__Theorem (Shannon).__ If a cipher satisfies perfect secrecy, then $|\mathcal{K}| \geq |\mathcal{M}|$.

__Intuition:__ Every ciphertext can decrypt to at most $|\mathcal{K}| < |\mathcal{M}|$ messages. This means that ciphertext leaks information about the message (not all messages equally likely). Cannot be perfectly secret.

__Proof.__ We will use a "counting" argument. Suppose $|\mathcal{K}| < |\mathcal{M}|$. Take any ciphertext $c \leftarrow \text{Encrypt}(k, m)$ for some $k \in \mathcal{K}$, $m \in \mathcal{M}$. This ciphertext can only decrypt to at most $|\mathcal{K}|$ possible messages (one for each choice of key). Since $|\mathcal{K}| < |\mathcal{M}|$, there is some message $m' \in \mathcal{M}$ such that

$$\forall k \in \mathcal{K} : \text{Decrypt}(k, c) \neq m'$$

By correctness of the cipher,

$$\forall k \in \mathcal{K} : \text{Encrypt}(k, m') \neq c$$

This means that

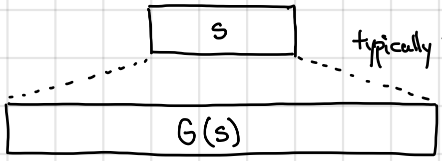
$$\left.\begin{array}{l} \Pr[k \xleftarrow{R} \mathcal{K} : \text{Encrypt}(k, m') = c] = 0 \\ \Pr[k \xleftarrow{R} \mathcal{K} : \text{Encrypt}(k, m) = c] > 0 \end{array}\right\} \text{Cannot be perfectly secret!}$$

__Take-away:__ Perfect secrecy requires long keys. Very impractical (except in the most critical scenarios — exchanging daily codebooks)

If we want something efficient / usable, we need to compromise somewhere.

- __Observe:__ Perfect secrecy is an __information-theoretic__ (i.e., a mathematical) property
  Even an __infinitely-powerful (computationally-unbounded)__ adversary cannot break security
  We will relax this property and only require
  security against __computationally-bounded__ (efficient) adversaries

Idea: "compress" the one-time pad: we will generate a long random-looking string from a short seed (e.g., $s \in \{0,1\}^{128}$).



typically: $s \in \{0,1\}^\lambda$ (λ is the seed length or security parameter)

$G(s) \in \{0,1\}^n$ where $n \gg \lambda$

n is the "stretch" of a PRG

Stream cipher: $\mathcal{K} = \{0,1\}^\lambda$

$\mathcal{M} = \mathcal{C} = \{0,1\}^n$

Encrypt(k, m): $c \leftarrow m \oplus \boxed{G(k)}$

Decrypt(k, c): $m \leftarrow c \oplus \boxed{G(k)}$

Instead of xor-ing with the key, we use the key to derive a "stream" of random-looking bits and use that in place of the one-time pad

If $\lambda < n$, then this scheme cannot be perfectly secure! So we need a different notion of security

Intuitively: Want a stream cipher to function "like" a one-time pad to any "reasonable" adversary.

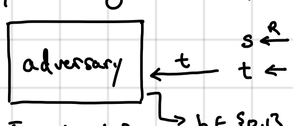
$\Rightarrow$ Equivalently: output of a PRG should "look" like uniformly-random string

What is a "reasonable" adversary?

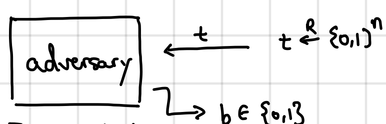
- Theoretical answer: algorithm runs in (probabilistic) polynomial time
- Practical answer: runs in time $< 2^{80}$ and space $< 2^{64}$ (can use larger numbers as well)

Goal: Construct a PRG so no efficient adversary can distinguish output from random.

Captured by defining two experiments or games:



Experiment 0



Experiment 1

the input to the adversary (t) is often called the challenge

Adversary's goal is to distinguish between Experiment 0 (pseudorandom string) and Experiment 1 (truly random string)

↳ It is given as input a string t of length n (either $t \leftarrow G(s)$ or $t \xleftarrow{R} \{0,1\}^n$)

↳ It outputs a guess (a single bit $b \in \{0,1\}$)

Remember: adversary knows the algorithm G; only seed is hidden!

Let $W_0 :=$ Pr[adversary outputs 1 in Experiment 0]

$W_1 :=$ Pr[adversary outputs 1 in Experiment 1]

} define the distinguishing advantage of $\mathcal{A}$ as

PRGAdv$[\mathcal{A}, G] := |W_0 - W_1|$

DO NOT RELY ON SECURITY BY OBSCURITY!

Definition. A PRG $G: \{0,1\}^\lambda \to \{0,1\}^n$ is secure if for all efficient adversaries $\mathcal{A}$,

probabilistic polynomial time

$$\text{PRGAdv}[\mathcal{A}, G] = \text{negl}(\lambda)$$

↳ negligible function (in the input length)

- Theoretical definition: $f(\lambda)$ is negligible if $f \in o(\lambda^{-c})$ for all $c \in \mathbb{N}$
- Practical definition: quantity $\leq 2^{-80}$ or $\leq 2^{-128}$

smaller than any inverse polynomial

e.g., $\frac{1}{2^\lambda}$, $\lambda^{-\log \lambda}$

Understanding the definition:

1. Can we ask for security against all adversaries (when $n \gg \lambda$)?

   No! Consider inefficient adversary that outputs 1 if t is the image of G and 0 otherwise.
   - $W_0 = 1$
   - $W_1 = \Pr[t \xleftarrow{R} \{0,1\}^n : \exists s \in \{0,1\}^\lambda : G(s) = t] = \frac{1}{2^{n-\lambda}}$

   $\Big\}$ $\text{PRGAdv}[A,G] = 1 - \frac{1}{2^{n-\lambda}} \approx 1$ if $n \gg \lambda$

2. Can the output of a PRG be biased (e.g., first bit of PRG output is 1 w.p. $\frac{2}{3}$)?

   No! Consider efficient adversary that outputs 1 if first bit of challenge is 1.
   - $W_0 = \frac{2}{3}$
   - $W_1 = \frac{1}{2}$

   $\Big\}$ $\text{PRGAdv}[A,G] = \frac{1}{6}$ NOT NEGLIGIBLE!

   More generally, no efficient statistical test can distinguish output of a secure PRG from random.

3. Can the output of a PRG be predictable (e.g., given first 10 bits, predict the 11th bit)?

   No! If the bits are predictable w.p. $\frac{1}{2} + \varepsilon$, can distinguish with advantage ε (since random string is unpredictable)

   In fact: unpredictable $\Rightarrow$ pseudorandom

Take-away: A secure PRG has the same statistical properties as the one-time pad to any efficient adversary.

$\Rightarrow$ Should be able to use it in place of one-time pad to obtain a secure encryption scheme (against efficient adversaries)

Need to define security of an encryption scheme.

Goal is to capture property that no efficient adversary can learn any information about the message given only the ciphertext. Suffices to argue that no efficient adversary can distinguish encryption of message m_0 from m_1, even if m_0, m_1 are adversarially-chosen.

Let (Encrypt, Decrypt) be a cipher. We define two experiments (parameterized by $b \in \{0,1\}$):

adversary		challenger ($b \in \{0,1\}$)
	$m_0, m_1 \in \mathcal{M}$ $\longrightarrow$	$k \xleftarrow{R} \mathcal{K}$
	$\longleftarrow$ $c_b \leftarrow \text{Encrypt}(k, m_b)$	
$\downarrow$ $b' \in \{0,1\}$		

$\Big\}$ semantic security experiment

Adversary chooses two messages and receives encryption of one of them. Needs to guess which one (i.e., distinguish encryption of m_0 from encryption of m_1)

Let $W_0 := \Pr[b' = 1 \mid b = 0]$
$W_1 := \Pr[b' = 1 \mid b = 1]$ $\Big\}$ probability that adversary guesses 1

(if adversary is good distinguisher, these two should be very different)

Define semantic security advantage of adversary A for cipher $\Pi_{SE} = (\text{Encrypt}, \text{Decrypt})$

$$\text{SSAdv}[A, \Pi_{SE}] = |W_0 - W_1|$$

Definition. A cipher $\Pi_{SE} = (\text{Encrypt}, \text{Decrypt})$ is semantically secure if for all efficient adversaries A,

$$\text{SSAdv}[A, \Pi_{SE}] = \text{negl}(\lambda)$$

λ is a security parameter (here, models the bit-length of the key)

Understanding the definition:

Can we learn the least significant bit of a message given only the ciphertext (assuming a semantically-secure cipher)

No! Suppose we could. Then, adversary can choose two messages m_0, m_1 that differ in their least significant bit and distinguish with probability 1.

This generalizes to <u>any</u> efficiently-computable property of the two messages.

How does semantic security relate to perfect secrecy?

<u>Theorem</u>. If a cipher satisfies perfect secrecy, then it is semantically secure.

<u>Proof</u>. Perfect secrecy means that $\forall\, m_0, m_1 \in \mathcal{M},\ c \in \mathcal{C}$:

$$\Pr[k \xleftarrow{R} \mathcal{K} : \mathrm{Encrypt}(k, m_0) = c] = \Pr[k \xleftarrow{R} \mathcal{K} : \mathrm{Encrypt}(k, m_1) = c]$$

Equivalently, the distributions

$$\underbrace{\{k \xleftarrow{R} \mathcal{K} : \mathrm{Encrypt}(k, m_0)\}}_{D_0} \quad \text{and} \quad \underbrace{\{k \xleftarrow{R} \mathcal{K} : \mathrm{Encrypt}(k, m_1)\}}_{D_1}$$

are <u>identical</u> ($D_0 \equiv D_1$). This means that the adversary's output b' is identically distributed in the two experiments, and so

$$\mathrm{SSAdv}[\mathcal{A}, \Pi_{SE}] = |W_0 - W_1| = 0.$$

<u>Corollary</u>. The one-time pad is semantically secure.

encryption key (PRG seed)

$$c \leftarrow G(s) \oplus m$$
$$m \leftarrow G(s) \oplus c$$

seems straightforward, but takes some care to prove

<u>Theorem</u>. Let G be a secure PRG. Then, the resulting stream cipher constructed from G is semantically secure.

<u>Proof</u>. Consider the semantic security experiments:

Experiment 0: Adversary chooses m_0, m_1 and receives $c_0 = G(s) \oplus m_0$

Experiment 1: Adversary chooses m_0, m_1 and receives $c_1 = G(s) \oplus m_1$

} Want to show that adversary's output in these two experiments are indistinguishable

Let $W_0 = \Pr[\mathcal{A} \text{ outputs } 1 \text{ in Experiment } 0]$

$W_1 = \Pr[\mathcal{A} \text{ outputs } 1 \text{ in Experiment } 1]$

<u>Idea</u>: If $G(s)$ is uniform random string (i.e., one-time pad), then $W_0 = W_1$. But $G(s)$ is like a one-time pad!

Define Experiment $0'$: Adversary chooses m_0, m_1 and receives $c_0 = t \oplus m_0$ where $t \xleftarrow{R} \{0,1\}^n$

Experiment $1'$: Adversary chooses m_0, m_1 and receives $c_1 = t \oplus m_1$ where $t \xleftarrow{R} \{0,1\}^n$

Define W_0', W_1' accordingly.

First, observe that $W_0' = W_1'$ (one-time pad is perfectly secure).

Now we show that $|W_0 - W_0'| = \mathrm{negl}$ and $|W_1 - W_1'| < \mathrm{negl}$.

$$\Rightarrow |W_0 - W_1| = |W_0 - W_0' + W_0' - W_1' + W_1' - W_1|$$
$$\leq |W_0 - W_0'| + |W_0' - W_1'| + |W_1' - W_1| \quad \text{by triangle inequality}$$
$$= \mathrm{negl.} + \mathrm{negl.} = \mathrm{negl.}$$

Show. If G is a secure PRG, then for all efficient A, $|W_0 - W_0'| = \text{negl}$.

Common proof technique: prove the contrapositive.

Contrapositive: If A can distinguish Experiments 0 and $0'$, then G is not a secure PRG.

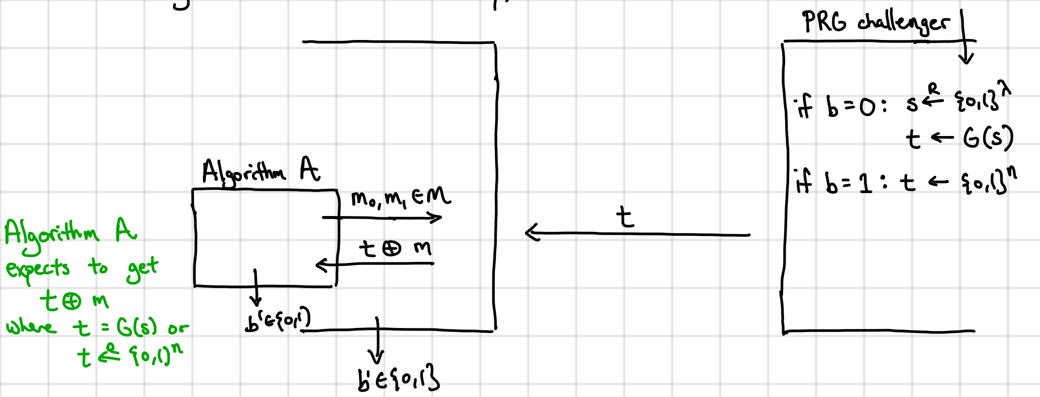
Suppose there exists efficient A that distinguishes Experiment 0 from $0'$

$\Rightarrow$ We use A to construct efficient adversary B that breaks security of G.

↳ this step is a reduction

[we show how adversary (i.e., algorithm) for distinguishing Exp. 0 and $0'$ $\Rightarrow$ adversary for PRG]

Algorithm B (PRG adversary):



Running time of B = running time of A = efficient

Compute $\text{PRGAdv}[B, G]$.

$\Pr[B \text{ outputs } 1 \text{ if } b = 0] = W_0$ ← if $b = 0$, then A gets $G(s) \oplus m$ which is precisely the behavior in Exp. 0

$\Pr[B \text{ outputs } 1 \text{ if } b = 1] = W_0'$ ← if $b = 1$, then A gets $t \oplus m$ which is precisely the behavior in Exp. $0'$

$\Rightarrow \text{PRGAdv}[B, G] = |W_0 - W_0'|$, which is non-negligible by assumption. This proves the contrapositive.

Important note: Security of above schemes shown assuming message space is $\{0,1\}^n$ (i.e., all messages are n-bits long)

In practice: We have variable-length messages. In this case, security guarantees indistinguishability from other messages of the same length, but length itself is leaked [inevitable if we want short ciphertexts]

↳ can be problematic — see traffic analysis attacks!

So far, we have shown that if we have a PRG, then we can encrypt messages efficiently (stream cipher)