Announcements/Reminders

HW2 Due Feb 27th
HW3 Due March 5th

PR1 Due March 10th
Previously on COS 433...
Length Extension for PRGs

Suppose I give you a PRG $G: \{0,1\}^\lambda \rightarrow \{0,1\}^{\lambda+1}$

On it’s own, not very useful: can only compress keys by 1 bit

But, we can use it to build PRGs with arbitrarily-long outputs!
Extending the Stretch of a PRG
Security Proof

Assume towards contradiction that breaks big PRG

Goal: build adversary that breaks $G$
Proof by Hybrids

$H_0: \{0,1\}^\lambda$
Security Proof

$H_1$: 

$\{0,1\}^\lambda$ 

state_1 \rightarrow state_2 \rightarrow state_3 \rightarrow \cdots
Security Proof

$H_2$: 

\[ \mathcal{D}_2 : \{0,1\} \]

\[ \{0,1\}^{\lambda} \]

\[ \text{state}_2 \]

\[ \text{state}_3 \]

\[ \ldots \]
Security Proof

$H_t:$

{0,1}  {0,1}  {0,1}  {0,1}  ...
Security Proof

$H_0$ corresponds to pseudorandom $x$
$H_t$ corresponds to truly random $x$

Let $q_i = \Pr[\hat{f}(x) = 1: x \leftarrow H_i]$.

By assumption, $|q_t - q_0| > \varepsilon$

Triangle ineq:

$$|q_t - q_0| \leq |q_1 - q_0| + |q_2 - q_1| + \ldots + |q_t - q_{t-1}|$$

$$\Rightarrow \exists i \text{ s.t. } |q_i - q_{i-1}| > \varepsilon/t$$
Security Proof

\[ y \]

\[ \text{state}_i \quad \text{state}_{i+1} \quad \text{state}_{i+2} \]

\[ \ldots \]
Today: Multiple Message Security
Left-or-Right Experiment

\[
\text{LoR-Exp}_b(\text{Robot}, \lambda)
\]

\[
(m_0, m_1 \in M) \rightarrow c
\]

\[
b \rightarrow \text{Challenger} \rightarrow k \leftarrow K \leftarrow \text{Enc}(k, m_b)
\]

(Same \( b \) for all queries)
LoR Security Definition

Definition: \((\text{Enc}, \text{Dec})\) has Left-or-Right indistinguishability if, for all \(\cdot\) running in polynomial time, \(\exists\) negligible \(\varepsilon\) such that:

\[
\left| \Pr[1 \leftarrow \text{LoR-Exp}_0(\cdot, \lambda)] - \Pr[1 \leftarrow \text{LoR-Exp}_1(\cdot, \lambda)] \right| \leq \varepsilon(\lambda)
\]
Alternate Notion: CPA Security

What if adversary can additionally learn encryptions of messages of her choice?

Examples:
• Midway Island, WWII:
  • US cryptographers discover Japan is planning attack on a location referred to as “AF”
  • Guess that “AF” meant Midway Island
  • To confirm suspicion, sent message in clear that Midway Island was low on supplies
  • Japan intercepted, and sent message referencing “AF”
Alternate Notion: CPA Security

What if adversary can additionally learn encryptions of messages of her choice?

Examples:

• Mines, WWII:
  • Allies would lay mines at specific locations
  • Wait for Germans to discover mine
  • Germans would broadcast warning message about the mines, encrypted with Enigma
  • Would also send an “all clear” message once cleared
CPA Experiment

**CPA Query**

- \( m \in M \)
- \( m_0, m_1 \in M \)
- \( m \in M \)

**Challenge Query**

- \( k \leftarrow K \)
- \( c \leftarrow \text{Enc}(k,m) \)
- \( c \leftarrow \text{Enc}(k,m_b) \)
- \( c \leftarrow \text{Enc}(k,m) \)

\( \text{CPA-Exp}_b( ) \)
CPA Security Definition

Definition: \((\text{Enc, Dec})\) is CPA Secure if, for all algorithms running in polynomial time, \(\exists\) negligible \(\varepsilon\) such that:

\[ \left| \Pr[1 \leftarrow \text{CPA-Exp}_0(\cdot, \lambda)] - \Pr[1 \leftarrow \text{CPA-Exp}_1(\cdot, \lambda)] \right| \leq \varepsilon(\lambda) \]
Generalized CPA Experiment

Queries in any order

GCPA-Exp\(_b\)(\(\bullet\), \(\lambda\))
GCPA Security Definition

Definition: \((\text{Enc}, \text{Dec})\) is Generalized CPA Secure if, for all running in polynomial time, \(\exists\) negligible \(\epsilon\) such that:

\[
\left| \Pr[1\leftarrow \text{GCPA-Exp}_0(\cdot, \lambda)] - \Pr[1\leftarrow \text{GCPA-Exp}_1(\cdot, \lambda)] \right| \leq \epsilon(\lambda)
\]
Equivalences

Theorem:

Left-or-Right indistinguishability ⇔
CPA-security ⇔
Generalized CPA-security
Proof

We will prove:

Generalized CPA-security
  $\Rightarrow$ CPA-security
  $\Rightarrow$ LoR indistinguishability
  $\Rightarrow$ Generalized CPA-security
Proof

Generalized CPA-security $\Rightarrow$ CPA-security

• Trivial: any adversary in the CPA experiment is also an adversary for the generalized CPA experiment that just doesn’t take advantage of the ability to make multiple challenge/LoR queries
Proof

Left-or-Right $\Rightarrow$ Generalized CPA
• Assume towards contradiction that we have an adversary $\Box$ for the generalized CPA experiment
• Construct an adversary $\Box$ that runs $\Box$ as a subroutine, and breaks the Left-or-Right indistinguishability
\[ Pr[1 \leftrightarrow \text{LoR-Exp}_b(c, \lambda)] = Pr[1 \leftrightarrow \text{GCPA-Exp}_b(c, \lambda)] \]
Pr[1←LoR-Exp_{b}(\text{robot}, \lambda)] = Pr[1←GCPA-Exp_{b}(\text{dog}, \lambda)]
Proof

Left-or-Right $\implies$ Generalized CPA

\[
\Pr[1\leftarrow \text{LoR-Exp}_0(\text{\textbullet}, \lambda)]
- \Pr[1\leftarrow \text{LoR-Exp}_1(\text{\textbullet}, \lambda)]
= \Pr[1\leftarrow \text{GCPA-Exp}_0(\text{\textbullet}, \lambda)]
- \Pr[1\leftarrow \text{GCPA-Exp}_1(\text{\textbullet}, \lambda)]
= \varepsilon
\]
Proof

(regular) CPA $\Rightarrow$ Left-or-Right

• Assume towards contradiction that we have an adversary for the **LoR Indistinguishability**

• Hybrids!
Hybrid $i$: 

If at most $i$ queries so far, 

$c \leftarrow \text{Enc}(k, m_0)$

If more than $i$ queries so far, 

$c \leftarrow \text{Enc}(k, m_1)$
Proof

(regular) CPA $\Rightarrow$ Left-or-Right

- Hybrid $0$ is identical to $\text{LoR-Exp}_1(\lambda)$

- Hybrid $q$ is identical to $\text{LoR-Exp}_0(\lambda)$

- We know that $\text{distinguishes Hybrid } q \text{ and Hybrid } 0$ with advantage $\epsilon$
  $\Rightarrow \exists i \text{ s.t. } \text{distinguishes Hybrid } i \text{ and Hybrid } i-1 \text{ with advantage } \epsilon/q$
$\Pr[1 \leftarrow \text{CPA-Exp}_b(\text{\#}, \lambda)] = \Pr[1 \leftarrow \text{\# in Hybrid i-b}]$
Proof

(regular) CPA $\Rightarrow$ Left-or-Right

\[
\Pr[1 \leftrightarrow \text{CPA-Exp}_0(\phantom{\text{image}}, \lambda)] - \Pr[1 \leftrightarrow \text{CPA-Exp}_1(\phantom{\text{image}}, \lambda)]
\]

\[
= \Pr[1 \leftrightarrow \text{in Hybrid i}] - \Pr[1 \leftrightarrow \text{in Hybrid i-1}] \geq \frac{\varepsilon}{q}
\]
Equivalences

Theorem:

Left-or-Right indistinguishability
⇔
CPA-security
⇔
Generalized CPA-security

Therefore, you can use whichever notion you like best.
Constructing CPA-secure Encryption

Starting point: stream ciphers = PRG + OTP for multiple messages

Problems:
- Stateful
- Need to synchronize with Bob
Constructing CPA-secure Encryption

Idea 1: Use random position to encrypt

\[ k \oplus \text{Randomly chosen position } i (i, ) \]
Analysis

As long as the two encryptions never pick the same location, we will have security

\[ \text{Pr[Collision]} = ? \]
Pr[Collision]

Consider event \( E_{j,k} = (i_j = i_k) \)

\[ \Rightarrow \Pr[E_{j,k}] = \frac{1}{n} \]

\( \Pr[\text{Collision}] = \Pr[E_{1,2} \text{ or } E_{1,3} \text{ or } \ldots \text{ or } E_{j,k} \text{ or } \ldots] \)

Union bound:

\[ \Pr[\text{Collision}] \leq \sum_{j,k} \Pr[E_{j,k}] = \sum_{j,k} \left(\frac{1}{n}\right) = \frac{q(q-1)}{2n} \]
Analysis

As long as the two encryptions never pick the same location, we will have security

$$\Pr[\text{Collision}] < \frac{q^2}{2n},$$
where

- $q =$ number of messages encrypted
- $n =$ number of blocks

If collision, then no security (“two-time pad”)

So we get LoR security, with $\varepsilon' = \varepsilon + \frac{q^2}{2n}$
What if…

The PRG has **exponential** stretch

\[ \text{Prob}[\text{collision}] \text{ is exponentially small} \]

However, computing PRG takes exponential time
What if...

The PRG has **exponential** stretch

AND, it was possible to compute any 1 block of output of the PRG
- In polynomial time
- Without computing the entire output

In other words, given a key, can efficiently compute the function \( F(k, x) = G(k)_x \)
Pseudorandom Functions

Functions that “look like” random functions

Syntax:
• Key space $K_\lambda$
• Domain $X_\lambda$
• Co-domain/range $Y_\lambda$
• Function $F: K_\lambda \times X_\lambda \rightarrow Y_\lambda$

Correctness: $F$ is a function (deterministic)
Pseudorandom Functions

Security:

\[ x \in X_\lambda \]

Challenger:

\[ b \]

\[ b' \]
Pseudorandom Functions

Security:

\[ x \in X_\lambda \]

Challenger

\[ b = 0 \]

\[ k \leftarrow K_\lambda \]

\[ y \leftarrow F(k, x) \]

\[ \text{PRF-Exp}_0(\cdot, \lambda) \]
Pseudorandom Functions

Security:

\[ x \in X_\lambda \]

\[ b = 1 \]

Challenger

\[ H \leftarrow \text{Funcs}(X_\lambda, Y_\lambda) \]

\[ y = H(x) \]

\[ \text{PRF-Exp}_1(\cdot, \lambda) \]
Definition: $F$ is a secure PRF if, for all $\mathcal{A}$ running in polynomial time, $\exists$ negligible $\varepsilon$ such that:

$$\left| \Pr[1 \leftarrow \text{PRF-Exp}_0(\mathcal{A}, \lambda)] - \Pr[1 \leftarrow \text{PRF-Exp}_1(\mathcal{A}, \lambda)] \right| \leq \varepsilon(\lambda)$$
Using PRFs to Build Encryption

\textbf{Enc}(k, m):
- Choose random \( r \leftarrow X_\lambda \)
- Compute \( y \leftarrow F(k, r) \)
- Compute \( c \leftarrow y \oplus m \)
- Output \((r, c)\)

\textbf{Dec}(k, (r, c)):
- Compute \( y' \leftarrow F(k, r) \)
- Compute and output \( m' \leftarrow c \oplus y' \)

\textbf{Correctness}:
- \( y' = y \) since \( F \) is deterministic
- \( m' = c \oplus y = y \oplus m \oplus y = m \)
Using PRFs to Build Encryption

\[
\text{Ciphertext} = (r \oplus m, c)
\]
Security

**Theorem**: If $F$ is a secure PRF with domain $X_\lambda$ and $|X_\lambda|$ is superpoly, then $(Enc, Dec)$ is LoR secure.
Proof

Assume toward contradiction that there exists a $\mathcal{B}$ breaking $(\text{Enc},\text{Dec})$

Hybrids...
Proof

Hybrid 0:

\[
m_0, m_1 \in M_\lambda
\]

Challenger

\[
b = 0
\]

\[
k \leftarrow K_\lambda
\]

\[
r \leftarrow X_\lambda
\]

\[
y \leftarrow F(k, r)
\]

\[
c \leftarrow y \oplus m_0
\]

\[
\text{LoR-Exp}_0(\text{LoR}, \lambda)
\]
Proof

Hybrid 1:

\[ m_0, m_1 \in M_\lambda \]

Challenger

\[ b = 0 \]

\[
\begin{align*}
H & \leftarrow \text{Funcs}(X_\lambda, Y_\lambda) \\
r & \leftarrow X_\lambda \\
y & \leftarrow H(r) \\
c & \leftarrow y \oplus m_0
\end{align*}
\]
Proof

Hybrid 2:

Challenger

\[
\begin{align*}
&m_0, m_1 \in M_\lambda \\
&b' \leftarrow \text{Funcs}(X_\lambda, Y_\lambda) \\
&b = 0 \\
&r \leftarrow X_\lambda \\
y \leftarrow H(r) \\
c \leftarrow y \oplus m_1
\end{align*}
\]
Proof

Hybrid 3:

LoR-Exp$_1$(∪, λ)
Proof

Assume toward contradiction that there exists a with advantage \( \varepsilon \) in breaking \((\text{Enc},\text{Dec})\)

\( \text{ distinguishes Hybrid 0 from Hybrid 3 with advantage } \varepsilon, \text{ so either } \)

- Dist. Hybrid 0 from Hybrid 1 with adv. \( \varepsilon - \frac{q^2}{4}|X| \)
- Dist. Hybrid 1 from Hybrid 2 with adv. \( \frac{q^2}{2}|X| \)
- Dist. Hybrid 2 from Hybrid 3 with adv. \( \varepsilon - \frac{q^2}{4}|X| \)
Proof

Suppose distinguishes Hybrid 0 from Hybrid 1

Construct
Proof

Suppose $\mathcal{A}$ distinguishes Hybrid 0 from Hybrid 1

Construct

- $\text{PRF-Exp}_0(\mathcal{A}, \lambda)$ corresponds to Hybrid 0
- $\text{PRF-Exp}_1(\mathcal{A}, \lambda)$ corresponds to Hybrid 1

Therefore, $\mathcal{A}$ has advantage $\varepsilon - q^2/4|X| \Rightarrow \text{contradiction}$
Proof

Suppose 🐜 distinguishes Hybrid 1 from Hybrid 2
Proof

Hybrid 1:

1. challenger
   \[ b = 0 \]
   \[
   H \leftarrow \text{Funcs}(X, Y)
   \]
   \[
   r \leftarrow X
   \]
   \[
   y \leftarrow H(r)
   \]
   \[
   c \leftarrow y \oplus m_0
   \]

2. challenger
   \[ \Rightarrow (r, c) \]

3. challenger
   \[ m_0, m_1 \in M \]

4. challenger
   \[ b' \]
Proof

Hybrid 2:

\[ m_0, m_1 \in M \]

Challenger

\[ b=0 \]

\[ H \leftarrow \text{Funcs}(X,Y) \]

\[ r \leftarrow X \]

\[ y \leftarrow H(r) \]

\[ c \leftarrow y \oplus m_1 \]
Proof

Suppose distinguishes Hybrid 1 from Hybrid 2

As long as the r’s for every query are distinct, the y’s for each query will look like truly random strings

In this case, encrypting $m_0$ vs $m_1$ will be perfectly indistinguishable
• By OTP security
Proof

Suppose distinguishes Hybrid 1 from Hybrid 2

Therefore, advantage is $\leq \Pr[\text{collision in the } r's] < \frac{q^2}{2|X|}$
Proof

Suppose distinguishes Hybrid 2 from Hybrid 3

Almost identical to the 0/1 case...
Using PRFs to Build Encryption

So far, scheme had fixed-length messages
• Namely, $M_\lambda = Y_\lambda$

Now suppose we want to handle arbitrary-length messages
Security for Arbitrary-Length Messages

\[ m_0, m_1 \text{ s.t. } |m_0| = |m_1| \]

Challenger

\[ k \leftarrow K_\lambda \]

\[ c \leftarrow Enc(k, m_b) \]

\[ IND-Exp_b(b', \lambda) \]
Theorem: Given any CPA-secure \((\text{Enc}, \text{Dec})\) for fixed-length messages (even single bit), it is possible to construct a CPA-secure \((\text{Enc}, \text{Dec})\) for arbitrary-length messages.
Construction

Let \((\text{Enc,Dec})\) be CPA-secure for single-bit messages

\textbf{Enc'}(k,m):

For \(i=1,\ldots, |m|\), run \(c_i \leftarrow \text{Enc}(k, m_i)\)

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

\textbf{Dec'}(k, (c_1, \ldots, c_l)):

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

Output \(m = m_1m_2\ldots,m_l\)
Theorem: If $(Enc, Dec)$ is LoR secure, then $(Enc', Dec')$ is LoR secure
Proof (sketch)

\[ m_0, m_1 \]

\[ (m_0)_1, (m_1)_1 \]
\[ (m_0)_2, (m_1)_2 \]
\[ (m_0)_3, (m_1)_3 \]

\[ \cdots \]
\[ c \leftarrow (c_1, \ldots) \]
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

$\textbf{Enc}(k, m)$:

- 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 $(r, c)$

$\textbf{Dec}(k, (r, c))$:

- 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

\[ \text{Solution 1: Add PRG/Stream Cipher} \]

\[ \begin{align*}
  r & \leftarrow X \\
  k & \leftarrow F \\
  y & \leftarrow z \\
  z & \oplus m \\
  (\text{,} & \ c) 
\end{align*} \]
Solution 2: Counter Mode

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

**Dec(k, (r,c) ):**
- For $i=1,\ldots,l$, 
  - Compute $y_i \leftarrow F(k, r||i)$
  - Compute $m_i \leftarrow y_i \oplus c_i$
- Output $m=m_1,\ldots,m_l$

Write $i$ as $\lambda/2$-bit string

Handles any message of length at most $2^{\lambda/2}$
Solution 2: Counter Mode

\[ F_k \oplus X \rightarrow r_1 \]
\[ F_k \oplus F_k \rightarrow r_2 \]
\[ F_k \oplus F_k \rightarrow r_3 \]
\[ F_k \oplus F_k \rightarrow r_4 \]
\[ F_k \oplus F_k \rightarrow r_5 \]
Summary

PRFs = “random looking” functions

Can be used to build security for arbitrary length/number of messages with stateless scheme

Next time: block ciphers and other “modes” of operation
Reminders

HW2 Due Feb 27\textsuperscript{th}
HW3 Due March 5\textsuperscript{th}

PR1 Due March 10\textsuperscript{th}