Announcements

Homework 3 up
Last Time

Stream Ciphers

Design of PRGs
Encryption Security Experiment

\[ \lambda \xrightarrow{\text{b}} \text{Challenger} \]

\[ k \leftarrow K_{\lambda} \]

\[ c \leftarrow \text{Enc}(k,m_b) \]

\[ b' \]

\[ \text{IND-Exp}_b(\text{Alice}, \lambda) \]
Definition: \((\text{Enc}, \text{Dec})\) has ciphertext indistinguishability if, for all probabilistic polynomial time (PPT) , there exists a negligible function \(\varepsilon\) such that

\[
\left| \Pr[1 \leftarrow \text{IND-Exp}_0(\cdot, \lambda)] - \Pr[1 \leftarrow \text{IND-Exp}_1(\cdot, \lambda)] \right| \leq \varepsilon(\lambda)
\]
This Time

Multiple message security

Stateless encryption

Pseudorandom Functions
Multiple Message Security
Left-or-Right Experiment

\[ \text{LoR-Exp}_b(\text{bot}, \lambda) \]
LoR Security Definition

Definition: \((\text{Enc}, \text{Dec})\) has **Left-or-Right indistinguishability** if, for all probabilistic polynomial time (PPT) algorithms, there exists a negligible function \(\varepsilon\) 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:
• Land 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

\[ \text{Challenger} \]

\[ \begin{align*}
  k & \leftarrow K_{\lambda} \\
  c & \leftarrow \text{Enc}(k,m) \\
  c & \leftarrow \text{Enc}(k,m_b) \\
  c & \leftarrow \text{Enc}(k,m)
\end{align*} \]

\[ \text{CPA-Exp}_b(Charlie, \lambda) \]
Generalized CPA Experiment

\[ \lambda \]

\[ m \in M_\lambda \]

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

\[ m \in M_\lambda \]

Challenger

\[ k \leftarrow K_\lambda \]

\[ c \leftarrow \text{Enc}(k,m) \]

\[ c \leftarrow \text{Enc}(k,m_b) \]

\[ c \leftarrow \text{Enc}(k,m) \]

Queries in any order

\[ \text{GCPA-Exp}_b(\text{Charlie}, \lambda) \]
Equivalences

Theorem:

Left-or-Right indistinguishability

⇔

CPA-security

⇔

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 Left-or-Right queries
Proof

Left-or-Right $\rightarrow$ Generalized CPA

• Assume towards contradiction that we have an adversary for the generalized CPA experiment
• Construct an adversary that runs as a subroutine, and breaks the Left-or-Right indistinguishability
\[
\Pr[1 \leftarrow \text{LoR-Exp}_b(m, \lambda)] = \Pr[1 \leftarrow \text{GCPA-Exp}_b(m, \lambda)]
\]
Proof

Left-or-Right $\rightarrow$ Generalized CPA

\[
\Pr[1\leftarrow \text{LoR-Exp}_0(\text{LoR}, \lambda)] - \Pr[1\leftarrow \text{LoR-Exp}_1(\text{LoR}, \lambda)]
\]

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

(regular) CPA $\rightarrow$ Left-or-Right
• Assume towards contradiction that we have an adversary for the LoR experiment
• Hybrids!
Hybrid $i$:

If at most $i$ queries so far,
\[ k \leftarrow K_\lambda \]
\[ c \leftarrow \text{Enc}(k,m_0) \]

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 \(\text{LoR-Exp}_1(\cdot, \lambda)\)
- Let \(t\) be maximum number of queries by \(\cdot\) \((t \leq \text{running time of } \cdot \leq \text{polynomial})\)
- Hybrid \(t\) is identical to \(\text{LoR-Exp}_0(\cdot, \lambda)\)

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

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

$$\Pr[1\leftrightarrow\text{CPA-Exp}_0(\mathcal{A}, \lambda)] - \Pr[1\leftrightarrow\text{CPA-Exp}_1(\mathcal{A}, \lambda)]$$

$$= \Pr[1\leftrightarrow\text{in Hybrid } i] - \Pr[1\leftrightarrow\text{in Hybrid } i-1] = \varepsilon/t$$
Equivalences

Theorem:

Left-or-Right indistinguishability

\( \uparrow \)

\( \downarrow \)

CPA-security

\( \uparrow \)

Generalized CPA-security

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

Starting point: A simple randomized encryption scheme from PRGs:

\[ G \oplus \text{Randomly chosen position} \]
Analysis

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

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

- \( q = \) number of messages encrypted
- \( n = \) number of blocks

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

For small \( q \), we get small, but non-negligible security
What if…

The PRG has **exponential** stretch

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

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 \( \{0,1\}^\lambda \)
• Domain \( X \) (usually \( \{0,1\}^m \), \( m \) may depend on \( \lambda \))
• Co-domain/range \( Y \) (usually \( \{0,1\}^n \), may depend on \( \lambda \))
• Function \( F: \{0,1\}^\lambda \times X \rightarrow Y \)
Pseudorandom Functions

Security:

\[ x \in X \]

Challenger

\[ \lambda \]

\[ b \]

\[ b' \]
Pseudorandom Functions

**Security:**

\[ x \in X \]

\[ \lambda \]

\[ b = 0 \]

**Challenger**

\[ k \leftarrow K_\lambda \]

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

**PRF-Exp_0( , , \lambda)**
Pseudorandom Functions

Security:

\[ x \in X \]

\[ y = H(x) \]

\[ PRF-Exp_1(\, b', \, \lambda) \]
PRF Security Definition

**Definition:** $F$ is a secure PRF if, for all probabilistic polynomial time (PPT), there exists a negligible function $\varepsilon$ such that

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

**Enc**(k, m):
- Choose random \( r \leftarrow X \)
- Compute \( y \leftarrow F(k,r) \)
- Compute \( c \leftarrow y \oplus m \)
- Output \((r,c)\)

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

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

Ciphertext = (F(k, m) \oplus r, c)
Theorem: If $F$ is a secure PRF and $X$ is exponentially large in $\lambda$ (e.g. $X=\{0,1\}^\lambda$), then $(Enc, Dec)$ is CPA-secure.
Proof

Assume toward contradiction that there exists a PPT \( \text{\textbullet} \) and non-negligible \( \varepsilon \) such that \( \text{\textbullet} \) has advantage \( \varepsilon \) in breaking \((\text{Enc},\text{Dec})\)

Hybrids...
Proof

Hybrid 0:

\[
\begin{align*}
\lambda & \quad b=0 \\
& \quad \text{Challenger} \\
& \quad k \leftarrow K_\lambda \\
& \quad r \leftarrow X \\
& \quad y \leftarrow F(k, r) \\
& \quad c \leftarrow y \oplus m_0
\end{align*}
\]

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

Hybrid 1:

\[ \lambda \]

\[ \begin{align*}
    & b = 0 \\
    & \text{Challenger} \\
    & H \leftarrow \text{Funcs}(X, Y) \\
    & r \leftarrow X \\
    & y \leftarrow H(r) \\
    & c \leftarrow y \oplus m_0
\end{align*} \]
Proof

Hybrid 2:

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

Challenger

\[ \lambda \]

\[ b = 0 \]

\[ r \leftarrow X \]

\[ y \leftarrow H(r) \]

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

Hybrid 3:

\[ \lambda \]

Challenger

\[ b = 0 \]

\[ k \leftarrow K_\lambda \]

\[ r \leftarrow x \]

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

\[ c \leftarrow y \oplus m_1 \]

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

\[ (r, c) \]

LoR-Exp_1(iard, \lambda)
Proof

Assume toward contradiction that there exists a PPT and non-negligible $\varepsilon$ such that has advantage $\varepsilon$ in breaking $(Enc, Dec)$

$\varepsilon$ distinguishes Hybrid 0 from Hybrid 3 with advantage $\varepsilon$

$\Rightarrow \exists i$ such that $\varepsilon$ distinguishes Hybrid $i-1$ from Hybrid $i$ with advantage $\varepsilon/3$
Proof

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

Construct

$b'$
Proof

Suppose $\text{Hybrid}_0$ distinguishes Hybrid 0 from Hybrid 1

Construct

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

Therefore, $\text{has advantage } \epsilon/3 \Rightarrow \text{contradiction}$
Proof

Suppose distinguishes Hybrid 1 from Hybrid 2
Proof

Hybrid 1:

$\lambda \leftarrow \text{Funcs}(X,Y)$

$b = 0$

Challenger

$r \leftarrow X$

$y \leftarrow H(r)$

$c \leftarrow y \oplus m_0$
Proof

Hybrid 2:

\[
\begin{align*}
\lambda &
\end{align*}
\]

\[
\begin{align*}
\text{Challenger} &
\end{align*}
\]

\[
\begin{align*}
H \leftarrow \text{Funcs}(X, Y)
\end{align*}
\]

\[
\begin{align*}
r &\leftarrow X \\
y &\leftarrow H(r) \\
c &\leftarrow y \oplus m_1
\end{align*}
\]
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 

\[ \Pr[\text{collision in the } r\text{'s}] \]

\[ \leq \Pr[r^{(1)}=r^{(2)} \text{ or } r^{(1)}=r^{(3)} \text{ or } ... \text{ or } r^{(1)}=r^{(d+1)} \]

\[ \text{ or } r^{(2)}=r^{(3)} \text{ or } ... \]

\[ \leq \Pr[r^{(1)}=r^{(2)}] + \Pr[r^{(1)}=r^{(3)}] + ... + \Pr[r^{(1)}=r^{(t)}] + ... \]

\[ + \Pr[r^{(2)}=r^{(3)}] + ... \]

\[ = \left( \frac{1}{|X|} \right)^t \binom{t}{2} \]

\[ \leq \frac{t^2}{2|X|} \]

Exponentially small  \Rightarrow  \text{contradiction}
Proof

Suppose distinguishes Hybrid 2 from Hybrid 3

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

**Enc(k, m):**
- Choose random $r \leftarrow X$
- Compute $y \leftarrow F(k, r)$
- Compute $c \leftarrow y \oplus m$
- Output $(r, c)$

**Dec(k, (r,c)) :**
- Compute $y' \leftarrow F(k, r)$
- Compute and output $m' \leftarrow c \oplus y'$

**Correctness:**
- $y' = y$ since $F$ is deterministic
- $m' = c \oplus y = y \oplus m \oplus y = m$
Using PRFs to Build Encryption

So far, scheme had fixed-length messages
• Namely, $M = Y$

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

$\textbf{IND-Exp}_b(X, \lambda)$
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 \((Enc, Dec)\) be CPA-secure for single-bit messages
• If messages are more than single bit, can always pad to message length

\(Enc'(k,m)\):
\[
\text{For } i=1,\ldots, |m|, \text{ run } c_i \leftarrow Enc(k, m_i) \\
\text{Output } (c_1, \ldots, c_{|m|})
\]

\(Dec'(k, (c_1, \ldots, c_l))\):
\[
\text{For } i=1,\ldots, l, \text{ run } m_i \leftarrow Dec(k, c_i) \\
\text{Output } m = m_1m_2\ldots,m_l
\]
Proof

Assume toward contradiction that there exists a PPT $\mathcal{A}$ and non-negligible $\varepsilon$ such that $\mathcal{A}$ has advantage $\varepsilon$ in breaking $(Enc', Dec')$

Construct $\mathcal{A}$ that has advantage $\varepsilon$ in breaking $(Enc, Dec)$
Proof (sketch)

\[ (m_0)_1, (m_1)_1 \]
\[ (m_0)_2, (m_1)_2 \]
\[ (m_0)_3, (m_1)_3 \]
\[ \ldots \]
\[ c \leftarrow (c_1, \ldots) \]
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

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

\textbf{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 \( c=(c_1,\ldots,c_{|m|})\)

\textbf{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 \( m=m_1,\ldots,m_l \)

Handles any message of length at most \( 2^{\lambda/2} \)
- Includes all polynomial-length messages

Write \( i \) as \( \lambda/2 \)-bit string
Solution 2: Counter Mode

\[ X \rightarrow r \rightarrow 1 \quad 2 \quad 3 \quad 4 \quad 5 \]

\[ F \oplus X \rightarrow r \rightarrow (, ) \]
Summary

PRFs = “random looking” functions

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

Pseudorandom Permutations/Block Ciphers
• PRFs that are permutations