Lecture 10: Symmetric Encryption

Instructor: Omkant Pandey

Spring 2017 (CSE 594)

The Setting (Secret Communication)

- Alice and Bob share a secret $s \in \{0,1\}^n$
- Alice wants to send a private message m to Bob
- Goals:
  - **Correctness:** Alice can compute an encoding c of m using s. Bob can decode m from c correctly using s
  - **Security:** No eavesdropper can distinguish between encodings of m and m'

Definition of Symmetric Encryption

- **Syntax:**
  - $\mathsf{Gen}(1^n) \to s$
  - $\mathsf{Enc}(s, m) \to c$
  - $\mathsf{Dec}(s, c) \to m'$ or $\perp$

  All algorithms are PPT in n (aka the security parameter).

- **Correctness:** $\forall m, s:\ \mathsf{Dec}(s, \mathsf{Enc}(s, m)) = m$, where $s \xleftarrow{\$} \mathsf{Gen}(1^n)$

- **Security:** ?
- **Indistinguishability security:** adversary cannot tell if m_0 or m_1 was encrypted.

Security

Definition (Indistinguishability Security)

A symmetric encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is secure if for all n.u. PPT adversaries $\mathcal{A}$, there exists a negligible function $\mu(\cdot)$ s.t.:

$$\Pr\left[\begin{array}{c} s \xleftarrow{\$} \mathsf{Gen}(1^n), \\ (m_0, m_1) \leftarrow \mathcal{A}(1^n), \\ b \xleftarrow{\$} \{0,1\} \end{array} : \mathcal{A}\left(\mathsf{Enc}\left(m_b\right)\right) = b\right] \leqslant \frac{1}{2} + \mu(n)$$

Definition (Indistinguishability Security (alternative))

A symmetric encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ secure if $\forall m_0, m_1$:

$$\left\{\mathsf{Enc}(s, m_0) : s \xleftarrow{\$} \mathsf{Gen}(1^n)\right\} \overset{c}{\approx} \left\{\mathsf{Enc}(s, m_1) : s \xleftarrow{\$} \mathsf{Gen}(1^n)\right\}$$

Note: Second definition is computational analogue of perfect secrecy.

Recall: these two are equivalent with a normalization factor 2 (“prediction advantage” vs “computational indistinguishability”).

One-Time Pads

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m) := m \oplus s$
- Security:

$$\mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_0\right) \equiv \mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_1\right)$$

1. <u>Think:</u> How to encrypt messages longer than n bits?

Encryption using PRGs

Note: m can be polynomially long if we use poly-stretch PRG

Encryption using PRGs

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$

<u>Note:</u> m can be polynomially long if we use poly-stretch PRG

Encryption using PRGs

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m) := m \oplus PRG(s)$

Note: m can be polynomially long if we use poly-stretch PRG

Encryption using PRGs

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s,m) := m \oplus PRG(s)$
- Security:

$$\mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_0\right) \overset{c}{\approx} \mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_1\right)$$

<u>Note:</u> m can be polynomially long if we use poly-stretch PRG

Encryption using PRGs

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m) := m \oplus PRG(s)$
- Security:

$$\mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_0\right) \overset{c}{\approx} \mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_1\right)$$

Note: m can be polynomially long if we use poly-stretch PRG

- Think: Proof?

Encryption using PRGs

- $\mathsf{Gen}(1^n) := s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m) := m \oplus PRG(s)$
- Security:

$$\mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_0\right) \overset{c}{\approx} \mathsf{Enc}\left(s \xleftarrow{\$} \{0,1\}^n, m_1\right)$$

<u>Note:</u> m can be polynomially long if we use poly-stretch PRG

- <u>Think:</u> Proof?
- <u>Think:</u> How to encrypt more than one message?

Stream Ciphers: Encryption with a PRG

- Roughly, another name for "encryption with a PRG"
- Recall our PRG stretch construction (from 1 bit to many)

$$\begin{array}{ccccccccccc} G(s_0 = s) & = & b_1 \| s_1 & \rightarrow G(s_1) & = & b_2 \| s_2 & \rightarrow & G(s_2) & = & b_3 \| s_3 & \rightarrow \\ m & = & m_1 & \| & & m_2 & \| & & & m_3 \ldots & \\ c & = & c_1 & \| & & c_2 & \| & & & c_3 \ldots & \end{array}$$

- Real world stream ciphers designed differently — much faster.
- Most of the old ones have known weaknesses or badly broken:
  - RC4: biases in initial output, was used for a long time.
  - CSS: badly broken, was used for DVD encryption
  - Modern stream ciphers (not yet broken): SOSEMANUK, Salsa20
  - Use a nonce/IV in addition to the seed.

Multi-message Secure Encryption

Definition (Multi-message Secure Encryption)

A symmetric encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is multi-message secure if for all n.u. PPT adversaries $\mathcal{A}$, for all polynomials $q(\cdot)$, there exists a negligible function $\mu(\cdot)$ s.t.:

$$\Pr\left[\begin{array}{r} s \xleftarrow{\$} \mathsf{Gen}(1^n), \\ \left\{\left(m_0^i, m_1^i\right)\right\}_{i=1}^{q(n)} \leftarrow \mathcal{A}(1^n), \\ b \xleftarrow{\$} \{0,1\} \end{array} : \mathcal{A}\left(\left\{\mathsf{Enc}\left(m_b^i\right)\right\}_{i=1}^{q(n)}\right) = b\right] \leqslant \frac{1}{2} + \mu(n)$$

1. <u>Think:</u> Computational Indistinguishability style definition
2. <u>Think</u> Security against *adaptive* adversaries?

Necessity of Randomized Encryption

Theorem (Randomized Encryption)

A multi-message secure encryption scheme cannot be deterministic and stateless.

Think: Proof?

Encryption using PRFs

Let $\{f_s : \{0,1\}^n \rightarrow \{0,1\}^n\}$ be a family of PRFs

Theorem (Encryption from PRF)

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a multi-message secure encryption scheme*

Encryption using PRFs

Let $\{f_s : \{0,1\}^n \to \{0,1\}^n\}$ be a family of PRFs

- $\mathsf{Gen}(1^n)$: $s \xleftarrow{\$} \{0,1\}^n$

Theorem (Encryption from PRF)

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a multi-message secure encryption scheme*

Encryption using PRFs

Let $\{f_s : \{0,1\}^n \to \{0,1\}^n\}$ be a family of PRFs

- $\mathsf{Gen}(1^n)$: $s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m)$: Pick $r \xleftarrow{\$} \{0,1\}^n$. Output $(r, m \oplus f_s(r))$

Theorem (Encryption from PRF)

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a multi-message secure encryption scheme*

Encryption using PRFs

Let $\{f_s : \{0,1\}^n \to \{0,1\}^n\}$ be a family of PRFs

- $\mathsf{Gen}(1^n)$: $s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s, m)$: Pick $r \xleftarrow{\$} \{0,1\}^n$. Output $(r, m \oplus f_s(r))$
- $\mathsf{Dec}(s, (r, c))$: Output $c \oplus f_s(r)$

Theorem (Encryption from PRF)

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a multi-message secure encryption scheme*

Encryption using PRFs

Let $\{f_s : \{0,1\}^n \to \{0,1\}^n\}$ be a family of PRFs

- $\mathsf{Gen}(1^n)$: $s \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{Enc}(s,m)$: Pick $r \xleftarrow{\$} \{0,1\}^n$. Output $(r, m \oplus f_s(r))$
- $\mathsf{Dec}(s,(r,c))$: Output $c \oplus f_s(r)$

Theorem (Encryption from PRF)

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a multi-message secure encryption scheme*

- <u>Think:</u> Proof?

Proof of Security

Proof via hybrids:

- H_1: Real experiment with $m_0^1, \ldots, m_0^{q(n)}$ (i.e., $b = 0$)
- H_2: Replace f_s with random function $f \xleftarrow{\$} \mathcal{F}_n$
- H_3: Switch to one-time pad encryption
- H_4: Switch to encryption of $m_1^1, \ldots, m_1^{q(n)}$
- H_5: Use random function $f \xleftarrow{\$} \mathcal{F}_n$ to encrypt
- H_6: Encrypt using f_s. Same as real experiment with $m_0^1, \ldots, m_0^{q(n)}$ (i.e., $b = 1$)

Think: Non-adaptive vs adaptive queries

Semantic Security

Definition (Semantic Security)

A symmetric encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is semantically secure if for every A there exists a PPT algorithm $\mathcal{S}$ (the "simulator") s.t. the following two experiments are computationally indistinguishable:

$$\left\{ \begin{array}{r} (m, z) \leftarrow A(1^n), \\ s \leftarrow \mathsf{Gen}(1^n), \\ \text{Output } (\mathsf{Enc}(s, m), {\color{red}z}) \end{array} \right\} \stackrel{c}{\approx} \left\{ \begin{array}{l} (m, z) \leftarrow A(1^n), \\ \text{Output } S(1^n, z) \end{array} \right\}$$

where A is an "adversarial" machine that samples a message from the message space and arbitrary auxiliary information.

- Indistinguishability security $\Leftrightarrow$ Semantic security
- Think: Proof?

Block Ciphers

- Encrypt blocks (say 64-bit) instead of bits as in steam ciphers
- AES is a block cipher
- Block cipher does not yield encryption direclty.
- The cipher comes with many "encryption modes" to encrypt arbitrarily long messages
- Poorest example: ECB or "Electronic Code Book"
  - identifiable patterns in the cipher (see wikipedia article)
- Other examples:
  - CBC or "Cipher Block Chaining"
  - PCBC, CFB, OFB, etc.