CS 600.442 – Modern Cryptography Oct 5, 2016

Lecture 6: Secret-Key Encryption

Instructor: Abhishek Jain *Scribe: Jesse Fowers*

1 Setting

We assume that Alice and Bob share a secret $s \in \{0,1\}^n$. We do not discuss here how they were able to share that secret (this will be discussed in a later lecture). Alice wants to send a private message to Bob, and we assume the "worst case" that the communication channel is public such that a third party Eve can read all messages sent on the channel. Alice will use an encryption scheme to communicate over the public channel with Bob.

2 Secret-key Encryption

Syntax A secret-key encryption consists of three algorithms described below:

- $\mathsf{KGen}(1^n) \to s$
- $\mathsf{Enc}(s, m) \to c$
- $\mathsf{Dec}(s, c) \to m'$

Each of these algorithms must run in polynomial time. In the above scenario, to send a message m to Bob, Alice uses the encryption algorithm to compute $\mathsf{Enc}(s, m) \to c$ and sends c on the public channel to Bob. Bob then uses the decryption algorithm $\mathsf{Dec}(s, c) \to m$ to recover the message.

A secret-key encryption scheme must satisfy the two properties described below:

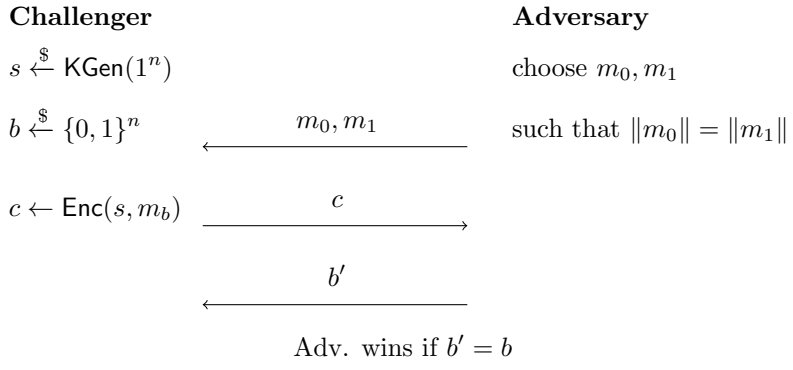
Correctness For every m, $\mathsf{Dec}(s, \mathsf{Enc}(s, m)) = m$, where $s \xleftarrow{\$} \mathsf{KGen}(1^n)$

Security Intuitively Eve must not be able to decipher the message m from the ciphertext c. More specifically, we require that she cannot distinguish between ciphertexts of two different messages m and m'. To formalize this we introduce the notion of IND-CPA Security (Indistinguishability under Chosen Plaintext Attack). Note here that n is known as the "security parameter" which expresses the degree of security of the scheme.

Definition 1 (IND-CPA) *A secret-key encryption scheme* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is IND-CPA secure if for all n.u. PPT adversaries* $\mathcal{A}$*, there exists a negligible function* $\mu(\cdot)$ *such that:*

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

Note that this is a game based definition of security, where the game operates in the following manner:



Note the lengths of the messages must be the same to prevent a simple attack that inspects the length of the ciphertext in order to differentiate.

Often it is easier to think of this definition as requiring computationally indistinguishability of the ciphertexts:

$$\mathsf{Enc}(m_0) \approx_c \mathsf{Enc}(m_1)$$

2.1 One-Time Pads

Consider the following encryption scheme:

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

Since s is random, $s \oplus m$ is also random so m is hidden from an informational theoretic perspective. That is to say $\mathsf{Enc}(s, m) \equiv U_n$, which is to say that they are identically distibuted. Thus two ciphertexts are more than just indistinguishable, they are in fact identically distributed.

$$\mathsf{Enc}(s \xleftarrow{\$} \{0,1\}^n, m_1) \equiv \mathsf{Enc}(s \xleftarrow{\$} \{0,1\}^n, m_2)$$

Note though that for two messages encrypted with the same key gives us $c_1 \oplus c_2 = (s \oplus m_1) \oplus (s \oplus m_1) = m_1 \oplus m_2$ which can break the security.

2.2 Encryption using PRGs

In the above scheme, the length of the secret-key grows with the length of the message being encrypted. We now discuss an encryption scheme where a secret-key of a fixed length can be used to encrypt a polynomially long message.

We will construct such an encryption using pseudorandom generators (PRG) by relying on the fact that the output of a PRG is computationally indistinguishable from uniform random. Thus we can use PRGs to convert a random key of a fixed length into a pseudorandom key of the necessary length to encrypt a message of arbitrary polynomially length.

Consider the following encryption scheme:

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

For security we do not have the identical distribution to uniform random as we did with one-time pads. We show security via indistinguishability.

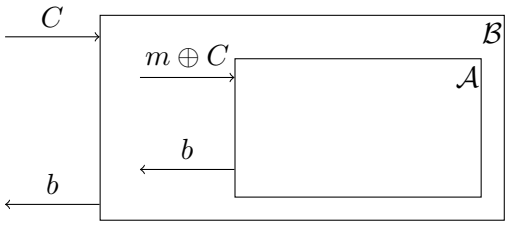
Proposition 1 (Security of Encryption using PRGs)

$$\mathsf{Enc}(s \xleftarrow{\$} \{0,1\}^n, m_1) \approx_c \mathsf{Enc}(s \xleftarrow{\$} \{0,1\}^n, m_2)$$

Proof. Security is proven via a hybrid argument. Rather than using the hybrid lemma though we prove in the forward direction by using the fact that indistinguishability is transitive over polynomial number of hybrids. Consider then the following list of hybrids:

$$\begin{aligned}
H_0: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n), \mathsf{Enc}(s, m_0) &&= m_0 \oplus PRG(s) \\
H_1: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n), \mathsf{Enc}(s, m_0) &&= m_0 \oplus R \xleftarrow{\$} \{0,1\}^n, \|R\| = \|m_0\| \\
H_2: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n), \mathsf{Enc}(s, m_1) &&= m_1 \oplus R \\
H_3: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n), \mathsf{Enc}(s, m_1) &&= m_1 \oplus PRG(s)
\end{aligned}$$

We claim that $H_0 \approx_c H_1$ and show this is true via a reduction argument. Assume that an adversary, $\mathcal{A}$, exists that can distinguish between H_0 and H_1. Then we construct an adversary $\mathcal{B}$ that can break the indistinguishability of the PRG.



The challenger to $\mathcal{B}$ flips a bit $b \xleftarrow{\$} \{0,1\}$ and then either sends $C \xleftarrow{\$} \{0,1\}^n$ or $C = PRG(s)$. Then $\mathcal{B}$ sends $C \oplus m$ to $\mathcal{A}$. Since $\mathcal{A}$ can distinguish between H_0 and H_0 it passes back the corresponding bit. This is then passed back to the challenger and is correct with non-negligible probability. The next pair of hybrids H_1 and H_2 are indistinguishable due to the indistinguishability of one-time pads. Finally H_2 and H_3 are indistinguishable by a symmetric argument to H_0 and H_1. Thus by transitivity, H_0 and H_3 are indistinguishable, which establishes that the scheme is IND-CPA secure.

■

3 Multi-message Secure Encryption

So far, we have only discussed encryption schemes where a key can be used to encrypt a *single* message. We now consider encryption schemes where a secret key can be used to encrypt multiple messages.

Definition 2 (Multi-message Secure Encryption) *A secret-key encryption scheme* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is multi-message IND-CPA secure if for all n.u. PPT adversaries* $\mathcal{A}$*, for all polynomials* $q(\cdot)$ *there exists a negligible function* $\mu(\cdot)$ *such that:*

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

This definition is very similar to the first but now in the security game the adversary sends two arrays of messages,$(m_0^1, m_0^2, \ldots, m_0^{q(n)})$ and $(m_1^1, m_1^2, \ldots, m_1^{q(n)})$ then the challenger encrypts one of them and returns an array of ciphertext $(c_1, c_2, \ldots, c_{q(n)})$. Here $q(\cdot)$ is an arbitrary polynomial chosen by the adversary $\mathcal{A}$.

Theorem 1 (Stateful Multi-message Encryption) *There exists a multi-message secret-key encryption scheme based on PRGs where the encryption algorithm is stateful*

The proof of the above theorem is straightforward and left as an exercise. Very roughly, the idea is that we can expand the key to a sufficiently long pseudorandom string using PRG (as in the previous construction) and then use different "chunks" of the randomness to encrypt different messages. We need to keep track of which chunk is used to encrypt which message, and therefore the encryption algorithm is stateful.

In practice, however, having a stateful encryption algorithm is not very desirable. Instead, we would like to construct multi-message encryption schemes where the encryption algorithm is stateless. The theorem below states that such an encryption scheme must also have a randomized encryption procedure.

Theorem 2 (Randomized Encryption) *A multi-message secure encryption scheme cannot be deterministic and stateless.*

Proof. Suppose such a scheme existed. Then an adversary could send

$$\begin{array}{cc} m_0^1 & m_1^1 \\ m_0^2 & m_1^2 \end{array}$$

such that $m_0^1 = m_0^2$ and $m_1^1 \neq m_1^2$, but since no state is kept and the algorithm is entirely deterministic $\mathsf{Enc}(m_0^1) = \mathsf{Enc}(m_0^2)$. So the adversary could just check if $c_1 = c_2$. ■

3.1 Encryption using PRFs

To construct a stateless multi-message secure encryption scheme, we will use a family of PRFs. Consider the following encryption scheme:

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

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

- $\mathsf{Dec}(s,(r,c)) := c \oplus f_s(r) = m$

Theorem 3 *Let* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *be based in PRFs as above, then it is a multi-message secure encryption scheme.*

Proof. The proof is done via a forward hybrid argument. Let RF be a purely random function. Consider the following list of hybrids:

$$\begin{aligned}
H_0: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_0^i) &= (r^i, m_0^i \oplus f_s(r^i)) \\
H_1: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_0^i) &= (r^i, m_0^i \oplus RF(r^i)) \\
H_2: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_0^i) &= (r^i, m_0^i \oplus R \xleftarrow{\$} \{0,1\}^n \\
H_3: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_1^i) &= (r^i, m_1^i \oplus R \xleftarrow{\$} \{0,1\}^n \\
H_4: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_1^i) &= (r^i, m_1^i \oplus RF(r^i)) \\
H_5: &\quad s \xleftarrow{\$} \mathsf{KGen}(1^n),\ \text{compute } \forall i \in [q(n)], \mathsf{Enc}(s, m_1^i) &= (r^i, m_1^i \oplus f_s(r^i))
\end{aligned}$$

Using a similar argument to the one used in the PRG case, $H_0 \approx_c H_1$ because a PRF is computationally indistinguishable from a RF. If these two hybrids were distinguishable then we could build an adversary that could distinguish between having oracle access to f_s and RF. Next, note that H_1 and H_2 are statistically indistinguishable: the only difference between them is that in H_2, we might sample the same string R for two different messages; however, this can only happen with exponentially small probability. Now, H_2 and H_3 are indistinguishable by the security of one-time pads. Now, by symmetry the remaining hybrids are indistinguishable giving us our result. ■