Pseudo-random Functions

Debdeep Mukhopadhyay
IIT Kharagpur

• We have seen the construction of PRG (pseudo-random generators) being constructed from any one-way functions.
• Now we shall consider a related concept:
  – Pseudo-random functions
  – instead of strings we consider functions
• It does not make much sense to call a fixed function pseudo-random.
• So, we have keyed functions.
• A keyed function $F : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$
• The first input is called the key.
• The key is chosen randomly and then fixed, resulting in a single argument function, $F_k : \{0,1\}^* \rightarrow \{0,1\}^*$
• Assume that the functions are length preserving, meaning that the inputs, output and key are all of the same size.

Pseudo-random functions

• No polynomial time adversary should be able to distinguish whether it is interacting with $F_k$ (for a randomly chosen $k$) or $f$ (where $f$ is chosen at random from the set of all functions mapping $n$ bit strings to $n$ bit strings).
• The former is chosen from a distribution over at most $2^n$ distinct functions.
• The later is from $2^{n2^n}$ functions.
• Despite this, the behavior of the functions must look the same to a PPT adversary.

Formally

Let $F : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ be an efficient length preserving, keyed function. $F$ is said to be pseudo-random function if for all probabilistic polynomial time distinguisher $D$, there exists negligible function $\varepsilon(n)$:

$$|\Pr[D^{k,()}(n)=1]-\Pr[D^{f,()}(n)=1]| \leq \varepsilon(n)$$

where $k$ is chosen uniformly at random and $f$ is chosen uniformly at random from the set of functions mapping $n$-bit strings to $n$-bit strings.
Encryption with a PRF

- Fresh Random string $r$
- Pseudorandom Function
- Pad
- $\text{plaintext} \xrightarrow{\text{xor}} \text{ciphertext}$

Some finer points

- If $x$ and $x'$ differ, outputs of $F_k(x)$ and $F_k(x')$ should not be correlated.
- Distinguisher $D$ is not given the key:
  - it is meaningless to talk about pseudorandomness once the key is given.
  - one can compute $y'=F_k(0^n)$
  - then query the oracle at $0^n$
  - if the oracle is for $F_k$, always $y=y'$
  - if the oracle is for random $f$, $y=y'$ with a probability of $2^{-n}$. thus we have a distinguisher.
Security against CPA

- **Defn:** A (adversary) should not be able to distinguish the encryptions of two arbitrary messages.

CPA Ind Exp

Experiment: Priv\(^{CPA}\)\(_{A,\Pi}(n)\)
1. A key is generated by running Gen(n)
2. Adversary A is given n and oracle access to Enc\(_i(.)\), and outputs a pair of messages \(m_0, m_1\) of the same length.
3. A random bit \(b \in \{0,1\}\) is chosen, and a ciphertext \(c = Enc_i(m_b)\) is computed and given to A as a challenge. We call c the challenge ciphertext.
4. Adversary A continues to have oracle access to Enc\(_i(.)\) and outputs a bit \(b'\).
5. Output of the experiment is 1, if \(b' = b\), and 0 otherwise.

A succeeds when Priv\(^{CPA}\)\(_{A,\Pi}(n) = 1\)
Definition of Indistinguishable under CPA

Any encryption scheme $\Pi=(\text{Gen},\text{Enc},\text{Dec})$ has indistinguishable encryptions under CPA (called CPA-secure) is for all PPT adversary $A$, there exists a negligible $\varepsilon(n)$ st.,

$$\Pr[\text{Priv}^{\text{CPA}}_{A,\Pi}(n) = 1] \leq \frac{1}{2} + \varepsilon(n)$$

where the probabilities are taken over the random coins used by $A$, as well as the random coins used in the experiment.

CPA secured encryption

• the scheme has to be probabilistic:
  – consider a deterministic encryption:
    $$\text{ENC}_k(m) = F_k(m)$$
  – Given $c=\text{ENC}_k(m_b)$ it is possible to ask for $\text{ENC}_k(m_0)$ and $\text{ENC}_k(m_1)$ and see for a match. Accordingly $b$ is discovered easily.
  – thus the scheme is not CPA secured.
A CPA secure encryption scheme from any PRF

Let $F$ be a PRF. Define an encryption as follows:

1. Gen: on input $n$ (security parameter), choose $k \leftarrow \{0,1\}^n$ uniformly at random as the key.
2. Enc: on input a key $k \in \{0,1\}^n$ and a message $m \in \{0,1\}^n$, choose $r \leftarrow \{0,1\}^n$ uniformly at random and output the ciphertext:
   \[ c = \langle r, F_k(r) \oplus m \rangle \]
3. Dec: On input a key $k$ and a ciphertext $\langle r, s \rangle$:
   \[ m = F_k(r) \oplus s \]

Theorem

If $F$ is a pseudorandom function, then the above construction is a fixed length symmetric key scheme for messages of length $n$ that has indistinguishable encryptions under a chosen plaintext attack.
Proof

• Follows a general principle.
• Prove that the system is secured when a truly random function is used.
• Next prove that if the system was insecure when the pseudorandom function was used, then we can make a distinguisher against the PRF.

Proof

Let $\Pi=(\widetilde{\text{Gen}}, \widetilde{\text{Enc}}, \widetilde{\text{Dec}})$ be an encryption scheme that is exactly the same as $\Pi=(\text{Gen},\text{Enc},\text{Dec})$, except that a true random function $f$ is used in place of $F_k$.
Thus $\text{Gen}(n)$ chooses a random function $f \leftarrow \text{Func}_n$ and $\widetilde{\text{Enc}}$ just like Enc except that $f$ is used instead of $F_k$. 
**Claim**: For every adversary $A$ that makes at most $q(n)$ queries to its encryption oracle:

$$\Pr[\text{Priv}_{\text{CPA}}(n) = 1] \leq \frac{1}{2} + \frac{q(n)}{2^n}$$

Proof: Each time a message $m$ is encrypted a random $r \leftarrow \{0,1\}^n$ is chosen and the ciphertext is $\{r, m \oplus f(r)\}$

Let $r_i$ be the random string used when generating the challenge ciphertext $c = \langle r_i, f(r_i) \oplus m \rangle$.

Define, Repeat as the event that $r_i$ is used by the encryption oracle to answer at least one of $A$'s queries.

Clearly, $\Pr[\text{Repeat}] \leq \frac{q(n)}{2^n}$

Also, $\Pr[\text{Priv}_{\text{CPA}}(n) = 1 | \text{Repeat}] = \frac{1}{2}$.

$$\therefore \Pr[\text{Priv}_{\text{CPA}}(n) = 1] = \Pr[\text{Priv}_{\text{CPA}}(n) = 1 \land \text{Repeat}] + \Pr[\text{Priv}_{\text{CPA}}(n) = 1 \land \neg \text{Repeat}]$$

$$\leq \Pr[\text{Repeat}] + \Pr[\text{Priv}_{\text{CPA}}(n) = 1 | \text{Repeat}] = \frac{1}{2} + \frac{q(n)}{2^n}$$

---

**Construct a Distinguisher for the PRF**

Let $\Pr[\text{Priv}_{\text{CPA}}(n) = 1] = \frac{1}{2} + \varepsilon(n)$

If $\varepsilon$ is not negligible then the difference between this is also non-negligible. Such a gap will enable us to distinguish the PRF from a true random function.
Distinguisher D:
D is given input n and oracle O: \{0,1\}^n \rightarrow \{0,1\}^n.
D answers the queries made by A in the CPA IND EXP.
1. Run A(n). Whenever A queries its encryption oracle on a message m, answer this query in the following way:
   a) Choose r \leftarrow \{0,1\}^n uniformly at random.
   b) Query O(r) and obtain response s'.
   c) Return to A the ciphertext <r,s' \oplus m>
2. When A outputs m, m \in \{0,1\}^n, choose a random bit b \leftarrow \{0,1\}.
   a) Choose r \leftarrow \{0,1\}^n uniformly at random.
   b) Query O(r) and obtain response s'.
   c) Return to A the ciphertext <r,s' \oplus m>
3. Continue answering A's queries as above. When A outputs a bit b', D outputs 1 if b = b' and 0 otherwise.

1. If D's oracle is a PRF, then the view of A when run as a sub-routine by D is distributed identically to the view of A in experiment Priv_{CPA}^{A,I} (n).
   Thus, Pr[D^k (n) = 1] = Pr[Priv_{CPA}^{A,I} (n) = 1].
2. If D's oracle is a random function, then the view of A when run as a sub-routine by D is distributed identically to the view of A in experiment Priv_{CPA}^{A,I} (n).
   Thus, Pr[D^f (n) = 1] = Pr[Priv_{CPA}^{A,I} (n) = 1].

Thus, \Pr[D^k (n) = 1] - \Pr[D^f (n) = 1] \geq \varepsilon(n) - \frac{q(n)}{2^n},
which is non-negligible if \varepsilon(n) is so.
This violates the PRF property of the F_k.
Modes of Encryption

• Electronic Code Book (ECB)

Deterministic encryption and thus cannot be CPA-secure.

Cipher Block Chaining (CBC)

A random IV (initial vector) of size n bits is chosen
Probabilistic and if F is a pseudo-random permutation then CBC is CPA-secure.
Output Feedback Mode (OFB)

If $F$ is a Pseudorandom function then this is secure against CPA.
Note that $F$ need not be a permutation.
Parallelism not possible.
But pre-processing of the key stream can lead to extremely fast operations.

Counter Mode
Theorem

If $F$ is a pseudo-random function, then randomized counter mode has indistinguishable encryptions under a chosen-plaintext attack (CPA).

Proof Idea

First consider that a truly random function, $f$, is used. Let $ctr^*$ denote the initial value $ctr$, when the challenge ciphertext is generated in the experiment $Priv^{opt}$. For the $i^{th}$ block of the message, thus $ctr^*+i$ was used to generate $f(ctr^*+i)$. Now, if $ctr^*+i$ was never accessed before, then the key stream is random and like a one time pad. Thus the adversary has no advantage in deciding whether $m_0$ or $m_1$ was the corresponding plaintext for the challenge ciphertext. So, we have to find what is the probability that $ctr^*+i$ was actually "matches" with one of the queries of the adversary $A$. 
Proof Idea

The adversary $A$ makes $q(n)$ queries. The starting IV value for the $i$th query is denoted by $ctr_i$. Let each message be of block-length $q(n)$.

We divide the entire scenario into two mutually exclusive cases:

1. There do not exist any $i, j, j'$ for which $ctr_i + j = ctr_i + j'$.
   
   Here: $Pr[Priv_{A,n}^{\mathrm{CPA}} = 1] = \frac{1}{2}$.

2. There exist $i, j, j'$ for which $ctr_i + j = ctr_i + j'$.
   
   In this case, $A$ can easily determine $f(ctr_i + j) = f(ctr_i + j')$ and thus compute $m_j$. Thus he can predict whether $m_i$ or $m_j$ was encrypted.

Let $Overlap_i$ denote the event that the sequence $ctr_i + 1, ..., ctr_i + q(n)$ overlaps the sequence $ctr_i + 1, ..., ctr_i + q(n)$.

Consider, $ctr_i + 1, ..., ctr_i + q(n)$

$Overlap_i$ occurs when $ctr_i + 1 \leq ctr_i + q(n)$ and

when $ctr_i + q(n) \geq ctr_i + 1$

This happens when: $ctr_i + 1 - q(n) \leq ctr_i \leq ctr_i + q(n) - 1$

Proof

We define the event $Overlap_i$, as when $Overlap_i$ occurs for any $i$, that is: $Pr[Overlap_i] \leq \sum_{i=1}^{q(n)} Pr[Overlap_i]$

Now, $Pr[Overlap_i] = \frac{2q(n) - 1}{2^n} \Rightarrow Pr[Overlap] \leq \frac{2q(n)^2}{2^n}$.

$Pr[Priv_{A,n}^{\mathrm{CPA}} = 1] \leq Pr[Overlap] + Pr[Priv_{A,n}^{\mathrm{CPA}} = 1 | Overlap]$

$= \frac{2q(n)^2}{2^n} + \frac{1}{2}$

The next step is to reason that if the random function is replaced by the pseudo-random function, and the scheme is not CPA-secure, then we can frame a PPT algorithm $D_i$ which is able to distinguish the function $F_k$ from a random function $f$. This proof is left as an exercise.
Block length and security

- Interestingly, we see that it is not only the key length but the block length also which decides the security.
- Consider a block length of 64 bits.
- The adversary’s success probability in the CPA sense is thus around \( \frac{1}{2} + \frac{q^2}{2^{63}} \). Thus if we have around \( 2^{30} \) guesses, then we have a practical attack! (only 1 GB queries and storage required).
- So, we need to increase the block length.