# Notes for Lecture 6 (Draft)

## Summary

The encryption scheme we saw last time, based on pseudorandom functions, works and is CPA-secure, but it is not used in practice. A disadvantage of the scheme is that the length of the encryption is twice the length of the message being sent.

Today we see the "counter mode" generalization of that scheme, which has considerably smaller overhead for long messages, and see that this generalizes preserves CPA-security.

We then give the definition of *pseudorandom permutation*, which is a rigorous formalization of the notion of *block cipher* from applied cryptography, and see two ways of using block ciphers to perform encryption. One is totally insecure, the other achieves CPA security.

## 1 The Randomized Counter Mode

Suppose we have a pseudorandom function $F : \{0,1\}^k \to \{0,1\}^m \to \{0,1\}^m$; we describe an encryption scheme that works for messages of variable length. We assume without loss of generality that the length of the message is a multiple of $m$, and we write a plaintext $M$ of length $cm$ as $M_1, \ldots, M_c$, a sequence of $c$ blocks of length $m$.

- $Enc(K, M_1, \ldots, M_k)$:
  - pick a random $r \in \{0,1\}^m$;
  - output $(r, F_K(r+1) \oplus M_1, \ldots, F_K(r+c) \oplus M_c$
- $Dec(K, C_0, C_1, \ldots, C_k) := C_1 \oplus F_K(C_0 + 1), \ldots, C_c \oplus F_K(C_0 + c)$

(When $r$ is a binary string in $\{0,1\}^m$ and $i$ is an integer, $r + i$ means the binary representation of the sum mod $2^m$ of $r$ (seen as an integer) and $i$.)

**Theorem 1** *Suppose $F$ is a $(t, \epsilon)$-secure pseudorandom function; then, when used to encrypt messages of length $cm$, the above scheme is $(t - O(cm), O(\epsilon + t^2 \cdot 2^{-m}))$-secure against CPA*

# 2 Pseudorandom Permutations

Denote by $\mathcal{P}_n$ the set of permutations $P : \{0,1\}^n \to \{0,1\}^n$,

**Definition 2 (Pseudorandom Permutation)** *A pair of functions $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ $I : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a $(t,q,\epsilon)$-secure pseudorandom permutation if:*

- *For every $r \in \{0,1\}^k$, the functions $x \to F_r(x)$ and $y \to I_r(x)$ are permutations and are one the inverse of the other;*
- *For every oracle algorithm $T$ that has complexity at most $t$ and that makes at most $q$ oracle queries we have*

$$|\mathop{\mathbb{P}}_{K\in\{0,1\}^k}[T^{F_K,I_K}() = 1] - \mathop{\mathbb{P}}_{P\in\mathcal{P}_n}[T^{P,P^{(-1)}}() = 1]| \leq \epsilon$$

That is, without knowing the random key $K$, the permutation $F_r(\cdot)$ and its inverse $I_r(\cdot)$ look like a completely random permutation and its inverse.

In the applied cryptography literature, pseudorandom permutations are called *block ciphers*.

How do we construct pseudorandom permutations? There are a number of block ciphers proposal, including the AES standard, that have been studied extensively and are considered safe for the time being. We shall prove later that any construction of pseudorandom functions can be turned into a construction of pseudorandom permutations; also, every construction of pseudorandom generators can be turned into a pseudorandom function, and every one-way function can be used to construct a pseudorandom generator. Ultimately, this will mean that it is possible to construct a block cipher whose security relies, for example, on the hardness of factoring random integers. Such a construction, however, would not be practical.

# 3 Encryption Using Pseudorandom Permutations

Here are two ways of using Pseudorandom Functions and Permutations to perform encryption. Both are used in practice.

## 3.1 ECB Mode

The Electronic Code-Book mode of encryption works as follows

- $Enc(K, M) := F_K(M)$
- $Dec(K, M) := I_K(M)$

**Exercise 1** *Show that ECB is message-indistinguishable for one-time encryption but not for two encryptions.*

## 3.2 CBC Mode

In its simplest instantiation the Cipher Block-Chaining mode works as follows:

- $Enc(K, M)$ : pick a random string $r \in \{0,1\}^n$, output $(r, F_K(r \oplus M))$
- $Dec(K, (C_0, C_1)) := C_0 \oplus I_K(C_1)$

Note that this similar to (but a bit different from) the scheme based on pseudorandom functions that we saw last time. In CBC, we take advantage of the fact that $F_K$ is now a permutation that is efficiently invertible given the secret key, and so we are allowed to put the $\oplus M$ inside the computation of $F_k$.

There is a generalization in which one can use the same random string to send several messages. (It requires synchronization and state information.)

- $Enc(K, M_1, \ldots, M_c)$ :
  - pick a random string $C_0 \in \{0,1\}^n$;
  - and output $(C_0, C_1, \ldots, C_c)$ where $C_i := F_K(C_{i-1} \oplus M_i)$
- $Dec(K, C_0, C_1, \ldots, C_c) := M_1, \ldots, M_c$ where $M_i := I_K(C_i) \oplus C_{i-1}$

This mode achieves CPA security.

# 4 The Ultimate Security

CPA security is not yet the strongest possible definition of security. It does not capture the possibility that an active adversary might obtain plaintext-ciphertext pairs in which the *ciphertext* depends arbitrarily on the challenge that the adversary is trying to decode. Suppose that the encryption scheme is part of a larger protocol in which sender and receiver are expected to exchange messages in a certain format. Then the attacker can send a ciphertext (of which it does not know the decoding),

and then see whether it receives a short reply (indicative that it is the encryption of "message invalid") or a longer one (which would be the continuation of the protocol).

In a *chosen ciphertext attack* (CCA), an attacker is allowed to gain decodings of ciphertexts of her choice. A system is secure if, for any two messages $M, M'$, their encodings are indistinguishable even to an adversary that can make arbitrary use of an encoding and a decoding oracle, with the only provision that it cannot query the decryption oracle on the challenge string.

**Definition 3 (CCA Security)** *An encryption scheme $(Enc, Dec)$ is $(t, \epsilon)$-CCA Secure if for every two messages $M, M'$ and for every oracle algorithm $T$ that satisfies the following two properties*

1. *has complexity $\leq t$;*
2. *when given input $C$ and oracles $E, D$, $T^{E,D}(C)$ never queries oracle $D$ with the string $C$;*

*We have*

$$\begin{aligned} &|\,\mathbb{P}[T^{Enc(K,\cdot),Dec(K,\cdot)}(Enc(K,M)) = 1] \\ &-\mathbb{P}[T^{Enc(K,\cdot),Dec(K,\cdot)}(Enc(K,M')) = 1]| \leq \epsilon \end{aligned}$$

It is not hard to show that all the encryption schemes we have seen so far fail CCA security.

Instead of trying to tweak the schemes in order to try and gain CCA security, we shall adopt a more modular design approach.

First, we will move on to discuss *message authentication*, which is an important goal in itself, for example when communicating with your online banking system. Then, we shall see that a CPA-secure encryption scheme, augmented with a proper message authentication scheme, automatically guarantees CCA security, because an attacker can *never make any use of a decryption oracle.*