Modern Cryptography

Construction of CPA-secure sechemes

Sep 10, 2025

Dr Shashank Singh

Table of Contents

Pseudorandom function

Keyed Function

We define a keyed function as a function

$$F_k : \{0,1\}^{\ell_{\text{in}}(n)} \mapsto \{0,1\}^{\ell_{\text{out}}(n)},$$

which takes as input a key, $k \leftarrow \{0,1\}^n$, completely specifies function

$$F_k : \{0,1\}^{\ell_{\text{in}}(n)} \mapsto \{0,1\}^{\ell_{\text{out}}(n)} \in \mathscr{F}_n.$$

Remark

- $\left|\left\{F_k : F_k \text{ is a keyed function and } k \in \{0,1\}^n\right\}\right| = 2^n \ll \left|\mathscr{F}_n\right|$.
- If $\ell_{\text{in}}(n) = \ell_{\text{out}}(n) = n$, F_k is called length preserving.

Pseudorandom Function

Definition 1

An efficient, length-preserving keyed function F_k, where $k \in \{0,1\}^n$ is said to pseudorandom function if for all probabilistic polynomial-time distinguishers (algorithms) $\mathscr{D}$, there is a negligible function $\varepsilon()$ such that,

$$\left|\Pr_{k \xleftarrow{\$} \{0,1\}^n}\left[\mathscr{D}\left(F_k(\cdot)\right) = 1\right] - \Pr_{f \xleftarrow{\$} \mathscr{F}}\left[\mathscr{D}\left(f(\cdot)\right) = 1\right]\right| \leq \varepsilon(n).$$

♣

Informally, if it's nearly impossible to determine whether a given function (oracle access) is a keyed function or a random function from the set $\mathscr{F}$ with a probability better than $\frac{1}{2} + \varepsilon(n)$, then we can consider the distribution of keyed functions to be pseudorandom.

Pseudorandom permutation

Pseudorandom permutation / Block Cipeher

- Let $\mathscr{P}_n$ be a set of all the permutations on $\{0,1\}^n$.
- A keyed function F_k is called a keyed permutations if $\ell_{\text{in}}(n) = \ell_{\text{out}}(n) = \ell(n)$ and $F_k : \{0,1\}^{\ell(n)} \mapsto \{0,1\}^{\ell(n)}$ is one-to-one.

> **Note**
>
> - A keyed permutation P_k is said to be efficient if there are PPT algorithms for conmputing $P_k(\cdot)$ and $P_k^{-1}(\cdot)$.
> - The value of $\ell(n)$ is termed as the block length of keyed permutation.

Pseudorandom permutation

Peudorandom Permutation

An efficient, length-preserving keyed permutation P_k, where $k \in \{0,1\}^n$ is said to be a pseudorandom permutation if for all probabilistic polynomial-time distinguishers (algorithms) $\mathscr{D}$, there is a negligible function $\varepsilon()$ such that,

$$\left|\Pr_{k \xleftarrow{\$} \{0,1\}^n}\left[\mathscr{D}\left(P_k(\cdot)\right) = 1\right] - \Pr_{p \xleftarrow{\$} \mathscr{P}_n}\left[\mathscr{D}\left(p(\cdot)\right) = 1\right]\right| \leq \varepsilon(n).$$

Strone Pseudorandom permutation

Strong Peudorandom Permutation

An efficient, length-preserving keyed permutation P_k, where $k \in \{0,1\}^n$ is said to be a strong pseudorandom permutation if for all probabilistic polynomial-time distinguishers (algorithms) $\mathscr{D}$, there is a negligible function $\varepsilon()$ such that,

$$\left|\Pr_{k \xleftarrow{\$} \{0,1\}^n}\left[\mathscr{D}\left(P_k(\cdot), P_k^{-1}(\cdot)\right) = 1\right] - \Pr_{p \xleftarrow{\$} \mathscr{P}_n}\left[\mathscr{D}\left(p(\cdot), p^{-1}(\cdot)\right) = 1\right]\right| \leq \varepsilon(n).$$

♣

Remark

It is fairly easy to construct a pseudorandom generator G from a pseudorandom function F. $G(s) := F_s(1) \parallel F_s(2) \parallel \ldots \parallel F_s(\ell)$ for desired ℓ.

Let F be a pseudorandom function. Define a private-key encryption scheme, $\Pi = (\text{GEN}, \text{ENC}, \text{DEC})$, for messages of length n as follows:

- The key $k \leftarrow \text{GEN}(n)$ is uniform on $\{0,1\}^n$.
- For $m \in \{0,1\}^n$, $\text{ENC}(k, m)$ picks $r \xleftarrow{\$} \{0,1\}^n$ and outputs c, where

$$c := \langle r, F_k(r) \oplus m \rangle$$

- On input $c = \langle r, s \rangle$ and a key k, $\text{DEC}(k, c)$ outputs m, where

$$m := F_k(r) \oplus s$$

Theorem 1

If F is a pseudorandom function, then the above scheme Π is a CPA-secure private-key encryption scheme for messages of length n.

♡

Note

- The scheme Π is efficient and CPA-secure. We can reuse the key as needed. But the ciphertext size is twice the message size. Is it possible to improve on this, too?

Remark

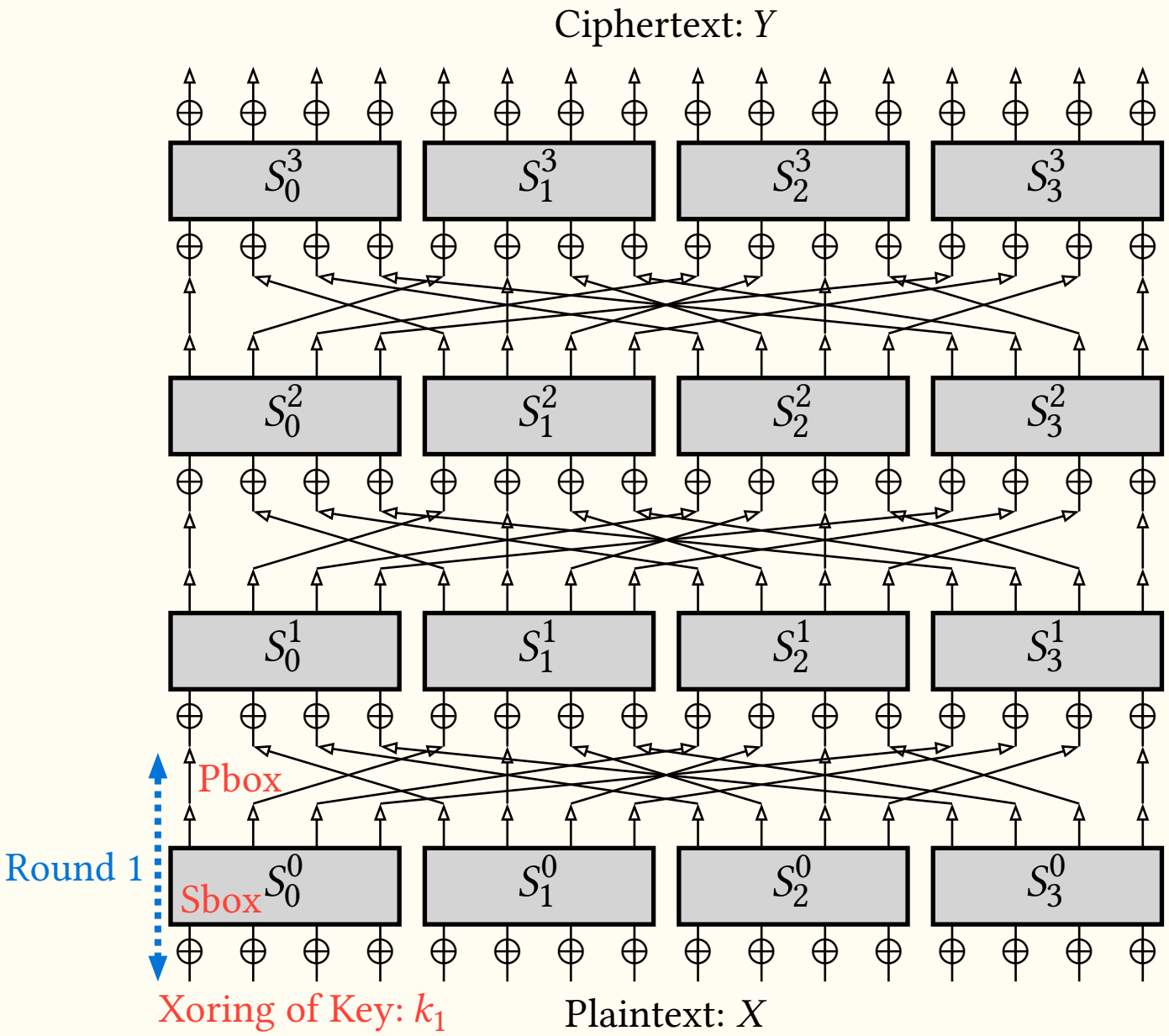
There are design principles, called the Modes of Operations, using which the ciphertext size can also be improved.

Block Ciphers

Block Ciphers

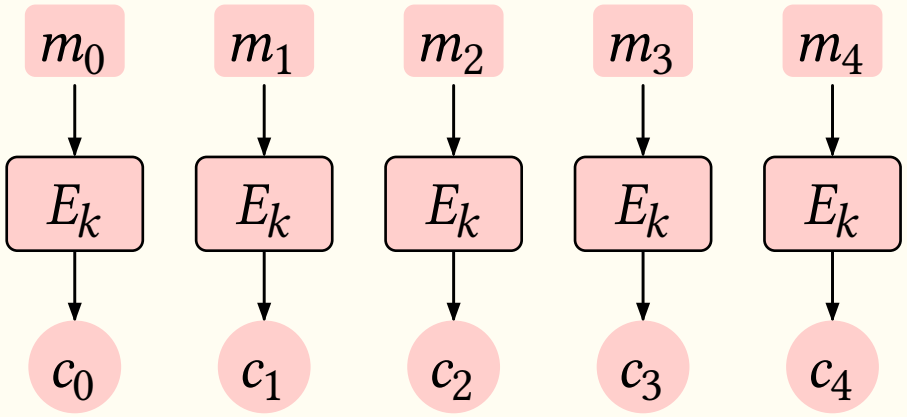
- The block ciphers are the concrete construction of secure (strong) pseudorandom permutations with some fixed key length called the block length.
- Block ciphers can be directly used as a fixed-length private key encryption scheme which is secure for single encryption only.
- Modes of operation (MOP) offer a secure and efficient way to encrypt long messages using block ciphers.

A toy example of SPN Block Cipher

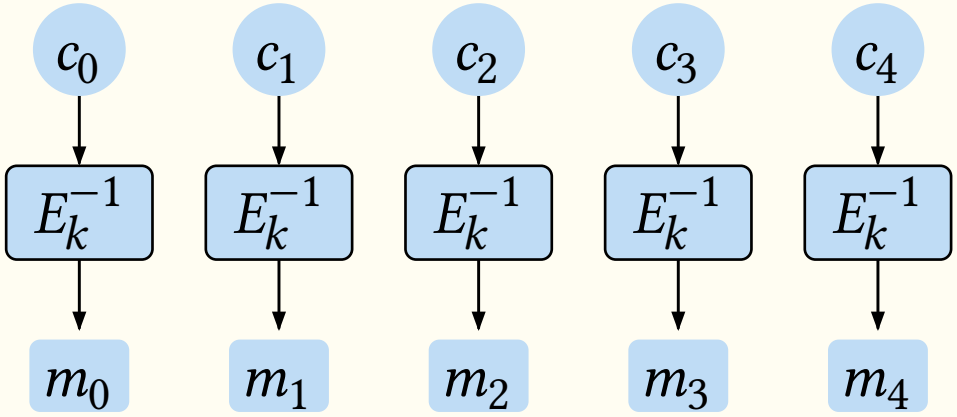


Electronic Codebook (ECB) MOP

- It is the simplest mode of operation, where each block of the message is encrypted using the same key. If needed, pad the last block.

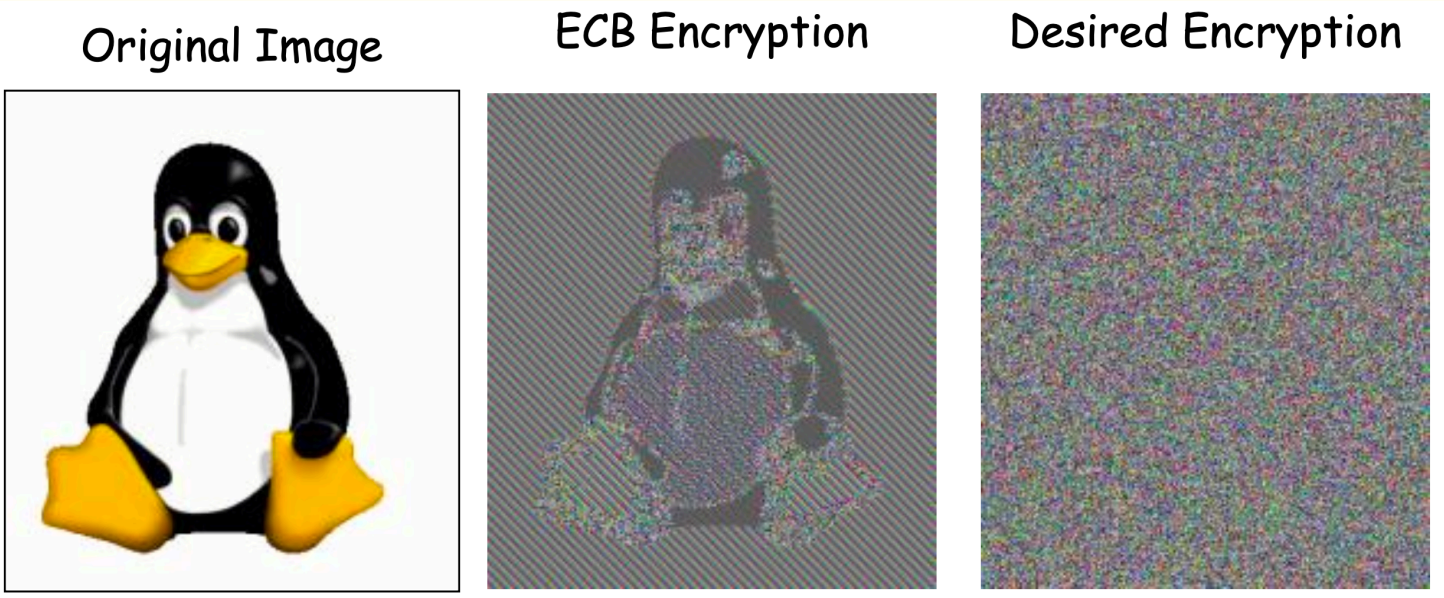


- Decryption is also straightforward.

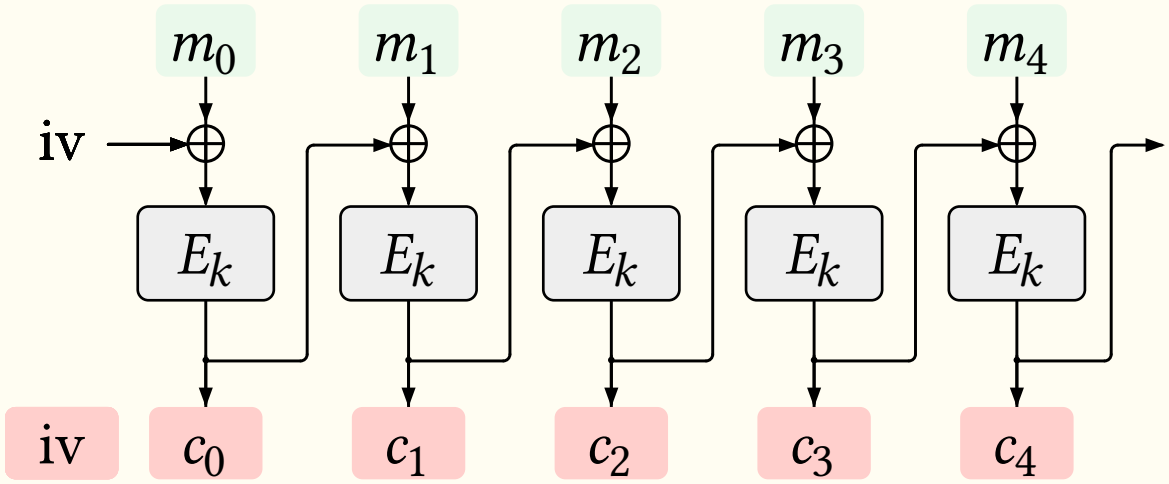


Electronic Codebook (ECB) MOP..

- Note that it is a deterministic encryption and hence it is not CPA secure.
- We should never use this scheme in any situation.

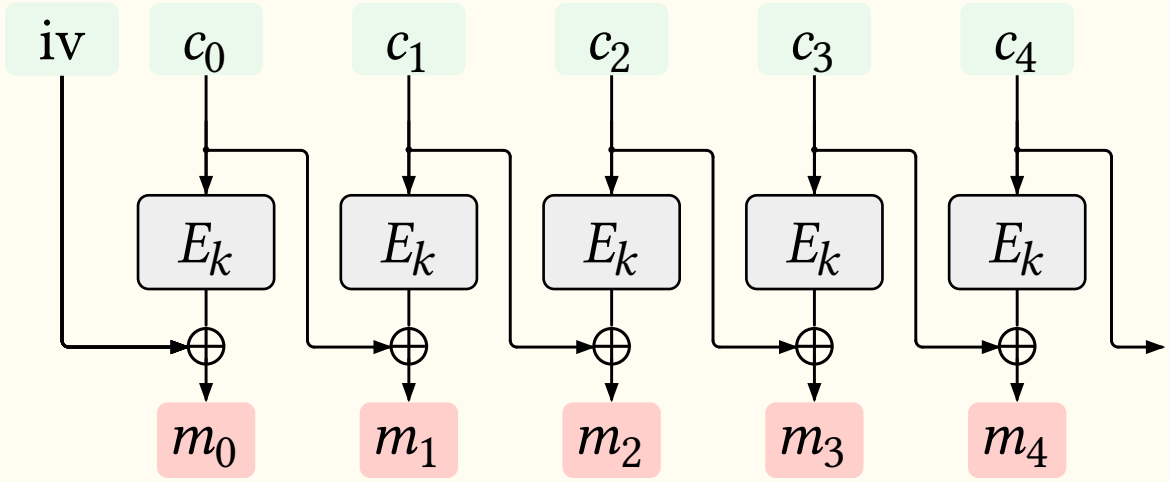


Cipher Block Chaining (CBC) MOP



Encryption:

- iv is the first cipher block of length n (the block size), randomly chosen.



Decryption:

Cipher Block Chaining (CBC) MOP..

- A random iv is required for each encryption.
- This scheme cannot be parallelised.
- Is it CPA secure? (Yes! The proof is left as an exercise.)
- **Stateful CBC**: In the stateful CBC mode of operation, for the first encryption, the iv is randomly chosen, and then onward, it is taken to be the last cipher block. (chained CBC mode used in SSL 3.0)

Theorem 1

Stateful CBC mode of operation is not CPA-secure.

♡

Proof. Proof by a counterexample. It will be done in class. □