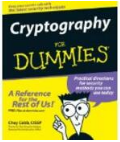
Block Cipher (分组密码)

Sheng Zhong   Yuan Zhang

Computer Science and Technology Department
Nanjing University

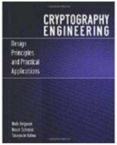
Outline

What is a block cipher?



“*An algorithm that encrypts data and cuts the data into small chunks and encrypts the chunks one after another.*”

From: Cryptography for Dummies (Chey Cobb)



“*An encryption function for fixed-size blocks of data.*”

From: Cryptography Engineering (N. Ferguson, et al.)

What is a block cipher

In this course, we adopt the definition in our textbook: A **block cipher** (分组密码，又称 “块密码”) is an efficient, keyed permutation function:

$$F : \{0,1\}^n \times \{0,1\}^l \to \{0,1\}^l.$$

- Essentially, it is just a keyed **permutation function**.
- “efficient”: Given k, both $F_k(x) \stackrel{def}{=} F(k,x)$ and its inverse F_k^{-1} can be computed within polynomial time.
- “permutation”: F_k is a bijection (i.e. a one-to-one correspondence).
- n is called key length, l is called *blocklength*.

Our security expectations for a block cipher

- Theoretically, we hope block ciphers to behave, at a minimum, as (strong) pseudorandom permutations.
- In practice, for a "good" block cipher, we often require the best known attack has time complexity $\approx 2^n$ (a brutal-force search for the key).

What are modes of operations?

- Block cipher (or stream cipher), is not used as encryption schemes on its own.
- Modes of operation (工作模式) provides a way to securely and efficiently encrypt **long messages** with stream or block ciphers.

" block/string ciphers + mode of operation "
= long-message encryption schemes

Block-cipher modes of operation

A few early, well-known modes of operation for Block ciphers include:

- Electronic Code Book (ECB) mode;
- Cipher Block Chaining (CBC) mode;
- Output Feedback (OFB) mode;
- Counter (CTR) mode.

ECB mode

Let F be a block cipher with block length n. Let the message to be encrypted be $m = m_1, m_2, \ldots, m_l$ where each $m_i \in \{0, 1\}^n$ represents a block of the plaintexts.

- The **Electronic Code Book** (ECB) mode is a naive mode:

$$c := < F_k(m_1), F_k(m_2), \ldots, F_k(m_l) >$$

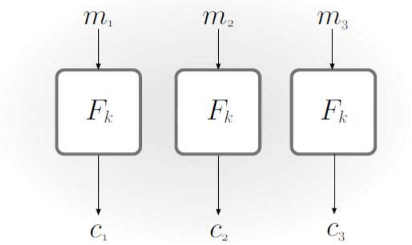


图 1: Electronic Code Book (ECB) mode

Security of ECB mode

- Deterministic, thus cannot be CPA-secure
- Not secure, should never be used.

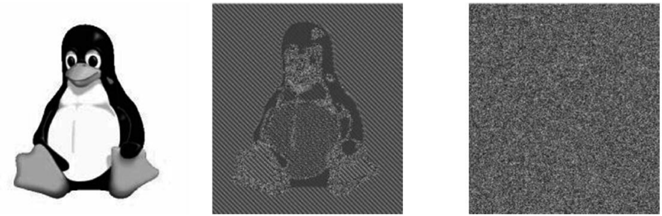


图 2: An illustration of the dangers of using ECB mode. The middle figure is an encryption of the image on the left using ECB mode; the figure on the right is an encryption of the same image using a secure mode. (Taking from http://en.wikipedia.org and derived from images created by Larry Ewing using The GIMP.

In **Cipher Block Chaining** (CBC) mode,

- Every time a message needs to be encrypted, *a uniform IV* is chosen.
- Plaintext blocks are "randomized" first, before being fed to F_k:

$$c_0 := IV$$
$$c_i := F_k(c_{i-1} \oplus m_i) \text{ for } i = 1, \ldots, l.$$

- Ciphertext is: $< c_0, c_1, \ldots, c_l >$.
- Decryption requires F_k^{-1} (F_k has to be invertible):

$$m_i := F_k^{-1}(c_i) \oplus c_{i-1}.$$

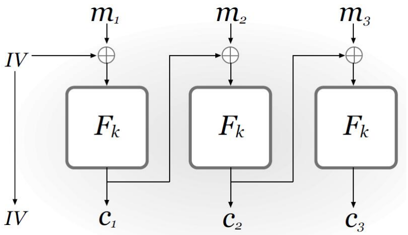


图 3: Cipher Block Chaining (CBC) mode

Security and drawback of CBC mode

- Encryption in CBC mode is probabilistic.
- If F is a PRF, then CBC-mode encryption is CPA secure.
- Major drawback: sequential encryption, cannot be parallelized.

OFB mode

In the **Output Feedback** (OFB) mode:

- A uniform IV is generated for every plaintext to be encrypted.
- "random" pads are generated for each block: $y_0 := IV$, $y_i = F_k(y_{i-1})$, and xor-ed to plaintext blocks: $c_i = m_i \oplus y_i$.
- The ciphertext is $< IV, c_1, c_2, \ldots, c_l >$.

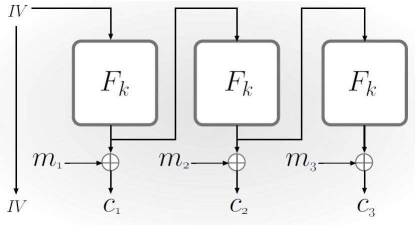


图 4: Output Feedback (OFB) mode

Pros and cons of OFB mode

- F_k is NOT required to be invertible.
- If F is a PRF, then OFB-mode encryption is CPA secure.
- Precomputation is supported: although both encryption and decryption are sequential, the pads for encryption/decryption can be pre-computed before the plaintext is known.

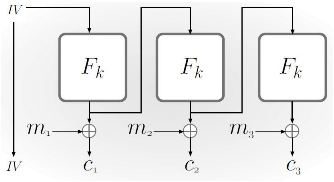


图 5: Output Feedback (OFB) mode

CTR mode

In the **Counter** (CTR) mode:

- A uniform value *ctr* is generated for every plaintext to be encrypted.
- "random" pads are generated for each block: $y_i = F_k(ctr + i)$, and xor-ed to plaintext blocks: $c_i = m_i \oplus y_i$.
- The ciphertext is $< ctr, c_1, c_2, \ldots, c_l >$.

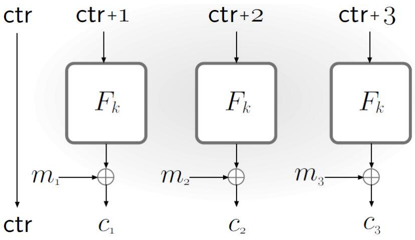


图 6: Counter (CTR) mode

Pros and cons of Counter (CTR) mode

- Very similar to OFB mode.
- If F is a PRF, then CTR-mode encryption is CPA secure.
- Decryption and encryption can be parallelized.

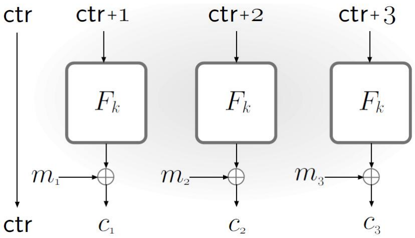


图 7: Counter (CTR) mode

The avalanche effect

Since we try to design a block cipher that is close to a random permutation, specifically we pay attentions to make it has an important property that a random permutation has:

- A small change in the input must "affect" every bit of the output.

We refer to this as the **avalanche effect**.

To construct a block cipher or a pseudo-random permutation, SPN follows the **confusion-diffusion paradigm**:

- It is introduced by Claude E. Shannon.
- It constructs a random-looking permutation F with a large block length from many smaller random or random-looking permutations $\{f_i\}$ with small block length.

图 8: Claude Elwood Shannon (1916–2001), photo downloaded from Shannon's wikimedia page

Details of the confusion-diffusion paradigm

The confusion-diffusion paradigm works as follows:

- The construction usually repeats multiple rounds of **confusion step** + **diffusion step**.
- The input of the block cipher is partitioned into several small blocks.
- In every round,
  - each small block is fed into a small random permutation (usually called a **round function**) to introduce *confusion* into the output.
  - Then, the bits of all blocks are mixed using a **mixing permutation** in the diffusion step.

Substitution-permutation networks

A **substitution-permutation network** (SPN) is a kind of practical block cipher construction that is based on a variant of the confusion-diffusion paradigm.

- In reach round, the SPN performs the following sequence of operations:
  1. **Key mixing**: in each round, the input is first xor-ed with the current-round **sub-key** or (**round key**)
  2. **Substitution**: after key mixing, each block i is inputted into a fixed, invertible "substitution function" (i.e. permutation) S_i called **S-box**.
  3. **Permutation**: the bits of all S-boxes' outputs are permuted.
- Details of the substitution step and the permutation step are public and know to any attacker. Only the keys are kept secret. (This setting is known as the **Kerckhoffs' principle**)

Substitution-permutation networks

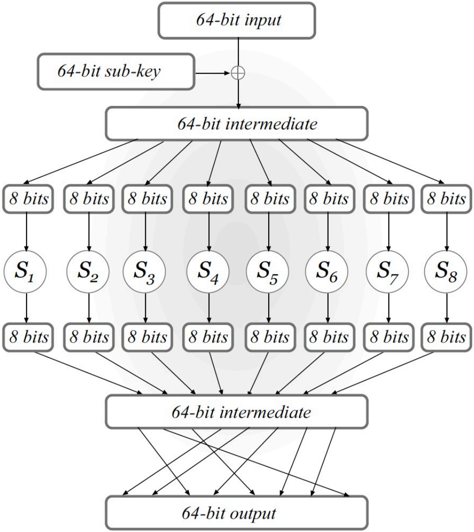


图 9: A single-round of a 64-bit substitution-permutation network

Substitution-permutation networks

- The output of each round is fed as input to the next round.
- After the last round, there is a final key-mixing step. The result is the output of the cipher.
- Different sub-keys are used in each round. Sub-keys are generated by a **master key** of the block cipher according to a **key schedule**.
- In summary, a r-round SPN has r (full) rounds of key mixing, S-box substitution, and application of a mixing permutation, followed by a final key-mixing step (Notice that in this SPN, r+1 sub-keys are used in total.).

Substitution-permutation networks

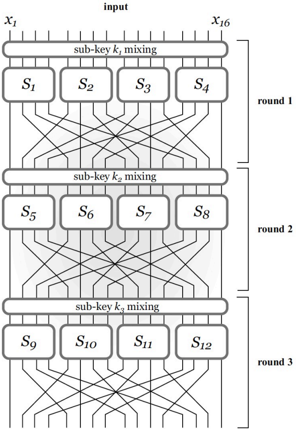


图 10: A 64-bit substitution-permutation network

The avalanche effect in the SPN

The avalanche effect is induced into the SPN mainly by the following designs:

- The S-boxes are deigned so that changing a single bit of the input to an S-box changes at least two bits in its output.
- The mixing permutations are designed so that the output bits of any S-box are used as input to multiple S-boxed in the next round.

Feistel network

Feistel network is another approach for constructing block ciphers:

- Named after the German-born physicist and cryptographer Horst Feistel (1915-1990) who did pioneering research while working for IBM.
- A Feistel network provides a way to construct an invertible function from non-invertible components. (Different from SPN, the underlying function need NOT be invertible).
- A Feistel network consists of several rounds. In each round, a **keyed round function** is applied.

In a balanced l-bit Feistel network, the i-th round function $\hat{f}_i$ takes as input a sub-key k_i and a $l/2$-bit string, and outputs an $l/2$-bit string. Define $f_i : \{0,1\}^{l/2} \to \{0,1\}^{l/2}$ via $f_i(R) \stackrel{def}{=} \hat{f}_i(k_i, R)$.

- The output (L_i, R_i) is computed as:
  $L_i := R_{i-1}$,
  $R_i := L_{i-1} \oplus f_i(R_{i-1})$.
- To invert, $R_i - 1 := L_i$,
  $L_{i-1} := R_i \oplus f_i(R_{i-1})$.
- Notice that the round function $\hat{f}_i$ are fixed, and publicly known, but the f_i are NOT.

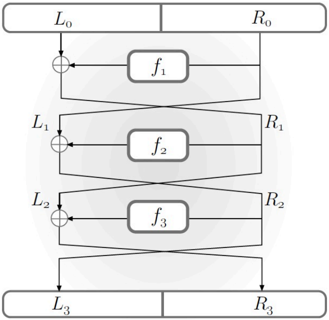


图 11: A three-round Feistel network

DES

The DES is a widely-used block cipher constructed based on the Feistel network:

- It consists of 16 rounds with a block length of 64 bits and a key length of 56 bits.
- The round function (sometimes called the mangler function) takes a 48-bit sub-keys and a 32-bit input, and output 32 bits.
- Well designed: the best known practical attack is still an exhaustive search through its key space.
- Cons: the key is too short.
- Replaced by AES.

The DES round function

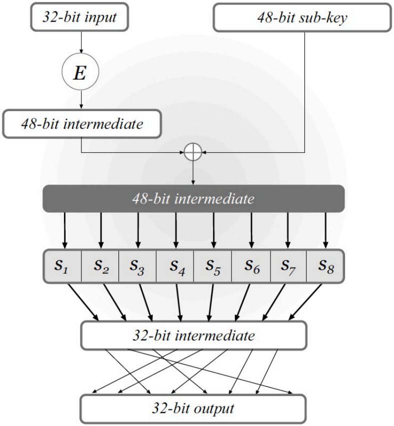


图 12: The DES mangler function

Triple-DES (3DES)

To improve the key-length issue of DES, Triple-DES is designed.

- Standardized in 1999.
- Achieves a 112-bit security.
- Since the minimum recommended key length nowadays is 128, 3DES is recomended to be replaced by the **Advanced Encryption Standard** (AES) (supports 128-bit, 192-bit, 256-bit keys).
- US National Institute of Standards and Technology (NIST) has deprecated DES and 3DES for all applications by the end of 2023.

Triple-DES (3DES)

To improve the key-length issue of DES, Triple-DES is designed:

- **Variant 1 (three keys)**: Choose three independent keys k_1, k_2, k_3, and define

$$F''_{k_1,k_2,k_3}(x) \overset{def}{=} F_{k_3}(F^{-1}_{k_2}(F_{k_1}(x))).$$

- **Variant 2 (two keys)**: Choose two independent keys k_1, k_2, and define

$$F''_{k_1,k_2}(x) \overset{def}{=} F_{k_1}(F^{-1}_{k_2}(F_{k_1}(x))).$$

AES - The Advanced Encryption Standard

- AES is a widely used encryption standard established by NIST in 2001.
- 128-bit block length.
- supports 128-bit (10 rounds), 192-bit (12 rounds), 256-bit(14 rounds) keys.
- adopts a substitution-permutation network structure.
- no practical cryptananlytic attacks better than brute-force key search.
- NSA allows to use AES256 to encrypt data with a classification level up to "TOP SECRET".[1]

[1] The United States has three levels of classification: Confidential, Secret, and Top Secret. From wikipedia page of "Classified information in the United States"

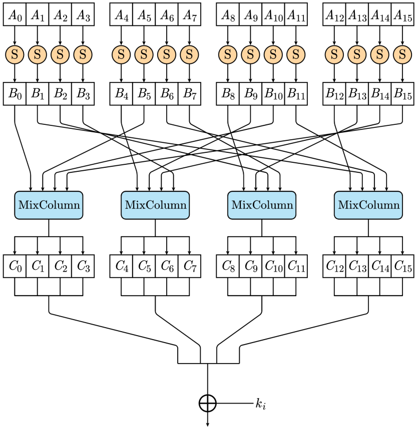


图 13: The AES round function[2]: In each round, the S Boxes performs "substitutions" on every byte-block A_i, then the results B_i undergo "permutations" via row-shifting and MixColumn operations, and the results C_i are xor-ed with the bytes of roundkey k_i.

A flashback: Cryptography is around us

- To the moment you made an online purchase.

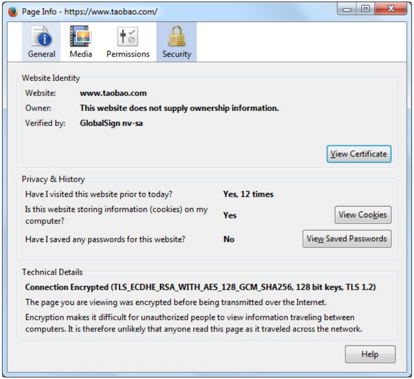


图 14: Page Info of www.taobao.com

- *"Website Identity Verified by GlobalSign nv-sa; Connection Encrypted (TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, 128 bit keys, TLS 1.2)"*