CHƯƠNG II
MÃ HÓA ĐỐI XỨNG

ThS. Nguyễn Cao Đạt
E-mail: dat@hcmut.edu.vn

Tham khảo

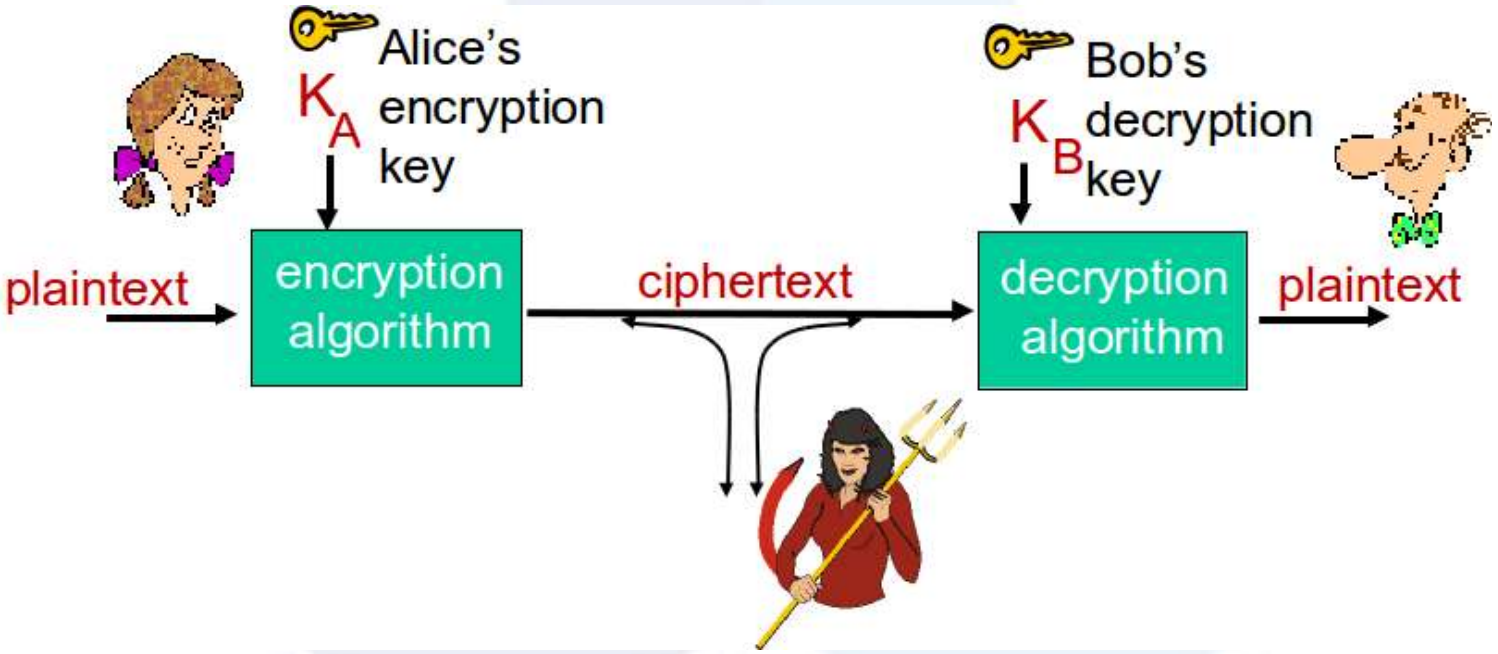
[1]. Cryptography and Network Security: chương 3, 4, 7 + AES

Nội dung trình bày

- Giới thiệu
  - Mô hình tổng quát
  - Các thuật ngữ cơ bản
  - Phân tích mã
  - Các phương pháp phân tích mã
- Mã hóa đối xứng
- Các hệ mã đối xứng truyền thống
- Chuẩn mã hóa dữ liệu
- Các vấn đề khác của mã hóa đối xứng

Mã hóa tổng quát

- m là thông điệp bản rõ
- $C = K_A(m)$ là thông điệp đã được mã hóa với khóa K_A
- $m = K_B(C)$ là thông điệp đã được giải mã với khóa K_B



Các thuật ngữ cơ bản

- Bản rõ(plaintext): thông điệp gốc.
- Bản mã(ciphertext): Thông điệp đã được mã hóa.
- Thuật toán mã hóa(cipher): thuật toán để chuyển đổi bản rõ thành bản mã.
- Khóa(key): thông tin được dùng trong mã hóa chỉ được biết bởi bên gởi và bên nhận.
- Mã hóa(encipher - encrypt): chuyển đổi bản rõ thành bản mã
- Giải mã(decipher - decrypt): phục hồi bản rõ từ bản mã.
- Mật mã(cryptography): nghiên cứu các nguyên tắc/phương pháp mã hóa.
- Phân tích mã(cryptanalysis - codebreaking): nghiên cứu các nguyên tắc/phương pháp giải mã từ bản mã nhưng không biết thông tin về khóa.
- Mật mã học(cryptology): ngành học về mật mã và phân tích mã.

Phân tích mã

- Cryptanalysis từ các từ của Hy Lạp là kryptós (che dấu) và analýein(nới lỏng, cởi trói, phân tích).
- Xác định ý nghĩa của thông điệp được mã hóa mà không có khóa giải mã.


VICTORY EXTRA !

NAZIS QUIT !

CITY EDITION EXTRA The Seattle Daily Times

TERMS SIGNED 5:41 SUNDAY, SEATTLE TIME

U. S. PLANES BAG 52 MORE JAP VESSELS

Tanks, Russ Bearing Down On Prague As Nazis Resist

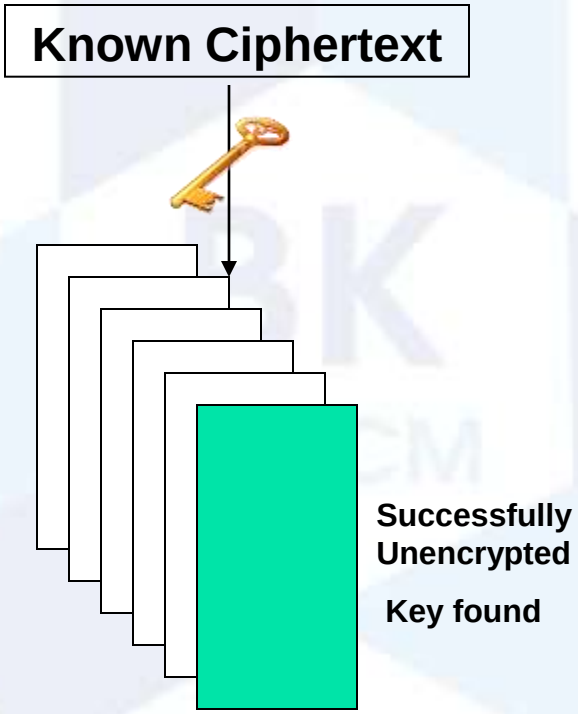
GREATEST OF ALL WARS IN EUROPE ENDS

BULLETINS

SURRENDER TOLD PEOPLE OF GERMANY

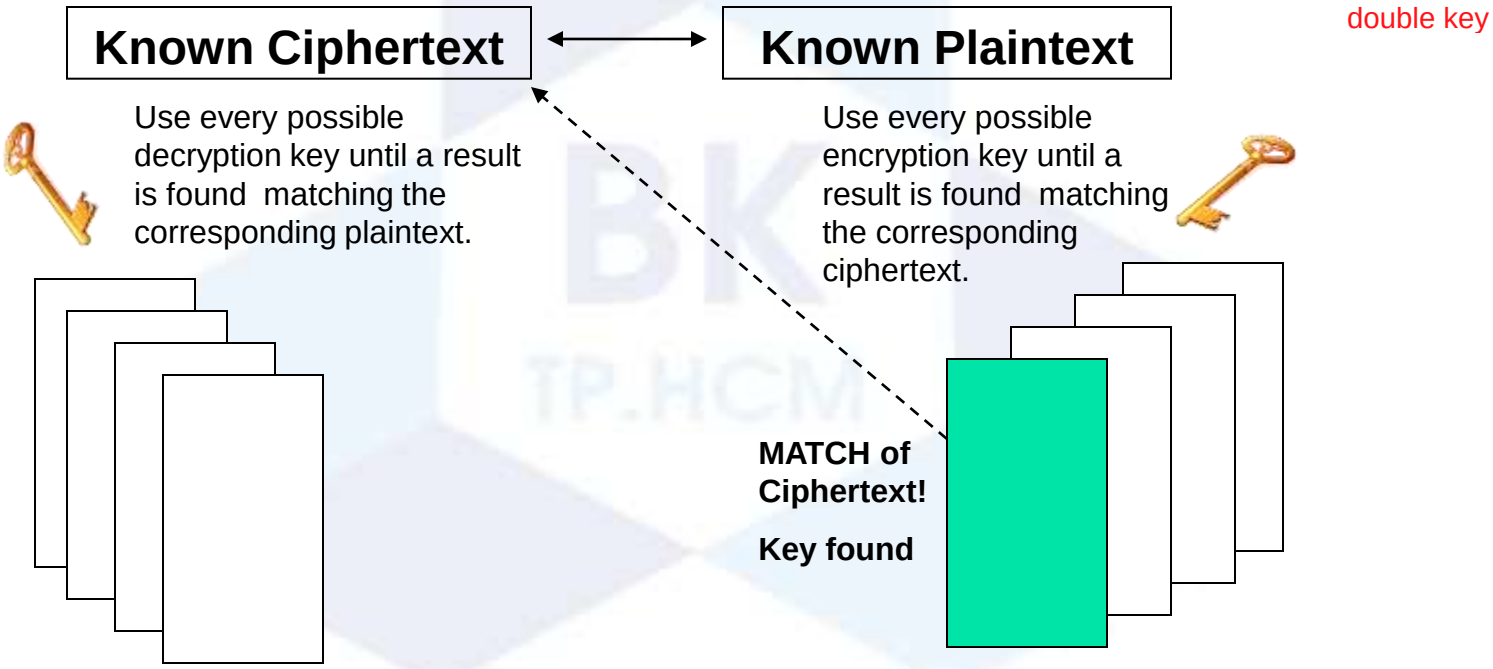
Các phương pháp phân tích mã

- **Tấn công vét cạn(brute force)**
  - Biết thông điệp mã hóa.
  - Cố gắng giải mã với tất cả các khóa có thể.



Các phương pháp phân tích mã

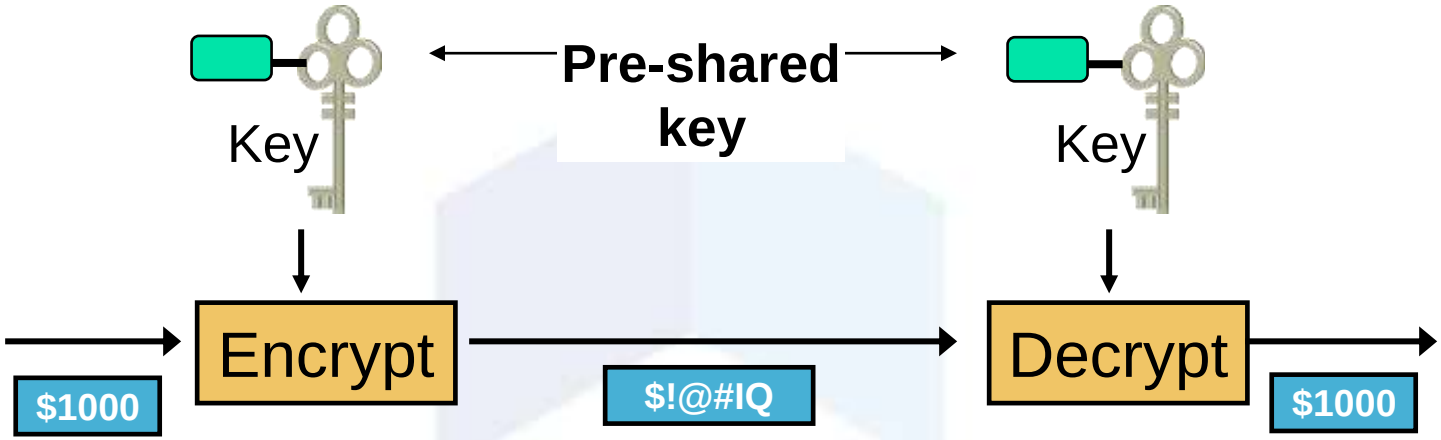
- **Tấn công gặp nhau ở giữa (Meet in the Middle)**
  - Có một số thông điệp rõ và thông điệp đã mã hóa tương ứng.
  - Mã hóa thông điệp bản rõ với các khóa có thể và cùng lúc giải mã thông điệp đã mã hóa với các khóa khác có thể cho đến khi tìm ra được một sự trùng khớp.



Nội dung trình bày

- Giới thiệu
- Mã hóa đối xứng
- Các hệ mã đối xứng truyền thống
- Chuẩn mã hóa dữ liệu
- Các vấn đề khác của mã hóa đối xứng

Mã hóa đối xứng



- Khóa thường dùng có chiều dài 80 - 256 bits
- Xử lý nhanh vì các thuật tuán chỉ dùng các phép toán đơn giản.
- Ví dụ các hệ mã đối xứng như DES, 3DES, AES, IDEA, RC2/4/5/6, và Blowfish.

Mã hóa đối xứng

- **Hai yêu cầu cho việc sử dụng an toàn mã hóa đối xứng**
  - Một thuật toán mã hóa mạnh.
  - Một khóa bí mật chỉ bên gởi và bên nhận biết.
- **Biểu diễn dưới dạng toán học**

  $Y = E_K(X)$

  $X = D_K(Y)$
- Giả sử thuật toán mã hóa được biết.
- Ngầm định có một kênh an toàn để phân phối khóa bí mật.

Khuyết điểm

>< fast -> use

- Mã hóa đối xứng dùng một khóa được chia xẻ giữa bên gởi và bên nhận. Nếu khóa này bị tiết lộ, thông tin liên lạc sẽ bị thương tổn.

  -> change key => key session

- Hệ mã đối xứng xem các bên là như nhau vì vậy không thể bảo vệ bên gửi từ việc giả mạo thông điệp bên nhận và việc tuyên bố thông điệp đó đã được gửi từ bên gửi.

  => NOT INTEGRITY

Nội dung trình bày

- Giới thiệu
- Mã hóa đối xứng
- Các hệ mã đối xứng truyền thống
  - Các hệ mã thay thế
  - Các hệ mã hoán vị
  - Các hệ mã nhân
  - Các máy Rotor
- Chuẩn mã hóa dữ liệu
- Các vấn đề khác của mã hóa đối xứng

Các hệ mã thay thế

miền xác định?

- Thay thế các ký tự của bản rõ bằng các ký tự khác hay bằng các con số hay bằng các biểu tượng.
- Nếu xem bản rõ như một chuỗi các bit thì mã thay thế sẽ thay thế các mẫu các bit thành mẫu các bit khác.
- Ví dụ
  - Mã thay thế đơn ký tự: mã Cesar

    $c = E(p) = (p + k) \bmod (26)$

    $p = D(c) = (c - k) \bmod (26)$
  - Mã thay thế đa ký tự: mã Vigenère

Các hệ mã hoán vị

- Sắp xếp lại thứ tự các ký tự mà không thay đổi các ký tự.
- Bản mã có sự phân bố tần số tương tự như bản rõ.
- Ví dụ mã Rail Fence
  - Viết các ký tự của thông điệp theo đường chéo trên một số hàng sau đó đọc ra bản mã theo hàng.
  - Viết thông điệp "meet me after the toga party" với hàng rào đường sắt có chiều dài bằng 2 là:

    m e m a t r h t g p r y

    e t e f e t e o a a t

    cho bản mã

    MEMATRHTGPRYETEFETEOAAT

Các hệ mã nhân

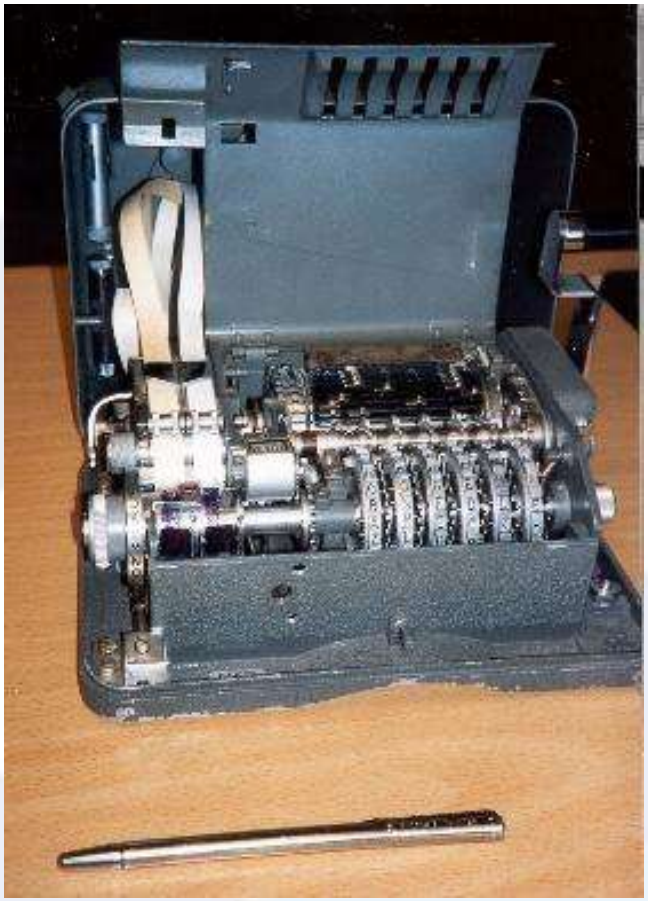
- Mã thay thế hoặc mã hoán vị không an toàn vì vì đặc điểm ngôn ngữ.
- Hai mã thay thế liên tiếp làm cho hệ mã thay thế phức tạp hơn.
- Hai mã hoán vị liên tiếp làm cho hệ mã hoán vị phức tạp hơn.
- Một mã thay thế theo sau một mã hoán vị làm cho hệ mã mới càng phức tạp hơn. Đây chính là hệ mã nhân.
- Cầu nối từ hệ mã cổ điển đến hệ mã hiện đại.

Các máy Rotor(1/2)

- Trước khi có các hệ mã hiện đại, các máy rotor là các hệ mã phức tạp được sử dụng rộng rãi trong thế chiến thứ 2.
  - German Enigma, Allied Hagelin, Japanese Purple
- Hiện thực rất phức tạp gồm nhiều mã thay thế.
- Sử dụng một loạt các xi lanh, một xi lanh cho một mã thay thế, nó di chuyển và thay đổi sau khi một ký tự được mã hóa.
- Với 3 xi lanh ta có $26^3=17576$ ký tự được dùng.

Các máy Rotor(2/2)

- **Máy Hagelin**



Nội dung trình bày

- Giới thiệu
- Mã hóa đối xứng
- Các hệ mã đối xứng truyền thống
- Chuẩn mã hóa đối xứng
  - Phân biệt mã hóa khối và mã hóa dòng
  - Các nguyên tắc cơ bản
  - Cấu trúc mã hóa Feistel
  - Chuẩn mã hóa đối xứng
- Các vấn đề khác của mã hóa đối xứng

Phân biệt mã hóa khối và mã hóa dòng

- **Mã hóa khối**
  - Xử lý các thông điệp theo từng khối.
  - Chiều dài từng khối là 64 bits hoặc lớn hơn.
  - Từng khối này sẽ được mã hóa hoặc giải mã.
  - Phạm vi ứng dụng lớn hơn.
- **Mã hóa dòng**
  - Xử lý các thông điệp theo từng bit hay byte tại mỗi thời điểm.
  - Từng bit hay byte sẽ được mã hóa hoặc giải mã.

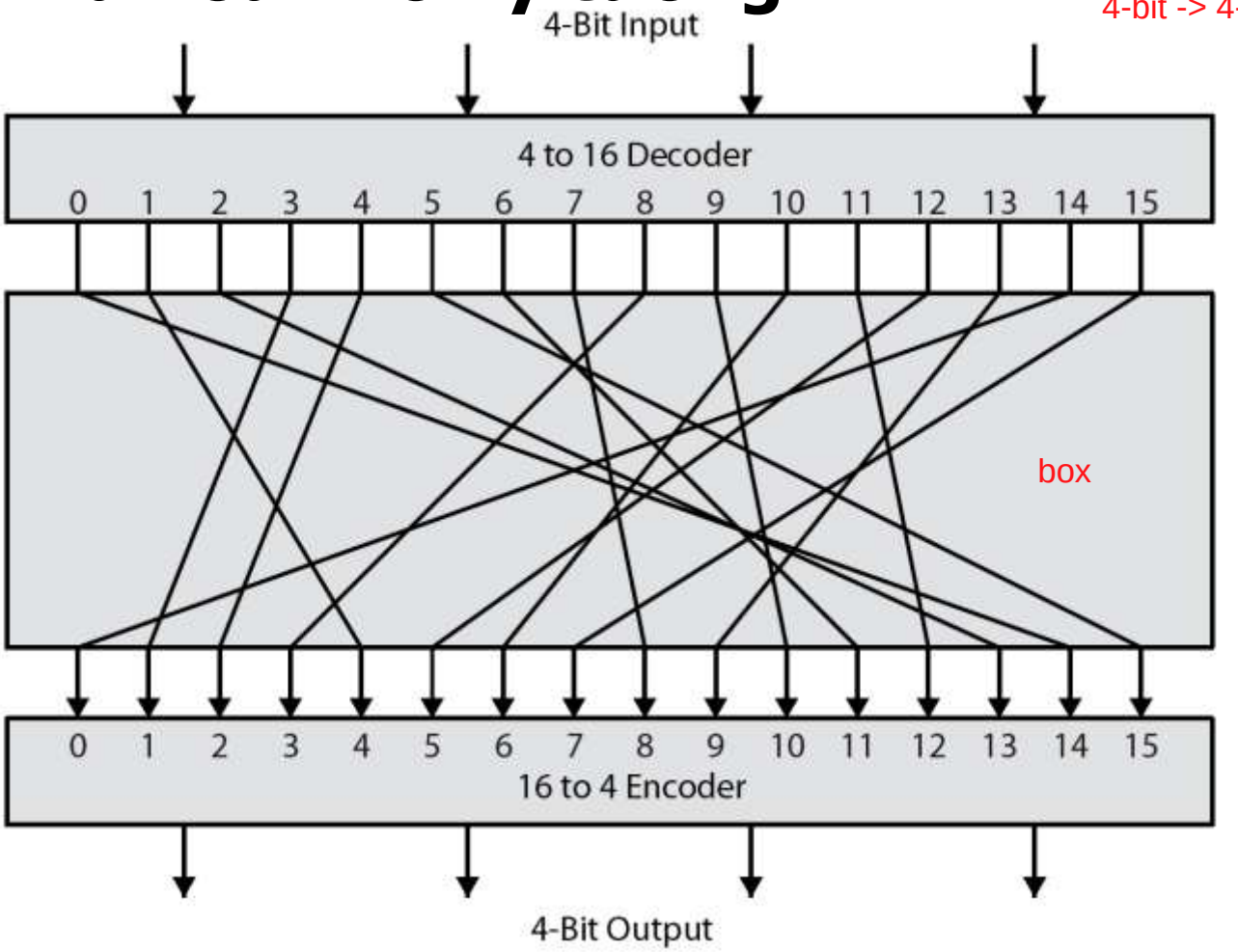
Các nguyên tắc cơ bản(1/4)

- Hầu hết các hệ mã đối xứng dựa trên cấu trúc mã hóa Feistel.
- Mã hóa giống như mã thay thế lớn đặc biệt.
- Mã thay thế n bits tổng quát thì chiều dài khóa là $n \times 2^n$
- Để giải quyết điều này, Feistel chỉ ra rằng:
  - Xây dựng dựa trên các khối nhỏ hơn tương tự như mã hóa khối lý tưởng.
  - Kết hợp chúng lại dựa trên ý tưởng của mã hóa nhân sử dụng mạng thay thế - hoán vị do Claude Shannon đề nghị.

Các nguyên tắc cơ bản(2/4)

- **Mã hóa khối lý tưởng**

4-bit -> 4-bit



non-linear vs linear

Các nguyên tắc cơ bản(3/4)

- **Mạng thay thế - hoán vị**
  - 1949: Claude Shannon giới thiệu mạng thay thế - hoán (substitution-permutation (S-P) network)
  - Mạng thay thế hoán vị dựa trên 2 phép toán mật mã nguyên thủy (đã xem xét trước đây)
    - Thay thế(*substitution* còn gọi là S-box)
    - Hoán vị(*permutation* còn gọi là P-box)
  - Cung cấp sự khuếch tán và sự nhầm lẫn cho thông điệp và khóa.

Các nguyên tắc cơ bản(4/4)

- **Khuếch tán** không gian search => Search all !!!
  - Làm mất đi cấu trúc thống kê của bản rõ trên bản mã nhận được.
  - Nhằm ngăn cản các cố gắng suy luận khóa.
- **Nhầm lẫn** 54 / 55 bit, feel not OK => right way?
  - Làm cho mối quan hệ giữa các bản mã và khóa phức tạp như có thể.
  - Nhằm ngăn cản các cố gắng phát hiện khóa.
- Khuếch tán(diffusion) và nhầm lẫn(confusion) là nền tảng cho thiết kế mã hóa khối hiện đại.

Cấu trúc mã hóa Feistel(1/4)

- Horst Feistel làm việc ở phòng thí nghiệm nghiên cứu Thomas J Watson - IBM, đưa ra một cấu trúc mã hóa Feistel vào thập kỷ 70.
- Dựa trên mã hóa nhân khả nghịch.
- Hiện thực khái niệm mạng thay thế - hoán vị của Shannon.

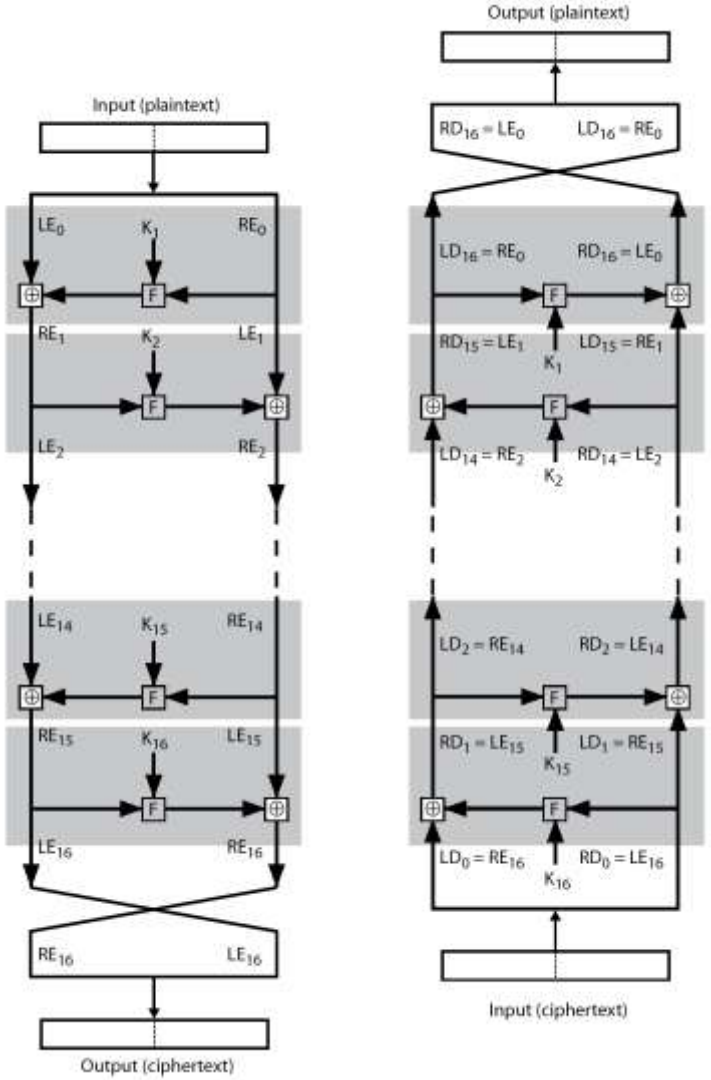
Cấu trúc mã hóa Feistel(2/4)

- **Cách mã hóa**
  - Xử lý qua nhiều vòng
  - Trong mỗi vòng
    - Chia khối dữ liệu đầu vào thành 2 phần
    - Thực hiện thay thế trong ½ dữ liệu bên trái.
    - Thực hiện hàm round trên ½ dữ liệu bên phải và khóa con sinh ra từ khóa ở mỗi vòng.
    - Sau đó hai nữa này sẽ được hoán vị và tiếp tục vòng khác.

Cấu trúc mã hóa Feistel(4/4)

- **Các đặc điểm cần lưu ý**
  - Mã hóa/giải mã nhanh.
  - Dể dàng phân tích.
- **Các tham số ảnh hưởng**
  - Kích thước khối.
  - Kích thước khóa.
  - Số lượng vòng.
  - Thuật toán tạo khóa con.
  - Hàm round.

Cấu trúc mã hóa Feistel(3/4)



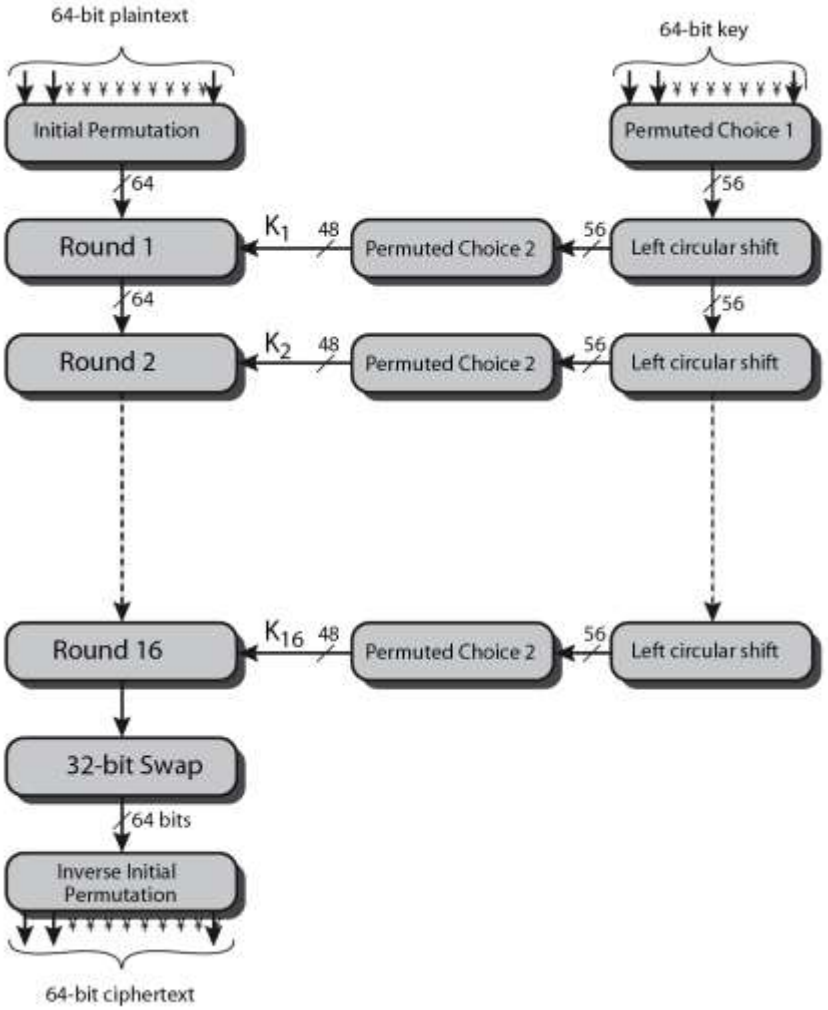
Chuẩn mã hóa dữ liệu

- **DES(Data Encryption Standard)**
  - Chuẩn mã hóa dữ liệu
  - Được NBS(National Bureau of Standards) đề nghị vào năm 1977.
  - Khối dữ liệu có chiều dài 64 bits
  - Khóa bí mật có chiều dài 56 bits

Lược đồ mã hóa DES(1/7)

- Một hoán vị khởi tạo nhằm trộn 64 bits của khối đầu vào.
- 16 vòng được thực hiện tương tự như cấu trúc mã Feistel.
- Một hoán vị kết thúc là ánh xạ ngược của hoán vị khởi tạo.
- Tạo khóa con
  - Một hoán vị khởi tạo trên khóa 64 bits để chọn ra 56 bits.
  - Mỗi vòng sẽ tạo ra một khóa con dựa trên một phép toán shift vòng trái và một phép hoán vị.

Lược đồ mã hóa DES(2/7)



Lược đồ mã hóa DES(3/7)

- **Hoán vị khởi tạo và ánh xạ ngược**
  - Được định nghĩa bởi các bảng 3.2a , 3.2b
  - Sắp xếp lại các bit dữ liệu đầu vào.
  - Dễ dàng hiện thực bằng phần cứng.
- **Ví dụ**

```
Hoán vị khởi tạo của 675a6967 5e5a6b5a
là: ffb2194d 004df6fb
```

Lược đồ mã hóa DES(4/7)

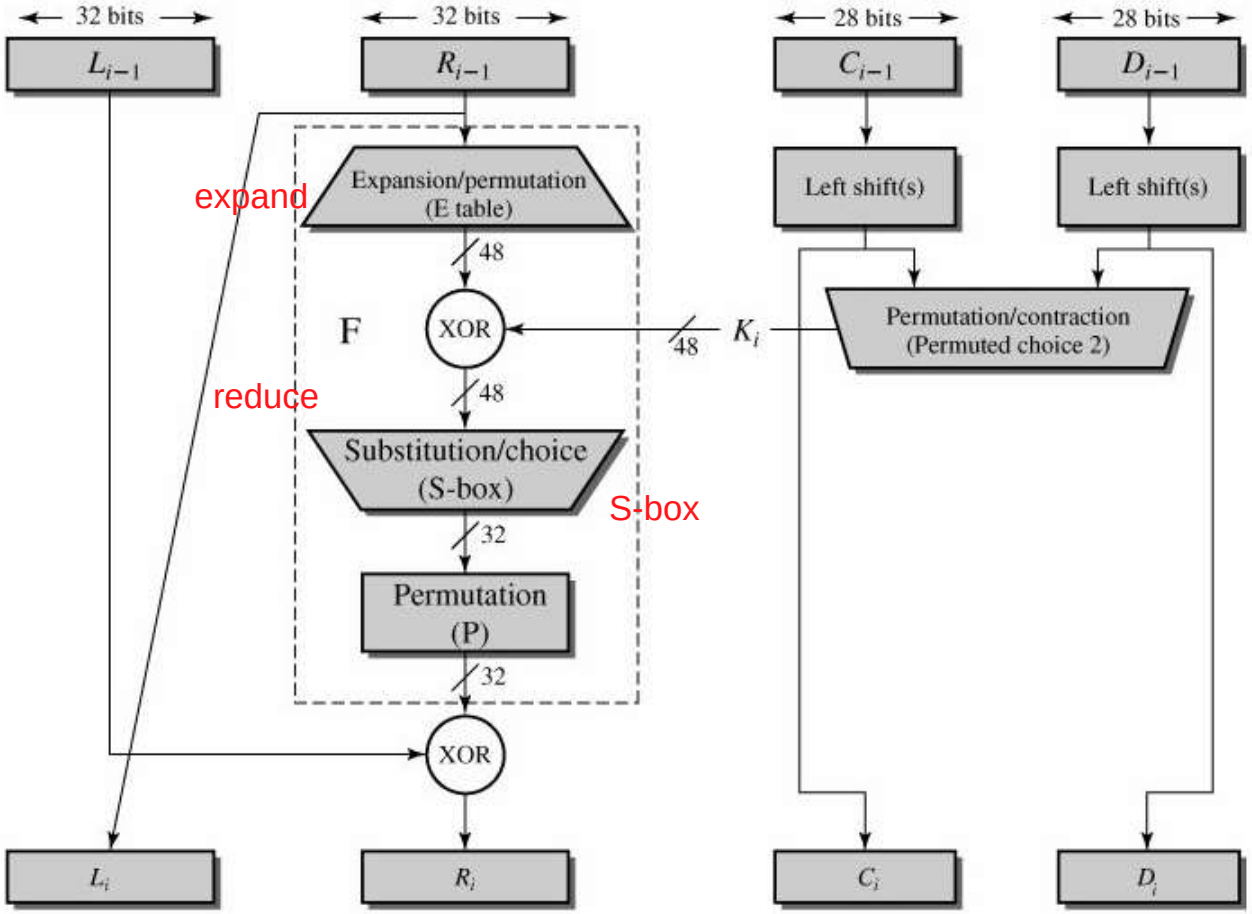
- **Chi tiết trên mỗi vòng**
  - Dùng hai nữa 32 bits bên trái và bên phải.
  - Được mô tả như sau:

    $L_i = R_{i-1}$

    $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$
- **Hàm Round F**
  - Lấy 32 bits bên phải và khóa con 48 bits
  - Mở rộng 32 bits bên phải thành 48 bits dùng E
  - Cộng với khóa con dùng XOR
  - Đưa qua 8 S-box để lấy ra kết quả là 32 bits.
  - Cuối cùng hoán vị dùng P.

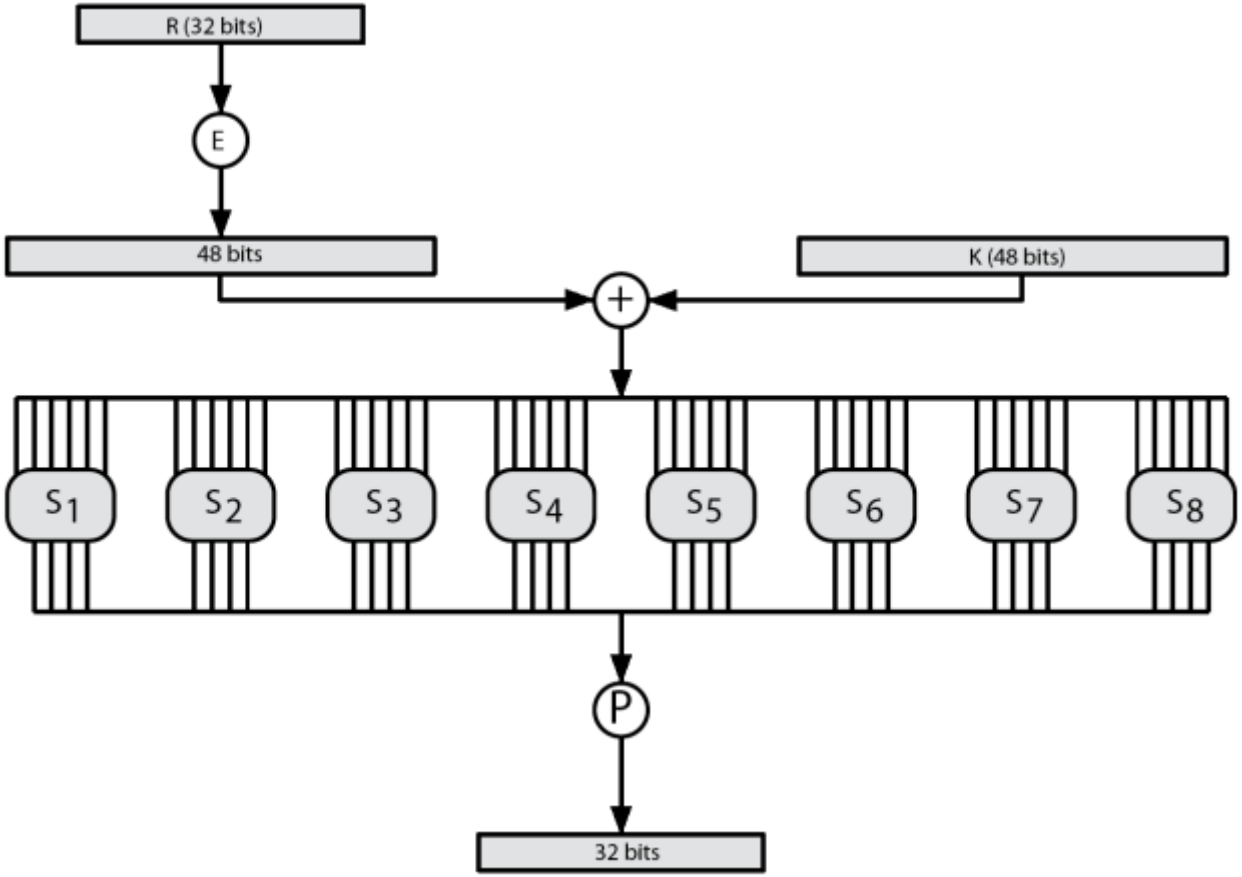
Lược đồ mã hóa DES(5/7)

- **Chi tiết trên mỗi vòng**



Lược đồ mã hóa DES(6/7)

- **Chi tiết hàm F**



Lược đồ mã hóa DES(7/7)

- **S box**
  - Ánh xạ 6 bits thành 4 bits
  - Bit 1 và 6 là giá trị hàng.
  - Các bit từ 2 đến 5 là giá trị cột.
  - Kết quả trên mỗi S box là 4 bits.
  - Việc lựa chọn hàng phụ thuộc vào cả dữ liệu và khóa.
- **Ví dụ**
  - `S(18 09 12 3d 11 17 38 39) = 5fd25e03`

Lược đồ giải mã DES

- Với cấu trúc Feistel, giải mã sử dụng cùng một thuật toán với mã hóa tuy nhiên các khóa con được sử dụng theo thứ tự đảo ngược.

Tiêu chuẩn thiết kế DES

- **Được Coppersmith báo cáo trong [COPP94]**
- **7 tiêu chuẩn cho S box nhằm:**
  - Không tuyến tính.
  - Chống lại phân tích mã sai biệt(differential cryptanalysis).
  - Nâng cao sự nhầm lẫn(good confusion).
- **3 tiêu chuẩn cho hoán vị P nhằm:**
  - Gia tăng sự khuếch tán.

Đánh giá sức mạnh của DES

- **Hiệu ứng dồn dập**

confusion

diffusion

(a) Change in Plaintext		(b) Change in Key	
Round	Number of bits that differ	Round	Number of bits that differ
0	1	0	0
1	6	1	2
2	21	2	14
3	35	3	28
4	39	4	32
5	34	5	30
6	32	6	32
7	31	7	35
8	29	8	34
9	42	9	40
10	44	10	38
11	32	11	31
12	30	12	33
13	30	13	28
14	26	14	26
15	29	15	34
16	34	16	35

6 round?

half change
-> best

Đánh giá sức mạnh của DES

- **Kích thước khóa**
  - Khóa có chiều dài 56 bits.
  - Có $2^{56} \sim 7.2 \times 10^{16}$ giá trị.
  - Tấn công brute force là không thực tế.
- **Các tấn công thực tế**
  - Vào 1997 trên Internet với ít tháng.
  - Vào 1998 trên phần cứng chuyên dụng của EFF(Electronic Frontier Foundation) trong ít ngày.
  - Vào 1999 với sự kết hợp thì chỉ còn lại 22 giờ.
- Tuy nhiên vẫn còn phải nhận ra bản rõ.
- Xem xét hệ mã thay thế cho DES.

Nội dung trình bày

- Giới thiệu
- Mã hóa đối xứng
- Các hệ mã đối xứng truyền thống
- Chuẩn mã hóa dữ liệu
- Các vấn đề khác của mã hóa đối xứng
  - Mã hóa DES nhiều lần
  - Chuẩn mã hóa tiên tiến
  - Các chế độ hoạt động của mã hóa

Mã hóa DES nhiều lần(1/4)

- **Một thay thế cho DES là cần thiết**
  - Các tấn công burte-force có khả năng làm tổn thương DES.
- **Một hướng tiếp cận là thiết kế một thuật toán hoàn toàn mới**
  - AES là ví dụ điển hình.
- **Một hướng tiếp cận khác sẽ dùng mã hóa nhiều lần với DES**
  - Triple-DES là hình thức được chọn.

Mã hóa DES nhiều lần(2/4)

- **Double DES**
  - Dùng mã hóa DES trên mỗi khối

    $C = E_{K2}(E_{K1}(P))$
- **Tấn công gặp nhau ở giữa**
  - Sử dụng khi mã hóa 2 lần
  - Vì : $X = E_{K1}(P) = D_{K2}(C)$
  - Tấn công bằng cách mã hóa P với tất cả các khóa và lưu trữ lại.
  - Sau đó giải mã C với các khóa và so trùng giá trị X.
  - Cần $O(2^{56})$ bước

Mã hóa DES nhiều lần(3/4)

- **Triple-DES**
  - Phải 3 lần mã hóa
  - Cũng sẽ cần 3 khóa riêng biệt
- **Triple-DES với 2 khóa**
  - Có thể dùng 2 khóa với thứ tự E-D-E

    $C = E_{K1}(D_{K2}(E_{K1}(P)))$
  - Vì mã hóa và giải mã là tương đương trong vấn đề an toàn
  - Nếu K1=K2 thì lại tương đương với DES
  - Chuẩn trong ANSI X9.17 & ISO8732
  - Không có một tấn công thực tế được biết hiện nay

Mã hóa DES nhiều lần(4/4)

- **Triple-DES với 3 khóa**
  - Mặc dù không có một cuộc tấn công thực tế nào trên Triple-DES với 2 khóa tuy nhiên có một số chỉ dẫn trong [1], chương 6, mục 1.
  - Có thể dùng Triple-DES với 3 khóa để tránh những vấn đề trên

    $C = E_{K3}(D_{K2}(E_{K1}(P)))$
  - Đã được sử dụng trong một vài ứng dụng như PGP, S/MIME.

Chuẩn mã hóa tiên tiến

- **AES(Advanced Encryption Standard)**
  - Chuẩn mã hóa tiên tiến
  - Khối dữ liệu có chiều dài 128 bits
  - Khóa bí mật có chiều dài 128 bits, 192 bits, 256 bits

Các chế độ hoạt động của mã khóa

- **Các hệ mã khối mã hóa trên các khối có kích thước cố định**
  - Ví dụ: DES mã hóa khối 64 bits với khóa 56 bits.
  - Cần một số cách để mã hóa/giải mã một số lượng dữ liệu tùy ý trong thực tế.
- **ANSI X3.106-1983 (bây giờ là FIPS 81)**
  - Định nghĩa 4 chế độ được dùng.
  - Sau đó định nghĩa 5 chế độ cho AES & DES.
  - Các chế độ được dùng với các hệ mã khối và các hệ mã dòng.

Thêm vào thông điệp

padding

- **Ở phía cuối thông điệp sẽ có một khối ngắn**
  - Thêm vào các byte 0
  - Thêm vào các byte khoảng trắng(0x20)
  - Thêm vào các byte với cùng một giá trị như số lượng byte cần thêm vào (PKCS5,PKCS7, …)

    DES INPUT BLOCK = f o r _ _ _ _ _

    (IN HEX) 66 6F 72 05 05 05 05 05
  - Thêm với 0x80 theo sau là các byte 0 (null).
  - Thêm các byte 0, còn byte cuối là số byte thêm vào.

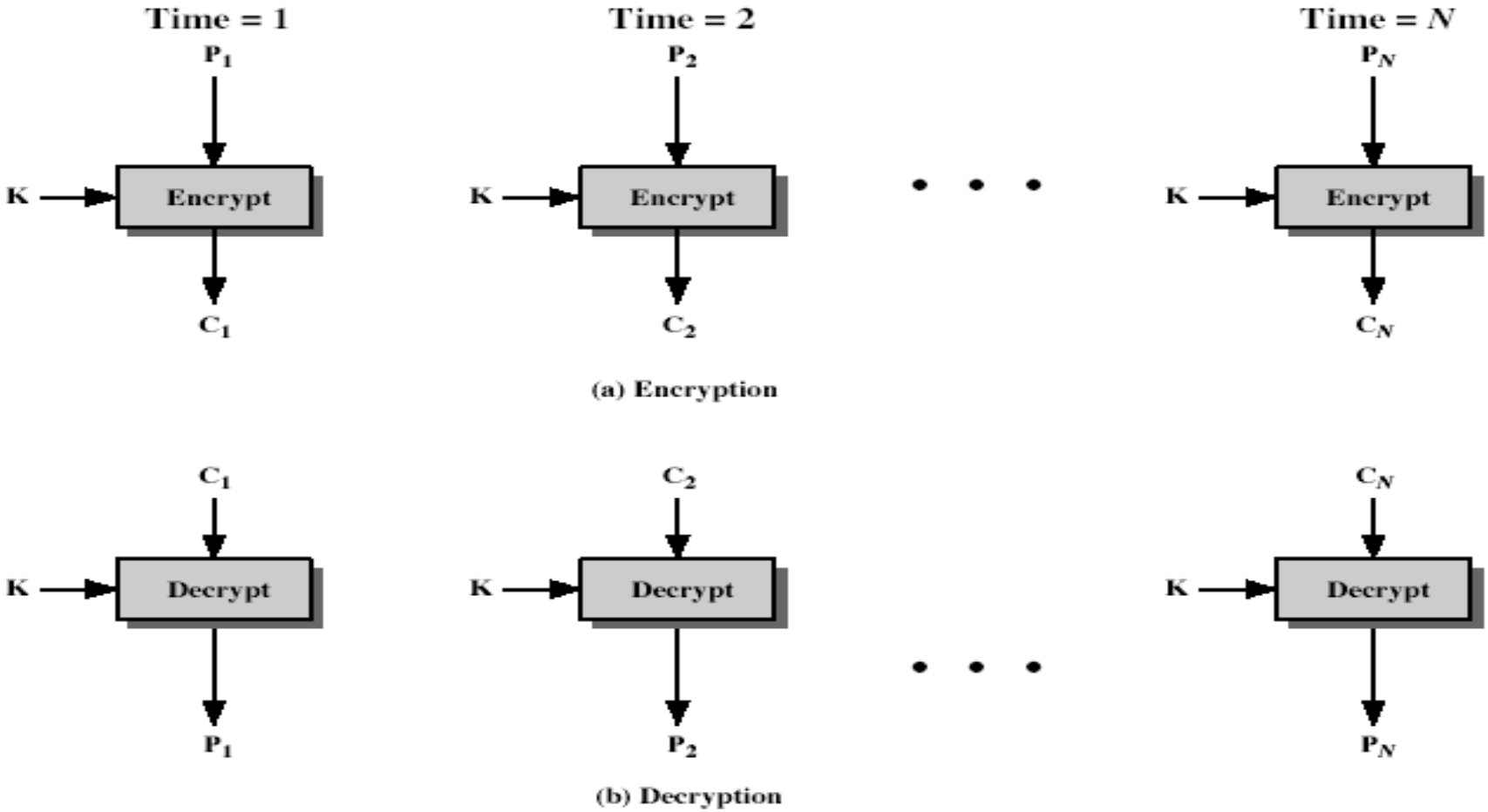
Electronic Code Book (ECB)

- Thông điệp được chia thành các khối độc lập.
- Mỗi khối được mã hóa độc lập với các khối khác.

$$C_i = DES_{K1}(P_i)$$

- Được dùng khi chỉ có một ít khối cần được gởi.

Electronic Codebook Book (ECB)



Thuận lợi và hạn chế của ECB

- **Thuận lợi**
  - Đơn giản
  - Thực hiện mã hóa/giải mã trên nhiều khối một cách đồng thời
- **Hạn chế**
  - Yếu do các khối thông điệp được mã hóa được độc lập.
  - Nếu các khối giống nhau thì các khối mã cũng lặp đi lặp lại trong bản mã dẫn đến dể dàng phân tích.
  - Ví dụ như dữ liệu đồ họa.

Cipher Block Chaining (CBC)

popular

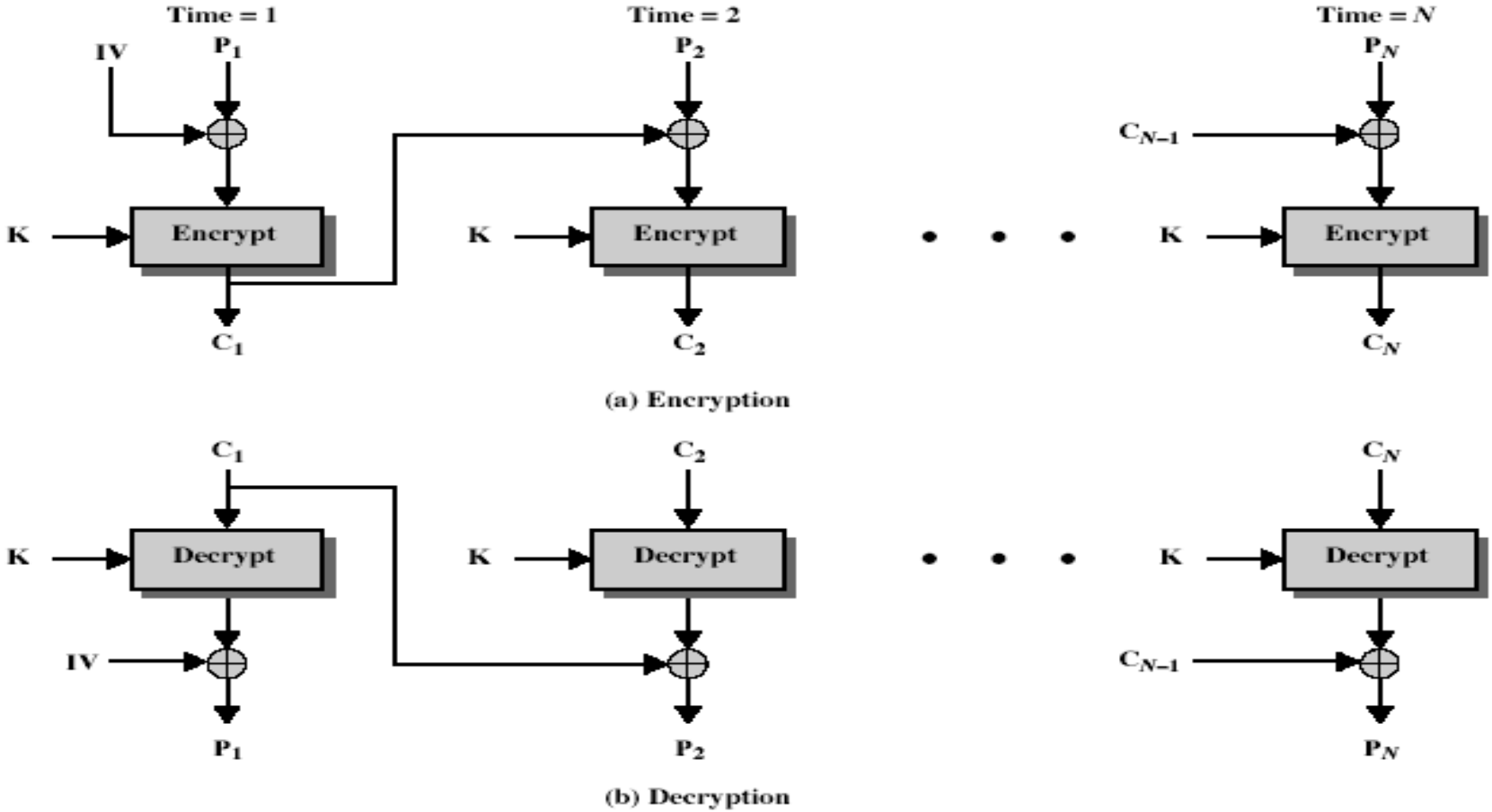
- Thông điệp được chia thành các khối.
- Các khối được liên kết với nhau trong phép toán mã hóa.
- Dùng vector khởi tạo (Initial Vector - IV) để bắt đầu:

$$C_{-1} = IV$$

$$C_i = DES_{K1}(P_i \text{ XOR } C_{i-1})$$

- Được dùng khi dữ liệu cần mã hóa là lớn.

Cipher Block Chaining (CBC)



Thuận lợi và hạn chế của CBC

- **Thuận lợi**
  - Mỗi khối mã hóa phụ thuộc vào tất cả các khối trước nó.
  - Bất kỳ thay đổi nào trên một khối bản rõ sẽ ảnh hưởng đến tất cả các khối mã hóa sau đó.
- **Hạn chế**
  - Cần có vector khởi tạo IV và phải được biết bởi cả bên gởi và bên nhận.
  - Nếu nó được gởi dạng bản rõ thì kẻ tấn công có thể thay đổi các bit trong khối đầu tiên và thay đổi IV để bù lại. integrity
  - Vậy IV phải là một giá trị được cố định hoặc được gởi dạng mã hóa trong chế độ ECB trước.

Cipher FeedBack (CFB)

- Thông điệp được xem như một dòng các bit.
- Thêm vào đầu ra của khối mã khóa một số các bit.
- Kết quả trở lại cho bước kế tiếp.
- Chuẩn cho phép các bit(1, 8, 64 hay 128 etc) được trở lại bước kế tiếp
  - CFB-1, CFB-8, CFB-64, CFB-128.

$$C_i = P_i \text{ XOR } DES_{K1}(C_{i-1})$$

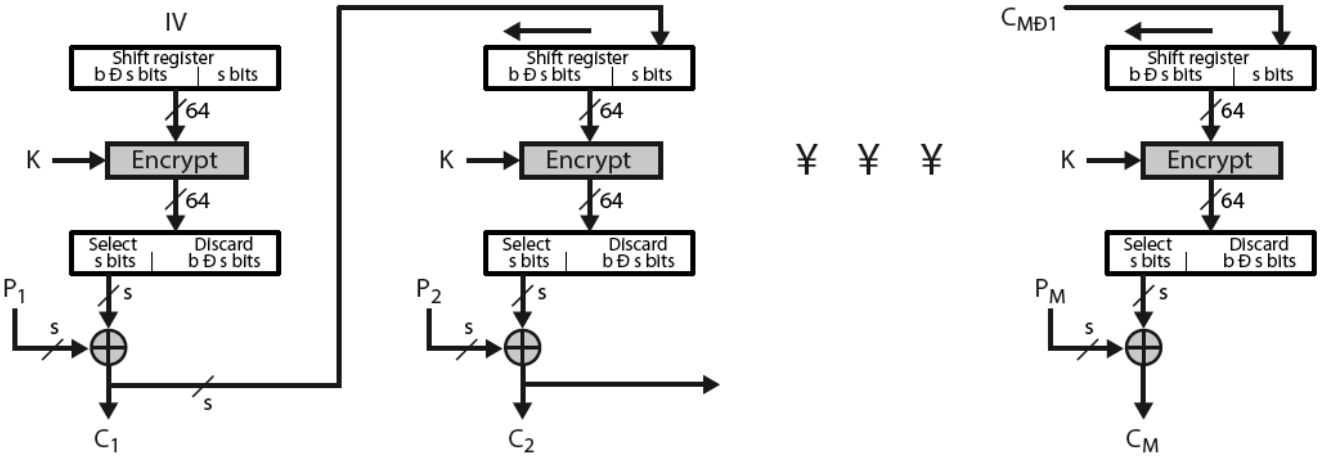
$$C_{-1} = IV$$

$$P_i = C_i \text{ XOR } DES_{K1}(P_{i-1})$$

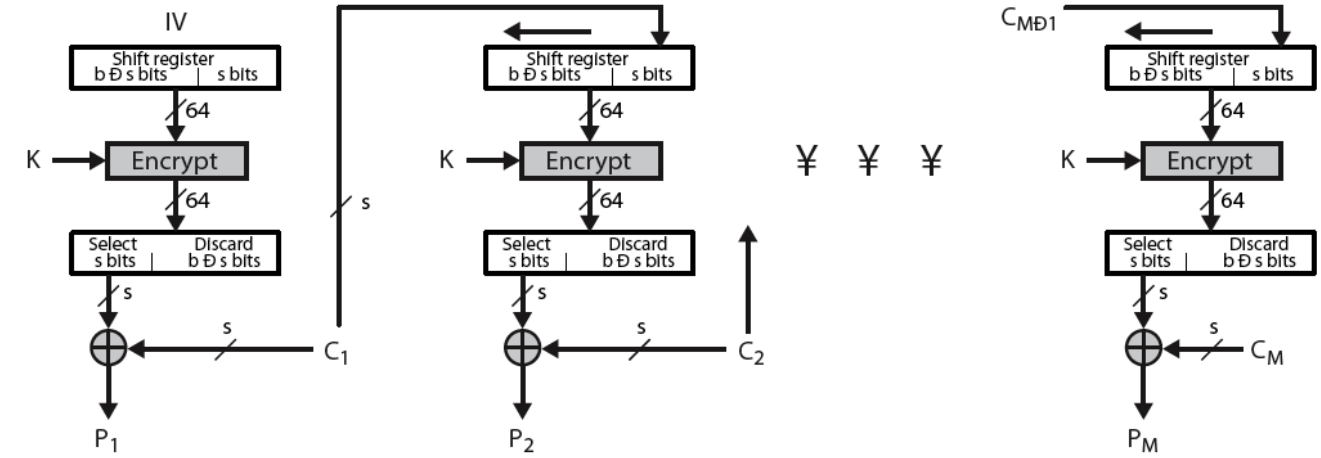
$$P_{-1} = IV$$

- Hiệu quả khi dùng tất cả các bit trong khối.
- Được dùng trong mã hóa dòng hay xác thực.

Cipher FeedBack (CFB)



(a) Encryption



(b) Decryption

Thuận lợi và hạn chế của CFB

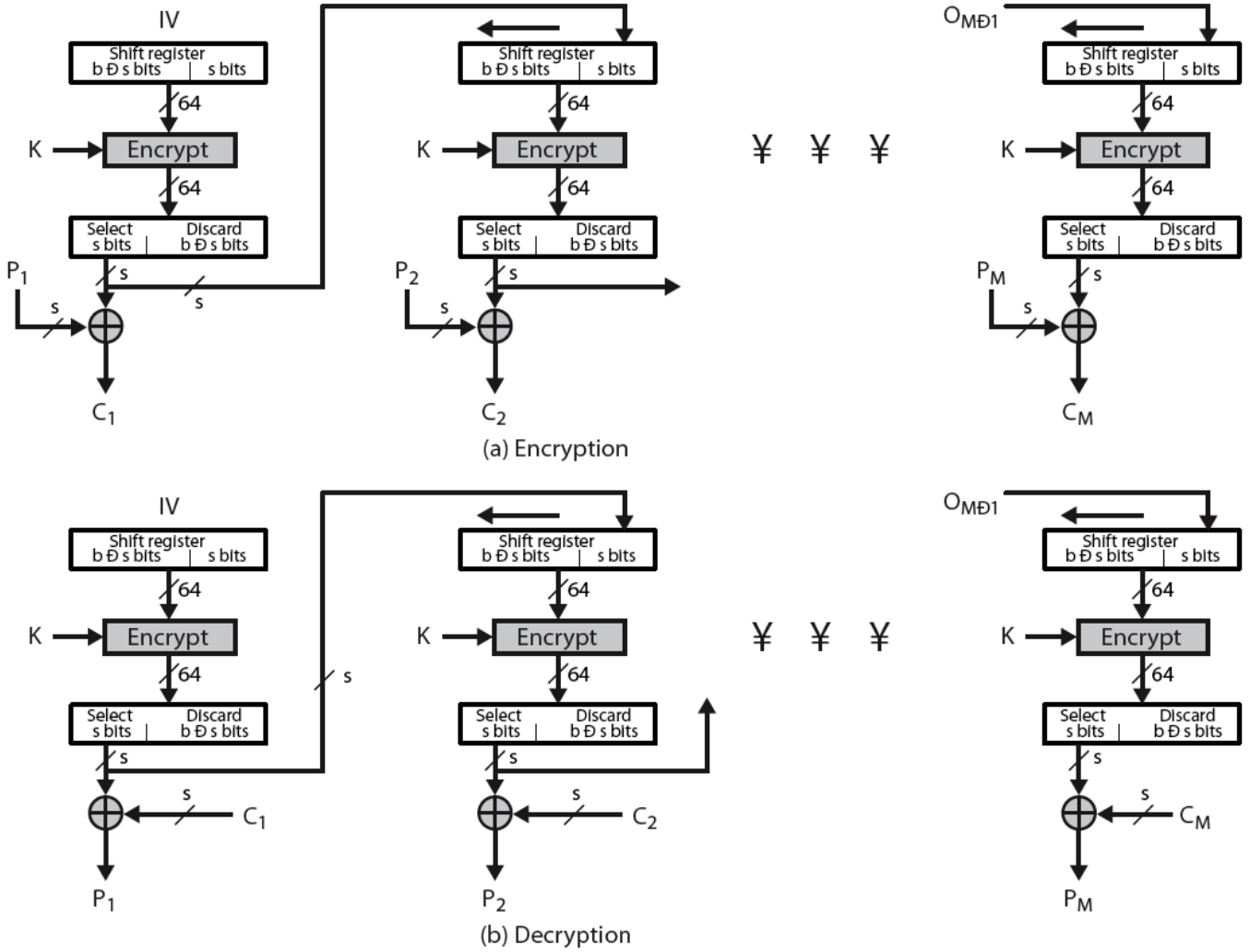
- **Thuận lợi**
  - Thích hợp với dữ liệu đến là bit/byte.
  - Là chế độ hoạt động phổ biến nhất của mã hóa dòng.
- **Hạn chế**
  - Lỗi lan truyền trên nhiều khối sau khi có lỗi xảy ra.
  - Phải truyền trên tầng vận chuyển tin cậy.

Output FeedBack (OFB)

- Thông điệp được xem như một dòng các bit.
- Thêm vào đầu ra của khối mã khóa một số các bit.
- Kết quả trở lại cho bước kế tiếp.
- Trở lại là độc lập với thông điệp.

$$C_i = P_i \text{ XOR } O_i$$
$$O_i = DES_{K1}(O_{i-1})$$
$$O_{-1} = IV$$
$$P_i = C_i \text{ XOR } O_i$$
$$O_i = DES_{K1}(O_{i-1})$$
$$O_{-1} = IV$$

Output FeedBack (OFB)



Thuận lợi và hạn chế của OFB

- **Thuận lợi**
  - Các lỗi trên chuỗi bit sẽ không bị lan truyền.
- **Hạn chế**
  - Có nhiều tổn thương trên tấn công thay đổi dòng thông điệp.
  - Các nghiên cứu đã chỉ ra rằng chỉ nên sử dụng OFB đầy đủ như OFB-64/128.

Counter (CTR)

- Là một phiên bản của OFB.
- Mặc dù được đề nghị trong nhiều năm trước nhưng nó chỉ thật sự là chuẩn khi dùng với AES.
- Tương tự như OFB nhưng mã hóa trên giá trị counter chứ không phải trên giá trị phản hồi.
- Giá trị khóa/counter là khác nhau cho mỗi khối bản rõ(không tái sử dụng)

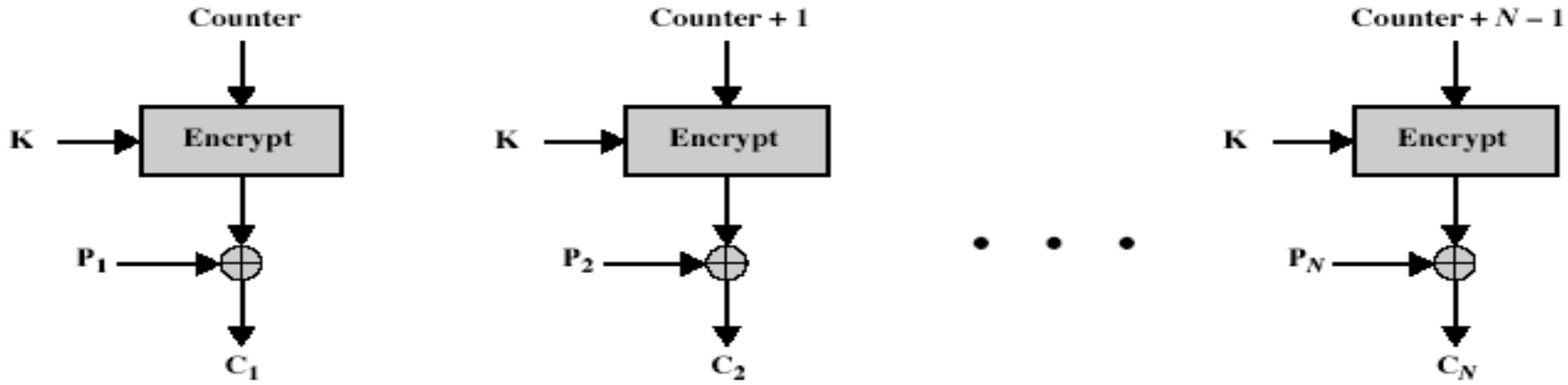
$$C_i = P_i \text{ XOR } O_i$$

$$O_i = DES_{K1}(i)$$

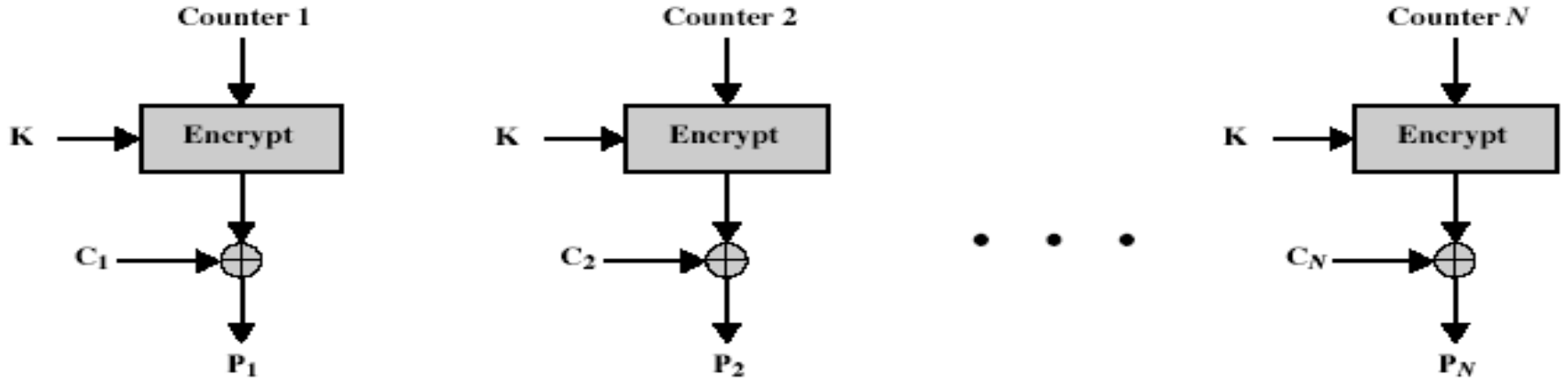
$$P_i = C_i \text{ XOR } O_i$$

- Được dùng mã hóa trên các mạng có tốc độ cao.

Counter (CTR)



(a) Encryption



(b) Decryption

Thuận lợi và hạn chế của CTR

- **Thuận lợi**
  - Tính hiệu quả cao.
    - Có thể thực hiện mã hóa song song trên phần cứng hay phần mềm.
    - Có thể tiền xử lý trước khi cần.
    - Phù hợp với các liên kết mạng tốc độ cao.
  - Truy cập ngẫu nhiên đến các khối dữ liệu đã mã hóa.
- **Hạn chế**
  - Phải chắc chắn rằng không bao giờ tái sử dụng lại cùng giá trị khóa và giá trị counter. Nếu không sẽ dễ bẻ gãy như OFB.

Tóm tắt

- Mã hóa đối xứng là một dạng mật mã, trong đó mã hóa và giải mã được thực hiện bằng cách sử dụng cùng một khóa.
- Các mã hóa đối xứng truyền thống sử dụng các kỹ thuật thay thế và hoán vị. Mã hóa nhân là cầu nối để đến với mã hóa đối xứng hiện đại.
- DES được sử dụng rộng rãi trước đây. DES thực hiện trên khối 64 bits và khóa 56 bits.
- Có nhiều chế độ hoạt động khi mã hóa đối xứng.