

Chuẩn Mã Hóa Dữ Liệu Data Encryption Standard (DES)

Mục đích

- ❑ To review a short history of DES

Sơ lược lịch sử về DES

- ❑ To define the basic structure of DES

Định nghĩa cấu trúc cơ bản của DES

- ❑ To describe the details of building elements of DES

Mô tả chi tiết về các thành phần cấu thành DES

- ❑ To describe the round keys generation process

Mô tả tiến trình tạo các khóa tuần hoàn

- ❑ To analyze DES

Phân tích DES

6-1 INTRODUCTION

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).

Chuẩn mật mã dữ liệu (DES) là một mã khối khóa đối xứng được công bố ra bởi Viện Quốc gia về Tiêu chuẩn và Công nghệ.

Topics discussed in this section:

6.1.1 History

6.1.2 Overview

6.1.1 History

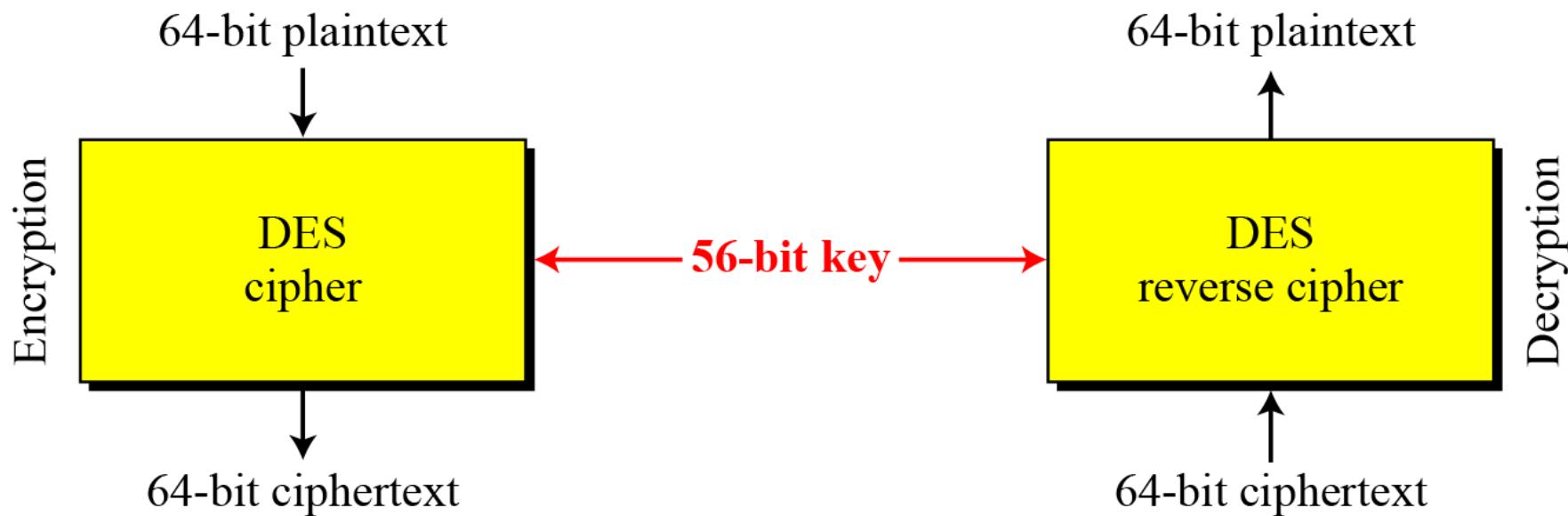
In 1973, NIST published a request for proposals for a national symmetric-key cryptosystem. A proposal from IBM, a modification of a project called Lucifer, was accepted as DES. DES was published in the Federal Register in March 1975 as a draft of the Federal Information Processing Standard (FIPS).

Năm 1973, NIST đưa ra yêu cầu cho đề xuất về hệ thống mật mã khóa đối xứng cho Quốc gia. Một đề xuất từ IBM, là dự án đã chỉnh sửa gọi là Lucifer để được chấp nhận gọi là DES. DES được công bố trong đăng ký Liên Bang tháng 3 năm 1975, gọi là phiên bản đầu của Chuẩn xử lý thông tin Liên Bang (FIPS).

6.1.2 Overview

*DES is a block cipher, as shown in Figure 6.1.
DES là một mã khối, chỉ ra như trên hình 6.1*

Figure 6.1 Encryption and decryption with DES



6-2 DES STRUCTURE

The encryption process is made of two permutations (P-boxes), which we call initial and final permutations, and sixteen Feistel rounds.

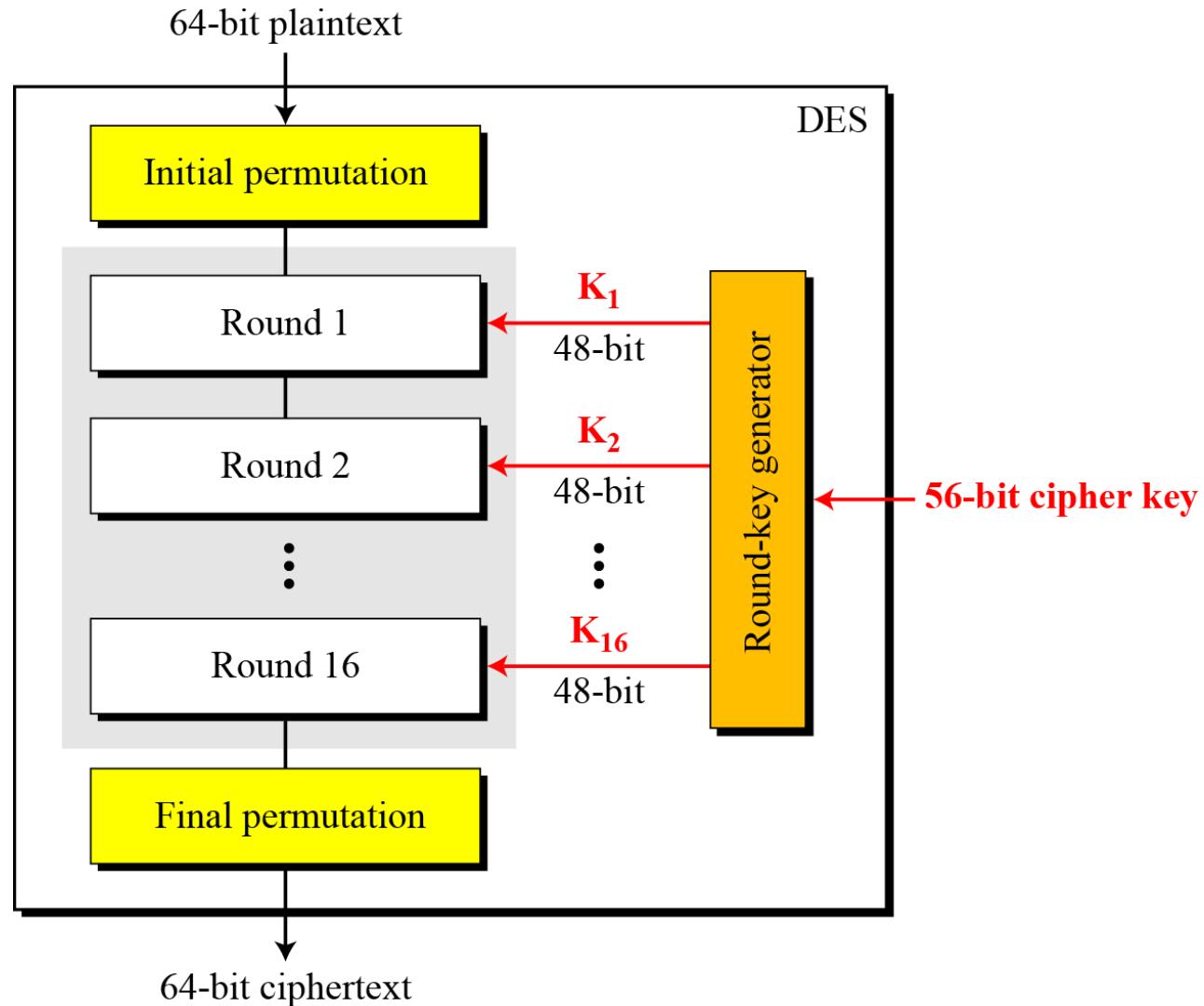
Quá trình mã hóa được tạo thành bởi 2 phép hoán vị P – box, đây là những phép hoán vị khởi tạo và kết thúc, và gồm 16 chu kỳ (vòng) Feistel.

Topics discussed in this section:

- 6.2.1 Initial and Final Permutations: Phép hoán vị khởi tạo và kết thúc
- 6.2.2 Rounds: chu kỳ (vòng)
- 6.2.3 Cipher and Reverse Cipher: mật mã và mật mã nghịch đảo
- 6.2.4 Examples: ví dụ

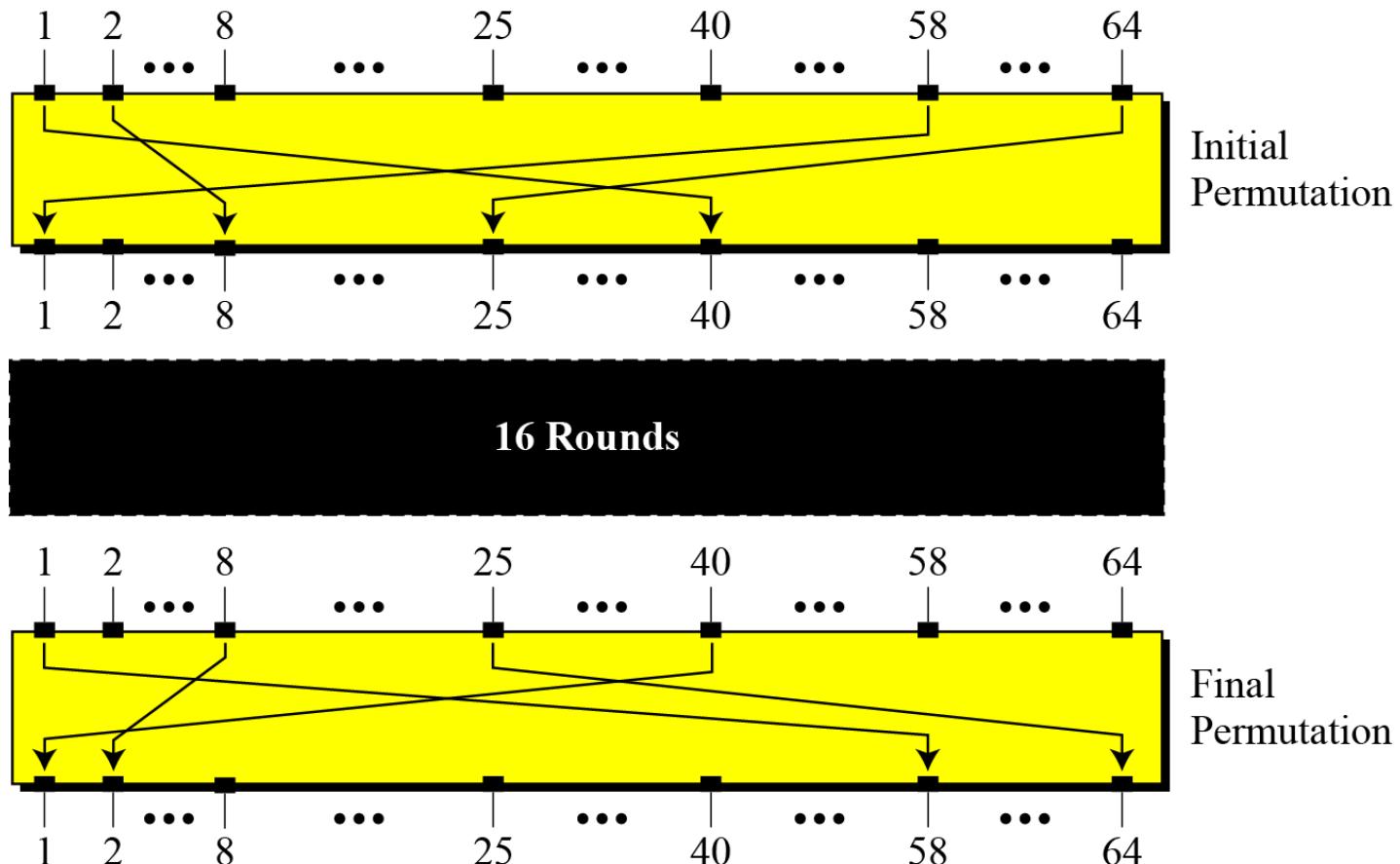
6-2 Continue

Figure 6.2 General structure of DES
Cấu trúc chung của mã hóa DES



6.2.1 Initial and Final Permutations

Figure 6.3 Initial and final permutation steps in DES
Các bước hoán vị khởi tạo và kết thúc trong DES



6.2.1 Continue

Table 6.1 Initial and final permutation tables
Bảng hoán vị khởi tạo và kết thúc

<i>Initial Permutation</i>	<i>Final Permutation</i>
58 50 42 34 26 18 10 02	40 08 48 16 56 24 64 32
60 52 44 36 28 20 12 04	39 07 47 15 55 23 63 31
62 54 46 38 30 22 14 06	38 06 46 14 54 22 62 30
64 56 48 40 32 24 16 08	37 05 45 13 53 21 61 29
57 49 41 33 25 17 09 01	36 04 44 12 52 20 60 28
59 51 43 35 27 19 11 03	35 03 43 11 51 19 59 27
61 53 45 37 29 21 13 05	34 02 42 10 50 18 58 26
63 55 47 39 31 23 15 07	33 01 41 09 49 17 57 25

6.2.1 *Continued*

Example 6.1

Find the output of the final permutation box when the input is given in hexadecimal as:

Tìm đầu ra của **hộp hoán vị kết thúc** khi đầu vào là số có cơ số 16

0x0000 0080 0000 0002

Solution

Only bit 25 and bit 63 are 1s; the other bits are 0s. In the final permutation, bit 25 becomes bit 64 and bit 63 becomes bit 15. The result is

Chỉ bit thứ 25 và 63 là các bit 1, còn lại các bits khác là bit 0. Trong hoán vị kết thúc, biến đổi bit 25 thành bit 64 và bit 63 thành bit 15. Khi đó kết quả là:

0x0002 0000 0000 0001

0x0000 0080 0000 0002

0000 0000 0000 0000
0000 0000 1000 0000
0000 0000 0000 0000
0000 0000 0000 0010

6.2.1 *Continued*

Example 6.2

Prove that the initial and final permutations are the inverse of each other by finding the output of the initial permutation if the input is

0x0002 0000 0000 0001

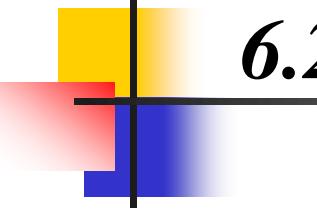
Solution

The input has only two 1s; the output must also have only two 1s. Using Table 6.1, we can find the output related to these two bits. Bit 15 in the input becomes bit 63 in the output. Bit 64 in the input becomes bit 25 in the output. So the output has only two 1s, bit 25 and bit 63. The result in hexadecimal is

0x0000 0080 0000 0002

0x0002 0000 0000 0001

0000 0000 0000 0010
0000 0000 0000 0000
0000 0000 0000 0000
0000 0000 0000 0001



6.2.1 Continued

Note

The initial and final permutations are straight P-boxes that are inverses of each other.

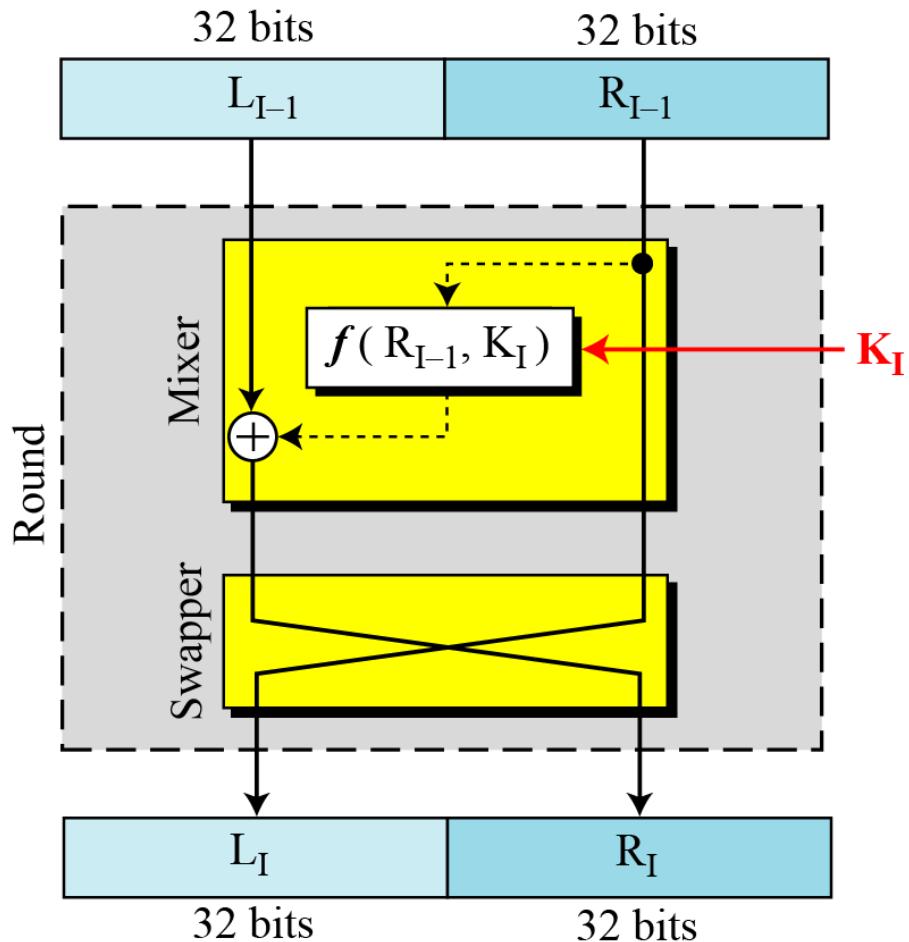
They have no cryptography significance in DES.

6.2.2 Rounds

DES uses 16 rounds. Each round of DES is a Feistel cipher.

DES sử dụng 16 vòng. Mỗi vòng trong DES là một mã Feistel.

Figure 6.4
*A round in DES
(encryption site)*



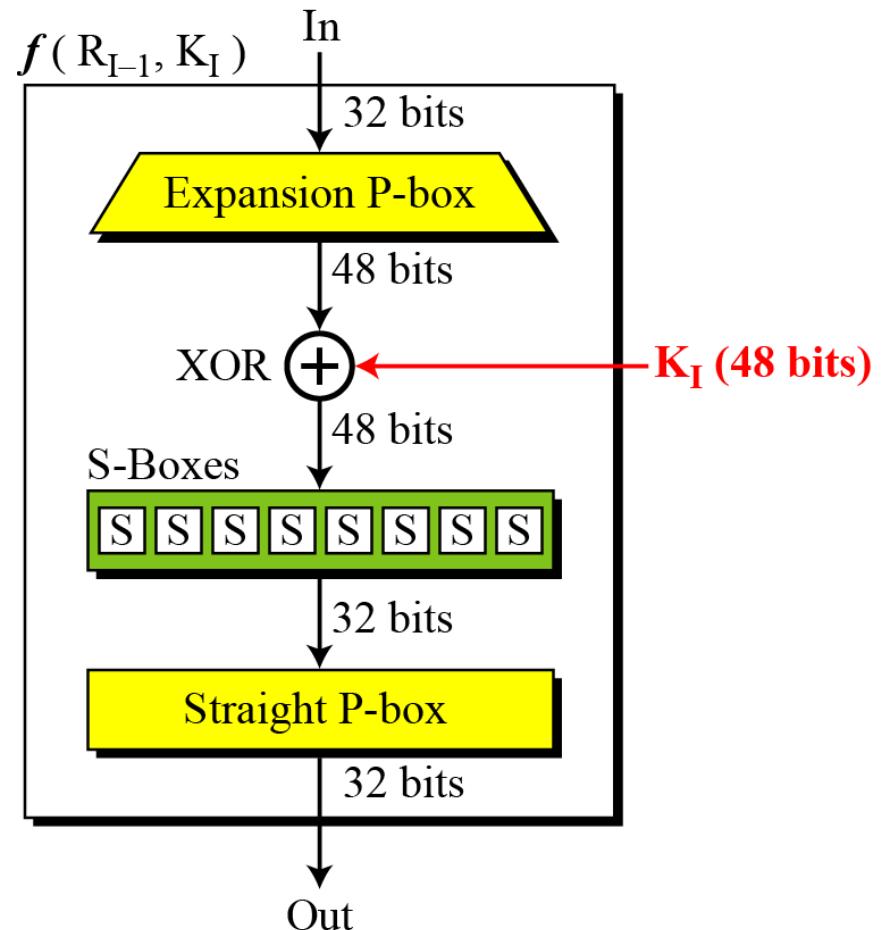
6.2.2 Continued

DES Function: Hàm DES

The heart of DES is the DES function. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.

Figure 6.5
DES function

Trái tim của DES là hàm DES. Hàm DES áp dụng khóa 48 bit cho 32 bit ngoài cùng bên phải để tạo ra đầu ra 32 bit.

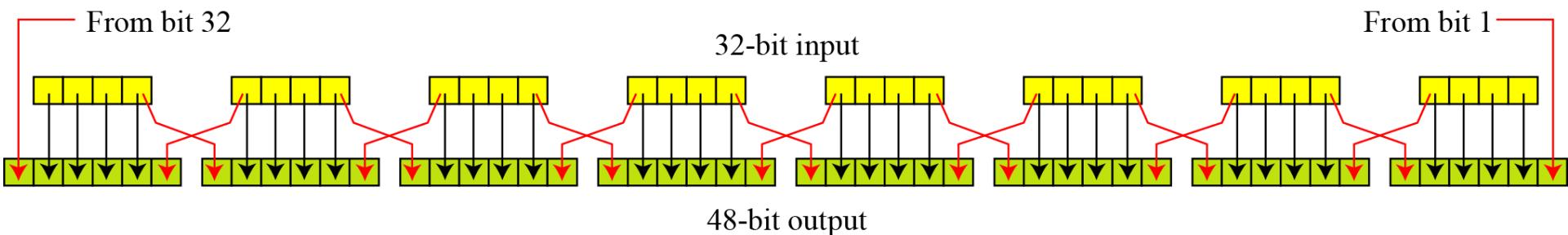


6.2.2 Continue

Expansion P-box: *Hộp P mở rộng*

Since R_{I-1} is a 32-bit input and K_I is a 48-bit key, we first need to expand R_{I-1} to 48 bits.

Figure 6.6 Expansion permutation



6.2.2 Continue

Although the relationship between the input and output can be defined mathematically, DES uses Table 6.2 to define this P-box.

Table 6.6 Expansion P-box table

32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

6.2.2 Continue

Whitener (XOR)

After the expansion permutation, DES uses the XOR operation on the expanded right section and the round key. Note that both the right section and the key are 48-bits in length. Also note that the round key is used only in this operation.

Sau khi mở rộng phép hoán vị, DES sử dụng phép XOR trong phần bên phải mở rộng và khóa vòng. Chú ý rằng phần bên phải và khóa vòng có độ dài 48bits. Khóa vòng chỉ được sử dụng trong phép toán này.

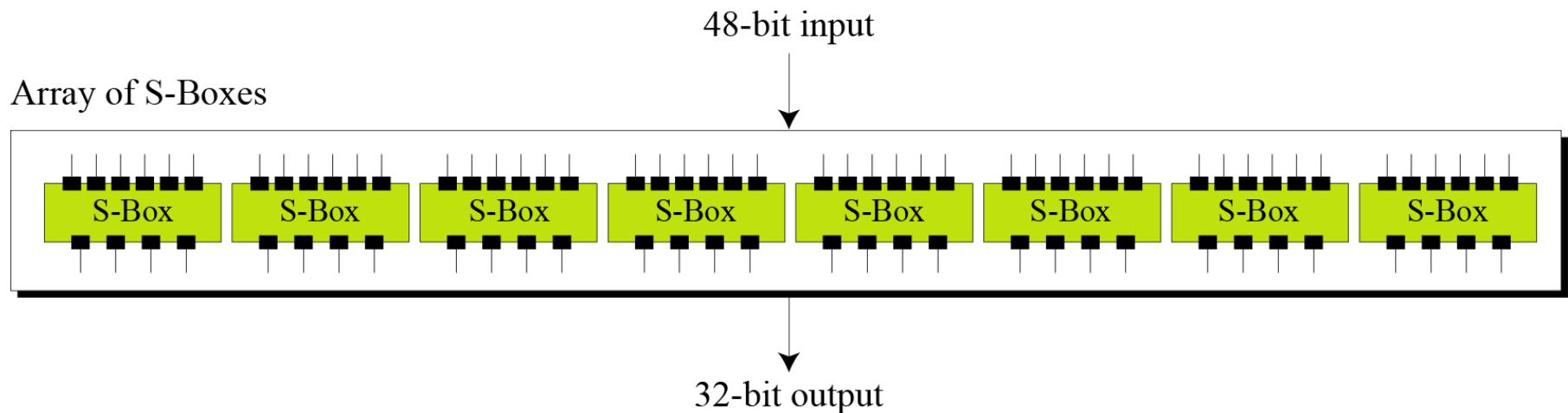
6.2.2 Continue

S-Boxes

The S-boxes do the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output. See Figure 6.7.

Các hộp S thực hiện sự trộn lẫn thực sự (nhầm lẫn). DES sử dụng 8 hộp S, mỗi hộp có đầu vào 6 bit và đầu ra 4 bit. Xem Hình 6.7.

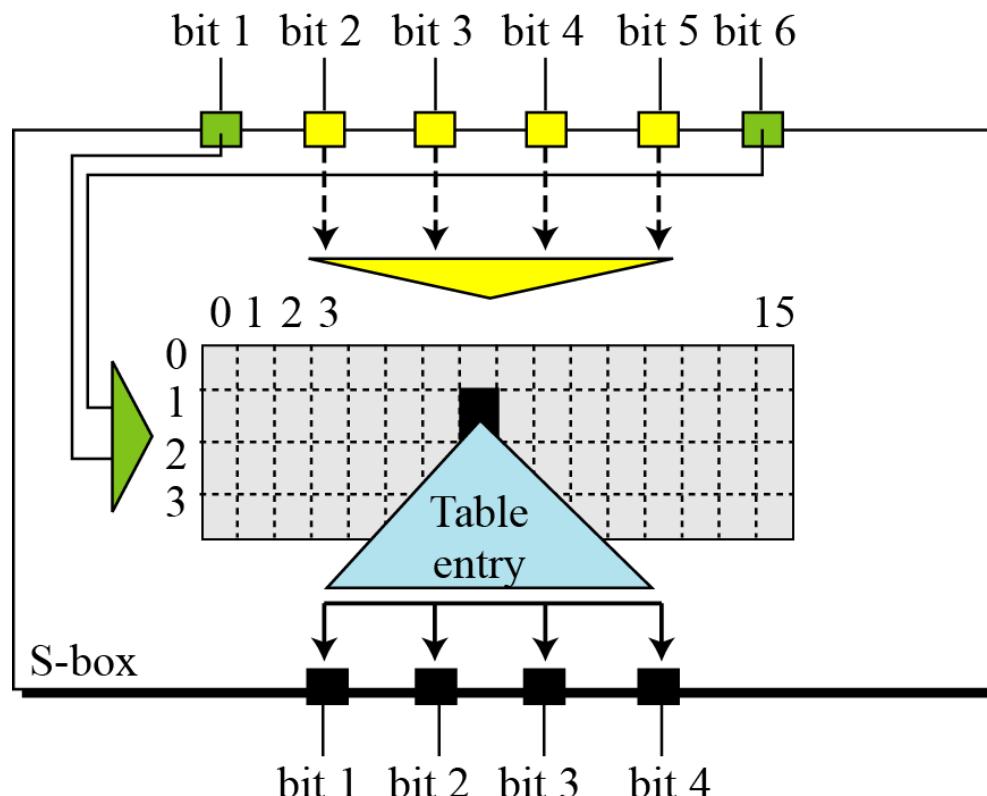
Figure 6.7 S-boxes



6.2.2 Continue

Figure 6.8 S-box rule

Quy tắc hộp chữ S



6.2.2 Continue

Table 6.3 shows the permutation for S-box 1. For the rest of the boxes see the textbook.

Dưới đây cho biết hoán vị của hộp S1. Đối với các hộp còn lại, xem tài liệu.

Table 6.3 S-box 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	10	03	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

6.2.2 Continue

S-box 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

6.2.2 Continue

S-box 3

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

S-box 4

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	07	13	14	03	00	6	09	10	1	02	08	05	11	12	04	15
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
2	10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

6.2.2 Continue

S-box 5

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

S-box 6

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
3	04	03	02	12	09	05	15	10	11	14	01	07	10	00	08	13

6.2.2 Continue

S-box 7

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	11	2	14	15	00	08	13	03	12	09	07	05	10	06	01
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

S-box 8

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
1	01	15	13	08	10	03	07	04	12	05	06	11	10	14	09	02
2	07	11	04	01	09	12	14	02	00	06	10	10	15	03	05	08
3	02	01	14	07	04	10	8	13	15	12	09	09	03	05	06	11

6.2.2 *Continued*

Example 6.3

The input to **S-box 1** is **100011**. What is the output?

Cho 8 bit đầu vào S-box 1 là 100011, tìm số bit đầu ra?

Solution

If we write the first and the sixth bits together, we get 11 in binary, which is 3 in decimal. The remaining bits are 0001 in binary, which is 1 in decimal. We look for the value in row 3, column 1, in Table 6.3 (S-box 1). The result is 12 in decimal, which in binary is 1100. So the input **100011** yields the output **1100**.

6.2.2 *Continued*

Example 6.4

The input to **S-box 8** is **000000**. What is the output?

Cho 8 bit đầu vào S-box 1 là **000000**, tìm số bit đầu ra?

Solution

If we write the first and the sixth bits together, we get **00** in binary, which is **0** in decimal. The remaining bits are **0000** in binary, which is **0** in decimal. We look for the value in row **0**, column **0**, in Table 6.10 (S-box 8). The result is **13** in decimal, which is **1101** in binary. So the input **000000** yields the output **1101**.

6.2.2 Continue

Straight Permutation (*Hoán vị thẳng*)

Table 6.11 *Straight permutation table*

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

6.2.3 Cipher and Reverse Cipher

Using mixers and swappers, we can create the cipher and reverse cipher, each having 16 rounds.

Sử dụng bộ trộn và trao đổi, có thể tạo ra mã và mã nghịch đảo 16 vòng

First Approach

To achieve this goal, one approach is to make the last round (round 16) different from the others; it has only a mixer and no swapper.

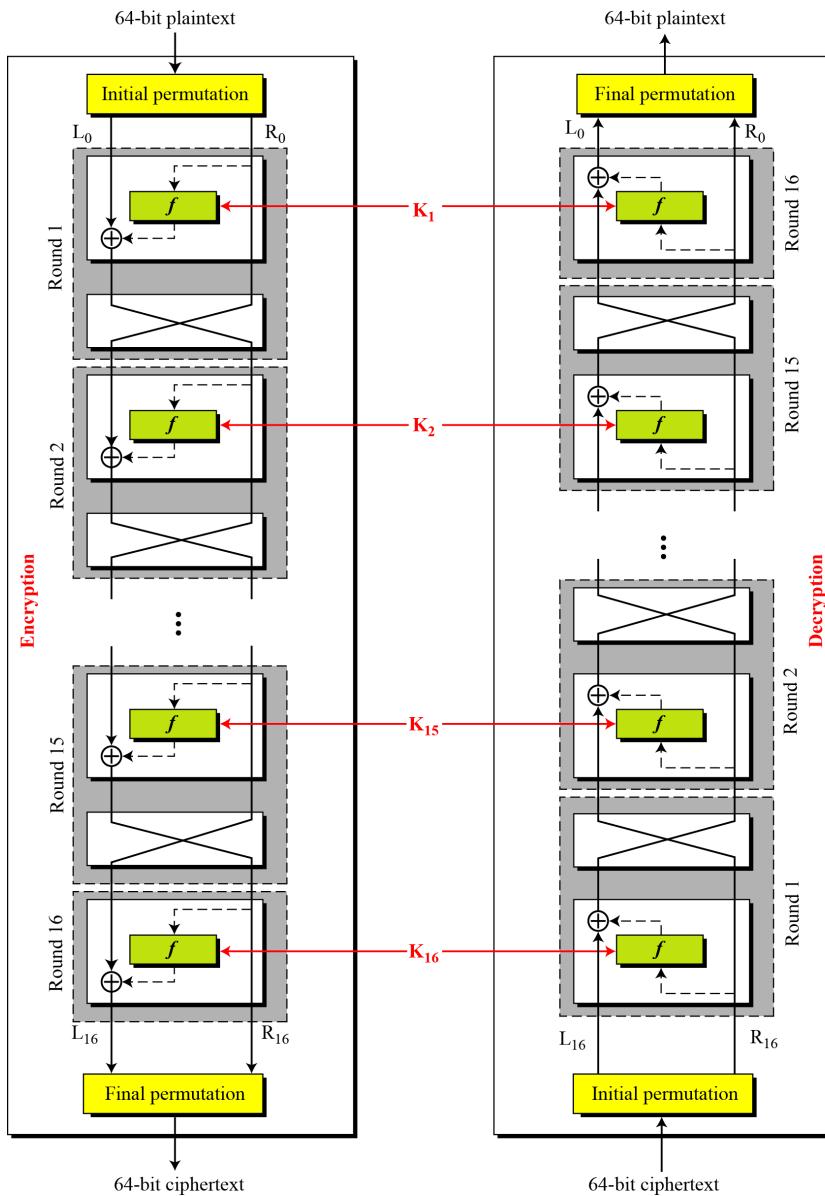
Vòng cuối cùng (16) khác với các vòng khác là chỉ có một bộ trộn và không có bộ trao đổi nào

Note

In the first approach, there is no swapper in the last round.

6.2.3 Continued

Figure 6.9 Mật mã DES và mật mã đảo ngược cho cách tiếp cận đầu tiên



6.2.3 *Continued*

Algorithm 6.1 *Pseudocode for DES cipher* *Mã giả cho DES*

```
Cipher (plainBlock[64], RoundKeys[16, 48], cipherBlock[64])
```

```
{
```

```
    permute (64, 64, plainBlock, inBlock, InitialPermutationTable)
```

```
    split (64, 32, inBlock, leftBlock, rightBlock)
```

```
    for (round = 1 to 16)
```

```
{
```

```
        mixer (leftBlock, rightBlock, RoundKeys[round])
```

```
        if (round!=16) swapper (leftBlock, rightBlock)
```

```
}
```

```
    combine (32, 64, leftBlock, rightBlock, outBlock)
```

```
    permute (64, 64, outBlock, cipherBlock, FinalPermutationTable)
```

```
}
```

6.2.3 *Continued*

Algorithm 6.1 Pseudocode for DES cipher (Continued) *Mã giả cho DES*

```
mixer (leftBlock[48], rightBlock[48], RoundKey[48])
```

```
{
```

```
    copy (32, rightBlock, T1)
```

```
    function (T1, RoundKey, T2)
```

```
        exclusiveOr (32, leftBlock, T2, T3)
```

```
        copy (32, T3, rightBlock)
```

```
}
```

```
swapper (leftBlock[32], rightBlock[32])
```

```
{
```

```
    copy (32, leftBlock, T)
```

```
    copy (32, rightBlock, leftBlock)
```

```
    copy (32, T, rightBlock)
```

```
}
```

6.2.3 *Continued*

Algorithm 6.1 *Pseudocode for DES cipher (Continued)* *Mã giả cho DES*

```
function (inBlock[32], RoundKey[48], outBlock[32])
{
    permute (32, 48, inBlock, T1, ExpansionPermutationTable)
    exclusiveOr (48, T1, RoundKey, T2)
    substitute (T2, T3, SubstituteTables)
    permute (32, 32, T3, outBlock, StraightPermutationTable)
}
```

6.2.3 Continued

Algorithm 6.1 Pseudocode for DES cipher (Continued) Mã giả cho DES

```
substitute (inBlock[32], outBlock[48], SubstitutionTables[8, 4, 16])
{
    for (i = 1 to 8)
    {
        row  $\leftarrow$  2  $\times$  inBlock[i  $\times$  6 + 1] + inBlock [i  $\times$  6 + 6]
        col  $\leftarrow$  8  $\times$  inBlock[i  $\times$  6 + 2] + 4  $\times$  inBlock[i  $\times$  6 + 3] +
            2  $\times$  inBlock[i  $\times$  6 + 4] + inBlock[i  $\times$  6 + 5]

        value = SubstitutionTables [i][row][col]

        outBlock[[i  $\times$  4 + 1]  $\leftarrow$  value / 8;           value  $\leftarrow$  value mod 8
        outBlock[[i  $\times$  4 + 2]  $\leftarrow$  value / 4;           value  $\leftarrow$  value mod 4
        outBlock[[i  $\times$  4 + 3]  $\leftarrow$  value / 2;           value  $\leftarrow$  value mod 2
        outBlock[[i  $\times$  4 + 4]  $\leftarrow$  value
    }
}
```

6.2.3 Continued

Alternative Approach: Phương pháp khác

We can make all 16 rounds the same by including one swapper to the 16th round and add an extra swapper after that (two swappers cancel the effect of each other).

Có thể tạo tất cả 16 vòng giống nhau bằng cách gộp một bộ trao đổi vào vòng thứ 16 và sau đó cộng thêm một bộ trao đổi (hai bộ trao đổi loại bỏ hiệu ứng lẫn nhau).

Key Generation: Tạo khóa

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.

Bộ tạo khóa vòng tạo ra mười sáu khóa 48-bit trong bộ khóa mã 56-bit

6.2.3 Continued

*Bộ tạo
khóa vòng
tạo ra
mười sáu
khóa 48-
bit trong
bộ khóa
mã 56-bit*

Shifting	
Rounds	Shift
1, 2, 9, 16	one bit
Others	two bits

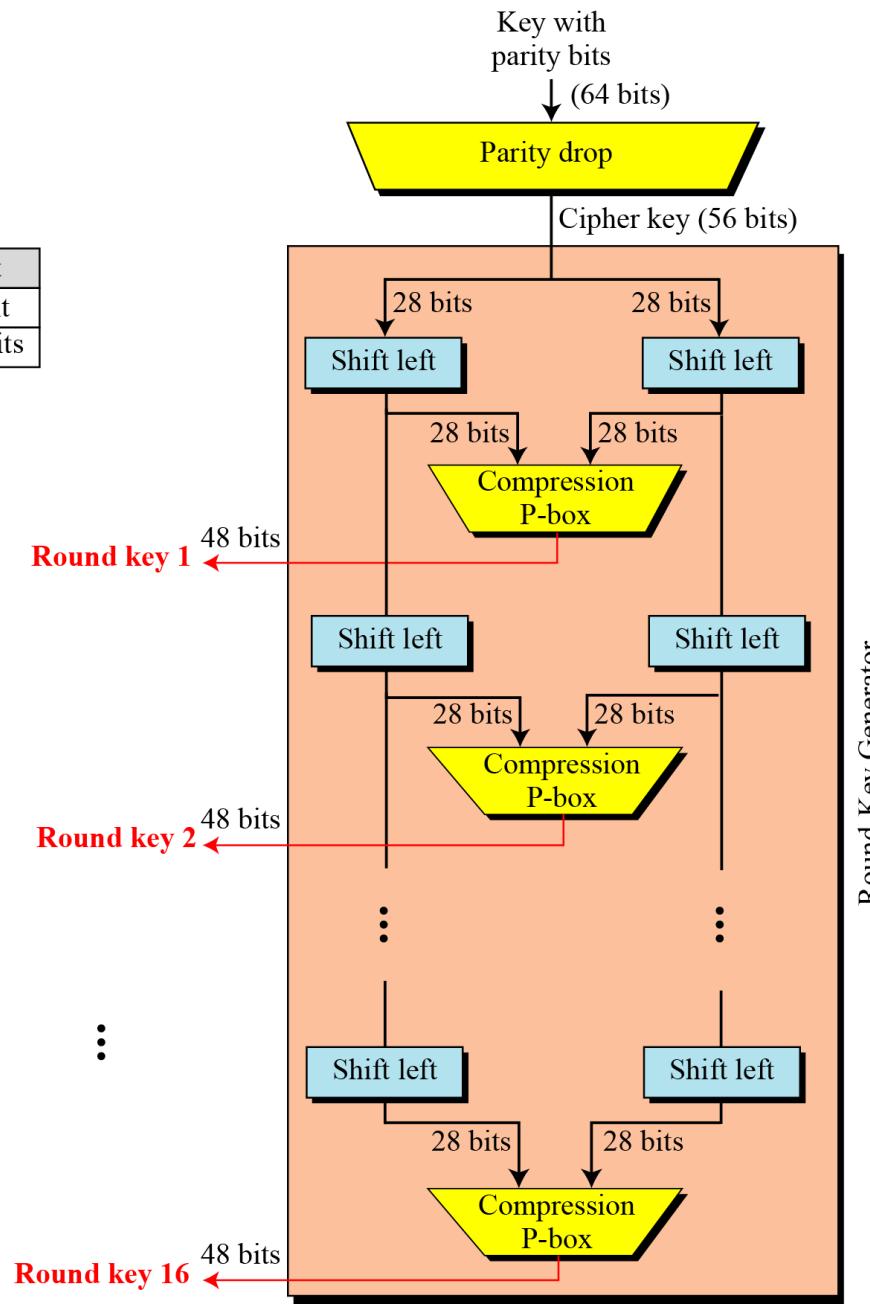


Figure 6.10
*Key generation
Tạo khóa*

Round-Key Generator

6.2.3 Continued

Table 6.12 Parity-bit drop table

+ Bỏ các bit
số 8, 16, 24,
32, 40, 48,
56, và 64 từ
64 bit khóa.
+ Sau đó
thực hiện
phép hoán
vị các bit
còn lại theo
bảng bên để
thực hiện
quá trình
tạo khóa
cho 16 vòng

Bảng thả bit chẵn lẻ

57	49	41	33	25	17	09	01
58	50	42	34	26	18	10	02
59	51	43	35	27	19	11	03
60	52	44	36	63	55	47	39
31	23	15	07	62	54	46	38
30	22	14	06	61	53	45	37
29	21	13	05	28	20	12	04

Table 6.13 Number of bits shifts

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

6.2.3 *Continued*

Dịch trái:

Chia 56 bit thành hai phần 28 bit. Mỗi phần sẽ được dịch vòng trái một hoặc 2 bit: vòng 1, 2, 9, 16 sẽ dịch 1 bit, còn lại dịch 2 bit.

Hai phần 28 bit sau khi dịch trái sẽ được đưa vào hộp P nép để tạo khóa 48 bit.

Table 6.13 *Number of bits shifts*

Shifting	
Rounds	Shift
1, 2, 9, 16	one bit
Others	two bits

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

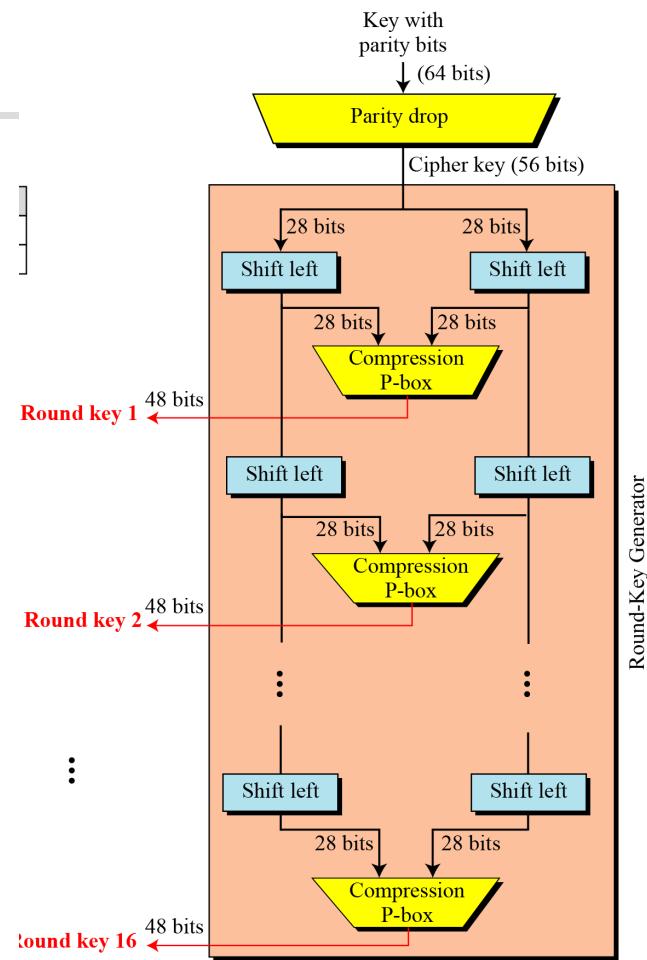
6.2.3 Continued

Hộp P nén: thực hiện biến đổi 56 bits vào thành 48bits ra, được sử dụng cho khóa vòng đó.

Table 6.14 Key-compression table

Bảng nén khóa

14	17	11	24	01	05	03	28
15	06	21	10	23	19	12	04
26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32



Bài tập về tìm khóa vòng Kn

- Cho khóa 64 bits vào K: **0123 ABCD 2562 1456**, tìm khóa vòng đầu tiên $K_1=?$

6.2.3 *Continued*

Algorithm 6.2 *Algorithm for round-key generation*

Thuật toán tạo khóa vòng

```
Key_Generator (keyWithParities[64], RoundKeys[16, 48], ShiftTable[16])
{
    permute (64, 56, keyWithParities, cipherKey, ParityDropTable)
    split (56, 28, cipherKey, leftKey, rightKey)
    for (round = 1 to 16)
    {
        shiftLeft (leftKey, ShiftTable[round])
        shiftLeft (rightKey, ShiftTable[round])
        combine (28, 56, leftKey, rightKey, preRoundKey)
        permute (56, 48, preRoundKey, RoundKeys[round], KeyCompressionTable)
    }
}
```

6.2.3 *Continued*

Algorithm 6.2 Algorithm for round-key generation (Continue) Thuật toán tạo khóa vòng

```
shiftLeft (block[28], numOfShifts)
{
    for (i = 1 to numOfShifts)
    {
        T ← block[1]
        for (j = 2 to 28)
        {
            block [j-1] ← block [j]
        }
        block[28] ← T
    }
}
```

6.2.4 Examples

Example 6.5

We choose a random plaintext block and a random key, and determine what the ciphertext block would be (all in hexadecimal):

Chọn một khối bản rõ ngẫu nhiên và một khóa ngẫu nhiên, và xác định khối bản mã sẽ là gì (tất cả đều ở dạng thập lục phân):

Plaintext: 123456ABCD132536

Key: AABB09182736CCDD

CipherText: C0B7A8D05F3A829C

Table 6.15 Theo dõi dữ liệu từ Ví dụ 6.5

Plaintext: 123456ABCD132536			
After initial permutation: 14A7D67818CA18AD			
After splitting: $L_0 = 14A7D678$ $R_0 = 18CA18AD$			
Round	Left	Right	Round Key
Round 1	18CA18AD	5A78E394	194CD072DE8C
Round 2	5A78E394	4A1210F6	4568581ABCCE
Round 3	4A1210F6	B8089591	06EDA4ACF5B5
Round 4	B8089591	236779C2	DA2D032B6EE3

6.2.4 Continued

Example 6.5 Continued

Table 6.15 Trace of data for Example 6.5 (Conintued)

Round 5	236779C2	A15A4B87	69A629FEC913
Round 6	A15A4B87	2E8F9C65	C1948E87475E
Round 7	2E8F9C65	A9FC20A3	708AD2DDB3C0
Round 8	A9FC20A3	308BEE97	34F822F0C66D
Round 9	308BEE97	10AF9D37	84BB4473DCCC
Round 10	10AF9D37	6CA6CB20	02765708B5BF
Round 11	6CA6CB20	FF3C485F	6D5560AF7CA5
Round 12	FF3C485F	22A5963B	C2C1E96A4BF3
Round 13	22A5963B	387CCDAA	99C31397C91F
Round 14	387CCDAA	BD2DD2AB	251B8BC717D0
Round 15	BD2DD2AB	CF26B472	3330C5D9A36D
Round 16	19BA9212	CF26B472	181C5D75C66D
After combination: 19BA9212CF26B472			
Ciphertext: C0B7A8D05F3A829C			(after final permutation)

6.2.4 Continued

Example 6.6

Let us see how Bob, at the destination, can decipher the ciphertext received from Alice using the same key. Table 6.16 shows some interesting points.

Hãy để chúng ta xem làm thế nào Bob, tại điểm đến, có thể giải mã bản mã nhận được từ Alice bằng cách sử dụng cùng một khóa. Bảng 6.16 cho thấy một số điểm thú vị.

Ciphertext: C0B7A8D05F3A829C			
After initial permutation: 19BA9212CF26B472			
After splitting: L ₀ =19BA9212 R ₀ =CF26B472			
Round	Left	Right	Round Key
Round 1	CF26B472	BD2DD2AB	181C5D75C66D
Round 2	BD2DD2AB	387CCDAA	3330C5D9A36D
...
Round 15	5A78E394	18CA18AD	4568581ABCCE
Round 16	14A7D678	18CA18AD	194CD072DE8C
After combination: 14A7D67818CA18AD			
Plaintext: 123456ABCD132536		(after final permutation)	

6-3 DES ANALYSIS

*Critics have used a strong magnifier to analyze DES.
Tests have been done to measure the strength of some
desired properties in a block cipher.*

Các nhà phê bình đã sử dụng một kính lúp mạnh để phân tích DES. Các thử nghiệm đã được thực hiện để đo độ mạnh của một số thuộc tính mong muốn trong mật mã khối.

Topics discussed in this section:

- 6.3.1 Properties: các thuộc tính của DES
- 6.3.2 Design Criteria: Tiêu chí đánh giá
- 6.3.3 DES Weaknesses: Điểm yếu của DES

6.3.1 Properties

Two desired properties of a block cipher are the avalanche effect and the completeness.

Hai thuộc tính mong muốn của mật mã khối là hiệu ứng thác lũ và tính đầy đủ.

Example 6.7

Để kiểm tra hiệu ứng thác lũ trong DES, chúng ta hãy mã hóa hai khối văn bản rõ (với cùng một khóa) chỉ khác nhau một bit và quan sát sự khác biệt về số lượng bit trong mỗi vòng.

Plaintext: 0000000000000000

Key: 22234512987ABB23

Ciphertext: 4789FD476E82A5F1

Plaintext: 0000000000000000000000000000

Key: 22234512987ABB23

Ciphertext: 0A4ED5C15A63FEA3

6.3.1 *Continued*

Example 6.7 *Continued*

Although the two plaintext blocks differ only in the rightmost bit, the ciphertext blocks differ in 29 bits. This means that changing approximately 1.5 percent of the plaintext creates a change of approximately 45 percent in the ciphertext.

Mặc dù hai khối bản rõ chỉ khác nhau ở bit ngoài cùng bên phải, các khối bản mã khác nhau **29 bit**. Điều này có nghĩa là việc thay đổi khoảng 1,5% văn bản rõ ràng sẽ tạo ra thay đổi khoảng 45 phần trăm trong bản mã.

Table 6.17 Số lượng bit khác biệt cho Ví dụ 6.7

Rounds	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit differences	1	6	20	29	30	33	32	29	32	39	33	28	30	31	30	29

6.3.1 Continued

Completeness effect

Completeness effect means that each bit of the ciphertext needs to depend on many bits on the plaintext.

Hiệu ứng hoàn thiện:

Hiệu ứng tính đầy đủ có nghĩa là mỗi bit của bản mã cần phụ thuộc vào nhiều bit trên bản rõ.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

S-Boxe

*The design provides **confusion and diffusion** of bits from each round to the next.*

Thiết kế cung cấp sự nhầm lẫn và khuếch tán các bit từ mỗi vòng sang vòng tiếp theo.

P-Boxes

They provide diffusion of bits.

Chúng cung cấp sự khuếch tán của các bit.

Number of Rounds

DES uses sixteen rounds of Feistel ciphers. the ciphertext is thoroughly a random function of plaintext and ciphertext.

DES sử dụng mười sáu vòng mã Feistel. bản mã hoàn toàn là một hàm ngẫu nhiên của bản rõ và bản mã.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

S-Boxe: Thiết kế cung cấp sự nhầm lẫn và khuếch tán các bit từ mỗi vòng sang vòng tiếp theo. Cụ thể:

1. Các giá trị của mỗi hàng được hoán vị từ 0 đến 15
2. Hộp S là phi tuyến
3. Nếu ta thay đổi một 1 bit vào, khi đó 2 hoặc nhiều hơn các bit ra bị thay đổi
4. Nếu 2 bit đầu vào hộp S khác hai bit nào đó ở giữa, thì đầu ra của hộp S sẽ khác ít nhất 2 bit.
5. Trong mỗi hộp S, nếu một bit đơn cố định (0, hoặc 1) còn các bit khác thay đổi ngẫu nhiên, sự khác nhau giữa các bit 0 và 1 là nhỏ nhất.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

S-boxes At least three weaknesses are mentioned in the literature for S-boxes.

1. In S-box 4, the last three output bits can be derived in the same way as the first output bit by complementing some of the input bits.
2. Two specifically chosen inputs to an S-box array can create the same output.
3. It is possible to obtain the same output in a single round by changing bits in only three neighboring S-boxes.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

P-Boxes: They provide diffusion of bits.

Chúng cung cấp sự khuếch tán của các bit.

Cụ thể: DES có 1 hộp P thẳng ($32 \rightarrow 32$) và 1 hộp P mở rộng ($32 \rightarrow 48$). Khi đó hai hộp P này cho phép khuếch tán các bit với nhau.

P-boxes One mystery and one weakness were found in the design of P-boxes:

1. It is not clear why the designers of DES used the initial and final permutations; these have no security benefits.
2. In the expansion permutation (inside the function), the first and fourth bits of every 4-bit series are repeated.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

Number of Rounds

DES uses sixteen rounds of Feistel ciphers. the ciphertext is thoroughly a random function of plaintext and ciphertext.

DES sử dụng mười sáu vòng mật mã Feistel. bản mã hoàn toàn là một hàm ngẫu nhiên của bản rõ và bản mã.

Tại sao DES lại sử dụng 16 vòng mà không phải là ít hơn hoặc hơn số đó?

→ phụ thuộc vào mức độ an toàn, khả năng chống lại các cuộc tấn công.

6.3.2 Design Criteria: Tiêu chuẩn thiết kế

Key Size Critics believe that the most serious weakness of DES is in its key size (56 bits). To do a brute-force attack on a given ciphertext block, the adversary needs to check 2^{56} keys.

- a. With available technology, it is possible to check one million keys per second. This means that we need more than two thousand years to do brute-force attacks on DES using only a computer with one processor.
- b. If we can make a computer with one million chips (parallel processing), then we can test the whole key domain in approximately 20 hours. When DES was introduced, the cost of such a computer was over several million dollars, but the cost has dropped rapidly. A special computer was built in 1998 that found the key in 112 hours.
- c. Computer networks can simulate parallel processing. In 1977 a team of researchers used 3500 computers attached to the Internet to find a key challenged by RSA Laboratories in 120 days. The key domain was divided among all of these computers, and each computer was responsible to check the part of the domain.
- d. If 3500 networked computers can find the key in 120 days, a secret society with 42,000 members can find the key in 10 days.

6.3.3 DES Weaknesses

Điểm yếu trong thiết kế mật mã: phụ thuộc chính vào việc thiết kế hệ mật này, cụ thể:

1. *Weaknesses in S-boxes: Hộp S4, đầu ra các hộp S có thể giống nhau*
2. *Weaknesses in P-boxes: Không rõ ràng việc sử dụng hộp P khởi tạo và kết thúc. Nó không đủ khả năng bảo mật.*
3. *Weaknesses in Key: Có nhiều điểm yếu khi mỗi vòng dùng khóa 56 bit → kiểm tra số lượng khóa 2^{56} .*

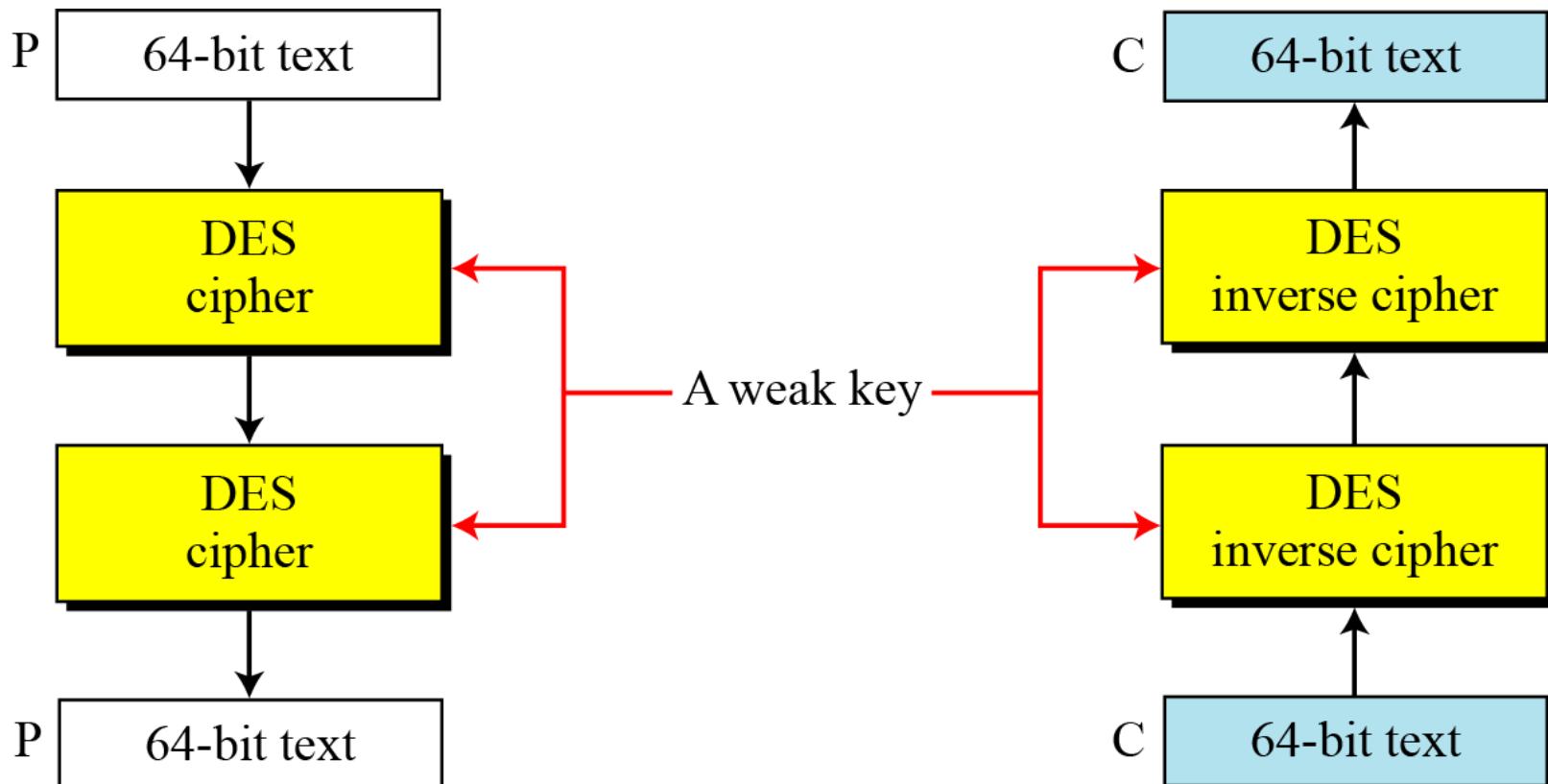
Trong 2^{56} trường hợp khóa K có 4 khóa có độ an toàn rất kém đó là các khóa toàn 0 hoặc 1

Table 6.18 Weak keys

Keys before parities drop (64 bits)	Actual key (56 bits)
0101 0101 0101 0101	0000000 0000000
1F1F 1F1F 0E0E 0E0E	0000000 FFFFFFFF
E0E0 E0E0 F1F1 F1F1	FFFFFFF 0000000
FEFE FEFE FEFE FEFE	FFFFFFF FFFFFFFF

6.3.3 Continued

Figure 6.11 *Mã hóa kép và giải mã bằng khóa yếu*



Khóa yếu là khóa khi đưa vào mã hóa bản tin gốc P hai lần thì lại được bản tin P, tức là $E_k(E_k(P))=P$.

6.3.3 *Continued*

Example 6.8

Khóa yếu

Chúng ta hãy thử khóa yếu đầu tiên trong Bảng 6.18 để mã hóa một khối hai lần. Sau hai lần mã hóa với cùng một khóa, khối văn bản rõ ban đầu được tạo. Lưu ý rằng chúng tôi đã sử dụng thuật toán mã hóa hai lần, không phải một lần mã hóa tiếp theo là một lần giải mã khác.

Key: 0x0101010101010101

Plaintext: 0x1234567887654321

Ciphertext: 0x814FE938589154F7

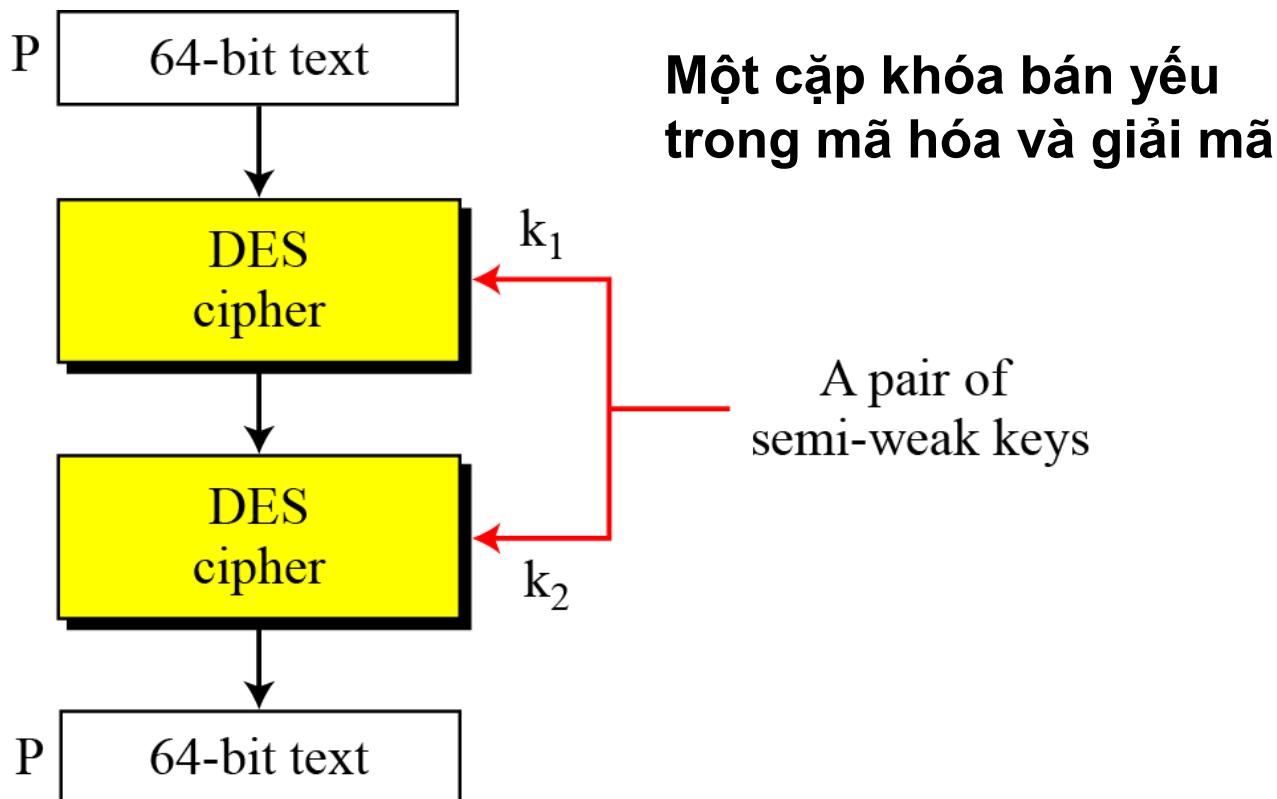
Key: 0x0101010101010101

Plaintext: 0x814FE938589154F7

Ciphertext: 0x1234567887654321

6.3.3 Continued

Figure 6.12 A pair of semi-weak keys in encryption and decryption



6.3.3 *Continued*

Table 6.19 *Semi-weak keys*

<i>First key in the pair</i>	<i>Second key in the pair</i>
01FE 01FE 01FE 01FE	FE01 FE01 FE01 FE01
1FE0 1FE0 0EF1 0EF1	E01F E01F F10E F10E
01E0 01E1 01F1 01F1	E001 E001 F101 F101
1FFE 1FFE 0EFE 0EFE	FE1F FE1F FE0E FE0E
011F 011F 010E 010E	1F01 1F01 0E01 0E01
E0FE E0FE F1FE F1FE	FEE0 FEE0 FEF1 FEF1

k1

k2

6.3.3 *Continued*

<i>Round key 1</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 2</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 3</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 4</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 5</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 6</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 7</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 8</i>	6EAC1ABCE642	9153E54319BD
<i>Round key 9</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 10</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 11</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 12</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 13</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 14</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 15</i>	9153E54319BD	6EAC1ABCE642
<i>Round key 16</i>	6EAC1ABCE642	9153E54319BD

6.3.3 *Continued*

Example 6.9

What is the probability of randomly selecting a weak, a semi-weak, or a possible weak key?

Xác suất để chọn ngẫu nhiên một khóa yếu, một nửa yếu, hoặc một khóa có thể yếu là bao nhiêu?

Solution

DES has a key domain of 2^{56} . The total number of the above keys are 64 ($4 + 12 + 48$). The probability of choosing one of these keys is 8.8×10^{-16} , almost impossible.

DES có miền khóa là 2^{56} . Tổng số khóa trên là 64 ($4 + 12 + 48$). Xác suất chọn một trong các khóa này là $8,8 \times 10^{-16}$, gần như là không thể.

6.3.3 *Continued*

Key Complement In the key domain (2^{56}), definitely half of the keys are *complement* of the other half. A **key complement** can be made by inverting (changing 0 to 1 or 1 to 0) each bit in the key. Does a key complement simplify the job of the cryptanalysis? It happens that it does. Eve can use only half of the possible keys (2^{55}) to perform brute-force attack. This is because

$$C = E(K, P) \rightarrow \bar{C} = E(\bar{K}, \bar{P})$$

In other words, if we encrypt the complement of plaintext with the complement of the key, we get the complement of the ciphertext. Eve does not have to test all 2^{56} possible keys, she can test only half of them and then complement the result.

6.3.3 *Continued*

Example 6.10

Hãy để chúng tôi kiểm tra xác nhận quyền sở hữu về các khóa bổ sung. Chúng tôi đã sử dụng một khóa và bản rõ tùy ý để tìm bản mã tương ứng. Nếu chúng ta có phần bổ sung khóa và bản rõ, chúng ta có thể nhận được phần bổ sung của bản mã trước đó (Bảng 6.20).

Table 6.20 Results for Example 6.10

	<i>Original</i>	<i>Complement</i>
Key	1234123412341234	EDCBEDCBEDCBEDCB
Plaintext	12345678ABCDEF12	EDCBA987543210ED
Ciphertext	E112BE1DEFC7A367	1EED41E210385C98

6-4 Multiple DES

The major criticism of DES regards its key length. Fortunately DES is not a group. This means that we can use double or triple DES to increase the key size.

Lời chỉ trích chính của DES liên quan đến độ dài khóa của nó. May mắn thay, DES không phải là một nhóm. Điều này có nghĩa là chúng ta có thể sử dụng DES gấp đôi hoặc gấp ba để tăng kích thước khóa.

Topics discussed in this section:

6.4.1 Double DES: DES đôi

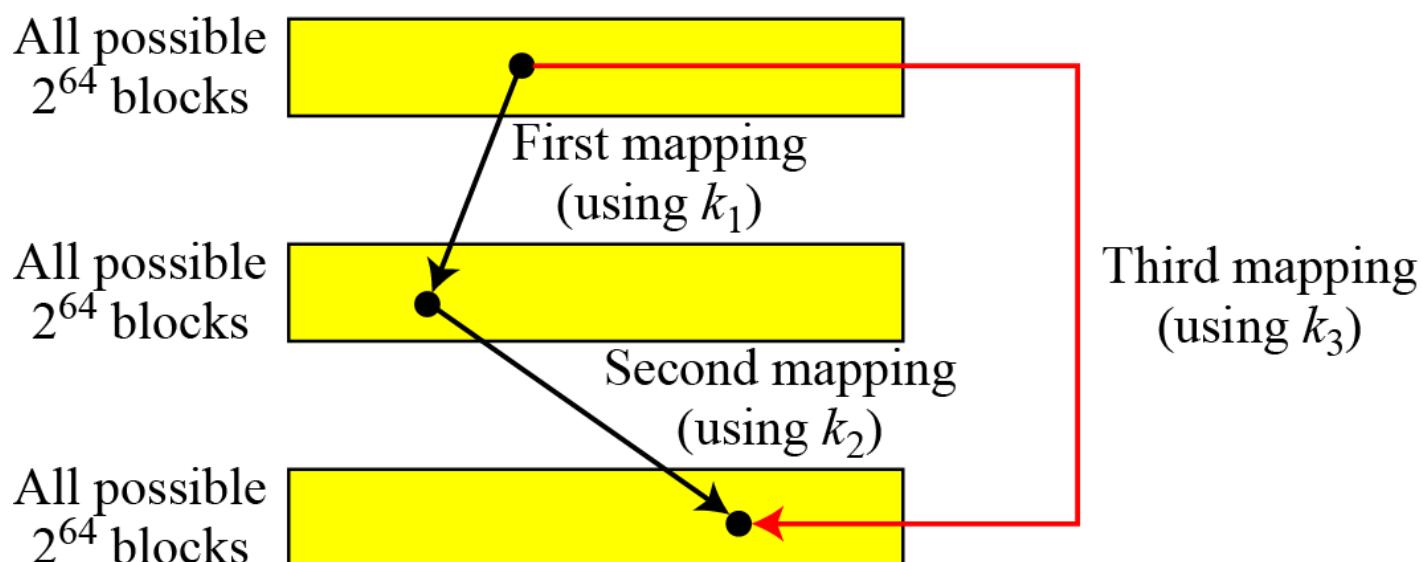
6.4.4 Triple DES: DES ba

6-4 Continued

A substitution that maps every possible input to every possible output is a group.

Sự thay thế ánh xạ mọi đầu vào có thể thành mọi đầu ra có thể là một nhóm.

Figure 6.13 Composition of mapping



6.4.1 Double DES

The first approach is to use double DES (2DES).

Cách tiếp cận đầu tiên là sử dụng DES kép (2DES).

Sử dụng mã hóa DES hai lần bằng hai khóa k1 và k2 khác nhau, khi đó kích thước khóa là 112 bits.

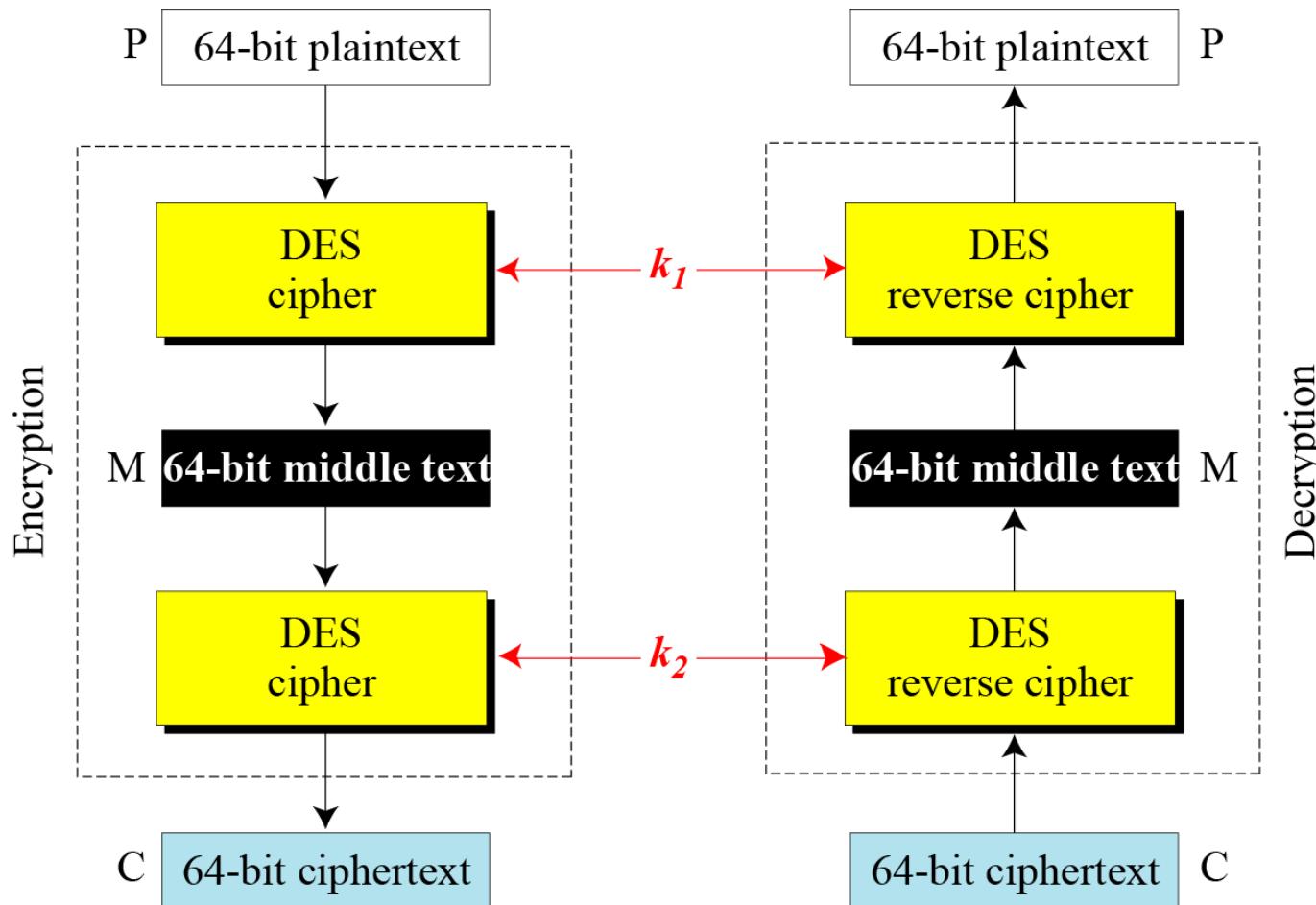
Meet-in-the-Middle Attack

*However, using a known-plaintext attack called **meet-in-the-middle attack**, that proves that double DES improves this vulnerability slightly (to 2^{57} tests), but not tremendously (to 2^{112}).*

Tuy nhiên, việc sử dụng một cuộc tấn công bằng văn bản rõ được gọi là cuộc tấn công gấp mặt ở giữa, nó chứng minh rằng DES kép cải thiện lỗ hổng này một chút (đến 2^{57} lần kiểm tra), nhưng không nhiều (đến 2^{112}).

6.4.1 Continued

Figure 6.14 Meet-in-the-middle attack for double DES



6.4.1 Continued

Figure 6.15 Tables for meet-in-the-middle attack

$$M = E_{k_1}(P)$$

M	k_1
●	

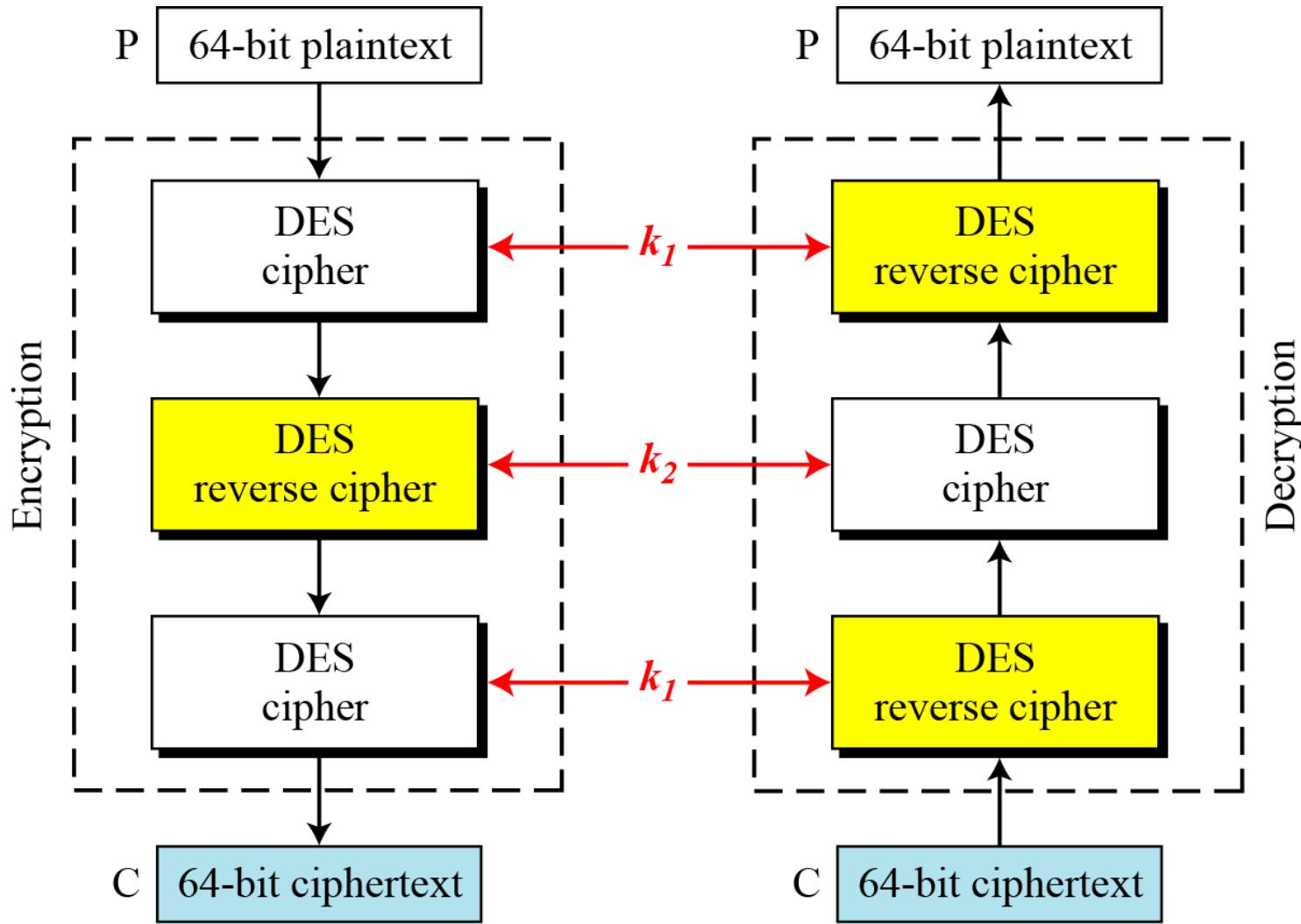
$$M = D_{k_2}(C)$$

M	k_2
●	

Find equal M's and record
corresponding k_1 and k_2

6.4.2 *Triple DES*

Figure 6.16 *Triple DES with two keys*



6.4.2 Continuous

Triple DES with Three Keys

The possibility of known-plaintext attacks on triple DES with two keys has enticed some applications to use triple DES with three keys. Triple DES with three keys is used by many applications such as PGP (See Chapter 16).

Khả năng xảy ra các cuộc tấn công bằng bản rõ đã biết vào DES ba với hai khóa đã lôi kéo một số ứng dụng sử dụng DES ba với ba khóa. Triple DES với ba khóa được sử dụng bởi nhiều ứng dụng như PGP (Xem Chương 16).

6-5 Security of DES

DES, as the first important block cipher, has gone through much scrutiny. Among the attempted attacks, three are of interest: brute-force, differential cryptanalysis, and linear cryptanalysis.

DES, với tư cách là mật mã khối quan trọng đầu tiên, đã trải qua nhiều nghiên cứu. Trong số các cuộc tấn công đã có gắng, ba cuộc tấn công được quan tâm: brute-force, phân tích mật mã vi sai và phân tích mật mã tuyến tính.

Topics discussed in this section:

6.5.1 Brute-Force Attack

6.5.2 Differential Cryptanalysis

6.5.3 Linear Cryptanalysis

6.5.1 Brute-Force Attack

We have discussed the weakness of short cipher key in DES. Combining this weakness with the key complement weakness, it is clear that DES can be broken using 2^{55} encryptions.

Chúng ta đã thảo luận về điểm yếu của khóa mật mã ngắn trong DES. Kết hợp điểm yếu này với điểm yếu bổ sung chính, rõ ràng là DES có thể bị phá vỡ bằng cách sử dụng 2^{55} mã hóa.

6.5.2 Differential Cryptanalysis

It has been revealed that the designers of DES already knew about this type of attack and designed S-boxes and chose 16 as the number of rounds to make DES specifically resistant to this type of attack.

Nó được tiết lộ rằng các nhà thiết kế của DES đã biết về kiểu tấn công này và đã thiết kế các hộp S và chọn 16 là số vòng để làm cho DES đặc biệt chống lại kiểu tấn công này.

Note

We show an example of DES differential cryptanalysis in Appendix N.

6.5.3 Linear Cryptanalysis

Linear cryptanalysis is newer than differential cryptanalysis. DES is more vulnerable to linear cryptanalysis than to differential cryptanalysis. S-boxes are not very resistant to linear cryptanalysis. It has been shown that DES can be broken using 2^{43} pairs of known plaintexts. However, from the practical point of view, finding so many pairs is very unlikely.

Note

We show an example of DES linear cryptanalysis in Appendix N.

6.5.3 Linear Cryptanalysis

Phương pháp phân tích mật mã tuyến tính mới hơn phương pháp phân tích mật mã vi sai. DES dễ bị phá mã tuyến tính hơn là phân tích mã vi sai. Hộp chữ S không chống lại được quá trình phá mã tuyến tính. Người ta đã chỉ ra rằng DES có thể bị phá vỡ bằng cách sử dụng 2^{43} cặp bản rõ đã biết. Tuy nhiên, trên quan điểm thực tế, việc tìm ra nhiều cặp số như vậy là rất khó xảy ra.

Note

Chúng tôi đưa ra một ví dụ về phân tích mật mã tuyến tính DES trong Phụ lục N.