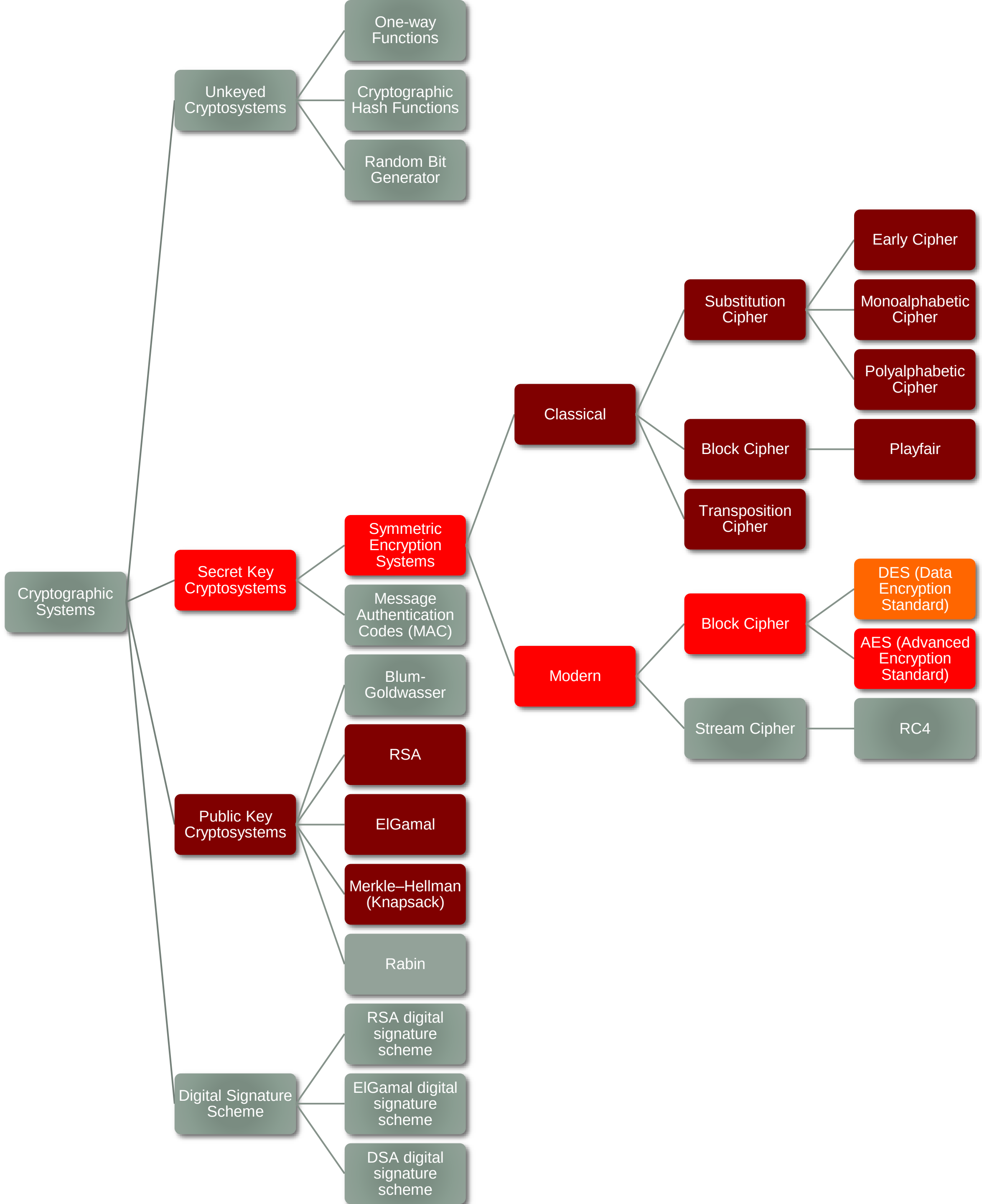
TUTORIAL

CSCI361 – Computer Security

Sionggo Japit

sjapit@uow.edu.au

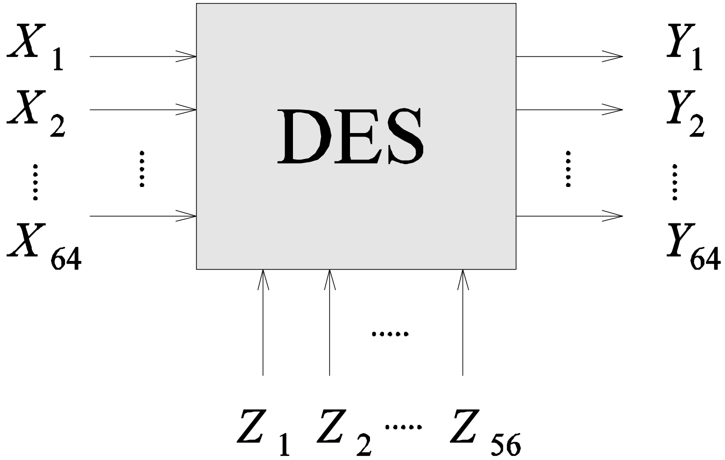
12 February 2024



Cryptographic Systems
Unkeyed Cryptosystems
One-way Functions
Cryptographic Hash Functions
Random Bit Generator
Secret Key Cryptosystems
Symmetric Encryption Systems
Message Authentication Codes (MAC)
Classical
Substitution Cipher
Early Cipher
Monoalphabetic Cipher
Polyalphabetic Cipher
Block Cipher
Playfair
Transposition Cipher
Modern
Block Cipher
DES (Data Encryption Standard)
AES (Advanced Encryption Standard)
Stream Cipher
RC4
Public Key Cryptosystems
Blum-Goldwasser
RSA
ElGamal
Merkle–Hellman (Knapsack)
Rabin
Digital Signature Scheme
RSA digital signature scheme
ElGamal digital signature scheme
DSA digital signature scheme

DES (DATA ENCRYPTION STANDARD)

DES



DES encrypts a plaintext bitstring *x* of length 64 using a key *K* which is a bitstring of length 56, obtaining a ciphertext bitstring which is again a bitstring of length 64.

DES

Encryption:

The algorithm proceeds in three stages:

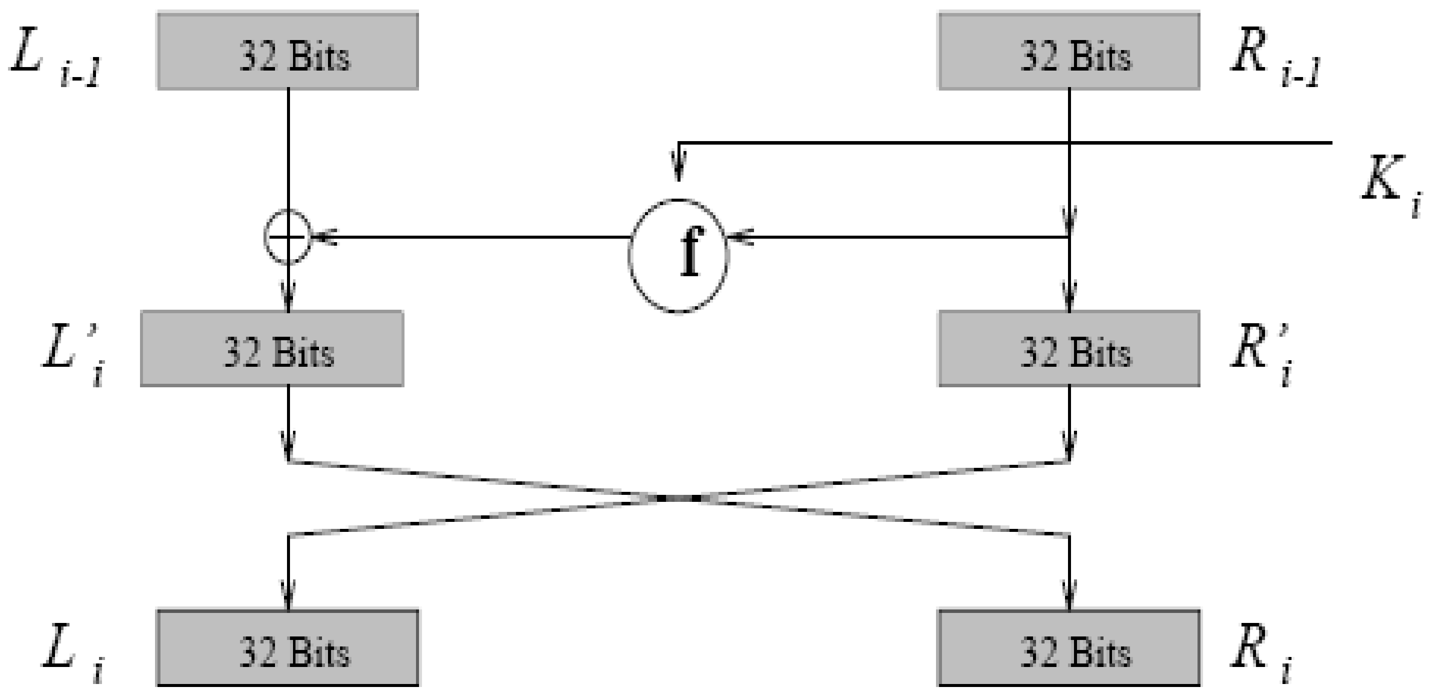
1. Given a plaintext x, a bitstring x_0 is constructed by permuting the bits of x according to a (fixed) *initial permutation* IP.

$$x_0 = \mathrm{IP}\,(x) = L_0 R_0,$$

where L_0 comprises the first 32 bits of x_0 and R_0 the last 32 bits.

DES

2. 16 iterations of a certain function are then computed. We compute L_iR_i for $1 \leq i \leq 16$.



DES

- The 16 iterations (rounds) of computation are computed according to the following rule:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i),$$

where

- $\oplus$ denotes the exclusive-or of two bitstrings.
- f is a function that we will describe later, and
- $K_1, K_2, \ldots, K_{16}$ are each bitstrings of length 48 computed as a function of the key K. (Actually, each K_i is a permuted selection of bits from K.) $K_1, K_2, \ldots, K_{16}$ comprises the *key schedule*.

DES

3. Apply the inverse permutation IP^{-1} to the bitstring $R_{16}L_{16}$, obtaining the ciphertext y. That is, $y = IP^{-1}(R_{16}L_{16})$. Note the inverted order of L_{16} and R_{16}.

DES

DES encryption algorithm:

(m,k)

$m \leftarrow IP(m)$

$L_0 \leftarrow$ leftmost 32 bits of m

$R_0 \leftarrow$ rightmost 32 bits of m

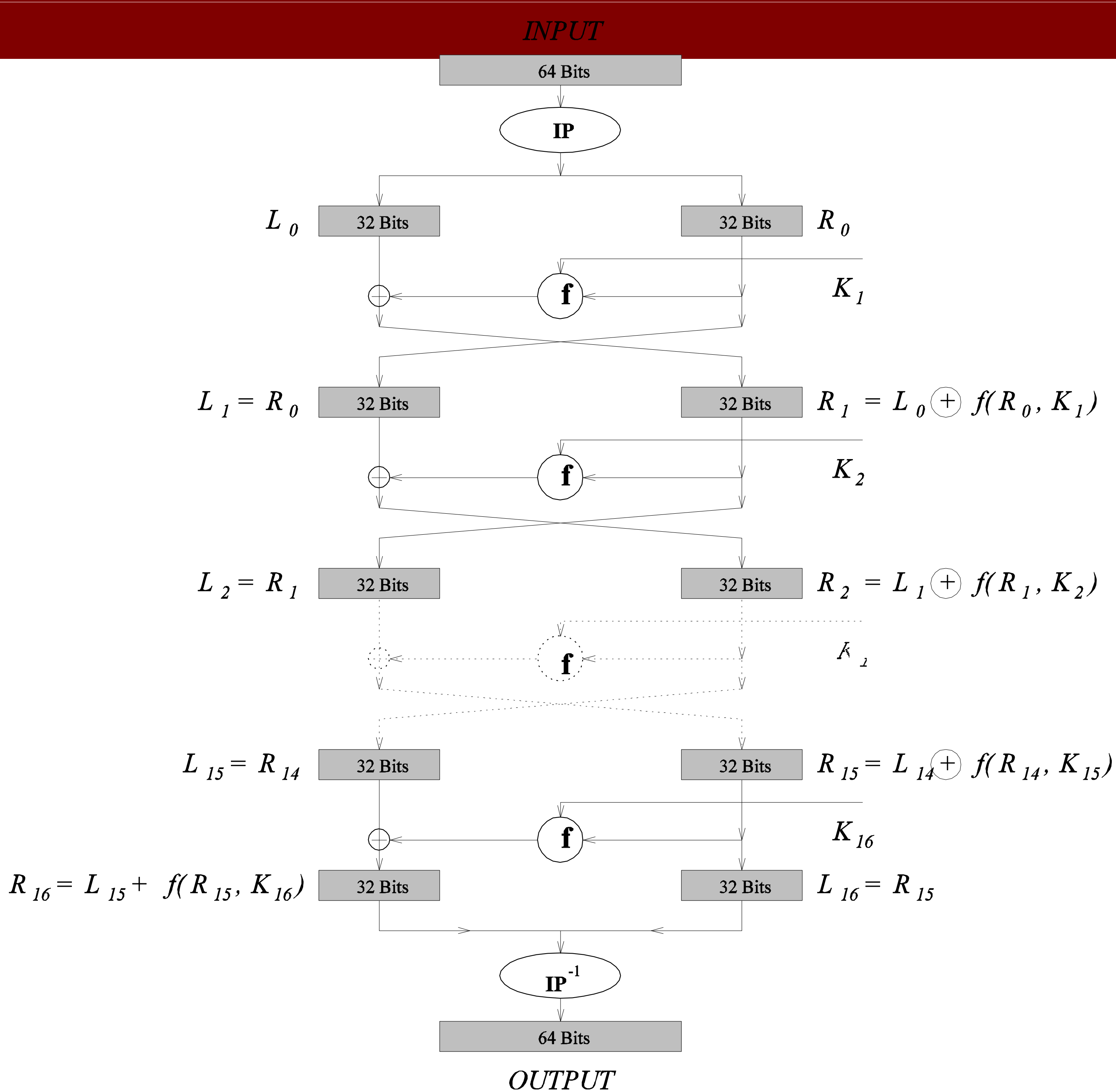
For i =1 to 16 do

$L_i \leftarrow R_{i-1}$

$R_i \leftarrow L_{i-1} \oplus f_{ki}(R_{i-1})$

$c \leftarrow IP^{-1}(R_{16},L_{16})$

(c)


INPUT
64 Bits
IP
L_0
32 Bits
32 Bits
R_0
K_1
f
$L_1 = R_0$
32 Bits
32 Bits
$R_1 = L_0 \oplus f(R_0, K_1)$
K_2
f
$L_2 = R_1$
32 Bits
32 Bits
$R_2 = L_1 \oplus f(R_1, K_2)$
f
$L_{15} = R_{14}$
32 Bits
32 Bits
$R_{15} = L_{14} \oplus f(R_{14}, K_{15})$
K_{16}
f
$R_{16} = L_{15} + f(R_{15}, K_{16})$
32 Bits
32 Bits
$L_{16} = R_{15}$
IP^{-1}
64 Bits
OUTPUT

DES

Decryption:

- For DES, the decryption algorithm is the same as the encryption algorithm. This means that the decryption algorithm discussed earlier can also be used for decryption.
- The only difference is that the key schedule must be reversed, meaning that the DES round keys must be used in reverse order (i.e., $k_{16}, \ldots, k_1$) to decrypt a given ciphertext.

The f function

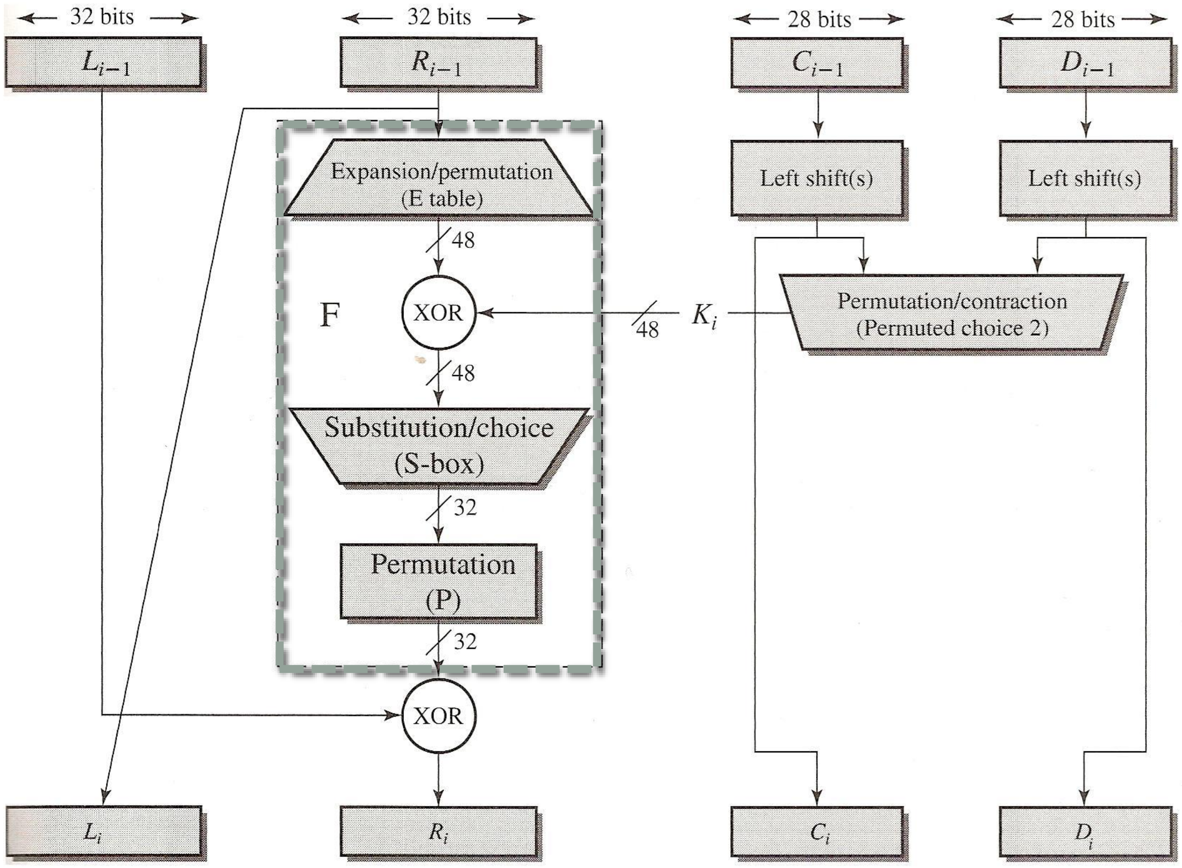


Figure 3.5 Single Round of DES Algorithm

The function F is a non-linear transformation, and is the source of the cryptographic strength of DES.

Source: Stalling W, *Cryptography and Network Security: Principles and Practices,* 4th ed, Prentice Hall, 2005

The f function

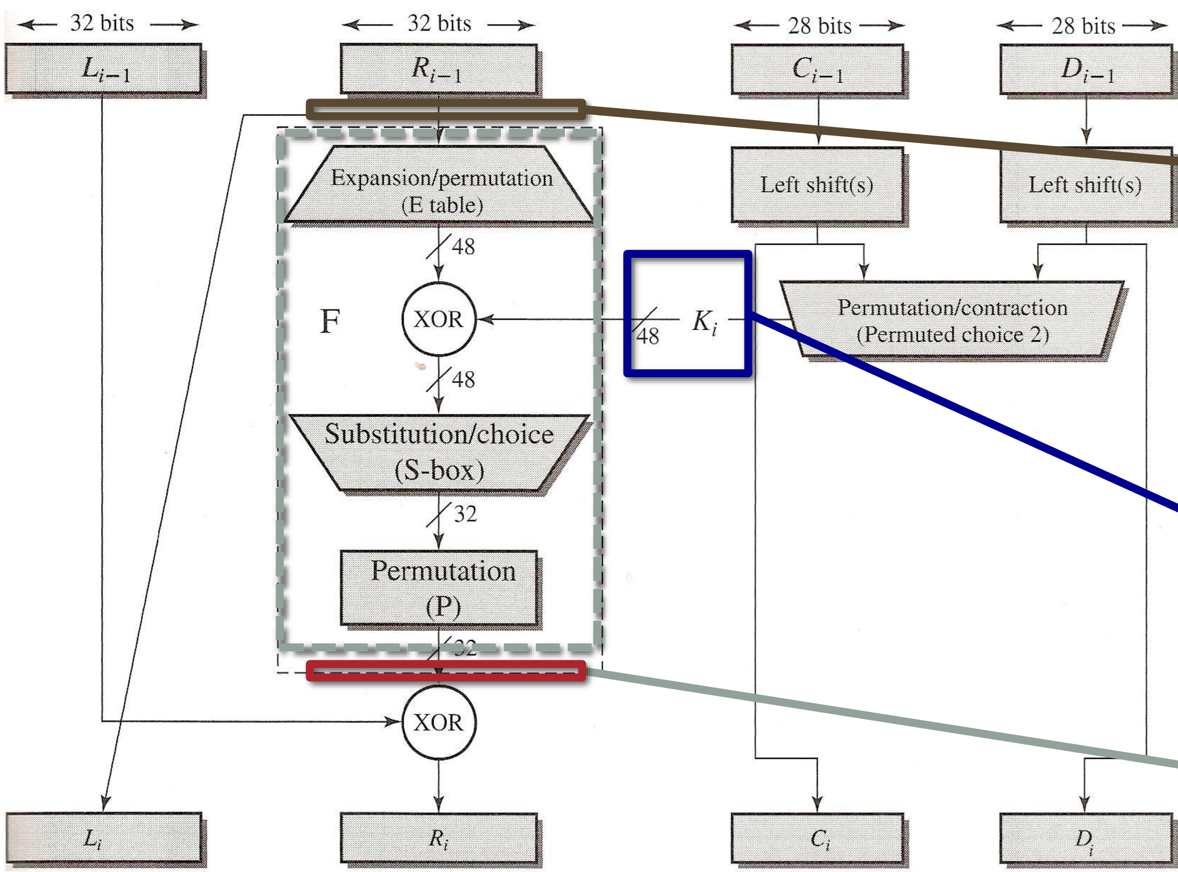


Figure 3.5 Single Round of DES Algorithm

The function F takes as input a first argument that is a bitstring of length 32

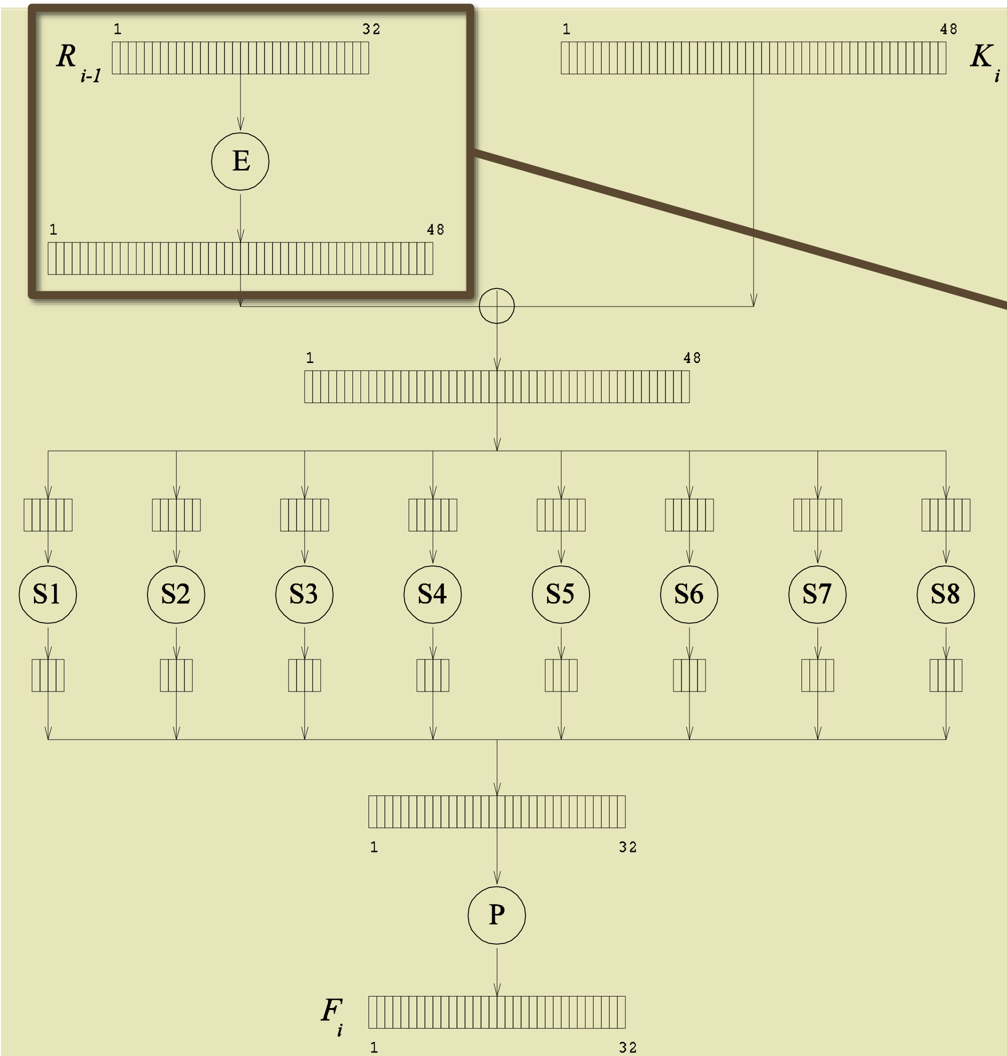
A second argument that is a bitstring of length 48

Produce as output a bitstring of length 32.

The detail processes are described next.

Source: Stalling W, *Cryptography and Network Security: Principles and Practices*, 4th ed, Prentice Hall, 2005

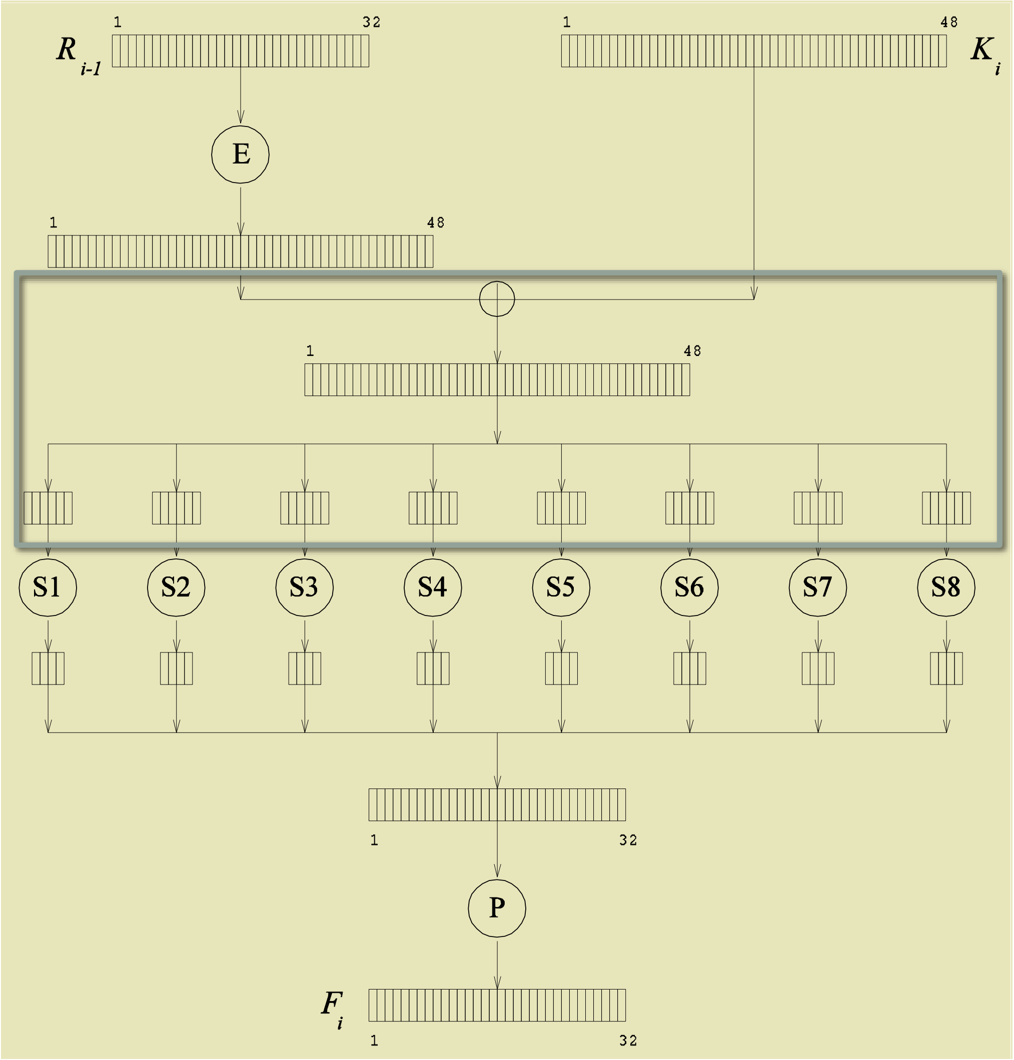
Processes of f function



The first argument is expanded to a bitstring of length 48 according to a fixed expansion function E.

The function E takes in the first argument (32 bits), permuted it in a certain way, with 16 of the bits appearing twice.

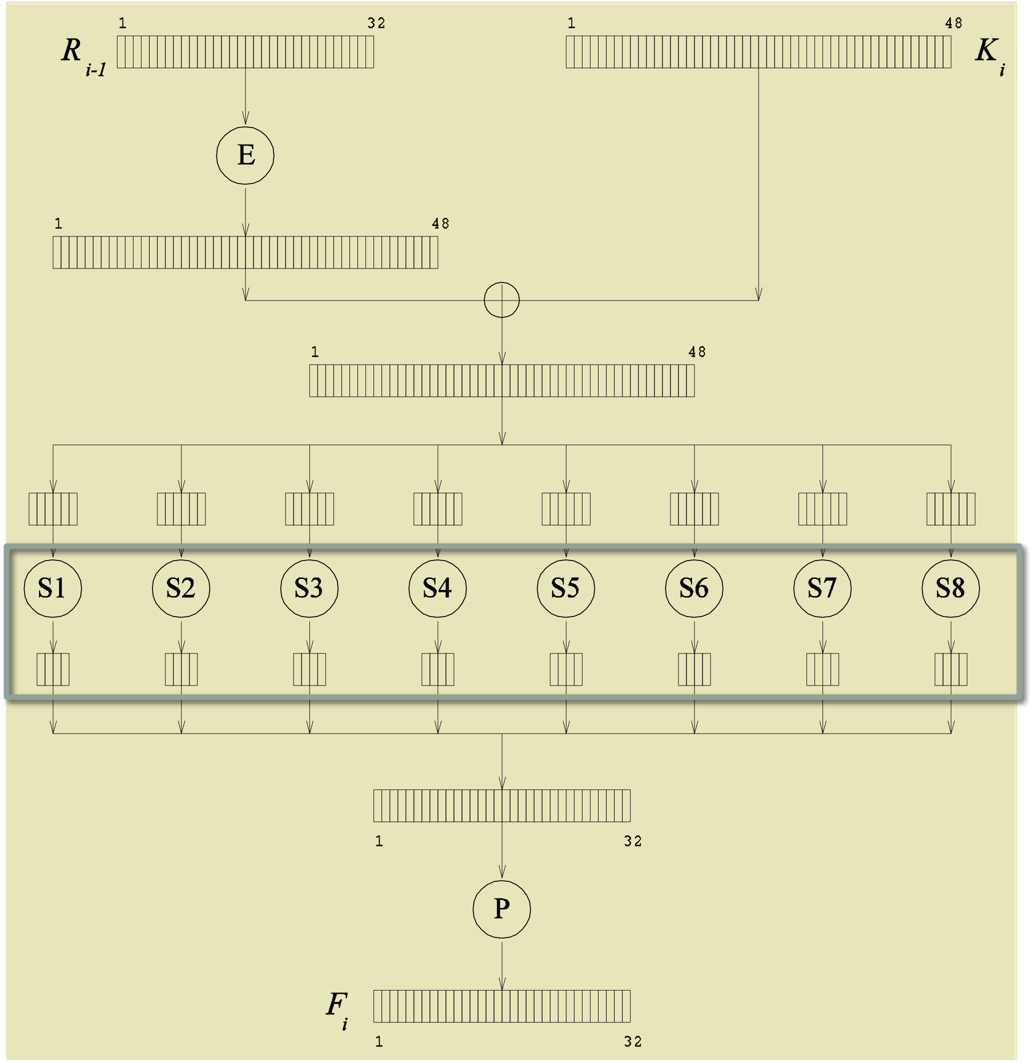
Processes of f function



Compute $E \oplus K$ and write the result as the concatenation of eight 6-bit strings

$$B = B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8.$$

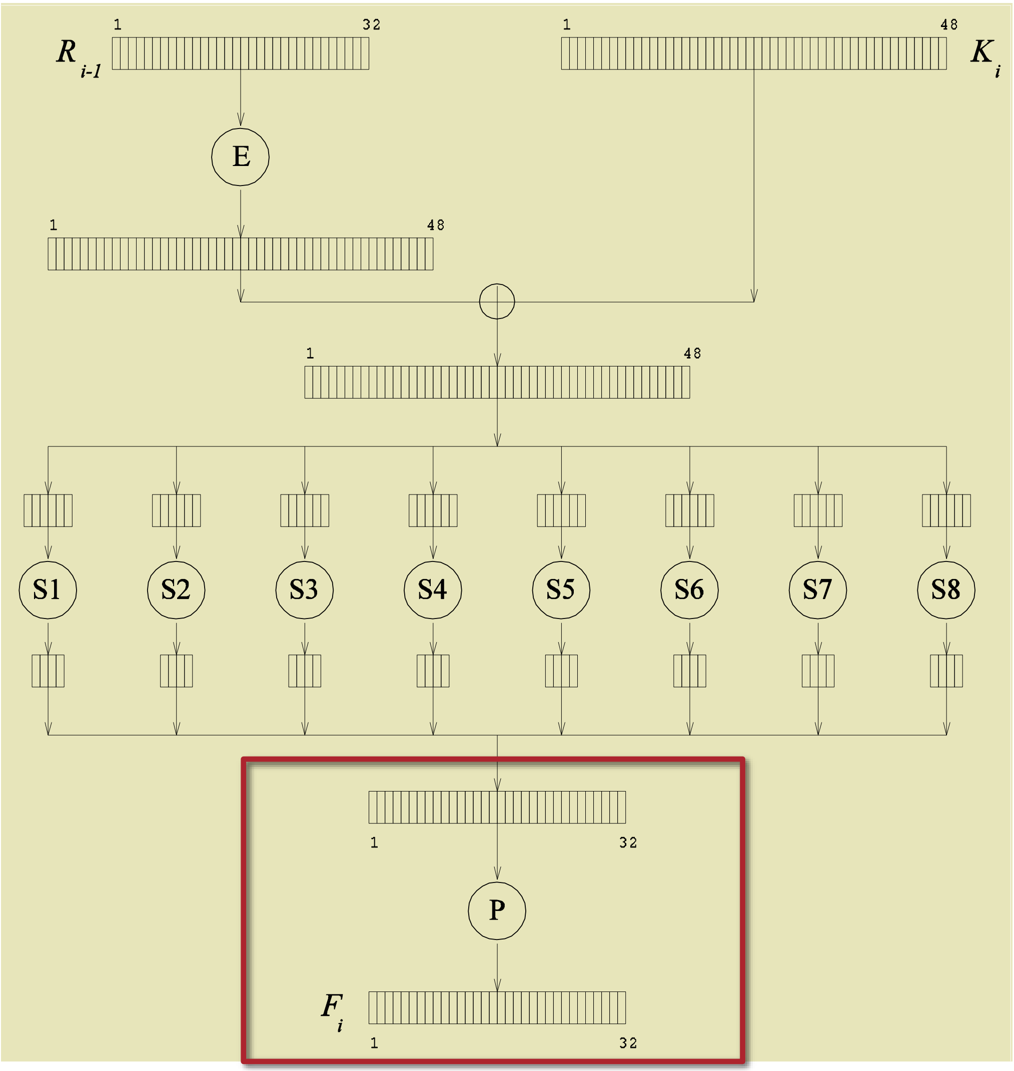
Processes of f function



Next the algorithm uses eight S-boses S_1, S_2, …, S_8. Each S_i is a fixed 4 x 16 array whose entries come from the integers 0 – 15.

Each of the S box S_j acts as a function that accepts as input a bitstring of length six. The first and the last bits are control bits, the other four bits forms the 16 possible values to select from the S-Box. The output of S_j is a bitstring of length four.

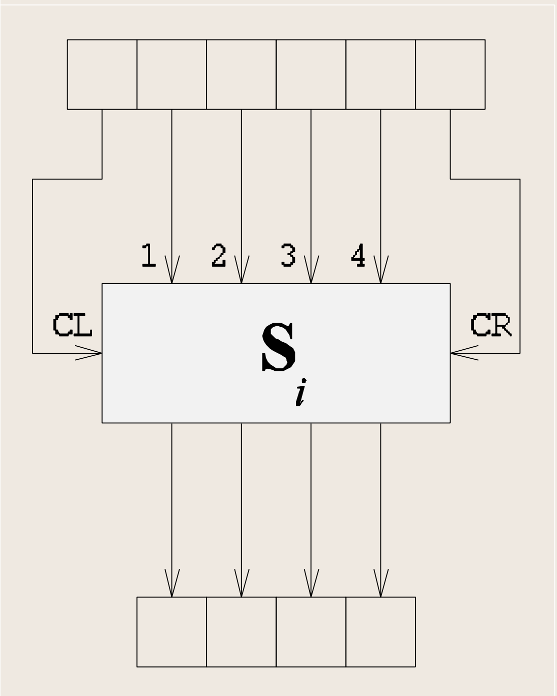
Processes of f function



The output of the eight S boxes form a bitstring $C = C_1C_2C_3C_4C_5C_6C_7C_8$ of length 32.

The $C = C_1C_2C_3C_4C_5C_6C_7C_8$ of length 32 is then permuted according to a fixed permutation P to produce the output of the function F_i.

Structure of S-Box



- CL = left control bit.
- CR= right control bit.
- The CL and CR select one row of the S-Box S_i to use.

Structure of S-Box

- For each of the 4 choices of (CL,CR), S_i performs a different substitution on the 16 possible values of the 4 inner input bits.
- For example $S_1(1,0,1,1,1,0)=[1,0,1,1]$.
- In the S-box below we represent the output values by the integer value of the four binary digits.

00	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
01	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
10	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
11	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Content of s-boxes

The contents of the six S boxes:

S_1															
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S_2															
15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

Content of s-boxes

S_3															
10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S_4															
7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

Content of s-boxes

S_5															
2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S_6															
12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

Content of s-boxes

S_7															
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S_8															
13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Initial permutation (IP)

Table 3.2 Permutation Tables for DES

(a) Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

This means that the 58^{th} bit of x is the first bit of IP(x); the 50^{th} bit of x is the second bit of IP(x), etc.

Inverse permutation (ip^{-1})

(b) Inverse Initial Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Expansion function (E)

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

Permutation function

(d) Permutation Function (P)

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Key computation

- DES expects 64 bits of key, but only uses 56, and the remaining 8 bits are parity-check bits for error detection.
- The 8 parity-check bits are in positions 8, 16, 24,…,64 are defined so that each byte contains an odd number of 1's, and thus a single error can be detected within each group of 8 bits. (These parity-check bits are not included in the computation of the 16 key schedules)

Key computation

The key generation is done as follows:

1. Given a 64-bit key K, discard the parity-check bits and permute the remaining bits of K according to a fixed permutation PC-1.

$$PC-1(K) = C_0 D_0$$

where

- C_0 comprises the first 28 bits of PC-1(K)
- D_0 the last 28 bits

Key computation

2. For i ranging from 1 to 16, compute

$$
\begin{aligned}
C_i &= LS_i(C_{i-1}) \\
D_i &= LS_i(D_{i-1}), \textit{ and} \\
K_i &= PC-2(C_i D_i)
\end{aligned}
$$

where LS_i represents a cyclic left-shift or either one or two positions, depending on the value of i. Shift one position if $i = 1, 2, 9$ *or* 16, and shift two positions otherwise.

Key computation

- The PC-1 and PC-2 are fixed permutation and are shown here.

Permuted Choice One (PC-1)

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Key computation

Permuted Choice Two (PC-2)

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

Key computation

The sixteen rounds of keys are as follow:

Round 1											
10	51	34	60	49	17	33	57	2	9	19	42
3	35	26	25	44	58	59	1	36	27	18	41
22	28	39	54	37	4	47	30	5	53	23	29
61	21	38	63	15	20	45	14	13	62	55	31

Round 2											
2	43	26	52	41	9	25	49	59	1	11	34
60	27	18	17	36	50	51	58	57	19	10	33
14	20	31	46	29	63	39	22	28	45	15	21
53	13	30	55	7	12	37	6	5	54	47	23

Key computation

Round 3											
51	27	10	36	25	58	9	33	43	50	60	18
44	11	2	1	49	34	35	42	41	3	59	17
61	4	15	30	13	47	23	6	12	29	62	5
37	28	14	39	54	63	21	53	20	38	31	7

Round 4											
35	11	59	49	9	42	58	17	27	34	44	2
57	60	51	50	33	18	19	26	25	52	43	1
45	55	62	14	28	31	7	53	63	13	46	20
21	12	61	23	38	47	5	37	4	22	15	54

Key computation

Round 5											
19	60	43	33	58	26	42	1	11	18	57	51
41	44	35	34	17	2	3	10	9	36	27	50
29	39	46	61	12	15	54	37	47	28	30	4
5	63	45	7	22	31	20	21	55	6	62	38

Round 6											
3	44	27	17	42	10	26	50	60	2	41	35
25	57	19	18	1	51	52	59	58	49	11	34
13	23	30	45	63	62	38	21	31	12	14	55
20	47	29	54	6	15	4	5	39	53	46	22

Key computation

Round 7											
52	57	11	1	26	59	10	34	44	51	25	19
9	41	3	2	50	35	36	43	42	33	60	18
28	7	14	29	47	46	22	5	15	63	61	39
4	31	13	38	53	62	55	20	23	37	30	6

Round 8											
36	41	60	50	10	43	59	18	57	35	9	3
58	25	52	51	34	19	49	27	26	17	44	2
12	54	61	13	31	30	6	20	62	47	45	23
55	15	28	22	37	46	39	4	7	21	14	53

Key computation

Round 9											
57	33	52	42	2	35	51	10	49	27	1	60
50	17	44	43	26	11	41	19	18	9	36	59
4	46	53	5	23	22	61	12	54	39	37	15
47	7	20	14	29	38	31	63	62	13	6	45

Round 10											
41	17	36	26	51	19	35	59	33	11	50	44
34	1	57	27	10	60	25	3	2	58	49	43
55	30	37	20	7	6	45	63	38	23	21	62
31	54	4	61	13	22	15	47	46	28	53	29

Key computation

Round 11											
25	1	49	10	35	3	19	43	17	60	34	57
18	50	41	11	59	44	9	52	51	42	33	27
39	14	21	4	54	53	29	47	22	7	5	46
15	38	55	45	28	6	62	31	30	12	37	13

Round 12											
9	50	33	59	19	52	3	27	1	44	18	41
2	34	25	60	43	57	58	36	35	26	17	11
23	61	5	55	38	37	13	31	6	54	20	30
62	22	39	29	12	53	46	15	14	63	21	28

Key computation

Round 13											
58	34	17	43	3	36	52	11	50	57	2	25
51	18	9	44	27	41	42	49	19	10	1	60
7	45	20	39	22	21	28	15	53	38	4	14
46	6	23	13	63	37	30	62	61	47	5	12

Round 14											
42	18	1	27	52	49	36	60	34	41	51	9
35	2	58	57	11	25	26	33	3	59	50	44
54	29	4	23	6	5	12	62	37	22	55	61
30	53	7	28	47	21	14	46	45	31	20	63

Key computation

Round 15											
26	2	50	11	36	33	49	44	18	25	35	58
19	51	42	41	60	9	10	17	52	43	34	57
38	13	55	7	53	20	63	46	21	6	39	45
14	37	54	12	31	5	61	30	29	15	4	47

Round 16											
18	59	42	3	57	25	41	36	10	17	27	50
11	43	34	33	52	1	2	9	44	35	26	49
30	5	47	62	45	12	55	38	13	61	31	37
6	29	46	4	23	28	53	22	21	7	63	39

An Example of DES encryption

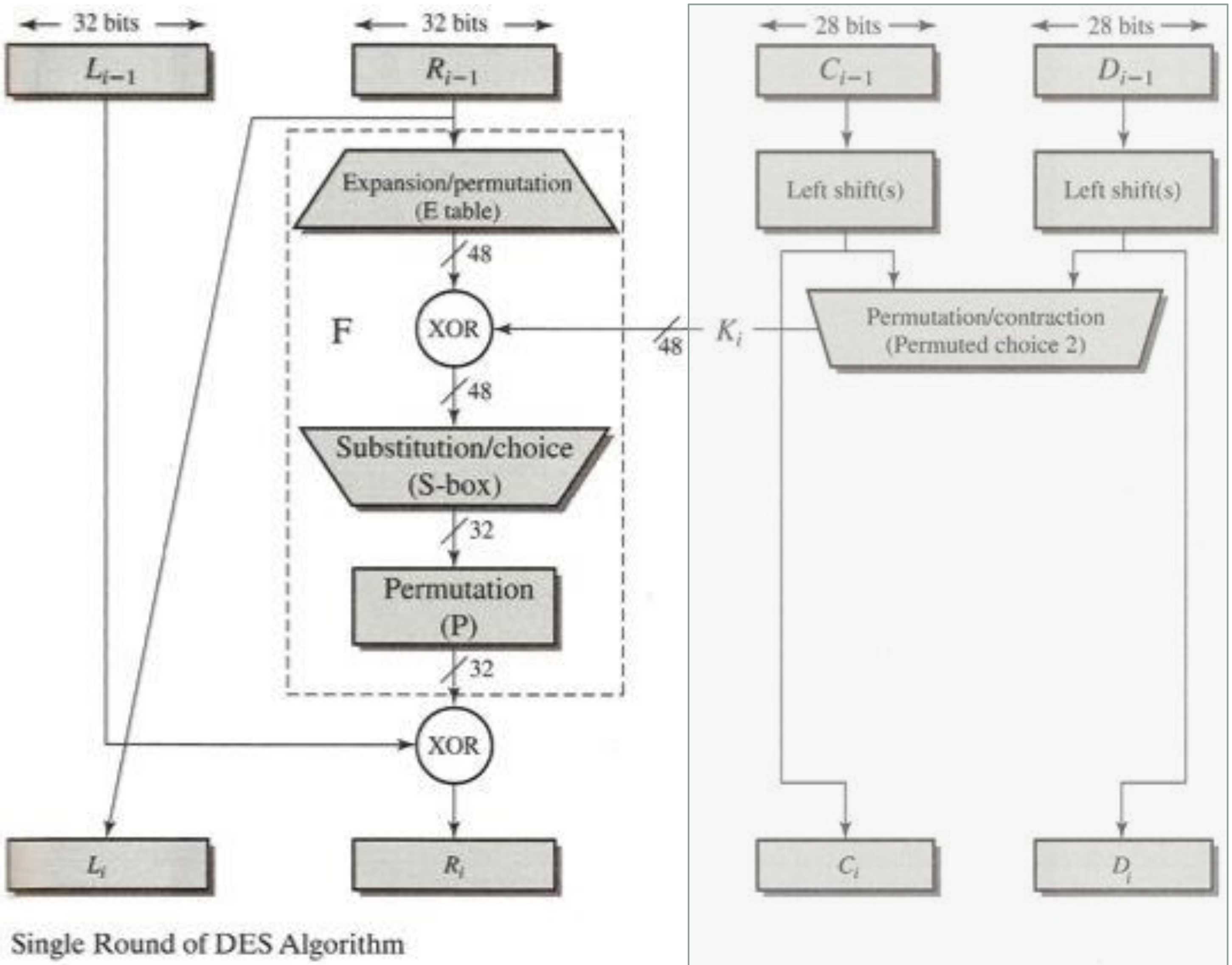
Encrypt the (hexadecimal) plaintext

0123456789ABCDEF

Using the (hexadecimal) key

133457799BBCDFF1

An Example of DES encryption



Single Round of DES Algorithm

An Example of DES encryption

- The key in binary without the parity-check bits is

00010010011010010101101111001001101101111011011111111000

1	3	3	4	5	7	7	9	9	B	B	C	D	F	F	1
00 01	00 1~~1~~	00 11	01 0~~0~~	01 01	01 1~~1~~	01 11	10 0~~1~~	10 01	10 1~~1~~	10 11	11 0~~0~~	11 01	11 1~~1~~	11 11	00 0~~1~~

00010010011010010101101111001001101101111011011111111000

An Example of DES encryption

First the key is permuted using PC-1 to obtain the 56-bit long key:

The key in binary with the parity bits highlighted in yellow:

									1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	3	3	3
1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2
0	0	0	1	0	0	1	1	0	0	1	1	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	1	1	0	0	1

3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	5	6	6	6	6	6
3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4
1	0	0	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1	0	0	0	1

An Example of DES encryption

- The 56-bit long key (after going through the PC-1 permutation) is as follow:

PC-1(K):

```
         11111111112222222222333333333344444444445555555
12345678901234567890123456789012345678901234567890123456
11110000110011001010101011110101010101100110011110001111
```

An Example of DES encryption

- Split the 56-bit key into c0 (first 28 bits) and d0 (next 28 bits) as follow:

c0	1	1	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	0	1	0	1	0	1	0	1	1	1	1
d0	0	1	0	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	1	1	1	0	0	0	1	1	1	1

An Example of DES encryption

- Perform a left-rotate by 1 bit to c0 and d0 to produce c1 and d1:

c0	1 1 1 1 0 0 0 0 1 1 0 0 1 1 0 0 1 0 1 0 1 0 1 0 1 1 1 1
d0	0 1 0 1 0 1 0 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 0 0 1 1 1 1

c1	1 1 1 0 0 0 0 1 1 0 0 1 1 0 0 1 0 1 0 1 0 1 0 1 1 1 1 1
d1	1 0 1 0 1 0 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 0 0 1 1 1 1 0

An Example of DES encryption

- Next we permute c1d1 using PC-2 to produce a 46-bit key:

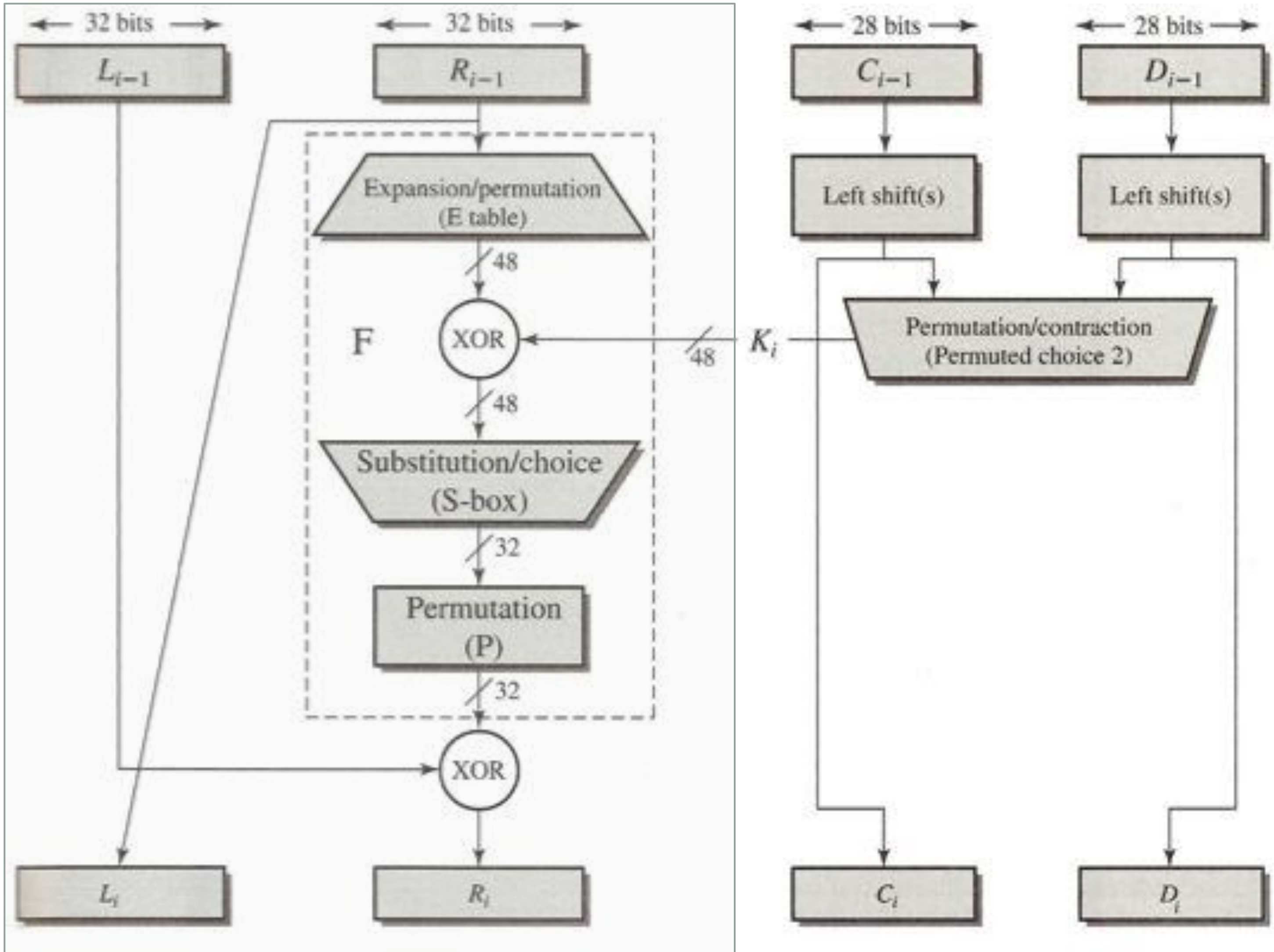
c1	1 1 1 0 0 0 0 1 1 0 0 1 1 0 0 1 0 1 0 1 0 1 0 1 1 1 1 1
d1	1 0 1 0 1 0 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 0 0 1 1 1 1 0

K1 = PC-2(c1d1):

000110110000001011101111111111000111000001110010

End of producing K1 ☺

An Example of DES encryption



Single Round of DES Algorithm

An Example of DES encryption

Applying IP to the plaintext, we obtain L_0 and R_0 (in binary):

$$L_0 = 11001100000000001100110011111111$$

$$L_1 = R_0 = 11110000101010101111000010101010$$

An Example of DES encryption

We’ll see how the first 4 bits are permuted:

The plaintext:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00 00	00 01	00 10	00 11	01 00	01 01	01 10	01 11	10 00	10 01	10 10	10 11	11 00	11 01	11 10	11 11

$$L_0 = \textcolor{red}{1100}1100000000001100110011111111$$

$$L_1 = R_0 = 11110000101010101111000010101010$$

An Example of DES encryptio

The 58^{th} bit becomes the first bit.

The plaintext:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00 00	00 01	00 10	00 11	01 00	01 01	01 10	01 11	10 00	10 01	10 10	10 11	11 00	11 01	11 10	11 11

The permuted plaintext (first 4 bits):

1

Permutation Tables for DES

(a) Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

An Example of DES encrypt

The 50^{th} bit becomes the second bit.

The plaintext:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00 00	00 01	00 10	00 11	01 00	01 01	01 10	01 11	10 00	10 01	10 10	10 11	1$\boxed{1}$ 00	11 01	1**1** 10	11 11

The permuted plaintext (first 4 bits):

11

Permutation Tables for DES

(a) Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

An Example of DES encryp

The 42nd bit becomes the third bit.

The plaintext:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00 00	00 01	00 10	00 11	01 00	01 01	01 10	01 11	10 00	10 01	10 10	10 11	11 00	11 01	11 10	11 11

The permuted plaintext (first 4 bits):

110

Permutation Tables for DES

(a) Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

An Example of DES encrypt

The 34th bit becomes the fourth bit.

The plaintext:

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00 00	00 01	00 10	00 11	01 00	01 01	01 10	01 11	10 00	10 01	1**0** 10	10 11	1**1** 00	11 01	1**1** 10	11 11

The permuted plaintext (first 4 bits):

1100

…and continue…

Permutation Tables for DES

(a) Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

An Example of DES encryption

- Next, the 16 rounds of encryption are performed as follow:

1. Expand R_0 (from 32 bits to 48 bits) according to the expansion table E.

$$R_0 = 11110000101010101111000010101010$$

$$E(R_0) = 011110100001010101010101011110100001010101010101$$

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

An Example of DES encryp

- **We will see how the first 12 bits are g**

Add 32[nd] bit to the front of the bit chunk 1 to 4.

$$R_0 = \boxed{1111}000010101010111100001010101\boxed{0}$$

The first 12 bits of expanded R_0:

01111

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

An Example of DES encryp

- **We will see how the first 16 bits are g**

Add 5^{th} bit to the back of the bit chunk 1 to 4.

$$R_0 = \boxed{1111}0000101010101111000010101010$$

The first 12 bits of expanded R_0:

011110

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

An E… DES encryption

Add 4^{th} bit to the front of the bit chunk 5 to 8.

- We… e first 16 bits are generated.

$$R_0 = 11110000101010101111000010101010$$

The first 12 bits of expanded R_0:

011110100000

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

An Example of ... ption

Add 9th bit to the front of the bit chunk 5 to 8.

- **We will see how th ... generated.**

$$R_0 = 11110000101010101111000010101010$$

The first 12 bits of expanded R_0:

011110100001

(c) Expansion Permutation (E)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

…and continue…

An Example of DES encryption

Next we compute $E \oplus K$:

$$E(R_0) = 011110100001010101010101011110100001010101010101$$

$$K_1 = 000110110000001011101111111111000111000001110010$$

$$E(R_0) \oplus K_1 = 011000010001011110111010100001100110010100100111$$

An Example of DES encryption

Next we substitute the output of the $E(R_0) \oplus K_1$ with eight S - boxes. Each S_i is a fixed 4 x 16 array whose entries come from the integers 0 – 15.

We will show the substitution of the first two s-boxes in the next two slides.

An Example of DES encryption

Substitution of the first 6 bits of $E(R_0) \oplus K_1$ using S - box 1:

The first 6 bits of $E(R_0) \oplus K_1 = 011000$

S_1															
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Left control bit (Cl) = 0 and right control bit (Cr) = 0,

thus index to row of S - box = 00 = 0

Index to column of the S - box = 1100 = 12

Output of S - box 1 = 5 = 0101

An Example of DES encryption

Substitution of the second 6 bits of $E(R_0) \oplus K_1$ using S-box 2:

The second 6 bits of $E(R_0) \oplus K_1 = 010001$

S_2															
15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

Left control bit (Cl) = 0 and right control bit (Cr) = 1,

thus index to row of S-box = 01 = 1

Index to column of the S-box = 1000 = 8

Output of S-box 2 = 12 = 1100

An Example of DES encryption

- The output of S-box: **01011100100000101011010110010111**
- Next the output of the S-box is permuted using the internal permutation function P.

Permutation Function (P)

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

An Example of DES encryption

The output of the permutation function $(f(E(R_0) \oplus K_1) =$
00100011010010101010101001110111011

The output of $f(E(R_0) \oplus K_1)$ is now $\oplus$ with L_0 to obtain R1:

$$f(E(R_0) \oplus K_1) = 00100011010010101010101001110111011$$

$$L_0 = 1100110\,0000000\,0011001100111111111$$

$$R_1 = 1110111\,1010010\,1001100101010000100$$

$$R_0 = L_1 = 1111000\,01010101011110\,00010101010$$

That is the end of the first round ☺
Fifteen more rounds to go ☹

An Example of DES encryption

- To summarize the process for one round:

$$
\begin{aligned}
E(R_0) &= 011110100001010101010101011110100001010101010101 \\
K_1 &= 000110110000001011101111111111000111000001110010 \\
E(R_0) \oplus K_1 &= 011000010001011110111010100001100110010100100111 \\
S\text{-}box\ outputs &\quad\ 01011100100000101011010110010111 \\
f(S\text{-}box\ outputs) &= 00100011010010101010100110111011 \\
{\color{red}L_0} &{\color{red}= 11001100000000001100110011111111} \\
f(S\text{-}box\ outputs) \oplus L_0 = R_1 &= 11101111010010100110010101000100 \\
R_0 = L_1 &= 11110000101010101111000010101010
\end{aligned}
$$

An Example of DES encryption

- The rest of the 15 rounds of processes are as follow:

Round 2:

$$E(R_1) = 011101011110101001010100001100001010101000001001$$

$$K_2 = 011110011010111011011001110110111100100111100101$$

$$E(R_1) \oplus K_2 = 000011000100010010001101111010110110001111101100$$

$$S\text{-}box\ outputs \qquad 11111000110100000011101010101110$$

$$f(S\text{-}box\ outputs) = 00111100101010111000011110100011$$

$$\color{red}{L_1 = 11110000101010101111000010101010}$$

$$f(S\text{-}box\ outputs) \oplus L_1 = R_2 = 11001100000000010111011100001001$$

$$R_1 = L_2 = 11101111010010100110010101000100$$

An Example of DES encryption

Round 3:

$$E(R_2) = 111001011000000000000010101110101110100001010011$$

$$K_3 = 010101011111110010001010010000101100111110011001$$

$$E(R_2) \oplus K_3 = 101100000111110010001000111110000010011111001010$$

$$S\text{ - }box\ outputs \qquad 00100111000100001110000101101111$$

$$f(S\text{ - }box\ outputs) = 01001101000101100110111010110000$$

$$\color{red}{L_2 = 11101111010010100110010101000100}$$

$$f(S\text{ - }box\ outputs) \oplus L_2 = R_3 = 10100010010111000000101111110100$$

$$R_2 = L_3 = 11001100000000010111011100001001$$

An Example of DES encryption

Round 4:

$$E(R_3) = 010100000100000101111100000000101011111111010101001$$

$$K_4 = 011100101010110111010110110110110011010100011101$$

$$E(R_3) \oplus K_4 = 001000101110111100101110110111100100101010110100$$

$$S\text{-}box\ outputs \quad 00100001111011011001111100111010$$

$$f(S\text{-}box\ outputs) = 10111011001000110111011101001100$$

$$L_3 = 11001100000000010111011100001001$$

$$f(S\text{-}box\ outputs) \oplus L_3 = R_4 = 01110111001000100000000001000101$$

$$R_3 = L_4 = 10100010010111000000101111110100$$

An Example of DES encryption

Round 5:

$$E(R_4) = 101110101110100100000100000000000000001000001010$$

$$K_5 = 011111001110110000000111111010110101001110101000$$

$$E(R_4) \oplus K_5 = 110001100000010100000011111010110101000110100010$$

$$S\text{-}box\ outputs \qquad 01010000110010000011000111101011$$

$$f(S\text{-}box\ outputs) = 00101000000100111010110111000011$$

$$\color{red}{L_4 = 10100010010111000000101111110100}$$

$$f(S\text{-}box\ outputs) \oplus L_4 = R_5 = 10001010010011111010011000110111$$

$$R_4 = L_5 = 01110111001000100000000001000101$$

An Example of DES encryption

Round 6:

$$E(R_5) = 110001010100001001011111110100001100000110101111$$

$$K_6 = 011000111010010100111110010100000111101100101111$$

$$E(R_5) \oplus K_6 = 101001101110011101100001100000001011101010000000$$

$$S\text{-}box\ outputs \qquad 01000000111110011010011000011101$$

$$f(S\text{-}box\ outputs) = 10011110010001011100110100101100$$

$$\color{red}{L_5 = 01110111001000100000000001000101}$$

$$f(S\text{-}box\ outputs) \oplus L_5 = R_6 = 11101001011001111100110101101001$$

$$R_5 = L_6 = 10001010010011111010011000110111$$

An Example of DES encryption

Round 7:

$$E(R_6) = 111101010010101100001111111001011010101101010011$$

$$K_7 = 111011001000010010110111111101100001100010111100$$

$$E(R_6) \oplus K_7 = 000110011010111110111000000100111011001111101111$$

$$S\text{-}box\ outputs \qquad 00010000011101010100000010101101$$

$$f(S\text{-}box\ outputs) = 100011000000001010001110000100111$$

$$\color{red}{L_6 = 10001010010011111010011000110111}$$

$$f(S\text{-}box\ outputs) \oplus L_6 = R_7 = 00000110010010101011101000010000$$

$$R_6 = L_7 = 11101001011001111100110101101001$$

An Example of DES encryption

Round 8:

$$E(R_7) = 000000001100001001010101010111110100000010100000$$

$$K_8 = 111101111000101000111010110000010011101111111011$$

$$E(R_7) \oplus K_8 = 111101110100100001101111100111100111101101011011$$

$$S\text{-}box\ outputs \quad 01101100000110000011111001010110$$

$$f(S\text{-}box\ outputs) = 00111100000011101000011011111001$$

$$\color{red}{L_7 = 11101001011001111100110101101001}$$

$$f(S\text{-}box\ outputs) \oplus L_7 = R_8 = 11010101011010010100101110010000$$

$$R_7 = L_8 = 00000110010010101011101000010000$$

An Example of DES encryption

Round 9:

$$E(R_8) = 011010101010101101010010101001010111110010100001$$

$$K_9 = 111000001101101111101011111011011110011110000001$$

$$E(R_8) \oplus K_9 = 100010100111000010111001010010001001101100100000$$

$$S\text{-}box\ outputs \qquad 00010001000011000101011101110111$$

$$f(S\text{-}box\ outputs) = 00100010001101100111110001101010$$

$$\color{red}{L_8 = 00000110010010101011101000010000}$$

$$f(S\text{-}box\ outputs) \oplus L_8 = R_9 = 00100100011111001100011001111010$$

$$R_8 = L_9 = 11010101011010010100101110010000$$

An Example of DES encryption

Round 10:

$$E(R_9) = 000100001000001111111001011000001100001111110100$$

$$K_{10} = 101100011111001101000111101110100100011001001111$$

$$E(R_9) \oplus K_{10} = 101000010111000010111111011011010100001011011011$$

$$S\text{ - }box\ outputs \quad 11011010000001000101001001110101$$

$$f(S\text{ - }box\ outputs) = 01100010101111001001110000100010$$

$$L_9 = 11010101011010010100101110010000$$

$$f(S\text{ - }box\ outputs) \oplus L_9 = R_{10} = 10110111110101011101011110110010$$

$$R_9 = L_{10} = 00100100011111001100011001111010$$

An Example of DES encryption

Round 11:

$$E(R_{10}) = 010110101111111010101011111010101111110110100101$$

$$K_{11} = 001000010101111111010011110111101101001110000110$$

$$E(R_{10}) \oplus K_{11} = 011110111010000101111000001101000010111000100011$$

$$S\text{-}box\ outputs \quad 0111001100000010111010001000000001$$

$$f(S\text{-}box\ outputs) = 1110000100000010011111101000000010$$

$$\color{red}{L_{10} = 001001000111111001100011001111010}$$

$$f(S\text{-}box\ outputs) \oplus L_{10} = R_{11} = 110001010111110000011110001111000$$

$$R_{10} = L_{11} = 10110111110101011101011110110010$$

An Example of DES encryption

Round 12:

$$E(R_{11}) = 011000001010101111110000000111111000001111110001$$

$$K_{12} = 011101010111000111110101100101000110011111101001$$

$$E(R_{11}) \oplus K_{12} = 000101011101101000000101100010111110010000011000$$

$$S\text{ - }box\ outputs \qquad 01111011100010110010011000110101$$

$$f(S\text{ - }box\ outputs) = 11000010011010001100111111101010$$

$$L_{11} = 10110111110101011101011110110010$$

$$f(S\text{ - }box\ outputs) \oplus L_{11} = R_{12} = 01110101101111010001100001011000$$

$$R_{11} = L_{12} = 11000101011110000001111000111100$$

An Example of DES encryption

Round 13:

$$E(R_{12}) = 001110101011110111111010100011110000001011110000$$

$$K_{13} = 100101111100010111010001111110101011101001000001$$

$$E(R_{12}) \oplus K_{13} = 101011010111100000101011011101011011100010110001$$

$$S\text{-}box\ outputs \qquad 10011010110100011000101101001111$$

$$f(S\text{-}box\ outputs) = 11011101101110110010100100100010$$

$$L_{12} = 11000101011110000011110001111000$$

$$f(S\text{-}box\ outputs) \oplus L_{12} = R_{13} = 00011000110000110001010101011010$$

$$R_{12} = L_{13} = 01110101101111010001100001011000$$

An Example of DES encryption

Round 14:

$$E(R_{13}) = 000011110001011000000110100010101010101011110100$$

$$K_{14} = 010111110100001110110111111100101110011100111010$$

$$E(R_{13}) \oplus K_{14} = 010100000101010110110001011110000100110111001110$$

$$S\text{ - }box\ outputs \quad 01100100011110011001101011110001$$

$$f(S\text{ - }box\ outputs) = 10110111001100011000111001010101$$

$$\color{red}{L_{13} = 01110101101111010001100001011000}$$

$$f(S\text{ - }box\ outputs) \oplus L_{13} = R_{14} = 11000010100011001001011000001101$$

$$R_{13} = L_{14} = 00011000110000011000101010101010$$

An Example of DES encryption

<u>Round 15:</u>

$$E(R_{14}) = 111000000101010001011001010010101100000001011011$$

$$K_{15} = 101111111001000110001101001111010011111100001010$$

$$E(R_{14}) \oplus K_{15} = 0101111111000101110101000111011111111111101010001$$

$$S\text{-}box\ outputs \quad 10110010111010001000110100111100$$

$$f(S\text{-}box\ outputs) = 01011011100000010010011101101110$$

$$\color{red}{L_{14} = 000110001100000110001010101011010}$$

$$f(S\text{-}box\ outputs) \oplus L_{14} = R_{15} = 010000110100000100011001000110100$$

$$R_{14} = L_{15} = 11000010100011001001011000001101$$

An Example of DES encryption

Round 16:

$$E(R_{15}) = 001000000110101000000100000110100100000110101000$$

$$K_{16} = 110010110011110110001011000011100001011111110101$$

$$E(R_{15}) \oplus K_{16} = 111010110101011110001111000101000101011001011101$$

$$S\text{ - }box\ outputs \quad 10100111100000110010010000101001$$

$$f(S\text{ - }box\ outputs) = 11001000110000000100111110011000$$

$$L_{15} = 11000010100011001001011000001101$$

$$f(S\text{ - }box\ outputs) \oplus L_{15} = R_{16} = 00001010010011001101100110010101$$

$$R_{15} = L_{16} = 01000011010000100011001000110100$$

An Example of DES encryption

$$R_{16} = 00001010010011001101100110010101$$

$$L_{16} = 01000011010000100011001000110100$$

- Applying IP^{-1} to L_{16} and R_{16}, we obtain the ciphertext.

85E813540F0AB405

The plaintext: 0123456789ABCDEF

An Example of DES encryption

We will see the inverse initial permutation (IP^{-1}) of the first four bits: (Take note that the order of the bits string is $R_{16}L_{16}$ not $L_{16}R_{16}$)

$$R_{16} = 0000101001 \quad \ldots \quad 010101$$

The 40th bit becomes the first bit.

$$L_{16} = 010000\boxed{1}1010000010001100100011010 0$$

The first four bits after going through the IP^{-1}:

1

(b) Inverse Initial Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

An Example of DES encryption

We will see the inverse initial permutation (IP^{-1}) of the first four bits: (Take note that the order of the bits string is $R_{16}L_{16}$ not $L_{16}R_{16}$)

$$R_{16} = 0000101\boxed{0}001001 \quad 0101$$

The 8th bit becomes the second bit.

$$L_{16} = 01000011010000100011001000110100$$

The first four bits after going through the IP^{-1}:

10

(b) Inverse Initial Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

An Example of DES encryption

We will see the inverse initial permutation (IP^{-1}) of the first four bits: (Take note that the order of the bits string is $R_{16}L_{16}$ not $L_{16}R_{16}$)

$$R_{16} = 00001010010011001$$

The 48th bit becomes the third bit.

$$L_{16} = 010000110100001\boxed{0}0011001000110100$$

The first four bits after going through the IP^{-1}:

100

(b) Inverse Initial Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

An Example of DES encryption

We will see the inverse initial permutation (IP^{-1}) of the first four bits: (Take note that the order of the bits string is $R_{16}L_{16}$ not $L_{16}R_{16}$)

$$R_{16} = 000010100100110\boxed{0}1$$

The 16th bit becomes the fourth bit.

$$L_{16} = 01000011010000010001100100110100$$

The first four bits after going through the IP^{-1}:

1000

…and so on…

(b) Inverse Initial Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

DES related questions

DES related question

What is meet-in-the-middle attack?

Meet-in-the-middle attack is an attack used by an attacker to attack a multiple encryption algorithm. This attack requires a known plaintext-ciphertext pair. In essence, the plaintext is encrypted to produce an intermediate value in the multiple encryption, and the ciphertext is decrypted to produce an intermediation value in the multiple encryption. Table lookup (dictionary attack) techniques can be used in such a way to dramatically improve on a brute-force try of all pairs of keys.

DES related question

A characteristics of Feistel network is 'involution'. Show that Feistel network has this characteristic.

$$f_i\left(L_{i-1}, R_{i-1}\right) = \left(L_{i-1} \oplus f\left(K_i, R_{i-1}\right), R_{i-1}\right)$$

$$\begin{aligned} f_i\left(f_i\left(L_{i-1}, R_{i-1}\right)\right) &= f_i\left(L_{i-1} \oplus f\left(K_i, R_{i-1}\right), R_{i-1}\right) \\ &= \left(L_{i-1} \oplus f\left(K_i, R_{i-1}\right) \oplus f\left(K_i, R_{i-1}\right), R_{i-1}\right) \\ &= \left(L_{i-1}, R_{i-1}\right) \end{aligned}$$