University of Salahaddin - Erbil
College of Engineering
Department of Software Engineering

Data Security

Academic year 2021-2022
4th Year Material
Chapter Three

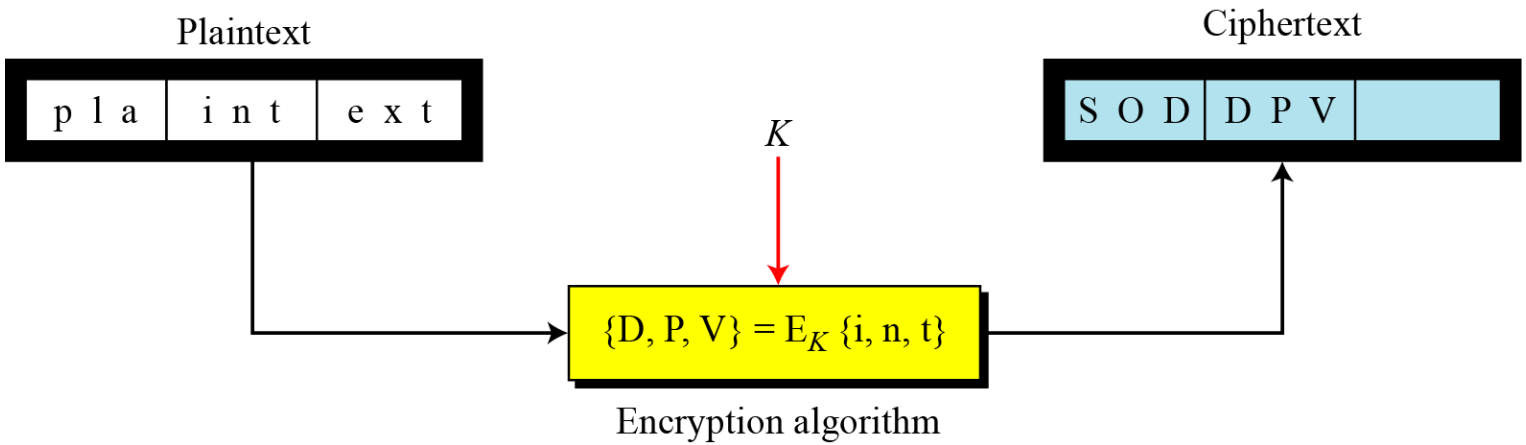
Modern Block Ciphers: Block cipher principles & Data Encryption Standard (DES)

Prepared By: Mr. **Zana Farhad Doghramachi, M.Tech(CSE)**
Zana.softeng@gmail.com

Block Ciphers

- Is an encryption/decryption scheme in which a block of plaintext is treated as a whole and used to produce a ciphertext block of equal length.
- Block cipher is a group of plaintext symbols of size m (m > 1) are encrypted together creating a group of ciphertext of the same size. A single key is used to encrypt the whole block.
- Many block ciphers have a Feistel structure. Such a structure consists of a number of identical rounds of processing. In each round, a substitution is performed on one half of the data being processed, followed by a permutation that interchanges the two halves. The original key is expanded so that a different key is used for each round.

Concept of Block Cipher



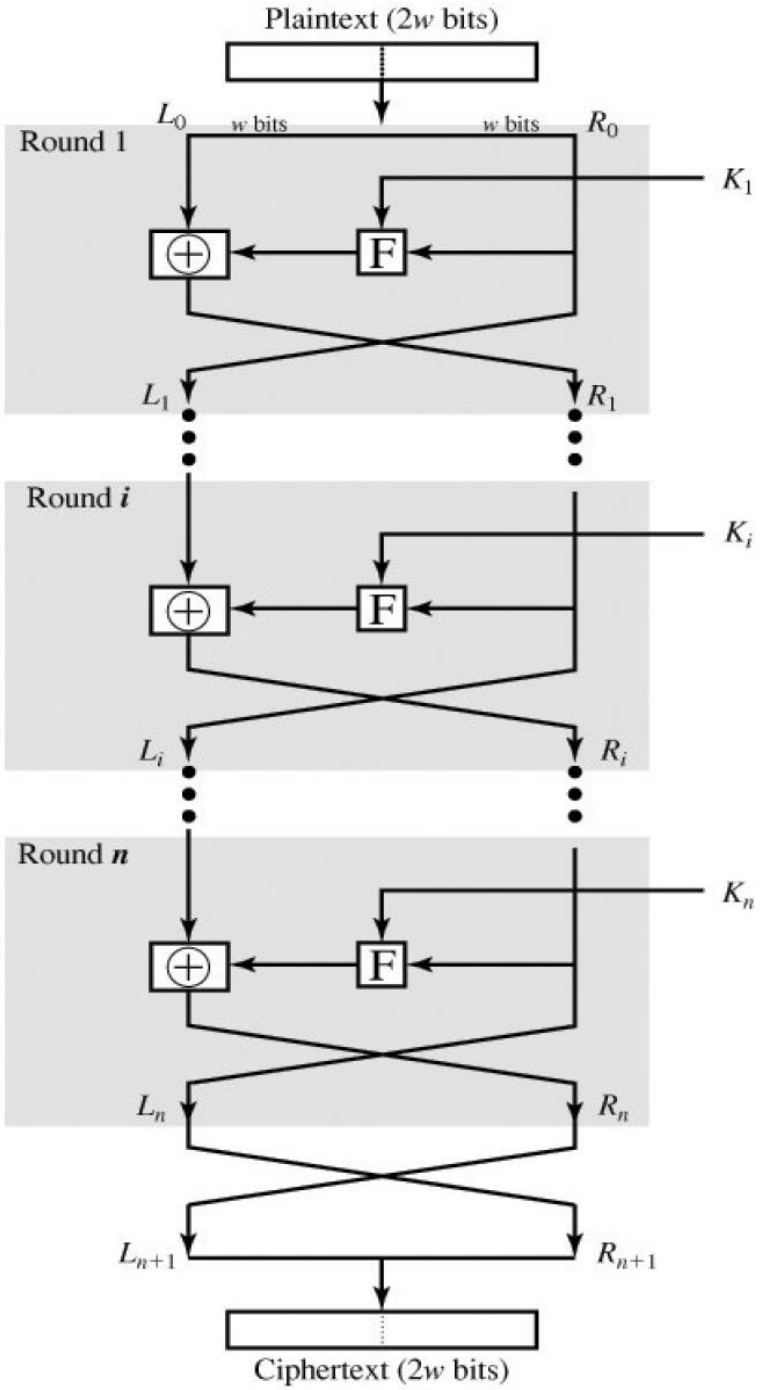
Feistel Cipher Structure I

- The inputs to the encryption algorithm are a plaintext block of length 2w bits and a key K. The plaintext block is divided into two halves, L_0 and R_0. The two halves of the data pass through n rounds of processing and then combine to produce the ciphertext block. Each round $_i$ has as inputs L_{i-1} and R_{i-1}, derived from the previous round, as well as a subkey K_i, derived from the overall K. In general, the subkeys K_i are different from K and from each other.

- All rounds have the same structure. A substitution is performed on the left half of the data. This is done by applying a round function F to the right half of the data and then taking the Exclusive-OR of the output of that function and the left half of the data.

Feistel Cipher Structure II

- The round function has the same general structure for each round but is parameterized by the round subkey K_i. Following this substitution, a permutation is performed that consists of the interchange of the two halves of the data.


Plaintext (2w bits)
L0
w bits
w bits
R0
Round 1
K1
F
L1
R1
Round i
Ki
F
Li
Ri
Round n
Kn
F
Ln
Rn
Ln+1
Rn+1
Ciphertext (2w bits)

Feistel Cipher I

The exact realization of a Feistel network depends on the choice of the following parameters and design features:

- **Block size**: Larger block sizes mean greater security but reduced encryption/decryption speed for a given algorithm. The greater security is achieved by greater diffusion. Traditionally, a block size of 64 bits has been considered a reasonable tradeoff and was nearly universal in block cipher design. However, the new AES uses a 128-bit block size.

- **Key size**: Larger key size means greater security but may decrease encryption/decryption speed. The greater security is achieved by greater resistance to brute-force attacks and greater confusion. Key sizes of 64 bits or less are now widely considered to be inadequate, and 128 bits has become a common size.

Feistel Cipher II

- **Number of rounds**: The essence of the Feistel cipher is that a single round offers inadequate security but that multiple rounds offer increasing security. A typical size is 16 rounds.

- **Subkey generation algorithm**: Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.

- **Round function:** Again, greater complexity generally means greater resistance to cryptanalysis.

Feistel Cipher III

There are two other considerations in the design of a Feistel cipher:

- **Fast software encryption/decryption**: In many cases, encryption is embedded in applications or utility functions in such a way as to preclude a hardware implementation. Accordingly, the speed of execution of the algorithm becomes a concern.

- **Ease of analysis**: Although we would like to make our algorithm as difficult as possible to cryptanalyze.

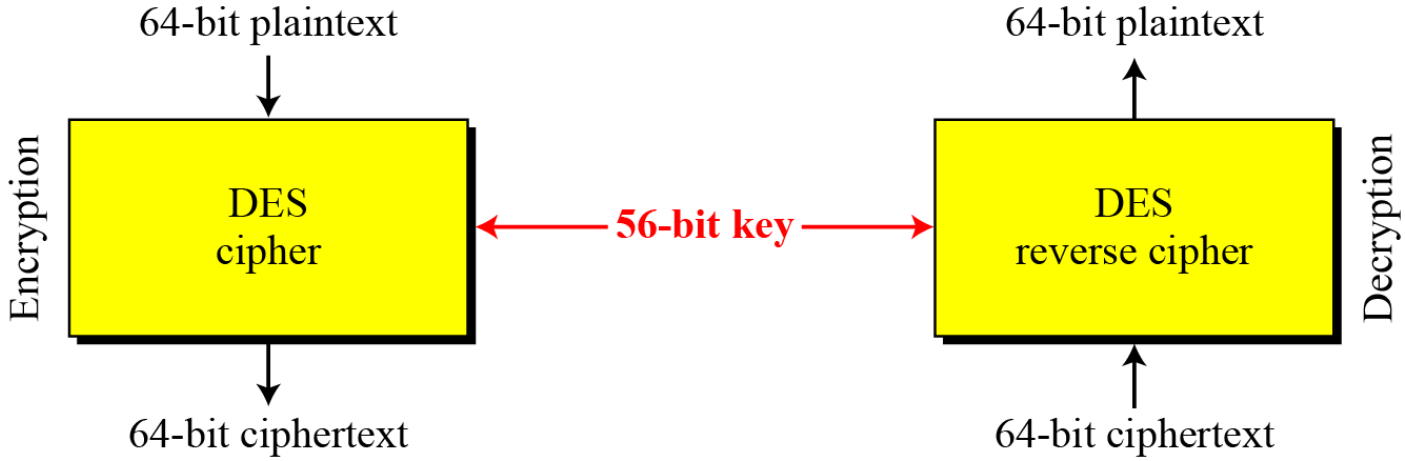
Homework

- Q1 What is Diffusion and Confusion?

Data Encryption Standard

- The ***Data Encryption Standard (DES)*** is a symmetric-key block cipher published in 1977 by the National Institute of Standards and Technology (NIST).

- ***DES*** is a block cipher, data are encrypted in 64 bit blocks using 56 bit key, the same 56 bit cipher key is used for both encryption and decryption. At the encryption site, DES takes a 64 bit plaintext, transforms it in a series of steps into a 64 bit ciphertext; at decryption site.



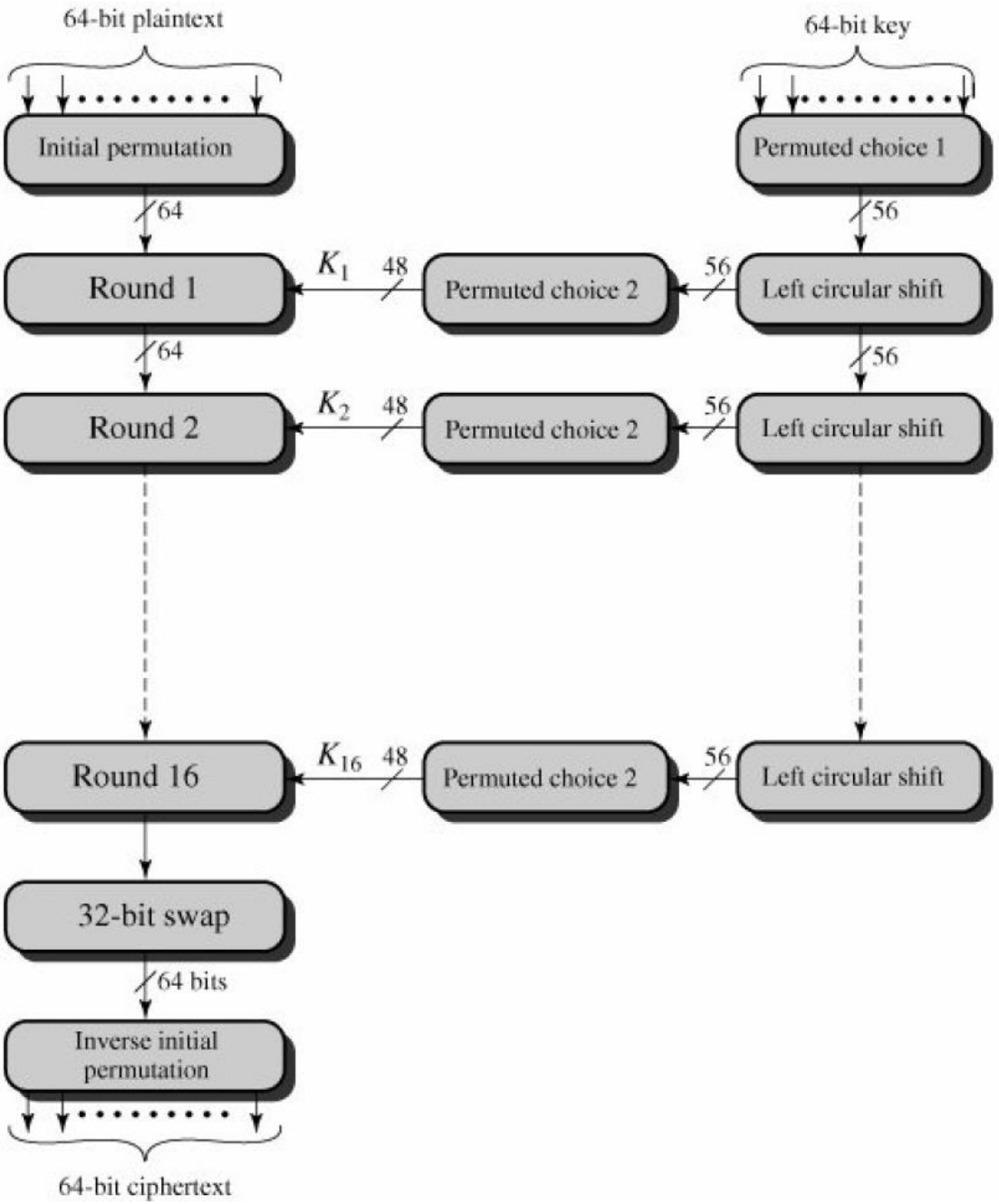
DES Encryption I

- As with any encryption scheme, there are two inputs to the encryption function: the ***plaintext*** to be encrypted and the ***key***. In this case, the plaintext must be 64 bits in length and the key is 56 bits in length.

- Looking at the left-hand side of the figure, we can see that the processing of the plaintext proceeds in three phases. First, the 64-bit plaintext passes through an initial permutation (IP) that rearranges the bits to produce the permuted input. This is followed by a phase consisting of 16 rounds of the same function, which involves both permutation and substitution functions. The output of the last (sixteenth) round consists of 64 bits that are a function of the input plaintext and the key. The left and right halves of the output are swapped to produce the preoutput.

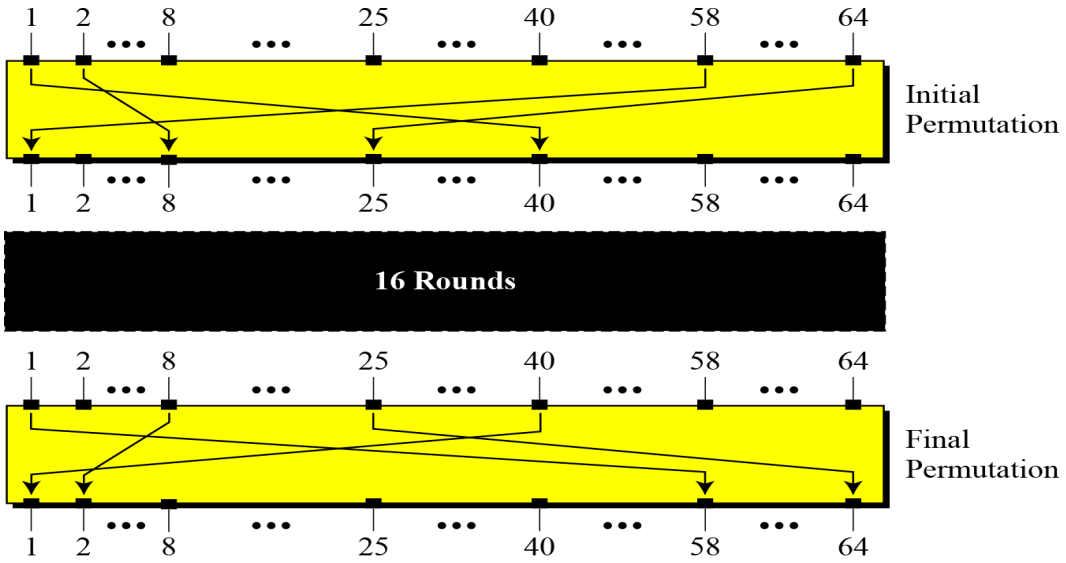
DES Encryption II

- Finally, the preoutput is passed through final permutation (IP^{-1}) that is the inverse of the initial permutation function, to produce the 64-bit ciphertext.


64-bit plaintext
64-bit key
Initial permutation
Permuted choice 1
64
56
Round 1
K_1
48
Permuted choice 2
56
Left circular shift
64
56
Round 2
K_2
48
Permuted choice 2
56
Left circular shift
Round 16
K_{16}
48
Permuted choice 2
56
Left circular shift
32-bit swap
64 bits
Inverse initial permutation
64-bit ciphertext

Initial & Final permutation I

- Each of these permutations takes a 64 bit input and permutes them according to a predefined rule. These permutations are keyless straight permutations that are the inverse of each other. In other words, if the rounds between these two permutations do not exit, the 58th bit entering the initial permutation is the same as the 58th bit leaving the final permutation.



Initial & Final permutation II

- The input to a table consists of 64 bits numbered from 1 to 64.

Initial Permutation	*Final Permutation*
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

Quiz

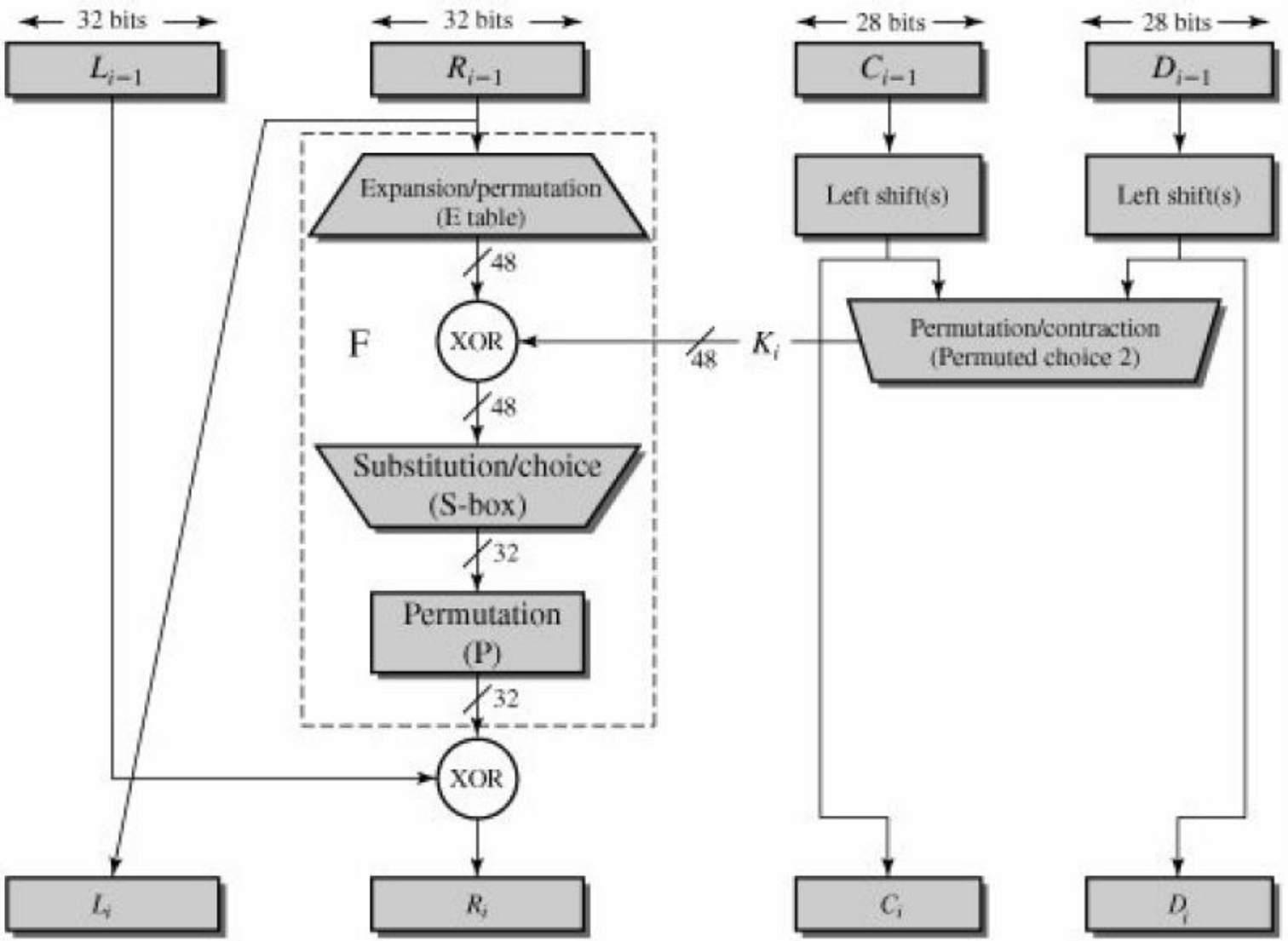
- Q1 Find the output of the final permutation box when the input is given in hexadecimal as:

0x0002 0000 0000 0001

- Q2 Find the output of the initial permutation if the input is

0x0000 0080 0000 0002

Single Round of DES



Details of Single Round I

- DES uses 16 rounds. Each round of DES is a Feistel cipher. The left and right halves of each 64-bit intermediate value are treated as separate 32-bit quantities, labeled L (left) and R (right). The overall processing at each round can be summarized in the following formulas:

- The round key K_i is 48 bits. The R input is 32 bits. This R input is first expanded to 48 bits by passing through an expansion/permutation. The resulting 48 bits are XORed with K_i. This 48-bit result passes through a substitution function that produces a 32-bit output.

- In the expansion table, the 32 bits of input are split into the groups of 4 bits (8 S-boxes), and then become groups of 6 bits by taking the outer bits from the two adjacent groups.

Details of Single Round II

- For example, if part of the input word is
  ... efgh ijkl mnop ... *this becomes*
  ... defghi hijklm lmnopq …

- The substitution consists of a set of eight S-boxes, each of which accepts 6 bits as input and produces 4 bits as output. These, which is interpreted as follows: The first and last bits of the input to box S_i form a 2-bit binary number to select one of four substitutions defined by the four rows in the table for S_i. The middle four bits select one of the sixteen columns.

- The decimal value in the cell selected by the row and column is then converted to its 4-bit representation to produce the output. For example, in S_1 for input 011001, the row is 01 (row 1) and the column is 1100 (column 12). The value in row 1, column 12 is 9, so the output is 1001.

Details of Single Round II

- Permutation (p) the last operation in the DES function is a permutation with a 32 bit input and a 32 bit output. The input/output relationship for this operation is shown in the next slide, for example the seventh bit of the input becomes the second bit of the output

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

Permutation table

DES S-Boxes I

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

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

DES S-Boxes II

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

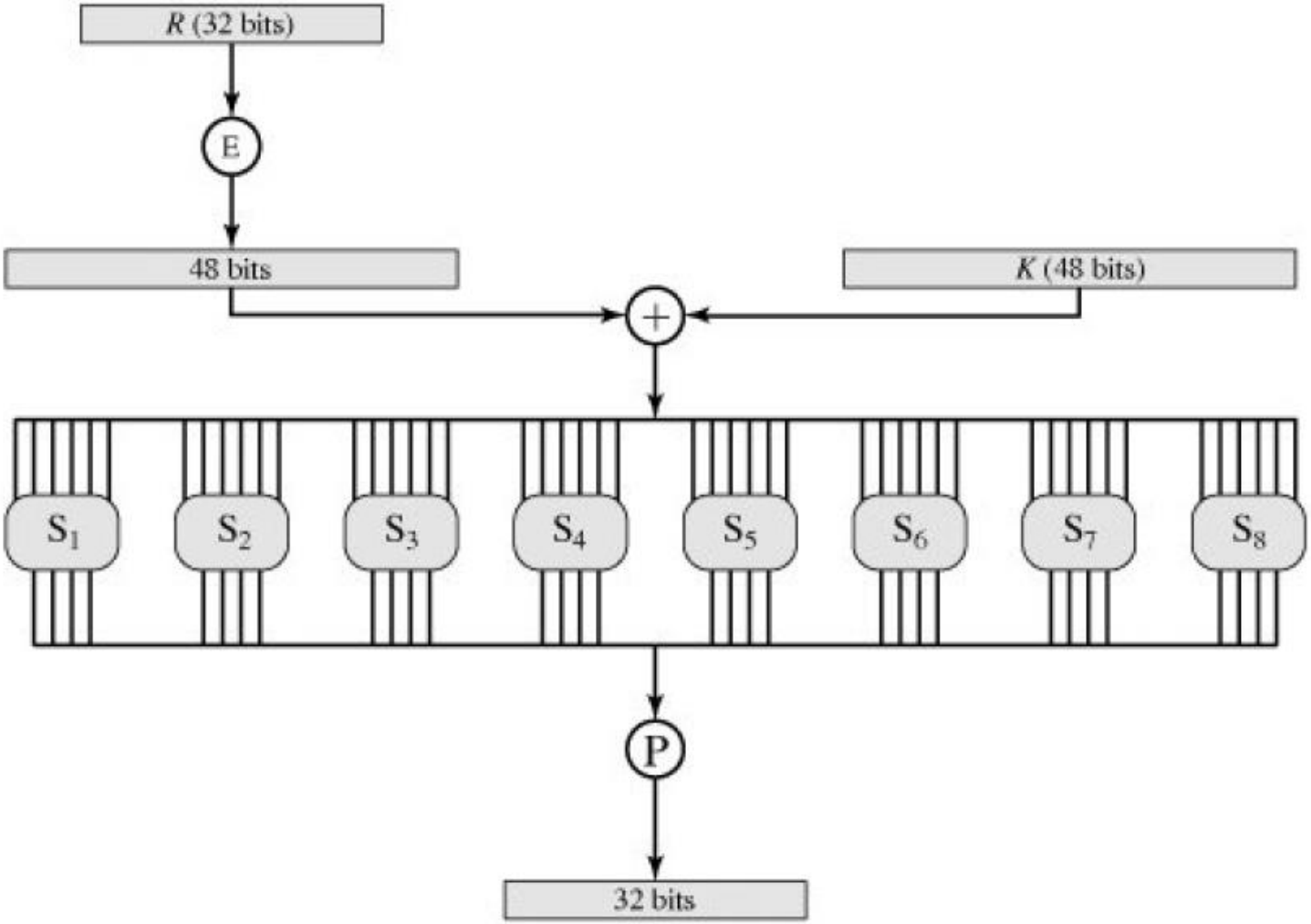
S_7

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

F(R,K)



Key Generation I

- 64-bit key is used as input to the algorithm. The bits of the key are numbered from 1 through 64; every eighth bit is ignored.
- The key is first subjected to a permutation governed by a table labeled Permuted Choice One .
- The resulting 56-bit key is then treated as two 28-bit quantities, labeled C_0 and D_0. At each round, C_{i-1} and D_{i-1} are separately subjected to a circular left shift, or rotation, of 1 or 2 bits.
- These shifted values serve as input to the next round. They also serve as input to Permuted Choice Two, which produces a 48-bit output that serves as input to the function $F(R_{i-1}, K_i)$.

Key Generation II

(a) Input Key

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

(b) 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

(c) 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

(d) Schedule of Left Shifts

Round number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bits rotated	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

DES Decryption

- As with any Feistel cipher, decryption uses the same algorithm as encryption, except that the application of the subkeys is reversed.

The Avalanche Effect

- A desirable property of any encryption algorithm is that a small change in either the plaintext or the key should produce a significant change in the ciphertext. In particular, a change in one bit of the plaintext or one bit of the key should produce a change in many bits of the ciphertext. DES exhibits a strong avalanche effect.

The Strength of DES

- The level of security provided by DES fall into two areas: key size and the nature of the algorithm. With a key length of 56 bits, there are 2^{56} possible keys, which is approximately 7.2×10^{16}. Thus, on the face of it, a brute-force attack appears impractical. Assuming that, on average, half the key space has to be searched, a single machine performing one DES encryption per microsecond would take more than a thousand years to break the cipher.

- DES finally and definitively proved insecure in July 1998, when the Electronic Frontier Foundation (EFF) announced that it had broken a DES encryption using a special-purpose "DES cracker" machine that was built for less than $250,000. The attack took less than three days.

The Strength of DES

- The EFF has published a detailed description of the machine, enabling others to build their own cracker, hardware prices will continue to drop, making DES virtually worthless.

- If the text message has been compressed before encryption, then recognition is more difficult. And if the message is some more general type of data, such as a numerical file, and this has been compressed, the problem becomes even more difficult to automate.

Number of Rounds

- The greater the number of rounds, the more difficult it is to perform cryptanalysis, even for a relatively weak F. In general, the criterion should be that the number of rounds is chosen so that known cryptanalytic efforts require greater effort than a simple brute-force key search attack.

Homework

- Q1 What is Differential and Linear Cryptanalysis?