CS6701 Cryptography and Network Security

Block Ciphers:Data Encryption Standard (DES)

Unit-II

Lecture -1

July 3, 2018

Session Meta Data

Author	Dr. J. Bhuvana
Reviewer	
Version Number	1.2
Date	July 3, 2018

Session Objectives

- Priciples of block Cipher

Session Outcomes

At the end of this session, participants will be able to

Agenda

1 Data Encryption Standard (DES)

Block Cipher Principles

- most symmetric block ciphers are based on a Feistel Cipher Structure
- block ciphers look like an extremely large substitution
- would need table of 2^{64} entries for a 64-bit block
- instead create from smaller building blocks
- using idea of a product cipher

Claude Shannon and Substitution- Permutation Ciphers

- Claude Shannon introduced idea of substitution-permutation (S-P) networks in 1949 paper
- form basis of modern block ciphers
- S-P nets are based on the two primitive cryptographic operations seen before:
  - **substitution (S-box)**
  - **permutation (P-box)**
- provide confusion & diffusion of message & key

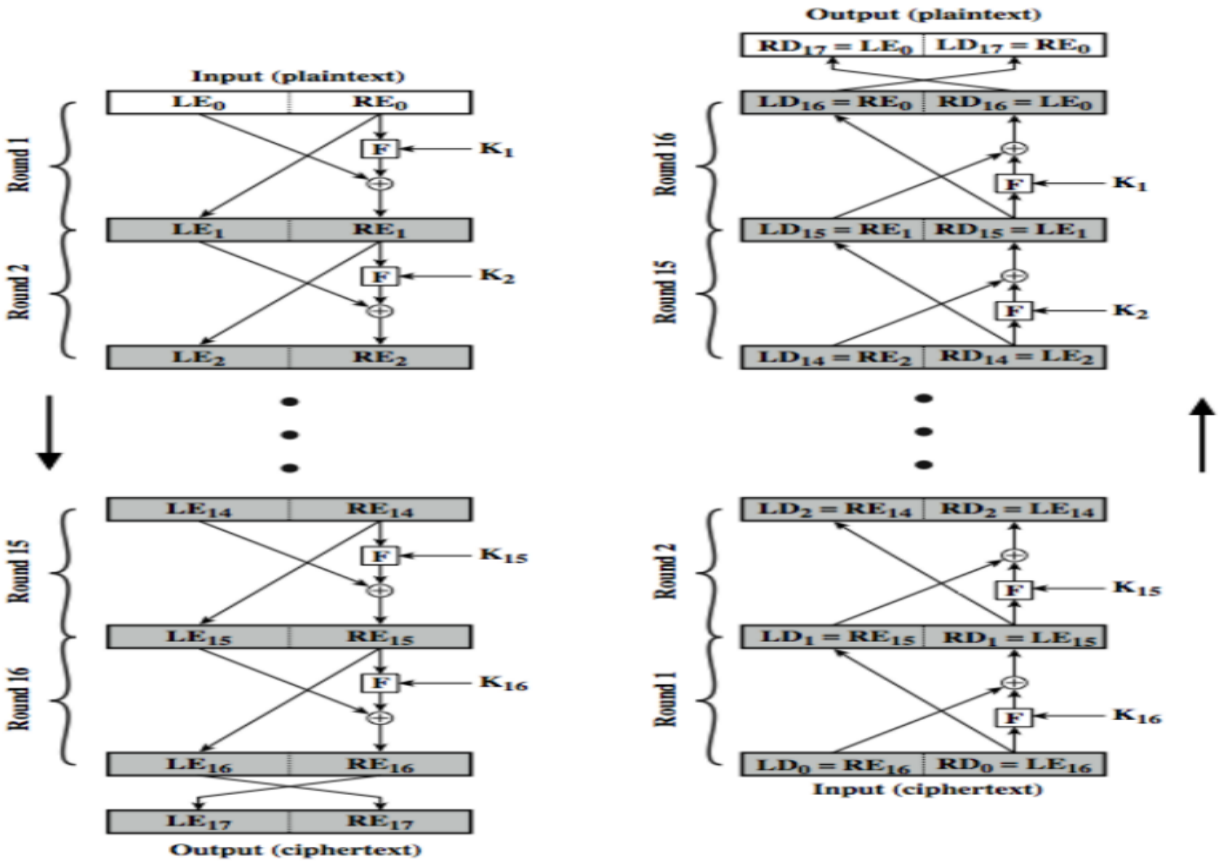
Confusion and Diffusion

- Cipher needs to completely obscure statistical properties of original message
- A one-time pad does this
- More practically Shannon suggested combining **S & P** elements to obtain:
- **Diffusion** dissipates statistical structure of plaintext over bulk of ciphertext
- **Confusion** makes relationship between ciphertext and key as complex as possible

Feistel Cipher Structure

- Horst Feistel devised the feistel cipher
  - based on concept of invertible product cipher
- partitions input block into two halves
  - process through multiple rounds which
  - perform a substitution on left data half
  - based on round function of right half & subkey
  - then have permutation swapping halves
- Implements Shannon's S-P net concept

Feistel Cipher Structure



Feistel Cipher Design Elements

- block size
- key size
- number of rounds
- subkey generation algorithm
- round function
- fast software en/decryption
- ease of analysis

Presentation Outline

1 Data Encryption Standard (DES)

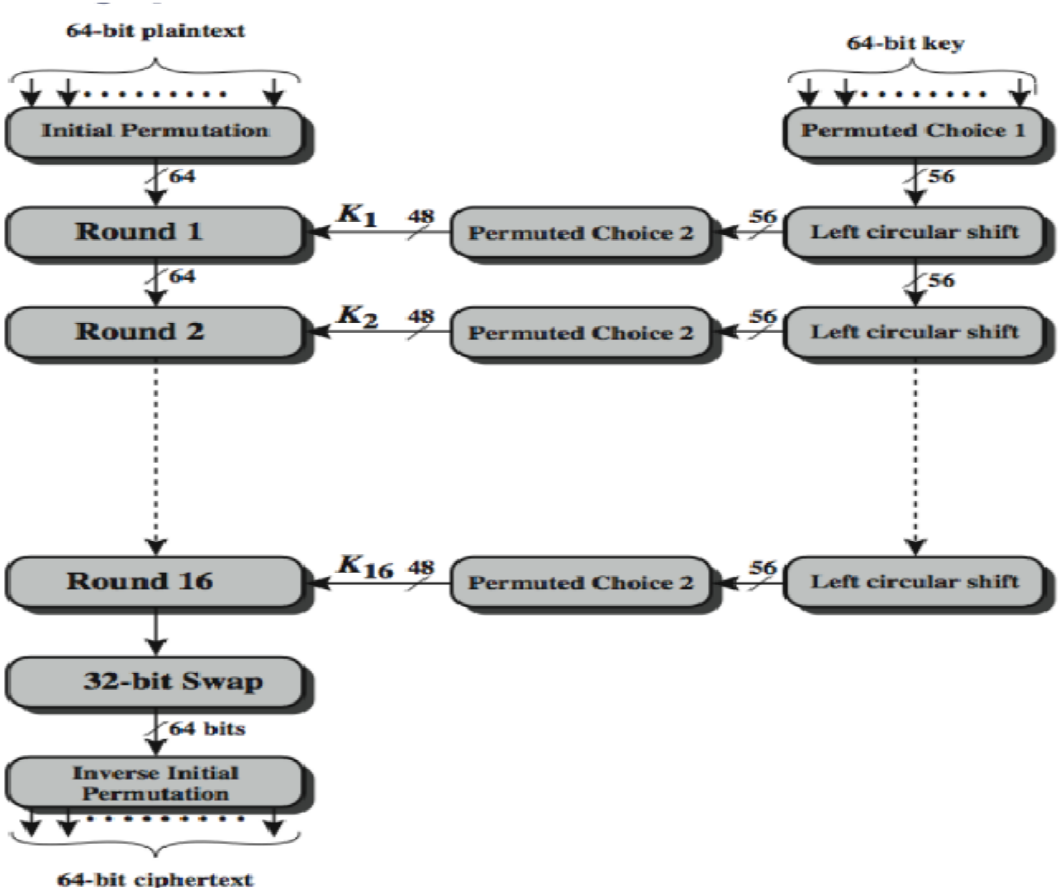
Data Encryption Standard (DES)

- most widely used block cipher in world
- adopted in 1977 by NBS (now NIST)
- encrypts 64-bit data using 56-bit key
- has been considerable controversy over its security
- IBM developed Lucifer cipher by team led by Feistel in late 60s
- used 64-bit data blocks with 128-bit key
- then redeveloped as a commercial cipher with input from NSA and others
- in 1973 NBS issued request for proposals for a national cipher standard
- IBM submitted their revised Lucifer which was eventually accepted as the DES

DES Design Controversy

- although DES standard is public
- was considerable controversy over design
  - in choice of 56-bit key (vs Lucifer 128-bit)
  - and because design criteria were classified
- subsequent events and public analysis show in fact design was appropriate
- use of DES has flourished
  - especially in financial applications
  - still standardised for legacy application use

DES Encryption Overview



Initial Permutation IP

- first step of the data computation
- IP reorders the input data bits
- even bits to LH half, odd bits to RH half
- quite regular in structure (easy in h/w)
- no cryptographic value

Encryption (IP, IP^{-1}

- IP

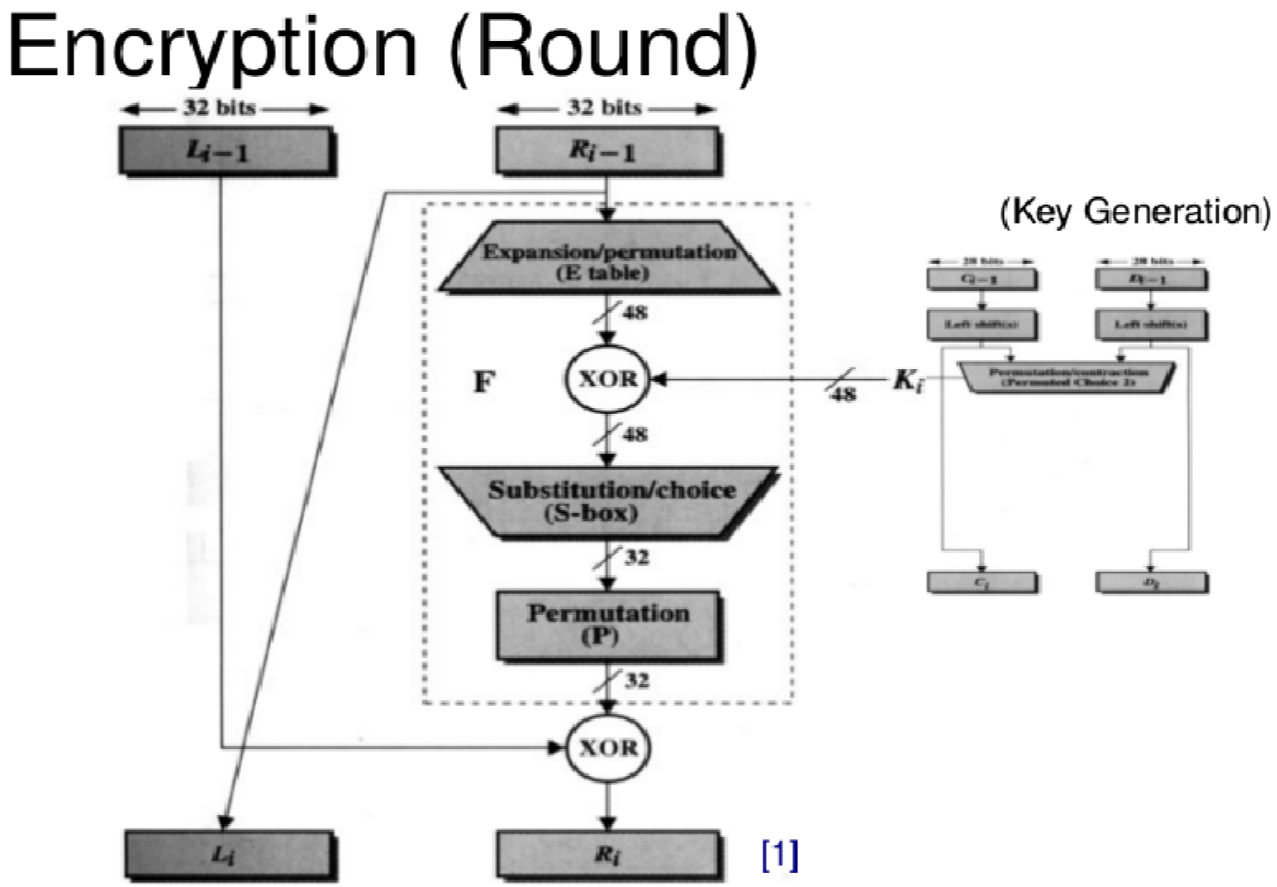
Bit	0	1	2	3	4	5	6	7
1	58	50	42	34	26	18	10	2
9	60	52	44	36	28	20	12	4
17	62	54	46	38	30	22	14	6
25	64	56	48	40	32	24	16	8
33	57	49	41	33	25	17	9	1
41	59	51	43	35	27	19	11	3
49	61	53	45	37	29	21	13	5
57	63	55	47	39	31	23	15	7

- IP^{-1}

Bit	0	1	2	3	4	5	6	7
1	40	8	48	16	56	24	64	32
9	39	7	47	15	55	23	63	31
17	38	6	46	14	54	22	62	30
25	37	5	45	13	53	21	61	29
33	36	4	44	12	52	20	60	28
41	35	3	43	11	51	19	59	27
49	34	2	42	10	50	18	58	26
57	33	1	41	9	49	17	57	25

- Note: $IP(IP^{-1}) = IP^{-1}(IP) = I$

DES - Round Operation

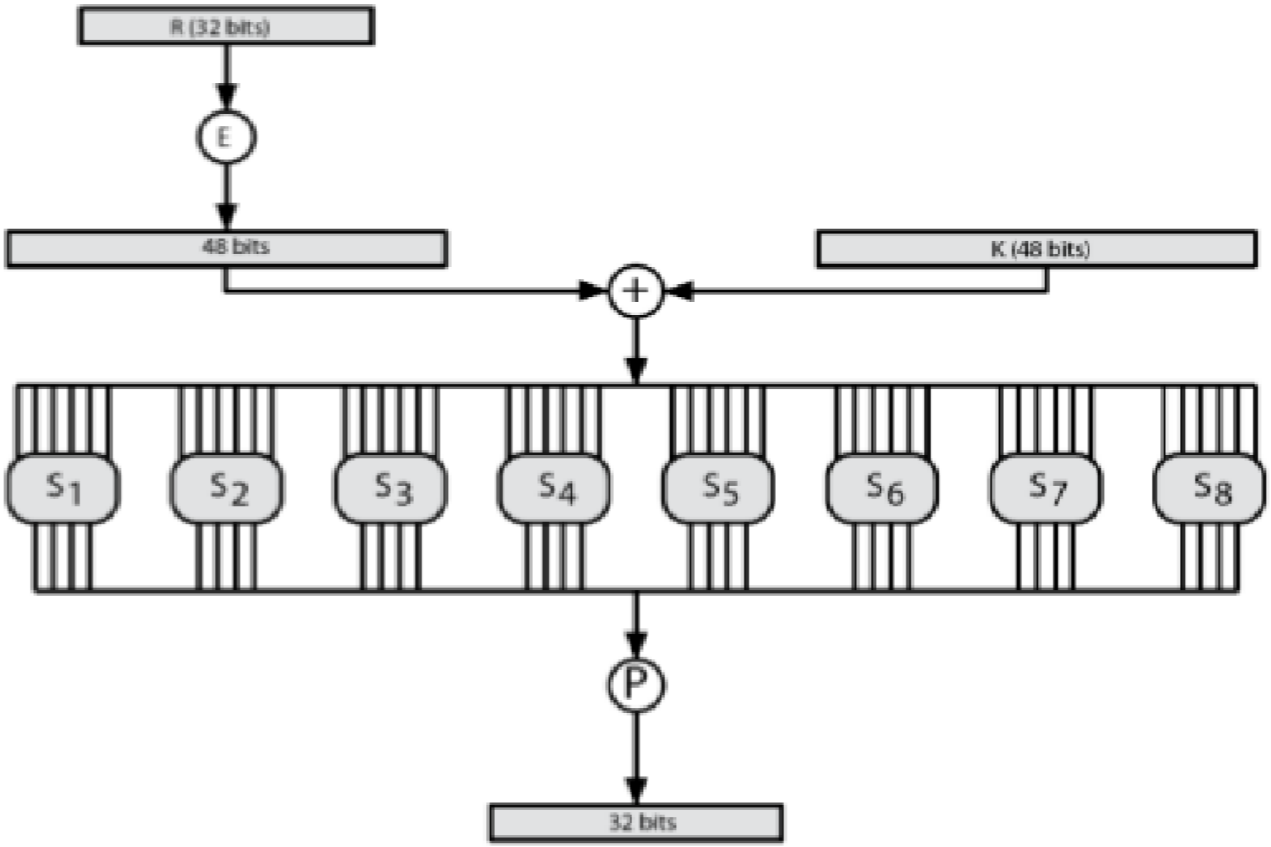


[1]

DES Round Structure

- uses two 32-bit L & R halves
- as for any Feistel cipher can describe as:
  $L_i = R_{i-1}$
  $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$
- F takes 32-bit R half and 48-bit subkey:
  - expands R to 48-bits using perm E
  - adds to subkey using XOR
  - passes through 8 S-boxes to get 32-bit result
  - finally permutes using 32-bit perm P

DES - Round structure



Substitution Boxes S

- have eight S-boxes which map 6 to 4 bits
- each S-box is actually 4 little 4 bit boxes
  - outer bits 1 & 6 (row bits) select one row of 4
  - inner bits 2-5 (col bits) are substituted
  - result is 8 lots of 4 bits, or 32 bits
- row selection depends on both data & key
  - feature known as autoclaving (autokeying)

Encryption (Round) (cont.)

- E

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	45	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

Expansion

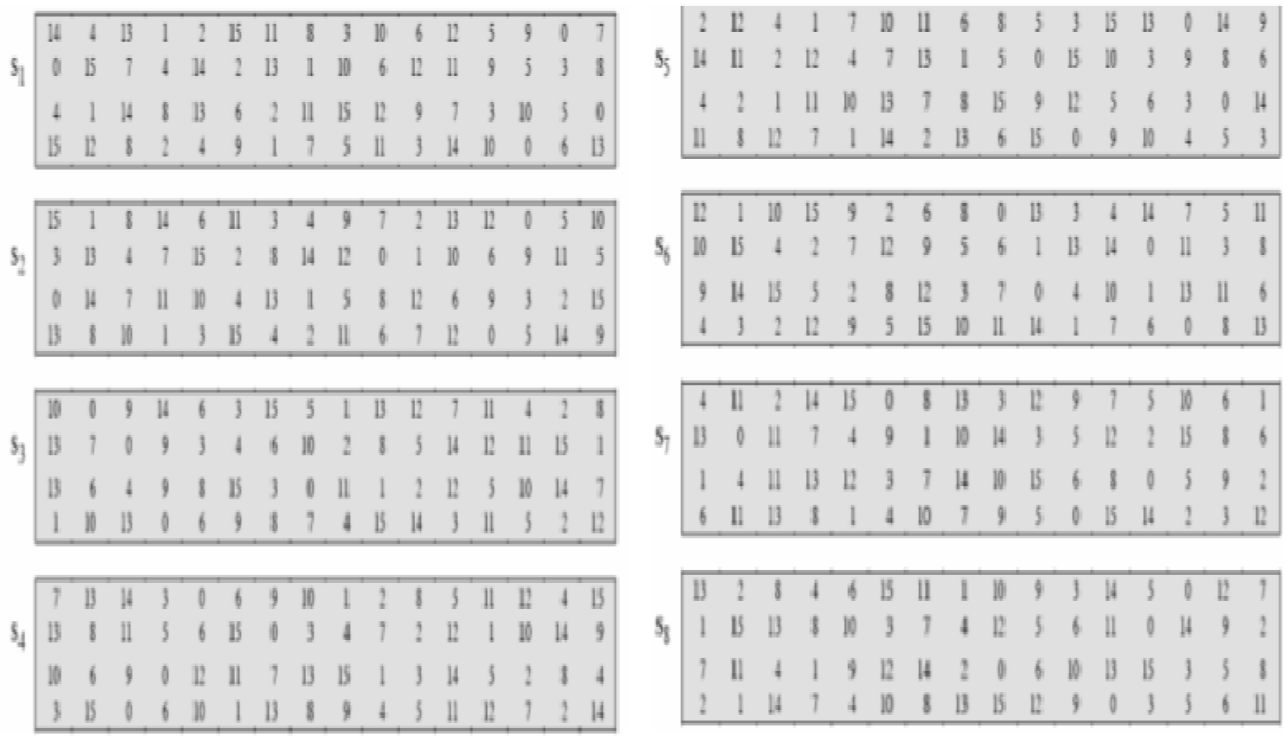
- P

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

Expansion

Encryption (Round) (cont.)

- S-box



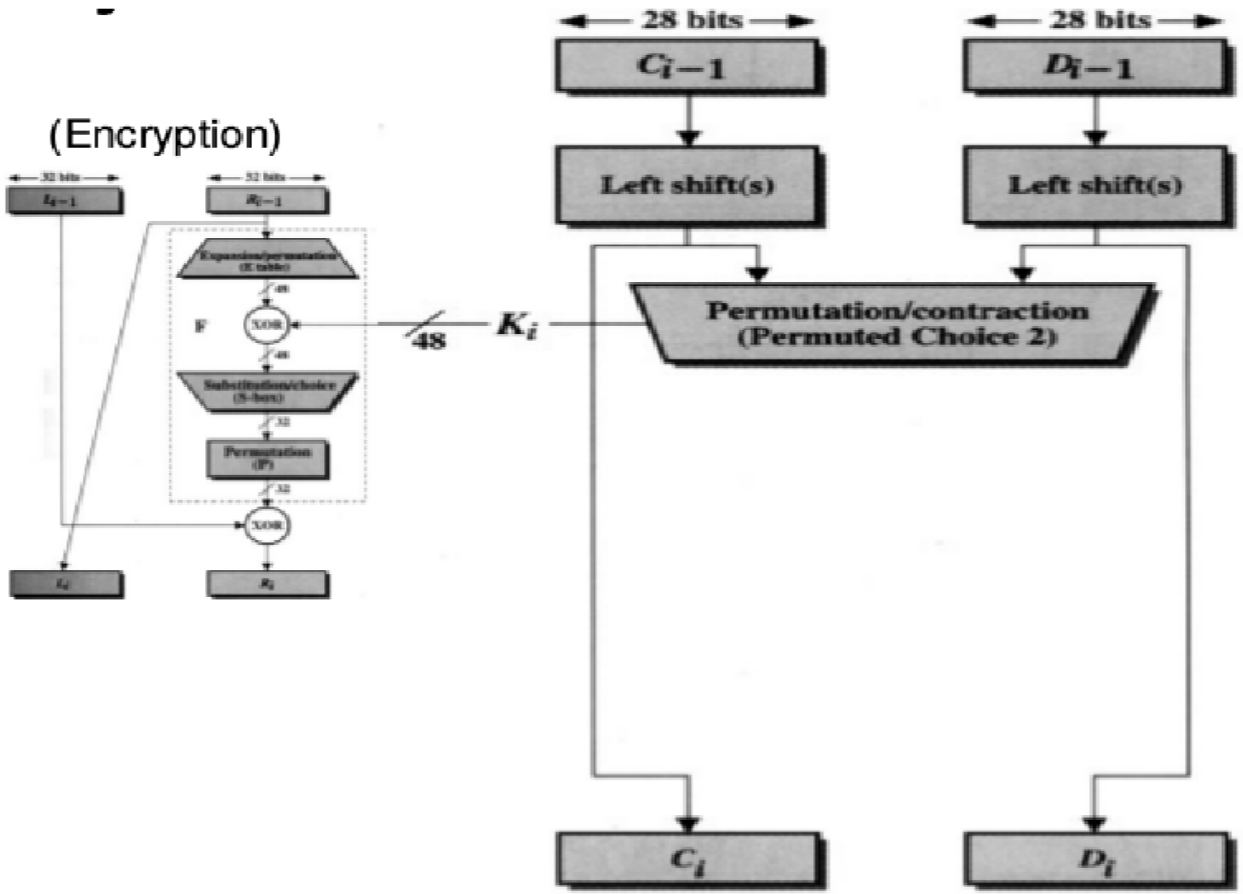
DES Key Schedule

- forms subkeys used in each round
  - initial permutation of the key (PC1) which selects 56-bits in two 28-bit halves
  - 16 stages consisting of:
    - rotating each half separately either 1 or 2 places depending on the key rotation schedule K
    - selecting 24-bits from each half & permuting them by PC2 for use in round function F

DES Decryption

- decrypt must unwind steps of data computation
- with Feistel design, do encryption steps again using subkeys in reverse order (SK16 ... SK1)
  - IP undoes final FP step of encryption
  - 1st round with SK16 undoes 16th encrypt round
  - 16th round with SK1 undoes 1st encrypt round
  - then final FP undoes initial encryption IP
  - thus recovering original data value

Key Generation



DES Key Schedule

- forms subkeys used in each round
  - initial permutation of the key (PC1) which selects 56-bits in two 28-bit halves
  - 16 stages consisting of:
    - rotating each half separately either 1 or 2 places depending on the key rotation schedule K
    - selecting 24-bits from each half & permuting them by PC2 for use in round function F

Avalanche Effect

- key desirable property of encryption alg
- where a change of **one input** or key bit results in changing approx **half output bits**
- making attempts to "home-in" by guessing keys impossible
- DES exhibits strong avalanche

Strength of DES Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
- brute force search looks hard
- recent advances have shown is possible
  - in 1997 on Internet in a few months
  - in 1998 on dedicated h/w (EFF) in a few days
  - in 1999 above combined in 22hrs!
- still must be able to recognize plaintext
- must now consider alternatives to DES

Strength of DES Analytic Attacks

- now have several analytic attacks on DES
- these utilise some deep structure of the cipher
  - by gathering information about encryptions
  - can eventually recover some/all of the sub-key bits
  - if necessary then exhaustively search for the rest
- generally these are statistical attacks
  - differential cryptanalysis
  - linear cryptanalysis
  - related key attacks

Strength of DES Timing Attacks

- attacks actual implementation of cipher
- use knowledge of consequences of implementation to derive information about some/all subkey bits
- specifically use fact that calculations can take varying times depending on the value of the inputs to it
- particularly problematic on smartcards

Summary

Topics discussed:

- block vs stream ciphers
- Feistel cipher design & structure
- DES - working & strength
- block cipher design principles