L2: Classical Encryption Techniques

2.2 Substitution Techniques

Caesar cipher: e.g.: replace each letter three places down

Plain: meet me after toga party

ciper: PHHW PH DIWHU WKH WRJD SDUWB

$C = E(3,P) = (P+3) \bmod 26$ // P = plain text
C = ciper //

$= E(K,P) = (P+K) \bmod 26$ // $1 \le K \le 25$

$P = D(K,C) = (C-K) \bmod 26.$

Brute-force cryptanalysis is easily performed.

Try with all 25 possible keys.

Soln: use large key space.

e.g.: 168-bit key for triple DES ($= 2^{168}$ possible keys)

Mono-alphabetic ciphers

cipher can be any permutation of 26 letters.

Then 26! ($= 4 \times 10^{26}$ possible keys, which is 10 times greater than DES) possible keys are there!

example page 37: The ~~cy~~ cipher given

WZQS ---

--- MQ

A powerful tool is - digrams.

Mono-alphabetic ciphers are easy to break ~~as~~ ~~for~~ ~~refute~~ as they reflect freq of data of the original alphabet.

Playfair cipher Soln: letter e could be replaced with 16, 74, 35, 21 - homophone.

Still breakable!

Playfair Cipher: multiple letter encryption cipher

1854, WWII

Your message: SECRET MESSAGE

SE CR ET ME SX SA GE

--double S er jonno S likhe then X likhbo

Encoding

cipher: ~~NORR~~ NO RD KU NK QZ PC ND

K	E	Y	W	O
R	D	A	B	C
F	G	H	I/J	L
M	N	P	Q	S
T	U	V	X	Z

Rules:
1. Split into pairs
2. for duplicated letters, insert X
3. for odd letter at end, insert x, ignore spa

First each pair into a seperate table

1. If in same column
   → Move each letter down ONE
   → upon reaching end of table wrap around.

2. if in same row.
   → Move each letter right ONE
   → upon ------, wrap around.

3. if it forms rectangle
   → swap the letters with ones on the end of the rectangle.

SE: ~~for~~ rule 3: S becomes N, E becomes O,

CR: rule 2: C = R, R = D

ET: rule 3: E = K, T = U

ME: rule 3:

SX: rule 3:

SA: rule 3:

GE: rule 1: G become N, E becomes D

26x26 = 676 digrams → a bit difficult. freq analysis is much more difficult.

Polyalphabetic Ciphers

Vigenere cipher: (ভিজেনিয়র সাইফার)

$K_0, K_1, K_2, \ldots, K_{m-1}$	Key ($m < n$)	3	4	2	4
$P_0, P_1, \ldots, P_{n-1}$	plaintext	22	4	0	17
$C_0, C_1, \ldots, C_{n-1}$	ciphertext	25	8	2	21

$$C = C_0, C_1, \ldots, C_{n-1} = E(K, P) = E[(K_0, \ldots, K_{m-1}), (P_1, \ldots P_{n-1})]$$

$$= (P_0 + K_0) \bmod 26, (P_1 + K_1) \bmod 26, \ldots (P_{n-1} + K_{m-1}) \bmod 26$$

$$C_i = \left(P_i + K_{i \bmod m}\right) \bmod 26.$$

Why?

Assignment: Make freq table for Bengali letters. graph

Letter freq information is obscured!

However, not all plaintext structure lost.

Assignment: Cryptanalysis of all ciphers

use non-repeating keywords → See book

Vernam Cipher: 1918, AT&T

Works on binary data.

$$C_i = P_i \oplus K_i \quad // \text{ compare eqn 2.3 of vegenere cipher}$$

$$P_i = C_i \oplus K_i$$

Plain Text = HELLO

Key : DGHBe

One-Time-pad: giving randomness!

Sender send a non-repeating numbers to rcvr.

↳ use RSA for secure Tx.

It works with same principle (as caesar cipher) — i.e. shifting letters. But with different key for each one!

8	19	13	4	23	20
11	10	41	----		

ONE-TIME-PAD

--doc

HAPPY

8 1 16 16 25

+8 +19 +13 +4 +23

16 20 29 20 48

= 3 (mod 26) = 22 (mode 26)

P T C T V

⇒ PP maps to CT!

Note ⇒ TT maps for A P!

Once used, never re-use the same key!

Hence, pattern like letter freq is no more! Unbreakable

P ⇒ (P-8) (T-19) (C-13) (T-4) (V-23)

⇒ H A P P Y

Weakness:

→ must keep the key!
→ The keys should long enough (at least same as the plain text) [length limitation]
→ Time consuming
→ Exchange of one-time-pad.
→ OTP must be potential random.

But random numbers generated by computer follow a pattern!

to Agon for website
1782=

Two powerful properties:

1. Shift/key is random
2. which is uniform freq distr.

What makes it so powerful?

1. ALICE → Caesar cipher needs only 26 possibilities using brute-force search.

Assume ALICE - every letter is shifted different letters by one-time-pad.

Hence 26×26×26×26×26 = 12million possible situation. Very hard to find.

The encrypted letter is equally likely distributed in 12m possibilities!

2.3: Transposition Techniques

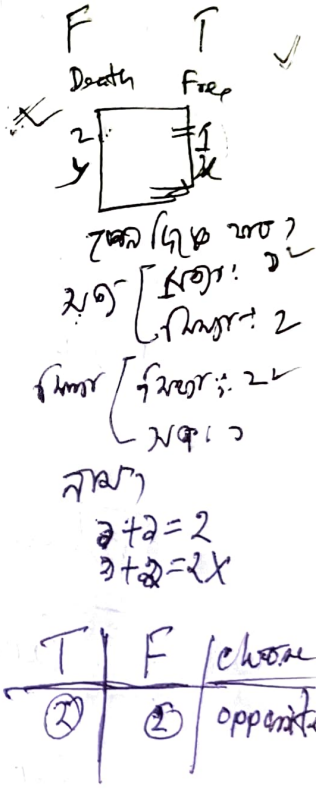
1. Re-arrange, No substitution.

rail fence (meet me after the toga party ই, with depth 2)

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

See Book page 49

vulnerable, not good.



Row Transposition Cipher:

key → Unique numbers from 0-9.

Key:	4	3	1	2	5	6	7
Plain Text:	a	t	t	a	c	k	p
	o	s	t	p	o	n	e
	d	u	n	t	i	l	t
	w	o	a	m	X	Y	Z

dummy chars.

cipher: start lowest column to highest in order.

ttna, aptm, tsuo, aodw, coix, knly, petz.

Easy to ~~nee~~ break!

do more than one stage of transposition.

See in book example.

Rotor Cipher

A rotor is electro-magnetic m/c to encrypt and decrypt. Famous → German Enigma

Rotor III	Rotor II	Rotor I
A → C	A → D	A → E
B → D	B → E	B → F
C → E	C → F	C → G
D → F	D → G	D → H
		E → I
		F → J
		G → K

CD FG JK

Plain Text = AB

become → CD → FG → JK

cipher ⇒ JK

Strength of encryption depends on ⇒

- → no. of rotors in system
- → size of each rotor
- → no. of rotor types.

Explain **Enigma Machine – Project** → can be a project → simple but ingenious invention

Youtube : 1. The inner Workings of an Enigma Machine.

1. Pringles can Enigma machine

Enigma m/c was an encryption device in WWII by German.

It had 158 quintillion combinations and it was tough to be unbreakable (158 quintillion)

See your tube: But Building Enigma machine using pringles tube.

youtube: Enigma II Encryption Machine & Puzzle - encode and decode cipher

OFB Meg
CGB - chairman
- ex CGB II
- cfg chairman
→ Saif, 0
→ Salahuddin, 4
→ Rasaij, 18
→ Hayat, 4
→ Kamal Hayat
→ Md Ali, 6
→ Nizam Selim, 4
principle

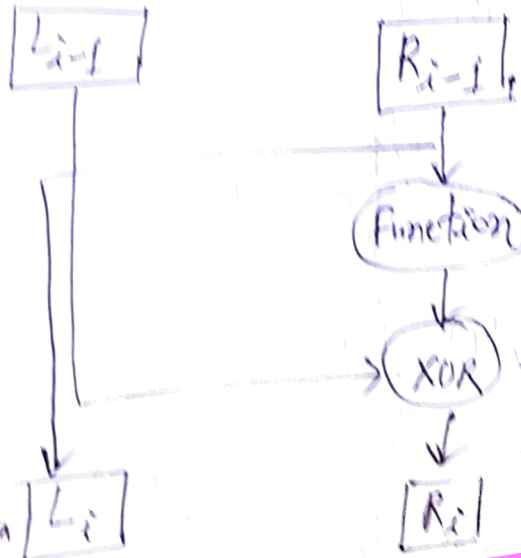
What change?

Chapter 3: Block Ciphers, DES

Feistal cipher/structure → block ciphers.

plain Text, P_i is divided into halves.

P_i :



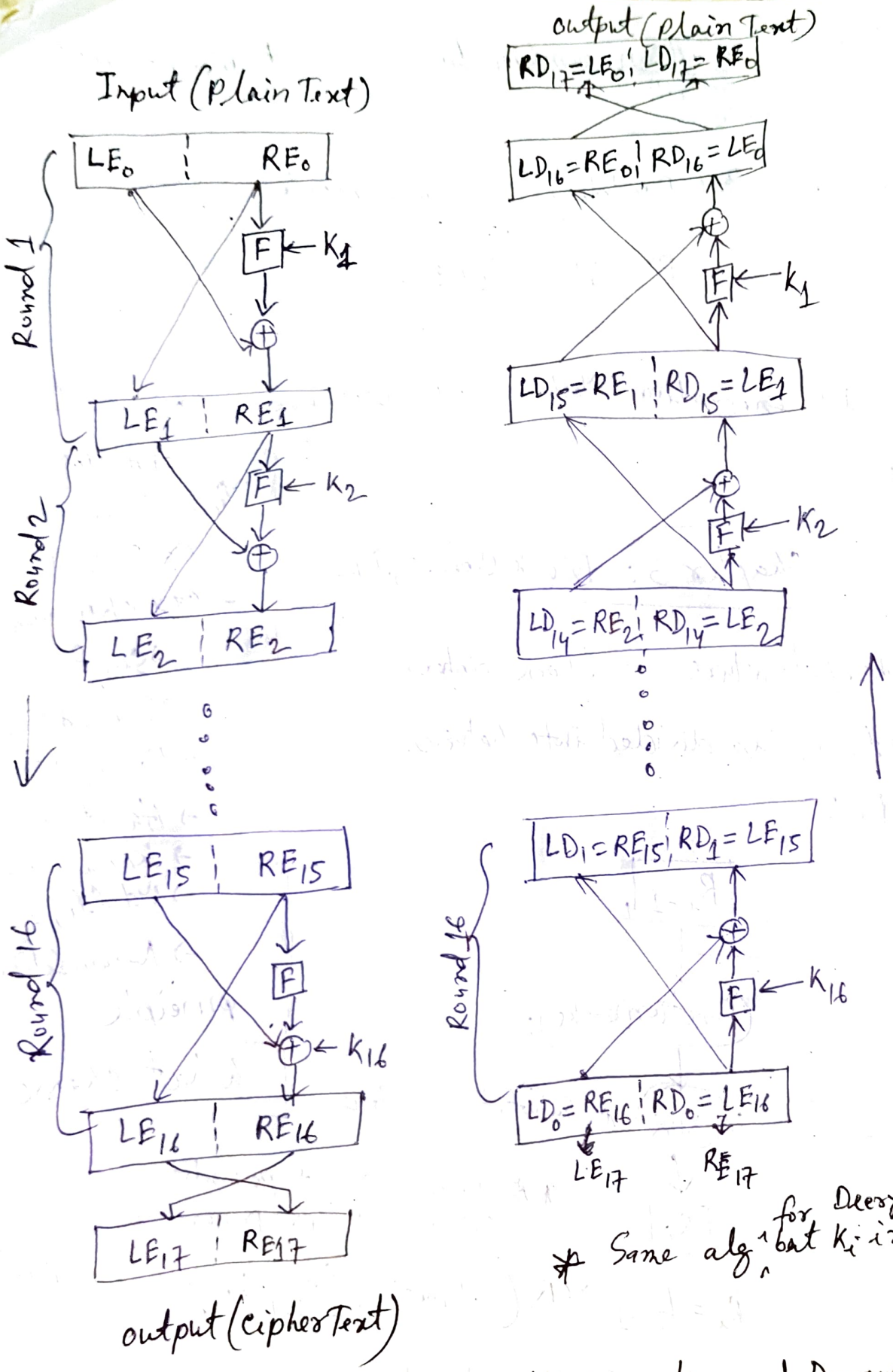
** substitution and permutation from Substitution Net (SPN) proposed by shannon in 1945

$L_i = R_{i-1}$

$R_i = L_{i-1}$ XOR (function (R_{i-1}, key_i))

Fig: 1 round for feistal structure.

Parameters: 1. No. of rounds (e.g. 16) 2. Block size (16 bytes)
3. Key (e.g. 128 bits) 4. Subkeys for each round



* Same alg. for Decryption but K_i in reverse order

figure 3.3 Feistel Encryption and Decryption (16 rounds)

At 16-th round

Encryption:

$LE_{16} = RE_{15}$

$RE_{16} = LE_{15} \oplus F(RE_{15}, K_{16})$

Decryption:

$LD_1 = RD_0 = LE_{16} = RE_{15}$

$RD_1 = LD_0 \oplus F(RD_0, K_{16})$

$= RE_{16} \oplus F(RE_{15}, K_{16})$

$= [LE_{15} \oplus F(RE_{15}, K_{16})] \oplus F($

The XOR has the following properties:

$[A \oplus B] \oplus C = A \oplus [B \oplus C]$

$D \oplus D = 0$

$E \oplus 0 = E$

Lecture 5: Data Encryption Standard (DES) (Youtube)

1st Part: DES Intro

Few facts:

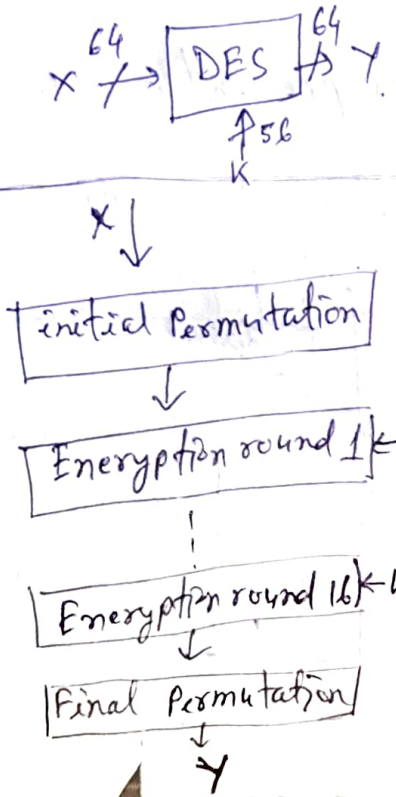
→ 1974 proposed by IBM with input of NSA.

→ Big event in history of cryptography.

→ 1977–1998 US standard, 80% of world used it. (e.g. e-passport with 3DES)

→ unsecure today (key too short) but 3DES is very secure.

How it works?



R. How do we build a block cipher?

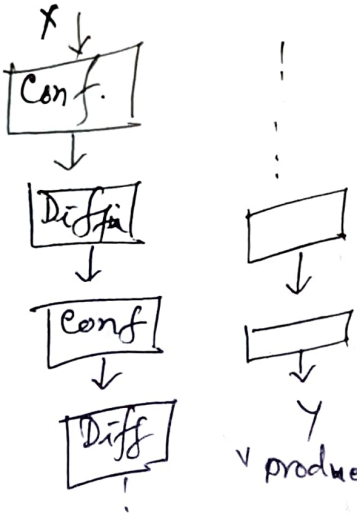
Shannon → proposed two atomic operations.

i) Confusion: Relationship between plain and cipher text is obscured.
e.g: Substitution table (caeser cipher)

ii) Diffusion: The influence of one each plaintext bit is spread over many cipher text bits.
e.g: Permutation

Combine confusion and diffusion many

many times to build a strong block cipher.



x_1 = 0010 1001 → block cipher → y_1 = 1011 1001

x_2 = 0000 1011 → block cipher → y_2 = 0110 1100

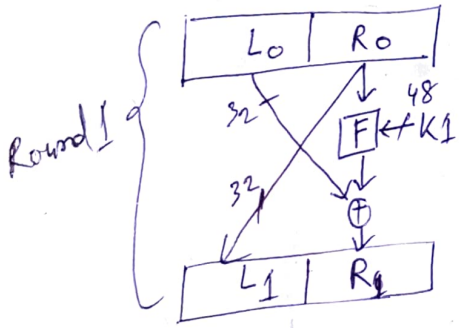
single bit flip — many bit flip (diffusion)

Most block ciphers ~~are~~ are product cipher, as they consists of rounds

2nd Part: Feistal Network

many of today's ciphers are feistal ciphers. (not all)

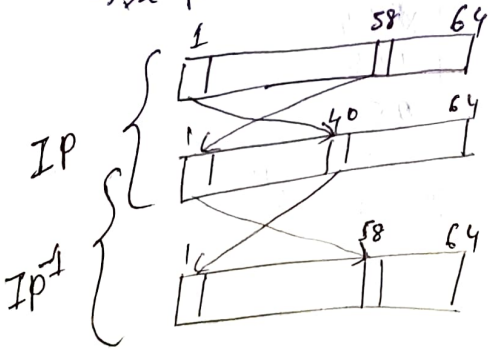
* See fig in prev pages of feistal structure



3rd Part: DES internals

a) IP and IP^{-1} (Initial Permutation)

→ simple bit permutation.



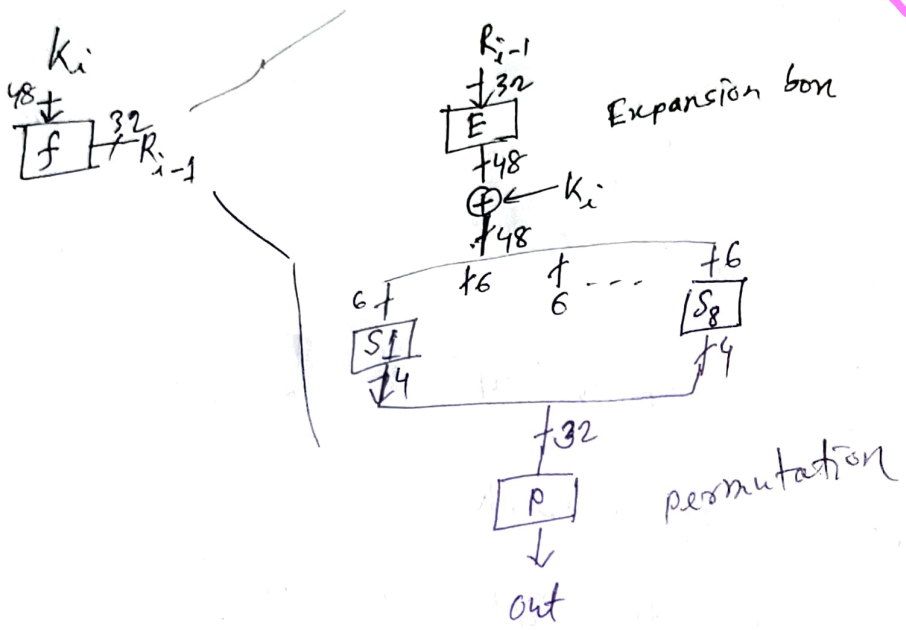
$IP(IP^{-1}) = I$ (original)

Why IP, IP^{-1} needed?

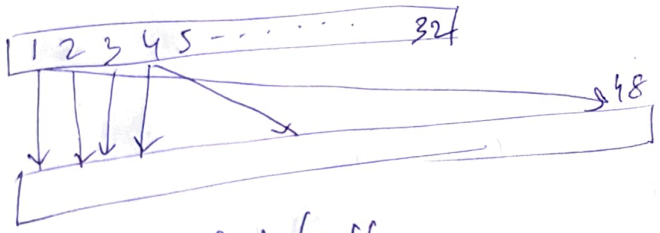
for electrical engg purpose. to integrated in H/w ckt. in s/w it cause slow down, critics ~~&~~ say IBM wanted to sell H/w.

DES have permutation in every round, no need to have IP!

36: Details of the F-function



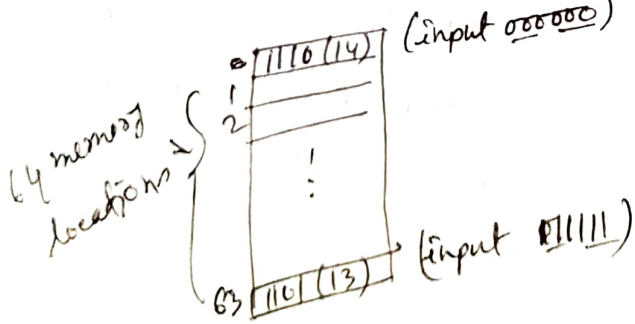
E-box: it provides ~~confusion~~ diffusion.



cross-wire !!

16 connected to 16 -
other 16 " to (16+16)

S-box: it is the heart of DES (it provides confusion).



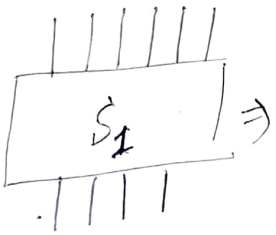
decoded from S-box

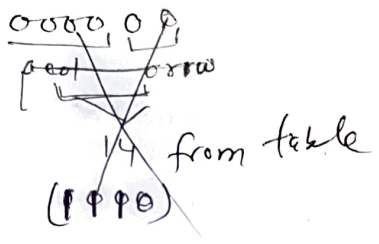
$2 - 2^1$
$4 - 2^2$
8
16
32
64
128
256
512
$1024 - 2^{10}$

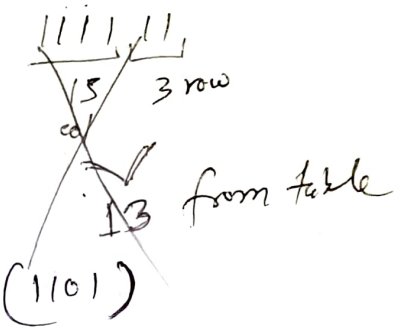
unusual decoding of S-box tables

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	03	10	6	12	5	9	0	7
1																
2																13
3	15	12	8	2												

↑ S-1 table

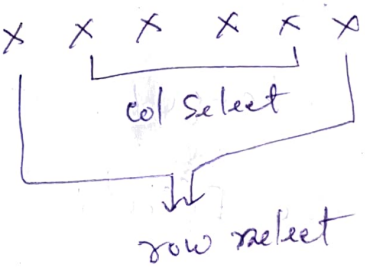






* unusual decoding of S-box tables

input bits



Example: what is S_1-box representation of 37?

$$S_1(37) = S_1(100101) = 08 = 1000$$

col 2

row 3

Why S-box? → IBM says for security. There are 8-tables for 8 S-boxes. Very hard to break.

- 1997-1997 broke by differential cryptanalysis.
- IBM, NSA knew that the attack coming before researcher.

4-Steps in DES

1. Expansion E
2. XOR with round key
3. S-box substitution
4. Permutation.

Q:: if 1 bit flips How many change in output?

- each of above 4-steps shows that lot of changes in a round.
- And that's done for 16 round!
- lot of change!
- Avalanche Effect:

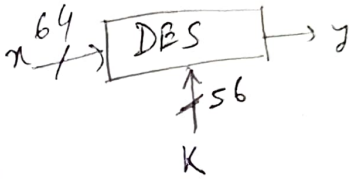
A small change in either the plain text or the key should produce a significant change in the cipher text.

(A change in 1 bit in plaintext should produce a change in many bits of the cipher-text)

Today:

1. key schedule
2. DES decrypt
3. DES security
4. DES Alternatives.

DES Key schedule:



Q. How to compute 16 subkeys $K_1 \dots K_{16}$?

See fig 3.14 of Paar Book.

Key schedule consists of simple operations.

a) PC-1, Permuted Choice-1

⇒ Drops bits 8, 16, 24, 32, 40, 48, 56, 64

⇒ Effective key lenght of DES : 64-8 = 56

b) LS_i :

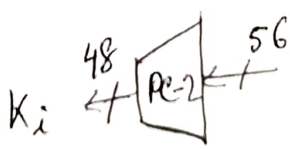
Left shift (in fact: left rotate) for each halves.

$$LS_i = \begin{cases} 1 \text{ pos shift}, & i = 1, 2, 9, 16 \\ 2 \text{ " " }, & \text{all other } i = 3, 4, \dots \end{cases}$$ (i = round no.)

Total number of bit position shift = $1 \cdot 4 + 2 \cdot 12 = 28$

⇒ $C_{16} = C_0$, $D_{16} = D_0$

c) PC-2, permuted choice-2

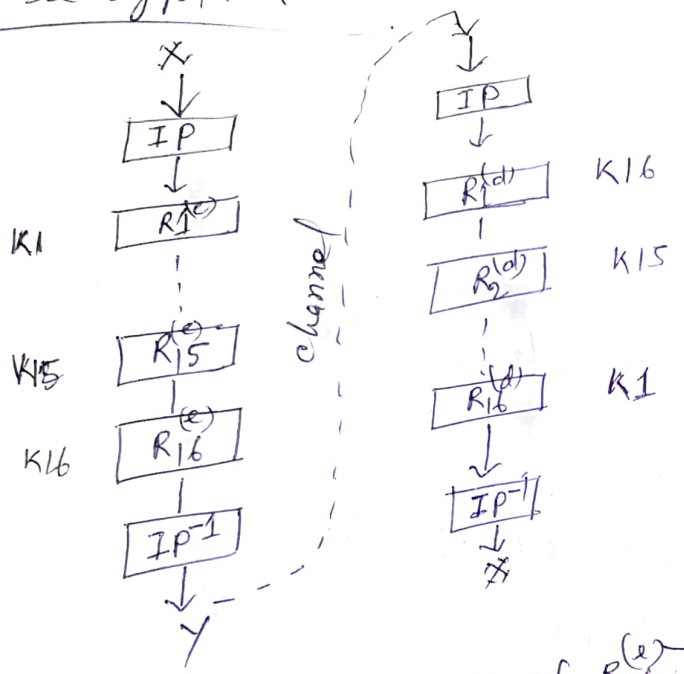


8 bits are dropped, remaining 48 input bits are permuted.

Observation:

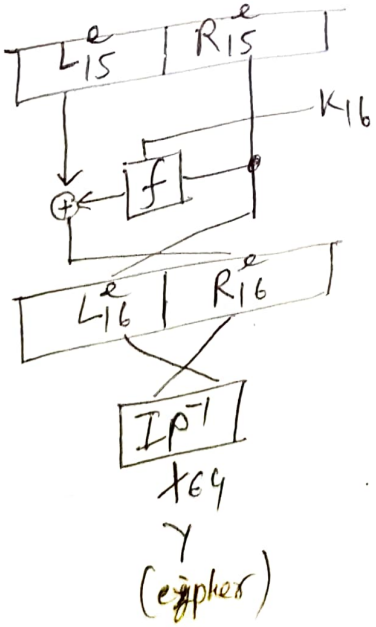
1. Each key $K_1, K_2, \ldots K_{16}$ is merely a permutation of the original 56 key bits.
2. Key schedule is easy to implement hardware.

2. Decryption

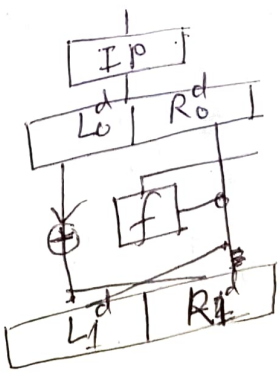


Note: $R_1^{(d)}$ is inverse of $R_{16}^{(e)}$
$R_{16}^{(d)}$ " " of $R_1^{(e)}$
* Reverse a round?

end of encryption.



beginning of Decryption



To show
$(L_1^d = R_{15}^e) \wedge (R_1^d = L_{15}^e)$

from figure, it is clear that
$L_1^d = R_{15}^e$

$R_1^d = \underline{L_0^d} \oplus f(K_i, R_0^d)$

L_0^d → is at left side
→ is at right " of IP
→ is " left " at IP^{-1}
→ is at right side $\Rightarrow (R_{16}^e)$
→ $L_{15}^e \oplus f(R_{15}^e, K_{16})$

$= L_{15}^e \oplus f(K_{16}, R_{15}^e) \oplus f(K_{16}, R_{15}^e)$

$= L_{15}^e \oplus 000\ldots0000$ (32 zero)

$= L_{15}^e$ #

General:
$L_i^d = R_{16-i}^e$ and $R_i^d = L_{16-i}^e$

at end of decryption:
$IP^{-1}(R_{16}^d, L_{16}^d) = IP^{-1}(L_0, R_0) = IP^{-1}(IP(x)) = x$

The remaining round reversals work the same:

Round 2^d reverses round 15^e
⋮
Round 16^d " " 1^e

3. DES Security:

Two families of attack

a) Analytical Attacks (until 1999 it was non-breakable)

Differential cryptanalysis : requires 2^{47} (x, y) pairs.

Linear " : " 2^{43} (x, y) pairs.

needs TB stands!

b) Brute-force attack:

given (X_0, Y_0)

$DES_{K_i}^{-1}(Y_0) \stackrel{?}{=} X_0$

$i = 0, 1, \ldots, 2^{56}-1.$

1998 - Deep Special-purpose DES hardware cracker
$250,000

2007 : COPACOBANA - $10,000

⇒ DES can be broken in few days.

4. DES Alternatives

Cipher	comment
AES	De facto world standard
3DES	Still very secure
AES-Finalists	

Hardware: Permutations such as E, P, IP and IP^{-1} are very easy to implement in H/W as they only require wiring. but no logic. S-box (6 to 4) also easily ~~refit~~ realizable in H/W. One S-box requires about 100 logic gates.

A single DES round can be done with 3000 gates. On Modern ASICs and FPGA [Application-Specific ICs, Field Programmable Gate Arrays)

⇒ Throughput rate 100 Gbit/sec possible.

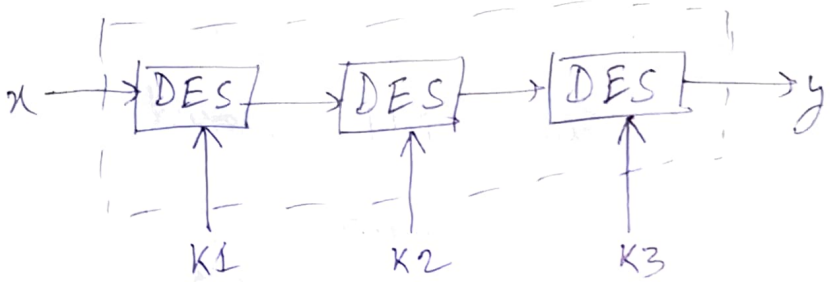
⇒ 3000 gates even fit on low cost RFID chip!

Software: E and P permutations are slow in software. S-box also slower.

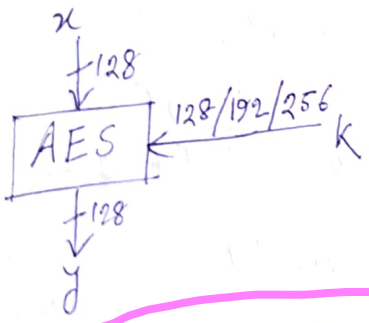
Triple DES (3DES):

consists of three subsequent DES encryption.

$$y = DES_{K3}\left(DES_{K2}\left(DES_{K1}(x)\right)\right)$$



AES:



All internal op are based on Finite Field

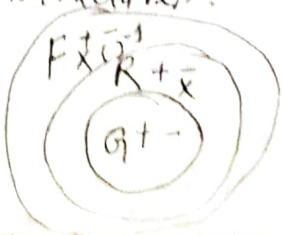
Introduction to Finite Fields:

02 Dec 2020

Initial, fee - 10pe chick (1-1.2kG), Separate packet consisting Chicks, Kaliya etc. Legs all in a packet. Shak, if available

Terminology: Finite Field (= Galois Field)

3 basic algebraic structures:



G: Group (+ −)

R: Ring (+ − ×)

F: Field (+ − × ÷)

See 4.3.2 of Book.

informally: "A field is a set of numbers in which we can add, subtract, multiply and divide".

Example: Real numbers, complex numbers, natural numbers.

- in crypto we always need finite sets.

Theory 4.3.1: Finite field only exist if they have p^m elements. (p is prime, m is integer)
(p smallest = 2, m smallest = 1)

Example:

i) There is FF with 11 element. GF(11)
ii) " " " " 81 " : GF(81) = GF(3^4)
iii) " " FF " 256 " : GF(256) = GF(2^8) ↑ "AES field"
iv) " " " " 12 = $2^2 \cdot 3$, NO FF

used in web browser...

Types of FF

GF(p^m)
- $m=1$ → GF(p) prime fields
- $m>1$ → GF(p^m) Extension Fields

imp In Crypto: GF(2^m)

3. Prime Field Arithmetic

The elements of a prime field, GF(p) are the integers $\{0, 1, \ldots$

a) Add, subtract, Multiply:

Let $a, b \in GF(p) = \{0, 1, \ldots, p-1\}$

$a + b \equiv c \bmod p$
$a - b \equiv d \bmod p$
$a \cdot b \equiv e \bmod p$

→ { $a+b$ may make the elements out of range but $(a+b) \bmod p$ make sure that it with range of $(p-1)$

Note that all conditions of fields ~~set~~ are satisfied with these computations.

b) Inversion:

$a \in GF(p)$

the inverse a^{-1} must satisfy $a \cdot a^{-1} \equiv 1 \bmod p$.

How to compute a^{-1}?

Ans: Extended Euclidean Alg.

4. Extension Field $GF(2^m)$ Arithmetic:

~~Arithmetic~~

a) Element representation:

The elements of $GF(2^m)$ are polynomials: ~~$a_{m-1}x^{m-1}$~~

$$a_{m-1}x^{m-1} + \cdots + a_1x^1 + a_0 = A(x) \in GF(2^m)$$

Note: AES, m is 8, Hence the ~~of~~ co-efficient is: $a_0, a_1, a_2 \cdots a_7$.

$a_i \in GF(2) = \{0, 1\}$

Example: $GF(2^3) = GF(8)$

$$A(x) = a_2x^2 + a_1x^1 + a_0x^0$$

$$= a_2x^2 + a_1x + a_0 \quad (a_0, a_1, a_2 \text{ all have elements } \{0,1\})$$

$= (a_2, a_1, a_0)$ 3 bit vectors, each may 0 or 1 $\Rightarrow$ 8 combinations.

$GF(2^3) = \{0, 1, x, x+1, x^2, x^2+1, x^2+x, x^2+x+1\}$

$000 \to 0$, $001 \to 1$, $010 \to x$, $011 \to x+1$, $100 \to x^2$, $101 \to x^2+1$, $110 \to x^2+x$, $111 \to x^2+x+1$

Quest: How to compute with these elements?

b) Addition and Subtraction in $GF(2^m)$

[Def 4.3.3]

⇒ use regular polynomial add or subtraction, where the co-efficients are computed in $GF(2)$.

Example: $GF(2^3)$

$GF(2) = \{0, 1\}$

$2^0 = 1$

$2^1 = 2 \bmod 2 = 0$

$GF(3) = \{0, 1, 2\}$

$2^0 = 1$

$2^1 = 2$

$2^2 = 4 \bmod 3 = 1$

$A(x) = x^2 + x + 1$

$B(x) = x^2 + 1$

$A + B = (1+1)x^2 + x + (1+1)$

see Why

$= 0 \cdot x^2 + x + 0$ ($2 \bmod 2$ is 0)

$= x$

Note: Add and Sub in $GF(2^m)$ are the same operations.

c) Multiplication in $GF(2^m)$

Intuition: Just do regular polynomial ~~mult~~ mult.

Example: $GF(2^3)$

$A \cdot B = (x^2 + x + 1)(x^2 + 1)$

$= x^4 + x^3 + x^2 + x^2 + x + 1$

$= x^4 + x^3 + (1+1)x^2 + x + 1$ // $2 \bmod 2$ is 0.

$= x^4 + x^3 + x + 1$, but it is not in the field!

$= c'(x)$

recall prime fields

Ex: $GF(7) = \{0, 1, \dots, 6\}$

$3 \cdot 4 = 12 \equiv 5 \bmod 7$ // modulo reduction

Solution: Reduce $c'(x)$ modulo a polynomial that "behave like a prime".

These are called irreducible polynomials.

Irreducible polynomial for $GF(2^3)$

$p(x) = x^3 + x + 1$

$$\overbrace{(x^4 + x^3 + x + 1)}^{A.B} \Big/ \overbrace{(x^3 + x + 1)}^{p(x)} = x + 1$$

$x^4 \quad +x^3+x$

$x^3 + x^2 + 1$

$x^3 + x + 1$

$\boxed{x^2 + x} \equiv A.B \bmod p(x)$

for every field $GF(2^m)$, there are several irreducible polynomials.

Ex: $p(x) = x^3 + x^2 + 1$

The "AES irreducible polynomial"

$p(x) = x^8 + x^4 + x^3 + x + 1$

d) Inversion of $GF(2^m)$

The inverse $A^{-1}(x)$ of an element $A(x) \in GF(2^m)$ must satisfy:

$$A(x)\, A^{-1}(x) \equiv 1 \bmod p(x)$$

↓

Ext Euel. Alg.