SPOOL : Simultaneous Peripheral Operations on Line

Information security

OSI → open system interconnections Architecture

1. Security Attacks
2. Security Services
3. Security Mechanism.

→ Active Attack
→ Passive Attacks → can just read or copy.
→ Attacker can modify content.

Active Attacks

- Masquerade → pretends as sender. (e.g. msg)
- Replay → attacking. multiple receivers by copying similar message
- Modification of msg.
- DOS

Passive

- Release of content message → leaks msg
- Traffic analysis → analyse how data is being send no of bytes sent or recieved.

Security Services

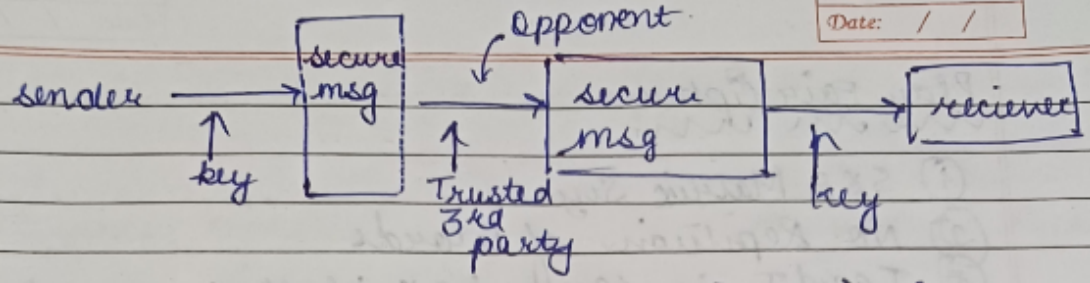
- Authentication
- Access Control
- confidentiality
- Integrity → No modification in data. Assurity.
- Non repudiation

Security Mechanism (Network Model)

↓ Plain Text, Cipher Text, Public key, Private key

Mediator.

0	1	2	3	4	5	6	7	8	9	10	11	12
a	b	c	d	e	f	g	h	i	j	k	l	m



Ciphers / Types of ciphers → Substitutional cipher
→ Transpositional

Algorithms providing secure encryption techniques

Types of substitutional ciphers

1. Ceasar Cipher (Subst cipher)
2. Mono Alphabetic → pattern is given for substitution
3. Poly Alphabetic → each word should be replaced with unique one
4. Play Fair Cipher

Poly Alphabetic cipher

Encryption → $C = (P+K) \bmod 26$
Decryption → $P = (C-K) \bmod 26$

Plain text →	W	E	D	I	S	C	O	V	E	R
key →	d	e	c	e	p	t	i	v	e	d

key	d_3
Plain text	w 22
cipher text	Z

$a \to 0$

d_2 4
d_3 8

$C = (22+3) \bmod 26$
$C = 25 \bmod 26$
$C = 25$

$C = 8 \bmod 26$

$5 \bmod 26$
5
f

12

Page :
Date: / /

Play Fair Cipher

1. 5 x 5 Matrix Size
2. No Repetition of words
3. I and J in same box block.

key - cipher in Cryptography

c	(r)	p	h	e
u	n	y	t	o
g	a	b	d	f
k	l	m	q	s
v	w	x	z	

Rule to encrypt data in play fair cipher

* Break the given words in alphabet sets each set should have two alphabets.
* If in any set both alphabets are same then write the alphabet one time and add bogus letter X.
* If only singe letter is left then add x with that letter.

→ If both alphabets are in same row. Then replace them with just next right alphabet.

→ If both alphabets are in same column than replace with just next below alphabet.

* If both alphabets are not in the same row or column then swap them with just left or right alphabet.

Page :
Date: / /

Transpositional ciphers

- Railfence cipher
- Columnar Ciphers.

→ Zig Zag Cipher.

For eg :- Plain Text = Secret MESSAGE
Key = 2

encrypt using Railfence cipher
key => number of rows.

S		C		E		M		S		A		E
	E		R		T		E		S		G	

Cypher Text = SCEMSAEERTESG

decryption

S	E	S	E	R	E	G
C	M	A	E	T	S	

S C E M S A E
E R T E S G

→ The Railfence works by writing your message on alternate lines and then reading of each line in turn.
→ All the spaces will be ignored in the plain Text.
→ In Railfence cipher read the top row first then bottom to get the cipher text.

→ Columnal Cipher :-

→ In columnar transp cipher plain Text represents in matrix form.

① → Writing row by row and reading the cipher text column by column [one by one]

② → Permutation is defined by the alphabetical order of the letters in the keyword.

③ → The message is written out in rows of fixed length.

④ → Any spare spaces are filled with blank, null or underscore character

eg :→ keyword → HACK
Plain Text → TREE IS GREEN

H	A	C	K
3	1	2	4
T	R	E	E
-	I	S	-
G	R	E	E
N	-	-	-

↓ encryption

cipher text :→ RIR_ESE_T_GNE_E_

↓ read

→ done according to alphabetical order.

decryption process of columnar cipher

→ To decypher it reciever has to work out on the length of columns by diviciding the message length by the key length.

→ $\frac{16}{4}$ → 4 → no of rows.

Page :
Date: / /

~~Steganography~~

Hill cipher (substitutional cipher) → solved in matrix form

Plain text → ACT → By default = 0.

Keyword → GYBNQKURP

$$\begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 19 \end{bmatrix} = \begin{bmatrix} 0+24\times2+19\times1 \\ 0+16\times2+190 \\ 0+34+19\times15 \end{bmatrix}$$

$$\begin{bmatrix} 48+19 \\ 32+190 \\ 319 \end{bmatrix}$$

24 × 2 = 48

U V W X Y Z
20 21 22 23 24 25

R S T 19

R S T U V 15

95
70
165
34
319

$$\begin{bmatrix} 67 \\ 222 \\ 319 \end{bmatrix} \bmod 26 \quad \begin{array}{r} 48 \\ 19 \\ \hline 67 \\ 190 \\ 32 \\ \hline 222 \end{array}$$

26 × 3 = 78

$$\begin{bmatrix} 15 \\ 14 \\ 7 \end{bmatrix} = \begin{bmatrix} P \\ O \\ H \end{bmatrix}$$

↕ encryption process.

Decryption process Find A^{-1} and multiply by CT mod 26 = PT

$$\begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 8 & 5 & 10 \\ 21 & 8 & 12 \\ 21 & 12 & 8 \end{bmatrix}$$

Page :
Date: / /

Steganography and cryptography

Advanced form of cryptography → secret writing

stegano → covered
graphy → listing

Audio files / text / video / images

↓ dig in frequency.

↓ in the form of said images which are encrypted.

↓ encryption algo's using pixels. Reciever should use same algo for decryption

Algebric structure

A non empty set S is called algebric structure w.r.t a non empty binary operation *
if $(a * b) \in S$
$\forall\ a, b \in S$

If the output belongs in same set then it's called algebric set and closure property.
for eg $(N, +)$ → operation
↓
Natural nos

Group :*

Euclid's Algorithm
↳ GCD Greatest common divisor

gcd(a,b) = gcd(b, a%b) ← Recursive case
gcd(a,0) = a ← Base case
eg gcd(10,14) = gcd(14, 10%14)
= gcd(14, 10)

gcd(a,b) = gcd(b,a)
(a<b) (b>a)

(swapping up of elements)

= gcd(14, 10)
= gcd(10, 14%10)
= gcd(10, 4)
= ~~gcd(10, 10%4)~~
= gcd(4, 10%4)
= gcd(4, 2)
= gcd(2, 4%2)
= gcd(2, 0)

gcd(10,4) gcd(10,4)
b>a
gcd(4,10)
~~gcd(10, 4%10)~~

1
0 | 14
10
4 | 10 | 2
8
2 | 4 | 2
4
0

Page :
Date: / /

Chinese Remainder Theorem (CRT)

used to solve set of diff congruent eq^{n}s. with one variable but diff mod values & modules which are relatively prime

$X = a_1 \pmod{m_1}$

$X = a_2 \pmod{m_2}$

$\vdots$

$X = a_n \pmod{m_n}$

~~$X = a_1 m_1 M_1^{-1} + a_2 m_2 m_2^{-1} + a_3 m_3$~~

$$X = a_1 M_1 M_1^{-1} + a_2 M_2 M_2^{-1} + a_3 M_3 M_3^{-1} + \cdots + a_n M_n M_n^{-1}.$$

$M = m_1 \times m_2 \times m_3$

$M_1 = \frac{M}{m_1}$; $M_2 = \frac{M}{m_2}$ ---- $M_n = \frac{M}{m_n}$

$M_1 \times M^{-1} = 1 \pmod{m_1}$

eg:7 $X = 2 \pmod 3$

$X = 3 \pmod 5$

$X = 2 \pmod 7$

$\frac{15}{\times 7}$
5

$M = 105$

$M_1 = \frac{105}{3}^{35}$ $M_2 = \frac{105}{5}^{21}$ $M_3 = \frac{105}{7}^{15}$

$35 \times M^{-1} = 1 \pmod 3$

$M^{-1} = \frac{1 \pmod 3}{35}$

$M_1^{-1} = 2 \qquad M_2^{-1} = 1 \qquad M_3^{-1} = 1$

$X = (2 \times 35 \times 2 + 3 \times 1 \times 21 + 2 \times 1 \times 15) \bmod 105$

$X = (140 + 63 + 30) \bmod 105$

$X = 233 \bmod 105$

$23.$

Fermat's Theorem

$x^{n-1} = 1 \bmod n$

where n is a prime number and x is not divisible by n.

$\phi(n) = n - 1$ → Eulers Tuotient

$x^{\phi(n)} = 1 \bmod n.$

Eulers Theorem → (Fermats Euler Theorem Or Eulers Tauotient Theorem)

Q. What is Eulers Tauotient | Rep using ϕ and is defined as the number of + integers less than n are called co-prime to n [co-primes means having gcd 1]

Block Cipher

Block → encrypt size is fixed and data is transferred in block.

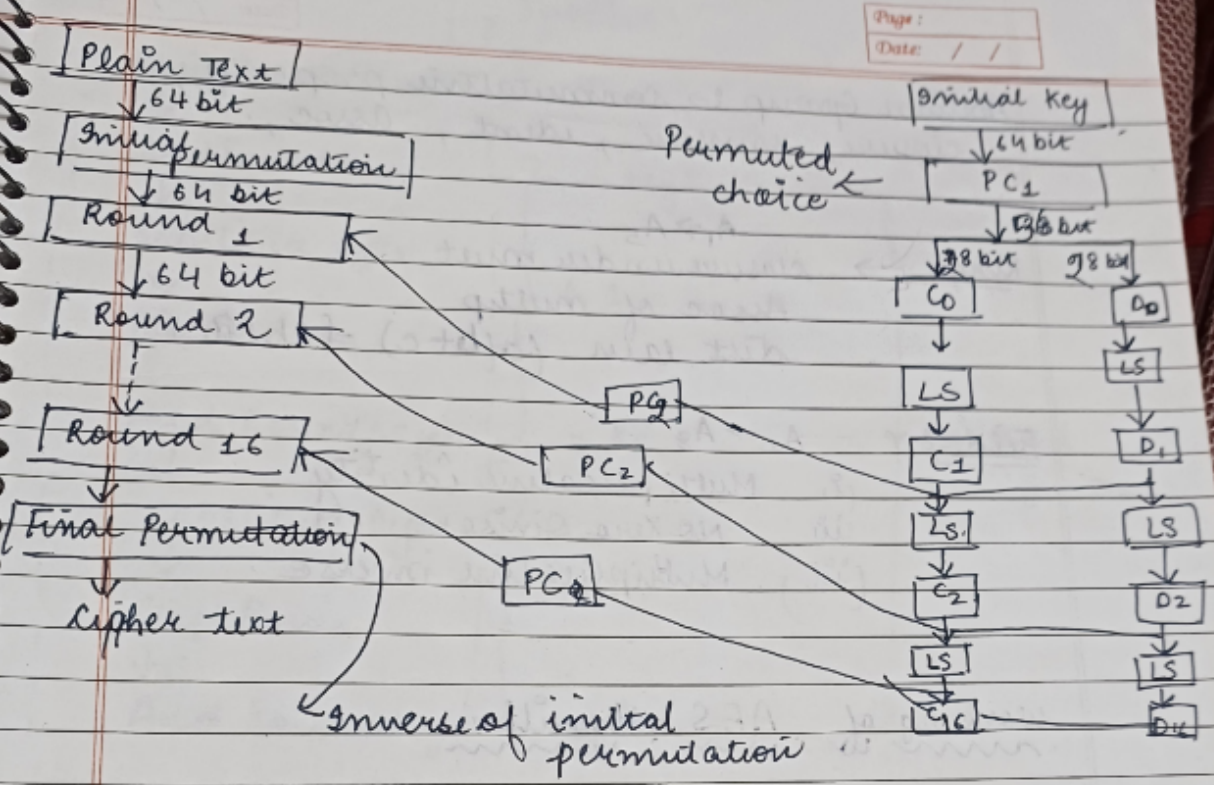
DES → Data encryption standard

AES → Advanced encryption standards.

DES	AES
* Data encryption Standard	* Advanced encryption stand.
* key length :- 56 bits	* key length :- 128, 192, 256 bits.
Plain Text → 64 bits { data }	
* It is bit oriented	* byte oriented.
* In DES we have 16 rounds of encryption	* We have 10, 12 and 14 rounds of encryption 10 → 128 bits 12 → 192 bits 14 → 256 bits
* designed by IBM	* Vincent Rijmen and Joan
* slower	* faster
* Non-flexible	* flexible.
* The structure of DES is based on Feistel Network	* The structure of AES is based upon substitution and permutation.
* DES developed in 1977	* AES is developed in 2001.

Working of DES Algorithm

Plain
64
Initia
Rou
Rou
Rou
Final



Initial permutation :→ Rearrangement of data.

8 bits removed are { parity bits in the key }
{ ↓always 1 and extra bit }

LS → Left Circular Shift

Semi Group :→ If it fulfils the closure property, associative property

$(a*b)*c = a*(b*c)$

Monoid :→ Identity element ; closure, associative

$(a \cdot e) = (e \cdot a) = a$.

Group :→ associ, closure, iden, inverse

$(aa^{-1}) = (a^{-1} \cdot a) =$

Page :
Date: / /

Abelian Group :→ Commutative property, closure, inverse, ident, Assoc,.

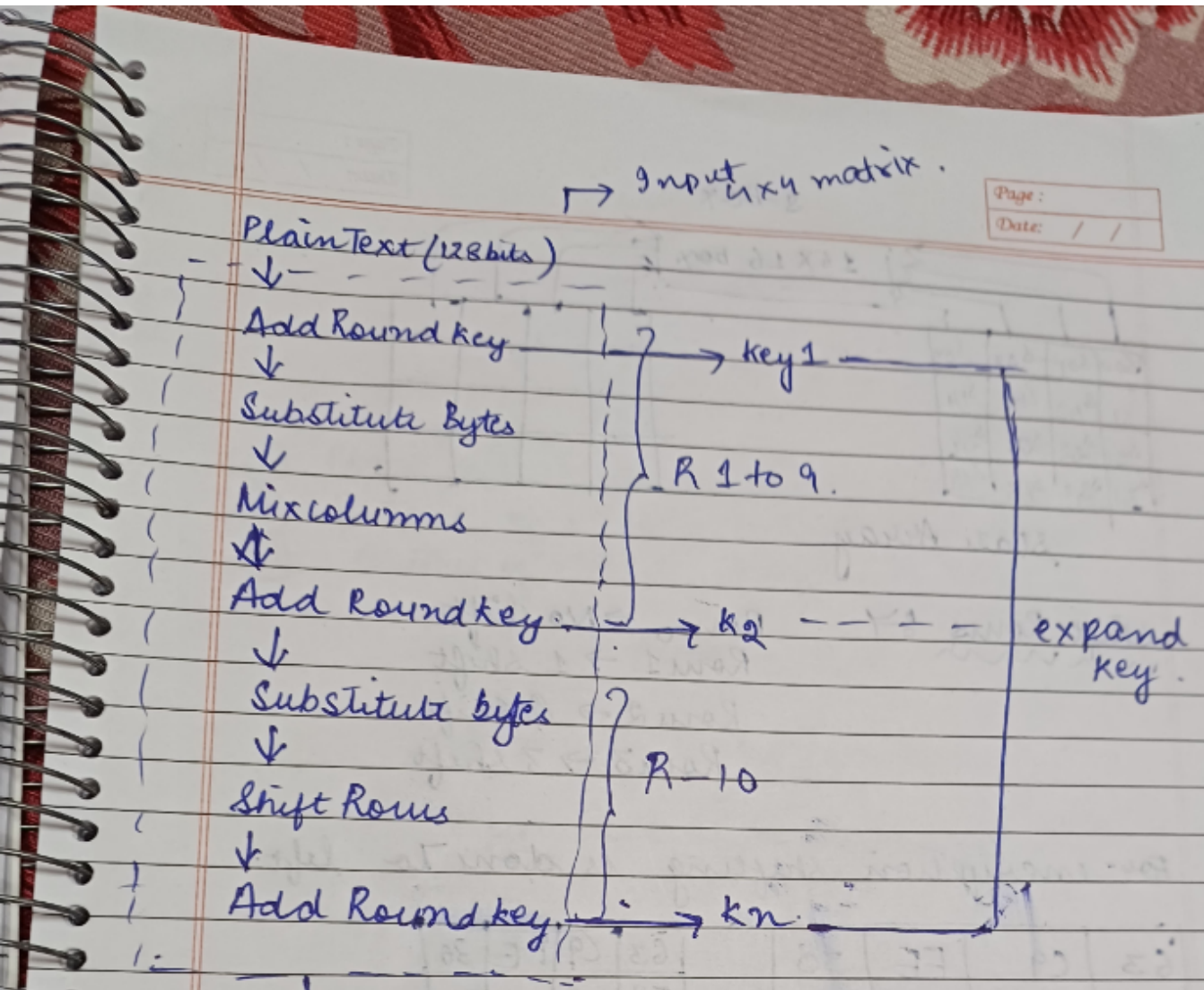
$A_1 \rightarrow A_5$

Rings → closure under mult.
Assoc of multip
dist law $(a(b+c) = (ab+ac)$

Field :→ $A_1 \rightarrow A_8$

(i) Multiplicative identity
(ii) No Zero Divisor
(iii) Multiplicative inverse.

Working of AES Algorithm



Structure of each round

4x4 matrix

↓

Add Round key

↓

4x4 matrix

↓

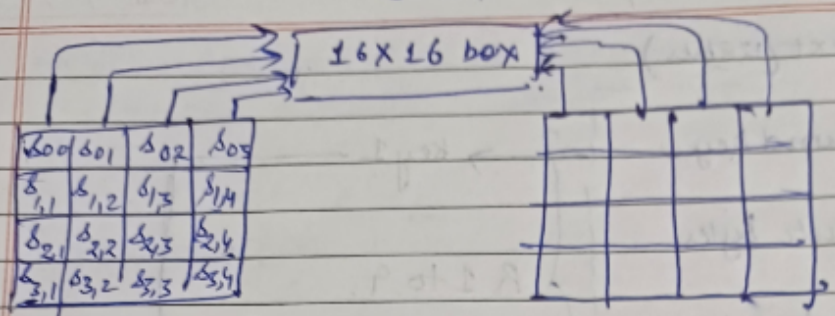
Substitute bytes

↓

4x4 matrix → output

↑ input to shift rows.

S-Box



State Array

Shift Rows

Row 0 → No shift.
Row 1 → 1 shift
Row 2 → 2 shift
Row 3 → 3 shift

For encryption shifting is done to left.

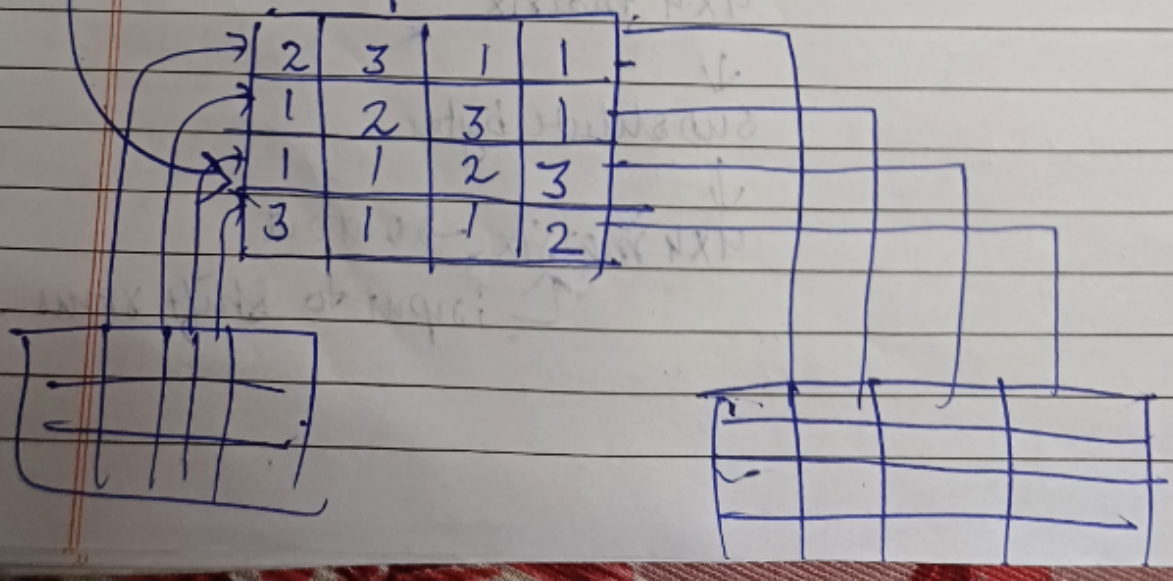
63	C9	FE	30
F2	F2	63	26
C9	C3	7D	A4
BA	63	82	D4

→

63	C9	FE	30
F2	63	26	F2
7D	D4	C9	C3
D4	BA	63	82

Mix Columns

Triple DES → 3 keys will be used



TRIPLE DES

K_1
K_2
K_3

1. Plain text → encrypted → K_1 → output
2. Output of 1st step decrypt with K_2
3. Output of 2nd " encrypted " K_3.

TRIPPLE DES

64 bits Plain Text

K_1 → DES cipher

↓A

K_2 → DES

↓B cipher Reverse.

K_3 → DES cipher

64 bits Cipher Text.

Blow Fish Algo

Encryption technique or alternative way to DES algorithm. It is faster than DES and provide better encryption. in this block size 64 bits [Input]

Key size [32 bit – 448 bits]

→ 16 rounds

→ Input 4x4 matrix.

64 Bits

↓

Round 1

Round 16

→ cipher text