NGFW → Network Security device → advancement of firewall
↳ Resolve the attacks & payload.

↳ Additional features → maintain Security in
↳ ① Application Security ② Control ③ Integrated ④ Cloud

Final

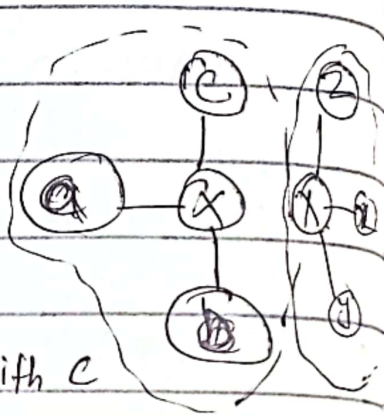
ARP → Layer3 (Network) Layer protocal.

① Convert
IP → MAC
Logic → physical

② It is always broadCast message

③ Reply UniCast

Example within SameNetwork
@ let say A want to Comunicate with C



④ a will contain Ip of C it will send message like this | aIp | aMac | CIp | FFFFF | → it will ~~be~~ received by C and he will return his Mac Address (Working on LAN)

Outside the Network a → B

a → x(port IP) → then → other network port Ip → z

DHCP

ⓐ Assign unique Ip dynamically with in the network

ⓑ Assign IP, gateway, DNS ip to Hosts

NAT Network address translation.

convert

→ Privat to public and public to private

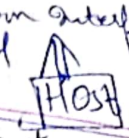
→ private Ip's can match with other ip

→ private network use range of classes.

→ NAT → make NAT Table

→ Request ⇒ private → Public | Response ⇒ public → Private

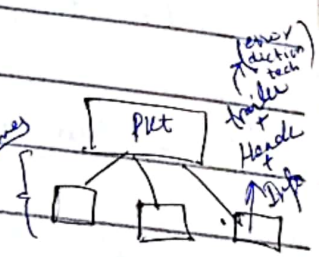
Implemented in DLL → Network interface Card



Data link layer Responsibilities

Transfer of data from

Network Layer → DLL → physical Layer



① Framing divide the streams of bits into manageable data frame.

② physical addressing → If frames are to be distributed on some network DLL add header to define Sender or recieved of device connect

③ Access Control → DLL protocol use to determine which device will control over link when 2 or mo

↳ To avoid collision

④ Flow Control → DLL impose the flow of data if sender send with high speed and recieve

to control

(Sliding window protocols)

⑤ Error Control

↳ @ Error detection ⓑ Error Correction } responsibility of DLL

Error Detection → used to detect error

↳ detection Range → (Hamming bit − 1)

How find Hamming bits

① Pair of two words ② Convert into binary ③ XOR both ④ No 1's

Hamming distance

```
 10101
 11110
-------
 01011  = 3 → Hamming distance =
```

H-b

Error detection techniques

(1) Parity checking tech

① Single or one dimensional Parity check

Even parity

Check no of 1's if that is even add 0 parity bit else 1 parity bit with data. which will

| 0 1 1 1 0 | 1 | , | 1 1 1 1 | 0 |

Confirm at receiver side by comparing parity with data bits.

↳ problems

① if even no of bits slip → do not detect error.

② Can detect all single bit errors / odd bits. (or)

② Two dimensional parity check :— ← update version

technique is same as single parity but here we consider data as matric we calculate parity in row wise an column wise

1	0	1	0	0
1	1	1	0	1
0	1	0	0	1

no of bits slips = no of parity error occur

parity bits of each row send along with data.

and compared at receiver side. → if data correct → accepted ↓ else rejected.

Problem

it can detect error in

Limit As it can not work if 4 bits slips.

(2) Checksum

Sender

① data segments add if 1 carry generate ~~[illegible] into~~ them → that will rape round

② Take 1's complent. to get sum.

③ No this is checksum

④ Send the data + checksum to receiver.

Reciever: add the data and checksum if all are all 1's then data is Correct else data corrupted

Sender

Fix: ① 10110011 } data
10101011
01011100
→ +1
01011101 } addition
10100000 → Checksum

Reciever

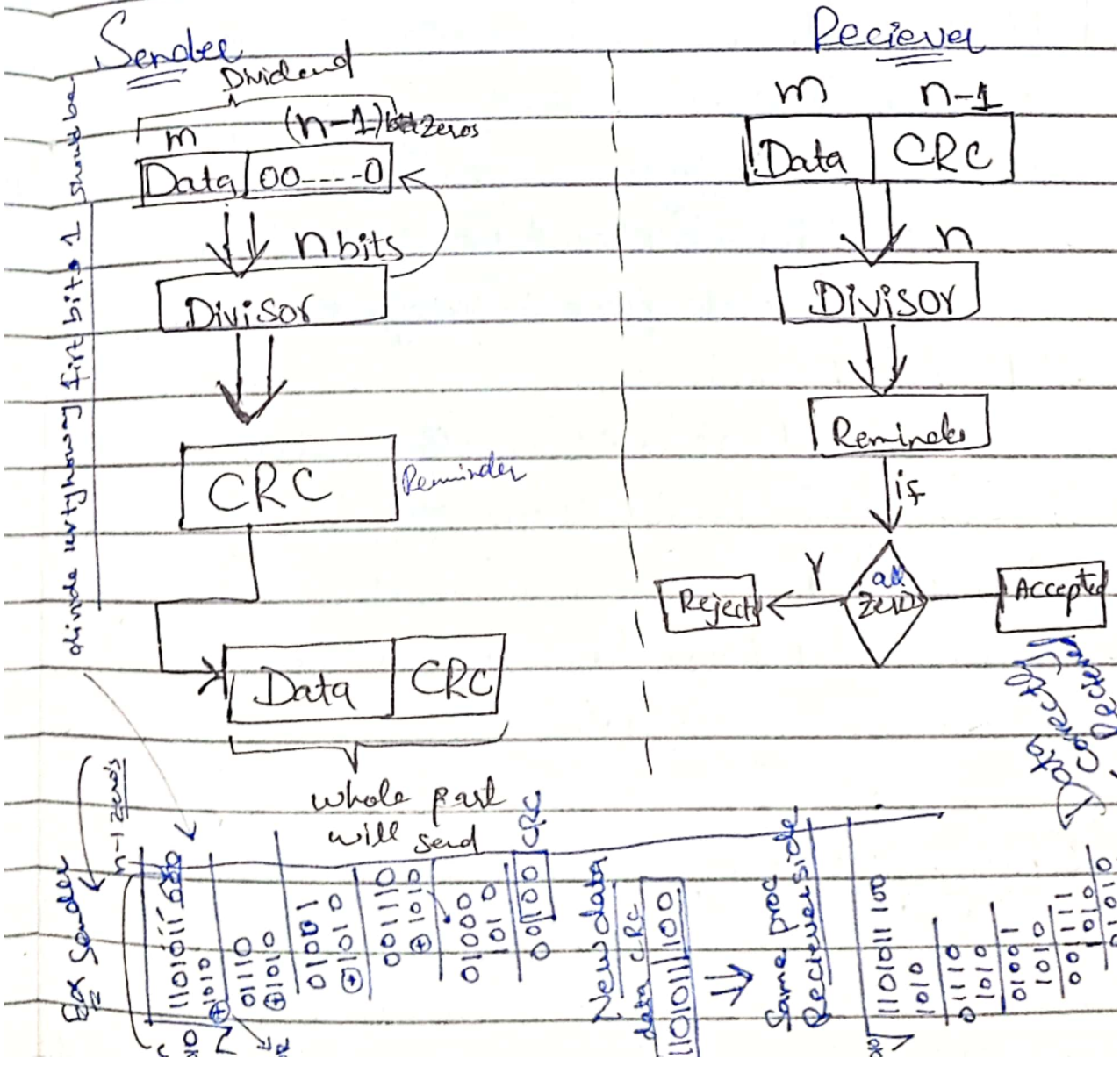
data addition
01011111
10100000
11111111
↳ No corrupted

Accepted.

Accepted ← 10100000

It can not detect all errors: → detect all errors involving odd + even bits

CRC ⇒ Cyclic Redundancy Check



Multiple Access ~~Link~~ Protocols

In Multiple Access link there are more than 1 host are connected through a link → data trafering

MAP → Prob
Shared medium where nodes try to transmit data simultaneously due to which collision occurs. data got corrupted. data.
↳ An algorithm → to determine → when node can transmit
↳ Here only 1 communication channel available

Ideal multiple access protocols:-
↳ M nodes, all have power to transmit data → full decentralized.

MAC Protocols: Taxonomy:

Three broad classes:

① Channel partitioning:-
↳ ① divide channel into small pieces.
② Alocate piece to specific node.

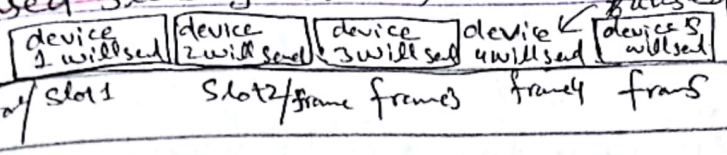
② Random Access:-
↳ ① Channel not divided → allow collisions.
② Recover from collision by retransmiting.

③ Taking turns:-
Node take turns, and can take more longer turn with size of data.

Channel Partitioning:-

TDMA
Access to channel in "rounds"

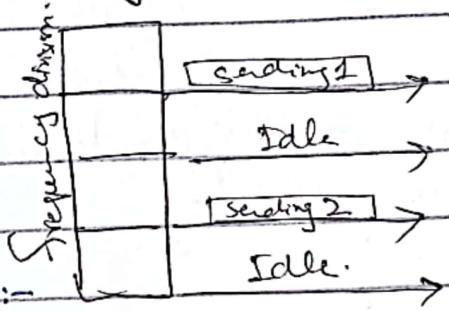
→ divide chnel in ~~diff~~ fixed legth slots according to time and assign them to ~~host~~ devices
→ Unused slots go Idle → no data send



FDMA

Chanel divided by frequency

pkts transfer according to frequency free slots will be unused / Idle.



Code Divison Multiple Access:-

each nod contain diffent code.

→ depending upon the code multiple user can transmit simaltaneusly.

Random Access Protocal (RAP):-

① A node can send pkt at full channel data rate R

② no priority here.

③ If collision occour.

- ⓐ how to detect collisions
- ⓑ how to recover from collision

} RAP

⇓

Example

① Slotted Aloha ② Alotla.

③ CSMA, CSMA/CD, CSMA/CA.

Sloted Aloha

→ data transit one by one node

Assumption → equal size frames → transit in start time.

Operation @ → if node transist data succefull in next slot next frame will send

ⓑ Else retransmit that frame

Pros:	Cons:
① Single node can transit data at its full rate on chanl	① Callision, wasting slots, Idle slots
② Highly decaterize → No ordering node	② If start of the transit Cellision occure the whole need to retransit.
③ Simple.	

Unslotted Aloha (PURE Aloha).

ⓐ No need of Syncronization → one by one send.

ⓑ Which node Contain (PKG) frame to transit. imideatly

ⓒ Collision probability increas.

(d) less efficent than Slotted Aloha.

CSMA (Carier Sense multiple Access).

first it detect the channel before transmit

- Channel free → Transmit entire frame
- Chanl Busy → wait for transisten

Prob: let say 2 node Senses at a time Chanel free they start transiting Callistens occurs.

Hidden Prob in multi trafering

B not visible to A

A // // // B ⇒ trafering data Collision occur

↓ Solutions is

RTS & CTS → clear to send.

↓

Ready to send ⇒ Recieve Ack → CTS → then transission occur.

CSMA/CD → (LAN)

Career Sense Multiple Access / Collision Detection

→ No Ack → wired.

A node traniting → Collision data recived at that node at that time. this mean this is its Collision data.

TT > TD else it cannot detect it is its collided data.

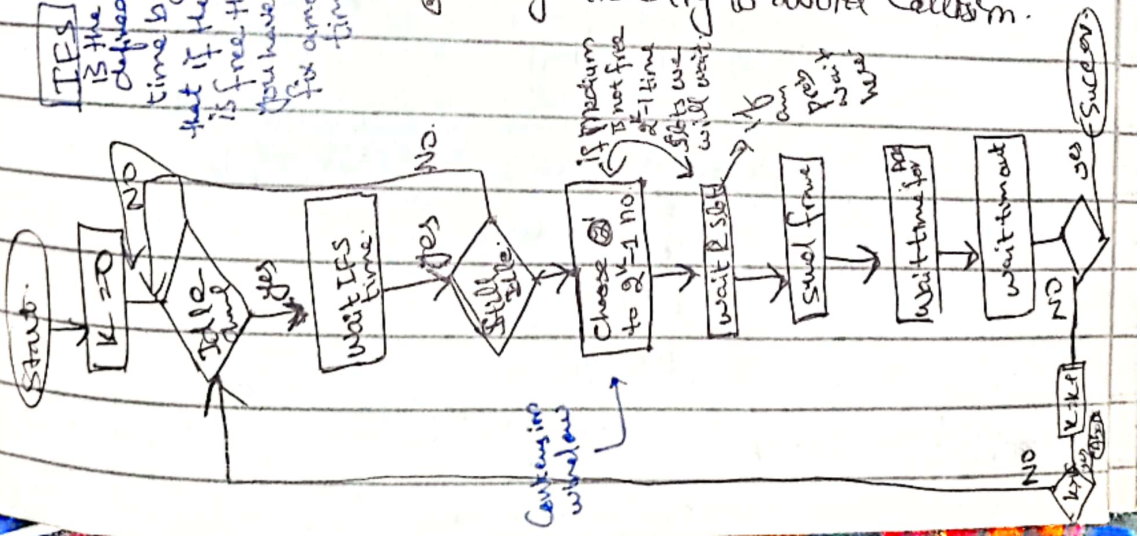
Worst Case

$TT \geq 2PD \Rightarrow \frac{L}{Bw} \geq 2P_D \Rightarrow L \geq 2 \times P_B \times Bw$

Efficiency $= \eta = \frac{1}{1+6.44a} \Rightarrow a = \frac{P_D}{TT}$

CDMA/CA → Collision Avoidance.

↳ used in wireless. in wirless not possible to dected Collision that's why here try to avoid Collision.

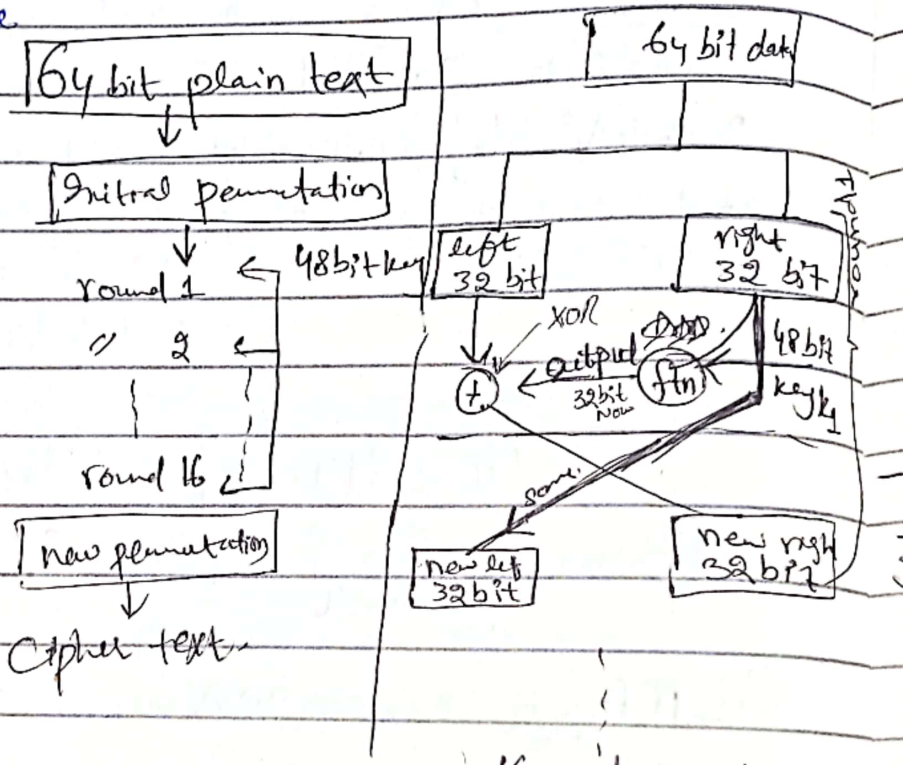


DES Data Encryption Standard.

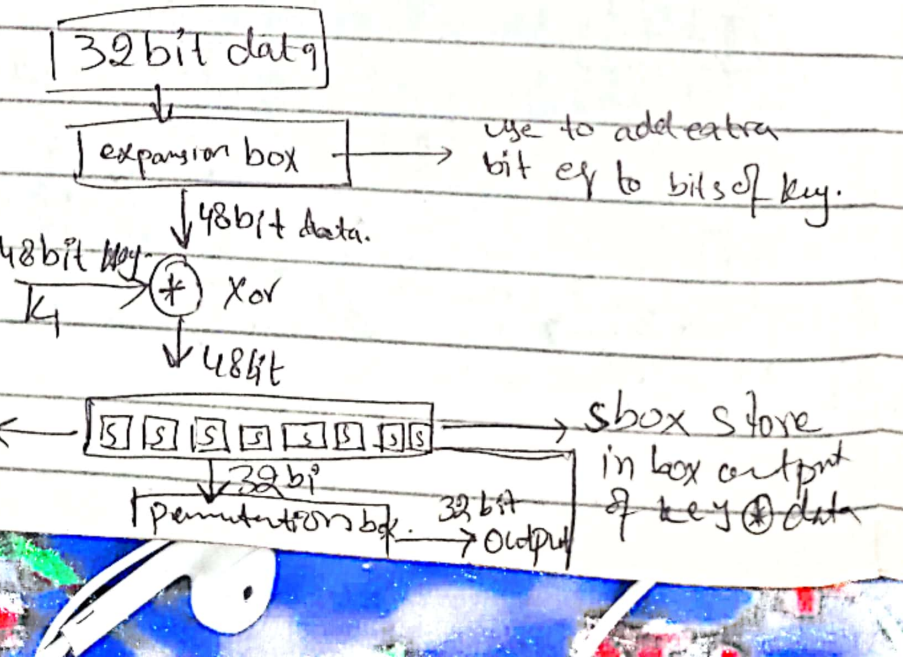
↳ 64 bit plain text block.

Working
1. permutations → ABC, BAC, BCA, CBA, CAB (No of arrangements)
2. pass 16 rounds
3. Swaping left right swap.
4. Final permutation.

Structure



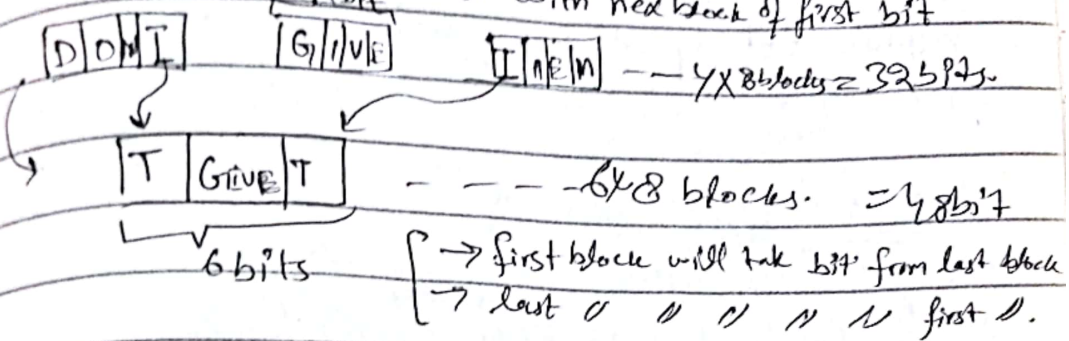
What is ftn (whenever we are passing key + 32 bits)?



what is expension box used in ftn?

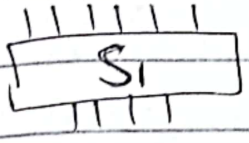
How expansion box convert 32 bit data to 48 bit?

① block names in block and its first and last bit place will filled with letter of previous box and last with next block of first bit



Sbox working?

↳ Conversion of 48 bits to 32 bit.



let say 6 bit are 0 0 1 0 1 1

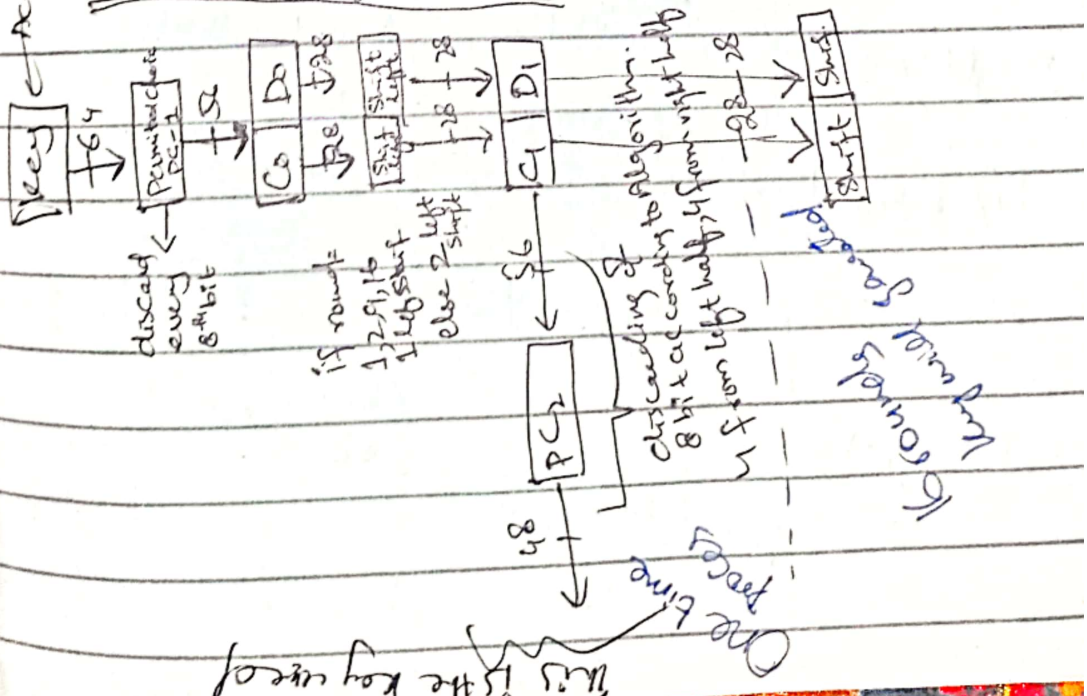
Row

Column

for each block of S there will be different table.

by Cal and row respect value will be (1–10) that can represent in 4 bits so its size will be 6 bits → 4 bits

How to keys generated?



Avalanc effect

With atleast 1 bit change in plain text
Can generate high effective cipher text.

Completeness effect :-

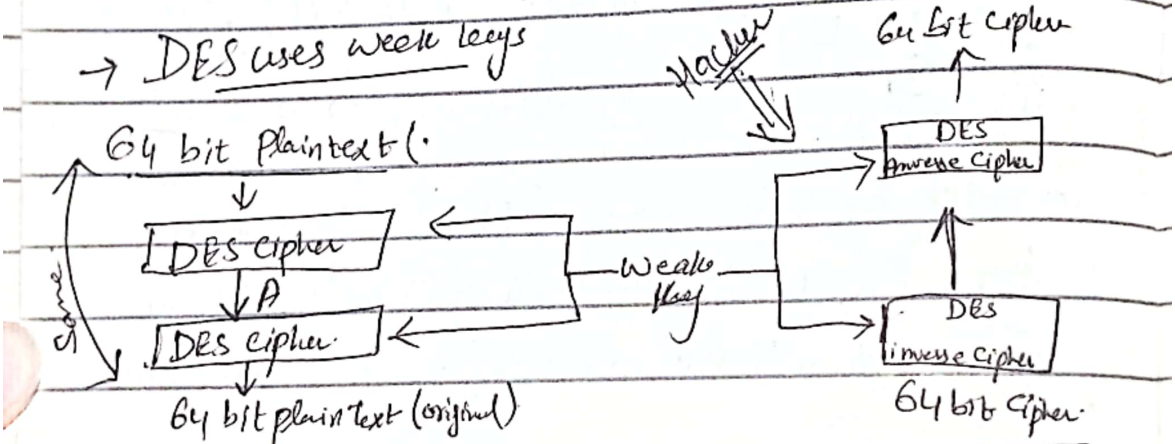
cipher text depends upon more than 1 bits of plain text.

Weakness of DES

→ Semi weak keys → order of keys (known by hacker) will affect if ⇒ Weak in cipher

→ 56 Combination is now easily breakable due to processing powers of PC's and parallel Computing.

→ DES uses week keys



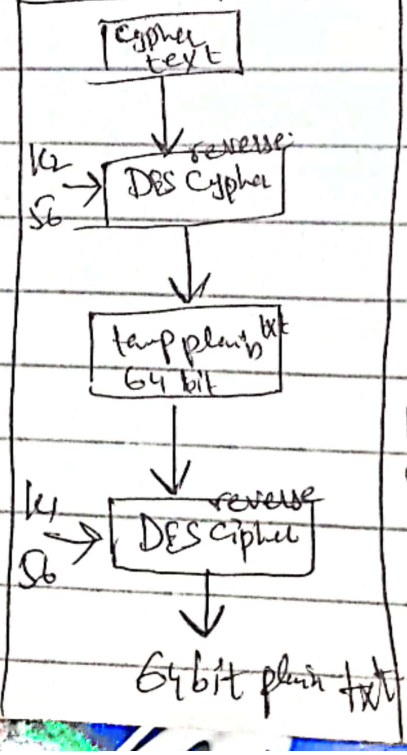
Double DES

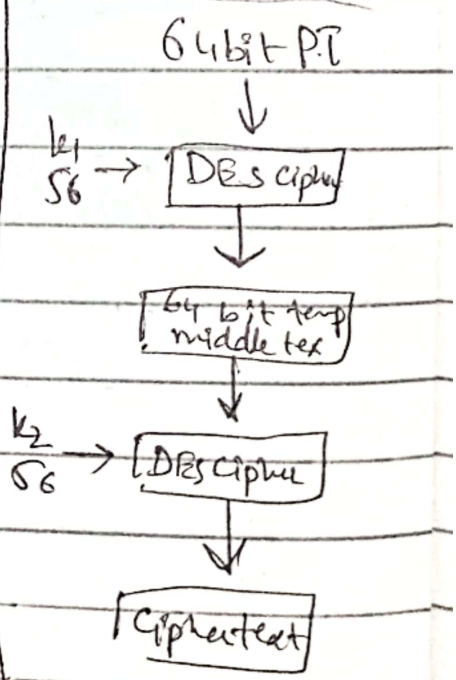
Encryption

↳ $C_{phe} = E(k_2, E(k_1, P_1))$

Decryption

plain txt $= D(k_1, D(k_2, C))$





Drawbacks of 2-DES

Key1 By 2^{56} Combination → Ciphertext generated

Key2 By 2^{56} Combination → Ciphertext generated

Both → Common pair values will be obtained

Usable

Attacker proceeds

Plain text = P, Cipher text = C

① encrypt P for all 2^{56} possible value of K_1 and store the results in a table and store it.

② Now decrypt © using all 2^{56} possible values of K_2. As each ie decryption result is produced checke against the tabble for a match.

③ When there is a match, we have locate a possibly correct pair of keys.

④ For multiple keys he has to try all possibilities.

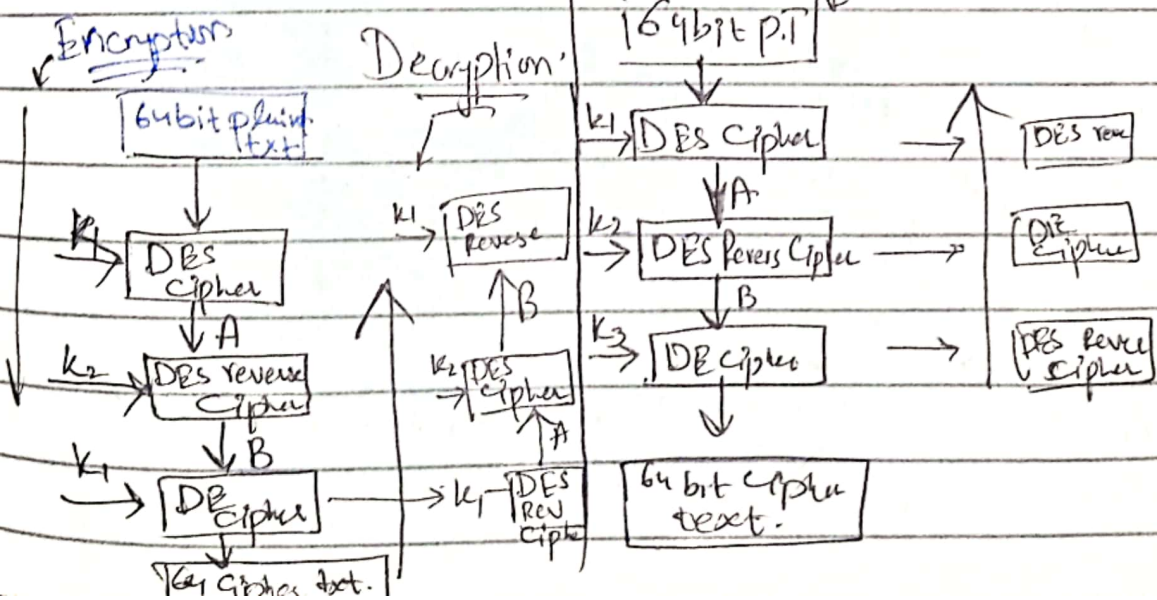
⇒ It take twice time the DES.

Triple DES

2, or 3 keys are used

Much Stronger than double DES.

2 keys



3 key Encryption Decryption

1 word = 32 bits

AES → Take data in block. [most secure unbreakable.]

↳ fixed size = 128 bits → 16 bytes = 4 words

There can be any version no of rounds and corresponding from bellow table:

	Rounds	no. of bits in key
Ver 128	10	128
Ver 192	12	192
Ver 256	14	256

Any key which is used that divides into parts.

and ** No of key generated by key Expansion Algorithm = (no of round + 1)

Input will be in 4x4 matrix = 4 words

State array which store key + data in matrix form.

If key = 128.

k_0	k_1	k_8	k_{12}
k_1	k_5	k_9	k_{13}
k_2	k_6	k_{10}	k_{14}
k_3	k_7	k_{11}	k_{15}

Expand key Algorithm →

w_0 w_1 w_2 ..... w_{43}

Expand key

10 rounds

Plain text (128 bit)

Only xor operation.

add Round key ← (w_{0-3}) → 4 words.

Substitute bytes → subbytes.

Shift operation

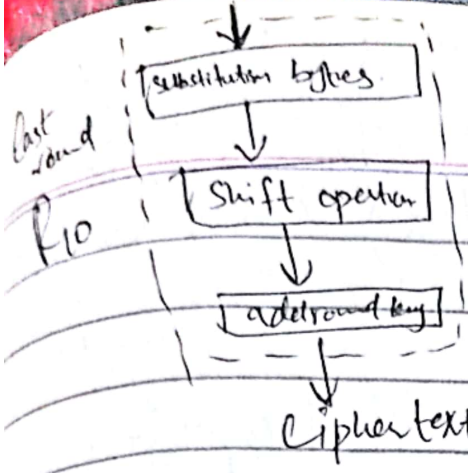
Mix Colu

Add round key. ← $w_3 - w_7$

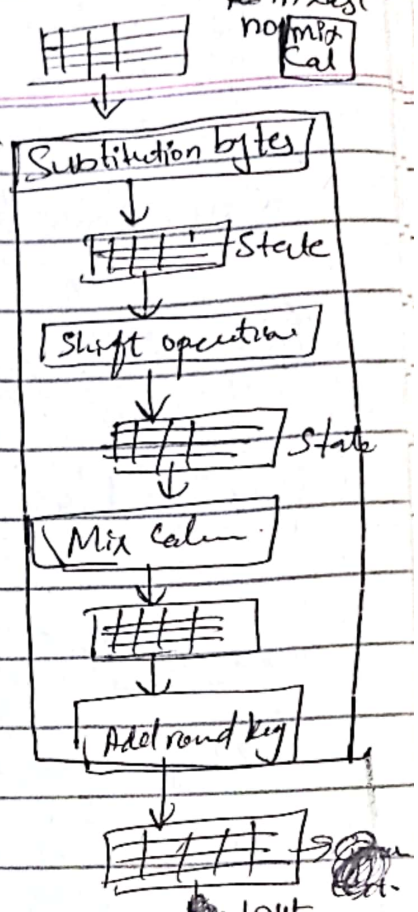
Round 1 = 4 transformation

All rounds will have 4 transformation and last will have 3 transfo

Pass to next round



Structure of 1 Round (in last no mix Col)



Substitution byte

16x16 → S box

we will take a value from state table ↑ Search by 1st 4 → Row
divid into 4,4 bits
2nd 4 → Col. by their digit value in
See in Substitution table (S-box) put
that same position of new mixing Statematrix.

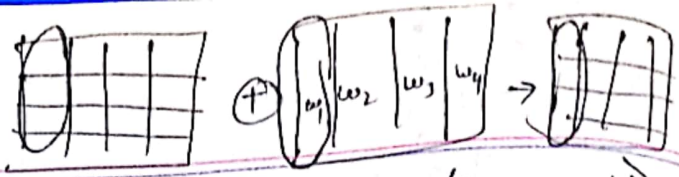
Shift Rows → Shifting in 4 byte (Couple row)

↳ Here we Shift rows left to the left
→ No of shifts depends upon the rows of State matrix

Row Zero → No Shift. left
Row 1 → 1 byte shift //
// 2 → 2 byte shift //
// 3 → 3 byte Shift left

Mixing Colm

take 1 Colom from State matrix Xing with 4X4 Co
matrix Inset into new State matrix Do one by
one

Add Round key 

(Take 1 Colm of stat matrix) XOR (1 word)
gust into new state matrix

This will be output of this round and input for next round.

Difference b/w AES & DES

Caesar Cipher → monalphabatic C.S, Simplest methode of encryption decryption tech.

↳ Replacement of alphabets according to the key

plaintext, key → ciphertext.

Ciphertext

$$c = E(k, p) = (p + k) \bmod 26 \quad \text{// Encryption}$$

key, message, plain text

* if (c-k) is -ve then add 26 to it then take mod

Decryption

$$p = D(k, C) = (C - k) \bmod 26$$

key, ciphertext

Classical Encryption Technique

2 types

① Substitution Techniques ② Transportation tech.

① // :- Technique we generate cipher text by replacing a letter with other letters.

① Transportation technique.

↳ perform permutation (rearrangement)

Permutation
ABC
ACB
BCA
BAC
⋮

② Mono alphabatic Substitution tech/ciphers-

if there is repeated alphabets in text then replace them with specific alphabet if that letter came in future that will also replace with same.

1 to 1 replacement

My NAME ⟶ NP OBNZ

③ polyalphabatic Substitution cipher(tech):-

→ There is no fixed substitution

→ we can use more the 1 substitution for same

My NAME ⟶ NP OBYZ

if in future same letter comes we will replace with other letter also.

RSA → Rivest - Shamir - Adleman. in 1978

↳ Asymetric Algorithm ⟶ 2 keys uses ⟶ public key, private key

↳ Block cipher

Public key ⟶ known to all user in Network.

Private key ⟶ kept secrete, not sharable to all.

ie → if public key of user A is used for encryption (for) we have to use the private key of same user dec

Algorithm (Numerical in video)

① Key generation

① Take 2 prime no p & q.

② Calculate n = p*q.

③ Calculate $\phi(n) = (p-1)(q-1)$

④ Choose value of e: or e will be given.

$1 < e < \phi(n)$ and $gcd(\phi(n), e) = 1$

⑤ Calculate

$d = e^{-1} \bmod \phi(n) \Rightarrow de \bmod \phi(n) = 1$

⑥ public key = {e, n} // Public key to encrypted

private key = {d, n} // Private key to decrypted

① Encryption:-

$C = M^e \bmod n$

M < n

M → no of letters.

n → Calculated in key generation

② Decryption:-

$M = C^d \bmod n$.

① Symmetric cryptography: Only 1 key same used to Encrypt & decrypt

② Asymmetric Chromatography: 2 diff key to encrypt or decrypt.

① public ② private key

SSL Certificate → Issued by 3rd party

↳ Secure Socket layer → encrypting data exchaged b/w client and server

this communication should be done browser use SSL.

① Client request through HTTPS server

② SSL certificate contain public key. assigned by 3rd Party

Working

(1) client request a page through webserver.

(2) webserver send SSL Cetificate + its public key

(3) browser verify signature + public key. → verification done.

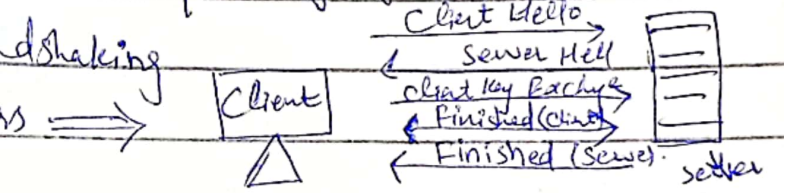
(4) Browser create Symmetric key it keeps one and copy send to server.

(5) Brower send the Symmetric key by encrypting with the public key of server

(6) server recieve the encrypted key it decrypt by its private key. and Communication starts.

⇒ SSL And TSL ⇒ Transport Layer Protocols.

① first handshaking process ⟹



② then after that both start Comunicate.

⇒ Digital Signature ⟷ it is a proof in hand of reciever → coming from correct entity verified.

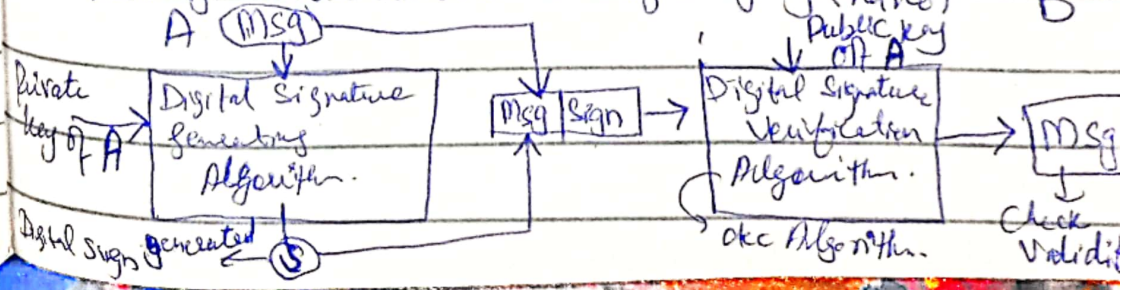
if sender to send encrypted message in which his private key is used the at reciever side to decrypt it it by Sender public k

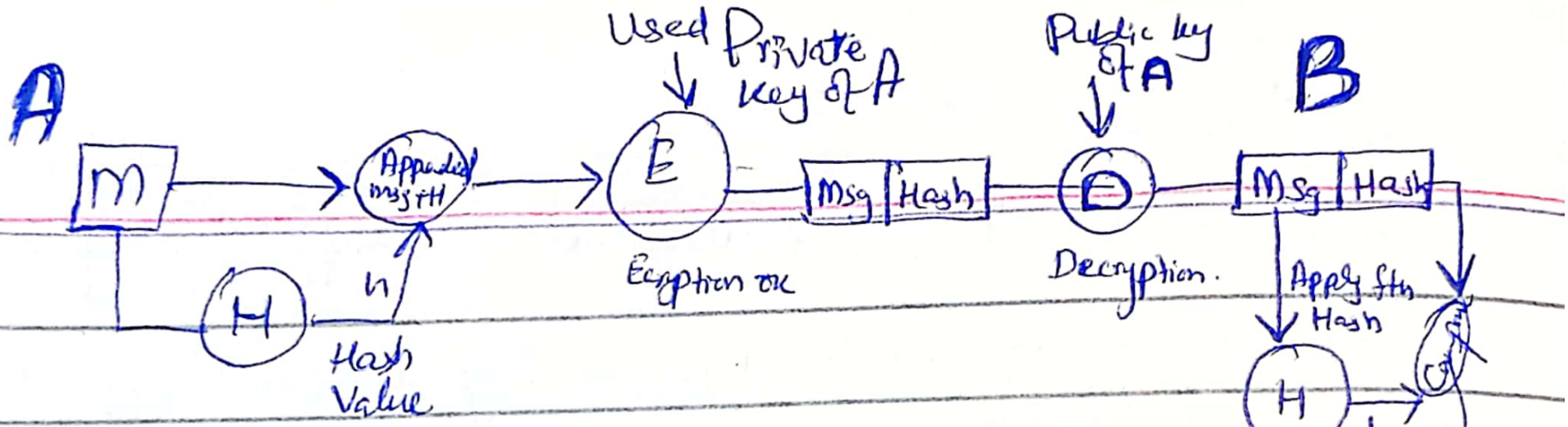
→ It is a symmetric key cryptography

encryption → private key.

decryption → public key

→ Message Authentication & msg integrity (change nahi)





→ Maintain data integrity. Using Hashing

☆ → When we sign a document digitally we send the sign as a separate.

Sender Send 2 docs → msg + its Signature.

→ Authenticity proved by using public key decryption.

→