18/2/25

- **Cryptography**

The Process of writing messages in a secure way is called cryptography.

objectives (of security) Imp.

i) **Integrity** → S ⟷ R

The content should be intact, which means it shouldn't change before and after sending.

ii) **Confidentiality** - Accessable for the team, not for the outsider.

iii) **Non repudiation** - Preventing users from denying actions they have performed, ensuring accountability.

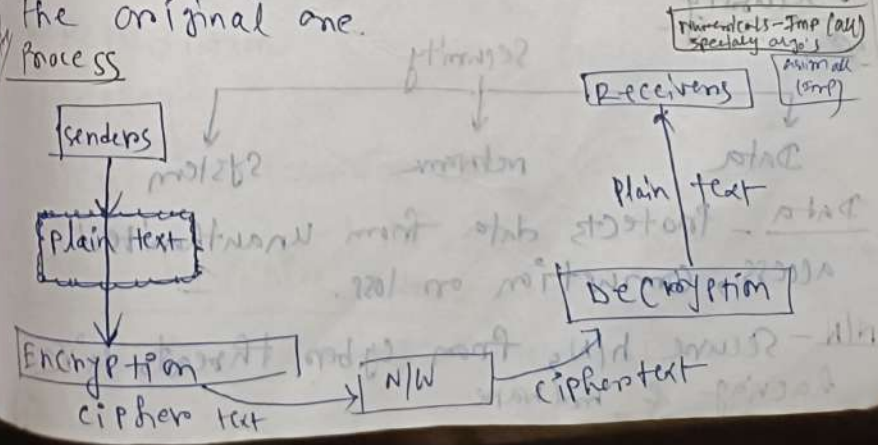
iv) **Access Control** - Ensuring that authorized users can access the information.

v) **Availability** - Ensuring that authorized users have access to the info's when needed.

- **Crypto Analysis**

To analyze the data to make sure it's the original one.

vi) **Process**



- **Diff b/w cryptography & crypto analysis**

Cryptography	Cryptoanalysis
i) The process of writing messages in a secure way is called Cryptography.	i) It is the process to analysis analyze the data to make sure that its the original one.
ii) It's goal is, Securing the data from the adversaries.	ii) It's goal is, breaking the secured data & finding the encryption key.
iii) Expert - Cryptographers	iii) Expert - Cryptoanalyst.
iv) It converts P.t. to C.t.	iv) It analyzes & deciphers C.t.

the algo used for encryption is called encryption algorithm. The algo used for decryption is called decryption algo.

- Threats & attacks

Threats are potential risky like eavesdropping, data tampering, replay.

Attacks that can harm data security. Means techniques like brute force is used to break encryption & steal or alter data.

Attack
- Active: can be detected after some time.
- Passive: They are very hard to detect. harmful than active attacks.

- Passive attacks are more harmful than active attacks.
- Security

Security
- Data
- network
- system

Data - Protects data from unauthorized access, corruption or loss.

N/N - Secure n/ws from cyber threats like hacking & malware.

system - Ensures the safety of the devices & s/ws from attacks.

Cryptography
- symmetric (eg - AES, DES, Blowfish, Twofish)
  - One key is used within the sender & the receiver.
- Asymmetric (eg - RSA, Diffie-Hellman)
  - A pair of keys are used (public for encryption, private for decryption) within the sender & receiver. (public key) (private key)

B/N N no. of people $^{N}C_2$ keys are required in symmetric encryptography.

- Types of active attacks
  - i) masquerade attack
  - ii) modification of message
  - iii) Repudiation
  - iv) Replay
  - v) Denial of service attack (DOS)
    - ↳ Distribute DOS (Prevention)
- Types of passive attack
  - i) Release of message content

25/2/25

ii) Traffic analysis.

Types of cipher → cipher

i) Substitution cipher
- Mono alphabetic → ceasar cipher
- Polyalphabetic → vigenere cipher

ii) Transposition
- Rail Fence cipher
- Columnar cipher
  - Single columnar transposition cipher

Mono alphabetic substitution cipher [1:1 mapping]
- one single character is replaced by another character

eg-①
Plain text: HELLO
cipher text: KHOOR (+3 (key))

eg-②
P.T: ALIAH
C.T: FQNFM (+5 (key))

Poly alphabetic substitution cipher
- one single char is replaced by more than 1 char

eg-①
P.T: GEEKSFORGEEKS [1:m mapping]
key word: AYUSH
item gap (_) is ignored

[If using 'F' for 'A', then all the 'A' will come after many times]
[A will be replaced by 'F' indexed many times]

GEEKSFORGEEKS
AYUSHAYUSHAYU
GCYZCZFMKYLEIN → C.T
[key/keyword]

[1st time A → F, 2nd time A → 2, i.e not fixed (based on algo)]

[If this is decrypted then we will get P.T]

[P_i = val. of P.T (that alphabet) which 'i' is key (=)]

- Encryption

$E_i = (P_i + K_i) \bmod 26$

Eg
G → $E_G = (6+0) \bmod 26$
$= 6 \rightarrow G$

- Decryption

$D_i = (E_i - K_i) \bmod 26$

[E_i = val. from encryption of that particular alphabet]

Eg
$E \rightarrow E_E = (4+24) \bmod 26 = 2 \rightarrow C$

Eg
$D_E = (4-24) \bmod 26 = -20 \bmod 26 = 6 \rightarrow G$
$D_C = (2-24) \bmod 26 = -22 \bmod 26 = 4 \rightarrow E$

Alphabet indexing starting from 0 & ends at 25

Anything + A → that anything

$E_F = (4+21) \bmod 26 = 25 \rightarrow Z$
$E_P = (10+18) \bmod 26 = 2 \rightarrow C$
$E_S = (18+7) \bmod 26 = 25 \rightarrow Z$
$F_F = (5+0) \bmod 26 = 5 \rightarrow F$
$E_O = (14+24) \bmod 26 = 12 \rightarrow M$ [Not correct]
$E_D = (17+20) \bmod 26 = 10 \rightarrow K$
$E_H = (6+18) \bmod 26 = 24 \rightarrow Y$
$E_E = (4+7) \bmod 26 = 11 \rightarrow L$
$E_S = (4+0) \bmod 26 = 4 \rightarrow E$
$E_A = (10+24) \bmod 26 = 8 \rightarrow I$

$E_S = (18+21) \bmod 26 = 13 \rightarrow N$ [correct one]

GCYCZFMLYLEIN

Assignment ①
① P.T: Today is tuesday
key word: AXZ
② Key: NICKS → single column
PT → ALPHA UNIVERSITY

ii) Traffic analysis.

Types of cipher

i) Substitution cipher
- Monoalphabetic → ceasar cipher
- Polyalphabetic → vigenere cipher → read from here

ii) Transposition

P.T: today is Tuesday (correct one)

A x 2 A x 2 A x 2 A x 2 A x

$E_T = (19+0) \bmod 26 = 19 = T$	$E_T = (19+23) \bmod 26 = 16 = Q$
$E_o = (14+23) \bmod 26 = 11 = L$	$E_u = (20+25) \bmod 26 = 19 = T$
$E_d = (3+25) \bmod 26 = 2 = C$	$E_e = (4+0) \bmod 26 = 4 = E$
$E_a = (0+0) \bmod 26 = 0 = A$	$E_s = (18+23) \bmod 26 = 15 = P$
$E_y = (24+23) \bmod 26 = 21 = V$	$E_d = (3+25) \bmod 26 = C$
$E_i = (8+25) \bmod 26 = 7 = H$	$E_a = (0+0) \bmod 26 = A$
$E_s = (18+0) \bmod 26 = 18 = S$	$E_y = (24+23) \bmod 26 = V$

C.T: FQNFM + 5 (A, then all the how many times 'A' will come that many times A will be replaced by 'F', fixed)

Polyalphabetic substitution cipher

eg-① one single char is replaced by more than 1 char. [1:m mapping]

P.T: GEEKSFORGEEKS

Key word: AYUSH (here gap (–) is ignored)

P.T: GEEKSFORGEEKS (first time A → F)

Key = AYUSH

G E E K S F O R G E E K S
A Y U S H A Y U S H A Y U

A 0, B 1, C 2, D 3, E 4, F 5, G 6, H 7, I 8, J 9, K 10, L 11, M 12, N 13, O 14, P 15, Q 16, R 17, S 18, T 19, U 20, V 21, W 22, X 23, Y 24, Z 25

$E_G = (6+0) \bmod 26 = 6 = G$
$E_E = (4+24) \bmod 26 = 2 = C$
$E_E = (4+20) \bmod 26 = 24 = Y$
$E_K = (10+18) \bmod 26 = 2 = C$
$E_S = (18+7) \bmod 26 = 25 = Z$
$E_F = (5+0) \bmod 26 = 5 = F$

$E_O = (14+24) \bmod 26 = 12 = M$
$E_R = (17+20) \bmod 26 = 11 = L$
$E_G = (6+18) \bmod 26 = 24 = Y$
$E_E = (4+7) \bmod 26 = 11 = L$
$E_E = (4+0) \bmod 26 = 4 = E$
$E_K = (10+24) \bmod 26 = 8 = I$
$E_S = (18+20) \bmod 26 = 12 = M$

∴ GEEKSFORGEEKS (P.T)
GCYCZFMLYLEIM (C.T)

Single column

PT → ALPHA UNIVERSITY

L.C.T

TLC AVH SGT EPCAV

today is tuesday (P.T).

TLCAVHS GTEPCAV (C.T)

FOR GEEKS

[I.m
mapping

Rail Fence cipher — keyvalue = 2

G E S O G E S
E K F R E K

C.T: GESOGESEKFREK

Single columnar transposition cipher

N	I	C	K
4	2	1	3
K	R	I	S
H	N	A	_
R	A	N	J
A	N	_	_

[_ = space/gap] — no word

Here gap (_) is not ignored, it is taken.

Given text: Krishna Ranjan (P.T), key = NFCK

C.T: IAN_RNANS_J_KHRA

- Masquerade attacks — The attacker disguises himself as an authorized person.

BOB [BOB Pretends to be Lily]

Lily → Internet → John

- Modification of message — Hacking & changing the message.

BOB → modifies the message

Lily → internet → John

- Repudiation — where some person does something such as financial transaction or sends a message & one doesn't want to send then denies having done it.

- Replay — 3rd Party captures the message & resends it.

It is done multiple time, Preventing the actual message which is to be received.

- Denial of service attack — The 3rd party sends calls or messages to the servers to make it engaged. It targets the n/w with traffic & Prevents the legitimate users to access them.

- The release of the message content — Reading the conversations with others & understanding the content instead of capturing them all.

Traffic analysis — Attacker observes the useful info & it's type instead of the actual message.

- Rail Fence cipher

27/2/25

- Block & Stream cipher → Either 1 bit or 1 byte but not more than that. (bit by bit)

Fixed length of data is being taken (64 bytes, 128 bits etc). Then the algo is being applied.

- Diff b/w block & stream cipher,

G E S O G E S

E K F

• Diff b/w replay attack & denial of service attack

Replay attack

i) Defi°

ii) It's purpose is to deceive the system / gain unauthorized access

iii) Targd - Comm° / data b/w 2 parties.

iv) method - capturing & reusing old messages.

v) Difficult to detect.

vi) Defence use of nonces, mechanism - timestamps, encryption.

Denial of service attack

i) Defi°

ii) It's purpose is to disrupt services & make them inaccessible.

iii) Target - Servers, N/Ws or services.

iv) method - overloading system resources.

v) Easier to detect.

vi) Defence use of firewalls, mechanism - ratelimiting etc.

Key val = 3

G S G S

E K F R E K

E O E

∴ C.t = G S G S E K F R E K E

O E

Rail Fence c. → It's a trans. position c. that encrypts P.t by writing it in a zigzag

...sing old messages.
...difficult to detect.
...efence Use of nonces,
...hanism - timestamps, every
...-ption.

resources.
vi) Easier to detect
vii) Defence use of fire
mechanism - rate limiting

across imaginary walls
Pattern, & then reading it off row by row.

Single Column transposition C. →
It's a ~~en~~ method of encrypting a P.t by rearranging the letters of the P.t into a grid (row & cols) & then reading them off in a diff. order, usually determined by a keyword.

...diation — where some person ...
as financial transaction or ...
...sage ... one doesn't want to

Block

i) less complex.

ii) key length 128 or 256 bits.

iii) modes: ECB, CBC, CTR, CFB & OFB

iv) It converts plain text to cipher by taking plain text block at a time. (Defn).

v) slow.

vi) Uses both confusion & diffusion.

vii) Reversibility of encrypted text is hard.

viii) 64 or more bits are used.

Stream

i) is more complex.

ii) key length = 128 or 256 bits

iii) modes: LFB, OFB, CTR

iv) It converts P.t to C.t by taking 1 byte of P.t. at a time. (Defn).

v) fast

vi) Uses confusion but not diffusion.

vii) Reversibility of encrypted ~~bit~~ text is easy.

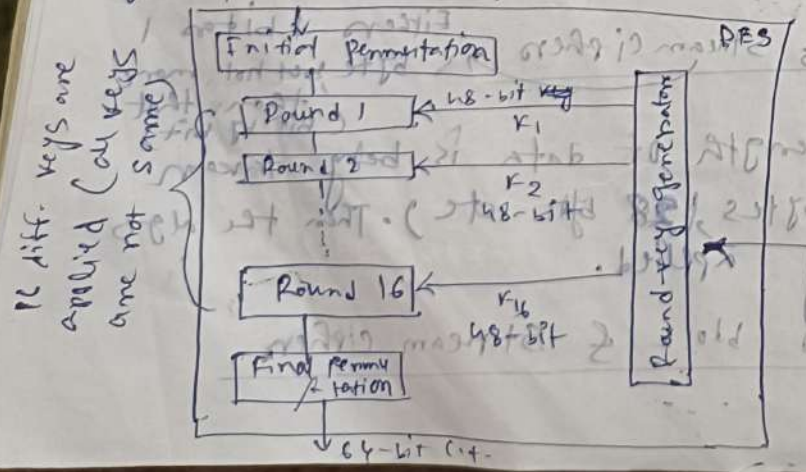
viii) 8 bits are used.

DES (Data encryption algo) Standard

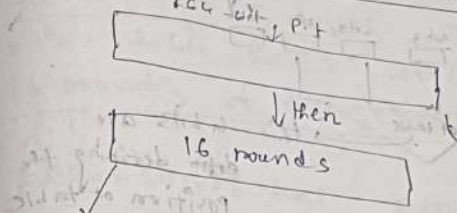
It's symmetric-key block cipher, with a 56-bit key length that has played a significant role in data security.

Algo

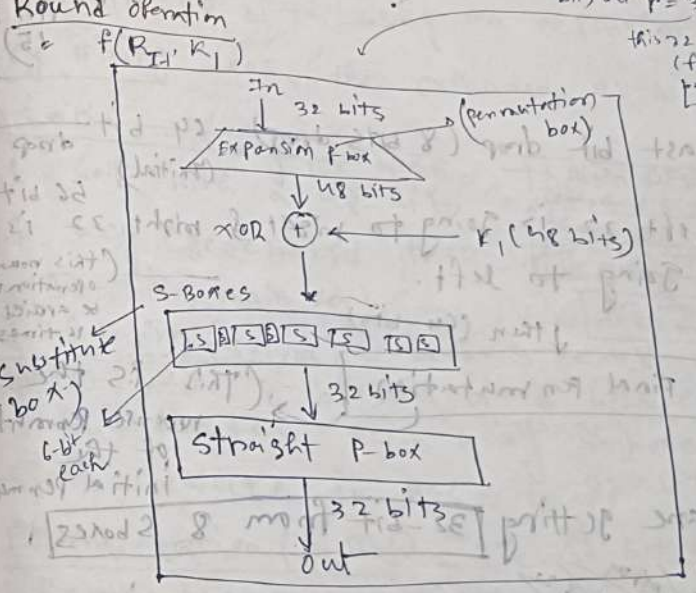
64-bit plain text

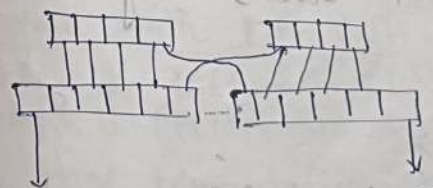


Initial & final Permutation

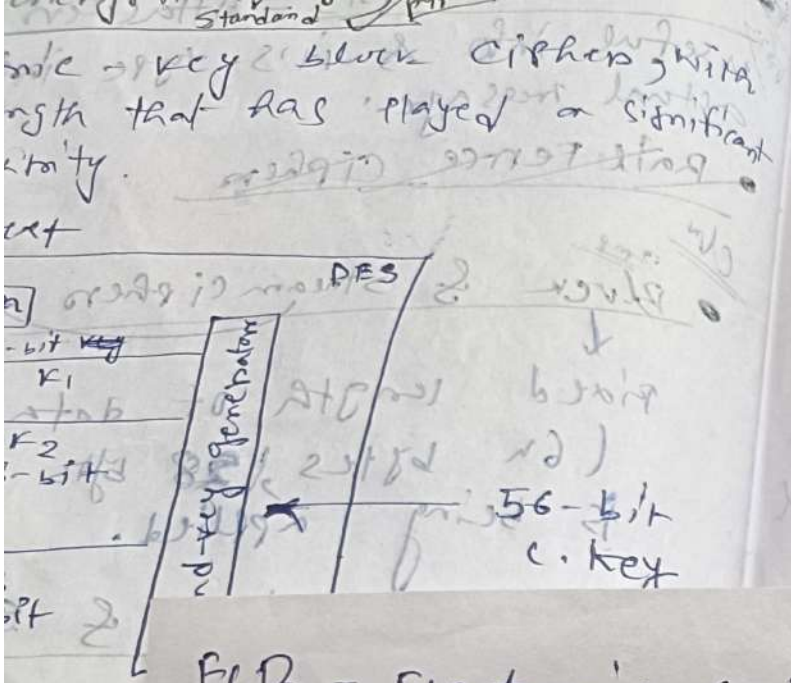


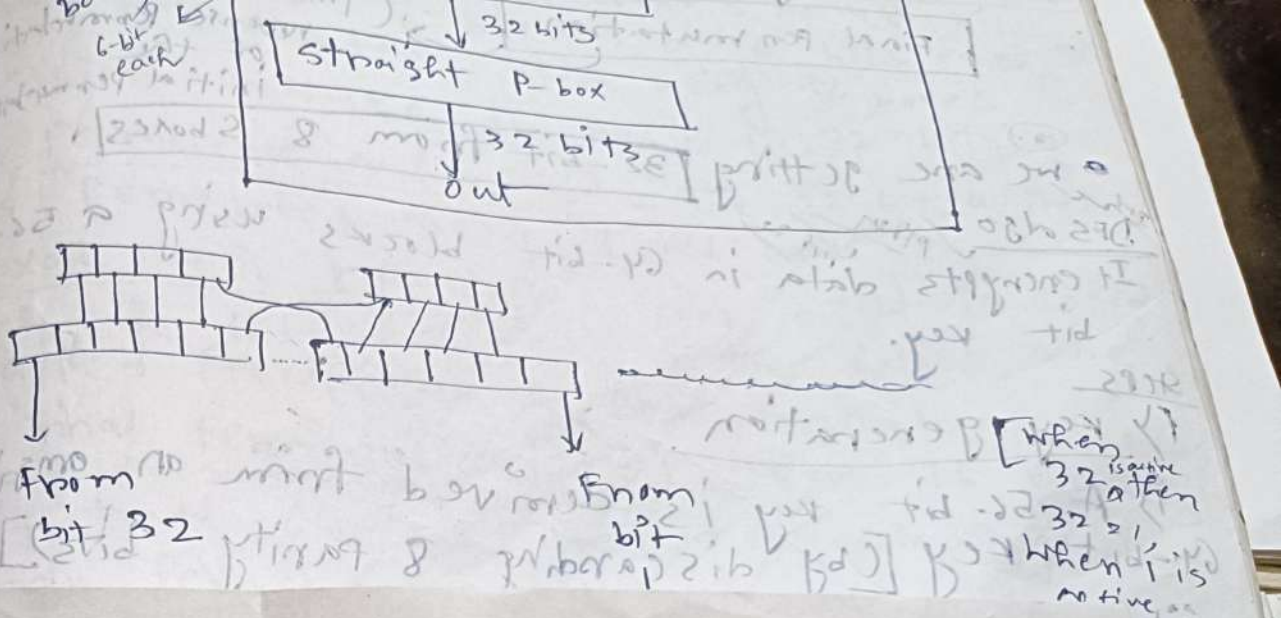
Round operation





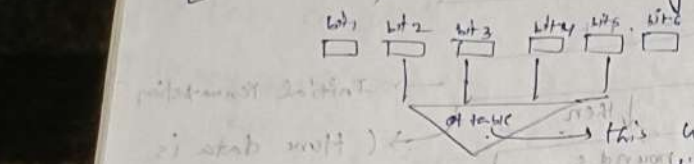
[When i is even then 32 = i, when i is odd i+1]





ECB = Electronic Codebook; CBC = cipher block chaining; CTR = Counter, CFB = Cipher feedback & OFB = Output feedback.

• [From 6 bit, 4 bit is coming.



• Last bit drop (8 bits drop) 64 bit (Initial) drop 56 bit

• [Left 32 is going to right & right 32 is going to left. (this round operation will be applied for 16 times)]

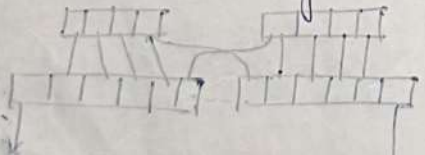
then (64-bits)

Final Permutation (This is the reverse Permutation of the initial permutation)

• we are getting 32-bit from 8 Sboxes.

DES algo:

It encrypts data in 64-bit blocks using a 56-bit key.



Steps

1) Key generation

i) A 56-bit key is derived from an original 64-bit key [(by discarding 8 parity bits)]

ii) The key is then divided into 2, 28-bit halves & transformed over 16 rounds.

2) Initial permutation (IP)

The 64 bits of block is permuted, using a fixed permutation table (to reorder the bits).

3) Splitting into 2 halves - The block is divided into 2, 32-bit halves: Left (L_0) & Right (R_0).

4) 16 rounds of encryption - Each round consists of:

i) Expansion (E-box) - Expands the 32-bit right half (R) into 48 bits using an expansion table.

ii) Key mixing - XORs the expanded 48-bit right half with a 48-bit round key (generated from the 56-bit main key).

Disadv - if key is weak.

→ After DES, AES comes & it's stronger than DES. If wanna implement (use python)

(this will be from the 1st round)

iii) Substitution (S-Box) - Breaks the 48-bit result into 8 blocks of 6 bits each, then substitutes them using S-boxes (which reduce it back to 32 bits).

iv) Permutation (P-Box) - The 32-bit output is permuted using a fixed table.

v) XOR with Left half - Then output is XORed with the left half (L_i), & the halves are swapped (except in the final round).

5) Final Permutation (FP-1) - After 16 rounds the final 64-bit result undergoes final permutation to produce the C.T.

• Disadv of DES

1) Key is weak. (Easy to break with modern computers)

iii) ~~can be b~~ The key is short (can be cracked by brute force).

ii) It works better in hardware than s/w.

iv) the block size is small, that's not secure for large data.

v) Can be hacked using Cryptanalysis. (2048 security)

vi) Replaced by AES & 3DES.

4/3/25

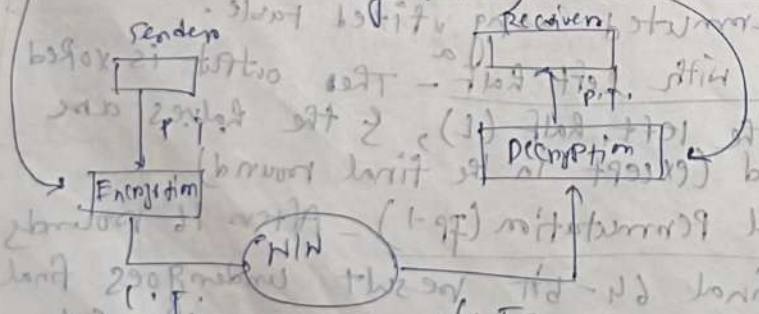
RSA → (Rivest, Shamir, Adleman) algo

Asymmetric key cryptography algo.

It's a public key encryption technique & is considered as the most secure way of encryption.

Public Pair key (N, e)

Private pair key (N, d)



Encryption (C) = $P^e \bmod N$

Decryption (P) = $C^d \bmod N$

Eg let my original message is F → [P]

N = 119

N = 119 e = 5 → (public key)

d = 77 (say) → (Private key)

Encryption

P.T → C = $(F)^5 \bmod 119$

= $(6)^5 \bmod 119$

= 41 → (C)

Here there will be normal alphabet no's A → 1, B → 2 ...

Decryption

PT → P = $(41)^{77} \bmod 119$ (Whenever the Case, will be like this) do

= 6 (= F) [By decryption we'll always get P.T → so write P.T if you're not getting ans by calculators]

Step 1 → PTD

Generate the RSA modulus - The initial procedure begins with selection of 2 prime nos namely p & q & then calculating their product N, as shown

N = p * q

S2:

Derived No. (e) - Consider no. e as a derived no. which should be > 1 & < (p-1) & (q-1). The primary condition is that there should be no common factors of (p-1) & (q-1) except 1

(Let N be a specified large no.)

S3
Public key – The specified pair of nos N & e forms the PSA Public key & it's made Public.

S4
Private key – Private key d is calculated from the nos p, q & e. The mathematical rep.

$$e * d = 1 \% \cdot (p-1)(q-1)$$

(Basic formula for extended Eudidean Algo)

Encryption formula

$$C = P^e \% \cdot N$$

[where, (N, e) is the public key of a P.T.o
P = P.T
C = C.T]

Decryption

$$P(P.T) = C^d \% \cdot N$$

Same as RSA algo (in step 5)

Not this

(a) How to calculate Private & Public keys?
(or) How to choose the Private & Public keys? PTO

(1) Choose 2 large Prime nos p & q
(2) Then calculate, $N = p * q$
(3) Choose e ($< N$) such that e & $(p-1) * (q-1)$ are relatively prime (means no common factors will be other than 1)
(4) Choose d such that $e * d \% \cdot (p-1)(q-1) = 1$

$e > 1$

Eg Assignment

(1) Using the RSA algo encrypt & decrypt the message: BE with key pairs (3, 13) & (5, 13). Given the 2 prime no. p = 19, q = 23. find N, e, d.

Sol^n p = 19, q = 23

$N = 19 \times 23$

$N = 437$

$e < 437$ such that e & $(18) * (22)$

e & 396

let, e = 13

$13 * d \% \cdot 396 = 1$

$d = 61$

(cross check $13 * 61 \% \cdot 396$
$= 793 \% \cdot 396$
$= 1$)

using the prev algo (Eclass)

P.T.O (see next)

Sol^n p = 17, q = 11

N = 187, e = 3, d = 107

Encryption

B,

Decryption C = 5

E,

RSA algo (Rivest, Shamir, Adleman)

1. Choose two diff. large prime nos. p, q
2. Calculate N = p * q
3. Calculate $\phi(N) = (p-1)*(q-1)$
4. Choose 'e' such that $1 < e < \phi(N)$, e is co prime to $\phi(N)$ (means gcd(e, $\phi(N)$) = 1)
5. Calculate d, such that $de \equiv 1 \bmod \phi(N)$
6. Public key e', private key 'd' for N

[Co prime = Relatively prime = no common factors b/w e & $\phi(N)$ except 1]

7. Encryption → $P^e \bmod N = C$, decryption → $P = C^d \bmod N$

[P = P·1, C = C·1]

Same as $de \bmod \phi(N) = 1$

P·1·0

(2) p = 19, q = 23

∴ N = p * q = 19 * 23 = 437 ∴ N = 437

$\phi(N) = (p-1)*(q-1)$

= 18 * 22

= 396

let e = 13

1 < 13 < 396

gcd(13, 396) = 1 (13, 396 / 13, 396)

∴ e = 13

gcd(a, b) = 1

∴ Co prime / Relatively prime

$de \equiv 1 \cdot \phi(N)$

(can be written as)

$de = 1 + k \cdot \phi(N)$ [k = constant starts from 0, k = 0, 1, 2 ——]

$\therefore d = \frac{1 + k \cdot \phi(N)}{e}$

$= \frac{1 + 0 * 396}{13}$ [k = 0]

= 0 ↳ (decimal is, fraction is coming so we're not considering it)

k = 1

$d = \frac{1 + 396}{13} = 30.53$ (x)

k = 2

$d = \frac{1 + 2 * 396}{13} = 61$ (decimal, absolute value, so we're considering it)

∴ d = 61

(Cross check = 61 * 13 = 1 · 396 = 1) (1) 22/9/25

Imp IDEA (International Data Encryption Algo)

This is a system cryptography algo. where;

i) P.T = 64 bits

ii) key size = 128 bits (into 52 subkeys)

iii) 8 (Rounds) + Last o/p transformation

Advs & disadvs of RSA

Advs – i) secure with large key sizes.
ii) No need to share secret keys
iii) supports digital signatures.
iv) Widely accepted & used.

Disadvs – i) slower than symmetric algs.
ii) Requires large keys for strong security
iii) Not ideal for encrypting large data.

iv) High Computational Cost.

In each round there are 14 steps.

P_1	P_2	P_3	P_4
16 bits	16 bits	16 bits	16 bits

We 6 keys → $K_1, K_2, K_3, K_4, K_5, K_6$ in each round.

Step 1 → $P_1 * K_1$

Step 2 → $P_2 + K_2$

Step 3 → $P_3 + K_3$

Step 4 → $P_4 * K_4$

Step 5 → Step 1 ⊕ Step 3

Step 6 → Step 2 ⊕ Step 4

Step 7 → Step 5 $* K_5$

Step 8 → Step 6 + Step 7

Step 9 → Step 8 $* K_6$

Step 10 → Step 7 + Step 9

Step 11 → Step 1 ⊕ Step 9 → R_1

Step 12 → Step 3 ⊕ Step 9 → R_2

Step 13 → Step 2 ⊕ Step 10 → R_3

Step 14 → Step 4 ⊕ Step 10 → R_4

R_1 R_3 R_2 R_4

After the rounds the sequence will be

R_1 R_2 R_3 R_4

O/P transformation

$R_1 * K_{48} \rightarrow C_1$

$R_2 + K_{50} \rightarrow C_2$

$R_3 + K_{51} \rightarrow C_3$

$R_4 * K_{52} \rightarrow C_4$

→ ultimate o/p

- There are 6 keys in each round so within the 8 rounds, there are 48 keys.

- **Digital signature** – It's a cryptog. mechanism that verifies the authenticity & integrity of digital docs/ messages.

The major aspects are –

1. Privacy
2. Integrity
3. Authentication
4. Non ~~repudiable~~ repudiation

→ these 3 are achieved by digital signature

- **Digest**

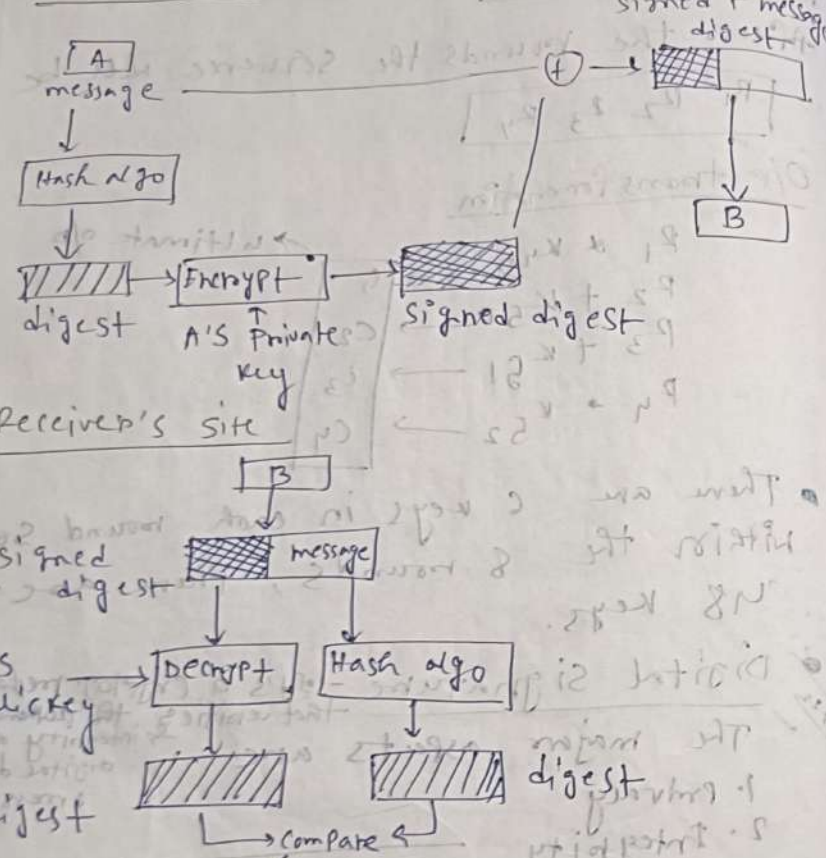
From the original message we get a fixed length message, which we call digest.

Hash algo

→ MD5 (120 bit digest) [message digest algo 5]

→ SHA (160 bit digest) [Secure hash algo]

Sender's site



(4) Assim

Explain how integrity, non repudiation authentiaty are achieved by using digital signature.

Imp

- user authenticatim

25/4/25

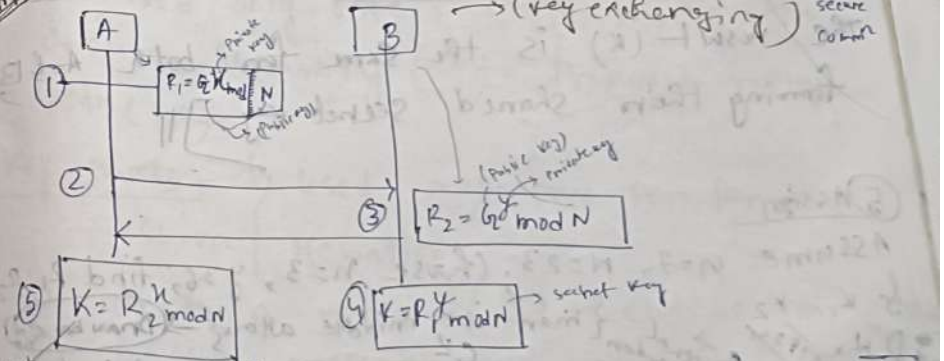
It is the process of confirming the identity of an user [in cryptography & n/w security] to allow secure access to systems.

Key Points - i) Ensures the user is genuine.
ii) Protects against impersonation.
iii) Often combined with authorization for full security.
iv) Can involve single [multi-factor authentication] methods: password, biometrics etc.

Imp

- Diffie hellman Protocal (DH) — key exchanging Protocol to secure comm

Imp



G & N are Public. (Public) not

x^A, y^B are not Private [G, N = Random no]

$K_1 = R_2^x \bmod N$

$= (G^y \bmod N)^x \bmod N$

$K_1 = G^{xy} \bmod N$

Either algo [numerical based on the

Algo

	A	B
(with agreed upon G, N) Step ①	Public keys available = G, N	Public keys available = G, N
Step ②	Private key selected = x	Private key selected = y
Step ③	key generated $R_1 = G^x \bmod N$	key generated $R_2 = G^y \bmod N$
	Exchange of generated keys takes place	
④	key received = R_2	key received = R_1
Step ⑤	Generated secret key $K = R_2^x \bmod N$	Generated secret key $K = R_1^y \bmod N$
	Algebraically, it can be shown that $K_1 = K_2$	
	Users now have a symmetric secret key to encrypt	

Steps

i) public values – A large Prime no. N & a primitive root G modulo N (G) are agreed on & made public.

ii) Private keys – A picks a secret no. x & B picks a secret no. y (These are their Private keys).

iii) Public keys – A Computes $R_1 = G^x \bmod N$ & sends R_1 to B. B computes $R_2 = G^y \bmod N$ & sends R_2 to A.

iv) Shared secret – A Computes $K = R_2^x \bmod N$; B Computes $K = R_1^y \bmod N$

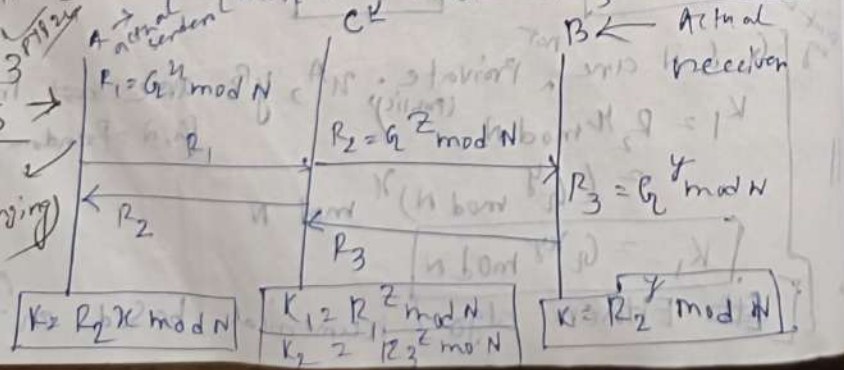
The result (K) is the same for both A & B forming their shared secret key.]||

⑤ Assign

Assume G=7, N=23. Choose x=3, y=6; find R_1, R_2 & K_1, K_2.

D H with 3 users → (Key Exchanging)

{man in middle attack} → Drawback



• Primite root – The primitive root of a prime no. n is an int r between [1, n-1] such that the values of $r^x \pmod n$, where x is in range [0, n-2] are different.

Ex I/P = 7 (n)

now, the smallest primitive root = 3 ← how

$3^0 \,\%\, 7 = 1$
$3^1 \,\%\, 7 = 3$
$3^2 \,\%\, 7 = 2$
$3^3 \,\%\, 7 = 6$
$3^4 \,\%\, 7 = 4$
$3^5 \,\%\, 7 = 5$

} they all are diff

• Nonce – A nonce is a random / unique no. that is used only once in Cryptography to ensure security (eg to Prevent replay attacks)

Key Points – i) Used only once per ~~Person~~ Session / operation.

ii) Often a random / sequential no.

iii) Ensures messages are fresh & not reused.

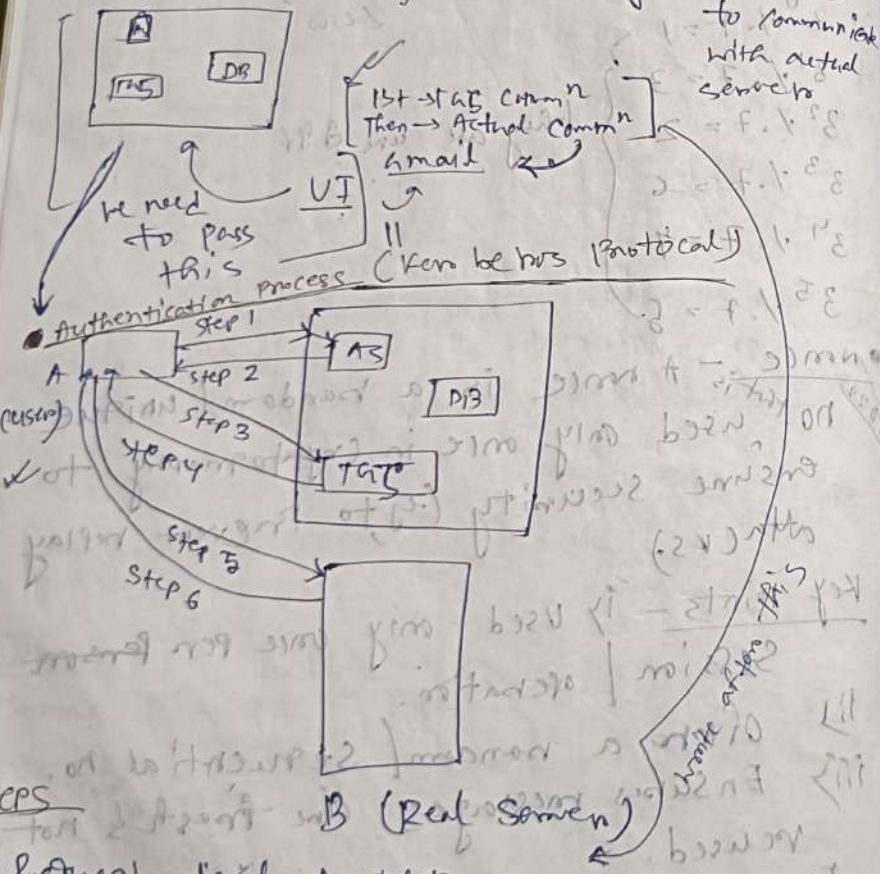
Eg. Authentication, encryption etc.

28/4/25

• Kerberos protocol → To auth. one to server & server to server.

3 components

1) AS (Authentication server) – Authentication
2) TGS (ticket generating server) → to grant the ticket
3) DB (Data base) granting

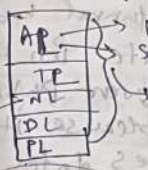


Steps

1. Request ticket to TGS
2. TGS session key & ticket for TGS server
3. Request ticket for the server
4. AB session key & ticket for B
5. Request service
6. Provide service

• once the user is verified by AS, the user can comm^n with diff. servers with diff. session keys.

• Security protocol (IPSec)



• layers: IP → IPSEC → AH (Authentication header) (Internet Protocol security)
TL → TLS → ESP (Encapsulating security payload)
AL → PGP

• Systems comm^n with c/a → systems are assigned with 64-bit addresses. (Unique IP address)

Host = user

• IP layer comm^n → Connectionless Protocol

• SA (security association) (In IPSEC Protocol)

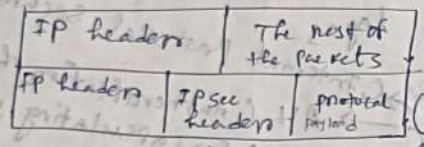
↳ It's also called Signalling Protocol. It's establishing the logical connection with the host before starting the actual comm^n.

Key points –
i) The components – SPI (Security Parameters index (32-bit))
ii) The types of Protocol is for security. (AH or ESP)
iii) Source Surveillance
iv) mode – Transport mode & tunnel mode

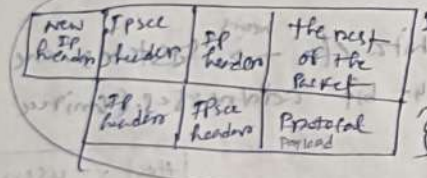
• SA, Comm^{n} (whether uni / bi directional)

If uni → only 1 SA is needed.
If bi → 2 SA are needed.
(Full / half duplex mode)

• transport mode – (The IPSEC tunnel mode is appropriate for sending data over public n/ws because it improves data security (against unauthorised parties).)



• Tunnel mode – It encrypts only the data packet payload while leaving the IP header unchanged.

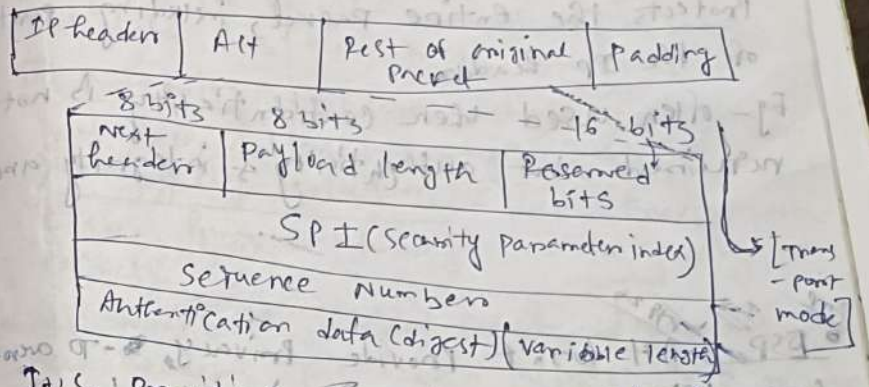


} → this will be treated as payload.

Diff

Transport mode	Tunnel mode
i) Extends to the payload of an IP packet	i) Provides protection to the entire IP packet
ii) typically used for end-to-end comm^{n} b/w 2 hosts.	ii) the entire original packet travels through a tunnel from one point of an IP n/w to another.
iii) ESP encrypts & optionally authenticates the IP payload but not the IP header.	iii) used when one / both ends of a security association are a security gateway.
iv) less secure	iv) more secure

• AH (Authentication header protocal)



It is providing source authentication. It authenticate source & provide the integrity of packets. But it is not able to provide the privacy. (only auth. & integrity is maintained)

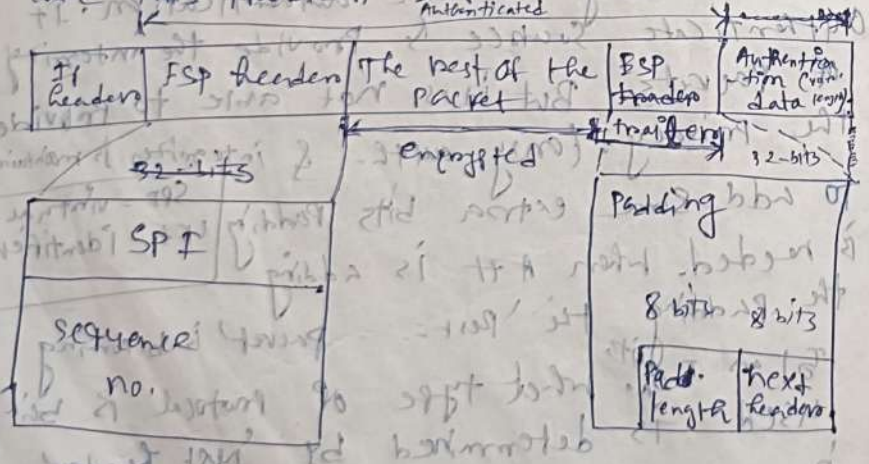
To add some extra bits Padding is needed. When AH is adding the padding the 'Rest ... packet' is becoming 32 bits. What type of protocol is being used is determined by 'next header'.

SPI = virtual ckt identifier

Reserved bits is acting as virtual ckt identifier. ~~SPI means acts as virtual ckt identifier~~. Sequence no uniquely identifies the message / packet.

key features - i) Authentication - verifies that the Packet comes from a trusted source.
ii) Integrity - Ensures that the contents of the Packet haven't been altered (in transit).
iii) No encryption - AH doesn't Provide confidentiality (Privacy)
iv) Covers most of the IP packet - AH Protects the entire packet, including parts of the IP header.
Eg - often used when confidentiality is not required, but authenticity & integrity are.

- ESP - AH doesn't provide Privacy, so to overcome this Problem ESP introduced.



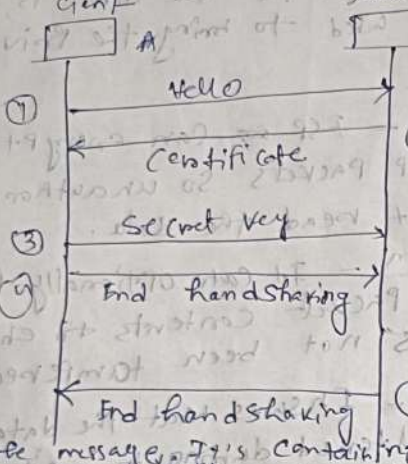
Here the auth. data is calculated by ESP header, the rest of packet, ESP trailer. (Hashing digest)
What type of header is being used is determined by next header. Here, the diff. is encryption is being used to bring the Privacy.

key features -
i) Encryption - ESP can encrypt the payload of IP packets so unauthorized Parties can't read the data.
ii) Authentication - It can optionally provide auth. of the packet contents to ensure the data has not been tampered with.
iii) Integrity - Ensures that the data has not been altered during transit.
iv) Anti-replay Protection - Helps to Prevent attackers from capturing and resending packets.

- ESP wraps the original packet data & adds a new header. This header includes info required for decryption & verification.
- IP header size = 50 - 51 bytes → AH header
- IP - size = 32 bits → ESP header

29/4/25

- TLS (Transport layer Security Protocol)
  SSL → while banking transaction is done, then it's being stored.
  → Handshake Protocol & Data exchange Protocol.



i) 'Hello' is the message. It's containing what world be the protocol etc.

ii) The certificate is containing Public key of the servers, which is encrypted by CA's Private key. The server is doing (Certification authority) authentication.

iii) The client is sending a secret key. That secret key is encrypted by Server Public key. (The decryption) is done by CA's public key.

iv) The client is sending a message of that message is encrypted by servers's Public key. After that the decryption is done by server's ~~Public~~ Private key of message.

v) The encrypted message is coming to client & the client side is doing the decryption.

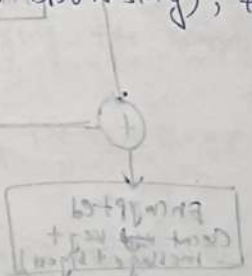
TLS is a cryptographic Protocol used to secure comm^n over a n/w. It ensures that data sent b/w 2 parties (like a web browser & a server) is —

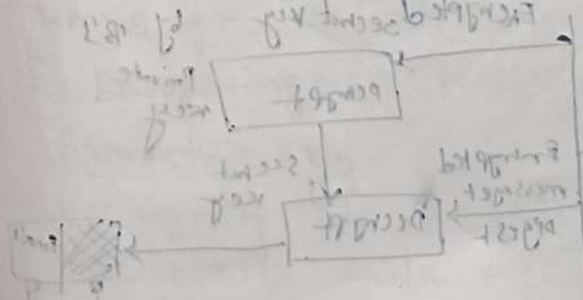
i) Encrypted. — [to prevent eavesdropping] ||

ii) Authenticated — [to confirm the identity of the comm^n parties.] ||

iii) Tamper-Proof — [to detect if data is altered during transmission.] ||

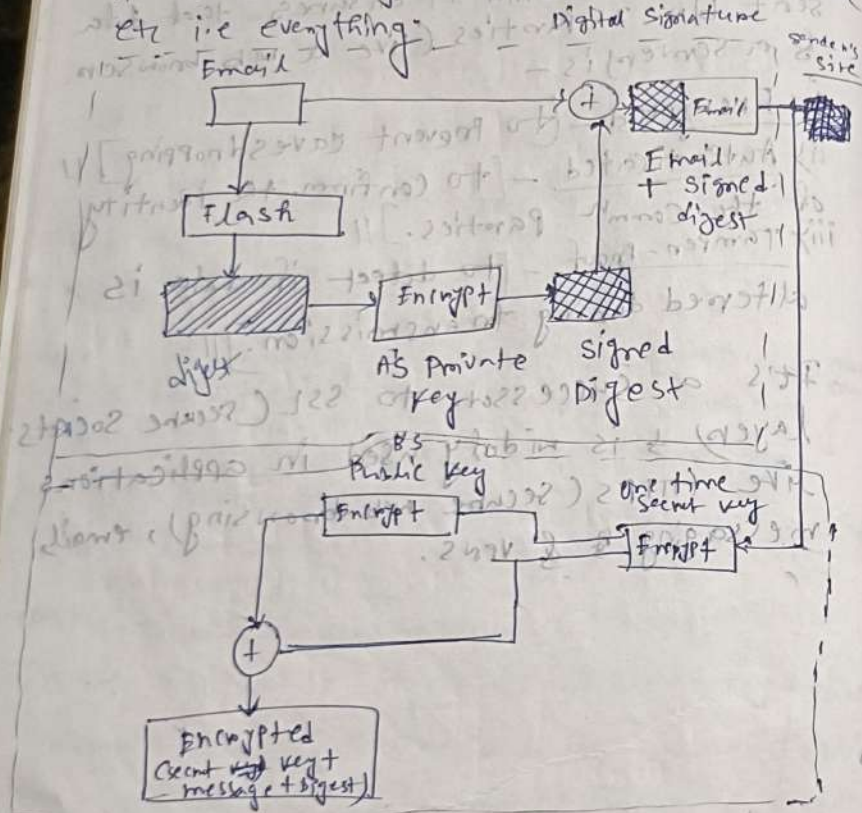
It's a successor to SSL (Secure Sockets layer) & is widely used in applications like HTTPS (secure web browsing), email, messaging & VPNs.





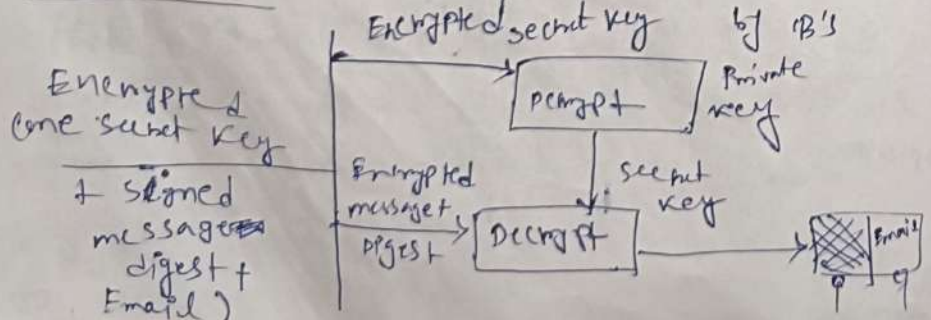
PGP (Pretty good Privacy) Protocol

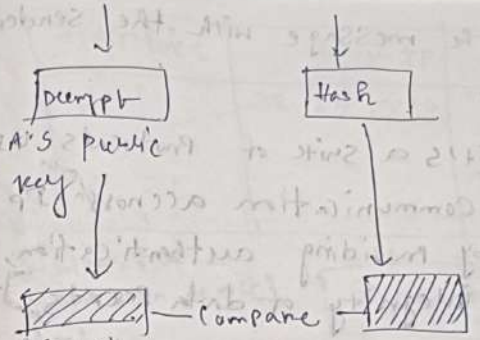
It gives Privacy, integrity, non-repudiation etc i.e everything.



Privacy

Receiver site





If the encryption is not being done by opposite key, then there is something (or vice versa) wrong

It's a cryptographic protocol/system used to secure emails & files by providing -

i) Confidentiality - [Using encryption to keep data private.]

ii) Authentication - [verifying the sender's identity.]

iii) Integrity - [Ensuring the message wasn't tampered with.]

iv) Non-repudiation - [Proving that the sender actually sent the message.]

- It uses a combination of symmetric encryption & asymmetric encryption. It works by -

i) Encrypting the message with a random symmetric key.

ii) Encrypting that symmetric key with the recipient's public key.

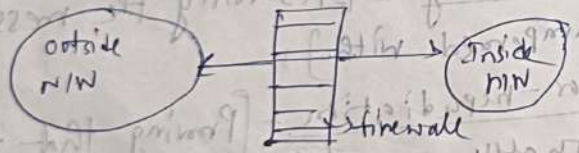
by signing the message with the sender's private key.

- IPsec – It's a suite of protocols used to secure communication accross IP networks by providing authentication, encryption, & integrity of data packets over IP

5/5/23

- Firewall

It acts as a protection of our system, whenever the system is vulnerable to get attacks. It's main purpose is to give the protection & security from other n/ws. There are 2 types of firewall is packet based & its proxy based firewall.



1) Packet based – It's filtering the packets. It filters at the n/w & transport layer. A packet-based firewall filters n/w traffic by checking each data packet's basic info (like IP address & port) to allow / block it's without tracking previous activity.

Eg – IP tables etc

Eg

Interface	Source IP	Destination IP
1	38.129.15.10	

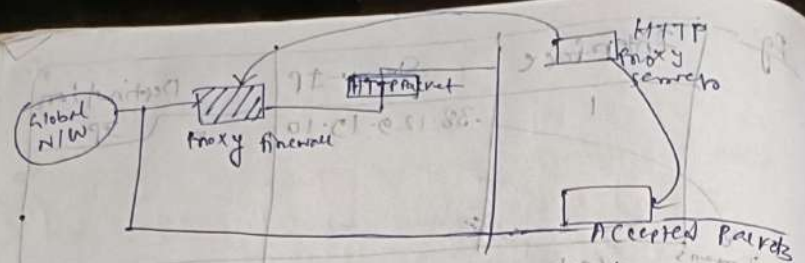
(It means it'll block that particular device)

Source port	Destination port

- Firewall can be in both h/w & s/w system.

2) Proxy firewall – (Here, whenever we're receiving the data then we are able to understand / know the info.) It filters at the application layers. If the message isn't legitimate, then it blocks it (doesn't allow to pass the message.

A Proxy firewall filters traffic by acting as an intermediary b/w users & the internet. It inspects data at the application level, improving security by hiding internal n/w details.



Eg- Squid proxy etc.

• Diff b/w them

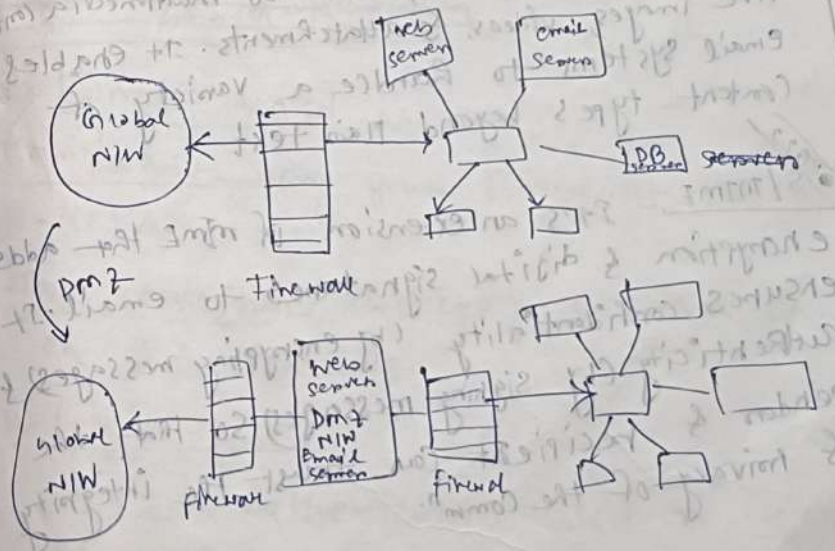
Features	Proxy servers	Proxy Firewall
i) Working Principle	Filters traffic based on IP addresses, ports & protocols.	Acts as an intermediary b/w users & the internet.
ii) OSI layer	Operates at n/w & transport layer.	Operates at application layer.
iii) Speed	Faster	Slower
iv) Security	Basic security	High security
v) Eg	IP tables etc	Web Proxy etc
vi) Alternative	Stateless firewall	Application firewall

Packet firewall

DMZ N/W — demilitarized

It stands for ~~militarized~~ N/W. It is providing extra layer of security. demilitarized zone

It's a buffer zone b/w an internal n/w & the internet. It holds publicly accessible servers (e.g., web servers) & uses cryptography to secure comm^{n}s b/w the DMZ & both external & internal n/w, preventing direct access to the internal n/w from external threats.



• MIME → It's multipurpose internet mail extensions

S/MIME → Secure multipurpose

(MIME) It's used for mail comm^n etc (to send voice image etc sending). It's drawbacks was it couldn't provide the security. That's why S/MIME came. It gives integrity auth. non-repudiation etc. so, it is more powerful than MIME.]||

• MIME - It is a standard that allows email to include not just text, but also multimedia content like images, videos, & attachments. It enables email systems to handle a variety of content types beyond plain text.

• S/MIME - It's an extension of MIME that adds encryption & digital signatures to email. It ensures confidentiality (by encrypting messages) & authenticity (by signing messages) so that the sender & recipient can trust the integrity & privacy of the comm^n.

• different types of authentication -

i) knowledg-based -

Password: A secret word / Phrase that only the user knows.

PIN - A short numerical code known only to the user.

ii) Possession-based -

Smartcards - physical cards, used to access systems / info.

mobile phones - Used for two-factors authentication like receiving sms (codes).

~~different types of authentication~~

iii) Biometric based

Fingerprint - Using your fingerprint for identification.

Face recognition - Identifying a person based on facial features.

iv) Location-based -

GPS - Authentication based on the user's physical location.

v) Behavioral-based -

typing patterns - Analyzing how you type to verify your identity.

Voice recognition - verifying identity based on the way you speak.

3DES - 3DES (Triple DES) is a type of encryption used to keep data secure. It is an upgraded ver. of the older DES. It applies DES 3 times with 3 diff keys, making it much harder to crack.

Algo

i) Step 1 - Encrypt the data with key 1.
ii) Step 2 - Decrypt the ~~data~~ result with key 2.
iii) Step 3 - Encrypt again with key 3.

This is called the 'Encrypt - Decrypt - Encrypt' (EDE) Process.

It's like $\boxed{\text{Encrypted data} = E(K_3, D(K_2, E(K_1, P_t)))}$