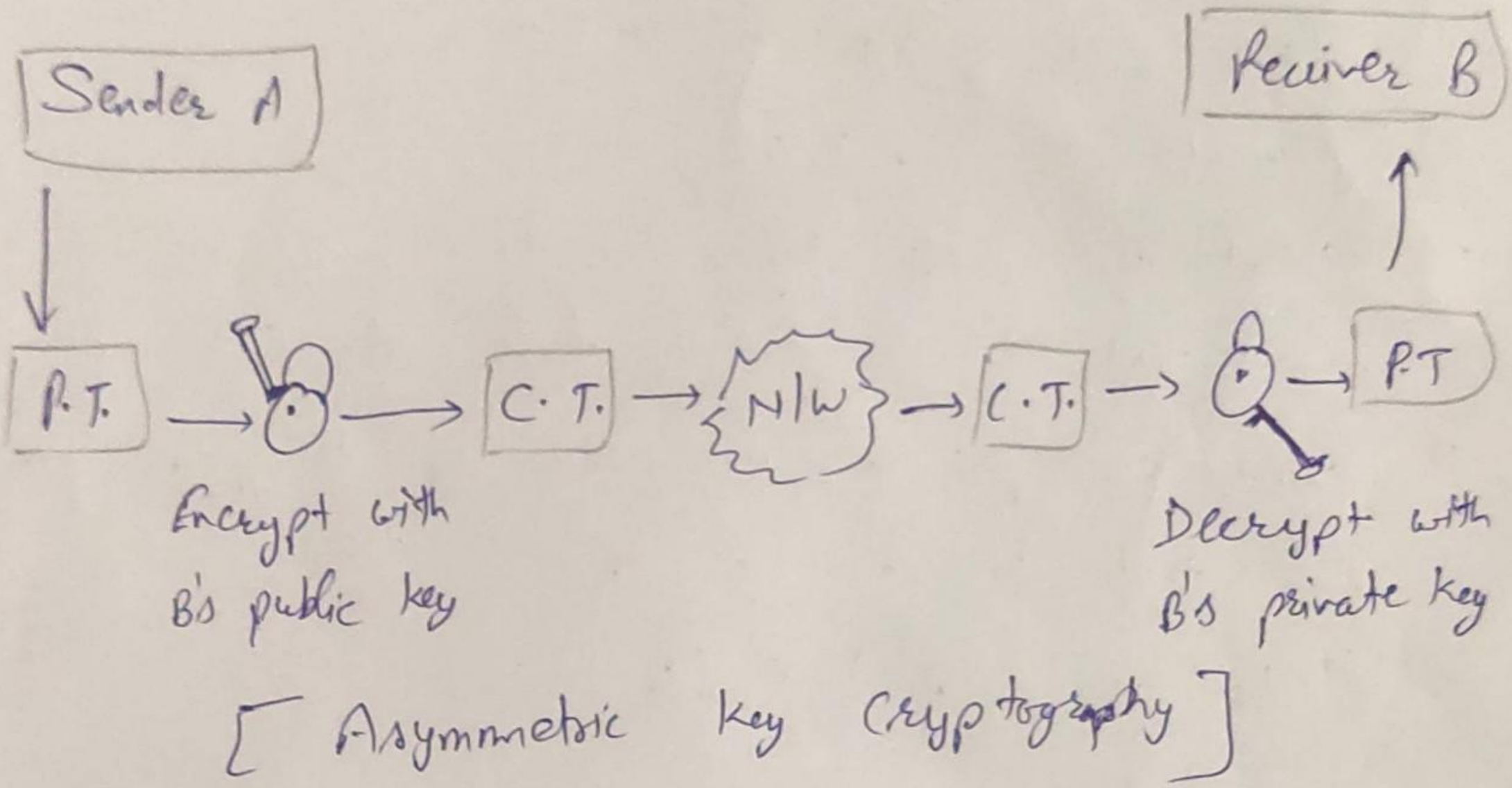
UNIT — III

Public key Cryptosystems with Applications

* Overview of Asymmetric key Cryptography:

It is also called Public key Cryptography. two diff. Keys are used. One key is used for encryption and only other corresponding key must be used for decryption.



[Asymmetric key Cryptography]

* The RSA Algorithm:

The RSA algo is the most popular and proven asymmetric key cryptographic algo.

A prime no. is one that is divisible by 1 and itself. For instance, 3 is a prime no., because it can be divided only by 1 or 3.

The RSA algo. is based on the mathematical fact that it is easy to find and multiply large prime no. together, but it is extremely difficult to factor their product. The private and public keys in RSA are based on very large prime no. However, the real challenge in case of RSA is the selection & generation of the public and private keys.

1) Choose two large prime no. P and Q.

2). Calculate $N = P \times Q$

3) Select the public key (i.e the encryption key) E such that it is not a factor of $(P-1)$ and $(Q-1)$.

4) Select the private key (i.e. the decryption key) D such that the following eq? is true:

$(D \times E) \bmod (P-1) \times (Q-1) = 1$

5) For encryption, calculate the C.T. from the P.T.

$CT = PT^E \bmod N$

6) Send CT as the Cipher Text to the receiver.

7) for decryption, calculate the plain text P.T from C.T.

$PT = CT^D \bmod N$

Q. 1) Choose two large prime no. P and Q.

Let P = 7 and Q = 17

2) Calculate N = P × Q

N = 7 × 17 = 119

3) Select the public key (i.e. the encryption key) E such that it is not a factor of (P-1) × (Q-1)

- Let us find (7-1) × (17-1) = 6 × 16 = 96
- the factors of 96 are 2×2×2×2×2×3.
- Thus we have to choose E such that none of the factor of E is 2 and 3.
  if we choose E as 4 then 2 is a factor of it.
  If we E as 6 then 2 & 3 is a factor of it.
  If we E as 15 then 3 is a factor.
- Let us choose E as 5 then both 2 and 3 not a factor of it.

4) Select the private key (i.e. the decryption key) D such that the following eq. is true:

(D × E) mod (P-1) × (Q-1) = 1

- Let us substitute the values of E, P and Q in eq".
- It have: (D × 5) mod (7-1) × (17-1) = 1
- i.e. (D × 5) mod (6) × (16) = 1
- i.e. (D × 5) mod (96) = 1

- After some calculation, Let us take D = 77. then following is true:

$$(77 \times 5) \bmod (96) = 385 \bmod 96 = 1$$

5) for encryption, calculate the C.T from P.T

$$CT = PT^{E} \bmod N$$

Let us assume that we want to encrypt P.T 10.

then

$$CT = 10^{5} \bmod 119$$

$$= 100000 \bmod 119$$

$$= 40$$

6) Send CT as the Cipher text to the receiver.

Send 40 as the CT to the receiver.

7) for decryption, Calculation the PT from the CT

$$PT = CT^{D} \bmod N$$

- $PT = 40^{77} \bmod 119$

$$= 10$$

which was the original PT of step 5.

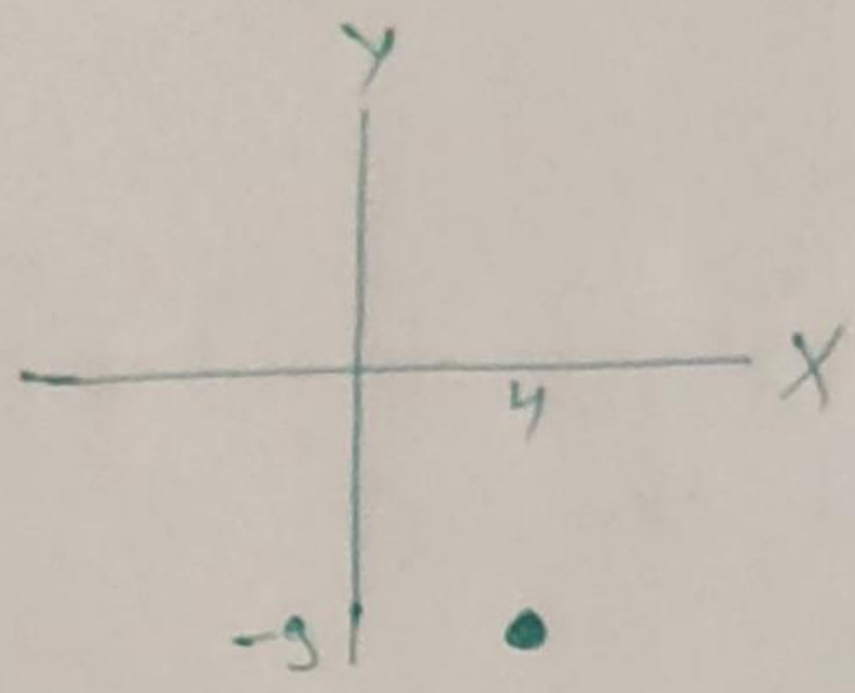
* Symmetric and Asymmetric key Cryptography Together :

Characteristic	Symmetric key Cryptography	Asymmetric key Cryptography
1) Key used for encryption / decryption	Same key is used for encryption and decryption	One key used for encryption and another, diff. key is used for decryption.
2) Speed of encrypt^{n} & decrypt^{n}	Very fast	Slower
3) Size of resulting encrypted text	Usually same as or less than the original clear text size	More the original clear text size
4) key agreement / exchange	A big problem	No problem at all
5) No. of keys required as compared to the no. of participants in the msg. exchange.	Equals about the square of the no. of participants So scalability is an issue.	Same as the no. of participants, so scale up quite well.
6) Usage	Used for encryption, decryption, not for digital signature.	Used for encryption, decryption and digital signature also.

* Elliptic Curve Cryptography (ECC):

RSA is the most prominent algo used in public key cryptography techniques for encryption and digital Signatures.

The main difference b/w RSA and ECC is that unlike RSA, ECC offers the same level of security for smaller key sizes. ECC is highly mathematical in nature therefore, we just have an overview.



[Point with reference to x and y axes]

An elliptic curve is similar to a normal curve drawn as a graph on x and y-axes. It has points. Each point can be designated by an (x,y) coordinate.

for instance, a point can be designated as (4, -9), which means that it is 4 units on right hand side of x-axis from the center and 9 below the y-axis from the center.

Consider an elliptic curve (E) with a point P. Now generate a random no. d. Let we have $Q = d \times P$. E, P and Q are public values and the challenge is to find d. This is called as elliptic curve discrete logarithm problem. As long as the curve is big enough, it is almost impossible to find d. Thus, E, P, Q together form the public key and d is the corresponding private key.

* ElGamal Cryptosystem :

The Elgamal technique is a public key algo., which can be used for both : digital signatures as well as encryption.

To generate a key pair, first select a prime no. p and two random no. g and x, so that both g and x are less than p. Then find out $y = g^x \bmod p$.

The public key becomes y, g and p. Both g and p can be shared in a group of users. The private key is x.

For encrypting a plain text msg M, first select a random no. k such that k is relatively prime to p-1. Then find out following:

$$a = g^k \bmod p$$

$$b = y^k M \bmod p$$

Here $M = (ax + kb) \bmod (p-1)$. Then the pair (a, b) becomes the C.T.

Note: C.T. is double the size of P.T.

To decrypt, (a, b) to find out the P.T. M,

Calculate $M = b/a^k \bmod p$.

* Rabin Cryptosystem:

→ it is a public key cryptosystem invented by Michael Rabin

→ it is uses asymmetric key encryption for comm^n b/w two parties and encrypting the msg.

→ it is a variation of RSA.

→ It is as ~~algo~~ secure as RSA.

→ RSA is based on exponentiation congruence; Rabin is based on quadratic congruence.

→ It uses public key = n

→ private key tuple (p, q)

→ Everyone can encrypt a msg. using n; only Bob can decrypt the msg. using p and q.

→ Decryption of msg. is infeasible because she does not know the values of p and q.

Procedure :

Key generation, Encryption & Decryption.

Key Generation :

1) Generate two very large prime no. p and q which satisfies the condition

$$p \neq q \rightarrow p \equiv q \equiv 3 \pmod 4$$

for Eg. $p = 139$ and $q = 191$

2) Calculate the value of n

$$n = p \cdot q$$

3) publish n as public key and save p and q as private key.

Encryption :

1) Get the public key n.

2) Convert the msg to ASCII value. Then convert it to binary and extend the binary value with

itself, and change the binary value back to decimal m.

3) Encrypt with the formula:

$$C = m^2 \bmod n$$

4) Send C to recipient.

Decryption:

1) Accept C from sender.
2) Specify a and b with extended Euclidean GCD such that, $a \cdot p + b \cdot q = 1$
3) Compute r and s:

$$r = C^{(p+1)/4} \bmod p \quad , \quad s = C^{(q+1)/4} \bmod q$$

4) Now Calculate X and Y

$$X = (a \cdot p \cdot r + b \cdot q \cdot s) \bmod p$$
$$Y = (a \cdot p \cdot r - b \cdot q \cdot s) \bmod q$$

5) The four roots are $m1 = X$, $m2 = -X$, $m3 = Y$, $m4 = -Y$
Now, Convert them to binary and divide them all in half.

6) Determine in which left and right half are same. Keep that binary's one half and convert it to decimal m. Get the ASCII characters for decimal value m.
The resultant char. gives the correct msg. sent by sender.

Decryption: The decryption is based on the solution of quadratic eq^n. Because the received C.T. is the square of P.T., it is guranteed that C has roots. So Chinese remainder theoran is used to find the four square roots. It is not deterministic.

The decryptim has four ansmes. It is upto the receiver of the msg. to choose one of the four as final ansmes. In many situations, the receiver can easily pick up the right assume.

$$a_1 = + (CT^{(p+1)/4}) \bmod p$$

$$a_2 = - (CT^{(p+1)/4}) \bmod p$$

$$b_1 = + (CT^{(q+1)/4}) \bmod q$$

$$b_2 = - (CT^{(q+1)/4}) \bmod q$$

$$P_1 = CRT(a_1, b_1, p, q)$$

$$P_2 = CRT(a_1, b_2, p, q)$$

$$P_3 = CRT(a_2, b_1, p, q)$$

$$P_4 = CRT(a_2, b_2, p, q)$$