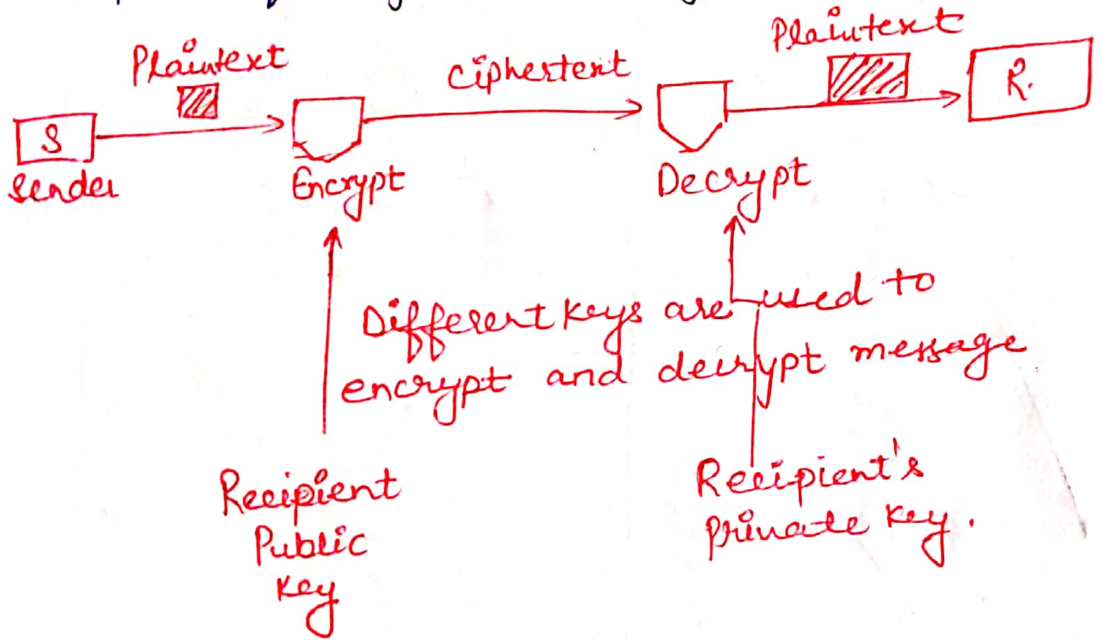
UNIT-III

Public key Crypto-System :-

Unlike symmetric key cryptography, we do not find historical use of public-key cryptography. With the spread of more unsecure computer networks in last few decades, a genuine need was felt to use cryptography at larger scale. The symmetric key was found to be non-practical due to challenges it faced for key mgmt. This gave rise to the public key cryptosystems.

The process of encryption & decryption is depicted as :-



Major Points -

- Different keys are used for encryption & decryption.
- Each reciever possesses a unique decryption key, generally refferred to as his/her private key
- Reciever needs to publish ~~his~~ an encryption key, referred to as his public key.
- Such cryptosystem involves trusted third party which certifies that a particular public key belongs to a specific person or entity only.

• Though private & public keys are related mathematically, it is not be feasible to calculate the private key from the public key.

Six Ingredients – Plaintext, Encryption Algorithm, Public key, Private key, Cipher text, Decryption Algorithm.

Applications of Public key Cryptosystems –

↳ Digital Signature – Content is digitally signed with an individual's private key & is verified by the individual's public key. (Provides → Authentication, Non-Repudiation, Integrity)

↳ Encryption – Content is encrypted using an individual's public key & can be decrypted by individual's private key. (Provides – confidentiality, Integrity)

RSA (Rivest - Shamir - Adleman) - RSA was first publicly published in 1977 by Ron-Rivest, Adi Shamir and Leonard Adleman of MIT.

But Diffie & Hellman introduced a new approach of public key cryptography as pioneer but RSA was supreme as the most widely accepted and implemented general purpose approach.

→ Plaintext is encrypted in blocks, with each block having a binary value less than some number 'n'. That is, the block size must be less than or equal to $\log_2(n)$ i.e in practical, the block size is 'i' bits, where $2^i < n \leq 2^{i+1}$.

<u>Hence</u> -

$$\boxed{\begin{aligned} C &= M^e \bmod n \\ P &= C^d \bmod n \end{aligned}}$$

Both sender and reciever must know the value of 'n'. Sender knows the value of 'e' only & Reciever knows the value of 'd'.

Hence public key PU = {e, n} & private key PR = {d, n}

<u>Key Generation</u> -

→ Select p, q   p & q both prime, p ≠ q {Private, chosen}

→ Calculate $n = p \times q$ {public, calculated}

→ Calculate $\phi(n) = (p-1)(q-1)$

→ Select Integer e $\gcd(\phi(n), e) = 1$; $1 < e < \phi(n)$
↳ {Public, chosen}

→ Calculate d $d \equiv e^{-1} \pmod{\phi(n)}$
↳ {Private, calculated}   or $\boxed{de = 1 \bmod \phi(n)}$

Scanned by CamScanner

→ Public key $PU = \{e, n\}$

→ Private key $PR = \{d, n\}$

Encryption -

Plain Text $M < n$

Cipher Text $C = M^e \bmod n$

Decryption -

Cipher Text C

Plain Text $M = C^d \bmod n$

Example :- Keys :-

1- Select two prime Numbers, $p = 17$ & $q = 11$

2- Calculate $n = pq = 17 \times 11 = 187$

3- Calculate $\phi(n) = (p-1)(q-1)$
$= 16 \times 10 = 160$

4- Select e such that 'e' is relatively prime to $\phi(n) = 160$ and less than $\phi(n)$, we choose $e = 7$.

5- Determine d such that $de \equiv 1 \bmod 160$ & $d < 160$.

The correct value of $d = 23$ because $23 \times 7 = 161 = 1 \bmod 160$)

Hence - Public key - $PU = \{7, 187\}$

Private key - $PR = \{23, 187\}$

Encryption: Plain Text 88 → [$88^7 \bmod 187 = 11$] (PU) → Cipher Text 11

Decryption: [$11^{23} \bmod 187 = 88$] (PR) → Plain Text 88

Hom Wok. a)- $p = 3$, $q = 11$, $e = 7$, $M = 5$

→ 13, 17 ; 35 $d = 11$,

Salesforce1 Mobile App. ... lop and customize

$P = 17, \; q = 11$

$n = 17 \times 11 = 187$

$\phi = 16 \times 10 = 160$

$1 < e < \phi$ coprime(ϕ)

$\hookrightarrow 160 = 80 \times 2$
$= 40 \times 2 \times 2$
$= 20 \times 2 \times 2 \times 2$
$= 2 \times 10 \times 2 \times 2 \times 2$
$= 2 \times 5 \times 2 \times 2 \times 2 \times 2$

Row	a	b	d	K
1	1	0	160	–
2	0	1	7	22
3	1	-22	6	1
4	-1	23	1	–

d=1 → Stop

$1-(-22 \times 1)$ $\quad 7-(6 \times 1)$
$= 1+22$ $\quad 7-$

$K_2 = \frac{d_1}{d_2}$
$= \frac{160}{7}$
$=$

Choose $e = '7'$ not divide by 160 also

Now calculate 'd'

$d \times e \bmod \phi = 1$

$d \times 7 \bmod 160 = 1$

Using Extended Eucledian theorem.

$ax + by = \gcd(a, b)$

$a = \phi, \; b = e$

$160 \cdot x + 7 \cdot \overset{d}{y} = \gcd(160, 7)$

$160 \cdot (-1) + 7 \times (23) = 1$

$-160 + 161$

$\boxed{= 1}$

$d = 23$

$a_3 = a_1 - (a_2 \times K)$
$= 1 - (0 \times 22)$
$a_3 = 1$

$b_3 = b_1 - (b_2 \times K)$
$= 0 - (1 \times 22)$
$= 0 - 22$

$d_3 = 160 - (7 \times 22)$
$= 160 - 154$
$= 6$

$0 - ($

16

Condition - if $d > \phi$ $\quad d = d \bmod \phi$ ✓

if d is -ve $\quad d = d + \phi$

if $d < \phi$ $= d = d = 23$ ✓

Scanned by CamScanner

Rabin Cryptosystem :- Asymmetric

* Published in January 1979 by Michael O. Rabin.

→ First Asymmetric cryptosystem where recovering the entire P.T from the C.T could be proven to be as hard as factoring.

Key Generation -

Encryption — $\cancel{P} \Rightarrow C = P^2 \bmod n$

Decryption $P = \sqrt{C} \bmod n$

Key Generation -

1- Choose two large distinct primes p & q. One may choose $p \equiv q \equiv$ ($p \bmod 4 = 3$, $q \bmod 4 = 3$)

2- Let $n = p \times q$

3- Then n is the public key.

4- The primes p & q are the private key.

Encryption -

→ For encryption only public key 'n' is used.

→ Let be Z_n, the plaintext space and m be the plain text. Now the cipher text c is determined by

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

→ C is the quadratic remainder of the square of the plaintext, modulo the key-number n.

Decryption -

→ Based on the solution of quadratic congruence.

→ Plain Text is $\sqrt{C} \bmod n$.

→ Four square roots are calculated out of which the correct Plaintext is selected.

→ Using Extended Euclidean algorithm find a & b.

$$a \times p + b \times q = 1$$

→ Compute

- $r = c^{(p+1) \div 4} \bmod p$
- $s = c^{(q+1) \div 4} \bmod q$
- $x = (a \times p \times s + b \times q \times r) \bmod n$
- $y = (a \times p \times s - b \times q \times r) \bmod n$

four square roots are -

$m_1 = x$, $m_2 = -x$, $m_3 = y$, $m_4 = -y$

four possible plain texts so difficult to find if the plain text is numeric.

~~Example~~

Soln - Introducing redundancy in the plain text.

- Pad the msg. such that only one of the four possible ways fit the padding, by replacing the bits.
- String of bits known as preset bits appended to the msg.

Example. Plain Text $= 5_{10}$, in binary $= 101$

by replacing bits, $101101 = 45_{10}$

- Let $p = 11$, $q = 7$ then $n \times q = 77$

public key $= 77$, private key $= 11, 7$

Encryption - cipher text $c = 45^2 \bmod 77 = 23$

Decryption - By Euclidean Algorithm

$$a \times p + b \times q = 1$$

Compute = $a = 2,\ b = -3$

- $x = 23^{(11+1) \div 4} \mod 11 = 1$
- $S = 23^{(7+1) \div 4} \mod 7 = 4$
- $x = (2 \times 11 \times 4 + (-3) \times 7 \times 1) \mod 77 = 67$
- $y = (2 \times 11 \times 4 - (-3) \times 7 \times 1) \mod 77 = 32$

This is the major disadvantage of Rabin.
↳ Decoding produces three false results in addition to the correct one, so that the correct result must be guessed.

Hence the roots are –

$m_1 = 67$

$m_2 = -67$ (We need to avoid (–)) So = $-67 + 77 = 10$

$m_3 = 32$

$m_4 = -32 = 45\ (-32+77)$

Out of the 4 –

$67_{10} = 1000011_2$

$10_{10} = 0001010_2$

$32_{10} = 0100000_2$

$45_{10} = 0101101_2$

1000011 ⇒

Only 45 has replicated bits hence it is the plain text.

Removing Replicating bits = $101_2 = 5_{10}$.

$P = 7,\ q = 11,\ n = 77,\ m = 20$ [64, 13, 20, 57) $\overset{a\ \ b}{(-3, 2)}$

Elgamal Cryptosystem → Asymmetric key.

→ Key Generation –

i)- Select large Prime NO. P

ii)- Select decryption key / Private Key (D)

iii)- Select second part of encryption key or public key (E_1)

iv)- Third part of the encryption key or public key (E_2)

$$\boxed{E_2 = E_1^{D} \mod P}$$

(v) - Public Key = $\{E_1, E_2, P\}$

Private Key = D

Encryption -

i) - Select Random Integer (R)

ii) - $C_1 = E_1^{R} \bmod P$

iii) - $C_2 = (PT \times E_2^{R}) \bmod P$

iv) - $C.T. = (C_1, C_2)$

Decryption -

$$P.T = \left[C_2 \times (C_1^{D})^{-1}\right] \bmod P$$

Example -

Plain Text = 7

Key Generation -

i) - $P = 11$

ii) - $D = 3$

iii) - $E_1 = 2$

iv) - $E_2 = 2^3 \bmod 11$

$= 8 \bmod 11$

$E_2 = 8$

Public key = $(2, 8, 11)$

Private key = 3

Encryption

i) ⇒ $R = 4$

ii) - $C_1 = 2^4 \bmod 11 = 5$

iii) - $C_2 = (7 \times 8^4) \bmod 11$

$C_2 = (28672) \bmod 11$

$C_2 = 6$

Cipher Text = $(5, 6)$

Decryption →

$$P.T = \left[6 \times (5^3)^{-1}\right] \bmod 11$$

$(5^3)^{-1} \bmod 11 = (125)^{-1} \bmod 11$

$(125 \times x) \bmod 11 = 1$

$\Rightarrow 125 \times x = 1$

$x = \frac{1}{125} = 125^{-1}$

$x = 3$

$(375 \bmod 11) = 1$

$$P.T = [6 \times (3)] \bmod 11$$

$= 18 \bmod 11$

$= \boxed{7}$

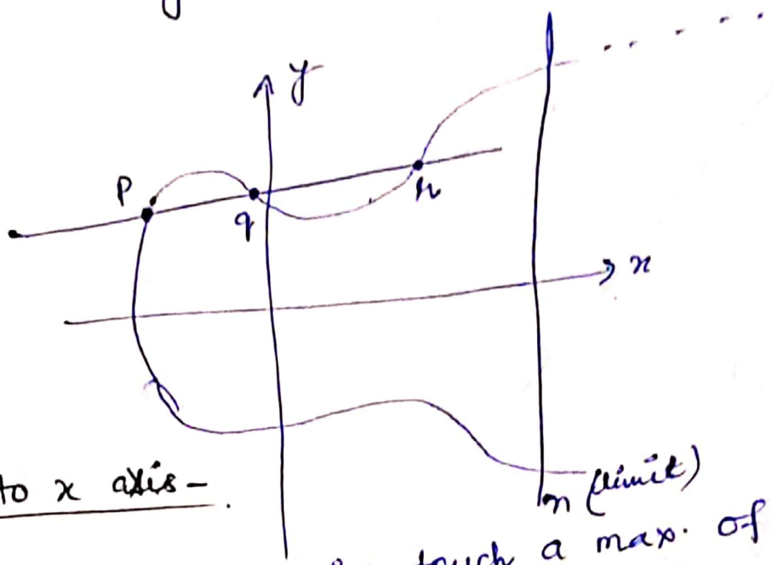
Elliptic Curve Cryptography →

Adv → It provides equal security with smaller key size as compared to non ECC algos.

→ It makes use of elliptic curves.

→ elliptic curves are defined by the mathematical f^n –

$$y^2 = x^3 + ax + b$$



Symmetric to x axis –

If we draw a line, it will touch a max. of 3 points.

ECC –

- Let $E_p(a,b)$ be the elliptic curve.
- Consider eq^n $Q = KP$ where $Q, P \in E_p(a,b)$ and $K < n$.
- It should be easy to find Q given K and P.
- But should be extremely difficult to find K given Q & P.
- Is a one way function → trap door function.
- It is called discrete logarithm problem.

Key Exchange →

→ Global Public elements –

$E_q(a,b)$ = Elliptic curve with parameters a, b & q.

q is a prime or integer of the form 2^m.

G : Point on elliptic curve whose order is large value n

User A key Generation -

Select private key $n_A = n_A < n$

Calculate public key $P_A = P_A = n_A \times G$

User B key Generation -

Select private key $n_B = n_B < n$

Calculate public key $P_B = P_B = n_B \times G$

Secret key by User A - $K = n_A \times P_B$

Secret Key by User B - $K = n_B \times P_A$

ECC - Encryption →

- Let the message be M.
- First encode the message M into a point on the elliptic curve.
- Let this point be P_m.
- Now this point is encrypted.
- For encrypting choose a random positive integer K.

Then $C_m = \{ \underset{①}{KG}, \underset{②}{P_m + K P_B} \}$ G is base point.

Decryption -

→ For decryption, multiply first point ① in the pair with reciever's secret key. i.e $\underline{KG} \times \underline{n_B}$

→ Then subtract it from second point in the pair.

i.e. $\underset{②}{\underline{P_m + K P_B}} - (K G \times n_B)$ $\{\because P_B = n_B \times G\}$

$\Rightarrow P_m + \cancel{K P_B} - \cancel{K P_B}$

$= \boxed{P_m}$ Original point.