Asymmetric Key Cryptography

Symmetric Key Cryptography issues:

- Needs a secure transmission channel to send key. (not possible)
- Key distribution is risky. (Key distribution is the most IMP part here)

Asymmetric Cryptographic (aka Public Key Cryptography)

- Two different keys: Public & Private
- Keys are Mathematically related to each other
- Two types:
  1] Two Keys; one for each user
  2] four Keys; a pair each
- No need to keep public key secure
- No way to authenticate sender
- Not suitable for large message.
- More secure, thus useful to send key for symmetric encryption.

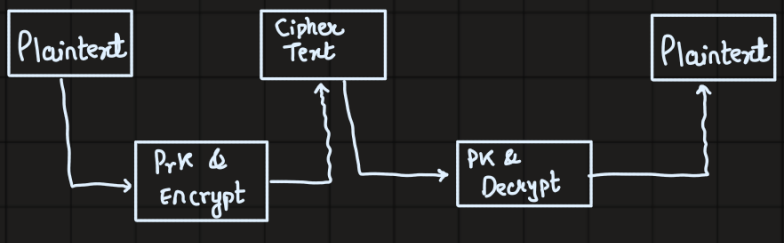
Key
↳ Private - Secret Key, never transmitted from owner

- How is secrecy of message provided, if Public Key is public?

→ Sender encrypts using receiver's public key (PK) and can only be decrypted by receiver's Private Key (PrK)

Authentication

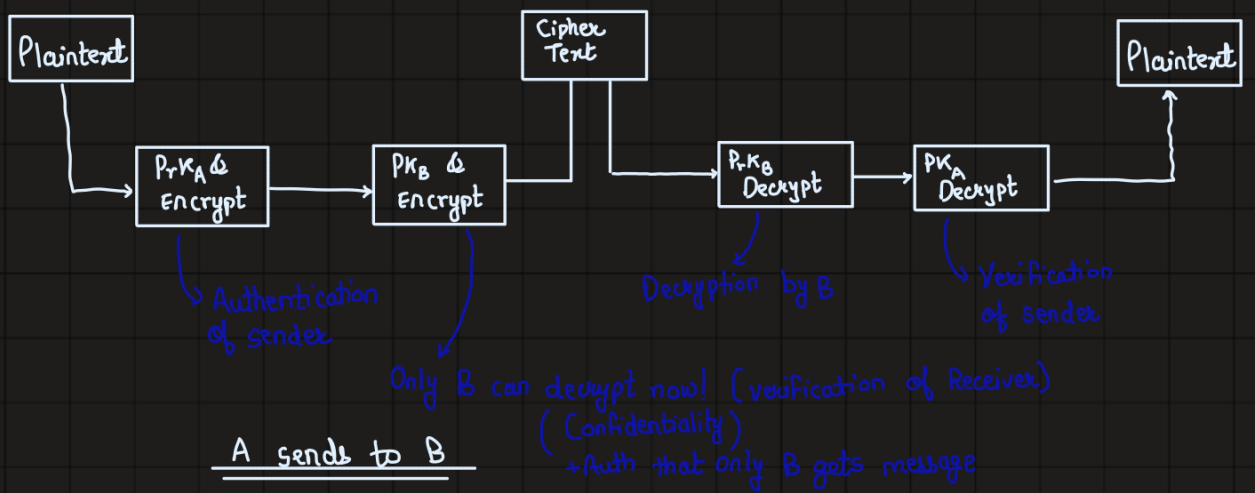
- Uses private key for encryption
- Recipient knows message encrypted by Sender's PrK.
- Attacker may get access to cipher text & decrypt it but to send it he needs PrK of Sender.
- Thus, Auth & origin of message is verifiable [but No Secrecy!]



PK encryption - Authentication

• Vice-Versa for only secrecy, but what if we need both?

PK encryption - Secrecy & Auth; Confidentiality



Only B can decrypt now! (verification of Receiver)
(Confidentiality)
+Auth that only B gets message

A sends to B

• Each User has two Keys
• Two Algos used for Encryption and decryption.
• Encryption Twice & Decryption Twice. Only intended user can decrypt message.

Confidentiality - Auth of both sender and receiver needed!

• Receiver's PK is used to encrypt. Only intended receiver can decrypt (thus confidentiality)

• Authentication & Secrecy & Confidentiality

PR_A & PU_A	Private - Public of A
PR_B & PU_B	-"- of B
M	Plain Text (Msg)
C	Cipher Text

1] A encrypts using Private key of its own: Encrypt/Decrypt(Msg, Key)

$temp = E(M, PR_A)$

2] A encrypts using public key of B (receiver):

$C = E(E(M, PR_A), PU_B)$

3] At receiver, decrypt using B's Private key:

$temp = D(E(E(M, PR_A), PU_B), PR_B) \Rightarrow E(M, PR_A)$

4) Receiver Decrypts using PU_A

$M = D(E(M, PR_A), PU_A)$. ⇒ Message

Asymmetric encryption → Used for Key distribution for symmetric Key.

↳ ∵ It is very slow, thus not suitable for encrypting large data.

RSA Algorithm

- Most Widely used Public Key Encryption Method
- Based on Modular Exponentiation property.
- No Exchange of Key ; Same algo for encryption/decryption.
- Variable Key length. (Common Size 1024)
- Larger Key, decrease Speed but Increase Security.

3 parts:

- Key Generation
- Encryption
- Decryption

Key Generation Steps:

1. Generate two large prime numbers p and q. Such that $p \neq q$.
2. Calculate modulus $n = p*q$ and $m = (p-1)*(q-1)$
   ↳ Since, n will be used as modulus in coming steps.
3. Select Key for encryption 'e', such that 'e' is relatively prime to 'm'; also $1 < e < m$
   - $e = x$, such that $\gcd(x, m) = 1$
4. Calculate Key for decryption:

   $d = e^{-1} \bmod m$

   ↳ Multiplicative inverse of e mod m, and $1 < d < m$

Public Key → $K(e, n)$ is used for encryption.

Private Key → $K(d, n)$ -11- decryption.

- P, q, m should be Kept Private.

Encryption: $C = P^e \bmod n$

Decryption: $P = C^d \bmod n$

Timing Attack on RSA:

- Cryptanalyst observes decryption process, based on time taken, he finds out exec time for each cipher text.
- Uses above info to deduce value for decryption key "d".

Protection?

- large value for 'e'.
- Use padding, delay during encryption.

* Key Distribution & Management:
  - Public Announcement
  - Publicly Available Directory
  - Public Key Authority
  - Public Key Certificates

Public Announcement

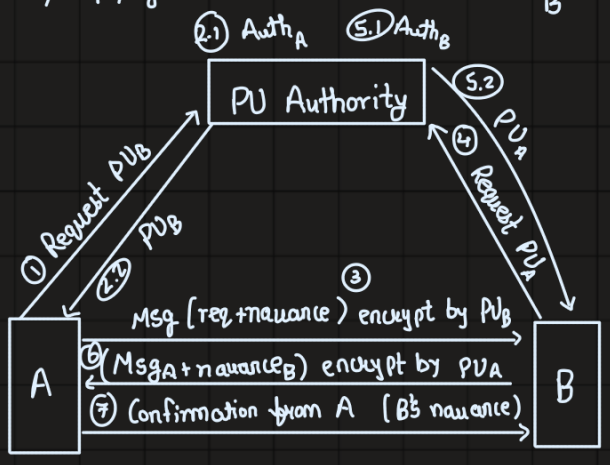
- PU is broadcasted by owner
- Drawback → Anyone can access & forge key to encrypt / decrypt.
- No control on who is accessing the key.

- Publically Available Directory
  - Directory is available to all legitmate users
  - PU of all users is maintained.
  - Directory maintained by trusted party.

PU Authority:

- Provides control over key distribution
- Steps:
  1. A sends request to authority for B's PU.
  2. Authority Authenticates A (PU_A decryption), sends message with B's PU encrypted using PK_{Auth}.
  3. A decrypts with PU_{auth}, A sends message to B requesting communication. This message contains network address of A and random number called nauance.

4. B receives message from A, B requests message Authority for A's PU ... (B decrypt's using PK_B)
5. (Same Step 1 but for B)
6. B Sends a reply to A, with a new nauance, in addition to random number received from A.
7. A confirms by replying back to B with nauance$_B$ encrypted in PU_B



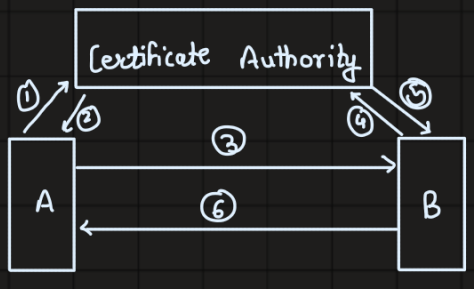
- Con → PU requests are slow & (SPOF ;))

PU Certificates:

- PU's are exchanged using certificates.
- PU Certificates are granted by Centralized Certificate Authority.
- PU Certificate consists of:
  - ~ Public Key
  - ~ Network Address ('name' in book)
  - ~ Time of request.
- PU is encrypted using private Key of Cert Auth.

Steps: (CA - Certificate Authority)

① A requests CA for PU Cert$_A$.
Sends (name + Public Key$_A$) to CA as Msg.
② Responds by Sending Cert encrypted using PK_{CA}
③ A sends cert to B, B decrypts using PU_{CA}.
Verifies authentication for A & stores PU_A.
④ Requests for cert$_B$ to CA. (name + PU_B)
⑤ CA sends cert$_B$ to B E((PU$_B$, time, network addr), PK_{CA})
⑥ B sends cert$_B$ to A.



Is the name & network address same here or different?

Diffie Hellman Key Exchange:

- Key Agreement Protocol.
  - PK is exchangable among users using PU.
  - This algo is for Key Exchange, not for encryption - decryption.
  - Using exchangable keys both participants generate a 'shared secret key', which is used for Encryption / Decryption.

STEPS:

1. User A and B agree on two numbers:
   $n \rightarrow$ A large prime number
   $g \rightarrow$ primitive root of n
2. A selects X_A (PK_A) randomly, such that $X_A < n$. $0 < X_A < n$
3. B selects X_B (PK_B) -11- -11- $X_B < n$. $0 < X_B < n$.
4. A computes Y_A (PU_A) using $Y_A = (g^{X_A}) \bmod n$.
5. B computes Y_B (PU_B) using $Y_B = (g^{X_B}) \bmod n$.
6. Both A & B exchange their PUs over the insecure channel.
7. A calculates k (shared secret key) using, $k = (Y_B^{X_A}) \bmod n$
8. B calculates k using, $k = (Y_A^{X_B}) \bmod n$.
9. Now, A and B communicate with each other using symmetric encryption techniques. k is used as the key.

* Private keys X_A, X_B & shared secret key 'k' should be kept secret.
* Y_A, Y_B, g, n need not be kept secret.
* n and g should be very large to avoid cryptoanalysis via bruteforcing.

> Sophaie Germain Prime Number:
> a prime number p is Sophaie Germain prime if $(2p+1)$ is also prime.

- Man in Middle aka bucket brigade attack.
- Diffie-Hellman does not authenticate users, so it might suffer from man in the middle.
- One of the solutions is to use digital signatures during PU transmission.

Elliptic Curve Cryptography:

- Mathematical operations defined over elliptic curve $y^2 = x^3 + ax + b$ where $4a^3 + 27b^2 \neq 0$.
- Each a, b value gives a different elliptic curve
- PU is a point <u>on</u> the curve, PK is a random number.
- PU is obtained by multiplying PK with generator point G in the curve.
- Major advantage of ECC is the smaller key size.

Point Multiplication:

- A point P on elliptic curve is multiplied by scalar k to obtain Q, another point on the elliptic curve.
- Point Multiplication is achieved by 2 operations:
  1. Point addition (adding 2 points) $L = J + K$
  2. Point doubling (adding J to itself) $L = 2J$

e.g. find $Q = kP$, where $k = 23$

$$kP = 2(2(2(2P) + P) + P) + P$$

$$5$$

$$23$$

- <u>Point Addition:</u>

Let $L = J + K$, where $L = (x_L, y_L)$

then,

$$x_L = s^2 - x_J - x_K$$

$$y_L = -y_j + s(x_j - x_L)$$

$$s = (y_J - y_K) / (x_J - x_K) \quad \ldots \text{slope of line through } K \text{ and } J.$$

Point Doubling:

if $J = 0$, $2J = \bigcirc$, ... $\bigcirc \rightarrow$ the point on elliptical curve at infinity.

Point $J = (x_J, y_J)$, where $y_J \neq 0$,

Let $L = (x_L, y_L)$, where $L = 2J$, Then

$$x_L = s^2 - 2x_J$$

$$y_L = -y_J + s(x_j - x_L)$$

$$s = (3x_j^2 + a) / (2y_j) \quad \ldots s \rightarrow \text{tangent at point } J, \ a \text{ is parameter chosen with elliptic curve.}$$

Finite Fields:

- To make operations on elliptic curve accurate & more efficient, curve cryptography defined over two fields:
  - Prime Field F_p
  - Binary Field F_2^m

Elliptic Curve on Prime Field: F_p

equation : $y^2 \bmod p = x^3 + ax + b \bmod p$, where $(4a^3 + 27b^2) \bmod p \neq 0$

Point Addition

- $J = (x_j, y_j)$
  $K = (x_k, y_k)$
  $L = (x_L, y_L)$ ... $L = J + K$
- $x_L = s^2 - x_j - x_k \bmod p$
- $y_L = -y_j + s(x_j - x_L) \bmod p$
- $s = (y_j - y_k)/(x_J - x_k) \bmod p$

Point Doubling:

$L = 2J$

- $x_L = s^2 - 2x_j \bmod p$
- $y_L = -y_j + s(x_j - x_L) \bmod p$
- $s = (3x_J^2 + a)/2y_j \bmod p$

if $K = -J$, then

$K = (x_J, -y_J \bmod p)$

↑ Point Negation

Elliptic Curve on Binary Field: F_2^m

$y^2 + xy = x^3 + ax^2 + b$, where $b \neq 0$

$J = (x_J, y_J)$
$K = (x_K, y_K)$
$L = (x_L, y_L)$

Point Addition:
$(J + K = K + J)$

$x_L = s^2 + s + x_j + x_K + a$

$y_L = s(x_j + x_L) + x_L + y_j$

$s = (y_j + y_K)/(x_j + x_k)$

Point Doubling

$x_L = s^2 + s + a$

$y_L = x_j^2 + (s + 1) * x_L$

$s = x_J + \frac{y_J}{x_J}$

↓ Tangent at point J.

Point Subtraction:

$-K = +(-K) = (x_K, x_K + y_K)$

Message Digest:

- Integrity of message checked using its hash value or message digest calculated from message.
- Initial version MD2. Used in PU infra and used as part of certificates generated with MD2 & RSA.
- Generated Hash value is called message digest.

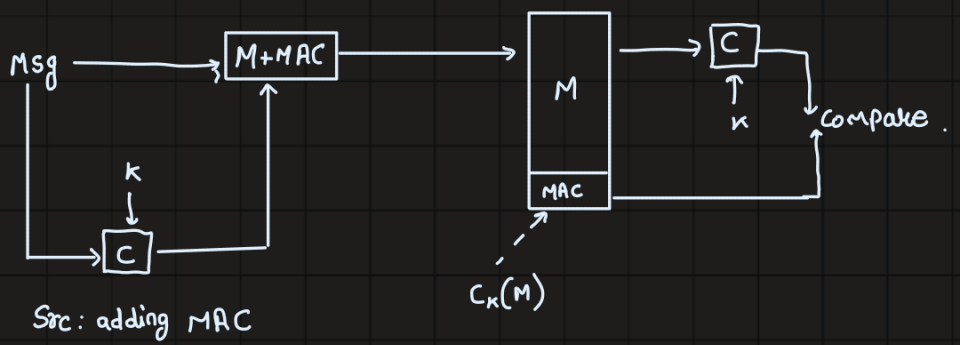
	MD2	MD4	MD5
block size	128 bits	512 bits	512 bits
checksum	16 bytes		
MD generated	128 bits	128 bit	128 bit
Vulnerable	Pre-image attack	//	
Operation size	16 bit	32 bit (non-secure hashing algorithm)	

MD5:

- There are 4 rounds.
  - Each round has 16 operations.
  -

Message Authentication Code: (MAC)

- A generated, small fixed size block appended to message as signature.
- Receiver performs computation on message & matches with MAC. (gives unalteration proof)



Src: adding MAC

- MAC is not digital signature, MAC uses symmetric key.
- MAC is a cryptographic checksum.

$$MAC = C_K(M)$$

K: key, M: Msg

MAC requirements:

1. Knowing Msg & MAC; but not the key, it should be infeasible to find other messages with the same MAC.
2. Uniform distribution of MAC data.
3. Depend equally on all bits of msg.

Hash Functions:

- To create digital signature.
- produces a fingerprint of file/data/message. $h = H(m)$
- Requirements:
  - fixed length output 'h' produce for any message of arbitrary length m.
  - One way property: $h = H(x)$ ... given h it is infeasible to find x.
  - Weak collission resistance: $H(y) = H(x)$; given x we still cannot compute y.
  - Strong collission resistance: $H(y) = H(x)$; infeasible to find any pair (x,y) that meets $H(x) = H(y)$

Solved Problems:

RSA: Message $M = 123$

$p = 11$

$q = 3$

$e = 13$

$n = 11 * 3 = 33$

$m = 10 * 2 = 20$

now, $d = e^{-1} \bmod m$

$\gcd(e, m) = \gcd(13, 20) = 1$

$20 = 13 * 1 + 7$ | $7 = 20 - 13(1)$

$13 = 7 * 1 + 6$ | $6 = 13 - 7(1)$

$7 = 6 * 1 + 1$ | $1 = 7 - 6(1)$

$6 = 1(6)$

$1 = 7 - 6(1)$

$= 20 - 13(1) - 6$

$= 20 - 13(1) - (13 - 7)$

$= 20 - 13 - 13 + 7$

$= 20 - 13(2) + 20 - 13(1)$

$= 20(2) - 13(3)$

$1 = 20(2) + 13(\underline{-3})$

$\hookrightarrow d$

$\therefore d = -3$

We would prefer residue of d, not -ve d,

$\therefore d = -3 \bmod 20 = 17 \bmod 20$

$\therefore e = 13, d = 17$

$C = p^e \bmod n$

$C_1 = 1^{13} \bmod 33 = 1 \bmod 33$

$C_2 = 2^{13} \bmod 33 = 8 \bmod 33$

$C_3 = 3^{13} \bmod 33 = 27 \bmod 33$

$P = C^d \bmod n$

$P_1 = 1^{17} \bmod 33 = 1 \bmod 33$

$P_2 = 8^{17} \bmod 33$

$= (8^3 \bmod 33)^5 + 8^2 \bmod 33$

$= (17 \bmod 33)^5$

$= 25 \bmod 33 * 31 \bmod 33 * 17 \bmod 33 * 25 \bmod 33$

$= 31 \bmod 33 * 31 \bmod 33 * 17 \bmod 33$

$= 4 \bmod 33 * 17 \bmod 33$

$= 2 \bmod 33$

$P_2 = 2$

$P_3 = 27^{17} \bmod 33$

$= (27^2 \bmod 33)^8 + 27 \bmod 33$

$= (3 \bmod 33)^8$

$= (3^4 \bmod 33)^2$

$= (15 \bmod 33)^2$

$= (27 \bmod 33)^2$

$= 3 \bmod 33$

$P_3 = 3$

Q.) Find PK for RSA if parameters given are:

$p = 93$

$q = 47$

$e = 21$

$n = 93 * 47 = 4371$

$m = 92 * 46 = 4232$

$d = e^{-1} \bmod m$

$d = 21^{-1} \bmod 4232$

gcd (21, 4232)

$4232 = 21(201) + 11$

$21 = 11(1) + 10$

$11 = 10(1) + 1$

$10 = 1(10)$

$11 = 4232 - 21(201)$

$10 = 21 - 11$

$1 = 11 - 10$

$1 = 4232 - 21(201) - (21 - 11)$

$= 4232 - 21(202) + 11$

$= 4232 - 21(202) + 4232 - 21(201)$

$= 4232(2) - 21(403)$

$= 4232(2) + 21(-403)$

$\therefore e^{-1} = -403$, residue value will be,

$= -403 \bmod 4232$

$= 3829$

P.K $= (d, n)$

$= (3829, 4371)$

7.2) $p = 5$

$q = 17$

$n = 5 * 17 = 85$

$m = 4 * 16 = 64$

$e = 1 < e < 64$ $\quad \gcd(e, m) = 1$

$\gcd(3, 64) = 1$ (3 → 5, 7 ...)

$\therefore$ let $e = 5$

PU $= (5, 85)$ $\quad (e, n)$

$d = e^{-1} \bmod m$

$d = 5^{-1} \bmod 64$

gcd (64,5)

$64 = 5(12) + 4$

$5 = 4(1) + 1$

$4 = 1(4)$

$4 = 64 - 5(12)$

$1 = 5 - 4$

$1 = 5 - (64 - 5(12))$

$= 5 - 64 + 5(12)$

$= 5(13) - 64$

$\therefore d = 13$

$\therefore PK = (13, 85)$

Encryption of msg = 4

$C = P^e \bmod n$

$= 4^5 \bmod 85$

$= 4^4 \bmod 85 * 4 \bmod 85$

$= 1 \bmod 85 * 4 \bmod 85$

$= 4 \bmod 85$

$= 4$

Decryption of msg

$P = C^d \bmod n$

$= 4^{13} \bmod n$

$= (4^4 \bmod 85)^3 * 4 \bmod 85$

$= (1 \bmod 85)^3 * 4 \bmod 85$

$= 4 \bmod 85$

$= 4$

7.3 $p = 3$

$q = 19$

$e = ?$

$n = 3 * 19 = 57$

$m = 2 * 18 = 36$

possible 'e' values (co-primes between 1 to m(36)) $\Rightarrow \phi(m)$

$\phi(36) = \phi(3 * 3 * 2 * 2)$

$= \phi(3^2) * \phi(2^2)$

$= (3^2 - 3) * (2^2 - 2)$

$= (9 - 3) * (4 - 2)$

$= 6 * 2$

$= 12$

$\phi(p^n) = p^n - p^{n-1}$

e has to be coprime with m, i.e. 36   & $1 < e < 36$

$\gcd(5, 36) = 1$

$\therefore e = 5$

$36 = 5(7) + 1$

$5 = 1(5)$

$1 = 36 - 5(7)$

$\downarrow$

$1 = 36 + 5(-7)$

$d = e^{-1} \bmod m$

$d = -7 \bmod 36$

$d = 29 \bmod 36$

PU = (5, 57)   (e, n)

PK = (29, 57)   (d, n)

Msg = P = 6

Encrypt:

$C = P^e \bmod n$

$= 6^5 \bmod 57$

$= 6^3 \bmod 57 * 6^2 \bmod 57$

$= 45 \bmod 57 * 36 \bmod 57$

$= 1620 \bmod 57$

$= 24 \bmod 57$

Decrypt:

$P = C^d \bmod n$

$= 24^{29} \bmod 57$

$= (24^5 \bmod 57)^5 * 24^4 \bmod 57$

$= (9 \bmod 57)^5 * 36 \bmod 57$

$= 54 \bmod 57 * 36 \bmod 57$

$= 6 \bmod 57$

$24^4 \rightarrow 36$

$24^5 \rightarrow 9$

$24^6 \rightarrow 45$

7.4) $C = 4$

$e = 89$

$p = 11$

$q = 47$

$n = 11 * 47 = 517$

$m = 10 * 46 = 460$

$d = e^{-1} \bmod m$

$= 89^{-1} \bmod 460$

gcd (460, 89)

$460 = 89(5) + 15$

$89 = 15(5) + 14$

$15 = 14(1) + 1$

$14 = 1(14)$

$15 = 460 - 89(5)$

$14 = 89 - 15(5)$

$1 = 15 - 14$

$1 = 15 - 14$

$1 = 460 - 89(5) - (89 - 15(5))$

$1 = 460 - 89(6) + (460 - 89(5))(5)$

$1 = 460 - 89(6) + 460(5) - 89(25)$

$= 460(6) - 89(31)$

$= 460(6) + 89(-31)$

$\therefore d = e^{-1} \bmod m$

$d = -31 \bmod 460$

$d = 429$

decryption:

$p = c^d \bmod n$

$= 4^{429} \bmod 517$

$= (4^3 \bmod 517)^{47} * 4^6 \bmod 517$

$= (25 \bmod 517)^{47}$

$= ((25^5 \bmod 517)^9 * 25^2 \bmod 517) * 4^6 \bmod 517$

$= ((12 \bmod 517)^9 * 108 \bmod 517) \ldots$

$= ((12^3 \bmod 517)^3 * 108 \bmod 517) \ldots$

$4^6 \rightarrow 477$

$4^7 \rightarrow 357$

$4^8 \rightarrow 394$

$4^9 \rightarrow 25$

$4^{12} \rightarrow 49$

$4^{15} \rightarrow 34$

$25^2 \rightarrow 108$

$25^3 \rightarrow 115$

$25^4 \rightarrow 290$

$25^5 \rightarrow 12$

$= ((56 \bmod 517)^3 \cdot \cdots$

$= 353 \bmod 517 * 108$

$= 383 * 477 \bmod 517$

$= 190$

$12^3 \rightarrow 177$

$12^4 \rightarrow 56$

* Diffie-Hellman key Exchange

- $n = 19$

  $g = 7$

  $X_A = 8$ $\quad \therefore Y_A = g^{X_A} \bmod n = 7^8 \bmod 19 = 11 \bmod 19$

  $X_B = 10$ $\quad \therefore Y_B = 7^{10} \bmod n = 7 \bmod 19$

$\therefore Y_A = 11$

$Y_B = 7$

k at A's side $\Rightarrow$ $k = (Y_B^{X_A}) \bmod n$

$= (7^8) \bmod 19$

$= 11$

k at B's side $\Rightarrow$ $k = (Y_A^{X_B}) \bmod n$

$= (11^{10}) \bmod 19$

$= (11^2)^5 \bmod 19$

$= (7)^5 \bmod 19$

$= 11$

$\therefore k = 11$

Elliptic Curve Equation

1] $y^2 = x^3 + 10x + 5$ define a group of F_{17}?

$a = 10$

$b = 5$

$4a^3 + 27b^2 = 4 * 10^3 + 27 * 5^2 \pmod{17}$

$= 4000 + 675 \mod 17$

$= 4675 \mod 17$

$= 0$

but for elliptic curve to exist, $4a^3 + 27b^2 \neq 0$!

∴ Does not define a group.

2] $P(2,0)$

$Q(6,3)$ lie on elliptic curve, $y^2 = x^3 + x + 7$ over F_{17}

∴ $a = 1, b = 7$

$4a^3 + 27b^2 = (4 + 27 * 49) \mod 17$

$= 1 \mod 17$ ∴ defines a group.

for P, $x = 2, y = 0$

$y^2 = x^3 + x + 7 \mod 17$

$0^2 = 2^3 + 2 + 7 \mod 17$

$= 8 + 2 + 7 \mod 17$

$= 17 \mod 17$

$0 = 0$ ∴ Point is valid.

for Q, $x = 6, y = 3$

$y^2 = x^3 + x + 7 \mod 17$

$3^2 = 6^3 + 6 + 7$

$9 = 229 \mod 17$

$9 = 8 \mod 17$

∴ Q does not lie on elliptic curve

(3) What are Negatives of Points over field F_{17}

1] $P(5,8)$

$-P = (5, -8 \bmod 17)$

$-P = (5, 9)$

2] $Q(3,0)$

$-Q(3,0 \bmod 17)$

$-Q(3,0)$

3] $R(0,6)$

$-R = (0, -6 \bmod 17)$

$-R = (0, 11)$

(4) $y^2 = x^3 + x + 7$ over F_{17}

$a = 1$

$b = 7$

$P(2,0) \quad \leftrightarrow J$

$Q(1,3) \quad \leftrightarrow K$

$P + Q = ?$

$J \quad K$

$S = (3-0)/(1-2) = 3/-1 = -3 \bmod 17 = 14 \bmod 17$

$x_L = S^2 - x_Q - x_P$

$= 14^2 - 1 - 2 \bmod 17$

$= 193 \bmod 17$

$= 6 \bmod 17$

$y_L = -y_J + S(x_J - x_L) \bmod 17$

$= -0 + 14(2 - 6) \bmod 17$

$= -0 + 14(-4) \bmod 17$

$= 12 \bmod 17$

$\therefore L = P + Q$

$= (6, 12)$

(5) $P = (1,3)$ ↔ 5

$2P = ?$

$y^2 = x^3 + x + 7$ over $F = 17$

$$s = (3x_j^2 + a) / 2y_j$$
$$= 3(1)^2 + 1 / 2(3)$$
$$= 4/6 \mod 17$$
$$= 4 * 6^{-1} \mod 17$$
$$= 4 * 3 \mod 17 \quad \because 6^{-1} = 3 \mod 17$$
$$= 12 \mod 17$$

6^{-1} of 17 = gcd(17,6)

$17 = 6(2) + 5$ ⟹ $5 = 17 - 6(2)$

$6 = 5(1) + 1$ ⟹ $1 = 6 - 5$

$5 = 1(5)$

$1 = 6 - 5$

$1 = 6 - (17 - 6(2))$

$= 6(3) - 17$

$= 6(3) + 17(-1)$

$$x_l = s^2 - 2x_j \mod 17$$
$$= 12^2 - 2(1)$$
$$= 142 \mod 17$$
$$= 6 \mod 17$$

$$y_l = -y_j + s(x_j - x_l) \mod 17$$
$$= -3 + 12(1 - 6) \mod 17$$
$$= -3 + 12(-5) \mod 17$$
$$= -63 \mod 17$$
$$= 5 \mod 17$$

$\therefore 2P = (6,5)$

Elliptic Curve Cryptography Over F_2^m ... binary field

1. Does the elliptic curve equation $y^2 + xy = x^3 + g^5 x^2 + g^6$ over F_2^3 define a group?

→ Yes,

$\because a = g^5$, $b = g^6$ and $\therefore b \neq 0$.

2. Do the points P, Q lie on elliptic curve given by equation $y^2+xy = g^2x^2 + g^6+x^3$ over f_2^3 ?

→ Given:

$P(g^3, g^6)$

$Q(g^5, g^2)$

Given Values:

$g_1 / g = 010$

$g_2 = 100$

$g_3 = 011$

$g_4 = 110$

$g_5 = 111$

$g_6 = 101$

$g_7 = 001$

for Point P:

- $x_P = g^3$, $y_P = g^6$

$y^2+xy = g^2x^2+g^6+x^3$

$(g^6)^2 + g^3 \cdot g^6 = (g^3)^3 + g^2(g^3)^2 + g^6$

$g^{12} + g^9 = g^9 + g^9 + g^6$

$a^{b^c} = a^{b*c}$; $a^b \cdot a^b = a^{b+b}$

mod w.r.t 7,

$g^5 + g^2 = g^2 + g^1 + g^6$

$111 \oplus 100 = 100 \oplus 010 \oplus 101$

$011 = 011$.

$Q = (g^5, g^2)$

$x = g^5$, $y = g^2$

$y^2+xy = x^3 + g^2x^2 + g^6$

$= (g^2)^2 + g^5 * g^2 = (g^5)^3 + g^2(g^5)^2 + g^6$

(g^{5+2})

$g^4 + g^7 = g^{15} + g^{2+10} + g^6$

mod w.r.t g^7

$g^4 + g = g^1 + g^5 + g^6$

$110 \oplus 010 = 010 \oplus 111 \oplus 101$

$100 = 001$

$\therefore$ Q does not lie on EC

$$\begin{array}{r} 011 \\ 111 \\ 101 \\ \hline 001 \end{array}$$

Q.3