Introduction – Shubham Parmar xyz

Today we are going to explain major topic of ISS; i.e., Symmetric and Asymmetric cryptography, Diffie Hellman, RSA and Euler's theorem.

Types to cryptography(symmetric,asymeteric)

Symmetric Key

- In symmetric key cryptography, we have a sender and a receiver who want to keep their messages secret from prying eyes.
- The key player here is the secret key they both share – it's like a secret language or a special code only they know.
- Unlike public key cryptography, where we have a pair of keys (public and private), symmetric key cryptography uses the same key for both encryption and decryption.
- These algorithm are crucial for securing sensitive information like passwords, financial transactions, and private communications between parties.

Example

1) Diffie-Hellman key exchange

Slide 01 – Here, we can see there are common colors which is Yellow. Alice's Common color is YELLOW and Bob's common color is YELLOW.
Below the common colors we can see secret color of Alice (RED) and Bob (BLUE).
Both the parties we merge their own individual colors and the output color will be kept as secret colors.

Slide 02 – After merging their own common color and their individual secret colors. The secret colors can be shared with each other for further process. Both Alice and Bob share the secret colors! The mixture of colors is not possible to separate again so it's an irreversible step.

Slide 03 – After exchanging the common secret colors. Alice and Bob will add their own individual secret color with the exchanged common secret colors. The final output will be the same identical color. That means with this whole process in the end you will get the same output with the process of merging different colors.

2) Sum

A large prime number, typically denoted as p.

- A primitive root modulo p, often denoted as g.

After that-

Each of them generates their own private key:

- Alice chooses a secret number a and calculates A = g^a mod p.

- Bob chooses a secret number b and calculates B = g^b mod p.

Exchange of public keys

They exchange their public key

Alice and Bob exchange their calculated values:

- Alice sends A to Bob.

- Bob sends B to Alice.

Shared Secret Calculation:

- Alice computes the shared secret s using B and her private key a: s = B^a mod p.

- Bob computes the shared secret s using A and his private key b: s = A^b mod p.

After the calculation both A,B have the shared secret key S which they can use for communication.

Asymmetric Key

Asymmetric key cryptography involves a pair of keys: a public key and a private key. The public key, like a lockbox in a public square, is used to encrypt messages that only the holder of the corresponding private key can decrypt. This system allows secure communication without the need for both parties to share a secret key beforehand. The private key is kept secret and is used for decrypting messages sent to the owner of the key. Additionally, the private key can be used to create digital signatures, providing a way to verify the authenticity of messages and documents.

Example: RSA

Before Starting with RSA we will understand GCD

Explaining GCD using Euclidean Algorithm

A	B	R
33	12	9
12	9	3
9	3	0
3	0	

The **biggest number** should be place in **column A** and **smallest number** should be place in **column B**

- Initially, place 33 in column A and 12 in column B.
- Divide 33 by 12, resulting in a remainder of 9, which is placed in column R.
- After division, shift the numbers to the left: the value in column B shifts to column A, and the value in column R shifts to column B.
- Repeat the division process with the values in columns A and B, placing the remainder in column R.
- Continue this process until column B reaches 0.
- Once column B becomes 0, The value in column A will represent the GCD of 12 and 33.
- If the number is **prime** then the gcd will be always **1** “**GCD(11,3)=1”.**

Euler Totient Function

There are three types of Method:

Φ(n)	**Criteria of ‘n’**	**Formula**
	‘n’ is prime	**Φ(n) = (n-1)**
	n=p x q **‘p’ and ‘q’ are primes**	**Φ(n)=(p-1) x (q-1)**
	n = a x b **Either ‘a’ or ‘b’ is composite.** **Both ‘a’ and ‘b’ are composite.**	**Φ(n)=n x (1-1/p1)*(1-1/p2)…** **where p1,p2… are distant primes**

Euler Totient Function is denoted as **Φ(n)**

Definition: It is number of positive integers less than ‘n’ that are relatively prime to n

There are three types of Criteria of ‘n’

1) First Criteria says if **‘n’ is prime** the formula will be **Φ(n) = (n-1)**

Example:

n=5

n is a prime number

Φ(n)=(n-1)

Φ(5)=(5-1)

Φ(5)=4

2) Second Criteria says If the **“n is the product of two prime numbers”** like **“p” and “q”** so the formula will be **Φ(n)=(p-1) x (q-1)**

Example:

n=5

n is a product of two prime numbers **5 and 7**

Let us assign **p=5 and q=7**

Φ(n)=(p-1)(q-1)

Φ(35)=(5-1)(7-1)

Φ(35)=4 x 6

Φ(35)=24

There are 24 numbers that are lesser than 35 and relatively prime to 35

3) Third Criteria says If the **"n =a x b"** means **"Either a or b is composite or Both a and b are composite"** the formula will be **Φ(n)=n x (1-1/p1)*(1-1/p2)… where p1,p2… are distant primes**

Example:

n=1000

factorization of **1000** = **2^3 x 5^3**

Φ(n)=n x (1-1/p1) (1-1/p2)…

Φ(1000)=1000 x (1-1/2) (1-1/5)

Φ(1000)=1000 x (1/2) (4/5)

Φ(1000)= 400

Euler's Theorem

For every positive integer 'a' & 'n', which are said to be relatively prime, then **a^Φn =1 mod n**

Euler's theorem a=3 and n=10

a^Φn =1 (mod n)

3^Φ10 = 1 (mod 10)

Φ10 = 4 ….By Applying the Euler Totient Function

3^4 = 1 (mod 10)

81 = 1 (mod 10)

Euler Theorem holds true for a=3 and n=10

It provides the mathematical foundation for the RSA algorithm to ensure the security and effectiveness of the encryption scheme. By applying Euler's theorem, RSA ensures that certain mathematical relationships hold true, allowing for secure encryption and decryption of messages using public and private keys.

RSA

Intro:

RSA (Rivest-Shamir-Adleman) is a widely used public-key cryptosystem that is named after its inventors: Ron Rivest, Adi Shamir, and Leonard Adleman. It was introduced in 1977 and remains an important algorithm in the field of cryptography. RSA is widely used for secure data transmission and digital signatures.

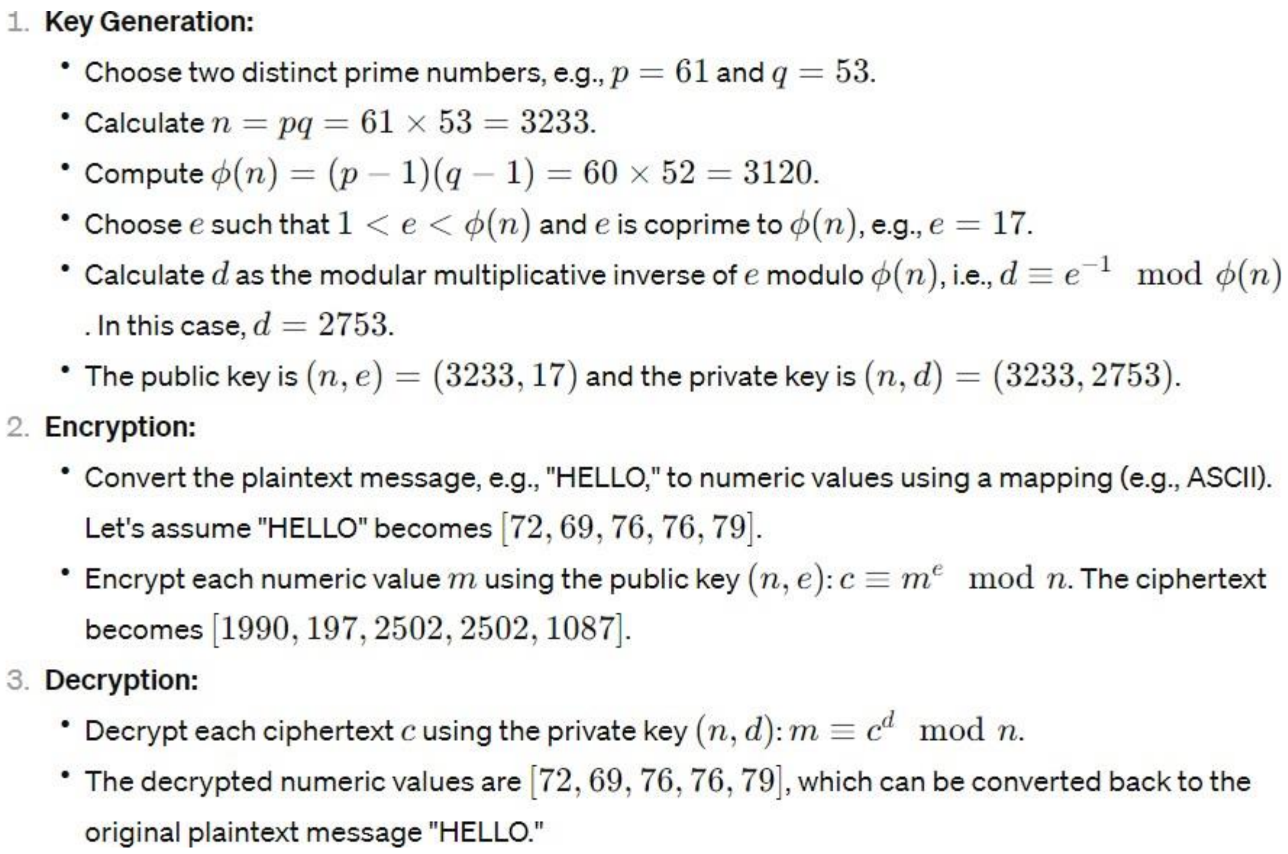
The RSA algorithm involves the use of a pair of keys: a public key and a private key. The public key is used for encryption, and the private key is used for decryption. The keys are mathematically related, but it is computationally infeasible to deduce the private key from the public key.

The security of RSA is based on the difficulty of factoring the product of two large prime numbers. The algorithm relies on the mathematical properties of large prime numbers and their difficulty to factorize, making it a one-way function that is easy to perform in one direction(encryption) but computationally hard in the other (decryption without the private key).

Working

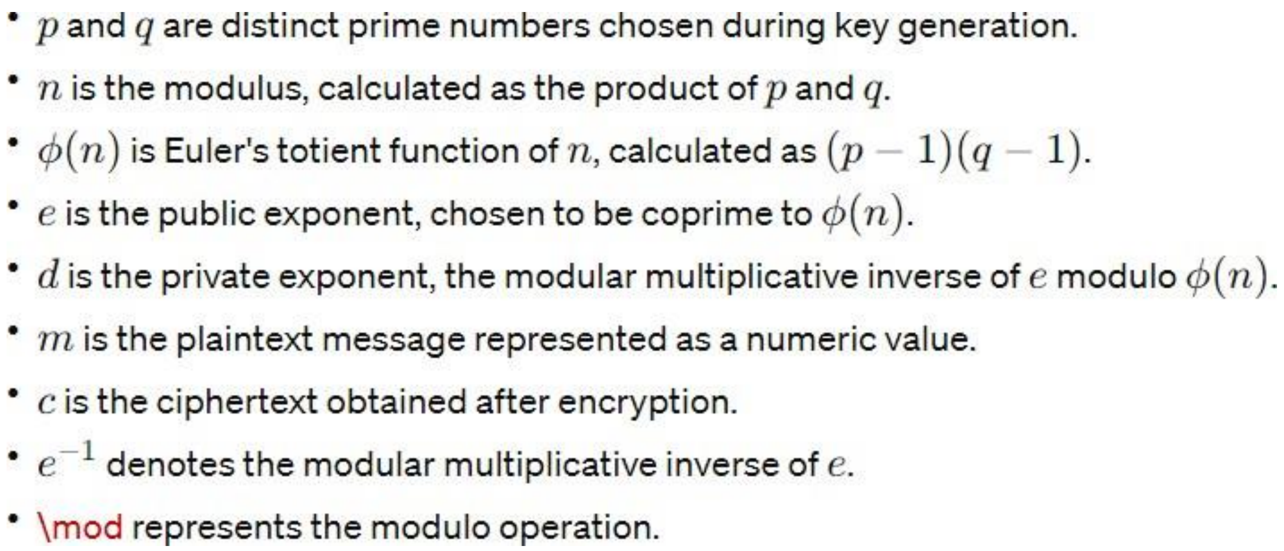
1. Tanay generates a pair of keys - a public key and a private key.
2. He keeps the private key secret and shares the public key with Harshil.
3. Tanay wants to send the message "CONFIDENTIAL" to Harshil.
4. He looks up Harshil's public key (which includes a modulus n and an exponent e).
5. He converts the message "CONFIDENTIAL" to a numerical representation.
6. For each number, he raises it to the power of e and takes the modulus n. These values become the encrypted message.
7. Tanay sends the encrypted message to Harshil.
8. Harshil receives the encrypted message.
9. He uses his private key (including modulus n and a different exponent d) to decrypt the message.
10. For each encrypted number, he raises it to the power of d and takes the modulus n. This retrieves the original numerical representation.
11. Harshil converts the numerical representation back to text, revealing the original message "CONFIDENTIAL."

Example:



1. **Key Generation:**
   - Choose two distinct prime numbers, e.g., $p = 61$ and $q = 53$.
   - Calculate $n = pq = 61 \times 53 = 3233$.
   - Compute $\phi(n) = (p-1)(q-1) = 60 \times 52 = 3120$.
   - Choose e such that $1 < e < \phi(n)$ and e is coprime to $\phi(n)$, e.g., $e = 17$.
   - Calculate d as the modular multiplicative inverse of e modulo $\phi(n)$, i.e., $d \equiv e^{-1} \mod \phi(n)$. In this case, $d = 2753$.
   - The public key is $(n, e) = (3233, 17)$ and the private key is $(n, d) = (3233, 2753)$.
2. **Encryption:**
   - Convert the plaintext message, e.g., "HELLO," to numeric values using a mapping (e.g., ASCII). Let's assume "HELLO" becomes $[72, 69, 76, 76, 79]$.
   - Encrypt each numeric value m using the public key (n, e): $c \equiv m^e \mod n$. The ciphertext becomes $[1990, 197, 2502, 2502, 1087]$.
3. **Decryption:**
   - Decrypt each ciphertext c using the private key (n, d): $m \equiv c^d \mod n$.
   - The decrypted numeric values are $[72, 69, 76, 76, 79]$, which can be converted back to the original plaintext message "HELLO."

Formula:



- p and q are distinct prime numbers chosen during key generation.
- n is the modulus, calculated as the product of p and q.
- $\phi(n)$ is Euler's totient function of n, calculated as $(p-1)(q-1)$.
- e is the public exponent, chosen to be coprime to $\phi(n)$.
- d is the private exponent, the modular multiplicative inverse of e modulo $\phi(n)$.
- m is the plaintext message represented as a numeric value.
- c is the ciphertext obtained after encryption.
- e^{-1} denotes the modular multiplicative inverse of e.
- \mod represents the modulo operation.