Caesar Cipher Algorithm – Brief Overview

Definition:
The **Caesar Cipher** is one of the simplest and oldest encryption techniques. It is a type of **substitution cipher**, where each letter in the plaintext is replaced by another letter obtained by shifting the alphabet by a fixed number of positions.

Algorithm Steps

1. Choose a **shift key** (e.g., 3).
2. For each letter in the plaintext:
    - Replace it with the letter that is **key positions ahead** in the alphabet.
    - Wrap around to the beginning if needed (after 'Z' comes 'A').
3. Non-alphabetic characters usually remain unchanged.

Example

Plaintext: `HELLO`
Shift (key): `3`

Plain	→	Cipher
H	→	K
E	→	H
L	→	O
L	→	O
O	→	R

Ciphertext: `KHOOR`

Strengths

- **Simple and easy to implement** – great for learning basic cryptography.
- **Fast encryption and decryption** – minimal computational effort.

Weaknesses

- **Easily broken** by **brute force** (only 25 possible shifts).
- **Letter frequency analysis** can quickly reveal the key.
- **Not secure** for any real-world application.

Summary

Aspect	Description
Type	Substitution Cipher
Key	Integer (shift value)
Security Level	Very Low
Use Case	Educational or toy encryption

In modern cryptography, Caesar Cipher is used only for teaching and demonstration purposes, not for securing real data.

RSA Algorithm – Brief Overview

Definition:
RSA (Rivest–Shamir–Adleman) is a widely used **asymmetric encryption algorithm**, meaning it uses **two keys** — a **public key** for encryption and a **private key** for decryption.
It is based on the mathematical difficulty of **factoring large prime numbers**.

Algorithm Steps

1. **Key Generation:**
    - Choose two large prime numbers: `p` and `q`.
    - Compute `n = p × q`.
    - Compute Euler's totient: `φ(n) = (p − 1)(q − 1)`.
    - Choose an integer `e` such that `1 < e < φ(n)` and `e` is **coprime** to `φ(n)`.
    - Compute `d` such that `(d × e) mod φ(n) = 1`.
        - `d` is the **private key exponent**.
    - The **public key** is `(e, n)` and the **private key** is `(d, n)`.
2. **Encryption:**
    - Convert plaintext message `M` into a number `< n`.
    - Compute ciphertext:
      `C = M^e mod n`
3. **Decryption:**
    - Compute plaintext:
      `M = C^d mod n`

P,q –> 2048 bits

Public modulus → n

Think of n like a lock made from two secret metals p and q.
 You can hand someone the lock (n), but without knowing the metals used (p and q), they can’t craft the key (d) to open it.

So while both keys are mathematically linked, **the private key cannot realistically be derived from the public key** – that’s the brilliance of RSA.

—--

Row Transposition Cipher

Key: 4 3 1 2 5 6 7

Key: zebra

Limitations of Row Transposition Cipher

1. **No letter substitution** → letter frequency remains the same, making it vulnerable to frequency analysis if multiple messages are available.

2. **Key reuse weakness** → if the same key is reused, patterns can appear.

3. **Simple permutation** → if message length or key is short, ciphertext can be guessed.

4. **Not secure for modern use** → easily broken by computer algorithms or statistical analysis.