

#### 1. XOR Each Character With 0 (XOR – Exclusive OR)

- XOR is a bitwise operator.
- XOR with 0 gives the same value.
- Each character is converted to ASCII.
- ASCII is XORED with 0
- No change occurs in bits.
- Output string matches input.
- Shows basic bitwise operations.
- Operation is very fast.
- Works character by character.
- Used for demonstration purpose only.

#### 2. AND / OR / XOR Each Character With 127 (AND – Logical AND, OR – Logical OR)

- 127 in binary is 01111111.
- AND 127 clears the highest bit.
- OR 127 sets most bits to 1.
- XOR 127 flips all seven bits.
- Each ASCII code is processed.
- Shows effect of bit masking.
- Useful in low-level operations.
- Output differs for each operator.
- Operation is reversible using XOR.
- Good for teaching bitwise logic.

#### 3. Caesar Cipher (CC – Caesar Cipher)

- It is a substitution cipher.
- Invented by Julius Caesar.
- Each letter is shifted by a fixed key.
- Uses modulo 26 arithmetic.
- Same key is used for both enc/dec.
- Very easy to implement.
- Works only on alphabets.
- Does not protect against attacks.
- Breakable by frequency analysis.
- Used as a basic cryptography example.

#### 4. Substitution Cipher (SC – Substitution Cipher)

- Each letter is replaced by another letter.
- A mapping table (key) is required.
- Same substitution is used for whole text.
- It is mono-alphabetic.
- Very simple encryption technique.
- Frequency patterns remain visible.
- Easily breakable by frequency attack.
- Uses direct letter mapping.
- Decryption uses reverse mapping.
- Useful for understanding classical cryptography

## 5. Hill Cipher (HC – Hill Cipher)

- It is a polygraphic cipher.
- Uses matrix multiplication for encryption.
- Key is a square matrix ( $2 \times 2$  or  $3 \times 3$ )
- Plaintext is converted to numbers ( $A=0$ ).
- Encryption uses  $C = KP \text{ mod } 26$ .
- Decryption uses inverse of key matrix.
- Requires key matrix to be invertible.
- More secure than simple substitution ciphers.
- Operates on blocks of letters
- Introduces linear algebra in cryptography.

## 6. DES Algorithm (DES – Data Encryption Standard)

- DES is a symmetric key algorithm.
- Block size = 64 bits.
- Key size = 56 bits.
- Uses 16 Feistel rounds.
- Applies initial permutation.
- Uses expansion, substitution, permutation operations.
- S-boxes provide confusion.
- P-boxes provide diffusion.
- Final permutation produces ciphertext.
- Considered outdated due to small key size.

## 7. Blowfish Algorithm (BF – Blowfish Algorithm)

- Fast symmetric encryption algorithm.
- Block size = 64 bits
- Key size is variable (32–448 bits).
- Uses 16 rounds
- Has a P-array of 18 subkeys.
- Contains four S-boxes of 256 entries.
- Uses Feistel network structure
- Resistant to many attacks.
- Good for software implementation.
- Designed as a free alternative to DES.

## 8. Rijndael / AES (AES – Advanced Encryption Standard, RJA – Rijndael Algorithm)

- AES is a symmetric block cipher.
- Block size = 128 bits.
- Key sizes: 128/192/256 bits.
- Uses SubBytes transformation.
- Uses ShiftRows transformation.
- MixColumns used for diffusion.
- AddRoundKey XORs with round key.
- Number of rounds: 10/12/14.
- Very strong and efficient.
- Current global encryption standard.

#### 9. RC4 Algorithm (RC4 – Rivest Cipher 4)

- RC4 is a stream cipher.
- Uses variable-length key (40–2048 bits).
- Works byte by byte.
- Has KSA (Key Scheduling Algorithm).
- Has PRGA (Pseudo-Random Generation).
- Produces a keystream.
- Encryption is XOR with keystream.
- Very fast in software.
- Used in older protocols like SSL/WEP.
- Now considered insecure.

#### 10. RSA Algorithm (RSA – Rivest–Shamir–Adleman)

- RSA is an asymmetric algorithm.
- Uses two keys: public and private.
- Based on prime numbers.
- Uses modulus  $n = p \times q$ .
- Public key =  $(e, n)$ .
- Private key =  $(d, n)$ .
- Encryption uses  $C = P^e \text{ mod } n$ .
- Decryption uses  $P = C^d \text{ mod } n$ .
- Security depends on factoring difficulty.
- Used for secure key exchange & signatures.

#### 11. Diffie–Hellman Key Exchange (DH – Diffie–Hellman)

- Used for secure key sharing.
- Uses a prime number  $p$ .
- Uses a generator  $g$ .
- A chooses secret  $a$ .
- B chooses secret  $b$ .
- A sends  $A = g^a \text{ mod } p$ .
- B sends  $B = g^b \text{ mod } p$ .
- Shared key =  $g^{ab} \text{ mod } p$ .
- Key is never transmitted directly.
- Provides secure mutual key generation.

#### 12. SHA-1 (SHA – Secure Hash Algorithm)

- SHA-1 is a hashing algorithm.
- Produces 160-bit hash.
- Input is padded to 512-bit blocks.
- Uses 80 rounds.
- Uses five registers (A–E).
- Uses four logical functions.
- Provides one-way hashing.
- Used in old digital signatures.
- Collision attacks discovered.
- Now replaced by SHA-256.

### 13. MD5 (MD5 – Message Digest 5)

- MD5 is a hashing algorithm.
- Produces a 128-bit hash
- Message padded to 512-bit blocks.
- Uses four 32-bit registers (A,B,C,D).
- Performs 64 rounds
- Uses four non-linear functions.
- Very fast in software.
- Not reversible.
- Collisions found (insecure).
- Used only for checksum purposes now.