5.Cryptographic Algorithm

1. Explain following terms:

i) Groups:- A Group is a fundamental algebraic structure consisting of a set G together with a binary operation (*)that satisfies the following four axioms:

- **Closure:** For all $a, b \in G$, the result of $a * b \in G$.
- **Associativity:** For all $a, b, c \in G$, $(a * b) * c = a * (b * c)$.
- **Identity Element:** There exists an element $e \in G$ such that for all $a \in G$, $a * e = e * a = a$.
- **Inverse Element:** For each $a \in G$, there exists an element $a^{-1} \in G$ such that $a * a^{-1} = a^{-1} * a = e$.

If the group also satisfies the commutative property: $a * b = b * a$ for all $a, b \in G$, then it is called an Abelian Group.

Example: The set of integers $\mathbb{Z}$ under addition forms an Abelian group.

ii) Rings:- A Ring is an algebraic structure $(R, +, \cdot)$ with two binary operations: addition and multiplication, satisfying:

- $(R, +)$ is an Abelian group.
- $(R, \cdot)$ is a semigroup (i.e., multiplication is associative).
- Distributive Laws hold:
  - $a \cdot (b + c) = a \cdot b + a \cdot c$
  - $(a + b) \cdot c = a \cdot c + b \cdot c$

Additional Notes:

- A commutative ring satisfies $a \cdot b = b \cdot a$ for all $a, b \in R$.

- A ring may or may not have a multiplicative identity (often denoted as 1).

Example: The set of integers $\mathbb{Z}$ with usual addition and multiplication is a commutative ring with identity.

iii) Prime Numbers:-

- A Prime Number is a natural number $p > 1$ that has exactly two positive divisors: 1 and itself.
- Formal Definition: A number p is prime if whenever $p = a \cdot b$, then either $a = 1$ or $b = 1$.

Important Properties:

- The number 1 is not a prime.

- The Fundamental Theorem of Arithmetic states: Every integer greater than 1 is either a prime or can be uniquely factored as a product of prime numbers (up to order).

Examples: 2, 3, 5, 7, 11, 13, …

2. Explain following terms:-

i) PKIX Model:-

- The PKIX (Public Key Infrastructure X.509) model is designed to extend the X.509 standard for use on the Internet, ensuring interoperability between different implementations.
-
- PKIX defines a document that outlines five areas of its architectural model: **profiles of internet users, certificate policies, certificate practice statements, time-stamping, and data certification/validation services**.
-
- The PKIX model supports checking certificate status in two modes: online using OCSP (Online Certificate Status Protocol) and offline using CRLs (Certificate Revocation Lists).

Key Components:

- **Certification Authority (CA)**: A trusted entity that issues digital certificates.
- **Registration Authority (RA)**: Verifies user identities before certificates are issued.
- **Certificate Repository**: Stores digital certificates and Certificate Revocation Lists (CRLs).
- **End Entities**: Users or devices that use PKI services.

Function: The PKIX model ensures secure communication over a network using public key cryptography. It binds public keys with user identities through digital certificates verified and signed by a CA.

ii) Digital Signature:-

- A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software, or digital document.
-
- A signature confirms that the information originated from the signer and has not been altered.

Key Generation Algorithms:
Digital signatures use a pair of keys — a private key (kept secret by the sender) and a public key (shared with others). These keys help prove that a message was really sent by the sender and protect against forgery or tampering.

Signing Algorithms:
To create a digital signature, the message is first converted into a short fixed-length value called a hash using a hash function. This hash is then encrypted using the sender's private key to form the digital signature. The signature is attached to the message and sent to the receiver. Hashing the message instead of signing the whole message makes the process faster and efficient.

Signature Verification Algorithms:
The receiver gets the message and digital signature, decrypts the signature using the sender's public key to get the original hash, and also computes a new hash from the received message. If both hashes match, the signature is valid and the message is authentic and unchanged. If not, the signature is invalid.

Applications of digital signatures:-

1. **Email Security**
   Digital signatures ensure that an email is sent by the claimed sender and that its content is not altered during transit.

2. **Software Distribution**
   Developers sign software or updates digitally to prove the software's authenticity and integrity, preventing tampering by attackers.

3. **Legal Documents**
   Digital signatures are used to sign contracts, agreements, and other legal documents electronically, making them legally binding and secure.

4. **Financial Transactions**
   Banks and financial institutions use digital signatures to secure online transactions, ensuring authenticity and preventing fraud.

5. **Authentication and Access Control**
   Digital signatures verify the identity of users or devices accessing systems, networks, or sensitive data.

6. **Blockchain and Cryptocurrencies**
   Transactions in blockchain networks are verified using digital signatures to ensure the legitimacy of each transaction.

iii) Digital Certificate:-

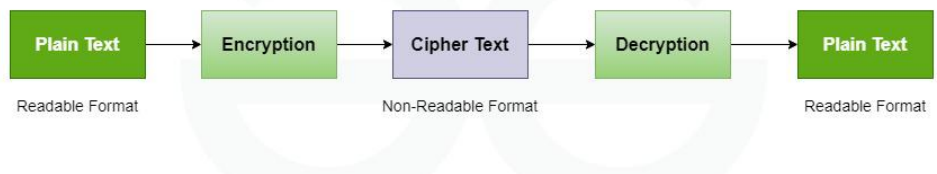
- A **Digital Certificate** is an electronic document used to prove the ownership of a public key.
- It helps others trust that the public key belongs to the person, organization, or device it claims to represent.
- **Types of Certificates:** There are various types like SSL/TLS certificates for websites, code signing certificates for software, and client certificates for user authentication.
    - It is issued by a trusted third party called a **Certificate Authority (CA)**.
    - The certificate contains information like:
        - The owner's name
        - The owner's public key
        - The issuer's name (CA)
        - Expiry date of the certificate
        - Digital signature of the CA to verify authenticity

Purpose:

- To verify the identity of the certificate holder.
- To enable secure communication by ensuring public keys are genuine and trustworthy.

iv) Cryptography:-

- Cryptography is the science of protecting information by transforming it into a secure format so that only authorized people can understand it.
- The prefix "crypt" means "hidden" and the suffix "graphy" means "writing". In Cryptography, the techniques that are used to protect information are obtained from mathematical concepts and a set of rule-based calculations known as algorithms to convert messages in ways that make it hard to decode them.



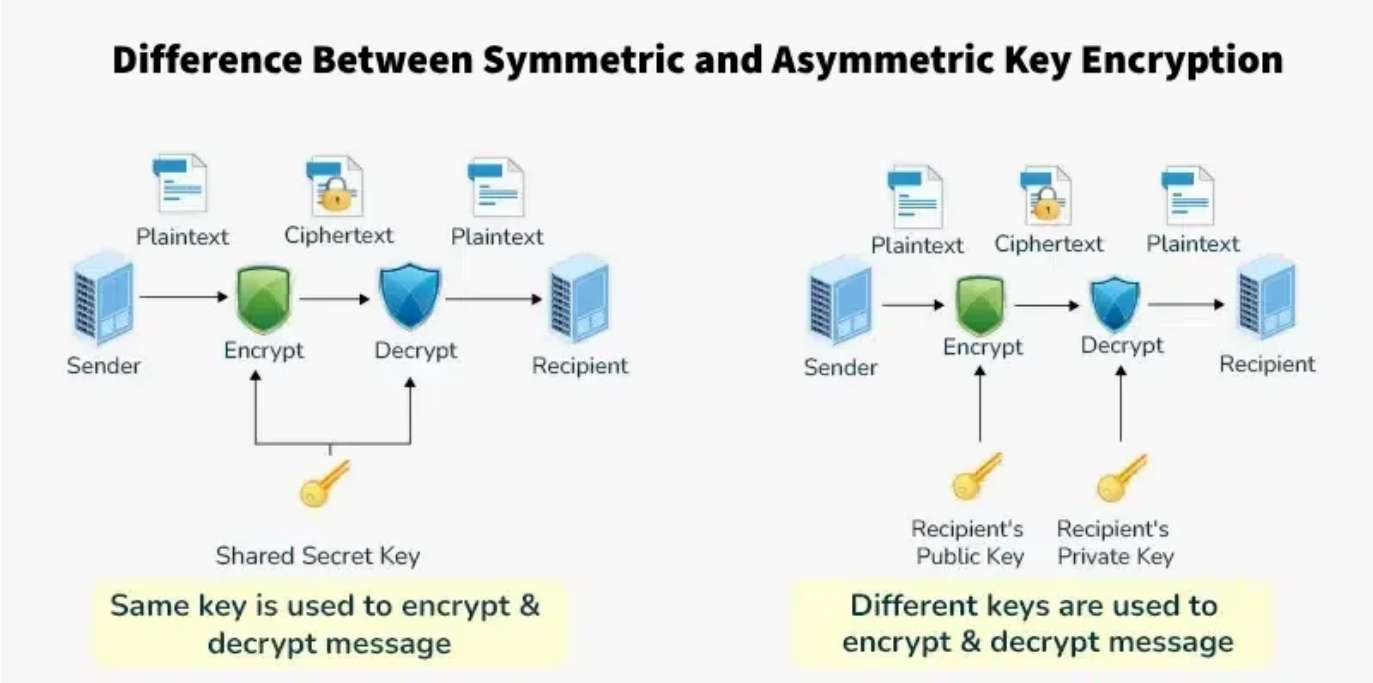
- It involves **encoding (encrypting)** data to keep it secret during communication or storage.
- The process of converting encrypted data back to its original form is called **decoding (decrypting)**.

- Cryptography ensures **confidentiality** (only authorized users can read data), **integrity** (data is not altered), **authentication** (verifying the identity of users), and **non-repudiation** (proof that a sender cannot deny sending a message).
- It uses mathematical algorithms and keys to perform encryption and decryption.

  - There are two main types of cryptography:
    - **Symmetric-key cryptography**: The same key is used for both encryption and decryption.
    - **Asymmetric-key cryptography**: Uses a pair of keys—a public key to encrypt and a private key to decrypt.

Features Of Cryptography:-
- Confidentiality:
- Integrity:
- Non-repudiation:
- Authentication:
- Interoperability:

v) Symmetric and Asymmetric Key:-



Symmetric Key Encryption	Asymmetric Key Encryption
It only requires a single key for both encryption and decryption.	It requires two keys, a public key and a private key, one to encrypt and the other to decrypt.
The size of ciphertext is the same or smaller than the original plaintext.	The size of ciphertext is the same or larger than the original plaintext.
The encryption process is very fast.	The encryption process is slow.
It is used when a large amount of data needs to be transferred.	It is used to transfer small amount of data.
It only provides confidentiality.	It provides confidentiality, authenticity, and non-repudiation.
The length of key used is 128 or 256 bits	The length of key used is 2048 or higher
In symmetric key encryption, resource utilization is low compared to asymmetric key encryption.	In asymmetric key encryption, resource utilization is high.
It is efficient as it is used for handling large amount of data.	It is comparatively less efficient as it can handle a small amount of data.
Security is lower as only one key is used for both encryption and decryption purposes.	Security is higher as two keys are used, one for encryption and the other for decryption.
The Mathematical Representation is as follows- P = D (K, E(K, P)) where K --> encryption and decryption key P --> plain text D --> Decryption E(K, P) --> Encryption of plain text using K	The Mathematical Representation is as follows- P = D(Kd, E (Ke,P)) where Ke --> encryption key Kd --> decryption key D --> Decryption E(Ke, P) --> Encryption of plain text using encryption key Ke. P --> plain text
Examples: 3DES, AES, DES and RC4	**Examples:** Diffie-Hellman, ECC, El Gamal, DSA and RSA

3. Define and list various computer network security mechanisms Also write short notes on the following terms
i) Encryption
ii) Decryption

Computer network security mechanisms are methods or techniques used to protect data and resources from unauthorized access, misuse, or attacks in a computer network.

Computer Network Security Mechanisms:

- Encryption and Decryption
- Authentication Mechanisms.
- Access Control.
- Intrusion Detection and Prevention Systems (IDS/IPS).
- Antivirus and Anti-malware Software.
- Public Key Infrastructure (PKI)

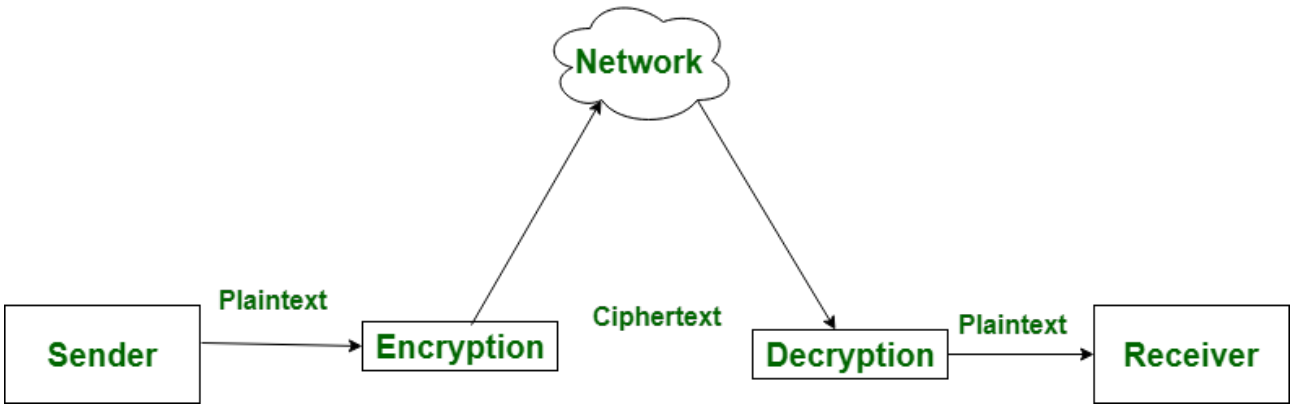
i) Encryption:

Encryption is the process of converting plain (readable) data into an unreadable form (called ciphertext) using a key and an algorithm. It protects the confidentiality of data during transmission or storage.

- **Example:** If you send an encrypted message, only the person with the correct decryption key can read it.

Application of Encryption:-

- **Online Banking:** To secure transactions, use online banking.
- **Email security:** To safeguard the contents of emails.
- **Secure Messaging:** To protect the privacy of discussions.
- **Data Storage:** To prevent unwanted access to data that has been stored.



ii) Decryption:

Decryption is the reverse of encryption. It is the process of converting the encrypted (ciphertext) data back into its original (plaintext) form using a key.

- **Example:** The receiver uses the decryption key to turn the unreadable data back into its original readable format.

Application of Encryption and Decryption

- **WhatsApp Messaging: It encrypts** It encrypts communications from beginning to end so that only the sender and recipient can read them.
- **HTTPS websites:** Encrypt user data to prevent third parties from intercepting it.
- **Encrypted Email Services:** Email services that use encryption, like ProtonMail, protect email contents.

4. Diffie-Hellman Algorithm:-

- The **Diffie-Hellman Key Exchange Algorithm** is a method used in cryptography to securely share a secret key between two parties over a public channel.
-
- It was developed by **Whitfield Diffie** and **Martin Hellman** in 1976 and is considered one of the first practical implementations of public-key exchange.
-
- The algorithm allows two people to generate a shared **secret key** that can be used for **encrypting messages**, even if an attacker is listening to the communication.
- The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret communications while exchanging data over a public network.

Steps :-

1. Publicly Shared Values

- Choose a **large prime number p** and a **base g** (also called generator), where $1 < g < p$.
- These values are **public** and known to everyone.

2. Private Key Generation

- Each party chooses a **private key**:
    - Alice chooses a secret number a.
    - Bob chooses a secret number b.

3. Compute Public Keys

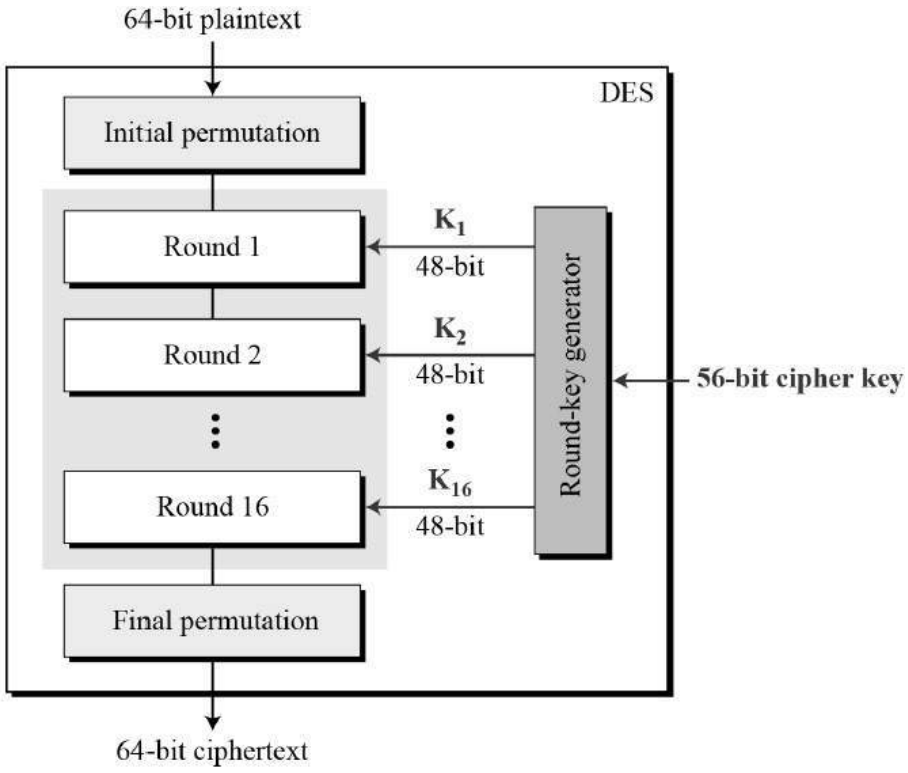
- Alice computes A = g^a mod p and sends A to Bob.
- Bob computes B = g^b mod p and sends B to Alice.

4. Compute Shared Secret Key

- Alice receives B and computes the key: K = B^a mod p.
- Bob receives A and computes the key: K = A^b mod p.

5. Data Encryption Standard (DES):-

- The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).
- It is used to **encrypt and decrypt** blocks of data using the same key.
- DES is based on the two attributes of Feistel cipher i.e. Substitution (also called confusion) and Transposition (also called diffusion).
- DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit.
- DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only).



Steps:-

1. **Initial Permutation (IP):**
   The DES algorithm starts by rearranging the 64-bit plain text using a fixed pattern called the Initial Permutation (IP). This step doesn't change the content but scrambles the order of bits to increase security.

2. **Splitting the Data:**
   After permutation, the 64-bit data is split into two equal parts: a left half (L0) and a right half (R0), each containing 32 bits. These two halves will be used repeatedly in the encryption process.

3. **16 Rounds of Processing:**
   The heart of DES is 16 rounds of the Feistel structure. In each round, the right half is expanded to 48 bits and combined with a 48-bit round key (generated from the original key). The result goes through substitution using S-boxes, then through a permutation (P-box), and finally is XORed with the left half. After this, the halves are swapped for the next round. This process is repeated 16 times, each time using a different round key.

4. **Final Round (No Swap):**
   After completing all 16 rounds, the last swap is skipped. The left and right halves from the final round are simply combined back together.

5. **Inverse Initial Permutation (IP^{-1}):**
   Finally, the combined 64-bit output is passed through the inverse of the initial permutation. This step reverses the initial scrambling and produces the final 64-bit ciphertext — the encrypted version of the original message.

6. RSA Algorithm in Cryptography:-

- RSA(**R**ivest-**S**hamir-**A**dleman) Algorithm is an **asymmetric** or **public-key cryptography** algorithm which means it works on two different keys: **Public Key** and **Private Key**.
-
- The Public Key is used for **encryption** and is known to everyone, while the Private Key is used for **decryption** and must be kept secret by the receiver.
-
- RSA Algorithm is named after Ron **R**ivest, Adi **S**hamir and Leonard **A**dleman, who published the algorithm in 1977.
-
- **Example of Asymmetric Cryptography:**

If Person **A** wants to send a message securely to Person **B**:

- Person A **encrypts** the message using Person B's **Public** Key.
- Person B **decrypts** the message using their **Private** Key.

Steps:-

1. **Key Generation:** Creating Public and Private Keys

- Choose two large prime numbers: p and q.
- Compute n = p × q. (This is used as the modulus for both keys.)
- Compute $\phi(n) = (p - 1)(q - 1)$.
- Choose an integer e such that $1 < e < \phi(n)$ and e is **coprime** to $\phi(n)$ (common choice: e = 65537).
- Compute the private key d such that $(d \times e) \% \phi(n) = 1$.

2. **Encryption:** Sender encrypts the data using Public Key to get cipher text.

To encrypt a message M:

- Convert M into an integer m such that m < n.
- Compute ciphertext: c = m^e mod n

3. **Decryption:** Decrypting the cipher text using Private Key to get the original data.

To decrypt the ciphertext c:

- Compute original message: m = c^d mod n

Applications of RSA

- Secure data transmission
- Digital signatures
- SSL/TLS certificates for websites
- Email encryption