UNIT 03:

NUMBER THEORY AND ASYMMETRIC KEY CRYPTOGRAPHY **AFTAB-1405**

01. DISCUSS COMPONENTS OF PUBLIC KEY CRYPTOSYSTEM AND THEN EXPLAIN SECRECY AND AUTHENTICATION IN IT. STATE APPLICATIONS OF IT. (MT)

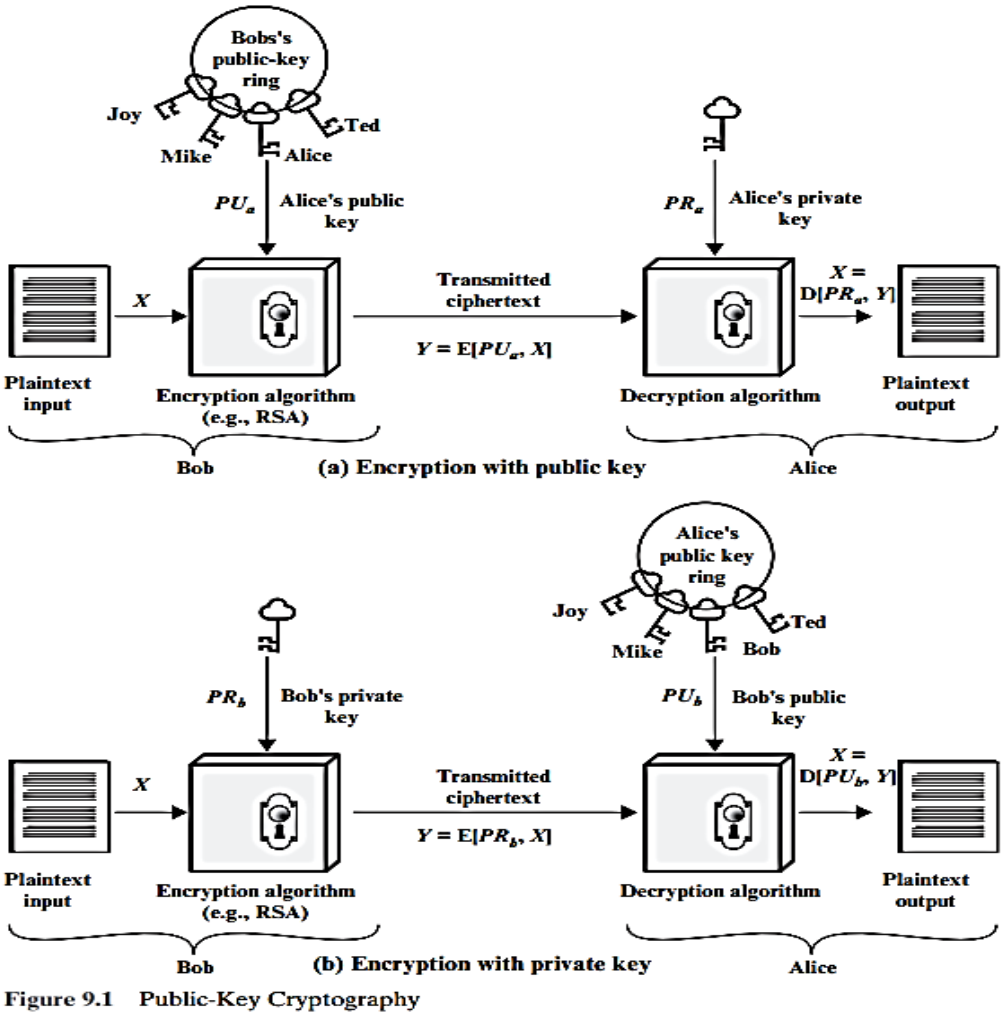

Figure 9.1 Public-Key Cryptography

Components of a Public-Key Cryptosystem

A public-key cryptosystem relies on a pair of ***mathematically linked keys:*** a ***public key and a private key.*** These keys work together to achieve encryption and decryption.

Here's a breakdown of the key components:

- **Plaintext:** The original message in a readable format.
- **Encryption Algorithm:** This ***algorithm transforms the plaintext into an unreadable format called ciphertext*** using either the public key or the private key.
- **Public Key & Private Key:** These are mathematically related keys. The public key is widely distributed and accessible to anyone. The ***private key is kept secret by the owner.***
- **Ciphertext:** The encrypted message that results from the encryption algorithm.
- **Decryption Algorithm:** This algorithm uses the matching key (either public or private depending on the encryption process) to transform the ciphertext back to the original plaintext message.

Here's how it works:

1. **Key Generation:** Each user generates a public-private key pair.
2. **Public Key Distribution:** The ***public key is made available to others,*** often through a ***public repository.***
3. **Encryption:** If Bob wants to send a secure message to Alice, he uses Alice's public key to encrypt the message.
4. **Decryption:** Only Alice can decrypt the message using her private key.

This system ***eliminates the need for pre-shared secret keys,*** making key distribution much simpler and more secure.

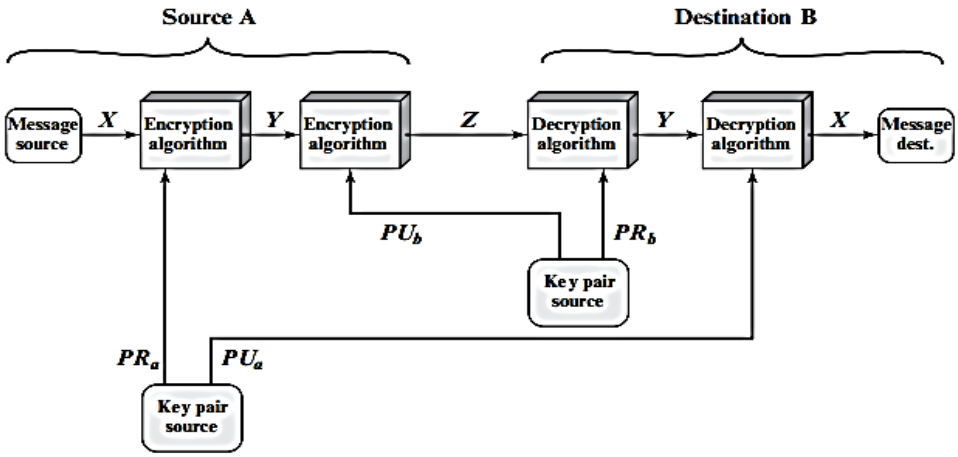

Figure 9.4 Public-Key Cryptosystem: Authentication and Secrecy

Secrecy and Authentication in Public-Key Cryptosystems

Secrecy:

- Public-key cryptography ***ensures secrecy by using the recipient's public key for encryption.*** Only the corresponding private key, held by the recipient, can decrypt the message.
- An eavesdropper intercepting the ciphertext wouldn't be able to decrypt it without the private key, protecting the confidentiality of the message.

Authentication:

Public-key cryptography can also provide a level of authentication:

- **Digital Signatures:** By signing a message with their private key before encryption, a sender can prove their identity. The recipient can then use the sender's public key to verify the signature and ensure the message originated from the claimed sender. This ***helps prevent forgery and repudiation (denial of sending the message).***

However, it's important to note that public-key cryptography alone doesn't guarantee complete sender authentication. Often, a combination of techniques like digital certificates and PKI (Public Key Infrastructure) is used for robust sender authentication.

Applications of Public-Key Cryptosystems

Public-key cryptography has numerous applications in securing digital communication and data:

- **Secure Email:** Protects email content from unauthorized access during transmission.
- **E-commerce Transactions:** Encrypts credit card information and other sensitive data during online transactions.
- **Secure Logins:** Enables secure logins for websites and applications using digital certificates and password encryption.
- **Virtual Private Networks (VPNs):** ***Creates a secure tunnel*** for encrypted communication over a public network.
- **Digital Signing:** Ensures the authenticity and integrity of digital documents.
- **Software Distribution:** ***Verifies the authenticity and integrity*** of software downloads.

Public-key cryptography plays a vital role in ***securing our digital interactions and protecting sensitive information.***

02. STATE AND EXPLAIN REQUIREMENTS OF PUBLIC KEY CRYPTOSYSTEM. EXPLAIN APPLICATIONS OF PUBLIC KEY CRYPTOSYSTEM.

Public-Key Cryptosystem

A public-key cryptosystem is a cryptographic system that utilizes a pair of mathematically linked keys: a public key and a private key. The public key is widely distributed and accessible to anyone, while the private key is kept secret by its owner. This system allows for secure communication and data protection through encryption and decryption.

Requirements of Public-Key Cryptosystems

Public-key cryptography relies on a specific set of properties to function securely. These requirements ensure the system can achieve confidentiality, authentication, and resistance to various attacks. Here's a breakdown of the key requirements:

1. **Efficient Key Generation:** It should be computationally easy for users to generate a key pair (public and private key). This ***ensures users can quickly set up the system without complex calculations.***
2. **Efficient Encryption:** Encrypting a message with the recipient's public key ***should be a quick and manageable process.*** This ***allows for smooth communication*** without significant delays.
3. **Efficient Decryption:** The recipient should be able to decrypt the message using their private key efficiently. This ***ensures timely access to the information after encryption.***
4. **Computational Infeasibility of Private Key Derivation:** Knowing only the public key, it should be computationally impossible for someone to determine the corresponding private key. This ***safeguards the secrecy of private keys,*** which are crucial for decryption.

5. **Computational Infeasibility of Message Recovery:** Even with the public key and the encrypted message (ciphertext), it should be computationally infeasible to recover the original message. This ***protects the confidentiality of the communication.***

These requirements are strict because very few algorithms meet them effectively. This highlights the importance of carefully chosen algorithms for secure public-key cryptography.

Applications of Public-Key Cryptosystems

Public-key cryptography offers a range of advantages in securing digital communication and data. Here are some of its prominent applications:

- **Secure Email:** Public-key cryptography can encrypt email content, ensuring only the intended recipient can decrypt and read the message.
- **E-commerce Transactions:** It protects sensitive data like credit card information during online transactions by encrypting it with the recipient's (e.g., merchant's) public key.
- **Secure Logins:** Websites and applications can leverage public-key cryptography with digital certificates to establish secure login sessions and protect passwords.
- **Virtual Private Networks (VPNs):** VPNs use public-key cryptography to create encrypted tunnels for secure communication over public networks.
- **Digital Signatures:** Public-key cryptography is a cornerstone of digital signatures. By signing a message with their private key before encryption, senders can prove their identity and ensure the message hasn't been tampered with.
- **Software Distribution:** Software downloads can be verified for authenticity and integrity using digital signatures based on public-key cryptography.

Overall, public-key cryptography plays a vital role in building trust and security in our digital interactions. It empowers secure communication, protects sensitive data, and safeguards electronic transactions.

03. EXPLAIN IN DETAIL SECURITY OF RSA WITH RESPECT TO DIFFERENT PARAMETERS.

The security of the RSA algorithm is primarily related to the difficulty of factoring large numbers, specifically the product of two large prime numbers (n = p * q). Here's a detailed explanation:

1. **Brute Force Attack:**
    - The ***defence against brute-force attacks is to use a large key space,*** i.e., a ***large number of bits in the private key (d).***
    - The ***larger the key size,*** the ***more secure the system,*** but also the ***slower the encryption/decryption process.***
2. **Mathematical Attacks:**
    - There are ***three main mathematical approaches*** to attacking RSA:

1. ***Factoring n*** into its ***two prime factors p and q.***
2. Determining the ***totient function φ(n) = (p-1)(q-1)*** without first determining p and q.
3. ***Determining the private key d directly,*** without first determining φ(n).

3. **Timing Attacks:**
   - Timing attacks exploit the fact that the ***decryption process may take extra amount of time*** depending on the private key bits.
4. **Hardware Fault-based Attacks:**
   - These attacks ***involve inducing hardware faults in the processor*** generating ***digital signatures to recover the private key.***
   - While a practical attack, it requires physical access to the target machine and the ability to control the input power to the processor, which is not a trivial requirement.
5. **Chosen Ciphertext Attacks (CCA):**
   - The basic RSA algorithm is vulnerable to CCA, where the attacker can exploit properties of RSA to obtain information about the private key.
   - To mitigate this, practical RSA-based cryptosystems use techniques like optimal asymmetric encryption padding (OAEP) to randomize the ciphertext and prevent the exploitation of such properties.

In summary, the ***security of RSA depends on the careful selection of parameters,*** such as the ***key size, the choice of prime factors,*** and the ***use of countermeasures*** against various attacks. Continuous advancements in factoring algorithms and computing power necessitate the periodic increase of the recommended key sizes to maintain the desired level of security.

04: STATE AND EXPLAIN IN DETAIL DIFFIE-HELLMAN KEY EXCHANGE ALGORITHM.

Purpose: The Diffie-Hellman key exchange algorithm enables two parties to securely establish a shared secret key over an insecure communication channel. This shared key can then be used for subsequent symmetric encryption of messages, ensuring confidentiality in communication.

Algorithm Overview:

1. **Setup**:
   - Both parties agree on two public parameters: a large prime number q and a primitive root a modulo q. These parameters are shared publicly.
2. **Key Generation**:
   - Each party independently generates a private key:
     - Alice selects XA, and Bob selects XB.

3. **Public Key Calculation**:
    - Using their private keys and the shared parameters, each party computes their public key:
        - Alice computes $YA=aXA \bmod q$.
        - Bob computes $YB=aXB \bmod q$.
4. **Shared Secret Calculation**:
    - Upon receiving the other party's public key, each party computes the shared secret key:
        - Alice computes $K=(YB)XA \bmod q$.
        - Bob computes $K=(YA)XB \bmod q$.
5. **Communication**:
    - Both parties now share a secret key K and can use it for symmetric encryption of messages.

Mathematical Principles:

- The security of Diffie-Hellman relies on the computational difficulty of calculating discrete logarithms modulo a prime number.
- Discrete logarithms are difficult to compute, making it infeasible to determine the private keys XA and XB from the public keys YA and YB.

Security Considerations:

- Diffie-Hellman is secure against passive eavesdropping attacks because private keys are never transmitted.
- The security lies in the difficulty of calculating discrete logarithms, particularly for large primes.

Potential Vulnerabilities:

- The algorithm is vulnerable to man-in-the-middle attacks, where an adversary intercepts and alters the communication between the two parties.
- Additional measures such as digital signatures or public-key certificates can be employed to authenticate the participants and mitigate such attacks.

Example:

- A numerical example demonstrating the key exchange process using specific parameters and calculations can further illustrate the algorithm's operation.