EXPT _1

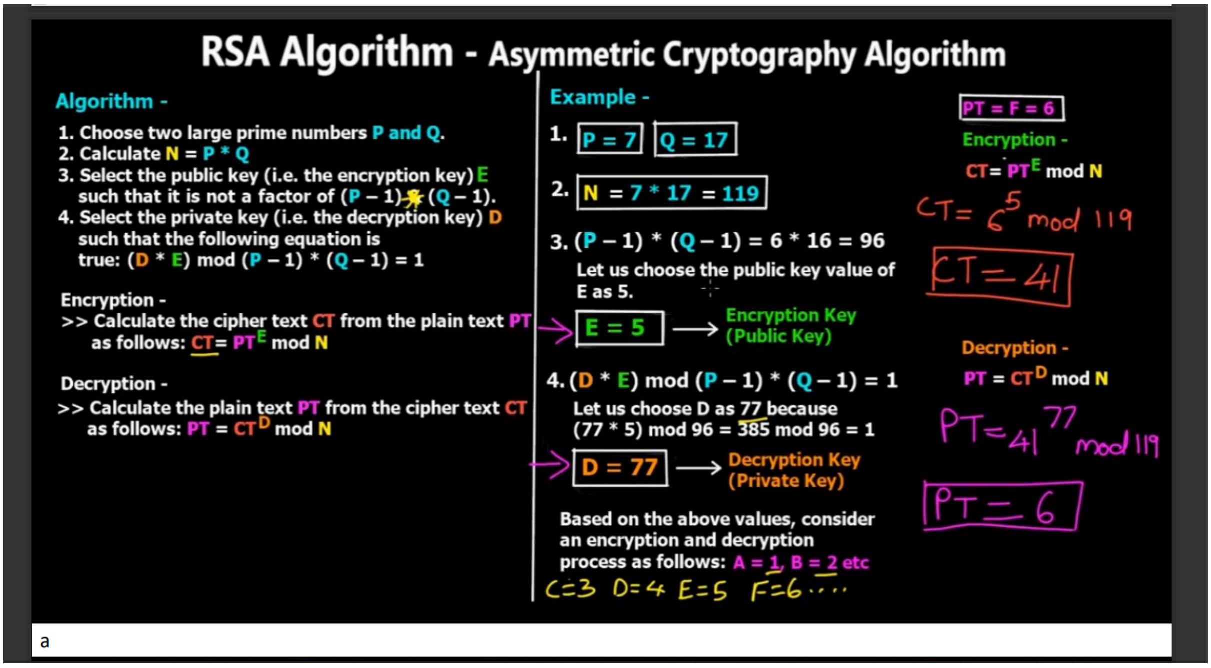
1. Write a program for default passwords, printed passwords and password in plain text form. Draw flowchart, algorithm and attach output results for the same.

   ANS -

```
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string storedPassword = "Secure@123";
    int tries = 4;

    while (tries > 0) {
        string enteredPassword;
        cout << "Enter a password: ";
        cin >> enteredPassword;

        if (enteredPassword == storedPassword) {
            cout << "Correct! You entered the correct password." << endl;
            break;
        } else {
              cout << "Incorrect! You entered the wrong password. Remaining tries: " <<
tries - 1 << endl;
            tries--;
        }

        if (tries == 0) {
            cout << "Access Denied! You have tried maximum number of times." << endl;
        }
        }

        return 0;
    }
```

2. Explain RSA algorithm in detail with example.

RSA (Rivest-Shamir-Adleman) is a widely used asymmetric cryptographic algorithm for secure data transmission and digital signatures.



RSA Algorithm - Asymmetric Cryptography Algorithm

Algorithm -

1. Choose two large prime numbers P and Q.
2. Calculate N = P * Q
3. Select the public key (i.e. the encryption key) E such that it is not a factor of (P – 1) * (Q – 1).
4. Select the private key (i.e. the decryption key) D such that the following equation is true: (D * E) mod (P – 1) * (Q – 1) = 1

Encryption -
>> Calculate the cipher text CT from the plain text PT as follows: $CT = PT^E \bmod N$

Decryption -
>> Calculate the plain text PT from the cipher text CT as follows: $PT = CT^D \bmod N$

Example -

1. $\boxed{P = 7}$ $\boxed{Q = 17}$

2. $\boxed{N = 7 * 17 = 119}$

3. (P – 1) * (Q – 1) = 6 * 16 = 96
   Let us choose the public key value of E as 5.

$\boxed{E = 5} \longrightarrow$ Encryption Key (Public Key)

4. (D * E) mod (P – 1) * (Q – 1) = 1
   Let us choose D as 77 because (77 * 5) mod 96 = 385 mod 96 = 1

$\boxed{D = 77} \longrightarrow$ Decryption Key (Private Key)

Based on the above values, consider an encryption and decryption process as follows: A = 1, B = 2 etc
C=3 D=4 E=5 F=6....

$\boxed{PT = F = 6}$

Encryption -

$CT = PT^E \bmod N$

$CT = 6^5 \bmod 119$

$\boxed{CT = 41}$

Decryption -

$PT = CT^D \bmod N$

$PT = 41^{77} \bmod 119$

$\boxed{PT = 6}$

a

3. Explain any two of following with example

    i. Playfair Cipher

    ii. Hill Cipher

    We have to encrypt the message ‘ACT’ (n=3).The key is ‘GYBNQKURP’ which can be written as the nxn matrix:

$$\begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix}$$

Cipherkey

The message ‘ACT’ is written as vector:

$$\begin{bmatrix} 0 \\ 2 \\ 19 \end{bmatrix}$$

message vector

The enciphered vector is given as:

$$\begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 19 \end{bmatrix} = \begin{bmatrix} 67 \\ 222 \\ 319 \end{bmatrix} \equiv \begin{bmatrix} 15 \\ 14 \\ 7 \end{bmatrix} \pmod{26}$$

enciphered vector

which corresponds to ciphertext of ‘POH’

Decryption

To decrypt the message, we turn the ciphertext back into a vector, then simply multiply by the inverse matrix of the key matrix (IFKVIVVMI in letters).The inverse of the matrix used in the previous example is:

$$\begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix}^{-1} \equiv \begin{bmatrix} 8 & 5 & 10 \\ 21 & 8 & 21 \\ 21 & 12 & 8 \end{bmatrix} \pmod{26}$$

inverse matrix

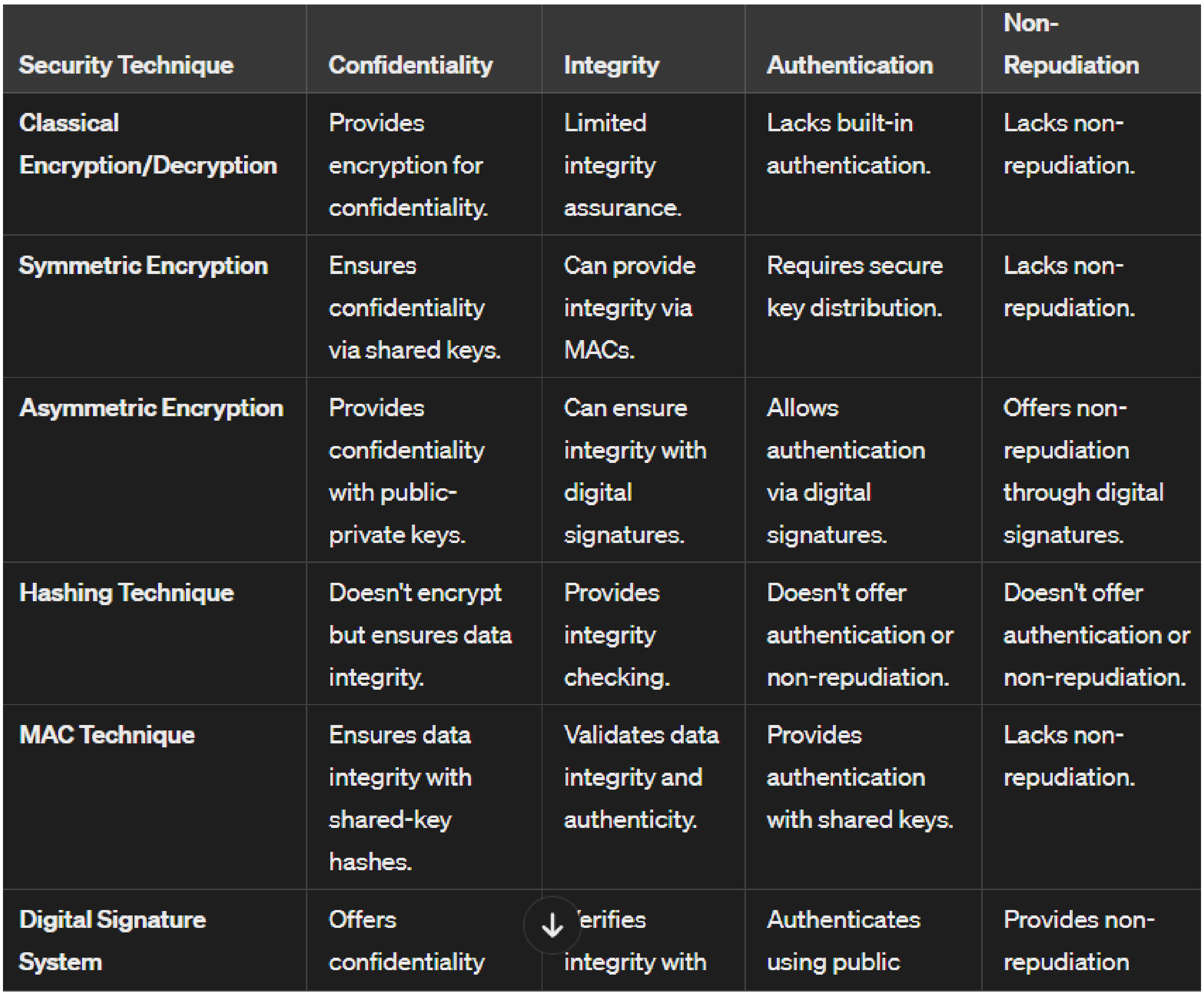
For the previous Ciphertext ‘POH’:

Decrypt

$$\begin{bmatrix} 8 & 5 & 10 \\ 21 & 8 & 21 \\ 21 & 12 & 8 \end{bmatrix} \begin{bmatrix} 15 \\ 14 \\ 7 \end{bmatrix} \equiv \begin{bmatrix} 260 \\ 574 \\ 539 \end{bmatrix} \equiv \begin{bmatrix} 0 \\ 2 \\ 19 \end{bmatrix} \pmod{26}$$

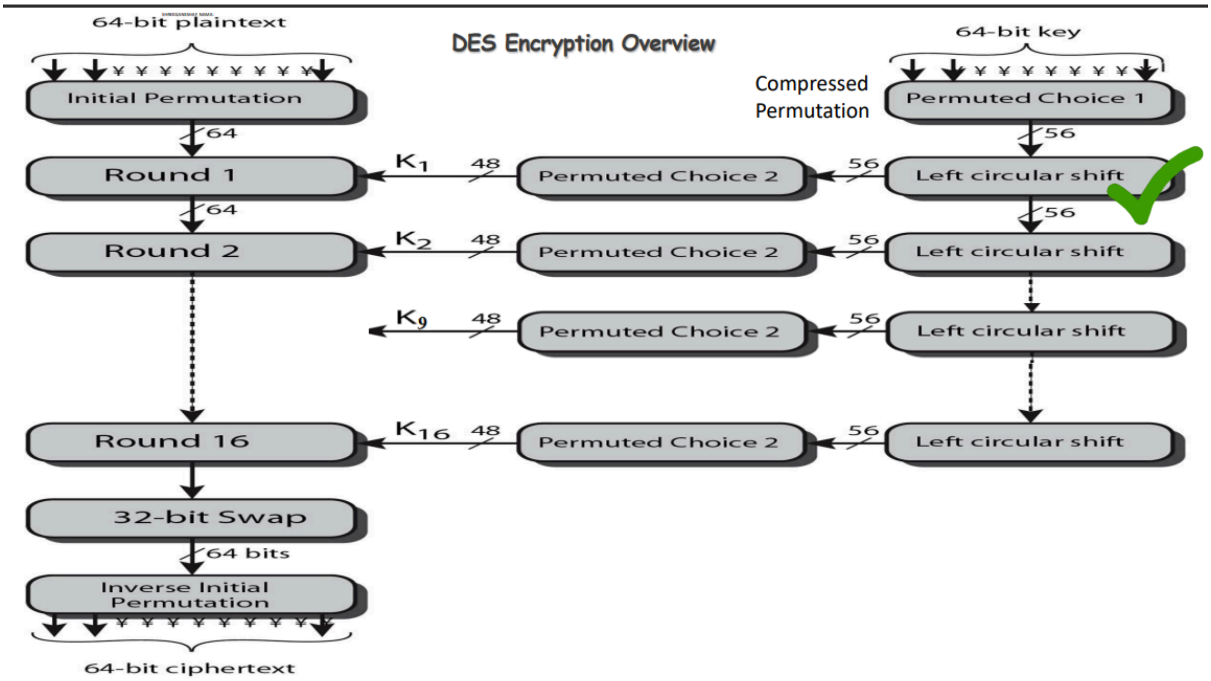
which gives us back ‘ACT’.

4. Compare in Tabular fashion, Confidentiality, Integrity, Authentication and Non repudiation for following security techniques
    a. Classical Encryption/Decryption
    b. Symmetric Encryption
    c. Asymmetric Encryption
    d. Hashing Technique
    e. MAC technique
    f. Digital Signature System



Security Technique	Confidentiality	Integrity	Authentication	Non-Repudiation
Classical Encryption/Decryption	Provides encryption for confidentiality.	Limited integrity assurance.	Lacks built-in authentication.	Lacks non-repudiation.
Symmetric Encryption	Ensures confidentiality via shared keys.	Can provide integrity via MACs.	Requires secure key distribution.	Lacks non-repudiation.
Asymmetric Encryption	Provides confidentiality with public-private keys.	Can ensure integrity with digital signatures.	Allows authentication via digital signatures.	Offers non-repudiation through digital signatures.
Hashing Technique	Doesn't encrypt but ensures data integrity.	Provides integrity checking.	Doesn't offer authentication or non-repudiation.	Doesn't offer authentication or non-repudiation.
MAC Technique	Ensures data integrity with shared-key hashes.	Validates data integrity and authenticity.	Provides authentication with shared keys.	Lacks non-repudiation.
Digital Signature System	Offers confidentiality	Verifies integrity with	Authenticates using public	Provides non-repudiation

5. Explain DES algorithm in detail.



Key Generation:

DES employs a 56-bit key, but the input key is typically 64 bits long. The extra 8 bits are used for parity checks.

The input key undergoes a process of permutation and compression to generate 16 round keys, each 48 bits long.

The round keys are derived from the original key using a combination of permutation tables and left shifts.

Encryption:

The plaintext is divided into blocks of 64 bits and undergoes an initial permutation (IP).

The 64-bit block is split into two 32-bit halves: left (L0) and right (R0).

Each round of encryption consists of several steps:

Expansion: The 32-bit right half is expanded to 48 bits using an expansion permutation table.

Key Mixing: The expanded right half is combined with the corresponding round key using bitwise XOR.

Substitution: The XOR result is divided into 8 groups of 6 bits each, which are substituted using eight S-boxes (substitution boxes).

Permutation: The output of the S-boxes is rearranged using a fixed permutation table.

XOR and Swap: The permuted output is XORed with the left half, and the new right half becomes the old left half.

After 16 rounds of processing, the left and right halves are swapped, and the final permutation (inverse of the initial permutation) is applied to produce the ciphertext.

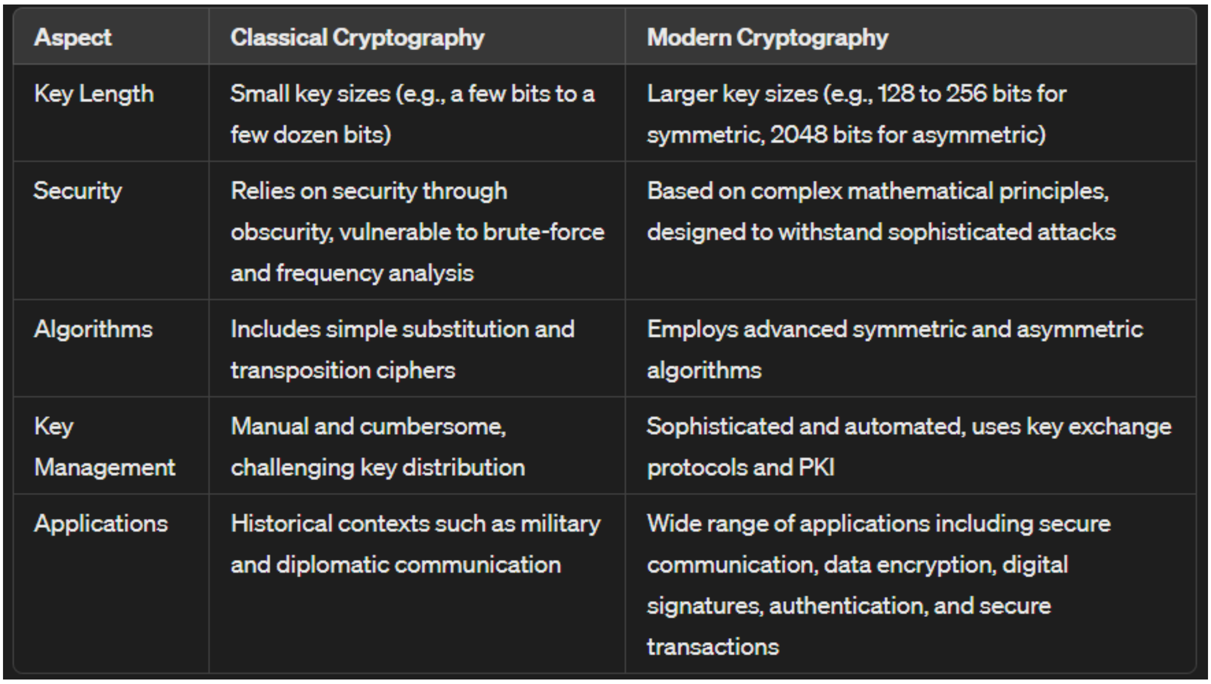
Decryption:

Decryption in DES is essentially the same process as encryption, but with the round keys applied in reverse order.

The ciphertext undergoes the initial permutation, followed by 16 rounds of decryption using the round keys in reverse order.

After the final round, the left and right halves are swapped, and the inverse of the initial permutation is applied to produce the plaintext.

6. Compare classical Vs Modern Cryptography.



Aspect	Classical Cryptography	Modern Cryptography
Key Length	Small key sizes (e.g., a few bits to a few dozen bits)	Larger key sizes (e.g., 128 to 256 bits for symmetric, 2048 bits for asymmetric)
Security	Relies on security through obscurity, vulnerable to brute-force and frequency analysis	Based on complex mathematical principles, designed to withstand sophisticated attacks
Algorithms	Includes simple substitution and transposition ciphers	Employs advanced symmetric and asymmetric algorithms
Key Management	Manual and cumbersome, challenging key distribution	Sophisticated and automated, uses key exchange protocols and PKI
Applications	Historical contexts such as military and diplomatic communication	Wide range of applications including secure communication, data encryption, digital signatures, authentication, and secure transactions

7. What is DoS attack? What is DDoS attack? How to Mitigate it?

DoS (Denial of Service) Attack:

Malicious attempt to disrupt a server, service, or network by overwhelming it with illegitimate traffic.

Usually from a single source.

Goal is to make the target unavailable to legitimate users.

DDoS (Distributed Denial of Service) Attack:

Variant of DoS attack where multiple compromised computers (botnets) flood the target simultaneously.

More challenging to mitigate due to distributed nature.

Mitigation:

Network Filtering, Traffic Shaping.

Anomaly Detection, Intrusion Prevention Systems.

Content Delivery Networks (CDNs), Scrubbing Centers.

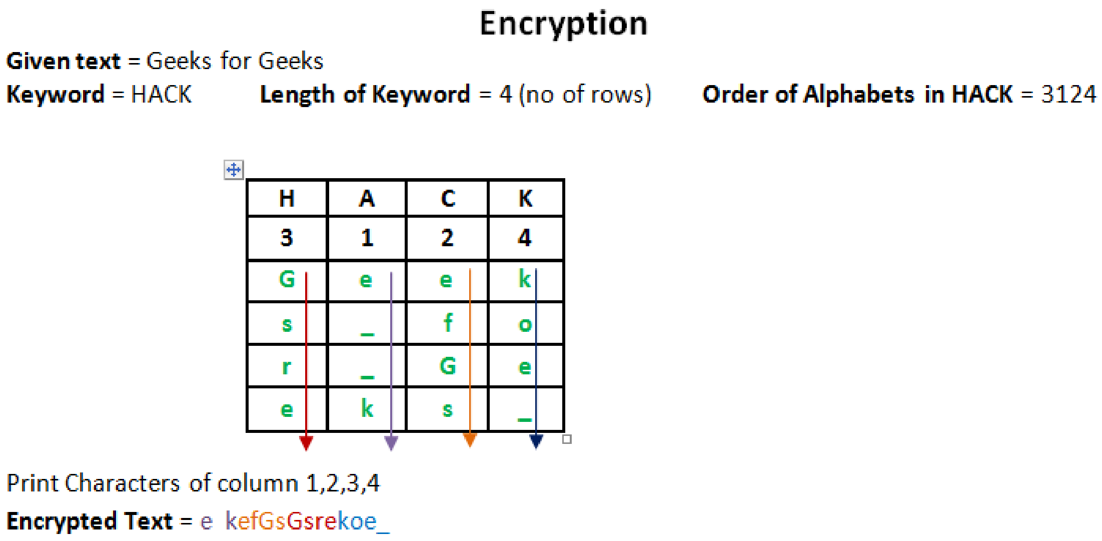
Load Balancers, Rate Limiting, Throttling.

Incident Response Plans.

By employing these techniques, organizations can better defend against and mitigate the impact of DoS and DDoS attacks.

EXPT _2

8. Write a program of encryption and decryption for transposition cipher. Draw flowchart, algorithm and attach output results for the same.



Encryption

Given text = Geeks for Geeks

Keyword = HACK **Length of Keyword** = 4 (no of rows) **Order of Alphabets in HACK** = 3124

H	A	C	K
3	1	2	4
G	e	e	k
s	_	f	o
r	_	G	e
e	k	s	_

Print Characters of column 1,2,3,4

Encrypted Text = e kefGsGsrekoe_

```
#include <bits/stdc++.h>

using namespace std;


//Encription function

string Encryption(int no_rows, int len_key, int len_msg, string msg, int col_val[])

{

   int x = 0;

   char enc_mat[no_rows + 1][len_key];

 //creating the matrix

   for (int i = 0; i < no_rows + 1; i++)

   {

      for (int j = 0; j < len_key; j++)

      {

        //initializes the positions with '_' after the end of message

         if (x >= len_msg)

         {

            enc_mat[i][j] = '_';
            }
            else
            {
                enc_mat[i][j] = msg[x];
            }
            x++;
        }
    }

    int t = 1;
    string cipher = "";
  //finding the cipher text according to the value of col_val matrix
    while (t <= len_key)
    {
        for (int i = 0; i < len_key; i++)
        {
            int k = col_val[i];
            if (k == t)
            {
                for (int j = 0; j < no_rows + 1; j++)
                {
                    cipher += enc_mat[j][i];
                }
                t++;
            }
        }
    }
    return cipher;
}
```

```
//decryption function
string Decryption(int no_rows, int len_key, string cipher,int col_val[])
{
    char dec_mat[no_rows + 1][len_key];
    int x = 0,t = 1;
  //rearrange the matrix according to the col_val
    while (t <= len_key)
    {
        for (int i = 0; i < len_key; i++)
        {
            int k = col_val[i];
            if (k == t)
            {
                for (int j = 0; j < no_rows + 1; j++)
                {
                    dec_mat[j][i]=cipher[x];
                    x++;
                }
                t++;
            }
        }
    }

    string message = "";
    for (int i = 0; i < no_rows + 1; i++)
    {
        for (int j = 0; j < len_key; j++)
        {
          //replacing the '_' with space
            if (dec_mat[i][j] == '_')
            {
                dec_mat[i][j] = ' ';
            }
            message += dec_mat[i][j];
        }
    }
    return message;
}

int main()
{
  //message
     string msg = "please transfer one million dollars to my swiss bank account six two
two four ";
  //key
    string key = "MEGABUCK";

    int len_key = key.length();
    int len_msg = msg.length();

    int val = 1,count = 0,ind;

    int col_val[len_key];
  //intializing col_val matrix with 0
    memset(col_val, 0, sizeof(col_val));
  //numbering the key alphabets according to its ACII value
    while (count < len_key)
    {
        int min = 999;
        for (int i = 0; i < len_key; i++)
        {
            if ((min > int(key[i])) && (col_val[i] == 0))
            {
                min = int(key[i]);
                ind = i;
            }
        }
        col_val[ind] = val;
        count++;
        val++;
    }

    int no_rows = len_msg / len_key;
//encrypted text
    string cipher_text = " ";
    cipher_text = Encryption(no_rows, len_key, len_msg, msg, col_val);
    cout << "Encrypted Message : " << cipher_text << endl;
//decrypted text
    string original_msg = " ";
    original_msg = Decryption(no_rows, len_key, cipher_text,col_val);
    cout << "Decrypted Message : " << original_msg << endl;
}
```

9. Use key
    a. MEGABUCK, or
    b. PICTENTG, or
    c. NBAISOKR
    d. for message: please transfer one million dollars to my swiss bank account six two two four

10. Explain C, I, A, Authentication and Non-Repudiation.

Confidentiality (C):

Protects sensitive information from unauthorized access or disclosure.

Achieved through encryption, access controls, and secure communication protocols.

Integrity (I):

Ensures data remains accurate, consistent, and unaltered.

Prevents unauthorized modification, deletion, or insertion of data.

Achieved through cryptographic hashing, digital signatures, and data validation.

Authentication (A):

Verifies the identity of users or systems accessing resources.

Confirms the legitimacy of claimed identities.

Utilizes methods like passwords, biometrics, and multi-factor authentication.

Non-Repudiation:

Prevents parties from denying actions they performed.

Ensures authenticity and integrity of messages or transactions.

Achieved through digital signatures, timestamps, and audit trails.

11. Explain the Interruption, Interception, Modification and Fabrication attack. Corelate the said attacks with C,I,A, Authentication and Non-Repudiation.

Interruption:

Attack: Disrupts availability of a service or system.

Correlation:

C: Compromises confidentiality by making data unavailable.

I: Prolonged unavailability raises concerns about integrity.

A: Prevents legitimate users from accessing authentication systems.

NR: Not directly related.

Interception:

Attack: Unauthorized access to data during transmission.

Correlation:

C: Compromises confidentiality by accessing sensitive data.

I: Integrity compromised if data is modified in transit.

A: Can lead to identity theft or impersonation.

NR: Undermined if intercepted communications are altered.

Modification:

Attack: Unauthorized alteration of data during transmission or storage.

Correlation:

C: Can compromise confidentiality if data is altered to reveal sensitive information.

I: Directly targets integrity by altering data.

A: Attackers may gain unauthorized access by modifying authentication credentials.

NR: Undermined if attackers falsify or manipulate data.

Fabrication:

Attack: Creation and insertion of counterfeit or unauthorized data.

Correlation:

C: Can compromise confidentiality by inserting false information.

I: Directly undermines integrity by inserting false data.

A: Involves creation of fake authentication credentials.

NR: Undermined if attackers create false evidence.

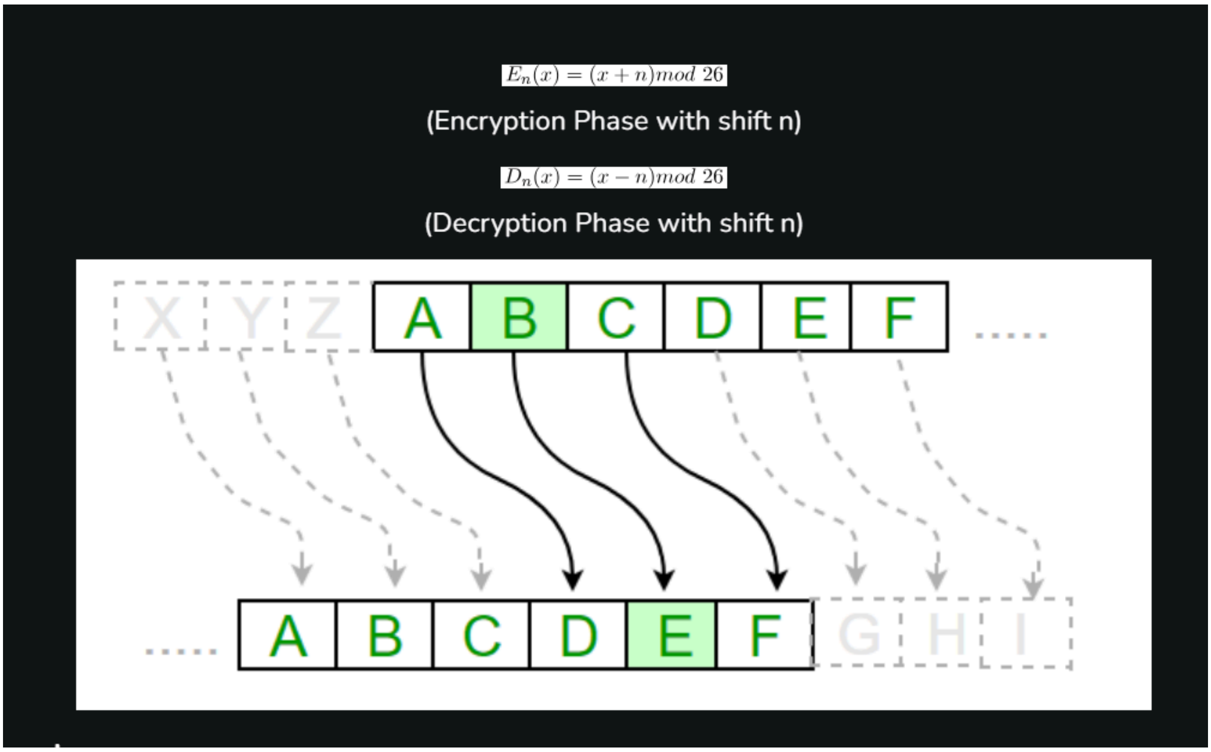
12. Explain RSA algorithm in detail. - **Expt _1**
13. Explain any two of following with example - **Expt_1**
    1. Playfair Cipher
    2. Hill Cipher

14. Compare in Tabular fashion, Confidentiality, Integrity, Authentication and Non repudiation for following security techniques - **Expt _1**
    a. Classical Encryption/Decryption
    b. Symmetric Encryption
    c. Asymmetric Encryption
    d. Hashing Technique
    e. MAC technique
    f. Digital Signature System

EXPT _3

15. Write a program of encryption and decryption for Substitution Cipher-Caesar Cipher.

Draw flowchart, algorithm and attach output results for the same.



$E_n(x) = (x+n) mod\ 26$

(Encryption Phase with shift n)

$D_n(x) = (x-n) mod\ 26$

(Decryption Phase with shift n)

```
#include <iostream>
#include <string>

using namespace std;

string encrypt(const string& text, int shift)
{
  string encrypted_text = "";
  for (int i =0; text[i] != '\0';i++)
        {
     encrypted_text += (text[i] - 'A' + shift) % 26 + 'A';
  }
  return encrypted_text;
}
```

```
string decrypt(const string& text, int shift)
{
    string decrypted_text = "";
    for (int i =0; text[i] != '\0';i++)
            {
        decrypted_text += (text[i] - 'A' - shift + 26) % 26 + 'A';
    }
    return decrypted_text;
}

int main() {
    string plain_text;
    int shift_value;

    cout << "Enter plain text (all uppercase): ";
    cin >> plain_text;

    cout << "Enter shift value (Enter 3 for Caesar cipher): ";
    cin >> shift_value;

    string encrypted = encrypt(plain_text, shift_value);
    cout << "Encrypted text: " << encrypted << endl;

    string decrypted = decrypt(encrypted, shift_value);
    cout << "Decrypted text: " << decrypted << endl;

    return 0;
}
```

16. Use key

a. 3, or

b. 5, or

c. 7

d. for message: please transfer one million dollars to my swiss bank account six two two four

17. Explain C, I, A, Authentication and Non-Repudiation. - **EXPT_2**
18. Explain the Interruption, Interception, Modification and Fabrication attack. Corelate the said attacks with C,I,A, Authentication and Non-Repudiation. - **EXPT_2**
19. Explain RSA algorithm in detail with example. - **EXPT_1**
20. Explain any two of following with example - **EXPT_1**

    i. Playfair Cipher

    ii. Hill Cipher

    iii. Vigenère cipher

    iv. One-Time Pad cipher

    v. Monoalphabetic cipher
21. Compare in Tabular fashion, Confidentiality, Integrity, Authentication and Non repudiation for following security techniques - **EXPT_1**

    a. Classical Encryption/Decryption

    b. Symmetric Encryption

    c. Asymmetric Encryption

    d. Hashing Technique

    e. MAC technique

    f. Digital Signature System

EXPT _4

22. Demonstrate installation and configuration of mobile Security app. Explain the different features and record the different working snapshots for the same.

23. What is WEP and WAP security techniques? Explain the details.

WEP (Wired Equivalent Privacy):

WEP was an early security protocol used to secure wireless networks.

It aimed to provide confidentiality and authentication similar to a wired network.

WEP uses a shared key mechanism for encryption, where all devices on the network share the same key.

However, WEP has significant security vulnerabilities, and it's now considered highly insecure and easily exploitable.

Attackers can easily crack WEP keys using readily available tools, exposing network traffic to interception and manipulation.

WPA (Wi-Fi Protected Access):

WPA is a successor to WEP and addresses many of its security flaws.

WPA improves security by using stronger encryption algorithms and implementing better key management techniques.

WPA supports various authentication methods, including Pre-Shared Key (PSK) and Extensible Authentication Protocol (EAP).

WPA also introduced Temporal Key Integrity Protocol (TKIP) and later Advanced Encryption Standard (AES) for encryption, providing stronger protection for wireless networks.

Overall, WPA offers better security than WEP and is widely recommended for securing Wi-Fi networks.

24. What are the different wireless components used for Wi-Fi, Bluetooth Communications?

Wi-Fi Components:

Wireless Router:

The central device that connects to your internet service provider's modem and creates a Wi-Fi network in your home or office.

It manages data traffic between devices on the network and connects them to the internet.

Wireless Access Point (WAP):

Similar to a router but used in larger networks or business environments to extend Wi-Fi coverage.

It connects to the router via a wired connection and broadcasts Wi-Fi signals for devices to connect to.

Wireless Network Interface Card (Wireless NIC or Wi-Fi Adapter):

Built into devices like laptops, smartphones, and tablets to enable wireless connectivity.

It communicates with the router or access point to send and receive data over the Wi-Fi network.

Wireless Range Extender:

Optional device used to increase the coverage area of a Wi-Fi network.

It picks up Wi-Fi signals from the router or access point and rebroadcasts them to areas with weak or no signal.

Bluetooth Components:

Bluetooth-enabled Devices:

Devices like smartphones, tablets, laptops, headphones, speakers, and smartwatches equipped with Bluetooth technology.

They can wirelessly connect to each other and exchange data or audio signals.

Bluetooth Transceiver:

Integrated into Bluetooth-enabled devices, the transceiver sends and receives Bluetooth signals.

It uses radio waves to establish connections and communicate with other Bluetooth devices in range.

Bluetooth Dongle:

External adapter plugged into a USB port on a computer or device to add Bluetooth functionality.

It allows devices without built-in Bluetooth capability to connect wirelessly to other Bluetooth devices.

Bluetooth Hub:

Optional device used to connect multiple Bluetooth devices together.

It acts as a central hub for managing Bluetooth connections and coordinating data exchange between devices.

25. Write details about ISM band frequencies, BT standards & Wi-Fi standards.

ISM Band Frequencies:

What is ISM Band: ISM stands for Industrial, Scientific, and Medical

Frequency Ranges: The most commonly used ISM bands include:

2.4 GHz: Widely used for Wi-Fi, Bluetooth, and other wireless communication technologies

5 GHz: Increasingly used for Wi-Fi due to its wider channels and reduced interference.

Regulatory Status: ISM bands are allocated by regulatory bodies. These bands are available for use without a license, but devices operating within ISM bands must comply with certain regulatory requirements to avoid interference with licensed users and other devices.

Bluetooth (BT) Standards:

Bluetooth Technology: Bluetooth is a wireless communication technology used for short-range data exchange between devices, such as smartphones, tablets, laptops, headphones, and IoT devices.

Bluetooth Standards:

Bluetooth 1.x: The initial versions of Bluetooth with limited data transfer rates and compatibility issues.

Bluetooth 2.0: Introduced Enhanced Data Rate (EDR) for faster data transfer and improved power efficiency.

Bluetooth 3.0: Introduced High-Speed Bluetooth with Enhanced Data Rate (HS-EDR) for faster file transfers.

Bluetooth 4.x: Introduced Low Energy (LE) technology for low-power IoT devices. Includes Bluetooth 4.0, 4.1, and 4.2 versions.

Bluetooth 5.x: The latest standard with significant improvements in range, data transfer speed, and compatibility with IoT devices. Includes Bluetooth 5.0, 5.1, and 5.2 versions.

Wi-Fi Standards:

Wi-Fi Technology: Wi-Fi is a wireless networking technology that allows devices to connect to a local area network (LAN) and access the internet wirelessly.

Wi-Fi Standards:

802.11b: Introduced in 1999, providing data rates up to 11 Mbps in the 2.4 GHz ISM band.

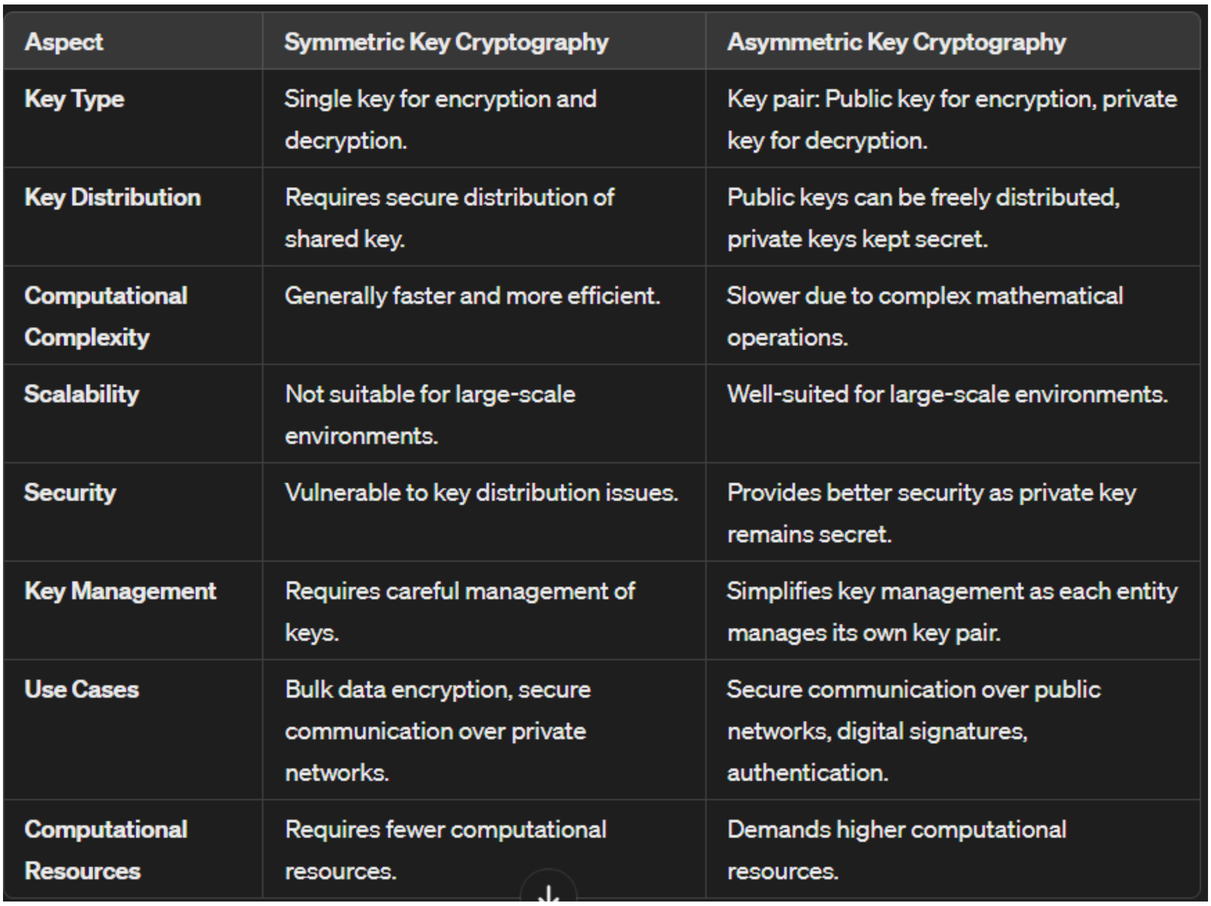
802.11a/g: Introduced around the same time, offering data rates up to 54 Mbps in the 5 GHz and 2.4 GHz bands, respectively.

802.11n: Introduced in 2009, providing higher data rates (up to 600 Mbps) and improved range using multiple antennas and channel bonding.

802.11ac: Introduced in 2013, offering even higher data rates (up to 1 Gbps) and improved performance in the 5 GHz band through wider channels and multi-user MIMO (MU-MIMO) technology.

802.11ax (Wi-Fi 6): The latest standard introduced in 2019, focusing on increasing network efficiency, capacity, and performance in dense environments. It supports data rates up to several Gbps and improves overall network performance.

26. Compare symmetric and asymmetric key cryptography.(Min 8 Points).



Aspect	Symmetric Key Cryptography	Asymmetric Key Cryptography
Key Type	Single key for encryption and decryption.	Key pair: Public key for encryption, private key for decryption.
Key Distribution	Requires secure distribution of shared key.	Public keys can be freely distributed, private keys kept secret.
Computational Complexity	Generally faster and more efficient.	Slower due to complex mathematical operations.
Scalability	Not suitable for large-scale environments.	Well-suited for large-scale environments.
Security	Vulnerable to key distribution issues.	Provides better security as private key remains secret.
Key Management	Requires careful management of keys.	Simplifies key management as each entity manages its own key pair.
Use Cases	Bulk data encryption, secure communication over private networks.	Secure communication over public networks, digital signatures, authentication.
Computational Resources	Requires fewer computational resources.	Demands higher computational resources.

27. Draw and explain how DES algorithm works in detail. - **EXPT_1**

28. Explain Transport and tunnel mode in IPSec.

Transport Mode:

Purpose: Transport mode in IPSec is used to secure communication between two hosts or devices on a network.

Functionality: In transport mode, only the payload (actual data) of the IP packet is encrypted and authenticated, leaving the IP header intact.

Use Case: Transport mode is commonly used for end-to-end communication between individual hosts or devices within a network.

Advantages:

Efficient for securing point-to-point communication.

Preserves original IP addresses of communicating hosts.

Tunnel Mode:

Purpose: Tunnel mode in IPSec is used to secure communication between two networks or between a host and a network.

Functionality: In tunnel mode, the entire original IP packet (including IP header and payload) is encapsulated within a new IP packet with its own IP header. This new packet is then encrypted and authenticated.

Use Case: Tunnel mode is commonly used for securing communication between networks, such as site-to-site VPNs or remote access VPNs.

Advantages:

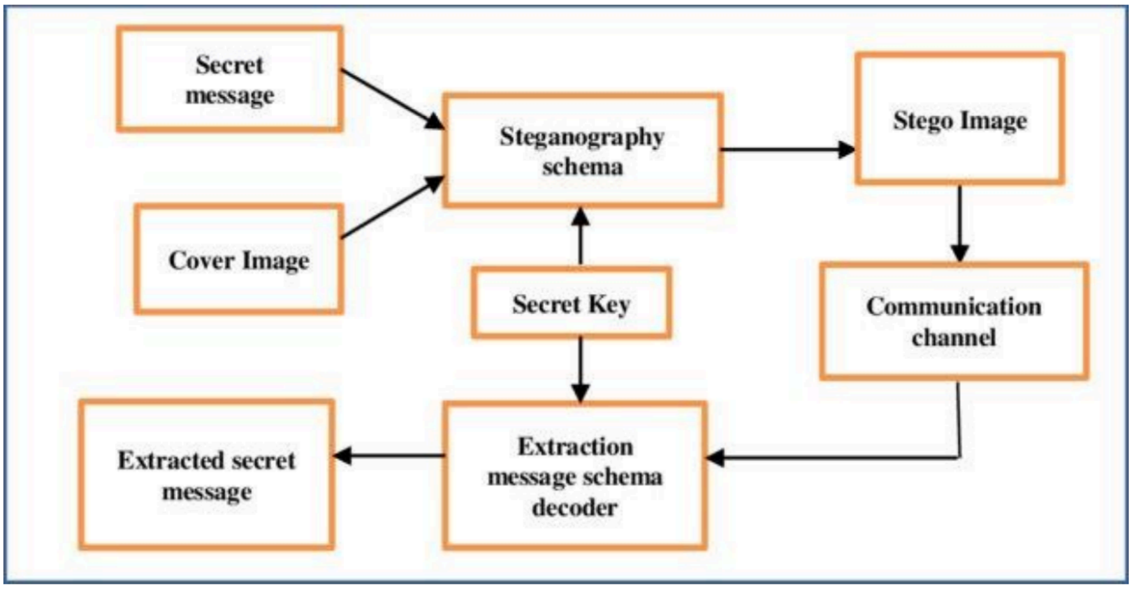
Provides end-to-end security for communication between networks.

Hides original IP addresses of communicating hosts.

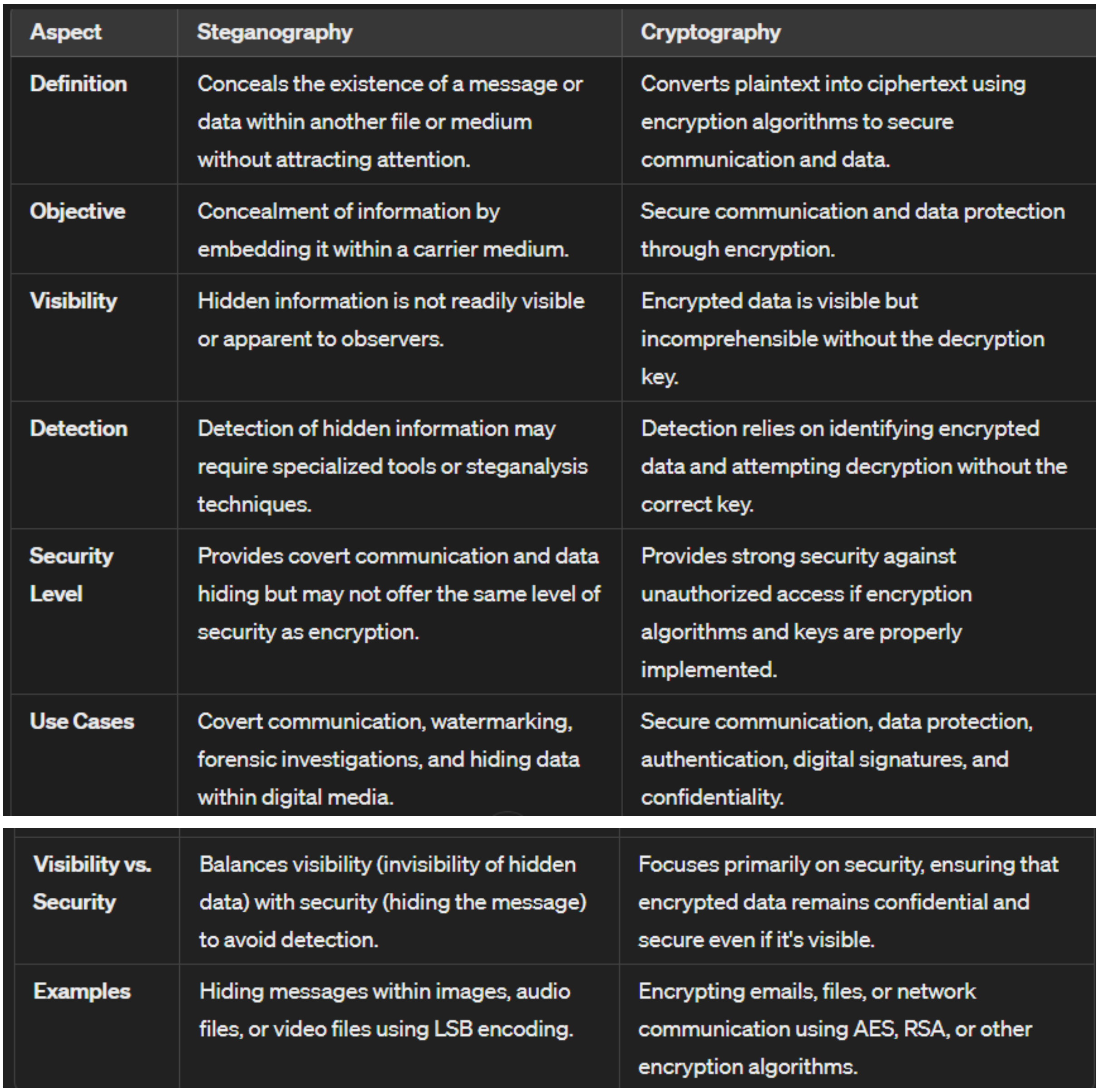
EXPT _5

29. Demonstrate installation and configuration of Steganography technique in view of network security. Explain the different features and record the different working snapshots for the same.

30. Draw and explain the block diagram of Steganography for Image as data.



31. Compare Steganography versus Cryptography.



Aspect	Steganography	Cryptography
Definition	Conceals the existence of a message or data within another file or medium without attracting attention.	Converts plaintext into ciphertext using encryption algorithms to secure communication and data.
Objective	Concealment of information by embedding it within a carrier medium.	Secure communication and data protection through encryption.
Visibility	Hidden information is not readily visible or apparent to observers.	Encrypted data is visible but incomprehensible without the decryption key.
Detection	Detection of hidden information may require specialized tools or steganalysis techniques.	Detection relies on identifying encrypted data and attempting decryption without the correct key.
Security Level	Provides covert communication and data hiding but may not offer the same level of security as encryption.	Provides strong security against unauthorized access if encryption algorithms and keys are properly implemented.
Use Cases	Covert communication, watermarking, forensic investigations, and hiding data within digital media.	Secure communication, data protection, authentication, digital signatures, and confidentiality.
Visibility vs. Security	Balances visibility (invisibility of hidden data) with security (hiding the message) to avoid detection.	Focuses primarily on security, ensuring that encrypted data remains confidential and secure even if it's visible.
Examples	Hiding messages within images, audio files, or video files using LSB encoding.	Encrypting emails, files, or network communication using AES, RSA, or other encryption algorithms.

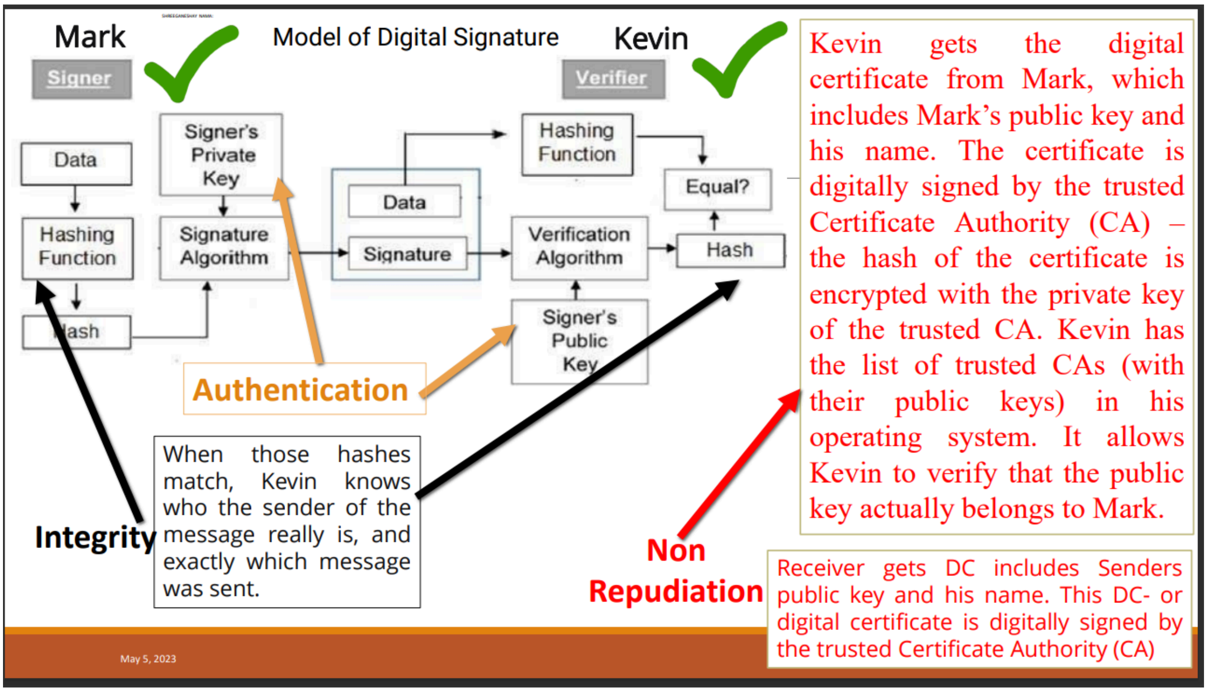
32. Draw and explain how DES algorithm works in detail. - **EXPT_1**

33. Explain RSA algorithm in detail with example. - **EXPT_1**

34. Compare symmetric and asymmetric key cryptography.(Min 8 Points). **EXPT_4**

35. Draw and explain the following block diagrams

    a. Digital Signature system.



When those hashes match, Kevin knows who the sender of the message really is, and exactly which message was sent.

Kevin gets the digital certificate from Mark, which includes Mark's public key and his name. The certificate is digitally signed by the trusted Certificate Authority (CA) – the hash of the certificate is encrypted with the private key of the trusted CA. Kevin has the list of trusted CAs (with their public keys) in his operating system. It allows Kevin to verify that the public key actually belongs to Mark.

Receiver gets DC includes Senders public key and his name. This DC- or digital certificate is digitally signed by the trusted Certificate Authority (CA)

b. End to End Email Communication system with Hashing, Digital signature and Digital Envelope processing blocks.

Hashing:

When Alice composes an email, her email client generates a hash value of the message content using a cryptographic hash function (e.g., SHA-256).

The hash value uniquely represents the message content and serves as a digital fingerprint.

Digital Signature:

Alice uses her private key to create a digital signature for the hashed message. The digital signature is created by encrypting the hash value using her private key.

This process ensures that only Alice, with access to her private key, can produce a valid digital signature for the email.

Digital Envelope:

To encrypt the email content and attachments, Alice generates a random symmetric encryption key (session key) specifically for this email.

Alice encrypts the email content and attachments using this session key with a symmetric encryption algorithm (e.g., AES).

Next, Alice encrypts the session key itself using Bob's public key. This step ensures that only Bob, with access to his private key, can decrypt the session key.

Alice sends the encrypted email content, encrypted session key, and digital signature to Bob.

Receiving and Verifying:

When Bob receives the email, his email client decrypts the session key using his private key.

With the decrypted session key, Bob decrypts the email content and attachments.

Bob's email client calculates the hash value of the received message content.

Bob verifies the digital signature by decrypting it using Alice's public key and comparing the decrypted hash value with the calculated hash value. If they match, it ensures that the email content has not been altered since it was signed by Alice.

EXPT _6

36. Install and configure firewall for Host security. Explain the different features and record the different working snapshots for the same.
37. Draw and explain following. A)Packet filtering firewall B)Application Layer Firewall C) Circuit Level Gateway firewall 38. Compare Firewall Versus Antivirus.
39. Compare IDS and IPS in detail.
40. Draw and explain how DES algorithm works in detail. - **EXPT_1**
41. Explain RSA algorithm in detail with example. - **EXPT_1**
42. Draw and explain the following block diagrams - **EXPT_5**
    a. Digital Signature system.
    b. End to End Email Communication system with Hashing, Digital signature and Digital Envelope processing blocks.

EXPT _7

43. Demonstrate process to ensure Security of web browser (Google Chrome) with respect to A) Cookies settings B) Website Blocking C) Phrase/Word blocking. Explain the different features and record the different working snapshots for the same.

44. What are different types of cookies?

Types of Cookies:

Session Cookies: Temporary, erased when the browser is closed.

Persistent Cookies: Remain on the device after browser closure.

Secure Cookies: Transmitted over HTTPS connections only.

HttpOnly Cookies: Inaccessible to client-side JavaScript.

Third-Party Cookies: Set by domains other than the current site.

45. What are advantages and drawback of cookies?

Advantages:

Personalization

Convenience

Tracking

Authentication

Drawbacks:

Privacy Concerns

Security Risks

Dependency

Legal Compliance

46. Explain “Session Hijacking” by misusing cookies information.

Session hijacking involves stealing a user's session cookie to gain unauthorized access.

Attackers intercept the cookie through methods like packet sniffing or XSS attacks.

With the stolen cookie, attackers can impersonate the user and access their active session.

This allows attackers to perform actions on the website as if they were the legitimate user.

Measures to mitigate session hijacking include using HTTPS connections for encryption.

Implementing HTTPOnly and Secure flags for cookies to prevent unauthorized access.

Enforcing session expiration policies to limit the lifespan of session identifiers.

Employing robust session management practices to detect and prevent suspicious activity.

Multi-factor authentication (MFA) adds an extra layer of security beyond session cookies.

Session hijacking poses a significant security risk and requires proactive measures to protect user sessions.

47. Draw and explain the following block diagrams - **EXPT_5**

a. Digital Signature system.

b. End to End Email Communication system with Hashing, Digital signature and Digital Envelope processing blocks.

Hashing:

When Alice composes an email, her email client generates a hash value of the message content using a cryptographic hash function (e.g., SHA-256).

The hash value uniquely represents the message content and serves as a digital fingerprint.

Digital Signature:

Alice uses her private key to create a digital signature for the hashed message. The digital signature is created by encrypting the hash value using her private key.

This process ensures that only Alice, with access to her private key, can produce a valid digital signature for the email.

Digital Envelope:

To encrypt the email content and attachments, Alice generates a random symmetric encryption key (session key) specifically for this email.

Alice encrypts the email content and attachments using this session key with a symmetric encryption algorithm (e.g., AES).

Next, Alice encrypts the session key itself using Bob's public key. This step ensures that only Bob, with access to his private key, can decrypt the session key.

Alice sends the encrypted email content, encrypted session key, and digital signature to Bob.

Receiving and Verifying:

When Bob receives the email, his email client decrypts the session key using his private key.

With the decrypted session key, Bob decrypts the email content and attachments.

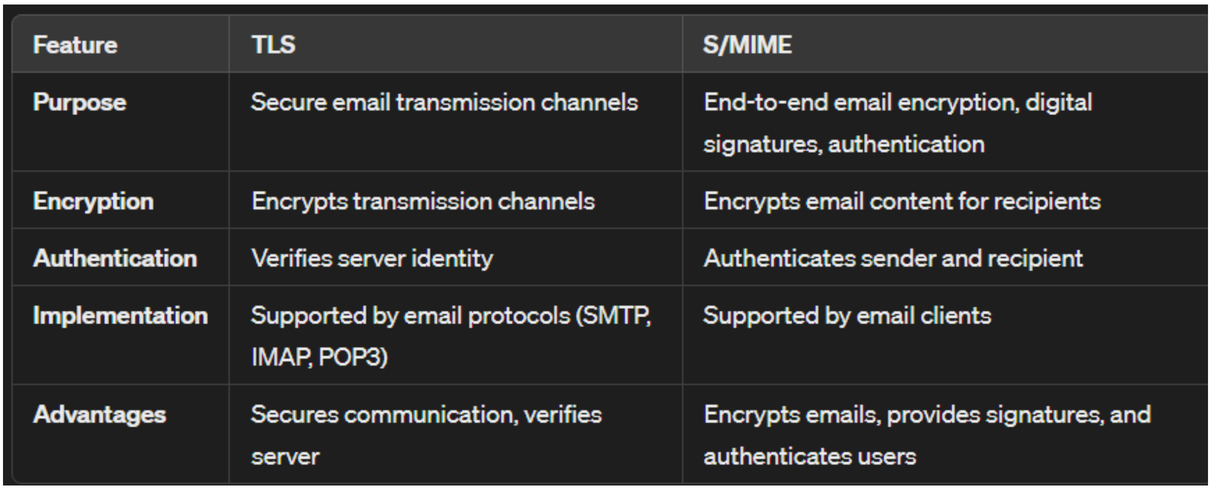
Bob's email client calculates the hash value of the received message content.

Bob verifies the digital signature by decrypting it using Alice's public key and comparing the decrypted hash value with the calculated hash value. If they match, it ensures that the email content has not been altered since it was signed by Alice.

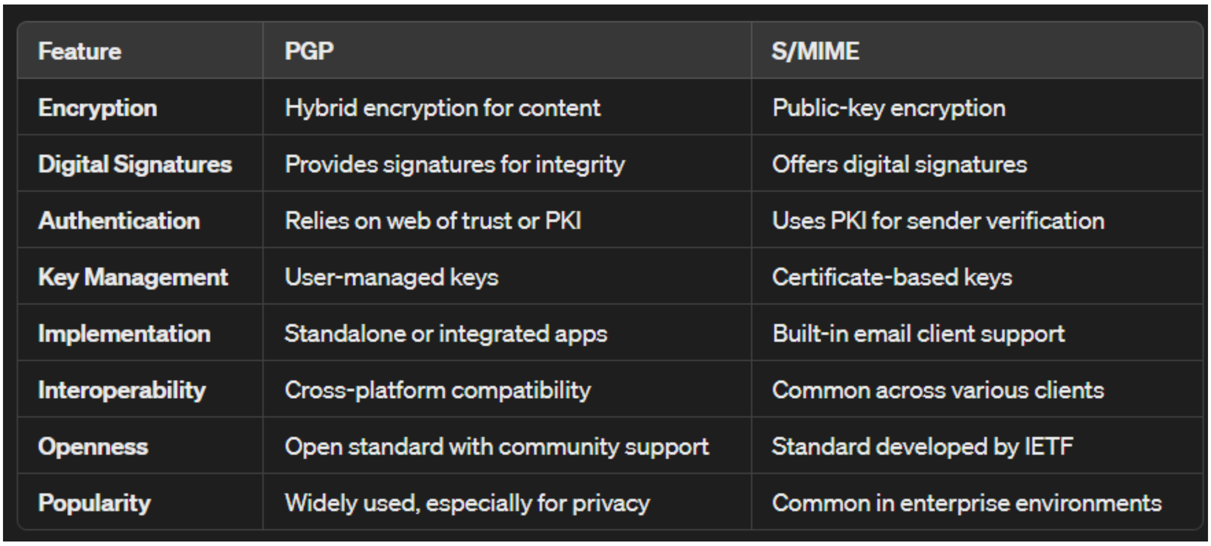
c.

48. Explain RSA algorithm in detail with example.   - **EXPT_1**

49. Explain TLS and S/MIME used in Email Security. Compare PGP Vs S/MIME.



Feature	TLS	S/MIME
Purpose	Secure email transmission channels	End-to-end email encryption, digital signatures, authentication
Encryption	Encrypts transmission channels	Encrypts email content for recipients
Authentication	Verifies server identity	Authenticates sender and recipient
Implementation	Supported by email protocols (SMTP, IMAP, POP3)	Supported by email clients
Advantages	Secures communication, verifies server	Encrypts emails, provides signatures, and authenticates users



Feature	PGP	S/MIME
Encryption	Hybrid encryption for content	Public-key encryption
Digital Signatures	Provides signatures for integrity	Offers digital signatures
Authentication	Relies on web of trust or PKI	Uses PKI for sender verification
Key Management	User-managed keys	Certificate-based keys
Implementation	Standalone or integrated apps	Built-in email client support
Interoperability	Cross-platform compatibility	Common across various clients
Openness	Open standard with community support	Standard developed by IETF
Popularity	Widely used, especially for privacy	Common in enterprise environments

EXPT _8

50. Implement Hash function technique for secured network using Suitable Hashing tool and validate using available online tools/Website tools. Explain the different features and record the different working snapshots for the same.
51. Explain following applications of Hash functions in detail

    a. **Protection to password storage**

    Hash functions convert passwords into irreversible hash values, safeguarding user confidentiality.

    Salting enhances security by adding random values to passwords before hashing.

    b. **Data Integrity check**

    Hash functions generate unique fingerprints of data, serving as checksums for verification.

Senders compute hash values of transmitted data, including them with the data for integrity verification.

Recipients independently compute hash values of received data, comparing them with transmitted values.

Any changes to data result in different hash values, indicating potential tampering.

Used in network protocols, file transfers, backups, and digital signatures for ensuring data authenticity and integrity.

52. Compare in Tabular fashion, Confidentiality, Integrity, Authentication and Non-repudiation for following security techniques - **EXPT_1**
    a. Classical Encryption/Decryption
    b. Symmetric Encryption
    c. Asymmetric Encryption
    d. Hashing Technique
    e. MAC technique
    f. Digital Signature System
53. What is Hash function? Explain How it works briefly? List the different applications of SHA2.

Hash Function:

Converts input data into fixed-size hash values.

Ensures data integrity and security.

Operates through mathematical algorithms.

Produces unique hash values for each input.

Characteristics include determinism and irreversibility.

SHA2 (Secure Hash Algorithm 2):

Family of cryptographic hash functions.

Includes SHA-224, SHA-256, SHA-384, SHA-512 variants.

Widely used for various cryptographic applications.

Provides data integrity verification and digital signatures.

Ensures password storage security and blockchain integrity.

Applications of SHA2:

Data integrity verification in networks.

Digital signatures for authentication.

Password storage protection.

Blockchain technology for hashing blocks.

SSL/TLS certificates for web security.

File integrity checking and message authentication codes.

Ensures confidentiality, integrity, and authenticity of digital assets.

54. Explain Transport and tunnel mode in IPSec. **- EXPT_4**
55. Explain TLS and S/MIME used in Email Security. Compare PGP Vs S/MIME. - **EXPT_7**
56. Explain RSA algorithm in detail with example. - **EXPT_1**

EXPT_9

57. Simulate Diffie-Hellman secure key exchange protocol using Vlabs simulation tool. Explain the different features of above protocol and record the different working snapshots for the same.
58. Draw and explain how DES algorithm works in detail. - **EXPT_1**
59. Explain RSA algorithm in detail with example. - **EXPT_1**
60. What is Hash function? Explain How it works briefly? List the different applications of SHA2. - **EXPT_8**
61. Compare symmetric and asymmetric key cryptography.(Min 8 Points). **EXPT_4**
62. Explain any two of following with example - **EXPT_1**
    1. Playfair Cipher
    2. Hill Cipher
    3. Vigenère cipher
    4. One-Time Pad cipher
    5. Monoalphabetic cipher
63. Write short note on the following
    - a. Transport and tunnel mode in IPSec. **- EXPT_4**
    - b. S/MIME for Email security. **- EXPT_7–**
    - c. TLS explanation with Suitable example **-EXPT_7**

EXPT_10

64. Simulate Vernam Cipher for encryption and decryption using Vlabs simulation tool. Explain the different features of above technique with suitable example and record the different working snapshots for the same.
65. Draw and explain how DES algorithm works in detail. - **EXPT_1**

66. Explain RSA algorithm in detail with example. - **EXPT_1**

67. Compare symmetric and asymmetric key cryptography.(Min 8 Points). **EXPT_4**

68. Explain any two of following with example - **EXPT_1**

    i. Playfair Cipher

    ii. Hill Cipher

    iii. Vigenère cipher iv.

    One-Time Pad cipher

    v. Monoalphabetic cipher

69. Write short note on the following **- EXPT_9**

    a. Transport and tunnel mode in IPSec.

    b. S/MIME for Email security.

    c. TLS explanation with Suitable example

70. Explain AH and ESP working in IPSec.

Authentication Header (AH):

Provides authentication and integrity protection for IP packets.

Adds a header containing a cryptographic hash of the packet's contents.

Hash covers the entire packet, including IP header, AH header, and payload.

Recipient recalculates the hash upon receipt for verification.

Ensures packet integrity and authenticates the sender.

Does not provide confidentiality as the payload remains unencrypted.

Encapsulating Security Payload (ESP):

Offers confidentiality, integrity, and optional authentication for IP packets.

Encrypts the payload using symmetric encryption algorithms like AES.

Adds a new header and optional trailer to encapsulate the payload.

Provides confidentiality through encryption and integrity protection via cryptographic hashes.

Can include optional authentication using algorithms like HMAC.

Operates in different modes, including transport and tunnel modes.

Ensures security services for IP communications, including flexibility in deployment.