Content from different reference book suggested by Prof. Prakash Sir and mentioned in CNS Syllabus file, Presentations, Notes provided in 2023

CNS UT1 and UT2 2023 Notes

Q. Encrypt the following using playfair cipher using the keyword MONARCHY. "SWARAJ IS MY BIRTH RIGHT". Use X as blank Space in points, with diagram and example
ANS.

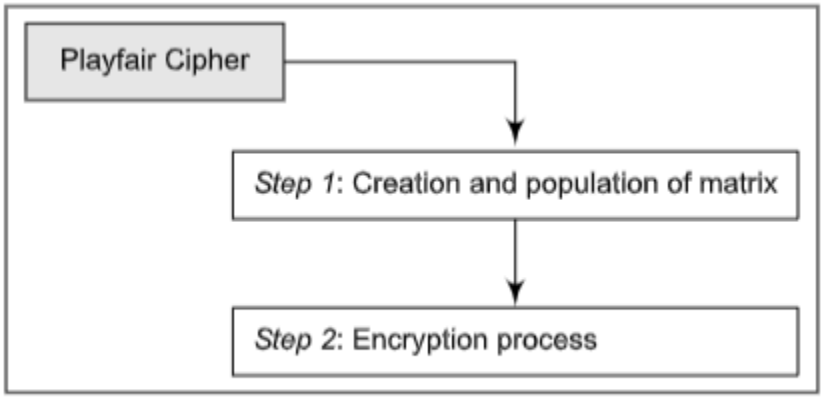


Fig. 2.16 Playfair Cipher steps

Encryption using Playfair Cipher with Keyword MONARCHY

To encrypt the given message "SWARAJ IS MY BIRTH RIGHT" using the Playfair cipher with the keyword MONARCHY, we first need to create the keyword matrix. The keyword matrix for MONARCHY is as follows:

M O N A R C H Y B D E F G I K L P Q S T U V W X Z

Next, we break down the plain text into pairs of alphabets, ignoring any spaces or punctuation marks. The pairs for the given message are: SW, AR, AJ, IS, MY, BI, RT.

Using the Playfair cipher based on the keyword matrix, we find the corresponding cipher text for each pair. The resulting cipher text would be: ZC, ZR, ZJ, YS, ZB, YD, ZT.

Therefore, the encrypted message using the Playfair cipher with the keyword MONARCHY for the given plain text "SWARAJ IS MY BIRTH RIGHT" would be "ZCZRZJYSZBYDZT".

Q. Convert "MEET ME" using hill cipher with key matrix in points, with diagram and example
3 5 1
6 7 2
1 4 0
ANS.

Encryption using Hill Cipher with Key Matrix

To encrypt the plain text "MEET ME" using the Hill cipher with the given key matrix, we need to break the plain text into pairs of alphabets and perform matrix multiplication with the key matrix.

First, we break the plain text into pairs: "ME", "ET", and "ME".

Next, we assign numerical values to the alphabets based on their positions in the alphabet. For example, "M" is 12, "E" is 4, and "T" is 20.

Then, we create matrices for each pair of alphabets:

For "ME": | 12 | 4 | |----|---|

For "ET": | 4 | 20 | |---|----|

For "ME": | 12 | 4 | |----|---|

Finally, we perform matrix multiplication with the key matrix:

For "ME": | 12 | 4 | | 3 | 5 | 1 | | 92 | 68 | 16 | |----|---| x |---|---|---| = |----|----|----| | 12 | 4 | | 6 | 7 | 2 | | 92 | 68 | 16 |

For "ET": | 4 | 20 | | 3 | 5 | 1 | | 32 | 140 | 24 | |---|----| x |---|---|---| = |----|-----|----| | 4 | 20 | | 6 | 7 | 2 | | 32 | 140 | 24 |

For "ME": | 12 | 4 | | 3 | 5 | 1 | | 92 | 68 | 16 | |----|---| x |---|---|---| = |----|----|----| | 12 | 4 | | 6 | 7 | 2 | | 92 | 68 | 16 |

The resulting cipher text is: "92 68 16 32 140 24 92 68 16".

Please note that the above encryption process is based on the given key matrix and the plain text "MEET ME".

Q. With respect to Information Security context explain four broad based categories of attacks in points, with diagram and example

ANS.

Four Broad Categories of Attacks in Information Security

1. Criminal Attacks: These attacks aim to maximize financial gain by targeting computer systems. Examples include fraud, scams, destruction, identity theft, and intellectual property theft. Criminal attackers are motivated by monetary benefits and may use various techniques to exploit vulnerabilities in systems.
2. Publicity Attacks: Publicity attacks are carried out to gain attention and recognition. Attackers, such as students or employees, seek publicity by adopting novel approaches to attack computer systems. One form of publicity attack is defacing web pages, as seen in the famous attacks on the US Department of Justice's website and the New York Times home page.
3. Legal Attacks: This unique form of attack aims to create doubt about the security of a computer system in a legal context. Attackers try to exploit the judge and jury's lack of technological knowledge to convince them that there are inherent weaknesses in the system. For example, an attacker may sue a bank for an online transaction they claim they never wanted to perform.
4. Passive and Active Attacks: Attacks can be classified as passive or active. Passive attacks involve eavesdropping or monitoring data transmission without modifying the contents. Examples include release of message contents and traffic analysis. Active attacks, on the other hand, involve modifying the contents of a message or disrupting the availability of a resource. Examples include alteration of messages, denial of service (DOS) attacks, and application-level and network-level attacks.

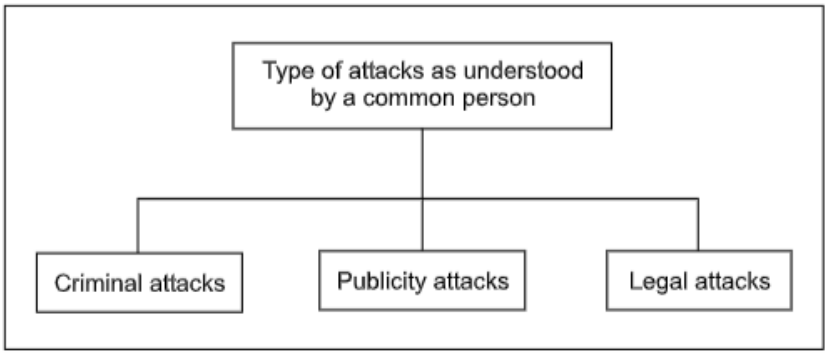


Fig. 1.9 Classification of attacks as understood in general terms

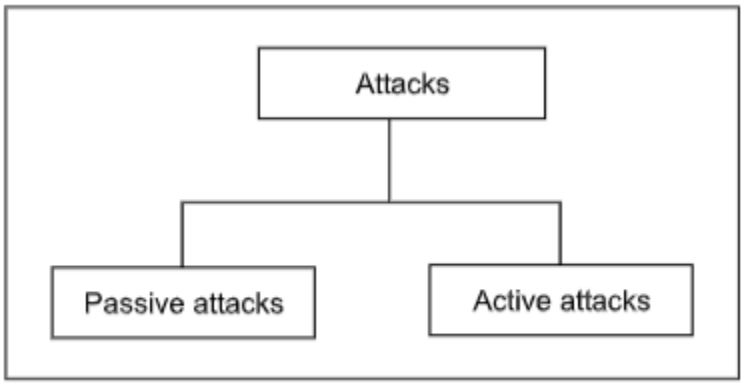


Fig. 1.10 Types of attacks

These categories of attacks demonstrate the diverse motivations and techniques employed by attackers in the field of information security. It is crucial for organizations and individuals to be aware of these attack types and implement appropriate security measures to protect their systems and data.

Q. Give difference between substitution and transposition cipher technique in points, with diagram and example
ANS.

Substitution Cipher Technique:

- In substitution cipher technique, the characters of a plain-text message are replaced by other characters, numbers, or symbols.
- It involves replacing each alphabet in the plain-text message with a different alphabet, number, or symbol.
- The Caesar cipher is an example of a substitution cipher, where each alphabet in the message is replaced by an alphabet three places down the line.
- Substitution ciphers are relatively easy to break, as the patterns in the language can be exploited.

Transposition Cipher Technique:

- In transposition cipher technique, the plain-text message is rearranged or permuted to form the cipher text.
- It does not involve replacing characters with other characters, but rather rearranging the order of the characters.
- The rail-fence technique and simple columnar transposition technique are examples of transposition ciphers.
- Transposition ciphers are also relatively easy to break, as the patterns in the language can still be exploited.

Diagram and Example:

Substitution Cipher Technique: Plain-text message: "HELLO" Cipher-text message: "URYYB"

Transposition Cipher Technique: Plain-text message: "HELLO" Cipher-text message: "HLELO"

In the substitution cipher, each alphabet is replaced with a different alphabet, while in the transposition cipher, the order of the alphabets is rearranged.

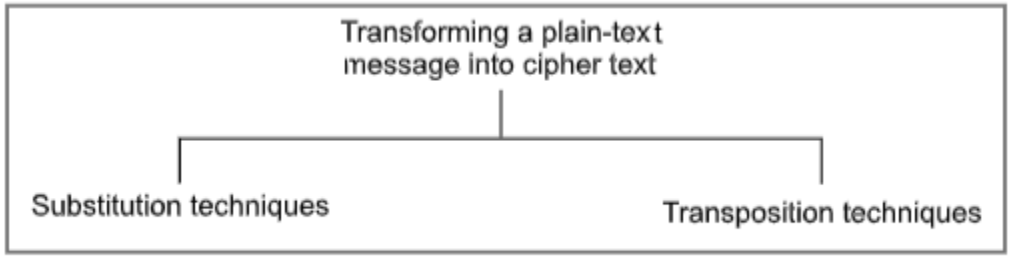


Fig. 2.9 Techniques for transforming plain text to cipher text

Q. Perform encryption and decryption process using mathematical modulor division method(viginere cipher) plain text:WOMENS UNIVERSITY Encryption key:UMITSNDT in points, with diagram and example
ANS.

Encryption Process using Vigenère Cipher

To perform encryption using the Vigenère cipher with the given plain text "WOMENS UNIVERSITY" and encryption key "UMITSNDT", we follow these steps:

1. Convert the plain text and encryption key into numerical values using a suitable mapping scheme. For example, we can use the ASCII values of the characters.
2. Repeat the encryption key to match the length of the plain text. In this case, the encryption key "UMITSNDT" can be repeated as "UMITSNDTUMI".
3. Perform the encryption by adding the numerical values of the plain text and the corresponding key values modulo 26. The result gives the encrypted text.

For example, let's consider the first letter "W" and the corresponding key letter "U". The numerical value of "W" is 87, and the numerical value of "U" is 85. Adding them modulo 26 gives us (87 + 85) % 26 = 20, which corresponds to the letter "U" in the encrypted text.

Repeat this process for each letter in the plain text and the corresponding key letters to obtain the complete encrypted text.

Decryption Process using Vigenère Cipher

To perform decryption using the Vigenère cipher, we follow similar steps as encryption but with a slight modification.

1. Convert the encrypted text and decryption key into numerical values using the same mapping scheme used for encryption.
2. Repeat the decryption key to match the length of the encrypted text.
3. Perform the decryption by subtracting the numerical values of the encrypted text and the corresponding key values modulo 26. The result gives the decrypted text.

For example, let's consider the first letter "U" in the encrypted text and the corresponding key letter "U". The numerical value of "U" is 20, and the numerical value of "U" is 85. Subtracting them modulo 26 gives us (20 - 85) % 26 = 11, which corresponds to the letter "L" in the decrypted text.

Repeat this process for each letter in the encrypted text and the corresponding key letters to obtain the complete decrypted text.

Example:

Plain text: WOMENS UNIVERSITY Encryption key: UMITSNDT

Numerical values: W(87) O(79) M(77) E(69) N(78) S(83) U(85) N(78) I(73) V(86) E(69) R(82) S(83) I(73) T(84) Y(89)

Repeated key: U(85) M(77) I(73) T(84) S(83) N(78) D(68) T(84) U(85) M(77) I(73) T(84) S(83) N(78) D(68) T(84)

Encrypted text: U(20) L(11) P(15) V(22) T(19) B(1) X(23) T(19) V(22) P(15) X(23) T(19) B(1) X(23) T(19) V(22)

Decrypted text: W(22) O(15) M(12) E(4) N(13) S(18) U(20) N(13) I(8) V(21) E(4) R(17) S(18) I(8) T(19) Y(24)

Therefore, the encrypted text for the given plain text "WOMENS UNIVERSITY" using the encryption key "UMITSNDT" is "ULPVTBXTVPXTV", and the decrypted text for the encrypted text using the same key is "WOMENSUNIVERSITY".

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*	*l*	*m*	*n*	*o*	*p*	*q*	*r*	*s*	*t*	*u*	*v*	*w*	*x*	*y*	*z*
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Fig. 2.15 Vigenére tableau

Q. State difference between threats and attacks, give brief information about program system threats in points, with diagram and example
ANS.

Threats vs Attacks:

Threats refer to potential risks or vulnerabilities that can harm a computer system or network. They can be accidental or intentional, and they can come from various sources such as viruses, hackers, or natural disasters. On the other hand, attacks are deliberate actions taken by an attacker to exploit vulnerabilities and cause harm to a computer system or network.

Program System Threats:

1. Viruses: Viruses are malicious programs that attach themselves to legitimate programs and can cause damage to computer systems. They can spread through email attachments or infected files.
2. Denial of Service (DoS) Attacks: These attacks aim to disrupt the normal functioning of a computer system or network by overwhelming it with excessive traffic or requests, making it inaccessible to legitimate users.
3. Alteration of Messages: This involves changing the content of a message, such as modifying the beneficiary or the amount in a transaction. It can lead to financial loss or manipulation of information.
4. Identity Theft: Attackers can steal personal information, such as passwords or credit card details, to impersonate someone else and carry out fraudulent activities.
5. Intellectual Property Theft: This involves stealing trade secrets, databases, software, or other digital assets belonging to organizations or individuals.

Diagram:

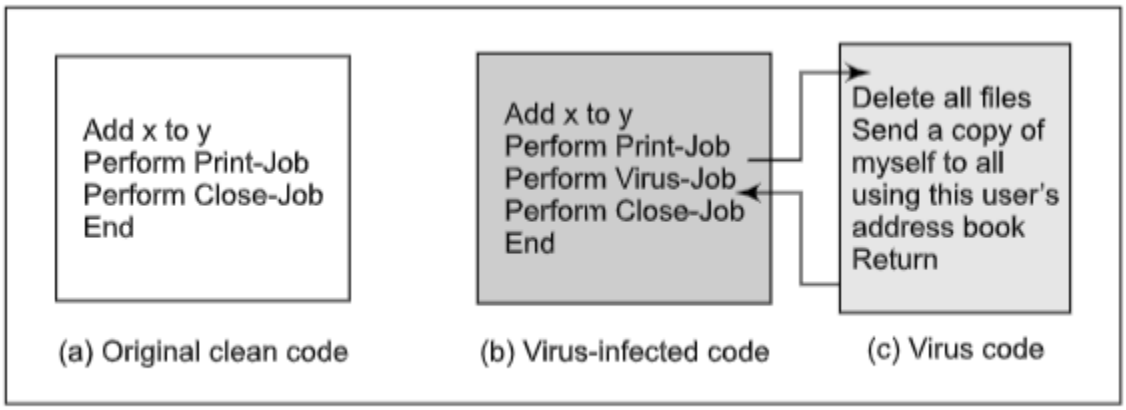


Fig. 1.14 Virus

Example:

An example of a program system threat is a virus attack. A virus is a piece of code that attaches itself to legitimate programs and can cause damage to computer systems. For instance, a user may unknowingly download an infected file, and when they run the program, the virus gets activated and starts infecting other files or spreading to other computers on the network. This can lead to data loss, system crashes, or unauthorized access to sensitive information.

Q. Give the difference between public key and private key cryptosystem in points, with diagram and example **ANS.**

Public Key Cryptosystem:

- In a public key cryptosystem, each user has a pair of keys: a public key and a private key.
- The public key is used for encryption, and it is freely available to anyone who wants to send a message to the user.

- The private key is kept secret and is used for decryption.
- The encryption process involves using the recipient's public key to encrypt the message, which can only be decrypted using the corresponding private key.
- This ensures that only the intended recipient can decrypt and read the message.

Private Key Cryptosystem:

- In a private key cryptosystem, there is only one key that is used for both encryption and decryption.
- This key is kept secret and is known only to the user.
- The encryption process involves using the private key to encrypt the message, and the same key is used to decrypt the message.
- Since the private key is the same for both encryption and decryption, it is crucial to keep it secure and prevent unauthorized access.

Diagram:

```
Public Key Cryptosystem:
Sender's Public Key --> Encryption --> Receiver's Private Key --> Decryption

Private Key Cryptosystem:
Sender's Private Key --> Encryption & Decryption --> Receiver's Private Key
```

Example:

Public Key Cryptosystem:

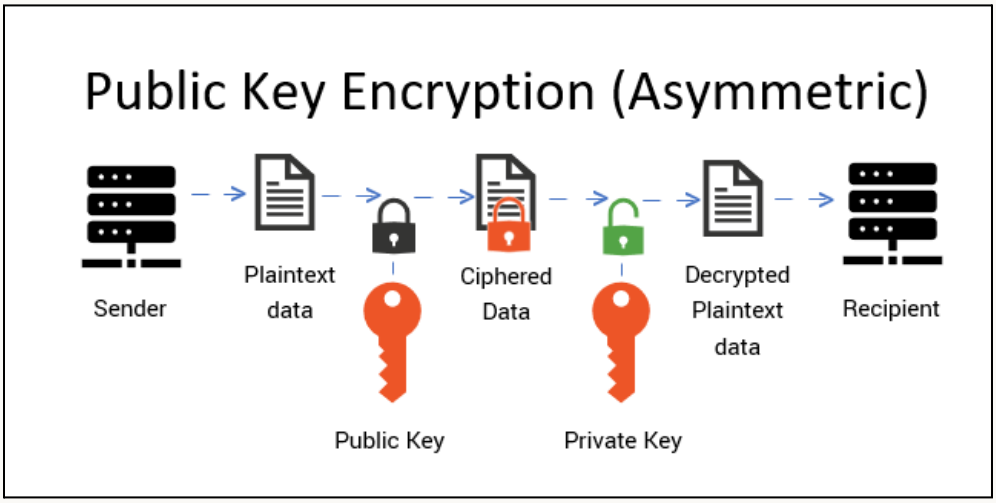
- Alice wants to send a secure message to Bob.
- Bob generates a pair of keys: a public key and a private key.
- Bob shares his public key with Alice.
- Alice uses Bob's public key to encrypt the message and sends it to Bob.
- Bob uses his private key to decrypt the message and read it.

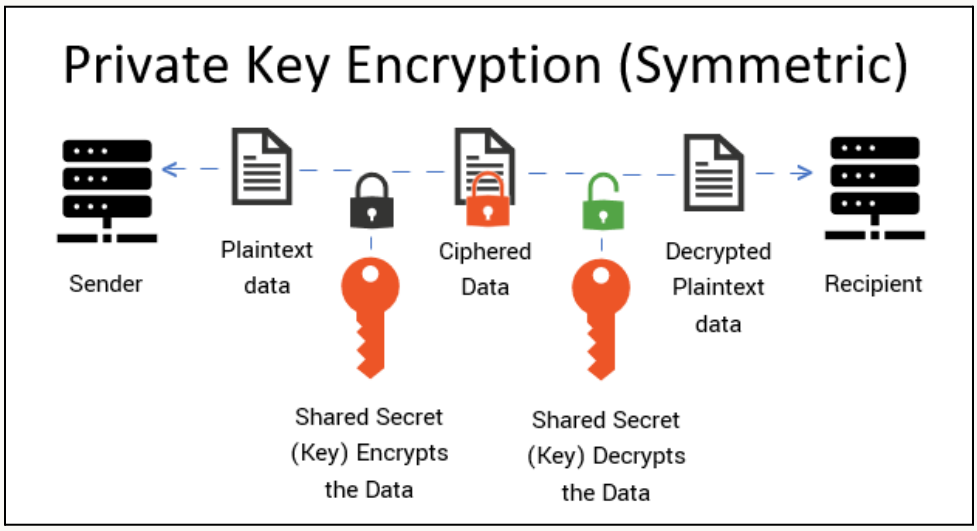
Private Key Cryptosystem:

- Alice wants to send a secure message to Bob.
- Bob generates a private key.
- Bob shares the private key with Alice.
- Alice uses the private key to encrypt the message and sends it to Bob.
- Bob uses the same private key to decrypt the message and read it.

Key details	*A* should know	*B* should know
A's private key	Yes	No
A's public key	Yes	Yes
B's private key	No	Yes
B's public key	Yes	Yes

Fig. 4.1 Matrix of private and public Keys





Q. Explain in detail about the security services(PGP or S/MIME) for email in points, with diagram and example
ANS.

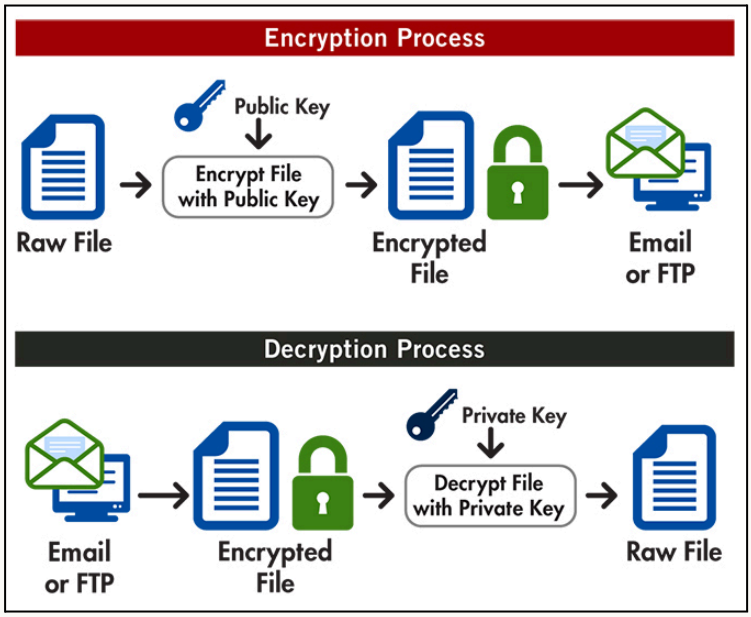
Security Services for Email: PGP and S/MIME

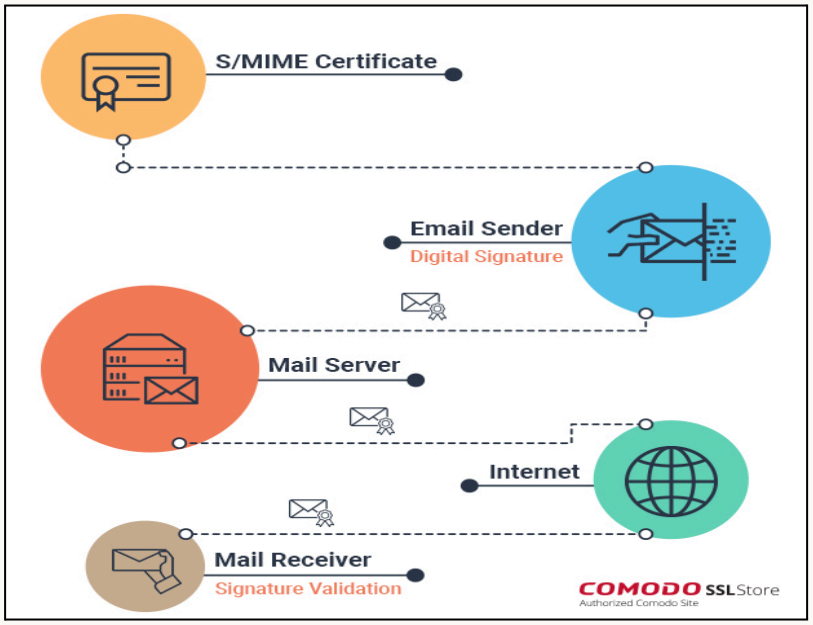
Pretty Good Privacy (PGP)

- PGP is a widely used email security protocol that offers encryption, digital signatures, and message integrity.
- It supports digital certificates or key rings to establish trust between users.
- PGP provides mechanisms like introducer trust, certificate trust, and key legitimacy to create trust relationships.
- The working of PGP involves steps like digital signature, compression, encryption, digital enveloping, and Base-64 encoding.
- PGP allows for different security options, including signature only, signature and Base-64 encoding, and signature, encryption, enveloping, and Base-64 encoding.

Secure MIME (S/MIME)

- S/MIME is another email security protocol that adds security to the MIME protocol, allowing non-text data to be sent via email.
- S/MIME secures MIME contents through encryption, message digests, and digital signatures.
- The output of S/MIME is a PKCS object, which is treated as a message content and wrapped inside MIME with appropriate headers.
- S/MIME supports digital signature, encryption, or both for email messages.
- It provides guidelines for cryptographic algorithms and supports various algorithms for secure communication.





Example:

- Let's say Alice wants to send a confidential email to Bob using PGP. She would first create a digital signature of the email using her private key. Then, she would compress the email and the digital signature together to reduce the message size. Next, she encrypts the compressed message using Bob's public key. Finally, she encodes the encrypted message in Base-64 format and sends it to Bob. Upon receiving the email, Bob would perform the reverse steps to retrieve the original plain-text message.

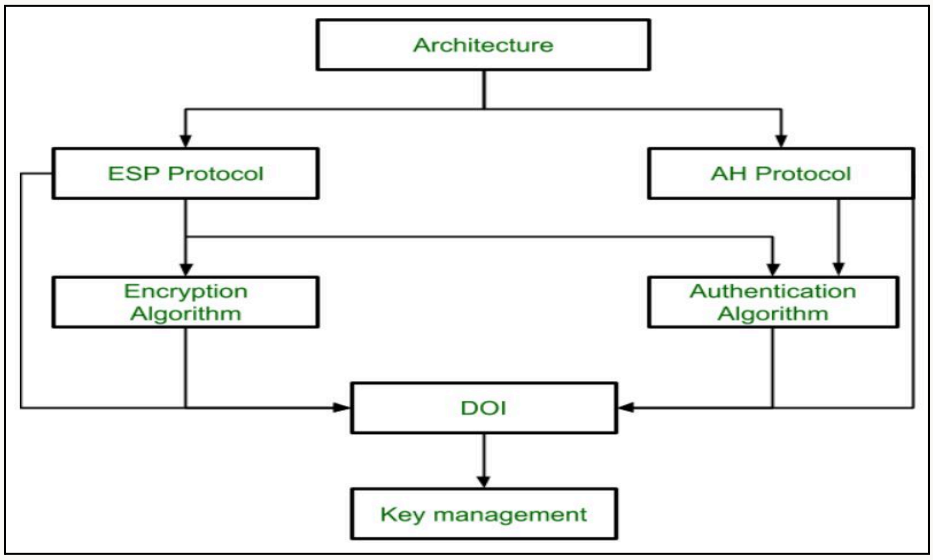
Note: The diagram provided in the document illustrates the security features offered by PGP and S/MIME, showing the encryption, digital signature, and other security mechanisms involved in securing email communication.

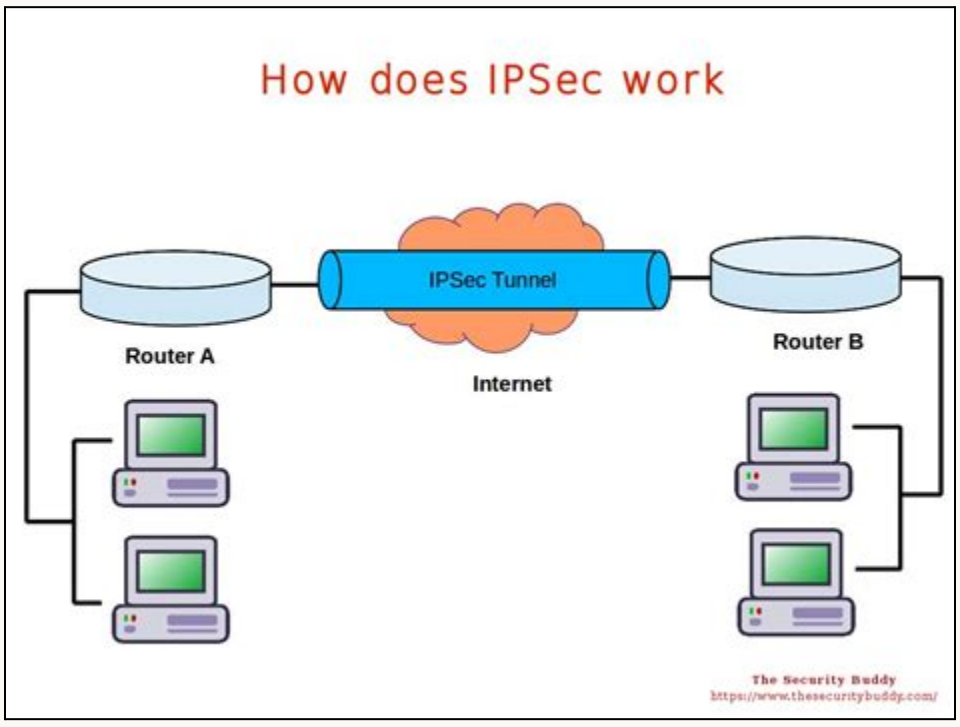
Q. Discuss IPsec with its architecture in points, with diagram and example
ANS.
IPsec (Internet Protocol Security) is a protocol suite that provides security services for IP packets. It operates at the network layer and offers authentication, integrity, and confidentiality services. The architecture of IPsec consists of several components:

1. Security Associations (SA): SAs are agreements between communicating parties that define the parameters for secure communication. This includes the IPSec protocol version, mode of operation (transport or tunnel), cryptographic algorithms, keys, and key lifetimes.

2. Authentication Header (AH): AH provides authentication, integrity, and optional anti-replay services. It adds a header to the IP packet, which includes a hash of the packet contents to ensure its integrity.
3. Encapsulating Security Payload (ESP): ESP provides data confidentiality by encrypting the IP packet payload. It also offers authentication and integrity services. ESP encapsulates the original IP packet and adds a new header and trailer for encryption and authentication.
4. Internet Key Exchange (IKE): IKE is responsible for negotiating and establishing SAs between communicating parties. It uses the ISAKMP/Oakley protocol for key management, which includes key agreement and distribution.
5. Tunnel Mode: IPsec can be implemented in tunnel mode, where the entire IP datagram, including the original header, is encrypted and a new IP header is added. This is useful for creating virtual tunnels between routers or connecting networks securely.
6. Transport Mode: IPsec can also be implemented in transport mode, where only the IP packet payload is encrypted, leaving the original IP header intact. This is suitable for securing communication between hosts.

Diagram





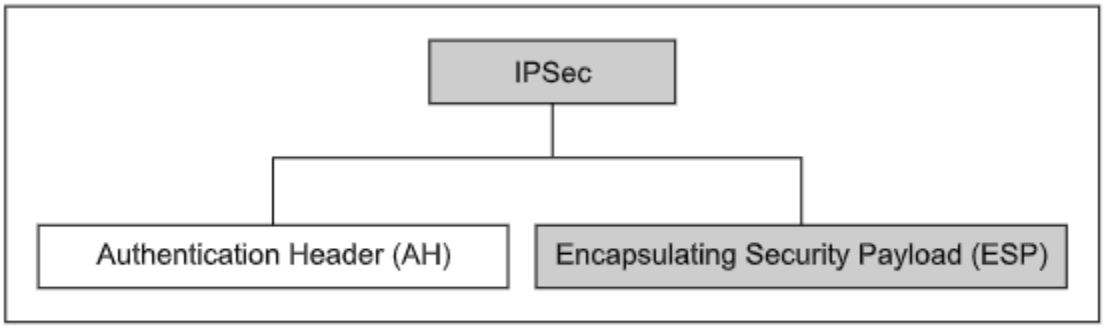


Fig. 9.28 IPSec protocols

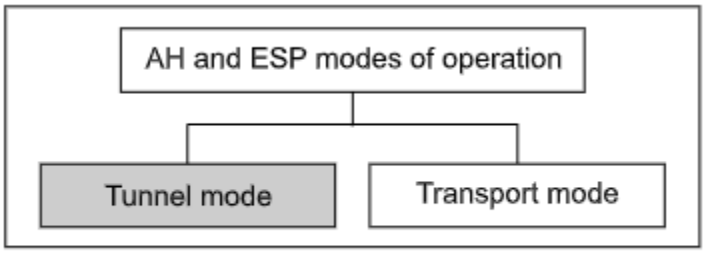


Fig. 9.29 AH and ESP modes of operation

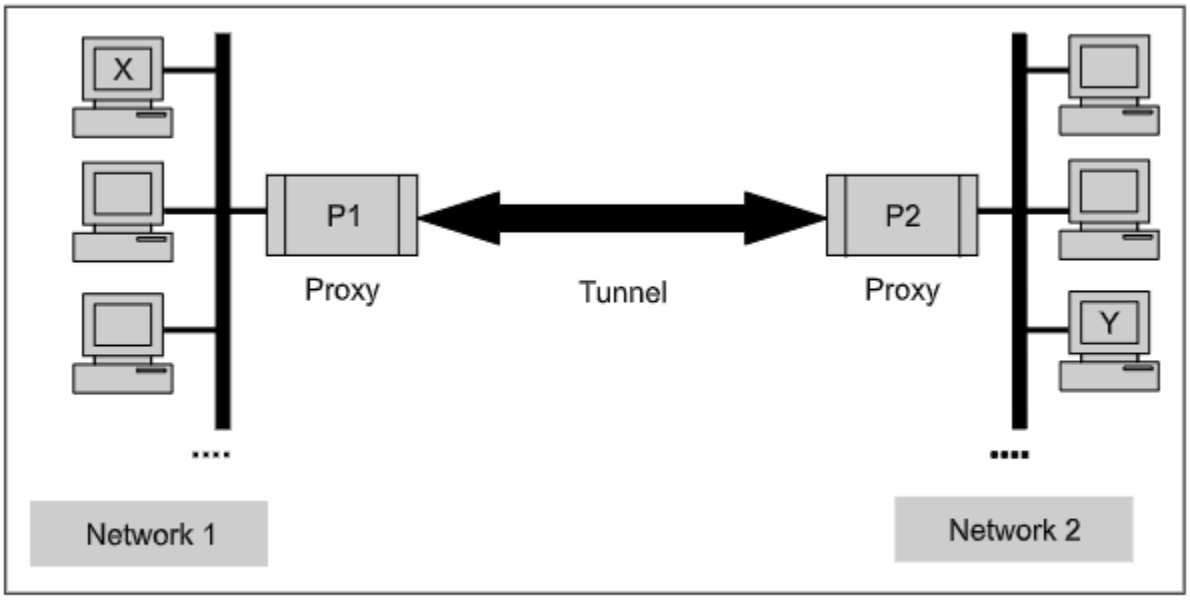


Fig. 9.30 Concept of tunnel mode

Example

For example, let's consider a scenario where two branch offices of an organization need to securely communicate over the internet. They can set up an IPsec-enabled network using SAs, AH, and ESP. The IP packets between the branch offices will be encrypted and authenticated, ensuring the confidentiality and integrity of the data. This secure communication can be established using tunnel mode, where the original IP packets are encapsulated and encrypted with a new IP header added.

Overall, IPsec provides a robust security solution for protecting network communication, allowing organizations to securely connect their branches, establish communication with other organizations, and provide secure remote internet access.

Q. Describe SSL architecture in detail in points, with diagram and example
ANS.

SSL Architecture

1. Introduction: The Secure Socket Layer (SSL) protocol is an Internet protocol that provides secure exchange of information between a web browser and a web server. It ensures authentication and confidentiality of data.
2. Position in TCP/IP Protocol Suite: SSL is located between the application layer and the transport layer in the TCP/IP protocol suite. It acts as an additional layer, providing a secure pipe between the web browser and the web server.
3. Working of SSL: SSL consists of three sub-protocols - the Handshake Protocol, the Record Protocol, and the Alert Protocol. The Handshake Protocol is used for initial communication between the client and the server, similar to a handshake before a conversation. The Record Protocol is responsible for encrypting and decrypting data, ensuring confidentiality. The Alert Protocol handles error messages and notifications.
4. SSL Handshake: The SSL handshake protocol involves a series of messages exchanged between the client and the server. It begins with a hello message, followed by the exchange of cryptographic parameters and authentication. This handshake establishes a secure connection between the client and the server.
5. Example: An example of SSL implementation is the secure communication between a web browser (client) and a web server. When a user accesses a secure website (https://), the SSL protocol is used to encrypt the data transmitted between the browser and the server, ensuring that sensitive information such as login credentials or credit card details are protected.

Diagram:

```
[Web Browser] <---SSL---> [Web Server]
```

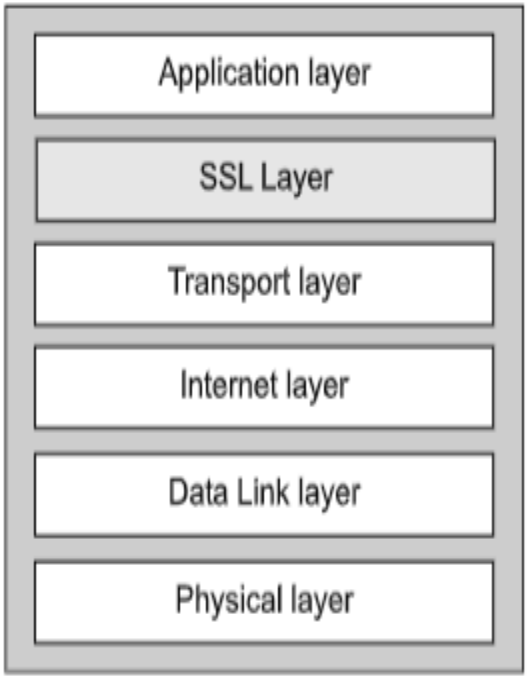


Fig. 6.9 Position of SSL in TCP/IP

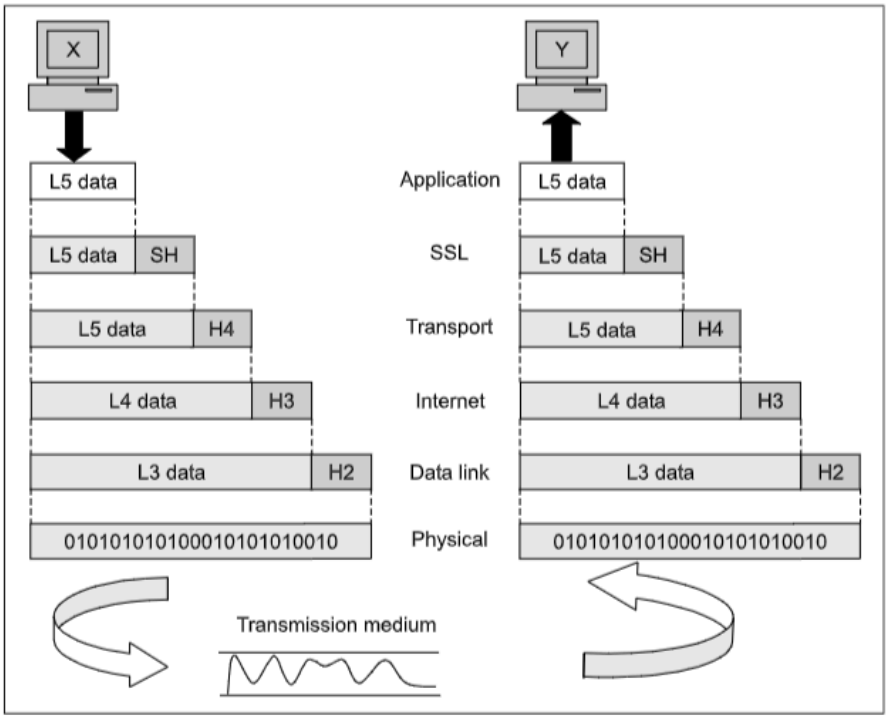


Fig. 6.10 SSL is located between application and transport layers

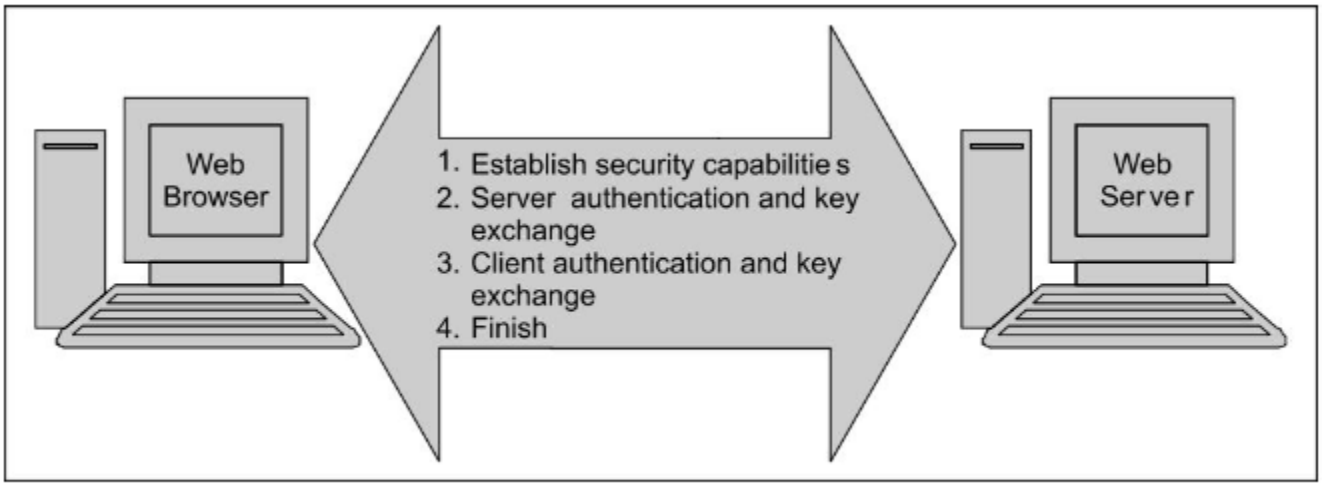


Fig. 6.13 SSL handshake phases

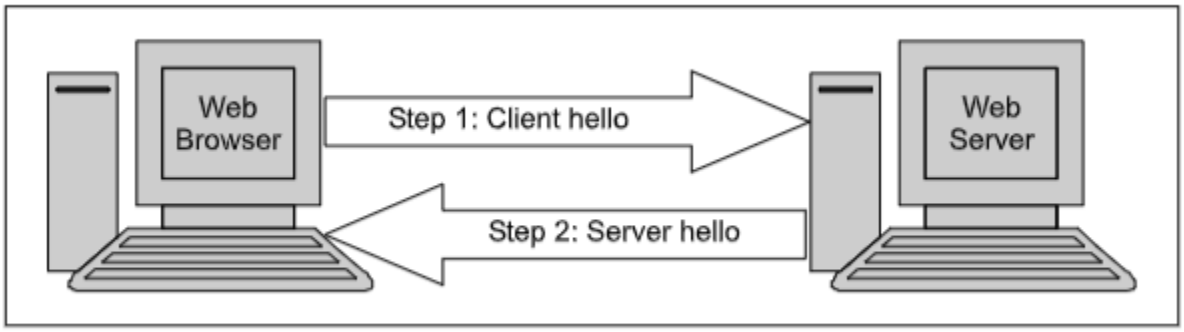


Fig. 6.14 SSL Handshake protocol *Phase 1: Establish security capabilities*

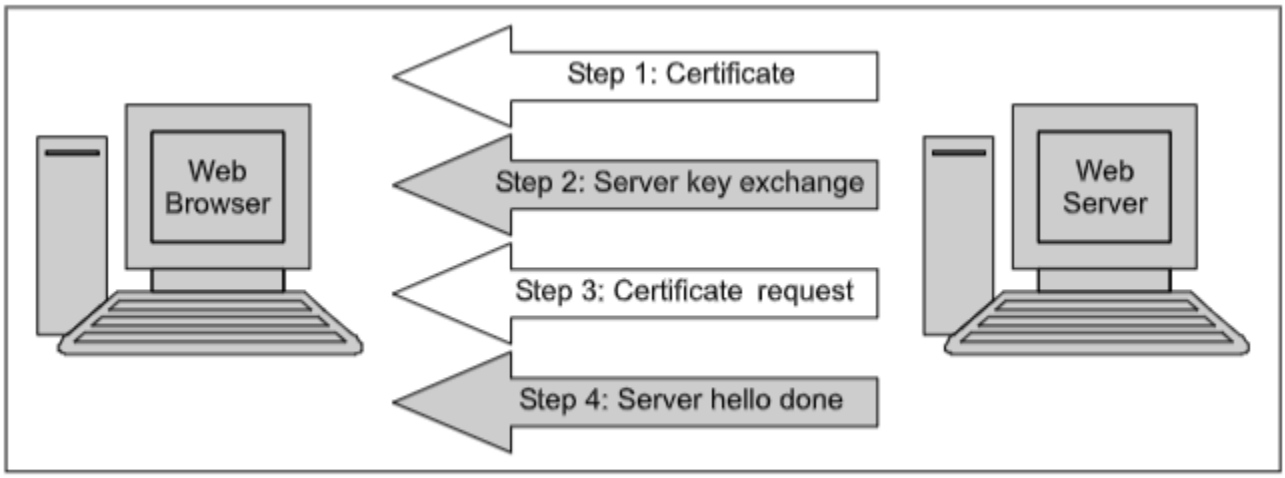


Fig. 6.15 SSL Handshake protocol *Phase 2: Server authentication and key exchange*

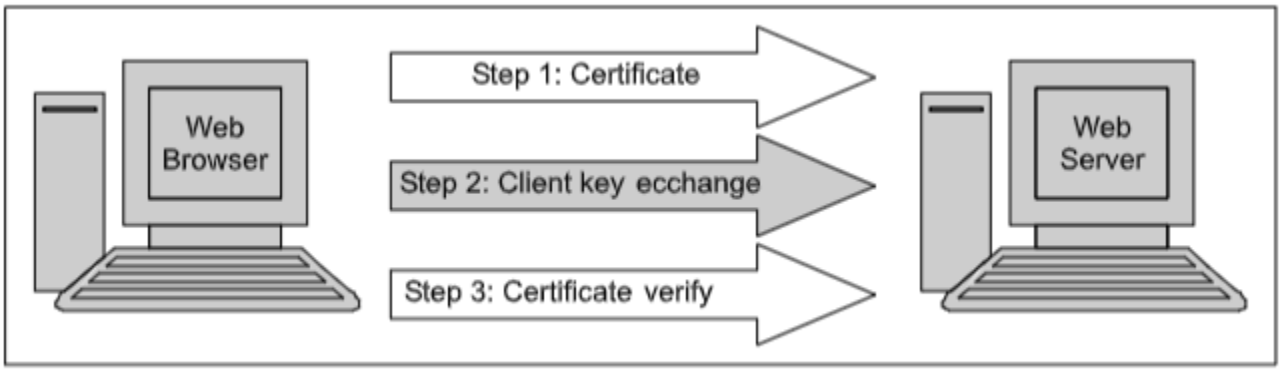


Fig. 6.16 SSL Handshake protocol *Phase 3: Client authentication and key exchange*

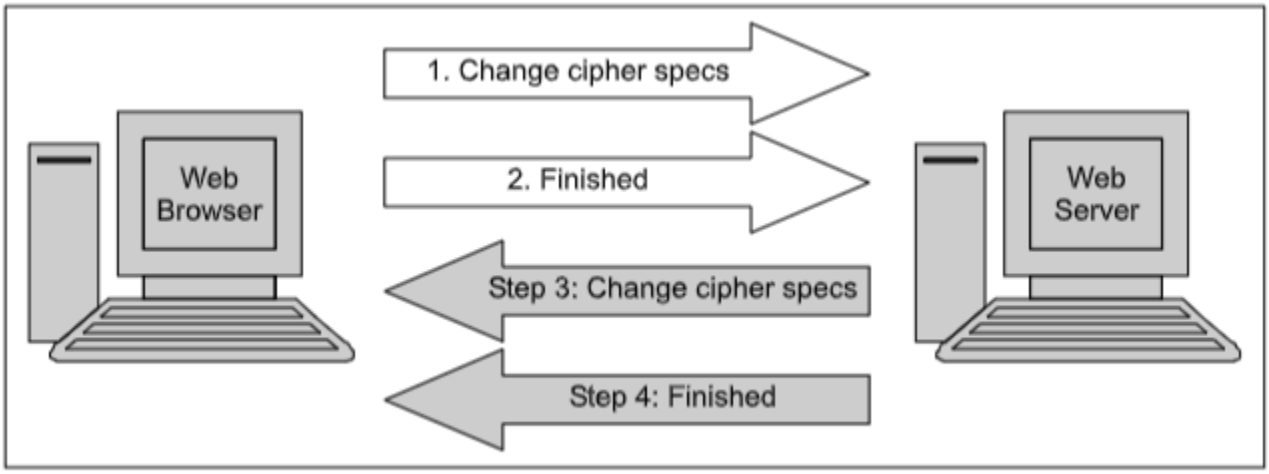


Fig. 6.17 SSL Handshake protocol *Phase 4: Finished*

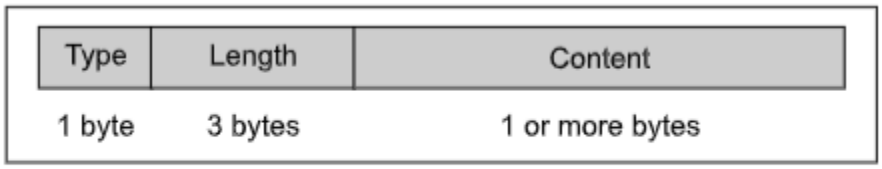


Fig. 6.11 Format of the *handshake protocol* messages

Q. In Diffie-hellman key exchange, Alice and Bob have chosen prime value q=17 and primitive root=5. If alice secret key is 4 and bob secret key is 6, what is the secret key they exchanged in points, with diagram and example

ANS.

Diffie-Hellman Key Exchange

In the given scenario, Alice and Bob have chosen the prime value q=17 and the primitive root=5. Alice's secret key is 4 and Bob's secret key is 6.

To calculate the secret key they exchanged, we can follow these steps:

1. Alice calculates her public key A by raising the primitive root (5) to the power of her secret key (4) modulo the prime value (17). So, A = (5^4) mod 17 = 3.
2. Bob calculates his public key B by raising the primitive root (5) to the power of his secret key (6) modulo the prime value (17). So, B = (5^6) mod 17 = 7.
3. Alice and Bob exchange their public keys A and B.
4. Alice calculates the shared secret key K by raising Bob's public key (B) to the power of her secret key (4) modulo the prime value (17). So, K = (7^4) mod 17 = 11.
5. Bob calculates the shared secret key K by raising Alice's public key (A) to the power of his secret key (6) modulo the prime value (17). So, K = (3^6) mod 17 = 11.

Therefore, the secret key they exchanged is 11.

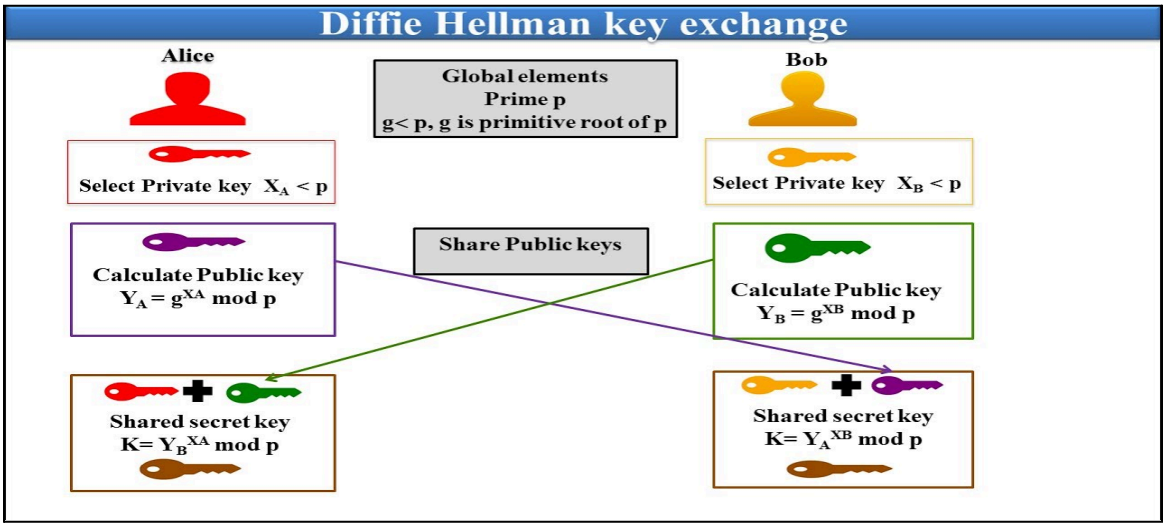
Here is a diagram to illustrate the Diffie-Hellman key exchange:

```
Alice's Side:
Secret Key: 4
Public Key: A = (5^4) mod 17 = 3

Bob's Side:
Secret Key: 6
Public Key: B = (5^6) mod 17 = 7

Shared Secret Key:
Alice: K = (B^4) mod 17 = 11
Bob: K = (A^6) mod 17 = 11
```

In this example, both Alice and Bob have successfully exchanged the secret key 11 using the Diffie-Hellman key exchange algorithm.



Q. Explain RSA algorithm with steps in selection and generation of public and private keys in points, with diagram and example

ANS.

RSA Algorithm: Selection and Generation of Public and Private Keys

1. Choose two large prime numbers P and Q.
    - Example: Let P = 47 and Q = 17.
2. Calculate N = P x Q.
    - Example: N = 47 x 17 = 119.
3. Calculate (P - 1) x (Q - 1) to find the factors of E.
    - Example: (P - 1) x (Q - 1) = 46 x 16 = 736.
4. Select the public key (encryption key) E such that it is not a factor of (P - 1) x (Q - 1).
    - Example: Choose E as 5.
5. Select the private key (decryption key) D such that (D x E) mod (P - 1) x (Q - 1) = 1.
    - Example: Choose D as 77.

Example of RSA Algorithm:

1. Encryption:
    - To encrypt a plain text PT, calculate the cipher text CT using the formula: CT = PT mod N^E.
    - Example: Encrypt plain text 10 using B's public key (E = 77, N = 119).
        - CT = 10 mod 119 = 100000 mod 119 = 40.
2. Decryption:
    - To decrypt the cipher text CT, calculate the plain text PT using the formula: PT = CT mod N^D.
    - Example: Decrypt cipher text 40 using B's private key (D = 5, N = 119).
        - PT = 40 mod 119 = 10 (original plain text).

Diagram:

```
A F F 6 6 5 Result modulo 119 = 41
1. Encode the original character using A = 1, B = 2, etc.
2. Raise the number to the power E, here 5.
3. Divide the result by 119 and get the remainder. The resulting number is the
cipher text.

B 41 41 77 Result modulo 119 6 F
1. Raise the number to the power D, here 77.
2. Divide the result by 119 and get the remainder. The resulting number is the
cipher text.
3. Decode the original character using 1 = A, 2 = B, etc.
```

This example demonstrates the encryption and decryption process using the RSA algorithm. The original character "F" is encoded as 6, encrypted using B's public key (77), and decrypted back to the original character using B's private key (5)

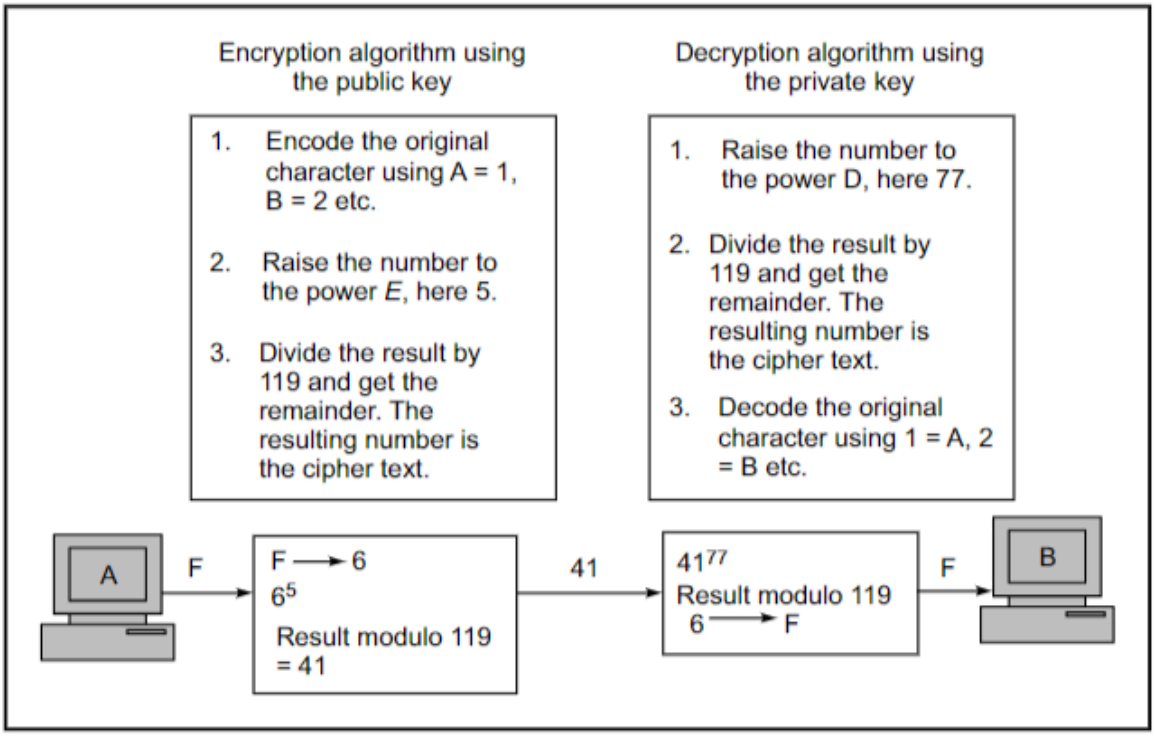


Fig. 4.6 Example of the RSA algorithm

Q. Explain MD5 in detail in points, with diagram and example
ANS.

Overview

MD5 (Message Digest Algorithm 5) is a cryptographic hash function developed by Ron Rivest. It is designed to produce a 128-bit message digest, which is a fixed-size representation of the input message. MD5 is fast and widely used for integrity checking and fingerprinting purposes. However, it has been found to have potential weaknesses and is no longer considered secure for cryptographic applications.

Key points

- MD5 is a message-digest algorithm developed by Ron Rivest.
- It is a fast algorithm that produces a 128-bit message digest.
- MD5 has its roots in a series of message-digest algorithms, with MD5 being the final version.
- Over the years, researchers have identified potential weaknesses in MD5.
- MD5 is no longer considered secure for cryptographic purposes.

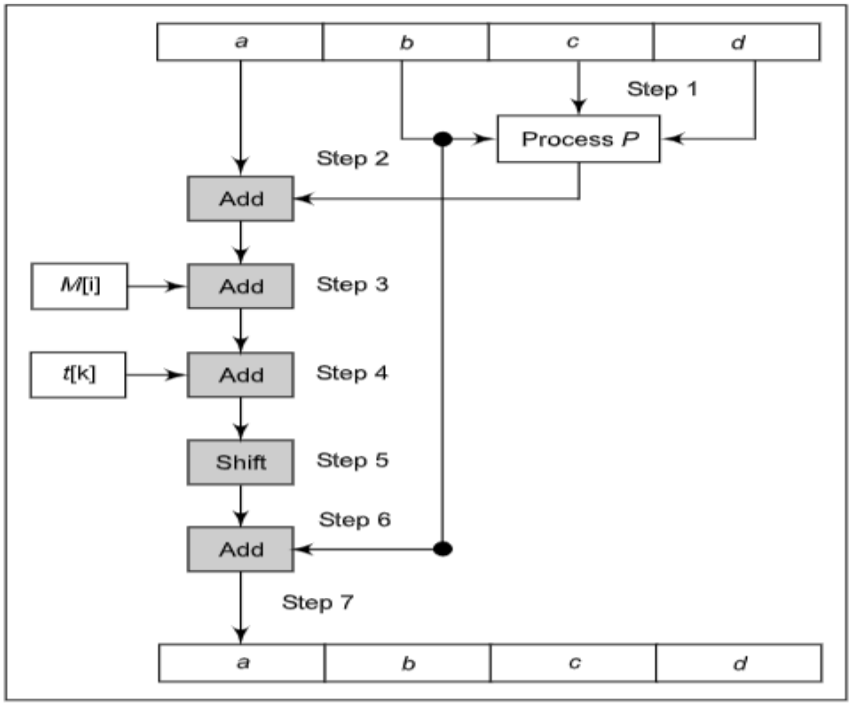


Fig. 4.33 One MD5 operation

Q. Describe the important algorithm modes which act as series of steps on block cipher in points, with diagram and example

ANS.

Algorithm Modes for Block Cipher

1. Electronic Code Book (ECB) Mode: In ECB mode, the plain-text message is divided into blocks of 64 bits each. Each block is encrypted independently using the same key. This mode is simple but lacks security as identical plain-text blocks produce identical cipher-text blocks.

Example: Suppose we have a plain-text message "HELLO WORLD" divided into two blocks "HELLO" and "WORLD". Each block is encrypted separately using the same key.

2. Cipher Block Chaining (CBC) Mode: In CBC mode, each plain-text block is XORed with the cipher-text block from the previous step before encryption. This introduces randomness and prevents identical plain-text blocks from producing identical cipher-text blocks.

Example: Suppose we have a plain-text message "HELLO WORLD" divided into two blocks "HELLO" and "WORLD". The first block is encrypted independently, but the second block is XORed with the cipher-text block of the first block before encryption.

3. Cipher Feedback (CFB) Mode: In CFB mode, a feedback mechanism is used where the output of the previous encryption is XORed with the plain-text block before encryption. This allows the encryption of individual bits or bytes rather than blocks.

Example: Suppose we have a plain-text message "HELLO WORLD" divided into two blocks "HELLO" and "WORLD". The first block is encrypted independently, but the second block is XORed with the output of the encryption of the first block before encryption.

4. Output Feedback (OFB) Mode: OFB mode is similar to CFB mode, but instead of XORing the plain-text block with the output of the previous encryption, it is XORed with the output of the encryption of the previous block.

Example: Suppose we have a plain-text message "HELLO WORLD" divided into two blocks "HELLO" and "WORLD". The first block is encrypted independently, but the second block is XORed with the output of the encryption of the first block before encryption.

These algorithm modes provide different ways to encrypt blocks of plain-text using a block cipher. Each mode has its own advantages and limitations in terms of security and efficiency.

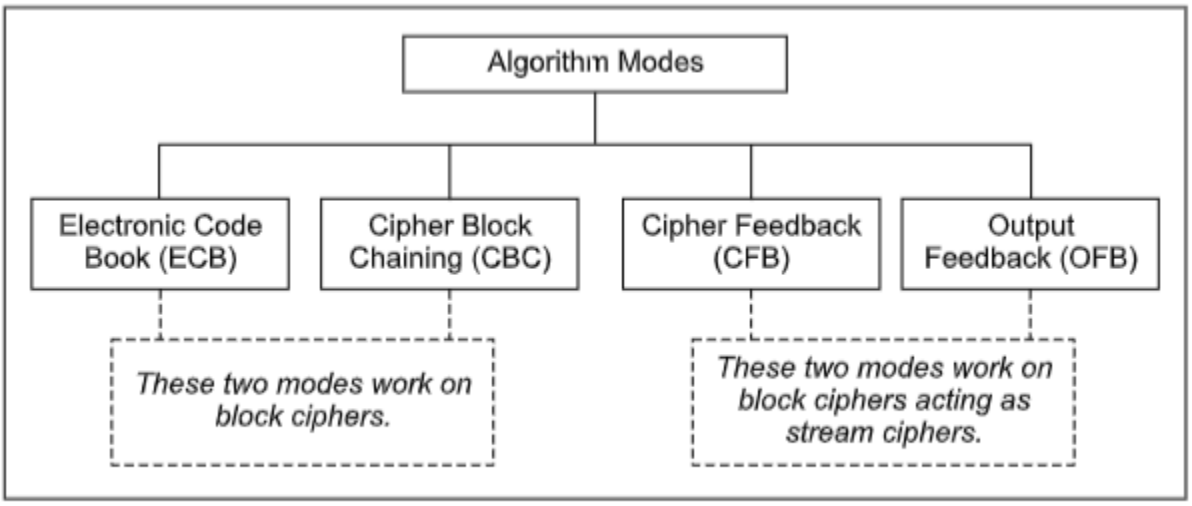


Fig. 3.5 Algorithm modes

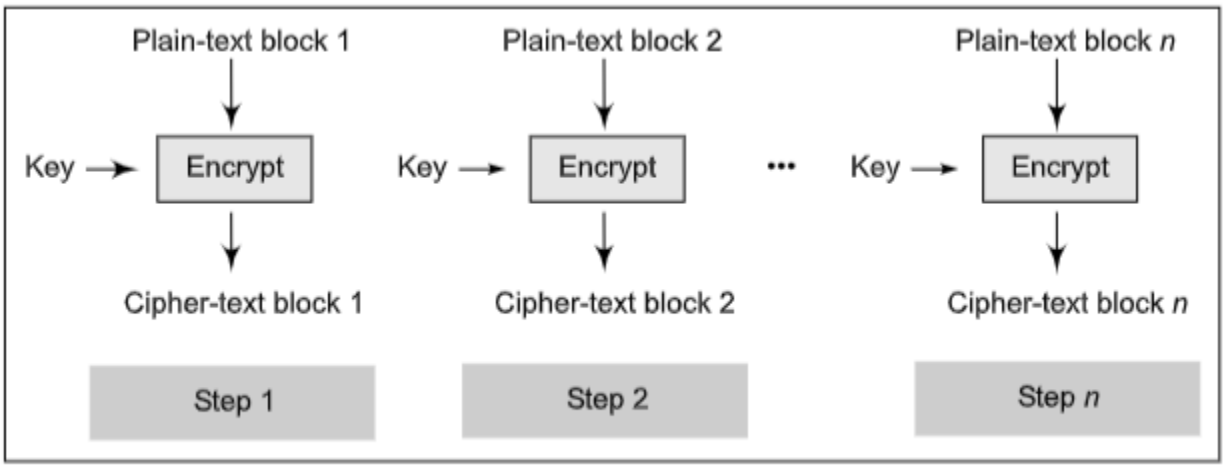


Fig. 3.6 ECB mode—the encryption process

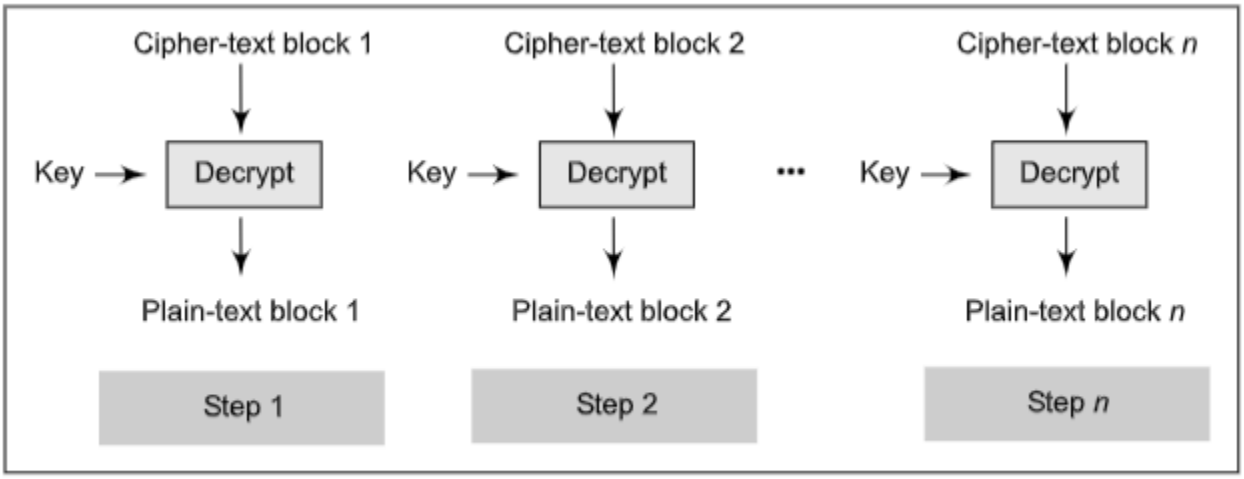


Fig. 3.7 ECB mode—the decryption process

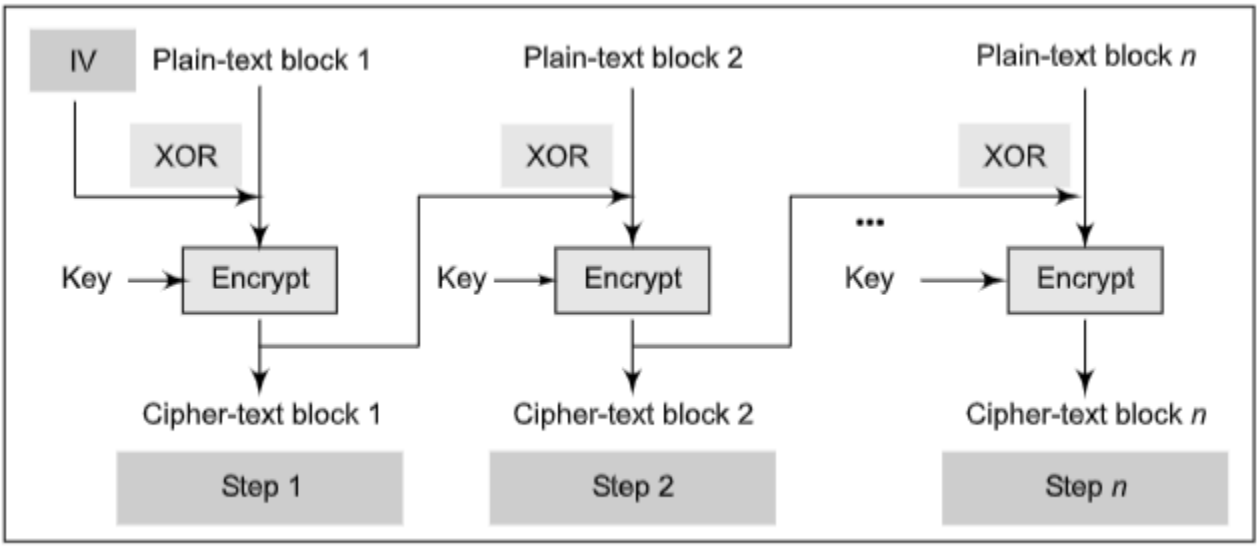
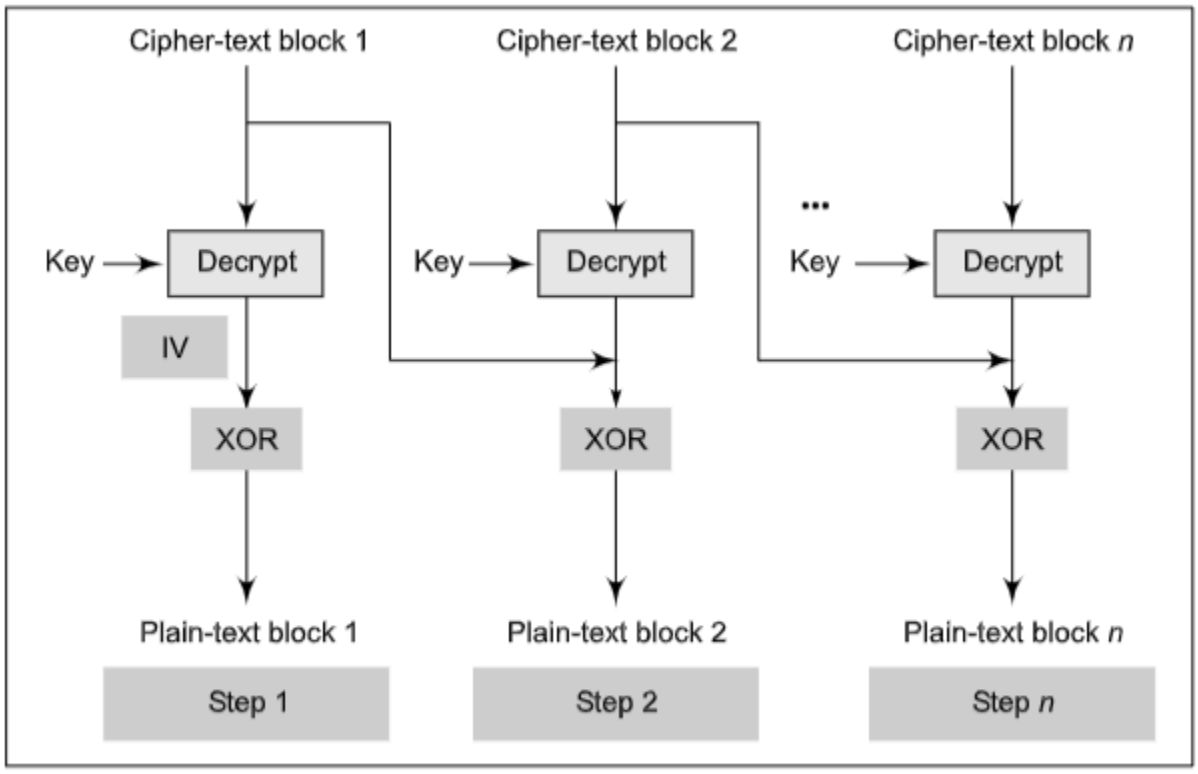


Fig. 3.8 CBC mode—the encryption process



ig. 3.9 CBC mode—the decryption process

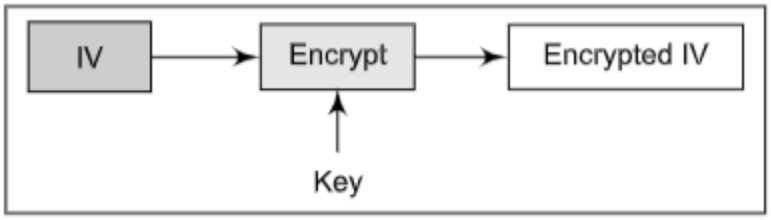


Fig. 3.10 CFB—Step 1

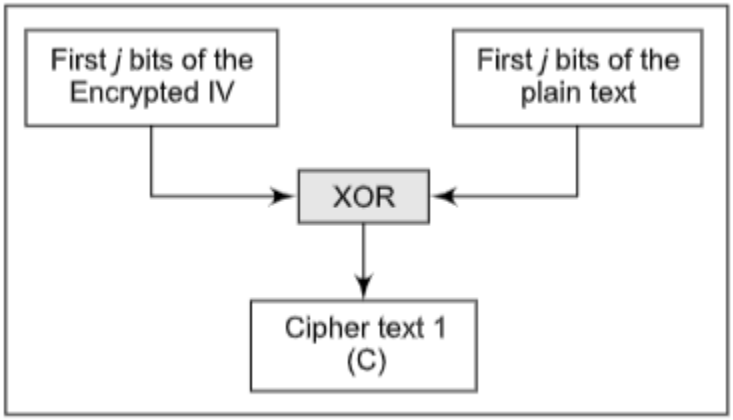


Fig. 3.11 CFB—Step 2

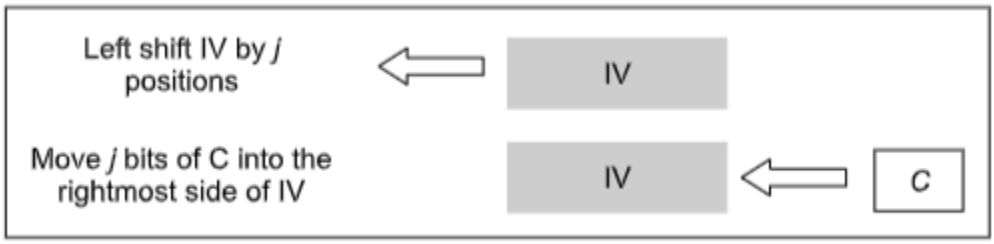


Fig. 3.12 CFB—Step 3

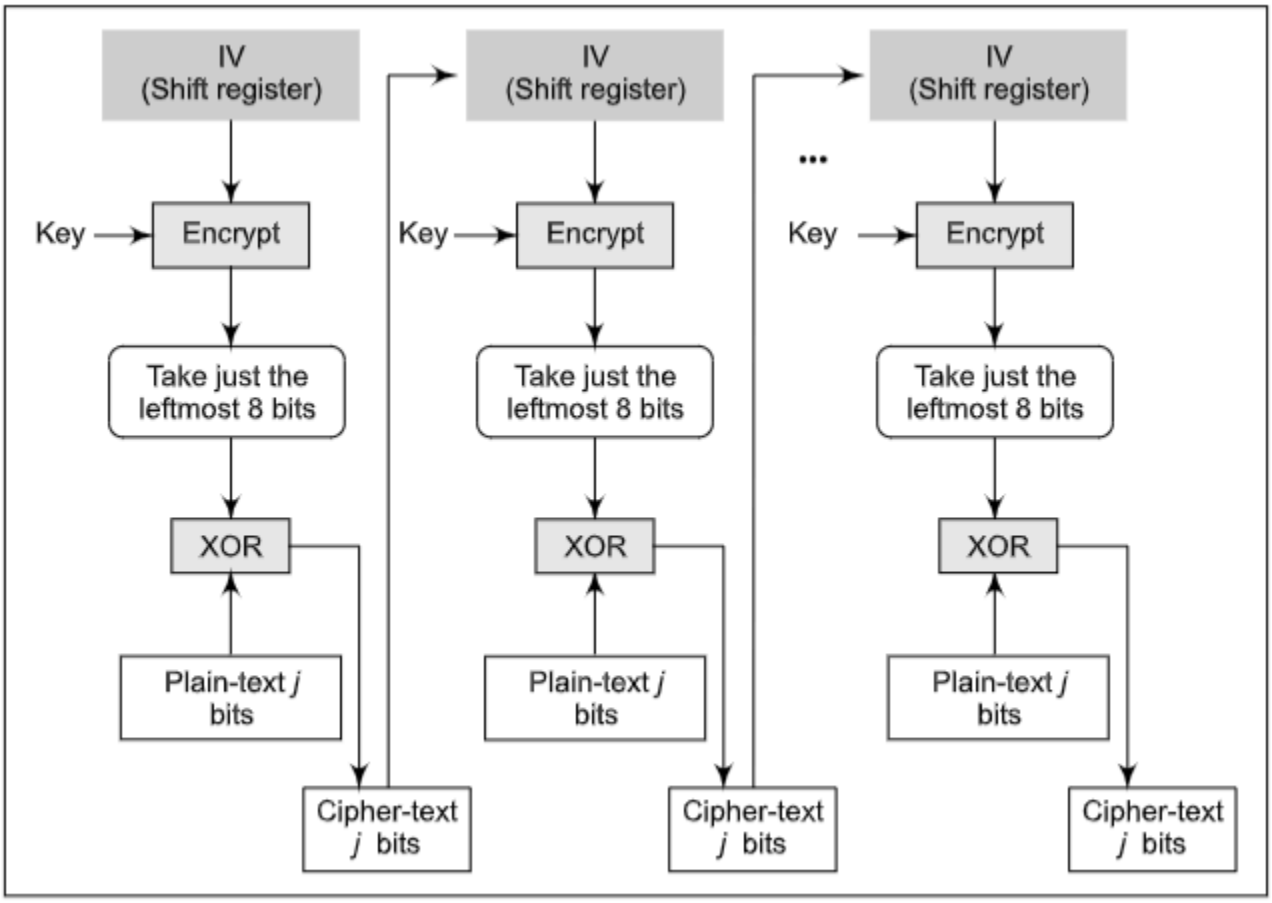


Fig. 3.13 CFB—the overall encryption process

Table 3.1 Algorithm modes: details and usage

Algorithm mode	Details	Usage
Electronic Code Book (ECB)	The same key independently encrypts blocks of text, 64 bits at a time.	Transmitting a single value in a secure fashion (e.g. password or key used for encryption).
Cipher Block Chaining (CBC)	64 bits of cipher text from the previous step and 64 bits of plain text of the next step are XORed together.	Encrypting blocks of text Authentication.
Cipher Feedback (CFB)	*K* bits of randomized cipher text from the previous step and *K* bits of plain text of the next step are XORed together.	Transmitting encrypted stream of data Authentication.
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption step is the preceding DES output.	Transmitting encrypted stream of data.
Counter (CTR)	A counter and plain-text block are encrypted together, after which the counter is incremented.	Block-oriented transmissions Applications needing high speed.

Table 3.2 Algorithm modes: advantages and problems

Feature	ECB	CBC	CFB	OFB/Counter
Security-related problems	Plain text patterns are not hidden. Input to the block cipher is the same as the plain text, and is not randomized. Plain text is easy to manipulate, blocks of text can be removed, repeated, or exchanged.	Plain-text blocks can be removed from the beginning and end of the message, and bits of the first block can be altered.	Plain-text blocks can be removed from the beginning and end of the message, and bits of the first block can be altered.	Plain text is easy to manipulate. Altering cipher text alters plain text directly.
Security-related advantages	The same key can be used for encrypting multiple messages.	XOR of plain text with previous cipher-text block hides the plain text. The same key can be used for encrypting multiple messages.	Plain-text patterns are hidden. The same key can be used for encrypting multiple messages, by using a different IV. Input to the block cipher is randomized.	Plain-text patterns are hidden. The same key can be used for encrypting multiple messages, by using a different IV. Input to the block cipher is randomized.
Problems related to effectiveness	Size of cipher text is more than the plain-text size by one padding block. Pre-processing is not possible.	Size of cipher text is more than the plain text size by one block. Pre-processing is not possible. Parallelism cannot be introduced in encryption.	Size of cipher text is the same as that of the plain-text size. Parallelism cannot be introduced in encryption.	Size of cipher text is the same as that of the plain-text size. Parallelism cannot be introduced (OFB only).