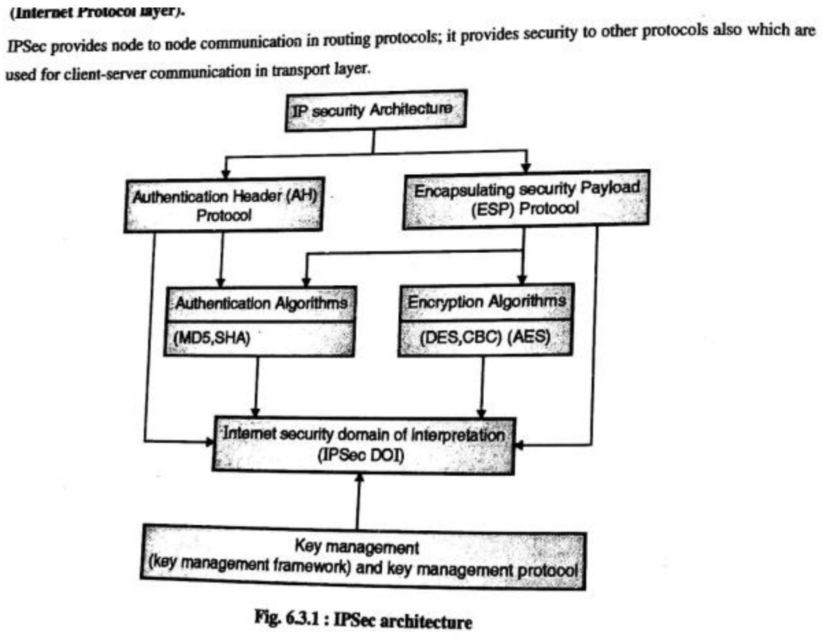
IP Security, Transport level security and Email Security

❋IP Sec:

- IP Sec (Internet Protocol Security) is an Internet Engineering Task Force (IETF) standard suite of protocols between two communication points across the IP network that provide data authentication, integrity, and confidentiality.
- It also defines the encrypted, decrypted, and authenticated packets.
- The protocols needed for secure key exchange and key management are defined in it.
- these components are very important in order to provide the three main services:

1. Confidentiality
2. Authentication
3. Integrity

❋IP Security Architecture:

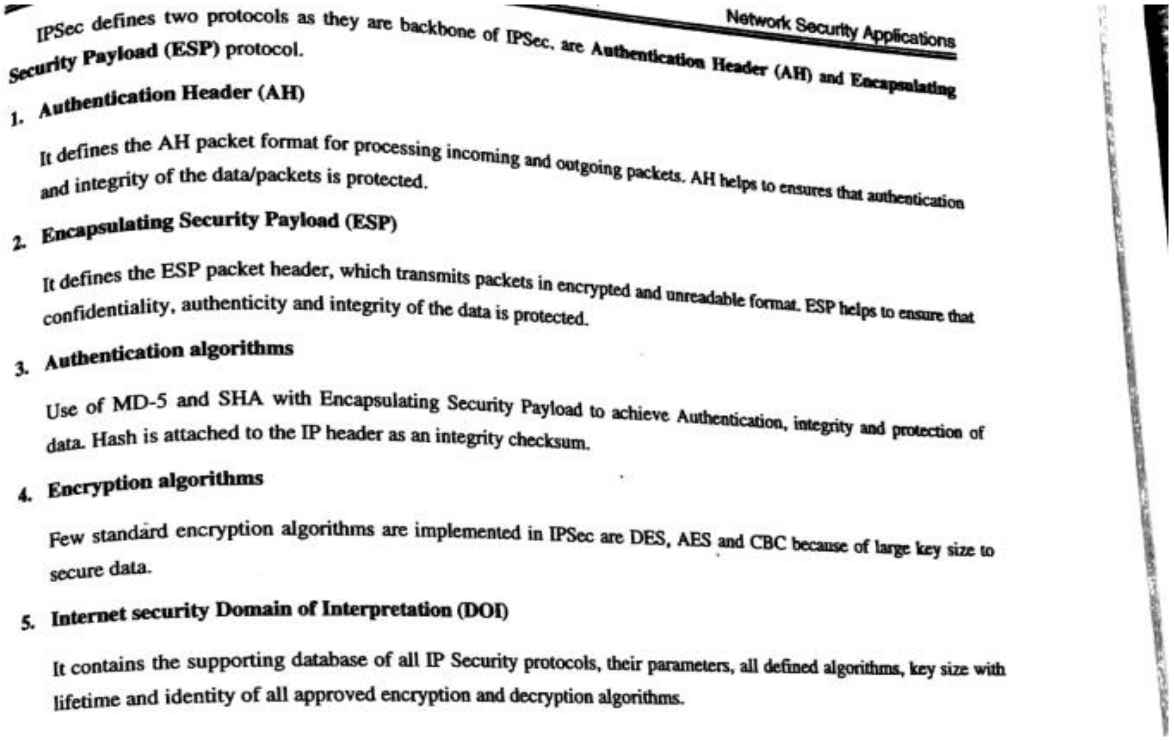


(Internet Protocol layer).

– IPSec provides node to node communication in routing protocols; it provides security to other protocols also which are used for client-server communication in transport layer.

IP security Architecture
Authentication Header (AH) Protocol
Encapsulating security Payload (ESP) Protocol
Authentication Algorithms
(MD5,SHA)
Encryption Algorithms
(DES,CBC) (AES)
Internet security domain of interpretation (IPSec DOI)
Key management (key management framework) and key management protocol

Fig. 6.3.1 : IPSec architecture



IPSec defines two protocols as they are backbone of IPSec, are **Authentication Header (AH) and Encapsulating Security Payload (ESP)** protocol.

1. **Authentication Header (AH)**

It defines the AH packet format for processing incoming and outgoing packets. AH helps to ensures that authentication and integrity of the data/packets is protected.

2. **Encapsulating Security Payload (ESP)**

It defines the ESP packet header, which transmits packets in encrypted and unreadable format. ESP helps to ensure that confidentiality, authenticity and integrity of the data is protected.

3. **Authentication algorithms**

Use of MD-5 and SHA with Encapsulating Security Payload to achieve Authentication, integrity and protection of data. Hash is attached to the IP header as an integrity checksum.

4. **Encryption algorithms**

Few standard encryption algorithms are implemented in IPSec are DES, AES and CBC because of large key size to secure data.

5. **Internet security Domain of Interpretation (DOI)**

It contains the supporting database of all IP Security protocols, their parameters, all defined algorithms, key size with lifetime and identity of all approved encryption and decryption algorithms.

6. **Key management**

As defined earlier key management is used to generate and distribute the keys required for IPSec protocols.

6.3.1 IPSec Modes

IPSec operates in two different modes :

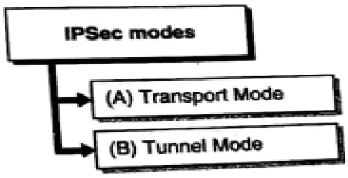


Fig. 6.3.2 : IPSec modes

6.3.1(A) Transport Mode

- In Transport mode IPSec protects the data that is delivered from transport layer to network layer or in other words, we can say that, transport mode protects the payload (a packet consist of controlled information and user data) of network layer.
- It encapsulates the transport layer payload by adding IPSec header and IP Sec trailer and sends this encapsulated packet to network layer.

- After that the IP headers of network layer is added to that encapsulated payload. IPSec transport mode is responsible for complete delivery of packet (traffic) from one host to another host or from host to gateways called as end-to-end communications.
- **For example :** Communications between client machine and a server machine, communications between two routers and from router to gateway is also considered as **end-to-end communication**. IPSec transport mode is responsible for secure communications between all these devices.
- Transport mode helps to protect user data, also known as IP payload through an AH or ESP header. In transport mode payload of IP packet is encrypted by the IPSec headers and trailers but the IP header information, which is remain unchanged as shown in Fig. 6.3.3.
- The payload of an IP packet is protected before it is handled by the network layer as shown in Fig. 6.3.3. Fig. 6.3.4 shows how the data exchange (end to end security) take place after encrypting payload.

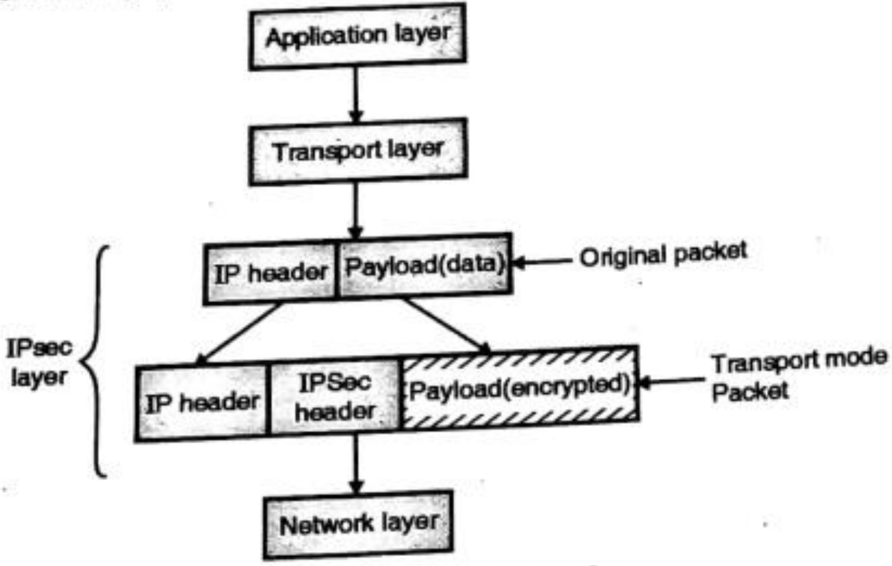
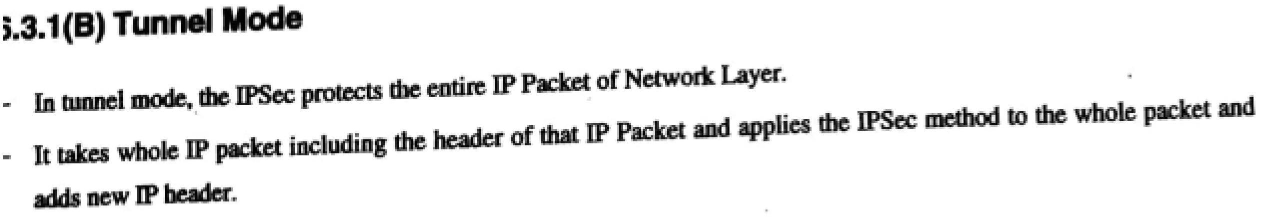
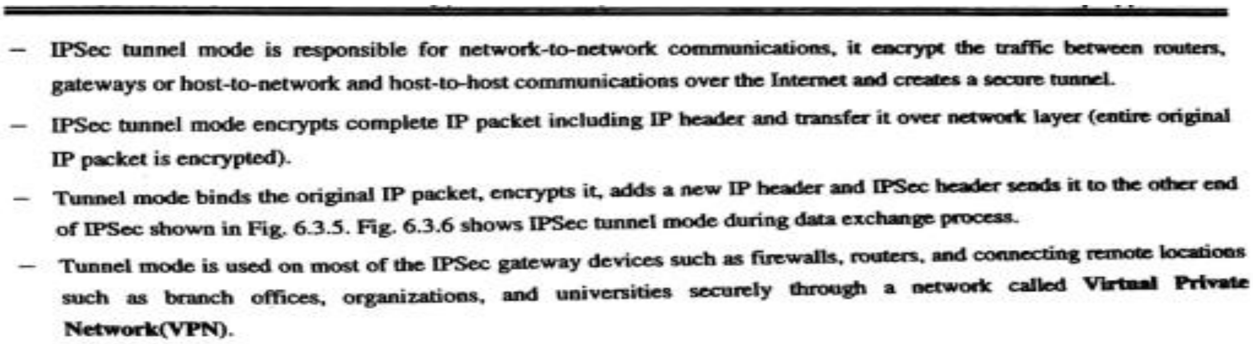


Fig. 6.3.3 : Transport mode



.3.1(B) Tunnel Mode

- In tunnel mode, the IPSec protects the entire IP Packet of Network Layer.
- It takes whole IP packet including the header of that IP Packet and applies the IPSec method to the whole packet and adds new IP header.



- IPSec tunnel mode is responsible for network-to-network communications, it encrypt the traffic between routers, gateways or host-to-network and host-to-host communications over the Internet and creates a secure tunnel.
- IPSec tunnel mode encrypts complete IP packet including IP header and transfer it over network layer (entire original IP packet is encrypted).
- Tunnel mode binds the original IP packet, encrypts it, adds a new IP header and IPSec header sends it to the other end of IPSec shown in Fig. 6.3.5. Fig. 6.3.6 shows IPSec tunnel mode during data exchange process.
- Tunnel mode is used on most of the IPSec gateway devices such as firewalls, routers, and connecting remote locations such as branch offices, organizations, and universities securely through a network called **Virtual Private Network(VPN).**

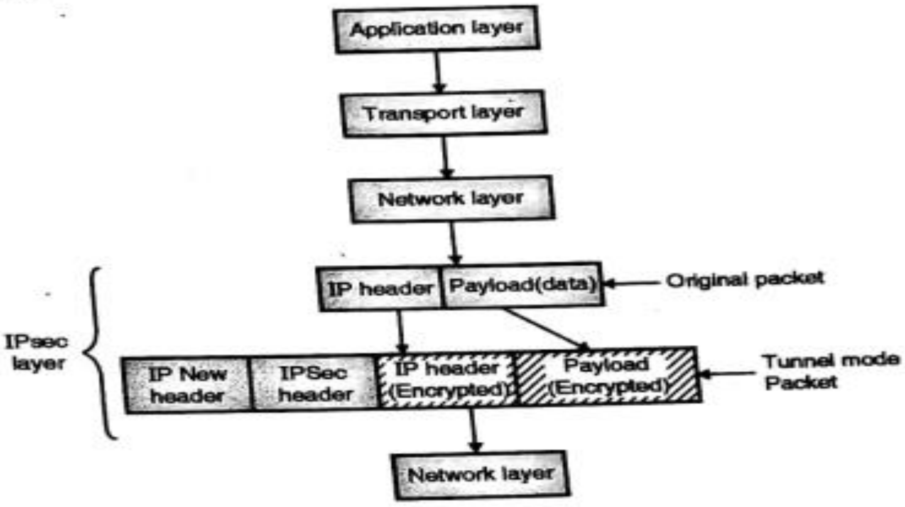


Fig. 6.3.5 : Tunnel Mode

✳<u>Advantages of IPSec:</u>

- Strong security: IPSec provides strong cryptographic security services that help protect sensitive data and ensure network privacy and integrity.
- Wide compatibility: IPSec is an open standard protocol that is widely supported by vendors and can be used in heterogeneous environments.
- Flexibility: IPSec can be configured to provide security for a wide range of network topologies, including point-to-point, site-to-site, and remote access connections.
- Scalability: IPSec can be used to secure large-scale networks and can be scaled up or down as needed.

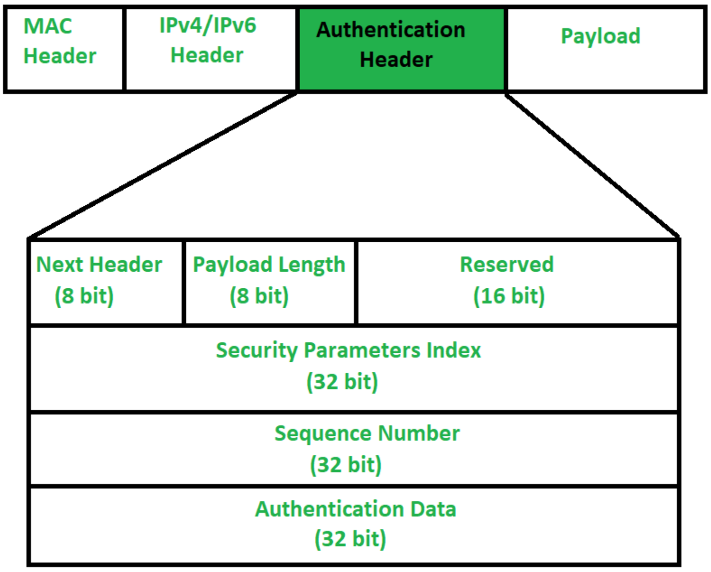
✳<u>Disadvantages of IPSec:</u>

- Configuration complexity: IPSec can be complex to configure and requires specialized knowledge and skills.
- Compatibility issues: IPSec can have compatibility issues with some network devices and applications, which can lead to interoperability problems.
- Performance impact: IPSec can impact network performance due to the overhead of encryption and decryption of IP packets.
- Key management: IPSec requires effective key management to ensure the security of the cryptographic keys used for encryption and authentication.

☢Internet Protocol Authentication (AH)Header :

When packet is sent from source A to Destination B, it consists of data that we need to send and header which consist of information regarding packet. Authentication Header verifies origin of data and also payload to confirm if there has been modification done in between, during transmission between source and destination.

Authentication Header: The question may arise, that how IP header will know that adjacent Extension header is Authentication Header. Well, there is protocol field in IP Header which tells type of header that is present in packet. So, protocol field in IP Header should have value of "51" in order to detect Authentication Header.



Next Header – Next Header is 8-bit field that identifies type of header present after Authentication Header. In case of TCP, UDP or destination header or some other extension header it will store correspondence IP protocol number . Like, number 4 in this field will indicate IPv4, number 41 will indicate IPv6 and number 6 will indicate TCP.

Payload Length – Payload length is length of Authentication header and here we use scaling factor of 4. Whatever be size of header, divide it by 4 and then subtract by 2. We are subtracting by 2 because we're not counting first 8 bytes of Authentication header, which is first two row of picture given above. It means we are not including Next Header, Payload length, Reserved and Security Parameter index in calculating payload length. Like, say if payload length is given to be X. Then (X+2)*4 will be original Authentication header length.

Reserved – This is 16-bit field which is set to "zero" by sender as this field is reserved for future use.

Security Parameter Index (SPI) – It is arbitrary 32-bit field. It is very important field which identifies all packets which belongs to present connection. If we're sending data from Source A to Destination B. Both A and B will already know algorithm and key they are going to use. So for Authentication, hashing function and key will be required which only source and destination will know about. Secret key between A and B is exchanged by method of Diffie Hellman algorithm. So Hashing algorithm and secret key for Security parameter index of connection will be fixed. Before data transfer starts security association needs to be established. In Security Association, both parties needs to communicate prior to data exchange. Security association tells what is security parameter index, hashing algorithm and secret key that are being used.

Sequence Number – This unsigned 32-bit field contains counter value that increases by one for each packet sent. Every packet will need sequence number. It will start from 0 and will go till 2^{32} – 1 and there will be no wrap around. Say, if all sequence numbers are over and none of it is left but we cannot wrap around as it is not allowed. So, we will end connection and re-establish connection again to resume transfer of remaining data from sequence number 0. Basically sequence numbers are used to stop replay attack. In Replay attack, if same message is sent twice or more, receiver won't be able to know if both messages are sent from a single source or not. Say, I am requesting 100$ from receiver and Intruder in between asked for another 100$. Receiver won't be able to know that there is intruder in between.

Authentication Data (Integrity Check Value) – Authentication data is variable length field that contains Integrity Check Value (ICV) for packet. Using hashing algorithm and secret key, sender will create message digest which will be sent to receiver. Receiver on other hand will use same hashing algorithm and secret key. If both message digest matches then receiver will accept data. Otherwise, receiver will discard it by saying that message has been modified in between. So basically, authentication data is used to verify integrity of transmission. Also length of Authentication data depends upon hashing algorithm you choose.

Modes of operations in Authentication Header:

There are two modes in the authentication header

- Authentication Header Transport Mode:
- Authentication Header Tunnel Mode:

Authentication Header Transport Mode: In the authentication header transport mode, it is lies between the original IP Header and IP Packets original TCP header.

Authentication Header Tunnel Mode: In this authentication header tunnel mode, the original IP packet is authenticated entire and the authentication header is inserted between the original IP header and new outer IP header. Here, the inner IP header contains the ultimate source IP address and destination IP address. whereas the outer IP header contains different IP address that is IP address of the firewalls or other security gateways.

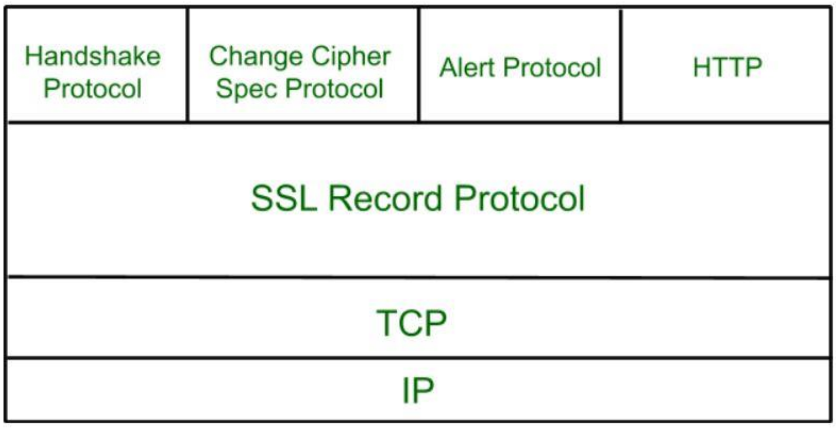
☣Secure Socket Layer Protocols:

- Secure Socket Layer (SSL) provides security to the data that is transferred between web browser and server.
- SSL encrypts the link between a web server and a browser which ensures that all data passed between them remain private and free from attack.
- Secure Sockets Layer (SSL) is a standard technique for transmitting documents securely across a network. SSL technology, created by Netscape, establishes a secure connection between a Web server and a browser, ensuring private and secure data transmission.
- SSL communicates using the Transport Control Protocol (TCP).
- The term "socket" in SSL refers to the method of sending data via a network between a client and a server.
    - ✓ SSL record protocol
    - ✓ Handshake protocol
    - ✓ Change-cipher spec protocol
    - ✓ Alert protocol

Salient Features of Secure Socket Layer:

- ❖ The advantage of this approach is that the service can be tailored to the specific needs of the given application.
- ❖ Secure Socket Layer was originated by Netscape.
- ❖ SSL is designed to make use of TCP to provide reliable end-to-end secure service.
- ❖ This is a two-layered protocol.

SSL Protocol Stack:



SSL Record Protocol:

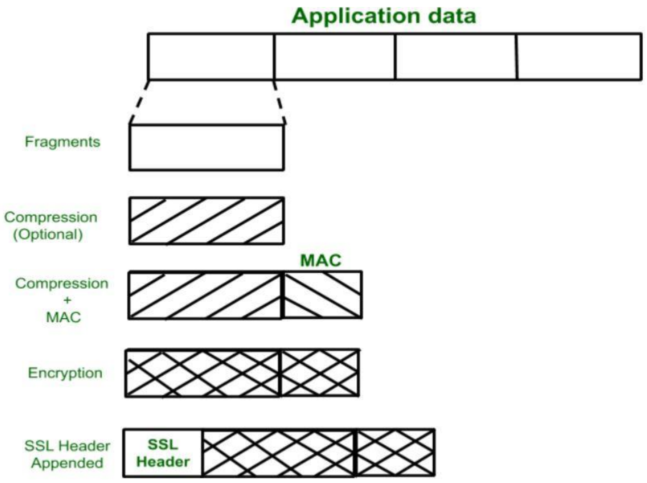
SSL Record provides two services to SSL connection.

- ✓ Confidentiality
- ✓ Message Integrity

In the SSL Record Protocol application data is divided into fragments.

The fragment is compressed and then encrypted MAC (Message Authentication Code) generated by algorithms like SHA (Secure Hash Protocol) and MD5 (Message Digest) is appended.

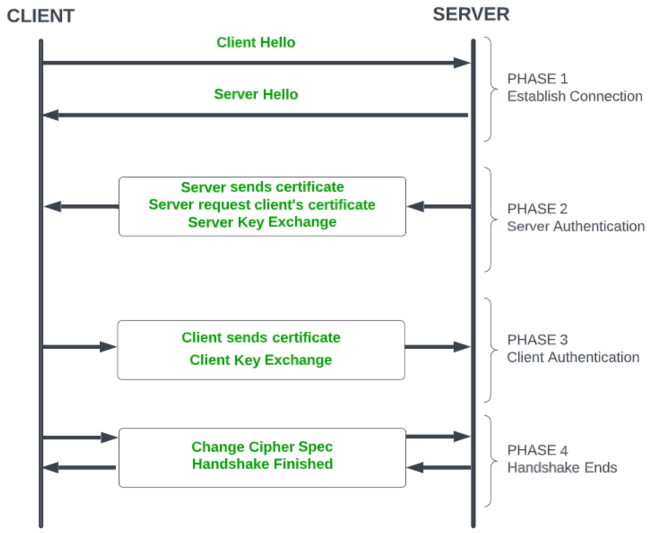
After that encryption of the data is done and in last SSL header is appended to the data.



Handshake Protocol:

Handshake Protocol is used to establish sessions. This protocol allows the client and server to authenticate each other by sending a series of messages to each other. Handshake protocol uses four phases to complete its cycle.

- ❖ Phase-1: In Phase-1 both Client and Server send hello-packets to each other. In this IP session, cipher suite and protocol version are exchanged for security purposes.
- ❖ Phase-2: Server sends his certificate and Server-key-exchange. The server end phase-2 by sending the Server-hello-end packet.
- ❖ Phase-3: In this phase, Client replies to the server by sending his certificate and Client-exchange-key.
- ❖ Phase-4: In Phase-4 Change-cipher suite occurs and after this the Handshake Protocol ends.



SSL HANDSHAKE PROTOCOL

Change-cipher Protocol:

This protocol uses the SSL record protocol. Unless Handshake Protocol is completed, the SSL record Output will be in a pending state. After the handshake protocol, the Pending state is converted into the current state.

Change-cipher protocol consists of a single message which is 1 byte in length and can have only one value. This protocol's purpose is to cause the pending state to be copied into the current state.

1 byte

Alert Protocol:

This protocol is used to convey SSL-related alerts to the peer entity. Each message in this protocol contains 2 bytes.

Level (1 byte)	Alert (1 byte)

Versions of SSL:

- SSL 1 – Never released due to high insecurity.
- SSL 2 – Released in 1995.
- SSL 3 – Released in 1996.
- TLS 1.0 – Released in 1999.
- TLS 1.1 – Released in 2006.
- TLS 1.2 – Released in 2008.
- TLS 1.3 – Released in 2018

SSL (Secure Sockets Layer) certificate is a digital certificate used to secure and verify the identity of a website or an online service. The certificate is issued by a trusted third-party called a Certificate Authority (CA), who verifies the identity of the website or service before issuing the certificate.

The SSL certificate has several important characteristics that make it a reliable solution for securing online transactions:

1. **Encryption**: The SSL certificate uses encryption algorithms to secure the communication between the website or service and its users. This ensures that the sensitive information, such as login credentials and credit card information, is protected from being intercepted and read by unauthorized parties.
2. **Authentication**: The SSL certificate verifies the identity of the website or service, ensuring that users are communicating with the intended party and not with an impostor. This provides assurance to users that their information is being transmitted to a trusted entity.
3. **Integrity**: The SSL certificate uses message authentication codes (MACs) to detect any tampering with the data during transmission. This ensures that the data being transmitted is not modified in any way, preserving its integrity.
4. **Non-repudiation**: SSL certificates provide non-repudiation of data, meaning that the recipient of the data cannot deny having received it. This is important in situations where the authenticity of the information needs to be established, such as in e-commerce transactions.
5. **Public-key cryptography**: SSL certificates use public-key cryptography for secure key exchange between the client and server. This allows the client and server to securely exchange encryption keys, ensuring that the encrypted information can only be decrypted by the intended recipient.

🦁Secure Shell (SSH) Protocol Stack:

- SSH stands for Secure Shell or Secure Socket Shell.
- It is a cryptographic network protocol that allows two computers to communicate and share the data over an insecure network such as the internet.
- It is used to login to a remote server to execute commands and data transfer from one machine to another machine.
- The SSH protocol was developed by SSH communication security Ltd to safely communicate with the remote machine.
- Secure communication provides a strong password authentication and encrypted communication with a public key over an insecure channel.
- It is used to replace unprotected remote login protocols **such as Telnet, rlogin, rsh, etc**., and insecure file transfer protocol FTP.
- Its security features are widely used by network administrators for managing systems and applications remotely.
- The SSH protocol protects the network from various attacks such as DNS spoofing, IP source routing, and IP spoofing.

How does SSH Works?

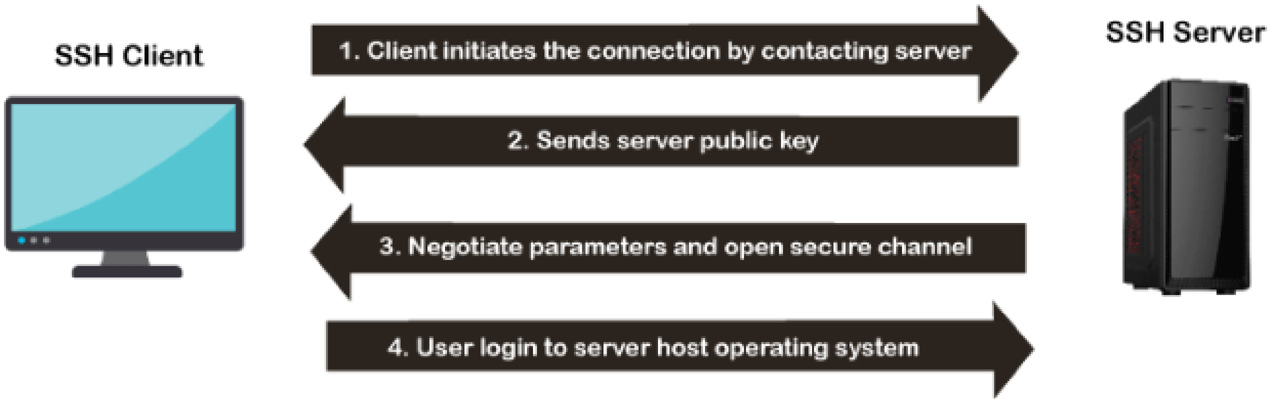
The SSH protocol works in a client-server model, which means it connects a secure shell client application (End where the session is displayed) with the SSH server (End where session executes).

As discussed above, it was initially developed to replace insecure login protocols such as Telnet, rlogin, and hence it performs the same function.

The basic use of SSH is to connect a remote system for a terminal session and to do this, following command is used:

```
ssh UserName@SSHserver.test.com
```

If we are connecting for the first time, it will prompt the remote host's public key fingerprint and ask to connect.


SSH Client
SSH Server
1. Client initiates the connection by contacting server
2. Sends server public key
3. Negotiate parameters and open secure channel
4. User login to server host operating system

The architecture of SSH Protocol

The SSH architecture is made-up of three well-separated layers. These layers are:

1. Transport Layer
2. User-authentication layer
3. Connection Layer

SSH Encryption Techniques

To make a secure transmission, SSH uses three different encryption techniques at various points during a transmission. These techniques are:

1. Symmetrical Encryption
2. Asymmetrical Encryption
3. Hashing

Usages of SSH protocol

The popular usages of SSH protocol are given below:

- It provides secure access to users and automated processes.
- It is an easy and secure way to transfer files from one system to another over an insecure network.
- It also issues remote commands to the users.
- It helps the users to manage the network infrastructure and other critical system components.
- It is used to log in to shell on a remote system (Host), which replaces Telnet and rlogin and is used to execute a single command on the host, which replaces rsh.

🐤Email security

Email security refers to the steps where we protect the email messages and the information that they contain from unauthorized access, and damage. It involves ensuring the confidentiality, integrity, and availability of email messages, as well as safeguarding against phishing attacks, spam, viruses, and another form of malware. It can be achieved through a combination of technical and non-technical measures.

(5 Marks)

Q. 6.4.2 How does PGP achieve confidentiality and authentication in emails? (Refer section 6.4) (6 Marks)

- We all are aware that most popular use of Internet is to send the email and chatting with the friend's, partner etc. But have you ever think that if we are sending mail to intended person is going in his inbox only?
- Security concerns have estimated that only about one in every 100 messages is secured against interception and modification attacks. Are we aware that sending an email to business partner or friends in clear text message is going through thousands of machines (between sender and receiver before it reaches to intended recipients) these machines might read and saved the contents of email for future use?
- Many people think that name given in sender of the mail identifies who actually sends it.
- When you send a message through email, we cannot guarantee that it will be deliver to correct destination or received exactly what you sent. And even there is a no way of knowing that the message is received read and forwarded by attacker.
- Because of wide spared problem of email modifications, sending it to wrong destination by intermediate parties, email spoofing, we need a competing solution to overcome and address the issues of authentication, integrity and reliability of the messages between sender and receiver.
- The public key cryptography play an important role because of two keys used, only intended sender can decrypt the message using his public key as message encrypted using private key of the sender. The solution is called as Pretty Good Privacy (PGP) program/ software which provide the secrecy and non-repudiation of data sent over Internet especially by email.
- Pretty Good Privacy (PGP) is a popular open-source freely available software package/ techniques used to encrypt and decrypt email messages over the Internet.
- PGP is an e-mail security program written by Phil Zimmermann in 1991, PGP program become a de facto standard for e-mail security used to store the encrypted files so that it can be non-readable by other users or intruders.
- This program also be used to send an encrypted digital signature, let the receiver verify the sender's identity and know that the message was not changed or modified while transmission.
- Once the file is encrypted using PGP program only the intended recipient can decrypt it. Once message content digitally singed by sender, the sender guarantee to the recipients that message or file comes from valid sender and not by attacker.
- Digital signature functionality of PGP guarantees that the message or file come from the sender and not from an intruder.

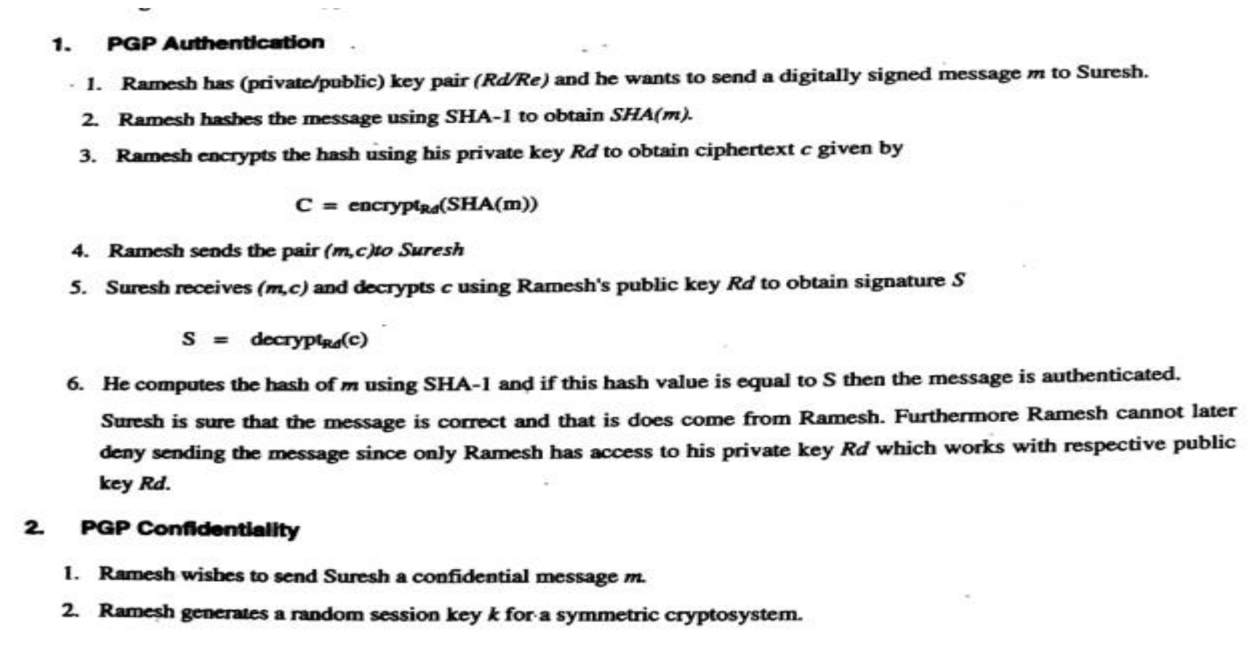
intruder.

6.4.1 Working of Pretty Good Privacy

- As mentioned earlier PGP uses the concept of public key cryptography, when user encrypts the plain text message using PGP, it first compress the plain text message.

- Because of data compression technique used to encrypt the plain text message which saves the transmission time and disk space and more important it strengthen the security.
- After data compression PGP generate the session key. Following table shows how PGP encrypt the message in order to achieve the confidentiality, integrity and non-repudiation.

Table 6.4.1 : Encryption and Decryption of Pretty Good Privacy



1. **PGP Authentication**

1. Ramesh has (private/public) key pair *(Rd/Re)* and he wants to send a digitally signed message *m* to Suresh.
2. Ramesh hashes the message using SHA-1 to obtain *SHA(m).*
3. Ramesh encrypts the hash using his private key *Rd* to obtain ciphertext *c* given by

$$C = \text{encrypt}_{Rd}(\text{SHA}(m))$$

4. Ramesh sends the pair *(m,c)to Suresh*
5. Suresh receives *(m,c)* and decrypts *c* using Ramesh's public key *Rd* to obtain signature *S*

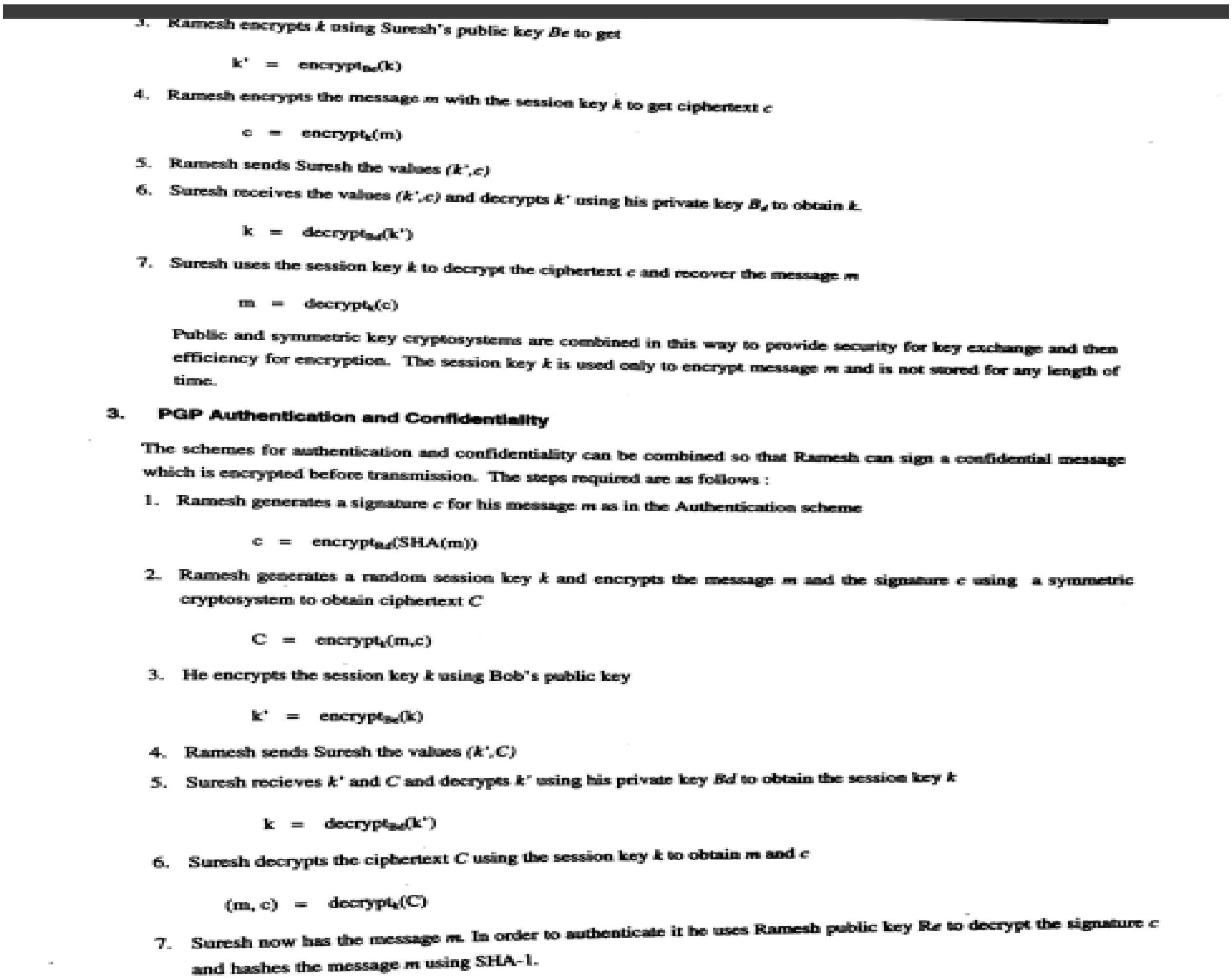
$$S = \text{decrypt}_{Rd}(c)$$

6. He computes the hash of *m* using SHA-1 and if this hash value is equal to S then the message is authenticated.

Suresh is sure that the message is correct and that is does come from Ramesh. Furthermore Ramesh cannot later deny sending the message since only Ramesh has access to his private key *Rd* which works with respective public key *Rd.*

2. **PGP Confidentiality**

1. Ramesh wishes to send Suresh a confidential message *m.*
2. Ramesh generates a random session key *k* for a symmetric cryptosystem.



3. Ramesh encrypts *k* using Suresh's public key *Be* to get

$$k' = \text{encrypt}_{Be}(k)$$

4. Ramesh encrypts the message *m* with the session key *k* to get ciphertext *c*

$$c = \text{encrypt}_k(m)$$

5. Ramesh sends Suresh the values *(k',c)*
6. Suresh receives the values *(k',c)* and decrypts *k'* using his private key B_d to obtain *k.*

$$k = \text{decrypt}_{Bd}(k')$$

7. Suresh uses the session key *k* to decrypt the ciphertext *c* and recover the message *m*

$$m = \text{decrypt}_k(c)$$

Public and symmetric key cryptosystems are combined in this way to provide security for key exchange and then efficiency for encryption. The session key *k* is used only to encrypt message *m* and is not stored for any length of time.

3. **PGP Authentication and Confidentiality**

The schemes for authentication and confidentiality can be combined so that Ramesh can sign a confidential message which is encrypted before transmission. The steps required are as follows :

1. Ramesh generates a signature *c* for his message *m* as in the Authentication scheme

$$c = \text{encrypt}_{Rd}(\text{SHA}(m))$$

2. Ramesh generates a random session key *k* and encrypts the message *m* and the signature *c* using a symmetric cryptosystem to obtain ciphertext *C*

$$C = \text{encrypt}_k(m,c)$$

3. He encrypts the session key *k* using Bob's public key

$$k' = \text{encrypt}_{Be}(k)$$

4. Ramesh sends Suresh the values *(k',C)*
5. Suresh recieves *k'* and *C* and decrypts *k'* using his private key *Bd* to obtain the session key *k*

$$k = \text{decrypt}_{Bd}(k')$$

6. Suresh decrypts the ciphertext *C* using the session key *k* to obtain *m* and *c*

$$(m, c) = \text{decrypt}_k(C)$$

7. Suresh now has the message *m.* In order to authenticate it he uses Ramesh public key *Re* to decrypt the signature *c* and hashes the message *m* using SHA-1.

MIME :

MIME stands for Multipurpose Internet Mail Extensions. It is used to extend the capabilities of Internet e-mail protocols such as SMTP. The MIME protocol allows the users to exchange various types of digital content such as pictures, audio, video, and various types of documents and files in the e-mail. MIME was created in 1991 by a computer scientist named Nathan Borenstein at a company called Bell Communications.

Need of MIME Protocol

MIME protocol is used to transfer e-mail in the computer network for the following reasons:

- The MIME protocol supports multiple languages in e-mail, such as Hindi, French, Japanese, Chinese, etc.
- Simple protocols can reject mail that exceeds a certain size, but there is no word limit in MIME.
- Images, audio, and video cannot be sent using simple e-mail protocols such as SMTP. These require MIME protocol.
- Many times, emails are designed using code such as HTML and CSS, they are mainly used by companies for marketing their product. This type of code uses MIME to send email created from HTML and CSS.

S/MIME?

Secure/Multipurpose Internet Mail Extension (S/MIME) is an industry-standard for email encryption and signature that is commonly used by businesses to improve email security. S/MIME is supported by the majority of corporate email clients.

S/MIME encrypts and digitally signs emails to verify that they are verified and that their contents have not been tampered with.

S/MIME Certificate Characteristics

You receive a slew of cryptographic security features when you use an S/MIME certificate for email apps.

1. **Authentication** − It refers to the verification of a computer user's or a website's identity.
2. **Message consistency** − This is a guarantee that the message's contents and data have not been tampered with. The message's secrecy is crucial. The decryption procedure entails checking the message's original contents and guaranteeing that they have not been altered.
3. **Use of digital signatures that invoke non-repudiation** − This is a circumstance in which the original sender's identity and digital signatures are validated so that there is no doubt about it.
4. **Protection of personal information** − A data breach cannot be caused by an unintentional third party.
5. **Encryption is used to protect data** − It relates to the procedures described above, in which data security is ensured by a mix of public and private keys representing asymmetric cryptography.

Support for S/MIME:

Some of the most popular email programs that support S/MIME are listed below.

- iPhone iOS Mai
- Apple Mail
- Gmail IBM Notes
- Mozilla Thunderbird MailMate Microsoft Outlook or Outlook on the Web
- CipherMail

Features of MIME Protocol:

- It supports multiple attachments in a single e-mail.
- It supports the non-ASCII characters.
- It supports unlimited e-mail length.
- It supports multiple languages.

Advantage of the MIME:

- The MIME protocol has the following advantages:
- It is capable of sending various types of files in a message, such as text, audio, video files.
- It also provides the facility to send and receive emails in different languages like Hindi, French, Japanese, Chinese etc.
- It is capable of sending the information contained in an email regardless of its length.
- It assigns a unique id to all e-mails.