Security Protocols

Title	Page Number
1. **Introduction**	3
2. **IP Security (IPsec)**	4
2.1. IPsec Components	4
2.2. Applications of IPsec	5
2.3. Benefits of IPsec	6
2.4. IPsec Modes	7
3. **Secure Sockets Layer (SSL) and Transport Layer Security (TLS)**	9
3.1. TLS Protocols	10
3.2. SSL/TLS Applications	14
4. **The Secure Shell (SSH)**	16
5. **Comparing IPsec, SSL/TLS, and SSH**	17
6. **Exercises**	18

Learning Objective

After studying this chapter, you should be able to:

- Investigate how to maintain the security of traffics traveling on the network wires.
- Investigate how IPsec, SSL/TLS and SSH provide security at the network, transport, and application layers.
- Study major applications of these protocols in e-commerce and remote administration.
- Compare between these protocols.

1.Introduction

One of the weaknesses of the original Internet Protocol (IP) is that it lacks any sort of general-purpose mechanism for ensuring the authenticity and privacy of data as it is passed over the internetwork. Since IP datagrams must usually be routed between two devices over unknown networks, any information in them is subject to being intercepted and even possibly changed. With the increased use of the Internet for critical applications, security enhancements were needed for IP. To this end, a set of protocols was developed.

Network security (or web security) protocols are used to protect computer data and communication in transit. A number of approaches to providing network security are possible. The various approaches that have been considered are similar in the services they provide and, to some extent, in the mechanisms that they use, but they differ with respect to their scope of applicability and their relative location within the TCP/IP protocol stack.

One way to provide network security is to use IP security (IPsec). The advantage of using IPsec is that it is transparent to end users and applications and provides a general-purpose solution. Another relatively general-purpose solution is to implement security just above TCP. The foremost example of this approach is the Secure Sockets Layer (SSL) and the follow-on Internet standard known as Transport Layer Security (TLS). Application-specific security services are embedded within the particular application. The Secure Shell (SSH) and Pretty Good Privacy are examples of this architecture. The advantage of this approach is that the service can be tailored to the specific needs of a given application.

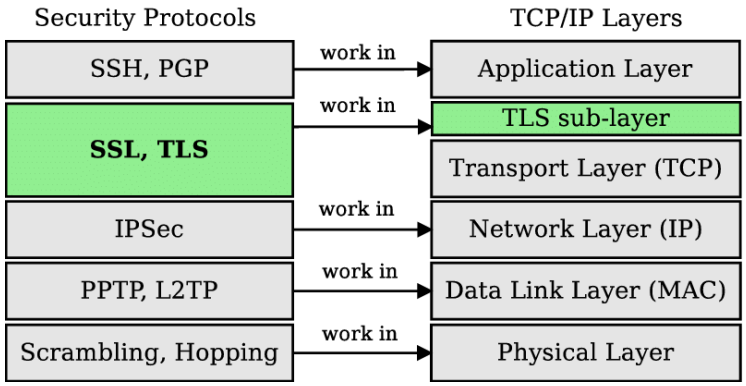


Figure (6.1): Relative Location of Security Protocols in TCP/IP Stack

2.IP Security (IPsec)

IPsec is not a single protocol, but rather a set of services and protocols that provide a complete security solution for an IP network. These services and protocols combine to provide various types of protection. Since IPsec works at the IP layer, it can provide these protections for any higher-layer TCP/IP application or protocol without the need for additional security methods, which is a major strength. Some of the kinds of protection services offered by IPsec include the following:

- Access Control
- Connectionless integrity
- Data origin authentication
- Rejection of replayed packets
- Confidentiality (encryption)
- Limited traffic flow confidentiality

Since IPsec is actually a collection of techniques and protocols, it is not defined in a single Internet standard. Instead, a collection of RFCs defines the architecture, services, and specific protocols used in IPsec. Some of the most important of these are:

RFC 4301, "Security Architecture for the Internet Protocol"

RFC 4302, "IP Authentication Header"

RFC 4303, "IP Encapsulating Security Payload"

RFC 7296, "Internet Key Exchange (IKEv2) Protocol"

2.1.IPsec Components

IPsec contains the following elements:

Authentication Header (AH): AH provides authentication, integrity, and optional anti-replay protection. The authentication header is inserted into the packet between the IP header and any subsequent packet contents. The payload is not touched. Although AH protects the packet's origin, destination, and contents from being tampered with, the identity of the sender and receiver is known. In addition, AH does not protect the data's confidentiality. If data is intercepted and only AH is used, the message contents can be read. Because message authentication is provided by

ESP, the use of AH is deprecated. It is included in IPsecv3 for backward compatibility but should not be used in new applications.

Encapsulation Security Payload (ESP): ESP provides all four security features of IPsec. These are confidentiality, integrity, origin authentication, and anti-replay protection. Confidentiality ensures data is encrypted. Integrity ensures data in transit has not been tampered with. Origin authentication ensures the remote peers are who they claim to be and anti-replay protection will ensure duplicated traffic is not accepted which would prevent DOS attacks, as well as spoofed traffic. The ESP header is inserted into the packet between the IP header and any subsequent packet contents. However, because ESP encrypts the data, the payload is changed. ESP does not encrypt the ESP header, nor does it encrypt the ESP authentication. ESP is more widely deployed than AH, because ESP provides all the benefits of IPsec.

Internet Key Exchange (IKE) Protocol: IKE is a network security Protocol designed to allow two devices to dynamically exchange Encryption Keys and negotiate Security Associations (SA). IKE Security Associations (SA) can be established dynamically and removed at a negotiated time period. IKE is made from the combination of many protocols, the most important one is Internet Security Association and Key Management Protocol (ISAKMP), which provide a framework for entity authentication and key exchange.

2.2.Applications of IPsec

IPsec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet. Examples of its use include:

- Secure branch office connectivity over the Internet: A company can build a secure virtual private network over the Internet or over a public WAN.
- Secure remote access over the Internet: An end user whose system is equipped with IP security protocols can gain secure access to a company network.
- Establishing extranet and intranet connectivity with partners: IPsec can be used to secure communication with other organizations.

- Enhancing electronic commerce security: Even though some Web and electronic commerce applications have built-in security protocols, the use of IPsec enhances that security. IPsec can add an additional layer of security to whatever is provided at the application layer.

The principal feature of IPsec that enables it to support these varied applications is that it can encrypt and/or authenticate all traffic at the IP level. Thus, all distributed applications (including remote logon, client/server, email, file transfer, Web access, and so on) can be secured.

2.3.Benefits of IPsec

Some of the benefits of IPsec:

- When IPsec is implemented in a firewall or router, it provides strong security that can be applied to all traffic crossing the perimeter. Traffic within a company or workgroup does not incur the overhead of security-related processing.
- IPsec in a firewall is resistant to bypass if all traffic from the outside must use IP and the firewall is the only means of entrance from the Internet into the organization.
- IPsec is below the transport layer (TCP, UDP) and so is transparent to applications. There is no need to change software on a user or server system when IPsec is implemented in the firewall or router. Even if IPsec is implemented in end systems, upper-layer software, including applications, is not affected.
- IPsec can be transparent to end users. There is no need to train users on security mechanisms, issue keying material on a per-user basis, or revoke keying material when users leave the organization.
- IPsec can provide security for individual users if needed. This is useful for offsite workers and for setting up a secure virtual sub network within an organization for sensitive applications.
- IPsec can play a vital role in the routing architecture required for internetworking. The following are examples of the use of IPsec in protecting routing applications: IPsec can assure that:
  - A router or neighbor advertisement comes from an authorized router.

- A redirect message comes from the router to which the initial packet was sent.
- A routing update is not forged.

2.4.IPsec Modes

Both AH and ESP support two modes of use: transport and tunnel mode.

Transport Mode: Transport mode provides protection primarily for upper-layer protocols. That is, transport mode protection extends to the payload of an IP packet. Examples include a TCP or UDP segment or an ICMP packet, all of which operate directly above IP in a host protocol stack. Typically, transport mode is used for end-to-end communication between two hosts (e.g., a client and a server, or two workstations). ESP in transport mode encrypts and optionally authenticates the IP payload but not the IP header. AH in transport mode authenticates the IP payload and selected portions of the IP header.

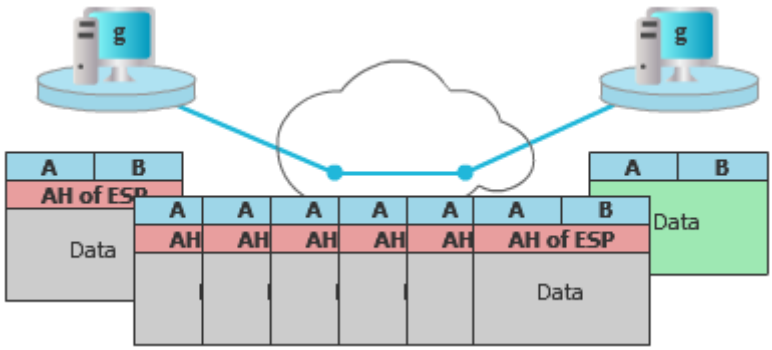
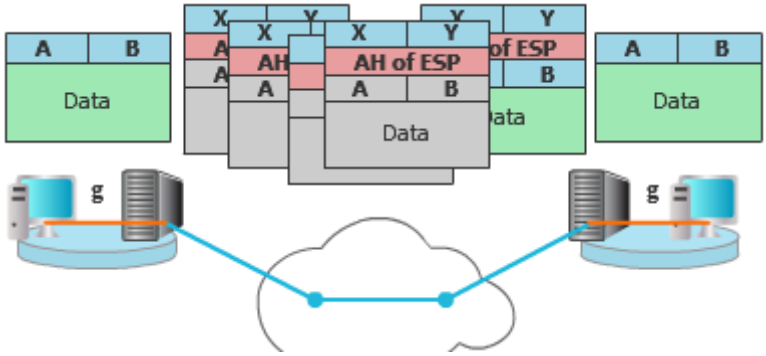


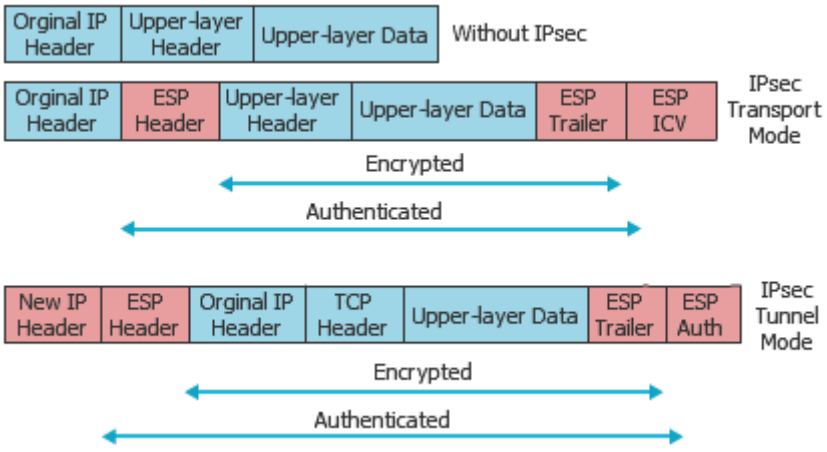
Figure (6.2): IPsec Transport Mode

Tunnel Mode: Tunnel mode provides protection to the entire IP packet. To achieve this, after the AH or ESP fields are added to the IP packet, the entire packet plus security fields is treated as the payload of new outer IP packet with a new outer IP header. The entire original, inner, packet travels through a tunnel from one point of an IP network to another, no routers along the way are able to examine the inner IP header. Because the original packet is encapsulated, the new, larger packet may have totally different source and destination addresses, adding to the security. Tunnel mode is used when one or both ends are a security gateway, such as a firewall or router that implements IPsec. With tunnel mode, a number of hosts on

networks behind firewalls may engage in secure communications without implementing IPsec. The unprotected packets generated by such hosts are tunneled through external networks by tunnel mode set up by the IPsec software in the firewall or secure router at the boundary of the local network. ESP in tunnel mode encrypts and optionally authenticates the entire inner IP packet, including the inner IP header. AH in tunnel mode authenticates the entire inner IP packet and selected portions of the outer IP header.



Figure(6.3): IPsec Tunnel Mode



Figure(6.4): IPsec Authentication and Encryption

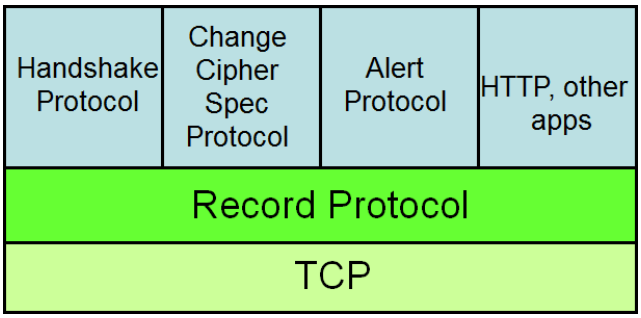
Tunneling example: Here is an example of how tunnel mode IPsec operates:

- Host A on a network generates an IP packet with the destination address of host B on another network.
- This packet is routed from the originating host to a firewall or secure router at the boundary of A's network.
- The firewall filters all outgoing packets to determine the need for IPsec processing.
- If this packet from A to B requires IPsec, the firewall performs IPsec processing and encapsulates the packet with an outer IP header. The source IP address of this outer IP packet is this firewall, and the destination address may be a firewall that forms the boundary to B's local network.
- This packet is now routed to B's firewall, with intermediate routers examining only the outer IP header.
- At B's firewall, the outer IP header is stripped off, and the inner packet is delivered to B.

3.Secure Sockets Layer (SSL) and Transport Layer Security (TLS)

One of the most widely used security services is the Secure Sockets Layer (SSL) and the follow-on Internet standard, the Transport Layer Security (TLS) Protocol (RFC 4346). TLS is a general-purpose service implemented as a set of protocols that rely on TCP. Most browsers come equipped with TLS, and most Web servers have implemented the protocol.

TLS is designed to make use of TCP to provide a reliable end-to-end secure service. TLS is not a single protocol but rather two layers of protocols, as illustrated in Figure(6.5). The Record Protocol provides basic security services to various higher-layer protocols. In particular, HTTP, which provides the transfer service for Web client/server interaction, can operate on top of TLS. Three higher-layer protocols are defined as part of TLS: the Handshake Protocol, the Change Cipher Spec Protocol, and the Alert Protocol. These TLS-specific protocols are used in the management of TLS exchanges.



Figure(6.5): SSL/TLS Protocol Stack

3.1.TLS Protocols

Record Protocol: The TLS Record Protocol provides two services for TLS connections:

- Confidentiality: The Handshake Protocol defines a shared secret key that is used for symmetric encryption of TLS payloads.
- Message integrity: The Handshake Protocol also defines a shared secret key that is used to form a message authentication code (MAC).

The steps of Record protocols operation are:

- Fragmentation: Each upper-layer message is fragmented into blocks of 214 bytes or less.
- Compression is optionally applied.
- Compute a message authentication code over the compressed data.
- The compressed message plus the MAC are encrypted using symmetric encryption.
- Prepend a header, which includes version and length fields.
- Submit the resulting unit in a TCP segment.
- Received data are decrypted, verified, decompressed, and reassembled, then delivered to higher-level users.

The Upper Layer Protocols: The Upper Layer protocols for TLS consist of three protocols, namely, the TLS Handshake, the TLS Change-Cipher-Spec and TLS Alert. The TLS Change-Cipher-Spec protocol consists of a single message of 1 byte. The main function of this protocol is to update the cipher suite i.e. a suite of algorithms for performing encryption and decryption, on the connection. The TLS Alert protocol consists of two bytes and its main function is management and error messages. The most important protocol in the Upper Layer protocols is the TLS Handshake protocol.

The Handshake Protocol: The most complex part of TLS is the Handshake Protocol. The Handshake Protocol is used before any application data are transmitted. The security goals of the handshake protocol are:

- Entity authentication of participating parties: Participants are called 'client' and 'server'. Server nearly always authenticated, client more rarely, which is appropriate for most web applications.
- Establishment of a fresh, shared secret: Shared secret used to derive further keys for confidentiality and authentication in record protocol.
- Secure ciphersuite negotiation: Encryption and hash algorithms that are used in record protocol and authentication and key establishment methods.

The Handshake Protocol consists of a series of messages exchanged by client and server. Figure(6.6) shows the initial exchange needed to establish a logical connection between client and server. The exchange can be viewed as having four phases.

Phase 1: Establish Security Capability

The purpose of this phase is to negotiate the best parameters that the two parties are able to support. This phase starts by the client sending a Client-Hello message which consists of the following parameters:

- The highest TLS version.
- Nonce which consists of a 4-byte time stamp and a 28-byte random number. The nonce is used during the key exchange to prevent replay attack.
- A session identifier (ID), which is used to update the connection parameters.
- A cipher suite, which is a list of cryptographic and hash algorithms the client is able to support.

- A compression method which lists all the compression methods the client is able to support.

The server replies by issuing a Server-Hello message which selects the best parameters the two parties are able to support.

Phase 2: Server Authentication and Key Exchange

The purpose of this phase is to authenticate the server to the client, if it is needed. The authentication is achieved by means of a public key certificate. In addition, the server sends the necessary parameters for key exchange. The server sends the following messages:

- Server public key certificate (optional).
- Server key exchange (optional) which carries the server's public key part when using the Diffie-Hellman (DH) algorithm for key exchange.
- Certificate request (optional), in which the server can ask the client for her public key certificate.
- Server Hello Done, which indicates the end of phase 2.

Phase 3: Client Authentication and Key Exchange

The purpose of this phase is to authenticate the client to the server, if it is needed. The authentication is achieved by means of a public key certificate. In addition, the client sends its part from the parameters for key exchange. The client first validates the server's certificate and then sends the following messages:

- Public key certificate: If the server requests the client's certificate in phase 2, the client provides the certificate if available, else the client sends a no-certificate Alert message.
- Client key exchange: This could be the client's DH public key, or a session key encrypted by the server's public key.
- Certificate verification (optional): In this message the client uses her private key to sign a message to the server. In this signature, the client proves that she has the private key which corresponds to the public key in the client's certificate.

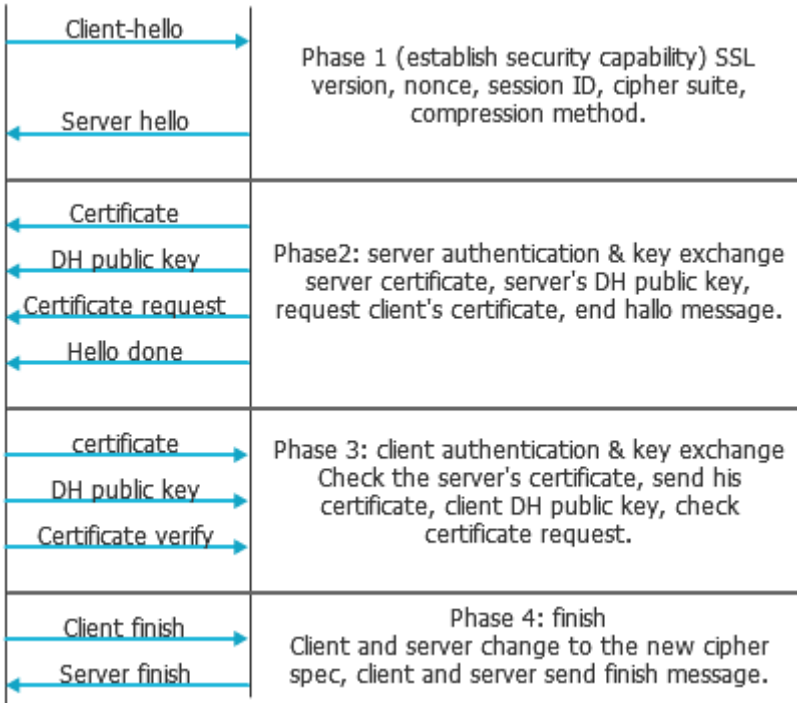


Figure (6.6): Handshake Protocol

Phase 4: Finish

In this phase the client and server check the new settings.

- The client changes to the new Cipher-Spec and sends a Change-Cipher-Spec message to the server.
- The client sends a Finish message under the new algorithm and keys.
- The server changes to the new Cipher-Spec and sends a Change-Cipher-Spec message to the client.
- The server sends a Finish message under the new algorithm and keys.

When this phase is complete the Handshake protocol is complete and the two parties are ready to exchange the data in a secure form.

3.2.SSL/TLS Applications

HTTPS (HTTP over SSL) refers to the combination of HTTP and SSL to implement secure communication between a Web browser and a Web server. There is no fundamental change in using HTTP over either SSL or TLS, and both implementations are referred to as HTTPS. The HTTPS capability is built into all modern Web browsers. Its use depends on the Web server supporting HTTPS communication. The principal difference seen by a user of a Web browser is that URL (uniform resource locator) addresses begin with https:// rather than http://. A normal HTTP connection uses port 80. If HTTPS is specified, port 443 is used, which invokes SSL.

We will take a very familiar application for SSL/TLS, which is protecting Web commerce transactions, and highlight some security issues. Most of these issues are applicable in other SSL/TLS applications:

- **The Authenticity of the Client**: SSL/TLS provides authentication for the client by using her public key certificate. In SSL/TLS, the public key certificate is optional and typically, it is difficult for a normal client to have a pair of public/private keys and the public key has to be certificated by a trusted third party. As a result, there is usually no authentication for the client. Anyone who knows the cardholder's credit card details is able to complete the transaction on behalf of the cardholder.
- **The Authenticity and Legitimacy of the Merchant**: SSL/TLS provides authentication for the merchant from his public key certificate. SSL/TLS does not provide any means for verifying the merchant's legitimacy. In fact any attacker, who has a Web site and a public key certificate, is able to make an SSL/TLS connection and deceive the client. In addition, does client understand meaning of certificate expiry and other security warnings? Does client software actually check complete certificate chain? Does the name in certificate match the URL of e-commerce site? Does the user check this? Is the site the one the client thinks it is? Determining the legitimacy of the merchant is completely the responsibility of the client.

- **Non-repudiation**: Neither the merchant nor the client has any proof of the transaction when using SSL/TLS. In fact, both of them are able to repudiate the transaction partially i.e. details of the transaction such as the price, goods specifications and so on, or totally, i.e. the transaction has not been carried out. The two parties could use other methods to confirm the transaction such as email or by post, but it is impossible to link this method with the transaction, which is carried out by using SSL/TLS.
- **Data Confidentiality**: SSL/TLS use encryption to maintain data confidentiality for the exchanged data. After the transaction, no guarantees about what happens to client data (including credit card details) may be stored on insecure server.
- **Data Integrity**: SSL/TLS maintains the data integrity for the exchanged data. SSL/TLS does not provide any means of data integrity after the transaction.
- **Freshness of the Transaction**: SSL/TLS provides freshness for the exchanged messages against third party. As we have seen, SSL/TLS session uses a random number (nonce) and a timestamp to prevent replay attacks. This protection is provided only during the transaction. Because the merchant knows all the payment and order information, therefore, it is possible to make other transactions using the payment information from this transaction.
- **The Security of the Client's Computer**: SSL/TLS is a temporary session. While the client does not have a pair of public/private keys, which is the typical case, the client can, however, use any Web browser, which supports SSL/TLS and typically most Web browsers support SSL/TL. If there is no malicious program, such as a Trojan horse, trap door, etcetera, which is capable of recording the user information during the transaction, whilst the user fills in the information fields, then the transaction is safe.
- **The Security of the Merchant's Computer**: By using SSL/TLS, the merchant knows all the payment information, which is enough to conduct any further transactions. The merchant usually stores the transactions information in his database, so any compromise of this database is catastrophic. If an attacker has access to this database, then he/she can choose any costumer's details to process spurious transactions.

4.The Secure Shell (SSH)

Secure Shell (SSH) is a protocol for secure network communications designed to be relatively simple and inexpensive to implement. The initial version was focused on providing a secure remote logon facility to replace Telnet and other remote logon schemes that provided no security. SSH also provides a more general client/server capability and can be used for such network functions as file transfer and e-mail. A new version, SSH2, is documented as a proposed standard in IETF RFCs 4250 through 4256.

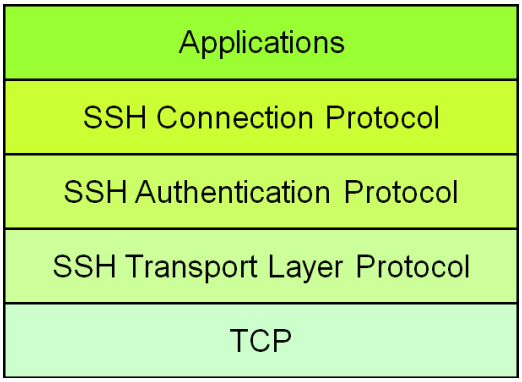
SSH is organized as three protocols that typically run on top of TCP:

1. Transport Layer Protocol: Provides server authentication, data confidentiality, and data integrity. The transport layer may optionally provide compression.
2. User Authentication Protocol: Authenticates the user to the server.
3. Connection Protocol: Multiplexes multiple logical communications channels over a single, underlying SSH connection.

SSH client and server applications are widely available for most operating systems. It has become the method of choice for remote login and X tunneling and is rapidly becoming one of the most pervasive applications for encryption technology. SSH provides security at Application layer. That is, it only covers traffic that explicitly protected.

Some examples of SSH applications are:

- Anonymous ftp for software updates: No client authentication is needed, but clients want to be sure of origin and integrity of software.
- Secure ftp: E.g. upload of webpages to webserver using sftp. Server now needs to authenticate clients. Username and password may be sufficient, transmitted over secure SSH transport layer protocol.
- Secure remote administration: SysAdmin (client) sets up terminal on remote machine. SysAdmin password is protected by SSH transport layer protocol, and SysAdmin commands are protected by SSH connection protocol.
- Virtual Private Network.



Figure(6.7): SSH Architecture

5.Comparing IPsec, SSL/TLS, and SSH

- All three have initial (authenticated) key establishment then key derivation, IKE in IPsec, Handshake Protocol in SSL/TLS, and Authentication Protocol in SSH. They use Asymmetric encryption in this phase.
- All three protect ciphersuite negotiation.
- All three use keys established to build a 'secure channel'. They use symmetric encryption in this phase.
- They operate at different network layers. This brings pros and cons for each protocol suite, and naturally support different application types.
- All three can be used to build VPNs.
- All practical, but not simple. Complexity leads to vulnerabilities, and makes configuration and management harder. Furthermore, complexity can create computational bottlenecks.

Security of all three undermined by:

- Implementation weaknesses.
- Weak server platform security, such as Worms, malicious code, rootkits,…
- Weak user platform security such as Keystroke loggers, malware,…
- Limited deployment of certificates and infrastructure to support them, especially client certificates.
- Lack of user awareness and education: Users click-through on certificate warnings. Users fail to check URLs. Users send sensitive account details to bogus websites ("phishing") in response to official-looking e-mail.

Exercises:

True or False

Question	True	False
1. The principal feature of IPsec is that it can encrypt and/or authenticate all traffic at the IP level.		
2. Transport mode provides protection to the entire IP packet.		
3. An end user whose system is equipped with IP security protocols can make a local call to an ISP and gain secure access to a company network.		
4. By implementing security at the IP level an organization can ensure secure networking not only for applications that have security mechanisms but also for the many security ignorant applications.		
5. IPSec can guarantee that all traffic designated by the network administrator is authenticated but cannot guarantee that it is encrypted.		
6. IPsec is executed on a packet-by-packet basis.		
7. Secure Sockets Layer (SSL) is an Internet standard that evolved from a commercial protocol known as Transport Layer Security (TLS).		
8. The Handshake Protocol is the simplest of the four TLS-specific protocols that use the TLS Record Protocol.		
9. The World Wide Web is fundamentally a client/server application running over the Internet and TCP/IP intranets.		
10. One way to classify Web security threats is in terms of the location of the threat: Web server, Web browser, and network traffic between browser and server.		
11. The Change Cipher Spec Protocol is one of the four TLS-specific protocols that use the SSL Record Protocol.		

12. The SSL Record Protocol is used before any application data is transmitted. L SEP		
13. The first element of the CipherSuite parameter is the key exchange method. L SEP		

Multiple Choice Questions

1. Authentication applied to the entire original IP packet is _____ .
   - **A.** security mode
   - **B.** cipher mode
   - **C.** tunnel mode
   - **D.** transport mode

2. Authentication applied to all of the packet except for the IP header is _____.
   - **A.** tunnel mode
   - **B.** transport mode
   - **C.** association mode
   - **D.** security mode

3. The _____ mechanism assures that a received packet was in fact transmitted by the party identified as the source in the packet header and assures that the packet has not been altered in transit.
   - **A.** confidentiality
   - **B.** authentication
   - **C.** security
   - **D.** key management

4. A _____ is a one way relationship between a sender and a receiver that affords security services to the traffic carried on it.
   - **A.** SAD
   - **B.** SPD
   - **C.** SA
   - **D.** SPI

5. The SSL Internet standard version is called _____ .

A. SSH

B. HTTP

C. SLP

D. TLS

6. The most complex part of TLS is the _____ .

A. SSL Record Protocol

B. Handshake Protocol

C. Change Cipher Spec Protocol

D. Alert Protocol

7. _____ attacks include impersonating another user, altering messages in transit between client and server and altering information on a Web site.

A. Active

B. Passive

C. Shell

D. Psuedo

8. _____ provides secure, remote logon and other secure client/server facilities.

A. SLP

B. HTTPS

C. TLS

D. SSH

9. An arbitrary byte sequence chosen by the server to identify an active or resumable session state is a _____ .

A. peer certificate

B. session identifier

C. compression

D. cipher spec

10. The _____ is used to convey TLS-related alerts to the peer entity.

A. Change Cipher Spec Protocol

B. Alert Protocol

C. SSL Record Protocol

D. Handshake Protocol

11. With each element of the list defining both a key exchange algorithm and a CipherSpec, the list that contains the combination of cryptographic algorithms supported by the client in decreasing order of preference is the _____ .

A. CipherSuite

B. Random

C. Session ID

D. Version

12. Phase _____ of the Handshake Protocol establishes security capabilities.

A. 4

B. 1

C. 2

D. 3

13. _____ is organized as three protocols that typically run on top of TCP for secure network communications and are designed to be relatively simple and inexpensive to implement.

A. SSL

B. SSH

C. TLS

D. SSI

Number of the Question	True	False
1	✓	
2		✓
3	✓	
4	✓	
5		✓
6	✓	
7		✓
8		✓
9	✓	
10	✓	
11	✓	
12		✓
13	✓	

Number of the Question	Answer
1	C
2	B
3	B
4	C
5	D
6	B
7	A
8	D
9	B
10	B
11	A
12	B
13	B