

Unit-VI

Network Security and Public Key Infrastructure

Digital Certificates

A digital certificate is a certificate issued by a Certificate Authority (CA) to verify the identity of the certificate holder. The CA issues an encrypted digital certificate containing the applicant's public key and a variety of other identification information. Digital certificate is used to attach public key with a particular individual or an entity. It is based on trust or chain of trust. The trusted third party is a certificate authority, an entity that issues digital certificate. It verifies the digital signature is truly signed by the claimed signer. The mostly used standard format for digital certificates is X.509.

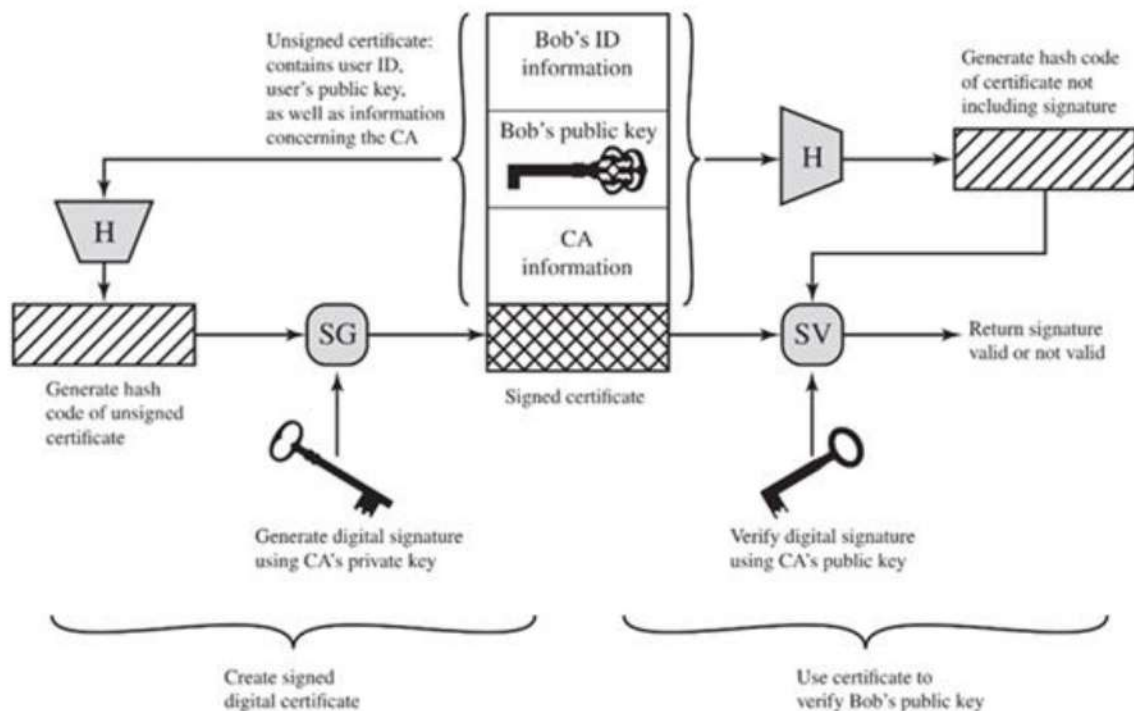
X.509 Certificate

X.509 is a standard defining the format of public key certificates. An X.509 certificate is a digital certificate that uses the widely accepted international X.509 public key infrastructure (PKI) standard to verify that a public key belongs to the hostname/domain, organization, or individual contained within the certificate. The X.509 certificate is either signed by a publicly trusted Certificated Authority or self-signed. When a certificate is signed by a trusted certificate authority, or validated by other means, someone holding that certificate can rely on the public key it contains to establish secure communications with another party, or validate documents digitally signed by the corresponding private key.

A typical X.509 standard digital certificate has following format:

Version	Version of X.509 to which the Certificate conforms
Serial Number	A number that uniquely identifies the Certificate
Signature Algorithm ID	The names of the specific Public Key algorithms that the CA has used to sign the Certificate (Ex.- RSA with SHA-1)
Issuer (CA) X.500 Name	The identity of the CA Server who issued the Certificate
Validity Period	The period of time for which the Certificate is valid with start date and expiration date
Subject X.500 Name	The owner's identity with X.500 Directory format (Ex.- cn=ausser, ou=SP, o=Alphawest)
Subject Public Key Info	The Public Key of the owner of the Certificate and the specific Public Key algorithms associated with the Public Key
Algorithm ID	
Public Key Value	
Issuer Unique ID	Information used to identify the issuer of the Certificate
Subject Unique ID	Information used to identify the Owner of the Certificate
Extension	Additional information like Alternate name, CRL Distribution Point (CDP)
CA Digital Signature	The actual digital signature of the CA

Generation and Use of Public-Key Certificate



Certificate Life Cycle Management

The life cycle of a certificate can be broken into six distinct stages:

1. **Certificate Enrollment:** Certificate enrollment is initiated by a user request to the appropriate CA. This is a cooperative process between a user and the CA. The enrollment request contains the public key and enrollment information. Once a user requests a certificate, the CA verifies information based on its established policy rules, creates the certificate, posts the certificate, and then sends an identifying certificate to the user.
2. **Certificate Validation:** When a certificate is used, the certificate status is checked to verify that the certificate is still operationally valid. During the validation process, the CA checks the status of the certificate and verifies that the certificate is not on a certificate revocation list (CRL).
3. **Certificate Revocation:** A certificate issued by a CA includes an expiration date that defines how long the certificate is valid. If a certificate needs to be revoked before that date, the CA can be instructed to add the certificate to its CRL. Reasons a certificate might need to be revoked include the certificate being lost or compromised, or the person the certificate was issued to leaving the company.
4. **Certificate Renewal:** When a certificate reaches its expiration date, if the certificate is allowed to be renewed, a user can ask the CA to renew the certificate. This might happen automatically, or it might require user intervention. When renewing a certificate you must choose whether or not to generate new public and private keys.
5. **Certificate Destruction:** When a certificate is no longer in use, the certificate and any backup copies or archived copies of the certificate should be destroyed, along with the private key associated with the certificate. This helps ensure that the certificate is not compromised and used.

6. **Certificate Auditing:** The CA performs certificate auditing, and the process varies depending on the CA and the management tools available to it. Certificate auditing involves tracking the creation, expiration, and revocation of certificates. In certain instances, it can also track each successful use of a certificate.

Public Key Infrastructure (PKI)

The public key infrastructure is defined as the set of hardware, software, people, policies and procedures needed to create, manage, store, distribute and revoke digital certificates based on asymmetric cryptography. The principal objective for developing a PKI is to enable secure, convenient, and efficient acquisition of public keys.

PKIX (Public Key Infrastructure X.509) model consists of following elements:

- **End Entity:** A generic term used to denote end users, devices (e.g., servers, routers), or any other entity that can be identified in the subject field of a public-key certificate. End entities typically consume and/or support PKI-related services.
- **Certificate Authority (CA):** The issuer of certificates and (usually) certificate revocation lists (CRLs). It may also support a variety of administrative functions, although these are often delegated to one or more Registration Authorities.
- **Registration Authority (RA):** An optional component that can assume a number of administrative functions from the CA. The RA is often associated with the end entity registration process but can assist in a number of other areas as well.
- **CLR Issuer:** An optional component that a CA can delegate to publish CRLs.
- **Repository:** A generic term used to denote any method for storing certificates and CRLs so that they can be retrieved by end entities.

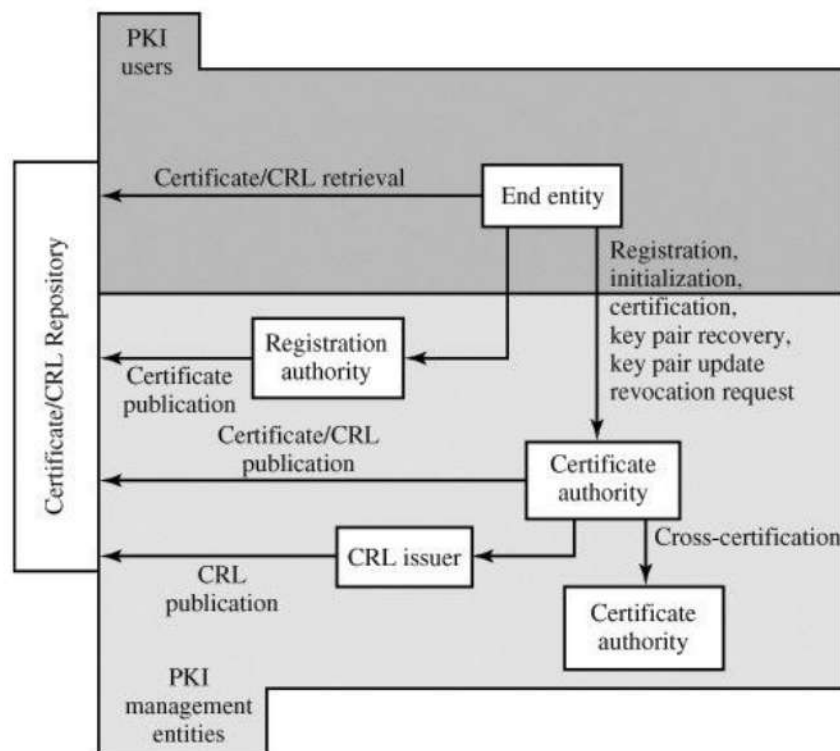


Fig: PKIX architecture model

PKIX Management Functions

- **Registration:** It is the process where an end entity (subject) makes itself known to a CA. Usually, this is via an RA.
- **Initialization:** This deals with the basic problems, such as the methodology of verifying that the end-entity is taking to the right CA. We have seen how this can be tackled.
- **Certification:** In this step, the CA creates a digital certificates for the end entity and returns it to the end entity, maintains a copy for its own records, and also copies it in public directories, if required.
- **Key Pair Recovery:** Keys used for encryption may be required to recover at a later date for decrypting some old documents. Key archival and recovery services can be provided by a CA or by an independent key recovery system.
- **Key Generation:** PKIX specifies that the end entity should be able to generate private and public key pair, or the CA/RA should be able to do this for the end-entity.
- **Key Update:** This allows a smooth transition from one expiring key pair to a fresh one, by the automatic renewal of digital certificates. However, there is a provision for manual digital certificate renewal request and response.
- **Cross Certification:** Helps in establishing trust models, so that end entities that are certified by different CAs can cross verify each other.
- **Revocation:** PKIX provides support for the checking of the certificate status in two modes: online and offline.

Pretty Good Privacy (PGP)

PGP is an open-source, freely available software package for email security.

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting, and decrypting texts, e-mails, files, directories, and whole disk partitions and to increase the security of e-mail communications.

The actual operation of PGP consists of 5 services: authentication, confidentiality, compression, e-mail compatibility and segmentation.

1. Authentication:

PGP authentication through the use of digital signature. A hash code of message is created using SHA-1. The hash code is encrypted using DSS or RSA with the sender's private key including message. The receiver uses RSA with the sender's public key to decrypt and recover the hash code. The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

2. Confidentiality:

PGP provides confidentiality through the use of symmetric block encryption. A message is encrypted using CAST-128 or IDEA or 3DES with one-time session key generated by the sender. The session key is encrypted using RSA with the recipient's public key and included with the message. The receiver uses RSA with its private key to decrypt and recover the session key. The session key is used to decrypt the message.

3. Compression:

PGP compresses the message after applying the signature but before encryption. This has the benefit of saving space both for email transmission and for file storage. Message encryption is applied after compression to strengthen cryptographic security.

The compression algorithm used ZIP algorithm.

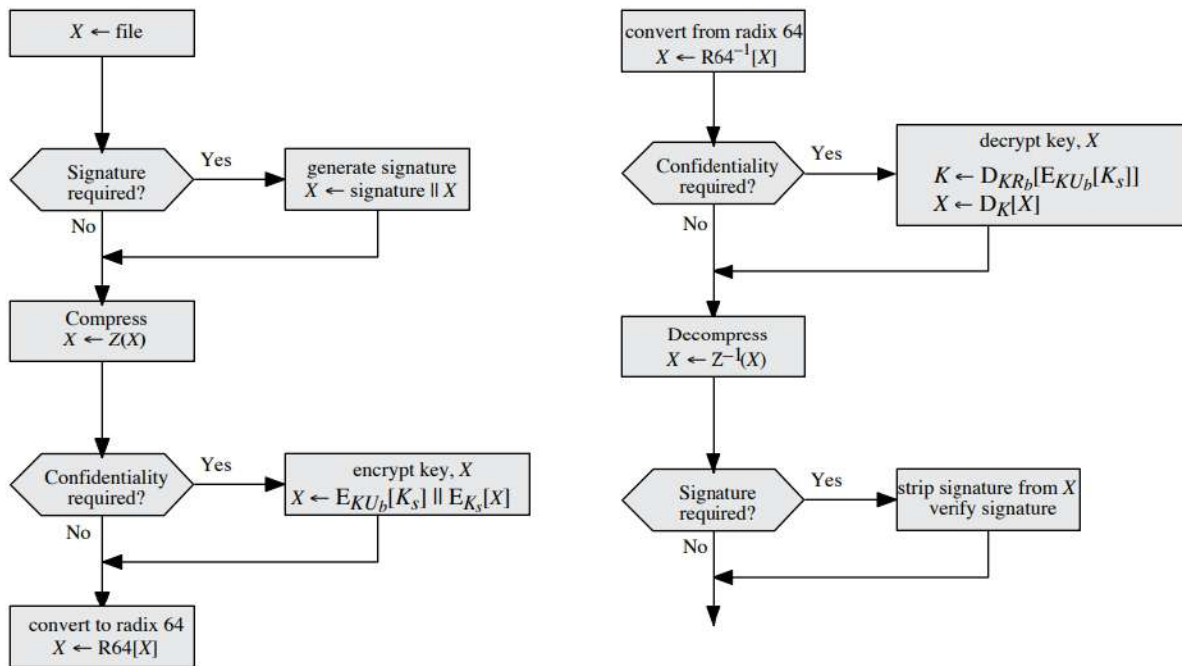
4. E-mail Compatibility:

The scheme used for e-mail compatibility is radix-64 conversion. To provide transparency for email application, an encrypted message may be converted to an ASCII string using radix 64-conversion. The use of radix 64 expands a message by 33%.

5. Segmentation:

Email facilities often are restricted to a maximum length. To accommodate this, PGP automatically subdivides a message that is too large into segments that are small enough to send via e-mail. The segmentation is done after all of the other processing, including the radix-64 conversion.

Transmission and Reception of PGP Messages



(a) Generic Transmission Diagram (from A)

(b) Generic Reception Diagram (to B)

On transmission, if it is required, a signature is generated using a hash code of the uncompressed plaintext. Then the plaintext, plus signature if present, is compressed. Next, if confidentiality is required, the block (compressed plaintext or compressed signature plus plaintext) is encrypted and prepended with the public-key-encrypted symmetric encryption key. Finally, the entire block is converted to radix-64 format.

On reception, the incoming block is first converted back from radix-64 format to binary. Then, if the message is encrypted, the recipient recovers the session key and decrypts the message. The resulting block is then decompressed. If the message is signed, the recipient recovers the transmitted hash code and compares it to its own calculation of the hash code.

Secure Socket Layer (SSL) Protocol

It is an internet protocol for secure exchange of information between a web browser and a web 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.

Goals:

- Confidentiality
- Integrity
- Authentication

SSL Architecture

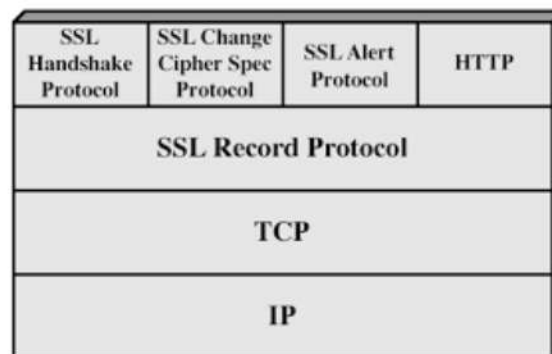


Fig: SSL Protocol Stack

SSL works in terms of connection and session between clients and servers.

- **SSL Connection:** A connection is a transport (in the OSI layering model definition) that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session.
- **SSL Session:** An SSL session is an association between a client and a server. Sessions are created by the Handshake Protocol. Sessions define a set of cryptographic security parameters, which can be shared among multiple connections.

Session State and Connection State:

A session state is defined by the following parameters:

- **Session identifier:** An arbitrary byte sequence chosen by the server to identify an active or resumable session state.
- **Peer certificate:** An X509.v3 certificate of the peer. This element of the state may be null.
- **Compression method:** The algorithm used to compress data prior to encryption.
- **Cipher spec:** Specifies the bulk data encryption algorithm and a hash algorithm used for MAC calculation. It also defines cryptographic attributes such as the hash size.
- **Master secret:** 48-byte secret shared between the client and server.
- **Is resumable:** A flag indicating whether the session can be used to initiate new connections.

A connection state is defined by the following parameters:

- **Server and client random:** Byte sequences that are chosen by the server and client for each connection.
- **Server write MAC secret:** The secret key used in MAC operations on data sent by the server.
- **Client write MAC secret:** The secret key used in MAC operations on data sent by the client.
- **Server write key:** The conventional encryption key for data encrypted by the server and decrypted by the client.
- **Client write key:** The conventional encryption key for data encrypted by the client and decrypted by the server.
- **Initialization vectors:** When a block cipher in CBC mode is used, an initialization vector (IV) is maintained for each key. This field is first initialized by the SSL Handshake Protocol. Thereafter the final ciphertext block from each record is preserved for use as the IV with the following record.
- **Sequence numbers:** Each party maintains separate sequence numbers for transmitted and received messages for each connection. When a party sends or receives a change cipher spec message, the appropriate sequence number is set to zero.

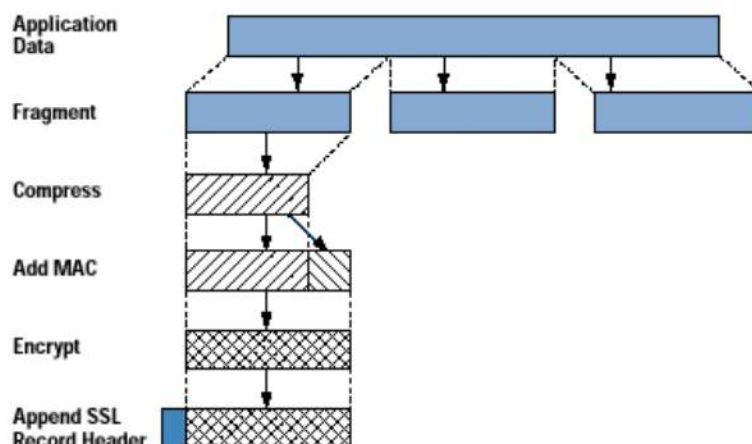
SSL Record Protocol

SSL Record Protocol provides two services for SSL connection:

- **Confidentiality:** The handshake protocol defines a shared secret key that is used for conventional encryption of SSL payloads.
- **Message Integrity:** The handshake protocol also defines a shared secret key that is used to form a MAC.

SSL Record Protocol Operation:

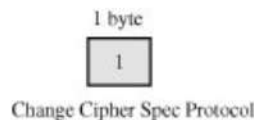
In SSL Record Protocol application data is divided into fragments. The fragment is compressed and then encrypted. MAC (Message Authentication Code) generated by algorithms like SHA (Secure Hash Protocol) and MD5 (Message Digest) is appended. After that encryption of the data is done and in last SSL header is appended to the data.



Change Cipher Spec Protocol

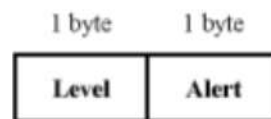
This protocol uses SSL record protocol. Unless Handshake Protocol is completed, the SSL record Output will be in pending state. After handshake protocol the Pending state is converted into Current state.

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

**Alert Protocol**

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

- The first bytes takes the value warning (1) or fatal (2) to convey the security of the message.
- If the level is fatal, SSL immediately terminates the connections.
- Other connection on the same session may continue, but no new connections on the session may be established.
- The second byte contains a code that indicates a specific alert.

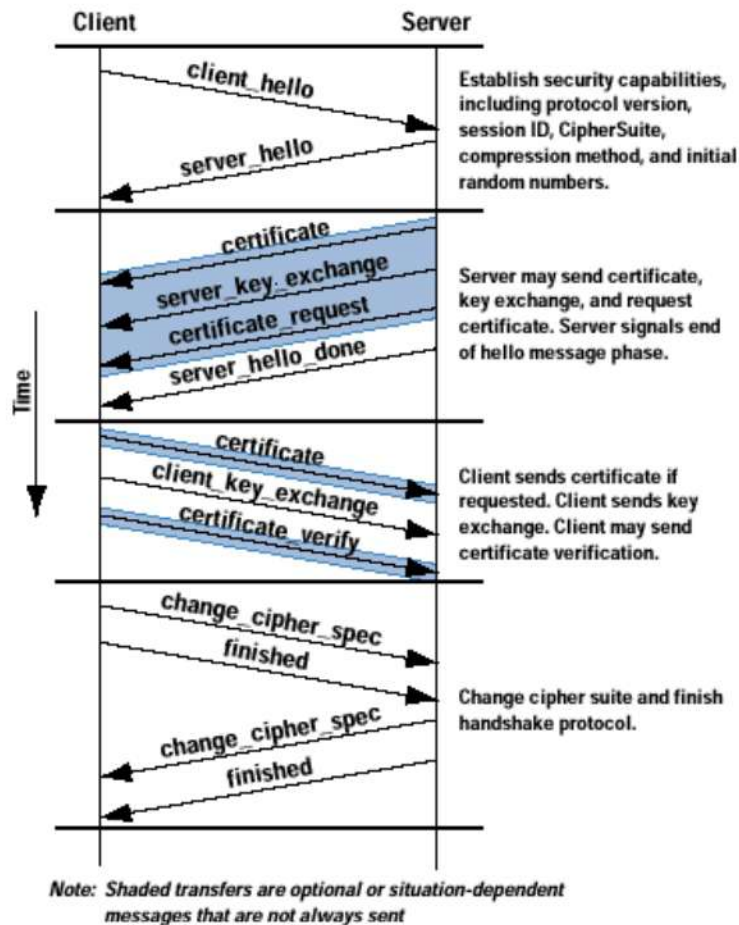


(b) Alert Protocol

SSL Handshake Protocol

Handshake protocol is used to establish a secure session between the client and the server. This protocol allows the server and client to authenticate each other and to negotiate an encryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSL record. The handshake protocol is used before any application data is transmitted. Handshake protocol uses four phases to complete its cycle:

1. **Establish Security Capabilities:** This phase is used to initiate a logical connection and to establish the security capabilities that will be associated with it. The exchange is initiated by the client, which sends a `client_hello` message.
2. **Server Authentication and Key Exchange:** The server begins this phase by sending its certificate, if it needs to be authenticated; the message contains one or a chain of X.509 certificates. The certificate message is required for any agreed-on key exchange method except anonymous Diffie-Hellman.
3. **Client Authentication and Key Exchange:** If the server has requested a certificate, the client begins this phase by sending a certificate message. If no suitable certificate is available, the client sends a `no_certificate` alert instead.
4. **Change Cipher Spec and Finish:** This phase completes the setting up of a secure connection. The client sends a `change_cipher_spec` message and copies the pending CipherSpec into the current CipherSpec. This message is not considered part of the Handshake Protocol but is sent using the Change Cipher Spec Protocol. The client then immediately sends the finished message under the new algorithms, keys, and secrets.



Transport Layer Security (TLS) Protocol

- The TLS protocol is the IETF standard version of the SSL protocol.
- The TLS protocol allows client-server application to communicate across a network in a way designed to prevent eavesdropping (secretly listening to the private conversation of others without their consent) and tampering (a device or process that makes unauthorized access to the protected object easily).
- Transport layer security (TLS) is a protocol that ensure privacy between communicating applications and their use on the internet.
- When a server and client communicate, TLS ensure that no third party may eavesdrop or tamper with any message.
- Two main ways of achieving TLS:
 1. Use a different port number for TLS connection (for e.g. port 443 for HTTP's).
 2. Use the regular port number and have the client request that the server switch the connection to TLS using a protocol.

SSL vs TLS

- **Version:** SSL 3.0 is used where TLS uses 1.0
- **Cipher suit:** SSL supports *FORTEZZA* but TLS does *not support*.
- **Cryptographic Secrets:** SSL uses *Message Digest to generate master secret* and TLS uses *pseudo random function* to generate master secret.
- **Record Protocol:** For integrity SSL uses *MAC* and TLS uses *HMAC (Hash based MAC)*.
- **Alert Protocol:** TLS *delete* some alert messages of SSL and *add some new* Alert Messages.
- **Certificate Verification:** In SSL it is *complex* and in TLS it is *simple*.
- **Security:** SSL is *less secured* as compared to TLS.

Internet Protocol Security (IPSec)

Internet protocol security (IPSec) is a set of security protocols and algorithms used to secure IP data at the network layer. IPSec provides data confidentiality, integrity, authentication of IP packets while maintaining the ability to route them through existing IP networks. IPSec is a framework, not an implementation.

IP-level security encompasses three functional areas:

- **Authentication:** The authentication mechanism ensures that the received packet was sent by the identified source. It also assures that the packet has not been altered in transit.
- **Confidentiality:** The confidentiality facility enables communicating nodes to encrypt messages to prevent eavesdropping by third parties.
- **Key Management:** It is concerned with secure exchange of keys.

Applications of IPSec

IPsec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet. Examples are:

- Secure branch office connectivity over the Internet.
- Secure remote access over the Internet.
- Establishing extranet and intranet connectivity with partners.
- Enhancing electronic commerce security.

Benefits of IPSec

- When IPsec is implemented in a firewall or router, it provides strong security to all traffic crossing the perimeter.
- 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.
- IPsec can provide security for individual users if needed.

IPSec Architecture

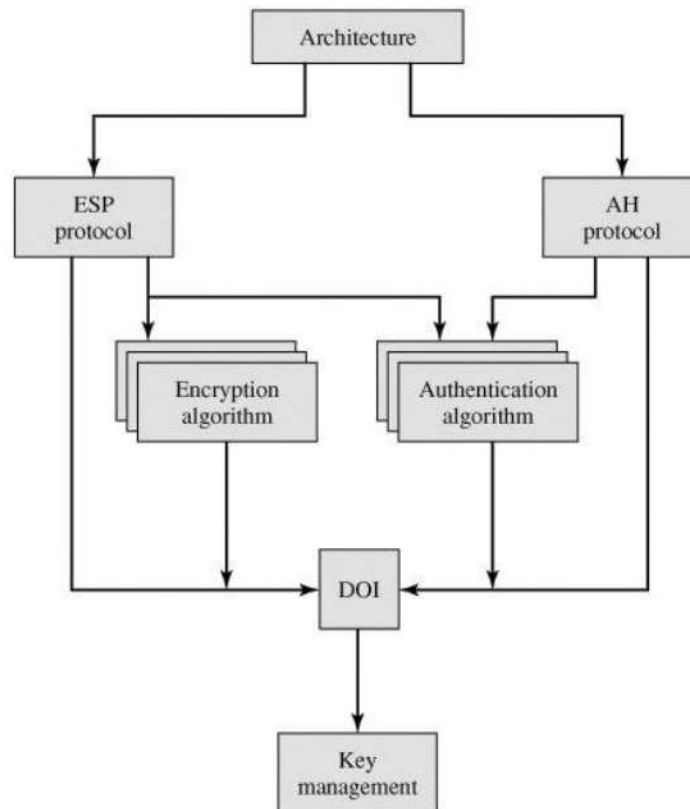


Fig: IPSec Architecture

- **Architecture:** Covers the general concepts, security requirements, definitions, and mechanism defining IPSec technology.
- **Encapsulating Security Payload (ESP):** Covers packet format and general issues related to the use of the ESP for packet encryption and, optionally, authentication.
- **Authentication Header (AH):** Covers the packet format and general issues related to the use of AH for packet authentication.
- **Encryption Algorithm:** A set of documents that describe how various encryption algorithm are used for ESP.
- **Authentication Algorithm:** A set of documents that describe how various authentication algorithms are used for AH and for authentication option of ESP.
- **Key Management:** Documents that describe key management schemes.
- **Domain of Interpretation (DOI):** Contains the values needed for the other documents to relate to each other.

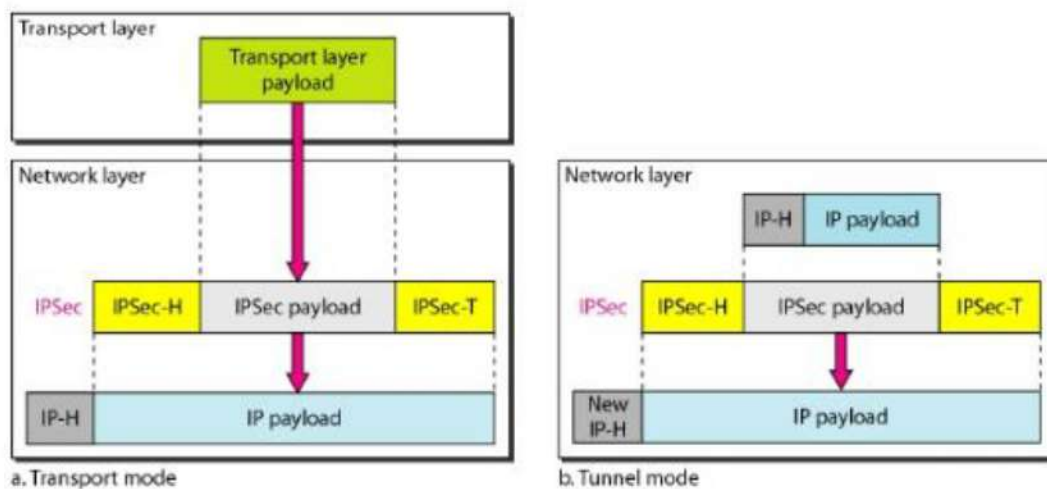
IPSec Security Services

- **Connectionless Integrity:** Assurance that received traffic has not been modified. Integrity includes anti-reply defenses.
- **Data origin authentication:** Assurance that traffic is sent by legitimate party or parties.
- **Confidentiality (encryption):** Assurance that user's traffic is not examined by non-authorized parties.
- **Access Control:** Prevention of unauthorized use of a resource.
- Limited traffic flow confidentiality.

IPSec Operation Mode

IPSec has two operation modes: Transport mode and Tunnel mode.

- **Transport Mode:** Only the payload or data of the original IP packet is protected (encrypted, authenticated, or both) in transport mode. The protected payload is then encapsulated by the IPSec headers and trailers while the original IP header remains intact and is not protected by IPSec.
- **Tunnel Mode:** The entire original IP packet is protected (encrypted, authenticated, or both) in tunnel mode. The packet is then encapsulated by the IPSec headers and trailers. Finally a new IP header is prefixed to the packet, specifying the IPSec endpoints as the source and destination.

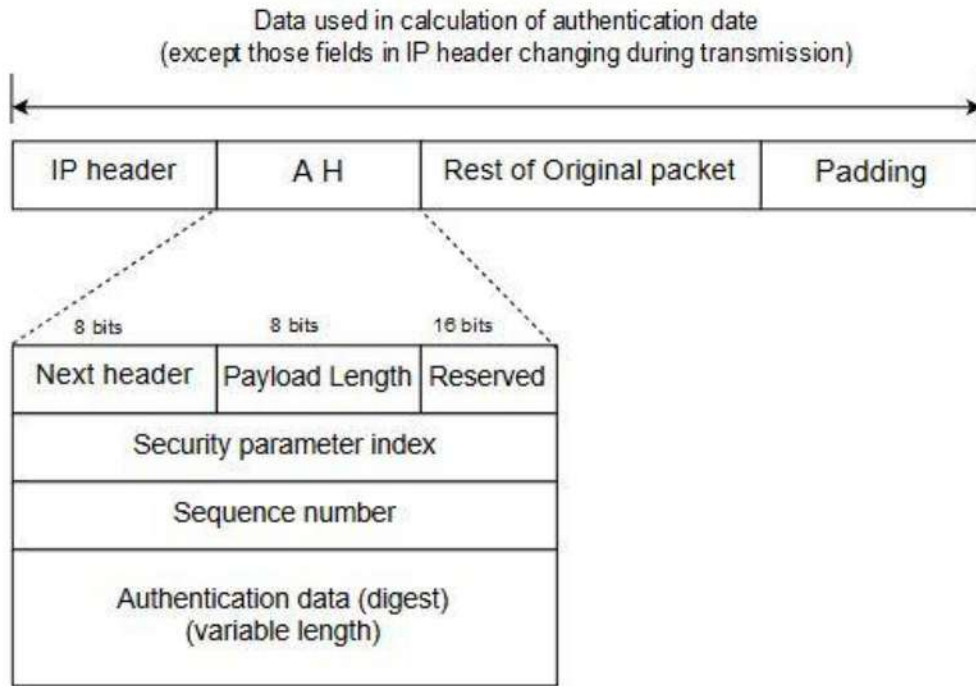


IPSec Protocols

IPSec uses two distinct protocols, **Authentication Header (AH)** and **Encapsulating Security Payload (ESP)**, which are defined by the IETF.

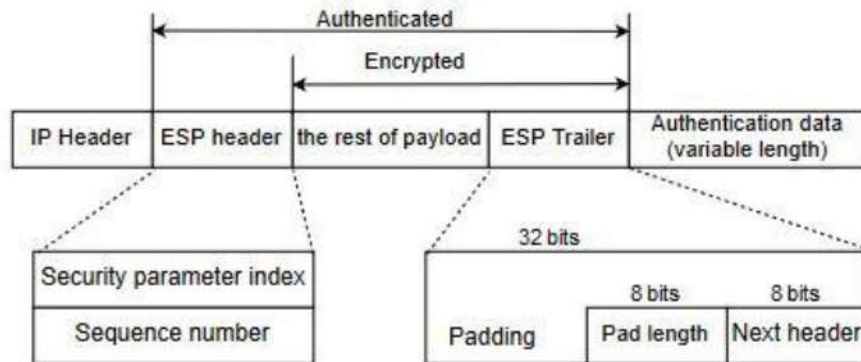
- The **AH protocol** provides a mechanism for authentication only. AH provides data integrity, data origin authentication, and an optional replay protection service. Data integrity is ensured by using a message digest that is generated by an algorithm such as HMAC-MD5 or HMAC-SHA. Data origin authentication is ensured by using a shared secret key to create the message digest. Replay protection is provided by using a sequence number field with the AH header. AH authenticates IP headers and their payloads, with the

exception of certain header fields that can be legitimately changed in transit, such as the Time To Live (TTL) field.



Authentication Header (AH) protocol

- The **ESP protocol** provides data confidentiality (encryption) and authentication (data integrity, data origin authentication, and replay protection). ESP can be used with confidentiality only, authentication only, or both confidentiality and authentication. When ESP provides authentication functions, it uses the same algorithms as AH, but the coverage is different. AH-style authentication authenticates the entire IP packet, including the outer IP header, while the ESP authentication mechanism authenticates only the IP datagram portion of the IP packet.



ESP

Firewalls

A firewall is a network security device, either hardware or software-based, which monitors all incoming and outgoing traffic and based on a defined set of security rules it accepts, rejects or drops that specific traffic.

Accept: allow the traffic

Reject: block the traffic but reply with an “unreachable error”

Drop: block the traffic with no reply

A firewall establishes a barrier between secured internal networks and outside untrusted network, such as the Internet.

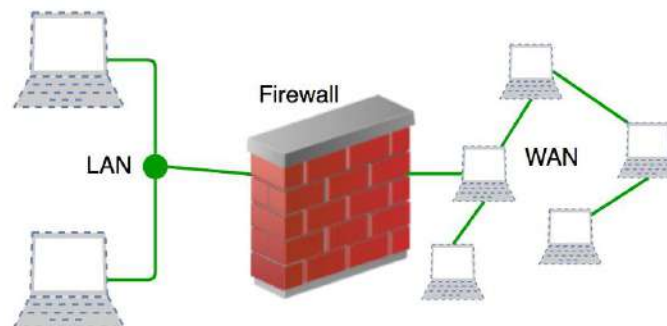


Fig: Firewall

Firewall Characteristics

Major **characteristics** related to firewall protection are described below:

1. Various protection levels
2. Wireless network (Wi-fi) Protection
3. Internet and network access
4. Blockage against unauthorized access
5. Protection against malware
6. Provide access only to valid data packets
7. Provision of different configurations
8. Provision of numerous security policies
9. Allowing to pass authorized traffic that fulfils a set of rules
10. Firewall functions like an immune system for malware and unauthorized access; therefore, it ensures a secure system and an OS.

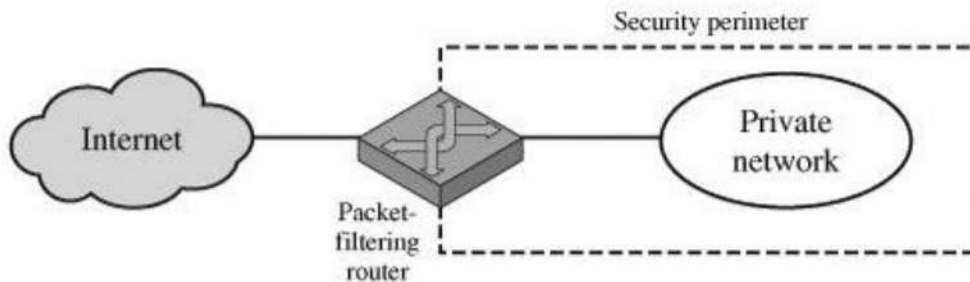
The following **capabilities** are within the scope of a firewall:

1. A firewall defines a single choke point that keeps unauthorized users out of the protected network, prohibits potentially vulnerable services from entering or leaving the network, and provides protection from various kinds of IP spoofing and routing attacks.
2. A firewall provides a location for monitoring security-related events. Audits and alarms can be implemented on the firewall system.
3. A firewall is a convenient platform for several Internet functions that are not security related.
4. A firewall can serve as the platform for IPSec.

Types of Firewalls

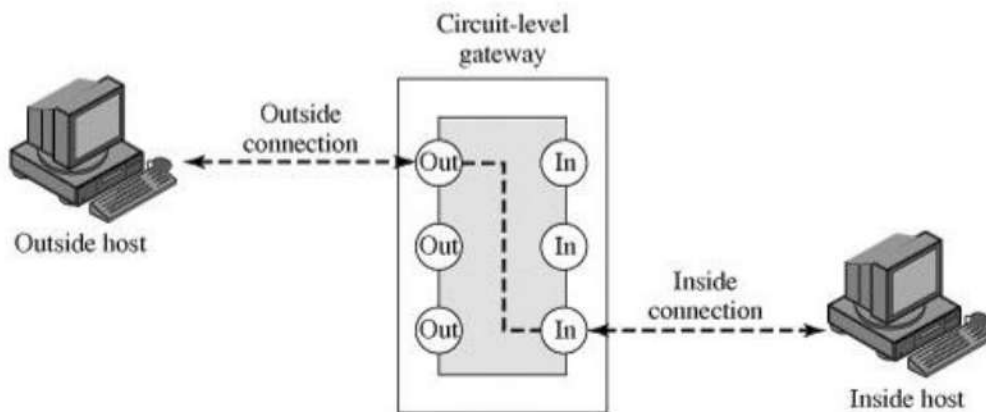
1. Packet Filtering Firewall:

Packet filtering firewalls work at the network layer (OSI model), or the IP layer (TCP/IP). In this each packet is compared to a set of criteria before it is forwarded. Depending on the packet and the criteria, the firewall can drop, forward the packet or send a message to the originator. Rules can be source and destination IP address, source and destination port number and protocol used. The advantages of packet filtering firewalls is their low cost and low impact on network performance.



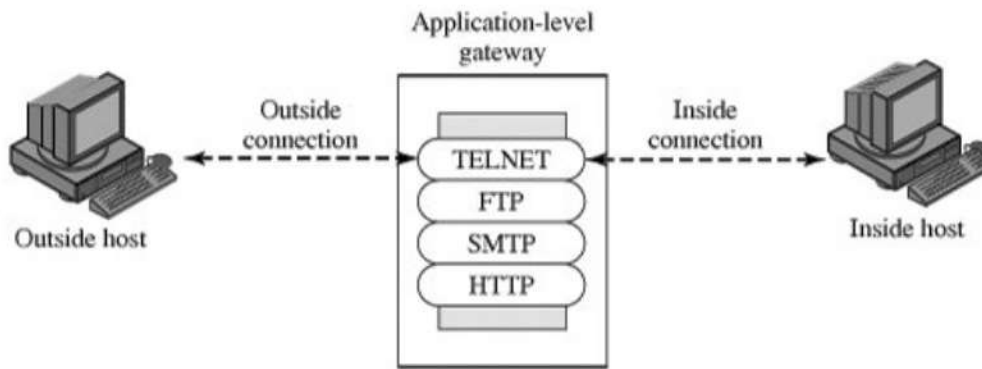
2. Circuit Level Gateway Firewall:

It work at the session layer (OSI model), or the TCP layer (TCP/IP). They monitor TCP handshaking between packets to determine whether a requested session is legitimate. Information passed to a remote computer through a circuit level gateway appears to have originated from the gateway. This is useful for hiding information about protected networks. Circuit level gateways are relatively inexpensive and have the advantage of hiding information about the private network they protect. On the other hand, they do not filter individual packets.



3. Application Level Gateway Firewall:

Application level gateways, also called proxies, are similar to circuit-level gateways except that they are application specific. They can filter packets at the application layer of the OSI model. Incoming or outgoing packets cannot access services for which there is no proxy. In plain terms, an application level gateway that is configured to be a web proxy acts as the server to the internal network and client to the external network. Because they examine packets at application layer, they can filter application specific commands such as http: post and get, etc. Application level gateways can also be used to log user activity and logins. They offer a high level of security, but have a significant impact on network performance.



4. *Stateful Multilayer Inspection Firewall:*

It combines the aspects of the other three types of firewalls. They filter packets at the network layer, determine whether session packets are legitimate and evaluate contents of packets at the application layer. They rely on algorithms to recognize and process application layer data instead of running application specific proxies. Stateful multilayer inspection firewalls offer a high level of security, good performance and transparency to end users. They are expensive however, and due to their complexity are potentially less secure than simpler types of firewalls if not administered by highly competent personnel.

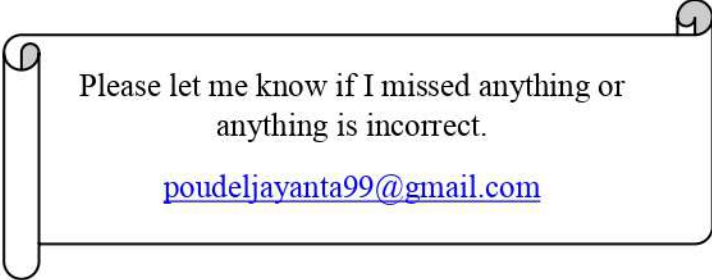
5. *Next-generation Firewalls (NGFW):*

Many of the latest released firewalls are usually defined as '**next-generation firewalls**'. However, there is no specific definition for next-generation firewalls. This type of firewall is usually defined as a security device combining the features and functionalities of other firewalls. These firewalls include **deep-packet inspection (DPI)**, surface-level packet inspection, and TCP handshake testing, etc.

NGFW includes higher levels of security than packet-filtering and stateful inspection firewalls. Unlike traditional firewalls, NGFW monitors the entire transaction of data, including packet headers, packet contents, and sources. NGFWs are designed in such a way that they can prevent more sophisticated and evolving security threats such as malware attacks, external threats, and advance intrusion.

Limitations of firewall

1. The firewall cannot protect against attacks that bypass the firewall. Internal systems may have dial-out capability to connect to an ISP. An internal LAN may support a modem pool that provides dial-in capability for traveling employees and telecommuters.
2. The firewall may not protect fully against internal threats, such as a disgruntled employee or an employee who unwittingly cooperates with an external attacker.
3. An improperly secured wireless LAN may be accessed from outside the organization. An internal firewall that separates portions of an enterprise network cannot guard against wireless communications between local systems on different sides of the internal firewall.
4. A laptop, PDA, or portable storage device may be used and infected outside the corporate network, and then attached and used internally.



Please let me know if I missed anything or
anything is incorrect.

poudeljayanta99@gmail.com