CHAPTER – 1

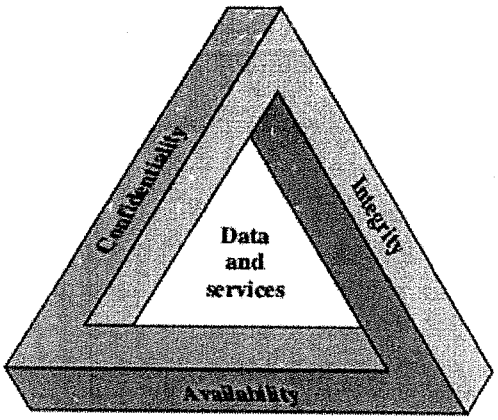
INTRODUCTION

We are leaving in the information age. So information is an asset that has value like other asset. As an asset, information needs to be secured from attacks.

Computer Security is the collection of tools designed to protect data to prevent from hackers.
Network Security measures to protect data during their transmission.
Internet Security measures to protect data during their transmission over a collection of interconnected networks.

SECURITY GOALS

The protection is affordable to an automated information system in order to attain the applicable objectives of preserving some specific security requirements, includes:



1. **CONFIDENTIALITY**: It is the most common aspect of information security, itensures that the information in a computer system and transmitted information are accessible only for reading by authorized parties.

2. **INTEGRITY:**Guarding against improper information modification or destruction.Ensures that only authorized parties are able to modify computer system assets and transmitted information. Modification includes writing, changing status, deleting, creating and delaying or replaying of transmitted messages.

3. **AVAILABILITY:** Ensuring timely and reliable access to and use of information.Network Services are continuously available to the legitimate users, whenever they require it.

SECURITY ATTACKS

Any action that compromises the security of information's owned by an organization.

Three goals of security **confidentiality, integrity, availability** can be threatened by the security attacks.

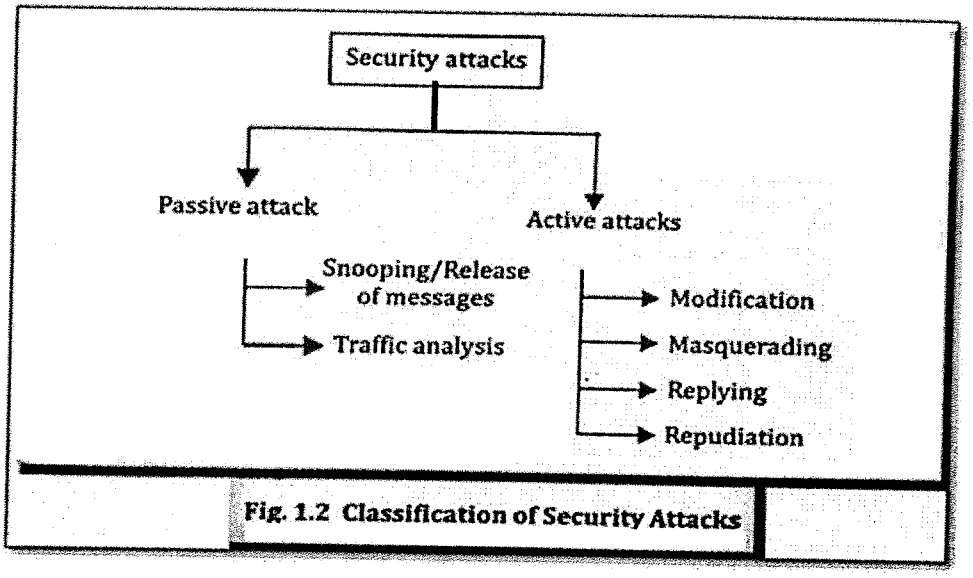


Fig. 1.2 Classification of Security Attacks

Note:

- A passive attack attempts to learn or make use of information but does not affect system resources.
- An active attack attempts to alter system resources or affect their operation.

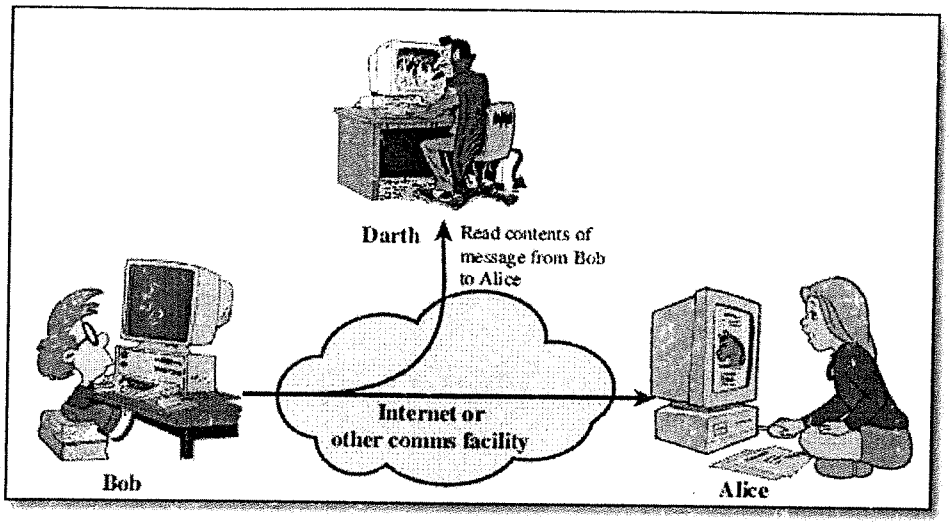
1. PASSIVE ATTACKS

- ✓ In the nature of eavesdropping on, or monitoring of, transmissions.
- ✓ The goal is to obtain information that is being transmitted.
- ✓ Very difficult to detect, because they do not involve any alteration of the data.
- ✓ Feasible to prevent the success of these attacks, usually by means of encryption.
- ✓ Emphasis in dealing with passive attacks is on prevention rather than detection.

There are two types of passive attacks, they are:

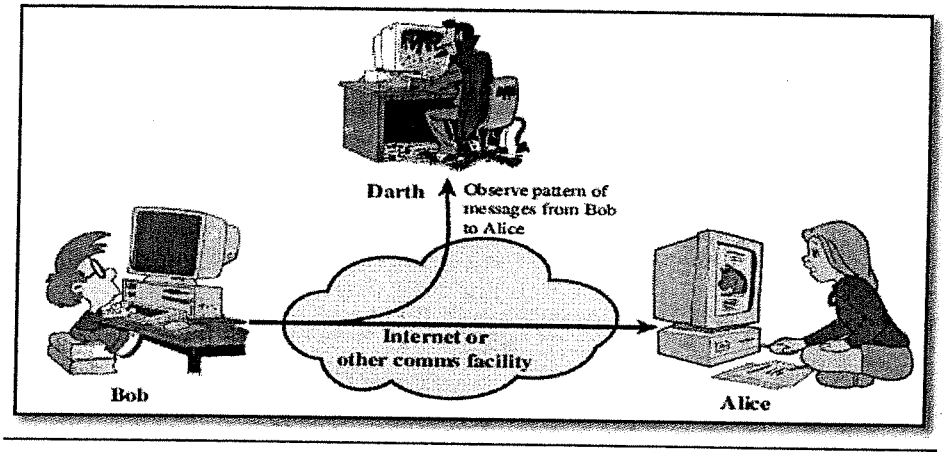
a. Release of message contents/ Snooping

- A telephone conversation, an electronic mail message, and a transferred file may contain sensitive or confidential information
- Prevent an opponent from learning the contents of these transmissions.



b. Traffic analysis

- Observe the pattern of these messages
- The opponent could determine the location and identity of communicating hosts and could observe the frequency and length of messages being exchanged.
- This information might be useful in guessing the nature of the communication that was taking place.



2. ACTIVE ATTACKS

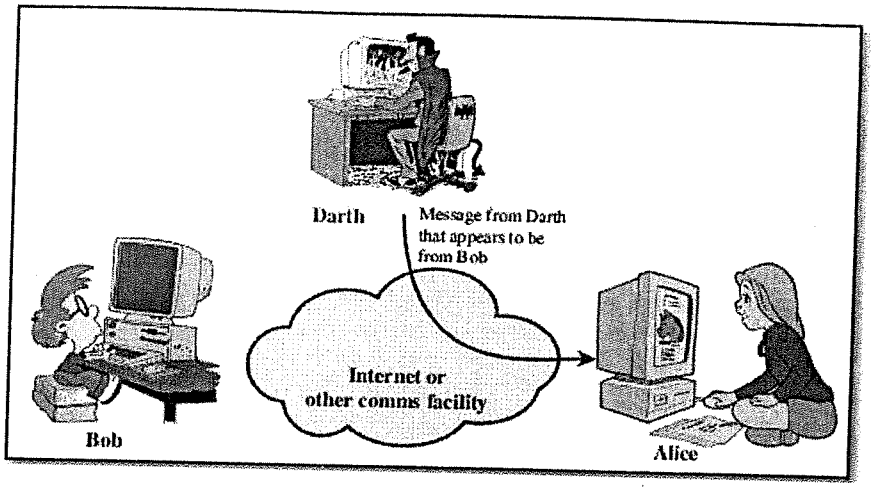
- ✓ Active attacks involve some modification of the data stream or the creation of a false stream.
- ✓ Detect and to recover from any disruption or delays caused by them.

There are four types of active attacks, they are:

a. Masquerade/Spoofing

- One entity pretends to be a different entity.
- It happens when the attacker impersonates somebody else.

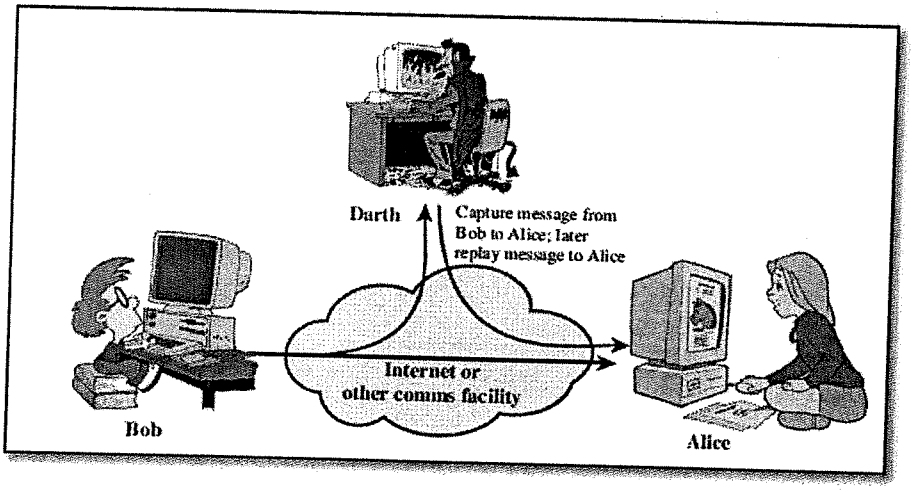
Example: An attacker might steal the bank card and PIN of the bank customer and pretend that he is that customer.



b. Replaying

- A hacker steals an authorized user's login information by stealing the session ID.
- Passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect.

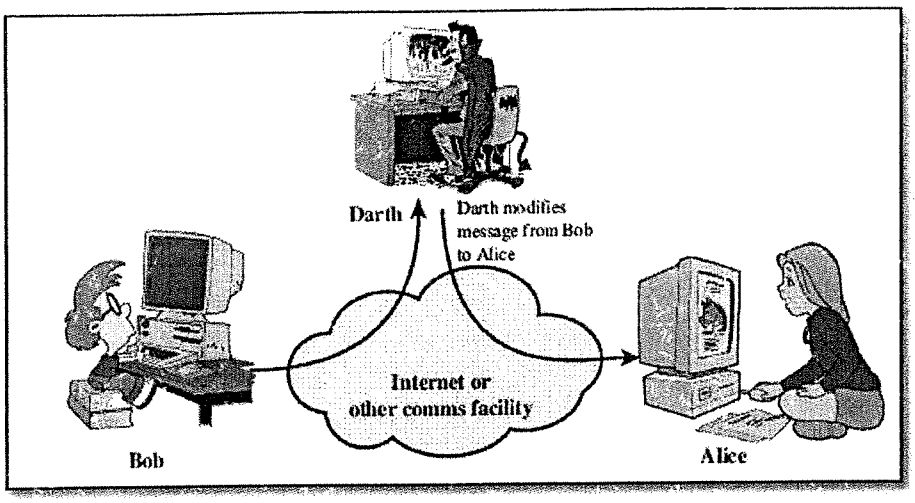
Example: A person sends request to her bank to ask for payment to the attacker, who has done a job for him. The intercepts the message and send it again to receive another payment from the bank.



c. Modification Of Messages

- Some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect.

Example: A message meaning "Allow John to read confidential file accounts" it is modified and sent by the attacker that "Allow Rama to read confidential file accounts".



d. Repudation (Denial Of Service attack - DOS)

- DOS is performed by one of the two parties in the communication.
- This attack may have a specific target.

Example:

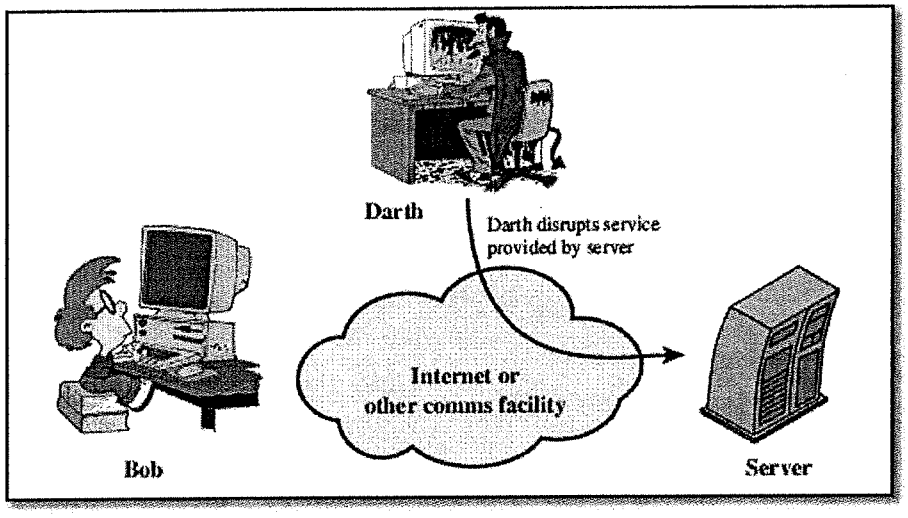
✓ An entity may suppress all messages directed to a particular destination.

✓ Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages so as a result performance of the will decrease.

✓ Crashing the victim.

✓ Forcing more computation.

✓ Taking long path in processing the packets.

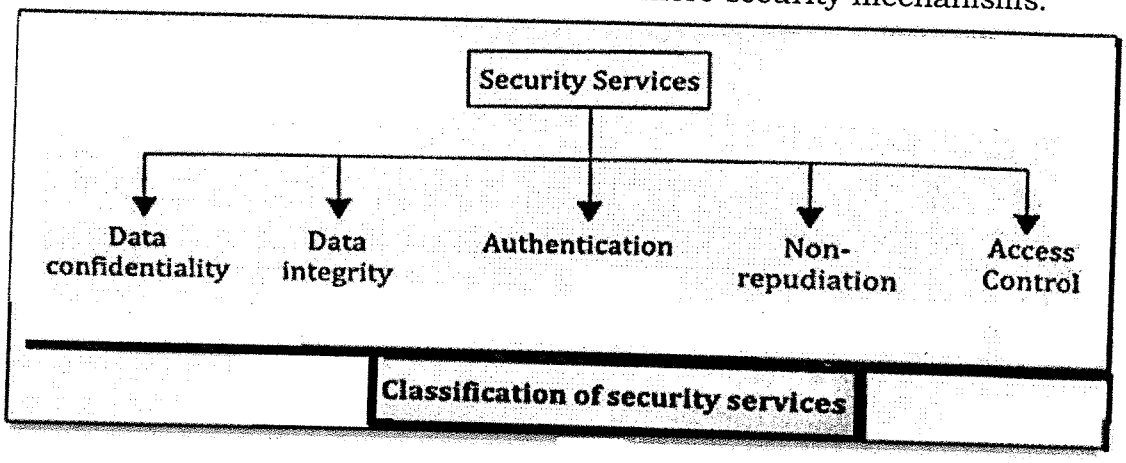


Difference between Passive Attack and Active Attack:

Passive Attack	Active Attack
In passive attack, Modification in information does not take place.	In active attack, Modification in information takes place.
Danger for Confidentiality	Danger for Integrity as well as Availability
Attention is on prevention	Attention is on detection.
It does not affect the system	It affects the system
While in passive attack, Victim does not get informed about the attack	In active attack, Victim gets informed about the attack
System resources are not changed	System resources can be changed

SECURITY SERVICES

A service that enhances the security of data processing system and information transfers. A security service makes use of one or more security mechanisms.



1. **DATA CONFIDENTIALITY:**Ensures that the information in a computer system and transmitted information are accessible for reading only by authorized parties.

2. **DATA INTEGRITY:**Ensures that only authorized parties are able to modify computer system assets and transmitted information. Modification includes writing, changing status, deleting, creating and delaying or replaying of transmitted messages.

3. **AUTHENTICATION:**Ensures that the origin of a message or electronic document is correctly identified, with an assurance that the identity is not false.

4. **NONREPUDIATION:**Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.Requires that neither the sender nor the receiver of a message be able to deny the transmission.

5. **ACCESS CONTROL:**The prevention of unauthorized use of a resource (i.e., this service controls that can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).

SECURITY MECHANISMS

A mechanism that is designed to detect, prevent, or recover from a security attack. Security mechanisms are used to provide security services.

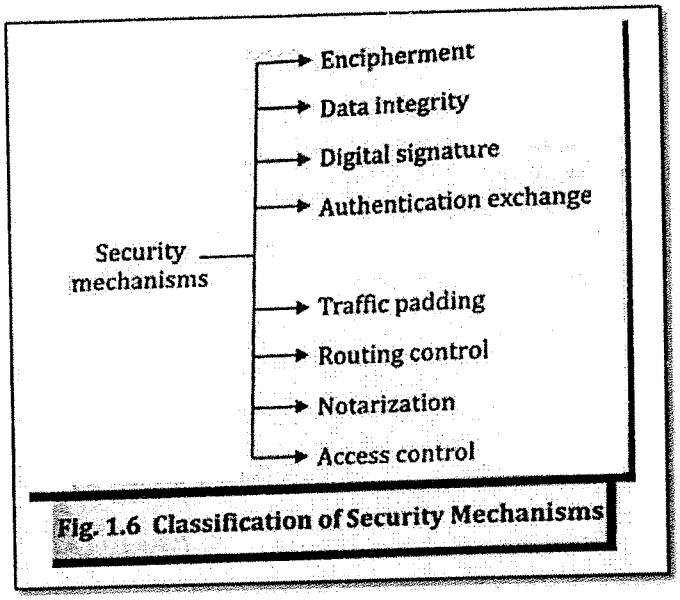
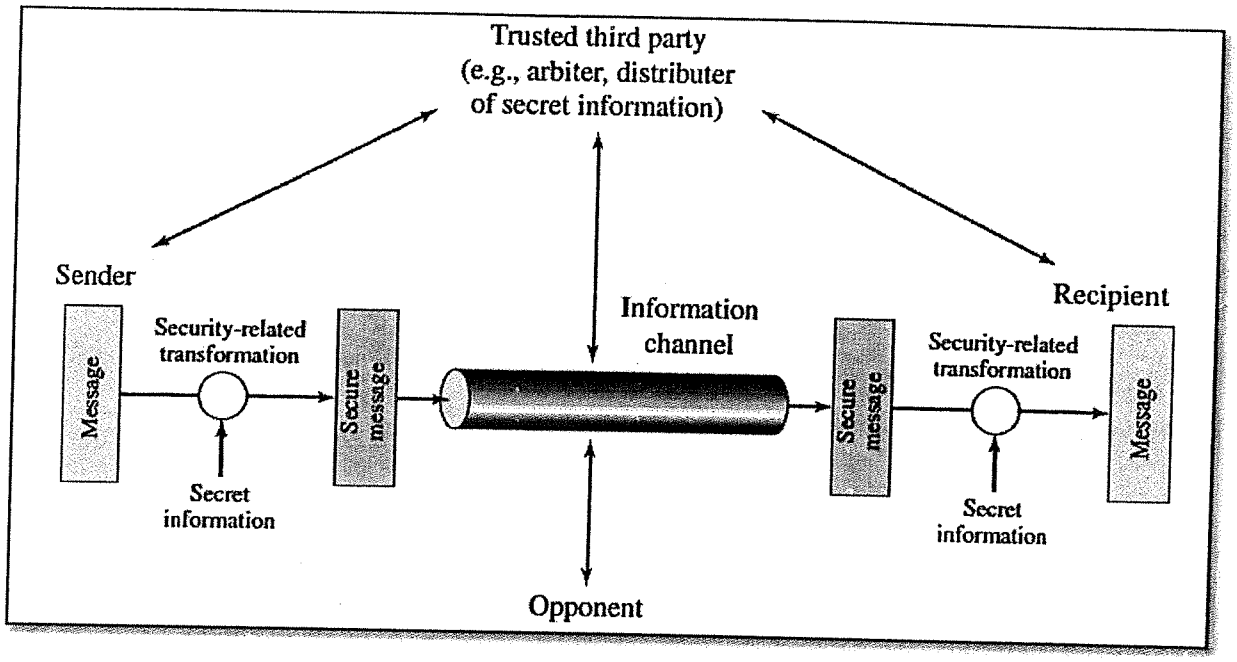


Fig. 1.6 Classification of Security Mechanisms

Classifications of security mechanism are as follows:

a. **Encipherment:** This is the process of using mathematical algorithms to transform data into a form that is not readily intelligible. The transformation and subsequent recovery of the data depend on an algorithm and zero or more encryption keys.
b. **Digital Signature:** A digital signature is a means by which the sender can electronically sign the data and the receiver can electronically verify the signature.
c. **Access Control:** A variety of mechanisms are available that enforce access rights to resources.
d. **Data Integrity:** A variety of mechanisms may be used to assure the integrity of a data unit or stream of data units.
e. **Authentication Exchange:** This is a mechanism intended to ensure the identity of an entity by means of information exchange.
f. **Traffic Padding:** The insertion of bits into gaps in a data stream is called traffic padding. This helps to thwart traffic analysis attempts.
g. **Routing Control:** Routing control enables selection of particular physically secure routes for certain data transmission and allows routing changes, especially when a breach of security is suspected.
h. **Notarization:** This is the use of a trusted third party to assure certain properties of a data exchange.

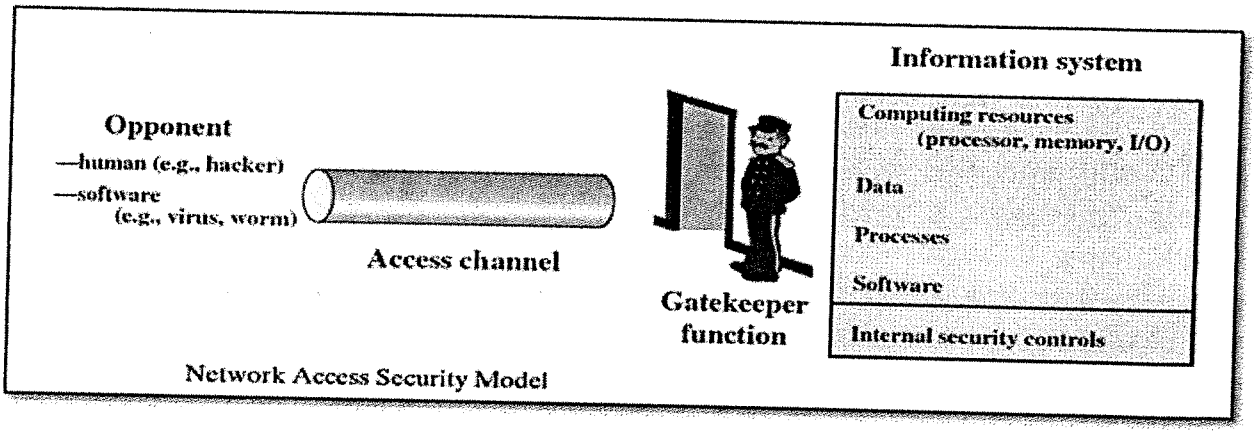
MODEL FOR NETWORK SECURITY



- A message is to be transferred from one party to another across some sort of Internet service.
- The two parties, who are the principals in this transaction, must cooperate for the exchange to take place.
- A logical information channel is established by defining a route through the Internet from source to destination and by the cooperative use of communication protocols (e.g., TCP/IP) by the two principals.

Four basic tasks in designing a particular security service:

1. Design an algorithm for performing the security-related transformation such that an opponent cannot defeat its purpose.
2. Generate the secret information to be used with the algorithm.
3. Develop methods for the distribution and sharing of the secret information.
4. Specify a protocol to be used by the two principals that makes use of the security algorithm and the secret information to achieve a particular security service.



CRYPTOGRAPHY

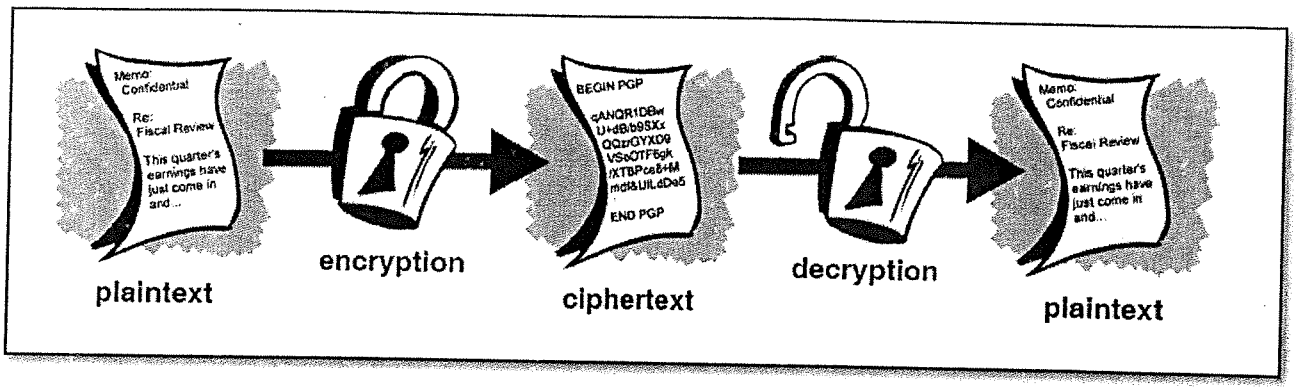
- Cryptography, a word with Greek origins, *crypto*means "**secret**" *graphy* means "**writing**". Cryptography means secret-writing.
- Cryptography is the science of using mathematics to encrypt and decrypt data.
- Cryptography enables you to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient.

Cryptographic systems are generally classified along three independent dimensions:

1. The type of operations used for transforming plaintext to ciphertext.
2. The number of keys used.
3. The way in which the plaintext is processed.

Basic Concepts

1. **Plaintext:** This is the original intelligible message or data that is fed into the algorithm as input.
2. **Encryption algorithm:** The encryption algorithm performs various substitutions and transformations on the plaintext.
3. **Ciphertext:** The encoded message resulting from the encryption.
4. **Cipher:**Cipher is an algorithm which is applied to plaintext to get ciphertext.
5. **Secret Key** Some critical information used by the cipher, known only to the sender & receiver.
6. **Encipher (encode):**The process of converting plaintext to cipher text using a cipher and a key.
7. **Decipher (decode):**The process of converting cipher text back into plaintext using a cipher and a key.
8. **Decryption algorithm:**This is the reverse of encryptionalgorithm. It takes the cipher text and secret key and produces the original plaintext.
9. **Encryption:** The process of turning plaintext back into ciphertext.
10. **Decryption:** The process of turning ciphertext back into plaintext.
11. **Cryptanalysis**:The art or science of transferring cipher text into plain text without initial knowledge of the key used to encrypt the plain text.Also called code breaking
12. **Cryptology:** The scientific study of both cryptography and cryptanalysis.
13. **Cryptanalyst:** A person expert in analysing and breaking codes and ciphers. The idea for cryptanalyst is to extract the secret key.



CRYPTOGRAPHY ATTACKS

The basic intention of an attacker is to break a cryptosystem and to find the plaintext from the ciphertext.

1. **Ciphertext only Attack:** Cryptanalyst obtains the sample of ciphertext, without the plaintext associated with it.
2. **Known-plaintext Attack:**In this method, the attacker knows the plaintext for some parts of the ciphertext. The task is to decrypt the rest of the ciphertext using this information. This may be done by determining the key or via some other method.
3. **Chosen-plaintext Attack:**In this method, the attacker has the text of his choice encrypted. So he has the ciphertext-plaintext pair of his choice. This simplifies his task of determining the encryption key.
4. **Chosen-ciphertext Attack:**In this method, the attacker chooses a portion of the decrypted ciphertext. He then compares the decrypted ciphertext with the plaintext and figures out the key.
5. **Dictionary Attack:** This attack has many variants, all of which involve compiling a 'dictionary'. In simplest method of this attack, attacker builds a dictionary of cipher texts and corresponding plaintexts that he has learnt over a period of time. In future, when an attacker gets the ciphertext, he refers the dictionary to find the corresponding plaintext.
6. **Brute Force Attack (BFA):** In this method, the attacker tries to determine the key by attempting all possible keys. If the key is 8 bits long, then the number of possible keys is 28 = 256. The attacker knows the ciphertext and the algorithm, now he attempts all the 256 keys one by one for decryption.
7. **Man in Middle Attack (MIM):** The targets of this attack are mostly public key cryptosystems where key exchange is involved before communication takes place.
8. **Timing Attacks:**They exploit the fact that different computations take different times to compute on processor. By measuring such timings, it is be possible to know about a particular computation the processor is carrying out. For example, if the encryption takes a longer time, it indicates that the secret key is long.

DECRYPTION AND CRYPTANALYSIS

Cryptanalysis is the study of cipher text, ciphers and cryptosystems with the aim of understanding how the system works and finding a secret key.

There are four basic steps in cryptanalysis:

i. **Determine the language being used:** To decode the cipher text into plaintext, we should have a general idea of what the plaintext is supposed to be like.
ii. **Identify the Encryption being used:** Certain Encryption systems are easily identifiable through the use of some telltale signs.
iii. **Find the Key:** Most ciphers and codes use a key to unlock them. Depending on the complexity of the encryption system, this process can be quite painstaking and laborious.
iv. **Decode the Message:** Once find the key and cipher text, we can decode into plaintext.

Public Key Infrastructure (PKI)

A Public Key Infrastructure (PKI) is a framework which supports the identification and distribution of public encryption keys.PKI is model for creating and distributing certificate based on X.509.

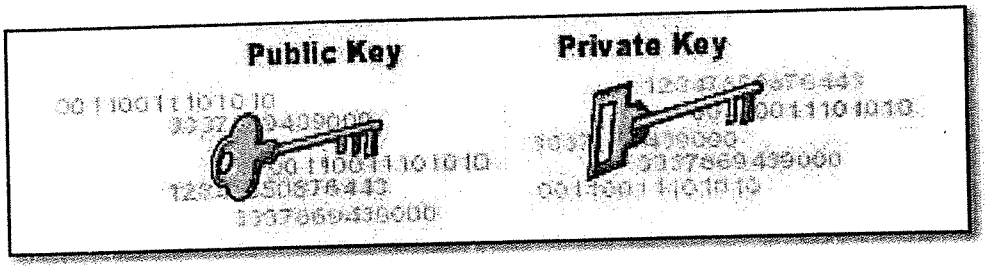
A public key infrastructure (PKI) is a system for the creation, storage, and distribution of digital certificates which are used to verify that a particular public key belongs to a certain entity.

The PKI creates digital certificates which map public keys to entities, securely stores these certificates in a central repository and revokes them if needed.

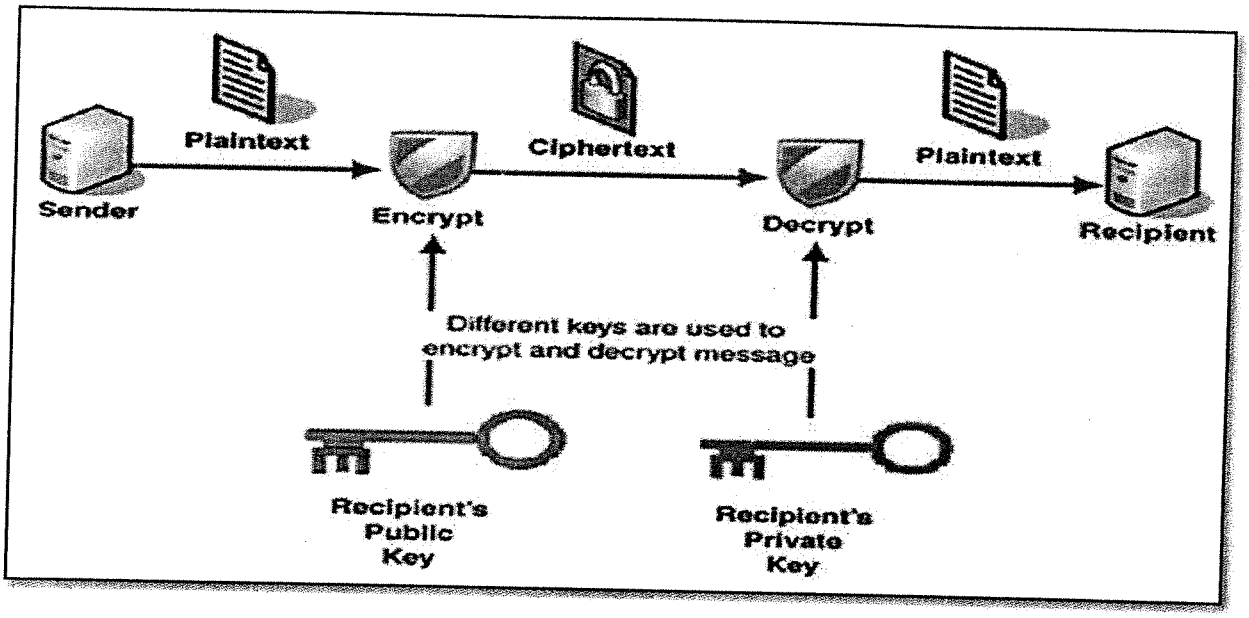
- **Certificate Authority (CA)** that stores, issues and signs the digital certificates.
- **Registration Authority (RA)** which verifies the identity of entities requesting their digital certificates to be stored at the CA.

PUBLIC KEY ENCRYPTION

Public-key encryption is a cryptographic system that uses two keys a ***public key*** known to everyone and a ***private or secret key*** known only to the recipient of the message.



Public and private keys are related in such a way that only the public key can be used to encrypt messages and only the corresponding private key can be used to decrypt them. Moreover, it is virtually impossible to deduce the private key if you know the public key.



DIGITAL SIGNATURE

- A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document.
- Digital signatures are encrypted messages that can be mathematically proven authentic.
- Asymmetric encryption processes are used to createdigital signatures. When an asymmetric cryptographic processuses the sender's private key to encrypt the message, the sender's public key must be used to decrypt the message.
- Main importance of Digital Signature is to maintain Message authentication, Data Integrity and Non- repudiation.

AUTHENTICATION

Authentication is the process of recognizing a user's identity. It is the mechanism of associating an incoming request with a set of identifying credentials. The credentials provided are compared to those on a file in a database of the authorized user's information on a local operating system or within an authentication server.

Certification Authority (CA)

CA is trusted entity which issues, manages, signs and revoke's digital certificate, which typically contains user name, public key and other identifying information.

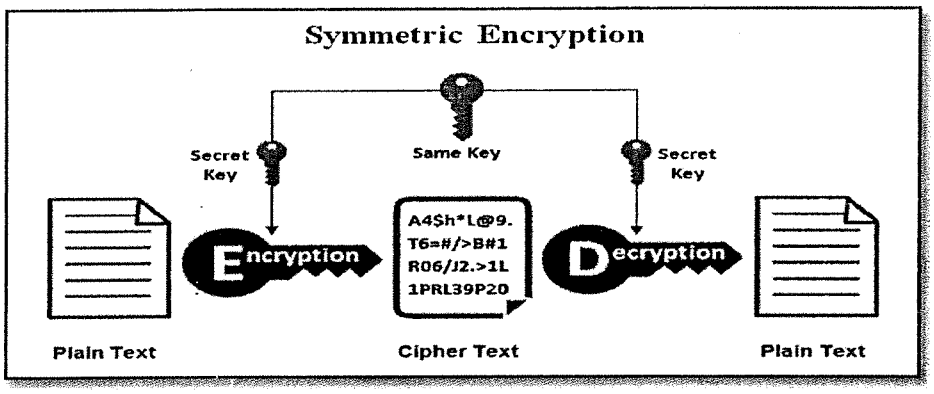
The key functions of CA are:

- Generating key pairs
- Issuing digital certificate
- Publishing Certificates
- Verifying certificates
- Revocation of certificates

SYMMETRIC-KEYCRYPTOGRAPHY

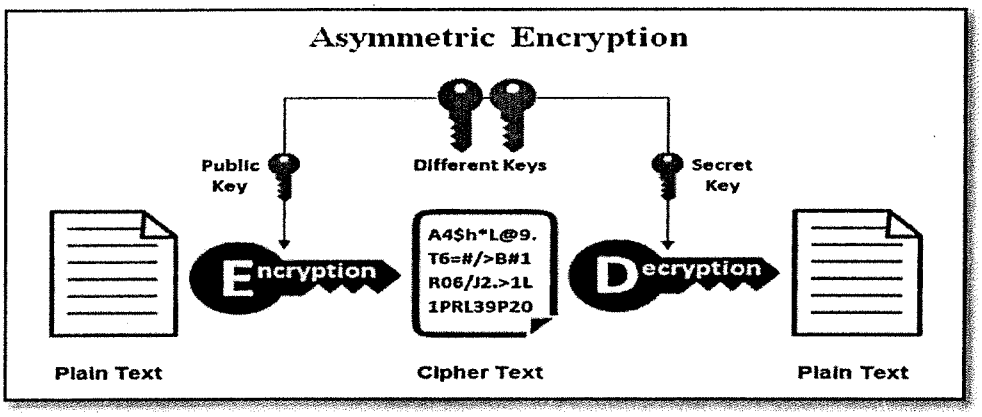
This is the simplest kind of encryption that involves only one secret key to cipher and decipher information. Symmetrical encryption is an old and best-known technique.

It uses a secret key known as 'Symmetric Key' that can either be a number, a word or a string of random letters. This key is applied to encode and decode the information. The sender uses this key before sending the message and the receiver uses it to decipher the encoded message.



ASYMMETRIC-KEYCRYPTOGRAPHY

- Asymmetric cryptography, also known as public key cryptography, uses public and private keys to encrypt and decrypt data.
- The keys are simply large numbers that have been paired together but are not identical (asymmetric).
- One key in the pair can be shared with everyone, it is called the *public key.*
- The other key in the pair is kept secret, it is called the *private key.*
- Either of the keys can be used to encrypt a message; the opposite key from the one used to encrypt the message is used for decryption.



Symmetric Encryption	Asymmetric Encryption
Symmetric encryption uses only one key for both encryption and decryption.	Asymmetric Encryption uses two different keys namely Public Key and Private Key.
Symmetric encryption is fast in execution.	Asymmetric Encryption is slow in execution due to the high computational burden
The symmetric encryption is used for bulk data transmission.	The asymmetric encryption is often used for securely exchanging secret keys.
Symmetric encryption is a simple technique compared to asymmetric encryption as only one key is employed to carry out both the operations.	Contribution from separate keys for encryption and decryption makes it a rather complex process.
Symmetric key cryptography utilizes less resource as compared to asymmetric key cryptography.	Asymmetric key cryptography utilizes more resource as compared to symmetric key cryptography.
Algorithms: RC4, AES, DES, 3DES, QUAD	Algorithms: RSA, DSA, Diffie-Hellman, ECC, El Gamal

STEGANOGRAPHY

- Steganography is data hidden within data.
- Steganography is an encryption technique in which hiding information within files that contains digital pictures or other images.
- Steganography techniques can be applied to images, a video file or an audio file.

The Forms of Steganography are:

i. Text Steganography
ii. Audio Steganography
iii. Video Steganography
iv. Images Steganography

Example:

- The sequence of first letters of each word of the overall message spells out the real (hidden) message.
  Since **E**veryone **C**an **R**ead, **E**ncoding **T**ext **I**n**N**eutral **S**entences **I**s **D**oubtfully **E**ffective ===→**SECRET INSIDE**
- Subset of the words of the overall message is used to convey the hiddenmessage.

Various other techniques have been used historically, some of them are:

- **Character marking** – selected letters of printed or typewritten text are overwritten inpencil. The marks are ordinarily not visible unless the paper is held to an angle to bright light.

- **Invisible ink–** a number of substances can be used for writing but leave no visible trace until heat or some chemical is applied to the paper.
- **Pin punctures–** small pin punctures on selected letters are ordinarily not visible unless the paper is held in front of the light.
- **Typewritten correction ribbon –** used between the lines typed with a black ribbon, the results of typing with the correction tape are visible only under a strong light.

Difference between Steganography and Cryptography:

Steganography	Cryptography
Steganography means cover writing	Cryptography means Secret writing
Steganography is less popular than Cryptography	While cryptography is more popular than Steganography
Attack's name in Steganography is Steganalysis.	While in cryptography, Attack's name is Cryptanalysis.
In Steganography, structure of data cannot be altered	While in cryptography, structure of data can be altered
Steganography supports Confidentiality and Authentication security principles	While cryptography supports Confidentiality and Authentication security principles as well as Data integrity and Non-repudiation
Steganography relies on parameter such as Key	While cryptography does not relies on any parameter

Questions asked from Chpater-1 in May-2019

1. Define cryptography.
2. Define computer security.
3. What do you mean by digital signature?
4. Define certificate authority.
5. Difference between passive and active attacks.
6. Explain Steganography.
7. Write a note on model for network security.
8. Explain the various forms of cryptographic techniques with neat labelled diagram.

Expected Questions from Chapter-1 in May-2020

1. What is cryptanalysis?
2. What is encryption?
3. What is decryption?
4. Explain cryptographic attacks.
5. Explain services and mechanisms of cryptography.
6. With neat diagram explain model of internetwork security.
7. Explain the various forms of cryptographic techniques with neat labelled diagram.
   Or
   Explain different types of cryptography techniques.
   Or
   Explain symmetric and asymmetric key cryptography.
8. Explain digital signature.
9. Difference between symmetric and asymmetric key cryptography.
10. Mention the difference between Steganography and cryptography.

CHAPTER – 2

SECURITY AT THE APPLICATION LAYER

2.1AUTHENTICATION

Authentication is a process that ensures and confirms a user's identity.

Any message authentication or digital signature mechanism has two levels of functionality. At the lower level, there must be some sort of function that produces an authenticator: a value to be used to authenticate a message. This lower -level function is then used as a primitive in a higher-level authentication protocol that enables a receiver to verify the authenticity of a message.

Types of Authentication Techniques

1. **Message encryption:** The ciphertext of the entire message serves as its authenticator.
2. **Message authentication code (MAC):** A function of the message and a secret key that produces a fixed-length value that serves as the authenticator.
3. **Hash function**: A function that maps a message of any length into a fixed length hash value, which serves as the authenticator.

1. Message Encryption:

Message encryption by itself can provide a measure of authentication. The analysis differs for symmetric and public-key encryption schemes.

2. Message authentication code (MAC):

- MAC is a key-dependent, one-way hash function that allows only specific recipients to access the message digest.
- MAC is an alternative authentication technique involves the use of a secret key to generate a small fixed size block of data, known as a cryptographic **checksum** or **MAC** that is appended to the message.
- **MAC = C (M, K)**, where M is the input message, C is the MAC function, K is theshared secret key, MAC is the message authentication code.

3. Hash Function:

- Hash function is a mathematical function that converts variable-length messages into a single fixed-length value.
- A hash function accepts a variable size message M as input and produces a fixed-size output, referred to as hash code **H(M)**.
- The main difference between Hash function and MAC is that, a hash code does not use any key.
- Value returned by hash function are called **message digest** or **simply hash value**.
- The hash value is appended to the message at the source at a time when the message is assumed or known to be correct.
- The receiver authenticates that message by re-computing the hash value.

The common authentication methods used for network security are:

1. Biometric Authentication
2. Token Authentication
3. Transaction Authentication
4. Two Way Factor Authentication or Multi-Factor Authentication (MFA)
5. Out-of-Band Authentication (OOB)

1. Biometrics for Network Security: Biometrics is refers to using the known and documented physical attributes of a user to authenticate their identity. Biometric authentication method includes fingerprint identification, voice recognition, retinal and iris scans and face scanning and recognition.

2. Token Authentication: A token is a material device that is used to access secure systems. A token makes it more difficult for a hacker to access an account since they must have long credentials and the tangible device itself, which is much harder for a hacker to obtain.

3. Transaction Authentication: Transaction authentication seeks out reasonable mistakes when comparing known data about a user with the details of a current transaction.

4. Multi-Factor Authentication (MFA): MFA is an authentication design that requires two or more independent ways of verifying identity. ATM's are prime examples of MFAs because you need a card (physical token) and a PIN (something known) in order for the transaction to take place.

5. Out-of-Band Authentication (OOB): OOB utilizes totally separate channels, like mobile devices, to authenticate transactions that originated on a computer. Any transaction that requires deposits from one place to another, like a large money transfer, would generate a phone call, text or notification on an app that there is more authenticationrequired for the transaction to be completed. With two necessary channels, it is much more difficult for a hacker to steal money.

2.2 KERBEROS

Kerberos is an authentication service designed for the use in a distributed environment. Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. Kerberos relies exclusively on conventional encryption, making no use of public-key encryption.

Use of Kerberos:

- Kerberos is used for decreasing the burden for server, means; Kerberos will take responsibility of authentication.
- It is designed for providing for strong authentication for client/server applications by using secret-key.

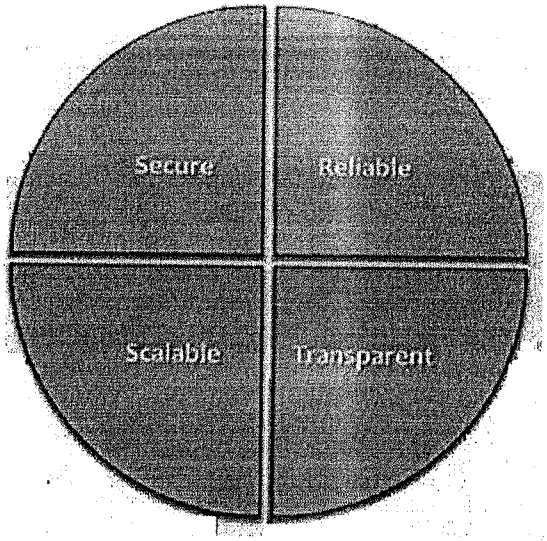
Versions:

- Kerberos Version4
- Kerberos Version5

Characteristics of Kerberos:

- It is secure: it never sends a password unless it is encrypted.
- Only a single login is required per session. Credentials defined at login are then passed between resources without the need for additional logins.
- The concept depends on a trusted third party – a Key Distribution Centre (KDC). The KDC is aware of all systems in the network and is trusted by all of them.
- It performs mutual authentication, where a client proves its identity to a server and a server proves its identity to the client.

Kerberos Requirements



The following are the requirements for Kerberos:

- **Secure:** A network eavesdropper should not be able to obtain the necessary information to impersonate a user. More generally, Kerberos should be strong enough that a potential opponent does not find it to be the weak link.

- **Reliable: S**hould be highly reliable and should employ distributed server architecture, with one system able to back up another.

- **Transparent:** Ideally, the user should not be aware that authentication is taking place, beyond the requirement to enter a password.

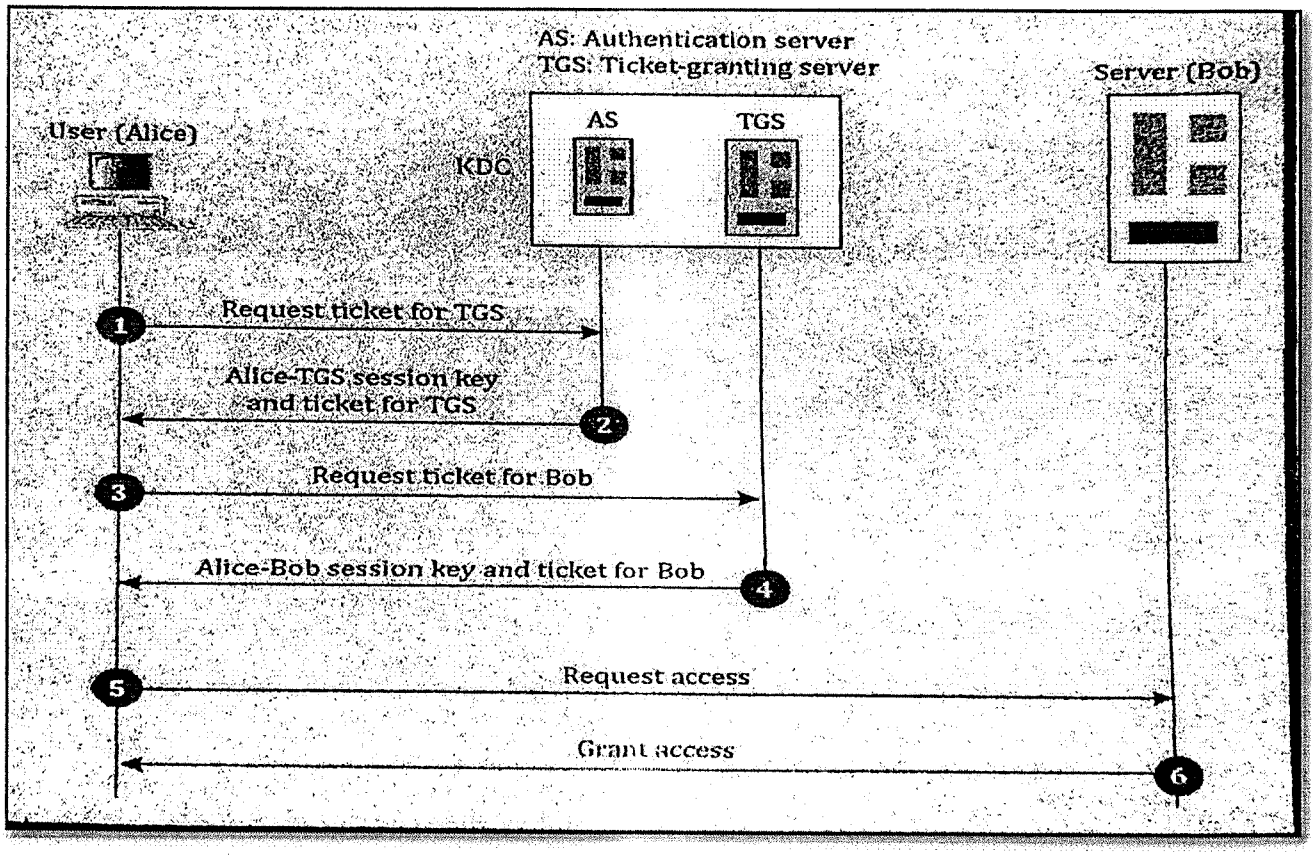
- **Scalable:** The system should be capable of supporting large numbers of clients and servers. This suggests a modular, distributed architecture.

Kerberos Operations

Kerberos is a network authentication protocol for client server applications. It is designed to provide strong authentication for client/server applications by using secret-key cryptography.

Main entities involved in Kerberos:

- **Client:** Initiates the communication for a service request. Acts on behalf of the user.
- **Server:** The server with the service the user wants to access.
- **Key Distribution Center (KDC):** The KDC is logically separated into three parts: Database (db), Authentication Server (AS) and Ticket Granting Server (TGS). Physically these 3 parts are existing in a single server and it is called as Key Distribution Center.
- **Authentication Sever (AS):** Performs client authentication. If the client is authenticated successfully the AS issues a ticket called TGT (Ticket Granting Ticket). TGT proves to other servers that client has been authenticated.
- **Ticket Granting Server (TGS):** An application server which provides the issuing of service tickets as a service.



A client process (Alice) can receive a service from a process running on the real server (Bob) in six steps:

Step 1: Alice sends her request to AS in plaintext, using her registered identity.

Step 2: The AS sends a message encrypted with Alice's symmetric key. The message contains two items: a session key Ks that is used by Alice to contact TGS and a ticket for TGS that is encrypted with the TGS symmetric key.

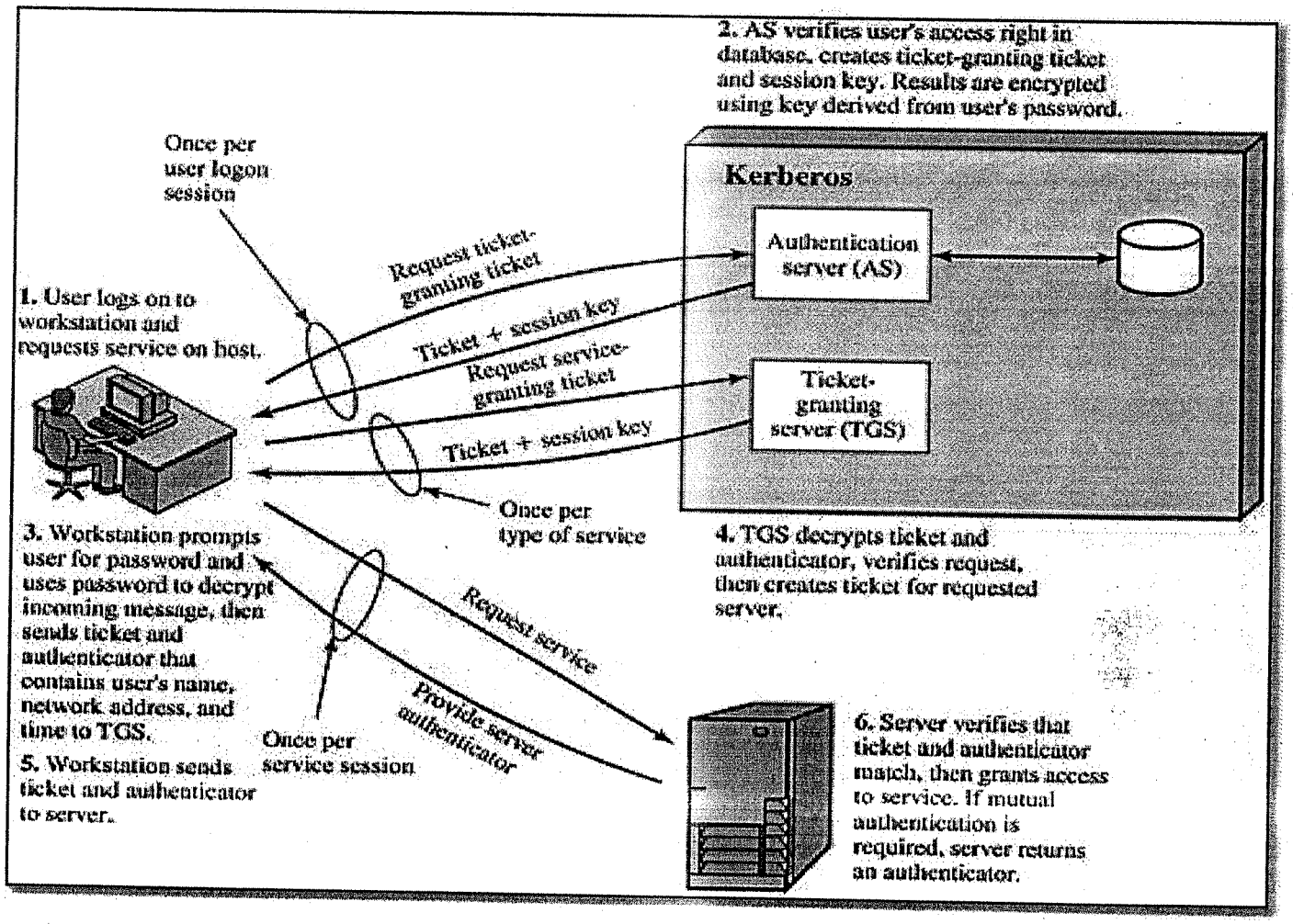
Step 3: Alice now sends three items to the TGS. The first is the ticket received from AS. The second is the name of the real server (Bob), and the third is a timestamp which is encrypted.

Step 4: Now, TGS sends two tickets, each containing the session key between Alice and Bob. The ticket for Alice is encrypted with Ks; the ticket for Bob is encrypted with Bob's key. She cannot replay step 3 because she cannot replace the time-stamp with a new one (she does not know Ks). Even if she is very quick and sends the step 3 messages before the time-stamp has expired, the still receives the same two tickets that she cannot decipher.

Step 5: Alice sends Bob's ticket with the encrypted time-stamp.

Step 6: Bob confirms the receipt by adding number one to the time-stamp. The message is encrypted and sent to Alice.

Kerberos 4 Overview



Differences between Kerberos version 4 and 5

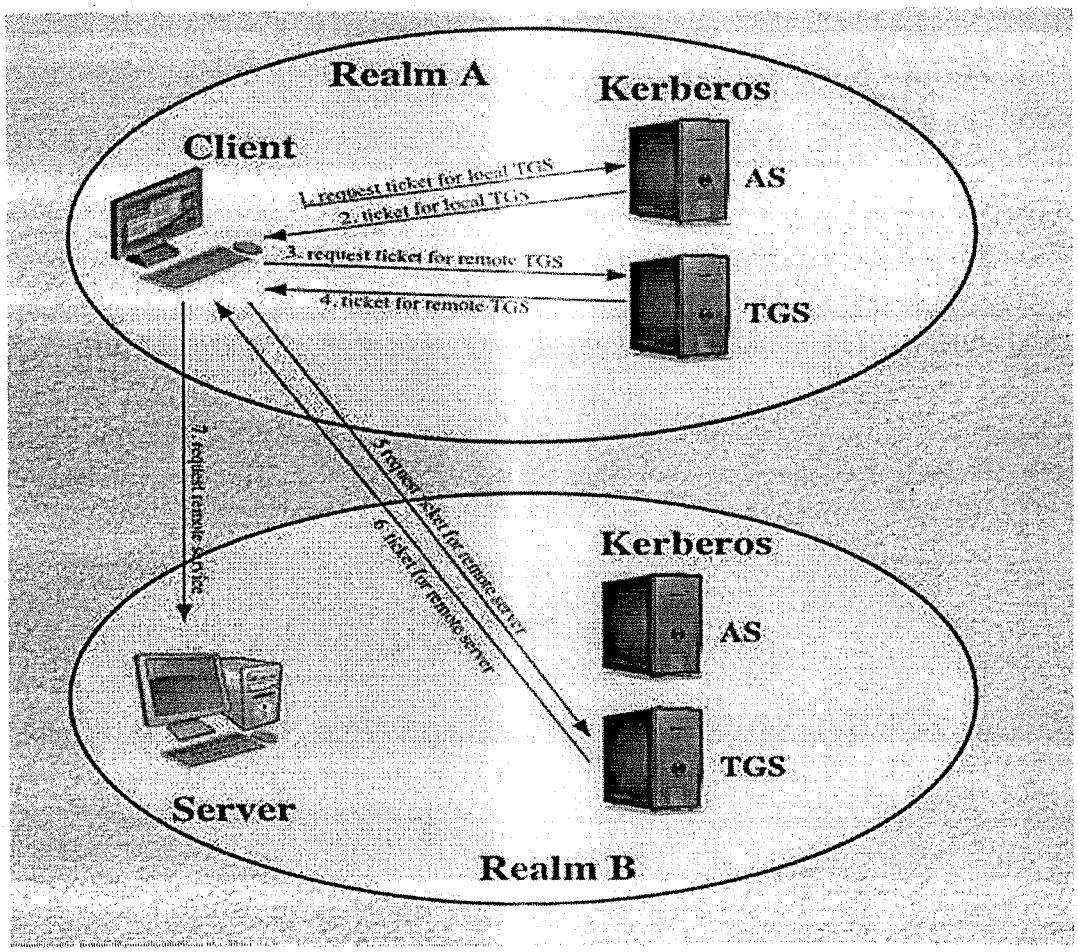
The minor difference between version 4 and version 5 are briefly listed below.

1. Version 5 has a longer ticket lifetime.
2. It allows tickets to be renewed.
3. It can accept any symmetric-key algorithm.
4. It uses a different protocol for describing data types.
5. It has more overhead than version 4.

Kerberos Realm

A Kerberos realm is a set of managed nodes that share the same Kerberos database.

Kerberos allows the global distribution of AS and TGS, with each system called realm. A user may get a ticket for local server or a remote server. In case of remote server, for example, Alice may ask her local TGS to issue a ticket that is accepted by a remote TGS. The local TGS can issue this ticket if remote TGS is registered with the local one. The Alice can use the remote TGS to access the remote real server.



Certification Authority (CA)

CA is a trusted entity that issues Digital Certificates and public-private key pairs. The role of the Certification Authority is to guarantee that the individual granted the unique certificate is, in fact, who he or she claims to be.

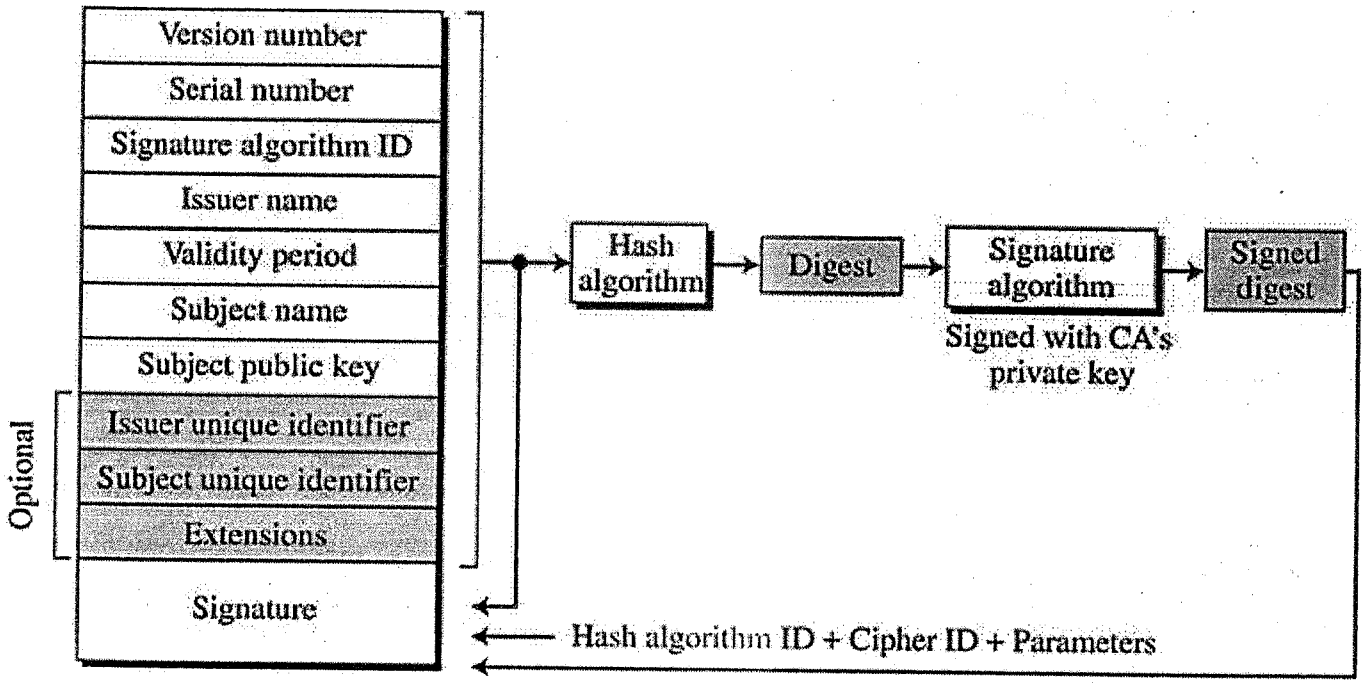
A Certificate Authority (CA) performs the following functions.

- Verifies the identity
- Issues digital certificates
- Maintains Certificate Revocation List (CRL)

2.3X.509CERTIFICATE

X.509 is a standard defining the format of public key certificates. X.509 certificates are used in Internet protocols, including TLS/SSL.An X.509 certificate contains a public key and an identity (a hostname, or an organization, or an individual), and is either signed by a certificate authority or self-signed.

The Format of X.509 certificate



- **Versionnumber:**The field is the version of X.509 (Current version is 3).
- **Serial number:**This field is the serial number assigned to each certificate and is unique for each certificate issuer.
- **Signature algorithm ID:**This field identifies signature algorithm used in the certificate. This field is repeatedin the signature field.
- **Issuer name:**This field identifies CA that created and signed this certificate.
- **Validity period:** Consists of two dates: the first and last on which the certificateis valid.
- **Subject name:**This field defines the entity that owns the public key stored in this certificate.
- **Subject's public-key:** The public key of the subject, plus an identifierof the algorithm for which this key is to be used, together with any associatedparameters.
- **Issuer unique identifier:** An optional-bit string field used to identify uniquelythe issuing CA in the event the X.500 name has been reused for differententities.
- **Subject unique identifier:** An optional-bit string field used to identify uniquely the subject in the event the X.500 name has been reused for different entities.
- **Extensions:** A set of one or more extension fields. Extensions were added in version 3 and are discussed later in this section.
- **Signature:** Covers all of the other fields of the certificate; it contains the hash code of the other fields encrypted with the CA's private key. This field includes the signature algorithm identifier.

Certificate Renewal: Each certificate has a period of validity. If there is no problem with the certificate, the CA issues a new certificate before the old one expires.

Certificate Revocation: In some cases a certificate must be revoked before its expiration (eg: The private key of the subject or CA has been compromised). The revocation is done periodically issuing a Certification Revocation List (**CRL**) that contains all revoked certificate that have not expired on the date the CRL is issued. To ensure the validity of a certificate, the user must check the latest CRL published by CA that issued the certificate.

CRL has the following fields: Signature algorithm ID, Issuer name, This update date, Next update date, Revoked certificate, Signature.

CA Hierarchy Use

In the example given below , user A can acquire the following certificates from the directory to establish a certification path to B:

X<<W>> W <<V>> V <<Y>><<Z>> Z <<B>>

When A has obtained these certificates, it can unwrap the certification path in sequence to recover a trusted copy of B's public key. Using this public key, A can send encrypted messages to B. If A wishes to receive encrypted messages back from B, or to sign messages sent to B, then B will require A's public key, which can be obtained from the following certification path:

Z<<Y>> Y <<V>> V <<W>> W <<X>>X <<A>>

B can obtain this set of certificates from the directory, or A can provide them as part of its initial message to B.

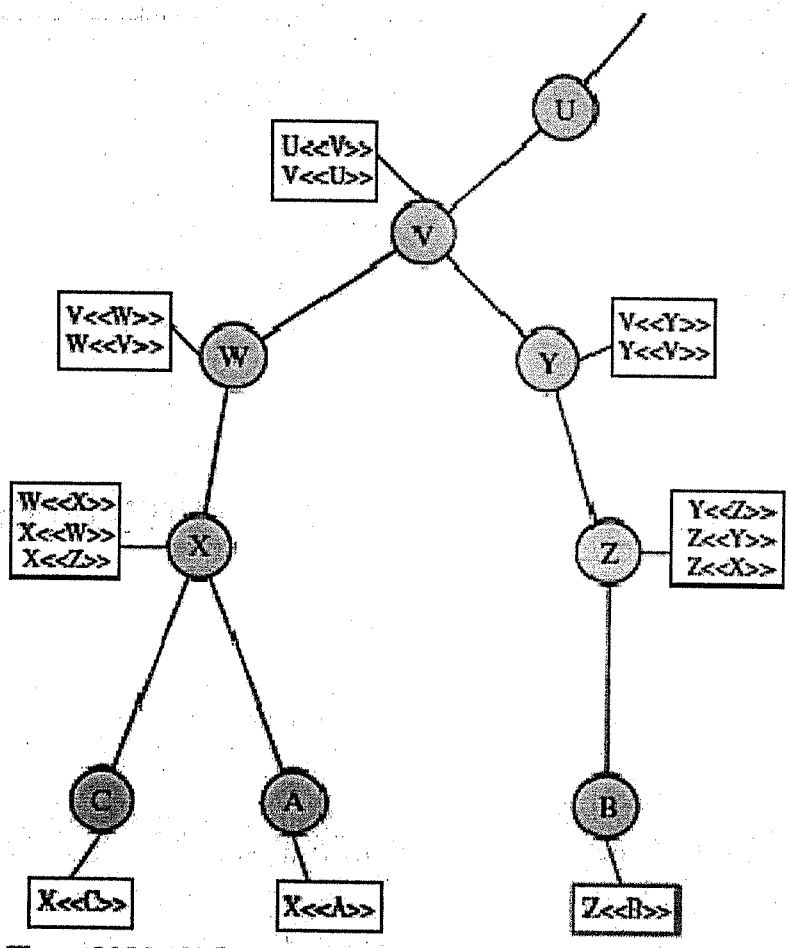
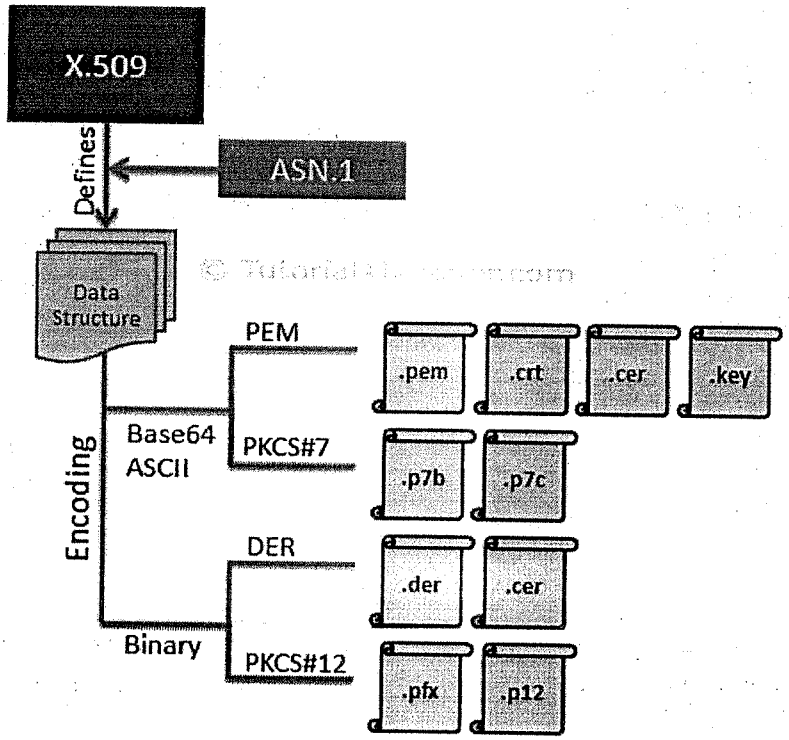


Figure 14.16 X.509 Hierarchy: A Hypothetical Example

X.509 Encoding formats and File extensions

There are different formats of X.509 certificates such as PEM, DER, PKCS#7 and PKCS#12. PEM and PKCS#7 formats use Base64 ASCII encoding while DER and PKCS#12 use binary encoding. The certificate files have different extensions based on the format and encoding they use.

The following figure illustrates the X.509 Certificate's encoding formats and file extensions.



PEM Format

Most CAs (Certificate Authority) provide certificates in PEM format in Base64 ASCII encoded files. The certificate file types can be .pem, .crt, .cer, or .key. The .pem file can include the server certificate, the intermediate certificate and the private key in a single file. The server certificate and intermediate certificate can also be in a separate .crt or .cer file. The private key can be in a .key file.

PKCS#7 Format

The PKCS#7 format is a Cryptographic Message Syntax Standard. The PKCS#7 certificate uses Base64 ASCII encoding with file extension .p7b or .p7c. Only certificates can be stored in this format, not private keys.

DER Format

The DER certificates are in binary form, contained in .der or .cer files. These certificates are mainly used in Java-based web servers.

PKCS#12 Format

The PKCS#12 certificates are in binary form, contained in .pfxor .p12 files.

The PKCS#12 can store the server certificate, the intermediate certificate and the private key in a single .pfx file with password protection. These certificates are mainly used on the Windows platform.

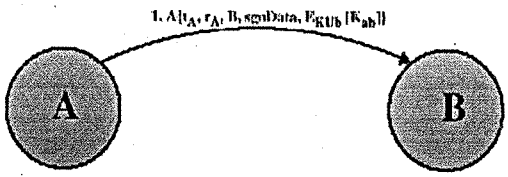
X.509 Authentication Procedures

X.509 includes three alternative authentication procedures:

1. One-Way Authentication
2. Two-Way Authentication
3. Three-Way Authentication

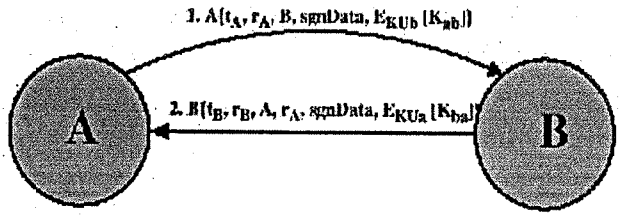
1. One-Way Authentication

- 1 message (A->B) used to establish:
  - The identity of A and that message is from A
  - Message was intended for B
  - Integrity & originality of message



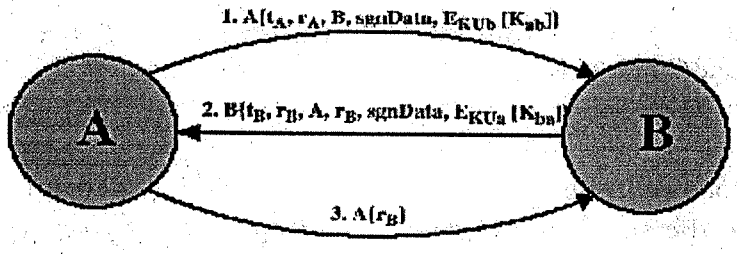
2. Two-Way Authentication

- 2 messages (A->B, B->A) which also establishes in addition:
  - The identity of B and that reply is from B
  - That reply is intended for A
  - Integrity & originality of reply



3. Three Way Authentication

- 3 messages (A->B, B->A, A->B) which enables above authentication without synchronized clocks
- Has reply from A back to B containing signed copy of nonce from B
- Eliminates the need to check timestamps.



2.4 E-MAIL

E-mail stands for electronic mail, is one of the most widely used features of the internet, along with the web. It allows you to send/receive messages to and from anyone with the mail address anywhere in the world.

Email messages are usually encoded in ASCII text. However, we can send non-text files such as graphic images and sound files, as attachments send in binary strings.

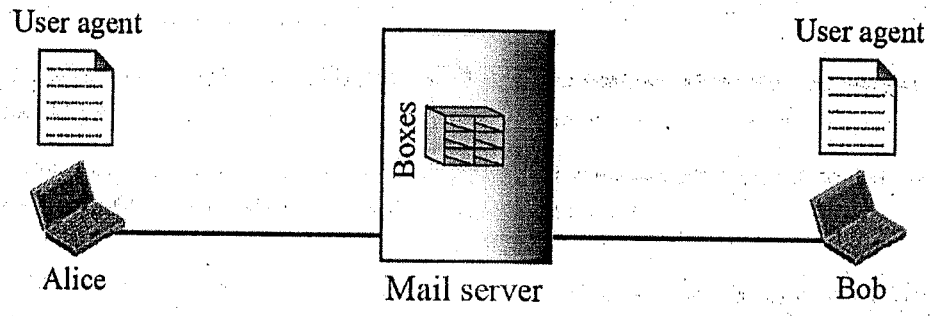
Email uses multiple protocols within TCP/IP protocol suite. A popular protocol for sending Email is SMTP(Simple Mail Transfer Protocol) & popular protocol for receiving is POP3(Post Office Protocol) & IMAP(Internet Mail Access Protocol).

E-mail Architecture

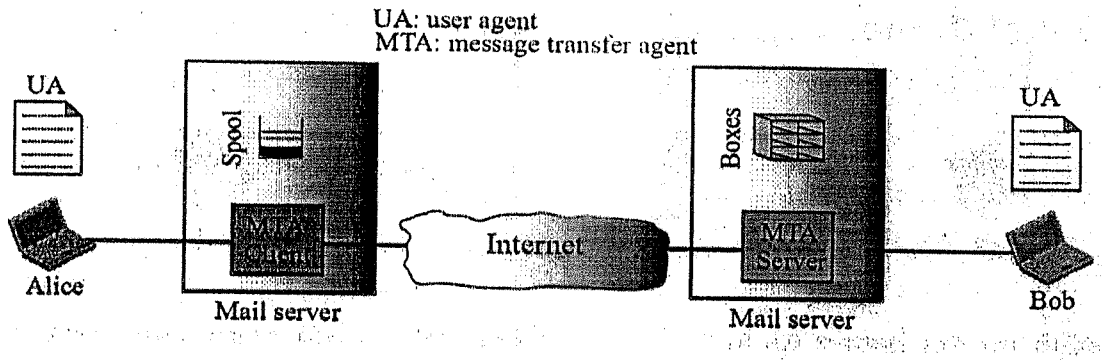
To explain the architecture of email, we divide architecture into four scenarios. The email system architecture is illustrated in following components:

- **User agent:** Software program that composes, reads, replies to and forward messages. It also handles mailboxes.
- **Message Transfer Agent(MTA):** The actual mail transfer is done through MTA. SMTP is example for MTA.
- **Message Access Agent(MAA):** This software pulls messages out of mailbox.
  - POP3 and IMAP4 are the examples for the MAA.

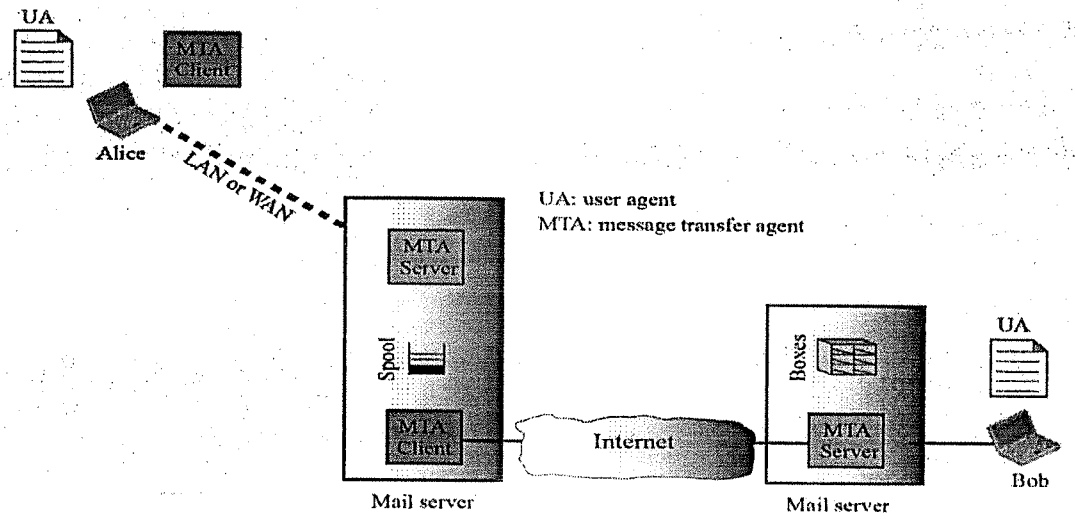
1. First Scenario: When the sender and the receiver of an e-mail are on the same mail server, we need only two user agents.



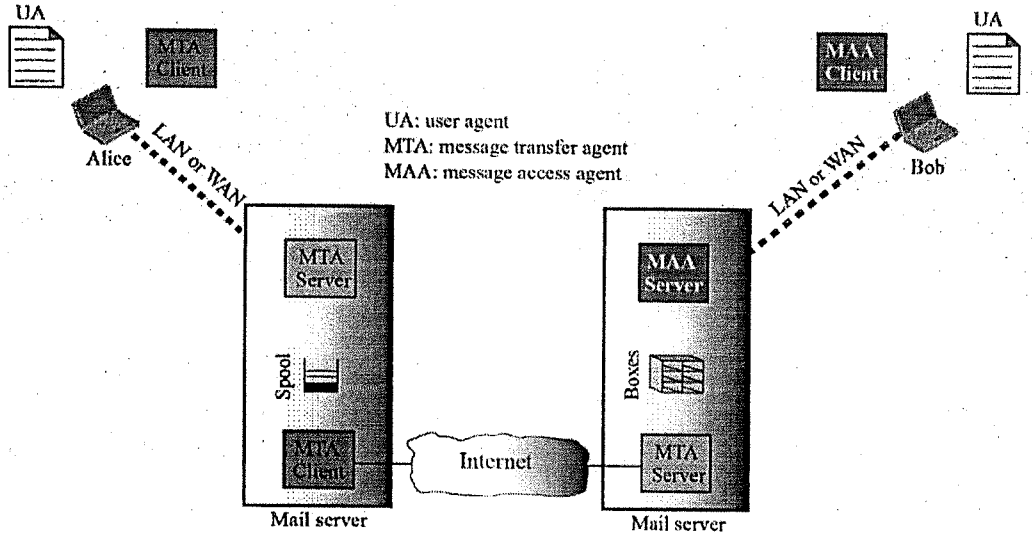
2. Second Scenario: When the sender and the receiver of an e-mail are on different mail servers,we need two UAs and a pair of MTAs (client and server).



3. Third Scenario: When the sender is connected to the mail server via a LAN or a WAN, we need two UAs and two pairs of MTAs (client and server).



4. Fourth Scenario: When both sender and receiver are connected to the mail server via a LAN or a WAN, we need two UAs, two pairs of MTAs (client and server), and a pair of MAAs (client and server). This is the most common situation today.



Electronic Mail Security

The protection of email from unauthorized access and inspection is known as electronicprivacy. There are mainly two methods for proving security for electronic mails:

- Pretty Good Privacy.
- S/MIME

PRETTY GOOD PRIVACY (PGP)

PGP provides the confidentiality and authentication service that can be used for electronic mail and file storage applications. The steps involved in PGP are:

- Select the best available cryptographic algorithms as building blocks.
- Integrate these algorithms into a general purpose application that is independent of operating system and processor and that is based on a small set of easy-to-use commands.
- Make the package and its documentation, including the source code, freely available via the internet, bulletin boards and commercial networks.
- Enter into an agreement with a company to provide a fully compatible, low cost commercial version of PGP.

Notations

The notations used in this chapter are:

Ks = session key used in symmetric encryption scheme
PRa = private key of user A, used in public-key encryption scheme
PUa= public key of user A, used in public-key encryption scheme
EP = public-key encryption
DP = public-key decryption
EC = symmetric encryption
DC = symmetric decryption
H = hash function
|| = concatenation

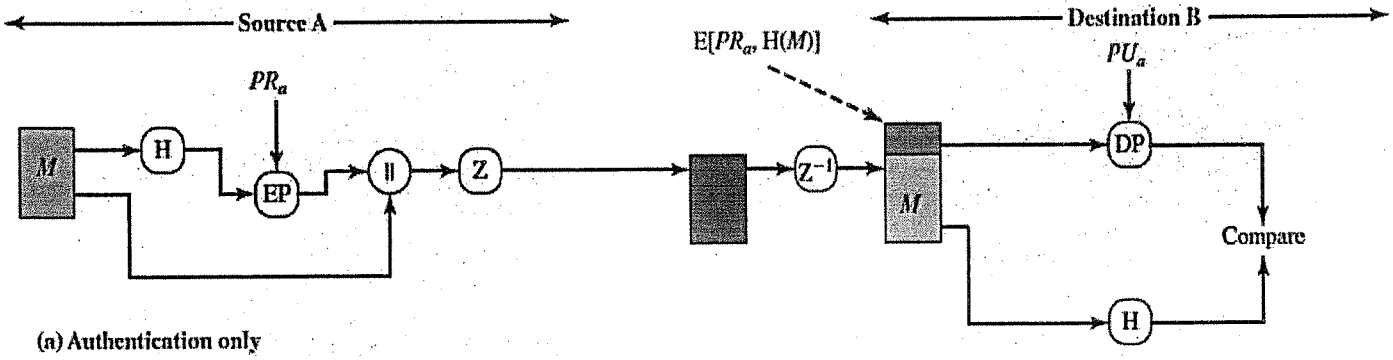
Operational description

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

1. Authentication:

The sequence for authentication is as follows:

- Consider a plain text message.Apply Hash function (independent of key).
- Encrypt hash code using private key of User A. Encrypted code is embedded with plain text message.
- Apply Zip function(Z) & send it to the receiver.
- At the receiver's side apply unzip function(Z^{-1}). Will get encrypted form of message with private key of sender and hash code.
- Apply Description algorithm, using public key of User A. Generated hash code is compared with hash code of plain text.
- If the two hash code matches, the message is accepted as authentic.



(a) Authentication only

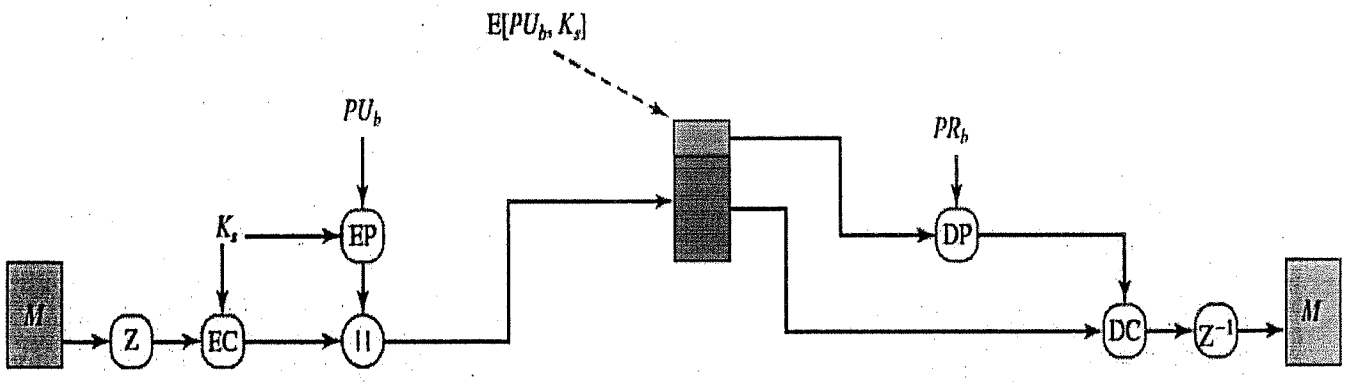
2. Confidentiality:

Confidentiality is provided by encrypting messages to be transmitted or to be stored locally as files.

In PGP, each conventional key is used only once. Thus although this is referred to as a session key, it is in reality a one-time key. To protect the key, it is encrypted with the receiver's public key.

The sequence for confidentiality is as follows:

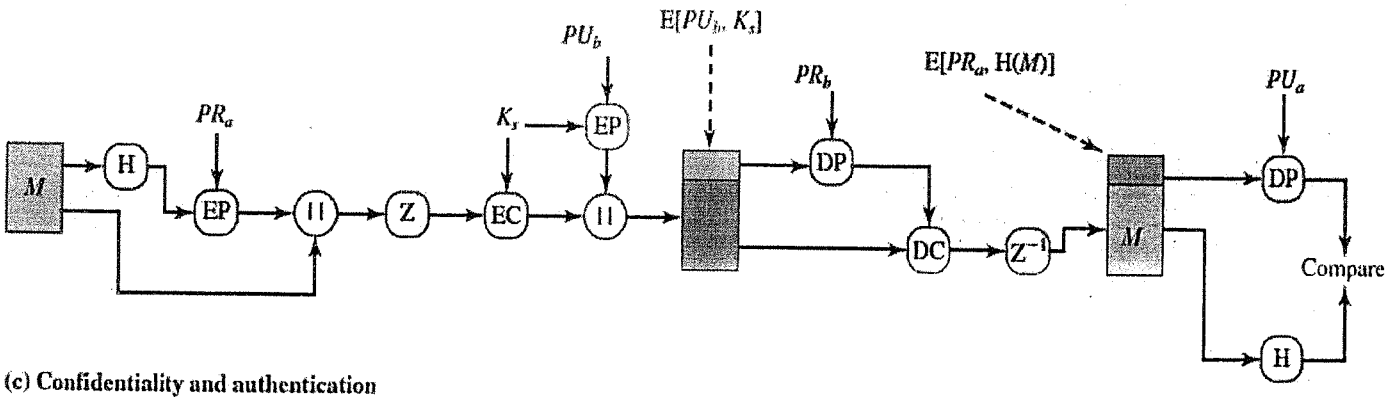
- Consider a plaintext message. Apply zip function to it.
- Apply conventional encryption using session key.
- The session key is encrypted, using the receiver's public key and is appended to the encrypted message.
- The receiver uses private key to decrypt and recover the session key.
- The session key is used to decrypt the message. Apply unzip function to get the plaintext message.



(b) Confidentiality only

3. Confidentiality and authentication

Here both services may be used for the same message. First, a signature is generated for the plaintext message and appended to the message. Then the plaintext plus the signature is encrypted and the session key is encrypted.



(c) Confidentiality and authentication

4. Compression

As a default, PGP compresses the message after applying the signature but before encryption. This has the benefit of saving space for both e-mail transmission and for file storage. Z is used for compression and Z^{-1} is used for decompression.

The signature is generated before compression for two reasons:

- It is preferable to sign an uncompressed message so that one can store only the uncompressed message together with the signature for future verification.
- If we generate signature after compression, then there is no need for the recompression for the message verification.

Message encryption is applied after compression to strengthen cryptographic security. Because the compressed message has less redundancy than the original plaintext, cryptanalysis is more difficult. The compression algorithm used is ZIP.

5. E-mail compatibility

- Many electronic mail systems only permit the use of blocks consisting of ASCII texts.
- To accommodate this restriction, PGP provides the service of converting the raw 8-bit binary stream to a stream of printable ASCII characters.
- The scheme used for this purpose is radix-64 conversion. Each group of three octets of binary data is mapped into four ASCII characters.

E.g: consider the 24-bit (3 octets) raw text sequence 00100011 01011100 10010001, we can express this input in block of 6-bits to produce 4 ASCII characters.

001000	110101	110010	010001
I	L	Y	R => corresponding ASCII characters

6. Segmentation and reassembly

E-mail facilities often are restricted to a maximum length. E.g., many of the facilities accessible through the internet impose a maximum length of 50,000 octets. Any message longer than that must be broken up into smaller segments, each of which is mailed separately.

To accommodate this restriction, 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 the other processing, including the radix-64 conversion. At the receiving end, PGP must strip off all e-mail headers and reassemble the entire original block before performing the other steps.

Cryptographic keys and key rings

There are four different keys used by PGP are:

1. One-time session symmetric keys
2. Public keys
3. Private keys
4. Passphrase-based symmetric keys

Three separate requirements can be identified with respect to these keys:

- A means of generating unpredictable session keys is needed.
- It must allow a user to have multiple public key/private key pairs.
- Maintain a file of public/private key pairs.

1. Session key generation

- Each session key is associated with a single message and is used only for the purpose of encryption and decryption of that message.
- Random 128-bit numbers are generated using CAST-128 itself.
- Input to the random number generator consists of a 128-bit key and two 64-bit blocks that are treated as plaintext to be encrypted.
  - ✓ Input is determined by the keystrokes and the times keystrokes are made.
  - ✓ Input is also effected by previous key outputs.

2. Key identifiers

With multiple private/public key pairs, there needs to be a way for the receiver to know which to use:

- ✓ How this is done is through the combination of a 64 bit key ID, which is unique to a user ID.
- ✓ With this key ID, the receiver can retrieve the correct public key of the sender to decrypt the message.
- ✓ A list of these key ID's are placed in what is called a key ring.

3. Key rings

PGP provides a pair of data structures at each node, one to store the public/private key pair owned by that node and one to store the public keys of the other users known at that node. These data structures are referred to as private key ring and public key ring.

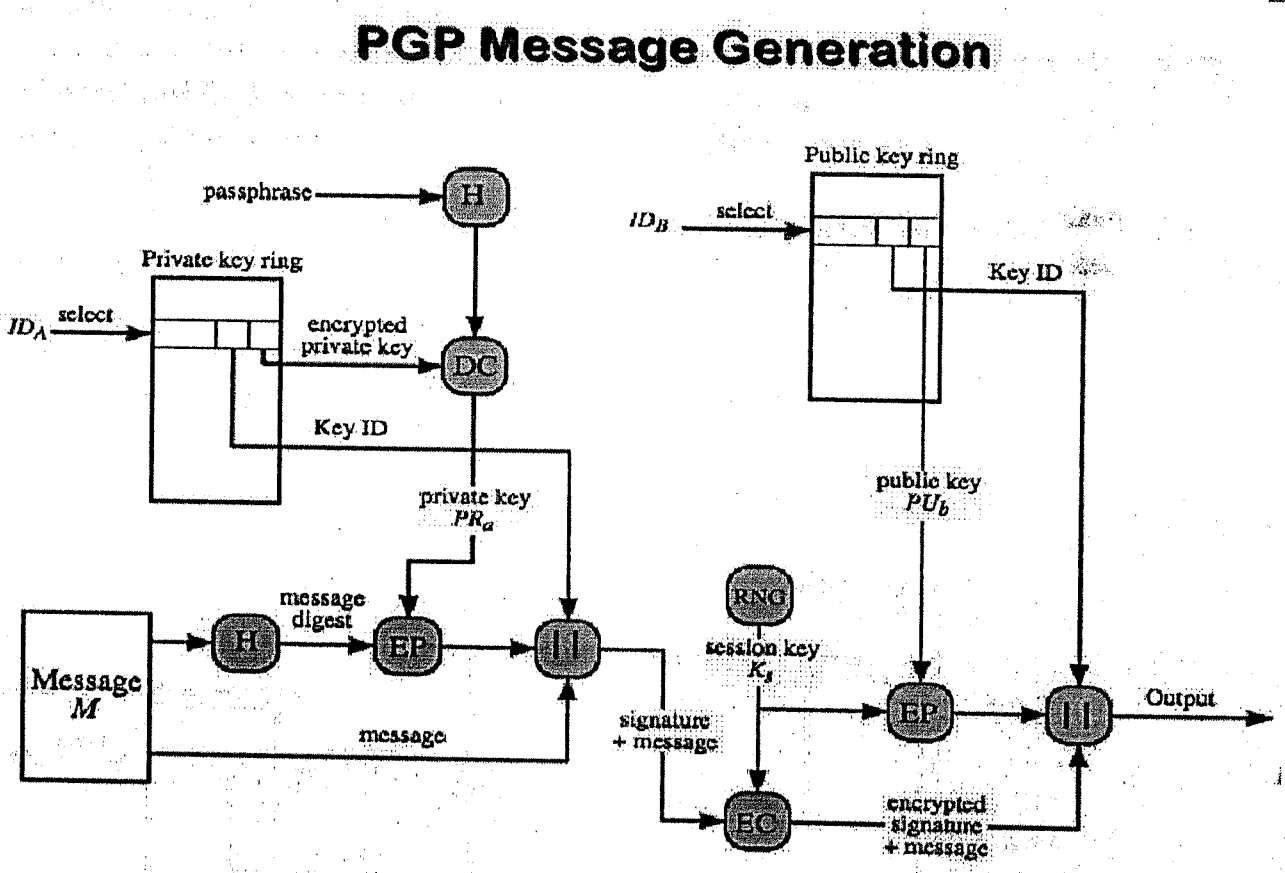
A message consists of three components.

1. **Message component** – includes actual data to be transmitted, as well as the filename and a timestamp that specifies the time of creation.

2. **Signature component** – includes the following
   - **Timestamp** – time at which the signature was made.
   - **Message digest** – hash code.
   - **Two octets of message digest** – to enable the recipient to determine if the correct public key was used to decrypt the message.
   - **Key ID of sender's public key** – identifies the public key.

3. **Session key component** – includes session key and the identifier of the recipient public key.

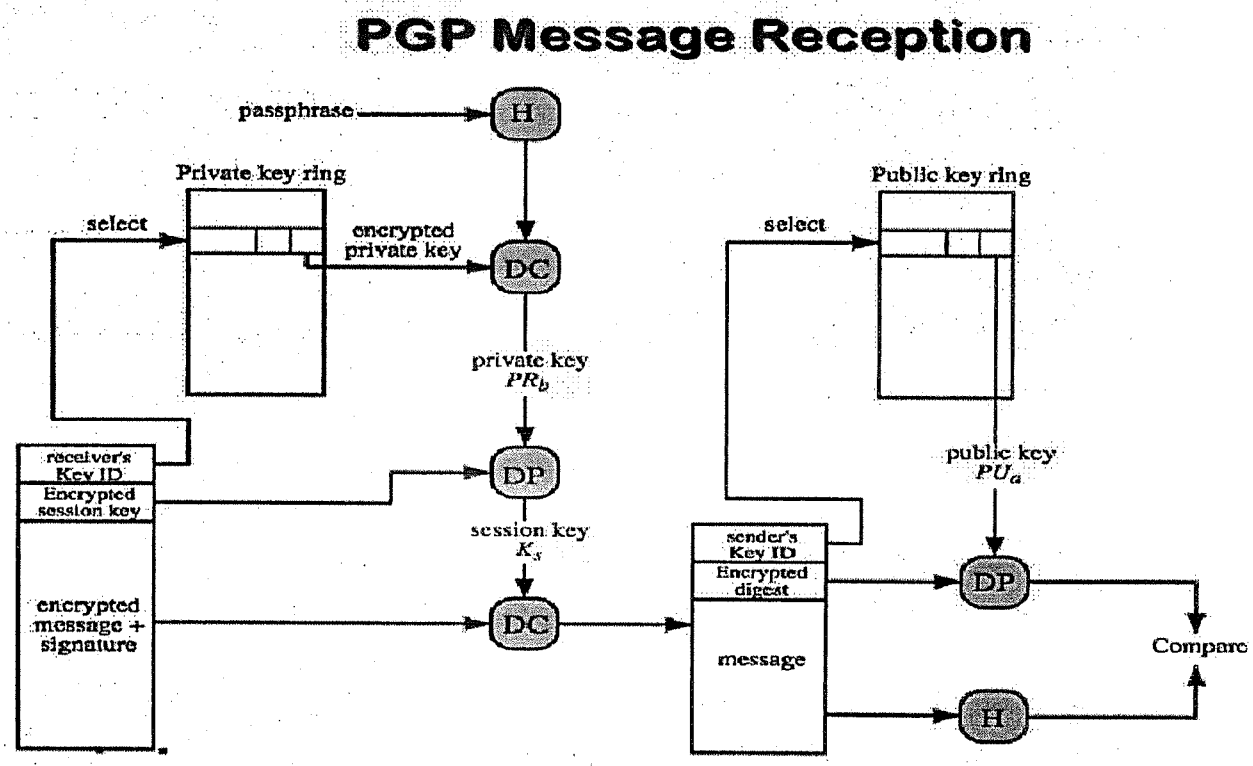
PGP MESSAGE GENERATION



First consider message transmission and assume that the message is to be both signed and encrypted. The sending PGP entity performs the following steps:

1. **Signing themessage**
   - PGP retrieves the sender"s private key from the private key ring using user ID as an index.
   - PGP prompts the user for the passphrase (password) to recover the unencrypted private key.
   - Thesignature component of the messageis constructed.
2. **Encrypting themessage**
   - PGP generates a session key and encrypts the message.
   - PGP retrieves the recipient's public key from the public key ring using user ID as index
   - The session key component of the message is constructed.

PGP MESSAGE RECEPTION



1. **Decrypting the message**
   - PGP retrieves the receiver's private key from the private key ring, using the key ID field in the session key component of the message as an index.
   - PGP prompts the user for the passpharse (password) to recover the unencrypted private key.
   - PGP then recovers the session key and decrypts the message.
2. **Authenticating the message**
   - PGP retrieves the sender's public key from the public key ring, using the key ID field in the signature key component of the message as an index.
   - PGP recovers the transmitted message digest.
   - PGP computes the message digest for the received message and compares it to the transmitted message digest to authenticate.

S/MIME

- **Secure/Multipurpose Internet Mail Extension** is a security enhancement to the MIME internet email standard, based on technology from R.S.A Data security.
- S/MIME is for industry standard for commercial and organizational use.
- It defined in number of documents that is RFC 2630, RFC 2632, RFC 2633

Multipurpose Internet Mail Extensions (MIME) is an extension to the RFC 5322 framework that is intended to address some of the problems and limitations of the use of Simple Mail Transfer Protocol (SMTP).

The following are limitations of the SMTP/5322 scheme.

1. SMTP cannot transmit executable files or other binary objects.
2. SMTP cannot transmit text data that includes national language characters, because these are represented by 8-bit codes and SMTP is limited to 7-bit ASCII.
3. SMTP servers may reject mail message over a certain size.
4. SMTP gateways that translate between ASCII and the character code EBCDIC do not usea consistent set of mappings, resulting in translation problems.
5. SMTP gateways to X.400 electronic mail networks cannot handle non-textual data included in X.400 messages.
6. Some SMTP implementations do not adhere completely to the SMTP standards defined in RFC 821. Common problems include:
   a. Deletion, addition, or reordering of carriage return and linefeed
   b. Truncating or wrapping lines longer than 76 characters
   c. Removal of trailing white space (tab and space characters)
   d. Padding of lines in a message to the same length
   e. Conversion of tab characters into multiple space characters

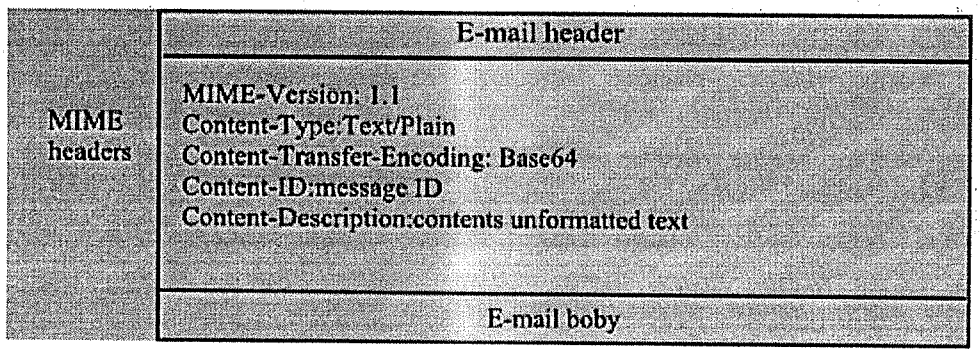
MIME is intended to resolve these problems in a manner that is compatible with existing RFC 5322 implementations. The specification is provided in RFCs 2045 through 2049.

Overview of MIME

The MIME specification includes the following elements:

1. **Five new message headers:** Five message header fields are defined. These fields provide information about the body of the message.
2. **A number of Content formats:** A number of content formats are defined, thus standardizing representations that support multimedia electronic mail.
3. **Transfer encodings:** Transfer encoding are defined that enable the conversion of any content format into a form that is protected from alteration by the mail system.

1. Five new Message header:



The five header fields defined in MIME are

- **MIME-Version:** Must have the parameter value 1.0. This field indicates that the messageconforms to RFCs 2045 and 2046.
- **Content-Type:** Describes the data contained in the body with sufficient detail.
- **Content-Transfer-**Encoding: Indicates the type of transformation that has been used torepresent the body of the message in a way that is acceptable for mail transport.
- **Content-ID:** Used to identify MIME entities uniquely in multiple contexts.
- **Content-Description:** A text description of the object with the body; this is useful whenthe object is not readable (e.g., audio data).

2. MIME Content formats:

The bulk of the MIME specification is concerned with the definition of a variety of contenttypes. This reflects the need to provide standardized ways of dealing with a wide variety ofinformation representations in a multimedia environment.

Type	*Subtype*	*Description*
Text	Plain	Unformatted
	HTML	HTML format (see Appendix E)
Multipart	Mixed	Body contains ordered parts of different data types
	Parallel	Same as above, but no order
	Digest	Similar to Mixed, but the default is message/RFC822
	Alternative	Parts are different versions of the same message
Message	RFC822	Body is an encapsulated message
	Partial	Body is a fragment of a bigger message
	External-Body	Body is a reference to another message
Image	JPEG	Image is in JPEG format
	GIF	Image is in GIF format
Video	MPEG	Video is in MPEG format
Audio	Basic	Single channel encoding of voice at 8 KHz
Application	PostScript	Adobe PostScript
	Octet-stream	General binary data (eight-bit bytes)

3. Transfer encodings:

The other major component of the MIME specification, in addition to content type specification, is a definition of transfer encodings for message bodies. The objective is to provide reliable delivery across the largest range of environments.

Type	*Description*
7bit	NVT ASCII characters and short lines
8bit	Non-ASCII characters and short lines
Binary	Non-ASCII characters with unlimited-length lines
Base64	6-bit blocks of data are encoded into 8-bit ASCII characters
Quoted-printable	Non-ASCII characters are encoded as an equal sign plus an ASCII code

S/MIME functionality

In terms of general functionality, S/MIME is very similar to PGP. Both offer the ability to sign and/or encrypt messages. In this subsection, we briefly summarize S/MIME capability. S/MIME provides the following functions.

- **Enveloped data:** This consists of encrypted content of any type and encrypted-content encryption keys for one or more recipients.
- **Signed data:** A digital signature is formed by taking the message digest of the content to be signed and then encrypting that with the private key of the signer. The content plus signature are then encoded using base64 encoding. A signed data message can only be viewed by a recipient with S/MIME capability.
- **Clear-signed data:** As with signed data, a digital signature of the content is formed. However, in this case, only the digital signature is encoded using base64.As a result, recipients without S/MIME capability can view the message content, although they cannot verify the signature.
- **Signed and enveloped data:** Signed-only and encrypted-only entities may be nested, so that encrypted data may be signed and signed data or clear-signed data may be encrypted.

CHAPTER – 3

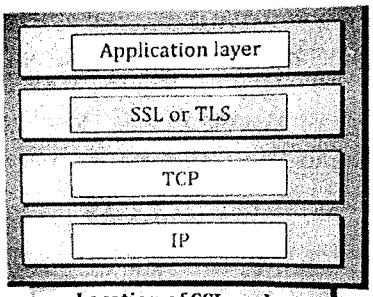
SECURITY AT TRANSPORT LAYER

Transport layer security provides end-end services to application that uses a reliable transport layer protocol such as TCP. The aim is to provide security services for transactions on the internet.

SECURE SOCKET LAYER/TRANSPORT LAYER SECURITY (SSL/TLS)

Secure Socket Layer (SSL) was developed by Netscape to provide security when transmitting information on the Internet. SSL ensures that all data passed between them remains private and free from attack.

- SSL provides security services between TCP and applications that use TCP. The Internet standard version is called Transport Layer Service (TLS).
- SSL/TLS provides **authentication**, **confidentiality** using symmetric encryption and **message integrity** using a message authentication code.
- SSL was designed to permit web browsers and web servers to exchange sensitive information and prevent programs that could view the network traffic from reading the sensitive data.
- TLS (Transport Layer Security) is just an updated, more secure, version of SSL.



Location of SSL and TLS in Internet Model

SSL ARCHITECTURE

SSL is designed to provide security and compression services to data generated from application layer:

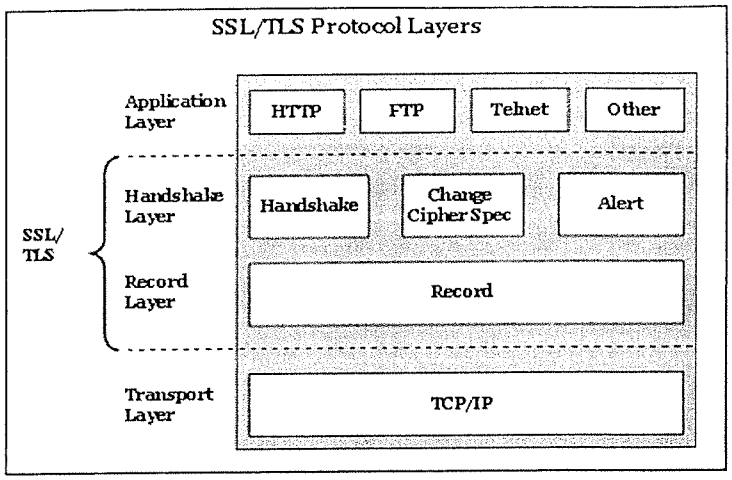
Services provided by SSL are:

- **Fragmentation:** Each upper layer messages are fragmented into block of 2^{14} bytes or less.
- **Compression:** Each fragment of data is compressed using one of the optionally applied. Compression must be lossless negotiated betweenclient and server.
- **Message Integrity:** To preserve the integrity of data, SSL uses keyed-hash function to create MAC.
- **Confidentiality:** To provide confidentiality, the original data and the MAC encrypted using symmetric encryption.
- **Framing:** A header is added to the encrypted payload. The payload is then passed to a reliable transport layer protocol.

FOUR PROTOCOLS

The four protocols of SSL are:

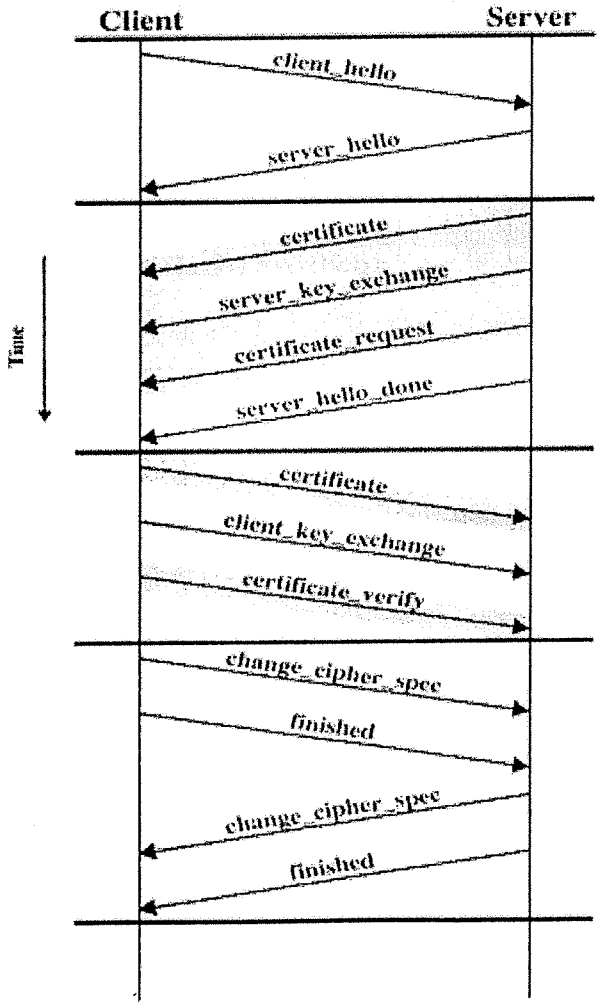
1. The Handshake Protocol
2. The SSL Record Protocol
3. The Change Cipher Spec Protocol
4. The Alert Protocol



1. The Handshake Protocol:

SSL Handshake protocol allows following between client and Server. The handshake is done before any data is transmitted

- To authenticate each other
- To negotiate encryption and MAC algorithms
- To create cryptographic keys to be used
- To establish a session and then a connection.



Phase 1: Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.

Phase 2: Server may send certificate, key exchange, and request certificate. At this stage, server signals end of the hello message phase.

Phase 3: Client transmits certificate if needed. Client transmits key exchange. Client may transmit certificate verification.

Phase 4: Change cipher suite and finish handshake protocol.

Note: **Shaded transfers are optional or situation dependent messages that are not always sent.**

There are four phases in SSL handshake protocol. Following series of messages are used in these 4 phases.

- **Phase-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, ***hello message***with different parameters. The 32 bit random number is generated the server which is independent of client random filed and sends ***server_hello*** message.

- **Phase-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.
  The ***server_key_exchange***message may sent if it is required.
  The second parameter in the ***certificate_request*** message is a list of distinguished names of acceptable certificate authorities.
  The final message in Phase 2 which is always required is the***server_done message*** which is sent by the server to indicate the end of the server hello and associated messages. After sending the message the server will wait for the client response.

- **Phase-3: Client Authentication and Key Exchange**
  After receiving the ***server_done*** message, the client should verify that the server provided a valid certificate if required and check that the ***server _hello*** parameters are acceptable. If all satisfactory the client sends one or more messages.
  The client begins this phase by sending a ***certificate message***. If no suitable certificate is available, the client sends a ***no_certificate***alert instead.
  Finally the client send **certificate_verify** message to explicit verification of a client certificate.

- **Phase-4: Finish**
  The client sends a **change_cipher_spec** message and copies the pending cipher spec into the current cipher spec.
  The client then immediately sends the **finished message**. The finished message verifies that the key exchange and authentication process were successful.

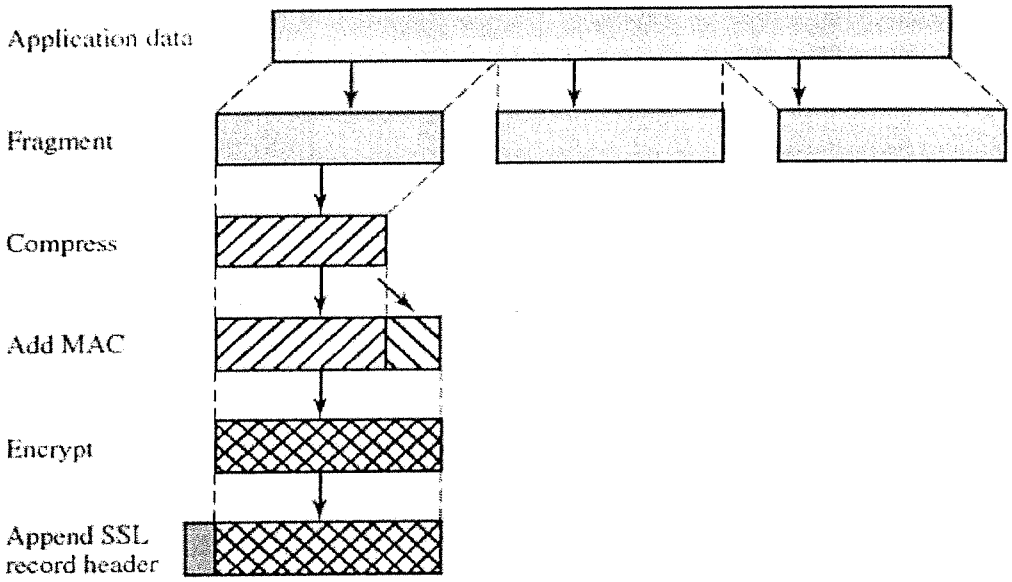
Following table mentions message types used in SSL handshake protocol between client and server.

Message Type	Parameters
•Hello_request	•Null
•Client_hello	•version random •session Id •cipher suite •compression method
•Server_hello	•version random •session Id •cipher suite •compression method
•Certificate	•Chain of X.509-v3 certificates
•Server_key_exchange	•Parameters •signature,
•Certificate_request	•type •authorities
•Server_done	•NULL
•Certificate_Verify	•Signature
•Client_Key_exchange	•parameters •signature
•Finished	•Hash Value

2. SSL Record Protocol:

The SSL Record Protocol provides two services for SSL connections:

- **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 message authentication code (MAC).



SSL Record Protocol Operation

Operations of SSL Record Protocol are:

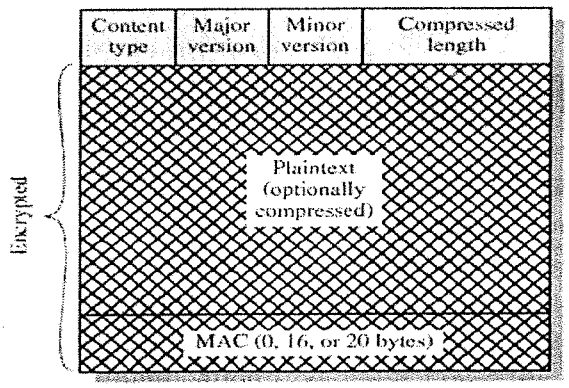
Step 1: The first step is fragmentation. Each upper-layer message is fragmented into blocks of 2^{14} bytes (16,384 bytes) or less. Next, compression is optionally applied.

Step 2: The compression is optionally applied. Compression must be lossless and may not increase the content length by more than 1924 bytes. The default compression algorithm is null.

Step 3: To compute Message Authentication Code (MAC) over the compressed data. For this purpose a shared secret key is used.

Step 4: The compressed message plus the MAC are encrypted using symmetric encryption. Encryption may not increase the content length by more than 1024 bytes, so that the total length my not exceed 214 + 2048bits

Step 5: The final step of SSL record protocol processing is to append a header.



SSL Record Format

Header consisting of the following fields:

- **Content Type (8 bits):**The higher-layer protocol used to process the enclosed fragment.
- **Major Version (8 bits):**Indicates major version of SSL in use. For SSLv3, the value is 3.
- **Minor Version (8 bits):**Indicates minor version in use. For SSLv3, the value is 0.
- **Compressed Length (16 bits):**The length in bytes of the plain-text fragment (or compressed fragment if compression is used).

3. Change CipherSpec Protocol:

The Change CipherSpec Protocol is one of the three SSL-specific protocols that use the SSL Record Protocol, and it is the simplest. This protocol consists of a single message, which consists of a single byte with the value 1. The sole purpose of this message is to cause the pending state to be copied into the current state, which updates the Cipher Suite to be used on this connection. This signal is used as a coordination signal. The client must send it to the server and the server must send it to the client. After each side has received it, all of thefollowing messages are sent using the agreed-upon ciphers and keys.

4. Alert Protocol:

The Alert Protocol is used to convey SSL-related alerts to the peer entity. As with other applications that use SSL, alert messages are compressed and encrypted, as specified by the current state.
Each message in this protocol contains 2 bytes.

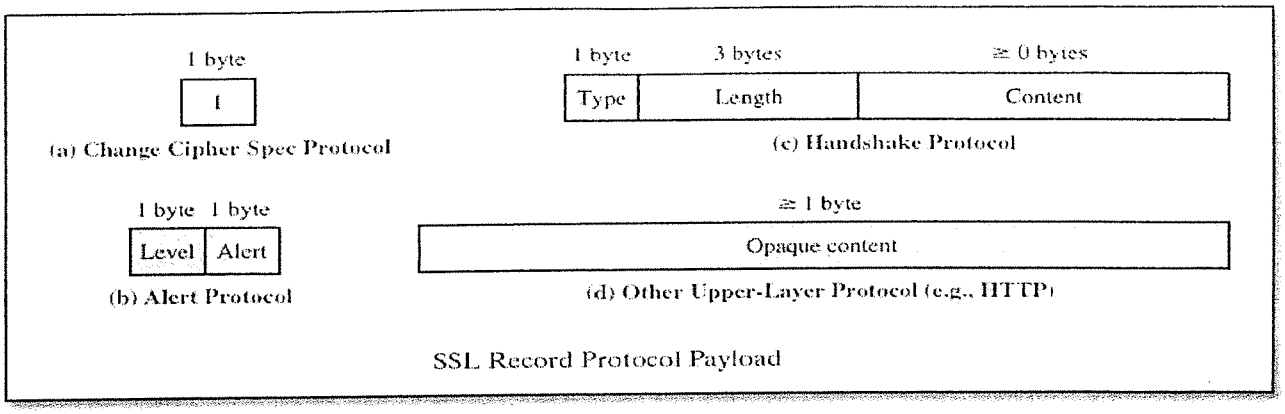
Level is further classified into two parts:

- **Warning:**
  This Alert have no impact on the connection between sender and receiver.

- **Fatal Error:**
  This Alert breaks the connection between sender and receiver.

Alerts defined for SSL

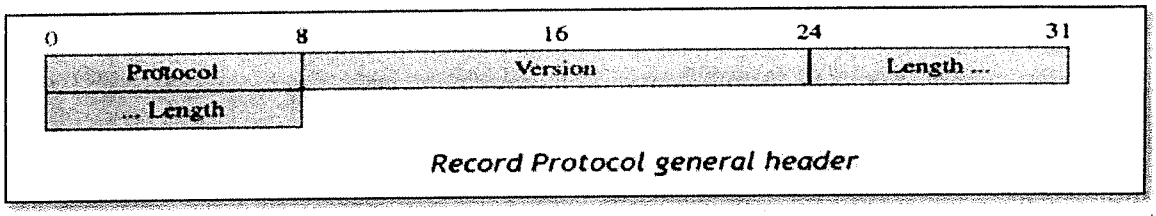
Value	Description	Meaning
0	*CloseNotify*	Sender will not send any more messages.
10	*UnexpectedMessage*	An inappropriate message received.
20	*BadRecordMAC*	An incorrect MAC received.
30	*DecompressionFailure*	Unable to decompress appropriately.
40	*HandshakeFailure*	Sender unable to finalize the handshake.
41	*NoCertificate*	Client has no certificate to send.
42	*BadCertificate*	Received certificate corrupted.
43	*UnsupportedCertificate*	Type of received certificate is not supported.
44	*CertificateRevoked*	Signer has revoked the certificate.
45	*CertificateExpired*	Certificate expired.
46	*CertificateUnknown*	Certificate unknown.
47	*IllegalParameter*	An out-of-range or inconsistent field.



SSL MESSAGE FORMATS

1. SSL Record Protocol:

Record Protocol message encapsulates messages from four different sources at the sender site. At the receiver site, the Record Protocol decapsulates the messages and delivers them to different destinations. The Record Protocol has a general header that is added to each message coming from the sources, as shown below:

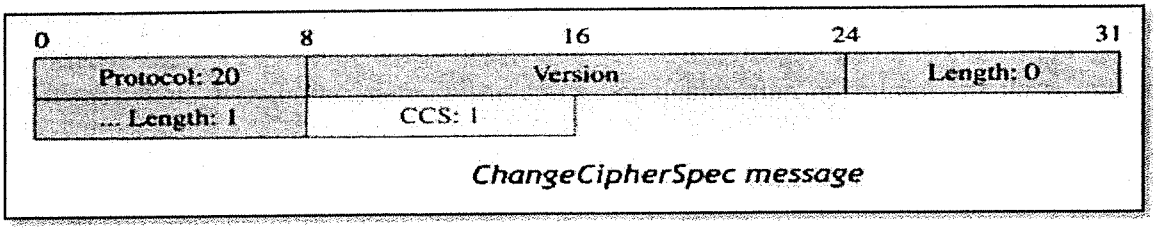


The fields in this header are listed below.

- **Protocol.** This 1-byte field defines the source or destination of the encapsulated message. The values are:
  - ✓ 20 - ChangeCipher Spec Protocol
  - ✓ 21 - Alert Protocol
  - ✓ 22 - Handshake Protocol
  - ✓ 23 - Data from the application layer.
- **Version.** This 2-byte field defines the version of the SSL.
  - ✓ 1- byte is the major version - current version 3.0
  - ✓ 1- byte is the minor version - current version 0
- **Length.** This 2-byte field defines the size of the message (without the header) in bytes.

2. Change Cipher Spec Protocol:

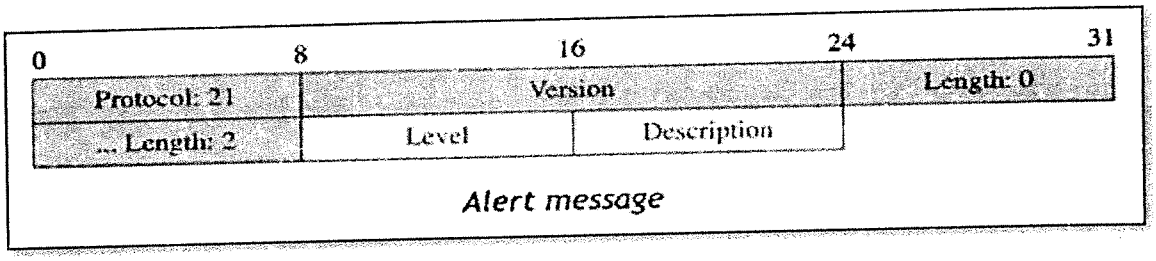
ChangeCipherSpec Protocol has one message and it is only one byte, encapsulated in the Record Protocol message with protocol value 20, as shown in Fig:



The one-byte field in the message is called the CCS and its value is currently 1.

3. Alert Protocol:

Alert Protocolhas one message that reports errors in the process, the value of protocol field is 21.



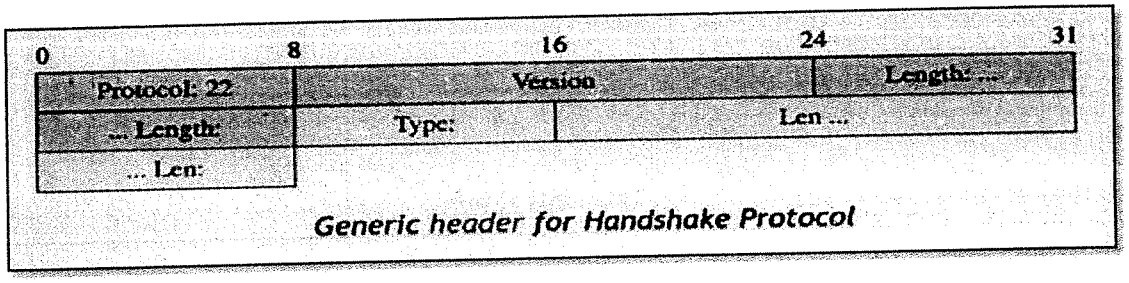
Alert message

The two fields of the Alert message are listed below.

- **Level:** This one-byte field defines the level of the error. i.e., warning and fatal.
- **Description:** The one-byte description defines the type of error.

4. Handshake Protocol:

Handshake Protocol messages have the four-byte generic header,the value of the protocol field is 22.



Generic header for Handshake Protocol

- **Type(1 byte):**Indicates one of 10 messages. (Listed in below table).
- **Length(3 bytes):** The length of the message in bytes.
- **Content(0 bytes):** The parameters associated with this message.

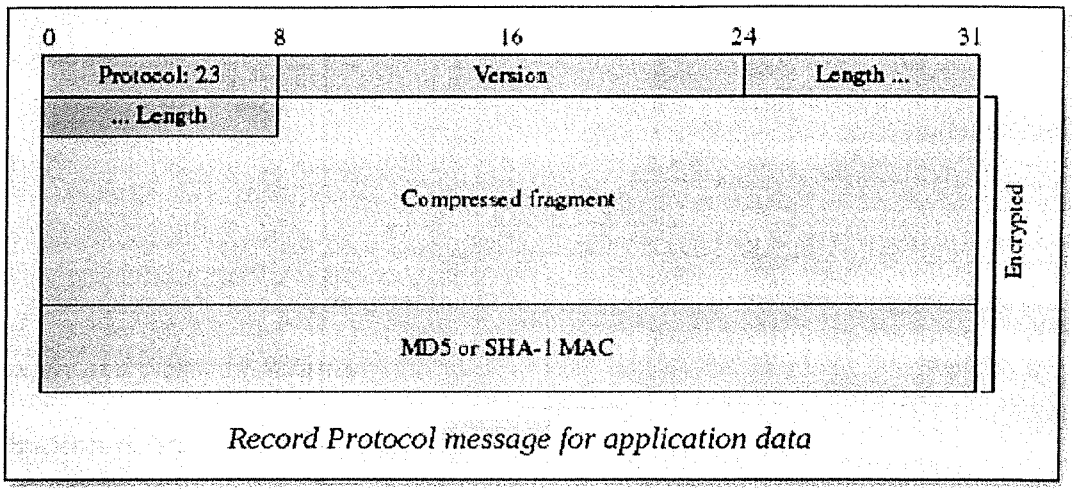
The handshake protocol consists of a series of messages exchanged by client and server. Handshake protocol message types are:

Types of Handshake messages

Type	*Message*
0	HelloRequest
1	ClientHello
2	ServerHello
11	Certificate
12	ServerKeyExchange
13	CertificateRequest
14	ServerHelloDone
15	CertificateVerify
16	ClientKeyExchange
20	Finished

5. Application data:

The format of Record protocol message for application data, the value of the protocol field is 22.



Record Protocol message for application data

The record protocol adds a signature MAC fragment coming from the application layer and then encrypts the fragment and the MAC.

TRANSPORT LAYER SECURITY (TLS)

TLS is a widely adopted security protocol designed to facilitate privacy and data security for communications over the Internet. TLS was proposed by the Internet Engineering Task Force (**IETF**).

- TLS is a protocol for establishing a secure connection between a client and a server. TLS is capable of authenticating both the client and the server and creating an encrypted connection.
- Many protocols use TLS to establish secure connections: (including HTTP, IMAP, POP3, and SMTP)
  - ✓ The TLS Handshake Protocol first negotiates key exchange using an asymmetric algorithm.
  - ✓ The TLS Record Protocol then begins opens an encrypted channel using a symmetric algorithm.
  - ✓ The TLS Record Protocol is also responsible for ensuring that the communications are not altered in transit. Hashing algorithms such as MD5 and SHA are used for this purpose.

- **Version:**Current version is TLS1.0.
- **Cipher Suite:** TLS cipher suite does not support Fortezza algorithm.
- **Cryptography Secret:** There are several differences in the generation of cryptographic secrets. TLS uses a *pseudorandom function (PRF)*to create the master key and the key materials.TLS makes use of a pseudorandom function to expand secrets into blocks of data for purposes of key generation or validation.
- **Message Authentication Code:** TLS uses HMAC (Hashed Message Authentication Code). An HMAC is a MAC which is based on a hash function. The basic idea is to concatenate the key and the message, and hash them together.
- **Alert Protocol:** TLS deletes some alert messages and adds some new ones.

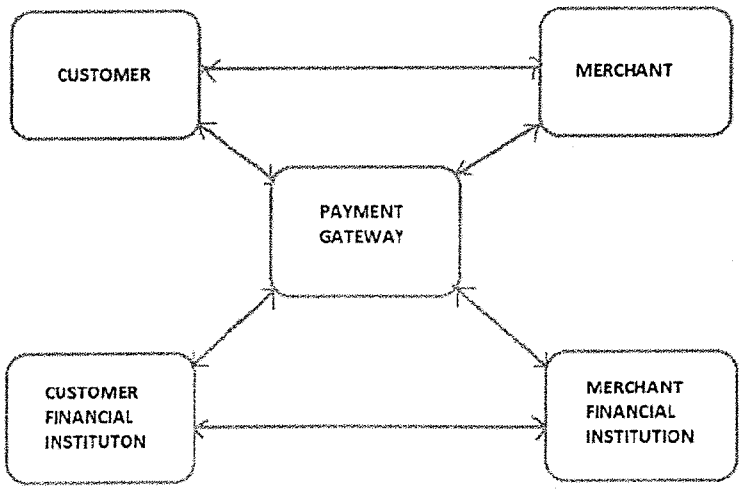
Comparison between SSL and TLS:

S.NO	SSL	TLS
1.	SSL stands for Secure Socket Layer.	TLS stands for Transport Layer Security.
2.	SSL supports Fortezza algorithm.	TLS does not supports Fortezza algorithm.
3.	Current SSL version is 3.0 version.	Current TLS version is the 1.0 version.
4.	In SSLMessage digest is used to create master secret.	In TLSPseudo-random function is used to create master secret.
5.	In SSLMessage Authentication Code protocol is used.	In TLSHashed Message Authentication Code protocol is used.
6.	SSL is complex than TLS	TLS is simple.
7.	The "No certificate" alert message is included.	It eliminates alert description (No certificate) and adds a dozen other values.

SECURE ELECTRONIC TRANSACTION (SET)

Secure Electronic Transaction (SET) is a system for ensuring the security of financial transactions on the Internet. It was supported initially by Mastercard, Visa, Microsoft, Netscape, and others. With SET, a user is given an electronic wallet (digital certificate) and a transaction is conducted and verified using a combination of digital certificates and digital signatures among the purchaser, a merchant, and the purchaser's bank in a way that ensures privacy and confidentiality. SET makes use of Netscape's Secure Sockets Layer (SSL),

Before discussing SET further, let's see a general scenario of electronic transaction, which includes client, payment gateway, client financial institution, merchant and merchant financial institution.



Requirements in SET:

SET protocol has some requirements to meet, some of the important requirements are :

- It has to provide mutual authentication i.e., customer (or cardholder) authentication by confirming if the customer is intended user or not and merchant authentication.
- It has to keep the PI (Payment Information) and OI (Order Information) confidential by appropriate encryptions.
- It has to be resistive against message modifications i.e., no changes should be allowed in the content being transmitted.
- SET also needs to provide interoperability and make use of best security mechanisms.

Participants in SET:

In the general scenario of online transaction, SET includes similar participants:

1. **Cardholder** – customer
2. **Issuer** – customer financial institution
3. **Merchant**
4. **Acquirer** – Merchant financial
5. **Certificate authority** – Authority which follows certain standards and issues certificates(like X.509V3) to all other participants.

SET functionalities:

1. **Provide Authentication**
   a. **Merchant Authentication** – To prevent theft, SET allows customers to check previous relationships between merchant and financial institution. Standard X.509V3 certificates are used for this verification.
   b. **Customer / Cardholder Authentication** – SET checks if use of credit card is done by an authorized user or not using X.509V3 certificates.
2. **Provide Message Confidentiality** : Confidentiality refers to preventing unintended people from reading the message being transferred. SET implements confidentiality by using encryption techniques. Traditionally DES is used for encryption purpose.
3. **Provide Message Integrity** : SET doesn't allow message modification with the help of signatures. Messages are protected against unauthorized modification using RSA digital signatures with SHA-1 and some using HMAC with SHA-1,

Dual Signature:

The dual signature is a concept introduced with SET, which aims at connecting two information pieces meant for two different receivers:

Order Information (OI) for merchant

Payment Information (PI) for bank

You might think sending them separately is an easy and more secure way, but sending them in a connected form resolves any future dispute possible. Here is the generation of dual signature:

Questions asked from Chpater-3 in May-2019

1. Expand MAC.
2. Define Issuer? Write a note on SSL handshake protocol message types.
3. Write a note on SET components.
4. Explain various SSL message formats in detail.
5. Explain SSL architecture with a neat labelled diagram.
6. Explain the various SET transaction types.

Expected Questions from Chapter-3

1. Explain handshake protocol in detail.
2. What are the steps involved in SSL record protocol transmission?
3. Explain CCS and alert protocol in detail.
4. Write a note on TLS.
5. What are the difference between SSL and TLS?
6. What do you mean by SET? Mention the participants of SET.
7. Explain SET functionalities.

CHAPTER – 4

SECURITY AT NETWORK LAYER

Network layer security controls have been used frequently for securing communications, particularly over shared networks such as the Internet because they can provide protection for many applications at once without modifying them.

The popular framework developed for ensuring security at network layer is Internet Protocol Security (IPsec).

IP SECURITY

IP security(**IPSec**) is a collection of protocols design by the Internet Engineering Task Force (IETF) to provide security for the packet at the network level. The network layer in the internet is often referred to as the Internet Protocol or IP layer.

IPsec encompasses three functional areas: Authentication, Confidentiality and Key management.

Definition: IPsec is a protocol suit for securing IP (internet protocol) communications by authenticating and encrypting each IP packet by authenticating and encrypting each packet of communication session.
It added to either current version of the IP (i.e., IPV4 or IPV6), by means of additional header.

Applications of IPSec:

- **Secure branch office connectivity over the Internet:** A company can build a secure virtual private network over the Internet or over a public WAN. Saving costs and network management overhead.
- **Secure remote access over the Internet:** reduces the cost of toll charges for traveling employees and telecommuters.
- **Establishing extranet and intranet connectivity with partners:** secure communication with other organizations, ensuring authentication and confidentiality and providing a key exchange mechanism.
- **Enhancing electronic commerce security:** Web and electronic commerce applications have built-in security protocols, the use of IPsec enhances security.

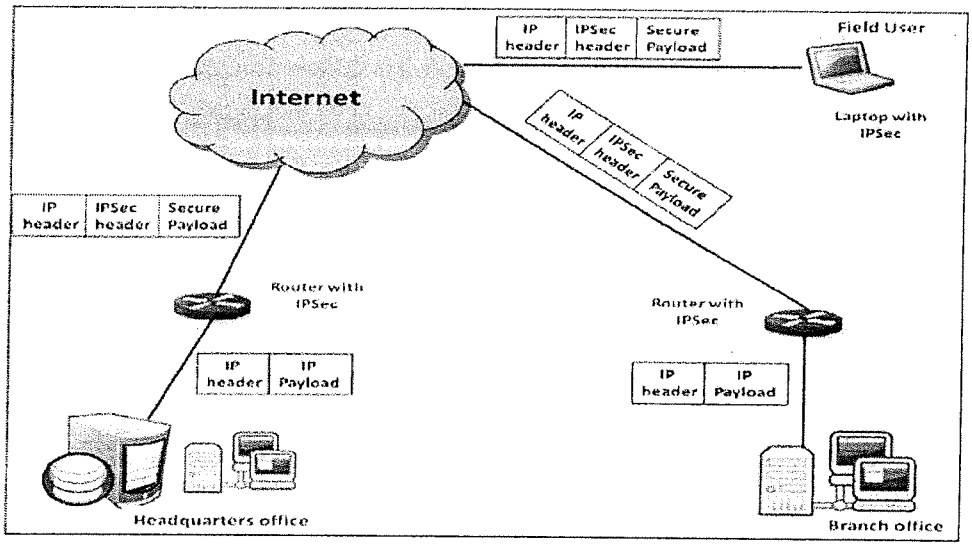
Benefits of IPSec:

- IPSec in a firewall/router provides strong security to all traffic crossing the perimeter.
- IPSec in a firewall is resistant to bypass.
- IPSec is below transport layer(TCP,UDP), hence transparent to applications.
- IPSec can be transparent to end users.
- IPSec can provide security for individual users if needed (useful for offsite workers and setting up a secure virtual subnetwork for sensitive applications).

Note:A Virtual Private Network (**VPN**) extends a private network across a public network, and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network.

Overview of IPSec:

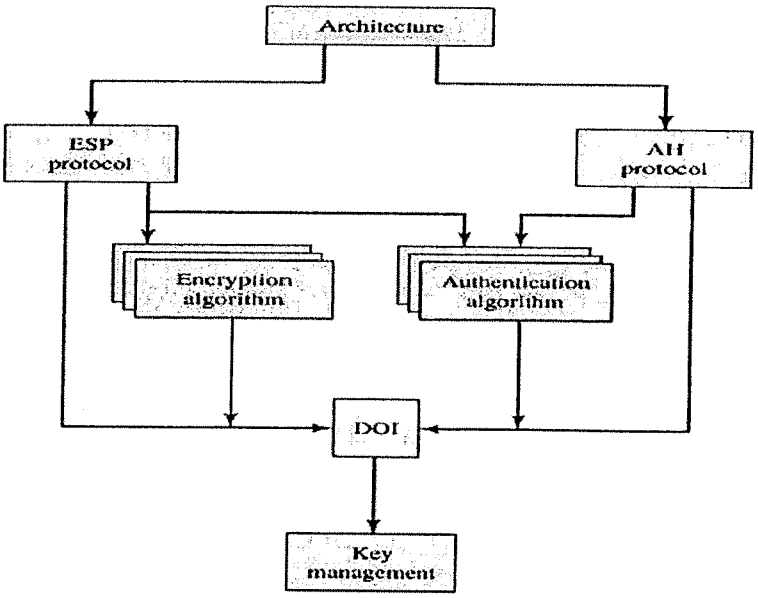
A typical scenario of IPSec usage is shown in the below diagram, an organization maintains LANs at dispersed locations. Non secure IP traffic is conducted on each LAN.



The IPSec protocols operate in networking devices, such as a router or firewall that connect each LAN to the outside world. The IPSec networking device will typically encrypt and compress all traffic going into the WAN, and decrypt and decompress traffic coming from the WAN; these operations are transparent to workstations and servers on the LAN. Secure transmission is also possible with individual users who dial into the WAN. Such user workstations must implement the IPSec protocols to provide security.

IP SECURITY ARCHITECTURE

The IP security architecture (IPsec) provides cryptographic protection for IP datagrams in IPv4 and IPv6 network packets. This protection can include confidentiality, strong integrity of the data, data authentication, and partial sequence integrity. Partial sequence integrity is also known as replay protection.



• **Architecture:** Covers the general concepts, security requirements, definitions, and mechanisms defining IPSec technology.
• **Encapsulating Security Payload (ESP):** Covers the 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 algorithms are used for ESP.
• **Authentication Algorithm:** A set of documents that describe how various authentication algorithms are used for AH and for the authentication option of ESP.
• **Key Management:** Documents that describe key management schemes.
Domain of Interpretation (DOI): Contains values needed for the other documents to relate to each other. These include identifiers for approved encryption and authentication algorithms, as well as operational parameters such as key lifetime.

IPSEC SERVICES

IPSec architecture makes use of two major protocols for providing security at IP level. This facilitates the system to beforehand choose an algorithm to be implemented, security protocols needed and any cryptographic keys required to provide requested services. The IPSec services are as follows:

- **Connectionless Integrity:**Data integrity service is provided by IPSec via AH which prevents the data from being altered during transmission.
- **Data Origin Authentication:** This IPSec service prevents the occurrence of replay attacks, address spoofing etc., which can be fatal
- **Access Control:** The cryptographic keys are distributed and the traffic flow is controlled in both AH and ESP protocols, which is done to accomplish access control over the data transmission.
- **Confidentiality:** Confidentiality on the data packet is obtained by using an encryption technique in which all the data packets are transformed into cipher-text packets which are unreadable and difficult to understand.
- **Limited Traffic Flow Confidentiality:** This facility or service provided by IPSec ensures that the confidentiality is maintained on the number of packets transferred or received. This can be done using padding in ESP.
- **Replay packets Rejection:** The duplicate or replay packets are identified and discarded using the sequence number field in both AH and ESP.

TWO MODES

IPSec operates in one of the two modes:

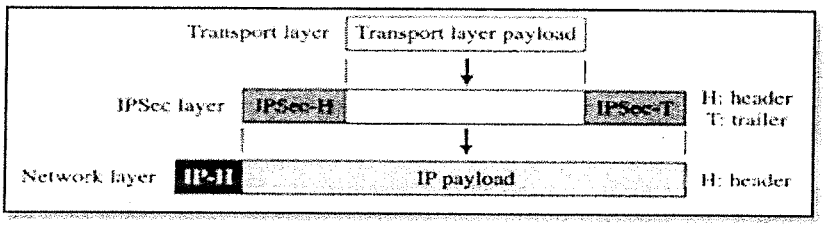
1. Transport Mode
2. Tunnel Mode

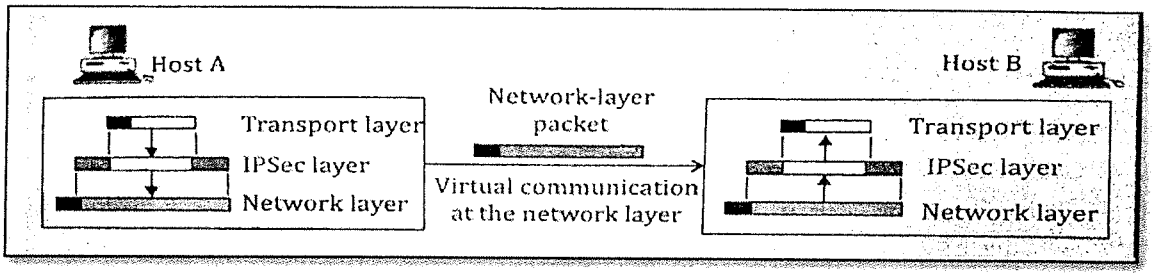
IPsec can be used (both AH packets and ESP packets) in two modes

1. Transport Mode:

TheIPSec header is inserted just after the IP header –this contains the security information, such as SA identifier, encryption and authentication.

- A transport mode provides protection primarily for upper-layer protocols, i.e., TCP, UDP, ICMP packet, operating directly above the IP layer.
- Typically used in end-to-end communication
- IP header not protected.
- It only protects the information coming from the transport layer.

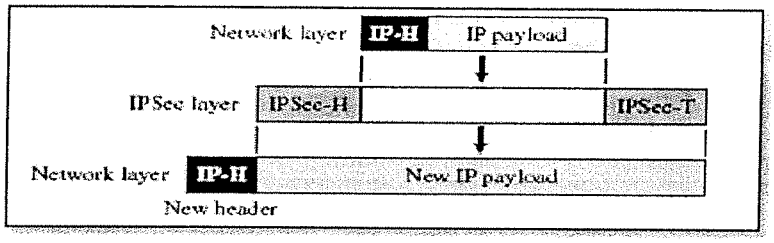


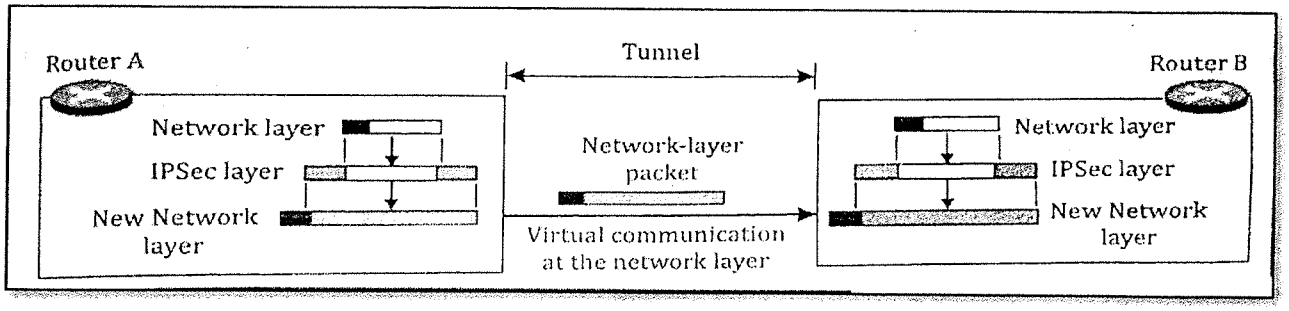


2. Tunnel Mode:

The entire IP packet, header and all, is encapsulated in the body of a new IP packet with a completely new IP header.

- Typically used in firewall-to-firewall communication.
- Provides protection for the whole IP packet.
- No routers along the way are able to examine the inner IP header.
- New larger packet may have totally different source and destination address.





Comparison between Transport and Tunnel mode:

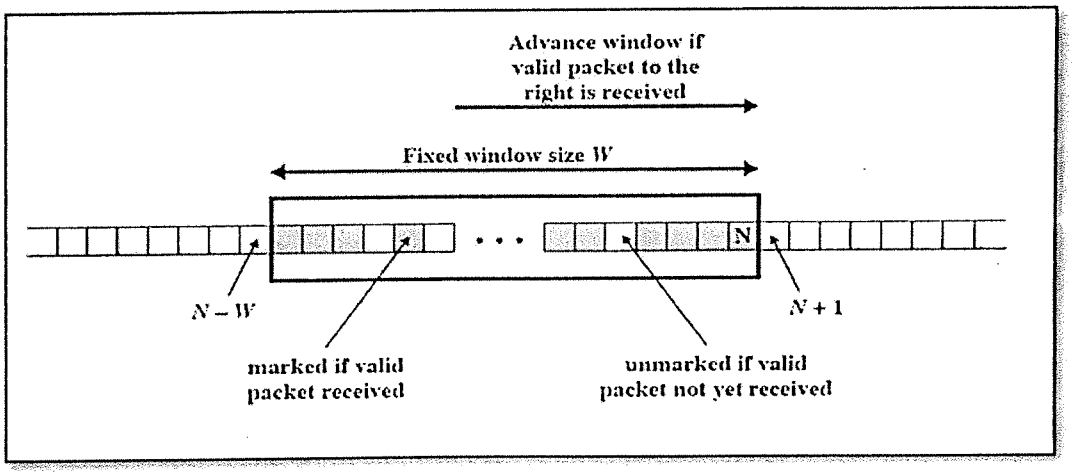
Transport mode	Tunnel mode
IP header not protected. It only protects the transport layer payload.	Provides protection for the whole IP packet.
Lower overhead than tunnel mode.	More overhead required.
No edits on IP header.	The entire packet is hashed or encrypted.
Used in securing communication from one device to another.	Used to tunnel traffic from one site to another
It is good for ESP host-to-host traffic.	It is good for VPNs, gateway-to-gateway security.
Provides protection primarily to upper layer protocols	Provides protection to entire IP packet
Original IP header is used for routing decisions.	New outer IP header is used for routing decisions.
Provides protection for the payload from end-to-end.	Provides protection for the payload and the header from the beginning of the tunnel to end of the tunnel.

Anti-Replay Service:

Anti-replay is a sub-protocol of IPsec that is part of IETF. The main goal of anti-replay is to avoid hackers injecting or making changes in packets that travel from a source to a destination.

The mechanism of Anti-replay window is as follows:

- Replay is when attacker resends a copy of an authenticated packet.
- Use sequence number to prevent this attack.
- Sender initializes sequence number to 0 when a new SA is established.
  - ✓ Increment for each packet.
  - ✓ Must not exceed limit of $2^{32} - 1$.
- Receiver then accepts packets with sequence number within window of $(N - W + 1)$.



Working of Anti-Replay attack is as follows:

- Received packet within window & new, check MAC, if authenticated mark slot
- Packet to the right of window, do check/mark & advance window to new sequencenumber which is the new right edge.
- Packet to the left, or authentication fails, discard packet.

TWO SECURITY PROTOCOLS

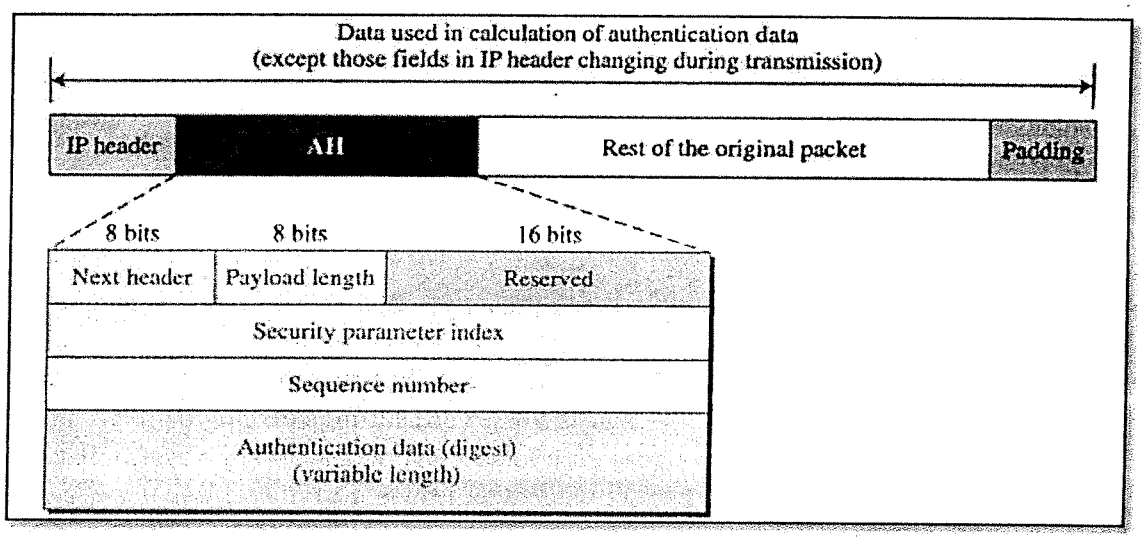
IPSec uses two protocols to provide security, they are:

1. Authentication Header (AH)
2. Encapsulating Security Payload (ESP)

1. Authentication Header:

- The Authentication Header provides support for data integrity and authentication of IP packets.
- The data integrity feature ensures that undetected modification to a packet's content in transit is not possible.
- The authentication feature enables an end system or network device to authenticate the user or application and filter traffic accordingly; it also prevents the address spoofing attacks observed in today's Internet.
- The AH also guards against the replay attack.
- Authentication is based on the use of a message authentication code (MAC), hence the two parties must share a secret key.

The Authentication Header consists of the following fields:

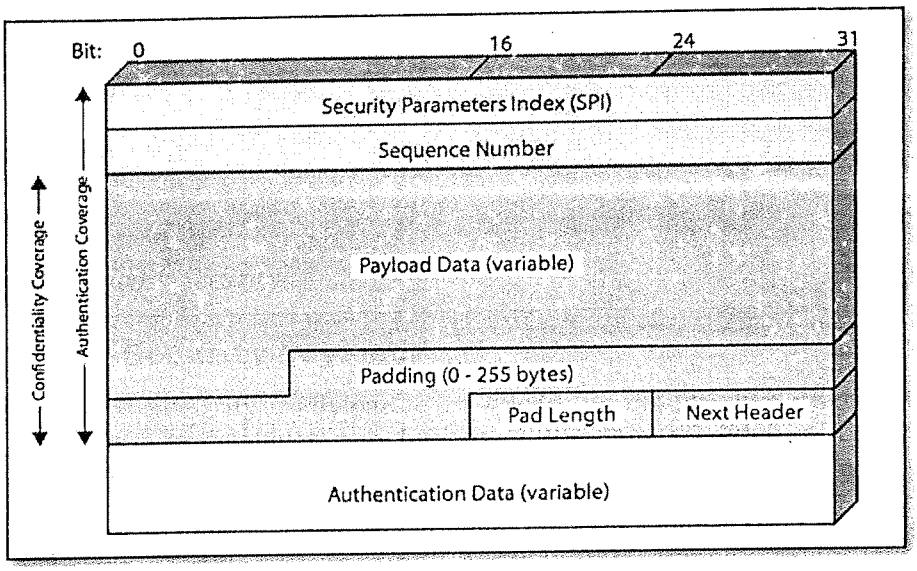


- **Next Header (8 bits):** Identifies the type of header immediately following this header.
- **Payload Length (8 bits):** Length of Authentication Header in 32-bit words.
- **Reserved (16 bits):** For future use.
- **Security Parameters Index (32 bits):** Identifies a security association.
- **Sequence Number (32 bits):** Sequence number prevents the playback and monotonically increasing counter value.
- **Authentication Data (variable):** A variable-length field (must be an integral number of 32-bit words) that contains the Integrity Check Value (ICV), or MAC, for this packet.

2. Encapsulating Security Payload:

The Encapsulating Security Payload provides confidentiality services, including confidentiality of message contents and limited traffic flow confidentiality. As an optional feature, ESP can also provide an authentication service.

The following figure shows the format of an ESP packet. It contains the following fields:



- **Security Parameters Index (32 bits):** Identifies a security association.
- **Sequence Number (32 bits):** A monotonically increasing counter value, this provides an anti-replay function.
- **Payload Data (variable):** This is a transport-level segment (transport mode) or IP packet (tunnel mode) that is protected by encryption.
- **Padding (0-255 bytes):** This field is used to make the length of the plaintext to be a multiple of some desired number of bytes. It is also added to provide confidentiality.
- **Pad Length (8 bits):** Indicates the number of pad bytes immediately preceding this field.
- **Next Header (8 bits):** Identifies the type of data contained in the payload data field by identifying the first header in that payload (for example, an extension header in IPv6, or an upper-layer protocol such as TCP).
- **Authentication Data (variable):** A variable-length field (must be an integral number of 32-bit words) that contains the Integrity Check Value computed over the ESP packet minus the Authentication Data field.

SECURITY ASSOCIATION

Security Association (SA) is a very important aspect of IPSec. A Security Association is a one-way logical connection between a sender and a receiver that affords security services to the traffic carried on it.
SA is a database record which specifies security parameters controlling security operations. They are referenced by the sending host and established by the receiving host.

A security association is uniquely identified by three parameters:

• **Security Parameters Index (SPI):** A bit string assigned to this SA and having local significance only. The SPI is carried in AH and ESP headers to enable the receiving system to select the SA under which a received packet will be processed.

• **IP Destination Address**: Currently, only unicast addresses are allowed; this is the address of the destination endpoint of the SA, which may be an end user system or a network system such as a firewall or router.
• **Security Protocol Identifier**: This indicates whether the association is an AH or ESP security association.

SA Parameters (SA Database):

In each IPSec implementation, there is a nominal Security Association Database that defines the parameters associated with each SA. A security association is normally defined by the following parameters:

- **Sequence Number Counter:** A 32-bit value used to generate the Sequence Number field in AH or ESP headers
- **Sequence Counter Overflow:** A flag indicating whether overflow of the Sequence Number Counter should generate an auditable event and prevent further transmission of packets on this SA (required for all implementations).
- **Anti-Replay Window:** Used to determine whether an inbound AH or ESP packet is a replay.
- **AH Information:** Authentication algorithm, keys, key lifetimes, and related parameters being used with AH (required for AH implementations).
- **ESP Information:** Encryption and authentication algorithm, keys, initialization values, key lifetimes, and related parameters being used with ESP (required for ESP implementations).
- **Lifetime of This Security Association:** A time interval or byte count after which an SA must be replaced with a new SA (and new SPI) or terminated, plus an indication of which of these actions should occur (required for all implementations).
- **IPSec Protocol Mode:** Tunnel, transport, or wildcard (required for all implementations). These modes are discussed later in this section.

SECURITY POLICY

Another important aspect of IPSec is the Security policy(SP),which defines the type of security applied to a packet when it is to be sent or received.

- **Security Policy Database (SA Selectors):**
  Each SPD entry is defined by a set of IP and upper-layer protocol field values, called selectors for each entry in SPD can be accessed using six tuples.

The following are the six tuples which determine an SPD entry:

1. **Remote IP Address**: This may be a single IP address, an enumerated list or range of addresses, or a wildcard (mask) address.
2. **Source IP Address:** This may be a single IP address, an enumerated list or range of addresses, or a wildcard (mask) address.
3. **UserID:** A user identifier from the operating system.
4. **Data sensitivity level:** Used for systems providing information flow security.
5. **Transport layer protocol:** this may be an individual protocol number, a list of protocol numbers, or a range of protocol numbers.
6. **Source and destination ports**: These may be individual TCP or UDP port values, an enumerated list of ports or a wildcard port

- **Outbound SPD:**
  When a packet is to be sent out, the outbound SPD is consulted. Outbound SPD uses the six tuples index.

- **Inbound SPD:**
  When a packet arrives, the inbound SPD is consulted. Each entry in the inbound SPD is also accessed using the six tuples index.

KEY MANAGEMENT

The key management portion of IPSec involves the determination and distribution of secret keys.The IPSec Architecture document mandates support for two types of key management:

- **Manual:**A system administrator manually configures each system with its own keys and with the keys of other communicating systems. This is practical for small, relatively static environments.
- **Automated:** An automated system enables the on-demand creation of keys for SAs and facilitates the use of keys in a large distributed system with an evolving configuration.

The default automated key management protocol for IPSec is referred to as ISAKMP/Oakley and consists of the following elements:

1. **Oakley Key Determination Protocol:** Oakley is a key exchange protocol based on the Diffie-Hellman algorithm but providing added security. Oakley is generic in that it does not dictate specific formats.
2. **Internet Security Association and Key Management Protocol (ISAKMP):** ISAKMP provides a framework for Internet key management and provides the specific protocol support, including formats, for negotiation of security attributes.

1. Oakley Key Determination:

Oakley is a refinement of Diffie -Hellman key exchange algorithm. The Diffie-Hellman algorithm has two attractive features:

- Secret keys are created only when needed. There is no need to store secret keys for a long period of time, exposing them to increased vulnerability.
- The exchange requires no pre-existing infrastructure other than an agreement on the global parameters.

However, Diffie-Hellman has got some weaknesses:

- No identity information about the parties is provided.
- It is possible for a man-in-the-middle attack.
- It is computationally intensive. As a result, it is vulnerable to a clogging attack, in.
- Which an opponent requests a high number of keys.

Oakley is designed to retain the advantages of Diffie-Hellman while countering its weaknesses.

Features of Oakley:

The Oakley algorithm is characterized by five important features:

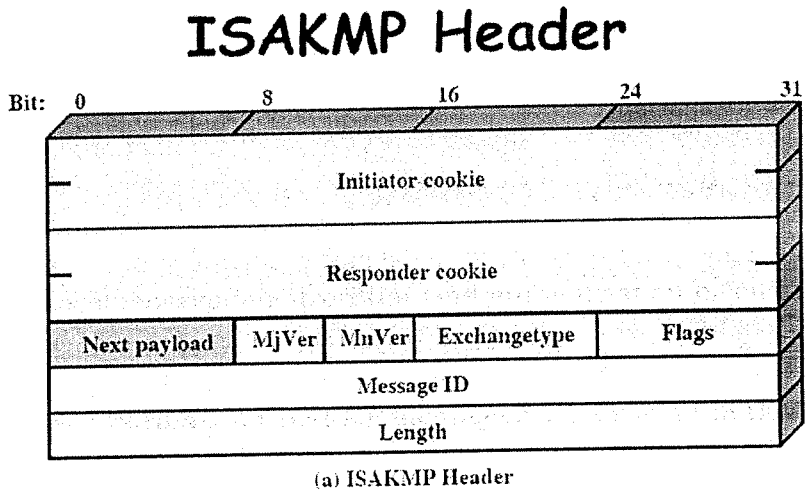
1. It employs a mechanism known as cookies to thwart clogging attacks.
2. It enables the two parties to negotiate a group; this, in essence, specifies the global parameters of the Diffie-Hellman key exchange.
3. It uses nonces to ensure against replay attacks.
4. It enables the exchange of Diffie-Hellman public key values.
5. It authenticates the Diffie-Hellman exchange to thwart man-in-the-middle attacks.

2. Internet Security Association and Key Management Protocol (ISAKMP):

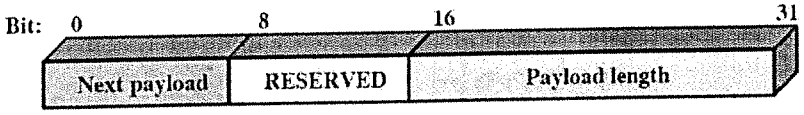
The ISAKMP protocol is designed to carry messages for the IKE exchange.

- It defines procedures and packet formats to establish, negotiate, modify, and delete security associations.
- As part of SA establishment, IKE defines payloads for exchanging key generation and authentication data.
- These payload formats provide a consistent framework independent of the specific key exchange protocol, encryption algorithm, and authentication mechanism.

ISAKMP Header Format: An ISAKMP message consists of an ISAKMP header followed by one or more payloads. All of this is carried in a transport protocol. The specification dictates that implementations must support the use of UDP for the transport protocol.



(a) ISAKMP Header



(b) Generic Payload Header

It consists of the following fields.

- **Initiator SPI (64 bits):** A value chosen by the initiator to identify a unique ISAKMP security association (SA).
- **Responder SPI (64 bits):** A value chosen by the responder to identify a unique ISAKMP SA.
- **Next Payload (8 bits):** Indicates the type of the first payload in the message
- **Major Version (4 bits):** Indicates major version of ISAKMP in use
- **Minor Version (4 bits):** Indicates minor version in use.
- **Exchange Type (8 bits):** Indicates the type of exchange
- **Flags (8 bits):** Indicates specific options set for this ISAKMP exchange.
- **Message ID (32 bits):** Used to control retransmission of lost packets and matching of requests and responses.
- **Length (32 bits):** Length of total message (header plus all payloads) in octets

Questions asked from Chpater-4 in May-2019

1. Define DOI?
2. What do you mean by key management?
3. What is replay attack?
4. What is SA?
5. Mention the three benefits of IPSec.
6. Write a note on IPSec AH?
7. Explain the IPSec architecture and services.

Expected Questions from Chapter-4

1. Define IPSec? Write the applications of IPSec.
2. Briefly explain the overview of IPSec.
3. Explain the IPSec architecture with a neat diagram.
4. Explain the IPSec services.
5. Explain two operating modes of IPSec.
6. Distinguish between two modes of IPSec.
7. Explain the mechanism of anti replay service.
8. Explain the AH header format.
9. Explain the various fields of ESP with a neat diagram.
10. What is SA? Explain SA parameters.
11. Write a note on security policy.

CHAPTER – 5

SYSTEM SECURITY

INTRUDERS

No matter how much secure a system is made, they would constantly try to find their way. We call them intruders, because they try to intrude into the privacy of a network. One of the common threats to security of a system is intruder, also referred to as hacker or cracker.

Intrusion: Any unauthorised access to the system.

Intruders: The person who try to intrude into the privacy of a network.

Intruders are classified into three classes:

- **Masquerader:**User with no authority to use the system but penetrates the security system as a legitimate or legal user.
- **Misfeasor:**
  - ✓ Legitimate users with no permission to access the application.
  - ✓ User misuses the privileges provided to him.
- **Clandestine user:**User tries to steal information such that he does not get trapped, another user get trapped in place of him. It can be internal or external.

Masquerader is mostly an outsider whereas misfeasor is generally an insider. Clandestine usercan be outsider or insider.

Intrusion Techniques:

The objective of the intruder is to gain access to a system or to increase the range of privilegesaccessible on a system. Generally this requires the intruder to acquire information that isprotected. Often this information is in the form of a user password.
The password filecan be protected in one of two ways:

- **One-way function:** The system stores only the value of a function based on the user's password. In practice, the system usually performs a one-way transformation (not reversible) in which the password is used to generate a key for the one way function and in which a fixed-length output is produced.
- **Access control:** Access to the password file is limited to one or a very few accounts

The following techniques are used for learning passwords:

1. Try default passwords used with standard accounts that are shipped with the system. Many administrators do not bother to change these defaults.
2. Exhaustively try all short passwords (those of one to three characters).
3. Try words in the system's online dictionary or a list of likely passwords. Examples of the latter are readily available on hacker bulletin boards.
4. Collect information about users, such as their full names, the names of their spouse and children, pictures in their office, and books in their office that are related to hobbies.
5. Try users' phone numbers, Social Security numbers, and room numbers.
6. Try all legitimate license plate numbers for this state.
7. Use a Trojan horse (described in next unit) to bypass restrictions on access.
8. Tap the line between a remote user and the host system.

INTRUSION DETECTION

An intrusion detection system (IDS) is a device or software application that monitors a network or systems for malicious activity or policy violations.

Intrusion Detection is the process of monitoring for and identifying attempted unauthorized system access or manipulation.

If an intrusion is detected quickly enough, the intruder can be identified and ejected from the system before any damage is done or any data are compromised.

Classification of IDS (Types of IDS):

Intrusion prevention systems can be classified into four different types:

1. **Network-based intrusion prevention system (NIPS)**: monitors the entire network for suspicious traffic by analysing protocol activity.
2. **Wireless intrusion prevention system (WIPS)**: monitor a wireless network for suspicious traffic by analysing wireless networking protocols.
3. **Network behaviour analysis (NBA)**: examines network traffic to identify threats that generate unusual traffic flows, such as distributed denial of service (DDoS) attacks, certain forms of malware and policy violations.
4. **Host-based intrusion prevention system (HIPS)**: an installed software package which monitors a single host for suspicious activity by analysing events occurring within that host.

Intrusion Detection methods:

Different Approaches for intrusion detection are:

1. **Statistical anomaly detection:** Involves the collection of data relating to the behaviour oflegitimate users over a period of time. Then statistical tests are applied to observed behaviour to determine with a high level of confidence whether that behaviour is not legitimate user behaviour.An IDS that looks at network traffic and detects data that is incorrect, not valid, or generally abnormal is called anomaly based detection.
   - **Threshold detection:** This approach involves defining thresholds, independent of user,for the frequency of occurrence of various events.
   - **Profile based:** A profile of the activity of each user is developed and used to detectchanges in the behaviour of individual accounts.

2. **Rule-based detection:** Involves an attempt to define a set of rules that can be used to decidethat a given behaviour is that of an intruder.
   - **Anomaly detection:** Rules are developed to detect deviation from previous usagepatterns.
   - **Penetration identification:** An expert system approach that searches for suspicious behaviour.

3. **Signature detection:** It is also known as knowledge based detection. Compares activity and behaviour to signatures of known attacks. Signature based IDPS is good for organizations concerned with known attacks. Signatures must be updated regularly to be effective.

Advantages:

- Simpler since it uses signatures of known attacks.
- The device can be and running upon installation.
- Each signature is assigned a number so it can be specified what activity is considered an attack.

Disadvantages:

- Signatures need to be updated often to be effective.
- Newer attack signatures may not be in the signature database.
- Attackers can make minor changes to attacks to avoid matching an attack signature.
- Might require extensive disk space for storage of database.

4. **Stateful Protocol Inspection:** Stateful protocol inspection is similar to anomaly based detection, but it can also analyse traffic at the network and transport layer and vender-specific traffic at the application layer, which anomaly-based detection cannot do.

Audit records:

A fundamental tool for intrusion detection is the audit record. Some record of on-going activity by users must be maintained as input to an intrusion detection system. It is used to record information about the action of the use. Basically, two plans are used:

- **Native audit records:** Virtually all multiuser operating systems include accounting software that collects information on user activity.
- **Detection-specific audit records:** A collection facility can be implemented that generates audit records containing only that information required by the intrusion detection system.

A good example of detection-specific audit records reported in literature contains the following fields:

a) **Subject:** Initiators of actions. A subject is typically a terminal user but might also be process acting on behalf of users or groups of users.
b) **Action:** Operation performed by the subject on or with an object; for example, login, read, perform I/O, execute.
c) **Object:** Receptors of actions. Examples include files, programs, messages, records, terminals, printers, and user or program-created structures. When a subject is the recipient of an action, such as electronic mail, then that subject is considered an object.
d) **Exception-Condition:** Denotes which, if any, exception condition is raised on return.
e) **Resource-Usage:** A list of quantitative elements in which each element gives the amount used of some resource (e.g., number of lines printed or displayed, number of records read or written, processor time, I/O units used, session elapsed time).
f) **Time-Stamp:** Unique time-and-date stamp identifying when the action took place.

Distributed Intrusion Detection Systems (DIDS):

Work on intrusion detection systems focused on single system standalone facilities. The typical organization need to defend a distribution collection of hosts supported by a LAN or internetwork, where a more effective defence can be achieved by coordination and cooperation among IDS across the network.

Major issues in design of a DIDS are:

- Dealing with varying audit record formats.
- Integrity and confidentiality of networked data.
- Centralized and decentralized architecture.

Honey Pot:

A Honey Pot is an intrusion detection technique used to study hacker movements and probing to help better system defenses against later attacks usually made up of a virtual machine that sits on a network or single client.

Definition: Honeypot is an intrusion detection technique used to detect, deflect or study hacking attempts in order to gain unauthorized access to information systems. Honeypot is a trap that attracts potential attackers.

PASSWORD MANAGEMENT

Passwords are a set of strings provided by users at the authentication prompts of web accounts. Password management is a set of principles and best practices to be followed by users while storing and managing passwords in an efficient manner to secure passwords as much as they can to prevent unauthorized access.

What are the challenges in password management?

There are many challenges in securing passwords in this digital era. When the number of web services used by individuals is increasing year-over-year on one end, the number of cyber-crimes is also skyrocketing on the other end. Here are a few common threats to protecting our passwords:

- **Login spoofing** - Passwords are illegally collected through a fake login page by cybercriminals.
- **Sniffing attack** - Passwords are stolen using illegal network access and with tools like key loggers.
- **Shoulder surfing attack** - Stealing passwords when someone types them, at times using a micro-camera and gaining access to user data.
- **Brute force attack** - Stealing passwords with the help of automated tools and gaining access to user data.
- **Data breach** - Stealing login credentials and other confidential data directly from the website database.

Traditional methods of password management

- Writing down passwords on sticky notes, post-its, etc.
- Sharing them via spreadsheets, email, telephone, etc.
- Using simple and easy to guess passwords
- Reusing them for all web applications
- Often forgetting passwords and seeking the help of 'Forgot Password' option

While hackers are equipped with advanced tools and attacks, individuals and businesses still rely on traditional methods of password management. This clearly raises the need for the best password management practices to curb security threats.

How to manage passwords

- Use strong and unique passwords for all websites and applications
- Reset passwords at regular intervals
- Configure two-factor authentication for all accounts
- Securely share passwords with friends, family, and colleagues
- Store all enterprise passwords in one place and enforce secure password policies within the business environment
- Periodically review the violations and take necessary actions.

Password selection strategies:

A password is a sequence of characters that allows access to a computer system, service or application.

The password selection strategy helps to eliminate guessable passwords while allowing user to select a memorable password. There are four basic techniques which are in use for selecting the password:

- User Education
- Computer-generated passwords
- Reactive password checking
- Proactive password checking

User Education

The user education strategy tells users the importance of using hard-to-guess passwords and provides guidelines for selecting strong passwords, but it needs their cooperation. The problem is that many users will simply ignore the guidelines.

Computer-generated passwords

This strategy let computer create passwords. If the passwords are quite random in nature, users will not be able to remember them. Even if the password is pronounceable, the user may have difficulty remembering it and so be tempted to write it down even pronounceable not remembered. It has history of poor user acceptance.

Reactive password checking

A reactive password checking strategy is one in which the system periodically runs its own password cracker to find guessable passwords. The system cancels any passwords that are guessed and notifies the user. Drawbacks are that it is resource intensive if the job is done right, and any existing passwords remain vulnerable until the reactive password checker finds them.

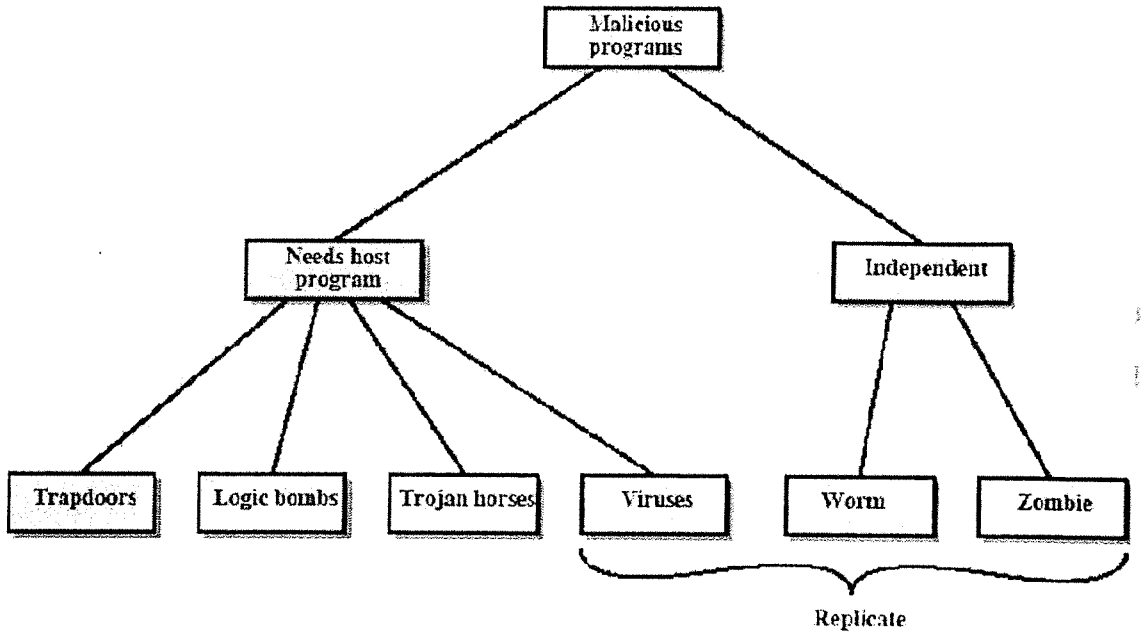
Proactive password checking

The most promising approach to improved password security is a proactive password checker, where a user is allowed to select his or her own password, but the system checks to see if it is allowable and rejects it if not. The trick is to strike a balance between user acceptability and strength.

VIRUSES AND RELATED THREADS

Malicious Programs:

Malicious programs causes harm to a computer system or network.



Malicious software can be divided into two categories:

- **Host program:** The former are essentially fragments of programs that cannot exist independently of some actual application program, utility, or system program.
  E.g.: Viruses, logic bombs, and backdoors.
- **Independent:** The latter are self-contained programs that can be scheduled and run by the operating system.
  E.g.: Worms and zombie.

NAME	DESCRIPTION
Virus	Attaches itself to a program and propagates copies of itself to other programs. It modifies the contents of the program.
Worm	Program that propagates copies of itself to other computers. It will not make any direct damages but consumes lot of system resources.
Logic bomb	Triggers action when condition occurs.
Trojan horse	Program that contains unexpected additional functionality. Simply resides on the system and transmit user information to attackers.
Backdoor (trapdoor)	Program modification that allows unauthorized access to functionality.
Exploits	Code specific to a single vulnerability or set of vulnerabilities.
Downloaders	Program that installs other items on a machine that is under attack. Usually, a downloader is sent in an e-mail.

NAME	DESCRIPTION
Auto-rooter	Malicious hacker tools used to break into new machines remotely.
Kit (virus generator)	Set of tools for generating new viruses automatically.
Spammer programs	Used to send large volumes of unwanted e-mail.
Flooders	Used to attack networked computer systems with a large volume of traffic to carry out a denial of service (DoS) attack.
Keyloggers	Captures keystrokes on a compromised system.
Rootkit	Set of hacker tools used after attacker has broken into a computer system and gained root-level access.
Zombie	Program activated on an infected machine that is activated to launch attacks on other machines.
Spyware	Software that collects information from a computer and transmit it to another system
Adware	Advertising that is integrated into software. It can result in pop-up ads or redirection of a browser to a commercial site.
Email virus	It sends itself to everyone on the mailing list in the user's email package.
Macro Virus	It is platform independent. Virtually it infects Microsoft word documents. It causes a sequence of actions to begin automatically when the application is opened.

Backdoor

A backdoor, also known as a trapdoor, is a secret entry point into a program that allows someone who is aware of the backdoor to gain access without going through the usual security access procedures. Programmers have used backdoors legitimately for many years to debug and test programs such a backdoor is called a maintenance hook. This is usually done when the programmer is developing an application that has an authentication procedure, or a long setup, requiring the user to enter many different values to run the application.

Logic Bomb

One of the oldest types of program threat, predating viruses and worms, is the logic bomb. The logic bomb is code embedded in some legitimate program that is set to "explode" when certain conditions are met. Examples of conditions that can be used as triggers for a logic bomb are the presence or absence of certain files, a particular day of the week or date, or a particular user running the application. Once triggered, a bomb may alter or delete data or entire files, cause a machine halt, or do some other damage.

Trojan Horses

A Trojan horse1 is a useful, or apparently useful, program or command procedure containing hidden code that, when invoked, performs some unwanted or harmful function. Trojan horse programs can be used to accomplish functions indirectly that an unauthorized user could not accomplish directly.
For example, to gain access to the files of another user on a shared system, a user could create a Trojan horse program that, when executed, changes the invoking user's file permissions so that the files are readable by any user. The author could then induce users to run the program by placing it in a common directory and naming it such that it appears to be a useful utility program or application.

Worms

A worm is a program that can replicate itself and send copies from computer to computer across network connections. Upon arrival, the worm may be activated to replicate and propagate again.

Network worm programs use network connections to spread from system to system. Once active within a system, a network worm can behave as a computer virus or bacteria, or it could implant Trojan horse programs or perform any number of disruptive or destructive actions. To replicate itself, a network worm uses some sort of network vehicle. Examples include the following:

Electronic mail facility: A worm mails a copy of itself to other systems.
Remote execution capability: A worm executes a copy of itself on another system.
Remote login capability: A worm logs onto a remote system as a user and then uses commands to copy itself from one system to the other.

The new copy of the worm program is then run on the remote system where, in addition to any functions that it performs at that system, it continues to spread in the same fashion.

Virus Structure:

A virus can be prepended or postpended to an executable program, or it can be embedded insome other fashion. The key to its operation is that the infected program, when invoked, will first execute the virus code and then execute the original code of the program.

An infected program begins with the virus code and works as follows:

The first line of code is a jump to the main virus program. The second line is a special markerthat is used by the virus to determine whether or not a potential victim program has already been infected with this virus.

When the program is invoked, control is immediately transferred to the main virus program. The virus program first seeks out uninfected executable files and infects them. Next, the virus may perform some action, usually detrimental to the system.

This action could be performed every time the program is invoked, or it could be a logic bombthat triggers only under certain conditions.

Finally, the virus transfers control to the original program. If the infection phase of the programis reasonably rapid, a user is unlikely to notice any difference between the execution of aninfected and uninfected program.

A virus such as the one just described is easily detected because an infected version of a programis longer than the corresponding uninfected one. A way to thwart such a simple means ofdetecting a virus is to compress the executable file so that both the infected and uninfectedversions are of identical length. The key lines in this virus are numbered. We assume thatprogram P1 is infected with the virus CV. When this program is invoked, control passes to itsvirus, which performs the following steps:

1. For each uninfected file P2 that is found, the virus first compresses that file to produce P'2,which is shorter than the original program by the size of the virus.
2. A copy of the virus is prepended to the compressed program.
3. The compressed version of the original infected program, P'1, is uncompressed.
4. The uncompressed original program is executed.

Types of Viruses:

There has been a continuous arms race between virus writers and writers of antivirus software since viruses first appeared. As effective countermeasures have been developed for existing types of viruses, new types have been developed. [STEP93] suggests the following categories as being among the most significant types of viruses:

- **Parasitic virus:** The traditional and still most common form of virus. A parasitic virus attaches itself to executable files and replicates, when the infected program is executed, by finding other executable files to infect.
- **Memory-resident virus:** Lodges in main memory as part of a resident system program. From that point on, the virus infects every program that executes.
- **Boot sector virus:** Infects a master boot record or boot record and spreads when a system is booted from the disk containing the virus.
- **Stealth virus:** A form of virus explicitly designed to hide itself from detection by antivirus software.
- **Polymorphic virus:** A virus that mutates with every infection, making detection by the "signature" of the virus impossible.
- **Metamorphic virus:** As with a polymorphic virus, a metamorphic virus mutates with every infection. The difference is that a metamorphic virus rewrites itself completely at each iterations, increasing the difficulty of detection. Metamorphic viruses my change their behaviour as well as their appearance.

Viruses counter measures:

Antivirus Approaches

The ideal solution to the threat of viruses is prevention: Do not allow a virus to get into the system in the first place. This goal is, in general, impossible to achieve, although prevention can reduce the number of successful viral attacks. The next best approach is to be able to do the following:

- **Detection:** Once the infection has occurred, determine that it has occurred and locate the virus.
- **Identification:** Once detection has been achieved, identify the specific virus that has infected a program.
- **Removal:** Once the specific virus has been identified, remove all traces of the virus from the infected program and restore it to its original state.

Advances in virus and antivirus technology go hand in hand. Early viruses were relatively simple code fragments and could be identified and purged with relatively simple antivirus software packages. As the virus arms race has evolved, both viruses and, necessarily, antivirus software have grown more complex and sophisticated.

The four generations of antivirus software are:

- First generation: Simple scanners
- Second generation: Heuristic scanners
- Third generation: Activity traps
- Fourth generation: Full-featured protection.

First-generation scanner requires a virus signature to identify a virus. The virus may contain "wildcards" but has essentially the same structure and bit pattern in all copies. Such signature-specific scanners are limited to the detection of known viruses. Another type of first generation scanner maintains a record of the length of programs and looks for changes in length.

Second-generation scanner does not rely on a specific signature. Rather, the scanner uses heuristic rules to search for probable virus infection. One class of such scanners looks for fragments of code that are often associated with viruses. For example, a scanner may look for the beginning of an encryption loop used in a polymorphic virus and discover the encryption key. Once the key is discovered, the scanner can decrypt the virus to identify it, then remove the infection and return the program to service.

Another second-generation approach is integrity checking. A checksum can be appended to each program. If a virus infects the program without changing the checksum, then an integrity check will catch the change. To counter a virus that is sophisticated enough to change the checksum when it infects a program, an encrypted hash function can be used. The encryption key is stored separately from the program so that the virus cannot generate a new hash code and encrypt that. By using a hash function rather than a simpler checksum, the virus is prevented from adjusting the program to produce the same hash code as before.

Third-generation programs are memory-resident programs that identify a virus by its actions rather than its structure in an infected program. Such programs have the advantage that it is not necessary to develop signatures and heuristics for a wide array of viruses. Rather, it is necessary only to identify the small set of actions that indicate an infection is being attempted and then to intervene.

Fourth-generation products are packages consisting of a variety of antivirus techniques used in conjunction. These include scanning and activity trap components. In addition, such a package includes access control capability, which limits the ability of viruses to penetrate a system and then limits the ability of a virus to update files in order to pass on the infection.

- ***First-generation*** – Scanner uses virus signature to identify virus. or change in length of programs
- ***Second generation*** – Usesheuristic rules to spot viral infection. or uses crypto hash of program to spot changes
- ***Third generation*** – Memory resident programs identify virus by actions
- ***Fourth generation*** – Packages with a variety of antivirus techniques. Eg: scanning & activity traps, access controls.
- Arms race continues

Advanced Antivirus Techniques:

More sophisticated antivirus approaches and products continue to appear. In this subsection, we highlight two of the most important.

Generic Decryption: Generic decryption (GD) technology enables the antivirus program to easily detect even the most complex polymorphic viruses, while maintaining fast scanning speeds. Recall that when a file containing a polymorphic virus is executed, the virus must decrypt itself to activate. In order to detect such a structure, executable files are run through a GD scanner, which contains the following elements:

- CPU emulator: A software-based virtual computer. Instructions in an executable file are interpreted by the emulator rather than executed on the underlying processor. The emulator includes software versions of all registers and other processor hardware, so that the underlying processor is unaffected by programs interpreted on the emulator.
- Virus signature scanner: A module that scans the target code looking for known virus signatures.
- Emulation control module: Controls the execution of the target code.

FIREWALLS

A firewall is a network security device that monitors incoming and outgoing network traffic and decides whether to allow or block specific traffic based on a defined set of security rules.

Firewalls can be an effective means of protecting a local system or network of systems from network-based security threats while at the same time affording access to the outside world via wide area networks and the Internet.

Firewall characteristics

The following are the design goals of a firewall:

- All traffic from inside to outside, and vice versa, must pass through the firewall. This is achieved by physically blocking all access to the local network except via the firewall. Various configurations are possible.
- Only authorized traffic, as defined by the local security policy, will be allowed to pass. Various types of firewalls are used, which implement various types of security policies, as explained later in this section.
- The firewall itself is immune to penetration. This implies that use of a trusted system with a secure operating system.

There are four essential controls exercised by a firewall:

- **Service control:** Determines the types of Internet services that can be accessed, inbound or outbound. The firewall may filter traffic on the basis of IP address and TCP port number; may provide proxy software that receives and interprets each service request before passing it on; or may host the server software itself, such as a Web or mail service.
- **Direction control:** Determines the direction in which particular service requests may be initiated and allowed to flow through the firewall.
- **User control:** Controls access to a service according to which user is attempting to access it. This feature is typically applied to users inside the firewall perimeter (local users). It may also be applied to incoming traffic from external users; the latter requires some form of secure authentication technology, such as is provided in IPSec .
- **Behavior control:** Controls how particular services are used. For example, the firewall may filter e-mail to eliminate spam, or it may enable external access to only a portion of the information on a local Web server.

Firewalls have their limitations, including the following:

- The firewall cannot protect against attacks that bypass the firewall. Internal systems may have dial-out capability to connect than ISP. An internal LAN may support a modem pool that provides dial-in capability for traveling employees and telecommuters.
- The firewall does not protect against internal threats, such as a disgruntled employee or an employee who unwittingly cooperates with an external attacker.
- The firewall cannot protect against the transfer of virus-infected programs or files. Because of the variety of operating systems and applications supported inside the perimeter, it would be impractical and perhaps impossible for the firewall to scan all incoming files, e-mail, and messages for viruses.

Types of firewalls:

There are three common types of firewalls:

1. Packet filters
2. Application-level gateways
3. Circuit-level gateways

1. Packet-Filtering Router:

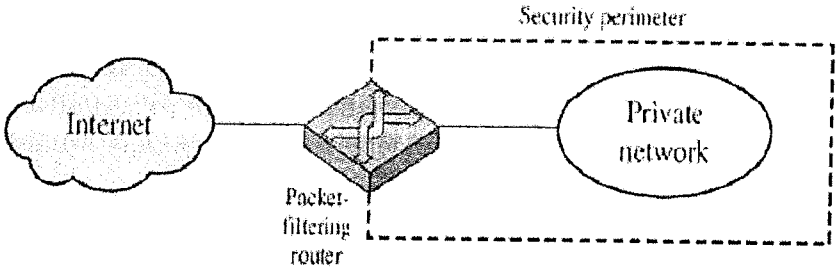
A packet-filtering router applies a set of rules to each incoming and outgoing IP packet and then forwards or discards the packet. The router is typically configured to filter packets going in both directions (from and to the internal network). Filtering rules are based on information contained in a network packet:

- **Source IP address:** The IP address of the system that originated the IP packet. (e.g., 192.178.1.1)
- **Destination IP address:** The IP address of the system the IP packet is trying to reach. (e.g., 192.168.1.2)
- **Source and destination transport-level address:** The transport level (e.g., TCP or UDP) port number, which defines applications such as SNMP or TELNET.
- **IP protocol field:** Defines the transport protocol.
- **Interface:** For a router with three or more ports, which interface of the router the packet came from or which interface of the router the packet is destined for.

The packet filter is typically set up as a list of rules based on matches to fields in the IP or TCP header. If there is a match to one of the rules, that rule is invoked to determine whether to forward or discard the packet. If there is no match to any rule, then a default action is taken. Two default policies are possible:

- **Default** = *discard*: That which is not expressly permitted is prohibited.
- **Default** = *forward*: That which is not expressly prohibited is permitted.

The default discard policy is more conservative. Initially, everything is blocked, and services must be added on a case-by-case basis. This policy is more visible to users, who are more likely to see the firewall as a hindrance. The default forward policy increases ease of use for end users but provides reduced security; the security administrator must, in essence, react to each new security threat as it becomes known.



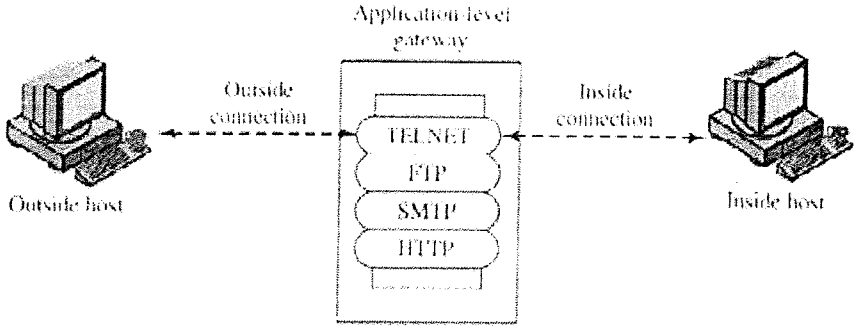
(a) Packet-filtering router

2. Application-Level Gateway:

An application-level gateway, also called a proxy server, acts as a relay of application-level traffic. The user contacts the gateway using a TCP/IP application, such as Telnet or FTP, and the gateway asks the user for the name of the remote host to be accessed. When the user responds and provides a valid user ID and authentication information, the gateway contacts the application on the remote host and relays TCP segments containing the application data between the two endpoints. If the gateway does not implement the proxy code for a specific application, the service is not supported and cannot be forwarded across the firewall. Further, the gateway can be configured to support only specific features of an application that the network administrator considers acceptable while denying all other features.

Application-level gateways tend to be more secure than packet filters. Rather than trying to deal with the numerous possible combinations that are to be allowed and forbidden at the TCP and IP level, the application-level gateway need only scrutinize a few allowable applications. In addition, it is easy to log and audit all incoming traffic at the application level.

A prime disadvantage of this type of gateway is the additional processing overhead on each connection. In effect, there are two spliced connections between the end users, with the gateway at the splice point, and the gateway must examine and forward all traffic in both directions.

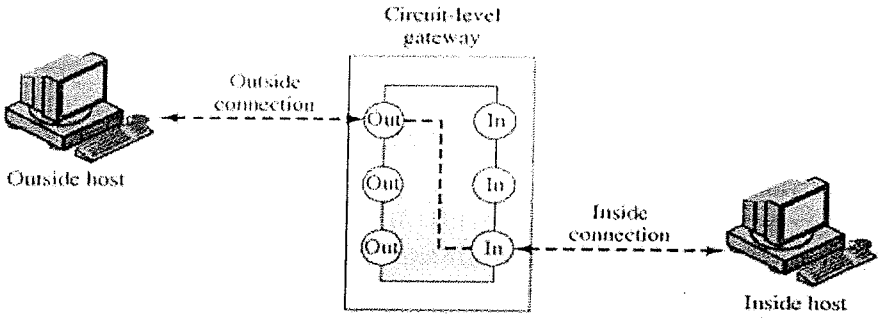


(b) Application-level gateway

3. Circuit-Level Gateway

A third type of firewall is the circuit-level gateway (Figure 16.2). This can be a stand-alone system or it can be a specialized function performed by an application-level gateway for certain applications. A circuit-level gateway does not permit an end-to-end TCP connection; rather, the gateway sets up two TCP connections, one between itself and a TCP user on an inner host and one between itself and a TCP user on an outside host. Once the two connections are established, the gateway typically relays TCP segments from one connection to the other without examining the contents. The security function consists of determining which connections will be allowed.

A typical use of circuit-level gateways is a situation in which the system administrator trusts the internal users. The gateway can be configured to support application-level or proxy service on inbound connections and circuit-level functions for outbound connections. In this configuration, the gateway can incur the processing overhead of examining incoming application data for forbidden functions but does not incur that overhead on outgoing data.



TRUSTED SYSTEMS

One way to enhance the ability of a system to defend against intruders and malicious programs is to implement trusted system technology.

A successful logon would not be sufficient for a system to grant access if it includes sensitive information in its database. A user can be identified to the system by user access control procedure, where each user is associated with a profile that specifies permissible operations and file accesses enabling the operating system to enforce them.

A general model of access control as exercised by an file or database management system is that of an access matrix. The basic elements of the model are as follows:

- **Subject**: An entity (typically a process) capable of accessing objects.
- **Object**: Anything to which access is controlled. Ex: include files, portion of files, programs and segments of memory.
- **Access right:** The way in which the object is accessed by a subject. Ex: read, write and execute.

One axis of an access matrix consists of identified subjects that may attempt data access, the other list objects that may be accessed and each entry in the matrix indicates the access rights of that subject for that object.

	Program1	...	SegmentA	SegmentB
Process1	Read Execute		Read Write	
Process2				Read
•				
•				
•				

a) Access Matrix

In practice, an access matrix is usually sparse and it implemented by decompossition in one of the two ways. If decompossition by columns, you have **access control lists**, which list users &their permitted access rights for each object. If decomposed by rows it yeilds **capability tickets**, which specify authorized objects & operation for a user. These tickets must be unforgeable which is made possible by having the operating system hold all tickets on behalf of users and hold them in a region of memory, inaccessible to users.

Access control list for Program1: Process1 (Read, Execute)
Access control list for Segment A: Process1 (Read, Write)
Access control list for Segment B: Process2 (Read)

b) Access Control List

Capability list for Process1: Program1 (Read, Execute) Segment A (Read)
Capability list for Process2: Segment B (Read)

c) Capability List

The concept of Trusted Systems:

When multiple categories or levels of data are defined, the requirement is referred to as multilevel security. The general statement of the requirement for multilevel security is that a subject at a highlevel may not convey information to a subject at a lower or noncomparable level unless that flowaccurately reflects the will of an authorized user. For implementation purposes, this requirementis in two parts and is simply stated. A multilevel secure system must enforce:

- **No read up:** A subject can only read an object of less or equal security level. This is referred to as **simple security property.**
- **No write down:** A subject can only write into an object of greater or equal security level. This is referred to as ***-property (star property).**

These two rules, if properly enforced, provide multilevel security.

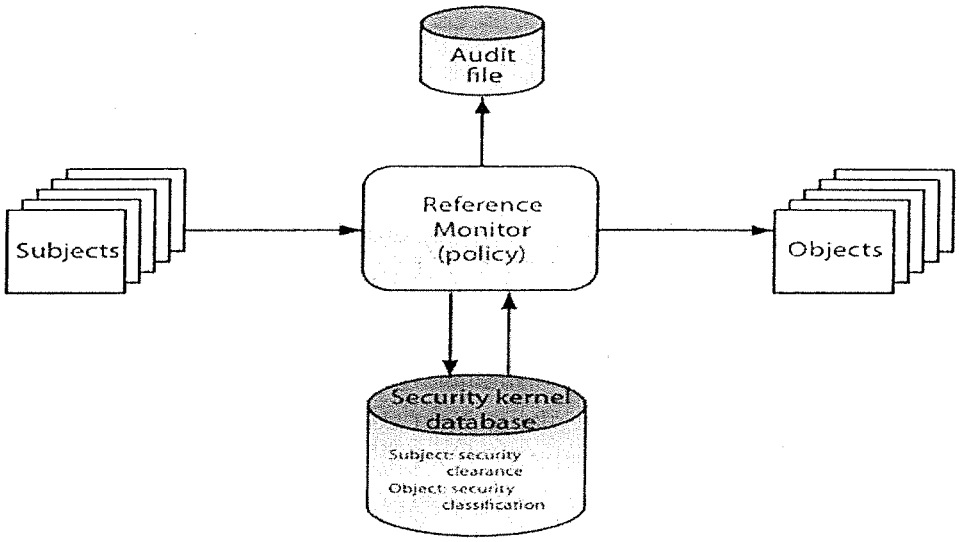
Reference Monitor concept:

The reference monitor concept was introduced as an ideal to achieve controlled sharing. The reference monitor is a controlling element in the hardware and operating system of a computer that regulates the access of subjects to objects on the basis of security parameters of the subject and object.

A combination of hardware, software and firmware that implements the reference monitor concept is called the *Reference validation mechanism.*

The reference monitor enforces the security rules and has the following properties:

- **Complete mediation:** The reference validation mechanism must always be invoked.
- **Isolation:** The reference validation mechanism must be tamperproof.
- **Verifiability:** The reference validation mechanism must be small enough to be subjected to analysis and tests to ensure that it is correct.



Questions asked from Chpater-5 in May-2019

1. Define audit record.
2. What do you mean by honey pots?
3. What is firewall? Explain its various types.

Expected Questions from Chapter-5

1. Explain the different approaches of IDS.
2. What is password management? Explain different password selector strategies.
3. Write a note on.
   a) Viruses. b) Worms. c) Trojan horses d) Logic bomb
4. Explain different characteristics and essential controls of firewall.
5. Write a note on trusted system.
6. Explain virus counter measures in detail.