UNIT-IV

Introduction

So far the delivery of data happens in the following two ways:

Node-to-node delivery: At the data-link level, delivery of frames take place between two nodes connected by a point-to-point link or a LAN, by using the data-link layers address.

Host-to-host delivery: At the network level, delivery of datagrams can take place between two hosts by using IP address.

From user's point of view, the internet can be considered as a set of application programs that use the internet to carry out useful communication tasks. Most popular internet applications include Electronic mail, File transfer, and Remote login.

IP allows transfer of IP datagrams among a number of stations or hosts, where the datagram is routed through the internet based on the IP address of the destination. But, in this case, several application programs (processes) simultaneously running on a source host has to communicate with the corresponding processes running on a remote destination host through the internet. This requires an additional mechanism called *process-to-process delivery*, which is implemented with the help of a transport-level protocol.

The transport level protocol will require an additional address, known as *port number*, to select a particular process among multiple processes running on the destination host. So, there is a requirement of the following third type of delivery system.

Process-to-process delivery: At the transport level, communication can take place between processes or application programs by using port addresses

Port Numbers

Transport layer address is specified with the help a 16-bit Port number in the range of 0 and 65535. Internet Assigned Number Authority (IANA) has divided the addresses in three ranges:

- **Port Number 0 to 1023:** These TCP/UDP port numbers are known well-known ports. These ports are assigned to specific server by the Internet Assigned Numbers Authority (IANA). For example, port 80 is used by web servers.
- **Port Numbers 1024 to 49151:** These are ports that an organization, such as application developers, can register with IANA to be used for a particular service. These should be treated as semi-reserved.
- **Port Numbers 49152 to 65535:** These are port numbers used by client programs, such as a web browser. When you visit a web site, your web browser will assign that session a port number from within this range. As an application developer, you are free to use any of these ports.

What are the most commonly used ports?

When configuring your server settings, you may be asked to enter a specific port number for a particular application or service. Here's a list of commonly requested port numbers you can use

- HTTP – Port 80
- HTTPS – 443
- FTP – 21
- FTPS / SSH – 22
- POP3 – 110
- POP3 SSL – 995
- IMAP – 143
- IMAP SSL – 993
- SMTP – 25 (Alternate: 26)

How to find the port number in Windows?

1. Open **Command Prompt** by typing **"Cmd"** in the search box.

2. Enter the **"ipconfig"** command.

3. Now, type **"netstat -a"** command for a list of connections and port numbers.

Transport Layer Functions

1.Process-Level Addressing-

Addressing is performed at the transport layer, where it is used to differentiate between software programs..

2.Multiplexing and Demultiplexing-

Using the transport layer addresses, transport layer protocols on a sending device multiplex the data received from many application programs for transport, combining them into a single stream of data to be sent. The same protocols receive data and then demultiplex it from the incoming stream of data grams, and direct each package of data to the appropriate recipient application processes.

3.Segmentation, Packaging and Reassembly-

The transport layer segments the large amounts of data it sends over the network into smaller pieces on the source machine, and then reassembles them on the destination machine.

4Connection Establishment, Management and Termination: Transport layer connection-oriented protocols are responsible for the series of communications required to establish a connection, maintain it as data is sent over it, and then terminate the connection when it is no longer required.

Connection Control: It includes 2 types:

- Connectionless Transport Layer : Each segment is considered as an independent packet and delivered to the transport layer at the destination machine.
- Connection Oriented Transport Layer : Before delivering packets, connection is made with transport layer at the destination machine.

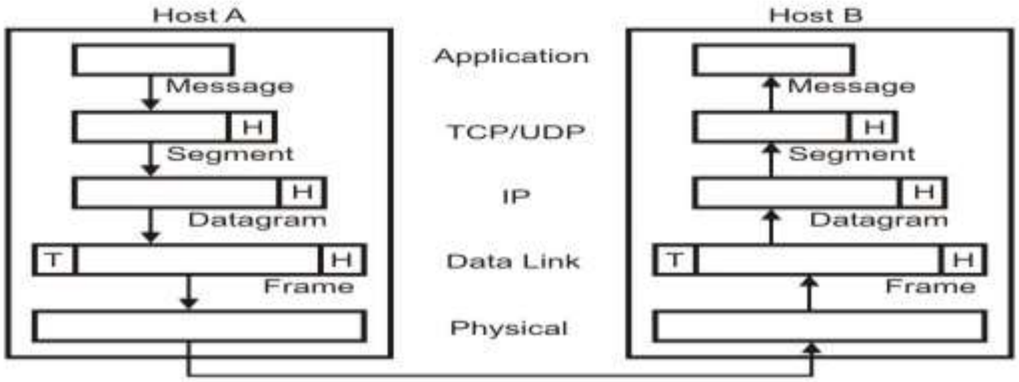
5.Acknowledgments and Retransmissions-

As mentioned above, the transport layer is where many protocols are implemented that guarantee reliable delivery of data. This is done using a variety of techniques, most commonly the combination of acknowledgments and retransmission timers. Each time data is sent a timer is started; if it is received, the recipient sends back an acknowledgment to the transmitter to indicate successful transmission. If no acknowledgment comes back before the timer expires, the data is retransmitted. Other algorithms and techniques are usually required to support this basic process.

6.Flow Control-

Transport layer protocols that offer reliable delivery also often implement flow control features. These features allow devices in communication to not overflow and to avoid bogging down the receiver with data. These allow mismatches in speed between sender and receiver to be detected and dealt with.

Basic communication mechanism is shown in Fig. The additional mechanism needed to facilitate multiple application programs in different stations to communicate with each other simultaneously can be provided by a transport level protocol such as UDP or TCP



Design Issues with Transport Layer

- Accepting data from Session layer, split it into segments and send to the network layer.
- Ensure correct delivery of data with efficiency.
- Isolate upper layers from the technological changes.
- Error control and flow control.

User Datagram protocol (UDP)

Provides an unreliable datagram service to network applications.

The User Datagram Protocol (UDP) does not present data as a stream of bytes, nor does it require that you establish a connection with another program in order to exchange information. Data is exchanged in discrete units called datagrams, which are similar to IP datagrams.

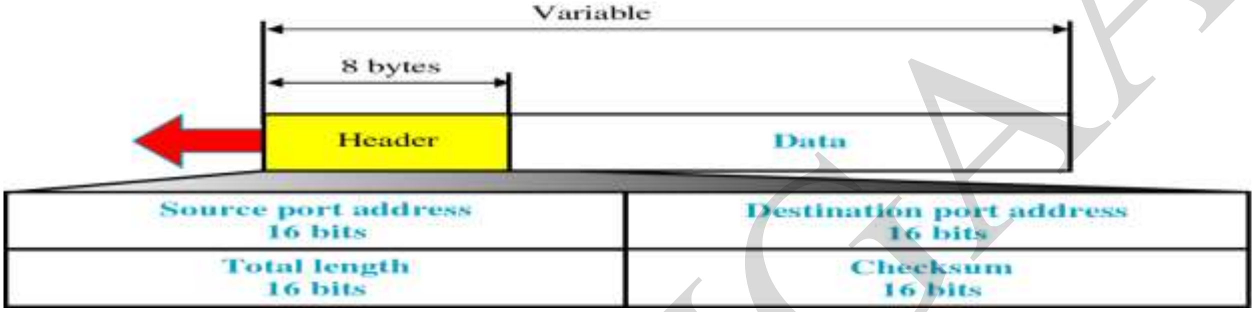
UDP is sometimes referred to as an unreliable protocol because when a program sends a UDP datagram over the network, there is no way for it to know that it actually arrived at its destination. Much of the work that TCP does transparently (such as generating checksums, acknowledging the receipt of packets, retransmitting lost packets and so on) must be performed by the application itself.

UDP Datagram

The UDP datagram format is shown in Fig. A brief description of different fields of the datagram is given below:

Source port (16 bits): It defines the port number of the application program in the host of the sender
Destination port (16 bits): It defines the port number of the application program in the host of the receiver
Length: It provides a count of octets in the UDP datagram, minimum length = 8
Checksum: It is optional, 0 in case it is not in use

UDP Datagram Format



UDP port numbers allow different applications to maintain their own "channels" for data; both UDP and TCP use this mechanism to support multiple applications sending and receiving data concurrently. The sending application (that could be a client or a server) sends UDP datagrams through the source port, and the recipient of the packet accepts this datagram through the destination port.

The datagram size is a simple count of the number of bytes contained in the header and data sections . Because the header length is a fixed size, this field essentially refers to the length of the variable-sized data portion.

The maximum size of a datagram varies depending on the operating environment. With a two-byte size field, the theoretical maximum size is 65535 bytes.

Transmission Control Protocol (TCP)

What is TCP?

TCP was specifically designed to provide a reliable end to end byte stream over an unreliable internetwork. The TCP entity on the sender machine accepts user data streams from local processes, breaks them up into pieces not exceeding 64KB and sends each piece as a separate IP datagram.

The IP layer gives no guarantee that datagram will be delivered properly, so it is up to TCP to timeout and retransmit, if needed. Duplicate, lost and out of sequence packets are handled using the sequence number, acknowledgements, retransmission, timers, etc to provide a reliable service.

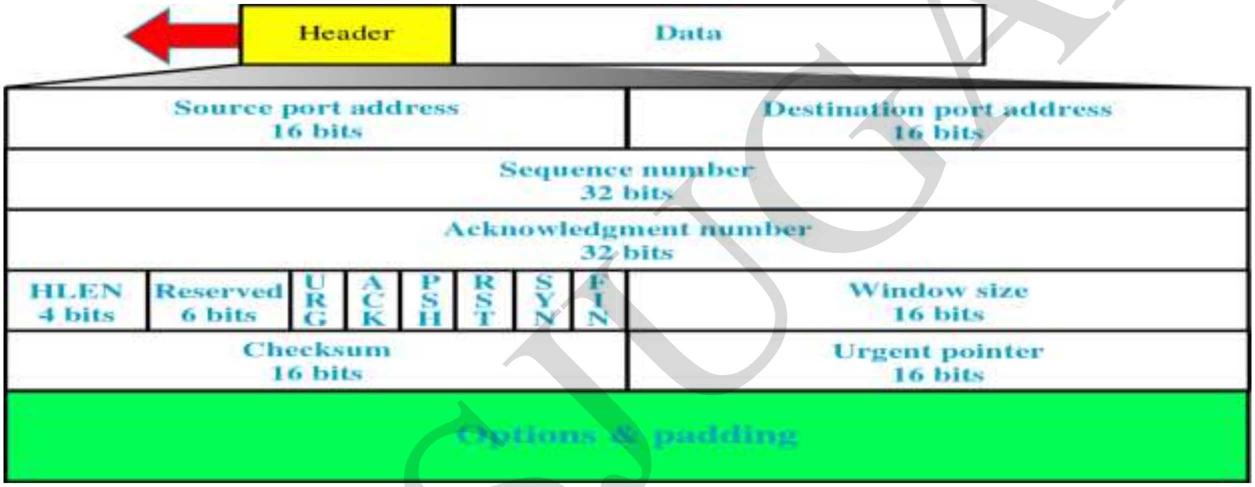
TCP connection is a *duplex connection*. That means there is no difference between two sides once the connection is established.

TCP provides a connection-oriented, full-duplex, reliable, streamed delivery service using IP to transport messages between two processes.
Reliability is ensured by:

- Connection-oriented service
- Flow control using sliding window protocol
- Error detection using checksum
- Error control using go-back-N ARQ technique

View this link for details
https://www.javatpoint.com/computer-network-transport-layer-protocols

TCP Segment Format



TCP Datagram

The TCP datagram format is shown in Fig.

A brief explanation of the functions of different fields are given below:

Source port (16 bits): It defines the port number of the application program in the host of the sender

Destination port (16 bits): It defines the port number of the application program in the host of the receiver

Sequence number (32 bits): It conveys the receiving host which octet in this sequence comprises the first byte in the segment

Acknowledgement number (32 bits): This specifies the sequence number of the next octet that receiver expects to receive

HLEN (4 bits): This field specifies the number of 32-bit words present in the TCP header

Control flag bits (6 bits): URG: Urgent pointer

ACK: Indicates whether acknowledge field is valid
PSH: Push the data without buffering
RST: Resent the connection
SYN: Synchronize sequence numbers during connection establishment
FIN: Terminate the connection

Window (16 bits): Specifies the size of window
Checksum (16 bits): Checksum used for error detection.

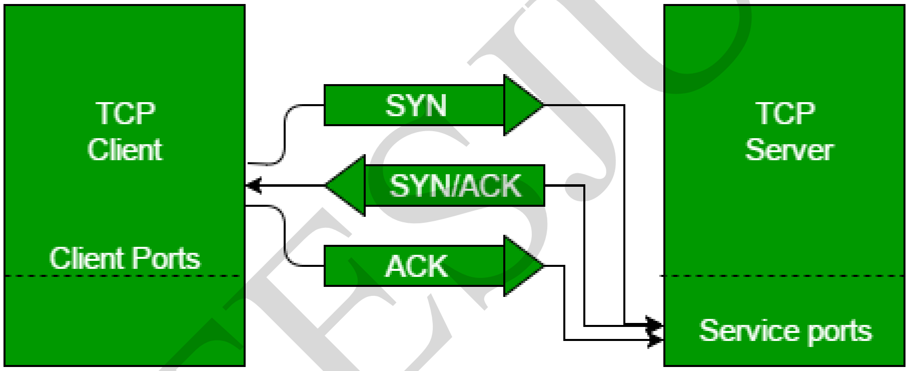
User pointer (16 bits): Used only when URG flag is valid

Options: Optional 40 bytes of information

TCP Connection establishment

https://www.youtube.com/watch?v=fQC4v07gs5k

The **"three-way handshake"** is the procedure used to establish a connection. This procedure normally is initiated by one TCP and responded to by another TCP. The procedure also works if two TCP simultaneously initiate the procedure.



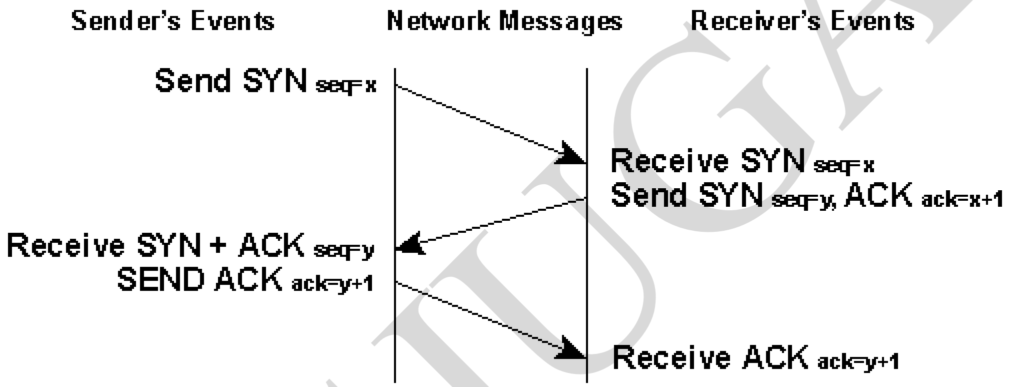
- **Step 1 (SYN) :** In the first step, client wants to establish a connection with server, so it sends a segment with SYN(Synchronize Sequence Number) which informs server that client is likely to start communication and with what sequence number it starts segments with
- **Step 2 (SYN + ACK):** Server responds to the client request with SYN-ACK signal bits set. Acknowledgement(ACK) signifies the response of segment it received and SYN signifies with what sequence number it is likely to start the segments with
- **Step 3 (ACK) :** In the final part client acknowledges the response of server and they both establish a reliable connection with which they will start the actual data transfer

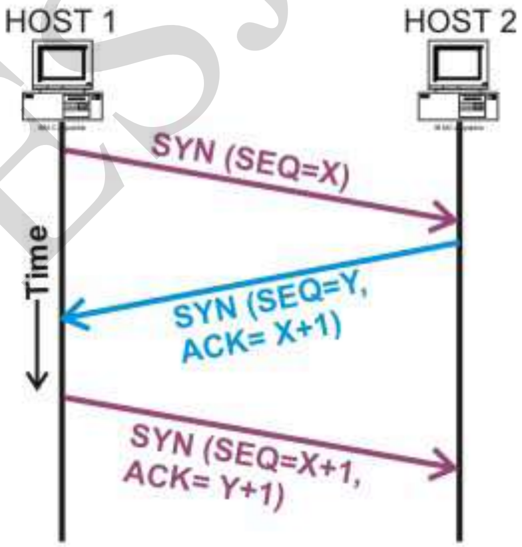
The steps 1, 2 establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps 2, 3 establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

The three-way handshake reduces the possibility of false connections. It is the implementation of a trade-off between memory and messages to provide information for this checking.

The simplest three-way handshake is shown in figure below.

Connection Establish



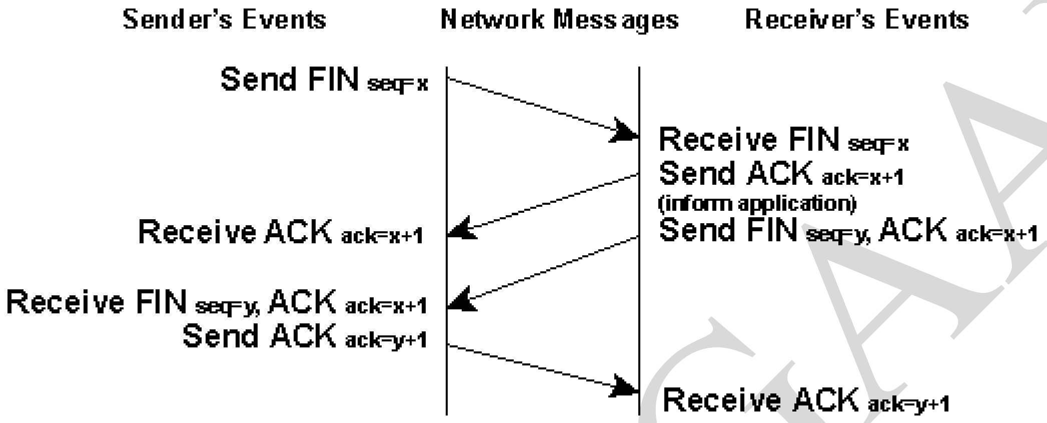


X, Y = Initialization sequence numbers

- The sender sends a SYN packet with sequence number say 'x'.
- The receiver on receiving SYN packet responds with SYN packet with sequence number 'y' and ACK with seq number 'x+1'

- On receiving both SYN and ACK packet, the sender responds with ACK packet with seq number 'y+1'
- The receiver when receives ACK packet, initiates the connection.

Connection Release



- The initiator sends a FIN with the current sequence and acknowledgement number.
- The responder on receiving this informs the application program that it will receive no more data and sends an acknowledgement of the packet. The connection is now closed from one side.
- Now the responder will follow similar steps to close the connection from its side. Once this is done the connection will be fully closed.

Error Control: Error control in TCP includes mechanism for detecting corrupted segments with the help of checksum field. Acknowledgement method is used to confirm the receipt of uncorrupted data. If the acknowledgement is not received before the time-out, it is assumed that the data or the acknowledgement has been corrupted or lost.

Congestion control: To avoid congestion, the sender process uses two strategies known as slow-start and additive increase, and the send one is known as multiplicative decrease as shown in Fig. To start with, the congestion widow size is set to the maximum segment size and for each segment that is acknowledged, the size of the congestion window size is increased by maximum segment size until it reaches one-half of the allowable window size. Ironically, this is known as *slow-start*, although the rate of increase is exponential as shown in the figure. After reaching the threshold, the window size is increased by one segment for each acknowledgement. This continues till there is no time-out. When a time-out occurs, the threshold is set to one-half of the last congestion window size.

Session Layer

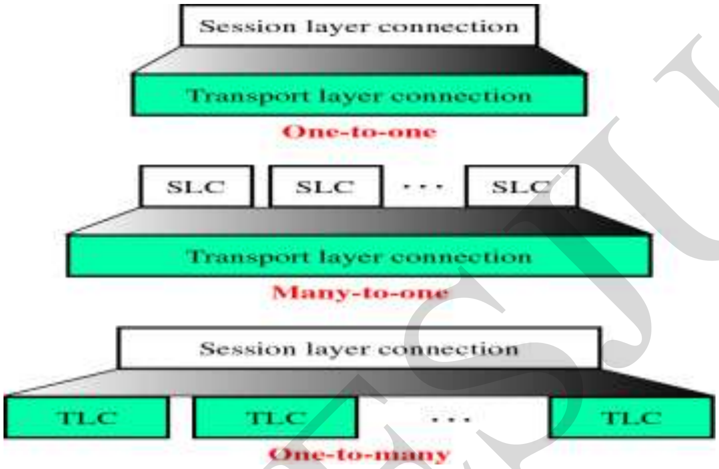
It deals with the concept of **Sessions** i.e. when a user logins to a remote server he should be **authenticated** before getting access to the files and application programs. Session layer can allow traffic to go in both direction at the same time, or in only one direction at one time.

Another job of session layer is to establish and maintain sessions. If during the transfer of data between two machines the session breaks down, it is the session layer which re-establishes the connection.

It also ensures that the data transfer starts from where it breaks keeping it transparent to the end user. e.g. In case of a session with a database server, this layer introduces **check points** at various places so that in case the connection is broken and reestablished, the transition running on the database is not lost even if the user has not committed. This activity is called **Synchronization**.

Another function of this layer is **Dialog Control** which determines whose turn is it to speak in a session. It is useful in video conferencing.

Session-to-Transport Layer Communication

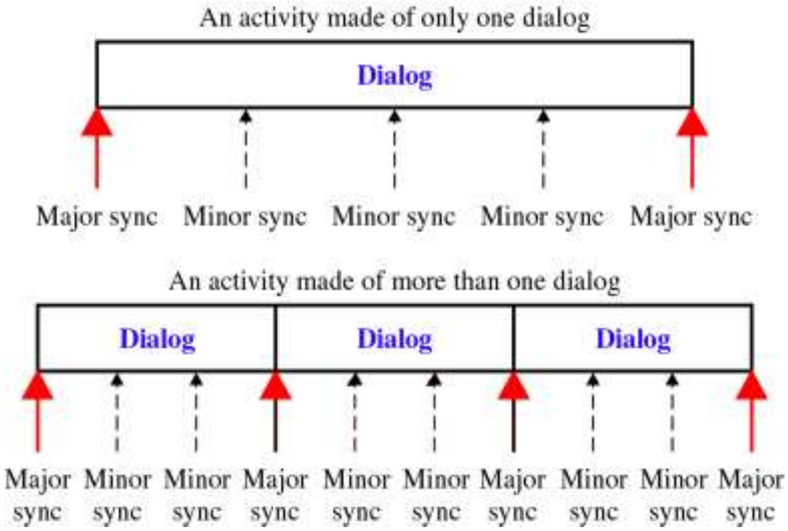


In one to one communication, there is one session layer connection for each transport layer connection. In many to one communication, multiple session layer connections share the services of one transport layer connection. In one to many communication, one session layer needs the services of many transport layer connections.

Synchronization Points

Two types of Synchronization points are used: **major and minor.** Major synchronization points divide an exchange into a series of dialogs. Generally each major sync point must be acknowledged before the session can continue. If error occurs, data can be recovered only upto the last major sync point.

A session can be of one dialog or multiple dialogs separated by major sync points. Minor synchronization points are added in the middle of the dialogs and may or may not require acknowledgement depending on the application. If an error occurs the control can go back to the previous sync points within a dialog to recover data.

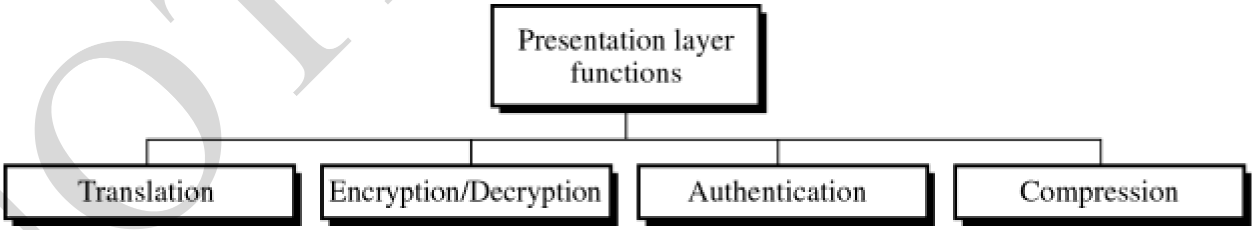


Presentation Layer

This layer is concerned with the syntax and semantics of the information transmitted. In order to make it possible for computers with different data representations to communicate, data structures to be exchanged can be defined in abstract way along with standard encoding. It also manages these abstract data structures. It encodes the data in standard agreed way (network format). Suppose there are two machines A and B which follows different data formats for data representation. This layer ensures that the data transmitted by one gets converted in the form compatible to other machine.

Other functions include compression, encryption etc.

Presentation Layer Functions



Translation: The information may vary from one machine to other machine in respect to the format. If the information sent by one computer in one format (say ASCII) is sent to the other computer and interpreted in other format (say EBCIDIC), the output will be wrong. The presentation layer solves the problem.

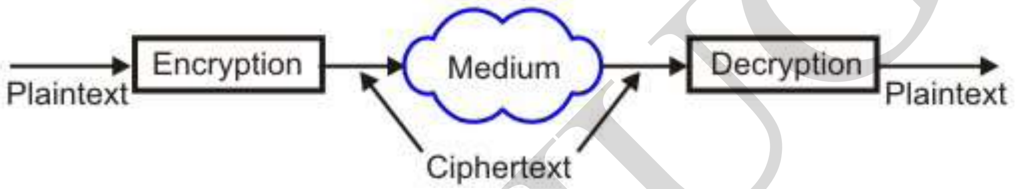
Cryptography: The word **cryptography** has come from a Greek word, which means *secret writing*. In the present day context it refers to the tools and techniques used to make messages secure for communication between the hosts and make messages immune to attacks by hackers.

The role of cryptography can be illustrated with the help of a simple model of cryptography as shown in Figure.

The message to be sent by the sender over a medium is known as **plaintext***,* which is encrypted before sending over the medium.

The encrypted message is known as **ciphertext**, which is received at the other end of the medium and decrypted to get back the original plaintext message.

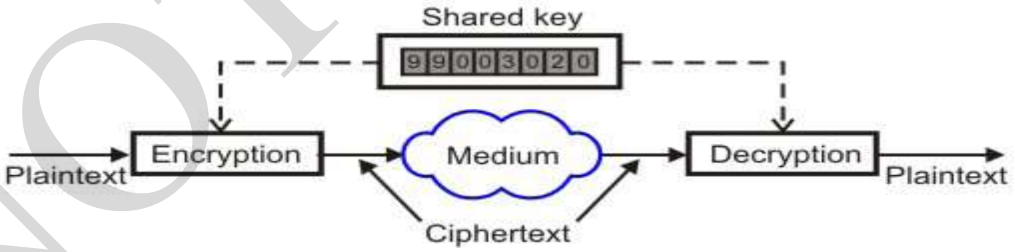
Encryption/Decryption methods are of 2 categories: **Symmetric key cryptography (conventional)** and **Public key cryptography.**



Symmetric Key Cryptography

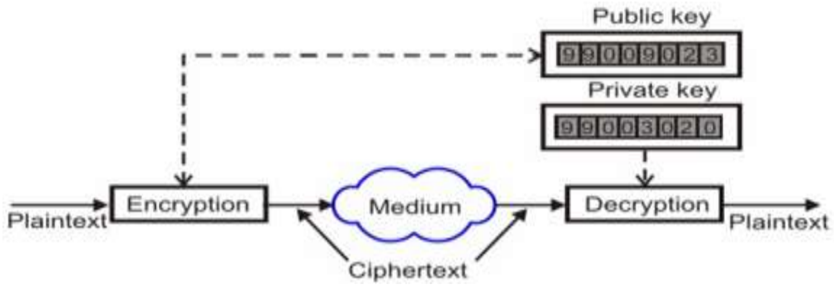
In symmetric key cryptography, same key is shared, i.e. the same key is used in both encryption and decryption. The algorithm used to decrypt is just the inverse of the algorithm used for encryption. For example, if addition and division is used for encryption, multiplication and subtraction are to be used for decryption.

Symmetric key cryptography algorithms are simple requiring lesser execution time. As a consequence, these are commonly used for long messages.



Public key Cryptography

In public key cryptography, there are two keys: a private key and a public key. The public key is announced to the public, where as the private key is kept by the receiver. The sender uses the public key of the receiver for encryption and the receiver uses his private key for decryption as shown in Fig.



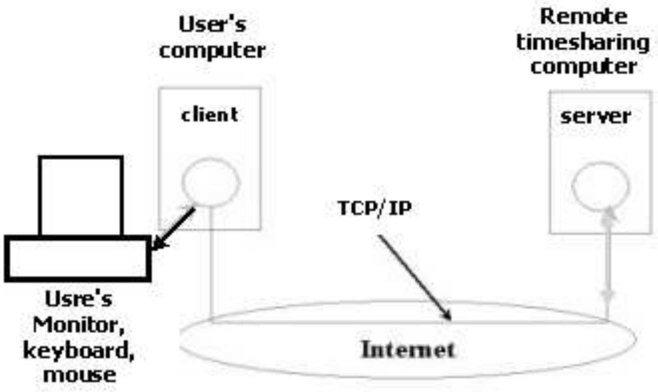
Advantages: The pair of keys can be used with any other entity The number of keys required is small

Application Layer

The seventh layer contains the application protocols with which the user gains access to the network. The choice of which specific protocols and their associated functions are to be used at the application level is up to the individual user.

Services – POP, SMTP (e-mail, Post office protocol, Simple Mail Transfer Protocol), Usenet (for news groups), HTTP (hyper text transfer protocol for web applications), FTP, TFTP (File Transfer protocol, trivial FTP for file transfer), Telnet (Terminal Network, A general purpose program enabling remote login into some other computer and function as if it is directly connected to that remote computer), Domain name server (finding ip addresses for domain names), SNMP (Simple Network Management Protocol).

Telnet is a network protocol which allows you to get connected to other hosts via remote login. Telnet remote login is widely used for file sharing. You can then easily browse the host machine and can download or upload files from/to the host. It works on a standard port no-23.



Once connection has been established between the client and server, the software allows the user to interact directly with the remote computer's operating system. For all user's inputs the server sends output and displays on user's screen.

After a user logs out of the remote computer, the server on the remote computer terminates the Internet connection, informs the client that the session has expired and control of the keyboard, mouse and display returns to the local computer.

The remote access by telnet has significant reasons. It makes computation remote from the user. Instead of sending a data file or a message from one computer to another, remote access allows a program to accept input, process it and send back the result to the remote user. Secondly, once user logs in to the remote computer the user can execute any kind of program residing in the remote server.

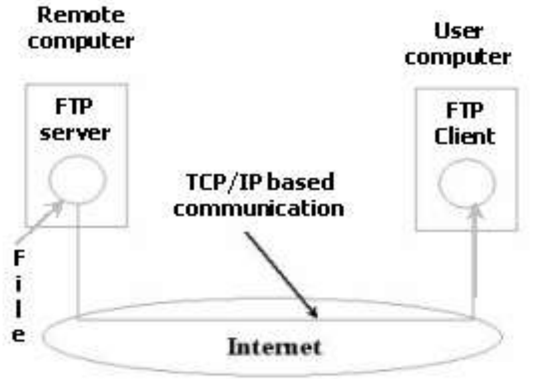
File Transfer Protocol (FTP)

Although services like email, Internet fax can be utilized for sending files over the net they are not designed for handling large volumes of data. For sending large volumes of data reliably over the net **File Transfer Protocol (FTP)** is preferred instead. FTP works in interactive environment. Just type ftp at the DOS command prompt to enter into ftp interactive session. FTP responds to each command the user enters. For example, when a session begins, the user enters a command to identify a remote computer. FTP then establishes a connection to the remote computer. In the same way, to terminate a session user tells FTP to relinquish its connection.

The user must login into the ftp site as an authentic user before performing any ftp based transactions. Usually the user will be provided with login name and password. This way the site is protected from malicious users and keeps the data secure.

FTP operation

FTP operation is also based on client server model.



The user invokes a local FTP program or enters a URL that specifies FTP. The local FTP program or the user's browser becomes an FTP client that uses TCP to contact an FTP server program on the remote computer. Each time the user requests a file transfer, The client and server program interacts to send a copy of the data across the Internet.

The FTP server locates the file that the user requested, and uses TCP to send a copy of the entire contents of the file across the Internet to the client As the client program receives data, it writes the data into a file on the user's local disk. After the file transfer completes, the client and server programs terminate the TCP connection used for the transfer.

BASIS FOR COMPARISON	TELNET	FTP
Basic	It allows a user to log in to the remote server.	It allows a user to transfer a file to the remote machine.
Functions on Port number	23	21 and 20

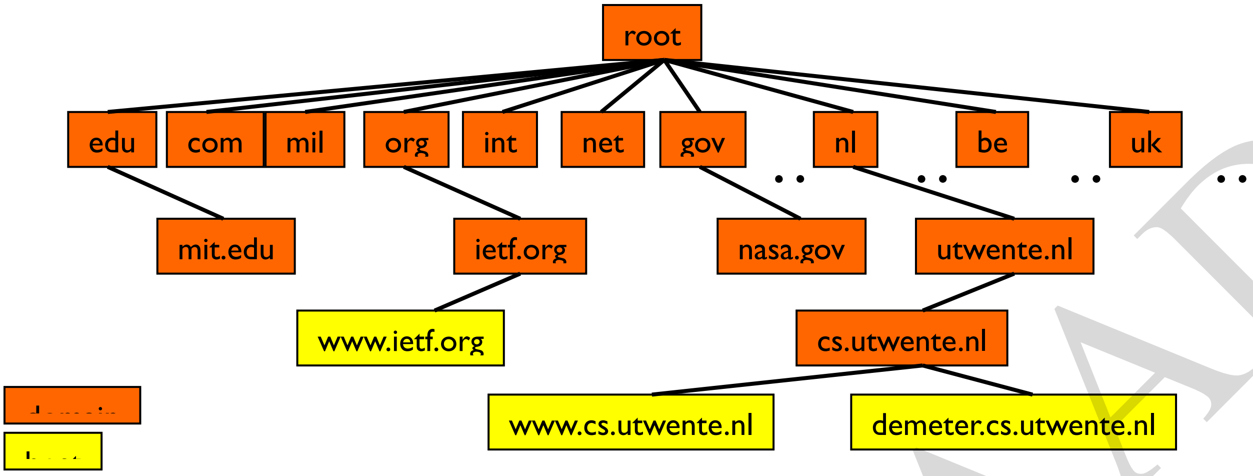
BASIS FOR COMPARISON	TELNET	FTP
Security	May have some security concerns.	More secure than Telnet.
Remote login	Is required to access the system resources.	Not necessarily needed.

DNS (Domain Name Service)

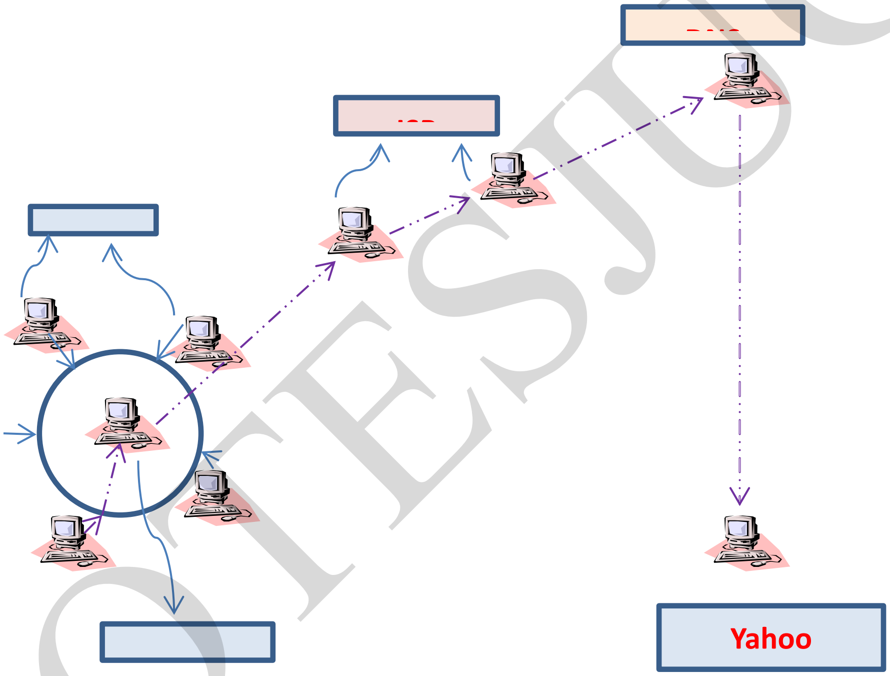
The internet primarily uses IP addresses for locating nodes. However, its humanly not possible for us to keep track of the many important nodes as numbers. Alphabetical names would be more convenient to remember than the numbers.

Conceptually, the internet is divided into several hundred top level domains where each domain covers many hosts. Each domain is partitioned in sub domains which may be further partitioned into sub sub domains and so on... So the domain space is partitioned in a tree like structure as shown below. It should be noted that this tree hierarchy has nothing in common with the IP address hierarchy or organization.

The internet uses a hierarchical tree structure of Domain Name Servers for IP address resolution of a host name.



Web page Retrieval



E-MAIL

- E-mail means or system for transmitting messages electronically (as between computers on a network)
- Messages sent and received electronically through an e-mail system.
- These messages usually consist of individual pieces of text which you can send to another computer user even if the other user is not logged in (i.e. using the computer) at the time you

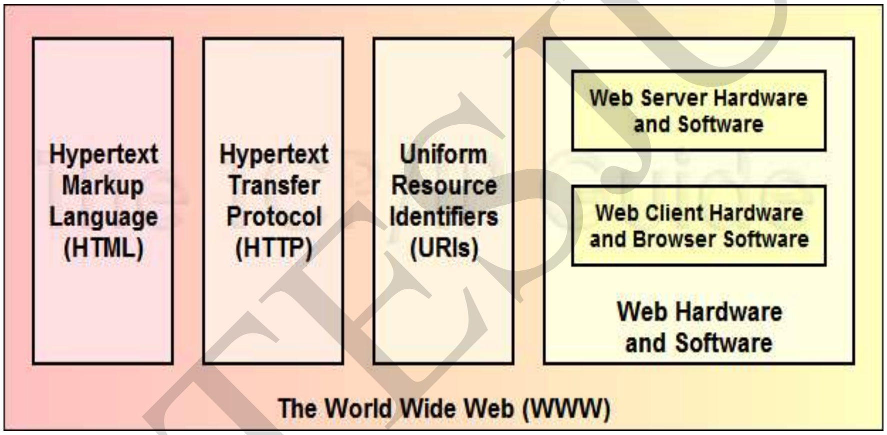
send your message. The message can then be read at a later time. This procedure is analogous to sending and receiving a letter.

- When mail is received on a computer system, it is usually stored in an electronic mailbox for the recipient to read later. Electronic mailboxes are usually special files on a computer which can be accessed using various commands. Each user normally has their individual mailbox.

COMMON EMAIL PROTOCOLS

- ◦ Sending Mail: SMTP (Simple Mail Transport Protocol)
- ◦ Receiving Mail: IMAP (Internet Message Access Protocol), POP3 (Post Office Protocol v3)

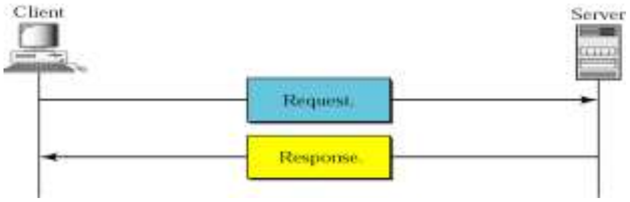
COMPONENTS OF WWW

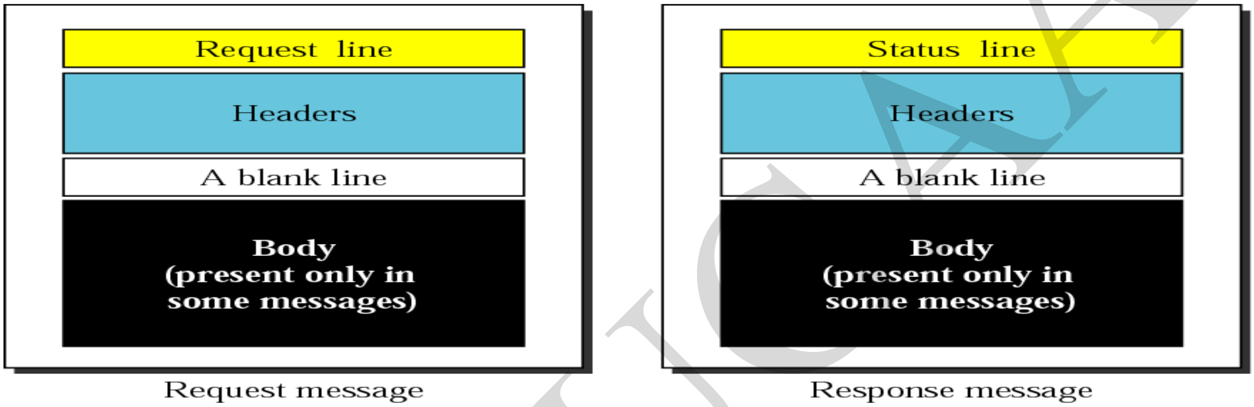

Hypertext Markup Language (HTML)
Hypertext Transfer Protocol (HTTP)
Uniform Resource Identifiers (URIs)
Web Server Hardware and Software
Web Client Hardware and Browser Software
Web Hardware and Software
The World Wide Web (WWW)

HTTP

- HTTP – Hyper text transfer protocol
- It is a software which is used across web to take the Request of a client and pass it to the web server and come back with the Response with some set of rules.
- HTTP is a request/response standard as is typical in client-server computing.
- The client is an application (e.g. web browser) on the computer used by an end-user.
- The server is an application running on the computer hosting the web site.

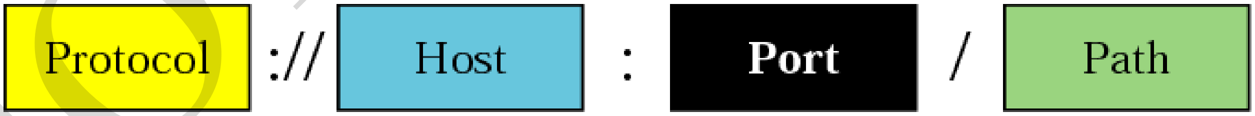
- The client which submits HTTP requests is also referred to as the user agent. The responding server—which stores or creates resources such as HTML files and images—may be called the origin server.





URL

- A common way to get to a Web site is to enter the URL of its home page file in your Web browser's address line. However, any file within that Web site can also be specified with a URL.
- The URL contains the name of the protocol to be used to access the file resource, a domain name that identifies a specific computer on the Internet, and a pathname, a hierarchical description that specifies the location of a file in that computer.
- A URL is a type of URI (Uniform Resource Identifier, formerly called Universal Resource Identifier.)


Protocol
://
Host
:
Port
/
Path