unit-3

1) IPV-4 and IPV-6

IPv4	IPv6
IPv4 has a 32-bit address length	IPv6 has a 128-bit address length
It Supports Manual and DHCP address configuration	It supports Auto and renumbering address configuration
In IPv4 end to end, connection integrity is Unachievable	In IPv6 end to end, connection integrity is Achievable
It can generate 4.29×10^{9} address space	Address space of IPv6 is quite large it can produce 3.4×10^{38} address space
The Security feature is dependent on application	IPSEC is an inbuilt security feature in the IPv6 protocol
Address representation of IPv4 is in decimal	Address Representation of IPv6 is in hexadecimal
Fragmentation performed by Sender and forwarding routers	In IPv6 fragmentation performed only by the sender
In IPv4 Packet flow identification is not available	In IPv6 packet flow identification are Available and uses the flow label field in the header
In IPv4 checksum field is available	In IPv6 checksum field is not available
It has broadcast Message Transmission Scheme	In IPv6 multicast and anycast message transmission scheme is available
In IPv4 Encryption and Authentication facility not provided	In IPv6 Encryption and Authentication are provided
IPv4 has a header of 20-60 bytes.	IPv6 has header of 40 bytes fixed

2)Quality of service in the networking protocol

Quality of service (QoS) refers to any technology that manages data traffic to reduce packet loss, latency and jitter on a network. QoS controls and manages network resources by setting priorities for specific types of data on the network.

QoS parameters

Organizations can measure QoS quantitatively by using several parameters, including the following:

- **Packet loss.** This happens when network links become congested, and routers and switches start dropping packets. When packets are dropped during real-time communication, such as in voice or video calls, these sessions can experience jitter and gaps in speech. Packets can be dropped when a queue, or line of packets waiting to be sent, overflows.
- **Jitter.** This is the result of network congestion, timing drift and route changes. Too much jitter can degrade the quality of voice and video communication.
- **Latency.** This the time it takes a packet to travel from its source to its destination. Latency should be as close to zero as possible. If a voice over IP call has a high amount of latency, users can experience echo and overlapping audio.
- **Bandwidth.** This is the capacity of a network communications link to transmit the maximum amount of data from one point to another in a given amount of time. QoS optimizes the network performance by managing bandwidth and giving high priority applications with stricter performance requirements more resources than others.
- **Mean opinion score (MOS).** This is a metric to rate voice quality that uses a five-point scale, with a five indicating the highest quality.

3)Explain about distance vector routing protocol(AODV) with an example

Ad-Hoc On Demand Distance Vector Routing Protocol(AODV) : Introduction

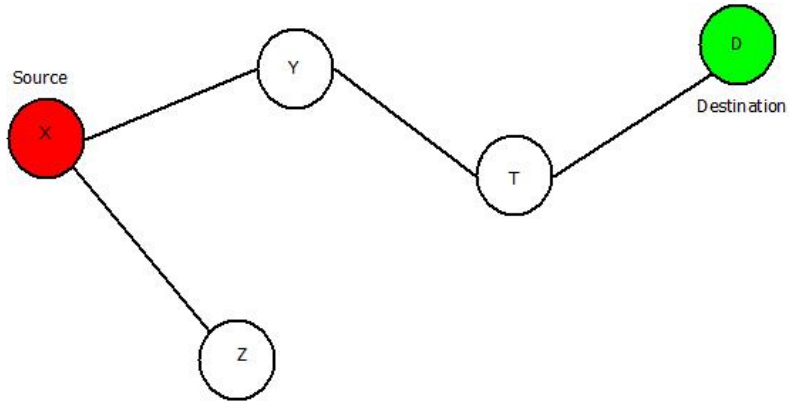
- Another type of reactive routing protocol which does not maintain routes but build the routes as per requirements is Ad-Hoc On Demand Distance Vector Routing Protocol.
- AODV is used to overcome the drawbacks of Dynamic Source Routing Protocol and Distance Vector Routing Protocol i.e. Dynamic Source Routing is capable of maintaining information of the routes between source and destination which makes it slow. If the network is very large containing a number of routes from source to destination, it is difficult for the data packets header to hold whole information of the routes.
- In case of Dynamic Source Routing, multiple routes are present for sending a packet from source to destination but AODV overcomes this disadvantage too.
- In AODV, along with routing tables of every node, two counters including Sequence Number(SEQ NO) and broadcast ID are maintained also.
- The destination IP is already known to which data is to be transferred from source. Thus, the destination **Sequence Number(SEQ NO)** helps to determine an updated path from source to destination.
- Along with these counters, **Route Request(RREQ)** and **Route Response(RRESP)** packets are used in which RREQ is responsible for discovering of route from source to destination and RRESP sends back the route information response to its source.

Ad-Hoc On Demand Distance Vector Routing Protocol : Working

- In Ad-Hoc On Demand Distance Vector Routing, the source node and destination nodes IP addresses are already known.
- **The goal is to identify, discover and maintain the optimal route between source and destination node in order to send/receive data packets and informative.**
- Each node comprises of a routing table along with below mentioned format of Route Request(RREQ) packet.

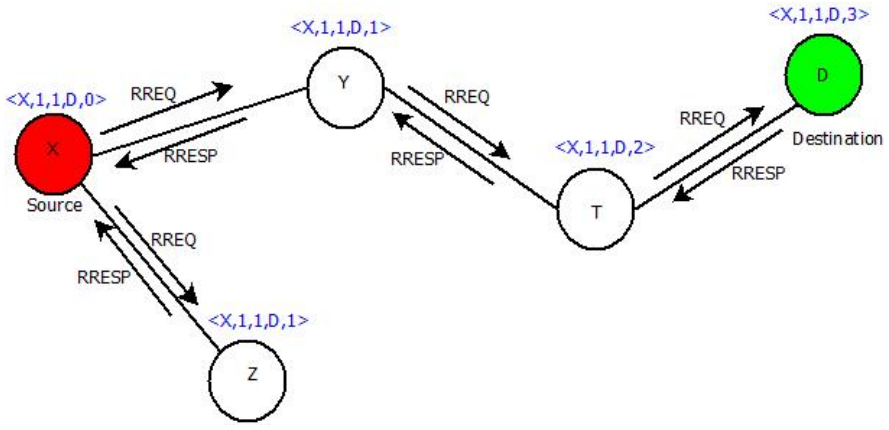
RREQ { Destination IP, Destination Sequence Number, Source IP, Source Sequence Number, Hop Count}

- Consider a network containing 5 nodes that are “X”, “Y”, “Z”,”T”,”D” present at unit distance from each other, where “X” being the source node and “D” being the destination node.



Ad-Hoc On Demand Distance Vector Routing : Sample Network

- **The IP addresses of source node "X" and destination node "D" is already known. Below mentioned steps will let you know how AODV works and concept of Route Request(REREQ) and Route Response(RRESP) is used.**
  - **Step 1: Source node "X" will send Route Request i.e. RREQ packet to its neighbours "Y" and "Z".**
  - **Step 2: Node "Y" & "Z" will check for route and will respond using RRESP packet back to source "X". Here in this case "Z" is the last node but the destination. It will send the RREQ packet to "X" stating "Route Not Found". But node "Y" will send RRESP packet stating "Route Found" and it will further broadcast the RRESP to node "T".**
  - **Step 3: Now the field of net hop in the RREQ format will be updated, Node "T" will send back the "Route Found" message to Node "Y" and will update the next hop field further.**
  - **Step 4: Then Node "T" will broadcast and RREQ packet to Node "D", which is the destination and the next hop field is further updated. Then it will send RRES packet to "T" which will further be sent back to the source node "X" via node "Y" and Node "T" resulting in generation of an optimal path. The updated network would be:**



Ad-Hoc On Demand Distance Vector Routing : Network

Advantages : Ad-Hoc On Demand Distance Vector Routing Protocol

- **Dynamic networks can be handled easily.**
- **No loop generation.**

Disadvantages : Ad-Hoc On Demand Distance Vector Routing Protocol

- **A delayed protocol because of its route discovery process.**
- **High bandwidth requirement.**

unit-4

1)Elements of simple transport protocol

Mid-2 Unit-4

1) Elements of simple Transport Protocols:

→ Addressing: *Transport layer deals with addressing.
* It also differentiates b/w connection & transaction
* ~~It helps in~~ *Ports or sockets address multiple conversations in same location*
* So, this helps in keeping track of multi-message conversations.

→ Connection Establishment/Release:

* Transport layer creates & releases connection across n/w.
* It also enables us to establish & delete conn^n across n/w to multiplex several message streams onto one communication channel.
* This also includes naming mechanism to indicate whom it wishes to communicate.

→ Flow Ctrl & Buffering:

* ~~Flow Ctrl~~. Transport layer enables fast process. to keep pace with slow one.
* Acknowledgements are used to manage end-to-end flow ctrl.
* Buffering is done by sender, if the n/w service is unreliable.
* the sender buffers all the TPDU's (Transaction Protocol Data Unit) sent to the receiver, but buffer size is not same as TPDU size as buffer size is fixed.
* So variable sized buffers are used.

→ Multiplexing / De multiplexing:

* Transport layer establishes seperate n/w connections for each transport connection by session layer.

* To improve throughput, multiple connections are established.

* If throughput isn't imp, it multiplexes all connections, thus reduces cost of n/w con^{n}.

* After multiplexing, at receiving end they are de mulplexed.

→ Crash Recovery / Error Ctrl:

* Error detection & Error ctrl are integral part of reliable service.

* It is performed on end-to-end basis.

* Transport layer uses sequence num for packets to ctrl errors.

2)Explain about TCP protocol

2) TCP Protocol: [Transmission Ctrl Protocol]

→ It is a transport layer that facilitates transmission of packets from source to destination.
→ It is a connection-oriented protocol.
→ This is used with IP. So, it is referred as TCP/IP
→ It divides data taken from application layer into packets & transmits them to destination.
→ The connection remains communication is not completed.
→ lies b/w Appⁿ & N/w layer.
→ Provides host to host communication

Features:

→ Transport Layer Protocol
→ Reliable
→ Order of the data is maintained
→ Connection-oriented
→ Full duplex — data transfer is bidirectional
→ Stream-oriented

Need:

→ In layered architecture, whole task is divided into smaller tasks & it is assigned to particular layer.
→ transport layer has crucial role in end-to-end commⁿ process.
→ It transfers data from upper layer & transfers it to network layer

Working:

→ Connn established using 3-way handshaking
→ Client sends segment with sequence num.
→ Server sends segment + its sequence num & acknowledgment sequence.
→ After receiving it, client sends acknowledgement to server.

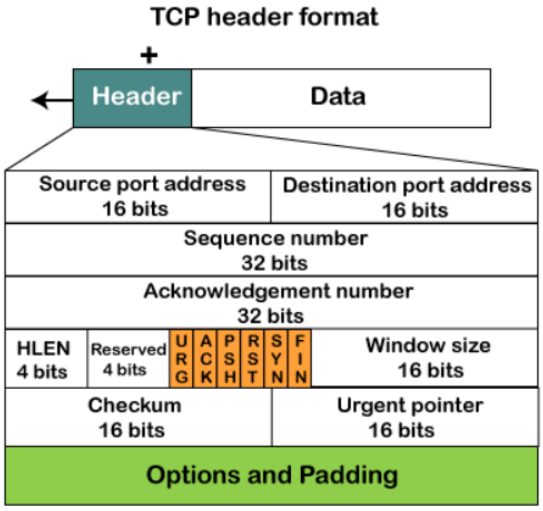
Advantages:

→ Provides connection-oriented reliable connn
→ Provides flow ctrl mechanism using sliding window protocol.
→ Provides error detection (checksum), error ctrl (Go back or ARP)
→ Eliminates congestion by using network congestion avoidance algo.

Dis-Adv:

→ Increases a large amt of overhead as each segment has its own overhead.
→ As it runs several layers, it slows down the speed of n/w
→ TCP is for WAN. Not suitable for small n/w.

TCP Header format



- **Source port:** It defines the port of the application, which is sending the data. So, this field contains the source port address, which is 16 bits.
- **Destination port:** It defines the port of the application on the receiving side. So, this field contains the destination port address, which is 16 bits.
- **Sequence number:** This field contains the sequence number of data bytes in a particular session.
- **Acknowledgment number:** When the ACK flag is set, then this contains the next sequence number of the data byte and works as an acknowledgment for the previous data received. For example, if the receiver receives the segment number 'x', then it responds 'x+1' as an acknowledgment number.
- **HLEN:** It specifies the length of the header indicated by the 4-byte words in the header. The size of the header lies between 20 and 60 bytes. Therefore, the value of this field would lie between 5 and 15.
- **Reserved:** It is a 4-bit field reserved for future use, and by default, all are set to zero.
- **Flags**
  There are six control bits or flags:
  1. **URG:** It represents an urgent pointer. If it is set, then the data is processed urgently.
  2. **ACK:** If the ACK is set to 0, then it means that the data packet does not contain an acknowledgment.
  3. **PSH:** If this field is set, then it requests the receiving device to push the data to the receiving application without buffering it.
  4. **RST:** If it is set, then it requests to restart a connection.
  5. **SYN:** It is used to establish a connection between the hosts.
  6. **FIN:** It is used to release a connection, and no further data exchange will happen.

- **Window size**
  It is a 16-bit field. It contains the size of data that the receiver can accept. This field is used for the flow control between the sender and receiver and also determines the amount of buffer allocated by the receiver for a segment. The value of this field is determined by the receiver.
- **Checksum**
  It is a 16-bit field. This field is optional in UDP, but in the case of TCP/IP, this field is mandatory.
- **Urgent pointer**
  It is a pointer that points to the urgent data byte if the URG flag is set to 1. It defines a value that will be added to the sequence number to get the sequence number of the last urgent byte.
- **Options**
  It provides additional options. The optional field is represented in 32-bits. If this field contains the data less than 32-bit, then padding is required to obtain the remaining bits.

3)Explain about UDP protocol

3) UDP : [User Datagram Protocol]

→ It allows computer applications to send msgs in the form of data grams b/w machines using IP (Internet Protocol) over n/w.

→ UDP is alternative to TCP.

→ UDP works by encapsulating data into packet.

→ UDP is used with IP. So, it is referenced as UDP/IP.

→ Provides process to process communication

→ Connectionless protocol.

→ Provides diff port num to distinguish users. & checksum to know if data has arrived.

Features:

→ Transport Layer Protocol.

→ Connectionless

→ Order of data is not guaranteed.

→ Ports [Defined b/w 0 to 1023]

→ Faster transmission

→ Acknowledgement mechanism

→ Segments are handled independently

→ Stateless [sender doesn't get ack for package sent]

Need:

→ Though UDP is unreliable, it is required in some cases.

→ When packets require large amt of bandwidth with data, UDP is used.

→ Eg: Video streaming

Adv:

→ Provides minimal no. of overheads

Dis Adv:

→ Provides unreliable conn° delivery service

→ Doesn't provide IP services except process to process comm°

→ Doesn't provide reliable transport delivery service

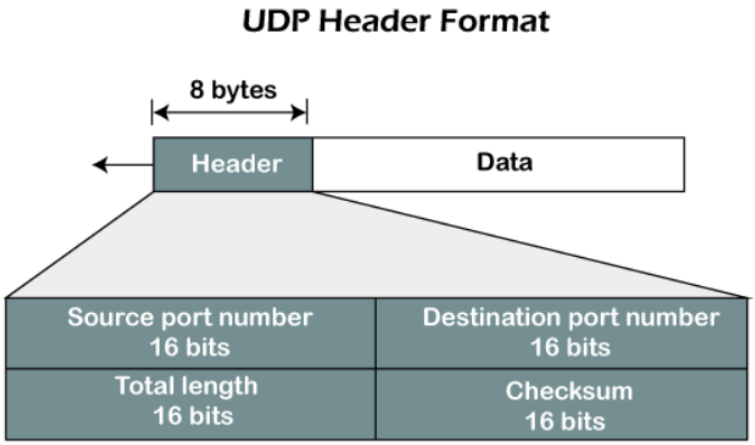
Working:

→ UDP works by gathering data in UDP packing & adding its own header info to the packet.

→ This data also consists of source & destination ports, packet length & checksum

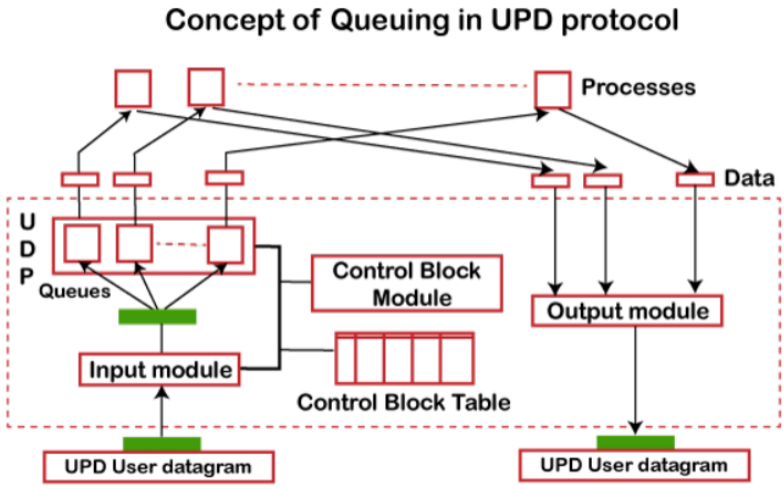
→ UDP packets are encapsulated in IP packets & sent to destinations.

UDP Header Format



In UDP, the header size is 8 bytes, and the packet size is upto 65,535 bytes. But this packet size is not possible as the data needs to be encapsulated in the IP datagram, and an IP packet, the header size can be 20 bytes; therefore, the maximum of UDP would be 65,535 minus 20. The size of the data that the UDP packet can carry would be 65,535 minus 28 as 8 bytes for the header of the UDP packet and 20 bytes for IP header.

Concept of Queuing in UDP protocol



In UDP protocol, numbers are used to distinguish the different processes on a server and client. We know that UDP provides a process to process communication. The client generates the processes that need services while the server generates the processes that provide services. The queues are available for both the processes, i.e., two queues for each process. The first queue is the incoming queue that receives the messages, and the second one is the outgoing queue that sends the messages. The queue functions when the process is running. If the process is terminated then the queue will also get destroyed.

UDP handles the sending and receiving of the UDP packets with the help of the following components:

- **Input queue:** The UDP packets uses a set of queues for each process.

- **Input module:** This module takes the user datagram from the IP, and then it finds the information from the control block table of the same port. If it finds the entry in the control block table with the same port as the user datagram, it enqueues the data.
- **Control Block Module:** It manages the control block table.
- **Control Block Table:** The control block table contains the entry of open ports.
- **Output module:** The output module creates and sends the user datagram.

Several processes want to use the services of UDP. The UDP multiplexes and demultiplexes the processes so that the multiple processes can run on a single host.

4)Diff btw TCP and UDP protocols

4) TCP v/s UDP

TCP	UDP
→ Transmission ctrl Protocol	→ User Data Gram protocol
→ Connection-oriented	→ Connection less.
→ Guarantees delivery of data from source to destination	→ Cannot guarantee
→ Provides flow ctrl & acknowledgment for error checking	→ Check sum is used for error checking
→ Acknowledgement is present	→ Not present
→ Order is data is guaranteed	→ not guaranteed
→ Slower than UDP	→ faster than TCP
→ Retransmission of lost packet is possible	→ Not possible
→ Doesn't support broadcasting	→ Supports
→ TCP conn is byte stream	→ Message stream

unit-5

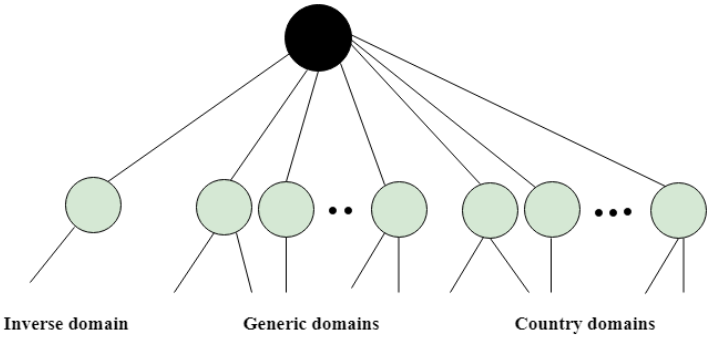
1)Define DNS(Domain Name System)

DNS

An application layer protocol defines how the application processes running on different systems, pass the messages to each other.

- DNS stands for Domain Name System.
- DNS is a directory service that provides a mapping between the name of a host on the network and its numerical address.
- DNS is required for the functioning of the internet.
- Each node in a tree has a domain name, and a full domain name is a sequence of symbols specified by dots.
- DNS is a service that translates the domain name into IP addresses. This allows the users of networks to utilize user-friendly names when looking for other hosts instead of remembering the IP addresses.
- For example, suppose the FTP site at EduSoft had an IP address of 132.147.165.50, most people would reach this site by specifying ftp.EduSoft.com. Therefore, the domain name is more reliable than IP address.

DNS is a TCP/IP protocol used on different platforms. The domain name space is divided into three different sections: generic domains, country domains, and inverse domain.



Generic Domains

- It defines the registered hosts according to their generic behavior.
- Each node in a tree defines the domain name, which is an index to the DNS database.
- It uses three-character labels, and these labels describe the organization type.

Country Domain

The format of country domain is same as a generic domain, but it uses two-character country abbreviations (e.g., us for the United States) in place of three character organizational abbreviations.

Inverse Domain

The inverse domain is used for mapping an address to a name. When the server has received a request from the client, and the server contains the files of only authorized clients. To determine whether the client is on the authorized list or not, it sends a query to the DNS server and ask for mapping an address to the name.

Working of DNS

- DNS is a client/server network communication protocol. DNS clients send requests to the. server while DNS servers send responses to the client.
- Client requests contain a name which is converted into an IP address known as a forward DNS lookups while requests containing an IP address which is converted into a name known as reverse DNS lookups.
- DNS implements a distributed database to store the name of all the hosts available on the internet.
- If a client like a web browser sends a request containing a hostname, then a piece of software such as **DNS resolver** sends a request to the DNS server to obtain the IP address of a hostname. If DNS server does not contain the IP address associated with a hostname, then it forwards the request to another DNS server. If IP address has arrived at the resolver, which in turn completes the request over the internet protocol.

2)Define ELectronic mail(e-mail)

Electronic mail (email) is a digital mechanism for exchanging messages through Internet or intranet communication platforms.

E-mail is defined as the transmission of messages on the Internet. It is one of the most commonly used features over communications networks that may contain text, files, images, or other attachments. Generally, it is information that is stored on a computer sent through a network to a specified individual or group of individuals.

Email messages are conveyed through email servers; it uses multiple protocols within the TCP/IP suite. For example, SMTP is a protocol, stands for simple mail transfer protocol and used to send messages whereas other protocols IMAP or POP are used to retrieve messages from a mail server. If you want to login to your mail account, you just need to enter a valid email address, password, and the mail servers used to send and receive messages.

Although most of the webmail servers automatically configure your mail account, therefore, you only required to enter your email address and password. However, you may need to manually configure each account if you use an email client like Microsoft Outlook or Apple Mail. In addition, to enter the email address and password, you may also need to enter incoming and outgoing mail servers and the correct port numbers for each one.

Email messages include three components, which are as follows:

- **Message envelope:** It depicts the email's electronic format.
- **Message header:** It contains email subject line and sender/recipient information.
- **Message body:** It comprises images, text, and other file attachments.

The email was developed to support rich text with custom formatting, and the original email standard is only capable of supporting plain text messages. In modern times, email supports HTML (Hypertext markup language), which makes it capable of emails to support the same formatting as websites. The email that supports HTML can contain links, images, CSS layouts, and also can send files or "email attachments" along with messages. Most of the mail servers enable users to send several attachments with each message. The attachments were typically limited to one megabyte in the early days of email. Still, nowadays, many mail servers are able to support email attachments of 20 megabytes or more in size.

3)Explain about WWW

World Wide Web (WWW)

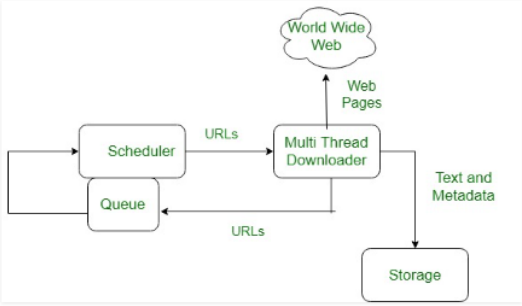
Difficulty Level : Medium • Last Updated : 22 Sep, 2021

The **World Wide Web** abbreviated as WWW and commonly known as the web. The WWW was initiated by CERN (European library for Nuclear Research) in 1989.

History:
It is a project created, by Timothy Berner's Lee in 1989, for researchers to work together effectively at CERN. is an organization, named World Wide Web Consortium (W3C), which was developed for further development in the web. This organization is directed by Tim Berner's Lee, aka the father of the web.

System Architecture:
From the user's point of view, the web consists of a vast, worldwide connection of documents or web pages. Each page may contain links to other pages anywhere in the world. The pages can be retrieved and viewed by using browsers of which internet explorer, Netscape Navigator, Google, Chrome, etc are the popular ones. The browser fetches the page requested interprets the text and formatting commands on it, and displays the page, properly formatted, on the screen.

The basic model of how the web works are shown in the figure below. Here the browser is displaying a web page on the client machine. When the user clicks on a line of text that is linked to a page on the abd.com server, the browser follows the hyperlink by sending a message to the abd.com server asking it for the page.



Here the browser displaying a web page on the client machine when the user clicks on a line of text that is linked to a page on abd.com, the browser follows the hyperlink by sending a message to abd.com server asking for the page.

Working of WWW:
The World Wide Web is based on several different technologies: Web browsers, Hypertext Markup Language (HTML) and Hypertext Transfer Protocol (HTTP).

A Web browser is used to access webpages. Web browsers can be defined as programs which display text, data, pictures, animation and video on the Internet. Hyperlinked resources on the World Wide Web can be accessed using software interface provided by Web browsers. Initially Web browsers were used only for surfing the Web but now they have become more universal. Web browsers can be used for several tasks including conducting searches, mailing, transferring files, and much more. Some of the commonly used browsers are Internet Explorer, Opera Mini, Google Chrome.

Features of WWW:

- HyperText Information System
- Cross-Platform
- Distributed
- Open Standards and Open Source
- Uses Web Browsers to provide a single interface for many services
- Dynamic, Interactive and Evolving.
- "Web 2.0"

Components of Web: There are 3 components of web:

1. **Uniform Resource Locator (URL):** serves as system for resources on web.
2. **HyperText Transfer Protocol (HTTP):** specifies communication of browser and server.
3. **Hyper Text Markup Language (HTML):** defines structure, organisation and content of webpage.

4)Simple Network Management protocol

Simple Network Management Protocol (SNMP)

Difficulty Level : Easy • Last Updated : 03 Nov, 2021

If an organization has 1000 devices then to check all devices, one by one every day, are working properly or not is a hectic task. To ease these up, Simple Network Management Protocol (SNMP) is used.

Simple Network Management Protocol (SNMP) –
SNMP is an application layer protocol that uses UDP port number 161/162.SNMP is used to monitor the network, detect network faults, and sometimes even used to configure remote devices.

SNMP components –
There are 3 components of SNMP:

1. **SNMP Manager –**
   It is a centralized system used to monitor network. It is also known as Network Management Station (NMS)

2. **SNMP agent –**
   It is a software management software module installed on a managed device. Managed devices can be network devices like PC, routers, switches, servers, etc.

3. **Management Information Base –**
   MIB consists of information on resources that are to be managed. This information is organized hierarchically. It consists of objects instances which are essentially variables.

SNMP messages –
Different variables are:

1. **GetRequest –**
   SNMP manager sends this message to request data from the SNMP agent. It is simply used to retrieve data from SNMP agents. In response to this, the SNMP agent responds with the requested value through a response message.

2. **GetNextRequest –**
   This message can be sent to discover what data is available on an SNMP agent. The SNMP manager can request data continuously until no more data is left. In this way, the SNMP manager can take knowledge of all the available data on SNMP agents.

3. **GetBulkRequest –**
   This message is used to retrieve large data at once by the SNMP manager from the SNMP agent. It is introduced in SNMPv2c.

4. **SetRequest –**
   It is used by the SNMP manager to set the value of an object instance on the SNMP agent.

5. **Response –**
   It is a message sent from the agent upon a request from the manager. When sent in response to Get messages, it will contain the data requested. When sent in response to the Set message, it will contain the newly set value as confirmation that the value has been set.

6. **Trap –**
   These are the message sent by the agent without being requested by the manager. It is sent when a fault has occurred.

7. **InformRequest –**
   It was introduced in SNMPv2c, used to identify if the trap message has been received by the manager or not. The agents can be configured to set trap continuously until it receives an Inform message. It is the same as a trap but adds an acknowledgement that the trap doesn't provide.

SNMP security levels –
It defines the type of security algorithm performed on SNMP packets. These are used in only SNMPv3. There are 3 security levels namely:

1. **noAuthNoPriv –**
   This (no authentication, no privacy) security level uses a community string for authentication and no encryption for privacy.
2. **authNopriv –** This security level (authentication, no privacy) uses HMAC with Md5 for authentication and no encryption is used for privacy.
3. **authPriv –** This security level (authentication, privacy) uses HMAC with Md5 or SHA for authentication and encryption uses the DES-56 algorithm.

SNMP versions –
There are 3 versions of SNMP:

1. **SNMPv1 –**
   It uses community strings for authentication and uses UDP only.
2. **SNMPv2c –**
   It uses community strings for authentication. It uses UDP but can be configured to use TCP.
3. **SNMPv3 –**
   It uses Hash-based MAC with MD5 or SHA for authentication and DES-56 for privacy. This version uses TCP. Therefore, the conclusion is the higher the version of SNMP, the more secure it will be.

5)FTP

- FTP stands for File transfer protocol.
- FTP is a standard internet protocol provided by TCP/IP used for transmitting the files from one host to another.
- It is mainly used for transferring the web page files from their creator to the computer that acts as a server for other computers on the internet.
- It is also used for downloading the files to computer from other servers.

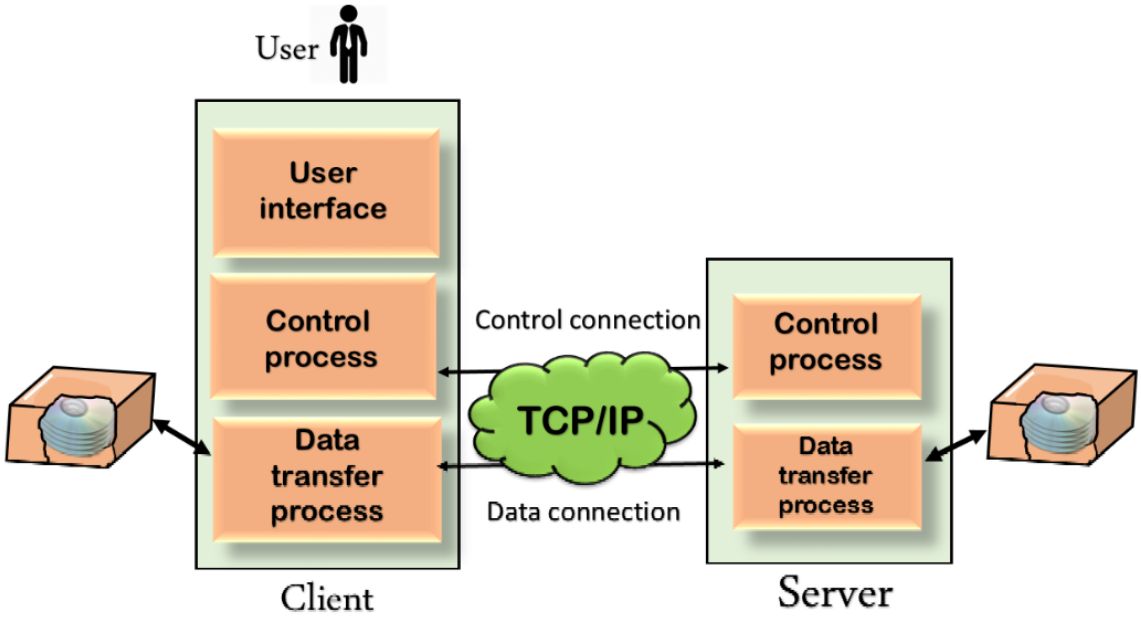
Objectives of FTP

- It provides the sharing of files.
- It is used to encourage the use of remote computers.
- It transfers the data more reliably and efficiently.

Why FTP?

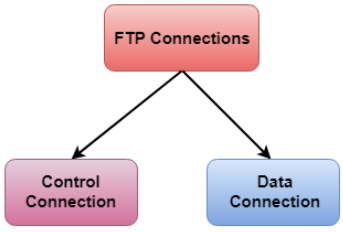
Although transferring files from one system to another is very simple and straightforward, but sometimes it can cause problems. For example, two systems may have different file conventions. Two systems may have different ways to represent text and data. Two systems may have different directory structures. FTP protocol overcomes these problems by establishing two connections between hosts. One connection is used for data transfer, and another connection is used for the control connection.

Mechanism of FTP



The above figure shows the basic model of the FTP. The FTP client has three components: the user interface, control process, and data transfer process. The server has two components: the server control process and the server data transfer process.

There are two types of connections in FTP:



- **Control Connection:** The control connection uses very simple rules for communication. Through control connection, we can transfer a line of command or line of response at a time. The control connection is made between the control processes. The control connection remains connected during the entire interactive FTP session.
- **Data Connection:** The Data Connection uses very complex rules as data types may vary. The data connection is made between data transfer processes. The data connection opens when a command comes for transferring the files and closes when the file is transferred.

FTP Clients

- FTP client is a program that implements a file transfer protocol which allows you to transfer files between two hosts on the internet.
- It allows a user to connect to a remote host and upload or download the files.
- It has a set of commands that we can use to connect to a host, transfer the files between you and your host and close the connection.
- The FTP program is also available as a built-in component in a Web browser. This GUI based FTP client makes the file transfer very easy and also does not require to remember the FTP commands.

FTP Clients

- FTP client is a program that implements a file transfer protocol which allows you to transfer files between two hosts on the internet.
- It allows a user to connect to a remote host and upload or download the files.
- It has a set of commands that we can use to connect to a host, transfer the files between you and your host and close the connection.
- The FTP program is also available as a built-in component in a Web browser. This GUI based FTP client makes the file transfer very easy and also does not require to remember the FTP commands.

Advantages of FTP:

- **Speed:** One of the biggest advantages of FTP is speed. The FTP is one of the fastest way to transfer the files from one computer to another computer.
- **Efficient:** It is more efficient as we do not need to complete all the operations to get the entire file.
- **Security:** To access the FTP server, we need to login with the username and password. Therefore, we can say that FTP is more secure.
- **Back & forth movement:** FTP allows us to transfer the files back and forth. Suppose you are a manager of the company, you send some information to all the employees, and they all send information back on the same server.

Disadvantages of FTP:

- The standard requirement of the industry is that all the FTP transmissions should be encrypted. However, not all the FTP providers are equal and not all the providers offer encryption. So, we will have to look out for the FTP providers that provides encryption.
- FTP serves two operations, i.e., to send and receive large files on a network. However, the size limit of the file is 2GB that can be sent. It also doesn't allow you to run simultaneous transfers to multiple receivers.
- Passwords and file contents are sent in clear text that allows unwanted eavesdropping. So, it is quite possible that attackers can carry out the brute force attack by trying to guess the FTP password.
- It is not compatible with every system.

6)Simple mail transfer protocol(SMTP)

Simple Mail Transfer Protocol (SMTP)

Difficulty Level : Medium • Last Updated : 05 Nov, 2021

Email is emerging as one of the most valuable services on the internet today. Most internet systems use SMTP as a method to transfer mail from one user to another. SMTP is a push protocol and is used to send the mail whereas POP (post office protocol) or IMAP (internet message access protocol) are used to retrieve those emails at the receiver's side.

SMTP Fundamentals

SMTP is an application layer protocol. The client who wants to send the mail opens a TCP connection to the SMTP server and then sends the mail across the connection. The SMTP server is an always-on listening mode. As soon as it listens for a TCP connection from any client, the SMTP process initiates a connection through port 25. After successfully establishing a TCP connection the client process sends the mail instantly.

SMTP Protocol

The SMTP model is of two types:

1. End-to-end method
2. Store-and- forward method

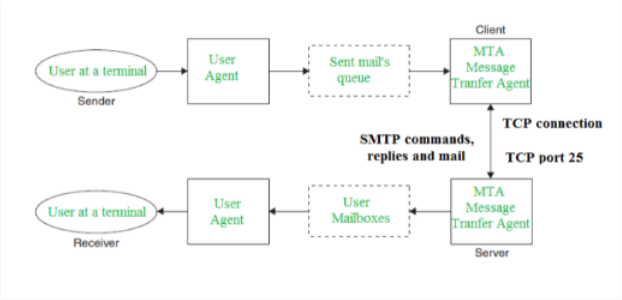
The end-to-end model is used to communicate between different organizations whereas the store and forward method is used within an organization. An SMTP client who wants to send the mail will contact the destination's host SMTP directly, in order to send the mail to the destination. The SMTP server will keep the mail to itself until it is successfully copied to the receiver's SMTP.
The client SMTP is the one that initiates the session so let us call it client- SMTP and the server SMTP is the one that responds to the session request so let us call it receiver-SMTP. The client- SMTP will start the session and the receiver-SMTP will respond to the request.

Model of SMTP system

In the SMTP model user deals with the user agent (UA), for example, Microsoft Outlook, Netscape, Mozilla, etc. In order to exchange the mail using TCP, MTA is used. The user sending the mail doesn't have to deal with MTA as it is the responsibility of the system admin to set up a local MTA. The MTA maintains a small queue of mails so that it can schedule repeat delivery of mails in case the receiver is not available. The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.

Model of SMTP system

In the SMTP model user deals with the user agent (UA), for example, Microsoft Outlook, Netscape, Mozilla, etc. In order to exchange the mail using TCP, MTA is used. The user sending the mail doesn't have to deal with MTA as it is the responsibility of the system admin to set up a local MTA. The MTA maintains a small queue of mails so that it can schedule repeat delivery of mails in case the receiver is not available. The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.



Both the SMTP-client and SMTP-server should have 2 components:

1. User-agent (UA)
2. Local MTA

Communication between sender and the receiver :

The sender's user agent prepares the message and sends it to the MTA. The MTA's responsibility is to transfer the mail across the network to the receiver's MTA. To send mails, a system must have a client MTA, and to receive mails, a system must have a server MTA.

SENDING EMAIL:

Mail is sent by a series of request and response messages between the client and the server. The message which is sent across consists of a header and a body. A null line is used to terminate the mail header and everything after the null line is considered as the body of the message, which is a sequence of ASCII characters. The message body contains the actual information read by the receipt.

RECEIVING EMAIL:

The user agent at the server-side checks the mailboxes at a particular time of intervals. If any information is received, it informs the user about the mail. When the user tries to read the mail it displays a list of emails with a short description of each mail in the mailbox. By selecting any of the mail users can view its contents on the terminal.

Some SMTP Commands:

- HELO – Identifies the client to the server, fully qualified domain name, only sent once per session
- MAIL – Initiate a message transfer, fully qualified domain of originator
- RCPT – Follows MAIL, identifies an addressee, typically the fully qualified name of the addressee, and for multiple addressees use one RCPT for each addressee
- DATA – send data line by line

7)Telnet

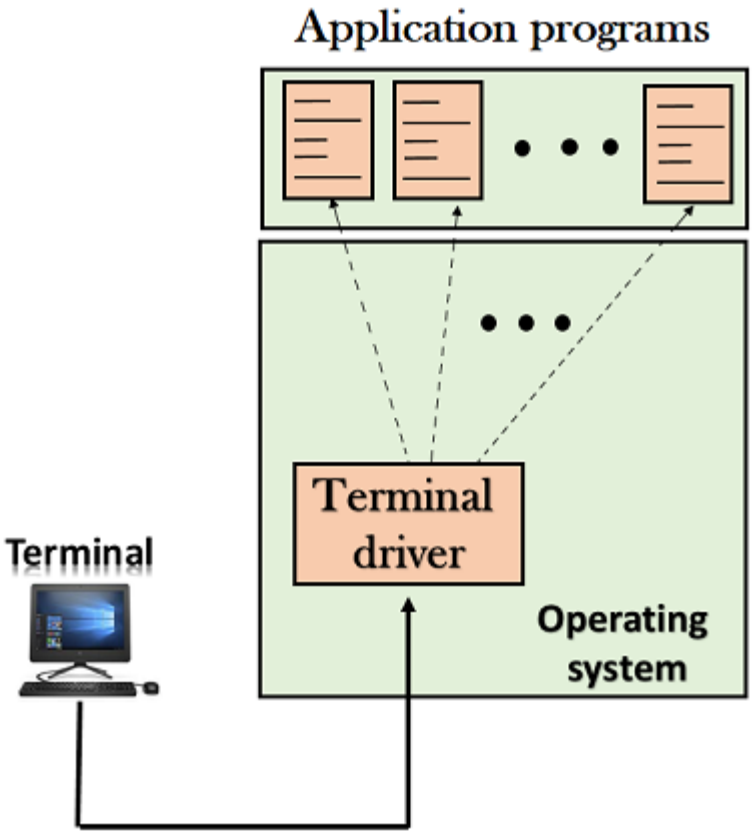
Telnet

- The main task of the internet is to provide services to users. For example, users want to run different application programs at the remote site and transfers a result to the local site. This requires a client-server program such as FTP, SMTP. But this would not allow us to create a specific program for each demand.
- The better solution is to provide a general client-server program that lets the user access any application program on a remote computer. Therefore, a program that allows a user to log on to a remote computer. A popular client-server program Telnet is used to meet such demands. Telnet is an abbreviation for **Terminal Network**.

- Telnet provides a connection to the remote computer in such a way that a local terminal appears to be at the remote side.

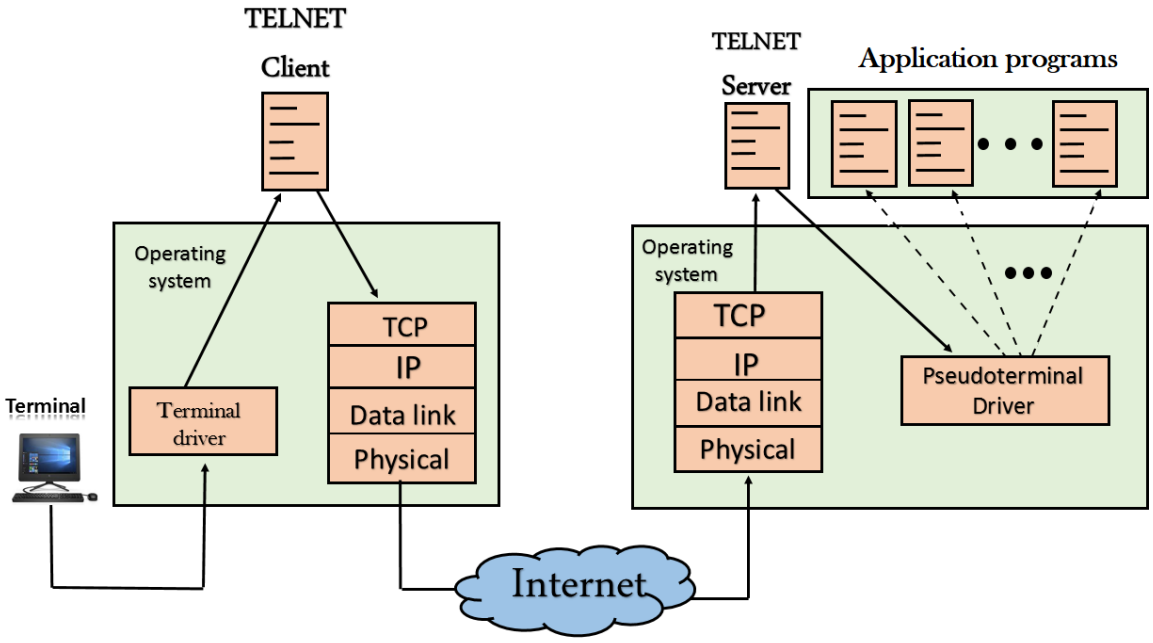
There are two types of login:

- Local Login



  - **When a user logs into a local computer, then it is known as local login.**
  - **When the workstation running terminal emulator, the keystrokes entered by the user are accepted by the terminal driver. The terminal driver then passes these characters to the operating system which in turn, invokes the desired application program.**
  - **However, the operating system has special meaning to special characters. For example, in UNIX some combination of characters have special meanings such as control character with "z" means suspend. Such situations do not create any problem as the terminal driver knows the meaning of such characters. But, it can cause the problems in remote login.**

- Remote login



  - When the user wants to access an application program on a remote computer, then the user must perform remote login.

- How remote login occurs

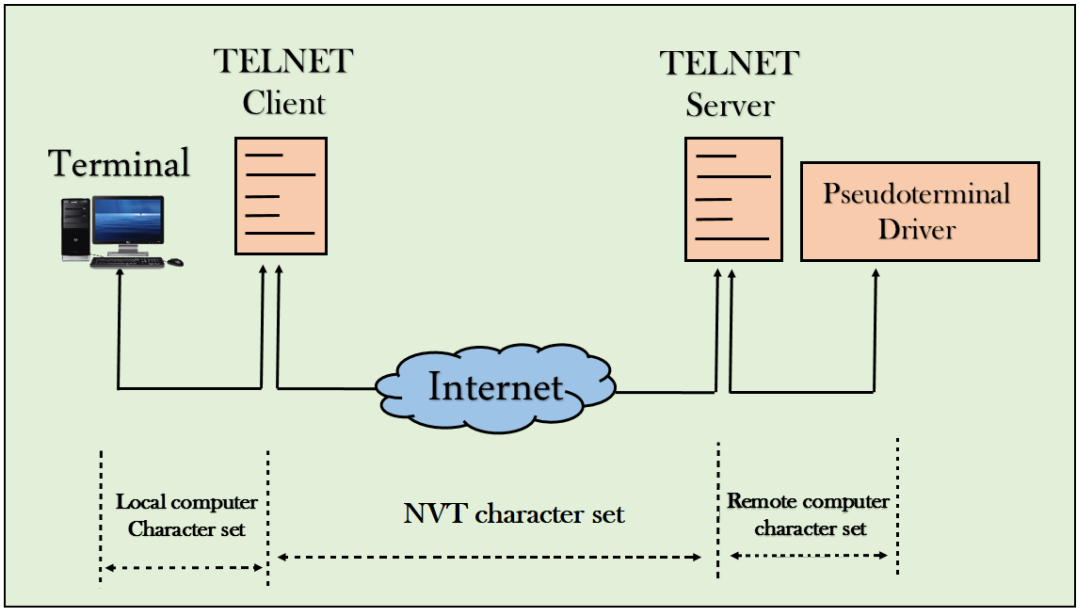
At the local site

The user sends the keystrokes to the terminal driver, the characters are then sent to the TELNET client. The TELNET client which in turn, transforms the characters to a universal character set known as network virtual terminal characters and delivers them to the local TCP/IP stack

At the remote site

The commands in NVT forms are transmitted to the TCP/IP at the remote machine. Here, the characters are delivered to the operating system and then pass to the TELNET server. The TELNET server transforms the characters which can be understandable by a remote computer. However, the characters cannot be directly passed to the operating system as a remote operating system does not receive the characters from the TELNET server. Therefore it requires some piece of software that can accept the characters from the TELNET server. The operating system then passes these characters to the appropriate application program.

Network Virtual Terminal (NVT)



- The network virtual terminal is an interface that defines how data and commands are sent across the network.
- In today's world, systems are heterogeneous. For example, the operating system accepts a special combination of characters such as end-of-file token running a DOS operating system *ctrl+z* while the token running a UNIX operating system is *ctrl+d*.
- TELNET solves this issue by defining a universal interface known as network virtual interface.
- The TELNET client translates the characters that come from the local terminal into NVT form and then delivers them to the network. The Telnet server then translates the data from NVT form into a form which can be understandable by a remote computer.