UNIT 1 INTRODUCTION TO TCP/IP

Structure **Page Nos**

1.0 INTRODUCTION

Transmission Control Protocol (TCP)/Internet Protocol (IP) is a set of protocols developed to allow computers of all sizes from different vendors, running different operating systems, to communicate or to share resources across a network. A packet switching: network research project was started by the USA Government in the late 1960s in 1990s, became the most widely used form of computer networking. This project centered around ARPANET. ARPANET is best-known TCP/IP network. TCP/IP is the principal UNIX networking protocol and was designed to provide a reliable end-to-end byte stream over an unreliable internetwork. TCP is a connection-oriented protocol while IP is a connectionless protocol. TCP supplies logic to give a reliable connection-oriented protocol above IP. It provides a virtual-circuit that two processes can use to communicate. IP provides a connectionless and unreliable delivery system and transfer each datagram independently in the network.

UDP is a connectionless and unreliable protocol running over IP. It adds a checksum to IP for the contents of the datagram and pass members. In this unit we are going to discuss all the protocols of TCPIIP in brief.

1 . OBJECTIVES

After going through this unit, you should be able to know:

- the need of Communication and Internetworking;
- how the Internet and TCP/IP came into existence;
- the meaning of the terms protocol and standard;
- the architecture of the TCP/IP Protocol Suite;
- difference between the OSI model and the TCP/IP Suite;
- the need and functionalities of Interface layer;
- the need and functionalities of Internet layer;
- the need and functionalities of Transport Layer;
- the need and functionalities of Application Layer, and
- how communication takes place on the Internet.

1.2 ORIGIN OF TCP/IP AND INTERNET

Before going through the origin of the Internet, let us examine what is communication.

1.2.1 Communication

Communication the process of sharing ideas, information, and messages with others at a particular time and place. Communication is a vital part of personal life and is also important in business, education; and any other situation where people encounter each other. Communication between two people is an outgrowth of methods developed over centuries of expression. Gestures, the development of language, and the necessity to engage in joint action all play a part. Communication, as we see it . today, has evolved a long way. We will discuss the primitive modes of communication briefly.

i) Early Methods

Early societies developed systems for sending simple messages or signals that could be seen or heard over a short distance, such as drumbeats, fire and smoke signals, or lantern beacons. Messages were attached to the legs of carrier pigeons that were released to fly home (this system was used until World War I, which started in 1914). Semaphore systems (visual codes) of flags or flashing lights were employed to send messages over relatively short but difficult-to-cross distances, such as from hilltop to hilltop, or between ships at sea.

ii) Postal Services

The postal system is a system by which written documents normally enclosed in envelopes, and also small packages containing other matter, are delivered to destinations around the world. Anything sent through the postal system is called post. . In India the East India Company in Mumbai, Chennai and Calcutta introduced the postal system in 1766, further these postal service became available to the general public. Even after implementing different electronic communication mediums, postal system is still one of the popular communication systems available.

iii) Telegraph

The first truly electronic medium for communication was the telegraph, which sent and received electrical signals over long-distance wires. The first practical commercial systems were developed by the physicist, Sir Charles Wheatstone and the inventor Sir William F. Cooke in Great Britain, and by the artist and inventor Samuel F. B. Morse in the United States. Morse demonstrated the first telegraph system in New York in 1837. But regular telegraph service, relaying Morse code (system of code using on and off signals), was not established until 1844. Telegraphers would translate the

letters of the alphabet into Morse code, tapping on an electrical switch, or key. The telegrapher at the other end of the line would decode the tapping as it came in, write down the message, and send it to the recipient by messenger. The telegraph made it possible for many companies to conduct their business globally for the first time.

iv) Telephone

Early devices capable of transmitting sound vibrations and even human speech appeared in the 1850s and 1860s. The first person to patent and effectively commercialise an electric telephone was Scottish-born American inventor Alexander Graham Bell. Originally, Bell thought that the telephone would be used to transmit musical concerts, lectures, or sermons.

The telephone network has also provided the electronic network for new computer-based systems like the Internet facsimile transmissions, and the World Wide Web. The memory and data-processing power of individual computers can be linked together to exchange the data transmitted over telephone line, by connecting computers to the telephone network through devices called modems (modulator-demodulators).

v) Computers and Internet

The earliest computers were machines built to make repetitive numerical calculations that had previously been done by hand. While computers continued to improve, they were used primarily for mathematical and scientific calculations, and for encoding and decoding messages. Computer technology was finally applied to printed communication in the 1970s when the first word processors were created.

At the same time computers were becoming faster, more-powerful and smaller, and networks were developed for interconnecting computers. In the 1960's the Advanced Research Projects Agency (ARPA) of the U.S. Department of Defence, along with researchers working on military projects at research centres and universities across the country, developed a network called the ARPANET, for sharing data and processing time of uniform computer connection over specially equipped telephone lines and satellite links. The network was designed to survive the attack or destruction of some of its parts and continue to work.

Soon, however, scientists using the ARPANET realised that they could send and receive messages as well as data and programs over the network. The ARPANET became the first major electronic-mail network; soon thousands of researchers all over the world used it. Later on the National Science Foundation (NSF) helped connect more universities and non-military research sites to the ARPANET, and renamed it the Internet because it was a network of networks among many different organisations.

Today, the Internet is the widely known computer network. It uses interconnection of computer system by both wired and wireless. Smaller networks of computers, called Local Area Networks (LANs), can be installed, in a single building or for a whole organisation. Wide Area Networks (WANs) can be used to span a large geographical area. LANs and WANs use telephone lines, computer cables, and microwave and laser beams to carry digital information around a smaller area, such as a single college campus. Internet can carry any digital signals, including video images, sounds, graphics, animations, and text, therefore it has became very popular communication tool.

1.2.2 Why do we need the Internet?

The main reason is that each computer network is designed with a specific purpose. For example, LAN is used to connect computers in a smaller area, and it provides fast communication. As a result, networks become specialised identify. In many cases, these networks do not use the same hardware and software technology. It means that, a computer can communicate with the computers attached to the same network, because they are intercompatible. As more and more organisations had multiple computer networks, this became a major issue. As a result, the concept of

internetworking (internet) came into being. This means that there should be a network of all physically separate networks.

1.2.3 Need of Protocol on Communication

All methods of communication described above follow a protocol. Protocol is nothing but a convention. To signify that "Everything is ok and the train can start by a green flag is also a protocol. When we write a letter we follow a protocol. If we look at them carefully, we will find that protocols normally have hidden layers. A good example is human conversation over the phone which can be used as an analogy for communication using computers.

Assume that X and Y, want to have conversation over the telephone about a cricket match. We call this an *idea.* Assume that each person is taking down what other has to say. Thus, the conversation takes place in the form of several messages. A message is a block of sentence. It could also consist of one world such as OK, yes denoting a positive *acknowledgement* (ACK). It could also mean a negative acknowledgement (NAK) or a request to repeat such as come again, pardon me. All this happens both ways.

At the level of idea, X and Y feel that they are discussing a cricket match. However, in reality, the conversation consists of a number of messages.

A message could be too long. It may not be wise for X to speak for half an hour, only to receive a request to repeat. It is therefore necessary to send/receive acknowledgements after each sentence like 'ok', 'come again' etc. A sentence is analogous to a packet in computer world. The sender X will not speak until s/he hears some form of acknowledgement, or will repeat the sentence if she receives a negative acknowledgement. An alternative is *timeout* strategy. The speaker speaks a sentence and waifs for some time for any acknowledgement. If s/he does not hear anything, s/he repeats the sentence.

Apart form this *error* control we take *care* of *flow* control. Flow control refers to the speed mismatch between the listener and speaker. If the speaker speaks too fast, the listener will say go-slow. In computer world, if the receiving computer is not fast enough, and cannot hold any more data, it requests the sender to wait or control the transfer by slowdown.

Therefore, in computer communication, both speaker and lister should agree on the communication language/syntax, scheme of acknowledgement, during flow control, machine error control mechanism, etc. Thus, we *can* say that the conversation is governed by some set of rules known to both the parties. This set of rules is called protocol and it necessary for disciplined manner of conversation/communication.

1.2.4 Problems in Computer Communication

The same concept of protocols described above applies to computer communication. The main difference is that earlier, the devices (telephones) on both sides are compatible. However, in an Internet work, it may or may not be so. The concept of Internetworking though, highly desirable, is not easily achievable.

Thus, any two networks, cannot directly communicate by connecting a wire between the networks. For example, one network could represent a binary 0 by–5 volts, another by +5 volts. Similarly, one could use a packet size of 128 bytes, whereas other could use 256 byte packets. The method of acknowledgement or error detection could be different. There could be many such differences.

1.2.5 Dealing with Incompatibility

The incompatibility issues are handled at two levels:

i) **Hardware Issues**

At the hardware level, an additional component called router is used to connect physically distinct networks as shown in Figure I. A router connects to the network in the same way as any other computer. Any computer connected to the network has a Network Interface Card (NIC), which has the address (network id+host id), hard

coded into it. A router is a device with more than one NICs. Router can connect incompatible networks as it has the necessary hardware (NIC) and protocols (TCP/IP).

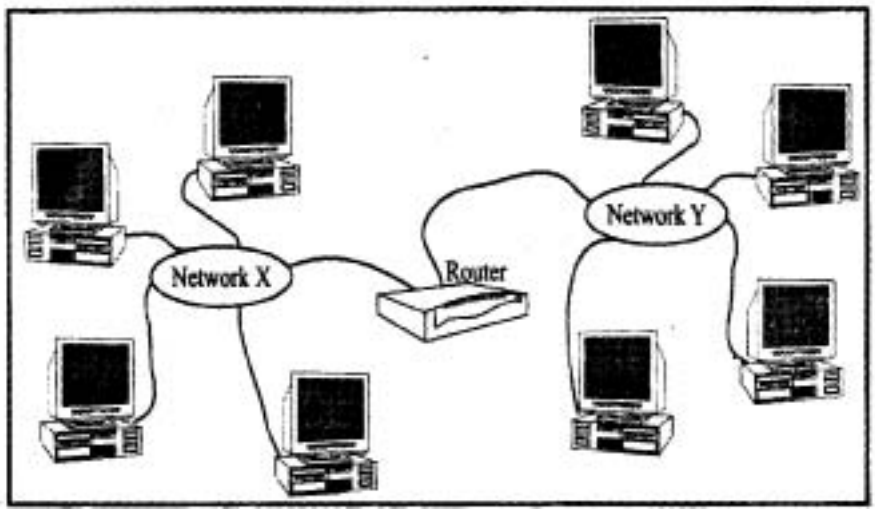

Figure 1: Router connects tbe Networks

ii) Software Issues

The routers must agree about the way information would be transmitted to the destination computer on a different network, since the information is likely to travel through different routers, there must be a predefined standard to which routers must confirm. Packet formats and addressing mechanism used by the networks may differ. One approach could be to perform conversion and reconversion corresponding to different networks. But this approach is difficult and cumbersome. Therefore, the Internet communication follows one protocol, the TCP/IP protocol suite. The basic idea is that it defines a packet size, routing algorithms, error control, flow control methods universally.

It would be unwise to club all these features in a single piece of software – it would make it very bulky. Therefore, all these features are logically sub-grouped and then the sub-groups are further grouped into groups called layers. Each layer has an interface with the adjacent layers, and performs specific functions.

Need for Layering

Since it is difficult to deal with complex set of rules, and functions required for computer networking, these rules and functions are divided with logical groups called layers. Each layer can be implemented interdependently with an interface to other layers providing with services to it or taking its services like flow control and error control functions are grouped together and the layer is called data link layer. Speech in telephone conversation is translated, with electrical segments and vice-versa. Similarly in computer system the data or pattern are converted into signals before transmitting and receiving. These function and rules are grouped together in a layer called physical layer.

1.2.6 A brief history of the Internet

Internet is made up of thousand and thousands of interconnected networks. Although it has become extremely popular in the last decade, it came into being only in 1969.

ARPANET

In the mid 1960s, the Advanced Research Projects Agency (ARPA) in the US Department of Defence (DoD) wanted to find a way to connect computers so that their funded researchers could share their findings. In 1967, ARPA proposed its idea for ARPANET, a small network for connecting computers.

Birth of Internet and TCP/IP

In 1972, Vint Cerf and Bob Kahn, who were part of core ARPANET group, started Internetting Project with the aim to connect different networks together. Diverse

interfaces, different packet sizes, diverse transmission rates presented many difficulties. Kahn Cerf proposed the idea of gateway as an intermediate hardware to transfer data from one network to another.

In 1973, they presented a landmark paper outlining the protocols for end-to-end communication. This paper on TCPIIP included concepts like datagram, gateway, and encapsulation. At this time, responsibility of ARPANET was handed to Defence Communication Agency (DCA).

Came 1977 and communication between networks was made possible. Internet consisting of three different networks, namely, ARPANET, packet radio, and packet satellite was now possible.

In 1978 noticing the popularity of BSD (Berkley Software Distribution) UNIX, which included network capabilities, ARPA signed a contract with Berkley under which TCP/IP software was incorporated in the operating system itself.

In 1983 original ARPANET protocols were abolished and TCP/IP was made de facto standard for Internet.

Time Line

- 1969 Four-node ARPA established
- 1972 Internetting Project begins
- 1973 Development of TCP/IP suite begins
- 1977 An Internet tested using TCP/IP
- 1978 UNIX distributed to universities
- 1983 TCPIIP became the official protocol for ARPANET.

1.2.7 Architecture of the Internet

Internet is organised to form a hierarchy. At the top, there is a very high-speed backbone and at the other end, there are users. There are Network Access Providers (NAP) and Internet Service Providers (ISP) at the intermediate level as shown in the Figure 2.

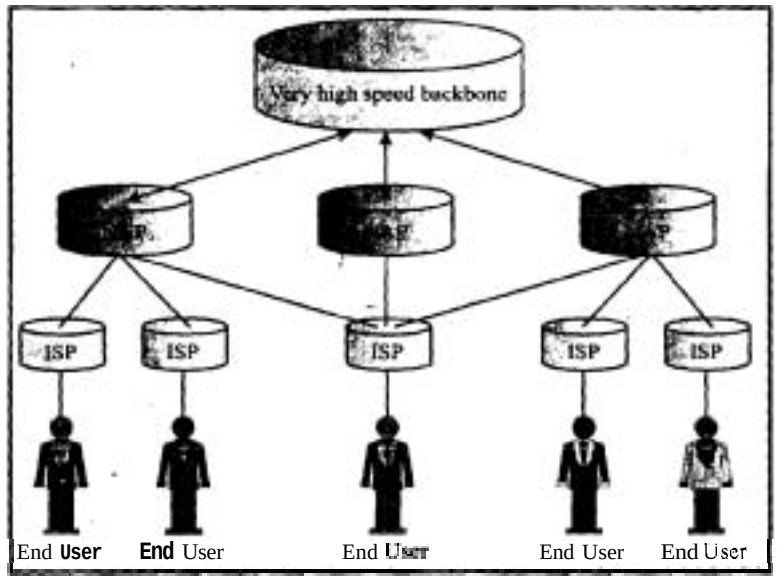

Figure 2: Architecture of Internet

A home user dials into the ISP, may be using a twisted pair telephone connection using a modem. The ISP connects to one of the Network Access Providers, which in turn, connects to the high-speed backbone at a Network Access Point. Network Access Point serves the purpose of connecting backbone networks to provide connectivity between end users.

1.3 TCP/IP LAYERS AND PROTOCOLS

The TCP/IP model is made up of four layers: interface layer, network, transport, and application. The first layer of TCPIIP (Application layer) is similar to the first three layers (Application, presentation and Session layer) of the OSI model. The services of transport layers of both the models are similar. Further, the services of network layers in both models are also similar, while some time network layer is also known as lnternet layer. The last layer of TCPIIP is interface layer, which includes the services of data link layer and physical layer of OSI model. In OSI model, each layer takes the services of the lower layer. Whereas the layers of TCP/IP protocol suite contain relatively independent protocols.

The mapping of OSI & TCPIIP model is shown in the Figure 3, it also shows different protocols included in the TCPIIP model.

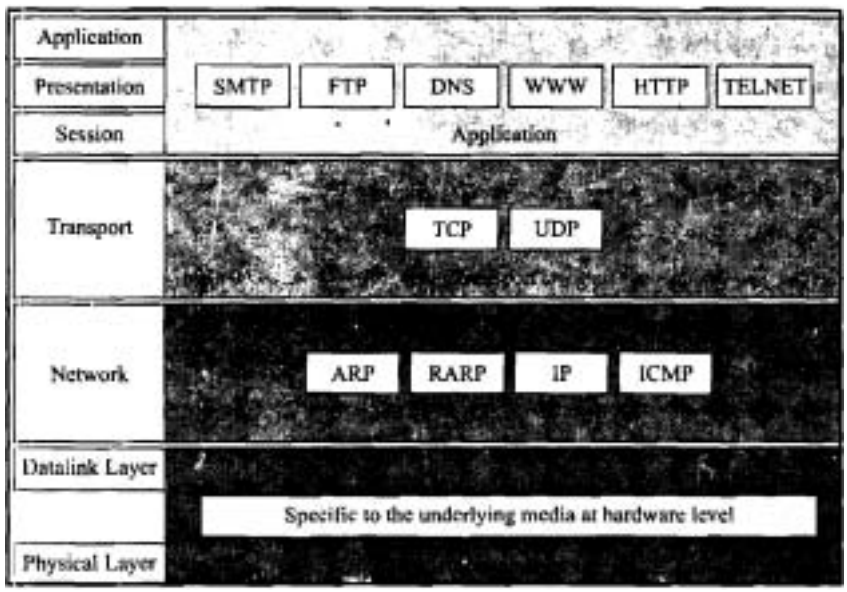

Figure 3: Mapping f OSI model and TCP/IP model

Layers of TCP/IP Protocol Suite

As we know TCP/IP contains four layers and each layer has its specific functions, in the following section lets find out the functions of each layer of TCPIIP.

Interface layer or (Physical + Data Link Layer)

The physical layer deals with the hardware level like, transmission media, connections and the voltage for digital signals. The data link layer deals with media access and control strategies, frame format etc.

Internet Layer or Network Layer

The Internet layer is an important layer in the protocol suite. At this layer, TCPIIP supports Internetworking Protocol (IP). IP is a host-to-host protocol. This layer is responsible for the format of datagrams as defined by IP, and routing a datagram or packet to the next hop, but is not responsible for the accurate and timely delivery of datagrams to the destination in proper sequence. IP allows raw transmission functions allouing user to add functionalities necessary for given application. Ensuring maximum efficiency, TCP/IP supports four other protocols: ARP, RARP, ICMP and IGMP in this layer.

- **Address Resolution Protocol (ARP)**

  On a LAN, each machine is identified with a unique physical address imprinted on the network interface card. ARP is used to find the physical address of a machine when its IP address is known.

- **Reverse Address Resolution Protocol (RARP)**

  It is used to find the IP address of a machine when its physical address is known. It is used when a diskless computer is booted or a computer is connected to the network for the first time.

- **Internet Control Message Protocol (ICMP)**

  IP is unreliable are best effort delivery. In case of failures ICMP is used to send notifications to the sender about packet problems. It sends error and query messages.

- **Internet Group Message Protocol (IGMP)**

  It is used for multicasting, which is transmission of a single message to a group of recipients.

Transport Layer

At this layer, TCP/IP supports two protocols: TCP, UDP, IP is host-to-host protocol, which can deliver the packet from one physical device to another physical device. TCP, UDP, are transport level protocols, responsible for delivering a packet from one process on a device to another process on the other device.

User Datagram Protocol (UDP)

It is simpler of the two protocols. It does not provide reliability. It is, therefore faster, and using for applications in which delay is intolerable (in case of audio and video).

Transmission Control Protocol (TCP)

TCP is reliable, connection oriented protocol. By connection oriented, we mean that a connection must be established between both ends before either can transmit data. It ensures that communication is error-free and in sequence.

Application Layer

As said earlier, it is closer to combined session, presentation, and application layer of OSI model. It allows the user to run various applications on Internet. These applications are File Transfer Protocol (FTP), remote login (TELNET), email (SMTP), WWW (HTTP). The session layer of OSI model is almost dropped in TCP/IP.

Check Your Progress 1

1) How is the TCP/IP model different from OSI model?

.....

.....

.....

2) What are the different layers of TCP/IP protocol suite? Explain.

.....

.....

.....

1.4 NETWORK ACCESS LAYER

The design of TCP/IP hides the function of this layer from users—it is concerned with getting data across a specific type of physical network (such as Ethernet, Token Ring, etc.). This design reduces the need to rewrite higher levels of a TCP/IP stack when new physical network technologies are introduced (such as ATM and Frame Relay).

The functions performed at this level include encapsulating the IP datagrams into *frames* that are transmitted by the network. It also maps the IP addresses to the

physical addresses used by the network. One of the strengths of TCP/IP is its addressing scheme, which uniquely identifies every computer on the network. This IP address must be converted into whatever address is appropriate for the physical network over which the datagram is transmitted.

1.5 INTERNET LAYER

The Internet layer is an important layer in the protocol suite. At this layer, TCP/IP supports Internetworking Protocol (IP). IP is host-to-host protocol. This layer is responsible for the format of datagrams as defined by IP, and routing and forwarding a datagram or packet to the next hop, but is not responsible for the accurate and timely delivery of datagrams to the destination in proper sequence. IP allows raw transmission functions allowing user to add functionalities necessary for given application, ensuring maximum efficiency.

1.5.1 Need for IP Address

The primary goal of the Internet is to provide an abstract view of the complexities involved in it. Internet must appear as single network of computers. At the same time network administrators or users must be free to choose hardware or various internetworking technologies like Ethernet, Token ring etc. Different networking technologies have different physical addressing mechanisms. Therefore, identifying a computer on Internet is a challenge. To have uniform addressing for computers over the Internet, IP software defines an IP address which is a logical address. Now, when a computer wants to communicate to another computer on the Internet, it can use logical address and is not bothered with the physical address of the destination and hence the format and size of data packet.

1.5.2 Classes of IP Address

Internet addresses are 32 bits long, written as four bytes separated by periods (full stops). They can range from 0.0.0.0 to 223.255.255.255. It's worth noting that IP addresses are stored in big-endian format, with the most significant byte first, read left to right. This contrasts with the little-endian format used on Intel- based systems for storing 32- bit numbers. This minor point can cause a lot of trouble for PC programmers and others working with raw IP data if they forget.

IP addresses comprise two parts, the network ID and the host ID. An IP address can identify a network (if the host part is all zero) or an individual host. The dividing line between the network ID and the host ID is not constant. Instead, IP addresses are split into five classes, which allow for a small number of very large networks, a medium number of medium- sized networks and a large number of small networks. The classes of IP address are briefly explained below, the structure of these classes are also shown in Figure 4.

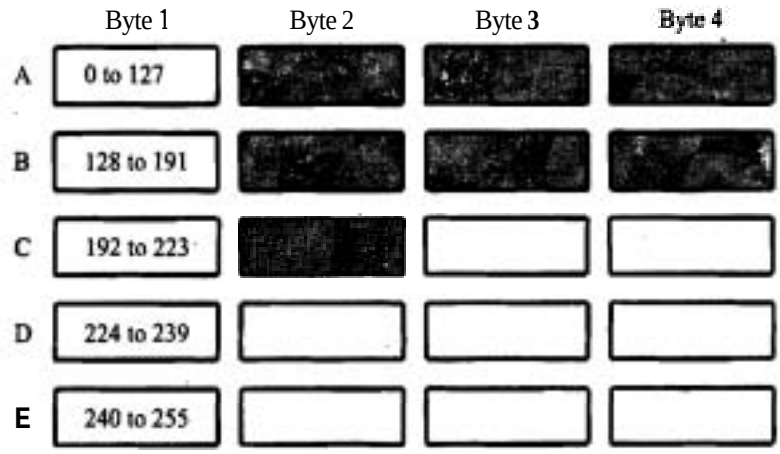

Figure 4: Classes of IP address

Class A addresses have a first byte in the range 0 to 127. The remaining three bytes can be used for unique host addresses. This allows for 126 networks each up to 16 million hosts.

Class B addresses can be distinguished by first byte values in the range 128 to 191 in these addresses, the first two bytes are used for the net ID, and the last two for the host ID, giving addresses for 16,000 networks, each with up to 65536 hosts.

Class C addresses have a first byte in the range 192 to 223. Here, the first three bytes identify the network, leaving just one byte for the individual hosts. This provides for 2 million networks of up to 256 hosts each.

Class D addresses have a first byte in the range 224 to 239. It is designed fo multicasting.

Class E addresses have a first byte in the range 240 to 255.It is reserved for future purposes. The concept is shown here.

- Although these addresses make it possible to uniquely identify quite a lot of networks and hosts, the number is not that large in relation to the current rate of expansion of the Internet. Consequently, a new addressing system has been devised which is a part of Internet Protocol version 6 (IPv6). IPv6 won't come into use for a couple of years, and understanding it isn't essential to understanding how IP works in general.

IP addresses can be further divided to obtain a subnet ID. The main net ID identifies a network of networks. The subnet ID lets you address a specific network within that network. This system of addressing more accurately reflects how real- world large networks re-connected together.

You decide how the subnet ID is arrived at by defining a 32- bit value called the subnet mask. This is logically ANDed with the IP address to obtain the subnet address. For example, if a subnet mask was 255.255.255.0 and an IP address was 128. 124. 14.5, 128. 124 would identify the Class B network, 128. 124. 14 would identify the sub network, and 5 would identify the host on that sub network.

1.5.3 Special Meanings

A few IP addresses have special meanings. A network ID of 0 in an address means "this network", so for local communication only the host ID need be specified. A host ID of 0 means "this host".

A network ID of 127 denotes the loopback interface, which is another way of specifying "this host". The host ID part of the address can be anything in this case, though the address 127.0.0. 1 is normally used. Packets sent to the loopback address will never appear on the network. It can be used by TCP/ IP applications that run on the same machine and want to communicate with one another.

Addresses in the range 224. x. x. x to 239. x. x. x are Class D addresses, which are used for multi- casting. Addresses 240.x. x. x to 247. x. x. x are reserved for experimental purposes.

Net, subnet and host ID's of all binary ones (byte value 255) are used when an IP packet is to be broadcast. Mercifully, an address of 255. 255.255.255 does not result in a broadcast to the entire Internet.

Three sets of addresses are reserved for private address space - networks of computers that do not need to be addressed from the Internet. There is one class A address (10.x. x. x), sixteen class B addresses (172. 16. x. x to 172. 31. x. x), and 256 class C addresses (192. 168.0. x to 192.168. 255. x). If you have equipment which uses IP addresses that have not been allocated by InterNIC then the addresses used should be within one of these ranges, as an extra precaution in case router misconfiguration allows packets to "leak" onto the Internet.

1.5.4 Who Decides the IP Addresses?

IP address of two Computers over the Internet is never same. To ensure this, there is a central authority that issues the IP add-ess. An organisation or individual wanting to

connect to the Internet needs to contact local ISP for obtaining a unique IP address at the Metadata the global level, Internet Assigned Number Authority (IANA) allocates a netid to the ISP.

1.5.5 Internet Protocol

IP is the transmission mechanism used by TCP/IP protocols for host-to-host communication Packets in IP layer are called datagrams. Figure 5 shows the IP datagram format:

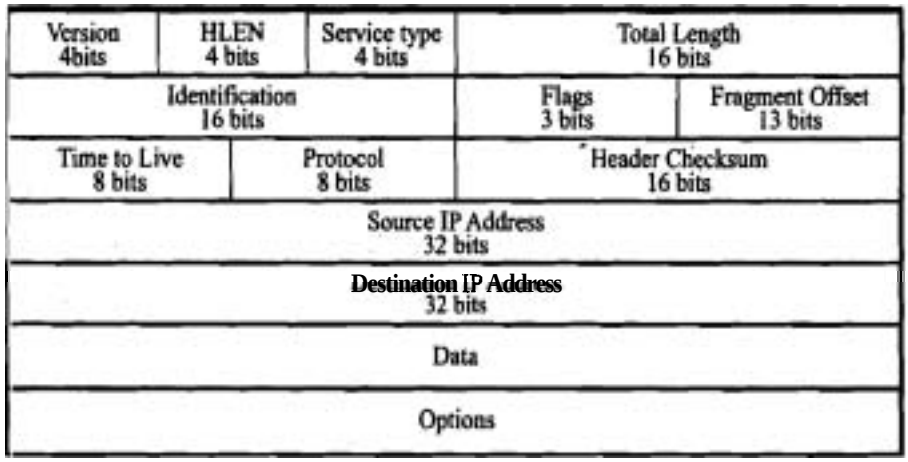

Figure 5: IP datagram

A brief description of header fields in order is given below:

Version (4 bits): It defines the version of IP protocol. Currently, the version is 4(IPv4), indicated by value 4.In future it would contain 6, for IPv6.

HLEN (4 bits): It is needed because length of header is variable. When the header size is 20 bytes, its value is 5(5*4=20). With options, the maximum size is 60 bytes, when the value is 15 (15*4=60). Each value represents number of 32-bit words.

Service Type (8 bits): It is used to define type of service in terms of reliability, precedence, delay and cost.

Total length (16 bits): it defines the total length of IP datagram. The maximum value can be 2^{16}=65536 bytes.

Identification (16 bits): This field is used to unusually identify a datagram. It is useful to know the fragments belonging to same datagram fragments that are part of a datagram which contain same value in identification fields, so that they can be put together in the order to reassemble the datagram at receiver.

Flags (3 bits): This field is used to uniquely identify a datagram. It is useful to know the fragments belonging to same datagrams.

Fragmentation Offset (13 bits): It is a pointer that indicates the offset of the fragment in the original datagram before fragmentation.

Time to Live (8 bits): It is used to control the maximum number of hops visited by the datagram. It is needed to restrict a datagram from continuing to travel in infinite loop without reaching the destination. This infinite looping may cause network congestion. This field limits the lifetime of datagram, after which the packet is discarded, so that datagram does not travel in infinite loop.

Protocol (8 bits): An IP datagram may encapsulate data from various higher-level protocols like TCP, UDP, ICMP, and IGMP. This field specifies the final destination protocol to which the IP datagram should be delivered. Each protocol TCP, UDP etc. identified with a unique number.

Source Address (32 bits): It stores the IP address of the source.

Destination Address (32 bits): It stores the IP address of the final destination.

Options: This field contains optional information such as routing details, timestamp etc. For instance, it can store route of a datagram, in the form of IP addresses of intermediate routers, optionally the time when it pass through that router.

The functions performed at this layer are as follows:

- **Define the datagram, which is the basic unit of transmission in the Internet:** The TCP/IP protocols were built to transmit data over the ARPANET, which was a packet switching network. A packet is a block of data that carries with it the information necessary to deliver it in a manner similar to a postal letter that has an address written on its envelope. A packet switching network uses the addressing information in the packets to switch packets from one physical network to another, moving them towards their final destination. Each packet travels the network independently of any other packet. The datagram is the packet format defined by IP.
- **Define the Internet addressing scheme:** IP delivers the datagram by checking the destination address in the header. If the destination address is the address of a host on the local network, the packet is delivered directly to the destination. If the destination address is not on the local network, the packet is passed to a router for delivery. Router is a devices that switch packets between the different physical networks. Deciding which router to use is called routing. IP makes the routing decision for each individual packet.
- **Move data between the Network Access Layer and the Host-to-Host Transport Layer:** When IP receives a datagram that is addressed to the local host, it must pass the data portion of the datagram to the correct host-to-host transport layer protocol. This selection is done by using the protocolfiled in the datagram header. Each host-to-host transport layer protocol has a unique number that identifies it to IP.
- **Route datagrams to remote hosts:** Internet gateways are commonly (and perhaps more accurately) referred to as IP routers because they use IP to route packets between networks. In traditional TCP/IP jargon, there are only two types of network devices: gateways and hosts. Gateways forward packets between networks and hosts do not. However, if a host is connected to more than one network (called a multi-homed host), it can forward packets between the networks. When a multi-homed host forwards packets, it acts like my other gateway and is considered to be a gateway.
- **Fragment and reassemble datagrams:** As a datagram is routed through different networks, it may be necessary for the IP module in a gateway to divide the datagram into smaller pieces. A datagram received from one network may be too large to be transmitted in a single packet on a different network. This condition only occurs when a gateway interconnects dissimilar physical networks. For example, a physical network may be using Ethernet which specify packet (Frame) size of main 1500 bytes. Other physical network may be using FDDI that specify packet size mark of 4464 bytes. Therefore, the MTV of Ethernet to 1500 & MTU of FDDI is 4464. An IP packet-being transmitters over Ethernet is fragments limit 1500 bytes each.

Each type of network has a maximum transmission unit *(MTU)*, which is the largest packet it can transfer. If the datagram received from one network is longer than the other network's MTU, it is necessary to divide the datagram into smaller fragments for transmission. This division process is called *fragmentation*.

1.5.6 Address Resolution Protocol (ARP)

We have seen that IP address makes the addressing uniform on the Internet. Routing of packets is done using the IP addresses of the packet. However, communication in a local network is broadcast, which is done using physical address. Therefore, when the packet reaches the destined network, there must be a process of obtaining the physical address corresponding to its IP address, of a computer in order to finally deliver the

datagram to the destined computer. The physical address corresponding to an IP address is resolved by using address resolution protocol (ARP). ARP maps given IP address to a physical address as shown in the ***Figure*** 6. It takes host's IP address as input and gives its physical address as output.

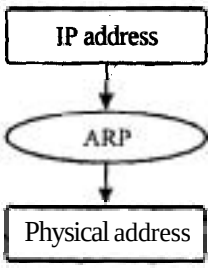

Figure 6: ARP maps the IP address to the physical address

ARP assumes that every host knows its IP address and physical address. Any time a host needs to know the physical address of another host on the network, it creates an ARP packet that includes the IP address X of the destination host asking—Are you the one whose IP address is X? If yes, please send back your physical address. This packet is then broadcasted over the local network. The computer, whose IP address matches X, sends a ARP reply packet, with its physical address. All the other hosts ignore the broadcast. Next time the host needs to send a datagram to the same destination, it need not broadcast an ARP query datagram; instead it can look up in its ARP cache. If the mapping is not found in the cache, then only the broadcast message is sent.

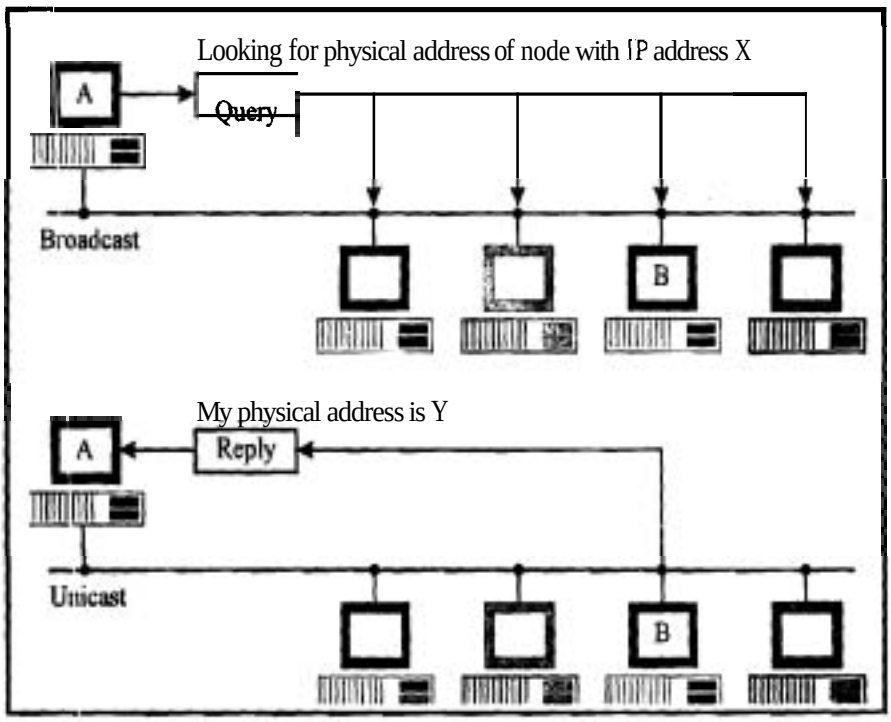

Figure 7: ARP query and reply

1.5.7 Reverse Address Resolution Protocol (RARP)

This protocol performs the job exactly opposite to ARP. It maps a physical address to its IP address as shown in ***Figure*** 8. Where is this needed? A node is supposed to have its IP address stored on its harddisk. However, there are situations when the host may not have hard disk at all, for example a diskless workstation. But also when a host is being connected to the network for the first time, at all such times, a host does not know it!; IP address. In that case RARP find out the IP address, this process is shown in ***Figure*** 9.

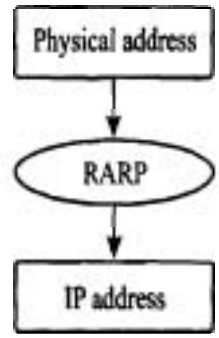

Figure 8: RARP maps the physical address to the IP address

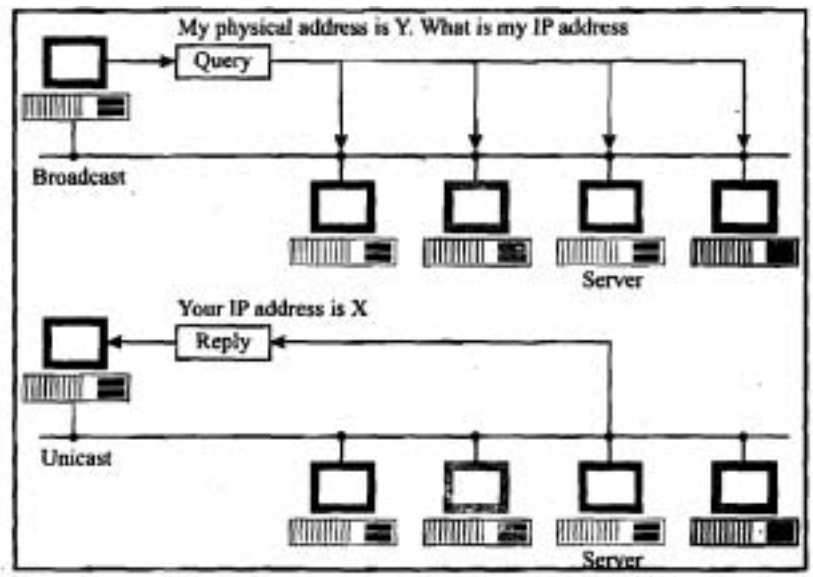

Figure 9: RARP query & reply

1.5.8 Internet Control Message Protocol (ICMP)

IP is best effort data delivery protocol. It means that **IP** makes best effort of delivering a datagram from source to destination. However, it does not guarantee that the datagram would be delivered correctly. IP has no error-reporting mechanism. A router or the final destination needs to inform the source if it must discard a datagram. **IP** also lacks a mechanism for host and management queries. A host sometimes needs to determine if a router or another host is alive to avoid sending datagrams to a router which is down. **ICMP** has been designed to report error occurred doing delivery of datagram.

The errors reported by **ICMP** are generally related to datagram processing. **ICMP** only reports errors involving fragment 0 of any fragmented datagrams. The **IP, UDP** or **TCP** protocols will usually take action based on **ICMP** messages. **ICMP** generally belongs to the IP layer of **TCP/IP** but relies on IP for support at the network layer. **ICMP** messages are encapsulated inside IP datagrams.

ICMP will report the following network information:

- Timeouts
- Network congestion
- Network errors such as an unreachable host or network.

The ping command is also supported by **ICMP,** and this can be used to debug network problems.

ICMP Messages

The ICMP message consists of an 8-bit type, an 8-bit code, an 8-bit checksum, and contents which vary depending on code and type.

ICMP is used for many different functions, the most important of which is error reporting. Some of these are *"port unreachable", "host unreachable", "network unreachable", "destination network unknown", and "destination host unknown".* Some not related to errors are:

- **Timestamp request and reply** allows one system to ask another one for the current time.

Address mask and reply is used by a diskless workstation to get its subnet mask at boot time.

- **Echo request and echo reply** is used by the ping program to see if another host is reachable. Thus two types of ICMP messages are defined: Error messages and Query messages.
- **Source Quench Message:** When datagrams arrive too quickly for processing, the destination host or an intermediate gateway sends an ICMP ***source quench message*** back to the sender. This message instructs the source to stop sending datagrams temporarily.
- **Redirect routes:** A gateway sends the ICMP ***redirect message*** to tell a host to use mother gateway, presumably because the other gateway is a better choice. This message can only be used when the source host is on the same network as both gateways.

16 TRANSPORT LAYER

The transport layer runs on top of the Internet layer and is concerned with process-to-process delivery of data packets. Here, process is a running application program on a host. The main task of transport layer is to ensure correct delivery of packets. This introduces several responsibilitieslike flow control mechanism. By flow control we mean that a faster sender must not drawn a slow receiver with data packets resulting in data loss. The transport layer also provides connection mechanism. It establishes connection with the receiver, transfers data, and terminates the connection. The transport layer includes acknowledgement service to check for packet loss in the network. In TCP/IP transport layer is represented by two protocols: TCP and UDP. UDP is simpler of the two protocols. It is unreliable, connectionless transport protocol. Unreliability means that it does not provide any flow control, error control, and acknowledgement services. Connectionless means that no connection is established between the sender and the receiver prior to sending data. If UDP detects an error, it silently drops the packet. What is the use of UDP if it is so unreliable? It is very useful when the speed of the delivery is critical and loss of a small number of packets is tolerable like images, voice or video. TCP is useful when it is necessary to ensure error free delivery of packets from source to the destination.

1.6.1 Transmission Control Protocol

We have seen that IP packets may travel through different routes and may arrive out of sequence. TCP puts the packets in sequence. An intermediate router may discard the IP packets and they may not arrive at the destination at all. TCP checks for missing packets and handles this issue retransmission request. Some packets may get duplicateddue to hardware malfunction. TCP discards duplicate packets. Hence, TCP takes care of all these situations and makes the Internet reliable. Before discussing TCP and UDP in detail, let us discuss process to process communication.

Process to Process Communication

IP provides host-to-host communication. TCP provides process-to-process communication using the client server paradigm as shown in ***Figure 10***, suppose out

of several application programs running on the host, TCP delivers the data packet to the destined application program.

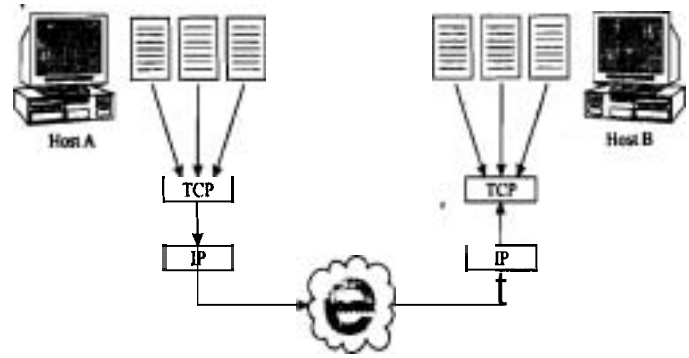

Figure 10: Domain of IP and TCP/UDP TCP like a multiplexer

Applications running on different hosts communicate with TCP with the help of concepts called ports. A port is a unique 16-bit number allocated to an application program. Ii is used when an application wants to open a TCP connection with another application on a remote computer. To understand how, let us take an analogy. A wants to call B staying in a hotel X, room **no.Y.** Now A dials the phone number of the hotel, after getting a response from the operator, A must tell the room number of B to be able to connect to **B.This information** is used to redirect the call to B.

Now, suppose an application X on host A wants to communicateto application Y on host B as given in Figure 11 below. It must first reach B and then Y. Reaching B is not enough because there may be multiple applications running on B which might want to communicate with another application running on some other host. Hence, specifying the host is not enough.

A process on local host, called client, needs some services from a process on remote host called server. The client uses a random port number called ephemeral port number. The server also defines a port **number** with itself, which is known to all, and not random. This is **because** a client wanting to access a particular service will not know the port number if it is random. Therefore, TCP uses **well-known** port numbers. This also allows multiple TCP connections between two hosts are possible. Although the IP address would be same, the port number would be different each time.

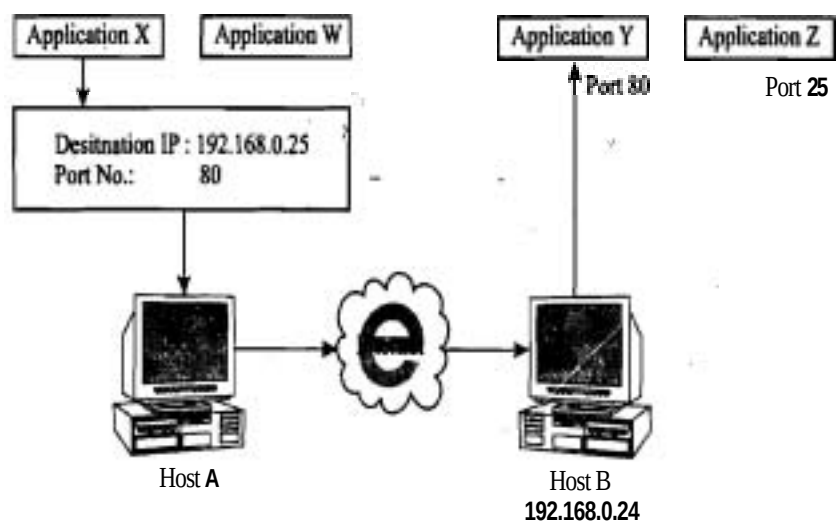

Figure 11: Use of port

The application X now provides the IP address of B, to identify the host B from thousands of hosts on the Internet. It provides port number. corresponding to application Y running on B to uniquely identify the application.

Sockets

The term Socket is used to identify the IP address and Port number concatenated together as shown in Figure 12.

Socket=IP address +port number

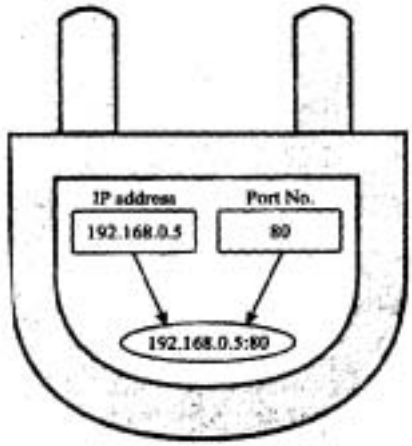

Figure 12: Socket

Socket Address

A client socket identifies a client process uniquely and a server socket identifies server process uniquely. Hence, a pair of sockets identifies a TCP connection between two applications on two different hosts.

Services Provided by TCP

Process to process communication: TCP provides process-to-process communication as discussed above in section 1.6.1.

- **Stream Delivery Service:** TCP is stream oriented protocol. It allows the sending process to deliver data as a stream of bytes and allows receiving process to obtain data as a stream of bytes. For this, TCP needs two buffers, the sending buffer, and the receiving buffer, one for each direction. Figure 13 shows the use buffers in TCP stream delivery.

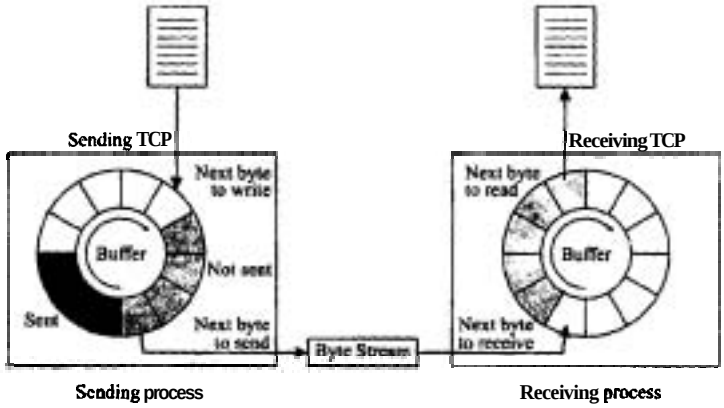

Figure 13: Stream Delivery Service

At the sending site, the buffer can have three types of chambers. The white section is the section of empty bytes which can be filled by the sender. The gray section contains bytes that have been sent but not acknowledged. They are kept in the buffer until the ACK is received. The black area contains the bytes to be sent by the sending TCP.

At the receiving site, the buffer is divided into two sections. The white section is the section of empty bytes which can be filled by the received bytes. The black section contains the bytes which can be read by the receiving process. It handles the speed

mismatch between the sender and the receiver. There is one more step before the data can be sent.

IP, the service provider of TCP, sends data as packet, and not as stream of bytes. TCP groups a number of bytes into a segment, add a TCP header and delivers the packet to the IP layer for transmission. These segments can be of different sizes. The segment is now encapsulated inside a IP datagram and transmitted.

- **Connection Oriented Service:** When a process host A wants to communicate to a process on host B, the two TCPs establish a connection, data is exchanged, and the connection is terminated. This connection is virtual and not physical. All the IP datagrams may follow different routes, may arrive out of sequence. TCP take the responsibility of delivering the bytes in order. Hence maintaining a virtual connection. TCP provides full duplex communication. Data can flow in both the directions.
- **Reliability: TCP** is a reliable transport protocol. TCP incorporates error control, flow control, and acknowledgement services.

TCP Connections

We have said again and again that TCP is connection oriented. You may wonder how TCP connections orients as it uses the services of IP, a connectionless service. It is because TCP connection is virtual and not physical. It uses the services of IP for packet delivery but controls the connection itself. It takes care of reordering, retransmission, duplication, of which IP is unaware.

Let's see this connection establishment in detail.

TCP uses a technique called **three-way handshaking**. It means that three messages are exchanged between the sender and the receiver to establish the connection as shown in *Figure 14*. After this, TCP guarantees a reliable data transfer. It has been roved that this three way handshake is necessary and sufficient for establishing a successful connection.

The server waits passively to accept any active connection request. The client always initiates the connection request. The client sends the SYN segment in which only the SYN flag is set. This segment is used for the synchronisation of sequence numbers. The client chooses a random number and sends it as the first sequence number. It does not carry any relevant data, and consumes one sequence number.
The server sends the SYN ACK segment, with SYN and ACK bits set by sending this, server informs the client about the sequence number it will use in future and acknowledges the receipt of first SYN segment. It also consumes one sequence number.

The client sends the third segment, an ACK segment. It is an acknowledgement for the second segment. It bears the same sequence number as the first segment and hence, does not consume any sequence number. After this handshake, communication can start between the two ends.

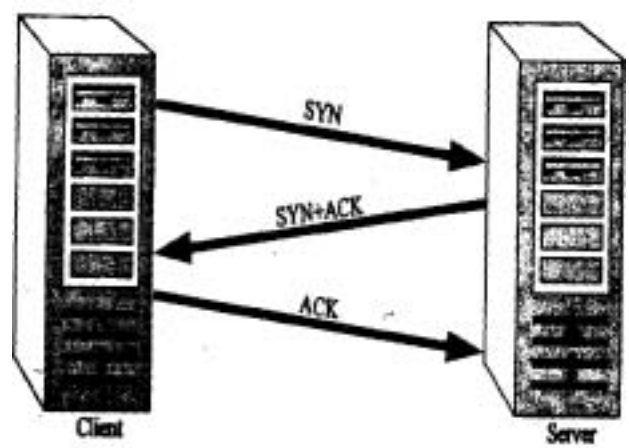

Figure 14: Three-way handshaking

The format of a TCP segment is shown in Figure 15. The header size is between 20 to 60 bytes, followed by data. It contains 20 bytes if the header does not contain any option.

Figure 15: TCP segment format

Here is a brief description of header fields in order.

Source port number (16 bit): It specifies the source port number corresponding to the application which is sending the segment.

Destination Port Number (16 bit): It specifies the port number of the destination computer, corresponding to the receiving applications.

Sequence Number (4 bytes): It specifies the number assigned to the first byte of the data portion contained in this TCP packet. Each byte to be transmitted is numbered in an increasing sequence. It tells the destination host which byte comprises the first byte of the TCP segment. During the connection establishment phase, the source and the destination generate a unique random number. If this random number is 3000, and the first segment contains 2000 bytes, then the sequence no. will be 3002.3000 and 3001 are used in connection establishment. The second segment would have a sequence number of 5002(3002+2000), and so on.

Acknowledgement Number (4 bytes): On receiving a packet with sequence number X, the receiver sends back X+1 as the acknowledgement number. It defines the sequence number which the receiver is expecting next.

Header length (4 bits): The header length can be between 20 to 60 bytes. therefore, the value of this field can be between 5 (5*4=20) and 15(15*4=20)
Reserved: These 6 bits are reserved for future use.

Flag (6 bits): This field signifies 6 control flags, each one of them occupying one bit. Out of these, two are most important. The SYN flag indicate that the source wants to establish a connection with the destination. The FIN flag means that the sender wants to close the TCP connection.

Window Size (2 bits): This field determines the size of the window the other party must maintain. It is useful for flow control.

Checksum (16 bits): It contains the checksum for error detection.

Urgent Pointer: This field is used in situations when the segment contains urgent data. It specifies the number that must be added to obtain the number of the first urgent byte in the data section of the segment.

Pseudoheader: It is the part of the header of the IP packet in which the segment is encapsulated with some fields set to zero. It is added to have better error control. This way we ensure that if the IP header is corrupted, the segment is not delivered to the wrong host.

The protocol field is added to ensure that the packet belongs to TCP and not UDP because a process can use either TCP or UDP, the destination port number can be same, The value of this field is 6. If this value is changed, the checksum calculation will detect it and discard the packet instead of delivering it to the wrong protocol.

1.6.2 User Datagram Protocol (UDP)

The other protocol in transport layer is UDP. UDP is connectionless protocol. It allows a computer to send data without needing to establish a virtual connection. There is no error checking except for checksum. It does not provide sequencing, flow control or acknowledgements. Thus, UDP packets may be lost, arrive out of sequence, or duplicated. It is left to the application program to take care of these issues.

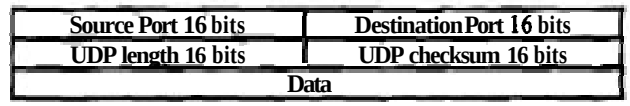

Figure 16: UDP packet format

Here is a brief description of header fields of DP packet is shown in Figure 16.

Source port number (16 bit): It specifies the source port number corresponding to the application which is sending the segment.

Destination Port Number (16 bit): It specifies the port number of the destination computer, corresponding to the receiving applications.

Total Packet Length (16 bits): It defines the total length of the UDP packet. However, this field is redundant, because there is a packet length field in IP, which encapsulates the UDP packet. Therefore UDP packet length=IP packet length-IP header length. This field is retained as an additional check.

Checksum: It is again used for error detection.

Pseudoheader: This field is added for the same purpose as in TCP.

☞ **Check Your Progress 2**

1) What are the functionalities of Internet layer?

.....

.....

.....

.....

.....

2) What do you mean by ICMP source quench message

.....

.....

.....

.....

1.7 APPLICATION LAYER

Till now, we have studied Internet layer and Network layer protocols which provide end-to-end delivery of packets. These protocols would have no meaning without the Application Layer Protocols like DNS, HTTP, TELNET, and FTP. We will discuss all the application layer protocols in brief.

1.7.1 Electronic Mail

Email is one of the most popular Internet services. At the beginning of the Internet era, emails were short and consisted of text only. Today it is much more complex allowing to send text, audio and video. It also allows to send a message to more than one recipient.

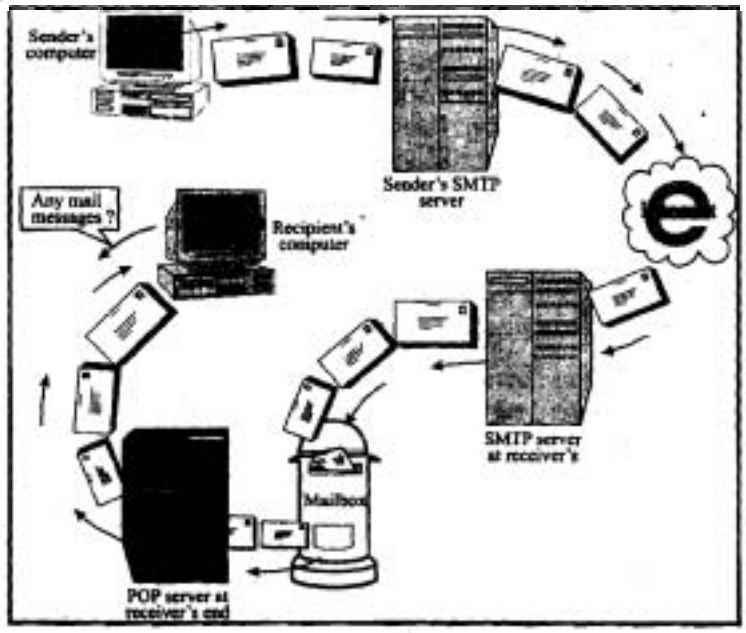

Figure 17: Journey of email message

In a typical scenario, both the sender and recipient are users on two different systems, and are connected to the system via a point to point WAN, or a LAN, which uses an email server for handling emails. Hence the message needs to be sent over the internet. How the sender computer sends the message through internet to the receipts computer is shown in Figure 17.

The sender uses an USER AGENT PROGRAM to prepare a message. It then sends the message through the LAN or WAN. This is done through a pair of message transfer agents (client and server). Whenever the sender has a message to send, he calls the user agent, which in turn, calls the Mail Transfer Agent (MTA) client. The MTA client establishes a connection with the MTA server, which is running all the time. The system at sender's site queues all the messages received. It then uses an MTA client to send message to the system at receiver's site. After the message arrives at Bob's mail server, another client server agents comes into picture which are called Message Access Agents (MAA). The receiver sends a request to the MAA server, which is running all the time, and requests the transfer of messages.

The email transfer protocols

For email messaging, every domain maintains an email server. The server runs protocols software that enable email communication. There are two main emails protocols: POP and SMTP. Because both the email protocol software programs run on server computers, the server computers are themselves called POP server and SMTP server. A single server can host both the POP and SMTP server programs.

SMTP sewer (Simple Mail Transfer Protocol)

SMTP is the Internet protocol used to transfer electronic mail between computers. The second generation of SMTP is called ESMTP (for Extended SMTP), but the differences are not important for this introduction.

It actually transfers the email message from the SMTP server of the sender to the SMTP server of the recipient. Its main job is to carry the message between the sender and the receiver. It uses TCP/IP underneath. That is, it runs on top of TCP/IP.

At the sender's site, an SMTP server takes the message sent by a user's computer. The SMTP server at the sender's end then transfers the message to the SMTP server of the recipient.

The SMTP server at the recipient's end takes the message and stores it in the appropriate user's mailbox.

POP server

The Post Office Protocol provides a standard mechanism for retrieving emails from remote server for a mail recipient. Suppose that a home user X usually connects to the Internet using a dial–up connection to an ISP.Also, another person Y has sent an email to X, when X is not connected to the Internet. Now, the email message gets stored in the mailbox of the user provided by the ISP.

When X connects to the Internet next time and wants to see the new mails that have arrived for him since the last time he had connected to the Internet, he opens his email program. That email client program invokes POP client, which contacts the POP server. The email client opens the mailbox for the user X and sends the emails arrived for him to the POP client.

POP server is needed because SMTP server expects the destination host to be online all the time so that it can make a TCP connection to it, and forward the mail. But desktop computers are usually powered off after business hours. Hence the SMTP server forwards the mails to the mailbox, and the POP server retrieves the mails from the respective email boxes of the user when requested by POP clients.

POP3 (version 3). This is simple and limited in its functionality. It has two modes: the delete mode and the keep mode, In delete mode, mail is deleted from the mailbox after retrieval. It is used when the user is working at his personal computer and can save and organise the information after reading or replying. The keep mode is used when the user would check his mail from his primary computer. The mail is read but kept in the system for later retrieval.

IMAP Protocol

Another mail access protocol is the Internet Mail Access protocol, version 4(IMAP4). It is similar to POP3, but it has more features as given below. In this, a user can-

* check the email header before downloading.
- search the contents of the email for a specific keyword prior to downloading.

  partially load the email it is helpful if the bandwidth is limited.
- create, rename, and delete mailboxes on the mail server.
- create a hierarchy of mailboxes in a folder for email storage.

File Transfer Protocol

We have seen how email works in Figure 17. But there are situations when we want to receive or send a file from or to a remote computer. Emails are just short messages. In fact, the greatest volume of data exchange in the lnternet today is due to file transfer. Special software and a set of rules called File Transfer Protocol (FTP) exists for this purpose. It is an application layer protocol that is aimed at providing a very simple interface for any user of the Internet to transfer files. Whether you know it or not, you most likely use FTP all the time.

Although transferring files form one system to another seems simple and straight forward task, some problems must be dealt with first. For example, two systems may use different file name conventions. Two systems may have different way to represent text or data. Two systems may have different directory structures. All these problems are elegantly solved by FTP.

FTP basics

FTP differs from other application layer protocols in one respect. All the other application layer protocols use a single connection between client and server. However, FTP uses two TCP/IP connections. One connection is used for actual data transfer, and the other is used for control information as shown in Figure 18. This makes FTP more efficient.

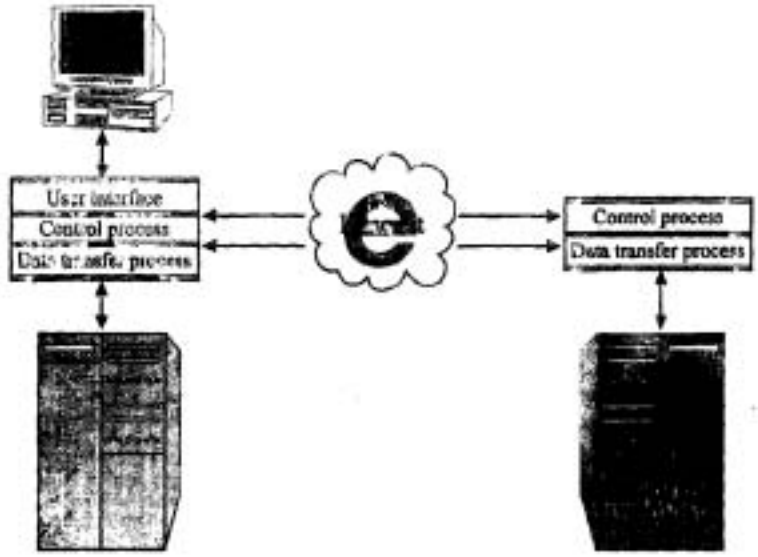

Figure 18: Data transfer through Internet

The components of the client and the server are shown in the Figure 18. The user interface component is not required at the server side, since there is no interaction exactly at the time of file transfer. The TCP control connection is made between the control processes of the client and the server. The TCP data transfer connection is made between the data transfer processes of the client and the server. While the file is being transferred, the server keeps track of how many bytes have been sent, and how much are remaining. It sends this information on the control connection, and this way reassures the user about the progress of data transfer. If multiple files are to be transferred in a single FTP session, then the control connection remains active throughout the entire FTP session. The data connection is opened and closed for each file transferred.

Prior to data transfer, the following attributes need to be specified:

Type of the file to be transferred: The file to be transferred can be an ASCIII, EBCDIC, or an image file. Image file is actually a misnomer. It has nothing to do with the image file. Instead, it signifies a binary file that is not interpreted by FTP in any manner, and it is sent as it is. If the file has to be sent as ASCII or EBCDIC, then the destination must be ready to accept in that mode. If the file has to be transferred without any regard to its content, then the third mcde is used.

The structure of the data: File can be transferred by FTP by interpreting its structure in the flowing ways:

Byte-oriented structure: The file can be transferred as a continuous stream of data, where in no structure is assumed.

Record-oriented structure: The file is divided into records and the records are transferred one by one.

The transmission mode: FTP can transfer a file using any of the three modes as described.

Stream mode: This is the default mode; the data is delivered from FTP to TCP as a continuous stream of bytes.

Block mode: The data can be delivered from FTP to TCP as a block. Each data block follows a three byte header. The first byte is the descriptor, whefeas the remaining to bytes defines the size of the block.

Compressed mode: The file can be compressed before sending. Normally Run Length Encoding is used for compressing a file.

This information is used by FTP to resolve the heterogeneity problem.
WU-FTP, CuteFTP is most commonly used FTP software applications.

What is an FTP Site?

An FTP site is like a large filing cabinet. With a traditional filing cabinet the person who does the filing has the option to label and organise the files, however they see fit. They also decide which files to keep locked and which remain public. It is the same with an FTP site.

The virtual 'key' to get into FTP site is the UserID and Password. If the creator of the FTP site is willing to give everyone access to the files, the UserID is 'anonymous' and the Password is your e-mail address (e.g. name@domain.com). If the FTP site is not public, there will be a unique UserID and Password for each person who is granted access.

When connecting to an FTP site that allows anonymous logins, you're frequently not prompted for a name and password. Hence, when downloading from the Internet, you most likely are using an anonymous FTP login and you don't even know it.
To make FTP connection you can use a standard Web browser (Internet Explorer, Netscape, etc.) or a dedicated FTP software program, referred to as an FTP 'Client'. When using a Web browser for an FTP connection, FTP uploads are difficult, or sometimes impossible, and downloads are not protected (not recommended for uploading or downloading large files).

When connecting with an FTP Client, uploads and downloads couldn't be easier, and you have added security and additional features. For once, you're able to resume a download that did not successfully finish, which is a very nice feature for people using dial-up connections'who frequently loose their Internet connection.

1.7.2 Domain Name System (DNS)

Machines on the Internet are identified by numerical "IP addresses" like 192.0.2.1. Domain names make it possible to refer to machines by a name rather than a number, which is definitely easier. If we had to remember the IP addresses of all of the Web sites we visit every day, we would all go nuts. Human beings just are not that good at remembering strings of numbers. In early days of Internet, all host names and associated IP addresses were recorded in a single file called hosts.txt, which was maintained by NIC (Network Information Center) in the US. Every night all the hosts attached to the Internet would obtain a copy of this file to refresh their domain name entries. As the internet grew, it became impossible to keep it up-to-date. To solve this problem, Domain Name System was developed. This DNS is consulted whenever any message is sent to any other computer on the Internet. It simply gives the mapping of domain names to IP addresses. Domain name servers translate domain names to IP addresses. That sounds like a simple task, and it would be able to handle four things:

- There are billions of IP addresses currently in use, and most machines have a human-readable name as well.
- There are many billions of DNS requests made every day. A single person can easily make a hundred or more DNS requests a day, and there are hundreds of millions of people and machines using the Internet daily.
- Domain names and IP addresses change daily.
- New domain names get created daily.

The DNS Name Space

The problem of mapping host names to IP addresses is very difficult given the numbers of machines on the Internet.

The postal system faces a similar challenge, which it has dealt by requiring the sender to specify the country, state, city, street name, house no. Internet uses the same principle. The Internet is divided into top-level domains, such having several hosts underneath. Again, each domain is divided into sub-domains, which can be further

classified into sub-domains. This creates a tree-like structure which is shown in *Figure* 19.

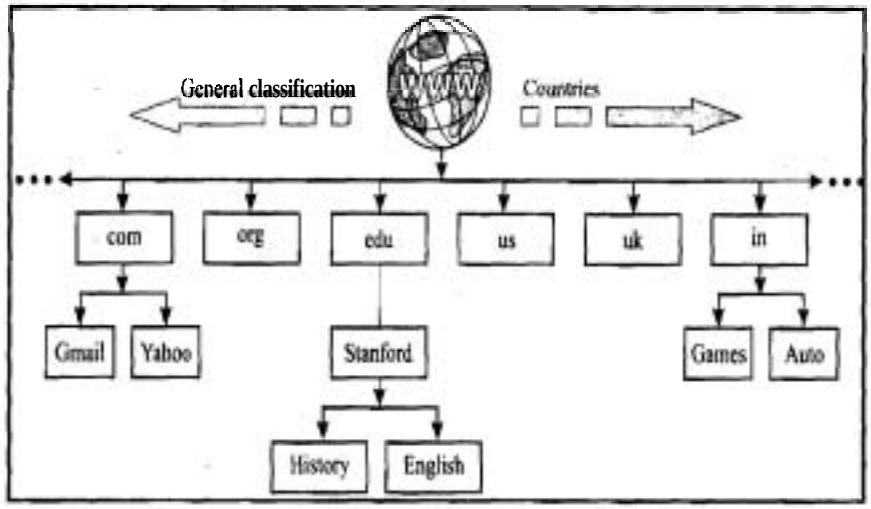

Figure 19: Domain Name space

The topmost domains are classified as: ***generic,*** and ***countries.*** The generic domains ***are*** classified into com (commercial), gov (the US federal government), edu (educational), org (non-profit organizations), net (network, etc. The country domains specify one entry for each country these clarifications are also shown in ***Figure*** 20. For example, uk (United Kingdom), in (India) and so on.

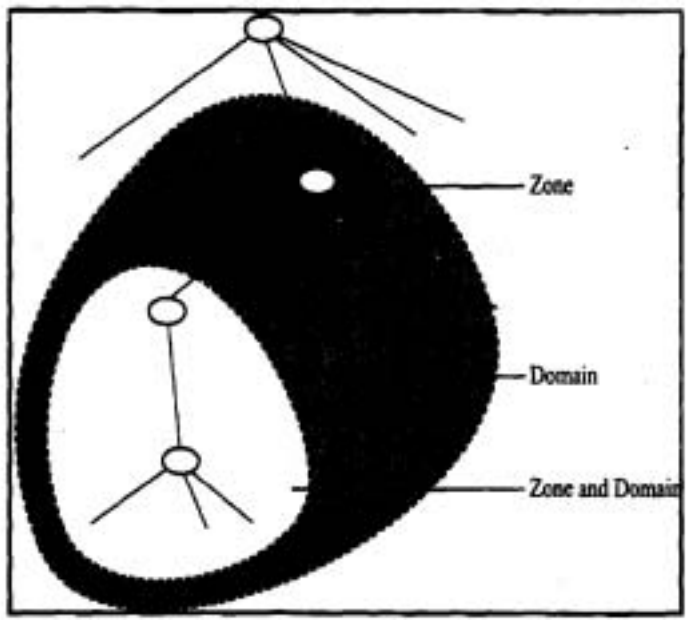

Figure 20: Zones and domains in DNS

For registering a name called ***bajaj under the category auto, which is within*** India(in), the full path will be bajaj.auto.in

Each domain is fully qualified by the path upward from it to the top most root. The names within a full path are separated by dot. Remember domain names are case-insensitive.

DNS Server

The information is contained in the domain name space. It must be stored at some place, but it would be inefficient and unreliable to store all the information at one place. The solution is to distribute the information among many computers called DNS servers. Thus, every domain has a domain name server.

Primary and Secondary DNS servers

DNS defines two types of servers: Primary server and secondary server.

A *primary server* is responsible for creating, maintaining and updating the zone file. It stores this zone file on its local disk.

A secondary server neither creates nor updates the zone file but stores the latest zone file from the primary server, The idea is to have redundancy so that if the primary server fails, the secondary server can continue serving the clients. A server can be primary server for one zone and secondary server for another zone.

1.7.3 How does the DNS server works?

DNS is designed to be a client server application. A host that needs to map a name to an address is called a revolver.

The DNS works in a manner similar a telephone directory inquiry service. You dial up the inquiry service and ask for a person's telephone number by providing his name. If the person is local, the inquiry service immediately comes up with the answer. If s/he stays in another state, the directory service directs your call to that state's telephone directory service, or asks you to call them. This is very similar to the way. The DNS server works. In this case, you specify the domain name and ask for its corresponding address.

All day, a DNS server does two things:

- Accepting requests from programs for mapping domain names into IP addresses (revolvers).
- Accepting requests from other DNS servers to map domain names into IP addresses.

When such a request comes, a DNS server has the following options:

- It can supply the IP address because it knows it from its zone file.
- It can contact another DNS server and try to locate the IP address for the name requested. Every DNS server has an entry called alternate DNS server, which is the DNS server it should get in touch with for unresolved domains. If this server is in authority, it responds, otherwise sends the query to another server. When the query is finally resolved, it travels back until it finally reaches the resolver. This type of resolution is called recursive resolution which is shown in ***Figure*** 21 given below.

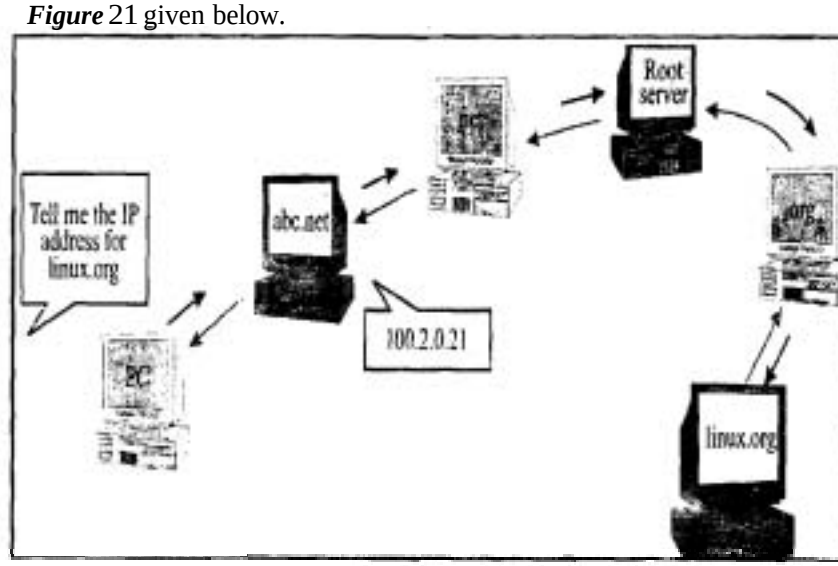

Figure 21: Recursive solution

It simply says i do not know the IP address for the requested domain but here is the IP address for a name server which knows more than me". The client is responsible for

repeating the query to this second server. If this newly addressed server can resolve the query, it answers the query, and otherwise, it returns address of the alternate server. This process is called iterative resolution which is shown below in Figure 22.

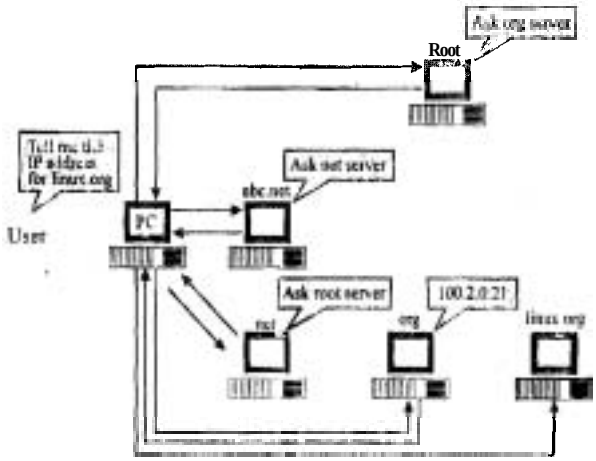

Figure 22: Iterative solution

It can return an error message because the requested domain name is invalid or does not exist.

Caching

Each time a server receives a query for the name that is not in its domain, it sends the query to the alternate server. It stores the information received from the response in its cache memory before sending it to the client. If any client asks for the same mapping, it an check its cache memory and resolve the problem. Thus caching speeds up the resolu ion process to avoid sending outdated mapping to clients. The server adds an information called time-to live (TTL) to the cache entry. It define this time in seconds that the receiving server can cache the information. After that the entry is invalid and any query must be sent to the alternate server. DNS requires that each server keeps a TTL counter for each mapping it caches.

1.7.4 Simple Network Management Protocol (SNMP)

Since it was developed in 1988, the Simple Network Management Protocol has become the de facto standard for internetwork management. Because it is a simple solution, requiring little code to implement, vendors can easily build SNMP agents to their products. SNMP is extensible, allowing vendors to easily add network management functions to their existing products. SNMP also separates the management architecture from the architecture of the hardware devices, which broadens the base of multivendor support. Perhaps most important, unlike other so-called standards, SNMP is not a mere paper specification, but an implementation that is widely available today.

SNMP is based on the manager/agent model consisting of a manager, an agent, a database of management information, managed objects and the network protocol. The manager provides the interface between the human network manager and the management system. The agent provides the interface between the manager and the physical device(s) being managed (see the illustration abovej.

A typical agent usually:

- Implements full SNMP protocol.
- Stores and retrieves management data as defined by the Management Information Base.
- Can asynchronously signal an event to the manager
- C'an be a proxy for some non-SNMP manageable network node.

A typical manager usually:

- Implemented as a Network Management Station (the NMS)
- Implements full SNMP Protocol Able to
- Query agents
- Get responses from agents
- Set's variables in agents
- Acknowledge's asynchronous events from agents.

The manager and agent use a **Management Information Base (MIB)** and a relatively small set of commands to exchange information. The MIB is organised in a tree structure with individual variables, such as point status or description, being represented as leaves on the branches. A long numeric tag or **object identifier** (OID) is used to distinguish each variable uniquely in the MIB and in SNMP messages.

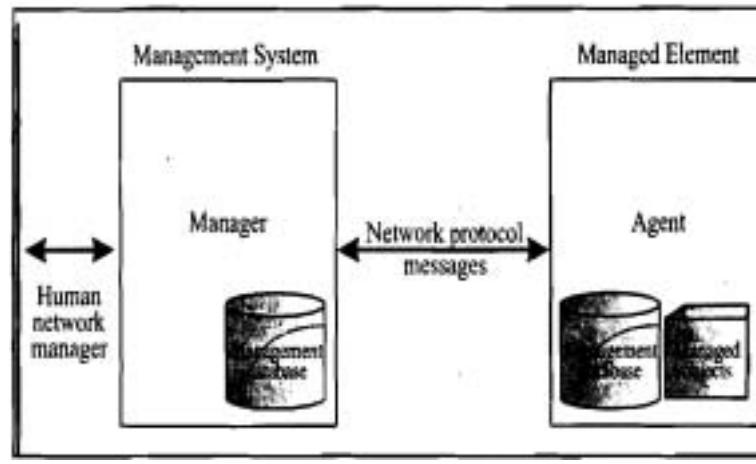

Figure 23; SNMP Protocol

SNMP uses five basic messages (GET, GET-NEXT, GET-RESPONSE, SET, and TRAP) to communicate between the manager and the agent. The GET and GET-NEXT messages allow the manager to request information for a specific variable. The agent, upon receiving a GET or GET-NEXT message, will issue a GET-RESPONSE message to the manager with either the information requested or an error indication as to why the request cannot be processed. A SET message allows the manager to request a change to be made to the value of a specific variable in the case of an alarm remote that will operate a relay. The agent will then respond with a GET-RESPONSE message indicating the change has been made or an error indication as to why the change cannot be made. The TRAP message allows the agent to spontaneously inform the manager of an 'important' event.

As you can see, most of the messages (GET, GET-NEXT, and SET) are only issued by the SNMP manager. Because the TRAP message is the only message capable of being initiated by an agent, it is the message used by DPS Remote Telemetry Units (RTUs) to report alarms. This notifies the SNMP manager as soon as an alarm condition occurs, instead of waiting for the SNMP manager to ask.

The small number of commands used is only one of the reasons SNMP is "simple." The other simplifying factor is its reliance on an unsupervised or connectionless communication link. This simplicity has led directly to its widespread use, specifically in the Internet Network Management Framework. Within this framework, it is considered 'robust' because of the independence of the managers from the agents, e.g. if an agent fails, the manager will continue to function, or vice versa. The unsupervised communication link does however create some interesting issues for network alarm monitoring.

Security levels with basic SNMP

Different Security levels (like authentication and authorization) are implemented in SNMP, let find out what we meant by authentication and authorization.

Authentication

Trivial authentication based on plain text community name is exchanged in SNMP messages. Authentication is based on the assumption that the message is not tampered with or interrogated.

Authorization

Once community name is validated then agent or manager checks to see if sending address is permitted or has the rights for the requested operation. "View" or "Cut" of the objects together with permitted access rights is then derived for that pair (community name, sending address).

Underlying Communication Protocols

SNMP assumes that the communication path is a connectionless communication subnetwork. In other words, no prearranged communication path is established prior to the transmission of data. As a result, SNMP makes no guarantees about the reliable delivery of the data. Although in practice most messages get through, and those that don't can be retransmitted. The primary protocols that SNMP implements are the User Datagram Protocol (UDP) and the Internet Protocol (IP). SNMP also requires Data Link Layer protocols such as Ethernet or TokenRing to implement the communication channel from the management to the managed agent. The connectionless nature of SNMP leaves the recovery and error detection up to the NMS (Network Management Station) and even up to the agent. However, keep in mind that SNMP is actually transport independent (although original design was connectionless transport function, which corresponds to the UDP protocol) and can be implemented on other transports as well:

1.7.5 Remote Login: TELNET

The main task of the Internet and its TCP/IP protocols is to provide services for users. For example, a user may want to run an application at the remote site and create results that can be transferred to the local site. One way to satisfy this demand is to create client-server applications program for each such desired service. Though, file transfer programs (FTP), email (SMTP), etc. are available, but it would be impossible to write a client-server application for each demand. The better solution would be a general purpose client-server program that lets a user access any application program on a ,remoteuser, or allow a user to log on to a remote computer. TELNET (TERminal NETwork) allows remote login services.

TELENT has two parts: client and server. Once a user using the services of a TELNET client connects to the remote TELNET server the keystrokes type by the user on the client are sent to the remote server to be interpreted upon to give the impression that the user is working on the remote computer.

Local login

When a user logs on to a local time sharing system, it is called local login. As user types at a terminal, the keystrokes are accepted by the terminal driver and passed to the operating system. The operating system interprets the combination of characters, and invokes the desired application. This mechanism is not simple as the operating system may associate special meanings to special characters. For example, Ctrl+z means suspend in UNIX. This does not create problem in local login because the terminal emulator and the driver know the exact meaning of characters.

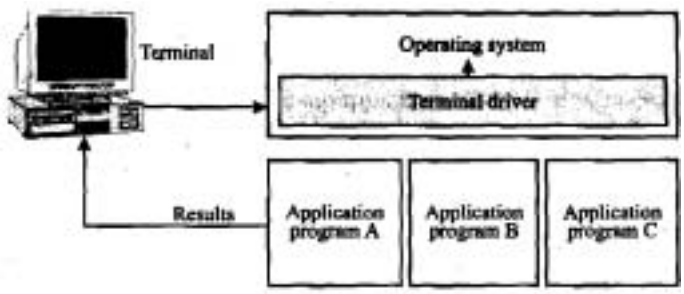

Figure 24: Local Login

Remote login

When a user logs on to a access an application on a remote computer, the users need to perform remote login.

Here the telnet client and server come into use. The user sends the keystrokes to the terminal driver where the local operating system accepts the characters but does not interpret them. The characters are sent to the TELNET client, which transform the characters to a universal character set called Network Virtual Terminal (NVT) and delivers them to the local TCPIIP stack. This is necessary because telnet is a general purpose application, and was designed to work between any terminal and any host. Thus, the client maps the terminal type to NVT. At the other end, server maps the NVT on to the actual terminal type the server is serving.

The commands or text, in NVT format travels through the Internet, and arrive at the TCP/IP stack of the remote machine. The characters are delivered to the local operating system, and are passed to the TELNET server, which performing mapping of NVT characters. However, they can be directly passed to the operating system, because it is designed to receive characters from terminal driver. The solution is to add a pseudo terminal driver, which pretends that the characters are coming from the terminal and not from the client. The operating system then passes the characters to the appropriate application.

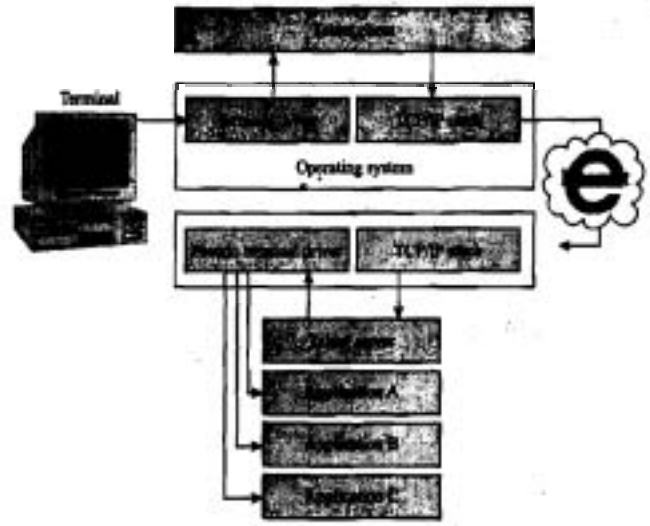

Figure 25: Remote Login

Communication in TELNET

Technically, TELNET server is quite complicated. It has to handle many clients at thq same time, and respond in real time. To handle this issue, TELNET server uses the principle of delegation. Whenever there a new client request for a TELNET connection, it creates a child process and lets the child handle the particular client.

TCP uses only one TCP connection. The same TCP connection is used to transfer both the control and data characters. How does the TELNET then distinguish between control and data characters? For this, it mandates that each sequence of control characters must be preceded by a special control character called Interpret As Command (IAC).

1.7.6 World Wide Web: HTTP

The WWW project was initiated by CERN to create a system to handle, distributed resources necessary for scientific research. Apart from email, the most popular applications running on Internet is the World Wide Web (WWW). It is so popular that people confuse it with Internet.

However, it is just an application such as email, FTP that uses TCP/IP. Many companies have internet websites, which is a collection s f web pages stored on a web

server. The server has web server software running on it. The function of web server is to store web pages and transmit them to a client computer when requested to do so.

In your company's website, you can display the information about the products, employees and policies. Sites can be dedicated to specialised tasks such as displaying news, share prices, sports etc. WWW consists of thousands of such websites of individuals and companies giving huge details about people, companies, events, news etc. WWW is an online repository of information that users can view using a program called web browser.

Architecture

The WWW today is a distributed client-server service. The client using a browser can access a service using a server. The service provider is distributed over many locations called sites.

Web server

A web server is a program running on a server computer, additionally, it consists of the web site, consisting of many web pages. It is simply a file written in HTML (Hlypertext Markup Language). It can consist of text, graphics, sound, video, animation. Every web site has a server process that passively listens for TCP connection requests at port 80. After the connection is established, the client sends one request and the server sends one response. The request – response model is governed by Hyper Text Transfer Protocol **(HTTP).**

Web browser

A web browser acts as a client in WWW interaction. Using this program, user can request for a web page on a web server. The browser then interprets and displays the document. A variety of vendors offer commercial browsers. A browser usually consists of three parts: a controller, client protocol and interpreters. The controller receives the input from the keyboard or the mouse and uses client programs to access the document. After the document has been accessed, the controller uses one of the interpreters to display the document on screen. The client protocol can be one of the protocols described previously like FTP, **TELNET,** or HTTP.

Uniform Resource Locator (URL)

A client that wants to access a web page needs to specify the address. The Uniform Resource locator is the standard for specifying any kind of information on the Internet. It has four things: protocol, host computer, port, and path.

Anatomy of URL

http://www.ignou.ac.in/80/index

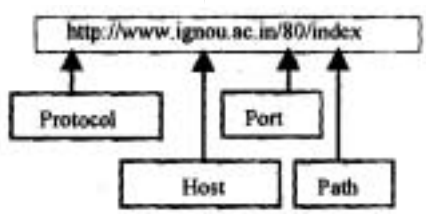

Figure 26: Components of URL

- ***The protocol*** is the client-server program used to retrieve the document. Most common is HTTP.
- ***Host*** is the name of the computer on which the information resides.
- **Port** number is transport address of the client or server program on a web site.
- ***Path*** is the pathname of the file on the website, and can consists of slashes.

HTTP

Hyper Text Transfer Protocol (HTTP) is a used mainly to transfer data on World Wide Web. The commands from the client are embedded in a request message .The

contents of the request message are embedded in a response message. HTTP uses the services of **TCP** at port 80.

HTTP is a stateless protocol since each transaction is independent of the previous **transaction.** The TCP connection **between** the client and the server is established for every page. **It does** not **remember** anything about **the** previous request. Keeping HTTP stateless was aimed at making the Web simple.

Sample **HTTP** request and response transaction is shown below:

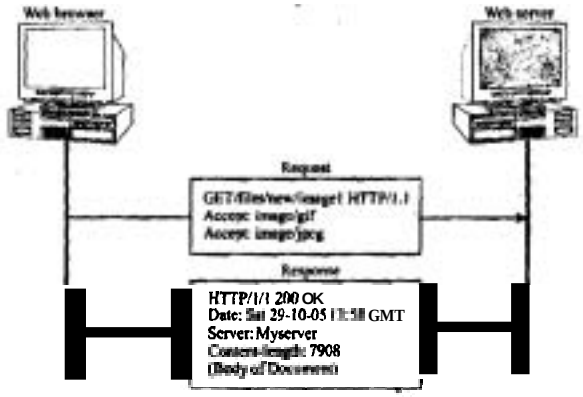

Figure 27: HTTP request and response

The **GET** command **requests** the web server at www.myserver.com for a image file **image1** .The **HTTp/1.0** indicates that the browser uses the 1.0 version of the HTTP protocol.

The first line of response indicates that the server is also using HTTP version 1.0 .the return code of 200 indicates that the server processed the request successfully. Request message can be of following types.

Method	Action
GET	Requests a document from the server
HEAD	Request information about the document but not the document itself; i.e. head of the HTML page
POST	Sends some information from client to server. It appends the data to the existing document.
PUT	Sends a document from to server. It replaces the existing document.
TRACE	Echoes the incoming request
CONNECT	Reserved
OPTION	Enquire about available options

Locating Information on the Internet

The amount of information available on the Internet is mind-boggling. However, finding the right information is usually very cumbersome. For enabling people to search infomation efficiently, **search** engines or used.

The search engines may continuously crawl through the Web Pages on the Internet and gather infomation about them to facilitate search for users in those Web pages.

Check Your Progress 3

1) **Explain** the concept of recursive and iterative resolution in of DNS?

.....

.....

.....

2) How does FTP differ from other application layer protocols?

.....

.....

.....

1.8 NETWORKING EXAMPLE: PUTTING IT ALTOGETHER

After seeing how communication takes place-using computers, we describe four scenarios to give a clear picture of it

Example 1: In the first example, there are just two computers, which are not part of any network as shown below in ***Figure*** 28.

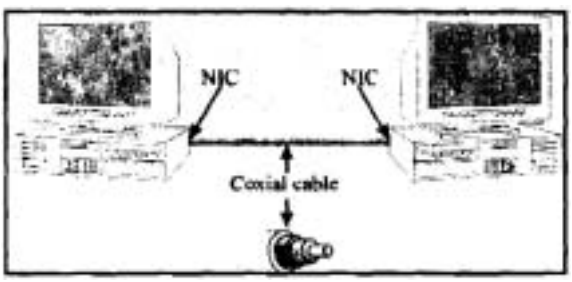

Figure 28: Two computers connectcd with Coaxer Cabk

In order to have communication, both computers should have Network Interface card and TCP/IP software running over it. They can connect using coaxial cable, and can choose any IP address, as they are not part of any network. The link is very fast as there is no sharing of bandwidth.

Example 2:

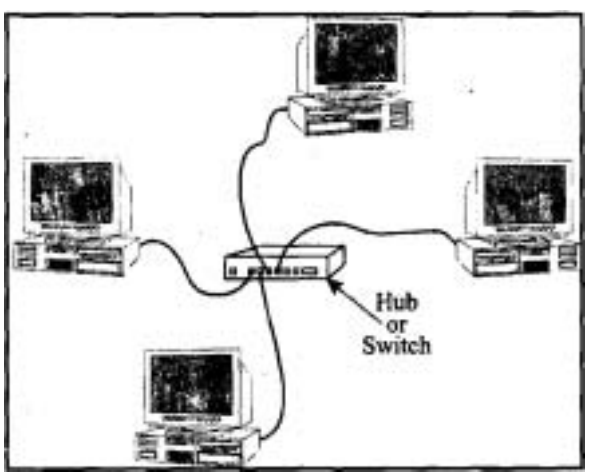

Figure 29: Computers connected with Hubs

A number of computers are connected to each other using a hub or a switch device *as* shown in ***Figure*** 29. Now the computers require uniform addressing, a network interface card and TCP/IP software they can connect to switch or hub using any physical medium. At a time only two computers can communicate. The packets from a computer are first transferred to the switch, which sends it to the destined computer.

Example 3:

A number of small networks like the ones in a computer Lab connect to each other, using routers *as* given in ***Figure*** 30. A computer wanting to communicate to another computer first checks if the destination computer is on the same network. If so, it uses ARP to know the IP address of the destination and forwards the packet to it. However,

if it is not on the same network, the packet is forwarded to the router connected to the network. A router is a multihome device, which is a part of more than one network. If the destination computer is connected to the router, it is delivered, otherwise it is forwarded to another router which is expected to know the way to destination. The path followed in order to toute the packets is determined by the routers which employ criteria like shortest path, minimum delay, minimum cost to decide the next router to which the packet should be fowared. The links and thus the bandwidth is shared among the computers.

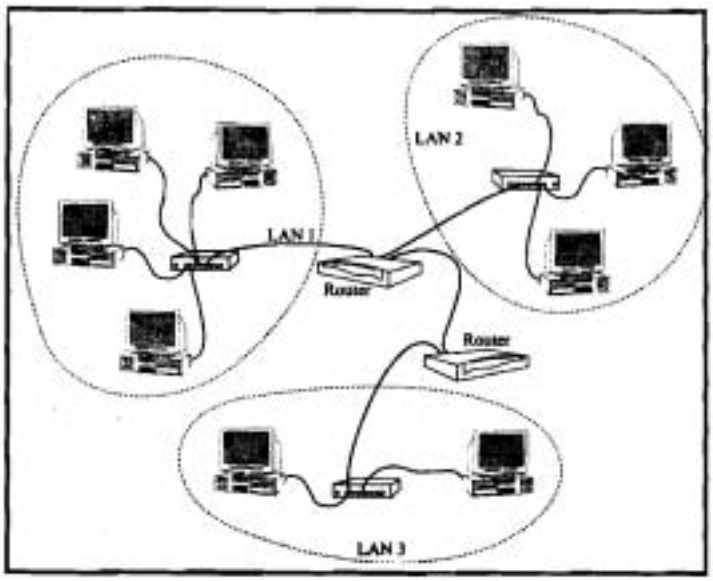

Figure 30: LANS connected with Routers

Example 4:

The fourth example is of the Internet given below in Figure 31, where a number of networks are connected to each other. These networks differ from each other in a number of ways like packet format, underlying hardware, and the protocols used. The scenario differs from the third one in that a gateway is required. A gateway can forward packets across different networks that may also use different protocols. Thus a gateway not only has the ability of translating between different packets formats, but also different protocols. Communication in the Internet is made possible by use of routers which forward packets between networks using the same protocol, and gateways connect large, incompatible networks.

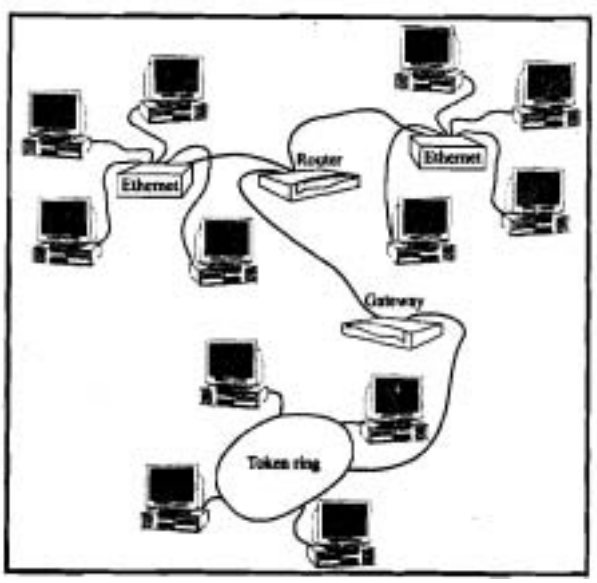

Figure 31: Different network connected with gateway and routers

1.9 SUMMARY

This unit is complete overview of TCPAP protocols. Till now we have studied that the Internet and TCPIIP came into being at around the same time. Some standards need to be flowed at the hardware and software level in order to connect incompatible networks. TCP is the software standard for the Internet. It is organised as a stack of layers. Each layer performs has interface with adjacent layers and certain functionalities. IP is connectionless, unreliable, and best effort packet delivery meclianism. It provides node-to node communication ARP, RAW, ICMP are other Internet layer protocols. TCP is reliable, connection-oriented protocol, and uses the services of IP. UDP is another transport layer protocol, which is unreliable, connectionless. It is much simpler than TCP and used in applications, which do not require the reliability but needs faster delivery. Together they provide end-to-end delivery of packets. Application layer protocols provide users with mechanisms to use internet for file transfer, (FTP) connection to remote hosts (TELNET), viewing web pages (WWW), manage network resources (SNMP) and lot of other functionalities. In the end, scenarios for typical communication between two computers are discussed. The next unit of this course covers about the Internet Protocol and its role in TCPAP.

1.10 SOLUTIONS/ANSWERS

Check Your Progress 1

1) The TCPIIP model is made up of four layers: interface layer, network, transport, and application. The first layer of TCPIIP (Application layer) is similar to the first three layers (Application, Presentation and Session layer) of the OSI model. The services of transport layers of both the models are similar. Further, the services of network layers in both models are also similar, while some time network layer is alsb known as Internet layer. The last layer of TCPIIP is Interface layer, which includes the services of data link layer and physical layer of OSI model. In OSI model, each layer takes the services of the lower layer. Whereas the layers of TCPAP protocol suite contain relatively independent protocols.

2) The TCPAP model is made up of four layers: interface, network, transport, and application. Layers of TCPAP Protocol Suite are explained below in detail:

i) Interface Layers

The physical layer deals with the hardware level, voltages. The data link layer deals with media access and control strategies, frame format etc. At this level, TCPIIP does not define any protocol. It supports all standards and protocols.

ii) Internet Layer or Network Layer

The Internet layer is an important layer in the protocol suite. At this layer. TCPAP supports Internetworking Protocol (IP). IP is a host-to host protocol. This layer is responsible for the format of data-grams as defined by IP, and routing a datagram or packet to the next hop, but is not responsible for the accurate and timely delivery of datagrams to the destination in proper sequence. IP allows raw transmission functions allovring user to add functionalities necessary for given application. An ensuring maximum efficiency, TCPIIP supports four other protocols: ARP, RAW, ICMP, IGMP.

iii) Transport Layer

At this layer, TCPIIP supports two protocols: TCP, UDP, IP is host-to-host protocol, which can deliver the packet from one physical device to another physical device. TCP.,are UDP, are transport level protocols, responsible for delivering a packet from one process on a device to another process on the other device. User Datgram Protocol (UDP)

It is simpler of the two protocols. It does not provide reliability. It is, therefore faster, and using for applications in which delay is intolerable (in case of audio and video) Transmission Control Protocol (TCP).

TCP is reliable, connection oriented protocol. By connection oriented, we mean that a connection must be established between both ends before either can transmit data. It ensures that communication is error-free and in sequence.

iv) **Application Layer**

As said earlier, it is closer to combined session, presentation, and application layer of OSI model. It allows user to run various applications on Internet. These applications are File Transfer Protocol (FTP), remote login (TELNET), email (SMTP), WWW (HTTP). The session layer of OSI model is almost dropped in TCP/IP.

Check Your Progress 2

1) Following are functions performed at the Internet layer:
   - Define the datagram, which is the basic unit of transmission in the Internet.
   - Define the Internet addressing scheme.
   - Move data between the Network Access Layer and the Host-to-Host Transport Layer.
   - Route datagrams to remote hosts.
   - Fragment and reassemble datagrams.
2) When datagrams arrive too quickly for processing, the destination host or an intermediate gateway sends an ICMP source quench message back to the sender. This message instructs the source to stop sending datagrams temporarily.

Check Your Progress 3

1) DNS server does two things one is accepting requests fiom programs for mapping domain names into IP addresses (revolvers) and another is accepting requests fiom other DNS servers to map domain names into IP addresses. When such a request comes, a DNS server has the following options.

i) It can supply the IP address because it knows it fiom its zone file.

ii) It can contact another DNS server and try to locate the IP address for the name requested. Every DNS server has an entry called alternate DNS server, which is the DNS server it should get in touch with for unresolved domains. If this server is in authority, it responds, otherwise sends the query to another server. When the query is finally resolved, it travels back until it finally reaches the resolver. This type of resolution is called ***recursive resolution.***

2) It simply says, I do not know the IP address for the requested domain but here is the IP address for a name server which knows more than me''. The client is responsible for repeating the query to this second server. If this newly addressed server *can* resolve the query, it answers the query, and otherwise, it returns address of the alternate server. This process is called ***iterative resolution.***

3) It *can* return an error message because the requested domain name is invalid or does not exist.

1.11 FURTHER READINGS

1) Achyut S Godbole, Web Technoligies, TATA McGrawHill, 2003.

2) Berhouz Forouzan, *TCP/IP* Protocol Suite, 3rd edition, TATA McGraw Hill, 2006.

3) http://www.tcpipguide.com

4) http://www.cisco.com