GCA S3 04

Data Communication and Computer Networks

SEMESTER III

BACHELOR OF COMPUTER APPLICATION

Block - 2



KRISHNA KANTA HANDIQUI STATE OPEN UNIVERSITY

Subject Expert

Prof. Anjana Kakati Mahanta, Gauhati University
Prof.(Retd.) Pranhari Talukdar, Gauhati University
Dr. Jyotiprakash Goswami, Assam Engineering College

Course Coordinator

Dr. Sanjib Kr. Kalita, KKHSOU
Dr. Tapashi Kashyap Das, KKHSOU
Sruti Sruba Bharali, KKHSOU

SLM Preparation Team

UNITS	CONTRIBUTORS
8	Swapnanil Gogoi, IDOL, GU
9	Barnali Gogoi, AEC
10 & 11	Ishita Chakraborty, Royal Global University
12	Pritam Medhi
13 & 14	Dr. Pranab Das, Don Bosco University
15	Dr. Chakradhar Das

Editorial Team

Content : Prof. Kandarpa Sarma, GU
Language : Prof.(Retd) Robin Goswami, Cotton College

Structure, Format & Graphics : Dr. Sanjib Kr. Kalita, KKHSOU

June 2018

This Self Learning Material (SLM) of the Krishna Kanta Handiqui State Open University is made available under a Creative Commons Attribution-Non Commercial-Share Alike 4.0 License (international): http://creativecommons.org/licenses/by-nc-sa/4.0/

Printed and published by Registrar on behalf of the Krishna Kanta Handiqui State Open University.

Headquarters : Patgaon, Rani Gate, Guwahati - 781017
City Office : Housefed Complex, Dispur, Guwahati-781 006; Web: www.kkhsou.in

he University acknowledges with thanks the financial support provided by the **Distance Education Bureau**, **UGC** for the preparation of this study material.

BLOCK INTRODUCTION

Block II contains 8 units. In block I we have already discussed 7 units. Unit 8 deals with the basic concept of Network layer including Routing, Network layer protocol like Internet protocols, IPV6 etc. Unit 9 introduces transport layer which includes Ports and Sockets, Datagram protocols, Transmission control protocols etc.The basics of session layers are discussed in Unit 10. Introduction to presentation layer like data representation, data conversion, translation, encryption compression are discussed in unit 11. Unit 12 deals with the Application layer like Client server model, Domain Name System, Simple Mail Transfer Protocol, File Transfer Protocol. The concept of Transmission Control Protocol, Internet Datagram Protocol are discussed in unit 13. Unit 14 discusses the details of Network Standard Protocol like FTP, Telnet, HTTP, email protocol, Browser, WWW, Ping, remote login etc. Unit 15 discusses network devices like Hub, Switch, Bridge, Router, Gateway, RS-232C etc. besides the concept of Network Address, Network Interface Card etc.

BACHELOR OF COMPUTER APPLICATION

Data Communication and Computer Networks

DETAILED SYLLABUS
BLOCK-2

Pages

Unit 8: Network Layer 5–38

Network Layer, Routing Protocols, Internet Protocol, IP Addresses, Sub-netting

Unit 9: Transport Layer 39–65

Process-To-Process delivery, User Datagram Protocol, Transmission Control Protocol: Characteristics, TCP Segment, Connection Establishment and termination

Unit 10: Session Layer 66–76

Interhost communication; Services- Dialog control, Token management, Synchronization; Protocols- Remote Procedure Call (RPC), NETBIOS.

Unit 11: Presentation Layer 77–96

Services - Data conversion, Translation, Encryption, Compression; Protocols - FTP, Telnet.

Unit 12: Application Layer 97–122

Client-Server Model: Client-Server model, Socket interface; A brief introduction to DNS, SMTP, FTP

Unit 13: Fundamentals of TCP and UDP 123–133

Connectionless and Connection Oriented Protocol, Transmission Control Protocol (TCP), User Datagram Protocol (UDP), IDP

Unit 14: Network Standard Protocol 134–147

FTP, Telnet, HTTP, Email Protocol, POP, SMTP, Browser, WWW, PING, Remote Login, Socket

Unit 15: Network Devices 148–168

Gateway, Bridge, Hub, Switch, RS-32C, Network Interface Card (NIC), Network Adapters, Components of NIC, Functions of NIC, Types of NIC; Ethernet: Basic Features, Types of Ethernet.

UNIT 8 : NETWORK LAYER

UNIT STRUCTURE

8.1 Learning Objectives
8.2 Introduction
8.3 Network layer
8.4 Addressing
8.5 Routing
  8.5.1 Unicast Routing
  8.5.2 Unicast Routing Protocols
  8.5.3 Multicast Routing
  8.5.4 Multicast Routing Protocols
8.6 Network Layer Protocols
  8.6.1 Internet Protocol (IP)
  8.6.2 IPv6
  8.6.3 Address Resolution Protocol (ARP)
  8.6.4 Internet Control Message Protocol (ICMP)
8.7 Let Us Sum Up
8.8 Further Reading
8.9 Answers to Check Your Progress
8.10 Model Questions

8.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- discuss about the basic concept of network layer
- describe about internetworks and network addressing
- describe the concept of routing and different routing protocols
- learn about different network layer protocols

8.2 INTRODUCTION

In this unit, we will learn about the basic concepts related to network layer. Different mechanisms of addressing and routing are discussed in detail in this unit. The different network layer protocols like address resolution protocol and internet control message protocol are discussed in this unit. In the next

unit we will exlpore the concepts related to transport layer.

8.3 NETWORK LAYER

The network layer provides the mechanism of transferring variable length data sequences from a source host on one network to a destination host on a different network. In this process, the quality of service requested by the transport layer is also maintained. The network layer adds a header that includes the logical addresses of the sender and receiver to the packet coming from the upper layer. The network layer performs network routing functions. Routers are used in this layer for sending data throughout the different networks. The network layer must know about the topology of the communication subnet and choose appropriate paths through it to perform its job properly. It must carefully choose routes to avoid overloading some of the communication lines and routers. When the source and destination are in different networks then different problems are occurred. It is up to the network layer to deal with these problems.

A collection of interconnected networks, permitting data to move freely among these large number of different networks and populations is called an internetwork or internet. So the people connected to one network can communicate with the people attached to a different network with the help of internetwork. The gateways are used to make the connection amongst different networks and to provide the necessary translation for hardware and software. In other words, internet is a collection of LANs connected by a WAN. Different organizations construct different parts of the internetwork and each organization maintains its own part.

The internetwork is a packet-switched network at the network layer. The internetwork uses universal addresses defined in the network layer to route packets from the source to the destination. Delivery of a packet can be accomplished by using either a connection-oriented or a connectionless network service. In connectionless service, the network layer protocol treats each packet independently. The packets in a message may or may not travel the same path to their destination. This type of service is used in the datagram approach to packet switching. The Internet has chosen this type of service at the network layer because the internet is made of different types of networks that it is impossible to create a connection from the source to the destination without knowing the nature of the networks in advance.

8.4 ADDRESSING

In network layer, the packet transmitted by the sending computer may travel through different LANs or WANs in the way to the destination computer. Now, in this part of communication, a global addressing scheme is used which is called the logical addressing. The term IP address is used to mean a logical address in the network layer of the TCP/IP protocol suite. The internet addresses used in current times are 32 bits in length which are referred as IPv4 (IP version 4) addresses or simply IP addresses. So in case of IPv4 address, the maximum 2^{32} addresses are possible.

Now a new version of internet address is designed due to the requirement of more than 2^{32} addresses. This new version is referred to as IPv6 (IP version 6). In this version, the internet uses 128-bit addresses which are called IPv6 addresses. So in case of IPv6 address, the maximum 2^{128} addresses are possible.

IPv4 Addresses:

An IPv4 address is a 32 bit address which uniquely and globally defines the connection of a device to the internet. So, two devices on the internet can never have the same IPv4 address at the same time and this addressing scheme must be accepted by any host which wants to be connected to the internet.

The length of an IPv4 address is 32-bit. Each bit of an IPv4 address can have two different values which are 0 or 1. So the address space of IPv4 is 2^{32} or 4,294,967,296.

To show an IPv4 address, two types of notations can be used which are binary notation and dotted-decimal notation.

In binary notation, the IPv4 address is displayed as 32 bits. For example:

00111000 10110011 00111011 00001100

In dotted-decimal notation, IPv4 addresses are written in decimal form with a decimal point separating the bytes. For example the dotted-decimal notation of the above address is given below:

56.179.59.12

In classful addressing, there are five classes of IPv4 addresses which are A, B, C, D and E. Each class occupies some part of the address space. If the address is given in binary notation, the class of the address can be found with the leftmost first few bits as follows:

- If the leftmost bit of the address is 0 then it is in class A. For example: 00000100 10001001 00010010 11101101 is class A address.

- If the leftmost two bits of the address are 10 then it is in class B. For example: 10100001 00010010 11100010 10111001 is a class B address.
- If the leftmost three bits of the address are 110 then it is in Class C. For example: 11000001 10001001 00001010 11111010 is a class C address.
- If the leftmost four bits of the address are 1110 then it is in class D. For example: 11101010 00010101 10111010 11111111 is a class D address.
- If the leftmost four bits of the address are 1111 then it is in class E. For example: 11111001 00010110 11101101 00011101 is a class E address.

Now, if the address is given in decimal-dotted notation, the decimal value of the first byte defines the class as follows:

- If the decimal value of the first byte of the address is in the range 0 to 127 then it is in class A. For example: 4.137.18.237 is a class A address.
- If the decimal value of the first byte of the address is in the range 128 to 191 then it is in class B. For example: 161.18.226.185 is a class B address.
- If the decimal value of the first byte of the address is in the range 192 to 223 then it is in class C. For example: 193.137.20.250 is a class C address.
- If the decimal value of the first byte of the address is in the range 224 to 239 then it is in class D. For example: 234.21.186.255 is a class D address.
- If the decimal value of the first byte of the address is in the range 240 to 255 then it is in class E. For example: 249.22.237.29 is a class E address.

Class A addresses were designed for large organizations with large number of attached hosts or routers. Class B addresses were designed for midsize organizations with tens of thousands attached hosts or routers. Class C addresses were designed for small organizations with a small number of attached hosts or routers. Class D addresses were designed for multicasting. The class E addresses were reserved for future use.

In clasful addressing, an IP address in class A, B or C divided into netid and hostid. In class A, one byte defines the netid and three bytes define the hostid. In class B, two bytes define the netid and two bytes define the hostid.

In class C, three bytes define the netid and one byte defines the hostid.
In classful addressing, mask or default mask is used for classes A, B and C. A mask is a 32-bit number in which the n leftmost bits are 1s and the 32-n rightmost bits are 0s. Class D and class E does not have any default mask. The masks for classes A, B, and C are given in the following table.

Class	Binary	Dotted-Decimal	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

In the given table, the last column shows the mask in the form /n where n can be 8, 16 or 24 in classful addressing. This notation is also called slash notation or Classless Interdomain Routing (CIDR) notation. This notation is used in classless addressing.

Due to the fast growing internet, depletion of the available addresses may occur in case of the classful addressing scheme. In recent times, the number of devices on the Internet is less than the 2^{32} address space but the availability of class A and class B addresses is decreasing and a class C block is too small for most midsize organizations.

To solve the drawback of classful addressing scheme, classless addressing was designed and implemented. This addressing scheme overcomes address depletion and gives more organization access to the internet. In this scheme there are no classes.

In classless addressing, when a device needs to be connected to the Internet then it is granted a block of addresses. The size of the block varies based on the nature and size of the device.
The internet authorities impose three restrictions on classless address blocks to simplify the handling of addresses which are:

1. The addresses in a block must be contiguous.
2. The number of addresses in a block must be a power of 2 like 2^1, 2^2, 2^4 etc.
3. The first address must be evenly divisible by the number of addresses.

In classless addressing the mask for a block can take any value from 0 to 32.

In IPv4 addressing, a block of addresses can be defined as x.y.z.t/n where x.y.z.t defines one of the addresses and the /n defines the mask.
Now the first address in the block can be found by setting the 32-n rightmost bits in the binary notation of the address to 0s and the Last address in the block can be found by setting the 32-n rightmost bits in the binary notation of the address to 1s.The number of addresses in the block is the difference between the last and first address. It can be found by using the formula 2^{32-n}.

Network Address Translation (NAT)

In near future, the address space of IPv4 will not be sufficient to accommodate all the devices in the internet because of the fast growth of internet. Now a solution to this problem is Network Address Translation (NAT). It is described in RFC 3022.
The basic concept of NAT is to assign each company a large set of addresses internally and one address or a small set of addresses externally. Each computer inside the company gets a unique IP address from the large set of addresses. When a data packet comes out from the company and goes to the ISP then an address translation takes place. In this scheme, three ranges of IP addresses have declared as private. These addresses can be used internally by different companies but packets containing these addresses can not appear on the internet itself. These three reserved ranges of addresses are given as follows:

```
10.0.0.0      to  10.255.255.255    (16,777,216 hosts)
172.16.0.0    to  172.31.255.255    (1,048,576 hosts)
192.168.0.0   to  192.168.255.255   (65,536 hosts)
```

The first range provides for 16,777,216 addresses .In general most companies choose this range of addresses. So using first range of addresses, every computer inside the company has a unique address of the form 10.x.y.z. When a packet comes out from the company network, it passes through a NAT box which converts the internal IP source address to the company's true IP address.

IPv6 Addresses

Internet Protocol version 6 (**IPv6**) is the latest version of internet addresses. IPv6 was developed by the Internet Engineering Task Force

(IETF).

IPv6 has an address space of 2^{128} addresses because it uses a 128-bit address. So the address space of IPv6 is much larger than IPv4 address space.

Hexadecimal Colon Notation: IPv6 specifies hexadecimal colon notation for its addresses where 128 bits is divided into eight sections and each 2 bytes in length. Two bytes in hexadecimal notation requires four hexadecimal digits. So an IPv6 address consists of 32 hexadecimal digits with every four digits separated by a colon. For example: EFED: 07E4: 0070: 0040: 00E0: D0FF: 0000: EEEE is an IPv6 address in hexadecimal colon notation.

In case of IPv6, the IP addresses are divided into several categories. A few leftmost bits in each IP address which are called the type prefix define the category of each address. The type prefix is designed such that no code is identical to the first part of any other code. So when an address is given then the type prefix can easily be determined. The following table shows the type prefix and its type of the address

Type Prefix	Type
0000 0000	Reserved
0000 0001	Unassigned
0000 001	ISO network addresses
0000 010	IPX(Novell) network addresses
0000 011	Unassigned
0000 1	Unassigned
0001	Reserved
001	Reserved
010	Provider-based unicast addresses
011	Unassigned
100	Geographic-based unicast addresses
101	Unassigned
110	Unassigned
1110	Unassigned
1111 0	Unassigned
1111 10	Unassigned
1111 110	Unassigned
1111 1110 0	Unassigned
1111 1110 10	Link local addresses
1111 1110 11	Site local addresses
1111 1111	Multicast addresses

The different categories of IPv6 addresses are discussed as follows:

Unicast Addresses:
A unicast address is used to define a single computer. The packet sent to a unicast address must be delivered to a specific computer. IPv6 defines two types of unicast addresses which are geographically based and provider based.

Multicast addresses:
Multicast addresses are used to define a group of hosts. A packet sent to a multicast address must be delivered to each member of the group.

Anycast Addresses:
IPv6 defines anycast addresses where each anycast address defines a group of nodes. Now a packet sent to an anycast address is delivered to only one of the members of the anycast group which is the nearest one with the shortest route. Anycast addresses can be assigned to all routers of an ISP that covers a large logical area in the Internet. No block is assigned for anycast addresses.

Reserved Addresses:
Reserved addresses start with eight 0s. There are some subcategories in this category which are unspecified address, loopback address, compatible address and mapped address.
In the unspecified address, all 128 bits are 0s. This address is used when a host does not know its own address and sends an inquiry to find its address.
In the loopback address, only the right most bit is 1 and all other 127 bits are 0s. This address is used by a host to test itself without going into the network.
In a compatible address, the right most 32 bits is an IPv4 address and all other 96 bits are 0s. These addresses are used when a host using IPv6 wants to send message to another host using IPv6 but the message needs to travel through a part of the network that is still using IPv4.
In a mapped address, all leftmost 80 bits are 0s and next 16 bits to these 80 bits are 1s. Here the rightmost 32 bits is an IPv4 address. These addresses are used when a host that has migrated to IPv6 wants to send a message to a host still using IPv4.

Local Addresses:

Local addresses are used when an organization wants to use IPv6 protocol without being connected to the global Internet. So in this case, no messages can be sent from outside the organization to the nodes using these addresses. There are two types of local addresses in IPv6 which are link local address and site local address.

A link local address is used in an isolated subnet and a site local address is used in an isolated site with several subnets.

8.5 ROUTING

The main function of the network layer is to transfer variable length data sequence from source machine to destination machine through the best possible path which is also called routing IP packets from the source machine to the destination machine. Now the algorithms called routing algorithms which are the part of network layer software responsible for deciding the routes and the data structures to transmit the incoming packets. If the subnet uses datagrams internally, this decision must be a new one for each time at every arrival of data packet because the best route may have changed since last time. If the subnet uses virtual circuits internally then routing decisions are made only when a new virtual circuit is being set up. This is also called as session routing because a route remains same for an entire user session.

The different goals of a routing algorithm are discussed as follows:

1. **Correctness:** The routing should be done in a proper way so that correctness can be maintained for sending the packets to their proper destination.
2. **Simplicity:** Simplicity should be maintained in the development of a routing algorithm so that the overhead is as low as possible.
3. **Robustness:** Once a major network becomes operative, it may be expected to run continuously for years without any failures. So the routing algorithms should be robust enough to handle hardware and software failures and should be able to cope with changes in the topology and traffic without requiring all jobs in all hosts to be aborted and the network rebooted every time when some router goes down.
4. **Stability:** The routing algorithms should be stable under all possible situations.

5. **Fairness:** Every node connected to the network should get a fair chance of transmitting their packets.
6. **Optimality:** The routing algorithms should be optimal in case of throughput and minimizing mean packet delays.

Routing algorithms can be divided into two major classes which are nonadaptive and adaptive algorithms.

Nonadaptive algorithms are static routing algorithms. Here the choice of the route to transmit IP packets from one node to another node is computed in advance and downloaded to the routers when the network is booted. It is not depended on the measurements of the current traffic and topology

On the other hand, in adaptive algorithms, the routing decisions are changed whenever there is a change in the topology and the network traffic. Now the different adaptive algorithms have differences amongst them in case of the following points:

(a) From where they get their information.

(b) When they change the routes

(c) When the load changes or when the topology changes.

(d) Type of metric used for optimization.

The Optimality Principle

The optimality principle is used in routing algorithms. It states that if router B is on the optimal path from router A to router C then the optimal path from B to C also falls along the same route.

So according to the optimality principle, the set of optimal routes from all sources to a particular destination form a tree rooted at that destination called a sink tree. A sink tree is not necessarily unique because other trees with the same path lengths may exist. The goal of all routing algorithms is to discover and use the sink trees for all routers.

Routing Protocols

Routing protocols are used to continuously update the routing tables that are consulted for forwarding and routing IP packets. A routing protocol is a combination of rules and procedures that allows routers to share whatever they know about the internet or their neighbourhood. Routing protocols are divided into categories which are unicast and multicast protocols.

8.5.1 Unicast Routing

A host or a router has a table called routing table which stores one

entry for each destination or a group of destinations host to route IP packets. A routing table can be either static or dynamic.

A static table is one with manual entries which are entered by the network administrator. It cannot be updated automatically when there is a change in the Internet. It is the responsibility of the administrator to update the table manually.

On the other hand, a dynamic routing table is updated automatically when there is a change in the internet. In recent times, dynamic routing tables are required in the internet for better performance.

In unicast routing, when a router receives a packet to route then it needs to

find the shortest path to the destination of the packet. The router uses its routing table to find the shortest path for that particular destination. Now the next-hop entry corresponding to the destination in the routing table is the start of the shortest path. The router has a shortest path tree to optimally reach all destinations. In unicast routing, each router needs only one shortest path tree to forward a packet.

a) Distance Vector Routing

Distance vector routing is a dynamic routing algorithm. This algorithm is also referred as the distributed Bellman-Ford routing algorithm or the Ford-Fulkerson algorithm. It was developed by Bellman in 1957 and Ford and Fulkerson in 1962.

In this algorithm, each router maintains a routing table which provide the best known distance to each destination and which path to use to reach there. Each entry in a routing table of a router contains two parts which are (a) the preferred outgoing line to use for the destination and (b) an estimate of the time or distance to the destination. So here, with the help of the routing tables, the route source host to destination host with minimum distance is calculated. Now the routing tables of the routers are automatically updated by exchanging information with the neighbours. Initially each router can know only the distance between itself and its immediate neighbours. Later, each router shares its routing table with its immediate neighbours periodically and when there is a change. By doing this, the routing tables are updated automatically and the path between any two routers with minimum distance can be found with the help of updated routing tables.

The Count-to-Infinity Problem

The count to infinity is a problem that may occur in distance vector routing. The count-to-infinity problem happens when a router tells another router that it has a path somewhere but there is no way for the second router to know that he is a part of the path. The count to infinity problem is caused by link failures that partition the network into two or more segments.

b) Link State Routing

Link state routing is also a dynamic routing algorithm. In this algorithm, each router must perform the following five steps:

1. Each router must discover its neighbours and learn their network addresses. It is done by sending a special HELLO packet from a router to all its neighbours connected to it. Now, the router on the other end needs to send back a reply telling who it is. These names must be globally unique.
2. Each router must have a reasonable estimate of the delay or cost to each of its neighbours. It can be done by sending a special ECHO packet to the neighbours of a router. Now, on receiving an ECHO packet by a router, it is required to send back immediately to the sending router. By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the delay. For better results, this test can be done several times and the average is used. In this method, the delays are assumed to be symmetric. Now here the load of the network traffic plays an important role. If the load is considered as a factor in this method then the round-trip timer must be started when the ECHO packet is queued and if the load is ignored then the timer should be started when the ECHO packet reaches the front of the queue.
3. After collecting the necessary information required to exchange, each router must construct a packet called link state packet, containing all these information. This packet contains the identity of the sender, a sequence number, age, a list of neighbours and the delay to each neighbour. Now, the most important point is when these packets should be constructed. In some cases, the link state packets are built at regular intervals. Another possibility is to build these packets when some significant event occurs like a neighbour

going down or coming back up again or changing its properties.

4. Each router must distribute its link state packet to all other routers. The routers getting the first packet will change their routes. Now the different routers may be using different versions of the topology so some problems like inconsistencies, loops, unreachable machines may occur. So, in this distribution process a distribution algorithm is used to distributing the link state packets reliably. The basic idea is to use flooding to distribute the packets. Here each packet contains a sequence number that is incremented for each new packet sent. When a new packet comes in, it is checked against the list of packets that have already arrived. If it is new one then it is forwarded on all lines except the one it comes in. Now, if it is a duplicate packet then it is discarded. If an incoming packet has a sequence number which is lower than the highest sequence of any packet that has arrived earlier then it is rejected because the router now has more recent data. Now this algorithm has some problems. The first problem is it may be possible that at some time the value of the sequence number will reach its maximum value. The solution for this problem is to use a 32 bit sequence number which will be large enough. The second problem is if a router crashes then it will lose track of its sequence number. If it starts again at 0 then the next packet will be rejected as a duplicate one. The third problem is, if a sequence number is corrupted. In this case it may be possible that some new packets will be rejected. For example if the actual sequence number of a new packet is 8 but it is corrupted to 640 then packets with sequence numbers from 9 to 640 will be rejected as obsolete because the current sequence number is thought to be 640. The solution to these problems is to include the age of each packet after the sequence number and decrement it once per second. When the age hits zero then the information from that router is discarded. The Age field is also decremented by each router during the initial flooding process to make sure no packet can get lost and live for an indefinite period of time. Some modifications can be done to

this algorithm. First modification is when a packet comes in to a router for flooding, it is not queued for transmission immediately. In this case, it is first put in a holding area to wait a short while. If another packet from the same source comes in before the first packet is transmitted, their sequence numbers are compared. If they are equal then the duplicate is discarded and if they are different then the older one is flooded. Second modification is all link state packets are acknowledged. It will help to handle errors occurred on the router to router lines. In this case when a line goes idle, the holding area is scanned in round-robin order to select a packet for acknowledgement to send.

5. Each router must compute the shortest path to every other router. In this step, Dijkstra's algorithm can be run locally to construct the shortest path to all possible destinations. The results of this algorithm can be installed in the routing tables. Now in case of this algorithm, for large subnets, the requirement of memory to store input data can be a problem because if a subnet has n routers and each router has k neighbours then the memory required to store the input data is proportional to k*n. The computation time can also be a problem in this algorithm.

8.5.2 Unicast Routing Protocols

Two unicast routing protocols are discussed as below.

a) Routing Information Protocol

The Routing Information Protocol (RIP) is an intradomain routing protocol used inside an autonomous system based on distance vector routing. RIP implements distance vector routing with some consideration:

1. An autonomous system has both routers and networks. Here the routers have routing tables but networks do not have any table.
2. The first column in routing tables defines a network address.
3. The metric used by RIP is the distance which is the number of networks to reach the destination and it is called hop count.
4. Any route in an autonomous system using RIP cannot have

more than 15 hops

5. The next-node column in a routing defines the address of the router to which the packet is to be sent to reach its destination.

b) Open Shortest Path First Protocol (OSPF)

The open shortest path protocol is an intradomain unicast routing protocol based on link state routing.

The OSPF divides an autonomous system into some areas to handle routing efficiently and in a timely manner. An area is a collection of networks, hosts, and routers of an autonomous system. All networks inside an area must be connected. Routers inside an area are required to distribute the routing information in the area. There are some special routers at the border of an area called area border routers which summarize the information about the area and pass it to other areas. There is a special area inside an autonomous system which is called the backbone and the routers inside the backbone are called the backbone routers. All other areas inside an autonomous system must be connected to the backbone.

When for some problem the connectivity between a backbone and an area is broken then the administrator must create a virtual link between routers to allow continuity of the functions of the backbone as the primary area.

The OSPF protocol allows the administrator to assign a metric to each

route. The metric is actually a cost which is based on a type of service such as minimum delay, maximum throughput etc. So a router can have multiple routing tables each based on a different type of service in OSPF protocol.

In OSPF protocol four types of links have been specified which are point to point, transient, stub and virtual.

A point to point link connects two routers without any other host or router in between.

A transient link is a network with several routers attached to it. The data can enter through any of the routers and exit the network through any router.

A stub link is a network that is connected to only one router. The data packets enter the network through this single router and exit

the network through this same router.

When the link between two routers is broken, the administrator may create a virtual link between them using a longer path which possibly passes through several routers.

8.5.3 Multicast Routing

In multicast routing, router receives multicast packet for routing to the destinations in more than one network. Now in this situation routing of a single multicast packet to each member of a group requires a shortest path tree. So if we there are n groups then n shortest path trees are required. In this case, the complexity of multicast routing is increased. Now to solve the problem, source-based trees and group-shared trees approaches are used.

- **Source-Based Tree:** In the source-based tree approach, each router needs to have one shortest path tree for each group.
- **Group Shared Tree:** In the group-shared tree approach, only one designated router, called the center core has m shortest path trees in its routing table. Here if a router receives a multicast packet then it encapsulates the packet in a unicast packet and sends it to the core router. The core router removes the multicast packet from the unicast packet and route the multicast packet with the help of its routing table.

a) Multicast Link State Routing

Multicast link state routing is an extension of unicast link state routing. This multicast routing algorithm uses a source based tree approach. In unicast link state routing, each node needs to advertise the state of its links. Now in multicast routing, state of a link specifies the groups which are active on that link. Here a node advertises every group which has any member on the link. Now the information about the group can be received by running Internet Group Management Protocol (IGMP).

When a router receives all the link state packets then it constructs n topologies from which *n* shortest path trees are made by using Dijkstra's algorithm. Here n is the number of groups. So each router has a routing table that specifies n shortest path trees.

Now the problem with this routing algorithm is that it requires more time and space to create and save the many shortest path trees than the unicast routing. The solution for this problem is that creation of the trees should be done only when it is required. When a router receives a packet with a multicast destination address then it use Dijkstra's algorithm to calculate the shortest path tree for that group. The result can be cached in case there are additional packets for that destination.

b) Multicast Distance Vector Routing

Multicast distance vector routing is the extension of unicast distance vector routing. In case of multicast routing, routers are not allowed to send its routing table to its neighbours. Here the idea is to create a table by using the information from the unicast distance vector tables.

Multicast distance vector routing uses source based trees but here the router never actually constructs a routing table. In this algorithm when a router receives a multicast packet then it forwards the packet as though it is consulting a routing table. After a packet is forwarded the table is destroyed.

The multicast distance vector algorithm uses a process to forward packets based on four decision making strategies discussed as follows:

1. **Flooding**. An another technique can be used to keep track of which packets have been flooded and avoid sending them second time. In this technique the source router put a sequence number in each packet it receives from its hosts. Here each router requires a list of sequence numbers per source router originating at that source which are already seen. If the sequence number of an incoming packet is on the list then it is discarded.
2. **Reverse Path Forwarding (RPF):** In RPF, a router forwards only the copy of a packet that has travelled the shortest path from the source to the router to prevent loops. Now RPF uses the unicast routing table to find this copy. The other copies of the packet are discarded.
3. **Reverse Path Broadcasting (RPB):** In RPF scheme, a network may receive two or more copies of a packet because

here forwarding is based on the source address. Now to eliminate this duplication, only one parent router for each network should be specified. So for this, a restriction can be made by which a network can receive a multicast packet from a particular source only through a designated parent router. This scheme is called reverse path broadcasting (RPB). Now the designated parent router can be the router with the shortest path to the source.

4. **Reverse Path Multicasting (RPM):** In RPB, broadcast of packets are done which is not efficient. So to increase efficiency, in RPM scheme the multicast packet is forwarded to only those networks that have active members for the particular group.

8.5.4 Multicast Routing Protocols

Two multicast routing protocols are discussed as below.

a) Multicast Open Shortest Path First (MOSPF) Protocol:
MOSPF protocol is an extension of the OSPF protocol. It uses multicast link state routing to create source-based trees. In this protocol, a tree is constructed that contains all the hosts belonging to a particular group. In this construction, the unicast address of the host is used. For efficiency, the router calculates the shortest path trees and the tree can be saved in cache memory for future use by the same source and group pair. MOSPF is a data-driven protocol. So an MOSPF router sees a datagram with a given source and group address for the first time, it constructs the Dijkstra shortest path tree.

b) Distance Vector Multicast Routing Protocol (DVMRP):
DVMRP is a multicast routing protocol which uses multicast distance vector routing. It is a source-based routing protocol based on RIP.

8.6 NETWORK LAYER PROTOCOLS

Four network layer protocols are discussed as follows:

8.6.1 Internet Protocol (IP)

The main network protocol in the internet model is the Internet Protocol (IP). The Internet Protocol version 4(IPv4) is used by the TCP/IP protocol.

IPv4 is an unreliable and connectionless protocol for a packet-switching network that uses the datagram approach. Here each datagram is handled independently and each datagram can follow a different route from source to the destination. So in this case, datagrams sent by the same source to the same destination may not arrive in the same order as the sending order. Some datagrams may also be lost or corrupted during transmission. The IPv4 does not provide any error control and flow control mechanism. It provides error detection on the header. IPv4 must be paired with a reliable protocol such as TCP to provide reliability.

A datagram in IPv4 is a variable-length packet consisting of two parts that are header and data. The header is 20 to 60 bytes in length and contains necessary information for routing and delivery. The different fields of header are given as follows:

Version (VER): It is a 4-bit field which defines the version of the IPv4 protocol. Currently the version is 4 but it may be replaced by version 6 in the future. This field tells the IPv4 software that the format of datagram is version 4. If the machine is using some other version of IPv4 then the datagram is discarded.

Header length (HLEN): It is a 4-bit field which defines the total length of the datagram header in 4-byte words. This field is required because the length of the header is variable. It is 20 to 60 bytes in length. Now when there are no options then the header length is 20 bytes and so the value of this field is 5 that mean 5 words with each word of 4 bytes in length. When the option field is at its maximum size then the value of this field is 15 that mean 15 words with each word of 4 bytes in length.

Services: It is an 8-bit field. Earlier this field is called service type and now it is called differentiated services.

Service Type: In this interpretation, the first 3 bits are called precedence bits. The next 4 bits are called type of service (TOS) bits and the last bit is not used.

Precedence bits is ranging from 0(000 in binary) to 7(111 in binary). The precedence bits provides the priority of the datagram in case of some problems like congestion. If a router is congested and needs to discard some datagrams then the datagrams with lowest precedence will be discarded first.

TOS bits is a 4-bit subfield with one and only one of the bits can have the value of 1 in each datagram. Each TOS bits has a special meaning given in the following table:

TOS Bits	Description
0000	Normal
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

Differentiated Services: In this interpretation, the first 6 bits make up the code point subfield and the last 2 bits are not used. Now the code point subfield can be used in two different ways given as follows:

a. If the 3 rightmost bits are 0s then the 3 leftmost bits are interpreted the same as the precedence bits in the service type interpretation.

b. If the 3 rightmost bits are not all 0s then the 6 bits defines 64 services based on the priority assignment by the Internet or local authorities. There are three categories of services. The first category contains 32 service types. The second and the third each contain 16 service types. The numbers for the first category are 0, 2, 4,…., 62 It is assigned by the Internet authorities (IETF). The numbers for the second category are 3, 7, 11, 15, and 63. It can be used by local authorities. The numbers for the third category are 1, 5, 9, 13, 17,…, 61. It is temporary and can be used for experimental purposes.

Total length: It is a 16 bit field which defines the total length of the IPv4 datagram in bytes.

Identification: It is a 16 bit field which identifies a datagram originating from the source host.

Flags: It is a 3 bit field. Here the first bit is reserved. The second bit is called “do not fragment bit”. If its value is 1 then the machine must

not fragment the datagram. If it cannot pass the datagram through any available physical network then it discards the datagram and sends an ICMP error message to the source host. If its value is 0 then the datagram can be fragmented if necessary. The third bit is called the “more fragment bit”. If its value is 1 then it means that the datagram is not the last fragment. If its value is 0 then it means that this is the last or only fragment.

Fragmentation offset: It is a 13 bit field which shows the relative position of the fragment with respect to the whole datagram. It is the offset of the data in the original datagram measured in units of 8 bytes.

Time-to-Live (TTL): It is an 8 bit field. It stores a value which specifies the number of router hops the packet is yet allowed to travel before it must be discarded or returned. When a source host sends a datagram, it stores a value in this field which is approximately 2 times the maximum number of routes between any two hosts. Each router that processes the datagram decrement the value of this field by 1 and now if this value is zero then the router discards the datagram.

This field is required because sometimes routing tables in the internet may be corrupted. As a result of this a datagram may travel between two or more routers infinitely without getting delivered to the destination host. So this field is used to limit the journey of the packet.

Protocol: It is an 8-bit field which defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP and IGMP. This field specifies the final destination protocol to which the IPv4 datagram is delivered. The following table shows the possible values in this field and corresponding higher level protocols:

Value	Protocol
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

Source Address: It is a 32-bit field which provides the IPv4 address of the source. This field must remain same during the time the IPv4

datagram travels from the source host to the destination host.
Destination address: It is a 32-bit field which provides the IPv4 address of the destination. This field must remain same during the time the IPv4 datagram travels from the source host to the destination.
IPv4 has some drawbacks and as a result of these, it will not be suitable for the fast growing Internet.

- Due to the fast growth of the internet, in near future the address space of IPv4 will not be sufficient to accommodate all the devices in the internet.
- The Internet must accommodate real-time audio and video transmission which requires minimum delay strategies and reservation of resources. But IPv4 does not provide these.
- IPv4 does not provide any encryption and authentication mechanism.

8.6.2 IPV6

IPv6 (Internetworking Protocol, version 6) is the new version of internet layer protocol for packet-switched internetworking and it provides end-to-end datagram transmission across multiple IP networks. IPv6 was first formally described in Internet standard document RFC 2460. IPv6 is also referred as IPng(Internetworking Protocol, next generation). In IPv6, the packet format and the length of the IP address were changed. Here related protocols like ICMP were also modified and other protocols in the network layer like ARP, RARP and IGMP were either deleted or included in the ICMPv6 protocol. Routing protocols like RIP and OSPF were also modified. The advantages of IPv6 are discussed as follows:

1. The length of IPv6 address is 128 bits and so the address space, 2^{128} is much larger than the address space, 2^{32} of IPv4.
2. IPv6 uses a new header format. In this format options are separated from the base header. In IPv6 the options are inserted when needed between the base header and the upper-layer data. So it simplifies and increases the speed of the routing process because most of the options do not need to be checked by routers.

3. IPv6 has new options to allow additional functionalities.
4. IPv6 allows the extension of the protocol if required by new technologies or applications.
5. In IPv6, new mechanism has been added to support traffic like real-time audio and video.
6. The encryption and authentication options are available in IPv6 which provide confidentiality and integrity of the packet.

Packet Format:

The IPv6 packet is consists of a base header followed by the payload. The payload consists of two parts which are optional extension headers and data from an upper layer. The length of the base header is 40 bytes and the payload contain up to 65535 bytes of information. The base header is consists of eight fields discussed as follows:

1. **Version:** It is a 4 bit field which defines the version number of the IP.
2. **Priority:** It is a 4 bit field which defines the priority of the packet with respect to traffic congestion. The following table shows different possible priority values and its meaning.

Priority	Meaning
0	No specific traffic
1	Background data
2	Unattended data traffic
3	Reserved
4	Attended bulk data traffic
5	Reserved
6	Interactive traffic
7	Control traffic

3. **Flow label**: It is a 24 bit field which is designed to provide special handling for a particular flow of data.
4. **Payload length:** It is a 16 bit field which defines the length of the IP datagram excluding the base header.
5. **Next header:** It is an 8 bit field which defines the header that follows the base header in the datagram.
6. **Hop limit:** It is an 8 bit field which stores the number of network segments on which the packet is allowed to travel before being discarded by a router. The hop limit is set by the sending host and is used to prevent packets from infinitely circulating on an IPv6 internetwork. At the time of forwarding an IPv6 packet, IPv6 routers

are required to decrease the Hop Limit by 1 and the IPv6 packet is discarded when the Hop Limit is 0.

7. **Source address:** It is a 128 bit field which stores a 128 bit internet address to identify the original source of the datagram.
8. **Destination address:** It is a 128 bit field which stores a 128 bit internet address used usually to identify the final destination of the datagram. But if source routing is used, this field contains the address of the next router.

Extension Headers:

There are six extension headers in IPv6 packet format discussed as follows:

Hop-by-Hop option: The hop-by-hop option is used when the source needs to send information to all routers which are visited by the datagram.

Source Routing: The source routing extension header is used to specify a list of intermediate nodes for a packet to travel on its path to its final destination.

Fragmentation: The fragmentation extension header is used in the purpose of fragmentation of datagram in IPv6. In IPv6, only the original source can fragment a datagram.

Authentication: The authentication extension header is used to validate the message sender and ensures the integrity of data.

Encrypted Security Payload: The encrypted security payload provides confidentiality.

Destination Option: The destination option is used when the source needs to send message to the destination only and this message is not permitted to access by the other routers.

8.6.3 Address Resolution Protocol (ARP)

The Address Resolution Protocol (ARP) is designed to create a mapping between physical and logical addresses. IP packets are encapsulated in a frame. IP packets use logical addresses and frame requires physical addresses. In internet, a packet starting from a source host may pass through different physical networks on the way to the destination host. Now the hosts and routers are recognized at the network level by their logical (IP) addresses and at the physical level, the hosts and routers are recognized by their physical addresses. A physical address is a local address and so it

must be unique in a local network but it may not unique universally. It is called a physical address because it is usually implemented in hardware.

Now delivery of a packet to a host or a router requires both logical and physical addressing. The mapping of a logical address to its corresponding physical address and vice versa can be done by using either static or dynamic mapping.

In static mapping, a table is created which associates a logical address with a physical address. This table is stored in each machine on the network. Static mapping table must be updated periodically because physical addresses may change in the following ways:

1. When a machine changes its NIC then a new physical address is created.
2. The physical address changes every time the computer is turned on in case of some LANs. For example: LocalTalk
3. In case of a mobile computer, physical address changes whenever it move from one physical network to another.

Now updating the static mapping table could degrade the network performance.

In case of dynamic mapping, ARP is used by a machine to find a logical address if it knows corresponding physical address and vice versa. When a host or a router has an IP datagram to send to another host or router, it has the logical (IP) address of the receiver. If the sender is the host then the logical (IP) address is obtained from the DNS and if the sender is a router then the logical address is obtained in a routing table. Now the sender requires the physical address of the receiver. So the host or the router sends an ARP query packet which includes the physical and IP addresses of the sender and the IP address of the receiver. This query is broadcast over the network and so every host or router on the network receives the ARP query packet. Now the actual recipient recognizes its IP address in the ARP query packet and sends back an ARP response packet. This ARP response packet contains the logical address (IP) and the physical address of the recipient. The ARP response packet is sent directly only to the sender of the ARP query packet by using the physical address received in the query packet.

The performance of ARP can be improved by using cache memory to store the ARP reply packets because a system normally sends

several packets to the same destination. A system that stores ARP reply packets in the cache memory always checks the cache memory to find the required mapping before sending an ARP request. The different fields of an ARP packet are as follows:

1. Hardware type is a 16-bit field defining the type of the network on which ARP is running. Each LAN has been specified by an integer based on its type.
2. Protocol type is a 16-bit field defining the protocol.
3. Hardware length is an 8-bit field defining the length of the physical address in bytes.
4. Protocol length is an 8-bit field defining the length of the logical address in bytes.
5. Operation is a 16-bit field defining the type of packet which can be either ARP request (1) or ARP reply (2).
6. Sender hardware address is a variable-length field defining the physical address of the sender.
7. the sender. In case of IP protocol, the length of this field is 4 bytes.
8. Target hardware address is a variable-length field defining the physical address of the receiver.
9. Target protocol address is a variable-length field defining the logical address of the receiver. In case of IPv4 protocol, the length of this field is 4 bytes.

Proxy ARP:

A proxy ARP is an ARP that acts on behalf of a set of hosts. When a router running a proxy ARP receives an ARP request looking for the IP address of one of these hosts then the router sends an ARP reply with its own physical address. After the router receives the actual IP packet, it sends the packet to the appropriate host or router. So using proxy ARP a network can be extended without the knowledge of the upstream router.

8.6.4 Internet Control Message Protocol (ICMP)

The IP protocol does not have any error-reporting or error-correcting mechanism and it also does not have any mechanism for host and management queries. Now the Internet Control Message Protocol

(ICMP) is designed to provide these mechanisms.

ICMP messages are divided into two categories which are error-reporting messages and query messages.

The error-reporting messages report problems which are occurred when a router or a destination host processes an IP packet.

A host or a network manager gets specific information from a router or another host from query messages which are occurred in pairs.

An ICMP message has two parts that are an 8-byte header and a variable-size data section. Now the first 4-bytes of the header are common for all messages. The first field of the header is ICMP type which defines the type of the message. The second field of the header is code field that specifies the reason for the particular message type and the last common field is the checksum field. The remaining part of the header is specific for each message type.

The data section in error message contains the information for finding the original packet where the error occurred. The data section in query messages contains information based on the type of the query.

ICMP uses the source IP address to send the error message to the source of the datagram. In general, ICMP reports five types of errors which are destination unreachable, source quench, time exceeded parameter problems and redirection.

When a router cannot route a datagram or a host cannot deliver a datagram then the datagram is discarded and the router or the host sends a destination-unreachable message created by either a router or the destination host, to the source host that has produced the datagram.

In case of IP protocol, there is no communication between the source host, the routers and the destination host. So in this situation, congestion can occur because IP does not have any flow control mechanism. If the datagrams are received much faster than they can be forwarded or processed, the buffer of a router or a host may overflow as its size is limited. In this case, the router or the host must discard some of the datagrams. Now the source-quench message in ICMP provides a kind of flow control to the IP. When a router or host discards a datagram due to congestion, it sends a source-quench message to the sender of the datagram. This message informs the source that the datagram has been discarded

and warns the source that it should slow down the sending process as there is congestion somewhere in the path from the source to the destination host.

Routers use routing tables to find the next router for sending packets. Now if there are errors in one or more routing tables then sometimes a packet may travel in a loop from one router to the next or a series of routers infinitely. So in this situation, each datagram contains a field called time to live to control it. When a datagram visits a router, the value of this field is decremented by 1. When the time-to-live value reaches 0, the router discards the datagram and a time-exceeded message must be sent by the router to the original source. Another time-exceeded message is also generated when all parts of a message does not arrive at the destination host within a certain time limit.

If a router or the destination host discovers an ambiguous or missing value in any part of the datagram then it discards the datagram and sends a parameter-problem message back to the source.

In IP, both routers and hosts require a routing table to find the address of the router or the next router. Routers update the routing tables constantly but hosts do not take part in the routing update process as the number of hosts in internet is much more than routers. So in general, the host uses static routing with a routing table that has a limited number of entries. As a result of this, the host may send a datagram to the wrong router. In this case, the router that receives the datagram will forward the datagram to the correct router and send a redirection message to the host for updating the routing table of the host.

A query message is encapsulated in an IP packet and the IP packet is encapsulated in a data link layer frame. The four pairs of query messages are discussed as follows:

The echo-request and echo-reply messages are used by network managers and users to identify network problems. This pair of messages can be used to determine the presence of communication at the IP level. ICMP messages are encapsulated in IP datagrams, so if a machine which sent an echo request, receive an echo-reply message then it means that the IP protocols in the sender and receiver are communicating with each other using the IP datagram and the intermediate routers are also receiving, processing and

forwarding IP datagrams.

The timestamp request and timestamp reply messages are used by any two machines to determine the round-trip time needed for an IP datagram to travel between them. This pair of query messages is also used to synchronize the clocks in two machines.

The address-mask request and reply messages are used by a host to obtain its mask. A host sends an address-mask-request message to a router on the LAN to obtain its mask. Now, if the host knows the address of the router; then it sends the request directly to the router otherwise it broadcasts the message. After receiving the address-mask-request message, the router sends an address-mask-reply message, providing the necessary mask for the host. A host that wants to send data to a host on another network can obtain the address of routers connected to its network by using the router solicitation and advertisement query messages. This pair of query messages is also used to know if the routers are alive and functioning. A host can broadcast a router-solicitation message. The routers receive the solicitation message and broadcast their routing information using the router-advertisement message.

There are two tools. These are Ping and Traceroute. These can be used in the Internet for debugging. These tools use ICMP for debugging. Ping program is used to determine if a host is alive and responding.

The Traceroute program in UNIX or tracert in Windows can be used to trace the route of a packet from the source to the destination.



CHECK YOUR PROGRESS

1. Multiple choice questions:

(I) The length of IPv4 address is

A. 32 bits

B. 128 bits

C. 32 bytes

D. None of above

(II) The size of the address space of IPv6 is

A. 2^{16}

B. 2^{32}

C. 2^{128}

D. Both B and C

(III) Which is not a subcategory of reserved addresses in IPv6?

A. Loopback address

B. Mapped address

C. Local address

D. Compatible address

(IV) Which is the decision making strategy of Multicast distance vector routing algorithm?

A. Flooding

B. Reverse path forwarding

C. Reverse path broadcasting

D. All of the above

(V) Which one is a unicast routing protocol?

A. Open Shortest Path First protocol

B. Address Resolution Protocol.

C. DVMRP protocol

D. None of the above

(VI) Which protocol is designed to create mapping between physical and logical addresses.

A. Internet Control Message Protocol

B. Internet Protocol

C. Address Resolution Protocol.

D. Both B and C

(VII) The length of the header part in an Internet Control Message Protocol message

A. 2 byte

B. 4 byte

C. 8 byte

D. 16 byte

(VIII) Routing Information Protocol is a

A. Multicast routing protocol

B. Broadcast routing protocol

C. Unicast routing protocol

D. None of the above

(IX) Which is not a type of link specified in OSPF protocol?

A. Point to point

B. Transient
C. Virtual
D. Backbone

(X) Which is not a dynamic routing algorithm?
A. Flooding
B. Distance vector routing algorithm
C. Link state routing algorithm
D. Both B and C

2. Fill in the blanks:

I. The internetwork is a _____ network.
II. _____service is used in the datagram approach to packet switching.
III. The term IP address is used to mean a _____ address in the network layer of the TCP/IP protocol suite.
IV. The length of an IPv6 address is _____ bytes.
V. In IPv6, reserved addresses start with _____s.
VI. In case of OSPF protocol, there is a special area inside an autonomous system called _____.
VII. Distance Vector Multicast Routing Protocol (DVMRP) is a source-based routing protocol based on -----_____.
VIII. The performance of ARP can be improved by using _____memory.
IX. An Internet Control Message Protocol message has two parts: _____ and a _____.
X. If the decimal value of the first byte of an IPv4 address is in the range _____ then it is a class B address.

3. State whether the following statements are True of False

I. If the decimal value of the first byte of an IPv4 address is in the range 192 to 223 then it is a class A address.
II. The basic concept of NAT is to assign each company a large set of addresses internally and one address or a small set of addresses externally.
III. Link state routing is a static routing algorithm.
IV. Hardware type is a 16-bit field defining the type of the network on which ARP is running.
V. MOSPF protocol uses multicast link state routing to create source-based trees.

VI. IPv4 is an unreliable and connection oriented protocol.
VII. The IPv6 packet is consists of a base header followed by the payload.
VIII. Using proxy ARP a network can be extended without the knowledge of the upstream router.
IX. The IP protocol has an error-reporting or error-correcting mechanism.
X. Ping program is used to determine if a host is alive and responding.



8.7 LET US SUM UP

- The network layer provides the mechanism of transferring variable length data sequences from a source host on one network to a destination host on a different network.
- A collection of interconnected networks, permitting data to move freely among different networks is called an internetwork or internet.
- An IPv4 address is a 32 bit address which uniquely and globally defines the connection of a device to the internet.
- Two types of notations are used to show an IPv4 address: binary notation and dotted-decimal notation.
- There are two addressing scheme in IPv4: classful addressing and classless addressing.
- Network Address Translation (NAT) is a solution for address depletion in IPv4 addressing.
- IPv6 has an address space of 2^{128} addresses because it uses a 128-bit address.
- IPv6 specifies hexadecimal colon notation for its addresses where 128 bits is divided into eight sections and each 2 bytes in length.
- In case of IPv6, the IP addresses are divided into several categories: Unicast Addresses, Multicast addresses, Anycast Addresses, Reserved Addresses and Local Addresses.
- The routing algorithms are the part of network layer software responsible for deciding the routes and the data structures to transmit the incoming packets efficiently.
- The different goals of a routing algorithm are: Correctness, Simplicity, Robustness, Stability, Fairness and Optimality.
- Routing algorithms can be divided into two major classes: nonadaptive and adaptive algorithms.
- Nonadaptive algorithms are static routing algorithms where the choice of the route to transmit IP packets from one node to another

node is computed in advance and downloaded to the routers when the network is booted.

- Adaptive algorithms are dynamic algorithms where the routing decisions are changed whenever there is a change in the topology and the network traffic.
- Routing protocols are divided into categories: unicast and multicast protocols.
- In unicast routing, when a router receives a packet to route then it needs to find the shortest path to the destination of the packet.
- In multicast routing, router receives multicast packet for routing to the destinations in more than one network.
- Distance vector routing and link state routing are two dynamic routing algorithm.
- The Routing Information Protocol (RIP) is an intradomain routing protocol used inside an autonomous system based on distance vector routing.
- The open shortest path protocol is an intradomain unicast routing protocol based on link state routing.
- Multicast link state routing is an extension of unicast link state routing.
- Multicast distance vector routing is the extension of unicast distance vector routing.
- Multicast Open Shortest Path First (MOSPF) Protocol is an extension of the OSPF protocol.
- DVMRP is a multicast routing protocol which uses multicast distance vector routing.
- The Internet Protocol version 4(IPv4) is used by the TCP/IP protocol.
- IPv4 is an unreliable and connectionless protocol for a packet-switching network that uses the datagram approach.
- IPv6 (Internetworking Protocol, version 6) is the new version of internet layer protocol for packet-switched internetworking and it provides end-to-end datagram transmission across multiple IP networks.
- The IPv6 packet is consists of a base header followed by the payload.
- The Address Resolution Protocol (ARP) is designed to create a mapping between physical and logical addresses.
- The Internet Control Message Protocol (ICMP) is designed to provide error-reporting or error-correcting mechanism. It also provides mechanism for host and management queries.
- ICMP messages are divided into two categories which are error-reporting messages and query messages.



8.8 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw- Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.



8.9 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: (I) A , (II) C , (III) C , (IV) D , (V) A
(VI) C , (VII) C , (VIII) C , (IX) D , (X) A

Answer 2: I. packet-switched ,II. connectionless , III. logical , IV. 16 , V. eight 0, VI. backbone, VII. RIP , VIII. cache , IX. Header, data section ,, X. 128 to 191.

Answer 3: I. False ,II. True , III. False , IV. True , V. True , VI. False, VII. True , VIII. True , IX. False , X. True



8.10 MODEL QUESTIONS

Q1. Explain the different addressing schemes in Pv4.
Q2. Define Routing. What are the different goals of routing algorithm?
Q3. Explain distance vector routing and link state routing.
Q4. Explain the different fields of an IPv4 packet.
Q5. Explain Address Resolution Protocol (ARP).

****

UNIT 9 : TRANSPORT LAYER

UNIT STRUCTURE

9.1 Learning Objections
9.2 Introduction
9.3 Transport Layer
    9.3.1 Relationship between Transport and Network Layers
    9.2.2 Overview of the Transport Layer in the Internet
    9.3.3 Ports and Sockets
        9.3.3.1 Ports
        9.3.3.2 Sockets
9.4 Process-To-Process Delivery
9.5 User Datagram Protocol (UDP)
    9.5.1 Applications of UDP
    9.5.2 Suitability of UDP for Certain Application
    9.5.3 Drawbacks of UDP
9.6 Transmission Control Protocol (TCP)
    9.6.1 Characteristics of TCP
    9.6.2 TCP Segment Structure
    9.6.3 Protocol Operation
    9.6.4 Connection Establishment
    9.6.5 Connection Termination
    9.6.6 Sliding Window
    9.6.7 Comparison of UDP and TCP
9.7 Let Us Sum Up
9.8 Further Reading
9.9 Answers to Check Your Progress
9.10 Model Questions

9.1 LEARNING OBJECTIVES

- define ports and sokets.
- describe process to process delivery
- describe user datagram protocol (UDP)
- describe transmission control protocol

9.2 INTRODUCTION

In the previous units we have discussed about the network layer. Different routing mechanisms and network layer protocols were discussed in the previous unit.

In this unit we will discuss about the transport layer of the OSI model and the relationship between transport and network layers. Other topics of discussion in the unit, include ports and sockets, application to application delivery, user datagram protocol and transmission control protocol. In the next unit, we will discuss topics related to session layer.

9.3 TRANSPORT LAYER

Transport Layer is 4th layer of the network OSI model. The transport layer of the OSI model offers end-to-end communication between end devices through a network. It also offers application-to-application communication between two applications in the both. Depending on the application, the transport layer offers reliable, connection-oriented or connectionless, best-effort communications.

The two most common transport layer protocols are the connection-oriented Transmission Control Protocol (TCP) and the connectionless User Datagram Protocol (UDP).

The transport layer ensures that error free messages are delivered, in sequence, and with no losses or duplications. It relieves the higher layer protocols from any concern with the transfer of data between them and their peers. The size and complexity of a transport protocol depends on the type of service it can get from the network layer. For a reliable network layer with virtual circuit capability, a minimal transport layer is required. If the network layer is unreliable and/or only supports datagram, the transport protocol should include extensive error detection and recovery.

The transport layer provides transfer of data between the end users, providing reliable data transfer services to the upper layers. It controls the reliability of a given link through flow control, segmentation/de-segmentation, and error control. The transport layer can keep track of the segments and retransmit those that fail. Transport layer also provides the acknowledgement of the successful data transmission and sends the next data if no errors occurred.

The transport layer provides:

- **Message segmentation:** It accepts a message from the (session) layer above it, splits the message into smaller units, if required, and passes the smaller units down to the network layer. The transport layer at the destination end re-assembles the message to get the original message back.
- **Message acknowledgment: I**t provides reliable end-to-end message delivery with acknowledgments.
- **Message traffic control:** It tells the transmitting host to 'wait-for-sometime' when no message buffers are available in the receiving host.
- **Session multiplexing:** It multiplexes several message streams, or sessions onto one logical link and keeps track of which messages belong to which sessions.

Typically, the transport layer can accept relatively large messages, but there are strict message size limits imposed by the network (or lower) layer. Consequently, the transport layer must break up the messages into smaller units, or frames, put a transport layer header to each frame. The transport layer header includes control information, such as message start and message end flags, to enable the transport layer on the other end to recognize message boundaries. In addition, if the lower layers do not maintain sequence, the transport header must contain sequence information to enable the transport layer on the receiving end to get the pieces back together in the right order before handing the received message up to the layer above.

Residing between the application and network layers, the transport layer has the critical role of providing communication services directly to the application processes running on different hosts. In this chapter we'll examine the possible services provided by a transport layer protocol and the principles underlying various approaches towards providing these services.

9.3.1 Relationship between Transport and Network Layers

In theory, the transport layer and network layer are distinct, but in practice, they are often very closely related to each other. We can

see this easily just by looking at the names of common protocol stacks—they are often named after the layer three and four protocols in the suite, implying their close relationship. For example, the name "TCP/IP" comes from the suite's most commonly used transport layer protocol (TCP) and network layer protocol (IP). Similarly, the Novell NetWare suite is often called "IPX/SPX" for its layer three (IPX) and layer four (SPX) protocols. Typically, specific transport layer protocols consider the network layers in the same family.

In the sending end, the transport layer of the network software takes input from the above layer. In case of TCP/IP protocol suit, it is the session layer. Then the transport layer processes them according to its own mechanism. After the process, the frames are handed over to the next lower layer in the protocol suit. For TCP/IP, it is the network layer. The network layer prepares them to transfer through the physical medium used in the network.

In the receiving end, the reverse process is executed. The incoming frames are accepted by the network layer. Network layer removes the network layer header and process those frames accordingly. Once this layer's processing is finished, the frames are handed over to the transport layer.

The most commonly used transport layer protocols are the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) in the TCP/IP suite; the Sequenced Packet Exchange (SPX) protocol in the NetWare protocol suite; and the NetBEUI protocol in the NetBIOS/NetBEUI/NBF suite.

9.3.2 Overview of the Transport Layer in the Internet

The backbone of Internet is the TCP/IP protocol suit. This protocol suit makes data transfer possible among heterogeneous networks. It is using two distinct transport-layer protocols and makes them available to the application layer. One is UDP (User Datagram Protocol), which provides an unreliable, connectionless service to the invoking application. The second one is TCP (Transmission Control Protocol), which provides a reliable & connection-oriented service to the invoking application. When designing a network

application using TCP/IP protocol suit, the application developer must specify any one of these two transport layer protocols.

9.3.3 Ports and Sockets

This section introduces the concepts of port and socket, which are needed to determine which local application in the sending end actually wants to communicates with which remote application in the receiving end.

9.3.3.1 Ports

A port is a 16-bit number, used by the host-to-host protocol to identify to which higher level protocol or application process it must deliver incoming messages.

There are two types of port:

- **Well-known:** Well-known ports belong to standard server applications. For example, Telnet uses well-known port number 23, HTTP uses well-known port number 80 etc. Well-known port numbers range between 1 and 1023. Most servers require only a single port. Exceptions are the FTP server, which uses two: 20 and 21. The well-known ports are controlled and assigned by the Internet Assigned Number Authority (IANA) and on most systems can only be used by system processes or by programs executed by privileged users. The reason for well-known ports is to allow clients to be able to find servers without configuration information. The well-known port numbers are defined in STD 2 – Assigned Internet Numbers

- **Ephemeral:** Clients do not need well-known port numbers because they initiate communication with servers and the port number they are using is contained in the TCP/UDP datagram sent to the server. Each client application is allocated a port number for as long as it needs it by the host it is running on. Ephemeral port numbers have values greater than 1023, normally in the range 1024 to 65535. A client can use any number allocated to it, as long as the combination of <transport protocol, IP address, port number> is unique. Ephemeral ports are not

controlled by IANA and can be used by ordinary user-developed programs on most systems.

Due to two different applications trying to use the same port numbers on one host, confusion is avoided by writing those applications to request an available port from TCP/IP. Because this port number is dynamically assigned, it may differ from one invocation of an application to the next. UDP and TCP all use the same port principle. To the best possible extent, the same port numbers are used for the same services on top of UDP and TCP.

9.3.3.2 Sockets

Since transport layer at the receiving host delivers data to the socket, there should be a unique identifier for each socket.
Socket identifier is called socket address.
Socket address = IP address : Port number (Figure 9.1)

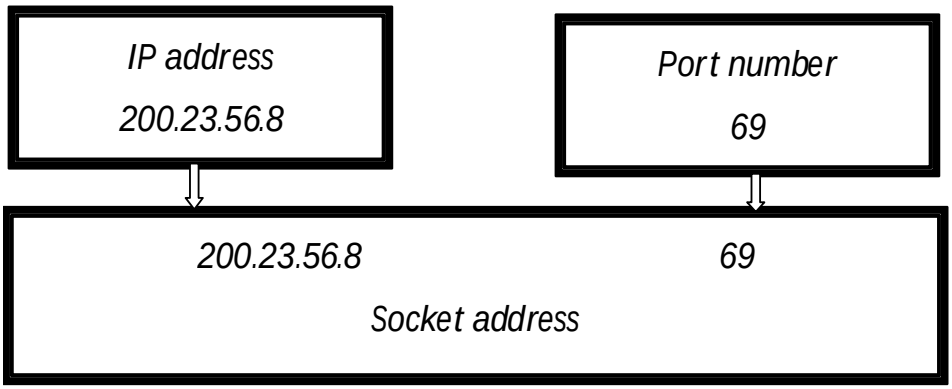


Figure 9.1: Socket Address

- A socket is a special type of file handle, which is used by an application to request network services from the operating system.
- A socket address is: <IP-address and local-application>

For example, in the TCP/IP suite: <193.44.234.3, 12345>
A socket is an endpoint for communication that can be named and addressed in a network. Two applications in two remote PCs communicate via TCP sockets. The socket model provides an application a connection to another application. These facilities are provided by TCP. TCP uses the same port principle as UDP to provide the communication. Like UDP, TCP uses well-known

and ephemeral ports. If two applications are communicating over TCP, they have a logical connection that is uniquely identifiable by the two sockets involved, that is, by the combination <local IP address, local port, remote IP address, remote port>.

9.4 PROCESS-TO-PROCESS DELIVERY

The transport layer is responsible for process-to-process delivery—the delivery of a packet, part of a message, from one application to another over network. Two processes communicate in a client /server architecture. UDP and TCP are transport-layer protocols that create a process-to-process communication. UDP is an unreliable and connectionless protocol that requires little overhead and offers fast delivery.

In the client-server paradigm, an application program on the local host, called the client, needs services from an application program on the remote host, called a server. Each application program has a unique port number that distinguishes it from other programs running at the same time on the same machine. The client program is assigned a random port number called the ephemeral port number. The server program is assigned a universal port number called a well-known port number. The combination of the IP address and the port number, called the socket address, uniquely defines a process and a host. TCP uses a sliding window mechanism for flow control.

9.5 USER DATAGRAM PROTOCOL (UDP)

UDP is a connectionless, unreliable transport protoco
l. It does not add anything to the services of IP except to provide process-to process communication instead of host-to-host communication. UDP is a very simple protocol using a minimum of overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP.

UDP uses a simple transmission model with a minimum of protocol mechanism. It has no handshaking dialogues and thus, exposes any unreliability of the underlying network protocol to the user's program. As this is normally IP over unreliable media, there is no guarantee of delivery, ordering or duplicate protection. UDP provides Checksum for data integrity, and port number for addressing different functions at the source and destination of the datagram. UDP is suitable for purposes where error checking and

correction is either not necessary or performed in the application, avoiding the overhead of such processing at the network interface level. Time-sensitive applications often use UDP because dropping packets is preferable to waiting for delayed packets, which may not be an option in a real-time system. If error correction facilities are needed at the network interface level, an application may use the Transmission Control Protocol (TCP) which is designed for this purpose. Figure 9.2 shows a UDP datagram. UDP datagram is having 8-byte header which is divided into four fields each with 2 bytes.

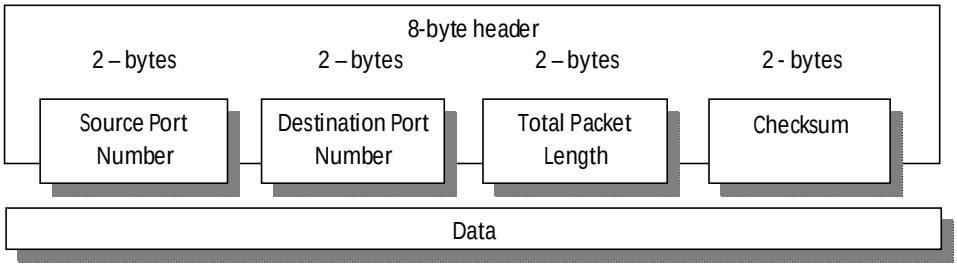


Figure 9.2: UDP datagram

The four files are:

1. **Source port number:** This is the port number used by the process running on the source host. This field identifies the sender's port when meaningful and should be assumed to be the port to reply to if needed. If not used, then it should be zero. If the source host is the client, the port number is likely to be an ephemeral port number. If the source host is the server, the port number is likely to be a well-known port number.
2. **Destination port number:** This is the port number used by the process running on the destination host. This field identifies the receiver's port and is required. Similar to source port number, if the client is the destination host then the port number will likely be an ephemeral port number and if the destination host is the server then the port number will likely be a well-known port number.
3. **Total Packet Length:** It defines the total length of the user datagram. A field that specifies the length in bytes of the entire datagram: header and data. The minimum length is 8 bytes since that's the length of the header. The field size sets a theoretical limit of 65,535 bytes (8 byte header + 65,527 bytes of data) for a UDP datagram. The practical limit for the data length which is imposed by the underlying IPV4 protocol is 65,507 bytes (65,535 " 8 byte UDP header " 20 byte IP header).
4. **Checksum:** The checksum field is used for error-checking of the header and data. If no checksum is generated by the transmitter, the field uses the value all-zeros. This field is not optional for IPv6.

9.5.1 Applications of UDP

Numerous key Internet applications use UDP, including: the Domain Name System (DNS), where queries must be fast and only consist of a single request followed by a single reply packet, the Simple Network Management Protocol (SNMP), the Routine Information Protocol (RIP) and the Dynamic Host Configuration Protocol (DHCP).Voice and video traffic is generally transmitted using UDP. Real-time video and audio streaming protocols are designed to handle occasional lost packets, so only slight degradation in quality occurs, rather than large delays if lost packets were retransmitted.

9.5.2 Suitability of UDP for Certain Application

- It is transaction-oriented, suitable for simple query-response pro tocols such as the Domain Name Systems or the Network Time Protocol.
- It provides datagrams, suitable for modeling other protocols such as in IP tunneling or Remote Procedure Call and the Network File System.
- It is simple, suitable for bootstrapping or other purposes without a full protocol stack, such as the DHCP and Trivial File Transfer Pro tocol.
- It is stateless, suitable for very large numbers of clients, such as in streaming media applications for example IPTV.
- The lack of retransmission delays makes it suitable for real-time applications such as Voice over IP (VoIP), online games, and many protocols built on top of the Real Time Streaming Protocol.
- Works well in unidirectional communication, suitable for broadcast information such as in many kinds of service discovery and shared information such as broadcast time or Routing Information Protocol.
- Sending a small message by using UDP takes much less interaction between the sender and receiver than using TCP or SCTP. UDP packets, called user datagrams, have a fixed size header of 8 bytes.

9.5.3 Drawbacks of UDP

- There is no flow control. The receiver may overflow with incoming messages.
- There is no error control mechanism in UDP except for the checksum.
- The sender does not know if a message has been lost or duplicated.
- When the receiver detects an error through the checksum, the user datagram is discarded.
- Each user datagram can travel on a different path. No relationship between the different user datagram even if they are coming from the same source process and going to the same destination program. Also, there is no connection establishment and no connection termination.
- UDP provides no guarantees to the upper layer protocol for message delivery and the UDP protocol layer retains no state of UDP messages once sent. For this reason, UDP is sometimes referred to as unreliable datagram protocol.

9.6 TRANSMISSION CONTROL PROTOCOL (TCP)

The TCP is one of the two original core protocols of the Internet protocol (IP) suite, and is so common that the entire suite is often called TCP/IP. TCP provides reliable, ordered, error-checked delivery of a stream of octets between programs running on computers connected to an intranet or the public Internet. Browsers use it when they connect to servers on the world wide web sites, and it is used to accurately deliver email and transfer files from one location to another. Applications that do not require the reliability of a TCP connection may instead use the connectionless UDP.

The protocol corresponds to the transport layer of TCP/IP suite. TCP provides a communication service at an intermediate level between an application program and the IP. That is, when an application program desires to send a large chunk of data across the Internet using IP, instead of breaking the data into IP-sized pieces and issuing a series of IP requests, the software can issue a single request to TCP and let TCP handle the IP details.

Due to network congestion, traffic load balancing, or other unpredictable network behavior, IP packets can be lost, duplicated, or delivered out of order. TCP detects these problems, requests retransmission of lost data, rearranges out-of-order data, and even helps minimize network congestion to reduce the occurrence of the other problems. Once the TCP receiver has reassembled the sequence of octets originally transmitted, it passes them to the application program. Thus, TCP abstracts the application's communication from the underlying networking details.

TCP is utilized extensively by many of the Internet's most popular applications, including the World Wide Web (WWW), E-mail, File Transfer Protocol (FTP), Secure Shell, peer-to-peer file sharing, and some streaming media applications.

TCP is optimized for accurate delivery rather than timely delivery, and therefore, TCP sometimes incurs relatively long delays (in the order of seconds) while waiting for out-of-order messages or retransmissions of lost messages. It is not particularly suitable for real-time applications such as VoIP. For such applications, protocols like the Real-time Transport Protocol (RTP) running over the UDP are usually recommended instead.

TCP is a reliable stream delivery service that guarantees that all the bytes received will be identical with the bytes sent and in the correct order. Since packet transfer is not reliable, a technique known as positive acknowledgment with retransmission is used to guarantee the reliability of packet transfers. This fundamental technique requires the receiver to respond with an acknowledgment message as it receives the data. The sender keeps a record of each packet it sends. The sender also keeps a timer from when the packet was sent, and retransmits a packet if the timer expires before the message has been acknowledged. The timer is needed in case a packet gets lost or corrupted.

TCP consists of a set of rules: for the protocol, that are used with the Internet Protocol, and for the IP, to send data "in a form of message units" between computers over the Internet. While IP handles actual delivery of the data, TCP keeps track of the individual units of data transmission, called segments that a message is divided into for efficient routing through the network. For example, when an HTML file is sent from a Web server, the TCP software layer of that server divides the sequence of octets of the file into segments and forwards them individually to the IP software layer (Internet Layer). The Internet Layer encapsulates each TCP segment into

an IP packet by adding a header that includes (among other data) the destination IP address. Even though every packet has the same destination address, they can be routed on different paths through the network. When the client program on the destination computer receives them, the TCP layer (Transport Layer) reassembles the individual segments and ensures they are correctly ordered and error free as it streams them to an application.

9.6.1 Characteristics of TCP

TCP can be characterized by the following facilities it provides for the applications using it:

- **Stream Data Transfer:**

From the application's viewpoint, TCP transfers a continuous stream of bytes through the network. TCP does this by grouping the bytes in TCP segments, which are passed to IP for transmission to the destination. Also, TCP itself decides how to segment the data and it can forward the data at its own convenience. Sometimes, an application needs to be sure that all the data passed to TCP has actually been transmitted to the destination. For that reason, a push function is defined. It will push all remaining TCP segment still in storage to the destination host. The normal close connection function also pushes the data to the destination.

- **Reliability:**

TCP uses a sequence number to identify each byte of data. The sequence number identifies the order of the bytes sent from each computer so that the data can be reconstructed in order, regardless of any fragmentation, disordering, or packet loss that may occur during transmission. For every payload byte transmitted, the sequence number must be incremented. In the first two steps of the 3-way handshake, both computers exchange an initial sequence number (ISN). This number can be arbitrary, and should in fact be unpredictable to defend against TCP sequence prediction attacks.

TCP primarily uses a cumulative acknowledgment scheme, where the receiver sends an acknowledgment signifying that the receiver has received all data preceding the acknowledged sequence number. The sender sets the sequence number field to the sequence number of the first payload byte in the segment's data field, and the receiver

sends an acknowledgment specifying the sequence number of the next byte they expect to receive. For example, if a sending computer sends a packet containing four payload bytes with a sequence number field of 100, then the sequence numbers of the four payload bytes are 100, 101, 102 and 103. When this packet arrives at the receiving computer, it would send back an acknowledgment number of 104 since that is the sequence number of the next byte it expects to receive in the next packet.

In addition to cumulative acknowledgments, TCP receivers can also send selective acknowledgments to provide further information. If the sender infers that data has been lost in the network, it retransmits the data.

- **Flow Control**

The receiving TCP, when sending an acknowledgment back to the sender, also indicates to the sender the number of bytes it can receive beyond the last received TCP segment, without causing overrun and overflow in its internal buffers. This is sent in the acknowledgment in the form of the highest sequence number it can receive without problems. This mechanism is also referred to as a window-mechanism.

TCP uses an end-to-end flow control protocol to avoid having the sender send data too fast for the TCP receiver to receive and process it reliably. Having a mechanism for flow control is essential in an environment where machines of diverse network speeds communicate. For example, if a PC sends data to a Smartphone that is slowly processing received data, the Smartphone must regulate the data flow so as not to be overwhelmed.

TCP uses a sliding window flow control protocol. In each TCP segment, the receiver specifies in the receive window field the amount of additionally received data (in bytes) that it is willing to buffer for the connection. The sending host can send only up to that amount of data before it must wait for an acknowledgment and window update from the receiving host.

TCP sequence numbers and receive windows behave very much like a clock. The receive window shifts each time the receiver receives and acknowledges a new segment of data. Once it runs out of sequence numbers, the sequence number loops back to 0.

When a receiver advertises a window size of 0, the sender stops sending data and starts the persist timer. The persist timer is used to protect TCP from a deadlock situation that could arise if a subsequent window size update from the receiver is lost, and the sender cannot send more data until receiving a new window size update from the receiver. When the persist timer expires, the TCP sender attempts recovery by sending a small packet so that the receiver responds by sending another acknowledgement containing the new window size.

- **Congestion control**

Another main aspect of TCP is congestion control. TCP uses a number of mechanisms to achieve high performance and avoid congestion collapse, where network performance can fall by several orders of magnitude. These mechanisms control the rate of data entering the network, keeping the data flow below a rate that would trigger collapse. They also yield an approximately max-min fair allocation between flows.

Acknowledgments for data sent, or lack of acknowledgments, are used by senders to infer network conditions between the TCP sender and receiver. Coupled with timers, TCP senders and receivers can alter the behavior of the flow of data. This is more generally referred to as congestion control and/or network congestion avoidance.

Enhancing TCP to reliably handle loss, minimize errors, manage congestion and go fast in very high-speed environments are ongoing areas of research and standards development. As a result, there are a number of TCP congestion avoidance algorithm variations.

- **Maximum segment size**

The maximum segment size (MSS) is the largest amount of data, specified in bytes, that TCP is willing to receive in a single segment. For best performance, the MSS should be set small enough to avoid IP fragmentation, which can lead to packet loss and excessive retransmissions. To try to accomplish this, typically the MSS is announced by each side using the MSS option when the TCP connection is established, in which case it is derived from the maximum transmission unit (MTU) size of the data link layer of the networks to which the sender and receiver are directly attached. Furthermore, TCP senders can use path MTU discovery to infer the

minimum MTU along the network path between the sender and receiver, and use this to dynamically adjust the MSS to avoid IP fragmentation within the network.

MSS announcement is also often called “MSS negotiation”. Strictly speaking, the MSS is not “negotiated” between the originator and the receiver, because that would imply that both originator and receiver will negotiate and agree upon a single, unified MSS that applies to all communication in both directions of the connection. In fact, two completely independent values of MSS are permitted for the two directions of data flow in a TCP connection. This situation may arise, for example, if one of the devices participating in a connection has an extremely limited amount of memory reserved (perhaps even smaller than the overall discovered Path MTU) for processing incoming TCP segments.

- **Selective acknowledgments**

Relying purely on the cumulative acknowledgment scheme employed by the original TCP protocol can lead to inefficiencies when packets are lost. For example, suppose 10,000 bytes are sent in 10 different TCP packets, and the first packet is lost during transmission. In a pure cumulative acknowledgment protocol, the receiver cannot say that it received bytes 1,000 to 9,999 successfully, but failed to receive the first packet, containing bytes 0 to 999. Thus, the sender may then have to resend all 10,000 bytes.To solve this problem TCP employs the selective acknowledgment (SACK) option, which allows the receiver to acknowledge discontinuous blocks of packets that were received correctly, in addition to the sequence number of the last contiguous byte received successively, as in the basic TCP acknowledgment. The acknowledgement can specify a number of SACK blocks, where each SACK block is conveyed by the starting and ending sequence numbers of a contiguous range that the receiver correctly received. In the example above, the receiver would send SACK with sequence numbers 1000 and 9999. The sender thus retransmits only the first packet, bytes 0 to 999.

An extension to the SACK option is the duplicate-SACK option. An out-of-order packet delivery can often falsely indicate the TCP sender of lost packet and, in turn, the TCP sender retransmits the suspected-to-be-lost packet and slows down the data delivery to

prevent network congestion. The TCP sender undoes the action of slow-down that is a recovery of the original pace of data transmission, upon receiving a D-SACK that indicates the retransmitted packet is duplicated.

The SACK option is optional and used only if both parties support it. This is negotiated when connection is established. SACK uses the optional part of the TCP header. The use of SACK is widespread — all popular TCP stacks support it. Selective acknowledgment is also used in Stream Control Transmission Protocol (SCTP).

- Multiplexing: Achieved through the use of ports, just as with UDP.
- Logical Connections:

The reliability and flow control mechanisms described above require that TCP initializes and maintains certain status information for each data stream. The combination of this status, including sockets, sequence numbers and window sizes, is called a logical connection. Each connection is uniquely identified by the pair of sockets used by the sending and receiving processes.

- Full Duplex: TCP provides for concurrent data streams in both directions.

9.6.2 TCP Segment Structure

TCP accepts data from a data stream, divides it into chunks, and adds a TCP header creating a TCP segment. The TCP segment is then encapsulated into an Internet Protocol (IP) datagram. A TCP segment is the packet of information that TCP uses to exchange data with its peers.

The term TCP packet is not in line with current terminology, where segment refers to the TCP Protocol Data Unit (PDU), datagram to the IP PDU and frame to the data link layer PDU.

Processes transmit data by calling on the TCP and passing buffers of data as arguments. The TCP packages the data from these buffers into segments and calls on the internet module, e.g. IP, to transmit each segment to the destination TCP.

A TCP segment consists of a segment header and a data section. The TCP header contains 10 mandatory fields, and an optional extension field. The data section follows the header. Its contents

are the payload data carried for the application. The length of the data section is not specified in the TCP segment header. It can be calculated by subtracting the combined length of the TCP header and the encapsulating IP header from the total IP datagram length. The Figure 9.3 shows the layout of a TCP segment header. The header is of size 20 to 60 bytes. The 20 bytes is reserved for the fixed fields and the other 40 bytes are reserved for options, which is the additional information that the header carries to the destination. These 40 bytes are not mandatory.

Let us discuss briefly these header fields:

1. **Source Port Number:** This is 2-bytes port number of the application on the source computer which wants to send the TCP segment.
2. **Destination Port Number:** This is 2-bytes port number of the application on the destination computer which is expected to receive the TCP segment.
3. **Sequence Number:** TCP sends multiple segments from source to destination in one TCP connection. So, it becomes important to number those segments in an increasing sequence to maintain the connectivity. The 4 bytes sequence number is assigned a number to the first byte of data portion in the TCP segment. It makes the destination host clear about the first byte of the TCP segment.

   During the TCP connection phase, the source and destination host randomly generate different unique numbers. Let's assume that for the source host, this random unique number is 2120 and first TCP segment is carrying 2000 bytes of data. The number 2120 and 2121 will be used in establishing the TCP connection with the destination host. Then the next number 2122 will be used in the sequence number field of the first TCP segment which will carry 2000 bytes of data. The second segment will carry the sequence number 4122 (= 2122+2000).
4. **Acknowledgement Number:** If the destination host receives a segment with sequence number X correctly, it sends 4 bytes acknowledgement number X+1 back to the source host which acknowledge the correct receipt of the previous segment from the source.

5. **Header Length:** This 4 bits field specifies the number of 4-bytes words in the TCP segment. Since the size of the TCP header can be 20 to 60 bytes, therefore, the value of this field can be between 5 and 15.
   If it is 5 then the length of the TCP header is 5 X 4 = 20 bytes, which is the minimum size it can have. Else if it is 15 then the size of the TCP header is 15 X 4 = 60, which is the maximum in size.

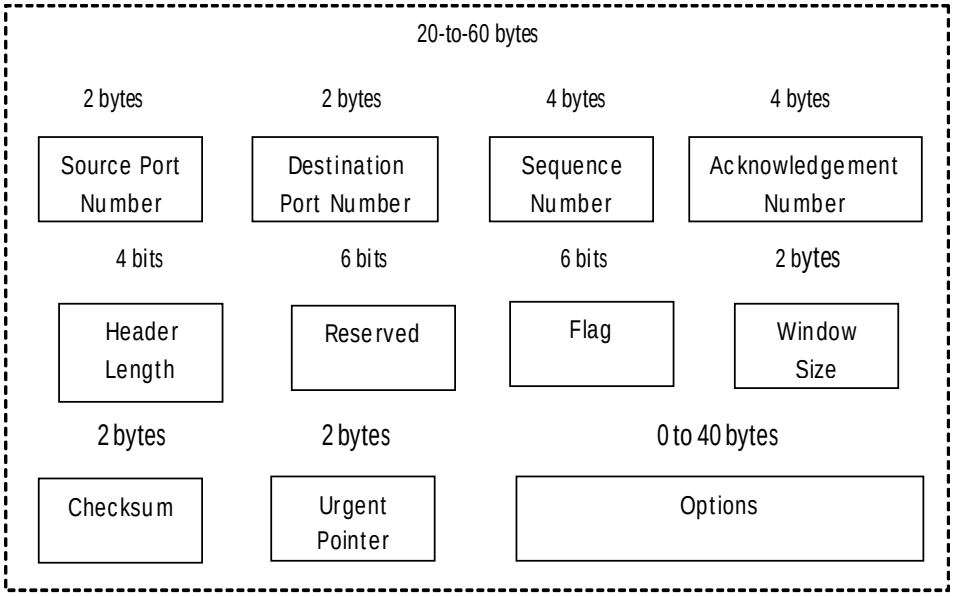


Figure 9.3: Layout of TCP segment header

6. **Reserved:** This 6-bits field is reserved for future use.

7. **Flag:** This 6-bits field defines 6 different control flag where each of them occupy 1 bit. They are

   a. URG: Set to 1 if Urgent Pointer is in use. It is used to indicate a byte offset from the current sequence number at which urgent data are to be found.

   b. ACK: Set to 1 to indicate that the Acknowledge Number is valid. If it is set to 0, Acknowledgement Number field is ignored.

   c. PSH: It is used to indicate Pushed Data. If it is set to 1, it instructed the receiver to deliver the data to the appropriate application once received. Else receiver waits till the buffer is full.

   d. RST: Used to reset a connection that has become confused due to some reason. It is also used to reject a segment or

refuse to open a connection.

e. SYN: It is used to establish a connection. A request for a connection contains SYN = 1 and ACK = 0. Reply to this request bears SYN = 1 and ACK = 1.

f. FIN: It is used to close a connection.

8. Window Size: This 2-bytes field determines the size of the sliding window that the other end must maintain.

9. Checksum: This 2-bytes field contains the checksum which is used for error detection and correction.

10. Urgent Pointer: This 2-bytes field is used to indicate that some data in a TCP segment is more urgent than others in the same connection.

9.6.3 TCP Protocol Operation

TCP protocol operations may be divided into three phases.
Phase 1: Connections must be properly established in a multi-step handshake process (connection establishment) before entering phase 2.

Phase 2: The data transfer phase. After data transmission is completed, phase 3 is executed.

Phase 3: The connection termination closes established virtual circuits and releases all allocated resources.

A TCP connection is managed by an operating system through a programming interface that represents the local end-point for communications, the Internet socket. During the lifetime of a TCP connection the local end-point undergoes a series of state changes:

a. LISTEN: (Server) represents waiting for a connection request from any remote TCP and port.

b. SYN-SENT: (Client) represents waiting for a matching connection request after having sent a connection request.

c. SYN-RECEIVED: (Server) represents waiting for a confirming connection request acknowledgment after having both received and sent a connection request.

d. ESTABLISHED: (Both server and client) represents an open

connection, data received can be delivered to the user. The normal state for the data transfer phase of the connection.

e. FIN-WAIT-1: (Both server and client) represents waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent.

f. FIN-WAIT-2: (Both server and client) represents waiting for a connection termination request from the remote TCP.

g. CLOSE-WAIT: (Both server and client) represents waiting for a connection termination request from the local user.

h. CLOSING: (Both server and client) represents waiting for a connection termination request acknowledgment from the remote TCP.

i. LAST-ACK: (Both server and client) represents waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request).

j. TIME-WAIT: (Either server or client) represents waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request.

k. CLOSED: (Both server and client) represents no connection state at all.

9.6.4 Connection Establishment

To establish a connection, TCP uses a three-way handshake. Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections. This is called a passive open. Once the passive open is established, a client may initiate an active open. To establish a connection, the three-way (or 3-step) handshake occurs:

1. SYN: The active open is performed by the client sending a SYN to the server. The client sets the segment's sequence number to a random value A.

2. SYN-ACK: In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number (A + 1), and the sequence

number that the server chooses for the packet is another random number, B.

3. ACK: Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value i.e. A + 1, and the acknowledgement number is set to one more than the received sequence number i.e. B + 1.

   At this point, both the client and server have received an acknowledgment of the connection. The steps 1, 2 establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps 2, 3 establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

9.6.5 Connection Termination

The connection termination phase uses a four-way handshake, with each side of the connection terminating independently. When an endpoint wishes to stop its half of the connection, it transmits a FIN packet, which the other end acknowledges with an ACK. Therefore, a typical tear-down requires a pair of FIN and ACK segments from each TCP endpoint. After both FIN/ACK exchanges are concluded, the side which sent the first FIN before receiving one waits for a timeout before finally closing the connection, during which time the local port is unavailable for new connections; this prevents confusion due to delayed packets being delivered during subsequent connections.

A connection can be “half-open”, in which case one side has terminated its end, but the other has not. The side that has terminated can no longer send any data into the connection, but the other side can. The terminating side should continue reading the data until the other side terminates as well.

It is also possible to terminate the connection by a 3-way handshake, when host A sends a FIN and host B replies with a FIN & ACK (merely combines 2 steps into one) and host A replies with an ACK. This is perhaps the most common method.

It is possible for both hosts to send FINs simultaneously then both just have to ACK. This could possibly be considered a 2-way handshake since the FIN/ACK sequence is done in parallel for both directions.

Some host TCP stacks may implement a half-duplex close sequence, as Linux or HP-UX do. If such a host actively closes a connection but still has not read all the incoming data the stack already received from the link, this host sends a RST instead of a FIN. This allows a TCP application to be sure the remote application has read all the data the former sent—waiting the FIN from the remote side, when it actively closes the connection. However, the remote TCP stack cannot distinguish between a Connection Aborting RST and this Data Loss RST. Both cause the remote stack to throw away all the data it received, but that the application still didn't read.

Some application protocols may violate the OSI model layers, using the TCP open/close handshaking for the application protocol open/close handshaking — these may find the RST problem on active close.

For a usual program flow like above, a TCP/IP stack like that described above does not guarantee that all the data arrives to the other application.

9.6.6 Sliding Window

A sliding window is used to make transmission more efficient as well as to control the flow of data so that the destination does not become overwhelmed with data.
TCP sliding windows are byte-oriented.
A simple transport protocol might use the following principle: send a packet and then wait for an acknowledgment from the receiver before sending the next packet. If the ACK is not received within a certain amount of time, retransmit the packet. See Figure 9.4 for more details.

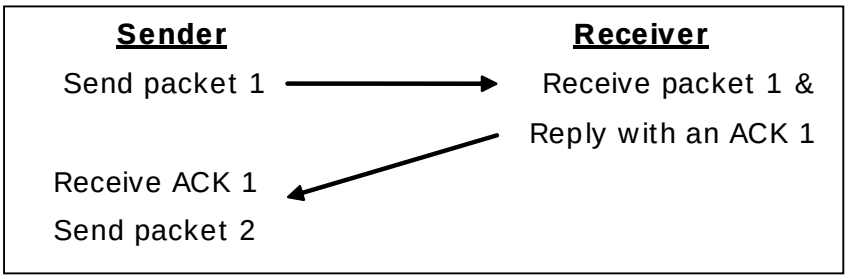


Figure 9.4: Sliding Window

While this mechanism ensures reliability, it only uses a part of the available network bandwidth. Now, consider a protocol where the sender groups its packets to be transmitted, and uses the following rules:

- The sender can send all packets within the window without receiving an ACK, but must start a timeout timer for each of them.
- The receiver must acknowledge each packet received, indicating the sequence number of the last well-received packet. The sender slides the window on each ACK received.

This window mechanism ensures:

a. Reliable transmission.
b. Better use of the network bandwidth (better throughput).
c. Flow-control, since the receiver may delay replying to a packet with an acknowledgment, knowing its free buffers are available and the window-size of the communication.

9.6.7 Comparison of UDP and TCP

Transmission Control Protocol is a connection-oriented protocol, which means that it requires handshaking to set up end-to-end communications. Once a connection is set up user data may be sent bi-directionally over the connection.

- Reliable – TCP manages message acknowledgment, retransmission and timeout. Multiple attempts to deliver the message are made. If it gets lost along the way, the server will re-request the lost part. In TCP, there's either no missing data, or, in case of multiple timeouts, the connection is dropped.
- Ordered – if two messages are sent over a connection in sequence, the first message will reach the receiving application first. When data segments arrive in the wrong order, TCP

buffers delay the out-of-order data until all data can be properly re-ordered and delivered to the application.

- Heavyweight – TCP requires three packets to set up a socket connection, before any user data can be sent. TCP handles reliability and congestion control.
- Streaming – Data is read as a byte stream, no distinguishing indications are transmitted to signal message (segment) boundaries.

  UDP is a simpler message-based connectionless protocol. Connectionless protocols do not set up a dedicated end-to-end connection. Communication is achieved by transmitting information in one direction from source to destination without verifying the readiness or state of the receiver. However, one primary benefit of UDP over TCP is the application to voice over internet protocol (VoIP) where latency and jitter are the primary concerns. It is assumed in VoIP UDP that the end users provide any necessary real time confirmation that the message has been received.
- Unreliable – When a message is sent, it cannot be known if it will reach its destination; it could get lost along the way. There is no concept of acknowledgment, retransmission, or timeout.
- Not ordered – If two messages are sent to the same recipient, the order in which they arrive cannot be predicted.
- Lightweight – There is no ordering of messages, no tracking connections, etc. It is a small transport layer designed on top of IP.
- Datagrams – Packets are sent individually and are checked for integrity only if they arrive. Packets have definite boundaries which are honored upon receipt, meaning a read operation at the receiver socket will yield an entire message as it was originally sent.
- No congestion control – UDP itself does not avoid congestion, and it's possible for high bandwidth applications to trigger congestion collapse, unless they implement congestion control measures at the application level.



CHECK YOUR PROGRESS

1. Which layer of the OSI model provides error correction and flow control?
   a) Presentation b) Transport
   c) Network d) Data link
2. The addressing especially used by Transport Layer is
   a) Station address
   b) Network address
   c) Application port address
   a) Dialog address
3. Both TCP and UDP belong to which layer of the OSI model?
   a) Session layer b) Transport layer
   c) Network layer d) Data Link layer
4. Which Protocol Data Unit (PDU) is employed at the Transport Layer?
   a. Bits b. Frames
   c. Packets d. Segments
5. What are the responsibilities of Transport Layer?
6. Which processes does TCP, but not UDP, use?
   a. Windowing b. Acknowledgements
   c. Source Port d. Destination Port
7. Which layer is responsible for providing mechanisms for multiplexing upper-layer application, session establishment, and tear down of virtual circuits?
   a. Session b. Network
   c. Physical d. Transport
   e. Application f. Presentation
8. Which two of the following protocols are used at the Transport layer?
   a) ARP
   b) UDP
   c) ICMP
   d)RARP
   e)TCP
   f) BootP



9.7 LET US SUM UP

- The transport layer provides Message segmentation, Message acknowledgment, Message traffic control & Session multiplexing.
- This layer uses two protocols to provide services. They are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).
- This layer uses the concepts of ports and sockets to provide services to the applications running in both the host computers.
- A port is a 16-bit number. It is used by the host-to-host protocol to identify to which higher level protocol or application process it must deliver incoming messages.
- TCP is connection oriented protocol as TCP always creates a virtual connection between the two hosts which are trying to communicate with each other using three way hand shaking method.
- TCP is reliable protocol as TCP carries out different mechanisms to detect errors like packet duplicacy, packet loss, sequencing and re-sequencing etc.
- UDP is connectionless and not reliable protocol.



9.8 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw. Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.



9.9 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: b

Answer 2: c

Answer 3: b

Answer 4: c

Answer 5: Write about reliability, flow control, congestion control, multiplexing, duplicacy control, message segmentation, loss

control & message acknowledgement.

Answer 6. b

Answer 7. d

Answer 8. b & e



9.10 MODEL QUESTIONS

Q1. What is the data unit of “Transport layer”? What are the functions of a Transport Layer in OSI Model?

Q2. What are ports and sockets?

Q3. Explain Process-To-Process delivery.

Q4. What is User Datagram? Explain the application of UDP and some drawback of UDP.

Q5. What is the source port number?

Q6. What is the destination port number?

Q7. What is sequence number?

Q8. What is the acknowledgment number?

Q9. What is the length of the header?

Q10. What is the type of the segment?

Q11. What is Transmission control protocol?

Q12. What is the TCP Protocol operation?

Q13. Explain on Connection Establishment and connection termination in terms of Transmission control protocol.

Q14. What is Sliding Window?

Q15. What is the window size?

Q16. What is Checksum?

Q17. What are the difference between TCP and UDP?

Q18. What is the way to establish a TCP connection?

Q19. Mark one of the most important differences between TCP and UDP. Which layer of OSI is responsible for end-to-end communication?

****

UNIT 10 : SESSION LAYER

UNIT STRUCTURE

10.1 Learning Objectives
10.2 Introduction
10.3 Session Layer
10.4 Session Layer Services
  10.4.1 Dialog Control
  10.4.2 Token Management
  10.4.3 Synchronization
10.5 Remote Procedure Call
10.6 Network Basic Input/Output System
10.7 Let Us Sum Up
10.8 Further Reading
10.9 Answers to Check Your Progress
10.10 Model Questions

10.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe the basic role of session layer in OSI model
- explain the concept of inter host communication
- define dialogue control mechanism
- describe token management process
- describe the process of synchronization
- explain the basic concept of Remote Procedure Call
- explain the fundamental concept of Network Basic Input/Output System

10.2 INTRODUCTION

In the previous unit, we have discussed different issues related to transport layer. Protocols like user datagram protocol and transmission control protocol were discussed in detail. In this unit, we will learn about session layer. Topics like session layer sevices and remote procedure call are discussed in this unit. In the next unit, we will explore the concepts related to presentation layer.

10.3 SESSION LAYER

Session layer is used for inter host communication. It establishes, manages and terminates sessions between two communicating hosts. It deals with Sessions i.e. when a user logins into a remote server, authentication of the user needs to be done before giving them access to the files and programs. It is also responsible for synchronizing dialogue between the presentation layers of the two hosts involved in data exchange. During the transfer of data between two machines if the session breaks, the session layer re-establishes the connection. It also ensures that the data transfer starts from where it was broken, keeping it transparent to the end user. In case of session with a database server it introduces check points at many places so that if the connection is broken, it should be able to re-established and recover from it, and the transition running on the database is not lost. This process is called as synchronization. This layer also deals with dialogue control which defines whose turn it is to speak in a session. It is useful in video conferencing. The session layer is usually implemented in application environments that use remote procedure calls.

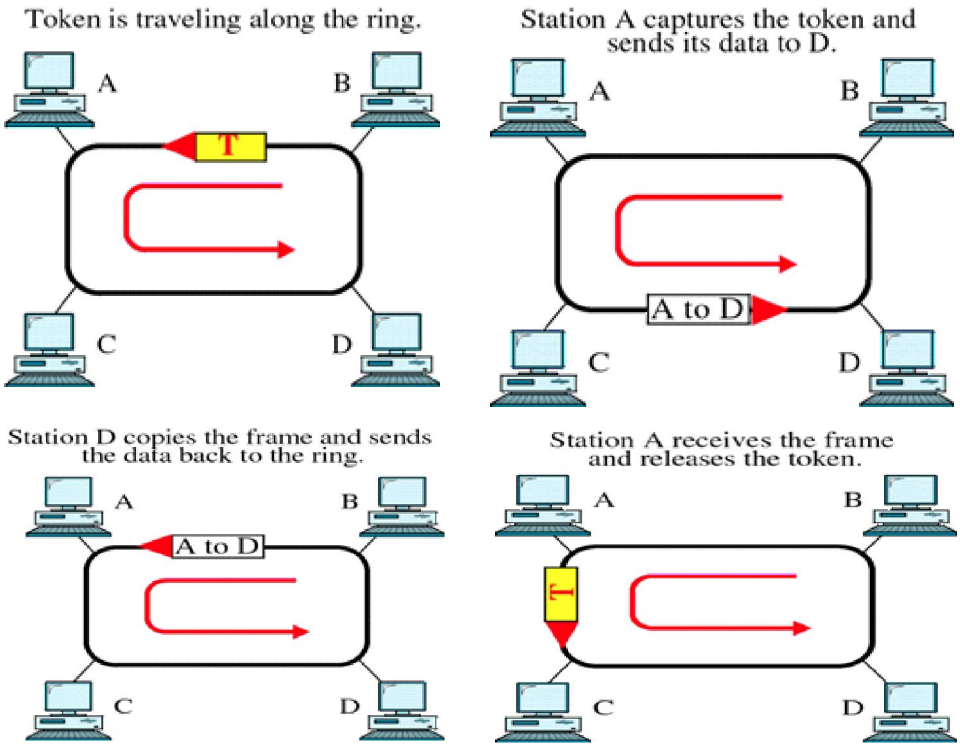


Figure 10.1: Session Layer

The session layer exists above the transport layer, and provides services to the transport layer services underneath it. The session layer along with the

presentation layer provides facilities to the transport layer that are expected to be of use to applications, such that each application doesn't have to provide its own implementation.

10.4 SESSION LAYER SERVICES

The session layer provides the following services:
i. Dialog control.
ii. Token management and
iii. Synchronization

10.4.1 Dialog Control

This service by session layer decides whose turn it is to talk during an active communication. The communication between two processes can take place in either half-duplex mode (one way at a time) or in full-duplex (two ways at a time) mode. In ISO protocols, dialog management is carried out with the help of a data token. The token is sent back and forth and a user may transmit only when it holds the token.

10.4.2 Token Management

In case of token-passing the stations in a network are arranged in a logical ring structure. Each station has a predecessor and a successor.

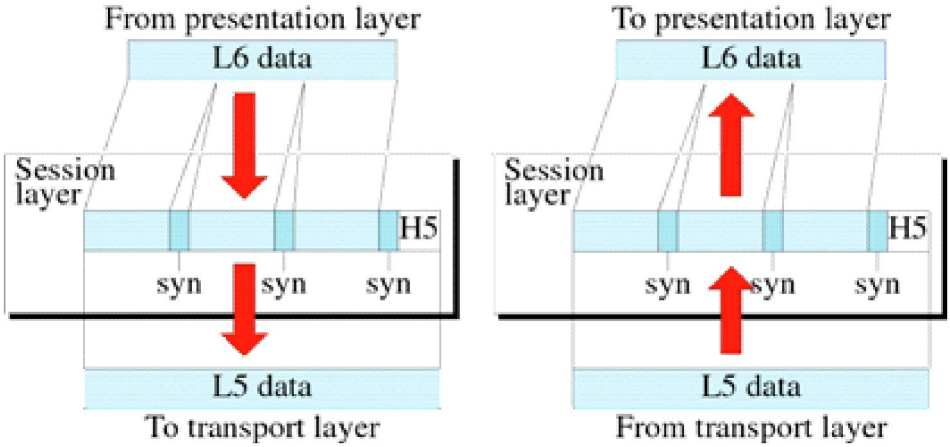
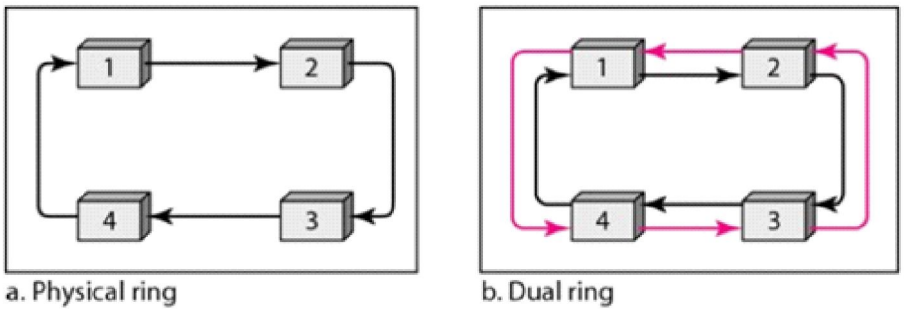


Figure 10.2: Token Passing

The predecessor is the station that is before the station in the ring

and the successor is the station which is after the station in the ring. The current station is the one that accesses the channel currently. To access the channel a packet called token circulates through the ring. The station that holds the token gives that station the right to access and sends data. When a station has some data to be sent, it waits till it receives the token from its predecessor. It then holds the token and sends its data. When the station has finished sending all data, it releases the token, passing it to the next logical station in the ring. This station cannot send data unless it receives the token again in the next round. The mechanism of token management is needed for finding methods to access the station. Every station should be limited in time for possession of the token. The token needs to be monitored to make sure that it has not been lost or destroyed. The token will disappear from the network if a station that is holding the token fails. Another purpose of token management is to assign priorities to the stations and the types of data which is being transmitted. A token management is necessary to make low priority stations release the token to higher priority stations. The token does not have to have the address of the next successor that means in a ring topology, when a station sends the token to its successor, the other stations cannot see the token. The disadvantage with this topology is that if one of the links between two adjacent stations fails, the entire system fails. The dual ring topology uses a second (auxiliary) ring which operates in the reverse direction in comparison with the main ring. The second ring is used for emergency; if one of the links in the main ring fails the system automatically combines the two rings to form a temporary ring. After the failed link is reestablished, the auxiliary ring becomes free again. For this topology each station needs to have two transmitter ports and two receiver ports.



a. Physical ring b. Dual ring

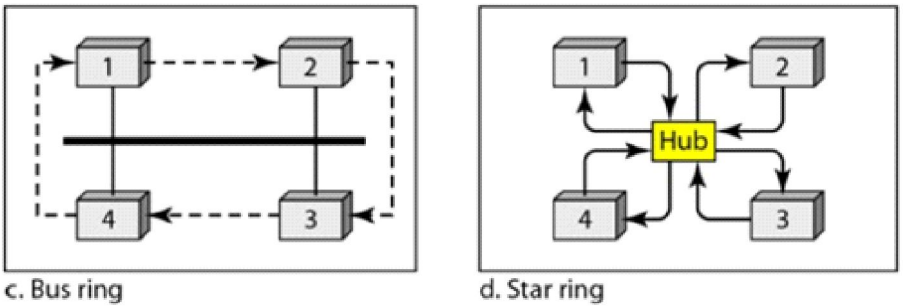


Figure 10.3: Physical topology in token passing access method

The high-speed Token Ring networks called FDDI (Fiber Distributed Data Interface) and CDDI (Copper Distributed Data Interface) use this topology. The bus topology, is also called a token bus, the stations are connected to a single cable called a bus. The hub in a star ring topology acts as the connector. Inside the hub the stations are connected to this ring through the two wire connections. This topology makes the network less prone to failure because if a link goes down, it will be bypassed by the hub and the rest of the stations can operate. Also adding and removing stations from the ring is easier.

10.4.3 Synchronization

The session layer adds checkpoints, or synchronization points, to a stream of data being transmitted. Users can fragment the data stream into pages, and insert synchronization points between each page. When an error occurs, the receiver can resynchronize the state of the session to a previous synchronization point. This requires that the sender hold data as long as may be needed. Synchronization is achieved through the use of sequence numbers. For example, if a system is sending a file of 3000 pages, it is advisable to insert checkpoints after every 100 pages to make sure that each 100 page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 623, the only pages that need to be resent after system recovery are pages 601 to 623. Pages previous to 601 need not be resent.



CHECK YOUR PROGRESS

1. What is inter host communication?
2. What is the major role of session layer?
3. What is a token?
4. Explain the process of synchronization?
5. State the advantage of dual ring architecture.

10.5 REMOTE PROCEDURE CALL

Remote Procedure Call (RPC) provides a different pattern for accessing network services. Instead of accessing the remote services by sending and receiving messages, a client invokes services by making a local procedure call which hides the details of the network communication.

While making a remote procedure call:

(I) The calling environment is put on hold, procedure parameters are shifted across the network to the environment where the procedure is to execute, and the procedures executed there.

(II) When the procedure finishes its execution the results produced are transferred back to the calling environment, where the execution resumes as if returning from a regular procedure call.

RPC is well suited for client-server (e.g., query-response) interaction in which the flow of control passes between the caller and callee. The client and server don't execute at the same time. Instead, the thread of execution jumps from the caller to the callee and then back again. The following steps take place during an RPC:

i. A client invokes a client stub procedure by passing parameters in the normal way. The client stub exists within the client's own address space.

ii. The clients stub marshalls the parameters into a form of message. Marshalling is a process of converting the representation of the parameters into a standard format, and copying each parameter into the message.

iii. The client stub passes the message to the transport layer, which in turn sends it to the remote server machine.

iv. On the server side, the transport layer passes the message to a server stub, which again demarshalls the parameters and calls the desired server routine using the regular procedure call mechanism.

v. When the server completes its procedure, it returns the values to the server stub using a normal procedure call return, that Marshalls the return values into a message. The server stub then hands over the message to the transport layer.
vi. The transport layer then sends the result message back to the client transport layer, which passes the message back to the client stub.
vii. The client stub demarshalls the return parameters and the execution returns to the caller.

10.6 NETBIOS

Network Basic Input/Output System (NetBIOS) was developed by IBM and Sytek as an Application Programming Interface (API) for client software to access LAN resources. From the time of its design, it has become the basis for many other networking applications. This interface specification is used for accessing networking services. It a layer of software which is developed to link a network operating system with specific hardware, which was designed as the network controller for IBM's Network LAN. It is now extended to allow programs written using the NetBIOS interface to operate on the IBM token ring architecture. It offers network applications a set of "hooks" to carry inter-application communication and data transfer. The main intention of NETBIOS is to isolate application programs from any type of hardware dependencies. In a NetBIOS LAN environment, computers are identified on the system by a name. Each computer on the network has a permanent name which is programmed in various different ways. PC's on a NetBIOS LAN communicate either by establishing a session or by using NetBIOS datagram or by broadcast methods. The Sessions are allowed for a larger message to be sent and handle error detection and correction. The communication process is based on one-to-one basis. Datagram and broadcast methods allow one computer to communicate with several other computers at the same time, but are limited to message size. There is no error detection or correction using datagram or broadcast methods. Datagram communication allows communication without having to establish a session. All communication in these environments are presented to NetBIOS in a format called Network Control Blocks (NCB). The allocation of the blocks in memory is dependent on the user program. These NCBs are divided into fields, which are reserved for input and output respectively. NetBIOS is supported on Ethernet, Token Ring, and IBM PC Networks. It supports both broadcasts and multicasting and supports three distinct services: Naming, Session, and Datagram. NetBIOS names are used to

identify resources on a network. Applications use these names to start and end sessions. We can configure a single machine with multiple applications, each of which has a unique NetBIOS name. Each PC that supports an application also has a NetBIOS station name that is user defined or that NetBIOS derives by internal means. NetBIOS consist of 16 alphanumeric characters. The combination of characters must be unique within the entire source routing network.



CHECK YOUR PROGRESS

6. An RPC (remote procedure call) is initiated by the
   a) server
   b) client
   c) both(a)and(b)
   d) none of the mentioned?
7. Remote procedure calls is
   a) inter-process communication
   b) a single process
   c) a single thread
   d) none of the mentioned?
8. RPC allows a computer program to cause a subroutine to execute in
   a) its own address space
   b) another address space
   c) both (a) and (b)
   d) none of the mentioned?
9. NetBIOS is developed by:
   a)Microsoft
   b)IBM
   c)Sun
   d) none of the mentioned?
10. RPC works between two processes. These processes must be
   a) on the same computer.
   b) on different computers connected with a network.
   c) both (a) and (b).
   d) none of the mentioned.



10.7 LET US SUM UP

- Session layer is used for inter host communication. It establishes, manages and terminates sessions between two communicating hosts.
- Session layer is responsible for synchronization and dialogue control.
- To access the channel a packet called token circulates through the ring. The station that holds the token gives that station the right to access and send data.
- In the token-passing method, the stations in a network are organized in a logical ring. Each station has a predecessor and a successor. A special packet called as a token circulates through the ring.
- A token management is necessary to make low priority stations release the token to higher priority stations.
- RPC is well suited for client/server interaction in which the flow of control passes between the caller and callee.
- In RPC instead of accessing the remote services by sending and receiving messages, a client invokes services by making a local procedure call which hides the details of the network communication making a remote procedure call.
- NETBIOS is an application Programming Interface (API) for client software to access LAN resources.
- NetBIOS is supported on Ethernet, Token Ring, and IBM PC Networks.
- NETBIOS supports both broadcasts and multicasting and supports three distinct services: Naming, Session, and Datagram.



10.8 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw- Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.

10.9 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: Inter host communication is a two-way process of communication between communicating entities/hosts in a network for reaching mutual understanding, in which participants not only exchange (encode -decode) information, news, ideas and feelings but also create and share meaning.

Answer 2: Session layer is responsible for synchronization and dialogue control.

Answer 3: A token is a special series of bits that travels around a token-ring network. As the token circulates, computers attached to the network can capture it. The token acts like a ticket, enabling its owner to send a message across the network. There is only one token for each network, so there is no possibility that two computers will attempt to transmit messages at the same time.

Answer 4: The session layer adds checkpoints, or synchronization points, to a stream of data being transmitted. Users can fragment the data stream into pages, and insert synchronization points between each page. When an error occurs, the receiver can resynchronize the state of the session to a previous synchronization point. This requires that the sender hold data as long as may be needed. Synchronization is achieved through the use of sequence numbers.

Answer 5: The dual ring topology uses a second (auxiliary) ring which operates in the reverse direction in comparison with the main ring. The second ring is used for emergency; if one of the links in the main ring fails the system automatically combines the two rings to form a temporary ring. After the failed link is reestablished, the auxiliary ring becomes free again.

Answer 6: (b) Client.

Answer 7: (a) Inter-Process communication.

Answer 8: (b)Another address space.

Answer 9: (b) IBM.

Answer 10: (c) Both (a) and (b)



10.10 MODEL QUESTIONS

Q 1. Explain the role of Session layer.

Q 2. How is Session layer useful to other layer & in OSI model?

Q 3. Explain the token passing mechanism.

Q 4. How does session layer establish, maintain and terminate session?

Q 5. Why is synchronization essential?

Q 6. What are marshalling and demarshalling?

Q 7. Explain with details how RPC works

Q 8. Describe NETBIOS in details.

****

UNIT 11 : PRESENTATION LAYER

UNIT STRUCTURE

11.1 Learning Objectives
11.2 Introduction
11.3 Presentation Layer
    11.3.1 Data Representation
        11.3.1.1 Text
        11.3.1.2 Number
        11.3.1.3 Image
        11.3.1.4 Audio
        11.3.1.5 Video
11.4 Data Conversion
11.5 Translation
11.6 Encryption
    11.6.1 Symmetric and Asymmetric Key Cryptography
    11.6.2 Classical Encryption Techniques
        11.6.2.1 Substitution Cipher
        11.6.2.2 Monoalphabetic Cipher
        11.6.2.3 Polyalphabetic Cipher
    11.6.3 Transposition Cipher
        11.6.3.1 Keyless Cipher
        11.6.3.2 Keyed Cipher
11.7 Compression
    11.7.1 Lossless Compression
        11.7.1.1 Run Length Encoding
        11.7.1.2 Huffman Encoding
        11.7.1.3 Lempel Ziv Encoding
    11.7.2 Lossy Compression
11.8 File Transfer Protocol
    11.8.1 Control connection
    11.8.2 Data connection
11.9 Telnet
11.10 Let Us Sum Up
11.11 Further Reading
11.12 Answers To Check Your Progress
11.13 Model Questions

11.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe the working of the presentation layer.
- explain data formats representation.
- describe various encryption techniques used in the presentation layer
- describe the compression techniques used in this layer.
- describe the difference between symmetric and asymmetric cryptography.
- describe the client/server architecture.
- explain the file transfer protocol.
- describe the concept of TELNET.

11.2 INTRODUCTION

In the previous unit, we have discussed the issues related to session layer. Topics like session layer services, remote procedure call were discussed in detail. In this unit, we will discuss the topics related to presentation layer. Topics like data representation and conversion, translation, encryption etc. are discussed in this unit. Different compression techniques are also covered in this unit. In the next unit, we will explore the issues related to application layer.

11.3 PRESENTATION LAYER

The presentation layer is the sixth layer of OSI Reference Model protocol stack. This layer deals with the syntax and semantics of the information being exchanged between two systems. The presentation layer is liable for the delivery and formatting of data to the application layer for further processing or display. The main responsibilities of the presentation layer include: Translation, Encryption and Compression of data. It also takes care about the data which needs to be sent to the receiver in such a way that the receiver understands the information which is communicated and should be able to use it. Syntax (Language) between two communicating systems can be different; under such circumstances presentation layer acts as a translator.

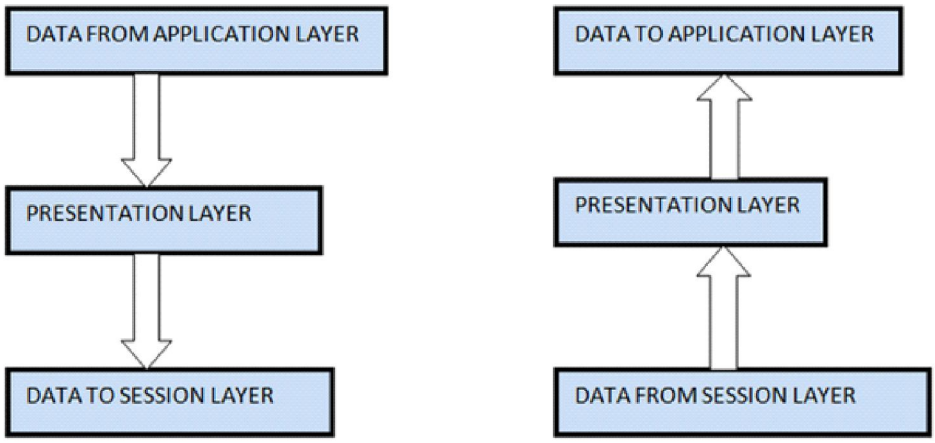


Figure 11.1: Presentation Layer.

For example in Pascal method strings are represented by an integer length field followed by the specified amount of bytes whereas in C/C++ method null-terminated strings are used e.g. "this is a string\0". The main idea is that the application layer should be capable of pointing the data to be moved, and the presentation layer will deal with the rest.

11.3.1 Data Representation

Information which needs to be exchanged between systems can be in the form of text, images, numbers, audio, video etc.

11.3.1.1 Text

In order to carry out data communication, text is represented as a sequence of bits (0's or 1's). Many sets of bit patterns are designed to represent text symbols. Every set is called as a code, and the process by which the symbols are represented is called as coding. The commonly used coding systems are

(i) Unicode system: The coding method in Unicode system uses 32 bits to represent a symbol or character in any language in the world.

(ii) The American Standard Code for Information Interchange (ASCII): It is developed in the United States.It is also referred to as Basic Latin, constituting the first 127 characters in Unicode.

11.3.1.2 Numbers

Numbers can also be represented by bit patterns. A number is directly converted to its binary form to simplify mathematical operations. Binary (base2), Decimal (base10), Hexadecimal (base 16), Octal (base 8) are few numbering system technique.

11.3.1.3 Images

An image is made up of matrix of pixels (picture elements), where each pixel represents a small dot. The resolution of the image depends on the size of the pixel. For example, consider an image which can be divided into 1000 pixels or 10,000 pixels. The second case produces image with better resolution. Since the image is divided into 10,000 pixels more memory is needed to store the image. Each pixel is allotted a bit pattern and the size and the value of the pattern depend on the type of image. If an image is made of only black and white dots like a chessboard, only 1-bit pattern is enough to represent a pixel. There are several techniques to represent color images like RGB and YCM. In RGB, each color is composed of a combination of three primary colors: red, green, and blue. The intensity of every color is measured, and a bit pattern is assigned to it. Whereas in YCM, a color is made of three other primary colors: yellow, cyan, and magenta.

11.3.1.4 Audio

Audio data refers to the recording/ broadcasting of sound or music. Audio data is continuous, and not discrete unlike text, numbers, or images.

11.3.1.5 Video

Video data can be the recording/broadcasting of a picture or movie. Video may either be represented as a continuous entity, or as a combination of images, each arranged with a discrete entity, to deliver the idea of motion.

11.4 DATA CONVERSION

Networks provide the ability for connecting different types of computers, servers and mainframes on the same network that may use different character sets. The presentation layer is in charge of fixing any type of irregularities while making translations transparent between networked systems. If two dissimilar computers are involved in exchanging text, e-mail, then the information needs to be converted to a particular format to carry out smooth exchange of information. For example, characters are 6 bits wide in control Data Corporation (C.D.C) brand computers, ASCII characters are 7 bits in P.C.'s and in IBM mainframes, EBCDIC characters are 8 bits. Several CPUs store integers in 2's complement form, whereas others store in 1's complement form. Therefore there are other representations which must be agreed upon so that two computers having different representation can communicate. The presentation layer executes data conversion algorithms and character translation functions.

(i) Character Code Translation: Here the presentation layer translates the characters from the American standard code for information interchange (ASCII) to the extended binary code decimal interchange code (EBCDIC).

(ii) Data Conversion: The presentation layer performs bit order reversal functions, which converts byte code for a carriage return (CR) to byte code for a carriage return with a line feed (CR/LF) and converts the integer numbers to floating point numbers.

11.5 TRANSLATION

Before transmitting data, the information which is in the form of characters, numbers, audio, video, text etc. should be changed to bit streams. As different computers use different encoding methods the presentation layer is accountable for interoperability between different encoding methods. It translates data between the formats the network and the format of the computer. Data can be encoded in different ways like data compression, or encryption, but the receiver will translate the encoding back to derive the original message. As a translator, presentation layer converts the data which is sent by the application layer of the transmitting node into an acceptable and compatible data format based on the appropriate network protocol and architecture. In the receiving computer, presentation layer translates the

data into an acceptable format which can be used by the application layer. Presentation layer looks after the related issues when the transmitted data must be viewed in a format different from the original format.

11.6 ENCRYPTION

Various applications need to exchange private information over the network. For a secure communication to take place in the network it should be noted that the confidentiality, integrity and authenticity of the information is maintained. Encryption is referred to as a process in which plaintext is converted into cipher text with the help of an encryption algorithm or a key. Decryption is a process of converting the cipher text back into plaintext with the help of a decryption algorithm or a key. Cryptography is a Greek word, which means "secret writing."It is a science of achieving security by encoding the messages into a non-readable form to avoid unauthorized access. When the message reaches the receiver, the hidden message is translated back into its original form. In cryptography plaintext is referred as the original message which the sender wishes to send, it can be in the form of text, picture, number etc. Cipher text is the encoded message which is sent into the network in unreadable form. Cryptography is classified into two categories symmetric key cryptography and asymmetric key cryptography.

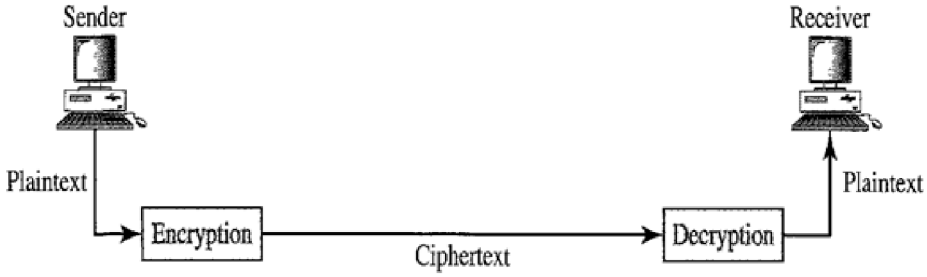


Figure 11.2: Cryptography components

11.6.1 Symmetric and Asymmetric Key Cryptography

In symmetric key cryptography both the communicating parties share the same key for both encryption and decryption.

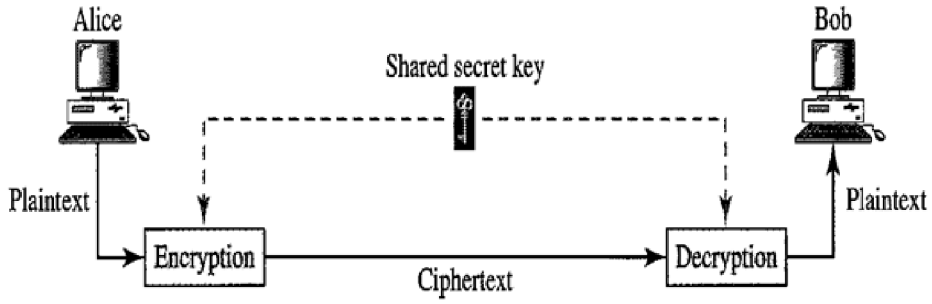


Figure11.3: Symmetric Cryptography.

Data encryption standard (DES) is one of the well-known symmetric key algorithms. In asymmetric cryptography two different keys are used for encryption and decryption. Public key is used for encryption and private key is used for decryption. The receiver sends its public key to all senders. All the senders use the public key to encrypt the data and the receiver upon receiving the encrypted data uses its private key to decrypt the data. It is very difficult to know the private key of the receiver by using the public key. RSA and Diffie-Hellman (Ron Rivest, Adi Shamir, and Len Adleman) algorithm are used in asymmetric key cryptography.

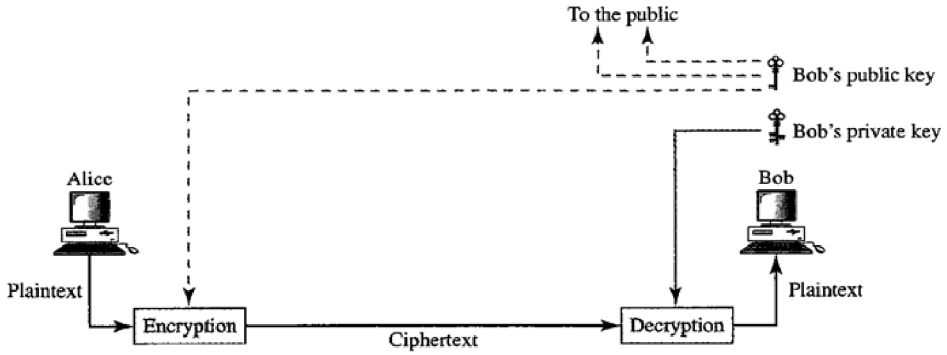


Figure 11.4: Asymmetric Cryptography.

11.6.2 Classical Encryption Technique

Encryption by using symmetric key can be done either by substitution technique, transposition technique, or by combining both the techniques. The traditional symmetric key ciphers can be categorized as shown in Figure 11.5.They are character-oriented and are now obsolete, their goal is to show how modern ciphers evolved from them.

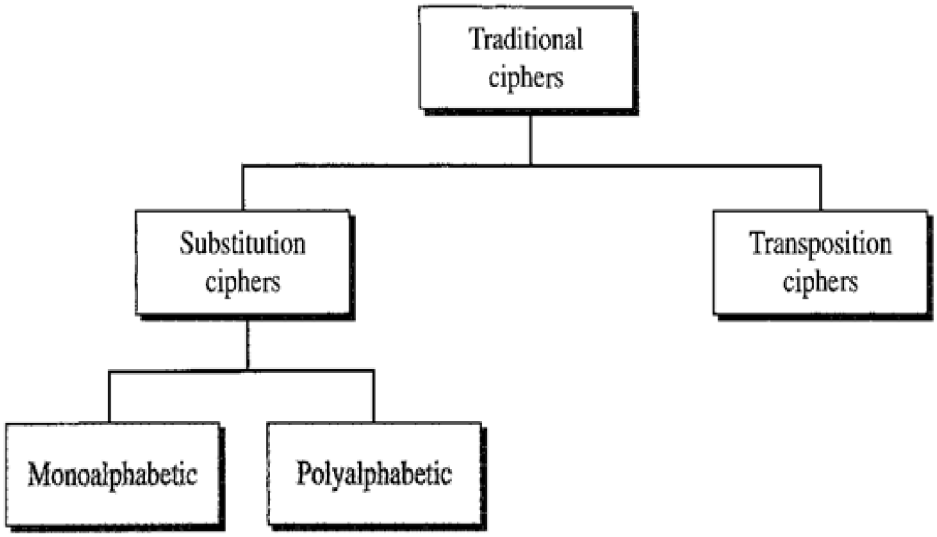


Figure 11.5: Encryption techniques

11.6.2.1 Substitution Cipher

In substitution cipher a symbol in the plaintext is substituted with another symbol. If the symbols in the plain text are alphabetic characters, we replace one character with another for example A can be replaced with D or else if the symbols are digits (0 to 9), one digit can be replaced with another digit. We can replace 3 with 7, and 2 with 6 and so on. It is further categorized into Mono alphabetic and poly alphabetic cipher.

11.6.2.2 Mono alphabetic Cipher

Here the symbol in plaintext has one-to-one relationship with the symbol in the cipher text. For example a symbol in the plaintext is always replaced with the same symbol in the cipher text irrespective of its position in the plaintext. In mono alphabetic cipher the word HELLO can be encrypted as IFMMP.

11.6.2.3 Poly alphabetic Cipher

Here the symbol in plaintext has one-to-many relationship with the symbol in the cipher text. For example same symbol appearing in the plaintext can be replaced with different symbol in the cipher text. In poly alphabetic cipher the word HELLO can be encrypted as AFINE.

11.6.3 Transposition Cipher

In a transposition cipher, there is no substitution of characters. Instead, reorders the character in the plain text(their locations change).It is categorized into keyed transposition cipher and keyless transposition cipher.

` 11.6.3.1 Keyless Cipher

In this cipher no key is used to permute the character in the plaintext. The characters in the plaintext are written in either column-wise or row wise and the cipher text is formed by reading the characters in row wise or column-wise respectively.

11.6.3.2 Keyed Cipher

In this cipher a key is used to permute the character in the plaintext. Key is a mapping between the position of the symbols in the plaintext and cipher text. Let us consider the plaintext: 2 4 1 3 .After encryption reordering of digits take place.2 is encrypted as 1, 4is encrypted with 2,1 is encrypted with 3 and 3 is encrypted with 4.Therefore the cipher text produced will be 1234.Decryption is just the reverse of the encryption process.

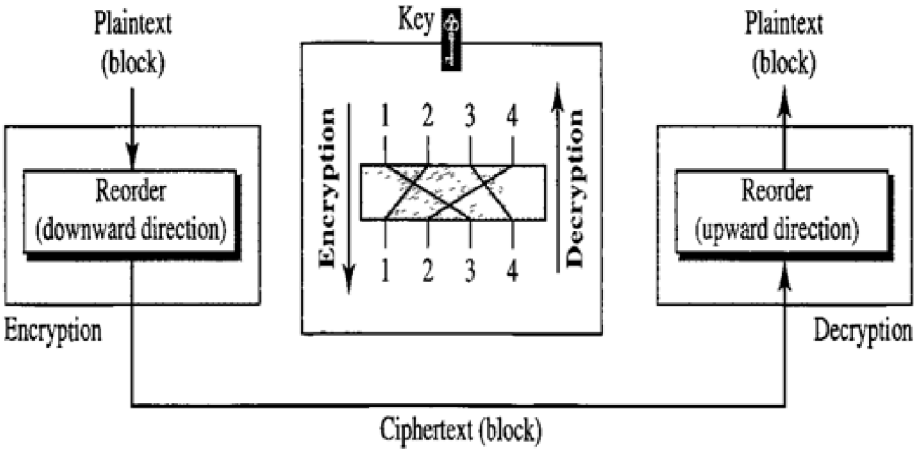


Figure 11.6: Encryption techniques



CHECK YOUR PROGRESS

1. What is presentation layer responsible for?
2. Define encryption and decryption?
3. Differentiate between symmetric and asymmetric cryptography?
4. Differentiate between monoaphapetic and polyalphabetic ciphers?
5. What are the various formats in which information can be transmitted?

11.7 COMPRESSION

Data compression reduces the number of bits to be transmitted and maximizes the use of bandwidth across a network. It also improves the overall throughput of the network. It is important in transmitting multimedia files such as audio, video, text etc. Compression algorithm can be categorized into: Lossless compression and Lossy compression.

11.7.1 Lossless Compression

In Lossless compression there is no loss of data during compression. When the data is decompressed it is found to be exactly same as it was before compression. It is used to compress executable code, text files, and numeric data, because the programs that process such information cannot tolerate errors in the data. Lossless data compression can be implemented using the following algorithm:

a. Run length encoding.
b. Huffman encoding.
c. Lempel Ziv encoding.

11.7.1.1 Run Length Encoding

Run length encoding method replaces the consecutive occurrences of a given symbol with only a single copy of the symbol along with a count as of how many times that symbol occurs. For example, the string AAAAAABBBBCCDDDD would be encoded as A6B4C2D4.Run-length encoding is used in fax

machine. Run length encoding scheme would actually increase the size of a file if there is not a lot of repetition in the data. Hence the compression technique must be carefully used.

11.7.1.2 Huffman Encoding

In Huffman coding fewer numbers of bits are assigned to symbols that occur more frequently and more bits to symbols that appear less often. Let us consider the string "mississippi river" which we want to send. There are 17 letters including one space represented a "_".to transmit this string we require 136 bits (since each letter take 8 bits).

In Huffman coding, the frequency of occurrences of each letter is found out as follows

m →1, i →5, s→4, p→2, r→2, v→1, e→1,→_1.

Sorting the numbers by the frequency of occurrences yields

i5 s4 p2 r2 m1 v1 e1 _1.

Now we add these respective frequencies. We start from the letters with the shortest frequencies and we form the tree with the above data as leaf nodes. So here first we add 'e' and space "_" and their frequencies to get e_2, where 2 is their added frequency. Similarly we add (v1, m1) (p2 , r2) (i5 , s4) to get vm2 , pr4 , is9 as shown in fig 11.7

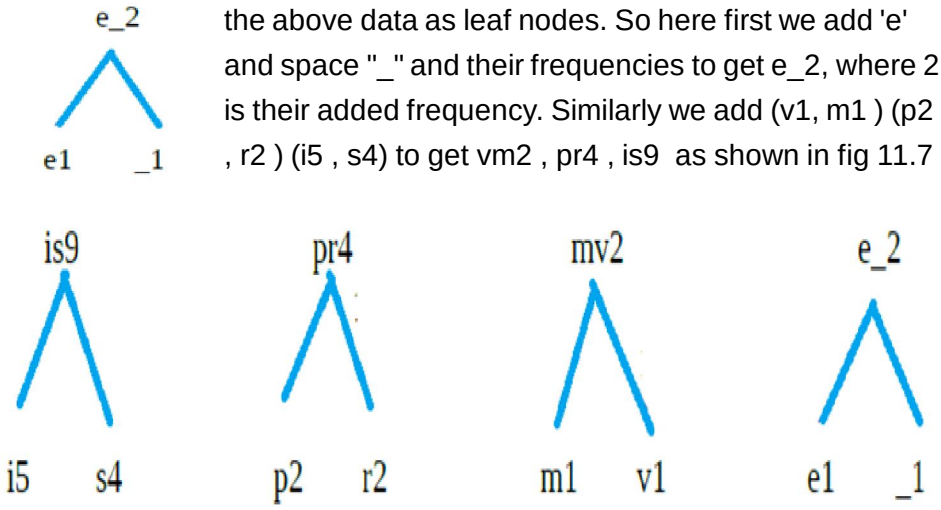


Figure 11.7: Huffman Coding

Similarly we go on adding to the top as shown in the figure below to create the tree. After the construction of the tree we start labeling the left branch with a "0" and right branch with "1"

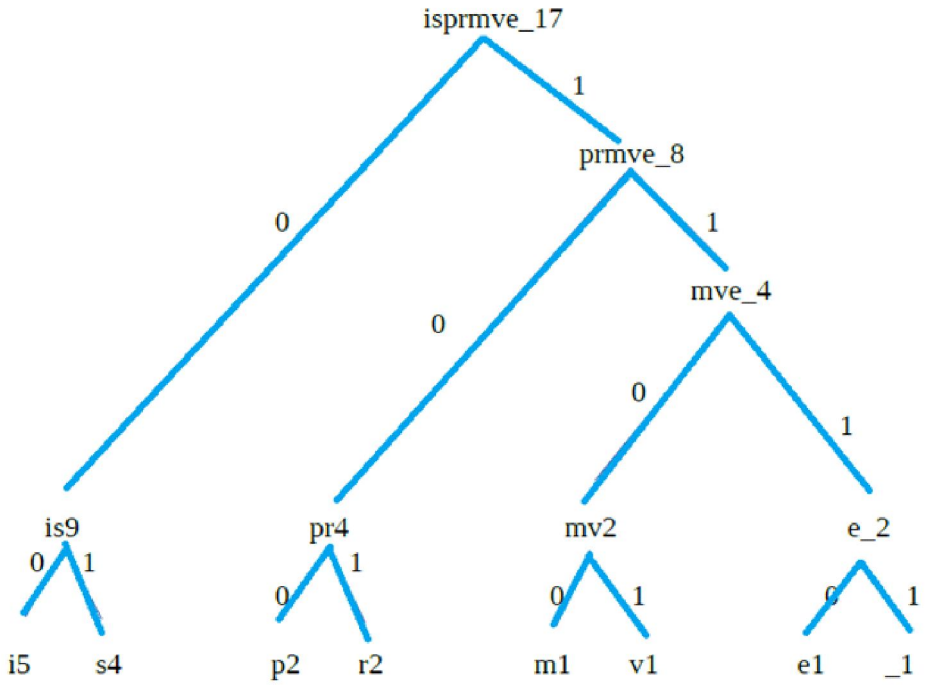


Figure 11.8: Huffman Compression

From the completed tree to get the codes for each letter we have to view the tree from the top to leaf for each respective letter. For e.g. taking the letter i. Starting from the root " isprmve_ " it goes to " is9 " and then reaches leaf i5 through the branch 00. Thus the assigned code for the letter "i" is 00 .Similarly we get codes for other letters by traversing the tree.

m→1→1100 i →5→00 s →4→01 p →2→100 r →2→101 v→1→1101
e→1→1110 _ →1→1111

letter with least frequencies like "v" and "e" needs more bits to represent than letters with higher frequency like "i" and "s" (2 bits each) .After compressing it with Huffman encoding the word "mississippi river" needs only 46 bits to transmit instead of 136 bits.

mi	s	s	i	s	si	p	p	i	_r	i	v	e	r
1100	00	101	00	0101	00	100	100	00	1111	101	00		

11011110101 =46

11.7.1.3 Lempel Ziv Encoding

In this method, a dictionary/table of variable length strings (common phrases) is built. This dictionary covers almost every string that is probable to occur in data. When any of these strings occur in the data, then they are substituted with the corresponding index to the dictionary. In this technique, instead of working with single characters in text data, we treat each word as a string and output the index in the dictionary for that particular word.

11.7.2 Lossy Compression

In Lossy compression the compressed data which is received by the receiver is not same as the data sent i.e. there may be data loss. It is because a lossy algorithm removes information which it cannot restore later. Lossy algorithms achieve better compression ratio than the lossless algorithms. It is used to compress still images, video and audio.

11.8 FILE TRANSFER PROTOCOL (FTP)

Now a day's internet is used mainly for downloading the file. Downloading is nothing but transferring files from a computer on the internet to our own computer. Uploading is a process of transferring the files from our own computer to another computer on the Internet. Most of these files are uploaded and downloaded using the Internet's File Transfer Protocol, usually referred as FTP. FTP is a standardized network protocol which is used to manipulate and exchange files over a TCP/IP based network, like the Internet. It is built on client/server architecture and uses separate connection for data transfer and control information between the client and server applications. The Split-up of commands and data transfer makes FTP more effective. In control connection, at a time we need to transfer only a line of command or a line of response. The data connection needs complex rules due to the variety of data types being transferred. FTP uses TCP ports 21 for the control connection, and port 20 is used for the data connection.

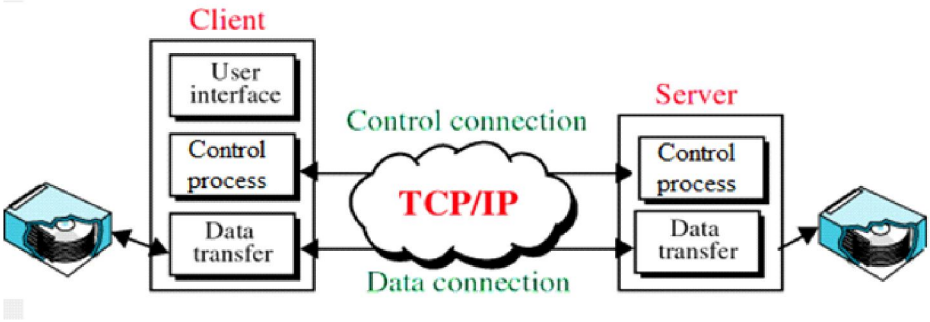


Figure 11.9: File Transfer Control Protocol

The client contains three components: user interface, control process, and data transfer process. The server control process and the server data transfer process are the two components of the server. Connections are built between the control processes and data connection between the data transfer processes. During the entire interactive FTP session the control connection remains connected. The data connection is opened when commands which are used for file transferring is used and then closes after the file is being transferred. When a user starts the FTP session, the control connection opens. Once the control connection is open, the data connection can be opened and can be closed multiple times if numerous files are transferred.

11.8.1 Control Connection

FTP uses the 7-bit ASCII character set. Communication is completed through commands and responses. This method is suitable for the control connection since we send one command or response at a time. Each command or response is the only short line, hence we don't have to worry about file format or file structure

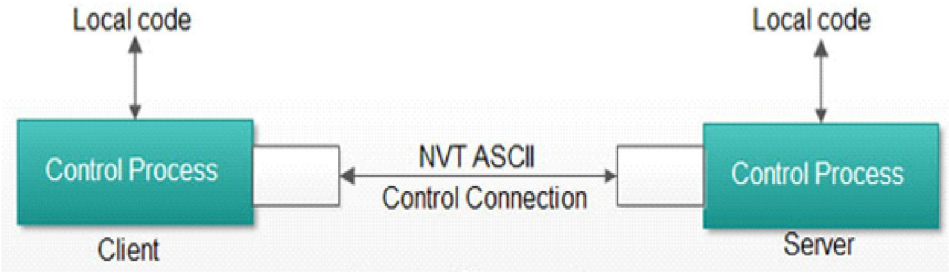


Figure 11.10: Control connection in FTP

11.8.2 Data Connection

Data connection is used for actual data transfer to take place. This connection is built between the data transfer process of client and server. The server uses port 20 for data connection. For each file transferred the data connection is opened and then closed after the transfer is complete. During the file transfer, the client must define the type of file, the data structure and the transmission mode. File transfer in FTP means either of three things:

a) Retrieving a file:When a file is copied from the server to the client. It is done with help of RETR command.

b) Storing a file:When a file is copied from the client to the server. It is done with STOR command.

c) Using LLST command: When a list of directory or file names needs to be sent from the server to the client.

FTP can transfer an ASCII file, EBCDIC file, or image file. The sender converts the file from its definite representation into ASCII characters, and the receiver transforms the ASCII characters to its own representation. If one or both ends of the connection use EBCDIC encoding, the files can be transferred using EBCDIC encoding. FTP can transfer a file using the data structure like: file structure, record structure, and page structure. The file is a continuous stream of bytes in file structure format. In the record structure format, the file is divided into records. It can be used only with text files. In page structure format, the file is divided into pages, where each page has a page number and a page header. The pages can be stored and accessed randomly or sequentially. The transmission mode used in FTP is the stream mode, block mode, and compressed mode. In stream mode data are delivered from FTP to TCP as a constant stream of bytes. TCP is liable for splitting data into segments of applicable size. In block mode; data is delivered from FTP to TCP in blocks or chunks. In the compressed mode, if the file is huge, the data can be compressed. The compression technique used is run-length encoding.

11.9 TErminaLNETwork (TELNET)

TELNET is a client/server application program. It is enable to establish a connection to a remote system in such a manner that the local terminal seems to be a terminal at the remote system. When user logs into a local system, it is called as local log-in. In local log in the user provides input through the keyboard that is accepted by the terminal driver. The terminal driver passes the input characters to the operating system. The operating system, in turn, infers the combination of characters and invokes the required application program whereas in remote log-in when a user wants to access an application program situated on a remote machine, it has to perform remote log-in.

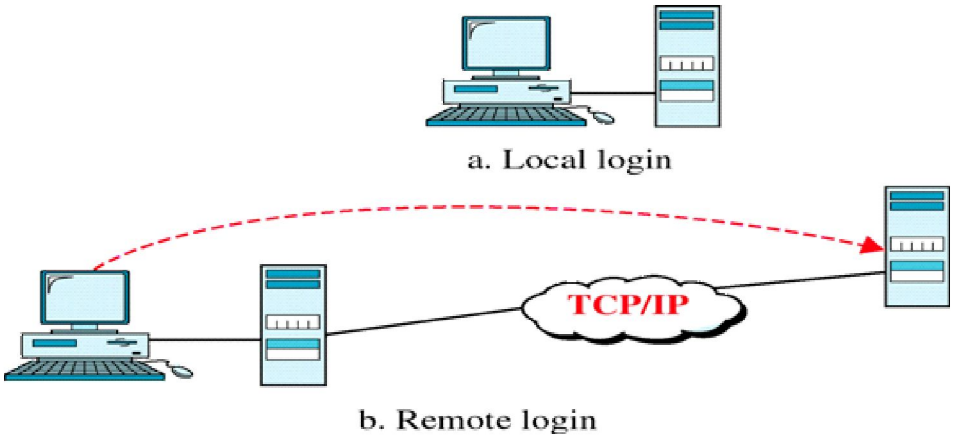


Figure 11.11: Local and remote log in

For remote login TELNET client and server programs are used. The user sends the request to the terminal driver the local operating system receives the characters but does not infer anything from them. The characters which were given as request are sent to the TELNET client, which converts the characters to a universal character set called the Network Virtual Terminal (NVT) characters and provides them to the local TCP/IP protocol stack.NVT format expresses all characters in 8 bits (one byte) form. Telnet is a text-only protocol where we won't be able to see graphics or any fancy images while. The text which is converted in NVT form, travel through the Internet and arrives at the TCP/IP stack in the remote machine. The TELNET server changes the characters to the corresponding characters which can be understood by the remote server. TELNET server receives characters from a terminal driver hence a software which is called as a pseudo terminal

driver is put in operating system of a TELNET server that pretends the characters are coming from a terminal. Then the operating system passes the characters to the suitable application program.

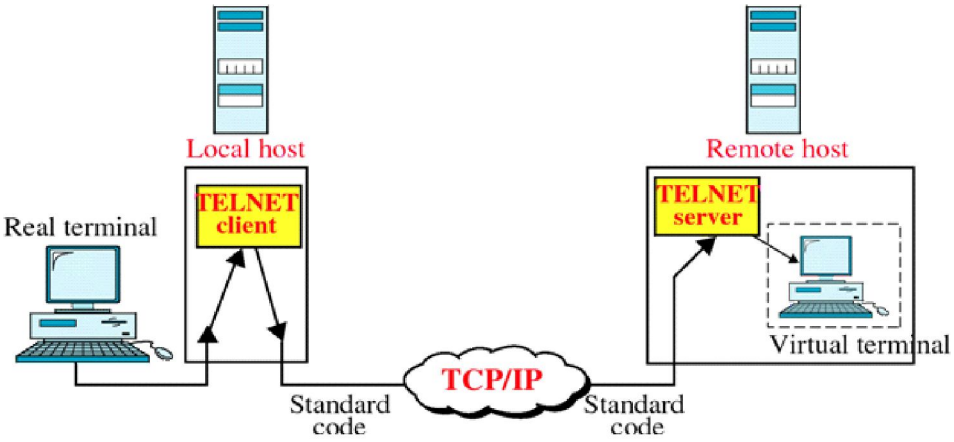


Figure 11.12: TELNET Architecture



CHECK YOUR PROGRESS

6. What do you mean by lossy and lossless compression?
7. Use run length encoding technique to compress wwwwaaadexxxxxx?
8. Compress **go go gophers** using huffman encoding?
9. What is the purpose of FTP?
10. TELNET stands for ----------



11.10 LET US SUM UP

- Data communication is a process of transferring data from one device to another through some form of transmission medium.
- A data communications scheme must transmit data to the accurate destination in an correct and timely manner.
- The major components that create data communications system are the message, sender, receiver, medium, and protocol.
- Text, numbers, images, audio, and video are different forms of information.
- TELNET is a client/server application that allows a user to log into a

remote machine, giving the user right to access the remote terminal.

- TELNET uses NVT system to encode characters on the local system.
- In the server machine, NVT decodes the characters to an acceptable form into the remote machine.
- NVT uses a set of characters for data and a set of characters for control.
- FTP protocol is used for file transfer in the Internet.
- FTP needs two connections for data transfer: a control connection and a data connection.
- FTP uses NVT ASCII for communication between dissimilar systems.
- Prior to any transfer of files, the file type, data structure, and transmission mode are well-defined by the client.
- During connection establishment responses are sent from the server to the client.
- The three types of file transfer are: file copied from the server to the client, file copied from the client to the server and a list of file names is sent from the server to the client.



11.11 FURTHER READING

- Behrouz A. Forouzan, E. (2006). *Data communications and Networking*. Tata McGraw Hill Education.
- William Stallings, E. (2006). *Data and Computer Communications*. Pearson Education India.



11.12 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: Presentation layer deals with syntax and semantics of the information being exchanged between two systems. The responsibility of the presentation layer includes translation, encryption and compression of data.

Answer 2: Encryption is referred as a process in which plaintext is converted into cipher text with the help of an encryption algorithm or a key. Decryption is a process of converting the cipher text

back into plaintext with the help of a decryption algorithm or a key.

Answer 3: Symmetric cryptography:

a) Same key is used for encryption and decryption.
b) Speed of encryption and decryption is fast.
c) Size of the encrypted text is same or less than the original text.

Asymmetric Cryptography:

a) Different keys are used for encryption and decryption.
b) Speed of encryption and decryption is slow.
c) Size of the encrypted text is more than the original text.

Answer 4: Monoalphabetic Cipher: The symbol in plaintext has one-to-one relationship with the symbol in the cipher text. A symbol in the plaintext is always replaced with the same symbol in the cipher text irrespective of its position in the plaintext.

Polyalphabetic cipher symbol in plaintext has one-to-many relationship with the symbol in the cipher text. A same symbol appearing in the plaintext can be replaced with different symbol in the cipher text.

Answer 5: Information which needs to be exchanged between systems can be in the form of text, images, numbers, audio, video etc.

Answer 6: In Lossless compression there is no loss of data during compression. When the data is decompressed it is found to be exactly same as it was before compression. In Lossy compression the compressed data which is received by the receiver is not as same as the data send i.e. there may be data loss

Answer 7: w4a3d1e1x6.

Answer 8: 00 01 100 00 01 100 00 01 1110 1101 101 1111 1100.

Answer 9: FTP is a standardized network protocol which is used to manipulate and exchange files over a TCP/IP based network, like the Internet. It is built on client/server architecture

Answer 10: Terminal network



11.13 MODEL QUESTIONS

Q 1. Explain in detail the role of presentation layer.

Q 2. What is a Transposition cipher?

Q 3. What is the necessity for encrypting the message?

Q 4. Explain lossless compression in details

Q 5. What is the difference between local and remote log-in in TELNET?

Q 6. How are control and data characters distinguished in NVT?.

Q 7. Describe the functions of the two FTP connections.

Q 8. What kinds of file types can FTP transfer?

Q 9. Explain TELNET architecture.

******

UNIT 12: APPLICATION LAYER

UNIT STRUCTURE

12.1 Learning Objectives
12.2 Introduction
12.3 Application Layer
  12.3.1 Client-Server Model
12.4 Domain Name System (DNS)
  12.4.1 DNS Name Space
  12.4.2 Domain Resource Records
  12.4.3 Name Servers
12.5 Simple Mail Transfer Protocol (SMTP)
12.6 File Transfer Protocol (FTP)
12.7 Let Us Sum Up
12.8 Further Reading
12.9 Answers to Check Your Progress
12.10 Model Questions

12.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe the client server model
- describe the Domain Name System
- describe the DNS namespace
- describe name servers and their use
- describe SMTP and FTP protocol

12.2 INTRODUCTION

In the previous unit, we have discussed about the issues related to presentation layer. Topic like data representation and conversion, translation, excryption and compression were discussed in detail. In this unit, we will discuss about the application layer. Topics like client server model, SMTP protocol and file transfer protocol are discussed in detail in this unit. In the next unit we will explore the fundamentals of TCP and UDP protocol.

12.3 APPLICATION LAYER

The Application Layer is the most important and most visible layer in computer networks. Applications reside in this layer and human users interact via those applications through the network. The layers below the application layer are there to provide transport services, but they do not do real work for the users. In this unit, we will study some real network applications. However, even in the application layer there is a need for support protocols, to allow the applications to function. Accordingly, we will look at an important one of these before starting with the applications themselves. The item in question is Domain Name System (DNS), which handles naming within the internet. We will learn about two very simple protocols namely SMTP and FTP which are the earlier ones to come up in the application layer.

Within the Internet, email is delivered by having the sending computer establish a TCP connection to port 25 of the receiving computer. Listening to this port is a mail server that speaks Simple Mail Transfer Protocol (SMTP). This server accepts incoming connections, subject to some security checks, and accepts messages for delivery. The FTP protocol is used to access files by FTP, the internet's file transfer protocol. FTP predates the web and has been in use for more than three decades. The web makes it easy to obtain files placed on numerous FTP servers throughout the world by providing a simple, clickable interface instead of a command-line interface. This improved access to information is one reason for the spectacular growth of the web.

12.3.1 CLIENT SERVER MODEL

A client-server model is the oldest model used to organize a networked application. In this model, a server provides services to clients that exchange information with it. This model is highly asymmetrical: clients send requests and servers perform actions and return responses. It is illustrated in Figure 12.1.

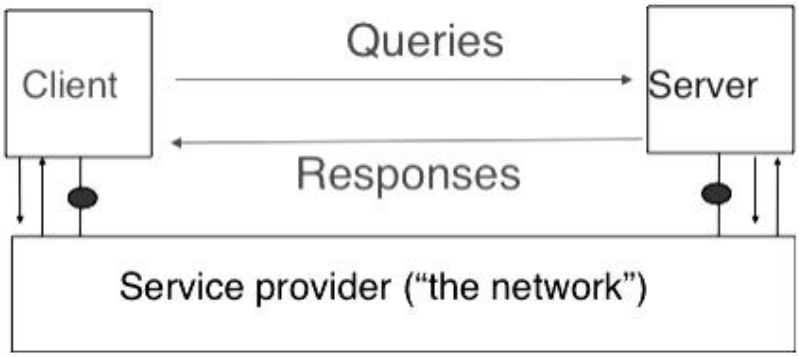


Figure 12.1: The Client-Server model

The client-server model was the first model to be used to develop networked applications. This model comes naturally from the mainframes and minicomputers that were the only networked computers used until the 1980s. A minicomputer is a multi-user system that is used by tens or more users at the same time. Each user interacts with the minicomputer by using a terminal. Those terminals are mainly a screen, a keyboard and a cable directly connected to the minicomputer. There are various types of servers as well as various types of clients. A web server provides information in response to the query sent by its clients. A print server prints documents sent as queries by the client. An email server will forward towards their recipient the email messages sent as queries while a music server will deliver the music requested by the client. From the viewpoint of the application developer, the client and the server applications directly exchange messages (the horizontal arrows labeled queries and responses in the above figure), but in practice these messages are exchanged thanks to the underlying layers (the vertical arrows in the above figure). Networked applications do not exchange random messages. In order to ensure that the server is able to understand the queries sent by a client, and also that the client is able to understand the responses sent by the server, they must both agree on a set of syntactical and semantic rules. These rules define the format of the messages exchanged as well as their ordering. This set of rules is called an application-level protocol. An application-level protocol is similar to a structured conversation between humans. Assume that Alice wants to know the current time but does not have a watch. If Bob passes close by, the following conversation could take place:

- Alice: Hello

- Bob: Hello
- Alice: What time is it?
- Bob: 11:55
- Alice: Thank you
- Bob: You're welcome

Such a conversation succeeds if both Alice and Bob speak the same language. If Alice meets Tchang who only speaks Chinese, she won't be able to ask him the current time. A conversation between humans can be more complex. For example, assume that Bob is a security guard whose duty is to only allow trusted secret agents to enter a meeting room. If all agents know a secret password, the conversation between Bob and Trudy could be as follows:

- Bob: What is the secret password?
- Trudy: 1234
- Bob: This is the correct password, you're welcome

If Alice wants to enter the meeting room but does not know the password, her conversation could be as follows:

- Bob: What is the secret password?
- Alice: 3.1415
- Bob: This is not the correct password.

Human conversations can be very formal, e.g. when soldiers communicate with their hierarchy, or informal such as when friends discuss. Computers that communicate are more akin to soldiers and require well-defined rules to ensure a successful exchange of information. There are two types of rules that define how information can be exchanged between computers:

- Syntactical rules that precisely define the format of the messages that are exchanged. As computers only process bits, the syntactical rules specify how information is encoded as bit strings.
- Organisation of the information flow. For many applications, the flow of information must be structured and there are precedence relationships between the different types of information. In the time example above, Alice must greet Bob before asking for the current time. Alice would not ask for the current time first and greet Bob afterwards. Such precedence relationships exist in networked applications as well. For example, a server must receive a username and a valid password before accepting more complex commands from its clients.

12.4 DOMAIN NAME SYSTEM (DNS)

Although programs theoretically could refer to Web pages, mailboxes, and other resources by using the network (e.g., IP) addresses of the computers on which they are stored, these addresses are hard for people to remember. Also, browsing a company's web pages from 128.111.24.41 means that if the company moves the Web server to a different machine with a different IP address, everyone needs to be told the new IP address. Consequently, high-level, readable names were introduced in order to decouple machine names from machine addresses. In this way, the company's web server might be known as www.cs.washington.edu regardless of its IP address. Nevertheless, since the network itself understands only numerical addresses, some mechanism is required to convert the names to network addresses. In the following sections, we will study how this mapping is accomplished in the internet. Way back in the ARPANET days, there was simply a file, hosts.txt, that listed all the computer names and their IP addresses. Every night, all the hosts would fetch it from the site at which it was maintained. For a network of a few hundred large timesharing machines, this approach worked reasonably well. However, well before many millions of PCs were connected to the Internet, everyone involved with it realized that this approach could not continue to work forever. For one thing, the size of the file would become too large. However, even more importantly, host name conflicts would occur constantly unless names were centrally managed, something unthinkable in a huge international network due to the load and latency. To solve these problems, Domain Name System (DNS) was invented in 1983. It has been a key part of the internet ever since. The essence of DNS is the invention of a hierarchical, domain-based naming scheme and a distributed database system for implementing this naming scheme. It is primarily used for mapping host names to IP addresses but can also be used for other purposes. DNS is defined in RFCs 1034, 1035, 2181, and further elaborated in many others. Very briefly, the way DNS is used is as follows. To map a name onto an IP address, an application program calls a library procedure called the resolver, passing it the name as a parameter. The resolver sends a query containing the name to a local DNS server, which looks up the name and returns a response containing the IP address to the resolver, which then returns it to the caller. The query and response messages are sent as UDP packets. Armed with the IP address, the program can then establish a TCP connection with the host or send it UDP packets.

12.4.1 The DNS Name Space

The Domain Name System is a distributed database that allows to map names on IP addresses. Managing a large and constantly changing set of names is a nontrivial problem. In the postal system, name management is done by requiring letters to specify (implicitly or explicitly) the country, state or province, city, street address, and name of the addressee. Using this kind of hierarchical addressing ensures that there is no confusion between segments of addresses like house, street, district etc. DNS works the same way. For the Internet, the top of the naming hierarchy is managed by an organization called Internet Corporation for Assigned Names and Numbers (ICANN). ICANN was created for this purpose in 1998, as part of the maturing of the Internet to a worldwide, economic concern. Conceptually, the Internet is divided into over 250 top-level domains, where each domain covers many hosts. Each domain is partitioned into subdomains, and these are further partitioned, and so on. All these domains can be represented by a tree, as shown in Figure 12.2. The leaves of the tree represent domains that have no subdomains. A leaf domain may contain a single host, or it may represent a company and contain thousands of hosts.

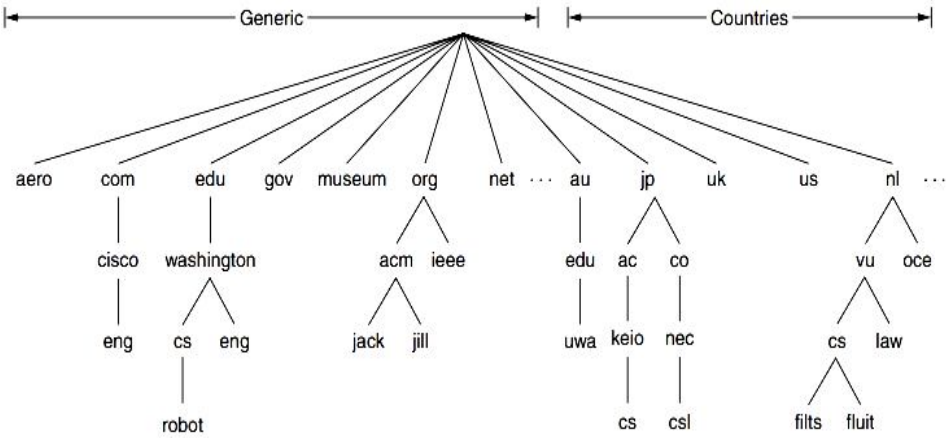


Figure 12.2: A portion of the Internet domain name space

The top-level domains come in two flavors: generic and countries. The generic domains, listed in Figure 12.3, include original domains from the 1980s and domains introduced via applications to ICANN. Other generic top-level domains will be added in the future. The country domains include one entry for every country, as defined in

ISO 3166. Internationalized country domain names that use non-Latin alphabets were introduced in 2010. These domains let people name hosts in Arabic, Cyrillic, Chinese, or other languages. Getting a second-level domain, such as name-of-company.com, is easy. The top-level domains are run by registrars appointed by ICANN. Getting a name merely requires going to a corresponding registrar (for com in this case) to check if the desired name is available and not somebody else's trademark. If there are no problems, the requester pays the registrar a small annual fee and gets the name. However, as the Internet has become more complex, registration of names have became difficult. Problems also arise with respect to content. Many considerations like country, profession, commercial etc come while registering domains. Some of the domains self-organise, while others have restrictions on who can obtain a name, as noted in Figure 12.3.

Domain	Intended use	Start date	Restricted?
com	Commercial	1985	No
edu	Educational institutions	1985	Yes
gov	Government	1985	Yes
int	International organizations	1988	Yes
mil	Military	1985	Yes
net	Network providers	1985	No
org	Non-profit organizations	1985	No
aero	Air transport	2001	Yes
biz	Businesses	2001	No
coop	Cooperatives	2001	Yes
info	Informational	2002	No
museum	Museums	2002	Yes
name	People	2002	No
pro	Professionals	2002	Yes
cat	Catalan	2005	Yes
jobs	Employment	2005	Yes
mobi	Mobile devices	2005	Yes
tel	Contact details	2005	Yes
travel	Travel industry	2005	Yes
xxx	Sex industry	2010	No

Figure 12.3: Generic top-level domains

The practice of registering a domain only to turn around and sell it off to an interested party at a much higher price even has a name. It is called cybersquatting. Many companies that were slow off the mark when the Internet era began found their obvious domain names already taken when they tried to acquire them. In general, as long as no trademarks are being violated and no fraud is involved, it is first-come, first-served with names. Nevertheless, policies to resolve naming disputes are still being refined. Each domain is named by the path upward from it to the (unnamed) root. The components are separated by periods. Example, eng.cisco.com represents engineering department of CISCO.

Domain names can be either absolute or relative. An absolute domain name always ends with a period (e.g., eng.cisco.com), whereas a relative one does not. Relative names have to be interpreted in some context to uniquely determine their true meaning. In both cases, a named domain refers to a specific node in the tree and all the nodes under it. Domain names are case-insensitive, so edu, Edu, and EDU mean the same thing. Component names can be up to 63 characters long, and full path names must not exceed 255 characters.

In principle, domains can be inserted into the tree in either generic or country domains. For example, cs.washington.edu could equally well be listed under the us country domain as cs.washington.wa.us. In practice, however, most organizations in the United States are under generic domains, and most outside the United States are under the domain of their country. There is no rule against registering under multiple top-level domains. Large companies often do so (e.g., sony.com, sony.net, and sony.nl). Each domain controls how it allocates the domains under it. For example, Japan has domains ac.jp and co.jp that mirror edu and com. The Netherlands does not make this distinction and puts all organizations directly under nl. Thus, all three of the following are university computer science departments:

1. cs.washington.edu (University of Washington, in the U.S.).
2. cs.vu.nl (Vrije Universiteit, in The Netherlands).
3. cs.keio.ac.jp (Keio University, in Japan).

To create a new domain, permission is required of the domain in which it will be included. For example, if a VLSI group is started at the University of Washington and wants to be known as vlsi.cs.washington.edu, it has to get permission from whoever

manages cs.washington.edu. Similarly, if a new university is chartered, say, the University of Northern South Dakota, it must ask the manager of the edu domain to assign it unsd.edu (if that is still available). In this way, name conflicts are avoided and each domain can keep track of all its subdomains. Once a new domain has been created and registered, it can create subdomains, such as cs.unsd.edu, without getting permission from anybody higher up the tree. Naming follows organizational boundaries, not physical networks. For example, if the computer science and electrical engineering departments are located in the same building and share the same LAN, they can nevertheless have distinct domains. Similarly, even if computer science is split over Babbage Hall and Turing Hall, the hosts in both buildings will normally belong to the same domain.



CHECK YOUR PROGRESS

1. Fill in the blanks:

(a) In the _____ model a server provides services to clients that exchange information with it.

(b) The client-server model was the first model to be used to develop _____.

(c) The DNS is primarily used for mapping _____ to _____.

(d) For the Internet, the top of the naming hierarchy is managed by an organization called _____.

(e) A leaf domain may contain a _____ host, or it may represent a company and contain thousands of _____.

(f) The top-level domains come in two flavors of _____ and _____.

(g) Domain names can be either _____ or _____.

12.4.2 Domain Resource Records

Every domain, whether it is a single host or a top-level domain, can have a set of resource records associated with it. These records are the DNS database. For a single host, the most common resource record is just its IP address, but many other kinds of resource records

also exist. When a resolver gives a domain name to DNS, what it gets back are the resource records associated with that name. Thus, the primary function of DNS is to map domain names onto resource records.

A resource record is a five-tuple. Although they are encoded in binary for efficiency, in most expositions resource records are presented as ASCII text, one line per resource record. The format we will use is as follows:

Domain_name Time_to_live Class Type Value

The Domain_name tells the domain to which this record applies. Normally, many records exist for each domain and each copy of the database holds information about multiple domains. This field is thus the primary search key used to satisfy queries. The order of the records in the database is not significant. The Time_to_live field gives an indication of how stable the record is. Information that is highly stable is assigned a large value, such as 86400 (the number of seconds in 1 day). Information that is highly volatile is assigned a small value, such as 60 (1 minute).

The third field of every resource record is the Class. For Internet information, it is always IN. For non-Internet information, other codes can be used, but in practice these are rarely seen.

The Type field tells what kind of record this is. There are many kinds of DNS records. The important types are listed in Figure 12.4.

An SOA record provides the name of the primary source of information about the name server's zone, the email address of its administrator, a unique serial number, and various flags and timeouts. The most important record type is the A (Address) record. It holds a 32-bit IPv4 address of an interface for some host. The corresponding AAAA, or "quadA," record holds a 128-bit IPv6 address. Every Internet host must have at least one IP address so that other machines can communicate with it. Some hosts have two or more network interfaces, in which case they will have two or more type A or AAAA resource records. Consequently, DNS can return multiple addresses for a single name.

A common record type is the MX record. It specifies the name of the host prepared to accept email for the specified domain. It is used because not every machine is prepared to accept email. If someone wants to send email to, for example, bill@microsoft.com, the sending

host needs to find some mail server located at microsoft.com that is willing to accept email. The MX record can provide this information. Another important record type is the NS record. It specifies a name server for the domain or subdomain. This is a host that has a copy of the database for a domain. It is used as part of the process to look up names.

Type	Meaning	Value
SOA	Start of authority	Parameters for this zone
A	IPv4 address of a host	32-Bit integer
AAAA	IPv6 address of a host	128-Bit integer
MX	Mail exchange	Priority, domain willing to accept email
NS	Name server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
SPF	Sender policy framework	Text encoding of mail sending policy
SRV	Service	Host that provides it
TXT	Text	Descriptive ASCII text

Figure 12.4: The principal DNS resource record types

CNAME records allow aliases to be created. For example, a person familiar with Internet naming in general and wanting to send a message to user paul in the computer science department at M.I.T. might guess that paul@cs.mit.edu will work. Actually, this address will not work, because the domain for M.I.T.'s computer science department is csail.mit.edu. However, as a service to people who do not know this, M.I.T. could create a CNAME entry to point people and programs in the right direction. An entry like this one might do the job:

cs.mit.edu 86400 IN CNAME csail.mit.edu

Like CNAME, PTR points to another name. However, unlike CNAME, which is really just a macro definition (i.e., a mechanism to replace one string by another), PTR is a regular DNS data type whose interpretation depends on the context in which it is found. In practice, it is nearly always used to associate a name with an IP address to allow lookups of the IP address and return the name of the corresponding machine. These are called reverse lookups. SRV is a newer type of record that allows a host to be identified for a

given service in a domain. For example, the Web server for cs.washington.edu could be identified as cockatoo.cs.washington.edu. This record generalizes the MX record that performs the same task but it is just for mail servers. SPF is also a newer type of record. It lets a domain encode information about what machines in the domain will send mail to the rest of the Internet. This helps receiving machines check that mail is valid. If mail is being received from a machine that calls itself dodgy but the domain records say that mail will only be sent out of the domain by a machine called smtp, chances are that the mail is forged junk mail. Last on the list, TXT records were originally provided to allow domains to identify themselves in arbitrary ways. Nowadays, they usually encode machine-readable information, typically the SPF information.

Finally, we have the value field. This field can be a number, a domain name, or an ASCII string. The semantics depend on the record type. A short description of the value fields for each of the principal record types is given in Figure 12.4.

For an example of the kind of information one might find in the DNS database of a domain of Figure 12.5. This figure depicts part of a (hypothetical) database for the cs.vu.nl domain shown in Figure 12.2. The database contains seven types of resource records.

The first non-comment line of Figure 12.5 gives some basic information about the domain, which will not concern us further. Then come two entries giving the first and second places to try to deliver email sent to person@cs.vu.nl. The zephyr (a specific machine) should be tried first. If that fails, the top should be tried as the next choice. The next line identifies the name server for the domain as star.

After the blank line (added for readability) come lines giving the IP addresses for the star, zephyr, and top. These are followed by an alias, www.cs.vu.nl, so that this address can be used without designating a specific machine. Creating this alias allows cs.vu.nl to change its World Wide Web server without invalidating the address people use to get to it. A similar argument holds for ftp.cs.vu.nl.

The section for the machine flits lists two IP addresses and three choices are given for handling email sent to flits.cs.vu.nl. First choice is naturally the flits itself, but if it is down, the zephyr and top are the second and third choices.

```
; Authoritative data for cs.vu.nl
cs.vu.nl.       86400   IN   SOA      star boss (9527,
cs.vu.nl.       86400   IN   MX       1 zephyr
cs.vu.nl.       86400   IN   MX       2 top
cs.vu.nl.       86400   IN   NS       star

star            86400   IN   A        130.37.56.205
zephyr          86400   IN   A        130.37.20.10
top             86400   IN   A        130.37.20.11
www             86400   IN   CNAME    star.cs.vu.nl
ftp             86400   IN   CNAME    zephyr.cs.vu.nl

flits           86400   IN   A        130.37.16.112
flits           86400   IN   A        192.31.231.165
flits           86400   IN   MX       1 flits
flits           86400   IN   MX       2 zephyr
flits           86400   IN   MX       3 top

rowboat                 IN   A        130.37.56.201
                        IN   MX       1 rowboat
                        IN   MX       2 zephyr

little-sister           IN   A        130.37.62.23

laserjet                IN   A        192.31.231.216
```

Figure 12.5: A portion of a possible DNS database for cs.vu.nl

The next three lines contain a typical entry for a computer, in this case, rowboat.cs.vu.nl. The information provided contains the IP address and the primary and secondary mail drops. Then comes an entry for a computer that is not capable of receiving mail itself, followed by an entry that is likely for a printer that is connected to the Internet.

12.4.3 Name Servers

In theory at least, a single name server could contain the entire DNS database and respond to all queries about it. In practice, this server would be so overloaded as to be useless. Furthermore, if it ever went down, the entire Internet would be crippled. To avoid the problems associated with having only a single source of information, the DNS name space is divided into non-overlapping zones. One possible way to divide the namespace of Figure 12.2 is shown in Figure 12.6. Each circled zone contains some part of the tree.

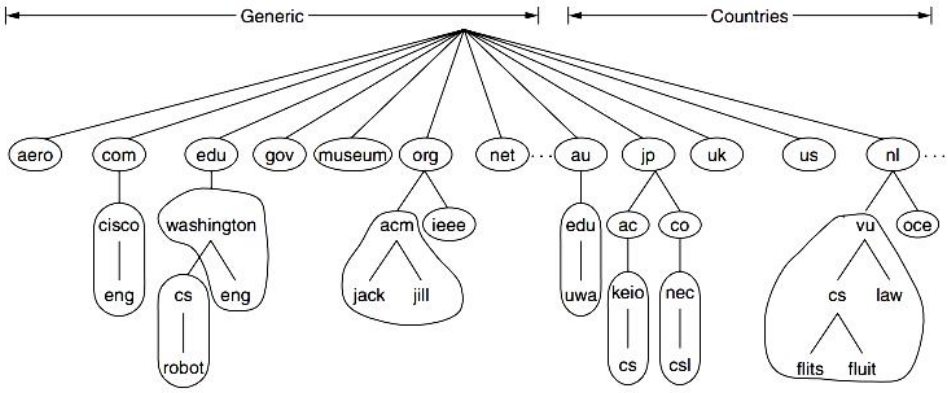


Figure 12.6: Part of the DNS name space divided into (circled) zones

Where the zone boundaries are placed within a zone is up to that zone's administrator. This decision is made in large part based on how many name servers are desired, and where. For example, in Figure 12.6, the University of Washington has a zone for washington.edu that handles eng.washington.edu but does not handle cs.washington.edu. That is a separate zone with its own name servers. Such a decision might be made when a department such as English does not wish to run its own name server, but a department such as Computer Science does.

Each zone is also associated with one or more name servers. These are hosts that hold the database for the zone. Normally, a zone will have one primary name server, which gets its information from a file on its disk, and one or more secondary name servers, which get their information from the primary name server. To improve reliability, some of the name servers can be located outside the zone. The process of looking up a name and finding an address is called name resolution. When a resolver has a query about a domain name, it passes the query to a local name server. If the domain being sought falls under the jurisdiction of the name server, such as top.cs.vu.nl falling under cs.vu.nl, it returns the authoritative resource records. An authoritative record is one that comes from the authority that manages the record and is thus always correct. Authoritative records are in contrast to cached records, which may be out of date. What happens when the domain is remote, such as when flits.cs.vu.nl wants to find the IP address of robot.cs.washington.edu at UW (University of Washington)? In this case, and if there is no cached information about the domain available locally, the name server begins a remote query. This query follows the process shown in Figure 12.7. Step 1 shows

the query that is sent to the local name server. The query contains the domain name sought, the type (A), and the class (IN).

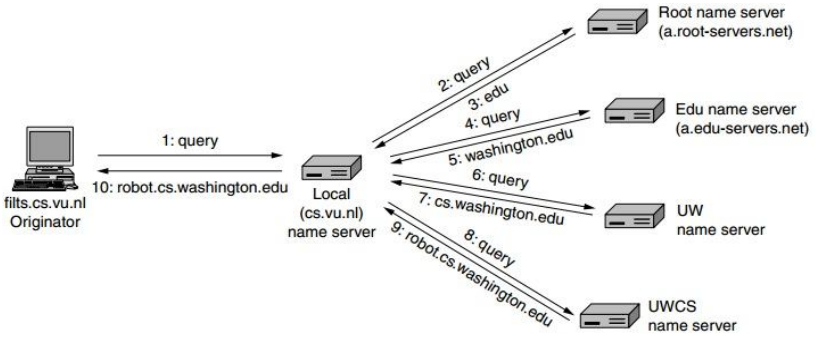


Figure 12.7: Example of a resolver looking up a remote name in 10 steps

The next step is to start at the top of the name hierarchy by asking one of the root name servers. These name servers have information about each top-level domain. This is shown as step 2 in Figure 12.7. To contact a root server, each name server must have information about one or more root name servers. This information is normally present in a system configuration file that is loaded into the DNS cache when the DNS server is started. It is simply a list of NS records for the root and the corresponding A records. There are 13 root DNS servers, unimaginatively called a- root-servers.net through m.root-servers.net. Each root server could logically be a single computer. However, since the entire Internet depends on the root servers, they are powerful and heavily replicated computers. Most of the servers are present in multiple geographical locations and reached using any cast routing, in which a packet is delivered to the nearest instance of a destination address.

The root name server is unlikely to know the address of a machine at UW, and probably does not know the name server for UW either. But it must know the name server for the edu domain, in which cs.washington.edu is located. It returns the name and IP address for that part of the answer in step 3. The local name server then continues its quest. It sends the entire query to the edu name server (a.edu-servers.net). That name server returns the name server for UW. This is shown in steps 4 and 5. Closer now, the local name server sends the query to the UW name server (step 6). If the domain name being sought was in the English department, the answer would be found, as the UW zone includes the English department. But the

Computer Science department has chosen to run its own name server. The query returns the name and IP address of the UW Computer Science name server (step 7).

Finally, the local name server queries the UW Computer Science name server (step 8). This server is authoritative for the domain cs.washington.edu, so it must have the answer. It returns the final answer (step 9), which the local name server forwards as a response to flits.cs.vu.nl (step 10). The name has been resolved.

You can explore this process using standard tools such as the dig program that is installed on most UNIX systems. For example, typing

dig@a.edu-servers.net robot.cs.washington.edu

will send a query for robot.cs.washington.edu to the a.edu-servers.net name server and print out the result. This will show you the information obtained in step 4 in the example above, and you will learn the name and IP address of the UW name servers.

There are three technical points to discuss about this long scenario. First, two different query mechanisms are at work in Figure 12.7. When the host flits.cs.vu.nl sends its query to the local name server, that name server handles the resolution on behalf of flits until it has the desired answer to return. It does not return partial answers. They might be helpful, but they are not what the query was seeking. This mechanism is called a recursive query.

On the other hand, the root name server (and each subsequent name server) does not recursively continue the query for the local name server. It just returns a partial answer and moves on to the next query. The local name server is responsible for continuing the resolution by issuing further queries. This mechanism is called an iterative query. One name resolution can involve both mechanisms, as this example showed. A recursive query may always seem preferable, but many name servers (especially the root) will not handle them. They are too busy. Iterative queries put the burden on the originator. The rationale for the local name server supporting a recursive query is that it is providing a service to hosts in its domain. Those hosts do not have to be configured to run a full name server, just to reach the local one.

The second point is caching. All of the answers, including all the partial answers returned, are cached. In this way, if another cs.vu.nl host queries for robot.cs.washington.edu the answer will already be known. Even better, if a host queries for a different host in the same

domain, say galah.cs.washington.edu, the query can be sent directly to the authoritative name server. Similarly, queries for other domains in washington.edu can start directly from the washington.edu name server. Using cached answers greatly reduces the steps in a query and improves performance. The original scenario we sketched is in fact the worst case that occurs when no useful information is cached. However, cached answers are not authoritative, since changes made at cs.washington.edu will not be propagated to all the caches in the world that may know about it. For this reason, cache entries should not live too long. This is the reason that the Time to live field is included in each resource record. It tells remote name servers how long to cache records. If a certain machine has had the same IP address for years, it may be safe to cache that information for 1 day. For more volatile information, it might be safer to purge the records after a few seconds or a minute.

The third issue is the transport protocol that is used for the queries and responses. It is UDP. DNS messages are sent in UDP packets with a simple format for queries, answers, and name servers that can be used to continue the resolution. We will not go into the details of this format. If no response arrives within a short time, the DNS client repeats the query, trying another server for the domain after a small number of retries. This process is designed to handle the case of the server being down as well as the query or response packet getting lost. A 16-bit identifier is included in each query and copied to the response so that a name server can match answers to the corresponding query, even if multiple queries are outstanding at the same time. Even though its purpose is simple, it should be clear that DNS is a large and complex distributed system that is comprised of millions of name servers that work together. It forms a key link between human-readable domain names and the IP addresses of machines. It includes replication and caching for performance and reliability and is designed to be highly robust.

There is also application demand to use names in more flexible ways, for example, by naming content and resolving to the IP address of a nearby host that has the content. This fits the model of searching for and downloading a movie. It is the movie that matters, not the computer that has a copy of it, so all that is wanted is the IP address of any nearby computer that has a copy of the movie. Content distribution networks are one way to accomplish this mapping.

12.5 SIMPLE MAIL TRANSFER PROTOCOL (SMTP)

Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks. SMTP was first defined by RFC 821 (1982, eventually declared STD 10), and last updated by RFC 5321 (2008) which includes the Extended SMTP (ESMTP) additions, and is the protocol in widespread use today. SMTP uses TCP port 25. The protocol for new submissions (MSA) is effectively the same as SMTP, but it uses port 587 instead. SMTP connections secured by SSL are known by the shorthand SMTPS, though SMTPS is not a protocol in its own right. While electronic mail servers and other mail transfer agents use SMTP to send and receive mail messages, user-level client mail applications typically use SMTP only for sending messages to a mail server for relaying. For receiving messages, client applications usually use either the Post Office Protocol (POP) or the Internet Message Access Protocol (IMAP) or a proprietary system (such as Microsoft Exchange or Lotus Notes/Domino) to access their mail box accounts on a mail server. Various forms of one-to-one electronic messaging were used in the 1960s. People communicated with one another using systems developed for specific mainframe computers. As more computers were interconnected, especially in the US Government's ARPANET, standards were developed to allow users of different systems to e-mail one another. SMTP grew out of these standards developed during the 1970s. SMTP can trace its roots to two implementations described in 1971: the Mail Box Protocol, whose implementation has been disputed, but is discussed in RFC 196 and other RFCs, and the SNDMSG program, which, according to RFC 2235, Ray Tomlinson of BBN invented for TENEX computers to send mail messages across the ARPANET. Fewer than 50 hosts were connected to the ARPANET at this time. Further implementations include FTP Mail and Mail Protocol, both from 1973. Development work continued throughout the 1970s, until the ARPANET converted into the modern Internet around 1980. Jon Postel then proposed a Mail Transfer Protocol (MTP) in 1980 that began to remove the mail's reliance on FTP. SMTP was published as RFC 788 in November 1981, also by Postel. The SMTP standard was developed around the same time as Usenet, a one-to-many communication network with some similarities. SMTP became widely used in the early 1980s. At the time, it was a complement to Unix to Unix Copy Program (UUCP) mail, which was better suited for handling e-mail transfers between machines that were

intermittently connected. SMTP, on the other hand, works best when both the sending and receiving machines are connected to the network all the time. Both use a store and forward mechanism and are examples of push technology. Though Usenet's newsgroups are still propagated with UUCP between servers, UUCP mail has virtually disappeared along with the "bang paths" it used as message routing headers.

Released with 4.1cBSD, right after RFC 788, Sendmail was one of the first (if not the first) mail transfer agents to implement SMTP. Over time, as BSD Unix became the most popular operating system on the Internet, sendmail became the most common MTA (mail transfer agent). Some other popular SMTP server programs include Postfix, qmail, Novell GroupWise, Exim, Novell NetMail, Microsoft Exchange Server, Sun Java System Messaging Server. Message submission (RFC 2476) and SMTP-AUTH (RFC 2554) were introduced in 1998 and 1999, both describing new trends in e-mail delivery. Originally, SMTP servers were typically internal to an organization, receiving mail for the organization from the outside, and relaying messages from the organization to the outside. But as time went on, SMTP servers (mail transfer agents), in practice, were expanding their roles to become message submission agents for Mail user agents, some of which were now relaying mail from the outside of an organization. This issue, a consequence of the rapid expansion and popularity of the World Wide Web, meant that SMTP had to include specific rules and methods for relaying mail and authenticating users to prevent abuses such as relaying of unsolicited e-mail (spam). Work on message submission (RFC 2476) was originally started because popular mail servers would often rewrite mail in an attempt to fix problems in it, for example, adding a domain name to an unqualified address. This behavior is helpful when the message being fixed is an initial submission, but dangerous and harmful when the message originated elsewhere and is being relayed. Cleanly separating mail into submission and relay was seen as a way to permit and encourage rewriting submissions while prohibiting rewriting relay. As spam became more prevalent, it was also seen as a way to provide authorization for mail being sent out from an organization, as well as traceability. This separation of relay and submission quickly became a foundation for modern email security practices.

As this protocol started out purely as ASCII text-based, it did not deal well with binary files, or characters in many non-English languages. Standards such as Multipurpose Internet Mail Extensions (MIME) were developed to

encode binary files for transfer through SMTP. Mail transfer agents (MTAs) developed after Sendmail also tended to be implemented 8-bit-clean, so that the alternate "just send eight" strategy could be used to transmit arbitrary text data (in any 8-bit ASCII-like character encoding) via SMTP. Mojibake was still a problem due to differing character set mappings between vendors, although the email addresses themselves still allowed only ASCII. 8-bit-clean MTAs today tend to support the 8 BITMIME extension, permitting binary files to be transmitted almost as easily as plain text. Recently the SMTPUTF8 extension was created to support UTF-8 text, allowing international content and addresses in non-Latin scripts like Cyrillic or Chines.

SMTP V/S Mail Retrieval

SMTP is a delivery protocol only. In normal use, mail is "pushed" to a destination mail server (or next-hop mail server) as it arrives. Mail is routed based on the destination server, not the individual user(s) to which it is addressed. Other protocols, such as the Post Office Protocol (POP) and the Internet Message Access Protocol (IMAP) are specifically designed for use by individual users retrieving messages and managing mail boxes. To permit an intermittently-connected mail server to pull messages from a remote server on demand, SMTP has a feature to initiate mail queue processing on a remote server. POP and IMAP are unsuitable protocols for relaying mail by intermittently-connected machines; they are designed to operate after final delivery, when information critical to the correct operation of mail relay (the "mail envelope") has been removed.

SMTP transport example

A typical example of sending a message via SMTP to two mailboxes (alice and theboss) located in the same mail domain (example.com or localhost.com) is reproduced in the following session exchange. (In this example, the conversation parts are prefixed with S: and C:, for server and client, respectively; these labels are not part of the exchange.) After the message sender (SMTP client) establishes a reliable communications channel to the message receiver (SMTP server), the session is opened with a greeting by the server, usually containing its fully qualified domain name (FQDN), in this case smtp.example.com. The client initiates its dialog by responding with a HELO command identifying itself in the command's parameter with its FQDN (or an address literal if none is available).

S: 220 smtp.example.com ESMTP Postfix

C: HELO relay.example.org

S: 250 Hello relay.example.org, I am glad to meet you

C: MAIL FROM:<bob@example.org>

S: 250 Ok

C: RCPT TO:<alice@example.com>

S: 250 Ok

C: RCPT TO:<theboss@example.com>

S: 250 Ok

C: DATA

S: 354 End data with <CR><LF>.<CR><LF>

C: From: "Bob Example" <bob@example.org>

C: To: "Alice Example" <alice@example.com>

C: Cc: theboss@example.com

C: Date: Tue, 15 January 2008 16:02:43 -0500

C: Subject: Test message

C:

C: Hello Alice.

C: This is a test message with 5 header fields and 4 lines in the message body.

C: Your friend,

C: Bob

C: .

S: 250 Ok: queued as 12345

C: QUIT

S: 221 Bye

{The server closes the connection}

The client notifies the receiver of the originating email address of the message in a MAIL FROM command. In this example, the email message is sent to two mailboxes on the same SMTP server: one each for each recipient listed in the To and Cc header fields. The corresponding SMTP command is RCPT TO. Each successful reception and execution of a command is acknowledged by the server with a result code and response message (e.g., 250 Ok). The transmission of the body of the mail message

is initiated with a DATA command after which it is transmitted verbatim line by line and is terminated with an end-of-data sequence. This sequence consists of a new-line (<CR><LF>), a single full stop (period), followed by another new-line. Since a message body can contain a line with just a period as part of the text, the client sends two periods every time a line starts with a period; correspondingly, the server replaces every sequence of two periods at the beginning of a line with a single one. Such escaping method is called dot-stuffing. The server's positive reply to the end-of-data, as exemplified, implies that the server has taken the responsibility of delivering the message. A message can be doubled if there is a communication failure at this time, e.g. due to a power shortage: Until the sender has received that 250 reply, it must assume the message was not delivered. On the other hand, after the receiver has decided to accept the message, it must assume the message has been delivered to it. Thus, during this time span, both agents have active copies of the message that they will try to deliver. The probability that a communication failure occurs exactly at this step is directly proportional to the amount of filtering that the server performs on the message body, most often for anti-spam purposes. The limiting timeout is specified to be 10 minutes. The QUIT command ends the session. If the email has other recipients located elsewhere, the client would QUIT and connect to an appropriate SMTP server for subsequent recipients after the current destination(s) had been queued. The information that the client sends in the HELO and MAIL FROM commands are added as additional header fields to the message by the receiving server. It adds a Received and Return-Path header field, respectively. Some clients are implemented to close the connection after the message is accepted (250 Ok: queued as 12345), so the last two lines may actually be omitted. This causes an error on the server when trying to send the 221 reply.

12.6 FILE TRANSFER PROTOCOL (FTP)

File Transfer Protocol (FTP) is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. FTP is built on a client-server architecture and uses separate control and data connections between the client and the server. FTP users may authenticate themselves using a clear-text sign-in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it. For secure transmission that hides (encrypts)

the username and password, and encrypts the content, FTP is often secured with SSL/TLS ("FTPS"). SSH File Transfer Protocol ("SFTP") is sometimes also used instead, but is technologically different. The first FTP client applications were command-line applications developed before operating systems had graphical user interfaces, and are still shipped with most Windows, Unix, and Linux operating systems. Dozens of FTP clients and automation utilities have since been developed for desktops, servers, mobile devices, and hardware, and FTP has been incorporated into hundreds of productivity applications, such as Web page editors.

Communication and data transfer

FTP may run in active or passive mode, which determines how the data connection is established. In active mode, the client creates a TCP control connection. In situations where the client is behind a firewall and unable to accept incoming TCP connections, passive mode may be used. In this mode, the client uses the control connection to send a PASV command to the server and then receives a server IP address and server port number from the server, which the client then uses to open a data connection from an arbitrary client port to the server IP address and server port number received. Both modes were updated in September 1998 to support IPv6. Further changes were introduced to the passive mode at that time, updating it to extended passive mode. The server responds over the control connection with three-digit status codes in ASCII with an optional text message. For example "200" (or "200 OK") means that the last command was successful. The numbers represent the code for the response and the optional text represents a human-readable explanation or request (e.g. <Need account for storing file>). An ongoing transfer of file data over the data connection can be aborted using an interrupt message sent over the control connection.

While transferring data over the network, four data representations can be used:

- ASCII mode: used for text. Data is converted, if needed, from the sending host's character representation to "8-bit ASCII" before transmission, and (again, if necessary) to the receiving host's character representation. As a consequence, this mode is inappropriate for files that contain data other than plain text.
- Image mode (commonly called Binary mode): the sending machine sends each file byte for byte, and the recipient stores the bytestream

as it receives it. (Image mode support has been recommended for all implementations of FTP).

- EBCDIC mode: use for plain text between hosts using the EBCDIC character set. This mode is otherwise like ASCII mode.
- Local mode: Allows two computers with identical setups to send data in a proprietary format without the need to convert it to ASCII

For text files, different format control and record structure options are provided. These features were designed to facilitate files containing Telnet or ASA.

Data transfer can be done in any of three modes:

- Stream mode: Data is sent as a continuous stream, relieving FTP from doing any processing. Rather, all processing is left up to TCP. No End-of-file indicator is needed, unless the data is divided into records.
- Block mode: FTP breaks the data into several blocks (block header, byte count, and data field) and then passes it on to TCP.
- Compressed mode: Data is compressed using a single algorithm.



CHECK YOUR PROGRESS

2. Fill in the blanks:

(a) The primary function of DNS is to map _____ onto _____.

(b) The process of looking up a name and finding an address is called _____.

(c) DNS messages are sent in _____ packets with a simple format for queries.

(d) SMTP connections secured by SSL are known by the shorthand _____.

(e) _____ has a feature to initiate mail queue processing on a remote server

(f) _____ is built on a client-server architecture and uses separate _____ and _____ between the client and the server.

(g) _____ users may authenticate themselves using a clear-text sign-in _____, normally in the form of a username and password.



12.7 LET US SUM UP

- The Application Layer is the most important and most visible layer in computer networks.
- A Client-Server model is the oldest model used to organize a networked application.
- The Domain Name System is a distributed database that allows to map names on IP addresses.
- The essence of DNS is the invention of a hierarchical, domain-based naming scheme and a distributed database system for implementing this naming scheme.
- Every domain, whether it is a single host or a top-level domain, can have a set of resource records associated with it.
- The DNS name space is divided into non-overlapping zones.
- Simple Mail Transfer Protocol is an Internet standard for electronic mail transmission across Internet Protocol networks.
- File Transfer Protocol is a standard network protocol used to transfer files from one host to another host over a TCP-based network.
- Multipurpose Internet Mail Extensions (MIME) were developed to encode binary files for transfer through SMTP.
- FTP may run in active or passive mode, which determines how the data connection is established.



12.8 FURTHER READING

- Tanenbaum, A. S., & Wetherall, D. (1996). *Computer networks*(pp. I-XVII). Prentice hall.
- Bonaventure, O. (2011). *Computer Networking: Principles, Protocols, and Practice*. The Saylor Foundation.



12.9 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: (a) Client Server.
(b) networked applications.
(c) host names, IP addresses.
(d) Internet Corporation for Assigned Names and Numbers.

(e) single, hosts.
(f) generic, countries.
(g) absolute, relative.

Answer 2. (a) domain names, resource records
(b) name resolution.
(c) UDP.
(d) SMTPS.
(e) SMTP.
(f) FTP, control, data connections.
(g) FTP, protocol.



12.10 MODEL QUESTIONS

Q1. Explain the concept of a Client Server model.

Q2. What is the Domain Name System? Explain its significance.

Q3. Explain the DNS Namespace with diagram.

Q4. What are Domain Resource Records? Explain their usefulness.

Q5. Describe the concept of Name servers.

Q6. Explain the concept of the Simple Mail Transfer Protocol giving suitable examples.

Q7. What is the File Transfer Protocol? Describe its application.

****

UNIT 13: FUNDAMENTALS OF TCP AND UDP

UNIT STRUCTURE

13.1 Learning Objectives
13.2 Introduction
13.3 Connectionless and Connection Oriented Protocol
  13.3.1 Connection Oriented Protocol
  13.3.2 Connectionless Protocol
13.4 Transmission Control Protocol (TCP)
13.5 User Datagram Protocol
13.6 Internet Datagram Protocol (IDP)
13.7 Let Us Sum Up
13.8 Further Reading
13.9 Answers To Check Your Progress
13.10 Model Questions

13.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe how connection oriented protocols work
- describe how connectionless protocols work
- describe how transmission control protocol is used as a connection oriented protocol
- learn about the various features of TCP
- describe how user datagram protocol is used as a connectionless protocol
- learn about the various features of UDP
- learn about Internet Datagram protocol

13.2 INTRODUCTION

In the previous unit, we have learnt issues related to the topic of application layer. Concepts like client server model simple mail transfer protocol and file transfer were discussed in detail in the previous unit. In this unit, we will discuss the fundamentals of TCP, UDP and IDP. In the next unit, we will explore the different network standard protocol.

13.3 CONNECTION ORIENTED AND CONNECTIONLESS PROTOCOLS

Computer systems interact with each other on the Internet or on a local area network using TCP/IP protocol suite. A TCP/IP protocol suit is a layered architecture where some numbers of protocols at each layer are responsible for communication between the systems. Of the 4 layers available in TCP/IP suite, two protocols TCP, UDP at transport layer are mainly responsible for communication. Depending on the type of communication required either TCP can be used for a reliable connection oriented service or UDP can be used for an unreliable connectionless service.

There are two types of end to end protocols defined in the transport layer for communication between two network terminals. These are Connection oriented protocol and Connectionless protocol.

13.3.1 Connection Oriented Protocols

In a connection oriented protocol, a logical connection is first established between the source station and the destination station. Once the connection is established, data is transmitted by the source station and acknowledgement for the data is sent by the destination station. If the acknowledgement receipt does not reach the source, it is assumed that data sent by source did not reach the destination or it may be lost in the network. So the data is retransmitted by the source station. After the transmission is completed successfully, connection between the source and destination system is terminated. This is similar to telephonic conversation between two persons. First of all a number that corresponds to the receiver is dialed, and the caller waits for the acknowledgement from other side. Once the acknowledgement is received by the caller, it is understood that connection between the two parties is established and a communication may begin.

Here a stream of data is delivered in the same order as it was sent. There is a sequence of operation to be followed by the users of connection oriented service. These are:

- Connection is established
- Information is sent
- Connection is released



Figure 13.1: Connection oriented Architecture

TCP is a connection-oriented protocol that is a connection is established and maintained until both the systems involved in the connection has finished exchanging the data. Telnet and rlogin are examples of connection-oriented protocol services. Connection-oriented communication may be a circuit switched connection, or a packet-mode virtual circuit connection. Asynchronous transfer mode (ATM) is an example of packet-mode virtual circuit connection.

Circuit Switching (CS): A circuit switching network communication starts by establishing a dedicated channel between the systems, before a message transfer is initiated. As long as both the systems are involved in data transfer the channel is reserved and kept in active mode. In case of half duplex and full duplex communication, one channel and two channels are respectively allocated. It is mainly used for communication requiring real time services without any much delay. Circuit switching takes place at the physical layer and there is no packet data transfer between the two stations. There is no addressing involved during data transfer, as end to end addressing is used during the setup phase.

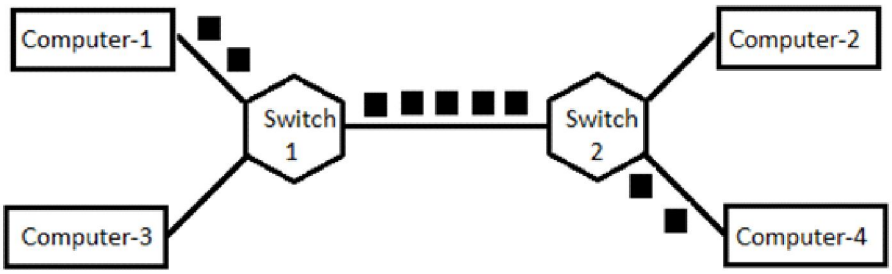


Figure 13.2: Circuit Switched Network

As shown in the Figure 13.2, if Computer-1 wants to use the network; it need to first request for a connection with Computer-4, which must be accepted by all the switches as well as by Computer-4. This phase is called

the setup phase. Then Computer-1 can communicate with Computer-4. All through the connection phase if computer-3 tries to communicate with computer-4 or any other user it will get busy signal from the network.

13.3.2 Connection less Protocols

Connectionsless protocol or service are similar to postal service where letter containing the destination address is delivered to the recipient. Each letter (message) may follow a different route to destination. The order in which a message is sent may not be received in the same order at the destination.

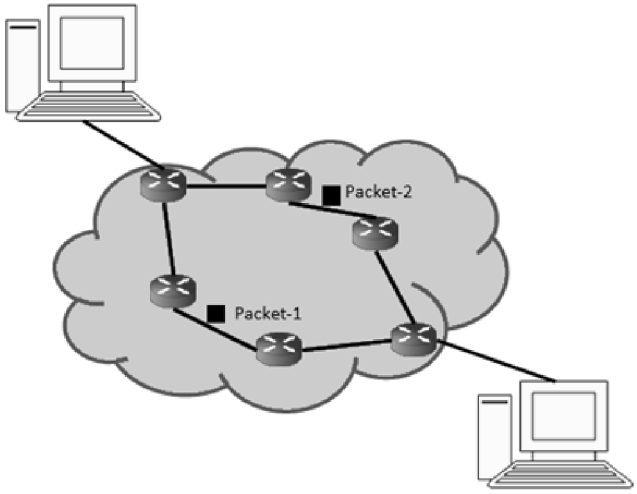


Figure 13.3: A Connectionless Network Communication

In connectionless protocol the destination is unaware of the data sent by the source station until it reaches the destination station. Initial authentication between the two stations involved in communication is not required as was required in connection oriented protocol. User Datagram Protocol (UDP) is a connectionless protocol. A connectionless protocol service does not need a prior connection to be established between the two systems involved in data transfer. Communication is achieved by sending a packet also known as datagram. In contrast to connection oriented service, it does not guarantee delivery of datagram to the destination. As there is no way of knowing the state at any point of time during a communication, it is therefore known as stateless protocol. The plus side of connectionless protocol is that it incurs low overhead compared to connection oriented protocol. It is also used in case of multicast or broadcast of messages where a number of receivers are involved.

Packet Switching (PS): Unlike circuit switching network, packet switching does not require any initial connection to be established. The channel may be available to several users simultaneously. A packet which may be part of a multipacket transmission is treated as independent packet. However increase in the number of users communicating in the network may lead to network congestion. In general packet switching is mainly used in non real time scenarios like data or voice applications.

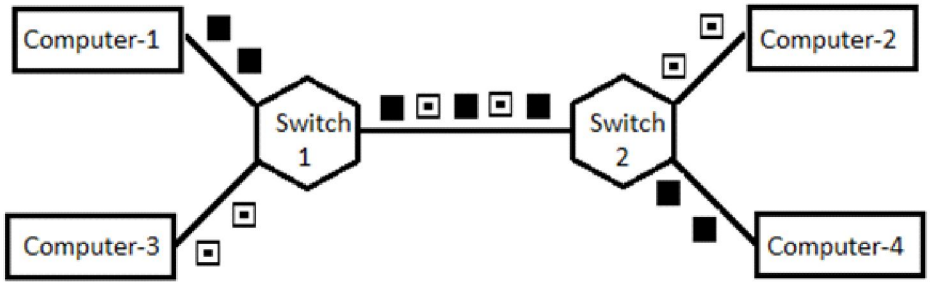


Figure 13.4: Packet Switched Network

As shown in the Figure 13.4, if computer-1 wants to send a message to computer-4 and if computer-3 wants to send a message to computer-2, it is possible simultaneously. For this type of communications the message is padded addresses of source and destination in the packet header. The intermediate switches sense the packet header to decide packet route and destination.

Packet switching can be carried out in two possible ways: Datagram or Virtual circuit.

- **Datagram:** In datagram a message is divided into multiple smaller packets and is sent across the network. Also each packet may follow a different route in the network to reach the destination. The switching nodes decide on the routing path based on the destination address. The packets may reach the destination out of order or may be lost in the network.

- **Virtual circuit:** In virtual circuit, a virtual route between the source and the destination is established. This is done by having a table entry for all the intermediate switches from source to destination. This setup is achieved through request and acknowledgement signals. Each packet contains virtual circuit identifier (VCI) instead of the destination address.

13.4 TRANSMISSION CONTROL PROTOCOL (TCP)

Transmission control protocol (TCP) is an end to end reliable connection oriented protocol. It works together with Internet protocol to provide services like packet exchange between systems in network especially over the Internet. Almost ninety percent of today's network communication and several applications like WWW, FTP, TELNET and EMAIL etc. uses TCP service. The term connection oriented means that connection is established and maintained by the applications running at both the ends until the exchange of information (message) is complete. To achieve this application message is broken into smaller data packets and sent over the communication network. The TCP works in layer-4 that is transport layer of Open Systems Interconnection (OSI) model.

When a client program request for a web page to the server it does so with the help of HTTP protocol. The web server in return sends the file using TCP. The file is divided into packets; each packet is then numbered and sent to IP for delivery to destination. The packets may be sent across multiple routes, where they are received by the TCP layer of the client. Acknowledgement is sent for received packets as well as for retransmission of lost packets. The packets are assembled according to their number and sent back to the receiving client application.

The key features of TCP can be listed as follows:

- Connection oriented protocol.
- Allows end to end communication.
- Supports full duplex server.
- Acknowledgements are used which makes it a reliable protocol.
- Order is maintained for sending packets.
- Allows error checking and recovery mechanism.
- Allows flow control and quality of service.

13.5 USER DATAGRAM PROTOCOL (UDP)

User Datagram Protocol (UDP) is an unreliable connectionless protocol used mainly for communication between application over the Internet which requires low response time and that allows partial loss of packets without affecting the communication. It runs mainly on the top of IP and is referred to as UDP/IP as like TCP/IP.

Communication where bandwidth is a constraint streaming a video by

sending thousands of packets and sending acknowledgement for each of the thousand packets results in significant wastage of bandwidth. The IP ensures delivery of almost every packet but loss of some of the packets can be ignored. As in the case of voice traffic or video streaming loss of some packets gets unnoticed. However UDP can be configured to retransmit lost packets as well as arrange correctly received packets so that applications requiring lossless data transmission and better packet transfer can be achieved compared to TCP. There are two types of services provided by user datagram protocol; it provides port number to each application request and also a checksum facility to verify correctness of the data that has arrived.

Like TCP, UDP also works in layer-4 of the transport layer of Open Systems Interconnection (OSI) model. It works in combination with other higher level protocols like Trivial File Transfer Protocol (TFTP), Simple Network Protocol (SNP), and Real Time Streaming Protocol (RTSP) etc to manage data transmission. UDP is a perfect protocol for applications such as voice, gaming and video communications, where some amount of data loss in the network does not affect the perceived quality.

The key features of UDP can be listed as follows:

- Acknowledgement of data not required.
- It is excellent protocol single direction flow of data.
- Uncomplicated and appropriate for query based communications.
- It is connectionless.
- Does not provide congestion control mechanism.
- Ordered delivery of packets not guaranteed.
- Suitable for streaming applications such as VoIP, multimedia streaming.

13.6 IDP

Internet Datagram Protocol (IDP) is a straightforward, unreliable datagram protocol, which is used to support the datagrams, which are connectionless, unreliable buffers of a fixed maximum length (SOCK_DGRAM) concept for the Internet Protocol (IP) family. Generally IDP sockets are used with the "sendto" and "recvfrom" functions. Also the destination can be fixed using "connect" function in conjunction with read and write functions. Xerox protocols are built on top of Internet Datagram Protocol. Therefore, IDP address formats are the same to those used by the Sequenced Packet

Protocol (SPP). The Internet Datagram Protocol port space may be "connected" to an SPP port. Also packets may be broadcast using a reserved broadcast address if the network supports IDP.



CHECK YOUR PROGRESS

1. The sub categories of connectionless services are:
   a) Acknowledged datagram
   b) Unreliable datagram
   c) Both a & b
   d) None of these
2. Unreliable connectionless service can be also termed as
   a) Byte stream service
   b) Data service
   c) Datagram service
   d) None of these
3. Transport layer protocols deals with
   a) application to application communication
   b) node to node communication
   c) process to process communication
   d) none of the mentioned
4. Which one of the following is a transport layer protocol?
   a) dynamic host configuration protocol
   b) internet control message protocol
   c) stream control transmission protocol
   d) neighbor discovery protocol
5. Which one of the following is a transport layer protocol used in internet?
   a) UCP
   b) TCP
   c) both (a) and (b)
   d) none of the mentioned
6. User datagram protocol is called connectionless as
   a) it sends data as a stream of related packets
   b) all UDP packets are treated independently by transport layer
   c) both (a) and (b)

d) none of the mentioned

7. Transmission control protocol is
   a) connection oriented protocol
   b) receives data from application as a single stream
   c) uses a three way handshake to establish a connection
   d) all of the mentioned

8. Which one of the following is a version of UDP with congestion control?
   a) stream control transmission protocol
   b) stream congestion transmission protocol
   c) structured stream transport
   d) none of the mentioned

9. A _____ is a TCP name for a transport service access point.
   a) pipe
   b) port
   c) node
   d) none of the above

10. The service that is similar to postal system is:
    a) Connection-oriented services
    b) Connectionless services
    c) Both a & b
    d) None of these

11. The different types of services provided by the interface to the layers above them are listed as follows
    a) Connection-oriented services
    b) Connectionless services
    c) Both a & b
    d) None of these

12. Which of the following addresses is used to deliver a message to the correct application program running on a host?
    a) IP
    b) Logical
    c) Physical
    d) Port



13.7 LET US SUM UP

- Connection oriented protocol and connectionless protocol are two end to end protocols defined in the transport layer for communication between two network terminals
- In a connection oriented protocol, a logical connection is established between the source station and the destination station before data is transmitted by the source and acknowledgement is sent by the destination.
- A circuit switching network communication starts by establishing a dedicated channel between the systems, before a message transfer is initiated.
- In connectionless protocol the destination is unaware of the data sent by the source station until it reaches the destination station.
- Packet switching does not require any initial connection to be established, channels may be available to several users simultaneously.
- Transmission control protocol (TCP) is an end to end reliable connection oriented protocol that works together with Internet protocol to provide services like packet exchange between systems in network.
- User Datagram Protocol (UDP) is an unreliable connectionless protocol used mainly for communication between application over the Internet which requires low response time and that allows partial loss of packets without affecting the communication.
- Internet Datagram Protocol is a simple, unreliable datagram protocol used to support the SOCK_DGRAM concept for the Internet Protocol family.



13. 8 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw- Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.

13.9 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: c
Answer 2: c
Answer 3: c
Answer 4: c
Answer 5: c
Answer 6: b
Answer 7: d
Answer 8: a
Answer 9: b
Answer 10: b
Answer 11: c
Answer 12: d

13.10 MODEL QUESTIONS

Q1. Explain how does a connection oriented protocol work?

Q2. Discuss a situation where connectionless protocol UDP is found to be better than connection oriented protocol TCP.

Q3. Explain how a connection oriented protocol ensures reliability in a communication between two stations?

Q4. Discuss the key features of Transmission control protocol.

Q5. Discuss the key features of User datagram protocol.

Q6. Differentiate between Packet Switching and Circuit switching.

Q7. Discuss how virtual circuit switching is different from circuit switching?

Q8. Describe Internet datagram protocol

****

UNIT 14 : NETWORK STANDARD PROTOCOL

UNIT STRUCTURE

14.1 Learning Objectives
14.2 Introduction
14.3 File Transfer Protocol (FTP)
14.4 Telnet
14.5 Hyper Text Transfer Protocol (HTTP)
14.6 Email Protocols
    14.6.1 Post Office Protocol (POP)
14.7 Browser
14.8 World Wide Web (WWW)
14.9 PING
14.10 Remote Login
14.11 Socket
14.12 Let Us Sum Up
14.13 Further Reading
14.14 Answers To Check Your Progress
14.15 Model Questions

14.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- describe the uses of FTP protocol for file transfer
- learn how remote communication with a device is done by using TELNET
- learn about the functionality of Hypertext Transfer Protocol for exchanging files
- describe various HTTP error codes
- describe commonly used protocols for transfer of emails
- define World Wide Web
- describe how a browser can locate, retrieve and view contents of World Wide Web
- describe how to check presence of a remote system online
- describe how to run applications in a remote system using remote login like SSH

- learn about communication between two applications running on different systems

14.2 INTRODUCTION

In the previous units we have learnt about the fundamentals of TCP, UDP and IDP protocols. In this unit, we will learn about the different network standars protocols like FTP, TELNET, HTTP etc.

It is essential to understand the basics of networking protocols as well as the applications that are used in day to day life for communication or for exchanging files between systems in a local area network or the Internet. It is important to know how to locate, retrieve and view the collection of documents and resources like web pages, text, images, audio, video and other multimedia that are available in the World Wide Web by using applications like web browser installed in a user's system. It is also important to understand online service like electronic mail that can be used for communication between users and how file exchange protocols and applications like FTP, TELNET and Remote Login make the communication easier. Programs like PING allows user to diagnose the network problems and HTTP error codes help the user to get insight about different possible reasons for a page error. This unit is going to discuss all the aspects of Network Standard Protocol. In the next unit, we will explore different types of network devices.

14.3 FILE TRANSFER PROTOCOL (FTP)

FTP is a short form for File Transfer Protocol. FTP is commonly used to exchange files between computers on a network or over the internet. FTP works over TCP/IP protocols to facilitate data exchange. FTP is based on client-server architecture secured with SSL/TLS. A user can use FTP to exchange files between two accounts or between an account and a user system, or to access software archives stored online. FTP is generally used to download or upload a file to or from a server using the Internet.

A FTP may be used in the following ways

- **Graphical FTP Software:** A Graphical FTP Software simplifies the file transfer by allowing the user to drag and drop files between windows. In the FTP software program, the user needs to enter the name of the FTP host

along with the username and the password which is provided on signup. For an anonymous FTP server, the user need not enter any username and password. Some examples of FTP programs are FileZilla, WS_FTP, Cute FTP, cURL.

- **Web browser:** To connect to a remote FTP server for file transfer, a user can type the FTP address in the web browser the same way as HTTP address is used to open a website. FTP transfers using a web browser is convenient and easy to browse directories, read and retrieve files, as connecting to the ftp server and transferring files is taken care of by the web browser itself. However, web browsers are slow, less reliable and have fewer features compared to dedicated FTP software. For example, to connect to a test FTP site for download and upload files enter a URL ftp://speedtest.tele2.net/.

- **Command-line FTP:** In any operating systems there are built-in command-line programs to connect list files and upload and download files. For example, to use FTP commands in windows command prompt

Step1: Open Command prompt
Step 2: At the C:\> prompt, type FTP. The C:\> prompt changes to ftp>.
Step 3: At the ftp> prompt, type: OPEN ftp_site.com and press ENTER
Step 4: When prompted, type the username and password provided by the administrator of the FTP site, and then press ENTER.

14.4 TELNET

TErminal NETwork in short Telnet is client software used most commonly as a command line interface for remote management and communication with a device. Telnet allows a user to communicate with remote computer over a TCP/IP network. Telnet client software on user computer can be used to make a connection to a remote host. Once a connection is established between telnet client and remote host, the client program becomes a virtual terminal and thus allows the user to communicate through his/her computer to the remote host. Generally, to log into the remote host, the user needs to have an account on that system. However, some remote systems do allow a guest or public login without having an account. Telnet client software exists in all operating systems.

Unlike HTTP which allows the user to exchange files to and from server. Telnet protocol allows the user to logon to the server as if he/she is an

authentic user and therefore grants control and permissions to the files and applications of the remote system.

Telnet uses port number 23 over TCP/IP network for communication with remote computers. Telnet by default is disabled in Windows and to enable Telnet utility following steps can be followed:

Step1: Go to Control Panel.

Step 2: Click on Programs and Features.

Step 3: Click on Turn Windows features on or off.

Step 4: Select the Telnet Client check box and Click OK

How to start a Telnet session: In the windows command prompt type telnet and press ENTER. The prompt for telnet is displayed as Telnet>. Type help to see the list of telnet commands.

14.5 HYPERTEXT TRANSFER PROTOCOL (HTTP)

Hypertext Transfer Protocol (HTTP) was coined by Ted Nelson. HTTP is a standard that allows the users to exchange files of type text, images, audio, video and multimedia files on the World Wide Web. HTTP runs on top of the TCP/IP suite of protocols and uses default port 80 for the connections. As the user opens any web browser he/she indirectly makes use of HTTP. The latest version of HTTP is HTTP 1.1.

A web page may include references to several other web pages, so a HTTP request to a web page may result in selection of additional web pages, that HTTP requests. A web server contains a HTTP daemon program running all the time in addition to the web pages that are being stored. A HTTP daemon program runs all the time so that it can handle any incoming HTTP request. The web browser runs a client HTTP program which is responsible for sending request to server machines. When the user requests for a web page by either typing the URL or clicking on the web link, the web browser creates a HTTP request and sends it to the IP address indicated by the URL. At the destination server, the HTTP daemon program receives the request sent by the client and acknowledges back by sending all the requested files.

HTTP Status Codes are Error Messages:

Sometimes it happens that when a user types in a URL for a web page on the Internet he/she gets an error message. For example “404 File Not Found”

error is a common HTTP error message. These error messages also called HTTP status code are responses given by the server machine to help recognize the problem.

A “404 File Not Found” HTTP status code means the web server cannot find the web page requested by the user. This means the requested webpage might have been deleted or moved or the user might have entered wrong URL. If the user knows the HTTP status code meaning, he/she can work out what has gone wrong. For example, in case of a 404 error the user may look for typing error in the URL then correct the error and try it again. In case error message appears again, the user may try deleting part of the URL between each backslash until the page opens without any error. It is another way to overcome the error message. Many websites help the user by creating custom 404 error pages where quick links to website navigation and website search features help find what is required by the user.

Some of the HTTP status codes are listed below:

- 1xx Informational responses
  - 100 Continue
  - 101 Switching Protocols
- 2xx Success
  - 201 Created
  - 202 Accepted
- 3xx Redirection
  - 301 Moved Permanently
  - 302 Found
- 4xx Client errors
  - 400 Bad Request
  - 401 Unauthorized
  - 408 Request Timeout
- 5xx Server error
  - 500 Internal Server Error
  - 502 Bad Gateway

14.6 EMAIL PROTOCOLS

The first electronic mail was developed in early 1960s. This system could send mails to the users of the same system only. In 1971 Ray Tomlinson

developed the first network based mail system via ARPANET. Today email system has become very popular in communication over the internet and thus resulting in a set of standardized network protocols. The email system is based on client/server architecture. The sender uses a client mail program to compose a message. The client program then sends the message to the server which then forwards the message to the email server of the receiver. The message then is sent to receiver's client program from where the message can be read. This process of sending and receiving emails through client programs follows a set of standard protocols working on different machines and operating systems. The most commonly used protocols for transfer of emails are categorized into mail transport protocol (example SMTP) and mail access protocol (example POP, IMAP)

14.6.1 Post Office Protocol (POP)

The Post Office Protocol (POP) is one of the primary mail access protocol used for downloading the recipient's email messages from mail servers. POP is facilitated by the TCP/IP protocol stack for network connection. POP works in conjunction with Simple Mail Transfer Protocol (SMTP) for email communication where POP pulls email messages and SMTP pushes the email messages to the server. In contrast to SMTP, this protocol requires authentication using the username and the password by the client programs. The current version of Post Office Protocol is POP3 and it is used generally for email communication using client/server architecture.

When a user checks the mailbox on the server using a client email program and downloads any mail perhaps the user is using POP3. Some of the popular email applications like Outlook express and Eudora have built-in this standard protocol. In addition to this, some web browsers like Microsoft Internet Explorer and Netscape browsers also have this built in protocol.

Email client applications used for downloading email messages using POP, automatically deletes the emails on the server after they are transferred successfully. As they are configured to do so, however, they can be reconfigured. POP also allows for email attachments as it is fully compatible with Multipurpose Internet Mail Extensions (MIME) which is an internet messaging standard. The users who do not have a steady internet connection or those who

access their mails on the same system, POP works well for them. POP requires downloading the entire email message upon authentication, however takes longer time for larger attachments.

14.7 BROWSER

In today's world anyone who wants to locate, retrieve and view the contents of World Wide Web needs to type the URL in the address bar of the web browser. It is the job of the web browser to interpret the commands contained in the web page. In short, a web browser is an application program used to view and explore information such as web pages, text, images, audio, video, any other multimedia files etc. It is based on a client server model, where the web browser acts like a client, and whose task is to communicate with the server to retrieve information. The server sends information back to the web browser, which then displays the information.

Today's web browsers are well-designed application that can translate and show HTML Web pages, JavaScript, AJAX and any other content hosted on Web servers. Some web browsers also contain built in plug-ins for extended capabilities like display multimedia information or perform video conferencing or provide other security features. In the past web browsers were mainly text-based but these days' graphical and voice-based web browsers are also available. Some of the most commonly available web browsers are listed below:

Browser	**Vendor**
Opera	Opera Software
Google Chrome	Google
Internet Explorer	Microsoft
Safari	Apple
Netscape Navigator	Netscape Communications Corp.
Mozilla Firefox	Mozilla

Although there are a variety of web browsers available and their features differ from one another, yet the basic component of web browser remains the same. A web browser mainly consists of three main components: 1) Controller 2) Interpreter and 3) Client Programs

Controller interprets the input taken from keyboard/mouse and provides services based on the input received. The information taken by the Interpreter from the controller is executed line by line. Client Program

specifies the protocols to be used for a particular service. Some of the commonly used protocols by the client program are: HTTP, SMTP, FTP, and POP.

14.8 WORLD WIDE WEB (WWW)

The World Wide Web in short WWW or often referred as “The Web” is a system of online content designed by Tim Berners-Lee in the year 1991. The World Wide Web uses HTTP to access interlinked HTML pages over the Internet. Most of the people think of World Wide Web as the Internet whereas it is the collection of documents and resources like web pages, text, images, audio, video and other multimedia linked by hyperlinks and uniform resource identifier. On the contrary, Internet is a collective organization of interconnected computer networks. The online content of World Wide Web can be accessed by a web browser.

Today so many websites are shifting their content design and style of development from large screen laptop and desktop computers to ever increasing demand of web in small screen mobile phone. Privacy over the internet has become a significant issue on the Web as individuals location information, browsing pattern and search history are captured and targeted for individual specific advertisements. An extra level of security is provided by web proxy services which direct the browsing through proxy web servers.

14.9 PING

Ping is a basic program used to check the presence of or the attention of a remote system online. If a user wants to verify the existence of a particular IP address, the ping program/command may be used. It is used as a network diagnostic tool by the user to check that the remote computer system is actually working. Ping sends an Internet Control Message Protocol (ICMP) Echo Request to a device on the network and waits for its reply. Ping is generally used for testing connectivity and determining response time.
For example to test an IP address (216.58.196.100) the user can open the command prompt in windows and type: ping 216.58.196.100. The Figure 14.1 describes the usage.

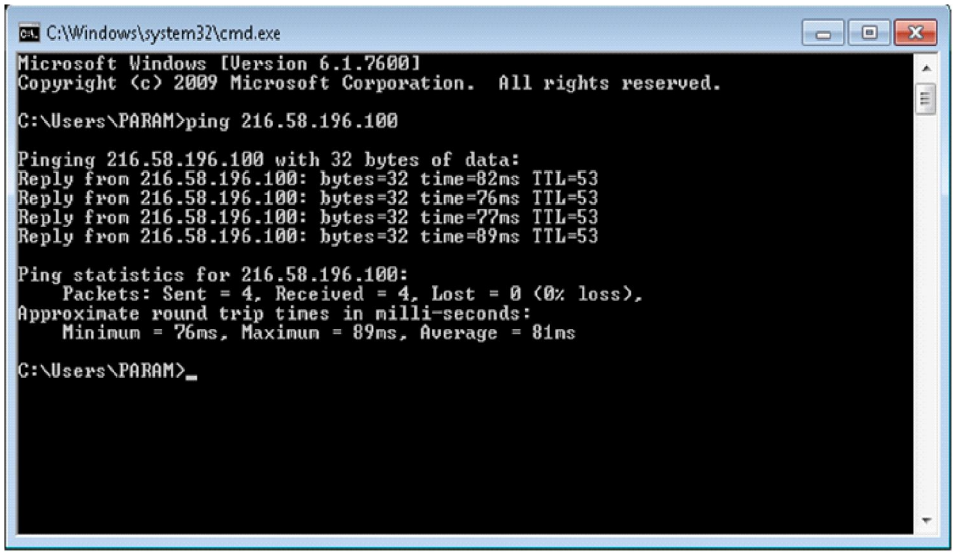


Figure 14.1: Test an IP address (216.58.196.100) using PING

The figure 14.2 shows that if the ping fails, the message request timed out is displayed. It means that the destination host system is down or there is no route back.

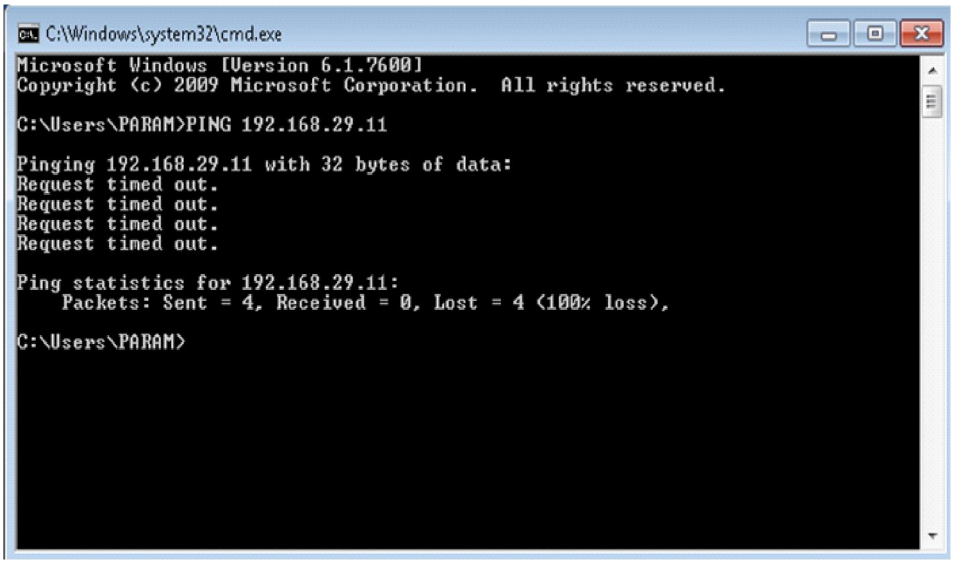


Figure 14.2: Request Times out message using PING

14.10 REMOTE LOGIN

Remote Login is a mechanism based on client server model to run applications in a remote system. For this the authorized user needs to establish a session in the remote (host) system. Once a session is established, it gives the user a feel that he/she is actually working in the host system. A client server application program can be used for remote

connection. Generally there are two programs that are available for such service, these are TELNET and SSH. Another way of remote login is via Virtual Private Network (VPN), which allows remote file access capability in Wide Area Network. For remote login the VPN needs to have a client program in the host system and VPN server on the target network. File sharing in a Local area network is not considered as remote login.

Secure shell an alternative to VPN for remote login. Secure Shell (SSH) protocol is a UNIX program that supports command line interface for remote login. SSH is better and easier than TELNET, it allows user to execute commands one at a time in the remote system. Some of the advantages of SSH are:

- SSH provides strong encryption as it uses public-key cryptography between the client system and remote servers. In case a intruder sneaks through the network to obtain a copy of the message, it still remains confidential.
- SSH provides strong authentication for connections in insecure environment like the Internet.
- Supports secure version utilities like ssh, slogin,and scp. Earlier insecure utilities were rsh, rlogin and rcp

SSH is present by default in most of the operating systems. The TCP port number 22 is used for listening by SSH server

14.11 SOCKET

A socket is a basic way of communication between two applications running on different systems on a local area network or across internet. Sockets can also be used for communication between two application on the same system. A socket helps in bidirectional communication on TCP/IP for sending data and receiving data with socket in another system.

Network programmers generally use API (Application programming Interface) to create sockets for communication. The API is nothing but function calls or socket libraries just like functions open (), read (), write () and close () used for doing file operations. The two most commonly used socket libraries are

- Berkeley Sockets for LINUX/UNIX systems
- WinSock for Windows systems

Following are the steps followed for establishing a connection between two systems: alignment client and server:

Client side connection:

- Create a socket
- Use IP address of the server to connect the socket
- Send or Receive data
- Socket Closed

Server side connection:

- Create a socket
- Bind socket to a port
- Listen for connection request
- Accept connection request
- Send or Receive data

Today most of the network sockets work in combination with TCP, IP and UDP. A socket library uses IP, TCP, and UDP respectively for raw packets, streams and datagrams. The network socket library facilitates communication between systems by first identifying the system with the help of IP address. As today's Internet provides name based service, instead of using IP address like 202.168.22.1, name like www.xyz.com may also be used. Port numbers are also used for distinguish between multiple applications. For example, port number 80 is used as a default port by web browsers for communication with servers on the Internet. There is also a networking protocol Secure Sockets Layer (SSL) that enables client server authentication and client server encrypted communication.



CHECK YOUR PROGRESS

1. Which of the statement is not correct
   a) FTP works over TCP/IP protocols
   b) Based on client-server architecture
   c) Used to download or upload a file
   d) Used only using Web browser
2. Telnet uses port number
   a) 21
   b) 22
   c) 23
   d) 24
3. HTTP uses port number
   a) 80

b) 81
c) 20
d) 40

4. HTTP error code _____means file permanently moved
   a) 201
   b) 301
   c) 401
   d) 501
5. SMTP generally uses port number
   a) 20
   b) 21
   c) 23
   d) 25
6. Today's web browser is based on
   a) Client server
   b) Client application
   c) AJAX
   d) Javascript
7. Which of the following is not a web browser?
   a) Opera
   b) Netscape
   c) Safari
   d) Weblorer
8. WWW is designed by _____ in the year 1991
   a) Tim Lee
   b) Tim Berners-Lee
   c) Berners-Lee
   d) John-Lee
9. Ping sends an _____ Echo Request to a device on the network and waits for its reply
   a) SMTP
   b) IMTP
   c) ICMP
   d) TCMP
10. TCP port number _____ is used for listening by SSH server
    a) 21
    b) 22
    c) 23
    d) 24

11. Sockets can be also be used for communication between two application on the
    a) same system
    b) different system
    c) Both a & b
    d) None of the above
12. Which of the following is not a step in establish a connection using Socket?
    a) Create a socket
    b) Send or Receive data
    c) Send IP address
    d) Socket Closed



14.12 LET US SUM UP

- File Transfer Protocol is commonly used to exchange files between computers on a network.
- FTP works over TCP/IP protocols to facilitate data exchange.
- TELNET is client software used most commonly as a command line interface for remote management and communication with a device.
- HTTP is a standard that allows users to exchange files of type text, images, audio, video and multimedia files on the World Wide Web.
- SMTP is a mail transport protocol responsible for sending messages between mail servers.
- The Post Office Protocol is a mail access protocol used for downloading recipient's email messages from mail servers.
- A web browser is an application program based on a client server model that is used to view and explore information such as web pages, text, images, audio, video, any other multimedia files etc.
- World Wide Web is the collection of documents and resources like web pages, text, images, audio, video and other multimedia linked by hyperlinks and uniform resource identifier.
- Ping is a basic program used to check presence of or attention of a remote system online.
- Remote Login is a mechanism based on client server model to run applications in a remote system.

- A socket is a bidirectional communication over TCP/IP between two applications running on different systems on a network.



14.13 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw- Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.



14.14 ANSWERS TO CHECK YOUR PROGRESS

Answer 1: d	Answer 2: c
Answer 3: a	Answer 4: b
Answer 5: d	Answer 6: a
Answer 7: d	Answer 8: b
Answer 9: c	Answer 10: b
Answer 11: c	Answer 12: c



14.15 MODEL QUESTIONS

Q 1. Define HTTP. List some of the HTTP Status Codes.
Q 2. How can simple mail transfer protocol be used to send mails?
Q 3. Differentiate between SMTP and POP email protocols.
Q 4. What is a PING program? How to use PING?
Q 5. How Remote login is different from TELNET?
Q 6. What is a Socket? What are the different functions of a socket? List all the steps to establish a connection using socket.
Q 7. Define FTP. How can FTP be used to transfer files.
Q 8. Write short notes on
   a) Remote Login
   b) TELNET
   c) WWW
   d) FTP
   e) Browser

****

UNIT 15 : NETWORK DEVICES

UNIT STRUCTURE

15.1 Learning Objectives
15.2 Introduction
15.3 Network Devices
  15.3.1 Hub
  15.3.2 Switch
  15.3.3 Bridge
  15.3.4 Router
  15.3.5 Gateway
  15.3.6 RS-232C
15.4 Network Adapters
15.5 Network Interface Card (NIC)
15.6 Component of NIC
15.7 Functions of NIC
15.8 Types of NIC
15.9 Ethernet
15.10 Let Us Sum Up
15.11 Further Reading
15.12 Answers to Check Your Progress
15.13 Model Questions

15.1 LEARNING OBJECTIVES

After going through this unit you will be able to:

- distinguish between a hub and a switch
- state the operation of a bridge
- describe routing
- state the functions of a gateway

15.2 INTRODUCTION

A computer network primarily consists of communicating components and their interconnecting components. Computers, printers are the communicating components and modems, hubs, repeaters, routers etc.

along with the transmission media are the interconnecting components. Transmission media are discussed in UNIT 6. In this unit, the networking devices are discussed.

15.3 NETWORK DEVICES

In computer network, there are many devices used to interconnect the different nodes of the network. These devices are considered to be interconnecting devices without which networking will not be possible. Some important networking devices are described follows.

15.3.1 Hub

In a LAN, each device is connected to a central controlling device by a dedicated point-to-point link. That central device is called a hub. So, a hub is a device which is used to connect all the nodes on a star or ring network. It is a broadcast device and nothing more than a box with a series of cable connectors in it. Various hubs are available in a variety of sizes, from four port device designed for home and small business networks to large rack-mounted units with up to 24 ports or more. A hub is installed simply by connecting it to a power source and plugging in cables connected to the network interface adapters in the computers. Hubs are associated with either physical layer or specific data-link layer protocols. It means that hubs are physical layer or data link layer devices depending upon its two variations viz., passive hub or active hub. Active Ethernet hubs are the most common, because Ethernet is the most popular data-link layer protocol. But



A 4 port HUB A Stackable HUB

Figure 15.1 A 4 port hub and a stackable hub

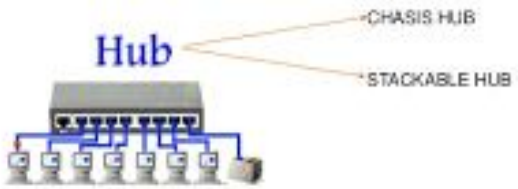


Figure 15.2 Hub in a star topology connecting eight devices

Token Ring hubs and other protocols, such as the Fiber Distributed Data Interface (FDDI) hubs are also available.

In a star topology, direct data transmission from one node to another is not allowed. The central connecting device, i.e. the hub, acts as an exchange. If one device wants to sends data to another device, it first send the data to the hub. Then the hub relays the data to the destination node.

15.3.2 Switch

Switch is a network device that interconnects different nodes of a local area network. It has similarity with a hub. A hub is a physical layer device that receives data from a source node and forwards it to all the nodes connected to it without inspecting the actual address of the destination node. On the other hand, a switch is a special type of hub that offers an additional layer of intelligence to basic, physical-layer repeater hubs. A switch must be able to read the MAC address of each frame it receives. This information allows switches to repeat the incoming data frames only to the computer or computers to which a frame is addressed. This speeds up the network and reduces congestion. Let us explain the case elaborately. Every network operates with a certain bandwidth. When only one node transmits data to only one node then the network operates with its full bandwidth speed. If two nodes start transmitting data to another two nodes, then the speed of the network becomes half of its bandwidth. Therefore, as more and more nodes start communication through the network, its speed becomes divided by the number pairs of participating nodes. But the speed of the network does not decreases if a switch is used in place of a hub in the situation described above. So the use of switch maintains the speed of data communication in a net and contributes to the performance of the network.

Switches operate at both the physical layer and the data link layer of the OSI Model. For this reason switches are termed as a layer-2 network device.

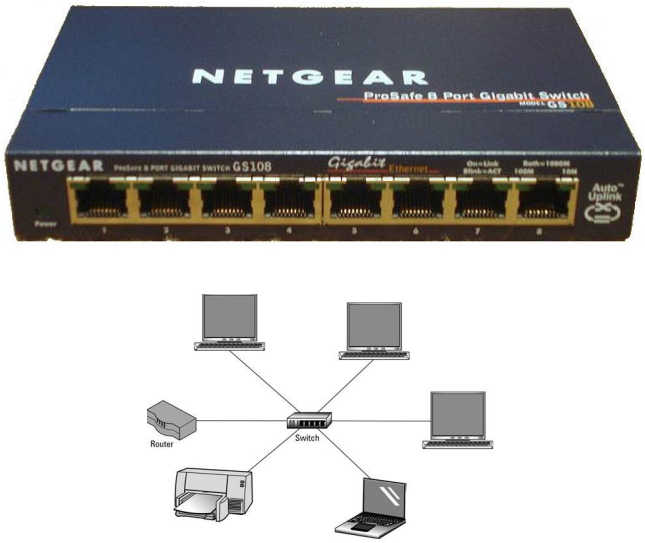


Figure 15.3 A switch and a star network connecting six devices using a switch



CHECK YOUR PROGRESS

1. What type of addressing used in Layer 2 switch?·
   a. IP addressing b. Bit and error checking·
   c. MAC addressing d. None of the above
2. At which layer of OSI model MLS switch works?·
   a. Layer 1 b. Layer 7·
   c. Layer 2 and 3 d. Layer 3 and 4
3. What is the difference between Hub and switch?·
   a. Hub broadcast data while switch first see and then unicast to its real destination only·
   b. Unlike switch Hub can easily secure the data and broadcast it
4. HUB stands for?·
   a. Hybird Universal Broadcast b. High Utilisation Broadast·
   c. Hybird Unicast Bradcast d. All of the above
5. In a network where security is the primary concerned, what will be used switch or Hub?

a. Both b. Switch·
c. Hub d. None of above

6. You are providing IT consultancy, your task is to give recommended suggestion to your client. Your client wants you to suggest either switch or Hub to be used in a medium-sized network. Which device will you recommend to such network?·

a. Both b. Switches·
c. Hubs d. Suggest something else

15.3.3 Bridge

A bridge is a network device that operates on layer 2 of OSI model. A bridge is used to join two network segments together. It provides facilities to computers on either segment to utilize resources on the other segment. Bridges are mostly used to divide large networks into smaller segments to improve network performance.

Bridges have all the capacities of a repeater, but can handle more nodes. Since the network is divided by the presence of a bridge, there are fewer computers competing for resources on each segment, thus improving network performance. Bridges can connect networks that run at different speeds, different topologies, or different protocols. But they cannot join an Ethernet segment with a Token Ring segment, because these use different networking standards.

Bridges operate at both the physical Layer and the MAC sublayer of the Data Link layer. Bridges read the MAC header of each frame to determine on which side of the bridge the destination device is located, the bridge then repeats the transmission to the segment where the device is located.

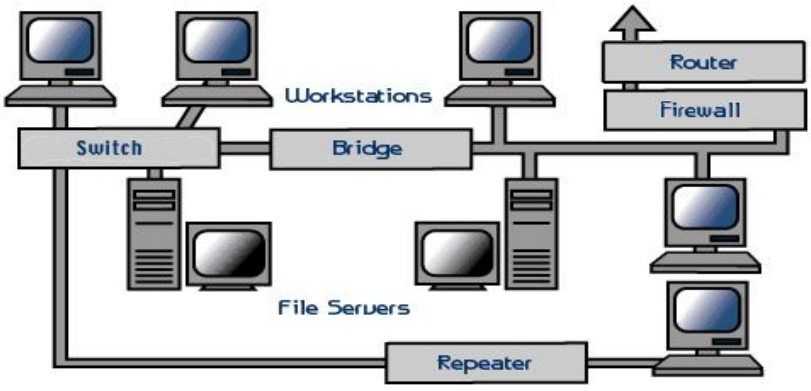


Figure 15.4 A bridge connecting two segments of a LAN

There are some variations of bridges viz., transparent bridge, source routing bridge, translational bridge, neighborhood bridge, remote bridge etc.

15.3.4 Router

Routers are the backbone of computer network. These are the networking devices used to extend or segment networks by forwarding packets from one logical network to another. Routers are most often used in large internetworks that use the TCP/IP protocol suite and for connecting TCP/IP hosts and local area networks (LANs) to the Internet using dedicated leased lines. In a subnet, routers are located at a telephone exchange and these are interconnected by long distance data transmission cable like optical fiber cable etc. In India routers and the subnet are operated by BSNL that provides the Internet service to most of the customers.

Routers work at the network layer (layer 3) of the OSI reference model for networking to move packets between networks using their logical addresses which is the IP addresses of destination hosts in the case of TCP/IP network. Because routers operate at a higher OSI level than bridges, so they have better packet-routing and filtering capabilities and greater processing power. That is why routers are costlier than bridges.

Routers maintain a table of information called routing tables that keep track of all known network addresses and possible paths throughout the internetwork. It also knows the cost of reaching each network. A router updates its routing table periodically by continuously exchanging information between the neighboring routers.

Routers route packets on the basis of on the available paths and their costs, thus taking advantage of the redundant paths that can exist in a mesh topology network. Because routers use destination network addresses of packets, they work only if the configured network protocol is a routable protocol such as TCP/IP.

Depending upon their operating principle routers are divided into two types, viz., static router and dynamic router. In case of static routers they have their routing tables configured manually with all network addresses and paths in the internetwork. On the other hand,

a dynamic router automatically creates its routing tables by listening to network traffic. Routers can be used to segment a large network and to connect local area segments to a single network backbone that uses a different physical layer and data link layer standard. They can also be used to connect LAN's to a WAN's.

15.3.5 Gateway

A gateway is a device which is used to connect different networks using different protocols. Gateways operate at all the layers of the OSI model. In order to communicate with a host on another network, an IP host must be configured with a route to the destination network. If a configuration route is not found, the host uses the gateway (default IP router) to transmit the traffic to the destination host. The default gateway is where the IP sends packets that are destined for remote networks. If no default gateway is specified, communication is limited to the local network. A common use of gateway is to connect the Internet of one country to the Internet of another country. In India, there are three International Internet Gateway (IIG) at Mumbai, Chennai and Agartala.

Gateways perform different responsibilities. They receive data from a network using one type of protocol stack, removes that protocol stack and repackages it with the protocol stack that the other network can use. They also change data format as per the necessity of the destination country. For example, gateway changes date format from dd/mm/yyyy to mm/dd/yyyy when an Internet user of India sends message to USA. They also convert rupee to dollar or pound accordingly. Gateways keep track of Internet billing.

They perform some protocol related conversion also. For example, a gateway that receives Simple Mail Transfer Protocol (SMTP) e-mail, translates it into a standard X.400 format, and forwards it to its destination. Again Gateway Service for NetWare (GSNW), which enables a machine running Microsoft Windows NT Server or Windows 2000 Server to be a gateway for Windows clients so that they can access file and print resources on a NetWare server.



CHECK YOUR PROGRESS

7. A device that links two homogeneous packet-broadcast local networks is-

a) Gateway b) Repeater c) Bridge d) Hub

8. What is the use of bridge in networks?

a) To connect LANs b) To separate LANs

c) To control network speed d) All of the above

9. In OSI reference model router operates in-

a) Layer 1 (Physical Layer) b) Layer 3 (Network Layer)

c) Layer 4 (Transport Layer) d) Layer 7 (Application Layer)

10. What does a router do in a network?

a) Forwards a packet to all outgoing links

b) Forwards a packet to the next free outgoing link

c) Determines on which outgoing link a packet is to be forwarded

d) Forwards a packet to all outgoing links except the originated link

11. An Internet gateway works in the OSI model

a) At the Transport layer

b) At the Application layer

c) At the Session layer

d) At all the layers

15.3.6 RS-232C

RS- 232 is a standard for serial communication of data. It formally defines the signals connecting between a *DTE* (Data Terminal Equipment) such as a computer and a *DCE* (Data Communication Equipment such as a modem. The RS-232 standard is commonly used in computer serial ports. The standard defines the electrical characteristics and timing of signals, the meaning of signals and the physical size and pin-out of connectors. RS-232 stands for Recommend Standard number 232 and C is the latest revision of the standard. The serial ports on most computers use a subset of the RS-232C standard.

An RS-232 serial port was once a standard feature of a pc, used for connections to modems, printers, mouse etc. and other peripheral devices. However, RS-232 is hampered by low transmission speed, large voltage swing, and large standard connectors. In modern personal computers, USB has displaced RS-232 from most of its peripheral interface roles. Many computers no longer come equipped with RS-232 ports and must use either an external USB-to-RS-232 converter or an internal expansion card with one or more serial ports to connect to RS-232 peripherals. Nevertheless, RS-232 devices are still used, especially in industrial machines, networking equipment, and scientific instruments.

Many PCs and compatible computers are equipped with two serial ports and one parallel port. Although these two types of ports are used for communicating with external devices, they work in different ways.

A parallel port sends and receives data eight bits at a time over 8 separate wires. This allows data to be transferred very quickly; however, the cable required is more bulky because of the number of individual wires it must contain. Parallel ports are typically used to connect a PC to a printer. A serial port sends and receives data one bit at a time through one wire. Though it requires eight times more duration to transfer each byte than a parallel connection, only a few wires are required. In fact, full duplex means two-way communications is possible with only three separate wires - one to send, one to receive and a common signal ground wire.

The full RS-232C standard specifies a 25-pin “D” connector of which 22 pins are used. Most of these pins are not needed for normal PC communications. Most new PCs are equipped with male D type connectors having only 9 pins.

DCE and DTE Device

DCE stands for Data Communications Equipment and DTE stands for Data Terminal Equipment. These terms are used to indicate the pin-out for the connectors on a device and the direction of the signals on the pins. Computer is a DTE device, while most other devices are usually DCE devices.

The RS-232 standard states that DTE devices use a 25-pin male connector and DCE devices use a 25-pin female connector.

Therefore DTE device can be connected to a DCE device using a straight pin-for-pin connection. However, to connect two like devices, a null modem cable should be used. Null modem cables cross *transmit* and *receive* lines in the cable. The listing below shows the connections and signal directions for both 25 and 9-pin connectors.

25 Pin Connector on a DTE device (PC connection)

Male RS232 DB25 

Pin Number	Direction of signal:
1	Protective Ground
2	Transmitted Data (TD) Outgoing Data (from a DTE to a DCE)
3	Received Data (RD) Incoming Data (from a DCE to a DTE)
4	Request To Send (RTS) Outgoing flow control signal controlled by DTE
5	Clear To Send (CTS) Incoming flow control signal controlled by DCE
6	Data Set Ready (DSR) Incoming handshaking signal controlled by DCE
7	Signal Ground Common reference voltage
8	Carrier Detect (CD) Incoming signal from a modem
20	Data Terminal Ready (DTR) Outgoing handshaking signal controlled by DTE
22	Ring Indicator (RI) Incoming signal from a modem

9 Pin Connector on a DTE device (PC connection)

Male RS232 DB9 

Pin Number	Direction of signal:
1	Carrier Detect (CD) (from DCE) Incoming signal from a modem
2	Received Data (RD) Incoming Data from a DCE
3	Transmitted Data (TD) Outgoing Data to a DCE
4	Data Terminal Ready (DTR) Outgoing handshaking signal
5	Signal Ground Common reference voltage
6	Data Set Ready (DSR) Incoming handshaking signal
7	Request To Send (RTS) Outgoing flow control signal

8	Clear To Send (CTS) Incoming flow control signal
9	Ring Indicator (RI) (from DCE) Incoming signal from a modem

The TD (transmit data) wire is the one through which data from a DTE device is transmitted to a DCE device. The RD (receive data) wire is the one on which data is received by a DTE device.

RTS means for Request To Send. This line and the CTS (Clear To Send) line are used when “hardware flow control” is enabled in both the DTE and DCE devices. The DTE device gives indication through this line to tell the remote device that it is ready and able to receive data. Sometimes the receive buffer is full and the DTE device is not able to receive data. Then it gives signal through this line to the DCE to stop sending data. When the DTE device is ready to receive data (i.e. when its receive buffer becomes empty), it will place this line back in the receiving condition. The complement of the RTS wire is CTS. The DCE device gives signal through this line to the DTE device that it is ready to receive the data. Likewise, if the DCE device is unable to receive data, it will place this line in the wait condition. Together, these two lines make up what is called RTS/ CTS or “hardware” flow control.

DTR stands for Data Terminal Ready. Its function is similar to the RTS line. DSR (Data Set Ready) is the companion to DTR in the same way that CTS is to RTS. Some serial devices use DTR and DSR as signals to simply confirm that a device is connected and is turned on. The DTR and DSR lines were originally designed to provide an alternate method of hardware handshaking.

CD stands for Carrier Detect. Carrier Detect is used by a modem to signal that it has made a connection with another modem, or has detected a carrier tone.

RI means Ring Indicator. A modem toggles the state of this line when an incoming call rings a phone.

The Carrier Detect (CD) and the Ring Indicator (RI) lines are only available in connections to a modem. Because most modems transmit status information to a PC when either a carrier signal is detected (i.e. when a connection is made to another modem) or when the line is ringing, these two lines are rarely used.

Sometime 9 to 25 Pin Adapters are required for interconnection

purpose. The following table shows the connections inside a standard 9 pin to 25 pin adapter.

9 Pin Connector	**25 Pin Connector**
Pin 1 DCD	Pin 8 DCD
Pin 2 RD	Pin 3 RD
Pin 3 TD	Pin 2 TD
Pin 4 DTR	Pin 20 DTR
Pin 5 GND	Pin 7 GND
Pin 6 DSR	Pin 6 DSR
Pin 7 RTS	Pin 4 RTS
Pin 8 CTS	Pin 5 CTS
Pin 9 RI	Pin 22 RI

Baud Rate Vs Bits Per Second

The baud unit is named after Jean Maurice Emile Baudot, who was an officer in the French Telegraph Service. Baud rate means the number of times per second that a line changes state – high or low. This is not always the same as bits per second (BPS). If two serial devices are connected together using direct cables then baud and BPS are the same. Thus, if the system is running at 19200 BPS, then the line is also changing states 19200 times per second.

But in a modem, this isn't the case. When a modem transfers signals over

a telephone line, the baud rate is actually limited to a maximum of 2400 baud. This is a physical restriction of the lines provided by the phone company. This limit can be crossed to 9600 or higher baud by using sophisticated phase modulation and data compression techniques.

Cables, Null Modems

Ideally all serial ports on every computer would be DTE devices with 25-pin male "D" connectors. All other devices to would be DCE devices with 25-pin female connectors. This would allow the use of a cable in which each pin on one end of the cable is connected to the same pin on the other end. Unfortunately, this is not the actual situation. Serial ports use both 9 and 25 pins and many devices can be configured as either DTE or DCE. In some other cases there may be use of completely non-standard or proprietary pin-outs. Because of this lack of standardization, a special cable called null

modem cable is often required.

Cables Lengths

The RS-232C standard imposes a cable length limit of 50 feet. But a cable can be as long as 10000 feet at baud rates up to 19200 if a high quality, well shielded cable is used. Unshielded cables may be effected by external environment. Even very short cables can pick up stray signals in electrically noisy environment. A cable can be greatly extended by using additional devices like optical isolators and signal boosters. Optical isolators use LEDs and Photo Diodes to isolate each line in a serial cable including the signal ground. Any electrical noise affects all the optically isolated cable equally thus canceling out the effect of any noise signals.

Baud Rate	Shielded Cable Length	Unshielded Cable Length
110	5000	1000
300	4000	1000
1200	3000	500
2400	2000	500
4800	500	250
9600	250	100

Null Modem Cables and Null Modem Adapter

If two DTE devices (or two DCE devices) are connected by using a straight RS-232 cable, then the transmit line on each device will be connected to the transmit line on the other device. Similarly the receive lines will likewise be connected to each other. A Null Modem cable or Null Modem adapter simply crosses receive and transmit lines so that transmit on one end is connected to receive on the other end and vice versa. In addition to transmit and receive, DTR & DSR, as well as RTS & CTS are also crossed in a Null modem connection.

Synchronous and Asynchronous Communications

There are two basic types of serial communications, synchronous and asynchronous. With Synchronous communications, the two devices initially synchronize themselves to each other and then continually send characters to stay in sync. Even when data is not really being sent, a constant flow of bits allows each device to know where the other is at any given time. That is, each character that is

sent is either actual data or an idle character. Synchronous communications allows faster data transfer rates than asynchronous methods, because additional bits to mark the beginning and end of each data byte are not required. The serial ports on IBM-style PCs are asynchronous devices and therefore only support asynchronous serial communications.

Asynchronous means "no synchronization" and thus does not require sending and receiving idle characters. However, the beginning and end of each byte of data must be identified by start and stop bits. The start bit indicate when the data byte is about to begin and the stop bit signals when it ends. The requirement to send these additional two bits cause asynchronous communications to be slightly slower than synchronous however it has the advantage that the processor does not have to deal with the additional idle characters.

An asynchronous line that is idle is identified with a value of 1, also called a mark state. By using this value to indicate that no data is currently being sent, the devices are able to distinguish between an idle state and a disconnected line. When a character is about to be transmitted, a start bit is sent. A start bit has a value of 0, also called a space state. Thus, when the line switches from a value of 1 to a value of 0, the receiver is alerted that a data character is about to come down the line.



CHECK YOUR PROGRESS

12. RS- 232 is a standard for
    a) Parallel communication b) Serial communication
    c) USB communication d) Wireless communication
13. Which of the following signals is not standard RS-232 signals?
    a) VDR b) RTS c) CTS d) DSR
14. RS-232 communication is
    a) Faster than parallel communication
    b) Faster than USB communication
    c) Faster than any form of communication
    d) Slower than parallel communication

15. The transmit and receive lines in a Null modem cable is
    a) crossed b) not crossed c) serially joined d) parallel joined
16. Baud rate means
    a) the number of times the line changes state – high or low in one second
    b) the number of bytes transmitted per second
    c) in all cases same as bit per seconds
    d) the cycles of the operating voltage
17. Synchronous communication is
    a) slower than asynchronous communication
    b) faster than asynchronous communication
    c) same in speed with asynchronous communication
    d) sometimes faster and sometimes slower than asynchronous communication

15.4 NETWORK ADAPTERS

A network adapter is a device that is used to connect to a network. Network adapter's types are generally dependent on network architecture and the medium used for transmission. For example, transmission can be through copper cabling or through optical fiber. Each computer on a network is identified by a unique number called physical address, which is actually network adapter's serial number. This address is also called MAC address. Most of the computers today have built in network adapters. However in early computers systems, network adapters were bought separately and installed in motherboard's PCI slot. As the earlier network adapters were external and were available in the form of a card they were also called network interface controller card.

A Network Adapter's job is to take digital data on an Ethernet represented in the form of 0s and 1s, and to convert it into electrical, radio or light signal for transmission. It also works the other way by converting electrical, radio or light signal from transmission medium to series of 0s and 1s. Hence it is called Transceiver that is a device which can both transmit and receive signals for communication. A transceiver represents a zero voltage by 0 and a positive voltage by 1. This method of representation of 0s and 1s is called encoding scheme and is dependent on the network architecture.

Some of the network adapters are listed below:

● ***Modem***: A modem is a type of network adapter that uses standard telephone lines to connect to the internet. A modem on the computer is connected via telephone line to the phone jack. The Internet service provider (ISP) provides a dial up phone number to access the resources on the network. Figure 16.1 shows a Modem.

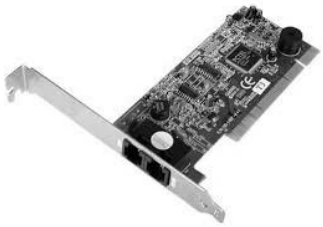


Figure 15.5: A Modem

● ***SFP Module:*** Small Form Factor Pluggable (SFP) are generally used in Routers and Switches to support any media type used by a port. Like SFP supports Copper or different types of optical fiber. Figure 16.2 shows a SFP module.



Figure 15.6: A SFP module

● GBIC: A GigaBit Interface Converter (GBIC) performs the same function as a SFP module. It is a larger-sized Transceiver that is designed for gigabit media including optical fiber and copper. Figure 16.3 shows a GBIC.



Figure 15.7: A GBIC

- ***Media Converter***: It is another type of network adapter that works at the physical layer. And is generally used to convert from one type of media to another type of media. For example an Ethernet network that uses copper cabling needs to use a media converter if it wants to connect to a server that supports fiber optic. A media converter works in a same network architecture and different transmission cable. For example it does not work from Ethernet to WAN architecture. Figure 16.4 shows a Media Converter.



Figure 15.8: A Media Converter.

15.5 NETWORK INTERFACE CARD (NIC)

If a user wants his computer to connect to a network or the internet, the user needs to install a network interface card (NIC). NIC is a circuit board which, when installed on a computer system, provides full time and dedicated connection to a network using an Ethernet cable with a RJ45 connector. Earlier computer systems lacked in built in NIC, so an external NIC card is installed in the PCI slot in the motherboard. However present day laptops or computers come with built in NIC cards designed especially for transmission over the Local Area Network. Also there are wireless network interface controller (WNIC) card which works by connecting to a wireless radio-based computer network instead of an Ethernet cable. The figure 16.5 a) and figure 16.5 b) shown below are examples of wired and wireless NIC card respectively.

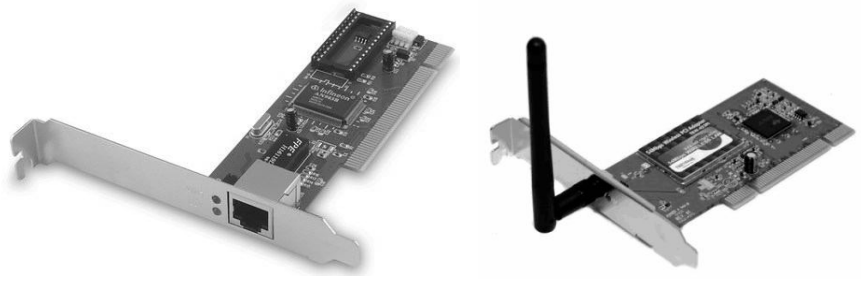


Figure 15.9 a) Wired NIC b) Wireless NIC

15.6 COMPONENT OF NIC

A network interface card contains a component called RJ45 connector port. This particular port serves as the point of connection between the system and the network. An optical fibre or a unshielded twisted pair (UTP) cable is inserted into the RJ45 connector port of the network interface card so as to establish the connection between the computer and the network.
A wireless network interface card in a laptop or a desktop computer uses radio waves to connect to a wireless access point.

15.7 FUNCTIONS OF NIC

Ethernet Network Interface Controller cards whether be wired or wireless has remained significant for servers and enterprises. These cards enable handling network traffic and controlling multiple connections. With the help of a Router or a Switch, a NIC card can handle a number of connections.
Today most offices connect their desktop computers with the help of Ethernet connections. Configuring the Ethernet cables is easy and provides superior performance than wireless NIC cards. As more and more offices are shifting towards cloud based technology, Ethernet cables are used to prevent congestion in wireless channels.
A network interface card acts as an intermediary between the user computer and the Internet. When a user opens a website in his/her computer, the computer sends the site information to the network interface card which then converts the site address information into electrical signal. The network carries this signal to the server on the internet. The site information is then sent back to the user via the network again till it reaches the network card that has initiated the request in the form of electric signal. The network interface card then converts the received signal back to digital data to be viewed by the user.

15.8 TYPES OF NIC

Network Interface Card is generally categorized based on their configuration into the following three types:

- **Jumper Configurable NIC Cards**
  Early computer equipments used Jumper to configure the devices without the need of software. Jumper configurable NIC cards also used

jumpers to control the computer hardware setting like type of transceiver, the interrupt line, upper memory block or input/output address.

- **Software Configurable NIC Cards**
  A software configurable NIC card comes with user manual and software to configure the device manually. The software contains a menu driven program for setting up the NIC card; the user may also use the auto configuration mode to select the most suitable configuration automatically.
- **Plug-and-Play Configurable NIC Cards**
  Today most of the devices are Plug and play configurable, that is, the user is not required to configure the device manually. As the device is plugged in it the auto configure at boot time. PnP NICs also once plugged in auto configure when the system boots.

15.9 ETHERNET

Ethernet was first developed in the year 1970 by Xerox, is capable of data transmission at 10Mbps. The latest technology available today also supports data transmission at 10Gbps. It is based on baseband architecture and has been successful in today's Local Area Network. The main reason for the success was the acceptance of Ethernet standard, IEEE 802.3. This allowed different vendors to fabricate network interfaces and media supporting the standard. The wireless equivalent of this Ethernet standard was IEEE 802.11a, b, g, n. The type of media supported by Ethernet includes twisted pair cables, coaxial cable or fiber optic cables. The best part of Ethernet media is that it does not require any power and is fail proof unless it is physically damaged or cut.

The Ethernet consists of three fundamentals:

- ***Physical Medium***: These are the media which are used to connect the systems.
- ***Medium Access Control Rules***: The rules that are used to access the Ethernet channel.
- ***Frame***: The data which need to be transmitted must follow a standard format.

The Ethernet allows all the systems connected over the network to access the channels independently, that is, there is no central control. The medium allows all the systems to receive and interpret the signal. Each system before transmitting data packet must check if the channel is free, otherwise

it may result in collision. For preventing collision IEEE standard has specified a mechanism called Carrier Sense Multiple Access with Collision Detection (CSMA/CD). The system before sending the data packet enters into Carrier sense mode, that is, it. senses the channel for presence of any other packet sent from some other system. It waits if the channel is not free, otherwise it places the data packet onto the channel. Actually, if the channel is free from any data packet all the systems have equal opportunity to place their respective data packet on to the channel. This condition is known as Multiple Access. In case two or more systems transmit data packets at the same time a detection state known as Collision Detection takes place. The system halts their transmission and waits for next Multiple Access state. In case of a collision the data packets must be resent by the respective systems.



15.10 LET US SUM UP

- Hubs are networking devices available in two types-passive and active.
- Switch is an active networking device work in layer2 and layer3 of OSI model.
- A big LAN can be sectionalized by using bridge to improve their performance.
- Routers are the workhorse of the subnet that provides the Internet service.
- Router forwards packets to their destination by consulting either a static or a dynamic routing table.
- Gateways are networking device that work in network layer of OSI model.
- Gateway works to interconnect networks that may operate on different protocols.
- RS-232C is a standard for serial communication.
- USB communication port greatly replaced the RS-232 port now-a-days.
- Baud rate is the number of times a line changes state from high to low or vice versa in one second.
- Bit per second (bps) is the number of bits transmitted per second.
- Baud rate and bps are same in certain cases, but not always.
- A Null Modem Cable is used to interconnect two DTE or DCE devices.



15.11 FURTHER READING

- Tanenbaum, A.S., & Wetherall, D. (2011). *Computer Networks.* Boston: Pearnson Prentice Hall.
- Forouzan, A.B. (2006). *Data Communications & Networking (sie).* Tata Mc Ggraw- Hill Education.
- Stallings, W. (2007). *Data and Computer Communications.* Pearson Education India.



15.12 ANSWER TO CHECK YOUR PROGRESS

Answer 1: (c)	Answer 2: (c)	Answer 3: (a)	Answer 4: (a)
Answer 5: (b)	Answer 6: (b)	Answer 7: (c)	Answer 8: (a)
Answer 9: (b)	Answer 10: (c)	Answer 11: (d)	Answer 12: (b)
Answer 13: (a)	Answer 14: (d)	Answer 15: (a)	Answer 16: (a)
Answer 17: (b)			



15.13 MODEL QUESTIONS

Q 1. Distinguish between a passive hub and an active hub.

Q 2. Distinguish between a hub and a switch.

Q 3. Describe the working principle of a switch.

Q 4. What is a network bridge? Explain its working principle.

Q 5. Describe router and its operating principle.

Q 6. Define gateway? How does it work in a computer network?

Q 7. Name the International Internet Gateway of India.

Q 8. Describe the RS-232 serial communication standard.

Q 9. Draw the DB25 connector and describe its various pin-outs.

Q 10. Draw the DB9 connector and describe its various pin-outs.

Q 11. Tabulate the connections of 9 pin to 25 pin RS-232 adapter.

Q 12. Describe synchronous and asynchronous serial communication.

Q 13. Define baud rate? When does it become same with bit per second? Explain.

Q 14. Draw a Null Modem Cable connections.

****