SYLLABUS

In-Sem. Exam

Unit I

Basics of Network and Physical Layer :

Types of networks, Network topologies, Design issues for layers, Network models, OSI model and TCP / IP protocol suite, Types of addressing.

Unit II

Data Link Layer :

Data link control, Framing, Flow and error control, Protocols for noiseless, and noisy channels, HDLC, Point to point protocol, **Media access control :** Random access, Controlled access- reservation, Channelization protocols.

End-Sem. Exam

Unit III

Network Layer - I :

Introduction to network layer : Network-layer services, Circuit switching, Packet switching, Network-layer performance, IPv4 addresses, Forwarding of IP packets, **Network layer protocols :** Internet Protocol (IP), ICMPv4, **Next Generation IP :** IPv6 addressing, The IPv6 protocol, The ICMPv6 protocol, Transition from IPv4 to IPv6.

Unit IV

Network Layer - II :

Unicast and Multicast Routing : Introduction, Routing algorithms, Unicast routing protocols, Introduction, multicasting basics, Intra-domain multicast protocols, Inter-domain multicast protocols, IGMP Distance vector, Link state, Path vector, **Routing in Internet :** RIP, OSPF, BGP.

Unit V

Transport Layer :

Introduction to transport layer, User datagram protocol, Transmission control protocol, TCP congestion policy, Stream control transmission protocol, Congestion control and QoS, socket programing.

Unit VI

Application Layer :

Introduction to application layer, **Standard client server Protocols :** World Wide Web and HTTP, Telnet, FTP, Email, SMTP, IMAP, POP, DNS, BOOTP, DHCP.

❑❑❑

Computer Networks

Chapter 4 : Network Layer- I

Q. 1 Explain different network layer services.

May 09, Dec. 13, Dec. 16

Ans. :

A. Services provided at the source computer :

- The following four services are provided by the network layer at the source computer :
  1. Packetizing.
  2. To find the logical address of the next hop.
  3. To find the physical or MAC address for the next hop.
  4. Fragmentation of the datagram if necessary.

1. Packetizing :

- Packetizing is the first duty of the network layer in which it encapsulates the payload (data received from the transport layer) in a packet at network layer at the source.
- Then at the destination the decapsulation process takes place.
- In this way the network layer is doing the job of a postal service in delivering the packages from source to destination.

At the source :

- At the sending end the events take place in the following sequence :
  1. The payload (data) from the upper layer is received.
  2. A header containing the source and destination address and some other information is added to the payload.
  3. This packet is then delivered to the data layer.
  4. If the payload is too large, then the host carries out **fragmentation** on it. Otherwise the host is not allowed to modify the contents of the payload.

2. Finding the logical address of the next hop :

- The datagram prepared with packetizing contains the source and destination addresses of the packet.
- The datagram is to be delivered to the next router. But the source and destination addresses in the datagram do not give any information about the logical address of the next hop.
- The network layer at the source computer finds the logical address of the next hop by consulting a routing table.

3. Finding MAC address of next hop :

- Note that it is the duty of data link layer (and not of network layer) to actually deliver the datagram to the next hop.
- And to do this the data link layer needs the physical or MAC address of the next hop.
- The network layer uses another table to map the logical address of next hop into the corresponding MAC address of next hop.
- However generally an auxiliary protocol called as ARP (Address Resolution Protocol) is used for this purpose.

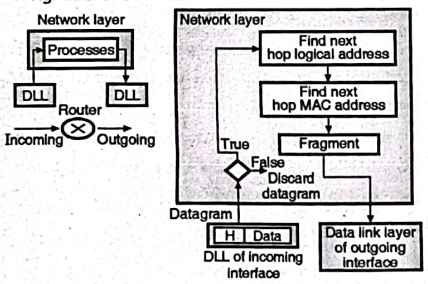
4. Fragmentation :

- The datagram at this stage may not always be ready to be given to the data link layer.
- The LANs and WANs can carry the data of a limited size in a frame.
- If the data is longer than the maximum specified size for LANs and WANs then it is not possible to fit it in one frame.
- In such circumstances, the datagram should be **fragmented** in to smaller data units before passing it to the data link layer.
- The datagram header is copied into all these fragments so that all the necessary information in the datagram is present in every fragment.

- In addition to this some more information regarding the position of that fragment in the whole datagram should be added to the header of the fragment.

B. Services provided at each router :

- The routers present in between the source and destination are supposed to check the source and destination addresses in the packet in order to forward it to the next network on the path.
- The router is not allowed to decapsulate the received packet unless it is too big and fragmentation needs to be carried out on it.
- The routers are not supposed to change the source and destination addresses.
- In the event of fragmentation, a router has to copy the header in all the fragments.
- At the router the services provided by the network layer are as follows :
  1. To find the next hop logical address.
  2. To find the next hop MAC address.
  3. To carry out fragmentation if required.
- Fig. 4.1 shows all these services.



(a) Process at the router (b) Services provided at the router

(G-2000) **Fig. 4.1**

- Before providing the services mentioned above the router checks the validity of the incoming datagram with the help of checksum.
- In checking the validation, the following two things are checked :
  1. Whether the datagram header is corrupted.
  2. Whether the datagram is delivered to the correct router.
- If the incoming datagram fails the validation test then it is simply discarded as shown in Fig. 4.1(b).

Other services :

- The other services expected from the network layer are as follows :
  1. Error control.
  2. Flow control.
  3. Congestion control.
  4. Quality of service (QoS).
  5. Security.

Q. 2 Compare virtual circuit and datagram networks.

May 07, Dec. 08, Dec. 09, Dec. 12

Ans. :

Comparison circuit and packet switching :

Table 4.1 : Comparison circuit and packet switching :

Sr. No.	Circuit switching	Datagram packet switching	Virtual-circuit packet switching
1.	Dedicated transmission path	No dedicated path	No dedicated path
2.	Continuous transmission of data	Transmission of packets	Transmission of packets
3.	Fast enough for interactive	Fast enough for interactive	Fast enough for interactive
4.	Messages are not stored.	Packets may be stored until delivered.	Packets stored until delivered.
5.	The path is established for entire conversation.	Route established for each packet	Route established for entire conversation
6.	Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay; packet transmission delay

Q. 3 Explain different performance parameters of network layer. Dec. 16

Ans. :

Performance parameters of network layer :

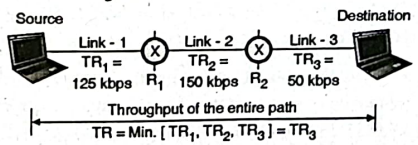
1. Delay.
2. Throughput.
3. Packet loss.
4. Congestion control.

1. Latency (Delay) :

- The latency or delay is defined as the time required for an entire message to reach its destination from the instant at which the first bit was sent out from the source.
- Latency is the sum of four delay components viz :
  1. Processing delay.
  2. Queueing delay.
  3. Transmission delay.
  4. Propagation delay.
- The sum of all these delays amounts to the total delay or latency.

2. Throughput (T) :

- The throughput is a parameter that is used to know the speed of data transmission over a network.
- The throughput of a system is defined as the actual rate at which the information is sent over the channel.
- It is measured in bits/second or frames/second.
- Throughput is a measure of performance of any network.
- However the throughput is not the same at every point in the network.
- It can have different values at different points.
- There we can define the throughput at any point of the network as the number of bits passing through that point per second.
- A packet passes through several links (networks) while travelling from a source to destination.
- Each link can have a different transmission rate as shown in Fig. 4.2.



(G-2228) **Fig. 4.2 : Throughput of a path**

- In such situations, the throughput of the entire path will be equal to the minimum of the transmission rates of different links.
- For Fig. 4.2, the throughput of the entire path is equal to 50 kbps i.e. the data rate of link-3.
- In general, the throughput of a path having "n" links in series is given by,

  Throughput = Minimum [TR_1, TR_2, ...TR_n]

- The definitions of bandwidth and throughput appear to be the same but they are not. They are different.
- If B is the bandwidth and T is the throughput of a network then T is always less than and at the most equal to B.
- Thus bandwidth is the theoretical measurement while throughput is the actual measurement of how fast data can be sent.

3. Packet loss :

- The performance of a network is dependent on one more factor, i.e. the number of packets lost during the communication.
- When a packet arrives at a router, when some other packet is being processed, the recently arrived packet is stored in the input buffer of the router and waits for its turn.
- But the size of input buffer is however finite. Therefore, a time may come when it will become full and the next packet arriving at the rotor input has to be dropped.
- The lost packets need to be retransmitted by the source, this may result in overflow of the input buffer and more loss of packets.
- Network designers have put in lots of time and efforts for design of queues for preventing the overflow and reducing the packet loss.

4. Congestion control :

- An important issue in a packet switching network is congestion.
- If an extremely large number of packets are present in a part of a subnet, the performance degrades.
- This situation is called as congestion.
- Congestion in a network may occur when the load on the network i.e. the number of packets sent to the network is greater than the capacity of the network (i.e. the number of packets a network can handle).

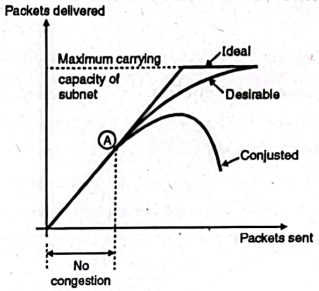
Q. 4 Write a short note on congestion control.

May 12, May 16, May 19, Dec. 19

Ans. :

Congestion control :

- If an extremely large number of packets are present in a part of a subnet, the performance degrades. This situation is called as congestion.
- Congestion in a network may occur when the load on the network i.e. the number of packets sent to the network is greater than the capacity of the network (i.e. the number of packets a network can handle).
- Fig. 4.3 explains the concept of congestion graphically.



(G-473) **Fig. 4.3 : Concept of congestion**

- Up to point A in Fig. 4.3, the number of packets sent into the subnet by the host is within the capacity of the network.
- So all these packets are delivered.
- In short the number of packets delivered is proportional to number of packets sent and no congestion takes place.
- But after point A, the traffic increases too far.
- The routers cannot cope with the increased traffic and they begin to lose packets. The congestion begins here.
- As the traffic increases further, the performance degrades more and more packets are lost and congestion worsens.
- At very high traffic, the performance collapses completely and almost all packets are lost.
- This is the worst possible congestion.

Need of congestion control :

- It is not possible to completely avoid the congestion but it is necessary to avoid it otherwise control it.
- Congestion will result in long queues, which results in buffer overflow and loss of packets.
- So congestion control is necessary to ensure that the user gets the negotiated QoS (Quality of Service).

Q. 5 What are the causes of congestion in a network ?

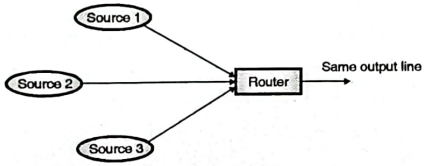
Dec. 10, Dec. 12

Ans. :

Causes of congestion :

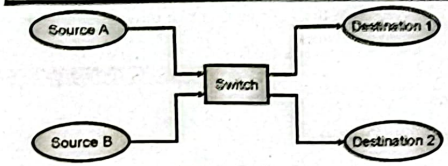
- Some of the causes of congestion are as follows :

1. If suddenly a flow of packets start coming on three or four senders which all needing the same output line.
   - Then a queue will become long.
   - If the memory capacity is not sufficient to hold all these packets, some of them will be lost.
   - This is shown in Fig. 4.4(a).



(G-474) **Fig. 4.4(a) : Causes of congestion**

   - This leads to congestion. Note that increasing the memory to infinity also does not solve the problem, in fact it worsens.
2. Congestion is caused by slow and low bandwidth links.
   - The problem will be solved when high speed links become available.
   - It is not always the case, sometimes increases in link bandwidth can aggravate the congestion problem because higher speed links may make the network more unbalanced.
   - For the configuration shown in Fig. 4.4(b), if both the sources begin to send to destination 1 at their maximum rate, congestion will occur at the switch.



(G-475) **Fig. 4.4(b) : Network with high speed links**

   - Higher speed links can make the congestion condition in the switch worse.
3. Congestion is caused by slow processors. The problem will be solved when processor speed is improved.
   - Faster processors will transmit more data in unit time.
   - If several nodes begin to transmit to one destination simultaneously at their maximum rate, the destination will be overwhelmed soon.
4. Congestion can make itself worse. If a router does not have any free buffers it should ignore (discard) new packets arriving at it.
   - But when a packet is discarded, the sender may retransmit it many times because it is not receiving the acknowledgement of the packet.
   - This multiple transmission of packets will force the congestion to take place at the sending end.

Q. 6 Explain the difference between flow control and congestion control.

May 07, Dec. 08, Dec. 13

Ans. :

Difference between flow and congestion control :

- Congestion control makes it sure that the subnet is able to carry the offered traffic i.e. the subnet is able to carry all the packets sent by all the senders to their destinations.
- Congestion control is dependent on the behaviour of all the hosts, all the routers and other factors which reduce the carrying capacity of a subnet.
- On the contrary, the flow control is related to point to point traffic between a sender and its destination.
- Flow control ensures that a fast sender does not send data at a rate faster than the rate at which the receiver can receive it.
- Flow control involves some kind of feedback from the receiver, which can control the sending rate of the sender.

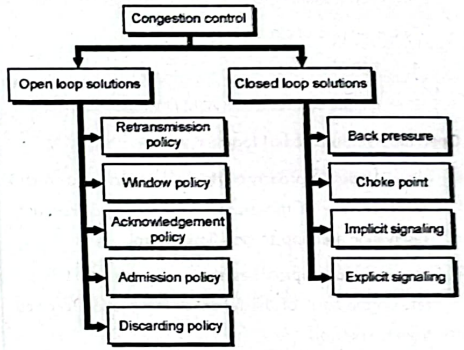
Q. 7 List various open-loop and closed-loop congestion control techniques. What is the difference between open-loop congestion control and closed-loop congestion control ?

Dec. 06, Dec. 07, May 09, Dec. 11

Ans. :

Classification of congestion control schemes :

- Fig. 4.5 shows the classification of congestion control schemes and various policies used in open loop and closed loop groups.



(G-476) **Fig. 4.5 : Classification of congestion control schemes**

Open loop control :

- Open loop solutions try to solve the congestion issue by excellent design to prevent the congestion from happening.
- Open loop control is exercised by using the tools such as deciding when to accept the new packets, when to discard the packets, which packets are to be discarded and making the scheduling decisions at various points.
- It is important to note that none of these decisions are made on the basis of the current status of a network, as no feedback is being used.

Closed loop control :

- The closed loop congestion control uses some kind of feedback.
- It takes into account the current status of the network.
- A closed loop control is based on the following three steps :
  1. Detect the congestion and locate it by monitoring the system.
  2. Transfer the information about congestion to places where action can be taken.
  3. Adjust the system operations to correct the congestion.
- Two examples of closed loop control are :
  1. TCP flow control.
  2. BR rate control for an ATM network.

Open loop versus closed loop :

- Open loop approaches do not need end-to-end feedback, one of the examples of this type are prior-reservation and hop-to-hop flow control.
- In closed-loop approaches, the source can adjust its cell rate on the basis of the feedback information received from the network.
- Some people feel that closed loop congestion control schemes are too slow in todays high-speed, large range network.
- Because it takes a long time for feedback to go back to source.
- Hence before any corrective action takes place thousands of packets have been already lost.
- But on other hand, if the congestion has already taken place and the overload is of long duration, the congestion cannot be released unless the source causing the congestion is asked to reduce its rate.
- Furthermore, ABR service is designed to use any bandwidth that is left over the source must have some knowledge of what is available when it is sending cells.

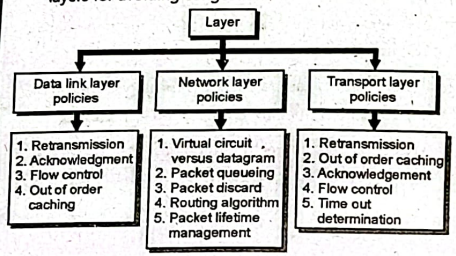
Q. 8 Explain the policies that can prevent congestion in a network.

Dec. 07, Dec. 09, May 10, Dec. 11

Ans. :

Prevention policies :

- Fig. 4.6 lists various policies corresponding to different layers for avoiding congestion.



(G-477) **Fig. 4.6 : Policies affecting the congestion**

Policies related to data link layer ;

1. Retransmission policy :

- The retransmission policy and the retransmission timers must be designed to optimise efficiency and at the same time prevent congestion.
- The retransmission policy deals with how fast a sender times out.
- If a sender times out early then it will retransmit all the packets and such a retransmission can lead to congestion.
- By designing the retransmission policy we can avoid this and prevent congestion.

2. Out of order caching policy :

- If the receivers routinely discard all the packets which are out of order, then retransmission of these packets will take place.
- This will increase the load and result in congestion. So a selective repeat (retransmission) should be adopted to avoid congestion.

3. Acknowledgement policy :

- If each received packet is promptly acknowledged then the acknowledgement packets will increase the traffic.

- If the acknowledgement is delayed (piggybacking) then there is a possibility of time out and retransmission.
- So a tight flow control has to be exercised to avoid congestion.

4. Window policy :

- The type of window at the sender may also affect congestion.
- The selective repeat window is better than the Go Back N window.

Policies related to network layer :

1. Choice between virtual circuit and datagrams :

- This choice at the network layer will affect the congestion because many congestion control algorithms work only with virtual circuit subnets.

2. Packet queuing and service :

- This policy is related to whether the routers have one queue per input line and one queue per output line or both.
- This policy is also related to the order in which the packets are processed e.g. round robin or priority based etc.

3. Discard policy :

- This policy lays a rule which tells the routers about which packet is to be discarded.
- A good discard policy can prevent congestion and a bad one will worsen the situation.

4. Routing algorithms :

- The routing algorithms can spread the traffic over all the lines.
- By doing so it is ensured that none of the lines are overloaded. This will certainly avoid congestion.

5. Package lifetime management :

- This policy decides the maximum time for which a packet may live before being discarded.
- This time should be of adequate value so that congestion can be avoided.

Policies related to transport layer :

- The policies at the transport layer are same as those at the data link layer.
- But at transport layer determining the time out interval is more difficult.
- If it is too short then extra packets are sent unnecessarily whereas if it is too long, congestion will reduce at the cost of increased response time (network will become slow).

Traffic shaping :

- One of the important reason behind congestion is the bursty nature of the traffic.
- If the traffic has a uniform data rate then congestion would not happen every now and then.
- But due to bursty traffic it can happen regularly.
- Traffic shaping is an open loop control. It prevents the congestion by making the packet transmission rate to be more predictable (bursty traffic is unpredictable).
- Thus traffic shaping will regulate the average rate or the burstiness of data transmission.
- Monitoring a traffic flow is called as **traffic policing**.
- Check if a packet stream (connection) is as per its descriptor, and if it is not as per its descriptor, then give penalty !
- In order to achieve this the network may want to monitor the traffic flow during the connection period.
- The process of monitoring and enforcing the traffic flow is called traffic policing.
- The types of penalties enforced are as follows :
  1. Drop packets that violate the descriptor.
  2. Give low priority to the packets violating the descriptor.

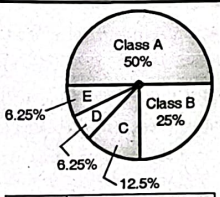
Q. 9 Explain various IPv4 address formats.

Dec. 10, May 13, Dec. 14, Dec. 15, May 17

Ans. :

IPv4 address formats :

- In the classful addressing architecture, the IP address space has been divided into five classes : A, B, C, D and E.
- Fig. 4.7 shows the percentage of occupation of the address space by each class.



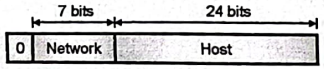
Class	No. of addresses	
A	2^{31}	50%
B	2^{30}	25%
C	2^{29}	12.5%
D	2^{28}	6.25%
E	2^{28}	6.25%

(G-2003) **Fig. 4.7 : Classful addressing occupation of address space**

- The number of class A addresses is the highest i.e. 50 % and those of classes D and E is the lowest i.e. 6.25 %.

Class A format :

- The formats used for IPv4 address are as shown in Fig. 4.7.
- The IPv4 address for class A networks is shown in Fig. 4.7(a).

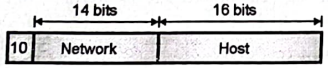


(G-531) **Fig. 4.7(a) : Class A IPv4 address formats**

- The network field is 7 bit long as shown in Fig. 4.7(a) and the host field is of 24 bit length.
- So the network field can have numbers between 1 to 126.
- But the host numbers will range from 0.0.0.0 to 127.255.255.255.
- Thus in class A, there can be 126 types of networks and 17 million hosts.
- The "0" in the first field identifies that it is a class A network address.

Class B format :

- The class B address format is shown in Fig. 4.7(b).

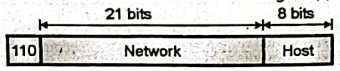


(G-532) **Fig. 4.7(b) : Class B format**

- The first two fields identify the network, and the number in the first field must be in the range 128 - 191.
- Class B networks are large. Host numbers 0.0 and 255.255 are reserved, so there can be upto 65,534 ($2^{16} - 2$) hosts in a class B network.
- Most of the 16,382 class B addresses have been allocated.
- The first block covers address from 128.0.0.0 to 128.255.255.255 and the last block covers from 191.255.0.0 to 191.255.255.255.
- **Example :** 128.89.0.26, for host 0.26 on net 128.89.

Class C format :

- The class C address format is shown in Fig. 4.7(c).



(G-533) **Fig. 4.7(c) : Class C format**

- The first block in class C covers addresses from 192.0.0.0 to 192.0.0.255 and the last block covers addresses from 223.255.255.0 to 223.255.255.255.

Class D format :

- The class D address format is shown in Fig. 4.7(d).

Fig. 4.7(d) : Class D format

- The class format allows for upto 2 million networks with upto 254 hosts each and class D format allows the multicast in which a datagram is directed to multiple hosts.

Class E address format :

- Fig. 4.7(e) shows the address format for a class E address.

Fig. 4.7(e) : IPv4 address for class E network

- This address begins with 11110 which shows that it is reserved for the future use.
- The 32 bit (4 byte) network addresses are usually written in dotted decimal notation.
- In this notation each of the 4-bytes is written in decimal from 0 to 255.

- So the lowest IP address is 0.0.0.0 i.e. all the 32 bits are zero and the highest IPv4 address is 255.255.255.255.

Q. 10 What is a mask in IPv4 addressing ? What is its default value ? Dec. 07

Ans. :

Network mask or Default mask :

- A network mask or default mask in classful addressing is defined as a 32-bit number obtained by setting all the "n" leftmost bits to 1s and all the (32 – n) rightmost bits to 0.

Default masks for different classes :

Table 4.2 : Default masks

Address class	Default mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

Q. 11 Briefly define subnetting. How do the subnet mask differ from a default mask in classful addressing ? Dec. 18

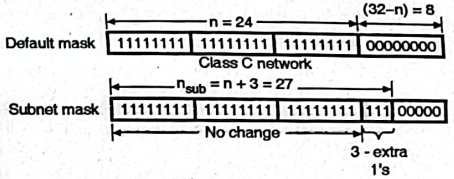
Ans. :

Definition of subnetting :

- We can define the subnetting as the principle of splitting a block of addresses into smaller blocks of addresses.
- In the process of subnetting we divide a big network into smaller subnetworks or subnets.
- Each such subnet has its own subnet address.

Subnet mask :

- The **network mask** or **default** mask is used when the given network is **not** to be divided into smaller subnetworks i.e. when **subnetting** is **not** to be done.
- But when the given network is to be divided into smaller subnets i.e. when subnetting is to be done, we need to create a **subnet mask** for each subnet.
- Fig. 4.8 shows the format of a subnet mask. Each subnet has its own **net id** and **host id.**
- If we want to divide a network into 8 subnets then the corresponding subnet mask will have three extra 1's because $2^3 = 8$, as compared to the default mask, as shown in Fig. 4.8.



(G-2011) **Fig. 4.8 : Default and subnet masks**

- In Fig. 4.8, we have shown the default mask and subnet mask when a class C network is to be divided into 8 subnets.

Difference between subnet mask and default mask :

Table 4.3 : Difference between subnet mask and default mask

Sr. No.	Subnet mask	Default mask
1.	To divide a given network address into two or more subnets, subnet mask is used.	The default mask signify a network without subnets.
2.	In office network the subnet mask is used.	In home networks default mask is used.
3.	A subnet mask can be changed as per the hosts / subnets requirement.	A default mask cannot be changed. It is fix for particular address class.
4.	Subnet mask is used to distinguish network part and host part in a IP address.	Default mask is subnet mask for a class of network.
5.	Subnet mask represents the number of bits used by network and rest for host.	Default mask is inbuild network portion which is already defined in IPv4 classes.

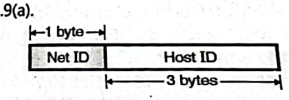
Q. 12 Show by calculations how many network each IP address class can have with one example ? Dec. 06

Ans. :

Number of networks in different IP address :

Class A address :

- The format of class A address is shown in Fig. 4.9(a).

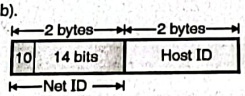


(G-560) **Fig. 4.9(a) : Class A address**

- Here one byte defines the network ID and three bytes define the host ID.
- The MSB in the network field is reserved. So actually there are only 7-bits in the network fields.
- So the number of networks in class A address will be 128.

Class B address :

- The format of class B address is shown in Fig. 4.9(b).

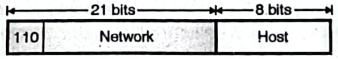


(G-561) **Fig. 4.9(b) : Class B address**

- Here 2-bytes are reserved for network field and remaining two bytes are for the host field.
- Out of 16-bits in the network field the first two bits (MSBs) are reserved. So actually 14 bits are available in the network field.
- So the number of networks in class B address is 2^{14} = 16, 368.

Class C address :

- The format of class C is shown in Fig. 4.9(c).
- Here 3-bytes are reserved for network field and only one byte for the host field.



(G-562) **Fig. 4.9(c) : Class C address**

- Out of 24-bits in the network field 3-bits are again reserved. So actually only 21-bits are available.
- So the number of networks in class C addresses is 2, 097, 152.

Q. 13 A router has following CIDR entries in its routing table :

Address/Mask	Next Hop
135.46.56.0/22	Interface 0
135.46.60.0/22	Interface 1
192.53.40.0/23	Router 1
Default	Router 2

For each of the following IP addresses, what does the router do if a packet with that address arrives ?

1. 135.46.63.10 2. 192.53.56.7

Dec. 10, Dec. 11, May 16

Ans. :

CIDR – Classless Inter Domain Routing :

- IP is being heavily used for decades.
- However, due to the exponential growth of internet, IP is running out of addresses.
- The CIDR is based on the principle of allocating the remaining IP addresses in variable-sized blocks regardless of the class.
- If a site needs say 2000 addresses, then a block of 2048 addresses on the 2048 byte boundary is given to it.
- However the classless routing makes forwarding of packets more complicated.

Forwarding algorithm in the old classful system :

- The steps followed in the old classful system for forwarding packets is as follows :
  1. As soon as a packet arrives at a router, a copy of the IP address was shifted right by 28 bits to obtain a 4 bit class number.
  2. A 16-way branch then sorts packets into class A, B, C and D (if supported) with eight of the cases for class A, four of the cases for class B, two of the cases for class C and one each for D and E.
  3. The code for each class then masked off the 8–, 16–, or 24– bit network number and right aligned it in a 32 bit word.
  4. The network number was then searched in the A, B or C table.
  5. As soon as the entry was found, the outgoing line was decided and the packet was forwarded upon it.

Forwarding with CIDR :

- The simple forwarding algorithm does not work with CIDR.
- Instead now each router table entry is extended by giving if a 32 bit mask.

- So now there is a single routing table for all networks (no different tables for class A, B, C, etc.) which consists of an array of triples. Each triple consists of an **IP address, subnet mask** and **outgoing line.**
- When a packet arrives at the input, the router first extracts its destination IP address.
- Then the routing table is scanned entry by entry to look for a match.
- It is possible that different entries with different subnet mask lengths match.
- In such a case the longest mask is used. For example if there is a match for a/20 mask and a/24 mask then /24 entry is used.

Solution of problem :

- Convert the IP address to bits and then AND it with the subnet mask of the interface whose address is closest to that of the IP addresses.
- The result of the ANDing will give you the network address and the interface to send the packet to.

1. IP = 135.46.63.10 :

- The interface whose address is closest to this IP is interface 1.
- This interface uses a 22 bit mask. So AND the given IP address with a 22 bit mask as follows :

(G-1973)

IP = 135.46.63.10 = 10000111.00101110.00111111.00001010
22 bit mask = 255.255.252.0 = 11111111.11111111.11111100.00000000

IP AND Mask = 10000111.00101110.00111100.00000000
∴ IP AND Mask = 135.46.60.0

- This result of ANDing matches with the network address of interface 1. Hence the router will forward this packet to interface 1.

2. IP = 192.53.56.7 :

- The interface whose address is closest to this IP is interface 2.
- This interface uses a 23 bit mask. So AND the packet IP address with a 23 bit mask as follows :

IP = 192.53.56.7 = 11000000.00110101.00111000.00000111
23 bit mask = 255.255.254.0 = 11111111.11111111.11111110.00000000

IP AND Mask = 11000000.00110101.00111000.00000000
= 192.53.56.0

(G-1974)

- This result of ANDing does not match with the network addresses of interface 0 or 1.
- Hence the packet will forwarded to the default i.e. Router 2.

Q. 14 Explain the concept of delivery and its different types used in network. **May 16**

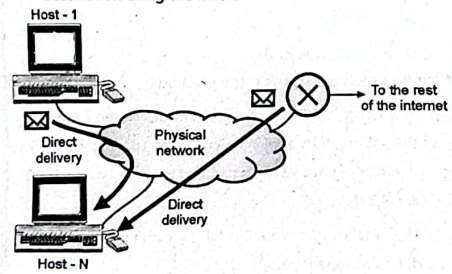
Ans. :

Concept of delivery :

- The network layer supervises how the packets are being handled by the underlying physical networks.
- This handling is known as the delivery of packets.
- The two different methods of delivery are :
  1. Direct delivery.
  2. Indirect delivery.

1. Direct delivery :

- In the direct delivery the destination host and the one who delivers the packet are in the same physical network as shown in Fig. 4.10(a).
- The sender can extract the network address of the destination using the mask.

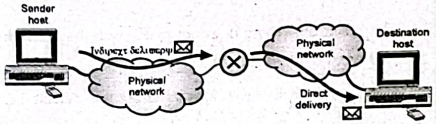


(G-440) **Fig. 4.10(a) : Direct delivery**

- It then compares this address with the addresses of the networks to which it is connected.
- If these two addresses are identical then the delivery is direct.

2. Indirect delivery :

- In the indirect delivery of packets, the sender host and the destination host are not the part of the same physical network as shown in Fig. 4.10(b).



(G-441) **Fig. 4.10(b) : Indirect delivery**

- In such a situation, the packets travel from one router to the other and are finally delivered to the destination host.
- The indirect delivery involves one direct and zero or more indirect deliveries. The last delivery is always a direct one.

Q. 15 Explain different forwarding techniques used in computer network. May 12, Dec. 17

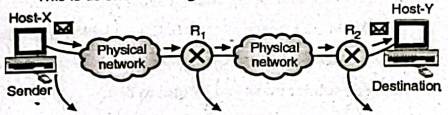
Ans. :

1. Forwarding techniques :

- Many techniques have been invented and tested in order to make the size of the routing tables manageable.
- Some of them are as follows :
  1. Next hop method versus route method.
  2. Network specific method versus Host specific method.
  3. Default method.

Next hop method versus route method :

- The Route method is the most basic method in which the information about the complete route is stored in the routing tables of hosts and routers as shown in Fig. 4.11(a).
- This makes the routing tables extremely large and difficult to manage.
- In order to reduce the size of routing tables, the next hop method is used in which the routing table contains only the address of the next hop (upto the next router) instead of information about the complete route.
- This is as shown in Fig. 4.11(a).



Routing tables at Host-X — Based on route

Destination	Route
Host Y	R_1,R_2 Host Y

Routing tables at router R_1 — Based on route

Destination	Route
Host Y	R_2, Host Y

Routing tables at router R_2 — Based on route

Destination	Route
Host Y	Host Y

Based on next hop

Destination	Next hop
Host Y	R_1

Based on next hop

Destination	Next hop
Host Y	R_2

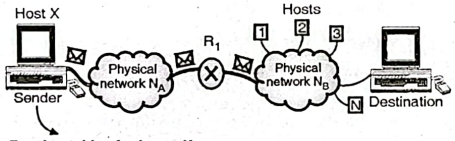
Based on next hop

Destination	Route
Host Y	–

(G-442) **Fig. 4.11(a) : Route method versus Next Hop method**

Network specific method versus host specific method :

- In the host specific method, the routing table of a host or router will specify each destination host connected to the same physical network.
- This increases the number of entries in a routing table and makes it large.
- But in the network specific method, we have only one entry corresponding to the destination network N_B only as shown in Fig. 4.11(b).



Routing tables for host - X

Host specific method

Destination	Next hop
Host - 1	R_1
Host - 2	R_2
⋮	⋮
Host - N	R_N

Network specific method

Destination	Next hop
Network N_B	R_1

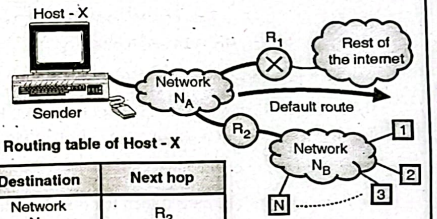
(G-443) **Fig. 4.11(b) : Host specific method versus network specific method**

- That means we consider all hosts connected to the same network N_B as one single entry.

- This will reduce the routing table and simplify the searching process considerably.

Default method :

- This is one more method of simplifying the routing tables. Refer Fig. 4.12 in which the sending host X is connected to a network with two routers R_1 and R_2.



Routing table of Host - X

Destination	Next hop
Network N_B	R_2
Any other network	R_1

(G-444) **Fig. 4.12 : Default method**

- Router R_2 routes the packets to the hosts connected to network N_B. However router R_1 is used for the rest of the Internet.
- Hence in the routing table instead of listing all networks in the entire Internet, host X will have only one entry called as the default entry (normally defined as network address 0.0.0.0).

Q. 16 List the various protocols giving their significance at network layer. Dec. 15

Ans. :

Various network layer protocols :

- Various network layer protocols and their functions (significance) are as listed below.

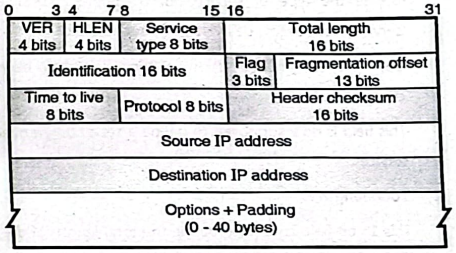
Sr. No.	Protocol	Function
1.	IP	Transports datagram from sender to destination. It acts like the postal service it is responsible for host to host delivery.
2.	ARP	It helps IP to find the MAC (physical address). It maps IP address to MAC address.
3.	ICMP	It is used alongwith IP to report presence of error and sends control message on behalf of IP. It provides feedback on special conditions.
4.	IGMP	It is a group management protocol used in multicasting environment alongwith IP.
5.	RARP	Mapping MAC address to IP address.

Q. 17 Draw IPv4 headers and explain briefly. May 10, Dec. 12, Dec. 15

Ans. :

IPv4 header format :

- The IP frame header contains routing information and control information associated with datagram delivery.
- The IP header structure is as shown in Fig. 4.13.



(G-2082) **Fig. 4.13 : IPv4 header format**

- Various fields in the header format are as follows :

1. VER (Version) :

- This is a 4 bit field which is used to define the version of IP protocol.
- The current version of IP is 4 i.e. IPv4 but in future it may be completely replaced by the latest version of IP i.e. IPv6.
- This field will indicate the IP software running on the processing machine that this datagram belongs to IPv4 version. If the processing machine is using some other version of IP, then the datagram will be discarded.

2. HLEN (Header length) :

- This 4-bit long field is used for defining the length of the datagram header in 4-byte words.
- The value of this field is multiplied by 4 to get the length of the IPv4 header which varies between 20 and 60 bytes.

- When there are no options, the value of this field is 5 and the **header length** is 5 × 4 = 20 bytes.
- When the value of option field is maximum the value of HLEN field is 15 and the corresponding header length is maximum i.e. 15 × 4 = 60 bytes.

3. Service type :

- In the earlier designs of IP header, this field was called as Type of Service (TOS) field and its job was to define how the datagram should be handled.
- At that time, a part of this field used to define the precedence of datagram and the remaining part used to define the type of service out of different possible services such as low delay, high throughput etc.
- But now the interpretation of this field has been changed by IETF.
- This field is now supposed to define a set of differential services.

4. Total length :

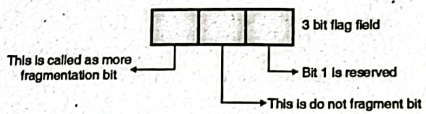
- This 16 bit field is used to define the total length of the IP datagram.
- The total length includes the length of header as well as the data field.
- The field length of this fields is 16 bits so the total length of the IP datagram is restricted to $(2^{16} - 1) = 65535$ bytes out of which 20 to 60 bytes constitute the header and the remaining bytes are reserved to carry data from upper layers.
- This field allows the length of a datagram to be upto 65,535 bytes, although such long datagrams are impractical for most hosts and networks.

5. Identification :

- This field is used to identify the datagram originating from the source host.
- When a datagram is fragmented, the contents of the identification field get copied into all fragments.
- This identification number is used by the destination to reassemble the fragments of the datagram.

6. Flags :

- **Flags :** This is a three bit field. The 3 bits are as shown in Fig. 4.14.



(G-527) **Fig. 4.14 : Flag bits**

- First bit is reserved, and it should be 0.
- The second bit is known as the "Do Not Fragment" bit. If this bit is "1" then machine understands that the datagram is not to be fragmented.
- But if the value of this bit is 0 then the machine should fragment the datagram if and only if necessary.
- The third bit is known as "More Fragment Bit" (M). M = 1 indicates that the datagram is not the last fragment and M = 0 indicates that this is the last or the only fragment.

7. Fragmentation offset :

- This is a 13 bit field which is used to indicate the relative position of this fragment with respect to the complete datagram.
- It is the offset of the data in the original datagram measured in units of 8 bytes.

8. Time to Live (TTL) :

- This is an 8-bit field which controls the maximum number of routers visited by the datagram during its lifetime.
- A datagram has a limited lifetime for travelling through an Internet.
- Originally the TTL field was designed to hold the timestamp.

9. Protocol :

- This is an 8-bit field which is used for defining the higher level protocol which uses the services of IP layer.
- The data from different high level protocols can be encapsulated into an IP datagram. These protocols could be UDP, TCP, ICMP, IGMP etc.

- The protocol field contents would tell the name of the protocol at the final destination to which this IP datagram is to be delivered.
- At the destination, the value of this field helps in the process of demultiplexing.

10. Header checksum :

- A checksum in IP packet covers on the header only. Since some header fields change, this field is recomputed and verified at each point that the Internet header is processed.

11. Source address :

- This field is used for defining the IP address of the source. It is a 32 bit field.

12. Destination address :

- This field is used for defining the IP address of the destination. It is also a 32 bit field.

13. Options :

- Options are not required for every datagram.
- They are used for network testing and debugging.

Q. 18 What is ICMPv4 ? Explain general format of ICMPv4 messages.

Dec. 16, Dec. 17, Dec. 16, May 19, Dec. 19

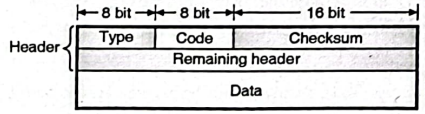
Ans. :

ICMPv4 :

- The long form of ICMPv4 is Internet Control Message Protocol version 4.
- The IP provides unreliable and connectionless datagram delivery and makes an efficient use of network resources.
- IP is a best-effort delivery (which does not provide any.
- The Internet Control Message Protocol (ICMP) is used to overcome these drawbacks.

Message format :

- Fig. 4.15 shows the general format of ICMP messages.



(G-2105) **Fig. 4.15 : General format of ICMP messages**

- As shown in Fig. 4.15, the header of an ICMP message is 8-byte long and the data section is of a variable size.
- The general header format for each ICMP message is different. But the first four bytes are common to all the message types.

1. Type :

- This 8-bit field is used for defining the types of message.

2. Code :

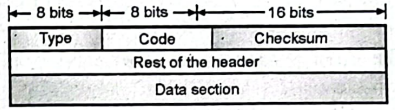
- This 8-bit field is used for specifying the reason for the particular message type.
- The last common field is the **checksum** field which is 16 bit (2 byte) long.
- The information to find the original packet that had error is included in the **data section** of the error messages.
- Whereas the **data section** in the query messages contains extra information depending on the type of query.

Q. 19 Give general format of ICMP and explain different types of error reporting messages used in ICMP. May 19, Dec. 19

Ans. :

General format of error reporting messages :

- Fig. 4.16 shows the general format of error reporting message in ICMPv4.



(G-2107(a)) **Fig. 4.16 : General format of error reporting message in ICMPv4**

- Depending on the values of the type and code fields, the type of error reporting message would change as shown in Table 4.4

Table 4.4

Sr. No.	Error reporting message	Type	Code
1.	Destination unreachable	03	0 to 15
2.	Source quench	04	0

Sr. No.	Error reporting message	Type	Code
3.	Time exceeded	05	0 to 3
4.	Parameter problem	11	0 and 1
5.	Redirection	12	0 and 1

1. Destination unreachable :

- When it is not possible for a router to route the datagram or when a host is unable to deliver a datagram, then the datagram is discarded and the destination unreachable error message is sent back by the respective host or router to the source host which originated the datagram.
- The general format of the destination unreachable error message is as shown in Fig. 4.16.
- The content of the type field for this error reporting message is 03.
- The code field for the destination unreachable error message has 16 different values (0 to 15) and each one specifies a reason for discarding a datagram.
- The destination host or routers can produce the destination unreachable message.
- Only the destination host can create code 2 and code 3 messages.
- The messages of other codes except codes 2 and 3 can be created only by the routers.
- The non-creation of destination unreachable message does not guarantee the delivery of datagram.
- It is not possible for the router to detect all the problems that prevent the packet delivery.

2. Source quench error message :

- A host or router uses source quench messages in order to tell the original source that congestion has occurred and to request it to reduce its current rate of packet transmission.

There is no flow control or congestion control mechanism in IP.

So the source quench message in ICMP is designed to add some kind of flow control and congestion control to IP.

- This message serves two purposes :
  1. It tells the source that the packet has been discarded.
  2. It gives a warning to the source that the source should slow down (quench) because congestion has taken place somewhere.
- Fig. 4.16 shows the format of the source quench error message.
- The content of the type and code fields for this error reporting message are 04 and 0 respectively.
- A source-quench message, one per discarded datagram due to congestion is sent back by a router or destination host, to the source host.
- But, the congestion relieved message cannot be sent to the source host as no such mechanism exists.
- As no such message could be sent back, the source host assumes that the congestion has continued to exist, and therefore it continues to reduce the rate of data transmission, until no more source-quench messages are received.
- The congestion can happen due to two types of communications :
  1. Due to one to one communication.
  2. Due to many to one communication.
- In the one to one communication, a single source host will be responsible for congestion because of its high data transmission rate.
- The source quench message will be useful under such operating conditions, for reducing the transmission rate of the source host and clear the congestion.
- But this message will not prove to be successful if congestion occurs in the many to one type communication.
- This is because the router or destination host does not know which source is fast and responsible for the congestion.
- As a result, it may discard the packets received from the slowest source instead of dropping them from a fast source which is actually responsible for congestion.

3. Time exceeded error message :

- This message is generated in two cases :
  1. If a router receives a packet with a 0 in the TTL field then it discards that datagram and send a time exceeded message back to the source originating that packet.
  2. If all the fragments which are parts of a message do not arrive at the destination host within a certain time limit then time exceeded message is sent back.
- The format of the time exceeded message is as shown in Fig. 4.16.
- The content of the type and code fields for this error reporting message are 11 and (0 or 1) respectively.
- If **code = 0,** then the router will discard the datagram because the value of TTL (time to live) field is zero.
- If **code = 1,** then destination host discards the fragments of datagram because some fragments could not arrive at the destination host within the time limit.

Parameter problem error message :

- There should not be any ambiguity in the header part of the packet.
- If a router or destination host comes across such ambiguity or missing value in any field of the datagram then it simply discards that datagram and sends the parameter problem message back to the source originating that message.
- This message can be created either by a router or the destination host.
- The content of the type and code fields for this error reporting message are 12 and (0 or 1) respectively.
  - (a) **If Code = 0 :** If code = 0, then the datagram is discarded because of an error or ambiguity present in one of the header fields. The erroneous byte is pointed at by the value of the pointer field. For example if pointer field = 0, then the first byte is an invalid field.
  - (b) **Code = 1 :** If code = 1, then the datagram is discarded because required part of an option is missing.
- The format of the parameter problem is as shown in Fig. 4.16.

5. Redirection error message :

- If a router or host wants to send a packet to another network then it should know the IP address of the next router.
- The routers and hosts must have a routing table to find the address of the next router and the routing table has to be updated automatically on a continuous basis.
- The redirection message is used for such updating.
- The ICMP sends a redirection message back to its host to carry out an automatic periodic updating.
- In order to ensure higher efficiency, the hosts do not participate in the process of routing table update.
- This is because the number of hosts in the Internet is much higher than the number of routers.
- If the routing tables of hosts are updated dynamically then it creates an unwanted traffic.
- Generally the static routing is used by the hosts. That means the routing table of a host contains limited number of entries.
- Generally a host knows the IP address of only one router that is the default router.
- Due to this, a host can send a datagram which is destined for another network, to a wrong router.
- Here the datagram receiving router will route the datagram the correct router.
- However it sends a redirection message to the host to update the routing table of the host.
- The format of the redirection message is as shown in Fig. 4.16.
- The content of the type and code fields for this error reporting message are 05 and (0 or 3) respectively.
- The second row of the redirection message contains the IP address of the appropriate target router.
- It is important to understand that the redirection message is different from the other error message even though it is considered as an error reporting message.

- What is the difference ? In this case the router does not discard the erroneous datagram. Instead it is sent to the appropriate router.
- This process of redirection is narrowed down by the contents of the **code field** as follows :
  - (a) **Code = 0 :** Redirection will be for a network specific route.
  - (b) **Code = 1 :** Redirection is to be done for a host specific route.
  - (c) **Code = 2 :** Redirection is to be done for a network specific route and based upon a specific type of service.
  - (d) **Code = 3 :** Redirection is to be done for a host specific route on the basis of a specified type of service.

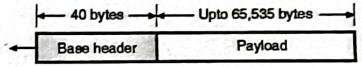
Q. 20 Draw and explain the header format for IPv6.

May 16, Dec. 16, May 19, Dec. 19

Ans. :

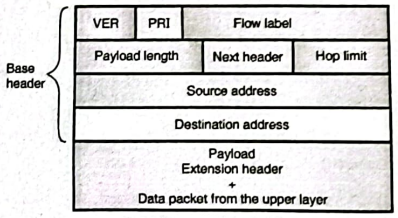
Header format for IPv6 :

- Fig. 4.17(a) shows IPv6 packet.



(G-2245) **Fig. 4.17(a) : IPv6 packet**

- Fig. 4.17(b) shows the packet format (Base header) of IPv6.



(G-550) **Fig. 4.17(b) : Format of an IPv6 datagram (Base header)**

- Each packet can be divided into two parts viz : base header and payload. Base header is the mandatory part and payload is an optional one.
- The payload follows the base header.
- The payload is made up of two parts :
  1. An optional extension headers.
  2. The upper layer data.
- The base header is 40 byte long whereas the payload consisting of the extension header and upper layer data can have information worth upto 65, 535 bytes.

Base header :

- Fig. 4.17(b) shows the base header. It has eight fields. These fields are as follows :

1. **Version (VER) :** The contents of this 4 bit field defines the version of IP such as IPv4 or IPv6. If VER = 6, then the version is IPv6.
2. **Priority :** This 4 bit field contents defines the priority of the packet which is important in connection with the traffic congestion.
3. **Flow label :** It is a 24 bit (3 byte) field which is supposed to provide a special handling for a particular flow of data.
4. **Payload length :** The contents of the 16 bit or 2 byte length field are used to indicate the total length of the IP datagram excluding the base header. That means it gives the length of only the payload part of the datagram.
5. **Next header :** It is an 8 bit field which defines the header which follows the base header in the datagram.
6. **Hop limit :** Contents of this 8 bit (1 byte) field have the same function as TTL (time to live) in IPv4.
7. **Source address :** It is a 16 byte (128 bit) Internet address which corresponds to the originator or source which has produced the datagram.
8. **Destination address :** This is a 16 byte (128 bit) internet address which corresponds to the address of the final destination of datagram.

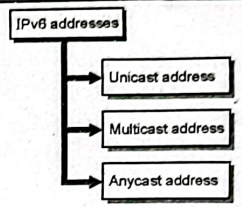
   But this field will contain the address of the next router and not the final destination if source routing is being used.

Q. 21 List and explain different types of addresses used in IPv6. Dec. 13, Dec. 17, Dec. 18

Ans. :

IPv6 addresses :

- IPv6 defines three different types of addresses as shown in Fig. 4.18.



(G-549) **Fig. 4.18 : Types of addresses**

1. **Unicast address :**
- A unicast address is meant for a single computer as a destination. A packet sent to a unicast address is meant to be delivered to the computer specified by the address.
- In IPv6 a large block of addresses has been designated from which it is possible to assign unicast addresses to the interfaces.

2. **Anycast address :**
- This is a type of address which is used to define a group of computers with addresses which have the same prefix.
- A packet sent to an anycast address must be delivered to only one of the member of the group which is the closest or the most easily accessible.
- No special or separate address block is assigned for anycasting in IPv6.
- Instead the anycast addresses are assigned from the block of unicast addresses.

3. **Multicast addresses :**
- A multicast address defines a group of computers which may or may not share the same prefix and may or may not be connected to the same physical network.
- A packet sent to a multicast address is meant to be delivered to each member of the group.
- There are no broadcast addresses in IPv6, because multicast addresses can perform the same function.
- The type of address is determined by the leading bits.
- All the multicast addresses start with FF (1111 1111) and all other addresses are unicast addresses.
- Anycast addresses are assigned from the unicast address space and they do not differ syntactically from unicast addresses.
- Anycast addressing is a rather new concept and there is not much experience about the widespread use of anycast addresses.
- Therefore, some restrictions apply to anycast addressing in IPv6 until more experience is gained.
- An anycast address may not be used as the Source Address of an IPv6 packet and anycast addresses may not be assigned to hosts but to routers only.
- A block is designated for multicasting in IPv6, from which the same address is assigned to the members of the group.

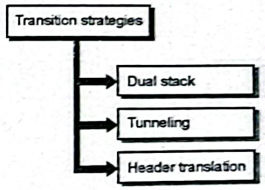
Q. 22 Explain transition strategies from IPv4 to IPv6 ?

Dec. 17

Ans. :

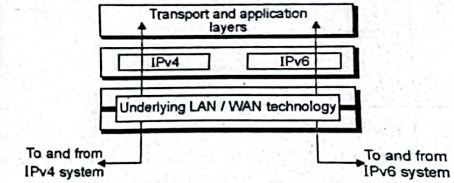
Transition strategies from IPv4 to IPv6 :

- Fig. 4.19(a) shows the strategies for transition from IPv4 to IPv6.



(G-2531) **Fig. 4.19(a) : Transition strategies**

1. **Dual stack :**
- Before completely migrating to version 6 it is recommended that all hosts should have a dual stack of protocols at the time of transition.
- Simultaneously station should run IPv4 and IPv6, until the Internet uses IPv6.
- The layout of dual stack configuration is as shown in Fig. 4.19(b).



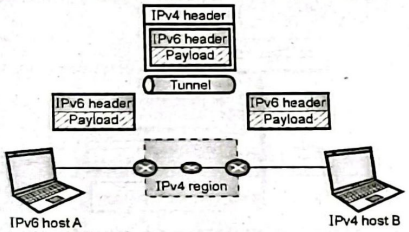
(G-2532) **Fig. 4.19(b) : Dual stack stategy**

- A source host send query to the DNS for deciding which version to use while sending a packet to a destination.

- A source host sends IPv4 packet if an IPv4 address is returned by the DNS, and sends IPv6 packet if DNS returns IPv6 address.

2. Tunneling :

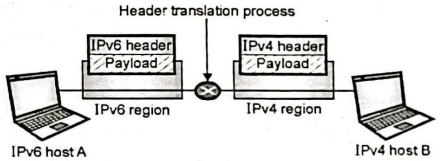
- When two computers are using IPv6 want to communicate with each other and a region through which the packet must pass uses IPv4, in such case **tunneling** strategy is used.
- The packet should have IPv4 address while passing through this region. When it enters in this region the IPv4 packet is encapsulated in IPv4 packet and when it exists the region it leaves its capsule.
- It looks like as if the IPv6 packet enters in a tunnel from one end and comes out from the other end. The protocol value is set to 41 for making it clear that IPv4 packet is holding an IPv6 packet as a data.
- The tunneling strategy is as shown in Fig. 4.19(c).



(G-2533) **Fig. 4.19(c) : Tunneling strategy**

3. Header translation :

- If some systems use IPv4 and the majority of the Internet has moved from IPv4 to IPv6, in that case header translation strategy is used where the receiver does not understand IPv6 but the sender wants to use IPv6 only.
- In this situation tunneling will not work because the packet should be in the IPv4 format which has to be understood by the receiver.
- In this strategy through header translation the format of header must be totally changed.
- The IPv6 packet header is converted into an IPv4 header. Fig. 4.19(d) shows header translation strategy.



(G-2534) **Fig. 4.19(d) : Header translation strategy**

Q. 23 Compare IPv4 and IPv6. **Dec. 15, May 19, Dec. 19**

Ans. :

Comparison between IPv4 and IPv6 :

- Table 4.5 shows the comparison of IPv4 and IPv6.

Table 4.5 : Comparison between IPv4 and IPv6

Sr. No.	IPv4	IPv6
1.	In IPv4 there are only 2^{32} possible ways to represent the address (about 4 billion possible addresses).	In IPv6 there are 2^{128} possible way (about 3.4×10^{38} possible addresses).
2.	The IPv4 address is written by dotted-decimal notation. e.g. 121.2.8.12	IPv6 is written in hexadecimal and consists of 8 groups, containing 4 hexadecimal digits or 8 groups of 16 bits each. e.g. FABC: AC77: 7834:2222:FACB: AB98: 5432:4567.
3.	The basic length of the IPv4 header comprises a minimum of 20 bytes (without option fields). The maximum total length of the IPv4 header is 60 bytes (with option fields) and it uses 13 fields to identify various control settings.	The IPv6 header is a fixed header of 40 bytes in length, and has only 8 fields. Option information is carried by the extension header, which is placed after the IPv6 header.

Sr. No.	IPv4	IPv6
4.	IPv4 header has a checksum, which must be computed by each router.	IPv6 has no header checksum because checksums are, for example, above the TCP/IP protocol suite, and above the Token Ring, Ethernet, etc.
5.	IPv4 contains an 8-bit field called Service Type. The Service Type field is composed of a TOS (Type of Service) field and a procedure field.	The IPv6 header contains an 8-bit field called the Traffic Class Field. This field allows the traffic source to identify the desired delivery priority of its packets.
6.	The IPv4 node has only stateful auto-configuration.	The IPv6 node has both a stateful and a stateless address autoconfiguration mechanism.
7.	Security in IPv4 networks is limited to tunneling between two networks.	IPv6 has been designed to satisfy the growing and expanded need for network security.
8.	Source and destination addresses are 32 bits (4 bytes) in length.	Source and destination addresses are 128 bits (16 bytes) in length.

Q. 24 In IPv4, class _____ has the greatest number of addresses in each block

(i) A (ii) B
(iii) C (iv) D

Ans. : (i)

Q. 25 The number of addresses in a class C block is _____.

(A) 65, 534 (B) 256
(C) 16,777,216 (D) none of these **Dec. 11**

Ans. : (B)

Q. 26 In IPv4, what is the value of the total length field in bytes if the header is 28 bytes and the data field is 400 bytes ?

(A) 428 (B) 407
(C) 107 (D) 427 **Dec. 11**

Ans. : (A)

Q. 27 In IPv4, when a datagram is encapsulated in a frame, the total size of the datagram must be less than the _____.

(A) MUT (B) MAT
(C) MTU (D) none of these **Dec. 11**

Ans. : (C)

Q. 28 _____ deals with the issues of creating and maintaining routing tables.

(A) Forwarding (B) Routing
(C) Directing (D) none of these **Dec. 11**

Ans. : (C)

Q. 29 Identify the class of the following IPv4 address : 191.1.2.3.

(A) C (B) A
(C) B (D) none of these **Dec. 11**

Ans. : (C)

Q. 30 An organization is granted a block ; one address is 2.2.2.64/20. The organization needs 10 subnets. What is the subnet prefix length ?

(A) /20 (B) /24
(C) /25 (D) none of these **Dec. 11**

Ans. : (B)

Q. 31 The _____ protocol is the transmission mechanism used by the TCP/IP suite.

(A) ARP (B) IP
(C) RARP (D) none of these **Dec. 11**

Ans. : (B)

Q. 32 The IPv4 header size _____.

(A) is 20 to 60 bytes long
(B) is 20 bytes long
(C) is 60 bytes long
(D) none of these **Dec. 11**

Ans. : (A)

Q. 33 The logical addresses in the Internet are called _____ addresses.

(A) port (B) IP
(C) Email (D) none of these **Dec. 11**

Ans. : (B)

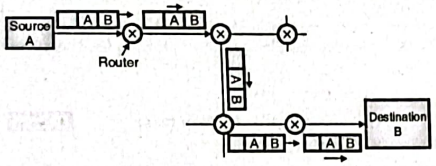
Chapter 5 : Network Layer-II

Q. 1 Explain unicast and multicast routing. May 13

Ans. :

Unicast routing :

- In unicast routing there is a one to one relation between the source and the destination.
- That means only one source sends packets to only one destination.
- The type of source and destination addresses included in the IP datagram are unicast addresses assigned to the hosts.
- The concept of unicast routing is illustrated in Fig. 5.1.



(G-448) **Fig. 5.1 : Unicast routing**

- In unicast routing when a router receives a packet, it forwards that packet through only one of its ports which corresponds to the optimum path.
- The router can discard the packet if it cannot find the destination address.

Multicast routing :

- In multicasting a message from a sender is to be sent to a group of destinations but not all the destinations in a network.
- A process has to send a message to all other processes in the group.
- For a small group it is possible to send a point-to-point message.
- But this is expensive if the group is large. So we have to send messages to a well defined groups which are small compared to the network size.
- Sending message to such a group is called **multicasting** and the routing algorithm used for multicasting is **multicast routing.**
- Multicast routing is a special class of broadcast routing.

Q. 2 Explain design goal often used in routing algorithm. May 17

Ans. :

Design goals for routing algorithms :

- Various routing algorithms are designed for one or more of the following design goals :
  1. Optimality.
  2. Simplicity and low overheads.
  3. Robustness and stability.
  4. Rapid convergence.
  5. Flexibility.

1. Optimality :

- We may define the optimality as the capability of a routing algorithm to select the best possible route, which depends on the metrics and metric weights used to make the calculations.

2. Simplicity :

- Routing algorithms are designed to be as simple as possible.
- That means the routing algorithm should work properly and efficiently with a minimum software and utilization overheads.

3. Robustness and stability :

- Routing algorithms should be designed for robustness.
- That means they should be able to perform correctly in all the unusual or unforeseen circumstances.
- The routing protocols are also supposed to withstand the test of time and prove stable under a variety of network conditions.

4. Rapid convergence :

- In addition, routing protocols should converge rapidly.
- Convergence can be defined as the process of agreement by all the routers, on optimal routes.
- That means in response to the routing update messages, the recalculation of optimal routes should be carried out quickly by all the routers.
- Routing algorithms that converge slowly can cause rooting loops or network outage.

5. Flexibility :

- Routing algorithms should also be flexible. That means they should adapt to different network circumstances quickly and accurately. That means in the event of failure of a network segment, different routers should quickly select the next best path for all routes which are using the failed segment. It is possible to program the routing algorithms to adapt to the changes in network bandwidth, router queue size and network delays.

Q. 3 What is dynamic routing ? Discuss distance vector routing. May 09, May 10, Dec. 15, Dec. 18

Ans. :

Dynamic routing algorithms :

- The algorithms in which the routing decision can be changed if there are any changes in topology or traffic etc. This is called as dynamic routing algorithms.

Distance Vector Routing Algorithm :

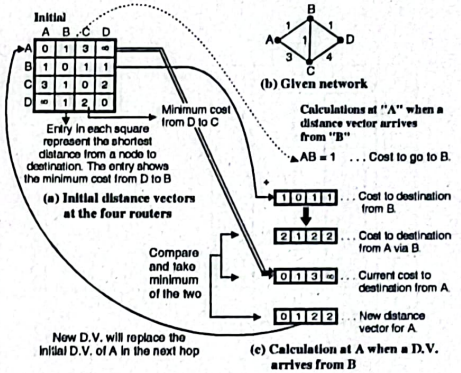
- In this algorithm, each router maintains a table called vector, such a table gives the best known distance to each destination and the information about which line to be used to reach there.
- This algorithm is sometimes called by other names such as :
  1. Distributed Bellman-Ford routing algorithm.
  2. Ford-Fulkerson algorithm.
- In distance vector routing, each router maintains a routing table.
- It contains one entry for each router in the subnet.
- This entry has two parts :
  1. The first part shows the preferred outgoing line to be used to reach the specific destination.
  2. Second part gives an estimate of the time or distance to that destination.

Distance vector :

- In distance vector routing, we assume that each router knows the identity of every other router in the network, but the shortest part to each router is not known.
- A distance vector is defined as the list of <destination, cost> tuples, one tuple per destination.
- Each router maintains a distance vector.
- The cost in each tuple is equal the sum of costs on the shortest path to the destination.

Updation of router tables :

- A router periodically sends a copy of its distance vector to all its neighbours.
- When a router receives a distance vector from its neighbour, it tries to find out whether its cost to reach any destination would decrease if it routed packets to that destination through that particular neighbouring router. This is illustrated in Fig. 5.2.

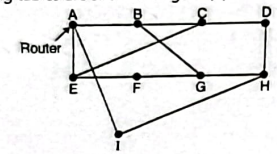


(G-463) **Fig. 5.2 : Distance vector algorithm at router A**

- Fig. 5.2 shows how the D.V. at A is automatically modified when a D.V. is received from B.
- A similar calculation takes place at the other routers as well.
- So the entries at every router can change. In Fig. 5.2(a) the initial distance vector is shown.
- The entries indicate to the costs corresponding to the shortest distance between the routers indicate to that square.
- For example, AC = 3 indicates the cost corresponding to the shortest path in terms of number of hops from A to C. Even if nodes asynchronously update their distance vectors the routing tables eventually converge.
- The well known example of distance vector routing is the Bellman-Ford algorithm.

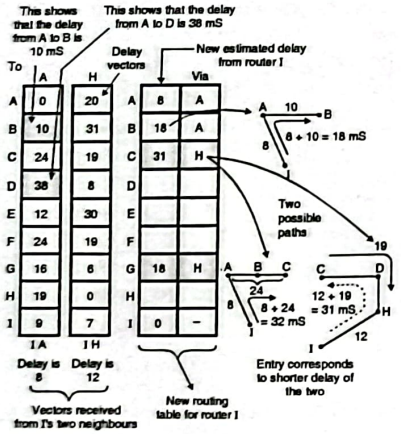
Routing procedure in distance vector routing :

- The example of a subnet is shown in Fig. 5.3(a) and the routing tables are shown in Fig. 5.3(b).



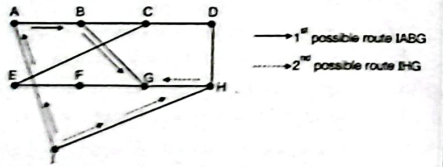
(G-464) **Fig. 5.3(a) : A subnet**

- The entries in router tables of Fig. 5.3(b) are the delay vectors.



(G-465) **Fig. 5.3(b) : Routing tables**

- For example consider the shaded boxes of Fig. 5.3(b).
- The entry in the first shaded box shows that the delay from A to B is 10 msec, whereas the entry in the other shaded box indicates that the delay from A to D is 38 msec.
- Consider how router I computes its new route to router G. Fig. 5.3(c) shows the two possible routes between I and G.



(G-466) **Fig. 5.3(c)**

- I knows that the reach G via A, the delay required is :

I to A Delay = 8mS
A to G Delay = 16mS
} ∴ I to G Delay = 8 + 16 = 24 msec

(L-891)

Whereas the delay between I and G via H (route IHG) is :

I to H Delay = 12mS
H to G Delay = 6mS
} ∴ I to G Delay = 12 + 6 = 18 msec

(L-892)

- The best of these values is 18 msec corresponding to the path IHG.
- Hence it makes an entry in its routing table (I's table) that the delay to G is 18 msec and that the route to use it is via H.
- The new routing table for router I is shown in Fig. 5.3(b).
- Similarly we can calculate the delays, from I to different destinations from A to I and enter the minimum possible delay into the I's router table.

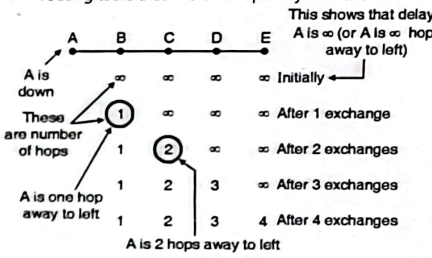
Q. 4 Explain the count-to-infinity entry in routing table. **Dec. 06, May 13**

Ans. :

Count to infinity problem :

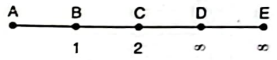
- Theoretically the distance vector routing works properly but practically it has a serious problem.
- The problem is that we get a correct answer but we get it slowly.
- In other words it reacts quickly to good news but it reacts too slowly to bad news. Consider a router whose best route to destination X is large.
- If on the next exchange neighbour A suddenly reports a short delay to X, the router will switch over and start using the line to A for sending the traffic to destination X.
- Thus in one vector exchange, the good news is processed. Let us see how fast does a good news propagate.
- Consider a linear subnet of Fig. 5.4 which has five nodes. The delay metric used is the number of hops.
- Assume that A is initially down and that all the other routers know this.

- So all the routers have recorded that the delay to A is infinity.
- When A becomes OK, the other routers come to know about it via the vector exchanges.
- Then suddenly a vector exchange at all the routers will take place simultaneously.
- At the time of first vector exchange, B comes to know that its left neighbour has a zero delay to A.
- So as shown in Fig. 5.4(a). B makes an entry in its routing table that A is one hop away to the left.



(G-467) **Fig. 5.4(a)**

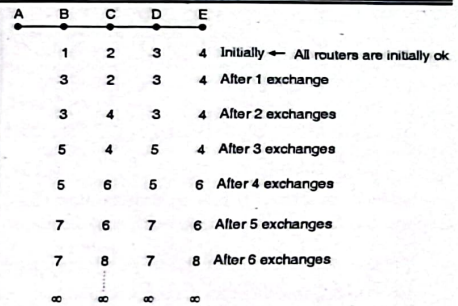
- All the other routers still think that A is down. So in the second row of Fig. 5.4(a), the entries below C D E are ∞.
- On the second vector exchange, C comes to know that B has a path of 1 hop length to A, so C updates its routing table and indicates a path of 2 hop length.
- But D and E do not change their table entries.
- So after the second vector exchange the entries in the third row of Fig. 5.4(a) are : (G-468(a))



- Similarly D and E will update their routing tables after 3 and 4 exchanges respectively.
- So we conclude that the good news of A has recovered has spread at a rate of one hop per exchange.

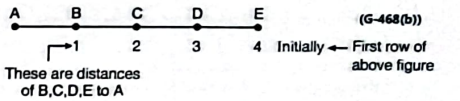
Explanation of Fig. 5.4(b) :

- Now refer Fig. 5.4(b). Here initially all routers are OK.



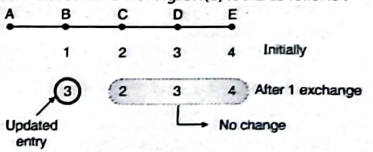
(G-468) **Fig. 5.4(b)**

- The routers B, C, D and E have distances of 1, 2, 3 and 4 respectively to A.
- So the first row of Fig. 5.4(b) is as follows :



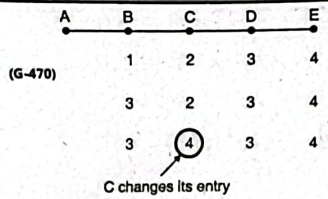
(G-468(b))

- Now imagine that suddenly A goes down or line between A and B is cut.
- At the first packet exchange B does not hear anything from A (because A is down). But C says " I have a path of length 2 to A". But poor B does not understand that this path is through B itself.
- So B thinks that it can reach A via C with a path length 3. (B to C 1 hop and C to A 2 hops) so it accordingly updates its routing table. But D and E do not update their entries.
- So the second row of Fig. 5.4(b) looks as follows :



(G-469)

- On the second exchange C realizes that both its neighbours (B and D) claim to have a path of length 3 to A.
- So it picks one of them at random and makes its new distance to A as 4.
- This is shown in row 3 of Fig. 5.4(b). It is repeated below.



(G-470)

- Similarly the other routers keep updating their tables after every exchange.
- It is expected that finally we should get ∞ in the router tables of B, C, D and E indicating that A is down.
- We do reach this state at the end in Fig. 5.4(b) but after a very long time. The conclusion is bad news propagates slowly. This problem is called as count-to-infinity problem.
- The solution to this problem is to use the split horizon algorithm.

Q. 5 Explain the link state routing in detail.

May 07, Dec. 07, Dec. 08, May 10, Dec. 10, Dec. 13

Ans. :

- Distance vector routing was used in ARPANET upto 1979. After that it was replaced by the link state routing.
- Variants of this algorithm are now widely used. The link state routing is simple and each router has to perform the following five operations.

Router operations :

1. Each router should discover its neighbours and obtain their network addresses.
2. Then it should measure the delay or cost to each of these neighbours.
3. It should construct a packet containing the network addresses and the delays of all the neighbours.
4. Send this packet to all other routers.
5. Compute the shortest path to every other router.

- The complete topology and all the delays are experimentally measured and this information is conveyed to each and every router.
- Then a shortest path algorithm such as Dijkshtra's algorithm can be used to find the shortest path to every other router.

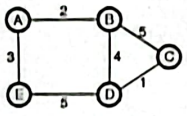
Protocols :

- Link state routing is popularly used in practice.
- The OSPF protocol which is used in the Internet uses the link state algorithm.
- IS-IS i.e. Intermediate system – Intermediate system is the other protocol which uses the link state algorithm.
- IS-IS is used in Internet backbones and in some digital cellular systems such as CDPD.

Building a routing table in link state routing :

Link state routing :

- Here the term link state is used for defining the characteristic of a link or edge, which represents a network in the Internet. The cost associated with each link is important.
- The links having lower costs are preferred to the links having higher costs.
- A nonexisting or broken link is indicated by an ∞ cost. In this method, each node must have a complete map of the network.
- That means each node should have complete information about the state of each link.
- The collection of states of all the links in an Internet is called as Link-State Database (LSDB).
- For the entire Internet, there is only one LSDB and its copy is available with each node.
- Each node uses it to create the least cost tree. The example of LSDB is as shown in Fig. 5.5(b) for the Internet shown in Fig. 5.5(a).



(a) Internetwork

	A	B	C	D	E
A	0	2	∞	∞	3
B	2	0	5	4	∞
C	∞	5	0	1	∞
D	∞	4	1	0	5
E	3	∞	∞	5	0

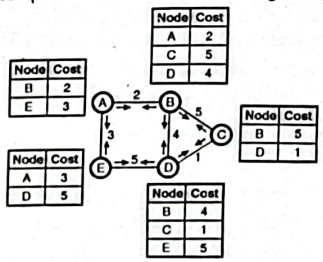
(b) Link state database (LSDB)

(G-2201) Fig. 5.5

- The next step is creation of LSDB (which contains all the information about the Internet) at each node.
- This can be achieved by a process called flooding.
- Each node sends a greeting message to all its immediate neighbours, so as to collect two important pieces of information as follows :

1. The identity of the neighbouring node.
2. Cost of the link.

- The packet containing this information is called as LS Packet (LSP), which is sent out of each interface.
- After receiving all the new LSPs each node will create the comprehensive LSDB as shown in Fig. 5.5(c).



(G-2202) Fig. 5.5(c)

- This LSDB is same for each node which shows the whole map of the internet. That means a node can use the LSDB to make the whole map of the Internet.

Q. 6 Compare distance vector routing with link state routing. May 14

Ans. :

Comparison of link state and distance vector routing :

Table 5.1 : Comparison of link state routing and distance vector routing

Sr. No.	Distance vector routing	Link state routing
1.	Each router maintains routing table indexed by and containing one entry for each router in the subnet.	It is the advanced version of distance vector routing.
2.	Algorithm took too long to converge.	Algorithm is faster.
3.	Bandwidth is less.	Wide bandwidth is available.
4.	Router measure delay directly with special ECHO packets.	All delays measured and distributed to every router.
5.	It doesn't take line bandwidth into account when choosing the routes.	It considers the line bandwidth into account when choosing the routes.

Q. 7 Compare static and dynamic routing algorithm with suitable example. May 14

Ans. :

Comparison between static and dynamic routing :

Table 5.2 : Comparison of static and dynamic routing

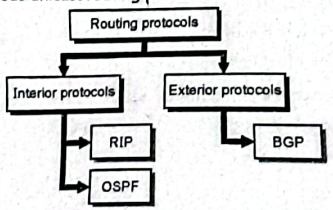
Sr. No.	Parameter	Static routing	Dynamic routing
1.	Updating of the routing tables	Manually done	Automatically done
2.	Bandwidth requirement	Less	More
3.	Application area	In small networks	In large networks
4.	Routing protocols	None	EIGRP, ARP etc.
5.	Security	Highly secure	Less secure
6.	Routing algorithms	Shortest path, flooding, flow based routing	Distance vector, link state
7.	Link failure	Any link failure affects the other routing paths.	Does not affect other routing paths.
8.	Additional resources	Not required	Required to store information
9.	Routing decision	Not based on the measured or estimated current traffic	Is based on the changes in topology or traffic

Q. 8 Give the classification of commonly used unicast routing protocols. Dec. 15

Ans. :

Classification of unicast routing protocols

- Various unicast routing protocols are shown in Fig. 5.6.



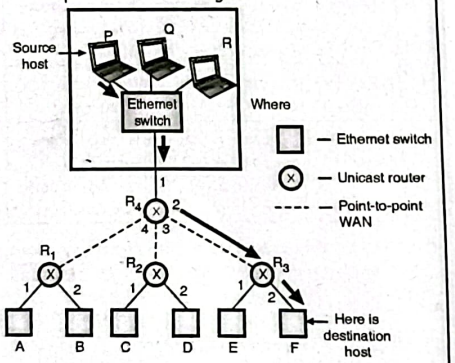
(G-497) Fig. 5.6 : Unicast routing protocols

Q. 9 Explain unicasting, broadcasting & multicasting. May 14

Ans. :

Unicasting :

- In the unicast communication, the communication takes place between one source and one destination.
- That means the relation between source and destination is one-to-one.
- In the IP datagram, both the source and destination addresses are the unicast addresses assigned to the hosts.
- In unicasting when a router receives a packet it forwards the packet through only one of its interfaces.
- This interface is the one corresponding to the optimum path. The router may discard the packet if the destination cannot be found.
- Fig. 5.7 shows the concept of unicasting. Delivery of unicast packet from a source computer to a destination computer is as shown in Fig. 5.7.

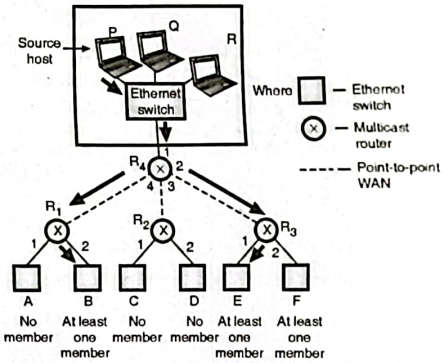


(G-2235) **Fig. 5.7 : Concept of unicasting**

- In Fig. 5.7, P is a source and destination computer is attached to Ethernet switch F.
- The responsibility of router R_4 is to forward the packet only through interface 2, the responsibility of router R_3 is to forward the packet only through interface 2.
- After arrival of packet at f, the responsibility of the network is to deliver the packet to destination host.
- The network broadcasts it to all hosts or Ethernet switch will deliver it to the destination host.

Multicasting :

- In multicast communication, the communication takes place between one source and a group of destinations i.e. the source to destination relationship is one-to-many.
- The type of source address is unicast address but the type of destination address is a group address that defines one or more destinations.
- The group address actually identifies all the members of a group.
- When a router receives a packet, it will forward the copies of packet to all the destinations through more than one of its interfaces, as shown in Fig. 5.8.



(G-2236) **Fig. 5.8 : Concept of multicasting**

- Router R_4 sends the datagram through interface 2 and 4. Router R_3 sends the datagram via both its interfaces 1 and 2.
- As R_3 knows that there is atleast one member is interested which belongs to this group in the area reached by interfaces 1 and 2.
- Router R_1, sends the datagram through interface 2. R_1 knows that there is no member interested in receiving datagram which belongs to the group in area reached by interface 1.

Broadcasting :

- In broadcast communication the source to destination relationship is of one-to-all type.

- That means there is only one source host and all other host act as destinations.
- On Internet the broadcasting does not take place due to the huge amount of traffic it would create and the corresponding bandwidth requirement.

Q. 10 What are multicasting applications ? May 10, Dec. 16

Ans. :

Applications of multicasting :

- Some of the applications of multicasting are as follows :
  1. In order to gain access to distributed databases.
  2. For information dissemination
  3. Dissemination of news.
  4. So as to have teleconferencing
  5. To help distance learning.

Q. 11 What is IGMP ? Give its message format. Dec. 10, May 13, May 14, May 19, Dec. 19

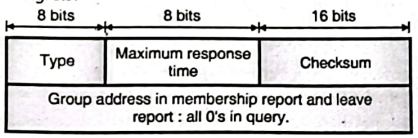
Ans. :

IGMP :

- IGMP is a necessary but not sufficient protocols used in multicasting environment. It is always used along with IP.

Message format :

- The IGMP (version-2) message format is shown in Fig. 5.9.



(G-587) **Fig. 5.9 : IGMP message format**

1. Type :

- It is an 8 bit field that defines the type of message as given in Table 5.3. The type and its value in hexadecimal and binary notation have also been shown in Table 5.3.

Table 5.3 : IGMP type field

Type	Value
General or special query	0x11 or 0001 0001
Membership report	0x16 or 0001 0110
Leave report	0x17 or 0001 0111

2. Maximum response time :

- This is the next 8-bit field which defines the amount of time allowed to answer a query.
- The value in this field shows the maximum response time in tenths of seconds.
- This value is a non zero number if the message is a query message and it is equal to zero for the other two message types.

3. IGMP checksum :

- The checksum is the 16-bit one's complement of the one's complement sum of the 8-byte IGMP message.
- When the checksum is computed, the checksum field should first be cleared to 0.
- When the data packet is transmitted, the checksum is computed and inserted into this field.
- When the data packet is received, the checksum is again computed and verified against the checksum field.
- If the two checksums do not match then an error has occurred.

4. Group address :

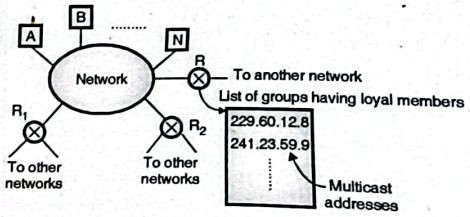
- This is a 32-bit field and its value depends on the type of message. For example the value of this field is zero for a general query message.
- The value in this field defines the multicast address of the group called **groupid**, in the other three types of messages.

Q. 12 How does IGMP is used ? Dec. 06, May 19, Dec. 19

Ans. :

IGMP operation :

- Refer Fig. 5.10 to understand the IGMP operation.



(G-588) **Fig. 5.10 : IGMP operation**

- IGMP operates locally. As shown in Fig. 5.10 multicast router R has a list of multicast addresses of the groups for which the router distributes packets.
- These packets are distributed to groups with at least one loyal member in that network.
- There is one router per group. Its duty is to distribute the multicast packets which are supposed to reach that group.
- So if there are three multicast routers (R_1, R_2 and R) connected to the network, then the lists of group identifications (ids) of all the routers are mutually exclusive i.e. they do not contain the same addresses.

Chapter 6 : Transport Layer

Q. 1 Explain the functions of transport layer.

May 12, Dec. 12, May 13, Dec. 13, Dec. 14, May 16, May 17, Dec. 18

Ans. :

Functions of transport layer :

1. Packetizing :

- The transport layer creates packets with the help of encapsulation on the messages received from the application layer. Packetizing is a process of dividing a long message into smaller ones.
- These packets are then encapsulated into the data field of the transport layer packet.
- The headers containing source and destination address are then added.
- The length of the message which is to be divided can vary from several lines (e-mail) to several pages.
- But the size of the message can become a problem.
- The message size can be larger than the maximum size that can be handled by the lower layer protocols.
- Hence the messages must be divided into smaller sections. Each small section is then encapsulated into a separate packet.
- Then a header is added to each packet to allow the transport layer to perform its other functions.

2. Connection control :

- Transport layer protocols are divided into two categories :
  1. Connection oriented.
  2. Connectionless.

Connection oriented delivery :

- A connection oriented transport layer protocol establishes a connection i.e. virtual path between sender and receiver.
- This is a virtual connection. The packet may travel out of order.
- The packets are numbered consecutively and communication is bi directional.

Connectionless delivery :

- A connectionless transport protocol will treat each packet independently.
- There is no connection between them. Each packet can take its own different route.

3. Addressing :

- The client needs the address of the remote computer it wants to communicate with.
- Such a remote computer has a unique address so that it can be distinguished from all the other computers.

4. Flow and error control :

- For high reliability the flow control and error control should be incorporated.
- **Flow control :** We know that data link layer can provide the flow control. Similarly transport layer also can provide flow control.
- But this flow control is performed end to end and not across a single link.
- **Error control :** The transport layer can provide error control as well.
- But error control at transport layer is performed end to end and not across a single link.
- Error correction is generally achieved by retransmission of the packets discarded due to errors.

Q. 2 Explain different services provided by transport layer.

May 12, Dec. 13, May 16, Dec. 16, Dec. 17, Dec. 19

Ans. :

Services provided by transport layer :

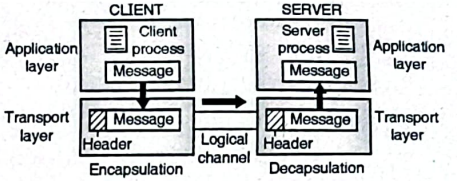
- The data link layer performs a node to node delivery.
- The network layer carries out the datagram delivery between two hosts (host to host delivery).
- But the real communication takes place between two processes or application programs for which we need the **process-to-process delivery.**
- The transport layer takes care of the **process-to-process delivery.**
- In this a packet from one process is delivered to the other process.
- The relationship between the communicating processes is the client-server relationship.

Addressing : Port number :

- There are several ways of achieving the process-to-process communication, but the most common method is using the client-server paradigm.
- **Client** is defined as the process on the local host.
- It needs services from another process called **server** which is on the other (remote) host.
- Both client and server have the same name. Some of the important terms related to the client-server paradigm are :
  1. Local host
  2. Remote host
  3. Local process
  4. Remote process
- We can use the IP addresses to define the local host and remote host.
- But this is not enough to define a process.
- In order to define a process, we have to use one more identifier called **Port numbers.**

Encapsulation and decapsulation :

- The transport layer carries out the **Encapsulation** of the message at the sending end and then **Decapsulation** at the receiving end when two computers communicate.
- This process has been illustrated in Fig. 6.1.



(G-2012) **Fig. 6.1 : Encapsulation and decapsulation**

Encapsulation :

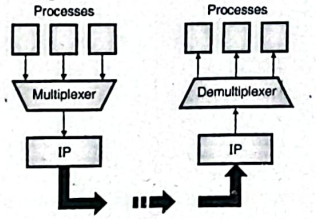
- At the sending end the process that has a message to send, will pass it to the transport layer alongwith a pair of socket addresses and some additional information.
- The transport layer adds its own header to this data. This packet at the transport layer in the Internet is known by different names such as **user datagram, segment or packet.**

Decapsulation :

- When the segment or datagram arrives at the receiving end, the header is isolated and destroyed, and the message is delivered to the process running at the application layer as shown in Fig. 6.1.
- The socket address of the sender process is then handed over to the destination process.

Multiplexing and Demultiplexing :

- The addressing mechanism allows multiplexing and demultiplexing taking place at the transport layer as shown in Fig. 6.2.



(G-597) **Fig. 6.2 : Multiplexing and demultiplexing**

Multiplexing :

- At the sending end, there are several processes that are interested in sending packets. But there is only one transport layer protocol (UDP or TCP).
- Thus it is a many processes-one transport layer protocol situation.

- Such a many-to-one relationship requires multiplexing.
- The protocol first accepts messages from different processes.
- These messages are separated from each other by their port numbers. Each process has a unique port number assigned to it.
- Then the transport layer adds header and passes the packet to the network layer as shown in Fig. 6.2.

Demultiplexing :

- At the receiving end, the relationship is one as to many. So we need a demultiplexer.
- First the transport layer receives datagrams from the network layer.
- The transport layer then checks for errors and drops the header to obtain the messages and delivers them to appropriate process based on the port number.

Q. 3 What are differences in IP address and port number ? May 19

Ans. :

Difference between IP address and port number :

Table 6.1 : Difference between IP address and port number

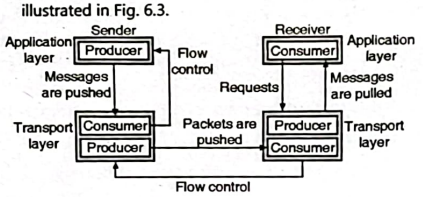
Sr. No.	IP address	Port number
1.	To identify a host IP address is used.	To identify an application/services on your system port number is used.
2.	IP address is the address of layer-3 (Network layer) protocol.	Port number is used for layer-4 (Transport layer) protocols.
3.	IP address is used by admin user of system.	Port number is provided by OS. This port number is called as port address.
4.	IP address is used to send datagram traffic access network from source to destination machine.	After packet delivery to destination, with the help of the port numbers OS sends the data to correct application.
5.	E.g. 191.168.0.1, 172.14.0.1 etc.	e.g. Port number 80, 68 etc.

Q. 4 Explain flow control in transport layer protocols. May 19

Ans. :

Flow Control at Transport Layer :

- The concept of flow control at transport layer has been illustrated in Fig. 6.3.



(G-2014) **Fig. 6.3 : Flow control at transport layer**

- It shows the communication taking place between a sender and a receiver.
- As shown in Fig. 6.3, there are four entities involved in this communication. They are as follows :
  1. Sender process.
  2. Sender transport layer.
  3. Receiver process.
  4. Receiver transport layer.

Sending end :

- The first entity on the sending end is the **sender process,** at the application layer. It works only as a **producer** which produces chunk of messages and pushes them to the transport layer on the sending end, as shown in Fig. 6.3.
- The second entity on the sending end is the **sender transport layer.** It has two different roles to play.
- First it acts as a **customer** and consumes all the messages produced and pushed by the producer.
- Then it encapsulates those messages into packets and pushes them to the receiver transport layer as shown in Fig. 6.3. Here it acts as a **producer.**

Receiving end :

- The first entity on the receiving end is the **receiver transport layer.** It also has two different roles to play.
- It acts as a **consumer** for the packets pushed by the senders transport layer and it also acts as the **producer.**

- It has decapsulate the messages and deliver them to the application layer as shown in Fig. 6.3.
- However the delivery of decapsulated messages to the application layer is a **pulling type delivery.**
- That means the transport layer waits till the application layer process requests for the decapsulated messages.

Flow control :

- As shown in Fig. 6.3, the flow control is needed for atleast two cases.
- First is from transport layer of sender to the application layer of sender.
- And secondly form the transport layer of receiver to the transport layer of sender.

Buffers :

- It is possible to implement the flow control in many different ways.
- One of the ways of implementation is to use two **buffers** one each at the sending and receiving transport layers.
- A **buffer** is nothing but a set of memory locations which can temporarily hold (store) packets.
- It is possible to exercise flow control communication by sending signals from the consumer to producer.
- The **flow control** at the **sending end** takes place as follows :
- As soon as the buffer at the transport layer becomes full it sends the stop message to its application layer in order to stop the chunk of messages that are being pushed into the buffer.
- The second flow control takes place at the receiver transport layer as follows :
- As soon as the buffer at receiver transport layer becomes full, it will inform the sender transport layer to stop pushing the packets.
- Whenever the buffer becomes partially empty, it again informs the sender transport layer to start sending the packets again.

Q. 5 Compare connection oriented versus connection less service. Dec. 09

Ans. :

Comparison of CLTS & COTS :

Table 6.2 : Comparison of connection oriented versus connection less service

Sr. No.	Parameter	Connection oriented	Connectionless
1.	Reservation of resources	Necessary	Not necessary
2.	Utilization of resources	Less	Good
3.	State information	Lot of information required	Not much information is required to be stored
4.	Guarantee of service	Guaranteed	No guarantee
5.	Connection	Connection needs to be established	Connection need not be established
6.	Delays	More	Less
7.	Overheads	Less	More
8.	Packets travel	Sequentially	Randomly
9.	Congestion due to overloading	Not possible	Very much possible

Q. 6 List the typical QoS parameters in the Transport Layer and explain each one. May 12, Dec. 12, Dec. 13, May 14, Dec. 14, May 16

Ans. :

1. Connection establishment delay :

- The time difference between the instant at which a request for transport connection is made and the instant at which it is confirmed is called as connection establishment delay.
- This delay should be as short as possible to ensure better service.

2. Connection establishment failure probability :

- Sometimes the connection may not get established even after the maximum connection establishment delay.

- This can be due to network congestion, lack of table space or some other problems.

3. Throughput :

- It is defined as the number of bytes of user data transferred per second, measured over some time interval.
- Throughput is measured separately for each direction.

4. Transit delay :

- It is the time duration between a message being sent by the transport user from the source machine and its being received by the transport user at the destination machine.

5. Residual error ratio :

- It measures the number of lost or garbled messages as a percentage of the total messages sent.
- Ideally the value of this ratio should be zero and practically it should be as small as possible.

6. Protection :

- This parameter provides a way to protect the transmitted data against reading or modifying it by some unauthorised parties.

7. Priority :

- Using this parameter the user can show that some of its connections are more important (have higher priority) than the other ones.
- This is important when congestions take place. Because the higher priority connections should get service before the low priority connections.

8. Resilience :

- Due to internal problem or congestion the transport layer spontaneously terminates a connection.
- The resilience parameter gives the probability of such a termination.

Q. 7 Explain various transport layer protocols.

Dec. 16, Dec. 17, Dec. 19

Ans. :

- The User Datagram Protocol is a very simple protocol. It adds little to the basic functionality of IP. Like IP, it is an unreliable, connectionless protocol.
- You do not need to establish a connection with a host before exchanging data with it using UDP, and there is no mechanism for ensuring that data sent is received.
- A unit of data sent using UDP is called a Datagram. UDP adds four 16-bit header fields (8 bytes) to whatever data is sent.
- These fields are : a length field, a checksum field, and source and destination port numbers. "Port number", in this context, represents a software port, not a hardware port.
- The concept of port numbers is common to both UDP and TCP. The port numbers identify which protocol module sent (or is to receive) the data.
- Most protocols have standard ports that are generally used for this. For example, the Telnet protocol generally uses port 23. The Simple Mail Transfer. Protocol (SMTP) uses port 25.
- The use of standard port numbers makes it possible for clients to communicate with a server without first having to establish which port to use.
- The port number and the protocol field in the IP header duplicate each other to some extent, though the protocol field is not available to the higher-level protocols. IP uses the protocol field to determine whether data should be passed to the UDP or TCP module.
- UDP or TCP use the port number to determine which application-layer protocol should receive the data.
- Although UDP isn't reliable, it is still a preferred choice for many applications.
- It is used in real-time applications like Net audio and video where, if data is lost, it's better to do without it than send it again out of sequence.
- It is also used by protocols like the Simple Network Management Protocol (SNMP).

Responsibilities of UDP :

- Being a transport layer protocol, the UDP has the following responsibilities :

1. To create a process to process communication, UDP uses port numbers to accomplish this.

2. To provide control mechanisms at the transport layer, UDP does not provide flow control or acknowledgements. It provides error detection. The erroneous packet is discarded.
3. UDP does not add anything to the services of IP except for providing process to process communication.

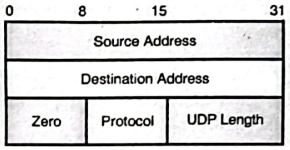
Q. 8 Draw and explain the UDP header in detail.

Dec. 13, May 14, Dec. 15, May 16, May 19, Dec. 19

Ans. :

UDP header :

- The purpose of using a pseudo-header is to verify that the UDP packet has reached its correct destination.
- The correct destination consists of a specific machine and a specific protocol port number within that machine.



(G-625) **Fig. 6.4 : UDP pseudo header**

- The UDP header itself specifies only the protocol port number.
- Thus, to verify the destination, UDP on the sending machine computes a checksum that covers the destination IP address as well as the UDP packet.
- At the ultimate destination, UDP software verifies the checksum using the destination IP address obtained from the header of the IP packet that carried the UDP message.
- If the checksum agrees, then it must be true that the packet has reached the intended destination host as well as the correct protocol port within that host.

User interface :

- A user interface should allow the creation of new receive ports, receive operations on the receive ports that return the data octets and an indication of source port and source address, and an operation that allows a datagram to be sent, specifying the data, source and destination ports and addresses to be sent.

IP Interface :

- The UDP module must be able to determine the source and destination Internet addresses and the protocol field from the Internet header.
- One possible UDP/IP interface would return the whole Internet datagram including the entire Internet header in response to a receive operation.
- Such an interface would also allow the UDP to pass a full Internet datagram complete with header to the IP to send.
- The IP would verify certain fields for consistency and compute the Internet header checksum.

Protocol application :

- The major uses of this protocol are the Internet Name Server, and the Trivial File Transfer.

Protocol number :

- This is protocol 17 (21 octal) when used in the Internet Protocol.

Q. 9 Explain various transport layer protocols.

Dec. 16

Ans. :

Transmission Control Protocol (TCP) :

- The TCP provides reliable transmission of data in an IP environment. TCP corresponds to the transport layer (Layer 4) of the OSI reference model.
- Among the services TCP provides are stream data transfer, reliability, efficient flow control, full-duplex operation, and multiplexing.
- TCP is the layer 4 protocol in the TCP/IP suite and it is a very important and complicated protocol.
- TCP has been revised multiple times in last few decades.
- With stream data transfer, TCP delivers an unstructured stream of bytes identified by sequence numbers.
- This service benefits applications because they do not have to chop data into blocks before handing it off to TCP.
- Instead, TCP groups bytes into segments and passes them to IP for delivery.

- TCP offers reliability by providing connection-oriented, end-to-end reliable packet delivery through an internetwork.
- It does this by sequencing bytes with a forwarding acknowledgment number that indicates to the destination the next byte the source expects to receive.
- Bytes not acknowledged within a specified time period are retransmitted.
- The reliability mechanism of TCP allows devices to deal with lost, delayed, duplicate, or misread packets.
- A time-out mechanism allows devices to detect lost packets and request retransmission.
- TCP offers efficient flow control, which means that, when sending acknowledgments back to the source, the receiving TCP process indicates the highest sequence number that it can receive without overflowing its internal buffers.
- TCP supports a full-duplex operation means that TCP processes can both send and receive at the same time.
- Finally, TCP's multiplexing means that numerous simultaneous upper-layer conversations can be multiplexed over a single connection.

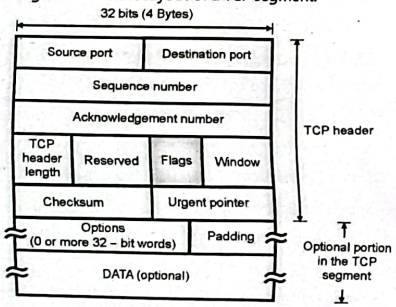
Q. 10 Draw and explain TCP header.

May 12, Dec. 12, May 14, Dec.15, Dec. 17, Dec. 18, May 19, Dec. 19

Ans. :

TCP header :

- *Fig. 6.5* shows the layout of a TCP segment.



(G-611) Fig. 6.5 : TCP header format

- Every segment begins with a 20 byte fixed format header.
- The fixed header may be followed by header options.
- After the options, if any, upto 65535 – 20 – 20 = 65495 data bytes may follow.
- Note that the first 20 bytes correspond to the IP header and the next 20 correspond to the TCP header.
- The TCP segment without data are used for sending the acknowledgements and control messages.

Source port :

- A 16-bit number identifying the application the TCP segment originated from within the sending host.
- The port numbers are divided into three ranges, well-known ports (0 through 1023), registered ports (1024 through 49,151) and private ports (49,152 through 65,535).
- Port assignments are used by TCP as an interface to the application layer.

Destination port :

- A 16-bit number identifying the application the TCP segment is destined for on a receiving host.
- Destination ports use the same port number assignments as those set aside for source ports.

Sequence number :

- A 32-bit number identifying the current position of the first data byte in the segment within the entire byte stream for the TCP connection.
- After reaching $2^{32} - 1$, this number will wrap around to 0.

Acknowledgement number :

- A 32-bit number identifying the next data byte the sender expects from the receiver.
- Therefore, the number will be one greater than the most recently received data byte.
- This field is only used when the ACK control bit is turned on.

Header length or offset :

- A 4-bit field that specifies the total TCP header length in 32-bit words (or in multiples of 4 bytes if you prefer). Without options, a TCP header is always 20 bytes in length. The largest a TCP header may be is 60 bytes.

- This field is required because the size of the options field(s) cannot be determined in advance.
- Note that this field is called "data offset" in the official TCP standard, but header length is more commonly used.

Reserved :

- A 6-bit field currently unused and reserved for future use.

Control bits or flags :

1. **Urgent pointer (URG) :** If this bit field is set, the receiving TCP should interpret the urgent pointer field.
2. **Acknowledgement (ACK) :** If this bit field is set, the acknowledgement field is valid.
3. **Push function (PSH) :** If this bit field is set, the receiver should deliver this segment to the receiving application as soon as possible.
   - An example of its use may be to send a Control-BREAK request to an application, which can jump ahead of queued data.
4. **Reset the connection (RST) :** If this bit is present, it signals the receiver that the sender is aborting the connection and all queued data and allocated buffers for the connection can be freely relinquished.
5. **Synchronize (SYN) :** When present, this bit field signifies that sender is attempting to "synchronize" sequence numbers.
   - This bit is used during the initial stages of connection establishment between a sender and receiver.
6. **No more data from sender (FIN) :** If set, this bit field tells the receiver that the sender has reached the end of its byte stream for the current TCP connection.

Window :

- A 16-bit integer used by TCP for flow control in the form of a data transmission window size.
- This number tells the sender how much data the receiver is willing to accept.
- The maximum value for this field would limit the window size to 65,535 bytes, however a "window scale" option can be used to make use of even larger windows.

Checksum :

- A TCP sender computes a value based on the contents of the TCP header and data fields.
- This 16-bit value will be compared with the value the receiver generates using the same computation.
- If the values match, the receiver can be very confident that the segment arrived intact.

Urgent pointer :

- In certain circumstances, it may be necessary for a TCP sender to notify the receiver of urgent data that should be processed by the receiving application as soon as possible.
- This 16-bit field tells the receiver when the last byte of urgent data in the segment ends.

Options :

- In order to provide additional functionality, several optional parameters may be used between a TCP sender and receiver.
- Depending on the option(s) used, the length of this field will vary in size, but it cannot be larger than 40 bytes due to the size of the header length field (4 bits).
- The most common option is the Maximum Segment Size (MSS) option.
- A TCP receiver tells the TCP sender the maximum segment size it is willing to accept through the use of this option.
- Other options are often used for various flow control and congestion control techniques.

Padding :

- Because options may vary in size, it may be necessary to "pad" the TCP header with zeros so that the segment ends on a 32-bit word boundary as defined by the standard.

Data :

- Although not used in some circumstances (e.g. acknowledgement segments with no data in the reverse direction), this variable length field carries the application data from TCP sender to receiver.
- This field coupled with the TCP header fields constitutes a TCP segment.

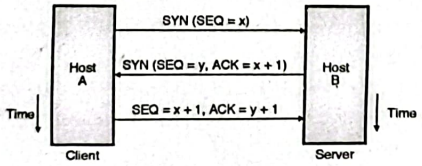
Q. 11 Explain connection establishment with respect to the transport layer.

Dec. 12, Dec. 13, Dec. 15, Dec. 16, May 17, Dec. 17, Dec. 18

Ans. :

TCP connection establishment :

- Connection establishment is performed by using a **three-way handshake** mechanism.
- A three-way handshake synchronizes both ends of a connection by allowing both sides to agree upon initial sequence numbers.
- This mechanism also guarantees that both sides are ready to transmit data and know that the other side is ready to transmit as well.
- This is necessary so that packets are not transmitted or re-transmitted during session establishment or after session termination.
- Each host randomly chooses a sequence number used to track bytes within the stream it is sending and receiving.
- Then, the three-way handshake proceeds in the manner shown in Fig. 6.6.



(G-613) **Fig. 6.6 : TCP connection establishment (Three-way handshake)**

- The requesting end (HOST A) sends a SYN segment specifying the port number of the server that the client wants to get connected to, and the client's initial sequence number (x).
- The server (HOST B) responds with its own SYN segment containing the server's initial sequence number (y).
- The server also acknowledges the client's SYN by acknowledging the client's SYN plus one (x + 1). A SYN consumes one sequence number.
- The client must acknowledge this SYN from the server by acknowledging the server's SYN plus one. (SEQ. = x + 1, ACK = y + 1).
- This is how a TCP connection is established.

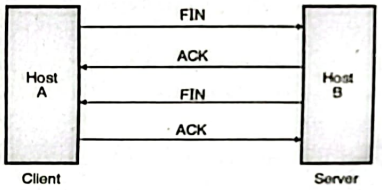
Q. 12 Explain connection termination with respect to the transport layer.

Dec. 12, Dec. 13, Dec. 15, Dec. 16, May 17

Ans. :

TCP connection termination :

- While it takes three segments to establish a connection, it takes four to terminate a connection.
- Since a TCP connection is full-duplex (that is, data flows in each direction independently of the other direction), the connection should be terminated in both the directions independently.
- The termination procedure in each direction is shown in Fig. 6.7(b).



(G-614) **Fig. 6.7(b) : TCP termination**

- The rule is that either side can send a FIN when it has finished sending data (FIN indicates finished).
- When a TCP program on a host receives a FIN, it informs the application that the other end has terminated the data flow.
- The receipt of a FIN only means there will be no more data flowing in that direction. A TCP can still send data after receiving a FIN.
- The end that first issues the close (e.g., sends the first FIN) performs the active close and the other end (that receives this FIN) performs the passive close.
- Now refer Fig. 6.7(b). When the server receives the FIN it sends back an ACK of the received sequence number plus one.
- A FIN consumes a sequence number, just like a SYN.
- At this point the server's TCP also delivers an end-of-file to the application (the discard server).

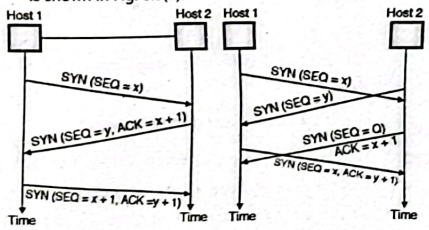
- The server then closes its connection and its TCP sends a FIN to the client. The client's TCP informs the application and sends an ACK to server by incrementing the received sequence number by one.
- Connections are normally initiated by the client, with the first SYN going from the client to the server. A client or server can actively close the connection (i.e. send the first FIN).
- But in practice generally the client determines when the connection should be terminated, since client processes are often driven by an interactive user, who enters something like quit to terminate.
- This is how the TCP connection is released.

Q. 13 Explain the TCP connection management in Client / Server model. May 16, May 17

Ans. :

TCP Connection management :

- Connections are established in TCP by following the three-way handshake technique.
- To establish a connection, one side, say the server, passively waits. It executes the LISTEN and ACCEPT primitives, to specify either a particular other side or nobody in particular.
- The other side (client) executes a connect primitive, with the IP and the port specified.
- The other information is the maximum TCP segment size, possible other options and optionally some user data (e.g. a password).
- The CONNECT primitive sends a TCP segment with the SYN bit on and the ACK bit off and waits for a response.
- The sequence of TCP segments sent in the normal case is shown in Fig. 6.8(a).



(a) Normal operation **(b) Call collision**

(G-615) **Fig. 6.8 : TCP connection management**

- When the segment sent by host – 1 reaches the destination i.e. host – 2 the receiving server checks to see if there is a process that has done a LISTEN on the port given in the destination port field.
- If not, it sends a reply with the RST bit on to reject the connection.
- Otherwise it gives the TCP segment to the listening process, which can accept or refuse (e.g. if it does not like the client) the connection.
- On acceptance a SYN is send, otherwise a RST.
- Note that a SYN segment occupies 1 byte of sequence space so it can be acknowledged unambiguously.

Call collision :

- If two hosts try to establish a connection simultaneously between the same two sockets then the events take place as shown in Fig. 6.8(b).
- Under such circumstances only one connection is established. Both the connections cannot be established simultaneously because connections are identified by their end points.
- If the first set up results in a connection which is identified by (x, y) and second connection is also set up, then only one table entry will be made i.e. for (x, y).
- For the initial sequence number a clock based scheme is used, with a clock pulse coming after every 4 μsec.
- For ensuring an additional safety, when a host crashes, it may not reboot for 120 sec which is maximum packet lifetime.
- This is to make sure that no packets from previous connections are still alive and travelling around.

Q. 14 Explain how QoS is improved. May 12, Dec. 12

Ans. :

Techniques for achieving good QoS :

1. Buffering.
2. Traffic shaping.
3. Leaky bucket algorithm.
4. Token bucket algorithm.
5. Resource reservation.
6. Admission control.

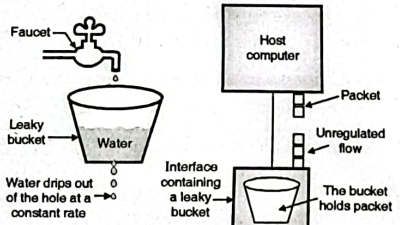
7. Proportional routing.
8. Packet scheduling.

Q. 15 Explain leaky bucket algorithm. Dec. 06, May 08

Ans. :

Leaky bucket algorithm :

- Leaky bucket algorithm is used to control congestion in network traffic. As the name suggests it's working is similar to a leaky bucket in real life.
- The principle of leaky bucket algorithm is as follows :
- Leaky bucket is a bucket with a hole at bottom.
- Flow of the water from bucket is at a constant rate (data rate is constant) which is independent of water entering the bucket (incoming data).
- If bucket is full, any additional water entering in the bucket is thrown out (Packets are discarded).
- Same technique is applied to control congestion in network traffic. Every host in the network is having a buffer (equivalent to a bucket) with finite queue length.
- Packets which are put in the buffer when buffer is full are thrown away.
- The buffer may send some number of packets per unit time onto the subnet (helpful if packets vary greatly in size) as shown in Fig. 6.9.
- the data flow at the input of the bucket is unregulated but that at the bucket output is a regulated one.

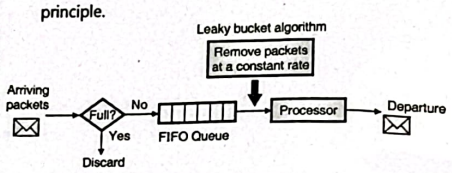


(a) (b)

(G-481) **Fig. 6.9 : Leaky bucket algorithm**

Leaky bucket implementation :

- Fig. 6.10 shows the implementation of leaky bucket principle.



(G-482) **Fig. 6.10 : Implementation of leaky bucket**

- A FIFO (First In First Out) queue is used for holding the packets which is equivalent to the leaky bucket.
- The implementation of Fig. 6.10 can be discussed under two different operating conditions, namely :
  1. For packets of fixed size.
  2. For packets of variable size.

1. Fixed size packets :

- If the arriving packets are of fixed size (e.g. cells in ATM networks), then the process of Fig. 6.10 will allow the removal of a fixed number of packets from the queue corresponding to every tick of the clock.

2. Packets of variable size :

- If the packets at the input of the process are of different size, then the fixed output rate will not correspond to the number of packets leaving the process but it will correspond to the number of bits leaving the process.

Algorithm :

- The algorithm for variable length packets is as follows :
  1. Initialize a counter to a number "n" at the tick of the clock.
  2. If "n" is greater than the packet size, then send the packet and decrement the counter by the packet size.
  3. Repeat step 2 until "n" becomes smaller than the packet size.
  4. Reset the counter and go back to step 1.

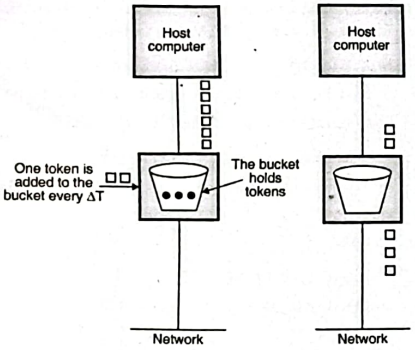
Note : Thus a leaky bucket algorithm shapes the bursty traffic to convert it into a fixed rate traffic. It does so by averaging the data rate. It drops the packets if the bucket (buffer) is full.

Q. 16 Explain token bucket algorithm. May 07, May 08, Dec. 08, Dec. 12

Ans. :

Token bucket algorithm :

- This algorithm is similar to the leaky bucket but it is possible to vary output rates.
- This is useful when larger burst of traffic is received.
- It enforces a long-term average transmission rate while permitting bounded bursts.
- In this approach, a token bucket is used to which manages the queue regulator that ultimately controls the rate of packet flow into the network.
- A token generator continuously produces tokens at a rate of R tokens per second and puts them into a token bucket with a depth of D tokens as shown in Fig. 6.10.



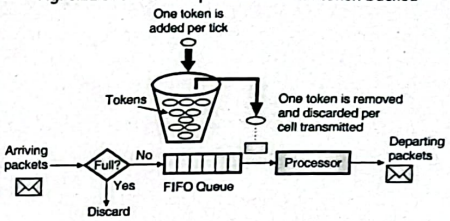
(a) Before (b) After

(G-483) **Fig. 6.10 : Token bucket algorithm**

- If the token bucket gets full then the extra tokens are discarded.
- Token bucket algorithm is a variant of leaky bucket algorithm. Here the bucket is filled with tokens.
- A packet which grabs and destroys a token is allowed to leave the bucket.
- Due to this mechanism, the packets never get lost but they just have to wait to grab a token.
- At the same time, an unregulated stream of packets arrive and are placed into a packet queue that has a maximum length of L.
- If the flow delivers more packets than the queue can store, the excess packets are discarded.

Implementation of token bucket :

- Fig. 6.11 shows the implementation of token bucket.



(G-484) **Fig. 6.11 : Implementation of token bucket**

- The token bucket can be easily implemented with a counter.
- The token is initialized to zero.
- Every time a token is added, the counter is incremented by 1 and every time a packet is dispatched, the counter is decremented by 1.
- If the counter contents go to zero, the host cannot send any data.

Note : The token bucket allows the bursty traffic at maximum possible rate.

Token bucket performance :

Let, s = Burst length (seconds),

c = Bucket capacity (bytes),

ρ = Token arrival rate (bytes/second),

and m = Maximum source rate (bytes/second)

What is the duration of a maximum-rate burst through a token bucket ?

1. Maximum bytes sent from the token bucket during a burst is, $c + \rho \cdot s$
2. Maximum bytes the source can send during a burst is,

   $m \cdot s$
3. Setting the two equal and solving for s,

$$s = \frac{c}{m - \rho}$$

Q. 17 A computer on a 6 Mbps network is regulated by a token bucket. The token bucket is filled at a rate 1 Mbps. It is initially filled to a capacity of 8 Mbit. How long can the computer transmit a full rate of 6 Mbps ? Dec. 06

Ans. :

Given :

C = Bucket capacity = 8 M bits

m = Maximum output rate = 6 Mbps.

ρ = Token arrival rate = 1 Mbps

To find :

S = Times for which maximum output is obtained.

$$S = \frac{C}{m - \rho}$$

$$\therefore S = \frac{8 \text{ M bits}}{6 \text{ Mbps} - 1 \text{ Mbps}} = \frac{8}{5}$$

$$= 1.6 \text{ sec}$$

- So the computer can transmit at the full 6 Mbps for 1.6 seconds.

Q. 18 Write a short note on congestion control. May 16

Ans. :

Principle of congestion control :

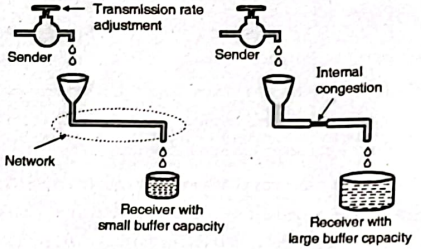
- The basic principle is do not inject a new packet into the network until an old one is delivered.
- TCP tries to do this by dynamically adjusting the window size.
- The steps followed in achieving the congestion control in TCP are as follows :

Step 1 : Detect the congestion :

- This is the first step in congestion control.
- Now-a-days packet loss due to transmission errors is very rare because the optical fiber links are being used.
- So most transmission time-outs (loss of packets) are due to congestions.
- So all the Internet TCP algorithms assume that time-outs are caused by congestion and so time outs can be used to detect the congestion.

Step 2 : Try to prevent congestion :

- After establishing a connection, a suitable window size is to be chosen.
- The receiver window size is based on its buffer capacity.
- If the sender adjusts its transmission rate according to this capacity as shown in Fig. 6.12(a), the congestion due to buffer overflow will never take place.



(a) No congestion **(b) Internal**

(G-618) **Fig. 6.12 : Congestion**

- Now consider Fig. 6.12(b). The sender is slow, the receiver has a large buffer capacity but the problem is low internal carrying capacity of the network.
- If the sender is too fast, the water will back up and some will be lost (loss of packets) and congestion will take place.

Conclusion :

- To prevent congestion TCP has to deal with two problems separately – receiver capacity and network capacity.

Q. 19 Compare TCP and UDP services for transport layer. May 12, May 13, May 14, Dec. 14, May 16

Ans. :

Comparison of UDP and TCP :

Table 6.3 : Comparison of UDP and TCP

Characteristic / Description	UDP	TCP
General Description	Simple, high-speed, low-functionality "wrapper" that interfaces applications to the network layer and does little else.	Full-featured protocol that allows applications to send data reliably without worrying about network layer issues.

Characteristic / Description	UDP	TCP
Protocol Connection Setup	Connectionless; data is sent without setup.	Connection-oriented; connection must be established prior to transmission.
Data Interface To Application	Message-based; data is sent in discrete packages by the application.	Stream-based; data is sent by the application with no particular structure.
Reliability and Acknowledgments	Unreliable, best-effort delivery without acknowledgments	Reliable delivery of messages; all data is acknowledged.
Retransmissions	Not performed. Application must detect lost data and retransmit if needed.	Delivery of all data is managed, and lost data is retransmitted automatically.
Features provided to manage flow of data	None	Flow control using sliding windows; window size adjustment heuristics; congestion avoidance algorithms.
Overhead	Very low	Low, but higher than UDP

Q. 20 Enlist and explain SCTP services. May 19, Dec. 19

Ans. :

SCTP Services :

1. Process to process communication.
2. Multiple streams.
3. Multihoming.
4. Full duplex communication.
5. Connection oriented service.
6. Reliable service.

Process to Process Communication :

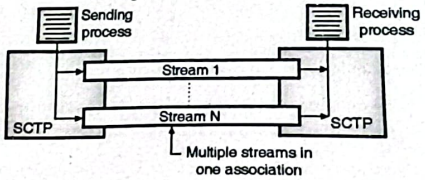
- All the well known ports in the TCP space are used by SCTP.
- But it uses some extra ports as well which have been listed in Table 6.4.

Table 6.4

Sr. No.	Protocol	Port number	Description
1.	IVA	9990	ISDN on IP
2.	M2UA	2904	SS7 Telephone signaling
3.	M3UA	2905	SS7 Telephone signaling
4.	H. 248	2945	Media gateway control
5.	H. 323	1718, 1719, 1720, 11720	IP Telephony
6.	SIP	5060	IP Telephony

Multiple streams :

- TCP being a stream oriented protocol, each connection between a TCP client and a TCP server is a stream.
- If there is a loss at any point in the steam, the data will be blocked there and there will be no delivery of the remaining data.
- This problem is acceptable when the data is in the text form.
- But it is not acceptable when the data is real time video or real time audio data.
- This problem in TCP is overcome in SCTP by providing multiple steams in a single connection between a client and a server.
- This concept is similar to a multilane highway.
- Due to such multiple stream service in SCTP, the delivery of data does not get blocked.
- If one stream is blocked, the other streams can continue delivering the data.
- The concept of SCTP multiple stream service has been illustrated in Fig. 6.13.

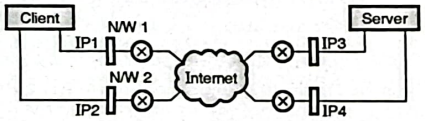


(G-2046) **Fig. 6.13 : SCTP multiple stream is one association**

- In SCTP, each connection is called as an association. Thus each **association** consists of multiple streams.

Multihoming :

- Assume that the sender and receiver hosts are **multihome** hosts.
- That means they are connected to multiple physical addresses with multiple IP addresses.
- In such a situation, if we use TCP, then the TCP connection involves only one source IP address and one destination address.
- Thus TCP does not support the multihoming services.
- However SCTP has been designed for providing multihome services.
- In an SCTP association the sending and receiving hosts can define more than one IP addresses at each end for an association.
- Thus multiple interfaces are established between the sending and receiving hosts in the same association.
- Multihoming is a **fault free** approach which ensures the data delivery without interruption.
- The concept of multihoming has been illustrated in Fig. 6.14 and it is very useful in the real time applications such as **Internet Telephony.**



(G-2047) **Fig. 6.14 : Concept of multihoming in SCTP**

- As shown in Fig. 6.14, the client has been connected to two different local networks with two different IP addresses. SCTP can allow an association between the client and the server using four different pairs of IP addresses.
- However the current SCTP version, it is possible to choose only one pair of IP addresses for data communication.
- The other pairs of IP addresses are used as alternatives only if the main choice fails. That means the current version of SCTP does not allow load sharing among different paths.

Full duplex communication :

- In SCTP also a full duplex service is offered, similar to TCP.
- That means the data can flow in both the directions simultaneously.
- SCTP at each end has a sending buffer and a receiving buffer and the packets are sent in both the directions.

Q. 21 State and explain the important features of SCTP. Dec. 15, Dec. 16, Dec. 18

Ans. :

Features of SCTP :

1. Transmission sequence number.
2. Stream identifier.
3. Stream sequence number.
4. Packets.
5. Acknowledgement number.

Transmission Sequence Number (TSN) :

- In TCP the unit of data is **a byte** because it is a byte oriented protocol.
- Therefore TCP controls the data transfer by numbering bytes with the sequence numbers.
- However the data unit in SCTP is a **Data chunk** and it may or may not have a one to one relationship with the messages produced by the sending process. (This happens due to the fragmentation).
- Therefore in SCTP the **data chunks** are numbered with **transmission sequence number (TSN)** in order to control the data transfer.
- Each TSN is a unique 32 bit number which is stored in the header of the data chunk. TSN has a value between $(2^{32} - 1)$.

Stream Identifier (SI) :

- In SCTP there are more than one stream in each association, and each such stream should be identified using a Stream Identifier (SI).
- The SI is a 16 bit number which starts from 0, and it is stored in header of the corresponding data chunk.

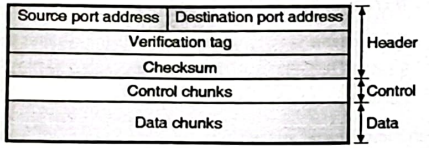
- This will help in placing the data chunk in its stream after receiving it at the destination.
- Thus SCTP uses SI to distinguish between different streams.

Stream Sequence Number (SSN) :

- SCTP uses the stream sequence number (SSN) to distinguish between different data chunks which belong to the same stream.
- The received data chunk at the destination is delivered to the appropriate stream in proper order by the destination SCTP.
- This becomes possible as SCTP defines each data chunk in each stream with stream sequence number (SSN) in addition to an SI.

Packets :

- The SCTP packet design is completely different than that of TCP.
- In SCTP, the data is carried in the form of data chunks while control information is carried as control chunks. Fig. 6.15 shows the SCTP packet.



(G-2048) **Fig. 6.15 : An SCTP packet**

- The role of an SCTP packet is same as that of a TCP packet.
- In SCTP the control information is not a part of the header, but it is included in the control chunks. The control chunks are of different types.
- In SCTP the data is not treated as one entity. Instead it is in the form of several data chunks, and each chunk can belong to a different stream.
- There is no option section is SCTP like TCP. We have to define new chunk types to handle options in SCTP.
- The length of general header in SCTP is 12 bytes as compared to 20 bytes in TCP.
- The checksum length in SCTP is 32 bit as compared to 16 bits in TCP.
- The verification tag field in SCTP packet is used as an association identifier.
- Each association is defined by a unique verification tag. We can have multihoming in SCTP by using different IP addresses.
- In an SCTP packet several different data chunks will be present and each one is defined by TSN, IS and SSN.
- In SCTP, control information and data information are carried in separate chunks.
- In SCTP the TSN, IS and SSN numbers (identifiers) are used only to identify the data chunks.
- The control chunks never use these three identifier. In SCTP the data is contained in data chunks, streams and packets.
- The relationship between these three is as follows :
  1. An association may send many **packets.**
  2. Each **packet** may contain many **chunks.**
  3. These **chunks** may belong to different **streams.**

Acknowledgement Number :

- In TCP the acknowledgement numbers are byte oriented and they refer to the sequence numbers.
- But the acknowledgement numbers in SCTP are chunk-oriented, and they refer to the TSN.
- In SCTP, the control chunks carry the control information.
- The control chunks do not need the TSN.
- These control chunks are acknowledged by another appropriate type of control chunk.
- The sequence number or acknowledgement number is not necessary for the control chunks in SCTP.

Chapter 7 : Application Layer

Q. 1 Write short note on : Domain name system.
Dec. 13, May 14, Dec. 14, Dec. 18, May 19, Dec. 19

Ans. :

Addressing :

- For communication to take place successfully, the sender and receiver both should have addresses and they should be known to each other.
- The addressing in application program is different from that in the other layers.
- Each program will have its own address format. For example an e-mail address is like abc@vsnl.net whereas the address to access a web page is like http :// www.google.com/
- It is important to note that there is an alias name for the address of remote host.
- The application program uses an alias name instead of an IP address.
- This type of address is very convenient for the human beings to remember and use. But it is not suitable for the IP protocol.
- So the alias address has to be mapped to the IP address. For this an application program needs service of another entity.
- This entity is an application program called DNS. Note that DNS is not used directly by the user. It is used by another application programs for carrying out the mapping.

Working of DNS :

- To map a name onto an IP address, an application program calls a library procedure called the **resolver.**
- The name is passed on to the resolver as a parameter.
- The resolver sends a UDP packet to a local DNS server which looks up the name and returns the corresponding IP address to the resolver.
- The resolver then sends this address to the caller. Then the program can establish a TCP connection with the destination or sends in the UDP packets.

Q. 2 What are various components of DNS ? Explain in brief. May 19, Dec. 19

Ans. :

Name Space :

- The names assigned to machines should be selected carefully from the name space.
- There should be a complete control over the relation between the names and the IP addresses.
- The names and corresponding addresses are uniquely defined.
- A name space maps each address to a unique name. It can be arranged in two different ways :
  1. Flat name space.
  2. Hierarchical name space.

Flat name space :

- In a flat name space, a name is assigned to every address.
- This type of name is simply the sequence of characters.
- That means it does not have any structure.
- The flat name space is not suitable for large systems like Internet, because there can be ambiguity and / or duplication.

Hierarchical name space :

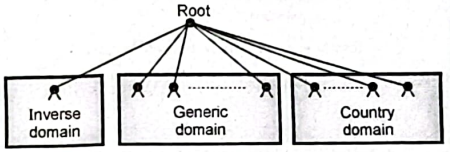
- In the hierarchical name space, each name is made of many parts.
- The first part may correspond to the name of an institution, the second part may define the department and so on.
- The part that defines the nature of institution and name of institution is assigned by a central authority.
- The responsibility of deciding the rest of the name can be given to that institute itself.
- That institute can add suffix or prefix to the name for defining its host or resources.

Q. 3 Explain DNS in Internet. Dec. 15

Ans. :

DNS in Internet :

- The domain name space (tree) is divided into three different sections as shown in Fig. 7.1.
  1. Generic domain.
  2. Country domain.
  3. Inverse domain.



(G-636) **Fig. 7.1 : Use of DNS in Internet**

Generic domains :

- The registered hosts are defined in the generic domains according to their generic behaviour e.g. com for commercial organizations.
- The first level in the generic domains section allows 14 possible lables. Some of them are given in Table 7.1.

Table 7.1 : Generic domain lables

Lable	Description
aero	Airline or aerospace related companies.
com	Commercial organizations.
coop	Cooperative business organizations.
edu	Educational institutions.
gov	Government institutions.
int	International organizations.
mil	Military organization.
net	Network support centers.
org	Non-profit organizations.

Country domain :

- This domain section uses two character country abbreviations eg. US for united states.
- Second lable in this domain can specify organization or national designations.

Inverse domain :

- The inverse domain is used for mapping an address to a name.

Q. 4 Write a short note on World Wide Web.
May 12, May 14, May 16, Dec. 16, May 17

Ans. :

- People have become aware of the power of Internet through WWW.
- HTTP is a file transfer protocol which is specifically designed to facilitate access to the WWW.
- The World Wide Web is an architectural framework for accessing documents which are spread out over a number of machines over Internet.
- It has a colourful graphical interface which is easy for the beginners to use.
- It provides information on almost every subject. The web (also known as WWW) began in 1989 at CERN the European center for nuclear research.
- The web was designed basically to connect scientists stationed all over the world. The web is basically a client-server system.
- The web pages are written in the languages HTML and Java.
- The growth of the World-Wide Web (WWW or simply Web) today is simply phenomenal.
- Each day, thousands of more people join the Internet (above 100 million users at recent estimates).
- Easy retrieval of electronic information along with the multimedia capabilities of Web browsers (like Mosaic or Netscape) are the factors responsible for this explosion.
- This topic provides some basic information behind some of this technology used in accessing the World-Wide Web.

Difference between Web and Internet :

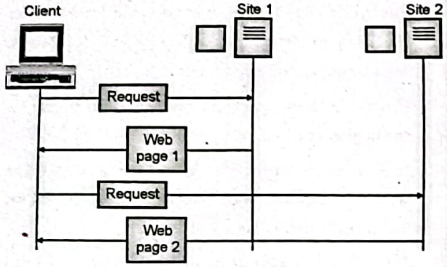
- The Web and the Internet are not the same thing.
- The Web is a collection of standard protocols or instructions, sent back and forth over the Internet to gain access to information.
- The Internet, on the other hand, is a "network of networks" -- a more physical entity.

Q. 5 Explain the architecture of WWW.
Dec. 12, Dec. 16

Ans. :

WWW architecture :

- The WWW is a distributed client/server service. A client (user) uses a browser to access a service using a server.
- But the service provided is distributed over a number of separate locations called as sites.
- Fig. 7.2 shows the architecture of WWW.



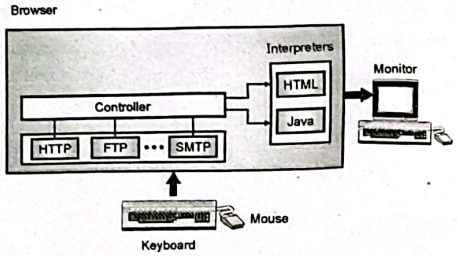
(G-655) **Fig. 7.2 : WWW architecture**

- As shown in Fig. 7.2, there are number of sites and each site holds a number of web pages.
- These pages can be retrieved and viewed by using browsers.
- The client sends a request through its browser to get a web document from a particular site.
- This request contains the site address and web page address (called URL) along with some other information.
- The server at the requested website finds the document and sends it to the client.

Q. 6 Explain how a web page is accessed through internet by a browser. Dec. 18

Ans. :

- Even though a number of browsers are available around, the browser architecture is nearly the same for all of them.
- Each browser consists of the following parts :
  1. A controller.
  2. Client programs.
  3. Interpreters.
- Fig. 7.3 shows the general architecture of a browser.



(G-665) **Fig. 7.3 : Browser architecture**

- The controller receives input from the keyboard or mouse.
- It then uses the client programs like HTTP, FTP etc to access the document.
- After accessing the document, the controller makes use of an interpreter such as HTML or Java (depending on type of document) and displays the accessed document on the screen.

Q. 7 Explain URL and cookies. Dec. 17, May 19, Dec. 19

Ans. :

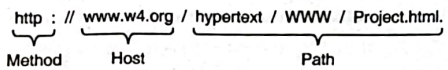
Uniform resource locator :

- The client accessing a web page needs an address.
- The HTTP uses the URL to facilitate the access of any document distributed over the world.
- The URL specifies any information on Internet by using four thing as shown in Fig. 7.4(a).
- They are as follows :
  1. Method or protocol. 2. Host computer.
  3. Port. 4. Path.

(G-660) **Fig. 7.4(a) : URL**

- Method is the protocol used such as FTP, HTTP which helps retrieving the desired information.
- Host is the computer where the required information is located.
- The name of the computer begins with www but this is not mandatory.

- URL can optionally contain the server's port number.
- If the port is to be included then it should be inserted between host and path and it should be separated by a colon, as shown in Fig. 7.4(a).
- Path is the name of the file where the information is located.
- The port and path fields are separated from each other by a slash.
- **Version :** The latest version of HTTP is 1.1 but the versions 0.9 and 1 are also used.
- The example of URL is shown in Fig. 7.4(b). Note that the port is not included.



(G-1969) **Fig. 7.4(b) : Example of URL**

Cookies : User-Server Interaction :

- We know that the HTTP servers are stateless. The disadvantage of being stateless is that the server cannot identify the client.
- The meaning of statelessness is that the client server relationship gets over as soon as their communication terminates.
- But the advantage of statelessness is that the server design is simplified to a great extent and it permits the engineers to develop high performance web servers which can handle thousands of TCP connections at a time.
- But many a times it is necessary for a web site to identify users. In such cases HTTP uses **cookies.**
- Cookies are defined in RFC 2109 and they allow sites to keep track of users.
- Cookies are not used by all the sites but some of the prominent sites that use cookies are : Yahoo, Amazon etc.

Components of cookie technology :

- Following are the four components of the cookie technology :
  1. A cookie header line in HTTP response message.
  2. A cookie header line in the HTTP request message.
  3. A cookie file kept on the user's end system and managed by user's browser.
  4. A back end database at web site.

Operating principle :

- If a new user X contacts a site (that uses cookies) for the first time, then that web site creates a unique identification number for this new user and then creates an entry in its back end data base.
- This entry is associated with the identification number of user X.
- The server will then respond to X's browser by including the header set-cookies : header, in the HTTP response number of user X.
- For example the header line can be :

  Set-cookie : 1 2 3 4 5 6 7

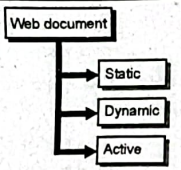
  Where, 1 2 3 4 5 6 7 is the identification number.
- When X's browser receives the HTTP response message, it reads the set cookie : header.
- The browser then appends a line to the special cookie file which is managed by the browser.
- This line will include the hostname of the server and the identification number 1 2 3 4 5 6 7.
- Next time when X visits this same site again, his browser will include the same identification number in each of his HTTP request.
- Thus it is now possible for the web site to track X's activities.
- It is then possible to know the areas of interest of X, which pages does he visit and at what time etc.
- Cookies simplify the internet shopping to a great extent but they remain highly controversial because they are thought as invasion in users privacy.
- It is possible to use cookies to gather personal information about X across a large number of websites.

Q. 8 Explain types of web documents in detail. May 19, Dec. 19

Ans. :

Types of web documents :

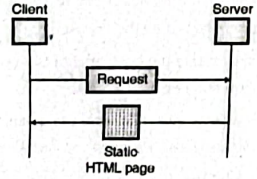
- The web documents can be classified into three categories as shown in Fig. 7.5.



(G-668) **Fig. 7.5 : Categories of web documents**

Static documents :

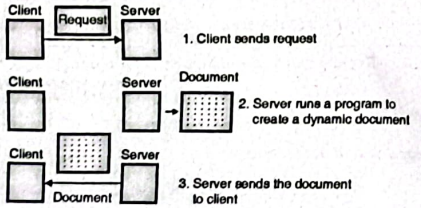
- The contents of static documents are fixed. These contents are created and stored in a server.
- If required the client can get a copy of static document.
- The contents of the static document are determined when it is created.
- These contents cannot be changed when the static document is being used.
- It is possible to change the contents of static document at the server but the user cannot change them.
- The user can display the static document by using a browser as shown in Fig. 7.6.



(G-669) **Fig. 7.6 : Static document**

Dynamic documents :

- The dynamic documents are not present in a predefined format, like static documents.
- A dynamic document is created by a web server on the request for the document from a browser.
- Refer Fig. 7.7 to understand how a dynamic document is created and passed on to the client.

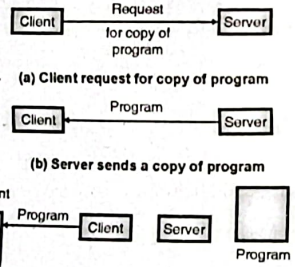


(G-670) **Fig. 7.7 : Dynamic document**

- First the client sends a request to the web server. After receiving this request, the web server will execute an application program to create a dynamic document.
- The server returns the dynamic document as a response of the request to the client.
- The contents of a dynamic document will be different corresponding to every request.
- A simple example of a dynamic document is to get time and data from the server.
- A server follows the steps given below to handle dynamic documents :
  1. The server checks the URL in order to find if it has defined a dynamic document.
  2. If the URL has defined the dynamic document, then the server executes the program.
  3. The output of this program is the dynamic document. It is returned back to the client.

Active documents :

- Active document can be defined as the program that is needed to be run at the client side.
- The examples of active documents are the programs creating animated graphics on the screen or the ones which help interaction with the user.
- Refer Fig. 7.8 to understand this concept.



(G-671) **Fig. 7.8 : Active document**

- It shows that whenever a browser requests for an active document, the server will send a copy of document in the form of byte code.
- The active document will then be run at the browser (client) site.

- The server stores the active document in the form of a binary code.
- The active document is stored on the server but it is not run on the server.
- The client receives the document and stores it, and can run it as many times as required without repeating the request.
- The server sends the active document to the client in the binary form.
- So it is possible to compress it at the server's site and then decompress it at the clients site.
- This will save the bandwidth as well as the transmission time.

Steps in creation of an active document :

- Refer Fig. 7.8 to understand the creation, compilation and execution of an active document.
  1. At the server, a program is written in source code and stored in a file.
  2. Then the program is compiled and binary code is creates and stored in a file at the server's site.
  3. A client (browser) requests for a copy of program as shown in Fig. 7.8(a). This program is transported from the server to the client in the compressed form.
  4. The client converts the received program from binary code into executable code using its own software.
  5. The client runs the program to create the desired result which can include animation or interaction with the user.

Q. 9 What is HTML ? State advantages and disadvantages of HTML. Dec. 10, Dec. 13, May 14, May 17

Ans. :

- The web pages are created by using a language called HTML
- It uses certain marks to format the text. For example if a part of text is required to be "boldface" then we can use the beginning and ending bold face tags (marks) in the text as shown below :

<B> – Beginning of boldface
</B> – End of boldface.

- Here <B> and </B> are the instructions for the browser.
- The browser will make the part of the text between these tags bold. HTML lets the user to use only ASCII characters for the main text as well as for formatting instructions.
- So every computer can receive the whole document as an ASCII document.
- The formatting instructions are used by the browser to format the data.

Advantages of HTML :

1. Any one can edit it.
2. It is easy to learn and use.
3. People located in different parts of world can work on the same document.
4. It widens the access to web publishing for non-technical users.
5. It is a very flexible tool which can be used for a number of applications.
6. It can be installed free of cost.

Disadvantages of HTML :

1. As anyone can edit, this may be too open for some applications (for example confidential documents).
2. It is open to SPAM and vandalism.
3. Requires Internet connectivity to collaborate.
4. Due to flexibility of its structure, the structure can become disorganized.
5. It takes a long time to choose the colour scheme of page and to create tables, forms etc.
6. It can only create static and plain pages. It is not useful to create dynamic pages.

Q. 10 Write a short note on : HTTP Dec. 16

Ans. :

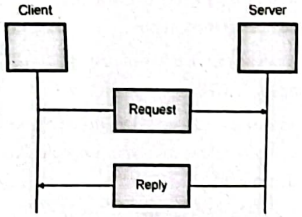
HTTP :

- The main function of HTTP is to access data on WWW.
- This protocol can access the data in various forms such as plaintext, hypertext, audio, video etc.

- The function of HTTP is equivalent to a combination of FTP and SMTP.
- It uses services of TCP. It uses only one TCP connection (port 80).
- There is no separate control connection like the one in FTP.
- Only the data transfer takes place between the client and server so there is only one connection and it is the data connection.
- The data transfer in HTTP is similar to SMTP. The format of the messages is controlled by MIME like headers.

Principle of HTTP Operation :

- The principle of HTTP is simple. A client sends a request. The server sends a response.
- The request and response messages carry data in the form of a letter with a MIME like format.
- Fig. 7.9 shows the HTTP transactions between client and server.



(G-657) **Fig. 7.9 : HTTP transaction**

- The client initializes the transaction by sending a request message and the server responds by sending a response.

Q. 11 Write short note on electronic mail system.

May 13, Dec. 13, Dec. 15, Dec. 16, May 19, Dec. 19

Ans. :

- One of the most popular network services is electronic mail (e-mail).
- Simple Mail Transfer Protocol (SMTP) is the standard mechanism for electronic mail in the internet.
- The first e-mail systems simply consisted by file transfer protocols.
- But some of the limitations of this system were as follows :
  1. It is difficult to send a message to a group of people.
  2. Message did not have any internal structure. So its computer processing was difficult.
  3. The sender never used to know if a message arrived or not.
  4. It was not easy to handover one's e-mails to someone else for the purpose of managing them when one is out of town or country for sometime.
  5. The user interface with the transmission system is poorly integrated.
  6. It was not possible to create and send messages containing a text, drawing, facsimile and voice together.
- So more elaborate e-mail systems were proposed. ARPANET e-mail proposals were published as RFC 821 (transmission protocol) and RFC 822 (message format).
- These are used in Internet.

Q. 12 Explain basic functions of electronic mail.

Dec. 15, Dec. 16, Dec. 18, May 19, Dec. 19

Ans. :

- An e-mail system consists of two subsystems :
  1. User agents.
  2. Message transfer agents.

  User agents : They enable users to read and send e-mail.

  Message transfer agents : They move the messages from the sender to the receiver.

Basic functions :

- E-mail systems support five basic systems which are as follows :
  1. Composition.
  2. Transfer.
  3. Reporting.
  4. Displaying.
  5. Disposition.

1. Composition :

- The process of creating messages and to answer them is known as composition.
- The system can also provide assistance with addressing and a number of header fields attached to each message.

2. Transfer :

- It is the process of moving messages from the sender to the recipient.
- This includes establishment of a connection from sender to destination or some intermediate machine, transferring the message, and breaking the connection.

3. Reporting :

- The reporting system is designed to tell the sender about whether the message was delivered or rejected or lost.

4. Displaying :

- It is the process of displaying the incoming messages so that it can be read by the user.
- For this purpose simple conversions and formatting are required to be done.

5. Disposition :

- This is concerned with what the recipient does with the received message. Disposition is the final step in e-mail system.
- Some of the possibilities are as follows :
  1. Throw after reading.
  2. Throw before reading.
  3. Save messages.
  4. Forward messages.
  5. Process messages in some other way.

Advanced features of E-mail systems :

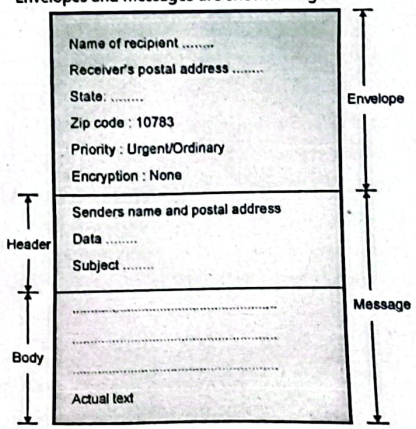
- Some of the advanced features included in addition to the basic functions are as follows :
  1. Forwarding an e-mail to a person away from his computer.
  2. Creating and destroying mailboxes to store incoming e-mail.
  3. Inspecting contents of mailbox, insert and delete messages from the mailboxes.
  4. Sending a message to a large group of people using the idea of mail list.
  5. To provide the facility of registered e-mail.
  6. Automatic notification of undelivered e-mails.
  7. Carbon copies.
  8. High priority E-mail (setting the priority of E-mail).
  9. Secret (encrypted e-mail).
  10. Alternative recipient : This allows automatic forwarding of an e-mail to an alternate recipient if the main recipient is not available.

E-mail envelope :

- In the modern e-mail systems, there is a distinction made between the e-mail and its contents.
- An e-mail envelope contains the message, destination address, priority, security level etc.
- The message transport agents such as SMTP use this envelope for routing.

Message :

- The actual message inside the envelope is made of two parts :
  1. Header 2. Body
- Header carries the control information while body contains the message contents.
- Envelopes and messages are shown in Fig. 7.10.



(G-640) **Fig. 7.10 : Envelope and message**

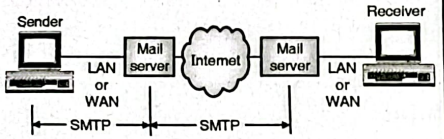
Q. 13 What is the role of SMTP and POP-3 server in E-mail system ?

Dec. 14, Dec. 16, May 17, Dec. 17, Dec. 18

Ans. :

Role of SMTP :

- The actual mail transfer is carried out through the message transfer agent.
- A system should have the client MTA in order to send a mail and it should have a server MTA in order to receive one.
- SMTP is the protocol which defines MTA client and server in the Internet.
- As shown in Fig. 7.11, the SMTP is used twice, once between the sender and sender's mail server and then between the two mail servers.

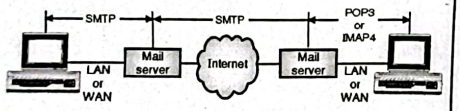


(G-641) **Fig. 7.11 : SMTP range**

- The job of SMTP is simply to define how commands and responses be sent back and forth.
- Each network can choose its software package for implementation.

POP 3 :

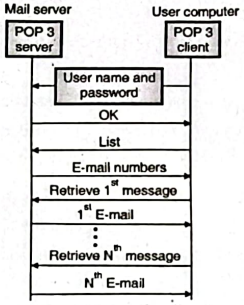
- The POP3 consists of client POP3 software and server POP3 software.
- Out of these, the client POP3 software is installed on the receiving computer whereas the mail server gets the server POP3 software installed on it.
- When the user wants to download email from the mailbox on the email server, the events take place in the following sequence. Refer Fig. 7.12.



(G-645) **Fig. 7.12 : Use of POP 3 or IMAP 4**

1. The client (user) establishes a connection with the server on TCP port 110.
2. The client then sends its user name and password to the server in order to access the mailbox.
3. The user is then allowed to list and get the mail messages one by one.

- This is called as downloading. It is illustrated in Fig. 7.13.



(G-647) **Fig. 7.13 : Downloading in POP3**

Modes of POP 3 :

- POP3 has two modes of operation :
  1. Delete mode and 2. Keep mode.
- **Delete mode :** In this mode the mail is deleted from the mailbox after each retrieval.
- This mode is used when the user is working on his permanent computer because it is then possible for him to save and rearrange the received mail after reading it.
- **Keep mode :** If operated in this mode, the mail remains in the mailbox after retrieval.
- This mode is used when the user accesses mail away from the primary computer. The read mail can be organized later.

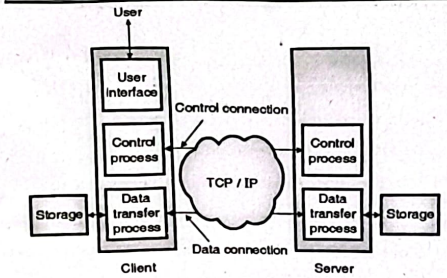
Q. 14 Explain file transfer protocol.

Dec. 17, Dec. 18, May 19, Dec. 19

Ans. :

File transfer protocol :

- A standard mechanism provided by the Internet which helps in copying a file from one host to the other is known as the File Transfer Program (FTP).
- The basic model of FTP is shown in Fig. 7.14.



(G-648) **Fig. 7.14 : Basic model of FTP**

- FTP establishes two types of connections between the client and server.
- One of them is used for data transfer and the other is for the control information.
- The fact that FTP separates control and data makes it very efficient.
- The control connection uses simple rules of communication.
- Only one line of command or a line of response is transferred at a time.
- But the data connection uses more complex rules due to the variety of data types being transferred.
- FTP uses port 21 for the control connection and port 20 for the data connection. Both these are well known TCP ports.
- As shown in Fig. 7.14 the client is made of three blocks namely :
  1. User interface.
  2. Control process.
  3. Data transfer process.
- The server has two blocks : the control process and data transfer process.
- The control connection connects the control processes while data connection connects the data transfer processes as shown in Fig. 7.14.
- The control connection is kept alive during the entire interactive FTP session.
- The data connection is first opened, file is transferred and data connection is closed. This is done for transferring each file.

Control connection :

- This connection is created in the same way as the other application programs.
- Control connection remains alive during the entire process.
- The IP uses minimize delay type service because this is an interactive connection between a user and a server.

Data connection :

- Data connection uses the port 20 at the server site.
- This connection is opened when data to be transferred is ready and it is closed when transfer of data is over.
- The data connection does not remain open continuously like control connection. It is opened and closed many times as per requirement.

Q. 15 Explain FTP in detail with respect to server and client communication.

Dec. 18

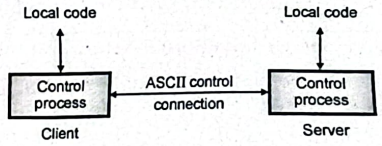
Ans. :

Communication in FTP :

- FTP operates in client – server environment.
- The two computers involved in communication may be different in terms of the operating systems, character sets, file structures and file formats etc.
- FTP can make them compatible.
- The approaches for communication over control connection and data connection are different from each other.

1. Communication over control connection :

- Refer Fig. 7.15 to understand the FTP's approach for the communication over the control connection.



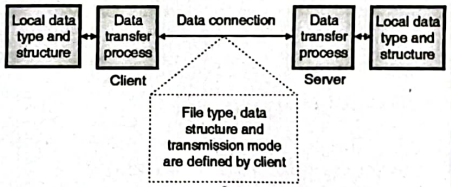
(G-649) **Fig. 7.15 : Communication over control connection**

- Similar to SMTP, FTP uses a set of ASCII characters to communicate across the control connection.

- Communication is achieved through a process of commands and response. One command is sent at a time.
- Each command or response is only of one short line.
- So it is not necessary to think about file format or file structure.
- Each line is ended with a two character token. The two characters used in the token are carriage return and line feed.

2. Communication over data connection :

- The purpose of implementing a data connection is to transfer a file.
- For this the client has to define the following :
  1. Type of file being transferred.
  2. Structure of data in the file.
  3. Mode of transmission.
- Before the transmission over data connection, the communication over control connection is performed.
- Refer Fig. 7.16 to understand communication over data connection.



(G-650) **Fig. 7.16 : Communication over the data connection**

- The problem of heterogeneity is solved by defining three attributes of communication : file type, data structure and transmission mode.

Q. 16 Explain the three FTP transmission modes.

May 16

Ans. :

FTP transmission modes :

- FTP uses one of the following modes to transfer a file :
  1. Stream mode.
  2. Block mode.
  3. Compressed mode.

1. Stream mode :

- In this mode the data is delivered from FTP to TCP in the form of continuous stream of bytes.
- TCP chops this data into segments of appropriate size.
- Stream mode is the default mode of transmission.

2. Block mode :

- In this mode, data delivery from FTP to TCP takes place in the form of data blocks.
- Each such block is preceded by a 3 byte header.

3. Compressed mode :

- For big files the data can be compressed. Generally a run length encoding is used for compression.

Q. 17 Explain : Telnet protocol.

Dec. 13, Dec. 16, Dec. 18

Ans. :

Telnet protocol :

- The long form of TELNET is TErminal NETwork. It was proposed by ISO as a standard TCP/IP protocol for a virtual terminal service.
- TELNET enables a user to establish a connection to a remote system.

Concepts related to TELNET :

- Some of the important concepts related to TELNET are as follows :
  1. Time sharing environment.
  2. Login : Local or Remote.
  3. Network Virtual Terminal.

Time sharing environment :

- TELNET was designed during those days when almost all the operating systems were operating on the time - sharing principle.
- In the time sharing environment there is a large central computer which supports all the users.
- All the processing is done by the central computer, and each user feels that it is a dedicated computer.

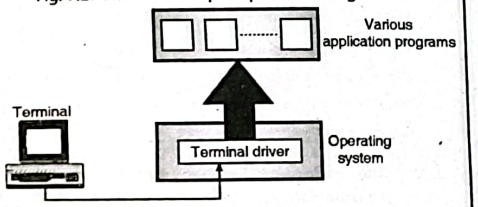
- The users can access all the common system resources, use all the programs or switch from one program to the other.

Login :

- In a system based on time sharing, every user must have an identification and a password for his authentication.
- Whenever a user wants to access the system he will log into the system with his user id and password.
- The system will check the password to allow only the authorised users to access the resources.
- The logic can be one of the following two types :
  1. Local login.
  2. Remote login.

1. Local login :

- The user login into a local time sharing system is called as local login.
- Fig. 7.17 illustrates the principle of local login.

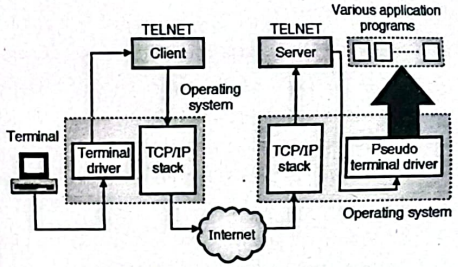


(G-1793) **Fig. 7.17 : Local login**

- The local login takes place in a step - by - step manner as follows :
  1. The user types at the keyboard of a terminal.
  2. The terminal driver accepts these keystrokes.
  3. It converts the keystrokes to characters.
  4. It passes the characters to operating system.
  5. The O.S. understands the combination of characters.
  6. It allows access of intended application to the user.

2. Remote login :

- The user will have to go for the remote login process when he wants to access an application program residing on a remote computer.
- He can do it using the TELNET client and server programs. Fig. 7.17(a) illustrates the principle of remote login.



(G-1794) **Fig. 7.17(a) : Principle of remote login**

- Remote login takes place in a step-by-step manner as follows :
  1. The user types at the keyboard of a terminal.
  2. The terminal driver at local O.S. accepts the characters but sends them to TELNET client without interpreting them.
  3. TELNET client converts them into NVT characters. NVT is Network Virtual Terminal. This is a universal character set.
  4. NVT characters are delivered to TCP/IP stack (local).
  5. The NVT characters travel on the Internet and reach the TCP/IP stack of the remote machine.
  6. The NVT characters are applied to the TELNET server which converts them appropriately so that the remote computer can understand them.
  7. These characters are applied to a software called pseudo terminal driver.
  8. The O.S. at the remote machine then passes the character to the intended application.

Q. 18 Explain Bootstrap protocol. May 17

Ans. :

Bootstrap protocol :

- BOOTP is more efficient than RARP because a single BOOTP message specifies many items needed at startup, including a computer's IP address, the address of router and the address of a server.

- BOOTP also includes a vendor-specific field in the reply that allows hardware vendors to send additional information.
- BOOTP uses UDP to carry messages and that UDP messages are encapsulated in IP datagrams for delivery.
- BOOTP places all responsibility for reliable communication on the client. BOOTP uses UDP for message delivery.
- Messages can be delayed, lost, delivered out of order or duplicated.
- But IP does not provide a checksum for data, the UDP datagram could arrive with some bits corrupted.
- To guard against corruption, BOOTP requires that UDP use checksums.
- It also specifies that requests and replies should be sent with the do not fragment bit set to accommodate clients that have too little memory to reassemble datagrams.
- To handle datagram loss, BOOTP uses the conventional technique of **timeout** and retransmission.

Two-Step Bootstrap Procedure :

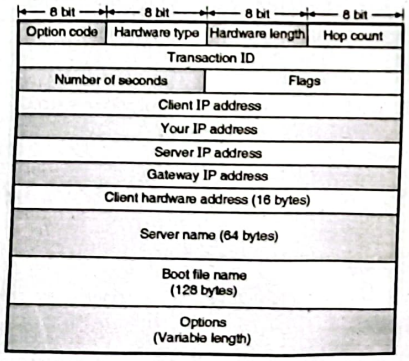
- BOOTP uses a two-step bootstrap procedure. It does not provide clients with a memory image - it only provides the client with information needed to obtain an image.
- The client then uses a second protocol (e.g. TFTP) to obtain the memory image.
- While the two-step procedure many seem unnecessary, it allows a clean separation of configuration and storage.
- A BOOTP server does not need to run on the same machine that stores memory images.
- In fact, the BOOTP server operates from a simple database that only knows the names of memory images.
- Keeping configuration separate from storage is important because it allows administrators to configure sets of machines so they act identically or independently.
- The BOOT FILE NAME field of a BOOTP message illustrates the concept.
- Suppose an administrator has several workstations with different hardware architectures and suppose that when users boot one of the workstations, they either choose to run UNIX or a local operating system.
- Because the set of workstations includes multiple hardware architectures, no single memory image will operate on all machines.
- To accommodate such diversity, BOOTP allows the BOOT FILE NAME field in a request to contain a generic name like "unix" which means, "I want to boot the UNIX operating system for this machine".
- The BOOTP server consults its configuration database to map the generic name into a specific file name that contains the UNIX memory image appropriate for the client hardware and returns the specific (i.e. fully qualified) name in its reply.
- Of course, the configuration database also allows completely automatic bootstrapping in which the client places zeros in the BOOT FILE NAME field and BOOTP selects a memory image for the machine.
- The advantage of the automatic approach is that it allows users to specify generic names that work on any machine; they do not need to remember specific file names or hardware architectures.

Q. 19 Draw and explain function of each field of DHCP message format. **May 17**

Ans. :

DHCP message format :

- The format of a DHCP packet has been shown in Fig. 7.18.



(G-1995) Fig. 7.18 : DHCP packet format

- Let us describe each field in the DHCP packet.

1. Operation code :

- This is an 8 bit field which is used to define the type of DHCP packet.
- If this field contains (1) then the packet is **request type** and if this field contains (2) then the packet is **reply type.**

2. Hardware type :

- This 8-bit field is used to define the type of physical network.
- An integer has been assigned to each type of network e.g. the value of this field is 1 for Ethernet.

3. Hardware length :

- This is an 8-bit field which is used for defining the length of the physical address in bytes.
- The value of this field is 6 for Ethernet because the physical address of Ethernet is 6 byte long.

4. Hop count :

- This is an 8-bit field which is used for define the maximum number of hops a packet can travel.

5. Transaction ID :

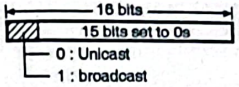
- This is a 32-bit or 4-byte long field which carries an integer in it.
- The contents of this field are known as transaction **identification** and it is set by the client.
- This field is used for matching a reply with the request.
- The same value is returned by the server in its reply packet.

6. Number of seconds :

- This is a 16-bit field which is used to indicate the amount of time (in seconds) elapsed from the instant at which the client started to boot.

7. Flag :

- This is a 16-bit long field, as shown in Fig. 7.19. Out of these 16 bits, only the leftmost bit is used and the remaining 15 bits are set to 0s.



(G-1996) Fig. 7.19 : Format of the flag field

- The leftmost bit is used to specify a forced broadcast reply (instead of unicast) from the server.

8. Client IP address :

- This 4-byte long field is used to carry the client IP address.
- A "0" in this field indicates that the client does not have this information.

8. Your IP address :

- This is also a 4-byte long field which is used to carry the clients IP address.
- This address is requested by the client and filled by the server in the reply message.

10. Server IP address :

- This is also a 4-byte long field which contains the IP address of the server.
- This address is sent by the server in the reply message.

11. Gateway IP address :

- This is a 4-byte or 32 bit long field that contains the IP address of a router which is filled in the reply message by the server.

12. Client hardware address :

- This is a 16-byte field which contains the physical address of the client.

13. Server name :

- This is a 64 byte long field which is filled on the optional basis by the server in a reply packet.
- This field consists of a null terminated string containing the domain name of the server.
- If no information about the server name is to be given, then the server should fill up this field with all zeros.

14. Boot filename :

- This is a 128-byte field which contains a null terminated string consisting of full pathname of the boot file.
- This path can be used by the client in order to obtain additional information about booting.

- This field is filled by the server in the reply message on the optional basis.
- If the server does not want to fill data in this field, then the entire field should be filled up with 0s.

15. Options :

- This is a 64-byte field which can be used for a dual purpose as follows :
  1. It is used to carry some additional information such as default router address or network mask.
  2. Or it is used to carry some specific information about the vendor.
- It is important to note that, the **options** field is used only in the **reply message.**

□□□