CHAPTER 11

Unit III

Network Layer

Syllabus :

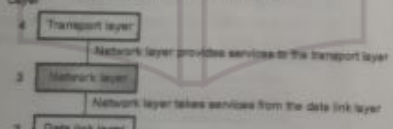
Introduction to network layer, Network layer services, Packetizing, Routing and forwarding, Other services, IPv4 addressing, Address space, Classful addressing, Unicast routing, General idea, Least cost routing, Routing algorithms, Distance vector routing, Link state routing, Path vector routing.

11.1 Network Layer :

- The network layer is responsible for carrying the packet from the source all the way to destination. In short it is responsible for host-to-host delivery.
- The network layer has a higher responsibility than the data link layer, because the data link layer is only supposed to move the frames from one end of the wire to the other end.
- Thus network layer is the lowest layer that deals with the end-to-end transmission.

Position of network layer :

- Fig. 11.1.1 shows the position of network layer in the 5 layer internet model. It is the third layer.



(G-433) **Fig. 11.1.1 : Position of network layer**

- It receives services from the data link layer and provides services to the transport layer.
- The network layer was designed to solve the problem of delivery through several links. The network layer is also called as the **Internetwork** layer.
- In addition to the host-to-host delivery the network layer is also responsible for **routing** the packets through the router.
- As a pure concept we can imagine that the Internet is a black box which connects a very large number of computers in the entire world together.
- But the Internet also is not a single network. It is infact the network of many networks or links.
- That means the Internet is an **internetwork** which is actually a combination of LANs and WANs.
- All these LANs and WANs are connected to each other via a connecting device such as a **router** which acts as a switch.

Routers :

Routers have many ports or interfaces. When it receives a packet at one of its ports, it forwards the packet through another port to the next switch or the final destination.

11.2 Network Layer Services :

- The duty of the network layer in TCP/IP is to provide the host-to-host delivery of datagrams.
- In this section we are going to discuss the services that are expected from the network layer.
- At the sending end, the network layer will accept a packet from its transport layer, encapsulate the packet into datagram and will deliver the packet to the data link layer.
- At the destination, exactly opposite process takes place. That means, at the destination the received datagram is decapsulated to extract the packet from it and the packet is delivered to the transport layer.

11.2.1 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 the 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 link 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.

At the destination :

The sequence of events taking place at the destination is as follows :

1. The network layer packet is received from the data link layer.
2. The received packet is decapsulated and the payload is delivered to the upper layer protocol.
3. If a large packet is fragmented by either the source host or a router, then the responsibility of the network layer at the destination is to wait until all fragments are received, reassemble them and deliver them to the upper layer protocol.

11.2.2 Router's Role :

- The router's 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.

11.2.3 Routing and Forwarding :

The other two important duties of the network layer, which are related to each other are routing and forwarding.

Routing :

- The responsibility of the network layer is to route the packets from its source to destination. The physical network through which the packets travel consists of LANs, WANs and routers.
- Due to this the source and destination are connected to each other via more than one routes.
- It is the responsibility of the network layer to find the best route out of all the possible routes.
- In order to achieve this goal, the network layer must have some concrete strategy for defining the best route.
- In the modern days, this is done by running an appropriate routing protocol which helps the routers to co-ordinate their knowledge about the neighbouring routers and prepare routing tables which can be used on the arrival of a packet.
- These routing protocols should be run before commencement of any communication.

Forwarding :

- We can define the process of forwarding as the action taken by a router when it receives a packet at one of its interfaces.
- A router takes such an action with the help of the decision making tables called as **forwarding table** or **routing table.**
- When a packet arrives at one of the interfaces of a router from one of the attached network, the router has to forward it to another attached network.
- The router has to make this decision with the help of piece of information present in the packet header.
- This piece of information can be the **destination address** or a **label.** The router can use this information to find the corresponding output interface number in the **forwarding table.**

11.3 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

Let's discuss these one by one.

11.3.1 Error Control :

- Eventhough it is possible to implement the error control at the network layer level, the design engineers have neglected this issue.
- One possible reason for this is that the packets may get fragmented at every router due to which the error checking becomes inefficient.
- However a **checksum** field has been added to the datagram in order to control any corruption in the header only. The error control is not applicable to the whole datagram.
- Thus there is no direct error control provided by the network layer in the Internet. But an auxiliary protocol ICMP is used by the Internet for providing some error control to the datagram.

11.3.2 Flow Control :

- The purpose of providing the flow control is to regulate the data rate of the source so as to avoid the receiver getting overwhelmed.
- The receiver will be overwhelmed if the upper layers at the sending end are producing data at a rate which is higher than the rate at which the upper layers at the destination can consume it (data).

- So as to control the sender's data rate, some kind of a feedback mechanism should be setup so that the receiver can tell the source that it (receiver) has overwhelmed with excess data.
- It is important to remember that the network layer does not directly provide any flow control
- The flow control is not provided at the network layer level because it is provided for most of the upper layer protocols and there is no need to provide flow control again which makes the design of network layer complex.

11.3.3 Congestion Control :

- This is another important issue to be handled at the network layer. **Congestion** will take place if the source computer sends more datagrams than the capacity of the network or routers.
- In this situation, the routers will drop some of the received packets.
- But this will make the congestion worse because the error control mechanism present at the upper layers will retransmit the packets dropped by the routers.
- Sometimes the congestion becomes so bad that the system collapses and no datagrams are delivered at the destination.
- The congestion control at the network layer is never implemented in the Internet.

11.3.4 Quality of Service (QoS) :

- The quality of service in the Internet has become more important since new applications like multimedia communication have been introduced.
- The Internet has grown as it successfully provides the quality of service to support all the modern day applications.
- However the QoS provisions are not implemented in the network layer. They are mostly implemented in the upper layers.

11.3.5 Security :

- During the early days of the Internet, security was not a major design concern due to limited (small) number of users. Hence the network layer was designed without any security provisions.
- But security has become a big concern now. But network layer is connectionless. Hence to provide security at the network layer we need to have another virtual level in order to change the connectionless service to connection oriented one.
- The virtual layer is known as IPsec.

11.4 IPv4 Addresses :

- Each computer connected to the Internet should be identified uniquely. The identifier used for this purpose is called as the **Internet address** or IP address.
- The hosts and routers on the Internet have unique IP addresses.
- The current version of IP (Internet Protocol) is IPv4 whereas the advanced version is IPv6.
- The IPv4 address is a 32-bit address and it is used for defining the connection of a host or router to the Internet. **Thus an IP address is an address of the interface.**

11.4.1 Uniqueness of IP Addresses :

- The IP address is **unique** and **universal.** That means each IP address defines only **one connection** to the Internet.
- At any given time, no two devices connected to the Internet can have the same IP address.
- But if a device is connected to the Internet via two connections through two different networks, then it can have two different IP addresses.
- All the IPv4 addresses are 32 bit long and they are used in the source address and destination address fields of the IP header.
- The IP addresses for hosts are assigned by the network administrator. For Internet it has to be obtained from the network information center.

11.4.2 Address Space :

- The IPv4 protocol has an address space. It is defined as the total number of addresses used by the protocol.
- If N number of bits are used for defining an address then the address space will be 2^N addresses.
- For IPv4, N is 32 bits. Hence its address space is 2^{32} or 4, 294, 967, 296 (more than 4 billion). So theoretically more than 4 billion devices could be connected to the Internet.
- Thus **the address space** of IPv4 is 2^{32}.

11.4.3 Notation :

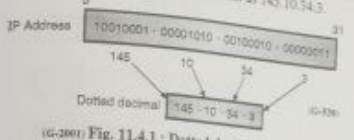
- The IPv4 addresses can be shown, use three different notations as follows :
  1. Binary notations (base 2).
  2. Dotted decimal notation (base 256).
  3. Hexadecimal notation (base 16).
- Out of these the **dotted decimal** notation is most commonly used.

Dotted decimal notation :

- This notation has become popular because of the two advantages it offers. This notation makes the IPv4 address more compact and easy to read.

- The 32 bit IPv4 address is grouped into groups of 8-bits each separated by decimal points (dots).
- Each 8-bit group is then converted into an equivalent decimal number as shown in Fig. 11.4.1.
- Each octet (byte) can take a value between 0 and 255. Therefore the IPv4 address in the dotted decimal notation has a range from 0.0.0.0 to 255.255.255.255.
- For example the IPv4 address of

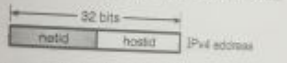
  1001 0001.00001010 00100010 00000011 is denoted in the dotted decimal form as 145.10.34.3.



(G-2001) **Fig. 11.4.1 : Dotted decimal notation**

11.4.4 IPv4 Address Format :

- A 32 bit IPv4 address consists of two parts. The first part is called as **net id** i.e. network identification which identifies a network on the Internet and the second part is called as the **host id** which identifies a host on that network.
- Fig. 11.4.2 shows the IPv4 address format. Note that the **net id** and **host id** are of variable lengths depending on the class of address.
- Note that class D and E addresses are not divided into net id and host id for the reasons discussed later on.



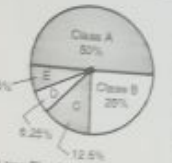
(G-2002) **Fig. 11.4.2 : IPv4 address format**

11.5 Classful Addressing :

- The concept of IP addresses is few decades old. It uses the concept of **classes.** This architecture is called as the **classful addressing.**
- Later on in mid 1990s a new architecture of addressing was introduced which was known as **classless addressing.** This new architecture has superseded the original architecture.
- In this section we are going to discuss the classful addressing.

11.5.1 IPv4 Address Classes :

- In the classful addressing architecture, the IP address space has been divided into five classes : A, B, C, D and E.
- Fig. 11.5.1 shows the percentage of occupation of the address space by each class.
- 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	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. 11.5.1 : Classful addressing occupation of address space**

11.5.2 Formats of Various Classes :

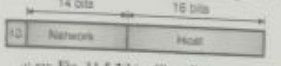
Class A format :

- The formats used for IPv4 addresses are as shown in Fig. 11.5.2. The IPv4 address for class A networks is shown in Fig. 11.5.2(a).
- The network field is 7 bit long as shown in Fig. 11.5.2(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.

(G-530) **Fig. 11.5.2(a) : Class A IPv4 address formats**

Class B format :

- The class B address format is shown in Fig. 11.5.2(b).
- The first two fields identify the network, and the number in the first field must be in the range 128 - 191.



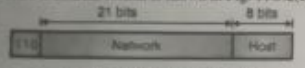
(G-531) **Fig. 11.5.2(b) : Class B format**

- Class B networks are large. Host numbers 0.0 and 255.255 are reserved, so there can be upto 65,534 (216-2) 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. 11.5.2(c).



(G-4355) **Fig. 11.5.2(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. 11.5.2(d).

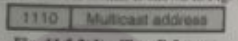


Fig. 11.5.2(d) : Class D format

- 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. 11.5.2(e) shows the address format for a class E address. This address begins with 11110 which shows that it is reserved for the future use.

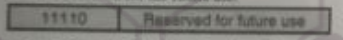
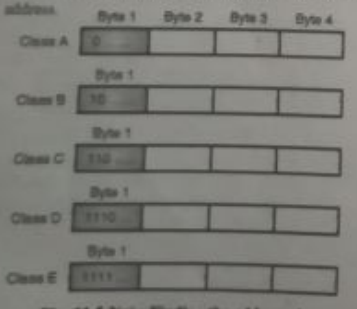


Fig. 11.5.2(e) : IPv4 address for class E network

- 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.

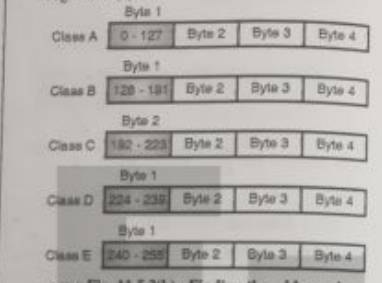
11.5.3 How to Recognize Classes :

- When an IPv4 address is given to us either in the binary or dotted decimal notation, we can find the class of the address.



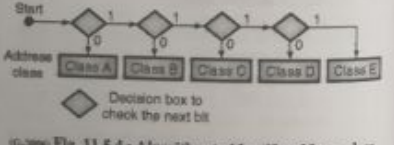
(G-2004) **Fig. 11.5.3(a) : Finding the address class**

- If the given address is in the binary notation then we can identify its class by inspecting the first few bits of the address. This is as shown in Fig. 11.5.3(a).
- If the given address is in the dotted decimal notation then we can identify the address class by inspecting the first byte of the address. This is as shown in Fig. 11.5.3(b).



(G-2005) **Fig. 11.5.3(b) : Finding the address class**

- It is important to note here that there are some special addresses which fall in class A or E. These special addresses are to be treated as the exceptions to the classful addressing. We have discussed them later in the chapter.
- In computers, the IPv4 addresses are generally stored in the binary notation format. Therefore it is possible to write an algorithm which can identify the address class by using the continuous checking process.
- The principle of such an algorithm has been shown in Fig. 11.5.4.

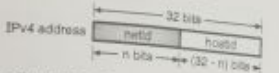


(G-2006) **Fig. 11.5.4 : Algorithm to identify address class**

11.5.4 Two Level Addressing :

- The IPv4 addressing is used for defining a destination for an Internet packet at the network layer.
- At the time when classful addresses were designed, the Internet was considered as the network of networks. In other words the whole Internet was divided into a number of smaller networks with many hosts connected to each network.

- Normally an organization which wants to connect to the Internet creates a network and the Internet authorities allocate a block of address to the organization. These addresses can be in class A, B or C.
- All the addresses allotted to an organization belong to a single block. Therefore each IPv4 address in classful addressing system is made up of two parts namely **net id** and **host id** as shown in Fig. 11.5.5.



(G-2007) **Fig. 11.5.5 : Two level addressing in classful addressing**

- The job of the **net id** is to define a network and that of the **host id** is to define a particular host in that network.
- As shown in Fig. 11.5.5 if n bits define **net id** then the remaining (32-n) bits define **host id**.
- The value of 'n" is not same for all the classes. Infact it is depend on the class as shown in Table 11.5.1.

Table 11.5.1

Class	Value of n
A	n = 8
B	n = 16
C	n = 24

11.5.5 Extracting Information in a Block :

- A block is nothing but a range of addresses. For any given block we would be interested to extract the following three pieces of information :
  1. The total number of addresses in the block.
  2. The first address of the block.
  3. The last address in the block.
- Before extracting all this information, we have to identify the class of the address as discussed earlier.
- Once we find the class of the block, we will have the values of "n" (the length of **net id** in bits) and (32 – n) i.e. the length of the **host id** in bits.
- It is now possible to obtain the three pieces of information mentioned above as shown in Fig. 11.5.6.

1. Total number of addresses in the block :

The total number of IPv4 addresses in the given block will be equal to,

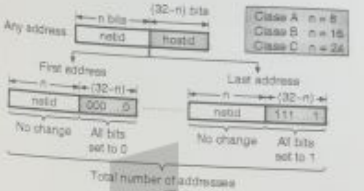
$$N = 2^{(32-n)} \quad \ldots(11.5.1)$$

2. First address in the block :

The first address in the given block can be obtained by keeping the leftmost "n" bits in the address as it is and setting all the (32 – n) rightmost bits to 0 as shown in Fig. 11.5.6.

3. Last address in the block :

The last address in the given block can be obtained by keeping the leftmost "n" bits in the address as it is and then setting all the (32 – n) rightmost bits to 1 as shown in Fig. 11.5.6.



(G-2008) **Fig. 11.5.6 : Information extraction in classful addressing**

11.5.6 Network Address :

- The network address is an address that defines the network itself. It cannot be assigned to a host. Fig. 11.5.7 shows the examples of network addresses for different classes.

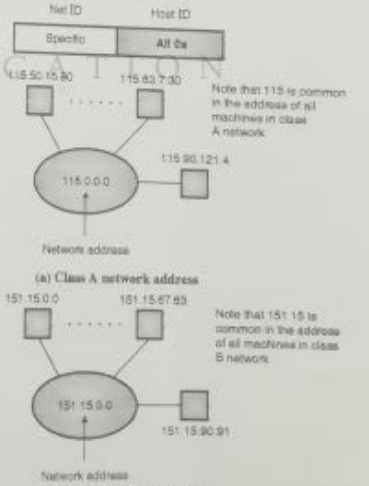
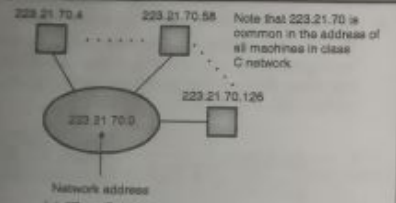


Fig. 11.5.7 (Contd...)



(c) Class C network address

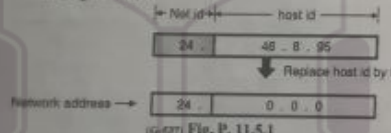
(G-536) Fig. 11.5.7

- The following examples will enable you to find the network address.

Ex. 11.5.1 : For the address 24.46.8.95 identify the type of network and find the network address.

Soln. :

- Examine the first byte. Its value is 24 i.e. it is between 0 and 127. So it is a class A network.
- So only the first byte defines the Net id. So we can find the network address by replacing the host id with 0s.
- The process of obtaining the network address is shown in Fig. P. 11.5.1.



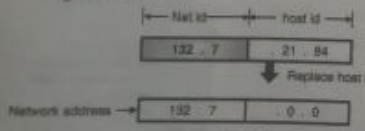
(G-537) Fig. P. 11.5.1

So the network address is 24.0.0.0.

Ex. 11.5.2 : For the address 132.7.21.84 find the type of network and the network address.

Soln. :

- Examine the first byte. It is 132 i.e. between 128 and 192. So it is a class B network.
- So the first two bytes define the net id. Replace the host id with 0's to get the network address as shown in Fig. P. 11.5.2.

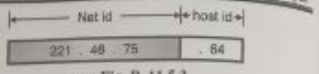


(G-538) Fig. P. 11.5.2

So the network address is 132.7.0.0.

Ex. 11.5.3 : Find the class of the network if the address is 221.46.75.84.

Soln. : The first byte is 221 i.e. between 192 and 255. So this is a class C network. The net id and host id are as shown in Fig. P. 11.5.3.



(G-539) Fig. P. 11.5.3

What is the difference between net id and network address ?

The network address is different from a net id. A network address has both net id and host id, with 0s for the host id.

Where to use the network address ?

The network address is used to route the packets to the desired location.

11.5.7 Network Mask or Default Mask :

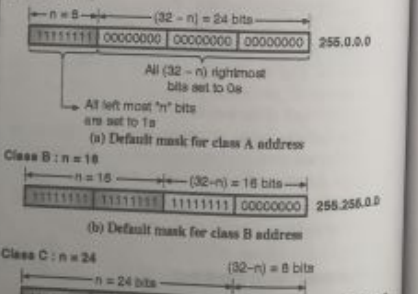
- Earlier we have discussed the methods for extracting different pieces of information. But all these methods are theoretical methods which are useful in explaining the concept.
- But practically these methods are not used. When a packet arrives at the input of the router in the Internet, it uses an algorithm to extract the **network address** from the destination address in the received packet.
- This can be achieved by using a **network mask.**

Definition of 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.

11.5.8 Default Masks for Different Classes :

- We know that the value of n is different for different classes. Therefore their default masks also will be different.
- The default masks for class A, B and C addresses are as shown in Fig. 11.5.8.



(a) Default mask for class A address

(b) Default mask for class B address

(c) Default mask for class C address

(G-3009) Fig. 11.5.8

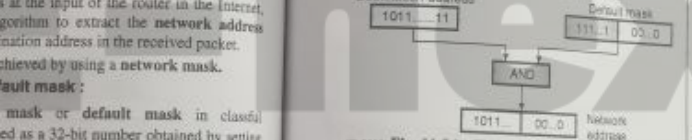
- Table 11.5.2 enlists the default masks of the three classes of IPv4 addresses.

Table 11.5.2 : Default masks

Address class	Default mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

11.5.9 Finding Network Address using Default Mask :

- The router uses the AND operation for extracting the network address from the destination address of the received packet.
- The router ANDs the destination address with the default mask to extract the network address as shown in Fig. 11.5.9.
- It is possible to use the default mask to find the number of addresses and the last address in the block.



(G-2010) Fig. 11.5.9 : Finding a network address using the default mask

11.5.10 Three Level Addressing : Subnetting :

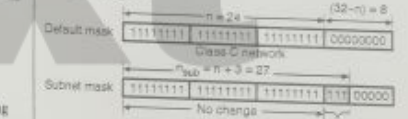
- As discussed earlier, the originally designed IP addresses were with two level addressing with **net id** and **host id.**
- The two level addressing is based on the principle that in order to reach a host on the Internet, we have to reach the network first and then the host.
- But very soon it became evident that the two level addressing would not be sufficient for the following two reasons :
  1. First it was needed to divide a large network of an organization (to which a block in class A or B is allotted) into many smaller **subnets** (subnetworks) for improved management and security.
  2. Second reason is more important. The blocks in class A and B were almost depleted and the blocks in class C were smaller than the needs of most organization. Therefore the organizations had to divide their allotted class A or B block into smaller subnetworks and share them.

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 :

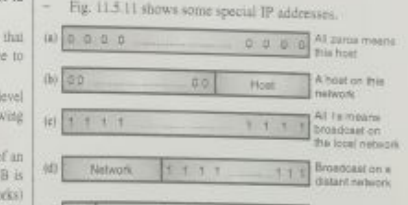
- The **network mask** or **default** mask that we discussed earlier is used when the given network is **not** to be divided into smaller subnetworks i.e. when **subnetting** is **not** to be done.
- But when the given network is to be divided into smaller subnets i.e. when subnetting is to be done, we need to create a **subnet mask** for each subnet.
- Fig. 11.5.10 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. 11.5.10.
- In Fig. 11.5.10, we have shown the default mask and subnet mask when a class C network is to be divided into 8 subnets.



(G-2011) Fig. 11.5.10 : Default and subnet masks

11.5.11 Special IP Addresses :

- Fig. 11.5.11 shows some special IP addresses.



(G-540) Fig. 11.5.11 : Special IP addresses

- All zeros means this host or this network and all 1s means broadcast address to all hosts on the indicated network.
- The IP address 0.0.0.0 is used by the hosts when they are being booted but not used afterward.
- The IP addresses with 0 as the network number refer to their own network without knowing its number as shown in Fig. 11.5.11(b).

- The address having all ones is used for broadcasting on the local network such as a LAN as shown in Fig. 11.5.11(c).
- Refer Fig. 11.5.11(d). This is an address with proper network number and all 1s in the host field. This address allow machines to send broadcast packets to distant LANs anywhere in the Internet.
- If the address is "127. Anything" as shown in Fig. 11.5.11(e) then it is a reserved address **loopback testing.** This feature is also used for debugging network software.

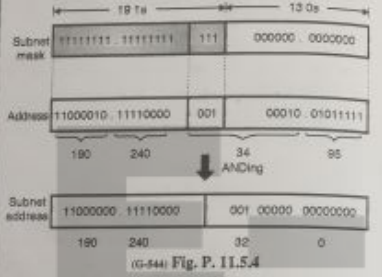
11.5.12 Limitations of IPv4 :

- The most obvious limitation of IPv4 is its address field. IP relies on network layer addresses to identify end-points on networks, and each networked device has a unique IP address.
- IPv4 uses a 32-bit addressing scheme, which gives it 4 billion possible addresses. With the proliferation of networked devices including PCs, cell phones, wireless devices, etc., unique IP addresses are becoming scarce, and the world could theoretically run out of IP addresses.
- If a network has slightly more number of hosts than a particular class, then it needs either two IP addresses of that class or the next class of IP address. For example, let use say a network has 300 hosts, this network needs either a single class B IP address or two class C IP addresses. If class B address is allocated to this network, as the number of hosts that can be defined in a class B network is $(2 \wedge 16 - 2)$, a large number of host IP addresses are wasted.
- If two class C IP addresses are allocated, as the number of networks that can be defined using a class C address is only $(2 \wedge 21)$, the number of available class C networks will quickly exhaust. Because of the above two reasons, a lot of IP addresses are wasted and also the available IP address space is rapidly reduced.
- Other identified limitations of the IPv4 protocol are : Complex host and router configuration, non-hierarchical addressing, difficulty in re-numbering addresses, large routing tables, non-trivial implementations in providing security, QoS (Quality of Service), mobility and multi-homing, multicasting etc.
- To overcome these problems the internet protocol version 6 (IPv6) which is also known as internet protocol, next generation (IPng) was proposed.
- In IPv6 the internet protocol was extensively modified for accommodating the unforeseen growth of the internet.
- The format and length of the IP addresses has been changed and the packet format also is changed.

Ex. 11.5.4 : A router inside an organization receives the same packet with a destination address 190.240.34.95. If the subnet mask is /19 (first 19-bits are 1s and following bits are 0s). Find the subnet address.

Soln. :

- To find the subnet address, AND the destination address with the subnet mask as shown in Fig. P. 11.5.4.



(G-544) **Fig. P. 11.5.4**

Thus the subnet address is 190.240.32.0

11.5.13 Classless Addressing :

- Even though the number of actual devices connected to Internet is much less than 4 billion, the address depletion has taken place due to flaws in the classful addressing scheme.
- We have run out of class A and B addresses. To overcome these problems, the classless addressing is now being tried out.
- In the classless addressing, there are no classes but the address generation take place in blocks.

Address blocks :

- Address block is defined as the range of addresses.
- In the classless addressing, when an entity wants to get connected to the internet, a block (range) of addresses is granted to it.
- The size of this block i.e. number of addresses depends on the size of the entity as well as its nature.
- That means for a small entity such as a household only one or two addresses will be given whereas for a larger entity like an organization, thousands of addresses can be allotted.

Restrictions :

Some of the restriction on classless address blocks have been imposed by the internet authorities in order to simplify the process of address handling.

1. The addresses in a block should be continuous, i.e. serial in manner.
2. The total number of addresses in a block has to be equal to some power of 2 i.e. $2^1, 2^2, 2^3 \ldots$etc.

3. The first address should be evenly divisible by the number of addresses.

11.5.14 Supernetting :

- The class A and class B addresses are almost depleted. But class C addresses are still available.
- But the size of class C address with a maximum number of 256 addresses does not satisfy the needs of an organization. More addresses will be required.
- The solution to this problem is **supernetting.**
- In supernetting an organization combines several class C blocks to create a large range of addresses i.e. several networks are combined to create a supernetwork.
- By doing this the organization can apply for a set of class C blocks instead of just one.

Example of supernetting :

- If an organization needs 1000 addresses, they can be obtained by using four C blocks (one C block corresponds to 256 addresses).
- The organization can then use these addresses as one supernetwork as a whole.

Note : The classful addressing is almost obsolete now and it is being replaced with classless addressing.

11.5.15 Who Decides the IP Addresses ?

- No two IP addresses should be same. This is ensured by a central authority that issues the prefix for the network number portion of the IP address.
- Locally an ISP is to be contacted in order to get a unique IP address prefix.
- At the global level the Internet Assigned Number Authority (IANA) allots an IP address prefix to the ISP. Thus it is ensured that the IP addresses are not duplicated.
- Conceptually IANA is a wholesales and ISP is a retailer of the IP addresses because ISP purchases IP addresses from IANA and sells them to the customers.

11.5.16 Registered and Unregistered Addresses :

- Registered IP addresses are required for computers which are accessible from the Internet but not every computer that is connected to the Internet.
- For security reasons, networks use firewalls or some other technologies for protecting the computers.
- The firewalls will enable the workstations to access the Internet but do not allow the other systems on the Internet to access them.
- These workstations are given the unregistered private IP addresses. These addresses are assigned by the network administrator without obtaining them from an ISP (Internet Service Provider) or IANA.
- These are special network addresses in each class as shown in Table 11.5.3. These addresses are to be used for private networks and are called **unregistered addresses.**
- We can choose any of these unregistered address while building our own private network.

Table 11.5.3 : IP addresses for private networks

Class	Network address
A	10.0.0.0 through 10.255.255.255
B	172.16.0.0 through 172.31.255.255
C	192.168.0.0 through 192.168.255.255

11.5.17 Solved Examples :

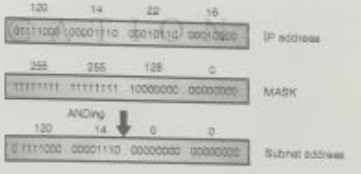
Ex. 11.5.5 : Find the sub-network address and the host id for the following :

Qr. No.	IP Address	MASK
(a)	120.14.22.16	255.255.128.0
(b)	140.11.36.22	255.255.255.0
(c)	141.181.14.16	255.255.224.0
(d)	200.34.22.156	255.255.255.240

Soln. :

Step 1 : To find the subnet address :

In order to find the subnet address we have to AND the IP address and the mask as follows :



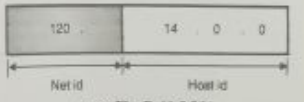
(G-553) **Fig. P. 11.5.5(a)**

So the subnet address is 120.14.0.0.

Similarly we can find the other subnet addresses.

Step 2 : Host id :

- Examine the first byte of the subnet address. It is 120 which is between 0 and 127. Hence this is a class A network.
- So only the first byte corresponds to the net id and the remaining three bytes correspond to the host id as shown in Fig. P. 11.5.5(b).



(G-554) **Fig. P. 11.5.5(b)**

So the host id is 14.0.0.

Similarly we can find the other host id.

Ex. 11.5.6 : The IP address of a host on class C network is 198.123.46.237. Four networks are allowed for this network. What is subnet mask ?

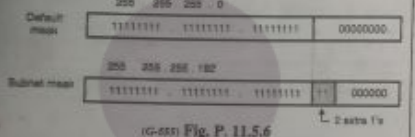
Soln. :

The default mask for a class C network is,

255.255.255.0

In order to have four networks, we must have two extra 1s.

Hence the default mask and subnet mask are shown in Fig. P. 11.5.6.



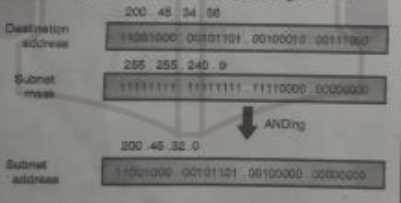
(G-555) **Fig. P. 11.5.6**

Thus the required subnet mask is 255.255.255.192.

Ex. 11.5.7 : What is the subnet address if the destination address is 200.45.34.56 and subnet mask is 255.255.240.0 ?

Soln. :

To find the subnet address we have to AND the IP address and the subnet mask as shown in Fig. P. 11.5.7.



(G-556) **Fig. P. 11.5.7**

Thus the required subnet address is 200.45.32.0.

Ex. 11.5.8 : A company is granted a site address 201.70.64.0. The company needs six subnets. Design the subnets.

Soln. :

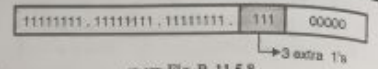
- This is a class C network. So the default mask is,

  255.255.255.0

- As we need 6 subnets, we need three extra 1s. So the subnet mask is,

  255.255.255.224

- In the binary form the subnet mask is as shown in Fig. P. 11.5.8.



(G-557) **Fig. P. 11.5.8**

- In order to have six subnets, we can have 6 different combinations of the 3-extra 1s as shown in Table P. 11.5.8(a).

Table P. 11.5.8(a)

Combination	Subnet number
000	Subnet 1
001	Subnet 2
010	Subnet 3
011	Subnet 4
100	Subnet 5
101	Subnet 6

- So the various addresses of 6 subnets are as shown in Table P. 11.5.8(b).

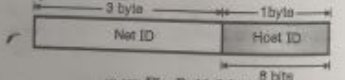
Table P. 11.5.8(b)

Subnet Number	Addresses
1	201.70.64.0 to 201.70.64.31
2	201.70.64.32 to 201.70.64.63
3	201.70.64.64 to 201.70.64.95
4	201.70.64.96 to 201.70.64.127
5	201.70.64.128 to 201.70.64.159
6	201.70.64.160 to 201.70.64.191

Ex. 11.5.9 : For a given class C network 195.188.65.0 design equal subnets in such a way that each subnet has atleast 60 nodes.

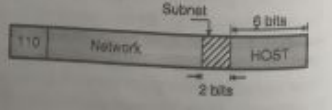
Soln. :

- Fig. P. 11.5.9(a) shows the structure of a class C address in which 3-bytes are reserved for net ID and 1-byte for host ID.



(G-558) **Fig. P. 11.5.9(a)**

- We are expected to design equal subnets such that each subnet has atleast 60 nodes (i.e. 60 users).
- In order to identify at least 60 users we need 6-bits in the host ID.
- The remaining 2-bits are assigned for subnetting as shown in Fig. P. 11.5.9(b).



(G-559) **Fig. P. 11.5.9(b)**

- This shows that there will be four equal subnets each one having at least 60 nodes.

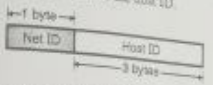
Ex. 11.5.10 : Show by calculations how many network each IP address class can have with one example ?

Soln. :

Number of networks in different IP address :

Class A address :

- The format of class A address is shown in Fig. P. 11.5.10(a). Here one byte defines the network ID and three bytes define the host ID.

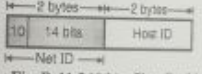


(G-560) **Fig. P. 11.5.10(a) : Class A address**

- The MSB in the network field is reserved. So actually there are only 7-bits in the network field.
- 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. P. 11.5.10(b). 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.

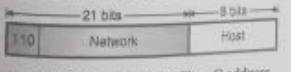


(G-561) **Fig. P. 11.5.10(b) : Class B address**

- 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. P. 11.5.10(c). Here 3-bytes are reserved for network field and only one byte for the host field.
- Out of 24-bits in the network field 3-bits are again reserved. So actually only 21-bits are available.



(G-562) **Fig. P. 11.5.10(c) : Class C address**

- So the number of networks in class C addresses is 2,097,152.

Ex. 11.5.11 : How many host per network in each IP address class can exist, show with example ?

Soln. :

Number of hosts in different IP addresses :

Class A :

There are 3-bytes (24-bits) in the host field. Hence the number of hosts in class A address will be $2^{24} = 16,7772,16$.

Class B :

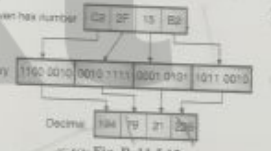
There are 2-bytes (16-bits) in the host field. So the number of hosts in class B address will be 65536 i.e. 2^{16} per network.

Class C :

There is 1-byte (8-bits) in the host field. So number of hosts in class C address will be $2^8 = 256$ per network.

Ex. 11.5.12 : Convert the IP address whose hexadecimal representation is C22F1582 to dotted decimal notation.

Soln. :



(G-563) **Fig. P. 11.5.12**

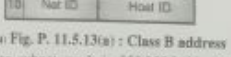
The IP address in the dotted decimal notation is as follows :

194.47.21.226

Ex. 11.5.13 : A class B network on internet has a subnet mask of 255.255.240.0. What is the maximum number of hosts per subnet ?

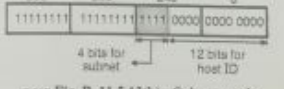
Soln. :

The structure of class B address is as shown in Fig. P. 11.5.13(a).



(G-564) **Fig. P. 11.5.13(a) : Class B address**

The given subnet mask is 255.255.2400. So it is as shown in Fig. P. 11.5.13(b).



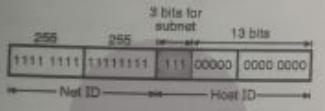
(G-565) **Fig. P. 11.5.13(b) : Subnet mask**

Thus there are 4 extra 1s as shown in Fig. P. 11.5.13(b). So there will be 16 subnets and each subnet can have $2^{12} = 4096$ hosts.

Ex. 11.5.14 : Perform the subnetting of the following IP address 160.111. X.X
Original subnet mask 255.255.0.0
Number of subnets 6 (six)

Soln. :

- The original subnet mask indicates that we are dealing with a class B address.
- In order to have six subnets we need to use 3 extra bits from the bits that are reserved for host ID. So the subnet mask is as shown in Fig. P. 11.5.14.



(G-566) **Fig. P. 11.5.14**

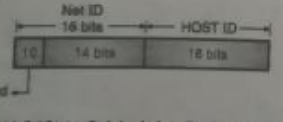
- The 3-bits reserved for subnetting will have 8 combinations from 000 to 111 out of which any six combinations can be used for 6 subnets.
- Let us decide that the combinations 000 to 001 are not to be used. Then the subnet masks for the 6 possible subnets will have the following addresses.

Subnet 1	255.255.64.0
Subnet 2	255.255.96.0
Subnet 3	255.255.128.0
Subnet 4	255.255.160.0
Subnet 5	255.255.192.0
Subnet 6	255.255.224.0

Ex. 11.5.15 : Suppose that instead of using 16-bits for the part of class B address originally, 20-bits had been used. How many class B network addresses would there have been? Give the range of IP addresses in decimal dotted form.

Soln. :

- Fig. P. 11.5.15(a) shows the original class B address format :



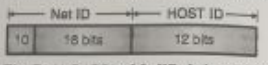
(G-567) **Fig. P. 11.5.15(a) : Original class B address format**

- The first two MSB bits of Net ID part are reserved. Hence, the number of bits actually available for network ID is 14.
- Hence the number of class B networks = 2^{14} = 16382.

Modification :

Now with 20 bits instead of 16 being available for the Net ID part the actually available number of bits for Network part becomes 18. This is shown in Fig. P. 11.5.15(b).

∴ Number of class B networks = 2^{18} = 2, 61, 888



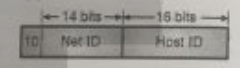
(G-568) **Fig. P. 11.5.15(b) : Modified class B address format**

The range of IP addresses in the decimal dotted form would be 128.0.0.0 to 191.255.255.255.

Ex. 11.5.16 : A network on the Internet has a subnet mask of 255.255.240.0. What is the maximum number of host it can handle? Give the range of IP addresses in decimal dotted form.

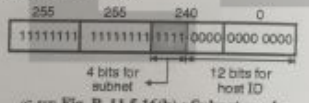
Soln. :

The structure of class B address is as shown in Fig. P. 11.5.16(a).



(G-564) **Fig. P. 11.5.16(a) : Class B address**

The given subnet mask is 255.255.240.0. So it is as shown in Fig. P. 11.5.16(b).



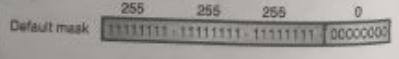
(G-565) **Fig. P. 11.5.16(b) : Subnet mask**

Thus there are 4 extra 1s as shown in Fig. P. 11.5.16(b). So there will be 16 subnets and each subnet can have 2^{12} = 4096 hosts.

Ex. 11.5.17 : For a given class-C network, design 4 equal subnets having minimum 50 nodes in each subnetwork.

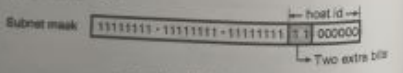
Soln. :

The default mask for a class C network is
255.255.255.0
This is as shown in Fig. P. 11.5.17(a).



(G-571) **Fig. P. 11.5.17(a)**

In order to design 4 equal subnets having a minimum 50 nodes in each subnetwork, we have to use two extra bits from the host id field. So the subnet mask is as shown in Fig. P. 11.5.17(b).



(G-572) **Fig. P. 11.5.17(b)**

In order to have four subnets, we can have four different combinations of the two extra bits as shown in Table P. 11.5.17(a).

Table P. 11.5.17(a)

Combination	Subnet
00	subnet 1
01	subnet 2
10	subnet 3
11	subnet 4

Let the class C address be 201.70.64.0. Then the addresses of the four subnets are as shown in Table P. 11.5.17(b).

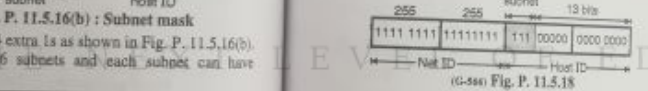
Table P. 11.5.17(b)

Subnet number	Addresses
1	201.70.64.0 to 201.70.64.63
2	201.70.64.64 to 201.70.64.127
3	201.70.64.128 to 201.70.64.191
4	201.70.64.192 to 201.70.64.255

Ex. 11.5.18 : For a given class B network 144.155.0.0 with default subnet mask, how can you divide it into 8 equal subnets ? How many hosts can be accommodated in each sub-network ?

Soln. :

Given class B network : 144.155.0.0. The default subnet mask is 255.255.0.0. In order to have 8 subnets we need to use 3 extra bits from the host id field as shown in Fig. P. 11.5.18.



(G-586) **Fig. P. 11.5.18**

- The 3-bits reserved for subnetting will have 8 combinations from 000 to 111 which can be used for 8 subnets.
- The subnet masks for the 8 possible subnets will have the following subnet masks :

Subnet	Mask
1	255.255.0.0
2	255.255.32.0
3	255.255.64.0
4	255.255.96.0
5	255.255.128.0
6	255.255.160.0
7	255.255.192.0
8	255.255.224.0

Number of hosts in each subnet :

Due to use of extra 3-bits for subnetting, now we have only 13-bits left in the host id field.

∴ No. of hosts in each subnet = 2^{13} = 8192 **...Ans.**

Ex. 11.5.19 : Consider any class – C network with default subnet mask. How many actual hosts can be connected in that network ? Divide that network into 4 equal subnets. What is the new subnet mask ? How many hosts can be connected in each subnet ?

Soln. :

- For a class C network, the default mask is
255.255.255.0
- For a class C network we can connect 2^8 = 256 total hosts.
- As we need 4 subnets, we need two extra 1s. So the subnet mask is
255.255.255.192
- In the binary from the subnet mask is as shown in Fig. P. 11.5.19.

(G-574) **Fig. P. 11.5.19**

- In order to have four subnets we can have the 4 combinations of the two extra 1s as shown in Table P. 11.5.19.

Table P. 11.5.19

Combination	Subnet number
0 0	Subnet 1
0 1	Subnet 2
1 0	Subnet 3
1 1	Subnet 4

- As we have used the 2 MSB bits of host ID field for subnet mask, we have only 6 bits remaining in the host id field.

∴ No. of hosts/subnet = 2^6 = 64.

Ex. 11.5.20 : Consider any class – C network with default subnet mask. Design the subnet in such a way that each has 62 nodes. Write the range of IP addresses for all subnets.

Soln. : Refer Ex. 11.5.19.

- But we want only 62 nodes on each subnet. So 2 nodes on each subnet will be inactive.
- Let the class C address be 201.70.64.0. Then the addresses of the four subnets are as shown in Table P. 11.5.20.

Table P. 11.5.20

Subnet number	Addresses
1	201.70.64.0 to 201.70.64.61
2	201.70.64.64 to 201.70.64.125
3	201.70.64.128 to 201.70.64.189
4	201.70.64.192 to 201.70.64.253

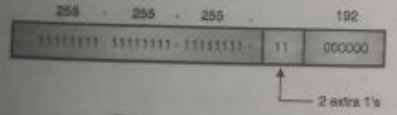
Ex. 11.5.21 : For a given C class network 210.50.60.0, how will you divide it into 4 equal subnets ? What will be the new subnet mask ? Give the network and broadcast address of each subnetwork.

Soln. :

Given : IP address : 210.50.60.0 (class C)

Step 1 : Subnet mask :

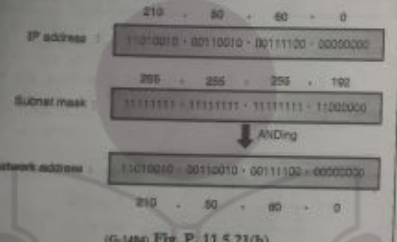
This is class C network. So default mask is given by 255.255.255.0.



(G-1483) **Fig. P. 11.5.21(a) : Subnet mask**

The new subnet mask is 255.255.255.192. ...Ans.

Step 2 : Find network address :

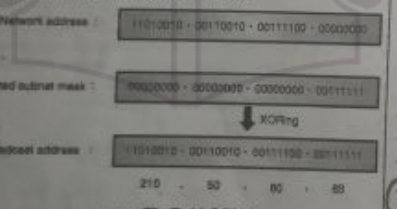


(G-1484) **Fig. P. 11.5.21(b)**

Network address is 210.50.60.0 ...Ans.

Step 3 : Find broadcast address :

To find broadcast address, take inverted subnet mask and perform XOR with network address.



(G-1485) **Fig P. 11.5.21(c)**

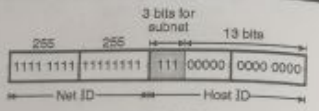
The broadcast address is 210.50.60.63 ...Ans.

Ex. 11.5.22 : For a given class B network 144.155.0.0 with default subnet mask, how can divide it into 8 subnets ? Write the :

1. Range of each subnet.
2. Network IP for 7th subnet.
3. Broadcast IP for the 7th subnet.
4. Subnet mask in subnets.

Soln. :

Given class B network : 144.155.0.0. The default subnet mask is 255.255.0.0. In order to have 8 subnets we need to use 3 extra bits from the host id field as shown in Fig. P. 11.5.22.



(G-566) **Fig. P. 11.5.22**

- The new subnet mask is 255.255.224.0.
- The 3-bits reserved for subnetting will have 8 combinations from 000 to 111 which can be used for 8 subnets.
- The subnet masks for the 8 possible subnets will have the following subnet masks :

Subnet	Mask
1	255.255.0.0
2	255.255.32.0
3	255.255.64.0
4	255.255.96.0
5	255.255.128.0
6	255.255.160.0
7	255.255.192.0
8	255.255.224.0

- The following is the range of subnets :

Subnet	Subnet range
1	144.155.0.0 to 144.155.31.255
2	144.155.32.0 to 144.155.63.255
3	144.155.64.0 to 144.155.95.255
4	144.155.96.0 to 144.155.127.255
5	144.155.128.0 to 144.155.159.255
6	144.155.160.0 to 144.155.191.255
7	144.155.192.0 to 144.155.223.255
8	144.155.224.0 to 144.155.255.255

- Network IP for 7th subnet is 144.155.192.0.
- Broadcast IP for 7th subnet is 144.155.223.255.
- Subnet mask is 255.255.224.0.

11.6 Routing :

- Routing is a very important issue in the network layer. A router creates its routing table so as to help forwarding a datagram in the connectionless services. It also helps in creating a virtual circuit in the connection oriented service.
- In the following sections we are going to discuss about the types of routing and different routing algorithms such as distance vector routing, link state routing and hierarchical routing.

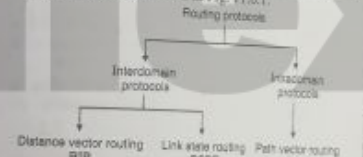
11.6.1 Types of Routing :

- Routing can be broadly classified into three types :
  1. Unicast routing.
  2. Broadcast routing
  3. Multicast routing.

- We can also classify the routing into two types as follows :
  1. Intradomain routing.
  2. Interdomain routing.

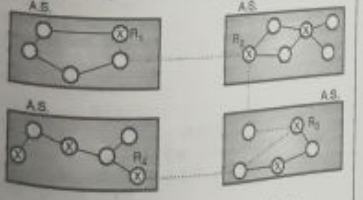
11.6.2 Intra and Interdomain Routing :

- Today the size of the internet is so big that one routing protocol cannot handle the task of updating the routing tables of all the routers.
- Hence an internet is divided into **Autonomous Systems** (AS). An Autonomous System (AS) is a group of networks and routers which is controlled by a single administrator. An AS is shown in Fig. 11.6.1.
- The **intradomain routing** is defined as the routing inside an autonomous system whereas the routing between autonomous system is known as the **interdomain routing.**
- Several intradomain and interdomain protocols are used. They are as shown in Fig. 11.6.1.



(G-1291) **Fig. 11.6.1 : Classification of routing protocols**

- The examples of interdomain routing protocols are :
  1. Distance vector routing
  2. Link state routing.
- An example of intradomain routing protocol is path vector routing.
- Each A.S. is allowed to choose one or more intradomain routing protocols in order to handle the routing inside the A.S. But only one interdomain routing protocol will handle routing between autonomous systems.
- The Routing Information Protocol (RIP) is an implementation of distance vector routing. Whereas the OSPF is an implementation of link state protocol. The BGP is an implementation of the path vector protocol.



(G-1292) **Fig. 11.6.2 : Autonomous systems**

11.6.3 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. 11.6.3.

(G-448) **Fig. 11.6.3 : 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 can not find the destination address.

Metric :

- A metric is defined as the cost assigned for passing through a network.
- The metric assigned to each network depends on the type of protocol.

Interior and exterior routing :

- An Internet is so large that for one routing protocol it is impossible to handle the task of updating the routing tables of all the routers.
- So an Internet is divided into a number of Autonomous Systems (AS). An AS is group of networks and routers.

Interior routing :

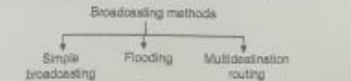
The routing that takes place inside an AS is called as interior routing.

Exterior routing :

The routing that takes place among various autonomous systems is called as exterior routing.

11.6.4 Broadcast Routing :

- In certain applications, the host has to send packets to many or all other hosts.
- If the sender send a packet to all destinations simultaneously then it is called as **broadcasting.**
- Various methods of broadcasting are as follows :



(G-449) **Fig. 11.6.4 : Various methods of broadcasting**

1. **Simple broadcasting :**
   - In this method the source will simply send a distinct (a separate) packet to each destination.
   - This method has two drawbacks :
     1. A lot of bandwidth is wasted.
     2. The source has to have a complete list of all destinations.
2. **Flooding :**
   - Flooding is another method used for broadcasting. The problem with flooding is that it has a point to point routing algorithm.
   - So it consumes a lot of bandwidth and generates too many packets.
3. **Multidestination routing :**
   - This is the third algorithm used for broadcasting.
   - In this algorithm each packet will contain a list of destinations or a bit map which indicates the desired destination.
   - When such a packet arrives at a router, the router first checks all the destinations. Then it decides the set of output lines that will be required based on the destination addresses.
   - The router then generates a new copy of the received packet for each output line to be used. It includes a list of only those destinations that are to use the line in each packet going out on that line. This will save bandwidth to a great extent. Also generation of too many packets right from the sending end will also be avoided.

11.6.5 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.

11.7 Routing Algorithms :

- One of the important functions of the network layer is to route the packets from the source machine to the destination machine.
- The major area of network layer design includes the algorithms which choose the routes and the data structures which are used.
- **Routing algorithm** is a part of network layer software. It is responsible for deciding the output line over which a packet is to be sent.
- Such a decision is dependent on whether the subnet is a virtual circuit or it is datagram switching.

11.7.1 Desired Properties of a Routing Algorithm :

- There are certain desirable properties of a routing algorithm as follows :
  1. Correctness
  2. Robustness
  3. Stability
  4. Fairness and
  5. Optimality.

11.7.2 Types of Routing Algorithms :

Routing algorithms can be divided into two groups :
1. Non-adaptive algorithms.
2. Adaptive algorithms.

1. **Non-adaptive algorithms :**
   - For this type of algorithms, the routing decision is not based on the measurement or estimation of current traffic and topology.
   - However the choice of the route is done in advance, off-line and it is downloaded to the routers.
   - This is called as static routing.
2. **Adaptive algorithms :**
   - For these algorithms the routing decision can be changed if there are any changes in topology or traffic etc.
   - This is called as dynamic routing.
   - In the following sections we are going to discuss various static and dynamic algorithms.

11.7.3 Optimality Principle :

- A general statement about optimality is called as optimality principle.
- It states that if router J is on the optimal path from router I to router K, then the optimal path from J to K will also be along the same route.

Sink tree :
- A set of optimal routes from all the sources to a given destination form a tree called sink tree and it is shown in Fig. 11.7.1. The root of the sink tree is at the destination.
- Note that a sink tree need not be unique. Other trees with the same path lengths may also exist.
- All the routing algorithms are supposed to discover and use the sink trees for all routers.
- In the sink tree of Fig. 11.7.1, the distance metric is the number of hops. In Fig. 11.7.1(b) a sink tree for router B has been shown. The paths from B to every router with minimum number of hops.

(a) A subnet

(b) A sink tree for router B

(G-4W) Fig. 11.7.1

11.8 Static Algorithms :

The examples of static algorithms are :
1. Shortest path routing.
2. Flooding.
3. Flow based routing.

11.8.1 Shortest Path Routing :

- This algorithm is based on the simplest and most widely used principle. Here a graph of subnet is prepared in which each node represents either a host or a router and each arc represents a communication link.
- So as to choose a path between any two routers, this algorithm simply finds the shortest path between them.

How to decide the shortest path ?
- One way of measuring the path length is the number of hops. Another way (metric) is the geographical distance in kilometres.
- Some other metrics are also possible. For example we can label each arc (link) with the mean queuing and transmission delay and obtain the shortest path as the fastest path.

Labels on the arcs :
- The labels on the arcs can be computed as a function of distance bandwidth, average traffic, mean queue length, cost of communication, measured delay etc.
- The algorithm compares various parameters and calculates the shortest path, on the basis of any one or combination of criterions stated above.

Various shortest path algorithms :
- There are many algorithms for computing the shortest path between two nodes.
- One of them is Dijkstra algorithm. The other one is Bellman-Ford algorithm.

11.9 Dynamic Routing Algorithms :

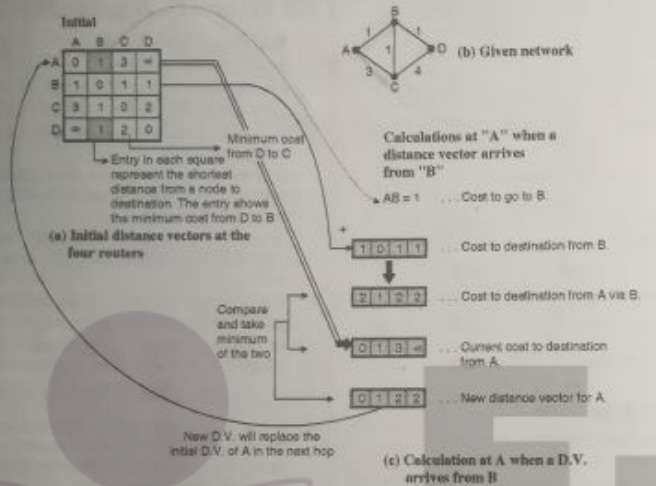
- The modern computer networks normally use the dynamic routing algorithms.
- Two dynamic routing algorithms namely distance vector routing and link state routing are used popularly.
- Both these algorithms are suitable for the packet switched networks.
- Both these algorithms assume that a router knows the address of each neighbouring router and the cost of reaching each neighbour.
- In the distance vector routing, each node tells its neighbours about its distance to every other node in the network.
- In the link state routing, a node tells every other node in the network the distance to its neighbours.
- So both these routing algorithms are distributed type and so they are suitable for large internetworks.

11.9.1 Distance Vector Routing Algorithm :

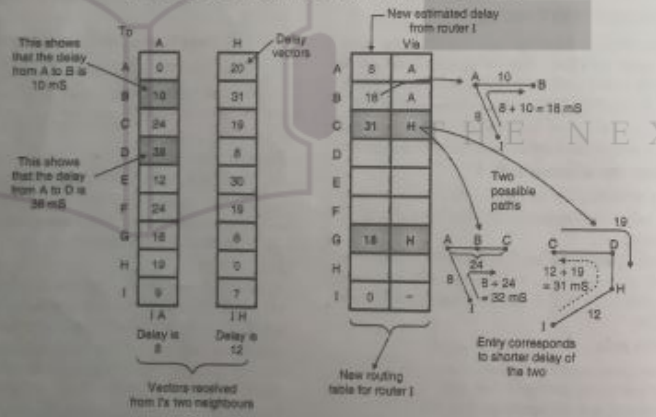
- In this algorithm, each router maintains a table called vector, such a table gives the best known distance to each destination and the information about which line is be used to reach there.
- This algorithm is sometimes called by other names such as :
  1. Distributed Bellman-Ford routing algorithm.
  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. 11.9.1.



(G-465) **Fig. 11.9.1 : Distance vector algorithm at router A**



(G-466) **Fig. 11.9.2(b) : Routing tables**

- Fig. 11.9.1 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. 11.9.1(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. 11.9.2(a) and the routing tables are shown in Fig. 11.9.2(b).

(G-464) **Fig. 11.9.2(a) : A subnet**

- The entries in router tables of Fig. 11.9.2(b) are the delay vectors. For example consider the shaded boxes of Fig. 11.9.2(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. 11.9.2(c) shows the two possible routes between I and G.

(G-466) **Fig. 11.9.2(c)**

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

1 to A Delay = 8 mS
A to G Delay = 16 mS
$\therefore$ I to G Delay = 8 + 16 = 24 msec

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

I to H Delay = 12 mS
H to G Delay = 6 mS
$\therefore$ I to G Delay = 12 + 6 = 18 msec

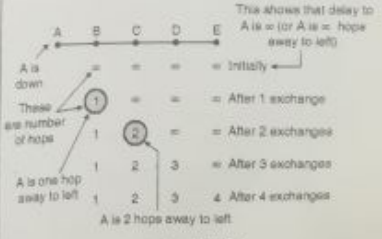
- The best of these values is 18 msec corresponding to the path IHG. Hence it makes an entry in its routing table (I's table) that the delay to G is 18 msec and that the route to use it is via H.
- The new routing table for router I is shown in Fig. 11.9.2(b).
- Similarly we can calculate the delays, from I to different destinations from A to I and enter the minimum possible delay into the I's router table.

Disadvantages :

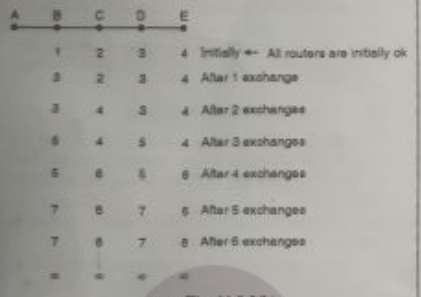
1. The distance vector routing takes a long time in converging to the correct answer. This is due to a problem called count-to-infinity problem. This problem can be solved by using the split horizon algorithm.
2. Another problem is that this algorithm does not take the line bandwidth into consideration when choosing a route. This is a serious problem due to which this algorithm was replaced by the Link State Routing algorithm.

11.9.2 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. 11.9.3 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. 11.9.3(a), B makes an entry in its routing table that A is one hop away to the left.
- All the other routers still think that A is down. So in the second row of Fig. 11.9.3(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.

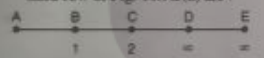


(G-467) **Fig. 11.9.3(a)**



(G-469) **Fig. 11.9.3(b)**

- So after the second vector exchange the entries in the third row of Fig. 11.9.3(a) are :

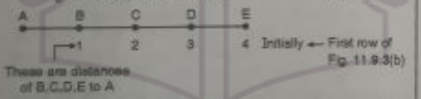


(G-468(a))

- 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.

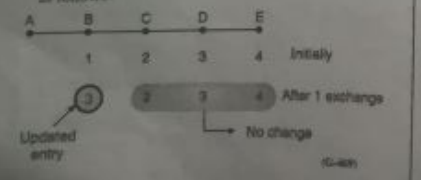
Explanation of Fig. 11.9.3(b) :

- Now refer Fig. 11.9.3(b). Here initially all routers are OK. The routers B, C, D and E have distances of 1, 2, 3 and 4 respectively to A. So the first row of Fig. 11.9.3(b) is as follows :



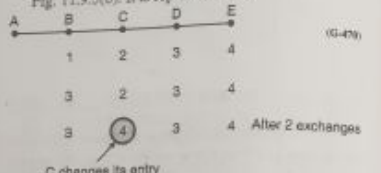
(G-468(b))

- 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. 11.9.3(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. 11.9.3(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. 11.9.3(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.

Split horizon algorithm :

- To avoid the count to infinity problem, several changes in the algorithm have been suggested. But none of them work satisfactorily in all situations.
- One particular method which is widely implemented is called as the **split horizon algorithm.**
- In this algorithm, the minimum cost to a given destination is not sent to a neighbour if the neighbour is the next node along the shortest path.
- For example if node A thinks that the best route to node B is via node C, then node A should not send the corresponding minimum cost to node C.

11.9.3 Link State Routing :

- 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.

11.9.4 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.

11.10 Path Vector Routing :

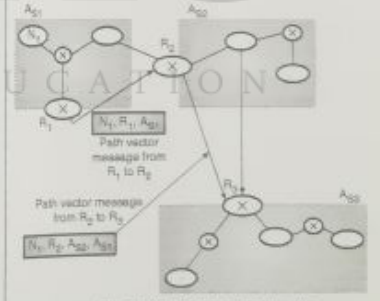
- It is different from both distance vector routing and link state routing.
- Table 11.10.1 shows the example of a path routing table. Each entry in the routing table will have the information about the destination network, the next router and the path to reach the destination.

Table 11.10.1 : Path vector routing table

Network	Next router	Path
N01	R01	AS 12, AS 21, AS 56
N02	R08	AS 20, AS 57, AS 06
.	.	.
.	.	.
.	.	.

11.10.1 Path Vector Messages :

- The autonomous boundary routers participate in path vector routing. Their job is to advertise the reachability of networks present in their A.S. to the neighbour autonomous boundary router.
- Each router that receives a path vector message verifies whether or not the advertised path is according to its policy. Such a policy is made up of rules that are imposed by the router controlling administrator.
- If yes then the router will update its routing table and will modify the message before it is sent to the next neighbour.
- In the modified message it sends its own AS number and replaces the next router entry with its own identification. This process is demonstrated in Fig. 11.10.1.
- Fig. 11.10.1 shows an internet containing three autonomous systems A_{S1} through A_{S3}.
- Router R_1 sends a path vector message to advertise that it is reachable to network N_1. Router R_2 on receiving this message will update its routing table. It then adds its own autonomous system (A_{S2}) to the path, inserts itself as the next router and sends this message to router R_3 as shown in Fig. 11.10.1.



(G-1799) **Fig. 11.10.1 : Path vector messages**

11.10.2 Loop Prevention :

- When a message is received, a router checks it to see if its autonomous system is in the path list to the destination. If it is present it indicates looping is involved which is undesirable and the message is ignored.
- In this way the looping problem and the associated instability which is present in distance vector routing is avoided in path vector routing.

11.10.3 Path Attributes :

- The path is specified in terms of attributes. Each attribute gives some information about the path. Hence the list of attributes helps the receiving router to make a better decision about when to apply its policy.
- Attributes are of two types :
  1. A well known attribute
  2. An optional attribute
- An attribute is called as a well known attribute if it is recognised by every BGP router.
- An optional attribute is the one that need not be recognised by every BGP router.
- The well known attributes are further classified into two categories :
  1. Well known mandatory attributes
  2. Well known discretionary attributes.
- The optional attributes also are classified into two types :
  1. An optional transitive attribute
  2. An optional nontransitive attribute.

Review Questions

Q. 1 State and explain the various services provided by network layer.

Q. 2 What is packetizing ?

Q. 3 Write short note on : routing and forwarding.

Q. 4 Explain error control and flow control.

Q. 5 Write short note on : IPv4 addresses.

Q. 6 What do you mean by uniqueness of IP addresses.

Q. 7 Draw IPv4 address format.

Q. 8 Define classful addressing.

Q. 9 Draw class B IPv4 address format.

Q. 10 How to recognize IPv4 classe.

Q. 11 Write short note on : Two level addressing in classful addressing.

Q. 12 How information is extracted in classful addressing ?

Q. 13 Define default mask.

Q. 14 Write default masks for different classes.

Q. 15 Define subnetting.

Q. 16 Write down limitations of IPv4.

Q. 17 Who decides the IP addresses ?

Q. 18 State the types of routing.

Q. 19 Explain unicast and broadcast routing.

Q. 20 Write down desired properties of a routing algorithm.

Q. 21 Write short note on : optimality principle.

Q. 22 Explain shortest path routing.

Q. 23 Explain distance vector routing algorithms.

Q. 24 Write short note on : Link state routing.

Q. 25 Compare link state routing and distance vector routing.

Q. 26 Write short note on : path vector routing.

□□□

CHAPTER 12

Unit III

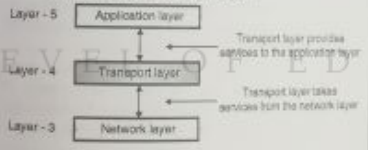
Introduction to Transport Layer

Syllabus :

Introduction to transport layer, Transport layer services, Connectionless and connection oriented protocols, Transport layer protocols, Services, Port number, User datagram protocol, User datagram, UDP services, UDP applications, Transmission control protocol, TCP services, TCP features, Segment.

12.1 Introduction :

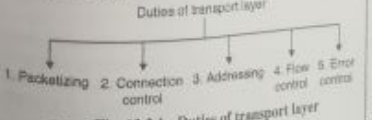
- The transport layer is the core of the Internet model. The application layer programs interact with each other using the services of the transport layer.
- Transport layer provides services to the application layer and takes services from the network layer.
- Fig. 12.1.1 shows the position of the transport layer in the 5-layer internet model. The transport layer is fourth layer in this model. It connects the lower three layers to upper three layers of an OSI layer.



(G-892) **Fig. 12.1.1 : Position of transport layer**

12.2 Transport Layer Duties and Functionalities :

- Transport layer is meant for the process to process delivery and it is achieved by performing a number of functions.
- Fig. 12.2.1 lists the functions of a transport layer.



(G-1407) **Fig. 12.2.1 : Duties 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 in 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.