Introduction to TCP/IP

3.1. Introduction :

In May, 1974, the Institute of Electrical and Electronic Engineers (IEEE) published a paper entitled "A Protocol for Packet Network Interconnection." The paper's authors Vint Cerf and Bob Kahn, described an internetworking protocol for sharing resources using packet-switching among the nodes. A central control component of this model was the transmission control program that incorporated both connection-oriented links and datagram services between hosts. the monolithic Transmission Control Program was later divided into a modular architectur consisting of the Transmission Control Protocol at the connection-oriented layer and the internet protocol at the internetworking (datagram) layer. The model became known informally as TCP/IP, although formally it was henceforth called the Internet Protocol Suite.

3.2. TCP/IP PROTOCOL

TCP/IP is an industry-standard protocol suite for Wide Area Networks (WANs) developed in the 1970s and 1980s by the U.S. Department of Defense (DoD). TCP/IP is a routable protocol that is suitable for connecting dissimilar systems (such as Microsoft Windows and UNIX) in heterogeneous networks. It is the protocol of the worldwide network known as the internet. The internet began as a project funded by the United States Department of Defence in the 1970 to interconnect educational institutions and government installations. At the time it was called ARPAnet (Advanced Research Projects Agency Network). Over time it has evolved into the huge, worldwide network known as the internet. The protocols that make up the internet protocol suite, the best known being TCP (Transmission Control Protocol) and IP (Internet Protocol), have become de-facto standards because of the success of the internet. The entire protocol suit is referred to as TCP/IP.

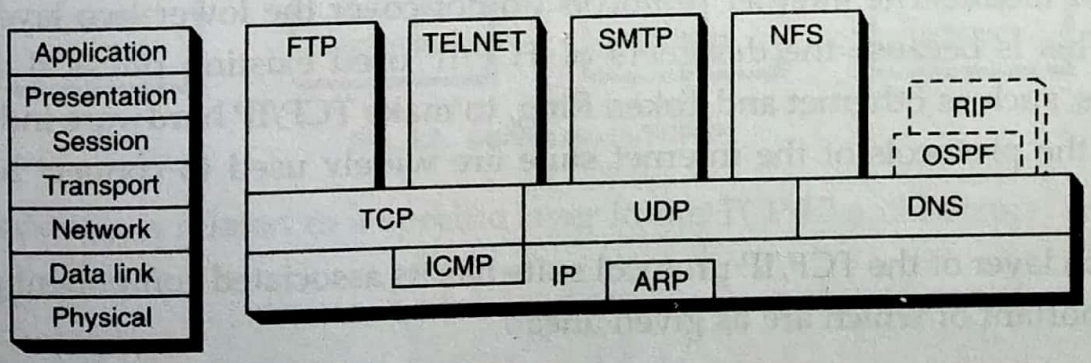


Fig. 3.1. The Relationship between the Internet protocol suite and the OSI model.

The internet suite was developed about ten years before the OSI model w defined and can therefore be only roughly mapped to it. The internet protocol suite w defined according to its own model, known as the internet or DOD model. Followi figure illustrates the relationship between the internet protocol suite and the O reference model.

The four DOD model layers as shown in figure 3.2 and the OSI model layers th correspond to are as follows:

(a) The network access layer corresponds to the physical and data link layers of t OSI model.

(b) The Internet layer corresponds to the OSI network layer. Protocols at this lay are concerned with transporting packets through the internetwork. The ma internet layer protocol is IP (Internet Protocol).

(c) The host-to-host layer corresponds roughly to the OSI transport layer. Protoco at this layer communicate with peer processes in other hosts or networke devices. An example of a host-to-host protocol is TCP

(d) The process/application layer corresponds to the OSI session, presentation an application layers. Protocols at this layer provide applications services on th network. Examples of protocols at this layer are Telnet (a terminal emulator) an FTP (a file transfer protocol).

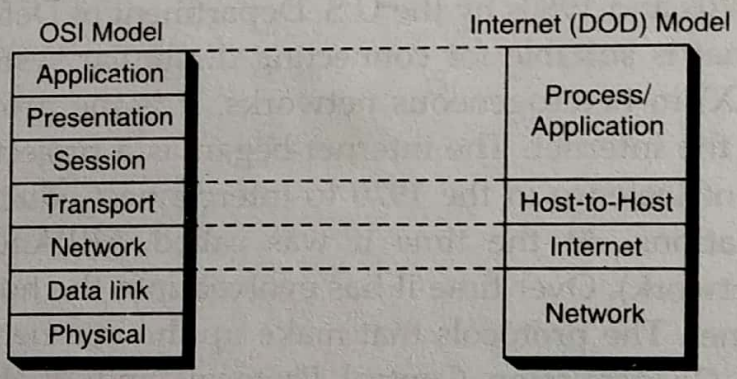


Fig. 3.2. The Internet (DOD) model mapped to the OSI model

The given figure shows the Internet (DOD) model and various protocols mappe to the OSI model. The internet protocols do not cover the lower two layers of the O model. This is because the designers of TCP/IP used existing physical and data lin standards, such as Ethernet and Token Ring, to make TCP/IP hardware independent. A a result, the protocols of the internet suite are widely used to connect heterogeneou systems.

Each layer of the TCP/IP protocol suite has its associated component protocols, th most important of which are as given ahead.

(a) **Application Layer Protocols :** Responsible for application-level access to TCP/IP networking services. These include Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Telnet, Simple Mail Transfer Protocol (SMTP), and Simple Network Management Protocol (SNMP). In the Microsoft implementation of TCP/IP, application layer protocols interact with transport layer protocols by using either Windows Sockets or Net BIOS over TCP/IP (Net BT).

(b) **Transport Layer Protocols :** Establish communication through connection-oriented sessions and connectionless broadcasts. Protocols at this layer include Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

(c) **Internet Layer Protocols :** Responsible for routing and encapsulation into IP packets. Protocols at this layer include Internet Protocol (IP), Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), and Internet Group Management Protocol (IGMP).

(d) **Network Layer Protocols :** It places frames on the network. The protocols include the various local area network (LAN) architectures (such as Ethernet and Token Ring) and WAN telecommunication service technologies-such as Plain Old Telephone Service (POTS), Integrated Services Digital Network (ISDN) and Asynchronous Transfer Mode (ATM).

3.3. Addressing

Four levels of addresses are used in an internet employing the TCP/IP protocols : **physical** (link) **addresses, logical** (IP) **addresses, port addresses,** and specific addresses.

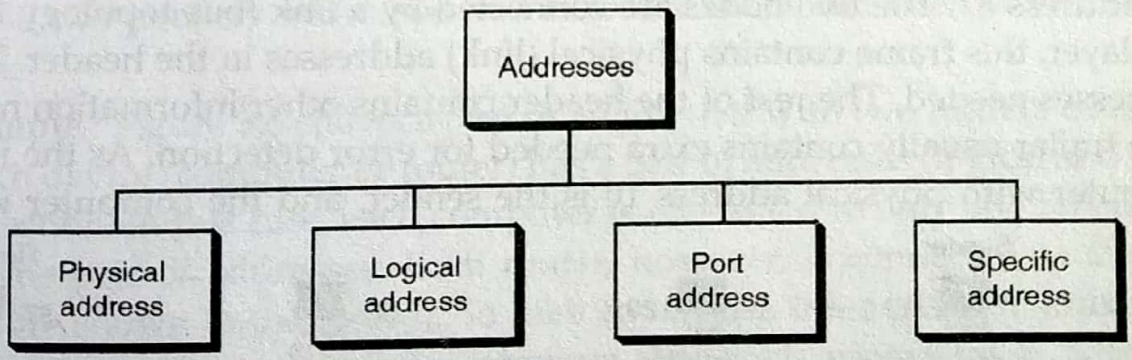


Fig. 3.3. Addresses in TCP/IP

Each address is related to a specific layer in the TCP/IP architecture, as shown in Fig. 3.4.

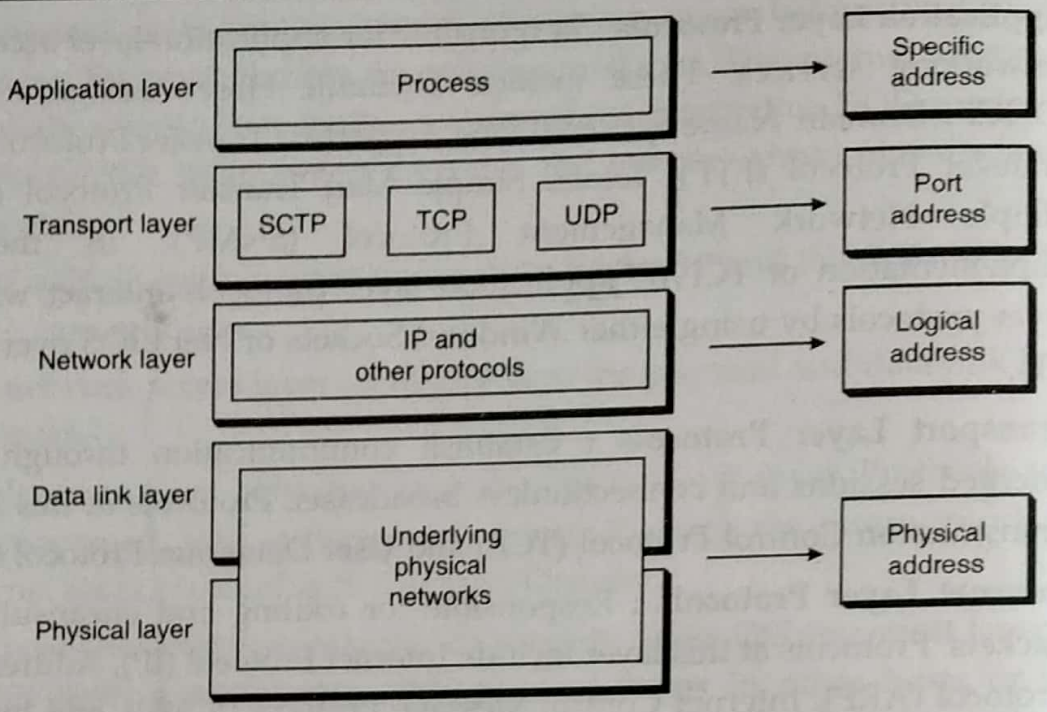


Fig. 3.4. Relationship of Layers and Addresses in TCP/IP

3.3.1. Physical Addresses

The physical address, also known as the link address, is the address of a node as defined by its LAN or WAN. It is included in the frame used by the data link layer. It is the lowest-level address.

The physical addresses have authority over the network (LAN or WAN). The size and format of these addresses vary depending on the network. For example, Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC). Local Talk (Apple), however, has a 1-byte dynamic address that changes each time the station comes up.

Example : In Figure a node with physical address 10 sends a frame to a node with physical address 87, The two nodes are connected by a link (bus topology LAN). At the data link layer, this frame contains physical (link) addresses in the header. These are the only addresses needed. The rest of the header contains other information needed at this level. The trailer usually contains extra needed for error detection. As the figure shows The computer with physical address 10 is the sender, and the computer with physical

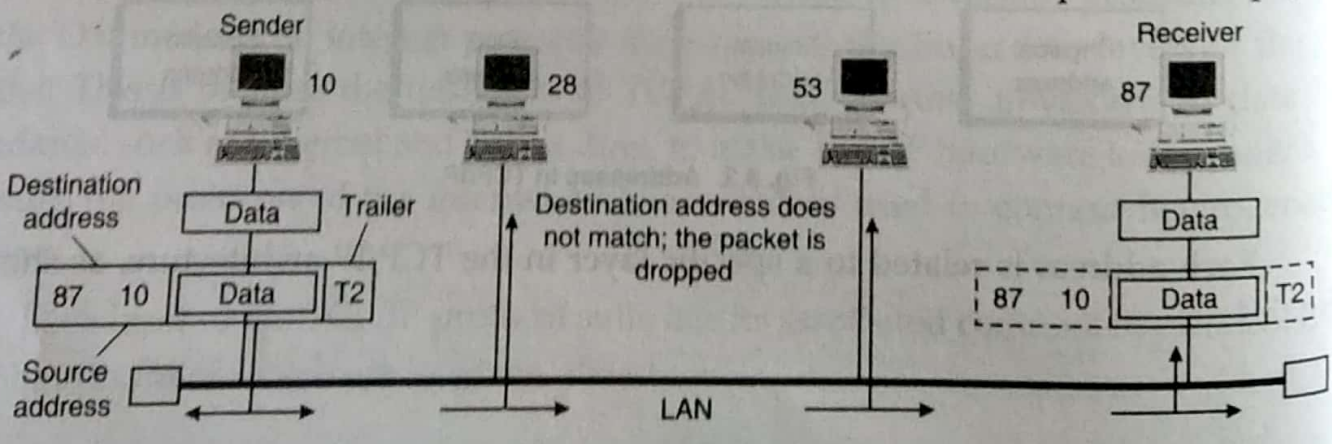


Fig. 3.5. Physical Addresses

address 87 is the receiver. The data link layer at the sender receives data from an upper layer. It encapsulates the data in a frame, adding a header and a trailer. The header, among other pieces of information, carries the receiver and the sender physical (link) addresses. Note that in most data link protocols, the destination address, 87 in this case, comes before the source address (10 in this case).

We have shown a bus topology for an isolated LAN. In a bus topology, the frame is propagated in both directions (left and right). The frame propagated to the left dies when it reaches the end of the cable if the cable end is terminated appropriately. The frame propagated to the right is sent to every station on the network. Each stations with physical addresses other than 87 drops the frame because the destination address in the frame does not match its own physical address. The intended destination computer, however, finds a match between the destination address in the frame and its own physical address. The frame is checked, the header and trailer are dropped, and the data part is decapsulated and delivered the the upper layer.

Example : Most local-area networks use a 48-bit (6-byte) physical address written as 12 hexadecimal digits; every byte (2 hexadecimal digits) is separated by a colon, as shown below :

07 : 01 : 02 : 01 : 2C : 4B

A 6-byte (12 hexadecimal digits) physical address

3.3.2. Logical Addresses

Logical addresses are necessary for universal communications that are independent of underlying physical networks. Physical addresses are not adequate in an internetwork environment where different networks can have different address formats. A universal addressing system is needed in which each host can be identified uniquely, regardless of the underlying physical network.

The logical addresses are designed for this purpose. A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the internet. No two publicly addressed and visible hosts on the Internet can have the same IP address.

Example : Figure 3.6 shows a part of an internet with two routers connecting three LANs. Each device (computer or router) has a pair of addresses (logical and physical) for each connection. In this case, each computer is connected to only one link and therefore has only one pair of addresses. Each router, however, is connected to three networks (only two are shown in the figure). So each router has three pairs of addresses, one for each connection. Although it may obvious that each router must have a separate physical address for each connection, it may not be obvious why it needs a logical address for each connection.

The computer with logical address A and physical address 10 needs to send a packet to the computer with logical address P and physical address 95. We use letters to show the logical addresses and numbers for physical addresses, but note that both are actually numbers.

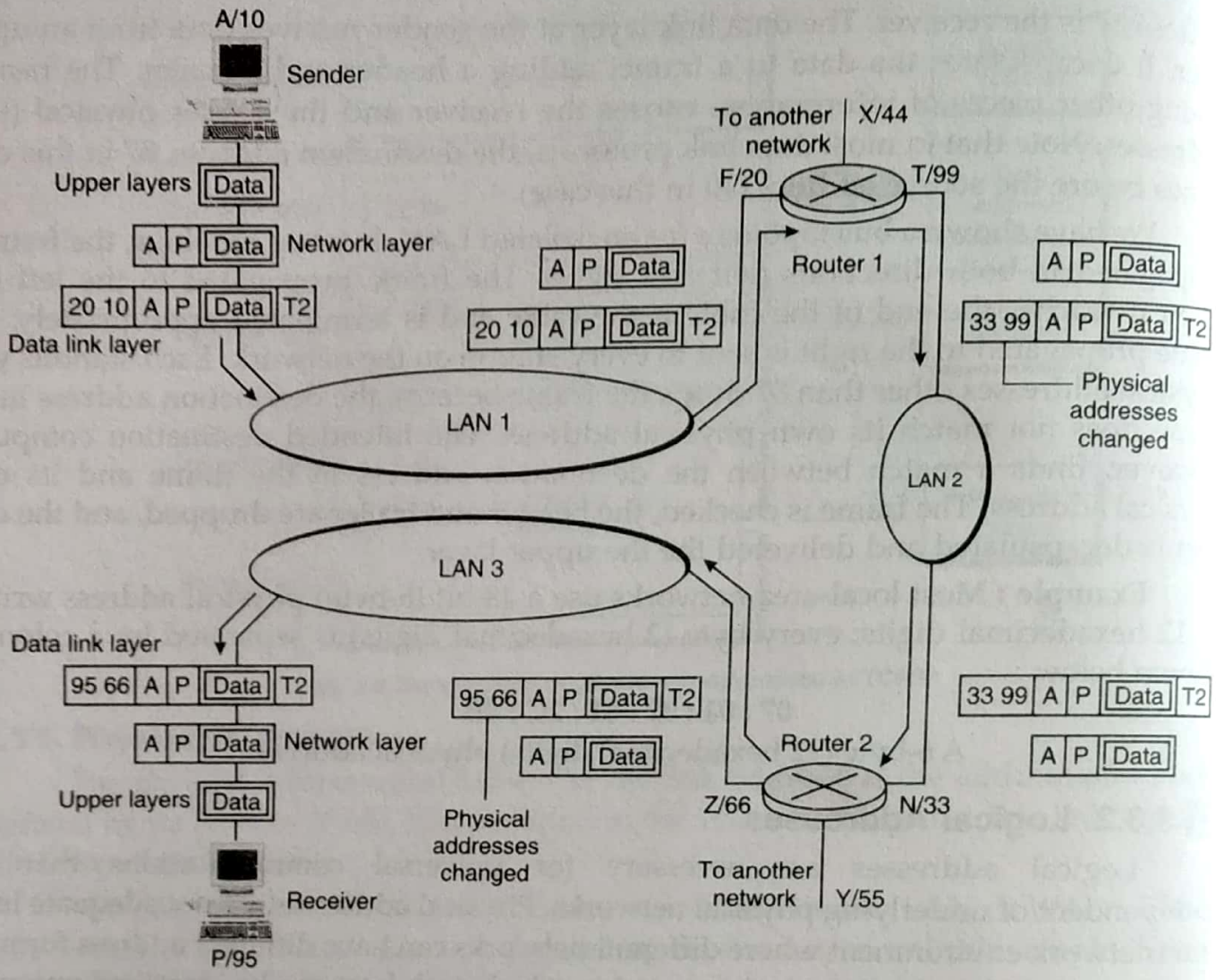


Fig. 3.6. IP Addresses

The sender encapsulates its data in a packet at the network layer and adds two logical addresses (A and P). Note that in most protocols, the logical source address comes before the logical destination address (contrary to the order of physical addresses). The network layer, however, needs to find the physical address of the next hop before the packet can be delivered. The network layer consults its routing table and finds the logical address of the next hop (router 1) to be F. The ARP discussed previously finds the physical address of router 1 that corresponds to the logical address of 20. Now the network layer passes this address to the data link layer, which in turn, encapsulates the packet with physical destination address 20 and physical source address 10.

The frame is received by every device on LAN 1, but is discarded by all except router 1, which finds that the destination physical address in the frame matches with its own physical address. The router decapsulates the packet from the frame to read the logical destination address P. Since the logical destination address does not match the router's logical address, the router knows that the packet needs to be forwarded. The router consults its routing table and ARP to find the physical destination address of the

next hop (router 2), creates a new frame, encapsulates the packet, and sends it to router 2.

Note the physical addresses in the frame. The source physical address changes from 10 to 99. The destination physical address changes from 20 (router 1 physical address) to 33 (router 2 physical address). The logical source and destination addresses must remain the same; otherwise the packet will be lost.

At router 2 we have a similar scenario. The physical addresses are changed and a new frame is sent to the destination computer. When the frame reaches the destination, The packet is decapsulated. The destination logical address P matches the logical address of the computer. The data are decapsulated from the packet and delivered to the upper layer. Note that although physical addresses will change from hop to hop, logical addresses remain the same from the source to destination.

3.3.3. Port Addresses

The IP address and the physical address are necessary for a quantity of data to travel from a source to the destination host. However, arrival at the destination host is not the final objective of data communications on the internet. A system that sends nothing but data from one computer to another is not complete. Today, computers are devices that can run multiple processes at the same time. The end objective of internet communication is a process communicating with another process. For example, computer A can communicate with computer C by using TELNET. At the same time, computer A communicates with computer B by using the File Transfer Protocol (FTP). For these processes to receive data simultaneously, we need a method to label the different processes. In other words, they need addresses. In the TCP/IP architecture, the label assigned to a process is called a port address. A port address in TCP/IP is 16 bits in length.

Example : Figure 3.7 shows two computers communicating via the internet. The sending computer is running three processes at this time with port addresses a, b and c. The receiving computer is running two processes at this time with port addresses j and k. Process a in the sending computer needs to communicate with process j in the receiving computer. Note that although both computers are using the same application, FTP, for example, the port addresses are different because one is a client program and the other is a server program. To show that data from process a need to be delivered to process j, and not k, the transport layer encapsulates data from the application layer in a packet and adds two port addresses (a and j), source and destination. The packet from the transport layer is then encapsulated in another packet at the network layer with logical source and destination addresses (A and P). Finally, this packet is encapsulated in a frame with the physical source and destination addresses of the next hop. We have not shown the physical addresses because they change from hop to hop inside the cloud designated as the internet. Note that although physical addresses change from hop to hop, logical and port addresses remain the same from the source to destination.

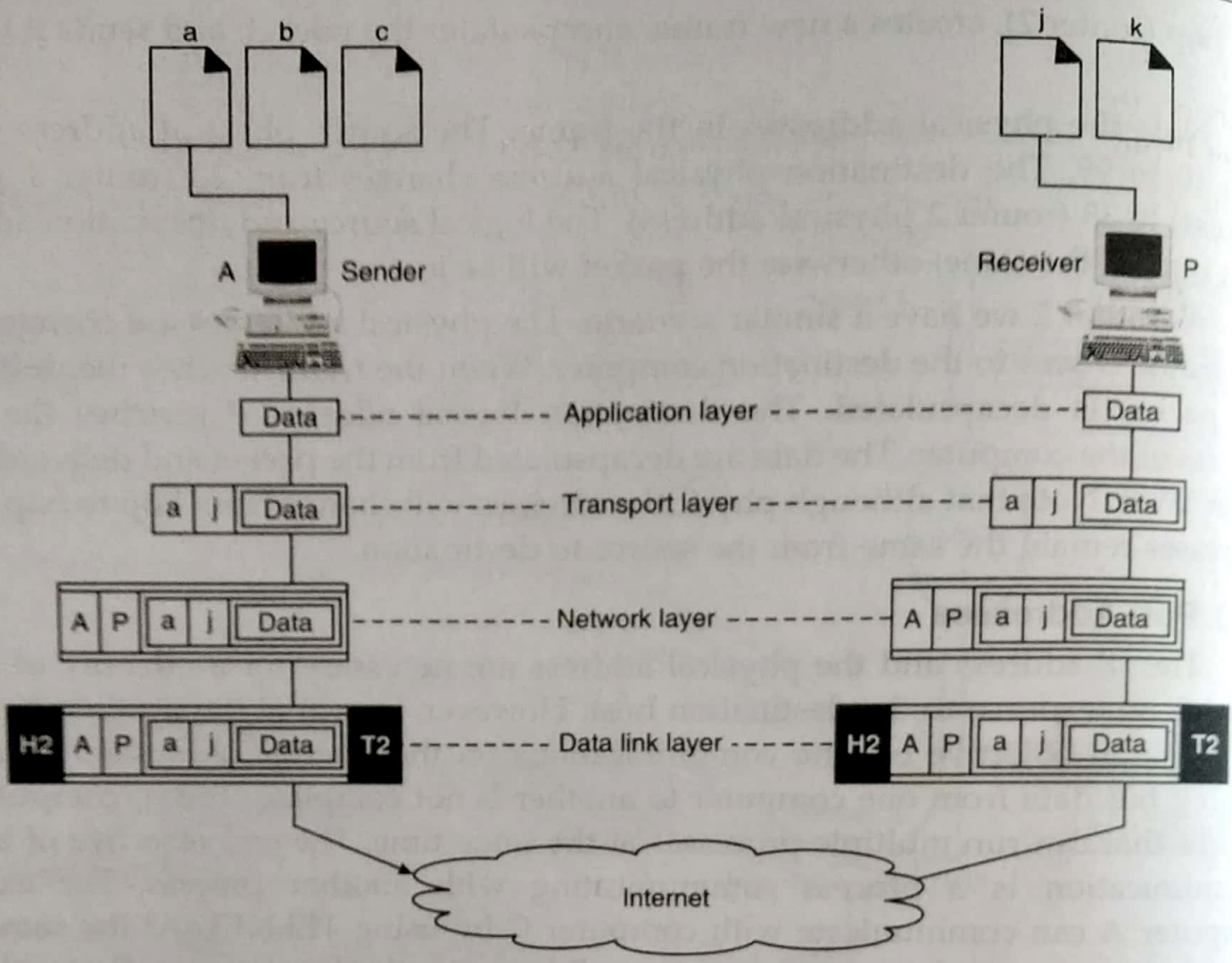


Fig. 3.7. Port Addresses

Example : A port address is a 16-bit address represented by one decimal numbe as shown.

753

A 16-bit port address represented as one single number.

3.3.4. Specific Addresses

Some applications have user-friendly addresses that are designed for that specifi address. Examples include the e-mail address (for example, ankit. verma aquarius@gmail.com) and the Universal Resource Locator (URL) (for example https://www.facebook.com/groups/smartmates/). The first defines the recipient of an e-mail; the second is used to find a document on the World Wide Web. These addresses however, get changed to the corresponding port and logical addresses by the sending computer.

3.4. IP Address

Every machine on the internet has a unique number assigned to it, called an IP address. Without a unique IP address on your machine, you will not be able to communicate with other devices, users, and computers on the internet. You can look at your IP address as if it were a telephone number, each one being unique and used to identify a way to reach you and only you.

IPv4 and IPv6 Addresses

There are two flavors of IP Addresses that can be used on a network. The first, and the version that the internet and most routers are currently configured for, is IPv4 or Internet Protocol version 4. This version uses 32-bit addresses, which limits the amount of addresses to 4, 294, 967, 296 possible unique addresses. Some of these addresses, about 290 million, are also reserved for special purposes. Due to the popular growth of the internet there has been concern that the pool of possible addresses would be exhausted in the near future. With this in mind, a new version of IP addresses was developed called IPv6, or Internet Protocol version 6, that would change the address size from 32-bit address to 128-bit addresses. This change would allow for generous IP address allocations to networks without any foreseeable problem with the amount of addresses available. In order to use IPv6 addresses, though, existing routers and hardware would need to be upgraded or configured to use this new version of IP addresses.

3.4.1. MAC Address v/s IP Address

A standalone computer (a computer that is not attached to a network) lives in its own world and carries out its tasks with its own inbuilt resources. But as soon as it becomes a workstation, it needs an interface to help establish a connection with the network because without this, hte workstations will not be able to share network resources.

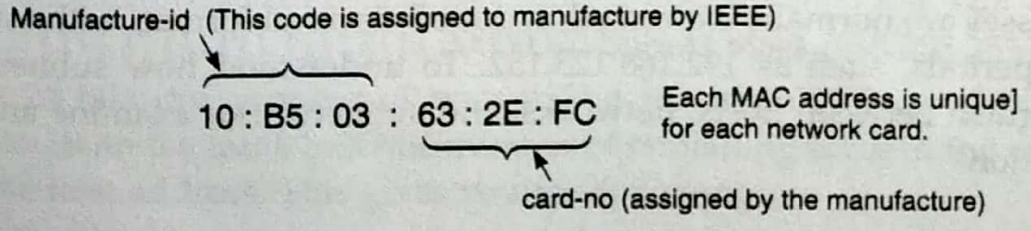
The network-interface-unit is a device (the network card) that is attached to each of the workstations and the server, and helps the workstation establish the all-important connection with the network. Each network-interface-unit that is attached to a workstation has a unique number identifying it which is known as the node address. The NIU is also called Terminal Access point (TAP). Different manufactures have different names for the interface. The NIU is also called NIC-Network Interface Card.

The NIC manufacture assigns a unique physical address to each NIC card ; this physical address is known as Media Access Control address (MAC address).

A MAC address is a 6-byte address with each byte separated by a colon e.g., a sample MAC address could be :

10 Z: B5 : 03 : 63 : 2E : FC

So, now you know that this MAC address is actually the numbe assigned to the network card of your computer. The first three bytes of MAC address are the manufacturer-id (assigned to the manufacture by an international organization namely IEEE) and the last three bytes are the card-no (assigned by manufacture)



All networks follow some agreed upon set of rules for communication. For example, when you speak to one another, you follow one rule which is "when one person is speaking the other would listen". Similarly, computers on a network also follow some set of rules for communicating with one another. These set of rules are called **protocols**. There are many networking protocols. One of the most common networking protocols is TCP/IP protocol. Today's most commonly known network, the internet, also follows this protocol. A network that follows TCP/IP protocol, can also be termed as TCP/IP network.

Each network device (a computer or any other network device) on a TCP/IP network needs to have a unique address on the network. This unique address on a TCP/IP network is the IP Address. IP addresses are needed so that different networks can communicate with each other.

IP addresses can be thought of as a unique series of numbers, uniquely identifying a computer on a network. Thus, you can say that just like, telephones are uniquely identified through their telephone-numbers, computers on a TCP/IP network (such as internet) are uniquely identified through their unique addresses—IP addresses.

Each IP address is actually a series. containing four numbers separated by dots or periods e.g., 192.168.1.1 is an IP address. Similarly 10.217.1.1 is also an IP address and so on.

IP addresses are normally written in dotted decimal form as listed above but computer internally convert them into binary from. For instance,

An IP address in dotted decimal form : 216. 27. 61. 137

Same IP address in binary form : 11011000 . 00011011. 00111101. 10001001

3.5. More on IP Address

The sucess of TCP/IP as the network protocol of the internet is largely because of its ability to connect together networks of different sizes and systems of different types. These networks are arbitrarily defined into three main classes (along with a few others) that have predefined sizes, each of which can be divided into smaller subnetworks by system administrators. A subnet mask is used to divide an IP address into two parts. One part identifies the host (computer), the other part identifies the network to which it belongs. To better understand how Ip addresses and subnet masks work, look at an IP (Internet Protocol) address and see how it is organized.

3.5.1. IP Addresses : Networks and Hosts

An IP address is a 32-bit number that uniquely identifies a host (computer or other device, such as a printer or router) on a TCP/IP network.

IP addresses are normally expressed in dotted-decimal format, with four numbers separated by periods, such as 192.168.123.132. To understand how subnet masks are used to distinguish between hosts, networks, and subnetworks, examine an IP address in binary notation.

For example, the dotted-decimal IP address 192.168.123.132 is (in binary notation) the 32 bit number 11000000010100011110111000100. This number may be hard to make sense of, so divide it into four parts of eight binary digits.

Thess eight bit sections are known as octets. The example IP address, then becomes 11000000.10101000.01111011.10000100. This number only makes a little more sense, so for most uses, convert the binary address into dotted-decimal format (192.168.123.132). the decimal numbers separated by periods are the octets converted from binary to decimal notation.

For a TCP/IP wide area network (WAN) to work efficiently as a collection of networks, the routers that pass packets of data between networks do not know the exact location of a host for which a packet of information is destined. Routers only know what network the host is a member of and use information stored in their route table to determine how to get the packet to the destination host's network. After the packet is delivered to the destination's network, the packet is delivered to the appropriate host.

For this process to work, an IP address has two parts. The first part of an IP address is used as a network address, the last part as a host address. If you take the example 192.168.123.132 and divide it into these two parts you get the following :

192.168.123. Network
132 Host

-or-

192.168. 123.0-network address.
0.0.0.132 -host address

3.5.2. Subnet mask

The second item, which is required for TCP/IP to work, is the subnet mask. The subnet mask is used by the TCP/IP protocol to determine whether a host is on the local subnet or on a remote network.

In TCP/IP, the parts of the IP address that are used as the network and host addresses are not fixed, so the network and host addresses above cannot be determined unless you have more information. This information is supplied in another 32-bit number called a subnet mask. In this example, the subnet mask is 255. 255. 255. 0. It is not obvious what this number means unless you know that 255 in binary notation equals 11111111; so, the subnet mask is :

11111111.11111111.11111111.0000000

Lining up the IP address and the subnet mask together, the network, and host portions of the address can be separated :

11000000.10101000.01111011.10000100-IP address (192. 168. 123. 132)
11111111.11111111.11111111.00000000-Subnet mask (255.255.255.0)

The first 24 bits (the number of ones in the subnet mask) are identified as the network address, with the last 8 bits (the number of remaining zeros in the subnet mask) identified as the host address. This gives you the following :

11000000.10101000.01111011.00000000-Network address (192. 168. 123. 0)
00000000.00000000.00000000.10000100-Host address (000. 000. 000. 132)

So now you know, for this example using a 255.255.255.0 subnet mask, that the network ID is 192.168.123.0, and the host address is 0.0.0.132. When a packet arrives on the 192.168.123.0 subnet (from the local subnet or a remote network), and it has a destination address of 192.168.123.132, your computer will receive it from the network and process it.

Almost all decimal subnet masks convert to binary numbers that are all ones on the left and all zeros on the right. Some other common subnet masks are :

Decimal	Binary
255.255.255.192	1111111.11111111.1111111.11000000
255.255.255.224	111111.11111111.1111111.11100000

3.5.3. IP Address Classes

IP addresses are organized into classes. For convenience of humans, IP addresses are expressed in the decimal format. Every number in each class in represented as binary to computers.

The four numbers in an IP address are known as 'octets' Each of them has eight bit positions. The octets are divided into two sections : Net and Host. The first octet represents Net for identifying the network and the host contains the last octet. There are five IP classes.

These classes are A, B, C, D, E and their possible ranges can be seen in the following figure :

Class	Start address	Finish address
A	0.0.0.0	126.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255.
D	224.0.0.0	239.255.255.255
E	240.0.0.0	255.255.255.255

Fig 3.8. IP address classes

If you look at the table you may notice something strange. The range of IP address from Class A to Class B skips the 127.0.0.0-127.255.255.255 range. That is because this range is reserved for the special addresses called loopback addresses that have already been discussed above.

The rest of classes are allocated to companies and organizations based upon the amount of IP addresses that they may need. Listed below are descriptions of the IP classes and the organizations that will typically receive that type of allocation.

Default Network : The special network 0.0.0.0 is generally used for routing.

Class A : From the table above you see that there are 126 class A networks. These networks consist of 16, 777, 214 possible IP addresses that can be assigned to devices and computers. This type of allocation is generally given to very large networks such as multi-national companies.

Loopback : This is the special 127.0.0.0 network that is reserved as a loopback to your own computer. These addresses are used for testing and debugging of your programs or hardware.

Class B : This class consists of 16,384 individual networks, each allocation consisting of 65,534 possible IP addresses. These blocks are generally allocated to internet service providers and large networks, like a college or major hospital.

Class C : There is a total of 2,097, 152 Class C networks available, with each network consisting of 255 individual IP addresses. This type of class is generally given to small to mid-sized companies.

Class D : The IP addressses in this class are reserved for a service called multicast.

Class E : The IP addresses in this class are reserved for experimental use.

Broadcast : This is the special network of 255.255.255.255, and is used for broadcasting messages to the entire network that your computer resides on.

3.5.4. Private Addresses

There are also blocks of IP addresses that are set aside for internal private use for computers not directly connected to the Internet. These IP addesses are not supposed to be routed through the Internet, and most service providers will block the attempt to do so. These IP addresses are used for internal use by company or home networks that need to use TCP/IP but do not want to be directly visible on the internet. These IP ranges are :

Class	Private Start Address	Private End Address
A	10.0.0.0	10.255.255.255
B	172.16.0.0	172.31.255.255
C	192.168.0.0	192.168.255.255

If you are on a home/office private network and want to use TCP/IP, you should assign your computers/devices IP addresses from one of these three ranges. That way your router/firewall would be the only device with a true IP address which makes your network more secure.

3.5.5. Common Problems and Resolutions

The most common problem people have is by accident assigning an IP address to a device on your network that is already assigned to another device. When this happens, the other computers will not know which device should get the information and you can experience erratic behavior. On most operating systems and devices, if there are two devices on the local network that have the same IP address, it will generally give you a "IP confilict" warnig. If you see this warning, that means that the device giving the warning, detected another device on the network using the same address.

The best solution to avoid a problem like this is to use a service called DHCP that almost all home routers provide. DHCP, or Dynamic Host Configuration Protocol, is a service that assigns addresses to devices and computers, You tell the DHCP server what range of IP addresses you would like it to assign, and then the DHCP server takes the responsibility of assigning those IP addresses to the various devices and keeping track so those IP addresses are assigned only once.

3.6. Special IP Addresses

There are several IP addresses that are special in one way or another. These addresses are for special purposes or are to be put to special use.

- Addresses significant to every IP subnet
  - Network Address
  - Broadcast Address
- Addresses significant to individual hosts
  - Loopback Address
- Special Addresses of Global Significance
  - Private Addresses
  - Reserved Addresses

3.6.1. IP-Subnets

Network Address : A network address is an address where all host bits in the IP address are set to zero (0). In every subnet there is a network address. This is the first and lowest numbered address in the range because the address is always the address where all host bits are set to zero. The network address is defined in the RFC's as the address that contains all zeroes in the host portion of the address and is used to communicate with devices that maintain the network equipment.

Today it is rare to see the network address in use.

Broadcast Address : A broadcast address is an address where all host bits in the IP address are set to one (1). This address is the last address in the range of addresses, and is the address whose host portion is set to all ones. All hosts are to accept and respond to the broadcast address. This makes special services possible.

3.6.2. Hosts

Loopback address (127.0.0.1) : The 127.0.0.0 class 'A' subnet is used for special local addresses, most commonly the loopback address 127.0.0.1. This address is used to test the local network interface device's functionality. All network interface devices should respond to this address from the command line of the local host. If you ping 127.0.0.1 from the local host, you can be assured that the network hardware is functioning and that the network software is also functioning. The addresses in the 127.0.0.0-127.255.255.255 range cannot be reached from outside the host, and so cannot be used to build a LAN.

3.6.3. Private IP Addresses

RFC 1918 defines a number of IP blocks which were set aside by the American registry of Internet Numbers (ARIN) for use as private addresses on private networks that are not directly connected to the internet. The private addresses are :

Class	Start	End
A	10.0.0.0	10.255.255.255
B	172.16.0.0	172.31.255.255
C	192.168.0.0	192.168.255.255

Multicast IP Addresses

There are a number of addresses that are set aside for special purposes, such as the IP's used in OSPF, Multicast, and experimental purposes that connot be used on the internet.

Class	Start	End
D	224.0.0.0	239.255.255.255

Special Use Addresses-Table From RFC 3330

Address Block	CIDR Mask	Used for	Reference
0.0.0.0	/8	Used to communicate with "this" network	RFC1700, p.4
10.0.0.0	/8	Private-Use Networks	RFC 1918
14.0.0.0	/8	Public-Data Network	RFC1700, p.181
24.0.0.0	/8	Cable TV Networks	—
39.0.0.0	/8	Previously Reserved Available for Regional Allocation	RFC1797
127.0.0.0	/8	Loopback address	RFC1700, p.5
128.0.0.0	/16	Previously Reserved Available for Regional Allocation	—
169.254.0.0	/16	Link local (e.g. Microsoft XP systems use Automatic Private IP Addressing (APIPA) which selects addresses in this range.)	
172.16.0.0	/12		
191.255.0.0	/16		
192.0.0.0	/16		
192.88.99.0	/16		
192.168.0.0	/16		

3.7. Subnetting

A subnet describes a set of networked computers which have common IP address routing prefix. Breaking the networking into smaller and more efficient subnets is known as subnets. Subnetting prevents Ethernet packet collision which has excessive rates in a large network. Routers are used to manage the traffic and constitute borders among subnets.

A Class A, B, or C TCP/IP network can be further divided, or subnetted, by a system administrator. This becomes necessary as you reconcile the logical address scheme of the internet (the abstract world of IP addresses and subnets) with the physical networks in use by the real world.

A system administrator who is allocated a block of IP addresses may be administering networks that are not organized in a way that easily fits these addresses. For example, you have a wide area network with 150 hosts on three networks (in different cities) that are connected by a TCP/IP router. Each of these three networks has 50 hosts. You are allocated the class C network 192.168.123.0 (For illustration, this address is actually from a range that is not allocated on the internet). This means that you can use the addresses 192.168.123.1 to 192.168.123.254 for your 150 hosts.

Two addresses that cannot be used in your example are 192.168.123.0 and 192.168.123.255 because binary addresses with a host portion of all ones and all zeros are invalid. The zero address is invalid because it is used to specify a network without specifying a host. The 255 address (in binary notation, a host address of all ones) is used to broadcast a message to every host on a network. Just remember that the first and last address in any network or subnet cannot be assigned to any individual host.

You should now be able to give IP addresses to 254 hosts. This works fine if all 150 computers are on a single network. However, your 150 computers are on three separate physical networks. Instead of requesting more address blocks for each network, you divide your network into subnets that enable you to use one block of addresses on multiple physical networks.

In this case, you divide your network into four subnets by using a subnet mask that makes the network address larger and the possible range of host addresses smaller. In other words, you are 'borrowing' some of the bits usually used for the host address and using them for the network portion of the address. The subnet mask 255.255.255.192 gives you four networks of 62 hosts each. This works because in binary notation, 255.255.255.192 is the same as

11111111.11111111.11111111.11000000. The first two digits of the last octet become network addresses, so you get the additional networks 00000000 (0), 01000000 (64), 10000000 (128) and 11000000 (192). In these four networks, the last 6 binary digits can be used for host addresses.

Using a subnet mask of 255.255.255.192, your 192.168.123.0 network then becomes the four networks 192.168.123.0, 192.168.123.64, 192.168.123.128 and 192.168.123.192. These four networks would have as valid host addresses :

192.168.123.1-62
192.168.123.65-126
192.168.123.129-190
192.168.123.193-254

Remember, again, that binary host addresses with all ones or all zeros are invalid, so you cannot use addresses with the last octet of 0, 63, 64, 127, 128, 191, 192, or 255.

You can see how this works by looking at two host addresses, 192.168.123.71 and 192.168.123.133. If you used the default Class C subnet mask of 255.255.255.0, both addresses are on the 192.168.123.0 network. However, if you use the subnet mask of 255.255.255.192, they are on different networks ; 192.168.123.71 is on the 192.168.123.64 network, 192.168.123.133 is on the 192.168.123.128 network.

3.7.1. Default Gateways

If a TCP/IP computer needs to communicate with a host on another network, it will usually communicate through a device called a router. In TCP/IP terms, a router that is specified on a host, which links the host's subnet to other networks, is called a default gateway. This section explains how TCP/IP determines whether or not to send packets to its default gateway to reach another computer or device on the network.

When a host attempts to communicate with another device using TCP/IP, it performs a comparison process using the defined subnet mask and the destination IP address versus the subnet mask and its own IP address. The result of this comparison tells the computer whether the destination is a local host or a remote host.

If the result of this process determines the destination to be a local host, then the computer will simply send the packet on the local subnet. If the result of the comparison determines the destination to be a remote host, then the computer will forward the packet to the default gateway defined in its TCP/IP properties. It is then the responsibility of the router to forward the packet to the correct subnet.

3.8. Supernetting

Supernetting, also called Classless Inter-Domain Routing (CIDR), is a way to aggregate multiple internet addresses of the same class. The original Internet Protocol (IP) defines IP addresses in four major classes of address structure, Classes A through D. Each class allocates one portion of the 32-bit internet address format to a network address and the remaining portion to the specific host machines within the network. Using supernetting, the network address 192.168.2.0/24 and an adjacent address 192.168.3.0/24 can be merged into 192.168.2.0/23. The "23" at the end of the address says that the first 23 bits are the network part of the address, leaving the remaining nine bits

for specific host addresses. Supernetting is most often used to combine Class C network addresses and is the basis for most routing protocols currently used on the internet.

Supernetting was created as a way to solve the problem of routing tables growing beyond the ability of current software and people to manage and to provide a solution to the exhaustion of Class B network address space. Supernetting allows one routing table entry to represent an aggregation of networks much like one area code represents an aggregation of telephone numbers in an area. The Border Gateway Protocol (BGP), the prevailing exterior (interdomain) gateway protocol and the Open Shortest Path First (OSPF) router protocol both support supernetting, but the older exterior or interdomain protocols, the Exterior Gateway Protocol (EGP) and the Routing Information Protocol (RIP) do not support it.

In other way supernetting simplifies network routing decisions and saves storage space on route tables. While supernetting. data bits are borrowed from the network ID and allocated to the host ID. A larger and more complicated network can block other routers from making topological changes, so a supernet improves convergence speed and enables a better and more stable environment.

Supernetting reuires the use of routing protocols that help to support CIDR. The other protocols-Interior Gateway Routing Protocol, Exterior Gateway Protocol and Routing Information Protocol Version 1-do not support the transmission of subnet mask information.

Network identifieres used in the supernet can have any length. This permits the organizations to customize network size based on their requirements. For instance, two blocks of class C can be supernetted for a total of approximately 500 addresses. The route aggregation feature of supernetting can be used to group routing information for multiple networks or hosts into one "summarized" route.

The supernet concept includes some drawbacks, the most notable of which is the complexity of CIDR compared to a classful addressing system and the need for new routing protocols that support the CIDR. The ability to customize the network identifier length also makes it harder for the system administrators to differentiate between a host identifier and a network identifier. In order to solve this issue, a new form of IP address writing called slash, or CIDR, notation was developed.

3.9. Loopback

Loopback, or loop-back, refers to the routing of electronic signals, digital data streams, or flows of items back to their source without intentional processing or modification. This is primarily a means of testing the transmission or transportation infrastructure.

Many example applications exist. It may be a communication channel with only one communication endpoint. Any message transmitted by such a channel is immediately and only received by that same channel. In telecommunications, loopback

devices perform transmission tests of access lines from the serving switching center, which usually does not require the assistance of personnel at the served terminal. Loop around is a method of testing between stations that are not necessarily adjacent, where in two lines are used, with the test being done at one station and the two lines are interconnected at the distant station. A patch cable may also function as loopback, when applied manually or automatically, remotely or locally, facilitating a loop-back test.

Where a system (such as a modem) involves round-trip analog-to-digital processing, a distinction is made between analog loopback, where the analog signal is looped back directly, and digital loopback, where the signal is processed in the digital domain before being re-converted to an analog signal and returned to the source.

3.9.1. Telecommunications

In telecommunications, loopback, or a loop, is a hardware or software method which feeds a received signal or data back to the sender. It is used as an aid in debugging physical connection problems. As a test, many data communication devices can be configured to send specific patterns (such as all ones) on an interface and can detect the reception of this signal on the same port. This is called a loopback test and can be performed within a modem or transceiver by connecting its output to its own input. A circuit between two points in different locations may be tested by applying a test signal on the circuit in one location, and having the network device at the other location send a signal back through the circuit. If this device receives its own signal back, this proves that the circuit is functioning.

A hardware loop is a simple device that physically connects the receiver channel to the transmitter channel. In the case of a network termination connector such asX. 21, this is typically done by simply connecting the pins together in the connector. Media such as optical fiber or coaxial cable, which have separate transmit and receive connectors, can simply be looped together with a single strand of the appropriate medium.

A modem can be configured to loop incoming signals from either the remote modem or the local terminal. This is referred to as loopback of software loop.

3.9.2. Serial Interfaces

A serial communications transceiver can use loopback for testing its functionality. For example, a device's transmit pin connected to its receive pin will result in the device receiving exactly what it transmits. Moving this looping connection to the remote end of a cable adds the cable to this test. Moving it to the far end of a modem link extends the test further. This is a common troubleshooting technique and is often combined with a specialized test device that sends specific patterns and counts any errors that come back Some devices include built-in loopback capability.

A simple serial interface loopback test, called paperclip test, is sometimes used to identify serial ports of a computer and verify operation. It utilizes a terminal emulator application to send characters, with flow control set to off, to the serial port and receive

the same back. For this purpose, a paperclip is used to short pin 2 to pin 3 (the receive and transmit pins) on a standard Rs-232 interface using D-subminiature DE-9 or DB-25 connectors.

3.9.3. Virtual Loopback Interface

Implementations of the Internet Protocol Suite include a virtual network interface through which network applications can communicate when executing on the same machine. It is implemented entirely within the operating system's networking software and passes no packets to any network interface controller. Any traffic that a computer program sends to a loopback IP address is simply and immediately passed back up the network software stack as if it had been received from another device.

Unix-like systems usually name this loopback interface lo or lo0.

Various Internet Engineering Tast Force (IETF) standards reserve the IPv4 address block 127.0.0.0/8, in CIDR notation and the IPv6 address : : 1 for this purpose. The most common IPv4 address used is 127.0.0.1. Commonly these loopback addresses are mapped to the hostnames, localhost or loopback.

MPLS

One notable exceptin to the use of the 127.0.0.0/8 network addresses is their use in Multiprotocol Label Switching (MPLS) traceroute error detecation, in which their property of not being routable provides a convenient means to avoid delivery of faulty packets to end users.

Martian Packets

Any IP datagram with a source or destination address set to a loopback address must not appear outside of a computing system, or be routed by any routing device. Packets received on an interface with a loopback destination address must be dropped. Such packets are sometimes referred to as Martian packets.[1] As with other bogus packets, they may be malicious and any problems they might cause can be avoided by applying bog on filtering.

Management Interface

Some computer network equipment use the term "loopback" for a virtual interface used for management purposes. Unlike a proper loopback interface, this type of loopback device is not used to talk with itself.

Such an interface is assigned an address that can be accessed from management equipment over a network but is not assigned to any of the physical interfaces on the device. Such a loopback device is also used for management datagrams, such as alarms, originating from the equipment. The property that makes this virtual interface special is that applications that use it will send or receive traffic using the address assigned to the virtual interface as opposed to the address on the physical interface through which the traffic passes.

Loopback interfaces of this sort are often used in the operation of routing protocols, because they have the useful property that, unlike real physical interfaces, they will not go down when a physical port fails.

3.10. Concept of Loopback Address

IP defines a **loopback address** used to test network applications. Programmers often use **loopback testing** for preliminary debugging after a **network application** has been created. To perform a **loopback test,** a programmer must have two application programs that are intended to communicate across a network. Each application includes the code needed to interact with **TCP/IP protocol** software. Instead of executing each program on a separate computer, the programmer runs both program on a single computer and instructs them to use a **loopback IP address** when communicating.

When one application sends data to another, data travels from the protocol stack to the IP software, which forwards it back up through the protocol stack to the second program. Thus, the programmer can test the porgram logic quickly without needing two computers and without sending packets across a network. IP reserves the network prefix 127/8 for use with loopback. The host address used with 127 is irrelevant that means all host addresses are treated the same. By convention, programmers often use host number 1, making 127.0.0.1 the most popular form of **loopback.** During **loopback testing** no packets ever leave a computer that means the IP software forwards packets from one application program to another. Consequently, the **loopback address** never appears in packet traveling across a network.

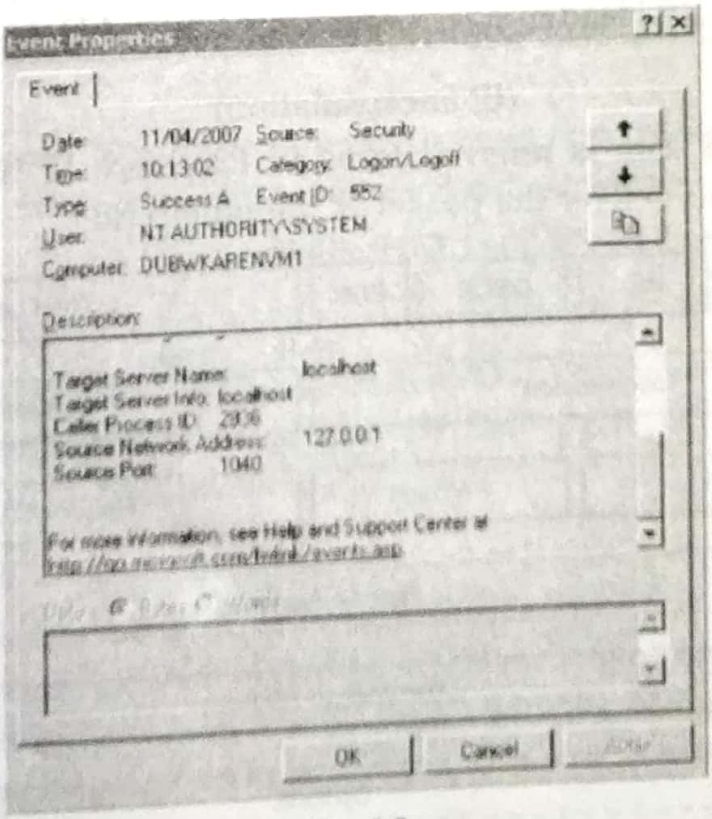


Fig. 3.9

3.11. IP Versions

Two versions of the Internet Protocol (IP) are in use : IP Version 4 and IP Version 6. Each version defines an IP address differently. Because of its prevalence, the generic term IP address typically still refers to the addresses defined by IPv4. The gap in version sequence betwen IPv4 and IPv6 resulted from the assignment of number 5 to the experimental Internet Stream Protocol in 1979, which however was never referred to as IPv5.

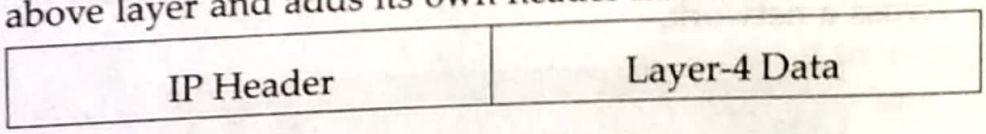
3.11.1. IPv4

Internet Protocol version 4 (IPv4) is the fourth version in the development of the Internet Protocol (IP) Internet, and routes most traffic on the Internet. However, a successor portocol, IPv6, has been defined and is in various stages of production deployment.

IPv4 is a connectionless protocol for use on packet-switched networks. It operates on a best effort delivery model, in that it does not guarantee delivery, nor does it assure proper sequencing or avoidance of duplicate delivery. These aspects, including data integrity, are addressed by an upper layer transport portocol, such as the Transmission Control Protocol (TCP).

3.11.2. Packet Format

Internet Protocol being a layer-3 protocol (OSI) takes data segments form layer-4 (Transport) and divides it into what's called packet. IP packet encapsulates data unit received from above layer and adds its own header information.



(IP Encapsulation)

The encapsulated data is referred to as IP Palyload. IP header contains all the necessary information to deliver the packet at the other end.

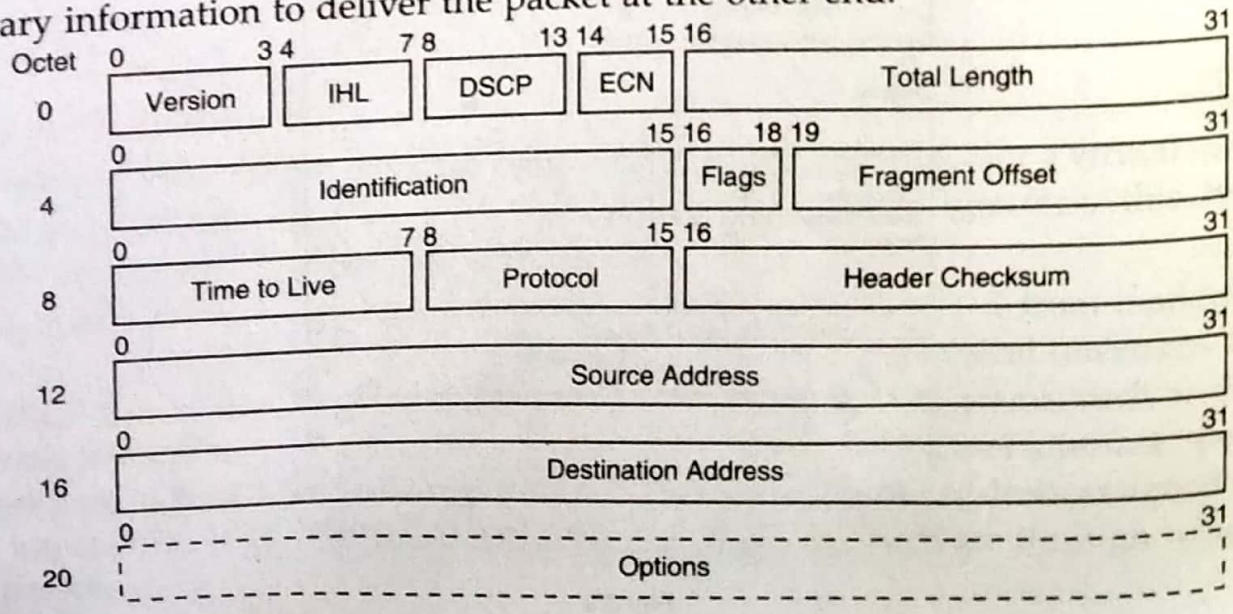


Fig. 3.10. IP Header

IP header includes many relevant information including Version Number, which, in this context, is 4. Other details are as follows :

Version : Version no. of Internet Protocol used (e.g. IPv4).

IHL : Internet Header Length, Length of entire IP header.

DSCP : Differentiated Services Code Point, this is Type of Service.

ECN : Explicit Congestion Notification, carries information about the congestion seen in the route.

Total Length : Length of entire IP Packet (including IP header and IP Payload)

Identification : If IP packet is fragmented during the transmission, all the fragments contain same identification no. to identify original IP packet they belong to.

Flags : As required by the network resources, if IP Packet is too large to handle these 'flags' tell that if they can be fragmented or not. In this 3-bit flag, the MSB is always set to '0'.

Fragment Offset : This offset tells the exact position of the fragment in the original IP Packet.

Time to Live : To avoid looping in the network, every packet is sent with some TTL value set, which tells the network how many routers (hops) this packet cna cross. At each hop, its value is decremented by one and when the value reaches zero, the packet is discarded.

Protocol : Tells the network layer at the destination host, to which Protocol this packet belongs to, *i.e.* the next level Protocol. For example protocol number of ICMP is TCP is 6 and UDP is 17.

Header Checksum : This field is used to keep checksum value of entire header which is then used to check if the packet is received error-free.

Source Address : 32-bit address of the sender (or source) of the packet.

Destination Address : 32-bit address of the Receiver (or destination) of the packet.

Options : This is optional field, which is used if the value of IHL is greater than 5. These options may contain values for options such as security, Record Route, Time Stamp etc.

3.11.3. IPv4-Addressing

IPv4 supports three different type of adressing modes :

Unicast Addressing Mode : In this mode, data is sent only to one destined host. The Destination Address field contains 32-bit IP address of the destination host. Here client sends data to the targeted server :

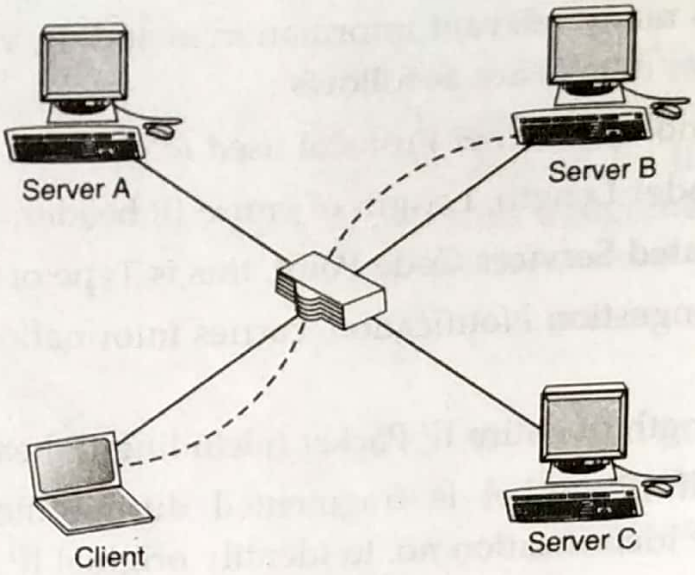


Fig.3.11

Broadcast Addressing Mode : In this mode the packet is addressed to all hosts in a network segment. The Destination Address field contains special broadcast address *i.e.* **255.255.255.255.** When a host sees this packet on the network, it is bound to process it. Here client sends packet, which is entertained by all the Servers.

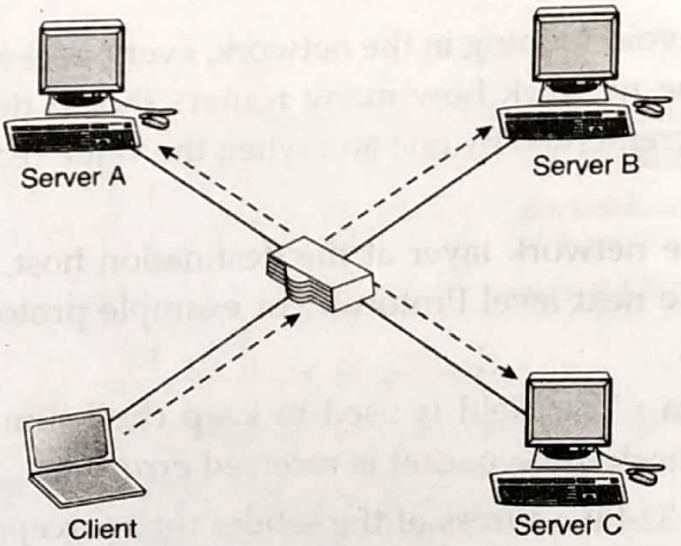


Fig.3.12

Multicast Addressing Mode : This mode is a mix of previous two modes, *i.e.* the packet sent is neither destined to a single host nor all the host on the segment. In this packet, the Destination Address contains special address which starts with 224.x.x.x and can be entertained by more than one host.

Here a server sends packets which are entertained by more than one servers. Every network has one IP address reserved for network number which represents the network and one IP address reserved for Broadcast address, which represents all the host in that network.

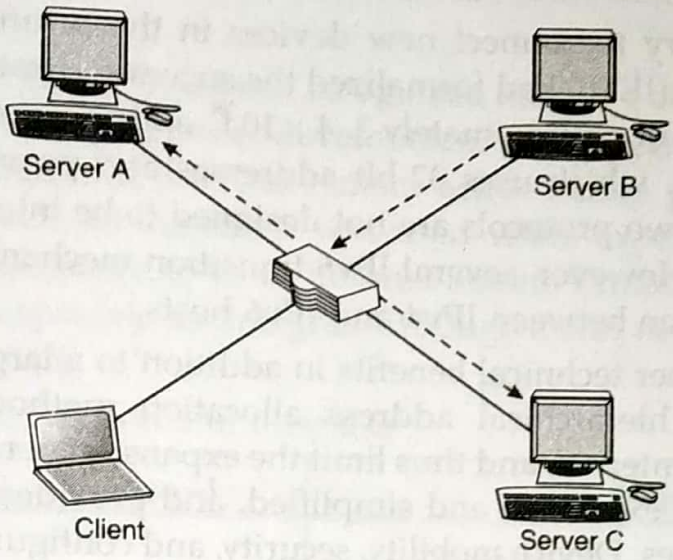


Fig.3.13

Hierarchical Addressing Scheme

IPv4 uses hierarchical addressing scheme. An IP address which is 32-bits in length, is divided into two or three parts as depicted :

8 bits	8 bits	8 bits	8 bits
Network	Network	Sub-Network	Host

A single IP address can contain information about the network and its sub-network and ultimately the host. This scheme enables IP address to be hierarchical where a network can have many sub-networks which in turn can have many hosts.

3.11.4. IPv6

Internet Protocol version 6, is a new addressing protocol designed to incorporate whole sort of requirement of future internet known to us as internet version 2. This protocol as its predecessor IPv4, works on network layer (layer-3). Along with its offering of enormous amount of logical address space, this protocol has ample of features which addresses today's shortcoming of IPv4.

Internet Protocol version 6 (IPv6) is the latest version of the Internet Protocol (IP) the communications protocol that provides an identification and location system for computers on networks and routes traffic across the internet. IPv6 was developed by the internet Engineering Task Force (IETF) to deal with the long-anticipated problem of IPv4 address exhaustion.

IPv6 is intended to replace IPv4, which still carries more than 96% of internet traffic worldwide as of May 2014. As of June 2014, the percentage of users reaching Google services with IPv6 surpassed 4% for the first time.

Every device on the internet is assigned an IP address for identification and location definition. With the rapid growth of the Internet after commercialization in the 1990s, it became evident that far more addresses than the IPv4 address space has

available were necessary to connect new devices in the future. By 1998, the Internet Engineering Task Force (IETF) had formalized the successor protocol. IPv6 uses a 128-bit address, allowing 2^{128}, or approximately $3 \cdot 4 \times 10^{38}$ addresses, or more than $7 \cdot 9 \times 10^{28}$ times as many as IPv4, which uses 32-bit addresses and provides approximately 4·3 billion addresses. The two protocols are not designed to be interoperable, complicating the transition to IPv6. However, several IPv6 transition mechanisms have been devised to permit communication between IPv4 and IPv6 hosts.

IPv6 provides other technical benefits in addition to a larger addressing space. In particular, it permits hierarchical address allocation methods that facilitate route aggregation across the internet, and thus limit the expansion of routing tables. The use of multicast addressing is expanded and simplified, and provides additonal optimization for the delivey of services. Device mobility, security, and configuration aspects have been considered in the design of the protocol.

IPv6 addresses are represented as eight groups of four hexadecimal digits separated by colons, for example 2001 : 0db 8:85a3 : 0042 : 1000 : 8a2e : 0370 : 7334, but methods of abbreviation of this full notation exist.

3.11.5. Why new IP version?

So far, IPv4 has proven itself as a robust routable addressing protocol and has served human being for decades on its best-effort-delivery mechanism. It was designed in early 80's and did not get any major change afterward. At the time of its birth, internet was limited only to a few universities for their research and to Department of Defense. IPv4 is 32 bits long which offers around 4, 294, 967, 296 (2^{32}) addresses. This address space was considered more than enough that time. Given below are major points which played key role in birth or IPv6:

- Internet has grown exponentially and the address space allowed by IPv4 is saturating. There is a requirement of protocol which can satisfy the need of future Internet addresses which are expected to grow in an unexpected manner.
- Using features such as NAT, has made the Internet discontiguous *i.e.* one part which belongs to intranet, primarily uses private IP addresses; which has to go through number of mechanism to reach the other part, the internet, which is on public IP addresses.
- IPv4 on its own does not provide any security feature which is vulnerable as data on Internet, which is a public domain, is never safe. Data has to be encrypted with some other security application before being sent on internet.
- Data prioritization in IPv4 is not up to date. Though IPv4 has few bits reserved for type of service or quality of service, but they do not provide much functionality.
- IPv4 enabled clients can be configured manually or they need some address configuration mechanism. There exists no technique which can configure a device to have globally unique IP address.

Why not IPv5?

Till date, Internet Protocol has been recognized has IPv4 only. Version 0 to 3 were used while the protocol was itself under development and experimented process. So, we can assume lots of background activites remain active before putting a portocol into production. Similarly, protocol version 5 was used while experimenting with stream protocol for internet. It is known to us as Internet Stream Protocol which used Internet Protocol number 5 to encapsulate its datagram. Though it was never brought into public use, but it was already used.

Here is a table of IP version and their use :

Decimal	Keyword	Version
0 – 1		Reserved
2 – 3		Unassigned
4	IP	Internet Protocol
5	ST	ST Datagram mode
6	IPv6	Internet Protocol Version 6
7	TP/IX	TP/IX The Next Internet
8	PIP	The P Internet Protocol
9	TUBA	TUBA
10 – 14		Unassigned
15		Reserved

3.11.6. Packet Format

The wonder of IPv6 lies in its header. IPv6 address is 4 times larger than IPv4 but the IPv6 header is only 2 times larger than of IPv4. IPv6 headers have one Fixed Header and zero or more Optional (Extension) headers. All necessary information which is essential for a router is kept in fixed header. Extension header contains optional information which helps routers to understand how to handle a packet/flow.

Fixed Header

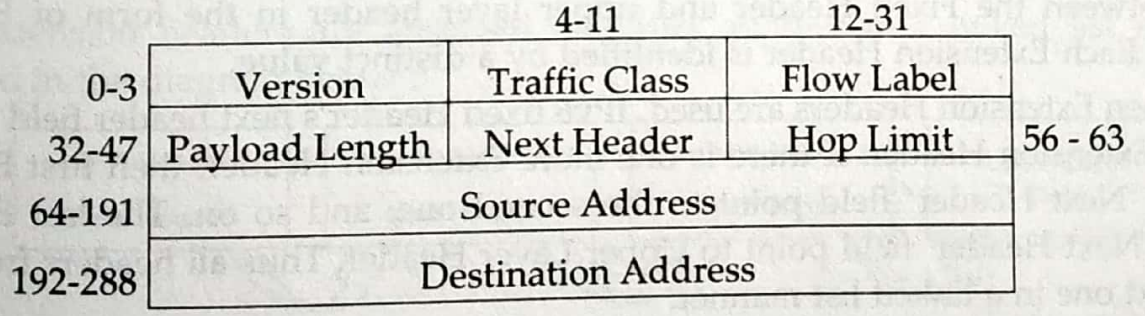


Fig.3.14. IPv6 Fixed Header

IPv6 fixed header is 40 bytes long and contains the following information.

- **Version** (4-bits) : This represents the version of Internet Protocol, *i.e.* 0110.
- **Traffic Class** (8-bits) : These 8 bits are divided into two parts. Most significant 6 bits are used for Type of Service, which tells the router what services should be provided to this packet. Least significant 2 bits are used for Explicit Congestion Notification (ECN).
- **Flow Label** (20-bits) : This label is used to maintain the sequential flow of the packets belonging to a communication. The source labels the sequence which helps the router to identify that this packet belongs to a specific flow of information. This field helps to avoid re-ordering of data packets. It is designed for streaming/real-time media.
- **Payload Length** (16-bits) : This field is used to tell the routers how much information this packet contains in its payload. Payload is composed of Extension Headers and Upper Layer data. With 16 bits, up to 65535 bytes can be indicated but if extension headers contain Hop-by-Hop Extension Header than payload may exceed 65535 bytes and this field is set to 0.
- **Next Header** (8-bits) : This field is used to indicate either the type of Extension Header, or if extension header is not present then it indicates the upper layer PDU. The values for the type of upper layer PDU is same as IPv4's.
- **Hop Limit** (8-bits) : This field is used to stop packet to loop in the network infinitely. This is same as TTL in IPv4. The value of hop limit field is decremented by 1 as it passes a link (router/hop). When the field reaches 0 the packet is discarded.
- **Source Address** (128-bits) : This field indicates the address of originator of the packet.
- **Destination Address** (128-bits) : This field provides the address of intended recipient of the packet.

Extension Headers

In IPv6, the fixed header contains only information which is necessary and avoiding information which is either not required or is rarely used. All such information is put between the Fixed Header and upper layer header in the form of Extension Headers. Each Extension Header is identified by a distinct value.

When Extension Headers are used, IPv6 fixed Header's next header field points to the first Extension Header. If there is one more Extension Header, then first Extension Header's 'Next Header' field point to the second one, and so on. The last Extension Header's Next-Header' field point to Upper Layer Header. Thus all headers from point to the next one in a linked list manner.

If the Next Header field contains value 59, it indicates that there's no header after this header, not even Upper Layer Header.

The following Extension Headers must be supported as per RFC 2460 :

Extension Header	Next Header Value	Description
Hop-by-Hop options header	0	read by all devices in transit network
Routing header	43	contains methods to support making routing decision
Fragment header	44	contains parameters of datagram fragmentation
Destination Options Header	60	read by destination devices
Authentication header	51	information regarding authenticity
Encapsulating Security Payload header	50	encryption information

The sequence of Extension Headers should be :

IPv6 header
Hop-by-Hop options header
Destination options header [1]
Routing header
Fragment header
Authentication header
Encapsulating Security Payload header
Destination Options header [2]
Upper-layer header

These headers :

1. Should be processed by First and subsequent destinations.
2. Should be processed by Final Destination.

Extension headers are arranged one after another in a linked list manner, as depicted in the diagram below :

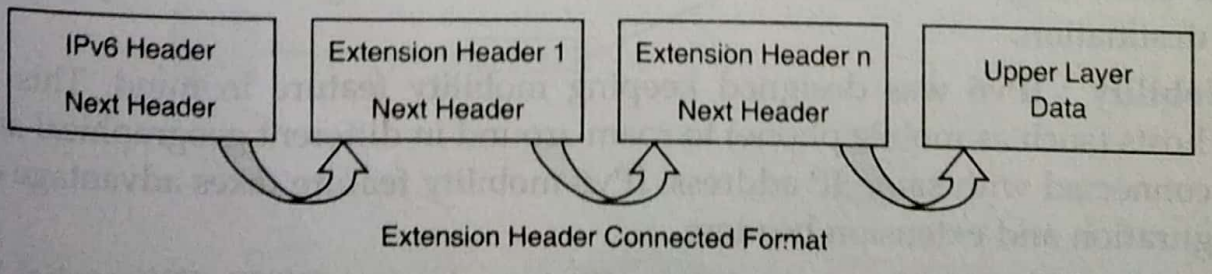


Fig. 3.15

3.11.7. IPv6 - Features

The successor of IPv4 is not designed to be backward compatible. Trying to keep the basic functionalities of IP addressing, IPv6 is redesigned entirely. It offers the following features :

Larger Address Space : In contrast to IPv4, IPv6 uses 4 times more bits to address a device on the internet. This much of extra bits can provide approximately $3 \cdot 4 \times 1038$ different combinations of addresses. This address can accumulate the aggressive requirement of address allotment for almost everything in this world. According to an estimate, 1564 addresses can be allocated to every square metre of this earth.

Simplified Header : IPv6's header has been simplified by moving all unnecessary information and options (which are present in IPv4 header) to the end of the IPv6 header. IPv6 header is only twice as bigger than IPv4 providing the fact the IPv6 address is four times longer.

End-to-end Connectivity : Every system now has unique IP address and can traverse through the internet without using NAT or other translating components. After IPv6 is fully implemented, every host can directly reach other host on the internet, with some limitations involved like Firewall, Organization's policies, etc.

Auto-configuration : IPv6 supports both stateful and stateless auto configuration mode of its host devices. This way absence of a DHCP server does not put halt on inter segment communication.

Faster Forwarding/Routing : Simplified header puts all unnecessary information at the end of the header. All information in first part of the header are adequate for a Router to take routing decision thus making routing decision as quickly as looking at the mandatory header.

IPSec : Initially it was decided for IPv6 to must have IPsec security, making it more secure than IPv4. This feature has now been made optional.

No Broadcast : Though Ethernet/Token Ring are considered as broadcast network because they support broadcasting, IPv6 does not have any broadcast support anymore left with it. It uses multicast to communicate with multiple hosts.

Anycast Support : This is another characteristic of IPv6. IPv6 has introduced anycast mode of packet routing. In this mode, multiple interfaces over the internet are assigned same anycast IP address. Routers, while routing, sends the packet to the nearest destination.

Mobility : IPv6 was designed keeping mobility feature in mind. This feature enables hosts (such as mobile phone) to roam around in different geographical area and remain connected with same IP address. IPv6 mobility feature takes advantage of auto IP configuration and extension headers.

Enhanced Priority Support : Where IPv4 used 6 bits DSCP (Differential Service Code Point) and 2 bits ECN (Explicit Congestion Notification) to provide Quality of

Service but it could only be used if the end-to-end devices support it, that is, the source and destination device and underlying network must support it.

In IPv6, Traffic class and flow label are used to tell underlying routers how to efficiently process the packet and route it.

Smooth Transition : Large IP address scheme in IPv6 enables to allocate devices with globally unique IP addresses. This assures that mechanism to save IP addresses such as NAT is not required. So devices can send/receive data between each other, for example VoIP and/or any streaming media can be used much efficiently.

Other fact is, the header is less loaded so routers can make forwarding decision and forward them as quickly as they arrive.

Extensibility : One of the major advantage of IPv6 header is that it is extensible to add more information in the option part. IPv4 provides only 40-bytes for options whereas options in IPv6 can be as much as the size of IPv6 packet itself.

3.11.8. IPv6- Addressing Modes

In computer networking, addressing mode refers to the mechanism how we address a host on the network. IPv6 offers several types of modes by which a single host can be addressed, more than one host can be addressed at once or the host at closest distance can be addressed.

Unicast : In unicast mode of addressing, an IPv6 interface (host) is uniquely identified in a network segment. The IPv6 packet contains both source and destination IP addresses. A host interface is equipped with an IP address which is unique in that network segment. A network switch or router when receives a unicast IP packet, destined to single host, sends out to one of its outgoing interface which connects to that particular host.

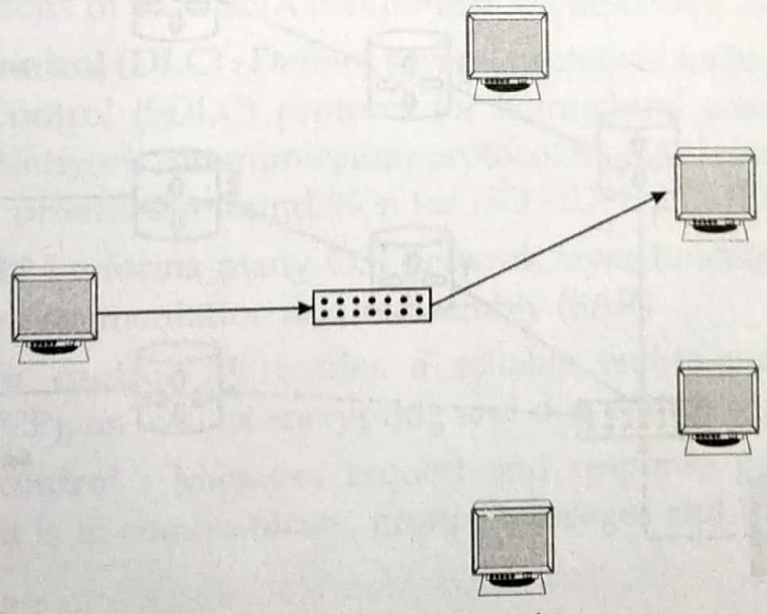


Fig.3.16. Unicast Messaging

Multicast : The IPv6 multicast mode is same as that of IPv4. The packet destined to multiple hosts is sent on a special multicast address. All hosts interested in that multicast information, need to join that multicast group first. All interfaces which have joined the group receive the multicast packet and process it, while other hosts not interested in mulitcast packets ignore the multicast information.

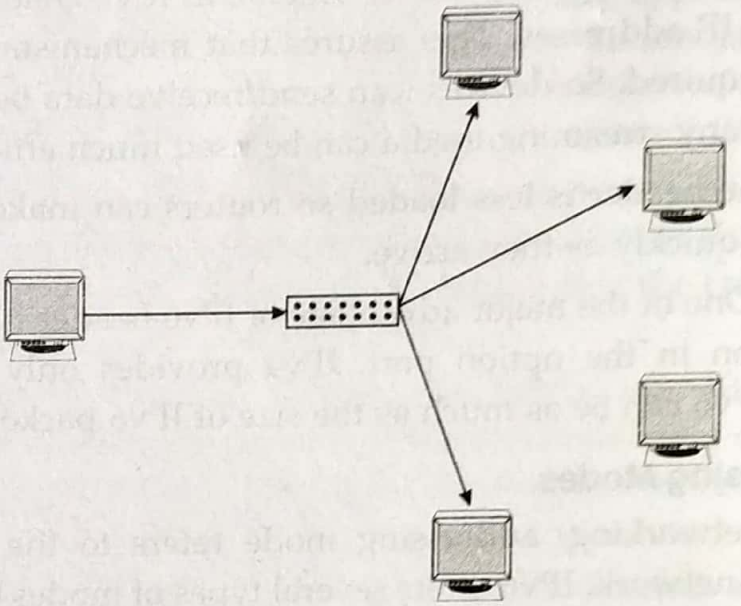


Fig.3.17. Multicast Messaging

Anycast : IPv6 has introduced a new type of addressing, which is called Anycast addressing. In this addressing mode, multiple interfaces (hosts) are assigned same Anycast IP address. When a host wishes to communicate with a host equipped with an Anycast IP address, sends a unicast message. With the help of complex routing mechanism, that Unicast message is delivered to the host clossed to the sender, in terms of routing cost.

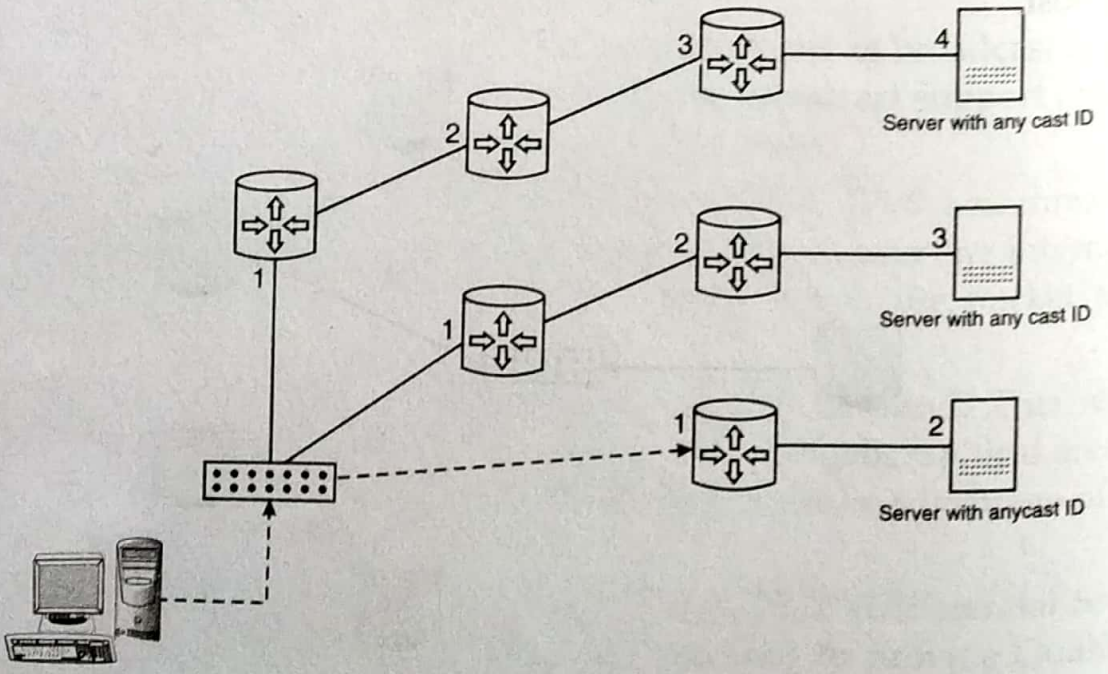


Fig. 3.18. Anycast Messaging

Let's take an example of Tutorial Points. com Web Serves, located in all continents. Assume that all Web Servers are assigned single IPv6 Anycast IP Address. Now when a user from Europe wants to reach TutorialsPoint.com the DNS points to the server which is physically located in Europe itself. If a user from India tries to reach Tutorialspoint.com the DNS will then point to Web Server physically located in Asia only. Nearest or Closest terms are used in terms of Routing Cost.

In the above picture, When a client computer tries to reach a Server, the request is forwarded to the serve with lowest Routing Cost.

3.12. Models and Protocols

In addition to the open architectural models such as OSI 7 layers model and the TCP/IP model, there exist a few popular vendor specific network communication models, such as IBM SNA (Systems Network Architecture), Digital Equipment Corporation's (DEC, now part of HP) DNA (Digital Network Architecture). We will only provide details on the IBM SNA here.

Although it is now considered a legacy networking architecture, the IBM SNA is still widely deployed. SNA was designed around the host-to-terminal communiation model that IBM's mainframes use. IBM expansion was deemed Advanced Peer-to-Peer Networking (APPN) and Advanced Program-to-Program Communication (APPC). Advanced Peer-to-Peer Networking (APPN) and Advanced Program-to-Program Communication (APPC). Advanced Peer-to-Peer Networking (APPN) represents IBM's second-generation SNA. In creating APPN, IBM moved SNA from a hierarchical, mainframe-centric environment to a peer-based networking environment. At the heart of APPN is an IBM architecture that supports peer-based communications, directory services, and routing between two or more APPC systems that are not directly attached.

SNA has many similarities with the OSI 7 layers reference model. However, the SNA model has only six layers and it does not define specific protocols for its physical control layer. The physical control layer is assumed to be implemented via other standards. The functions of each SNA component are described as follows :

- **Data Link Control (DLC) :** Defines several protocols, including the Synchronous Data Link Control (SDLC) protocol for heirarchical communication, and the Token Ring Network communication protocol for LAN communication between peers. SDLC provided a foundation for ISO HDSL and IEEE 802.2.
- **Path control :** Performs many OSI network layer functions, including routing and datagram segmentation and reassembly (SAR)
- **Transmission control :** Provides a reliable end-to-end connection service (similar to TCP), as well as encrypting and decrypting services.
- **Data flow control :** Manages request and response processing, determines whose turn it is to communicate, groups messages and interrupts data flow on request.

- **Presentation services :** Specifies data-transformation algorithms that translate data from one format to another, coordinate resource sharing and synchronize transaction operations.
- **Transaction services :** Provides application services in the form of programs that implement distributed processing or management services.
- The following figure illustrates how the IBM SNA model maps to the ISO OSI reference model.

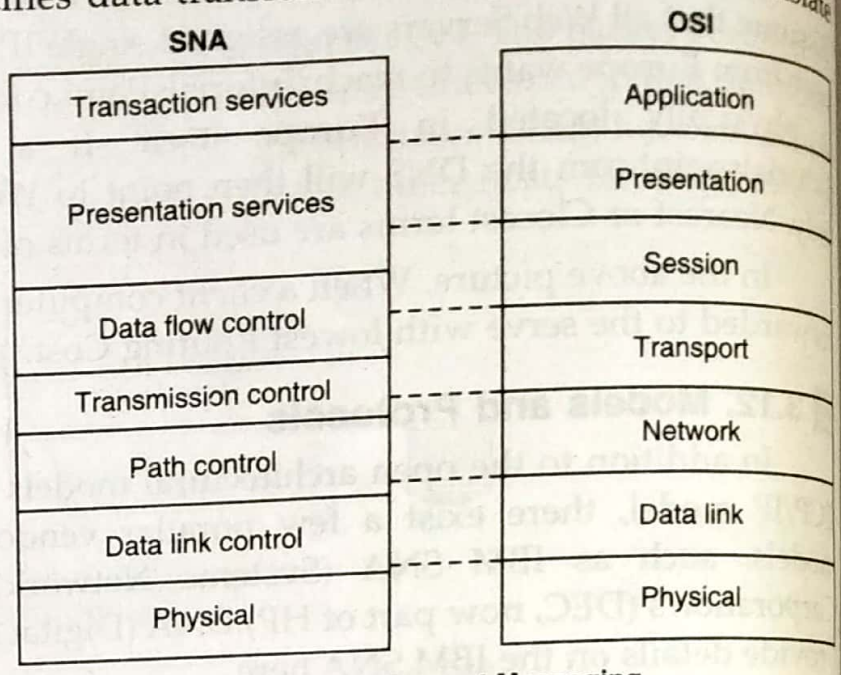


Fig. 3.19. Anycast Messaging

A typical SNA Network Topology

SNA supports the following types of networks :

- A subarea network is a hierarchically organized network consisting of subarea nodes and peripheral nodes. Sub-area nodes, such as hosts and communication controllers, handle general network routing. Peripheral nodes, such as terminals, attach to the network without awareness of general network routing.

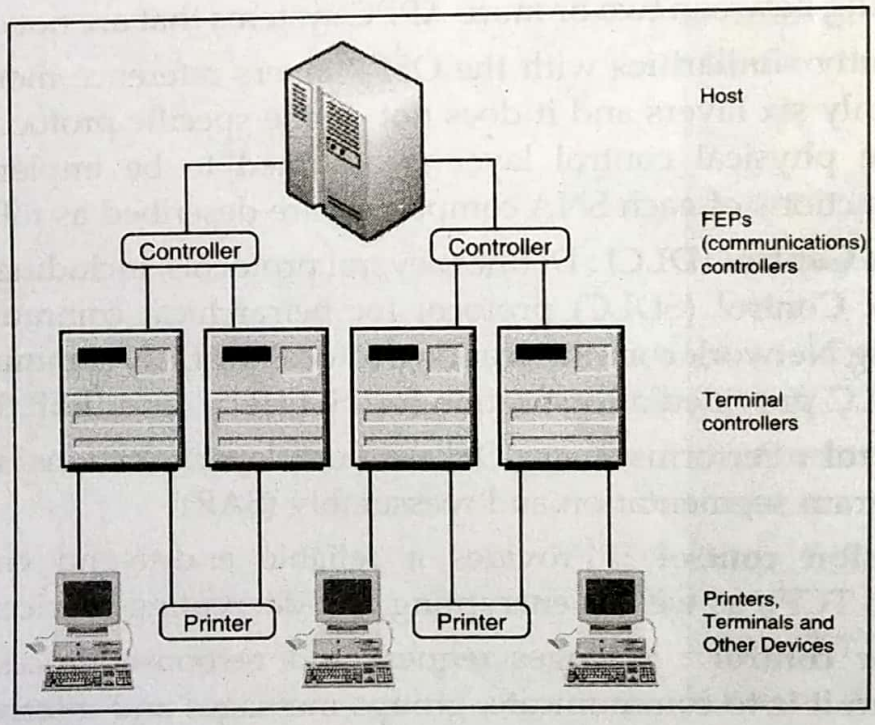


Fig. 3.20. SNA Network Topology

- A peer network is a cooperatively organized network consisting of peer nodes that all participate in general network routing.
- A mixed network is a network that supports both host-controlled communications and peer communications.
- In SNA networks, programs that exchange information across the SNA network are called transaction programs (TPs). Communication between a TP and the SNA network occurs through network occurs through network accessible units or NAUs (formerly called "network addressable units"), which are unique network resources that can be accessed (through unique local addresses) by other network resources. There are three types of NAU : Physical Unitl, Logic Units and Control Points.

Communication between Transaction Programs (TP) and Logic Units (LU) is shown as follows :

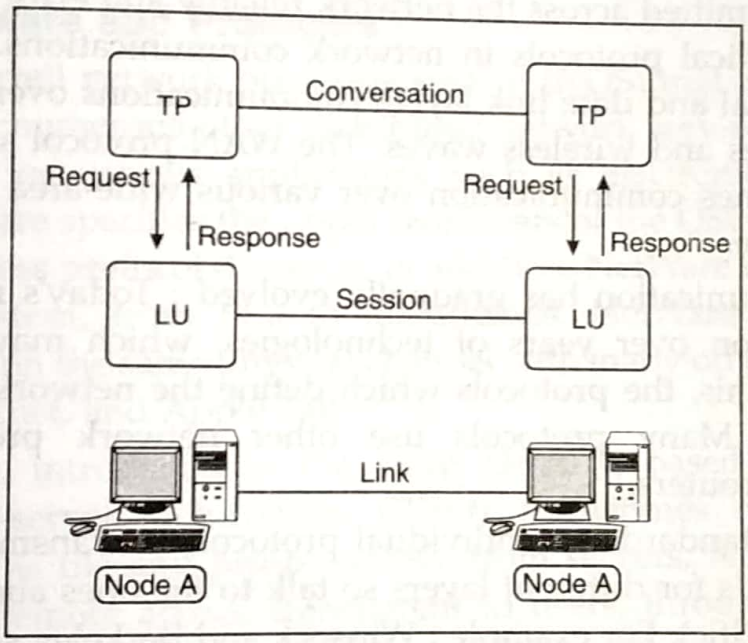


Fig. 3.21. Communication TP and LU in SNA

3.13. Network Protocol

The OSI model, and any other network communication model, provide only a conceptual framework for communication between computers, but the model itself does not provide specific methods of communication. Actual communication is defined by various communication protocols. In the context of data communication, a protocol is a formal set of rules, conventions and data structure that governs how computers and other network devices exchange information over a network. In other words, a protocol is a standard procedure and format that two data communication devices must understand, accept and use to be able to talk to each other.

In modern protocol design, protocols are "layered" according to the OSI 7 layer model or a similar layered model. Layering is a design principle which devides the protocol design into a number of smaller parts, each part accomplishing a particular sub-task, and interacting with the other parts of the protocol only in a small number of

well-defined ways. Layering allows the parts of a protocol to be designed and tested without a combinatorial explosion of cases, keeping each design relatively simple. Layering also permits familiar protocols to be adapted to unusual circumstances.

The header and/or trailer at each layer reflect the structure of the protocol. Detailed rules and procedures of a protocol or protocol stack are often defined by a lengthy document. For example, IETF user RFCs (Request for Comments to define protocols and updates to the protocols.

A wide variety of communication protocols exist. These protocols are defined by many standard organizations throughout the world and by technology vendors over years of technology evolution and development. One of the most popular protocol suites is TCP/IP, which is the heart of Internetworking communication. The IP, the Internet Protocol, is responsible for exchanging information between routers so that the routers can select the proper path for network traffic, while TCP is responsible for ensuring the data packets are transmitted across the network reliably and error free. LAN and WAN protocols are also critical protocols in network communications. The LAN protocols suite is for the physical and data link layers communications over various LAN media such as Ethernet wires and wireless waves. The WAN protocol suite is for the lowest three layers and defines communication over various wide-area media, such as fiber optic and copper cable.

Network communication has gradually evolved : Today's new technologies are based on accumulation over years of technologies, which may be still existing or obsolete. Because of this, the protocols which define the network communication, are highly inter-related. Many protocols use other network protocols to exchange information between routers.

In addition to standards for individual protocols in transmission, there are now also interface standards for different layers so talk to the ones above or below (usually operating-system-specific). For example : Winsock and Berkeley sockets between layers 4 and 5, NDIS and ODI between layers 2 and 3.

The protocols for data communication cover all areas as defined in the OSI model. However, the OSI model is only loosely defined. A protocol may perform the functions of one or more of the OSI layers, which introduces complexity to understand protocols relevant to the OSI 7 layer model. In real-world protocols, there is some argument as to where the distinctions between layers are drawn, there is no one black and white answer.

To develop a complete technology that is useful for the industry, very often a group of protocols is required in the same layer or across many diferent layers. Different protocols often describe differnet aspects of a single communication; taken together, these form a protocol suite. For example, Voice over IP (VOIP), a group of protocols developed by many venders and standard organizations, has many protocols across the 4 top layers in the OSI model.

Protocvols can be implemented either in hardware or software, or a mixture of both. Typically, the lower layers are implemented in hardware, with the higher layers being implemented in software.

Protocols could be grouped into suites (or families, or stacks) by their technical functions, or origin of the protocol introduction, or both. A protocol may belong to one or multiple protocol suites, depending on how you categorize it. For example, the Gigabit Ethernet protocol IEEE 802.3z is a LAN (Local Area Network) protocol and it can also be used in MAN (Metropolitan Area Network) communications.

Most recent protocols are designed by the IETF for internet-working communications, and the IEEE for local area networking (LAN) and metropolitan area networking (MAN). The ITU-T contributes mostly to wide area networking (WAN) and telecommunications protocols. ISO has its own suite of protocols for internetworking communications, which is mainly deployed in European countries.

3.14. Novell Netware and Protocols

NetWare is a Novell network operating system (NOS) that provides transparent remote file access and numerous other distributed network services, including printer sharing and support for various applications such as electronic mail transfer and database access. NetWare specifies the upper five layers of the OSI reference model and runs on any media-access protocol (Layer 2). In addition, NetWare runs on virtually any kind of computer system, from PCs to mainframes. NetWare and its supporting protocols often coexist on the same physical channel with many other popular protocols, including TCP/IP, DECnet, and Apple Talk.

Novell NetWare, introduced in the early 1980s, is based on Xerox Network Systems (XNS) client-server architecture. Clients (sometimes called work-stations) request services, such as file and printer access, from servers, NetWare's client/server architecture supports remote access, transparent to users, through remote procedure calls. A remote procedure call begins when the local computer program running on the client sends a procedure call to the remote server. The server then executes the remote procedure call and returns the requested information to the local clinet.

The most popular protocols in the Novell NetWare suite are :

IPX : Internetwork Packet Exchange protocol-Routing and networking protocol at layer 3. When a device to be communicated with is located on a different network, IPX routes the information to the destination through any intermediate networks. IPX is similar to IP (Internet Protocol) in the TCP/IP suite.

SPX : Sequenced Packet Exchange protocol, control protocol at the transport layer (layer 4) for reliable, connection-oriented datagram transmission. SPX is similar to TCP in the TCP/IP suite.

NCP : Network Core Protocol is a series of server routines designed to satisfy application requests coming from, for example, the NetWare shell. Services provided by

NCP include file access printer access, name management, accounting, security and file synchronization.

NetBIOS : Network Basic Input/Output System (NetBIOS) session-layer interface specification from IBM and Microsoft. NetWare's NetBIOS emulation software allows programs written to the industry-standard NetBIOS interface to run within the NetWare system.

NetWare Application-layer Services : NetWare Message Handling Service (NetWare MHS), Btrieve, NetWare Loadable Modules (NLMs), and various IBM connectivity features. NetWare MHS is a message delivery system that provides electronic mail transport. Btrieve is Novell's implementation of the binary tree (btree) database access mechanism. NLMs are implemented as add-on modules that attach into the NetWare system. NLMs for alternate protocol stacks, communication services, database services, and many other services are currently available from Novell and third parties.

Since NetWare 5·0, all Novell network services can be run on top of TCP/IP. These, IPS and SPX became Novell legacy network and transport layer protocols.

Architecture

The following figure illustrates the NetWare protocol suite, the media-access protocols on which NetWare runs, and the relationship between the NetWare protocols and the OSI reference model.

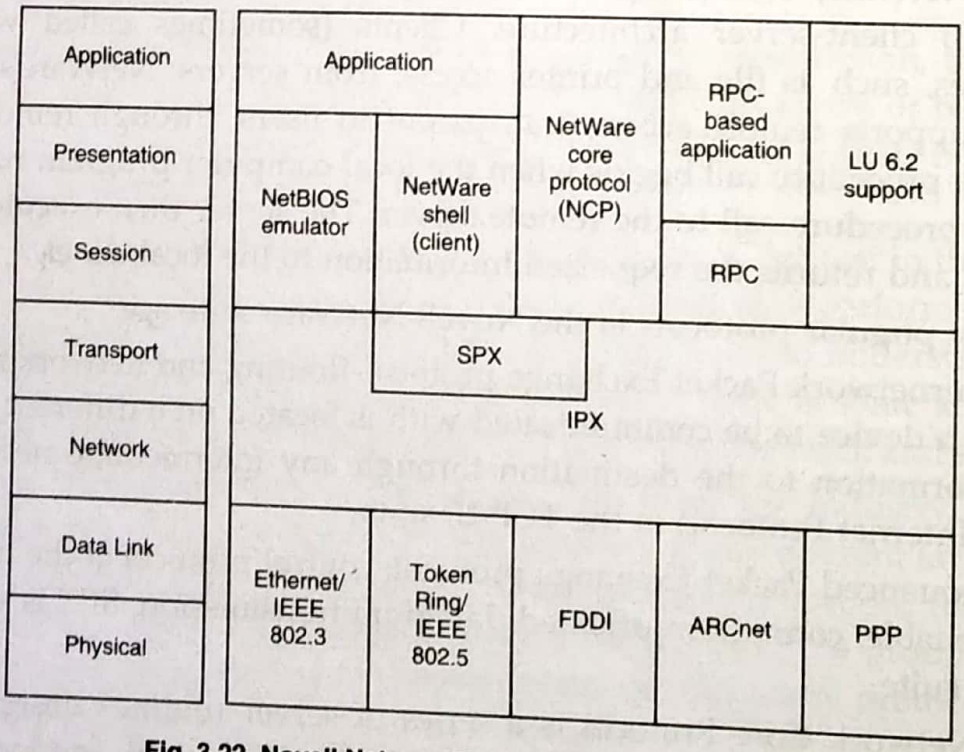


Fig. 3.22. Novell Netware Protocol Stack Architecture

3.14.1. Internetwork Packet Exchange Protocol (IPX)

Internetwork Packet Exchange (IPX) is the legacy network protocol used by the Novell NetWare operating systems to route packets through an internetwork. IPX is a datagram protocol used for connectionless communications similar to IP (Internet Protocol) in the TCP/IP suite. Higher-level protocols, such as SPX and NCP, are used for additional error recovery services.

To make best-path routing decisions, IPX uses the services of a dynamic distance vector routing protocol such as Routing Information Protocol (RIP) or NetWare Link-State Protocol (NLSP).

Novell IPX network addresses are unique and are represented in a hexadecimal format that consists of two parts : a network number and a node number. The IPX network number, which is assigned by the network administrator, is 32 bits long. The node number, which usually is the Media Access Control (MAC) address for one of the system's network interface cards (NICs), is 48 bits long. IPX's use of a MAC address for the node number enables the system to send nodes to predict what MAC address to use on a data link.

Novell NetWare IPX supports four encapsulation schemes on a single router interface :

Novell Proprietary : Also called 802.3 raw or Novell Ethernet_802.3, Novell proprietary serves as the initial encapsulation scheme that Novell uses.

802.3 : Also called Novell_802.3, 802.3 is the standard IEEE 802.3 frame format.

Ethernet version 2 : Also called Ethernet-II or ARPA, Ethernet verison 2 includes the standard Ethernet Version 2 header, which consists of Destination and Source Address fields followed by an EtherType field.

SNAP : Also called Ethernet_SNAP, SNAP extends the IEEE 802.3 header by providing a type code similar to that defined in the Ethernet version 2 specification.

The maximum length of the data section of an IPX packet varies from a minimum of 30 bytes (the header only) depending on the lower layer MAC protocol (Ethernet or token ring) that is being used.

Protocol Structure

The NetWare IPX Packet Header.

8	16 bit
Checksum	
Packet length	
Transport control	Packet type
Destination network (4 bytes)	
Destination node (6 bytes)	
Destination socket (2 bytes)	
Source network (4 bytes)	
Source node (6 bytes)	
Source socket (2 bytes)	

- **Checksum :** Indicates that the checksum is not used when this 16-bit field is set to 1s (FFFF).
- **Packet Length :** Specifies the length, in bytes, of a complete IPX datagram. IPX packets can be any length, up to the media maximum transmission unit (MTU) size (no packet fragmentation allowed).
- **Transport Control :** Indicates the number of routers through which the packet has passed. When this value reaches 16, the packet is discarded under the assumption that a routing loop might be occuring.
- **Packet Type :** Specifies which upper-layer protocol should receive the packet's information. It has two common values :
  - **5 :** Specifies Sequenced Packet Exchange (SPX)
  - **17 :** Specifies NetWare Core Protocol (NCP)
- **Destination network, Destination node, and Destination socket :** Specify destination information.
- **Source network, Source node, and Source socket :** Specify source information.

3.14.2. NetWare Core Protocol (NCP)

The Novell NetWare Core Protocol (NCP) manages access to the primary NetWare server resources. NCP makes procedure calls to the NetWare File Sharing Protocl (NFSP) that services requests for NetWare file and print resources. NCP is the principal protocol for transmitting information between a NetWare server and its clients.

NCP handles login requests and many other types of requests to the file system and the printing system. NCP is a client/server LAN protocol. Workstations create NCP requests and use IPX to send then over the network. At the server, NCP requests are received, unpacked and interpreted.

NCP services include file access, file locking, security, tracking of resource allocation, event notification, synchronization with other servers, connection and communication, print services and queue and network management.

NCP uses the underlying Internetwork Packet Exchange Layer (IPX). More recent NetWare versions (after NetWare 5·0) can also use TCP/IP.

Protocol Structure

The format of the NCP Request header is shown below :

8	16bit
Request type	
Sequence number	Connection number low
Task number	Connection number high
Request code	

- Request type-Identifies the packet type :

1111H.	Allocate slot request.
2222H.	File server request.
3333H.	File server reply.
5555H	Deallocate slot request.
7777H	Burst mode packet (BMP).
9999H	Positive acknowledge.

- **Sequence Number :** Number used by the workstation and file server to identify packets which are sent and received.
- **Connection Number Low** : Low connection ID number assigned to the workstation.
- **Task Number :** Identifies the operating system *e.g.*, DOS, task.
- **Connection Number High** : High Connection ID number assigned to the workstation. Used only on the 1000-user version of NetWare, on all other versions will be set to 0.
- **Request Code :** Identifies the specific request function code.

The structure of the NCP Reply header is the same as the Request header, but the last 2 bytes differ after Connection Number High. This is shown below :

Completion code
Connection status

- **Completion Code :** The completion code indicates whether or not the Client's request was successful. A value of 0 in the Completion Code field indicates that the request was successful. Any other value indicates an error.
- **Connection Status :** The fourth bit in this byte will be set to 1 if DOWN is typed at the console prompt, to bring the server down.

3.14.3. NetWare Link Service Protocol (NLSP)

The NetWare Link Services Protocol (NLSP) is a link-state routing protocol in the Novell NetWare architecture. NLSP is based on the OSI Intermediate System-to-Intermediate System (IS-IS) protocol and was designed to replace IPX RIP (Routing Information Protocol) and SAP (Service Advertisement Protocol), Novell's original routing protocols that were designed for small scale internetworks.

- Compared to RIP and SAP, NLSP provides improved routing, better efficiency, and scalability. The following are the key features of the NLSP.
- NLSP-based routers use a reliable delivery protocol, so delivery is guaranteed.
- NLSP facilitates improved routing decisions because NLSP-based routers store a complete map of the network, not just next-hop information.

- NLSP is efficient, particularly over a WAN link, because its support of IPX header compression makes it possible to reduce the size of packets. NLSP also supports multicast addressing so that routing information is sent only to other NLSP routers, not to all devices, as RIP does.
- NLSP supports load balancing across parallel paths and improves link integrity. It periodically checks links for connectivity and for the data integrity of routing information.
- NLSP is scalabe because NLSP can support up to 127 hops (RIP supports only 15 hope) and permits hierarchical addressing of network nodes, which allows networks to contain thousands of LANs and servers.
- NLSP-based routers are backward compatible with RIP-based routers.
- Similar to IS-IS, NLSP supports hierarchical routing with area domain, and global internetwork components. Areas can be linked to create routing domains, and domains can be linked to create a global internetwork. NLSP supports three levels of hierarchical routing : Level 1, Level 2 and Level 3 routing.
- An NLSP router extracts certain information from the adjacency database and adds locally derived information. Using this information, the router constructs a link-state packet (LSP) that describes its immediate neighbors. All LSPs constructed by all routers in the routing area make up the link-state database for the area. The link-state database is synchronized by reliably propagating LSPs throughout the routing area when a router observes a topology change. Two methods ensure that accurate topology-change information is propagated : flooding and receipt confirmation.
- NLSP supports a hierarchical addressing scheme. Each routing area is identified by two 32-bit quantities : a network address and a mask.

Protocol Structure

NLSP WAN Hello Packet :

1	2	3	4	5		6	8			9 bytes
Protocol ID	Length Ind.	Minor version	Rsvd	Rsvd	Packet type	Major version	Reserved	Rsvd	State	Cct type
Source ID							Holding time	Packet length		
Packet length	Local Wan Circuit ID	Variable Length Fields								

- **Protocol ID :** Identifies the NLSP routing layer with the 0×83 hex number.
- **Length Indicator :** Determines the number of bytes in the fixed portion of the header.
- **Minor Version :** Contains one possible decimal value and is ignored on receipt.

- **Reserved :** Contains no decimal values and is ignored on receipt.
- **Packet Type (5 bits) :** Contains 17 possible decimal values.
- **Major Version :** Contains one possible decimal value.
- **Reserved :** Contains no decimal values and is ignored on receipt.
- **State (2 bits) :** Sends the router's state assiciated with the link (0 = up, 1 = initializing, 2 = down).
- **Circuit Type (Cct type) :** Consists of 2 bits. This field can have one of the following values :
- **0 :** Reserved value; ignore entire packet.
- **1 :** Level 1 routing only.
- **2 :** Level 2 routing only. (The sender uses this link for level 2 routing.)
- **3 :** Both Level 1 and Level 2. (The sender is a Level 2 router and uses this link for level 1 and Level 2 traffic.)
- **Source ID :** Serves as the system identifier of the sending router.
- **Holding Time :** Contains the holding timer, in seconds, to be used for the sending router.
- **Packet Length :** Determines the entire length of the packet, in bytes, including the NLSP header.
- **Local WAN Circuit ID :** Acts as a unique identifier assigned to this circuit when it is created by the router.
- **Variable Length Field :** Consists of a series of optional fields.

1	2	3	4	5		6	8	9 bytes			
Protoc -ol ID	Length Ind.	Minor version	Rsvd	Rsvd	Packet type	Major version	Reserved	Rsvd	NM	Res	Cct type
Source ID							Holding time	Packet length			
Packet Length	R	Priority	LAN ID								
Variable Length Fields											

- **Protocol ID :** Identifies the NLSP routing layer with the 0×83 hex number.
- **Length Indicator :** Determines the number of bytes in the fixed portion of the header (up to and including the LAN ID field).
- **Minor Version :** Contains one possible decimal value and is ignored on receipt.
- **Reserved :** Contains no possible decimal values and is ignored on receipt.
- **Packet Type (5 bits) :** Contains 15 possible decimal values.
- **Major Version :** Contains one possible decimal value.
- **Reserved :** Contains no possible decimal values and is ignored on receipt.

- **No Multicast (NM) (1 bit) :** Indicates, when set to 1, that the packet sender cannot receive traffic addressed to a multicast address. (Future packets on this LAN must be sent to the broadcast address.)
- **Circuit Type (Cct Type) (2 bits) :** Can have one of the following values :
  - **0 :** Reserved value; ignore entire packet.
  - **1 :** Leval 1 routing only.
  - **2 :** Level 2 routing only. (The sender uses this link for Level 2 routing).
  - **3 :** Both Level 1 and Level 2. (The sender is a Level 2 router and uses this link for Level 1 and Level 2 traffic.)
- **Source ID :** Contains the system ID of the sending router.
- **Holding Time :** Contains the holding timer, in seconds, to be used for the sending router.
- **Packet Length :** Determines the entire length of the packet, in bytes, including the NLSP header.
- **R :** Contains no possible decimal values and is ignored on receipt.
- **Priority (7 bits) :** Serves as the priority associated with being the LAN Level 1 designated router. (Higher numbers have higher priority.)
- **LAN ID :** Contains the system ID (6 bytes) of the LAN Level 1 designated router, followed by a field assigned by that designated router.
- **Variable Length Fields :** Consists of a series of optional fields.

3.14.4. Sequenced Packet Exchange Protocol (SPX)

The Sequenced Packet Exchange (SPX) protocol is Novell's legacy transport layer protocol providing a packet delivery service for Novell NetWare network. SPX is based on the Xerox Sequenced Packet Protocol (SPP). SPX, operates on top of IPX and is used in Novell NetWare (prior to NetWare 5·0) systems for communications in client/server application programs, *e.g.*, BTRIEVE (ISAM manager). SPX performs equivalent functions to TCP. The newer versions of NetWare services are run on top of TCP/IP.

IPX receives packets from the network and passes on those for SPX to handle. SPX guarantees that packets are received intact, in the order they were sent, and eliminates duplicate packets. SPX prepares the sequence of packets that a message is divided into and manages the reassembly of received packets, confirming that all have been received and requesting retransmission when they haven't. SPX works directly with the Internetwork Packet Exchange (IPX) protocol, which manages the forwarding of packets in the network. SPX does not provide connections to the file server itself, which uses the NetWare Core Protocol (NCP). SPX has been extended as SPX-II (SPX2).

SPX does not provide group broadcast support; packets can only be sent to a single session partner. SPX can detect if its partner has disappeared.

Protocol Structure

The structure of the SPX packet is shown in the following illustration :

8	16bit
Connection control flag	Datastream type
Source connection ID	
Destination connection ID	
Sequence number	
Acknowledge number	
Allocation number	
Data (0-534 bytes)	

- **Connection Control Flag :** Four flags which control the bi-directional flow of data across an SPX connection. These flags have a value of 1 when set and 0 if not set.

  Bit 4 — Eom : End of message.
  Bit 5 — Att : Attention bit, not used by SPX
  Bit 6 — Ack : Acknowledge required.
  Bit 7 — Sys : Transport control.

- **Datastream Type :** Specifies the data within the packet :
- **Source Connection ID :** A 16-bit number assigned by SPX to identify the connection.
- **Destination Connection ID :** The reference number used to identify the target end of the transport connection.
- **Sequence Number :** A 16-bit number, managed by SPX, which indicates the number of packets transmitted.
- **Acknowledge Number :** A 16-bit number, indicating the next expected packet.
- **Allocation Number :** A 16-bit number, indicating the number of packets sent but not yet acknowledged.
- The SPX II header is the same as the SPX header described above, except for the following differences :
  - **Connection Control Flag :** Bit 2 - Size negotiation. Bit 3 - SPX II type.
  - **Datastream Type-252 :** Orderly release request. 253 Orderly release acknowledgment.

There is also an additional 2-bites Extended Acknowledgment field at the end.

3.15. Apple Talk Protocols

Apple Talk is the architecture used on with Apple brand computers and is a suite of protocols for networking Apple computers. Some of the protocols are :

- **Apple Share :** Works at the application layer to provide services.
- **AFP (Apple Talk Filing protocol) :** Makes network files appear local by managing file sharing at the presentation layer.

- **ATP :** Apple Talk Transaction Protocol provides a Transport Layer connection between computers. Three transaction layers are :
  - Transaction requires (TREQ)
  - Transaction response (TRESP)
  - Transaction release (TREL)
- **DDP :** Datagram Delivery Protocol is a routable protocol that provides for data packet transportation. It operates at the network layer at the same level of the IP protocol.

The Apple Talk networking scheme puts computers into groups called zones. This is similar to workgroups on a Windows network.

Four Session Layer Protocols

- **ASP :** AppleTalk Session Protocol controls the starting and ending of sessions between computers called nodes. It works at the session level. The NBP, described below is used to get addresses from computer names. ATP is used at the transport level.
- **ADSP :** AppleTalk Data Stream Protocol manages the flow of data between two established socket connections.
- **ZIP :** Zone Information Protocol used with RTMP to map zones. Routers use zone information tables (ZITs) to define network addresses and zone names.
- **PAP :** Printer Access Protocol manages information between workstations and printers.

Other Protocols

- **NBP :** Name Binding Protocol translates addresses into names.
- **AEP :** AppleTalk Echo Protocol uses echoes to tell if a computer, or node, is available.
- **RTMP :** Routing Table Maintenance Protocol is used to update routers with information about network status and address tables. The whole address table is sent across the network.
- **ARUP :** AppleTalk Update Routing is a newer version of RTMP.