UNIT 4: THE NETWORK LAYER

4.1. FUNCTION OF NETWORK LAYER

- The Network Layer is the third layer of the OSI model.
- It handles the service requests from the transport layer and further forwards the service request to the data link layer.
- The network layer translates the logical addresses(IP address) into physical addresses
- It determines the route from the source to the destination and also manages the traffic problems such as switching, routing and controls the congestion of data packets.
- The main role of the network layer is to move the packets from the sending host to the receiving host.

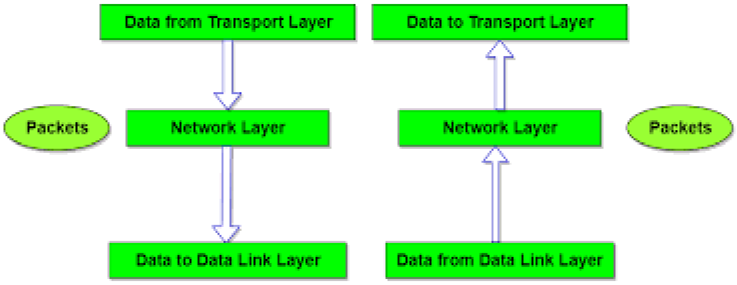
The main functions performed by the network layer are:

- **Routing:** When a packet reaches the router's input link, the router will move the packets to the router's output link. For example, a packet from S1 to R1 must be forwarded to the next router on the path to S2.
- **Logical Addressing:** The logical addressing at the network layer while physical addressing at the data link layer is defined by the MAC address of a device, whereas the IP addressing is determined at the network layer of the OSI model. This addressing is also called as logical addressing. The network layer adds a header to the packet which is coming from the upper layer includes the logical addresses of the sender and receiver.
- **Internetworking:** This is the main role of the network layer that it provides the logical connection between different types of networks.

- **Fragmentation and Reassembly**
  The network layer must send data down to the data link layer for transmission. The data or information that the network layer receives is in the form of a packet and the data that data link layer forwards is called a frame.
- The network layer has the responsibility of Fragmentation and reassembly because some data link layer technologies have limits on the length of any message that can be sent.
- If the packet of data that the network layer has to send is too large, the network layer must break the packet up, send each packet to the data link layer, and then have pieces reassembled once they arrive at the network layer on the destination system.

Example:

If I want to access some data from Facebook then I will open my laptop, type URL of Facebook and send an HTTP request to facebook.com for some data. Since the server of Facebook is situated outside my local area network, my request is forwarded to Facebook through the default gateway or router of my institution.



Design Issues with Network Layer

A major design issue in the network layer is to determine the packet routing that is how

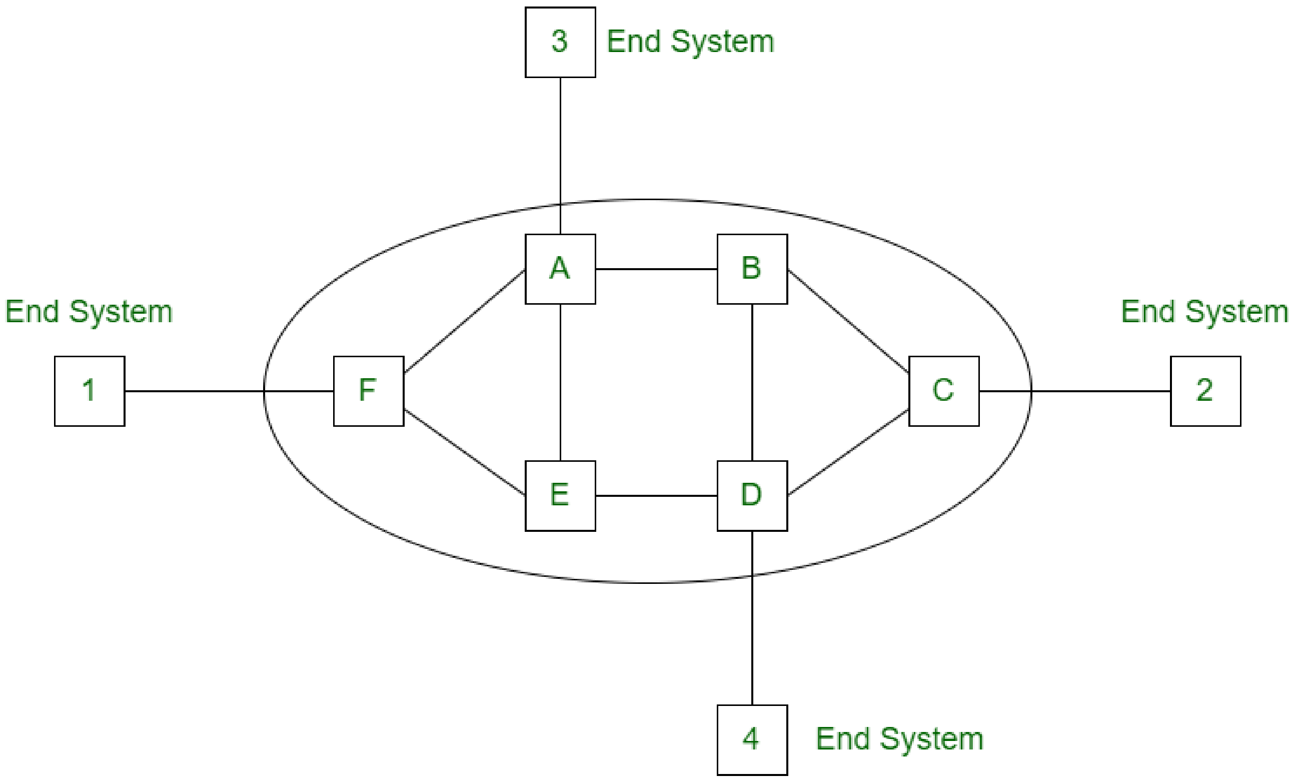
each packet routed from source to destination. Routes can be based on static tables and also highly dynamic, that is each packet has a predefined route or it can be changed for each packet. If there are too many packets available in the subnet at the particular time, they will get into one another's way, forming bottlenecks. The network layer issue is the quality of service provided such as delay, transmit time, jitter, etc.
When packet travel from one network to another to reach its destination,
many problems can arise such as:

- The addressing being used by two networks may be different from each other.
- It is necessary to have different protocols.

4.2. VIRTUAL CIRCUIT AND DATAGRAM SUBNET

Virtual Circuit:

Virtual Circuit is the computer network providing connection-oriented service. It is a connection-oriented network. In virtual circuit resource are reserve for the time interval of data transmission between two nodes. This network is a highly reliable medium of transfer. Virtual circuits are costly to implement.



Working of Virtual Circuit:

- In the first step a medium is set up between the two end nodes.
- Resources are reserved for the transmission of packets.
- Then a signal is sent to sender to tell the medium is set up and transmission can be started.
- It ensures the transmission of all packets.
- A global header is used in the first packet of the connection.
- Whenever data is to be transmitted a new connection is set up.

Datagram:

This approach uses a different, more dynamic scheme, to determine the route through the network links.

Each packet is treated as an independent entity, and its header contains full information about the destination of the packet.

The intermediate nodes examine the header of the packet, and decide to which node to send the packet so that it will reach its destination.

In this method, the packets don't follow a pre-established route, and the intermediate nodes (the routers) don't have pre-defined knowledge of the routes that the packets should be passed through.

Packets can follow different routes to the destination, and delivery is not guaranteed .

Due to the nature of this method, the packets can reach the destination in a different order than they were sent, thus they must be sorted at the destination to form the original message.

This approach is time consuming since every router has to decide where to send each packet. The main implementation of Datagram Switching network is the Internet, which uses the IP network protocol.

Sno	**Datagram Packet Switching**	**Virtual-circuit Packet Switching**
1	Two packets of the same user pair can travel along different routes.	All packets of the same virtual circuit travel along the same path.
2	The packets can arrive out of sequence.	Packet sequencing is guaranteed.
3	Packets contain full Src, Dst addresses	Packets contain short VC Id. (VCI).
4	Each host occupies routine table entries.	Each VC occupies routing table entries.
5	Requires no connection setup.	Requires VC setup. First packet has large delay.
6	Also called Connection less	Also called connection oriented.
7	Examples: X.25 and Frame Relay	Eg. Internet which uses IP Network protocol.

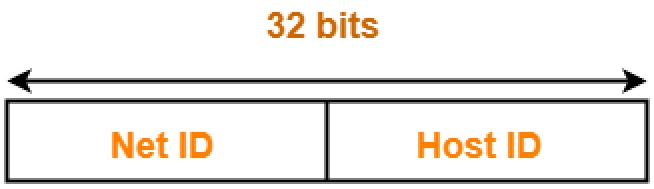
4.3. IPv4 Addresses: Address Space, Notations, Classful addressing, Classless addressing, Subnetting and Network Address Translation(NAT)

ADDRESS SPACE AND NOTATION

______________ .______________.______________.________________

IPv4 addresses are 32-bit numbers that are typically displayed in dotted decimal notation. A 32-bit address contains two primary parts: the network prefix and the host number. Each host also has an address that uniquely identifies it.

- An IPv4 address is 32 bits long
- The IPv4 addresses are unique and universal
- IPv4 uses 32-bit addresses, which means that the address space is 2^32 or 4,294,967,296(Maximum available theoretically)



Format of an IP Address

IPv4 have 2 types of notations:

1. Dotted decimal notations
Denoted in decimal format each byte is separated by dot eg: 117.149.29.2 Mostly used by human configurations

0.0.0.0-255.255.255.255

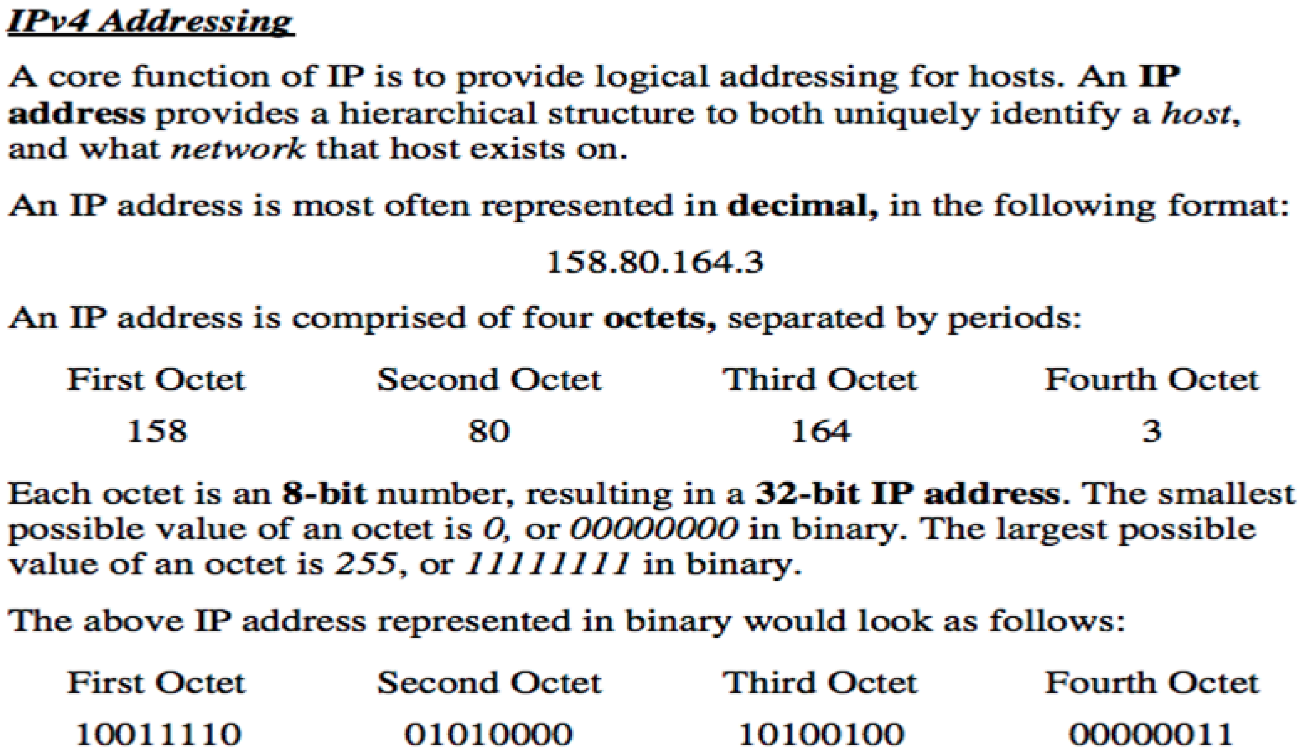
2. Binary notation
In binary format eg: 01110101 10010101 00011101 00000010
00000000.00000000.00000000.00000000-11111111.11111111.11111111.11111111

CLASSFUL ADDRESSING (Classification of IP Addresses)

IP address is an address having information about how to reach a specific host, especially outside the

LAN. An IP address is a 32 bit unique address having an address space of 2^32.

1. The value of any segment (byte) is between 0 and 255 (both included).
2. There are no zeroes preceding the value in any segment (054 is wrong, 54 is correct)



IPv4 Addressing

A core function of IP is to provide logical addressing for hosts. An **IP address** provides a hierarchical structure to both uniquely identify a *host*, and what *network* that host exists on.

An IP address is most often represented in **decimal,** in the following format:

158.80.164.3

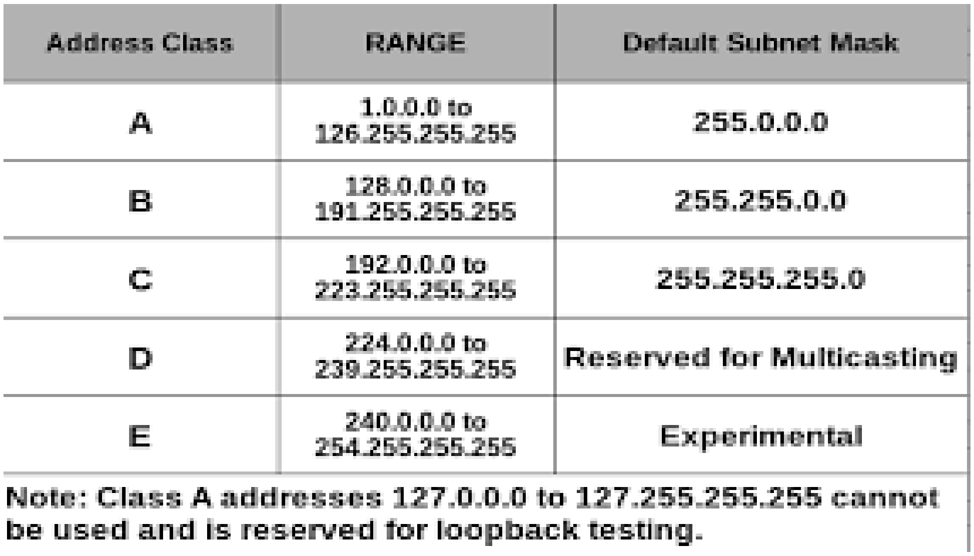
An IP address is comprised of four **octets,** separated by periods:

First Octet	Second Octet	Third Octet	Fourth Octet
158	80	164	3

Each octet is an **8-bit** number, resulting in a **32-bit IP address**. The smallest possible value of an octet is *0,* or *00000000* in binary. The largest possible value of an octet is *255*, or *11111111* in binary.

The above IP address represented in binary would look as follows:

First Octet	Second Octet	Third Octet	Fourth Octet
10011110	01010000	10100100	00000011



Address Class	RANGE	Default Subnet Mask
A	1.0.0.0 to 126.255.255.255	255.0.0.0
B	128.0.0.0 to 191.255.255.255	255.255.0.0
C	192.0.0.0 to 223.255.255.255	255.255.255.0
D	224.0.0.0 to 239.255.255.255	Reserved for Multicasting
E	240.0.0.0 to 254.255.255.255	Experimental

Note: Class A addresses 127.0.0.0 to 127.255.255.255 cannot be used and is reserved for loopback testing.

0.0.0.0-default route

255.255.255.255=broadcasting

A=1-127 B=128-191 C=192-223 D=224-239 E=240-255

Classful Addressing

The 32 bit IP address is divided into five sub-classes. These are:

- *Class A*
- *Class B*
- *Class C*
- *Class D*
- *Class E*

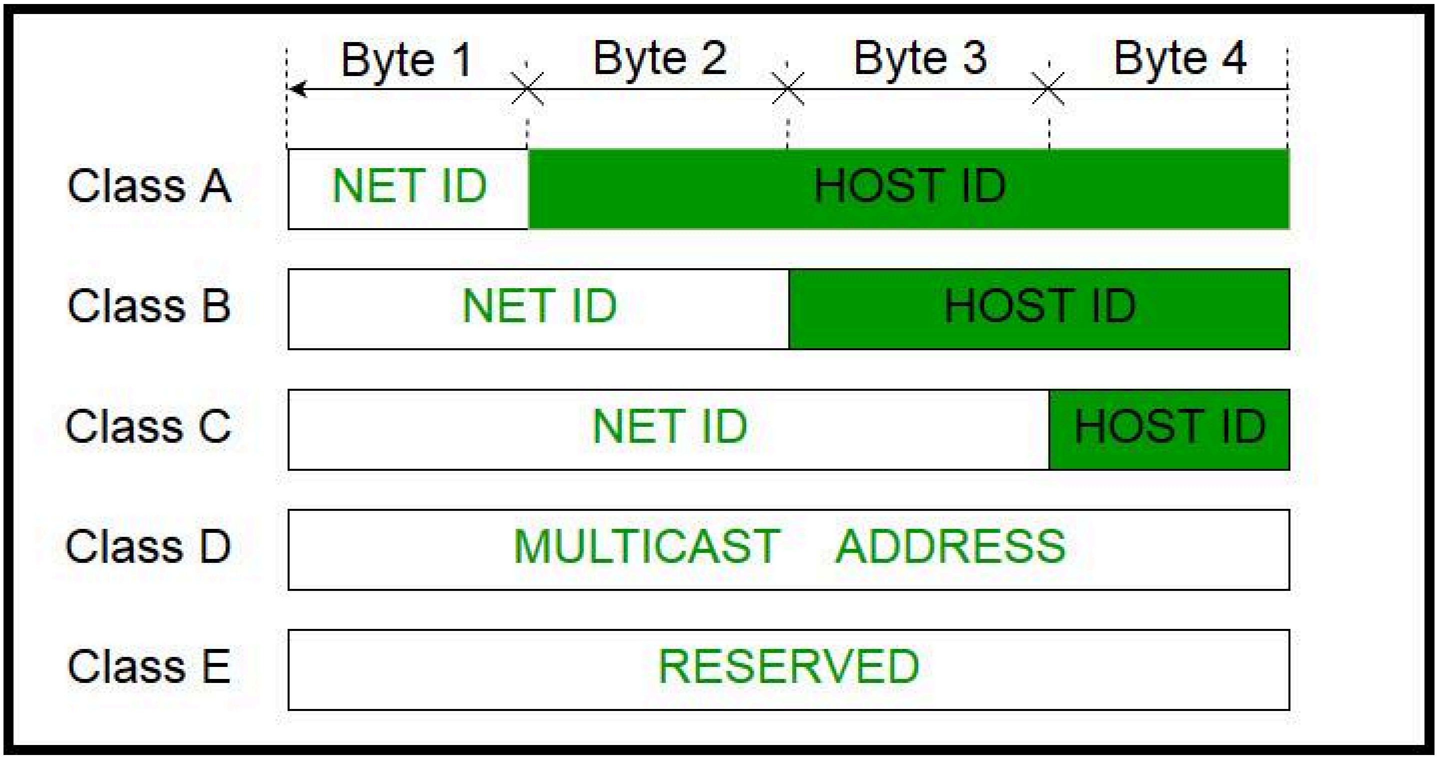
Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively. The order of bits in the first octet determine the classes of IP address.

IPv4 address is divided into two parts:

- *Network ID*
- *Host ID*

The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that particular class. Each ISP or network administrator assigns IP address to each device that is connected to its network.

A=2^24=   B=2^16   C=2^8=256



Note: IP addresses are globally managed by Internet Assigned Numbers Authority(IANA) and regional Internet registries(RIR).

Note: While finding the total number of host IP addresses, 2 IP addresses are not counted and are therefore, decreased from the total count because the first IP address of any network is the network number and whereas the last IP address is reserved for broadcast IP.

Class A:RANGE=(1-127)   NID=8 BIT   HID=24   TOTAL ADDRESS=2^24=16,777,214

SUBNET MASK= NID denoted by 1 HID part denoted by 0

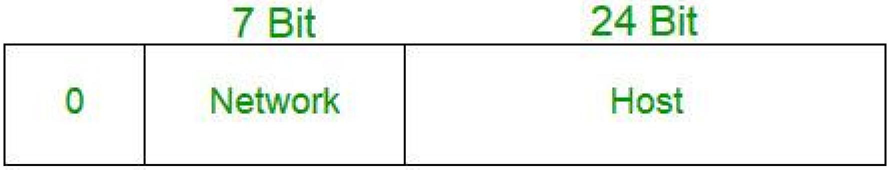
SUBNET MASK=255.0.0.0

IP address belonging to class A are assigned to the networks that contain a large number of hosts.

- *The network ID is 8 bits long.*
- *The host ID is 24 bits long.*

The higher order bit of the first octet in class A is always set to 0. The remaining 7 bits in first octet are used to determine network ID. The 24 bits of host ID are used to determine the host in any network. The default subnet mask for class A is 255.x.x.x. Therefore, class A has a total of:

- 2^7-2= 126 network ID(Here 2 address is subracted because 0.0.0.0 and 127.x.y.z are special address.)
- 2^24 – 2 = 16,777,214 host ID



Class A

IP addresses belonging to class A ranges from 1.x.x.x – 126.x.x.x

Class B:RANGE=128-191 NID=16 HID=16 TOTAL ADDRESS=2^16=65534

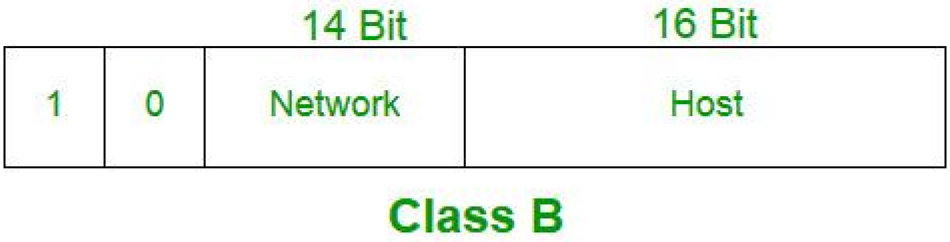
SUBNET MASK:255.255.0.0

IP address belonging to class B are assigned to the networks that ranges from medium-sized to large-sized networks.

- The network ID is 16 bits long.
- The host ID is 16 bits long.

The higher order bits of the first octet of IP addresses of class B are always set to 10. The remaining 14 bits are used to determine network ID. The 16 bits of host ID is used to determine the host in any network. The default sub-net mask for class B is 255.255.x.x. Class B has a total of:

- 2^14 = 16384 network address
- 2^16 – 2 = 65534 host address
    - IP addresses belonging to class B ranges from 128.0.x.x – 191.255.x.x.

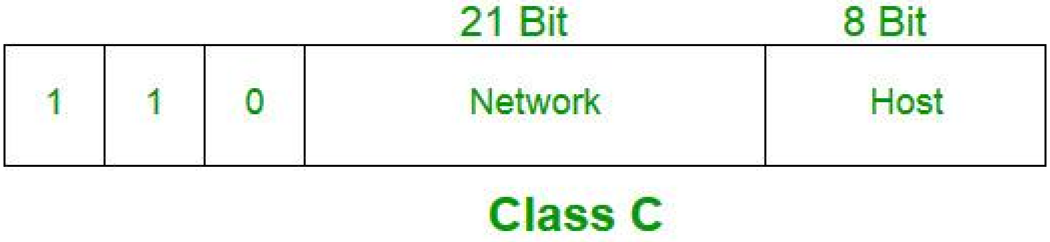


Class C: RANGE=192-223 NID=16   HID=8   TOTAL ADDRESS=2^8=256

SUBNET MASK:255.255.255.0
IP address belonging to class C are assigned to small-sized networks.

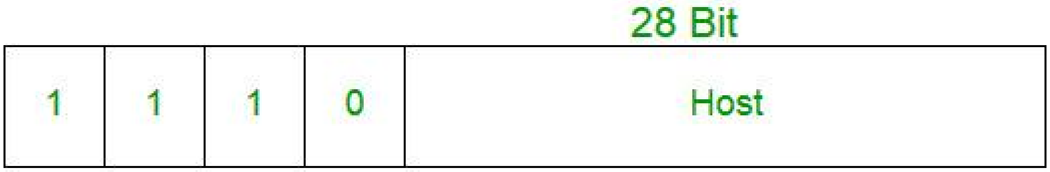
- *The network ID is 24 bits long.*
- *The host ID is 8 bits long.*

- *The higher order bits of the first octet of IP addresses of class C are always set to 110. The remaining 21 bits are used to determine network ID. The 8 bits of host ID is used to determine the host in any network. The default sub-net mask for class C is 255.255.255.x. Class C has a total of:*
  - *2^21 = 2097152 network address*
  - *2^8 – 2 = 254 host address*
- *IP addresses belonging to class C ranges from 192.0.0.x – 223.255.255.x.*



Class D:
IP address belonging to class D are reserved for multi-casting. The higher order bits of the first octet of IP addresses belonging to class D are always set to 1110. The remaining bits are for the address that interested hosts recognize.
Class D does not posses any sub-net mask. IP addresses belonging to class D ranges from

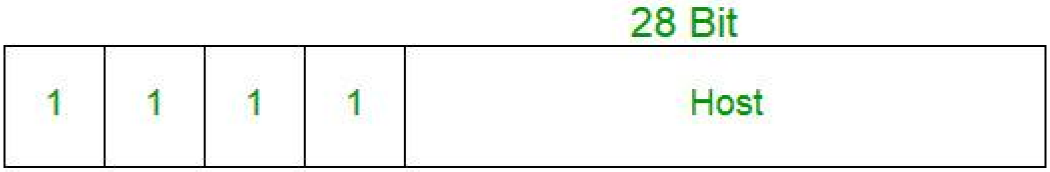
224.0.0.0 – 239.255.255.255.



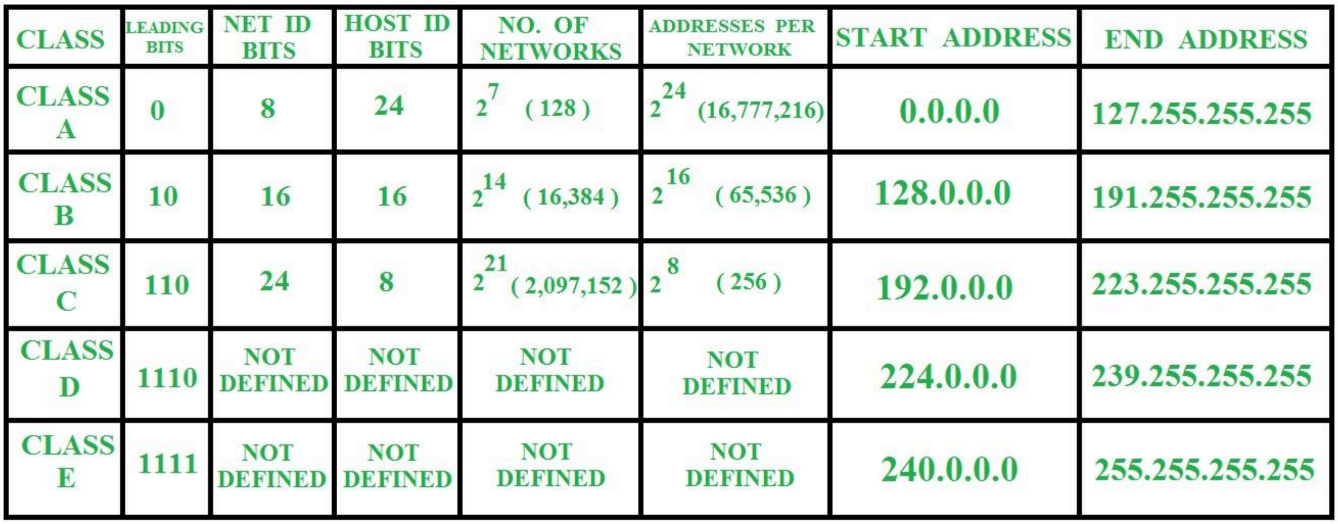
Class D

Class E:

IP addresses belonging to class E are reserved for experimental and research purposes. IP addresses of class E ranges from 240.0.0.0 – 255.255.255.254. This class doesn't have any sub-net mask. The higher order bits of first octet of class E are always set to 1111.



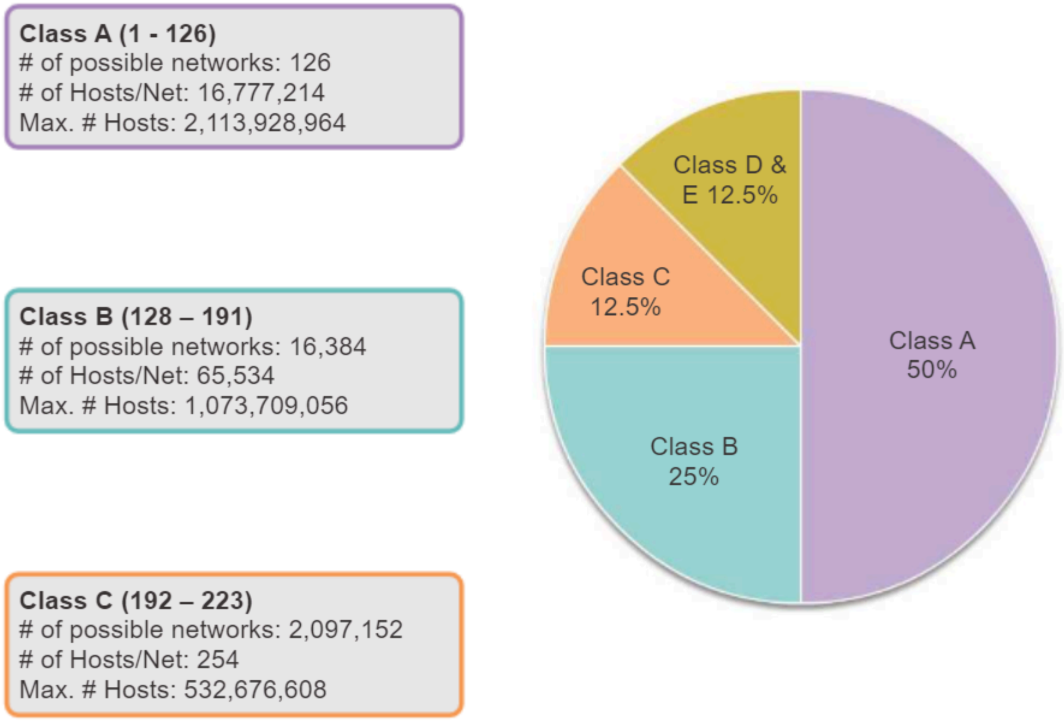
Class E



CLASS	LEADING BITS	NET ID BITS	HOST ID BITS	NO. OF NETWORKS	ADDRESSES PER NETWORK	START ADDRESS	END ADDRESS
CLASS A	0	8	24	2^7 (128)	2^{24} (16,777,216)	0.0.0.0	127.255.255.255
CLASS B	10	16	16	2^{14} (16,384)	2^{16} (65,536)	128.0.0.0	191.255.255.255
CLASS C	110	24	8	2^{21} (2,097,152)	2^8 (256)	192.0.0.0	223.255.255.255
CLASS D	1110	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	224.0.0.0	239.255.255.255
CLASS E	1111	NOT DEFINED	NOT DEFINED	NOT DEFINED	NOT DEFINED	240.0.0.0	255.255.255.255

Problems with Classful Addressing:

Classful IP Address Allocation = Inefficient



Class A (1 - 126)
# of possible networks: 126
# of Hosts/Net: 16,777,214
Max. # Hosts: 2,113,928,964

Class B (128 – 191)
# of possible networks: 16,384
# of Hosts/Net: 65,534
Max. # Hosts: 1,073,709,056

Class C (192 – 223)
# of possible networks: 2,097,152
# of Hosts/Net: 254
Max. # Hosts: 532,676,608

The classful addressing specified in RFCs 790 and 791 resulted in a tremendous waste of address space. In the early days of the Internet, organizations were assigned an entire classful network address from the A, B, or C class.

As illustrated in the figure:

- Class A had 50% of the total address space. However, only 126 organizations could be assigned a class A network address. Ridiculously, each of these organizations could provide addresses for up to 16 million hosts. Very large organizations were allocated entire class A address blocks. Some companies and governmental organizations still have class A addresses. For example, General Electric owns 3.0.0.0/8, Apple Computer owns 17.0.0.0/8, and the U.S. Postal Service owns 56.0.0.0/8.
- Class B had 25% of the total address space. Up to 16,384 organizations could be assigned a class B network address and each of these networks could support up to 65,534 hosts. Only the largest organizations and governments could ever hope to use all 65,000 addresses. Like class A networks, many IP addresses in the class B address space were wasted.
- Class C had 12.5 % of the total address space. Many more organizations were able to get class C networks, but were limited in the total number of hosts that they could connect. In fact, in many

cases, class C addresses were often too small for most midsize organizations.
- Classes D and E are used for multicasting and reserved addresses.

The overall result was that the classful addressing was a very wasteful addressing scheme. A better network addressing solution had to be developed. For this reason, Classless Inter-Domain Routing (CIDR) was introduced in 1993.

CLASSLESS ADDRESSING

- To reduce the wastage of IP addresses in a block, we use subnetting.
- We give the IP address and define the number of bits for the mask along with it (usually followed by a '/' symbol), like, 192.168.1.1/28.
- Here, a subnet mask is found by putting the given number of bits out of 32 as 1, like, in the given address, we need to put 28 out of 32 bits as 1 and the rest as 0, and so, the subnet mask would be 255.255.255.240.There is no classes hierarchy in the IP address but address is still granted in blocks.

Restriction

To simplify the handling of addresses, the Internet authorities impose three restrictions on classless address blocks:

- The addresses in a block must be contiguous, one after another.
- The number of addresses in a block must be a power of 2 (I, 2, 4, 8, ..)
- The first address must be evenly divisible by the number of addresses. Ex: if a block contains 4 addresses, the beginning address must be divisible by 4.

SUBNETTING

- **Subnetting** is the practice of dividing a network into two or smaller networks. It increases routing efficiency, which helps to enhance the security of the network.
- IP Subnetting designates high-order bits from the host as part of the network prefix. This method divides a network into smaller subnets.
- It also helps you to reduce the size of the routing tables, which is stored in routers.
- Subnetting means increasing networks bits(i.e. 1s) in subnet mask
  - If network bit is increased host bits will be decreased, so number of host will be decreased
  - A Class A network have 8 bits for network (224 IP address available) if you wanted smaller block IP from class A increase the network bits /decreasing host bits

Here are important reasons for using Subnetting:

- It helps you to maximise IP addressing efficiency.
- Extend the life of IPV4.
- Public IPV4 Addresses are scarce.
- This method allows you to apply network security policies at the interconnection between subnets.
- Optimized IP network performance.
- Subnetting process helps to allocate IP addresses that prevent large numbers of IP network addresses from remaining unused.
-

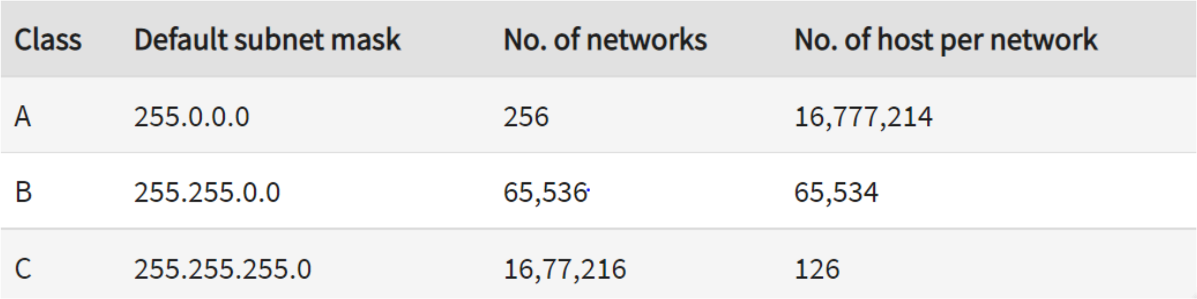
Subnet Mask

A subnet mask is a 32 bits address used to distinguish between a network address and a host address in IP address. A subnet mask identifies which part of an IP address is the network address and the host address.

How to Use a Subnet Mask?

The subnet mask is used by the router to cover up the network address. It shows which bits are used to identify the subnet.

Every network has its own unique address, Like here, class B network has network address 172.20.0.0, which has all zeroes in the host portion of the address.



Class	Default subnet mask	No. of networks	No. of host per network
A	255.0.0.0	256	16,777,214
B	255.255.0.0	65,536	65,534
C	255.255.255.0	16,77,216	126

Supernetting

- Supernetting means creating bigger network from smaller one
- Supernetting means decreasing networks bits(i.e. 1s) in subnet mask
- If network bit is decreased host bits will be increased, so number of host will be decreased
- A Class C network have 24 bits for network (28 IP address available) if you wanted bigger block IP from class C decrease the network bits / increasing host bits
- Supernetting just opposite of subnetting

VLSM (Variable Length Subnet Mask)

- Subnetting and supernetting is achieved by varying default subnet mask
- Usually in classful IP address have 8,16,24 default CIDR values for Class A, B, C respectively, but in classless IP no default CIDR value / subnet mask is available CIDR value may be varying

- **Change the following IPv4 addresses from binary notation to dotted-decimal notation.**

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add

dots for separation.

a. 129.11.11.239

b. 193.131.27.255

- **Change the following IPv4 addresses from dotted-decimal notation to binary notation.**

a. 111.56.45.78

b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent (see Appendix B).

a.• 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

- **Find the error, if any, in the following IPv4 addresses.**

a. 111.56.045.78

b. 221.34.7.8.20

c. 75.45.301.14-------0.0.0.0-255.255.255.255

d. 11100010.23.14.67

Solution

a. There must be no leading zero (045).

b. There can be no more than four numbers in an IPv4 address.

c. Each number needs to be less than or equal to 255 (301 is outside this range).

d. A mixture of binary notation and dotted-decimal notation is not allowed.

- **Find the class of each address.**

a. 00000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

c. 14.23.120.8

d. 252.5.15.111

Solution

a. The first bit is O. This is a class A address.

b. The first 2 bits are 1; the third bit is O. This is a class C address.

c. The first byte is 14 (between 0 and 127); the class is A.

d. The first byte is 252 (between 240 and 255); the class is E

- **A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block ?**

Solution

11001101 00010000 00100101 0010**0111-IP ADDRESS**

11001101 000100000100101 001**0000- FIRST ADDRESS**

11001101 000100000100101 0010**1111**-LAST ADDRESS

First Address The first address in the block can be found by setting the 32 - n rightmost bits in the binary notation of the address to Os.

The binary representation of the given address is 11001101 00010000 00100101 00100 I 11. If we set 32 - 28 rightmost bits to 0, we get 11001101 000100000100101 0010000 or 205.16.37.32.

The first address in the block can be found by setting the rightmost 32 - n bits to Os.

Find the last address for the block?

Last Address The last address in the block can be found by setting the 32 - n rightmost bits in the binary notation of the address to Is.

The last address in the block can be found by setting the rightmost 32 - n bits to Is.

The binary representation of the given address is 11001101 00010000001001010010011l. If we set 32 - 28 rightmost bits to 1, we get 11001101 00010000 001001010010 1111 or 205.16.37.47.

Find the number of addresses?

The value of n is 28, which means that number of addresses is 232- 28 or 16.

- **Suppose an organization is given the block 17.12.40.0/26, which contains 64 addresses. The organization has three offices and needs to divide the addresses into three sub blocks of 32, 16, and 8 addresses.**

- An ISP provides a organization with IP addresse 192.168.10.0/24. You are told to divide the IP address among four department. Department A=60,B=45, C=32, D=12 address .

NETWORK ADDRESS TRANSLATION (NAT)

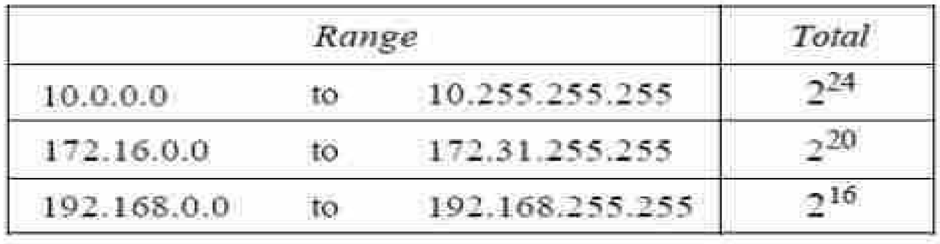
Network address translation is a method of mapping an IP address space into another by modifying network address information in the IP header of packets.

IP addresses have public range and private range. A private **IP address** is the address your network router assigns to your device.Private (internal) addresses **are not routed on the Internet** and no traffic can be sent to them from the Internet, they are only supposed to work within the local network. Private addresses include IP addresses from the following subnets.
Private addresses can be assigned by **the router** using the Dynamic Host Configuration Protocol or be manually set, after which the addresses can communicate with one another through the router.

Public range is used for communication in internet and can used only with permission of internet authorities
Private IP can be used for local communication without permission of Internet

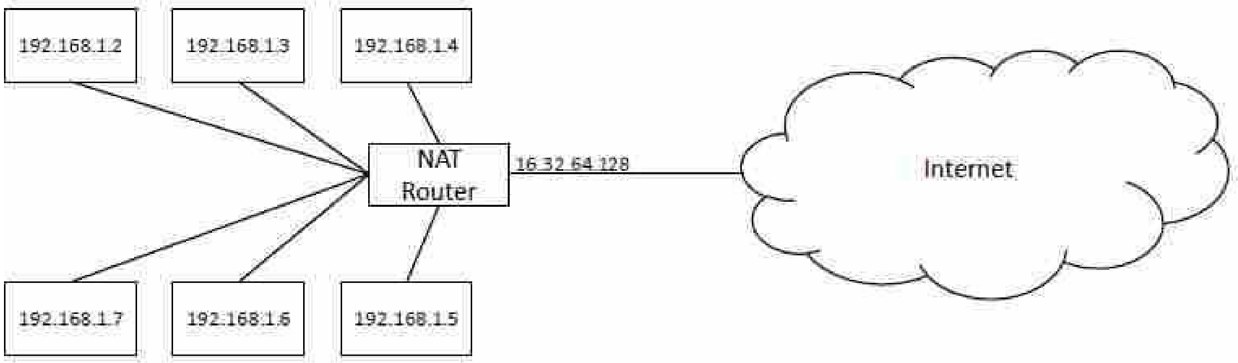
authorities

Given below table shows private ranges of class A,B,C



Range			Total
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

- Public IP should be unique globally
- Private IP should be unique inside a organization, not globally
- **Private IP addresses** are provided by network devices, such as routers, using network **address** translation (NAT).
- NAT router consist of public IP in exit interface and internal interface consist of Private IPs

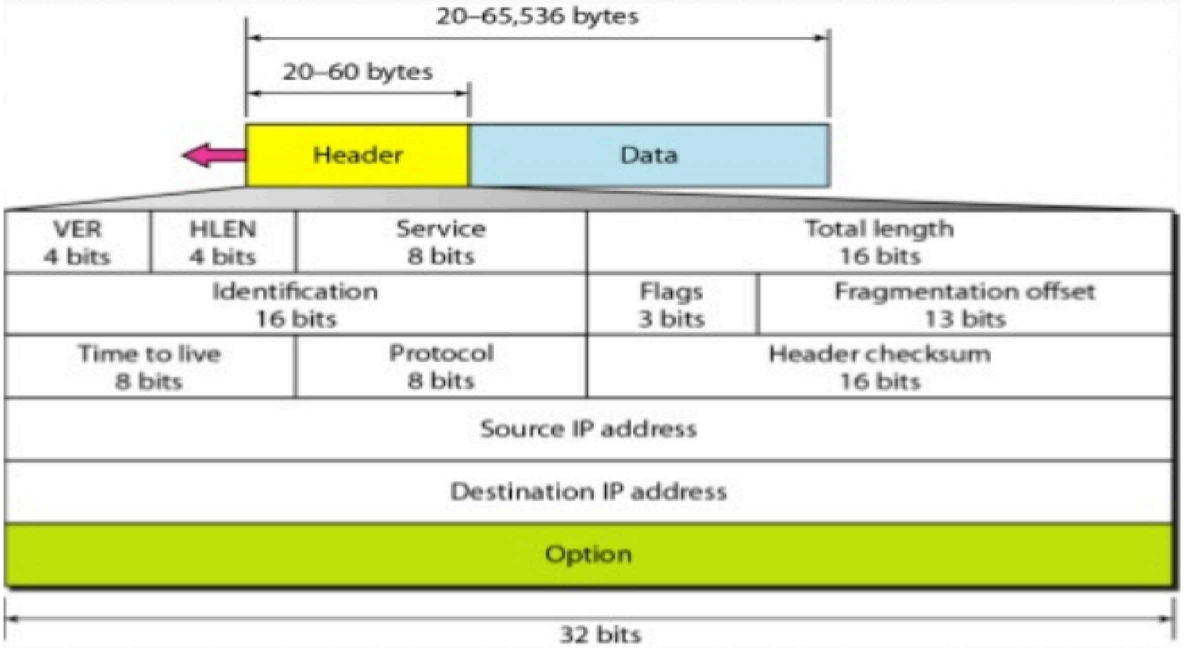


Address Translation : Replace outgoing packets Source IP address as NAT router public IP and replaces incoming packet Destination IP with private (Private to public and public to private)

Translation is done with help of translation table which consist of IP address of private range and public range and port address

Generally, the border router is configured for NAT i.e the router which has one interface in the local (inside) network and one interface in the global (outside) network. When a packet traverses outside the local (inside) network, NAT converts that local (private) IP address to a global (public) IP address. When a packet enters the local network, the global (public) IP address is converted to a local (private) IP address.

4.4. IPv4 Datagram Header



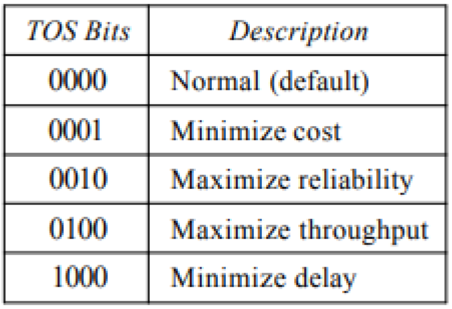
Packets in the IPv4 layer are called datagrams. A datagram is a variable-length packet consisting of two parts: header and data. The header is 20 to 60 bytes in length and contains information essential to routing.

- **Version (VER):** This 4-bit field defines the version of the IPv4 protocol. Currently, the version is 4. However, version 6 (or IPng) may totally replace version 4 in the future.

- **Header length (HLEN):** This 4-bit field defines the total length of the datagram header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes).

- **Services:** IETF has changed the interpretation and name of this 8-bit field. This field, previously called service type, is now called differentiated services. **Type of service:** Low Delay, High Throughput, Reliability (8 bits)

Table 20.1 *Types ofservice*



TOS Bits	*Description*
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

- **Total Length:** Length of header + Data (16 bits), which has a minimum value 20 bytes and the maximum is 65,535 bytes

- **Identification:** Unique Packet Id for identifying the group of fragments of a single IP datagram (16 bits)

- **Flags:** 3 flags of 1 bit each: reserved bit (must be zero), do not fragment flag, more fragments flag.
- **Fragment Offset:**
  The Fragment Offset field (13 bits) is used to indicate the starting position of the data in the fragment .This information is used to reassemble the data from all the fragments (whether they arrive in order or not).

- **Time to live:** Datagram's lifetime (8 bits), It prevents the datagram to loop in the network

- **Protocol:** Name of the protocol to which the data is to be passed (8 bits)

- **Header Checksum:** 16 bits header checksum for checking errors in the datagram header

- **Source IP address:** 32 bits IP address of the sender

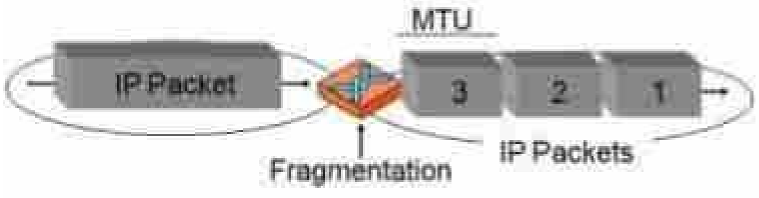
- **Destination IP address:** 32 bits IP address of the receiver

- **Option:** Optional information such as source route. Due to the presence of options, the size of the datagram header can be of variable length (20 bytes to 60 bytes).

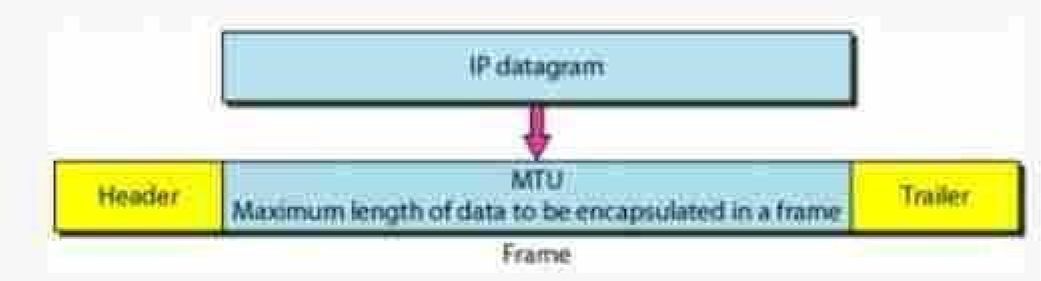
Fragmentation

Fragmentation is done by **the network layer when the maximum size of datagram is greater than maximum size of data** that can be held in a frame i.e., its Maximum Transmission Unit (MTU). The network layer divides the datagram received from the transport layer into fragments so that data flow is not disrupted.

Fragmentation is done by routers. Fragmentation may be done multiple times along the route.
If the IP packet is longer than the MTU, the router breaks the packet into smaller packets called IP fragments.
Fragments are still IP packets.



Maximum Transfer Unit (MTU)
Each data link layer protocol has its own frame format in most protocols.
One of the fields defined in the format is the maximum size of the data field.
In other words, when a datagram is encapsulated in a frame, the total size of the datagram must be less than this maximum size.



1.6. Limitations of IPv4

- Exponential growth of the Internet and the impending exhaustion of the IPv4 address space
- Requirement for security at the IP level
- Need for better support for prioritized and real-time delivery of data

4.5. IPV6 address structure and advantage over IPV4

IPV6

Internet Protocol version 6 (IPv6) is the most recent 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.

With the rapid growth of the Internet after commercialization in the 1990s, it became evident that far more addresses would be needed to connect devices than the IPv4 address space had available. IPv6 uses a 128-bit address, theoretically allowing 2^{128}, or approximately 3.4×10^{38} addresses. Several IPv6 transition mechanisms have been devised to permit communication between IPv4 and IPv6 hosts.

IPv6 addresses are represented as eight groups of four hexadecimal digits with the groups being separated by colons, for example 2001:0db8:0000:0042:0000:8a2e:0370:7334, but methods to abbreviate this full notation exist.

LIMITATIONS OF IPV4

IPv4 has some deficiencies that make it unsuitable for the fast-growing Internet.

o Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.

o The Internet must accommodate real-time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.

o The Internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

ADVANTAGES OF IPV6

The next-generation IP, or IPv6, has some advantages over IPv4 that can be summarized as follows:

- Larger address space. An IPv6 address is 128 bits long, Compared with the 32-bit address of IPv4, this is a huge increase in the address space.
- Better header format. IPv6 uses a new header format in which options are separated from the base header.
- New options. IPv6 has new options to allow for additional functionalities.
- Allowance for extension. IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.
- Support for traffic such as real-time audio and video.
- Support for more security. The encryption and authentication options in IPv6 provide

PACKET FORMAT

The IPv6 packet is shown in Figure. Each packet is composed of a mandatory base header followed by the payload. The payload consists of two parts: optional extension headers and data from an upper layer. The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65,535 bytes of information.

Base Header

Figure shows the base header with its eight fields. These fields are as follows:

o Version. This 4-bit field defines the version number of the IP. For IPv6, the value is 6.

o Priority. The 4-bit priority field defines the priority of the packet with respect to traffic congestion.

Figure 20.15 *IPv6 datagram header and payload*

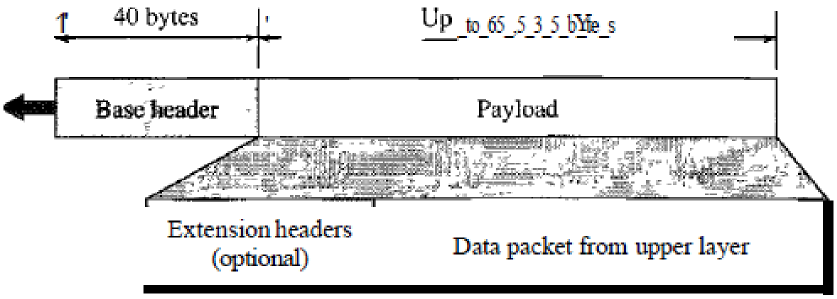
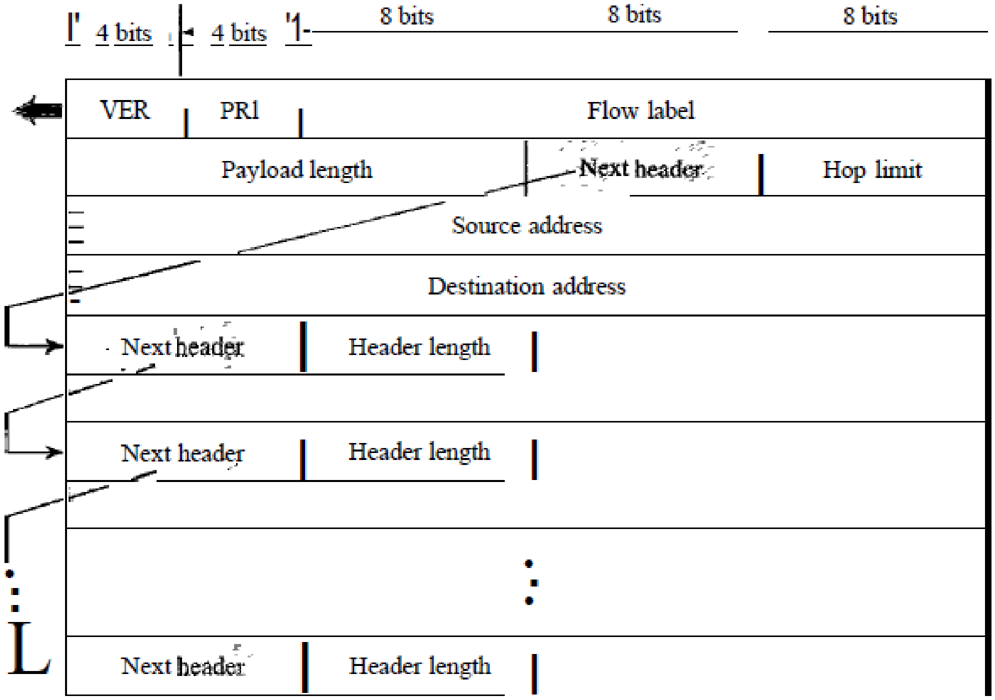


Figure 20.16 *Format ofan IPv6 datagram*



- Flow label. The flow label is a 3-byte (24-bit) field that is designed to provide special handling for a particular flow of data.
- Payload length. The 2-byte payload length field defines the length of the IP datagram excluding the base header.
- Next header. The next header is an 8-bit field defining the header that follows the base header in the datagram.
- Hop limit. This 8-bit hop limit field serves the same purpose as the TTL field in IPv4.
- Source address. The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.
- Destination address. The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram. However, if

source routing is used, this field contains the address of the next router.

Priority

The priority field of the IPv6 packet defines the priority of each packet with respect to other packets from the same source. For example, if one of two consecutive Datagrams must be discarded due to congestion, the datagram with the lower packet priority will be discarded.

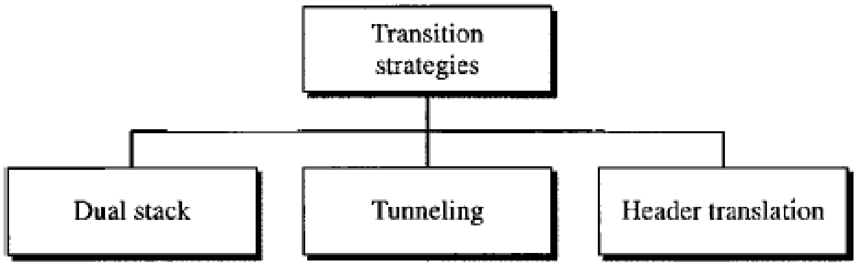
Flow Label

A sequence of packets, sent from a particular source to a particular destination that needs special handling by routers is called a flow of packets. The combination of the source address and the value of the flow label uniquely define a flow of packets.

TRANSITION FROM IPv4 TO IPv6

Because of the huge number of systems on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly. It takes a considerable amount of time before every system in the Internet can move from IPv4 to IPv6. The transition must be smooth to prevent any problems between IPv4 and IPv6 systems. Three strategies have been devised to help the transition

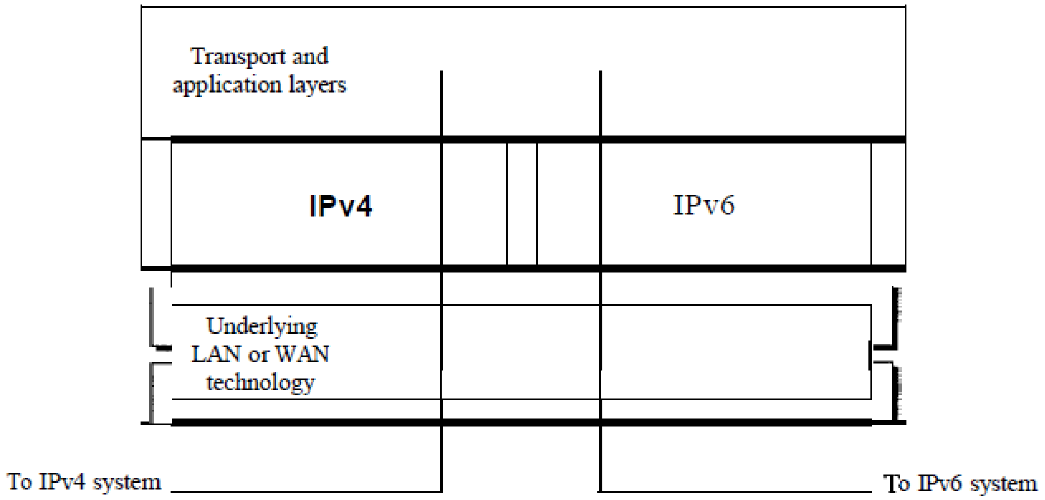
Figure 20.18 *Three transition strategies*



Dual Stack

It is recommended that all hosts, before migrating completely to version 6, have a dual stack of protocols. In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6.

Figure 20.19 *Dual stack*

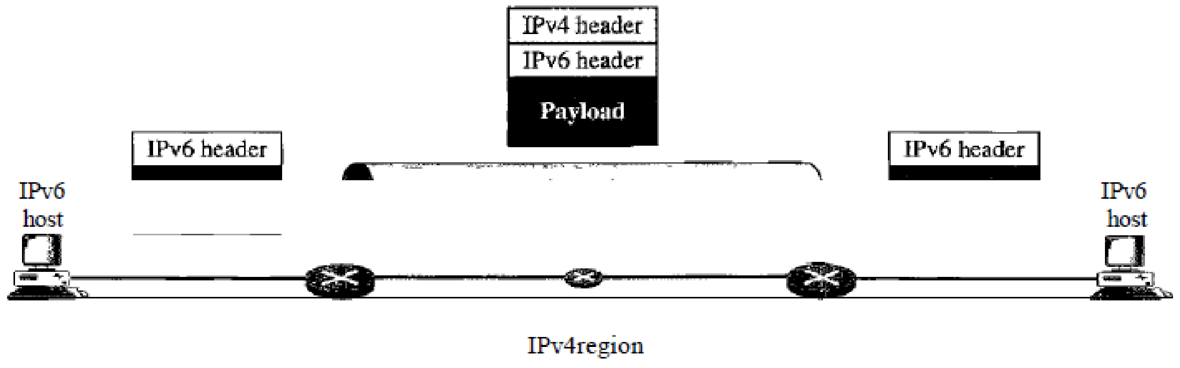

Transport and application layers
IPv4
IPv6
Underlying LAN or WAN technology
To IPv4 system
To IPv6 system

To determine which version to use when sending a packet to a destination, the source host queries the DNS. If the DNS returns an IPv4 address, the source host sends an IPv4 packet. If the DNS returns an IPv6 address, the source host sends an IPv6 packet.

Tunneling

Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4. To pass through this region, the packet must have an IPv4 address. So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region. It seems as if the IPv6 packet goes through a tunnel at one end and emerges at the other end.

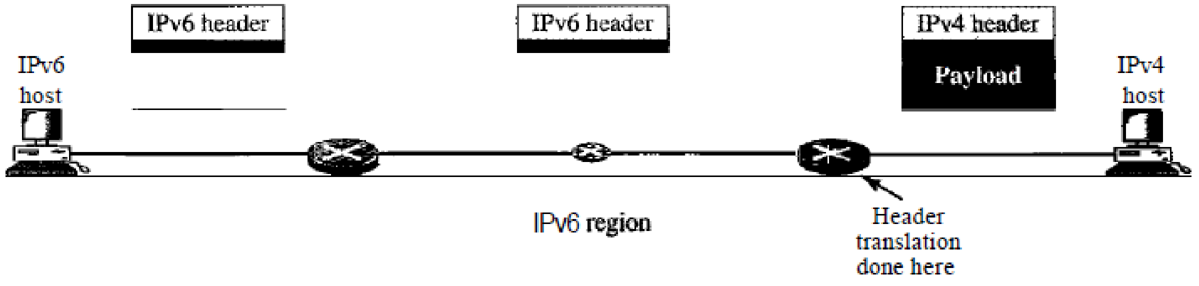
Figure 20.20 *Tunneling strategy*



Header Translation

Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4. The sender wants to use IPv6, but the receiver does not understand IPv6. Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver. In this case, the header format must be totally changed through header translation. The header of the IPv6 packet is converted to an IPv4 header

Figure 20.21 *Header translation strategy*

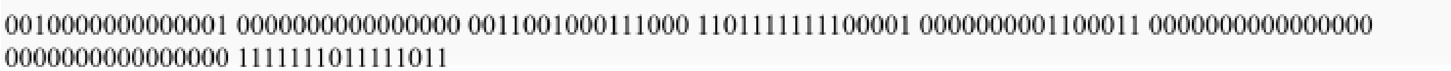


Header translation uses the mapped address to translate an IPv6 address to an IPv4 address.

Address Structure

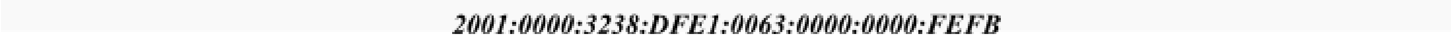
An IPv6 address is made of 128 bits divided into eight 16-bits blocks. Each block is then converted into 4-digit Hexadecimal numbers separated by colon symbols.

For example, given below is a 128 bit IPv6 address represented in binary format and divided into eight 16-bits blocks:



```
0010000000000001 0000000000000000 0011001000111000 1101111111100001 0000000001100011 0000000000000000
0000000000000000 1111111011111011
```

Each block is then converted into Hexadecimal and separated by ***':'*** symbol:

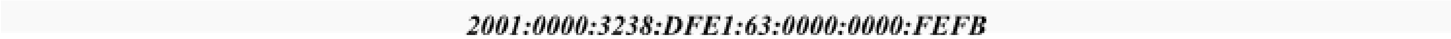


2001:0000:3238:DFE1:0063:0000:0000:FEFB

Even after converting into Hexadecimal format, IPv6 address remains long. IPv6 provides some rules to shorten the address. The rules are as follows:

Rule.1: Discard leading Zero(es):

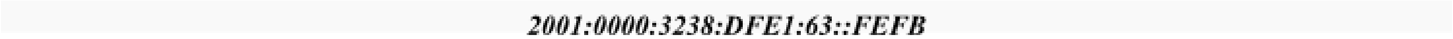
In Block 5, 0063, ***the leading two 0s*** can be omitted, such as (5th block):



2001:0000:3238:DFE1:63:0000:0000:FEFB

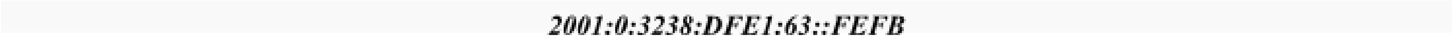
Rule.2: If two of more blocks contain consecutive zeroes, omit them all and replace

with double ***colon sign ::,***
such as (6th and 7th block):



```
2001:0000:3238:DFE1:63::FEFB
```

Consecutive blocks of zeroes can be replaced ***only on ce by : :*** so if there are still blocks of zeroes in the address, they can be shrunk down to a single zero, such as (2nd block):



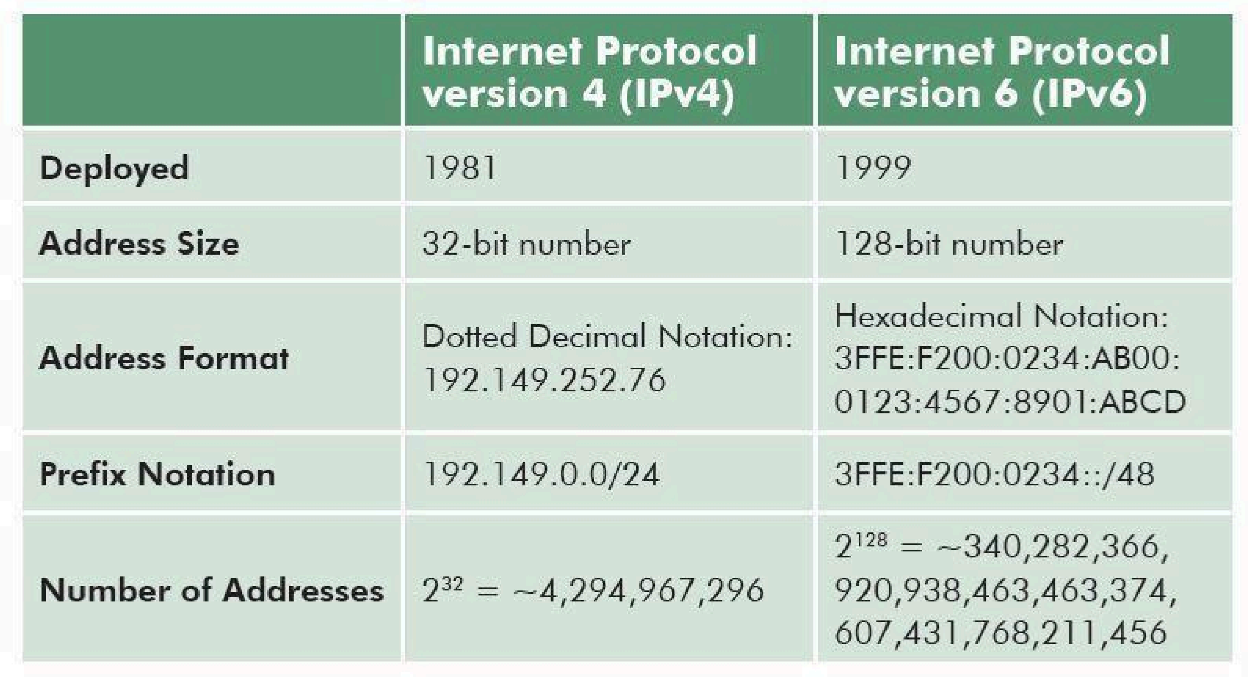
```
2001:0:3238:DFE1:63::FEFB
```

ADVANTAGES OF IPV6

The next-generation IP, or IPv6, has some advantages over IPv4 that can be summarized as follows:

- Larger address space. An IPv6 address is 128 bits long, Compared with the 32-bit address of IPv4, this is a huge increase in the address space.
- Better header format. IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper-layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
- New options. IPv6 has new options to allow for additional functionalities.
- Allowance for extension. IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.
- Support for resource allocation. In IPv6, the type-of-service field has been removed, but a mechanism has been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.
- Support for more security. The encryption and authentication options in IPv6 provide

COMPARISON OF IPV4 AND IPV6 ADDRESS FORMAT



	Internet Protocol version 4 (IPv4)	Internet Protocol version 6 (IPv6)
Deployed	1981	1999
Address Size	32-bit number	128-bit number
Address Format	Dotted Decimal Notation: 192.149.252.76	Hexadecimal Notation: 3FFE:F200:0234:AB00: 0123:4567:8901:ABCD
Prefix Notation	192.149.0.0/24	3FFE:F200:0234::/48
Number of Addresses	2^{32} = ~4,294,967,296	2^{128} = ~340,282,366, 920,938,463,463,374, 607,431,768,211,456

4.7. INTERNET CONTROL PROTOCOL

1. Address Resolution Protocol (ARP) –

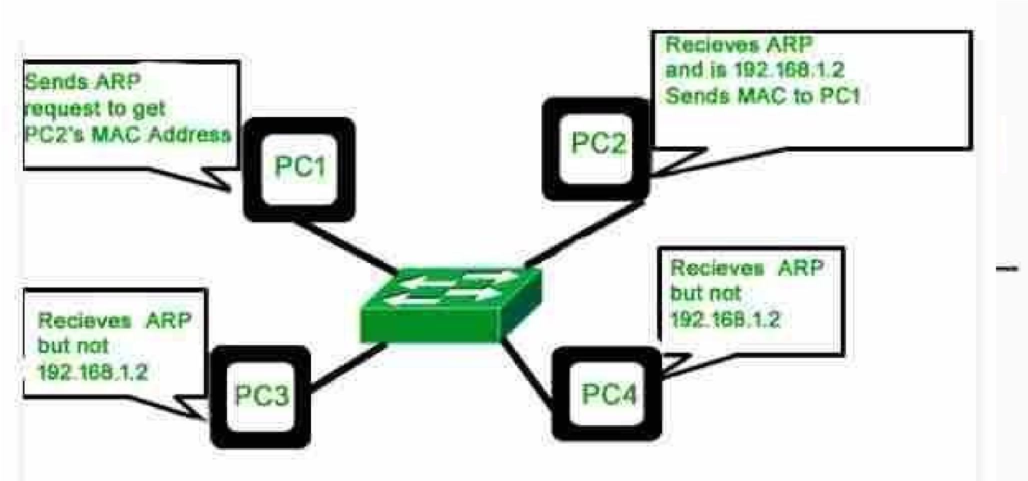
- Address Resolution Protocol is a communication protocol used for discovering physical addresses associated with a given network address.
- Typically, ARP is a network layer to data link layer mapping process, which is used to discover MAC addresses for a given Internet Protocol Address.
- In order to send the data to a destination, having an IP address is necessary but not sufficient; we also need the physical address of the destination machine.
- ARP is used to get the physical address (MAC address) of the destination machine.

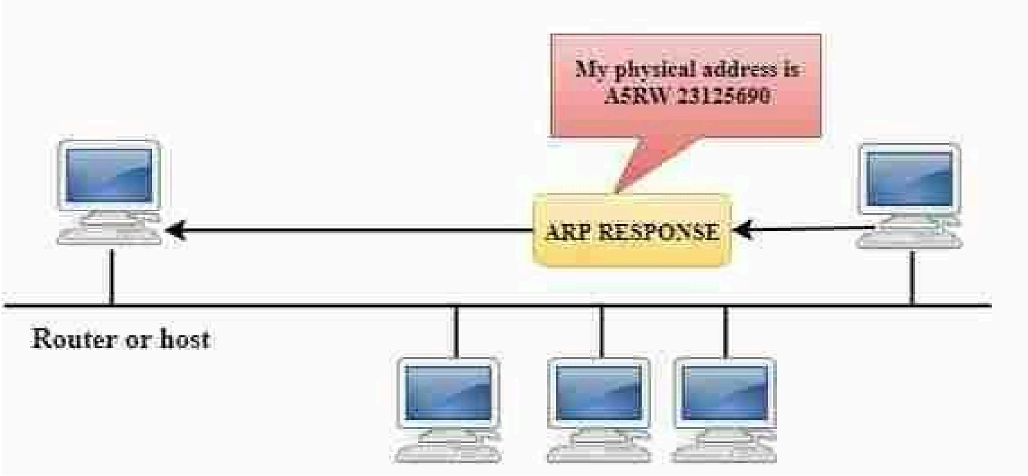
  - It is used to associate an IP address with the MAC address.

  - Each device on the network is recognized by the MAC address imprinted on the NIC. Therefore, we can say that devices need the MAC address for communication on a local area network.ARP is used to find the MAC address of the node when an internet address is known.

How ARP works:

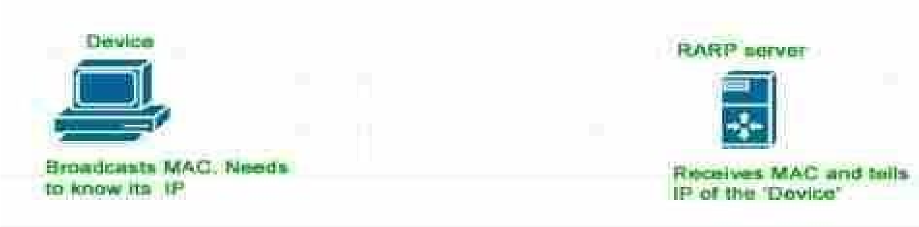
If the host wants to know the physical address of another host on its network, then it sends an ARP query packet that includes the IP address and broadcasts it over the network. Every host on the network receives and processes the ARP packet, but only the intended recipient recognizes the IP address and sends back the physical address. The host holding the datagram adds the physical address to the cache memory and to the datagram header, then sends it back to the sender.

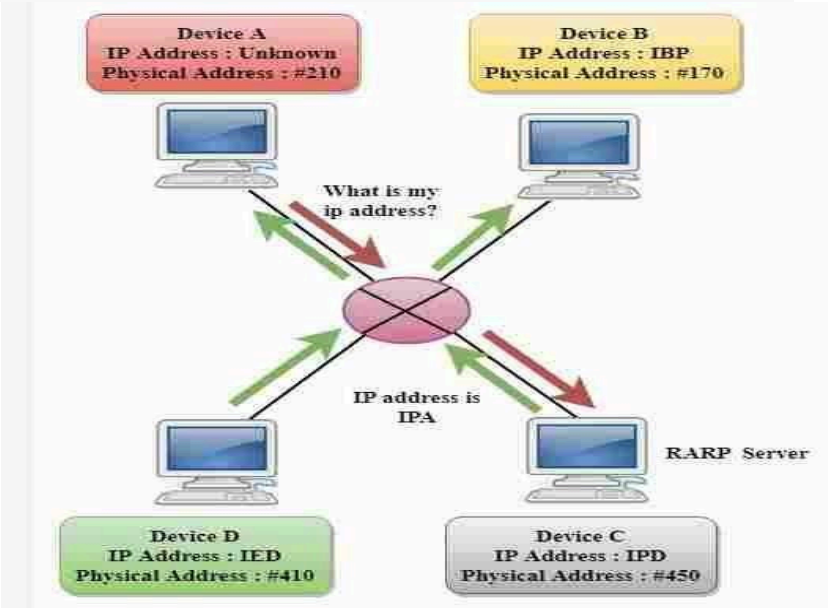

Sends ARP request to get PC2's MAC Address
PC1
Recieves ARP and is 192.168.1.2 Sends MAC to PC1
PC2
Recieves ARP but not 192.168.1.2
PC3
Recieves ARP but not 192.168.1.2
PC4



- **Dynamic entry:** It is an entry which is created automatically when the sender broadcasts its message to the entire network. Dynamic entries are not permanent, and they are removed periodically.
- **Static entry:** It is an entry where someone manually enters the IP to MAC address association by using the ARP command utility.

- **Reverse Address Resolution Protocol (RARP) –**
  - Reverse ARP is a networking protocol used by a client machine in a local area network to request its Internet Protocol address (IPv4) from the gateway-router's ARP table.
  - The network administrator creates a table in gateway-router, which is used to map the MAC address to corresponding IP address.
  - When a new machine is set up or any machine which doesn't have memory to store an IP address, needs an IP address for its own use.
  - So the machine sends a RARP broadcast packet which contains its own MAC address in both the sender and receiver hardware address field.
- If the host wants to know its IP address, then it broadcasts the RARP query packet that contains its physical address to the entire network.
- A RARP server on the network recognizes the RARP packet and responds back with the host IP address.
- The protocol which is used to obtain the IP address from a server is known as **Reverse Address Resolution Protocol**.
- The message format of the RARP protocol is similar to the ARP protocol.
- Like ARP frame, RARP frame is sent from one machine to another encapsulated in the data portion of a frame.


Device
RARP server
Broadcasts MAC. Needs to know its IP
Receives MAC and tells IP of the 'Device'



A special host configured inside the local area network, called a RARP-server is responsible to reply for these kinds of broadcast packets. Now the RARP server attempts to find out the entry in IP to MAC address mapping table. If any entry matches in the table, the RARP server send the response packet to the requesting device along with IP address.

> ***LAN technologies like Ethernet, Ethernet II, Token Ring and Fiber Distributed Data Interface (FDDI) support the Address Resolution Protocol. RARP is not being used in today's networks. Because we have much great featured protocols like BOOTP (Bootstrap Protocol) and DHCP(Dynamic Host Configuration Protocol).***

ICMP

The IP protocol has no error-reporting or error-correcting mechanism. The IP protocol also lacks a mechanism for host and management queries. A host sometimes needs to determine if a router or another host is alive. And sometimes a network administrator needs information from another host or router. The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies. It is a companion to the IP protoco1.

Types of Messages

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

The error-reporting messages report problems that a router or a host (destination) may encounter when it processes an IP packet.

The query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host. For example, nodes can discover their neighbors. Also, hosts can discover and learn about routers on their network, and routers can help a node redirect its messages.

Message Format

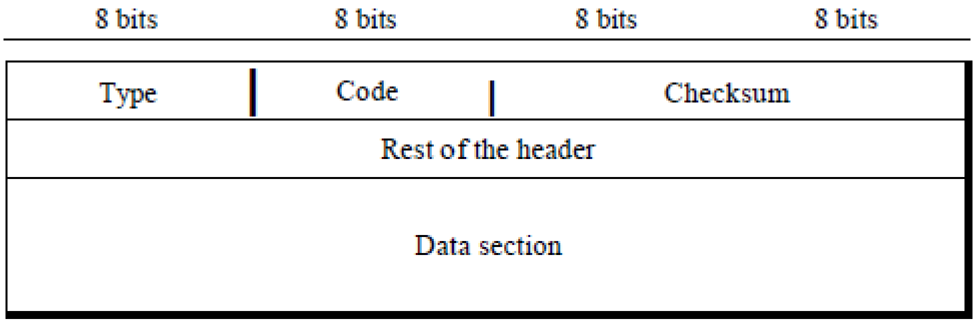
An ICMP message has an 8-byte header and a variable-size data section. Although the general format of the header is different for each message type, the first 4 bytes are common to all.

The first field, ICMP type, defines the type of the message.

The code field specifies the reason for the particular message type.

The last common field is the checksum field. The rest of the header is specific for each message type.

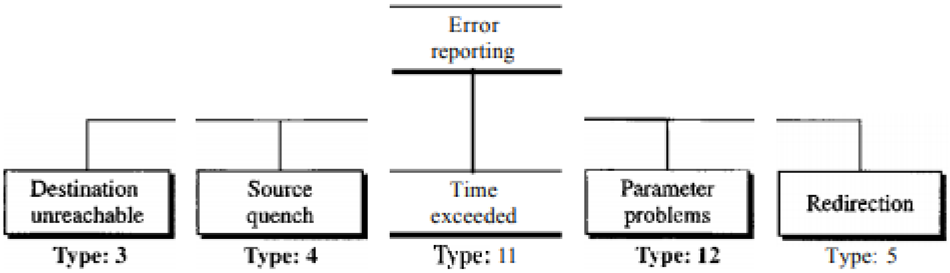
The data section in error messages carries information for finding the original packet that had the error. In query messages, the data section carries extra information based on the type of the query.



Error Reporting

Five types of errors are handled: destination unreachable, source quench, time exceeded, parameter problems, and redirection

Figure 21.9 *Error-reporting messages*



Destination Unreachable

When a router cannot route a datagram or a host cannot deliver a datagram, the datagram is discarded and the router or the host sends a destination-unreachable message back to the source host that initiated the datagram.

Source Quench

When a router or host discards a datagram due to congestion, it sends a source-quench message to the sender of the datagram. This message has two purposes. First, it informs the source that the datagram has been discarded. Second, it warns the source that there is congestion somewhere in the path and that the source should slow down (quench) the sending process.

Time Exceeded

The time-exceeded message is generated in two cases:routers use routing tables to find the next hop (next router) that must receive the packet. If there are errors in one or more routing tables, a packet can travel in a loop or a cycle, going from one router to the next or visiting a series of routers endlessly. When the time-to-live value reaches 0, after decrementing, the router discards the datagram.

Parameter Problem

Any ambiguity in the header part of a datagram can create serious problems as the datagram travels through the Internet. If a router or the destination host discovers an ambiguous or missing value in any field of the datagram, it discards the datagram and sends a parameter-problem message back to the source.

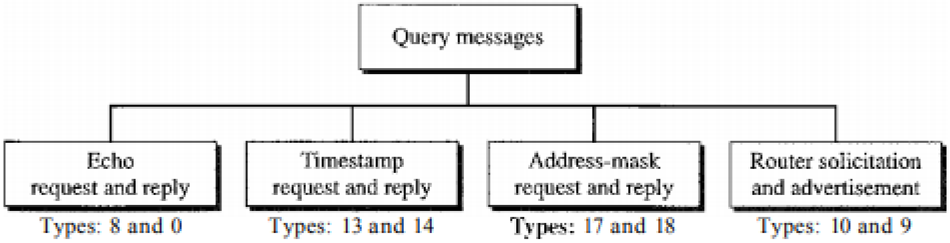
Redirection

When a host comes up, its routing table has a limited number of entries. It usually knows the IP address of only one router, the default router. For this reason, the host may send a datagram, which is destined for another network, to the wrong router. In this case, the router that receives the datagram will forward the datagram to the correct router. However, to update the routing table of the host, it sends a redirection message to the host.

Query Message

In addition to error reporting, ICMP can diagnose some network problems. This is accomplished through the query messages, a group of four different pairs of messages, as shown in Figure 21.12. In this type of ICMP message, a node sends a message that is answered in a specific format by the destination node. A query message is encapsulated in an IP packet, which in turn is encapsulated in a data link layer frame.

Figure 21.12 *Query messages*



Echo Request and Reply

The echo-request and echo-reply messages are designed for diagnostic purposes. The echo-request and echo-reply messages can be used to determine if there is communication at the IP level. Because ICMP messages are encapsulated in IP datagrams, the receipt of an echo-reply message by the machine that sent the echo request is proof that the IP protocols in the sender and receiver are communicating with each other using the IP datagram.

Timestamp Request and Reply

Two machines (hosts or routers) can use the timestamp request and timestamp reply messages to determine the round-trip time needed for an IP datagram to travel between them. It can also be used to synchronize the clocks in two machines.

Address-Mask Request and Reply

A host may know its IP address, but it may not know the corresponding mask. The router receiving the address-mask-request message responds with an address-mask-reply message, providing the necessary mask for the host. This can be applied to its full IP address to get its subnet address.

Router Solicitation and Advertisement

The host must know if the routers are alive and functioning. The router-solicitation and router-advertisement messages can help in this situation. A host can broadcast (or multicast) a router-solicitation message. The router or routers that receive the solicitation message broadcast their routing information using the router-advertisement message. Checksum

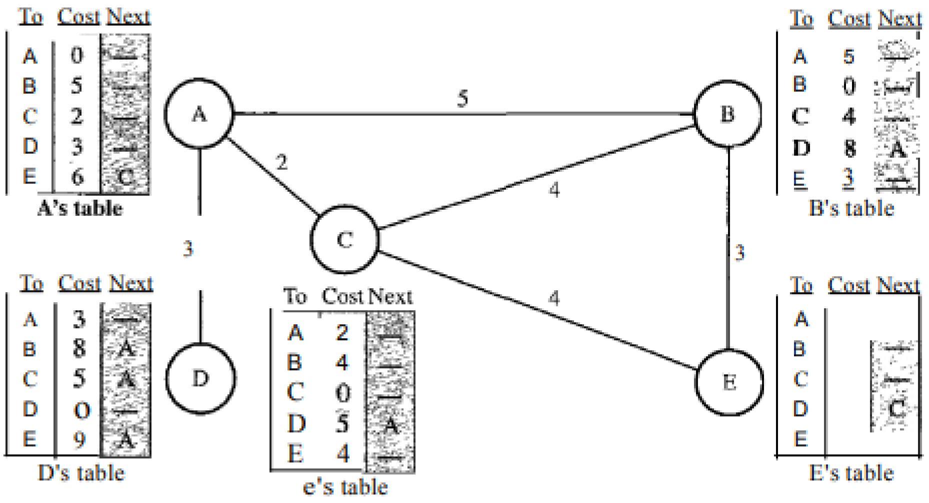
In ICMP the checksum is calculated over the entire message (header and data).

4.6. ROUTING ALGORITHM :

DISTANCE VECTOR ROUTING

In distance vector routing, the least-cost route between any two nodes is the route with minimum distance. In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node. The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing). We can think of nodes as the cities in an area and the lines as the roads connecting them. A table can show a tourist the minimum distance between cities. In Figure 22.14, we show a system of five nodes with their corresponding tables.

Figure 22.14 *Distance vector routing tables*



A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

D's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	O	—
E	9	A

e's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

E's table

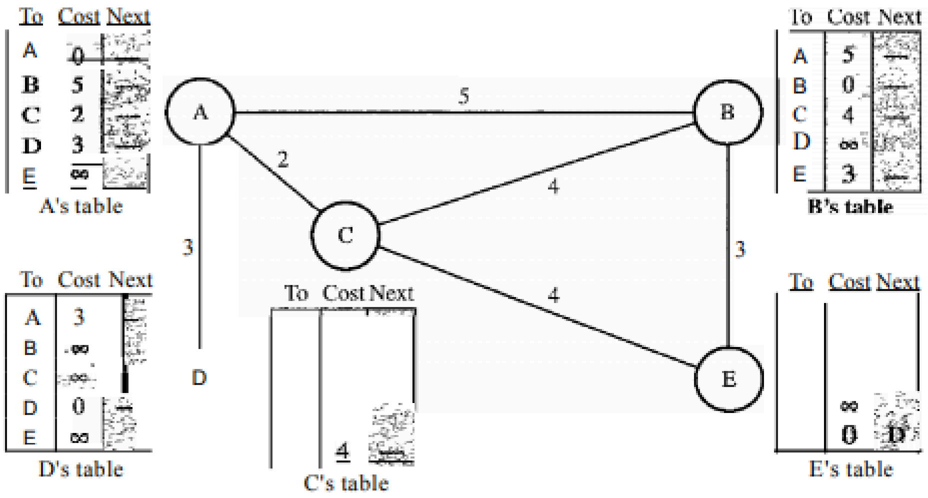
To	Cost	Next
A		
B		—
C		—
D		C
E		

The table for node A shows how we can reach any node from this node. For example, our least cost to reach node E is 6. The route passes through C.

Initialization

The tables in Figure 22.14 are stable; each node knows how to reach any other node and the cost. At the beginning, however, this was not the case. Each node can know only the distance between itself and its immediate neighbors, those directly connected to it. So for the moment, we assume that each node can send a message to the immediate neighbors and find the distance between itself and these neighbors. Figure 22.15 shows the initial tables for each node. The distance for any entry that is not a neighbor is marked as infinite (unreachable).

Figure 22.15 *Initialization oftables in distance vector routing*



To	Cost	Next
A	0	[illegible]
B	5	[illegible]
C	2	[illegible]
D	3	[illegible]
E	∞	[illegible]

A's table

To	Cost	Next
A	5	[illegible]
B	0	[illegible]
C	4	[illegible]
D	∞	[illegible]
E	3	[illegible]

B's table

To	Cost	Next
A	3	[illegible]
B	∞	[illegible]
C	∞	[illegible]
D	0	[illegible]
E	∞	[illegible]

D's table

To	Cost	Next
[illegible]	4	[illegible]

C's table

To	Cost	Next
[illegible]	∞	[illegible]
[illegible]	0	D

E's table

Sharing

The whole idea of distance vector routing is the sharing of information between neighbors. Although node A does not know about node E, node C does. So if node C shares its routing table with A, node A can also know how to reach node E. On the other hand, node C does not know how to reach node D, but node A does. If node A shares its routing table with node C, node C also knows how to reach node D. In other words, nodes A and C, as immediate neighbors, can improve their routing tables if they help each other. There is only one problem. How much of the table must be shared with each neighbor? A node is not aware of a neighbor's table. The best solution for each node is to send its entire table to the neighbor and let the neighbor decide what part to use and what part to discard. However, the third column of a table (next stop) is not useful for the neighbor. When the neighbor receives a table, this column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table. A node therefore can send only the first two columns of its table to any neighbor. In other words, sharing here means sharing only the first two columns

Updating

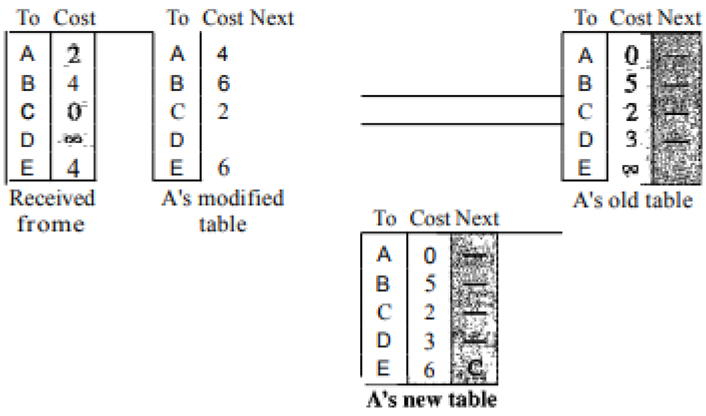
When a node receives a two-column table from a neighbor, it needs to update its routing table. Updating takes three steps:

1. The receiving node needs to add the cost between itself and the sending node to each value in the second column. The logic is clear. If node C claims that its distance to a destination is x mi, and the distance between A and C is y mi, then the distance between A and that destination, via C, is x + y mi.

2. The receiving node needs to add the name of the sending node to each row as the third column if the receiving node uses information from any row. The sending node is the next node in the route. 3. The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table. a. If the next-node entry is different, the receiving node chooses the row with the smaller cost. If there is a tie, the old one is kept. b. If the next-node entry is the same, the receiving node chooses the new row. For example, suppose node C has previously advertised a route to node X with distance

3. Suppose that now there is no path between C and X; node C now advertises this route with a distance of infinity. Node A must not ignore this value even though its old entry is smaller. The old route does not exist anymore. The new route has a distance of infinity. Figure 22.16 shows how node A updates its routing table after receiving the partial table from node C.

Figure 22.16 *Updating in distance vector routing*



To	Cost
A	2
B	4
C	0
D	∞
E	4

Received frome

To	Cost	Next
A	4	
B	6	
C	2	
D		
E	6	

A's modified table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

A's old table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

A's new table

There are several points we need to emphasize here. First, as we know from mathematics, when we add any number to infinity, the result is still infinity. Second, the modified table shows how to reach A from A via C. If A needs to reach itself via C, it needs to go to C and come back, a distance of 4. Third, the only benefit from this updating of node A is the last entry, how to reach E. Previously, node A did not know how to reach E (distance of infinity); now it knows that the cost is 6 via C.

When to Share

The question now is, when does a node send its partial routing table (only two columns) to all its immediate neighbors? The table is sent both periodically and when there is a change in the table. Periodic Update A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing. Triggered Update A node sends its two-column routing table to its neighbors anytime there is a change in its routing table. This is called a triggered update. The change can result from the following.

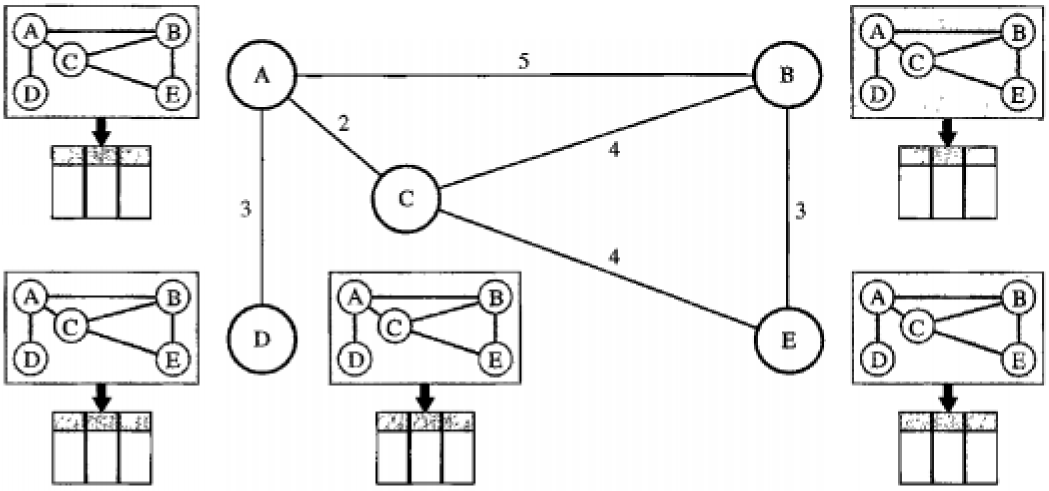
1. A node receives a table from a neighbor, resulting in changes in its own table after updating.

2. A node detects some failure in the neighboring links which results in a distance change to infinity.

LINK STATE ROUTING

Link state routing has a different philosophy from that of distance vector routing. In link state routing, if each node in the domain has the entire topology of the domain the list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down)-the node can use Dijkstra's algorithm to build a routing table. Figure 22.20 shows the concept.

Figure 22.20 *Concept oflink state routing*

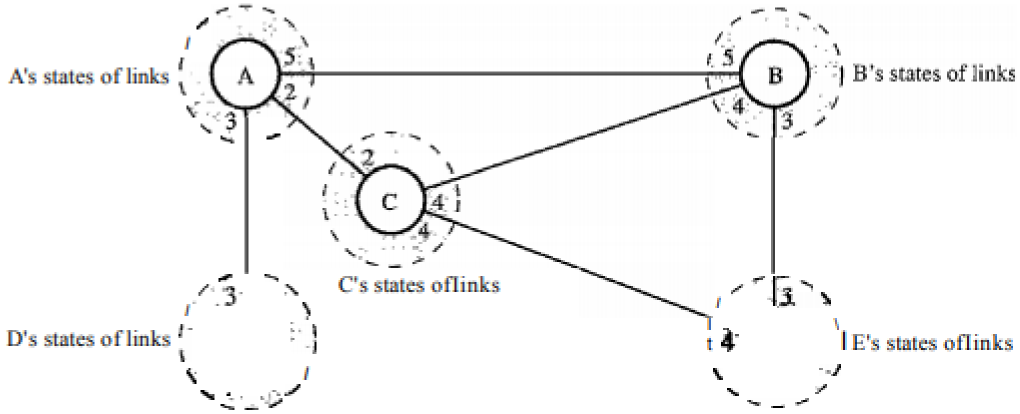


The figure shows a simple domain with five nodes. Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology. This is analogous to a city map. While each person may have the same map, each needs to take a different route to reach her specific destination.

The topology must be dynamic, representing the latest state of each node and each link. If there are changes in any point in the network (a link is down, for example), the topology must be updated for each node. How can a common topology be dynamic and stored in each node? No node can know the topology at the beginning or after a change somewhere in the network.

Link state routing is based on the assumption that, although the global knowledge about the topology is not clear, each node has partial knowledge: it knows the state (type, condition, and cost) of its links. In other words, the whole topology can be compiled from the partial knowledge of each node. Figure 22.21 shows the same domain as in Figure 22.20, indicating the part of the knowledge belonging to each node.

Figure 22.21 *Link state knowledge*



Node A knows that it is connected to node B with metric 5, to node C with metric 2, and to node D with metric 3. Node C knows that it is connected to node A with metric 2, to node B with metric 4, and to node E with metric 4. Node D knows that it is connected only to node A with metric 3. And so on. Although there is an overlap in the knowledge, the overlap guarantees the creation of a common topology-a picture of the whole domain for each node.

Building Routing Tables

In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.

1. Creation of the states of the links by each node, called the link state packet (LSP).

2. Dissemination of LSPs to every other router, called flooding, in an efficient and reliable way. 3. Formation of a shortest path tree for each node.

4. Calculation of a routing table based on the shortest path tree. Creation of Link State Packet (LSP) A link state packet can carry a large amount of information. For the moment, however, we assume that it carries a minimum amount of data: the node identity, the list of links, a sequence number, and age. The first two, node identity and the list of links, are needed to make the topology. The third, sequence number, facilitates flooding and distinguishes new LSPs from old ones. The fourth, age, prevents old LSPs from remaining in the domain for a long time.

LSPs are generated on two occasions:

1. When there is a change in the topology of the domain. Triggering of LSP dissemination is the main way of quickly informing any node in the domain to update its topology.

2. On a periodic basis. The period in this case is much longer compared to distance vector routing. As a matter of fact, there is no actual need for this type of LSP dissemination. It is done to ensure that old information is removed from the domain. The timer set for periodic dissemination is normally in the range of 60 min or 2 h based on the implementation. A longer period ensures that flooding does not create too much traffic on the network.

Flooding of LSPs

After a node has prepared an LSP, it must be disseminated to all other nodes, not only to its neighbors. The process is called flooding and based on the following:

1. The creating node sends a copy of the LSP out of each interface.

2. A node that receives an LSP compares it with the copy it may already have. If the newly arrived LSP is older than the one it has (found by checking the sequence number), it discards the LSP. If it is newer, the node does the following:

a. It discards the old LSP and keeps the new one.

b. It sends a copy of it out of each interface except the one from which the packet arrived. This guarantees that flooding stops somewhere in the domain (where a node has only one interface).

Formation of Shortest Path Tree: Dijkstra Algorithm

After receiving all LSPs, each node will have a copy of the whole topology. However, the topology is not sufficient to find the shortest path to every other node;

a shortest path tree is needed. A tree is a graph of nodes and links; one node is called the root. All other nodes can be reached from the root through only one single route. A shortest path tree is a tree in which the path between the root and every other node is the shortest. What we need for each node is a shortest path tree with that node as the root.

The Dijkstra algorithm creates a shortest path tree from a graph. The algorithm divides the nodes into two sets: tentative and permanent. It finds the neighbors of a current node, makes them tentative, examines them, and if they pass the criteria, makes them permanent. We can informally define the algorithm by using the flowchart in Figure 22.22. Let us apply the algorithm to node A of our sample graph in Figure 22.23. To find the shortest path in each step, we need the cumulative cost from the root to each node, which is shown next to the node. The following shows the steps. At the end of each step, we show the permanent (filled circles) and the tentative (open circles) nodes and lists with the cumulative costs.

Figure 22.22 *Dijkstra algorithm*

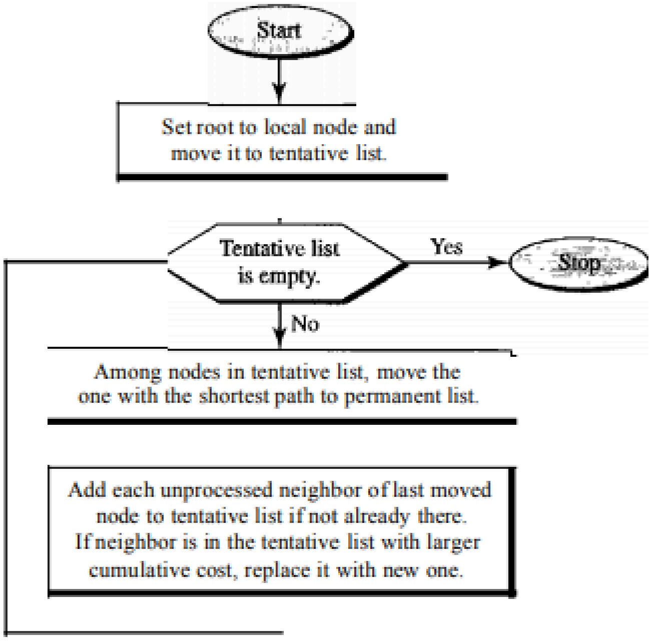
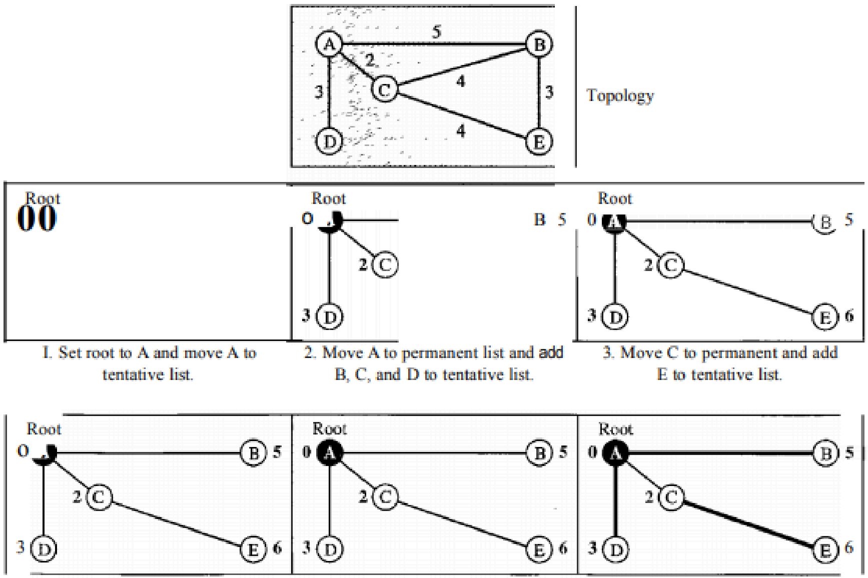

Start
Set root to local node and move it to tentative list.
Tentative list is empty.
Yes
Stop
No
Among nodes in tentative list, move the one with the shortest path to permanent list.
Add each unprocessed neighbor of last moved node to tentative list if not already there. If neighbor is in the tentative list with larger cumulative cost, replace it with new one.

Figure 22.23 *Example offormation ofshortest path tree*



1. Set root to A and move A to tentative list.

2. Move A to permanent list and add B, C, and D to tentative list.

3. Move C to permanent and add E to tentative list.

4. Move D to permanent list.

5. Move B to permanent list.

6. Move E to permanent list (tentative list is empty).

1. We make node A the root of the tree and move it to the tentative list.

Our two lists are Permanent list: empty Tentative list: A(O)

2. Node A has the shortest cumulative cost from all nodes in the tentative list. We move A to the permanent list and add all neighbors of A to the tentative list. Our new lists are Permanent list: A(O) Tentative list: B(5), C(2), D(3)

3. Node C has the shortest cumulative cost from all nodes in the tentative list. We move C to the permanent list. Node C has three neighbors, but node A is already processed, which makes the unprocessed neighbors just B and E. However, B is already in the tentative list with a cumulative cost of 5. Node A could also reach node B through C with a cumulative cost of 6. Since 5 is less than 6, we keep node B with a cumulative cost of 5 in the tentative list and do not replace it. Our new lists are Permanent list: A(O), e(2) Tentative list: B(5), 0(3), E(6)

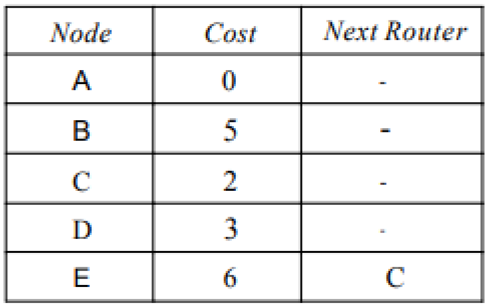
4. Node D has the shortest cumulative cost of all the nodes in the tentative list. We move D to the permanent list. Node D has no unprocessed neighbor to be added to the tentative list. Our new lists are Permanent list: A(O), C(2), 0(3) Tentative list: B(5), E(6)

5. Node B has the shortest cumulative cost of all the nodes in the tentative list. We move B to the permanent list. We need to add all unprocessed neighbors of B to the tentative list (this is just node E). However, E(6) is already in the list with a smaller cumulative cost. The cumulative cost to node E, as the neighbor of B, is 8. We keep node E(6) in the tentative list. Our new lists are Permanent list: A(O), B(5), C(2), 0(3) Tentative list: E(6)

6. Node E has the shortest cumulative cost from all nodes in the tentative list. We move E to the permanent list. Node E has no neighbor. Now the tentative list is empty. We stop; our shortest path tree is ready. The final lists are Permanent list: A(O), B(5), C(2), D(3), E(6) Tentative list: empty

Calculation of Routing Table from Shortest Path Tree Each node uses the shortest path tree protocol to construct its routing table. The routing table shows the cost of reaching each node from the root. Table 22.2 shows the routing table for node A.

Table 22.2 *Routing table for node A*



Node	*Cost*	*Next Router*
A	0	-
B	5	-
C	2	-
D	3	-
E	6	C

Compare Table 22.2 with the one in Figure 22.14. Both distance vector routing and link state routing end up with the same routing table for node A.

4.8. ROUTING PROTOCOLS

OPEN SHORTEST PATH FIRST (OSPF)

Open Shortest Path First (OSPF) is an active routing protocol used in internet protocol. Particularly it is a link state routing protocol and includes into the group of interior gateway protocol. Open Shortest Path First (OSPF) operating inside a distinct autonomous system.

Routers connect networks using the Internet Protocol (IP), and OSPF (Open Shortest Path First) is a router protocol used to find the best path for packets as they pass through a set of connected networks. OSPF is designated by the Internet Engineering Task Force (IETF) as one of several Interior Gateway Protocols (IGPs) -- that is, protocols aimed at traffic moving around within a larger autonomous system network like a single enterprise's network, which may in turn be made up of many separate local area networks linked through routers.

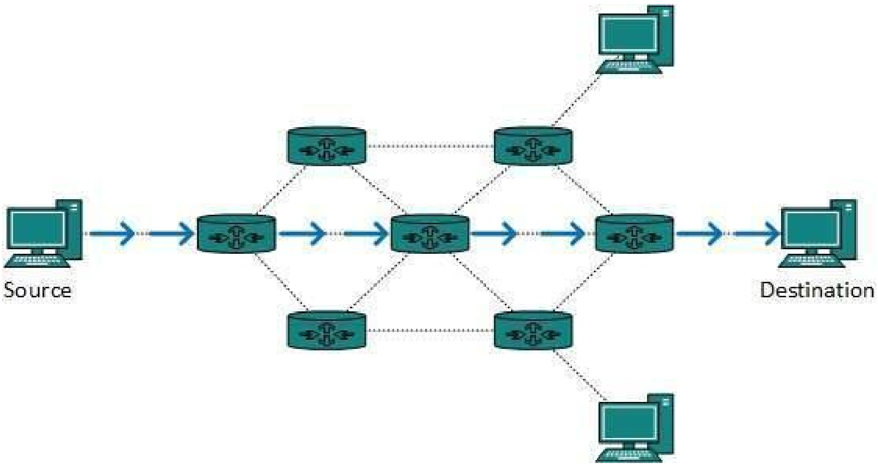
The OSPF routing protocol has largely replaced the older Routing Information Protocol (RIP) in corporate networks. Using OSPF, a router that learns of a change to a routing table (when it is reconfigured by network staff, for example) or detects a change in the network immediately multicasts the information to all other OSPF hosts in the network so they will all have the same routing table information. Unlike RIP, which requires routers to send the entire routing table to neighbors every 30 seconds, OSPF sends only the part that has changed and only when a change has taken place? When routes change -- sometimes due to equipment failure -- the time it takes OSPF routers to find a new path between endpoints with no loops (which is called "open") and that minimizes the length of the path is called the convergence time.

BORDER GATEWAY PROTOCOL (BGP)

Border Gateway Protocol (BGP) are the core routing protocol of the internet and responsible to maintain a table of Internet protocol networks which authorize network reaching capability between AS. The Border Gateway Protocol (BGP) expressed as path vector protocol. BGP router maintains a standard routing table used to direct packets in transit. This table is used in conjunction with a separate routing table, known as the routing information base (RIB), which is a data table stored on a server on the BGP router. The RIB contains route information both from directly connected external peers, as well as internal peers, and continually updates the routing table as changes occurs. BGP is based on TCP/IP and uses client-server topology to communicate routing information, with the client-server initiating a BGP session by sending a request to the server.

UNICAST ROUTING

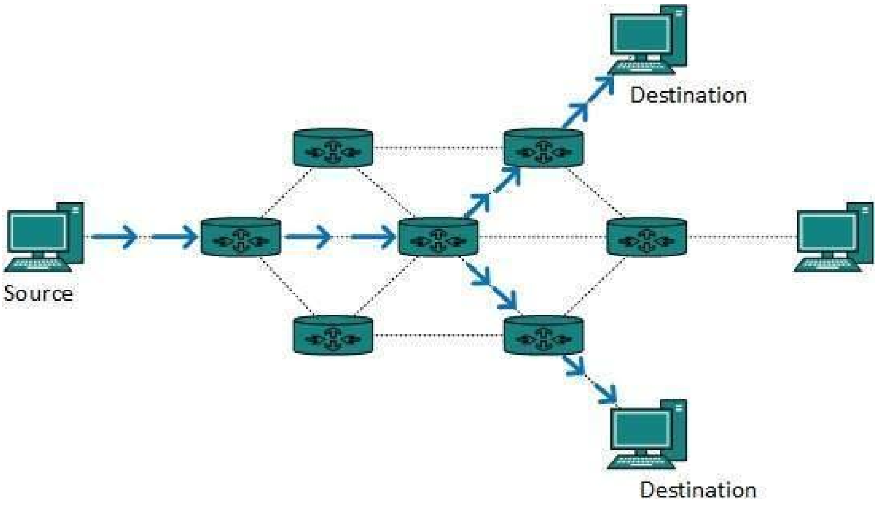
Most of the traffic on the internet and intranets known as unicast data or unicast traffic is sent with specified destination. Routing unicast data over the internet is called unicast routing. It is the simplest form of routing because the destination is already known. Hence the router just has to look up the routing table and forward the packet to next hop.



MULTICAST ROUTING

Multicast routing is special case of broadcast routing with significance difference and challenges. In broadcast routing, packets are sent to all nodes even if they do not want it.

But in Multicast routing, the data is sent to only nodes which wants to receive the packets.

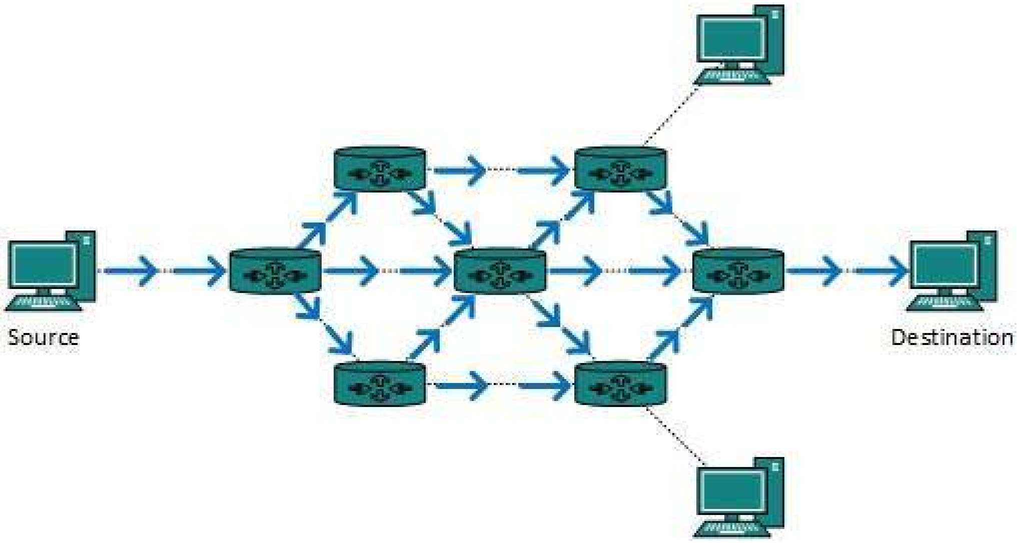


The router must know that there are nodes, which wish to receive multicast packets (or stream) then only it should forward. Multicast routing works spanning tree protocol to avoid looping.

Multicast routing also uses reverse path Forwarding technique, to detect and discard duplicates and loops.

BROADCAST ROUTING

- In broadcast routing, packets are sent to all nodes even if they do not want it.
- Routers create broadcast domains. But it can be configured to forward broadcasts in some special cases. A broadcast message is destined to all network devices.
- A router creates a data packet and then sends it to each host one by one. In this case, the router creates multiple copies of single data packet with different destination addresses.


Source
Destination