Computer Network

Lecture taken by

Dharmendra Kumar

(Associate Professor)

United College of Engineering and Research, Prayagraj

Unit-3

Logical Addressing

IPv4 ADDRESSES

An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.

Note: Two devices on the Internet can never have the same address at the same time.

Note: If a device operating at the network layer has m connections to the Internet, then it needs to have m addresses. Router is such a device that uses many addresses.

Logical Addressing

Address Space

- A protocol such as IPv4 that defines addresses has an address space. An address space is the total number of addresses used by the protocol. If a protocol uses N bits to define an address, the address space is 2^N values.
- IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than 4 billion).

Logical Addressing

Notations

There are two prevalent notations to show an IPv4 address: binary notation and dotted decimal notation.

Binary Notation

In binary notation, the IPv4 address is displayed as 32 bits. Each octet is often referred to as a byte.

The following is an example of an IPv4 address in binary notation:

01110101 10010101 00011101 00000010

Dotted-Decimal Notation

The following is the dotted-decimal notation of the above address:

117.149.29.2

Logical Addressing

Example

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

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

a. 129.11.11.239

b. 193.131.27.255

Logical Addressing

Example

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

a. 111.56.45.78

b. 221.34.7.82

Solution

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Logical Addressing

Example

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

a. 111.56.045.78

b. 221.34.7.8.20

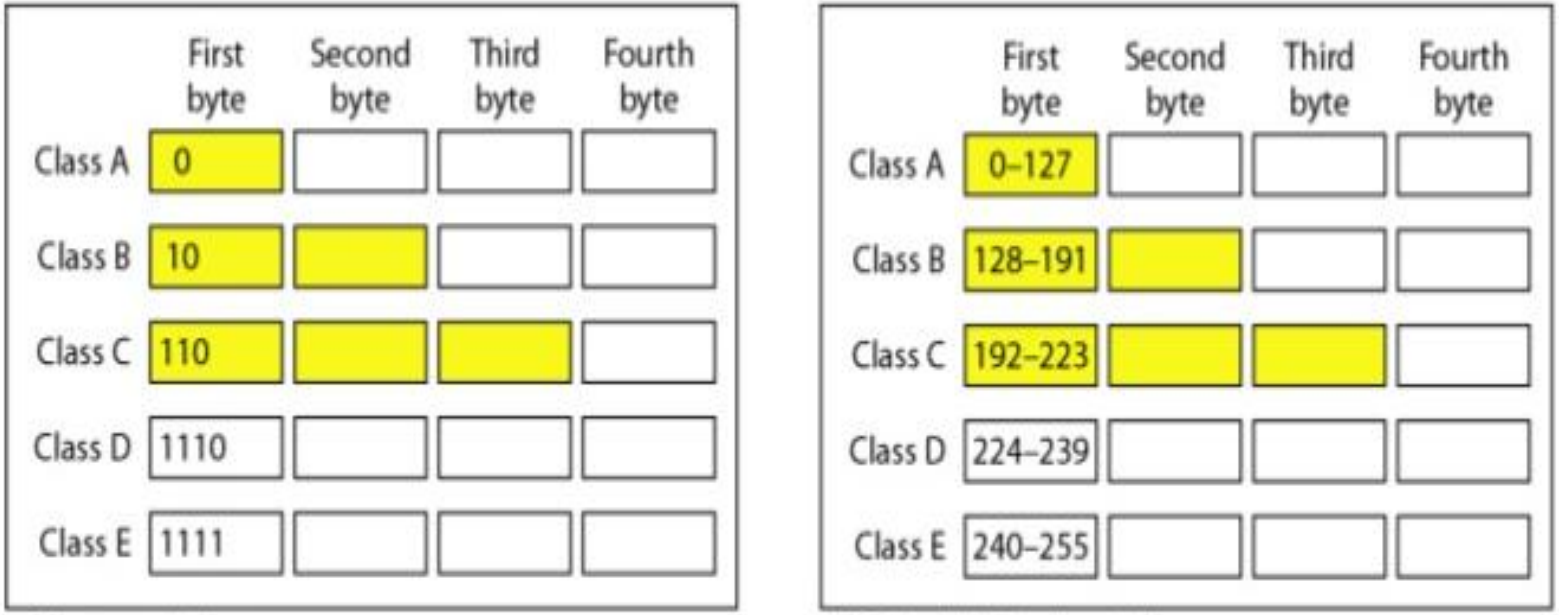
c. 75.45.301.14

d. 11100010.23.14.67

Logical Addressing

Classful Addressing

In classful addressing, the address space is divided into five classes: A, B, C, D, and E. Each class occupies some part of the address space.



	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10	▨		
Class C	110	▨	▨	
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0–127			
Class B	128–191	▨		
Class C	192–223	▨	▨	
Class D	224–239			
Class E	240–255			

b. Dotted-decimal notation

Logical Addressing

Example

Find the class of each address.

(a) 00000001 00001011 00001011 11101111

(b) 11000001 10000011 00011011 11111111

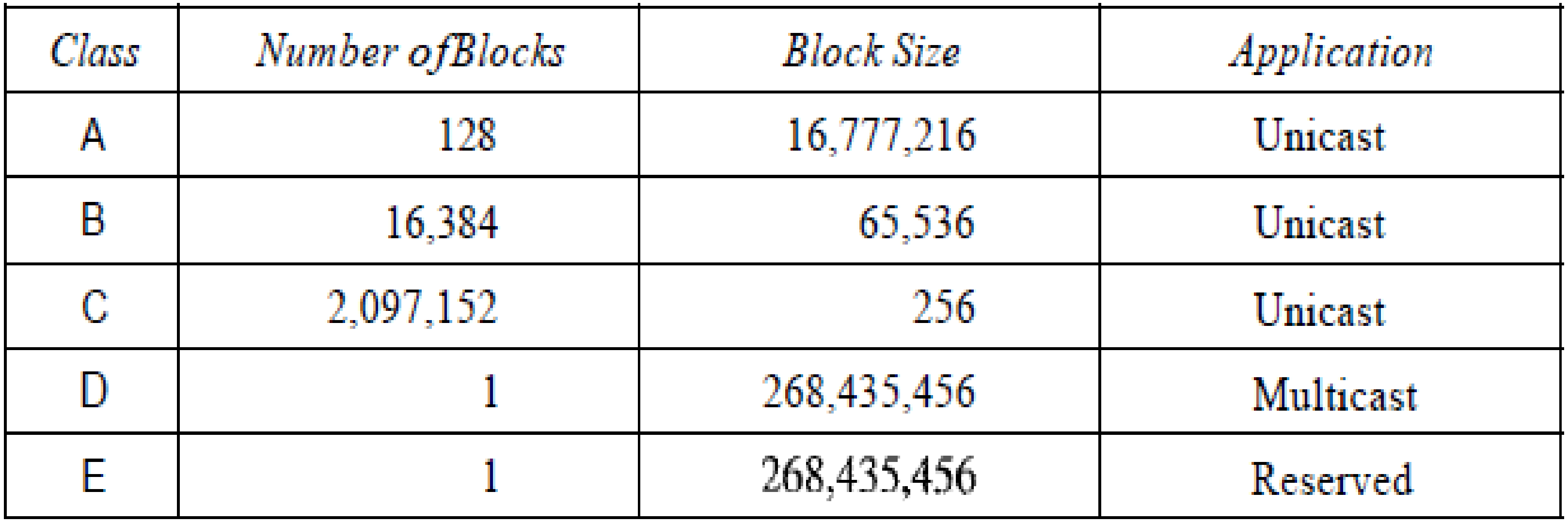
(c) 14.23.120.8

(d) 252.5.15.111

Logical Addressing

Classes and Blocks

One problem with classful addressing is that each class is divided into a fixed number of blocks with each block having a fixed size as shown in the following table:-



Class	*Number ofBlocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

Note: In classful addressing, a large part of the available addresses were wasted.

Logical Addressing

Netid and Hostid

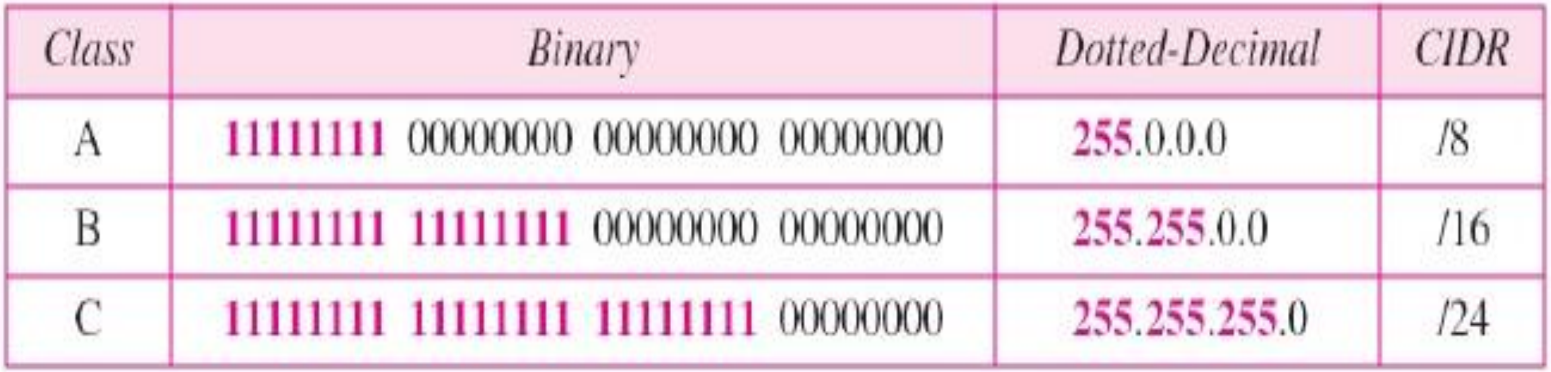
In classful addressing, an IP address in class A, B, or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address.

- In class A, one byte defines the netid and three bytes define the hostid.
- In class B, two bytes define the netid and two bytes define the hostid.
- In class C, three bytes define the netid and one byte defines the hostid.

Logical Addressing

Mask

Mask is a 32-bit number made of contiguous 1's followed by contiguous 0's. The masks for classes A, B, and C are shown in the following table. The concept does not apply to classes D and E.



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

The mask is used to find the netid and the hostid. For example, the mask for a class A address has eight 1s, which means the first 8 bits of any address in class A define the netid; the next 24 bits define the hostid.

Logical Addressing

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

Logical Addressing

Subnetting

- If an organization was granted a large block in class A or B, it could divide the addresses into several contiguous groups and assign each group to smaller networks called subnets.
- Subnetting increases the number of 1’s in the mask.

Logical Addressing

Supernetting

- In supernetting, an organization can combine several class C blocks to create a larger range of addresses. In other words, several networks are combined to create a super network or a supernet. An organization can apply for a set of class C blocks instead of just one.

- For example, an organization that needs 1000 addresses can be granted four contiguous class C blocks. The organization can then use these addresses to create one super network.
- Supernetting decreases the number of 1’s in the mask. For example, if an organization is given four class C addresses, the mask changes from /24 to /22.

Logical Addressing

Classless Addressing

In this scheme, there are no classes, but the addresses are still granted in blocks.

Address Blocks

- In classless addressing, when an entity, small or large, needs to be connected to the Internet, it is granted a block of addresses. The size of the block (the number of addresses) varies based on the nature and size of the entity.
- For example, a household may be given only two addresses; a large organization may be given thousands of addresses. An ISP may be given thousands or hundreds of thousands based on the number of customers it may serve.

Logical Addressing

Restriction

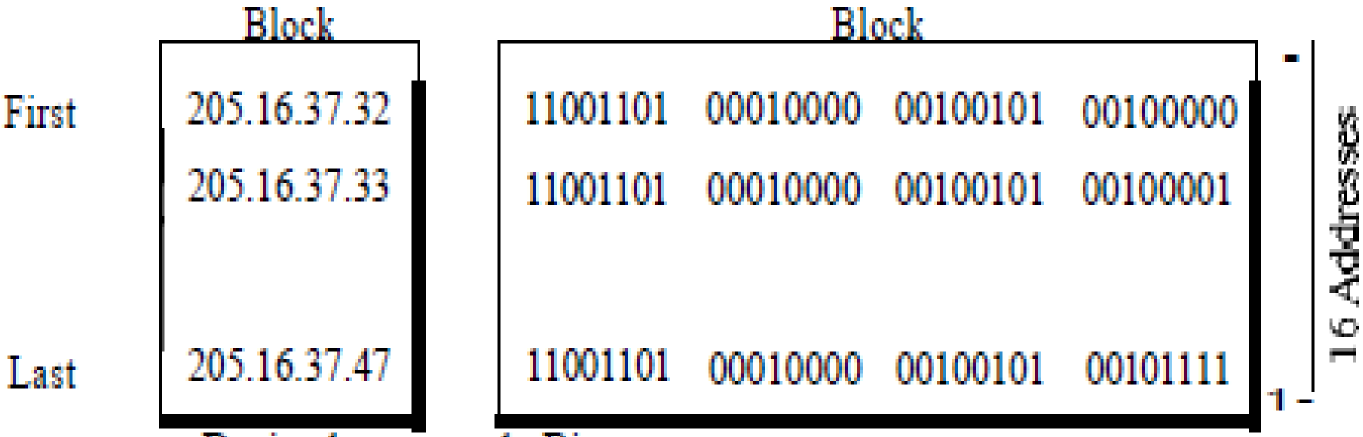
The Internet authorities impose three restrictions on classless address blocks:

1. The addresses in a block must be contiguous, one after another.
2. The number of addresses in a block must be a power of 2.
3. The first address must be evenly divisible by the number of addresses.

Logical Addressing

Example

Following figure shows a block of addresses, in both binary and dotted-decimal notation, granted to a small business that needs 16 addresses.



	Block	Block
First	205.16.37.32	11001101 00010000 00100101 00100000
	205.16.37.33	11001101 00010000 00100101 00100001
Last	205.16.37.47	11001101 00010000 00100101 00101111

a. Decimal

b. Binary

Logical Addressing

Mask

- A better way to define a block of addresses is to select any address in the block and the mask.

- In IPv4 addressing, a block of addresses can be defined as

  x.y.z.t/n

  in which x.y.z.t defines one of the addresses and the /n defines the mask.

- The address and the /n notation completely define the whole block (the first address, the last address, and the number of addresses).

Logical Addressing

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 0s.

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 1s.

Number of Addresses

The number of addresses in the block is the difference between the last and first address. It can easily be found using the formula $\mathbf{2^{32-n}}$.

Logical Addressing

Example

A block of addresses is granted to a small organization. We know that one of the addresses is

205.16.37.39/28.

(a) What is the first address in the block?

(b) What is the last address in the block?

(c) Find the number of addresses in this block.

Logical Addressing

Network Addresses

A very important concept in IP addressing is the network address.

The first address is called the network address and defines the organization itself to the rest of the world.

Logical Addressing

Hierarchy

IP addresses have levels of hierarchy.

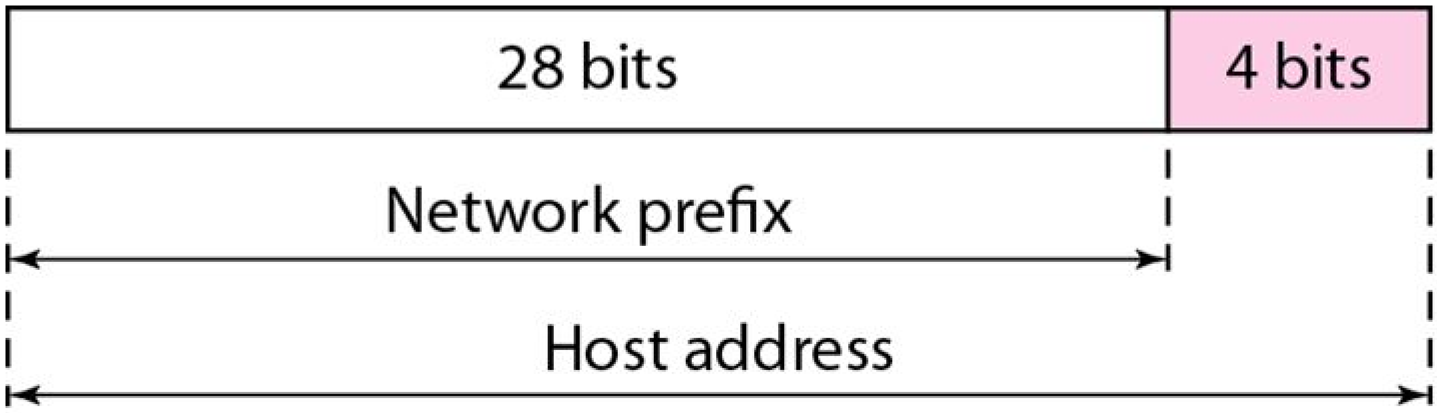
Two-Level Hierarchy: No Subnetting

- An IP address can define only two levels of hierarchy when not subnetted. The n leftmost bits of the address x.y.z.t/n define the network (organization network); the 32–n rightmost bits define the particular host (computer or router) to the network.
- The two common terms are prefix and suffix. The part of the address that defines the network is called the prefix; the part that defines the host is called the suffix.

Logical Addressing

Following figure shows the hierarchical structure of an IPv4 address.

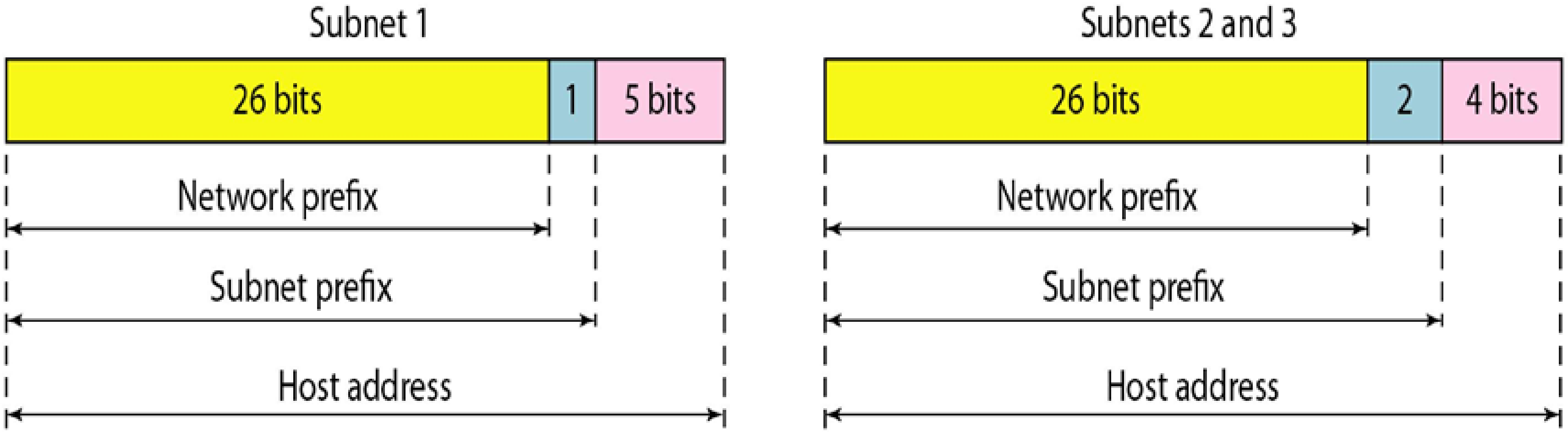
Two levels of hierarchy in an IPv4 address



Logical Addressing

Three-Levels of Hierarchy: Subnetting

An organization that is granted a large block of addresses may want to create clusters of networks (called subnets) and divide the addresses between the different subnets. The rest of the world still sees the organization as one entity; however, internally there are several subnets. The organization has its own mask; each subnet must also have its own.



Logical Addressing

Example:

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 subblocks of 32, 16, and 16 addresses. Find the mask of these three offices.

Solution:

(a) Suppose the mask for the first office is n1. Therefore, $2^{32-n1} = 32$, n1 = 27.

(b) Suppose the mask for the first office is n2. Therefore, 232-n2 = 16 , n2 = 28.

(c) Suppose the mask for the first office is n3. Therefore, 232-n3 = 16 , n3 = 28.

Logical Addressing

Example

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

(a) The first group has 64 customers; each needs 256 addresses.

(b) The second group has 128 customers; each needs 128 addresses.

(c) The third group has 128 customers; each needs 64 addresses.

Design the subblocks and find out how many addresses are still available after these allocations.

Logical Addressing

Solution:

(a) Group 1

For this group, each customer needs 256 addresses. This means that 8 bits are needed to define each host. The prefix length is then 32 - 8 = 24. The addresses are

1st Customer: 190.100.0.0/24 to 190.100.0.255/24

2nd Customer: 190.100.1.0/24 to 190.100.1.255/24

………………………………………………………………………

………………………………………………………………………

64th Customer: 190.100.63.0/24 to 190.100.63.255/24

Total = 64 X 256 =16,384

Logical Addressing

Solution:

(b) Group 2

For this group, each customer needs 128 addresses. This means that 7 bits are needed to define each host. The prefix length is then 32 - 7 = 25. The addresses are

1st Customer: 190.100.64.0/25 to 190.100.64.127/25

2nd Customer: 190.100.64.128/25 to 190.100.64.255/25

………………………………………………………………………….

………………………………………………………………………….

128th Customer: 190.100.127.128/25 to 190.100.127.255/25

Total = 128 X 128 =16,384

Logical Addressing

Solution:

(c) Group 3

For this group, each customer needs 64 addresses. This means that 6 bits are needed to define each host. The prefix length is then 32 - 6 = 26. The addresses are

1st Customer: 190.100.128.0/26 to 190.100.128.63/26

2nd Customer: 190.100.128.64/26 to 190.100.128.127/26

……………………………………………………………………………..

……………………………………………………………………………..

64th Customer: 190.100.159.192/26 to 190.100.159.255/26

Total = 128 X 64 = 8192

Number of granted addresses to the ISP = 65536

Number of allocated addresses by the ISP = 16,384 + 16,384 + 8192

= 40960

Number of available addresses = 65536 – 40960

= **24576**

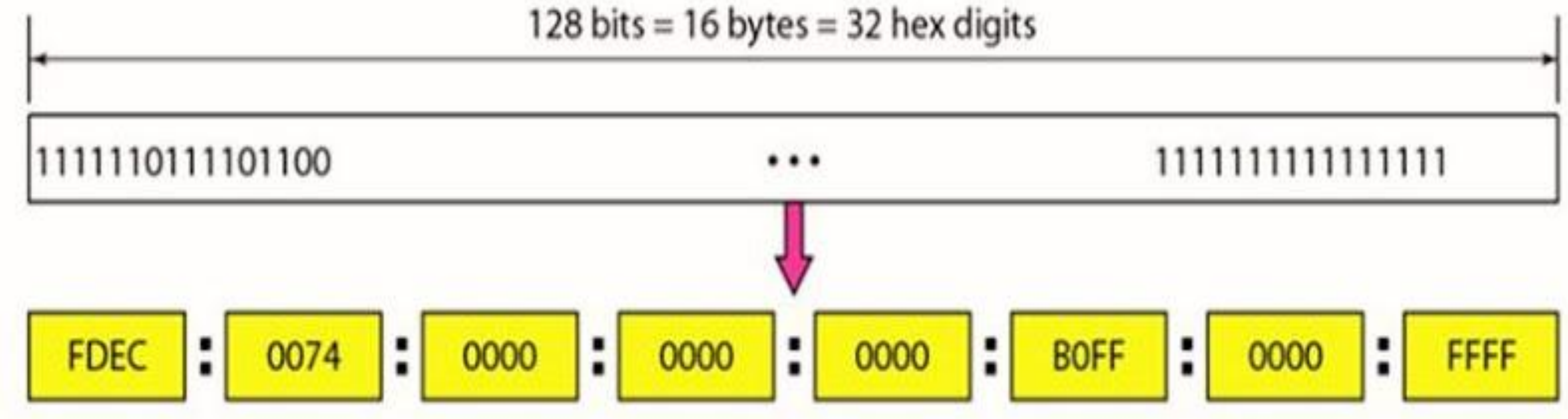
Logical Addressing

IPv6 ADDRESSES

An IPv6 address consists of 16 bytes (octets) i.e. it is 128 bits long.

Hexadecimal Colon Notation

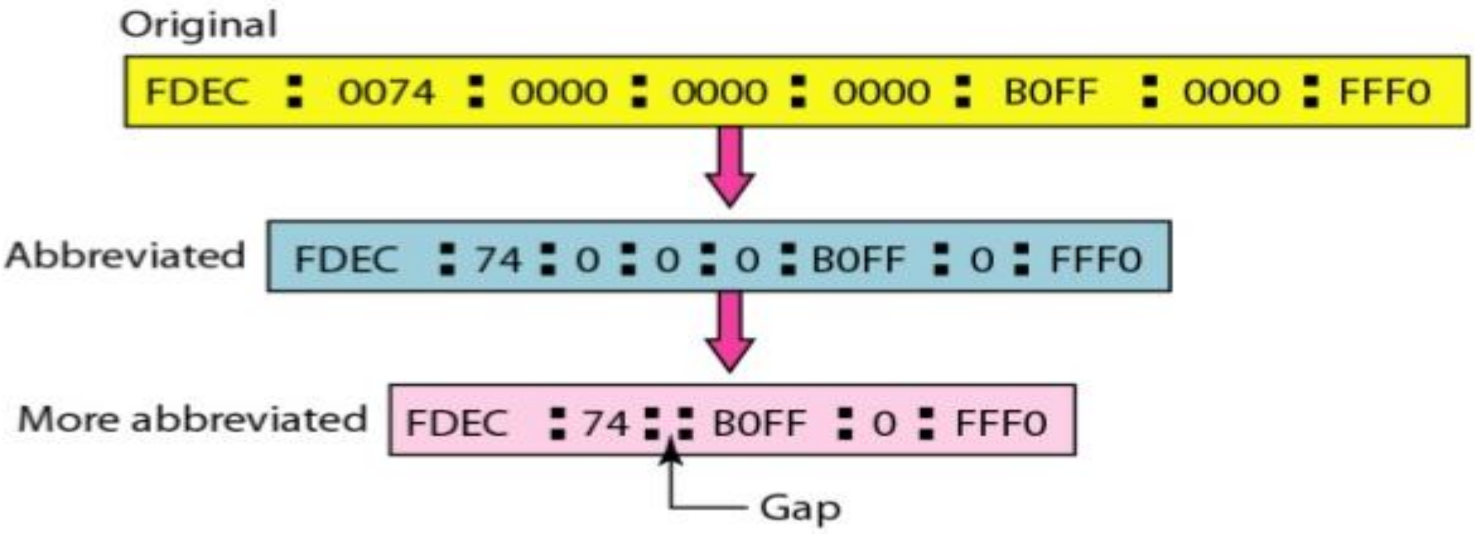
In this notation, 128 bits is divided into eight sections, each 2 bytes in length. Two bytes in hexadecimal notation requires four hexadecimal digits. Therefore, the address consists of 32 hexadecimal digits, with every four digits separated by a colon, as shown in figure:-



Logical Addressing

Abbreviation

- The leading zeros of a section can be omitted.
- If there are consecutive sections consisting of zeros only. We can remove the zeros altogether and replace them with a double colon. Note that this type of abbreviation is allowed only once per address.



Original
FDEC : 0074 : 0000 : 0000 : 0000 : B0FF : 0000 : FFF0
Abbreviated
FDEC : 74 : 0 : 0 : 0 : B0FF : 0 : FFF0
More abbreviated
FDEC : 74 : : B0FF : 0 : FFF0
Gap

Logical Addressing

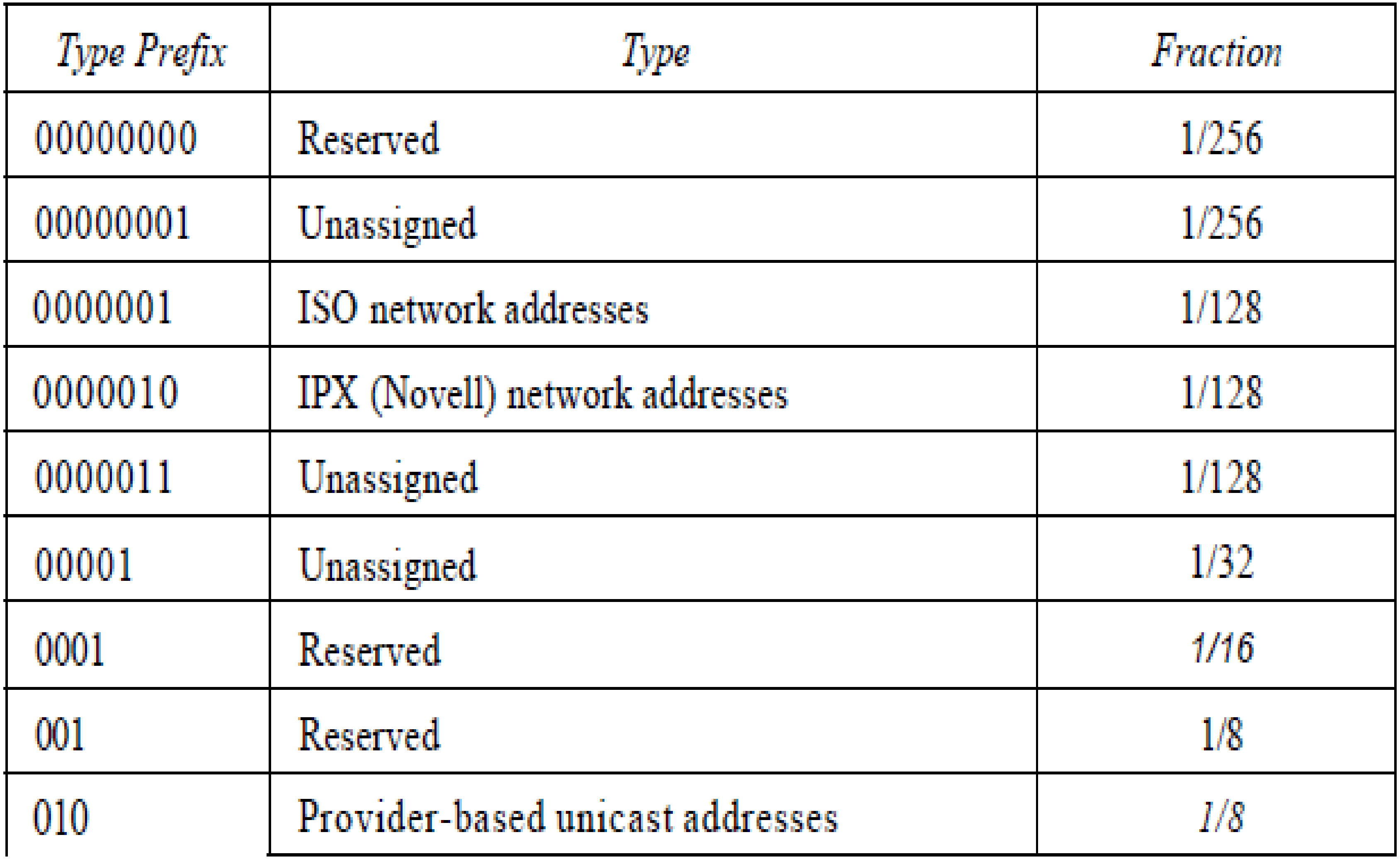
Example

Expand the address 0:15::1:12:1213 to its original.

Address Space

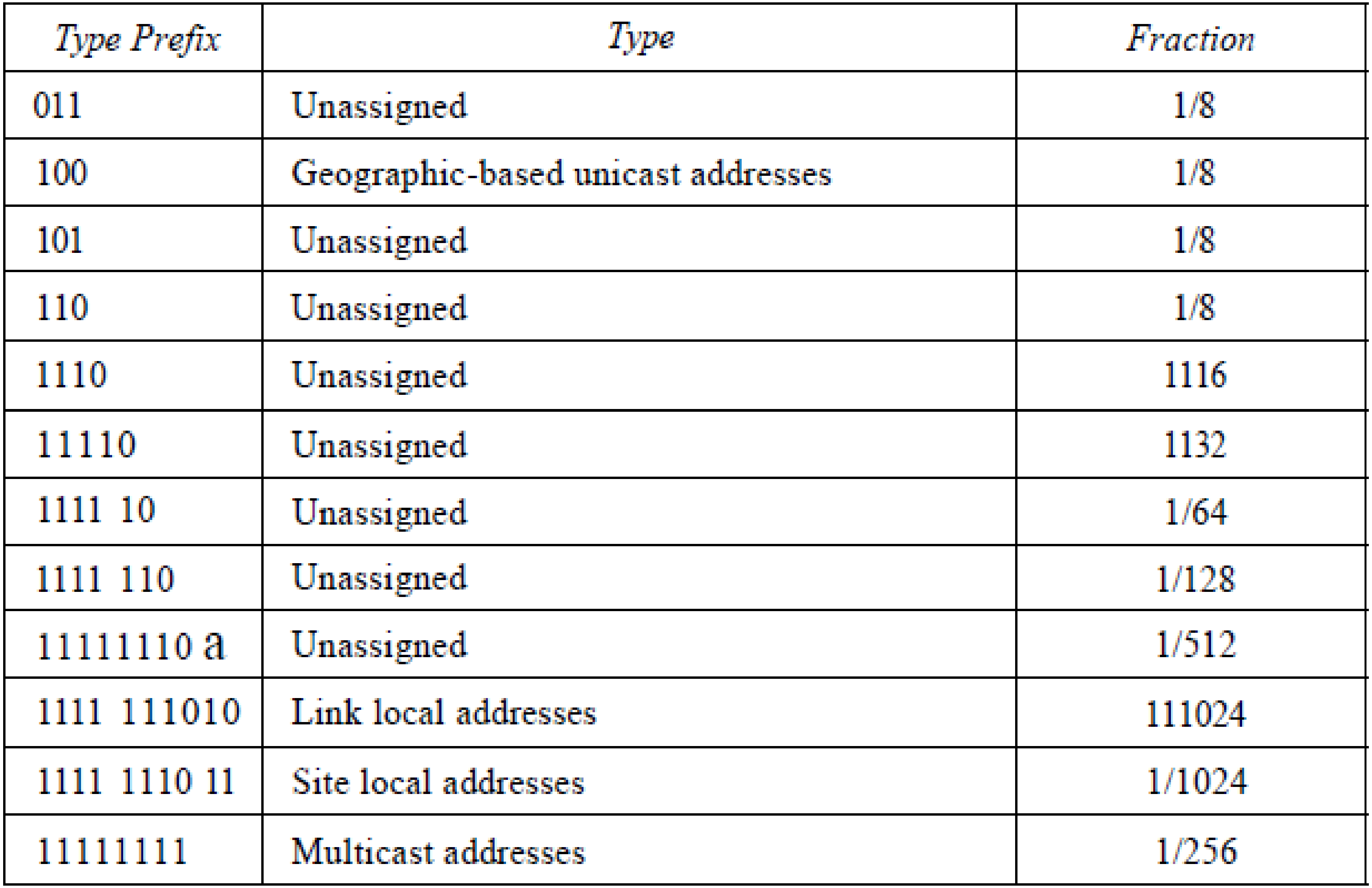
- IPv6 has a much larger address space; 2^{128} addresses are available. The designers of IPv6 divided the address into several categories. A few leftmost bits, called the **type prefix**, in each address define its category.
- Following table shows the prefix for each type of address. The third column shows the fraction of each type of address relative to the whole address space.

Logical Addressing



Type Prefix	*Type*	*Fraction*
00000000	Reserved	1/256
00000001	Unassigned	1/256
0000001	ISO network addresses	1/128
0000010	IPX (Novell) network addresses	1/128
0000011	Unassigned	1/128
00001	Unassigned	1/32
0001	Reserved	1/16
001	Reserved	1/8
010	Provider-based unicast addresses	1/8

Logical Addressing



Type Prefix	*Type*	*Fraction*
011	Unassigned	1/8
100	Geographic-based unicast addresses	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1116
11110	Unassigned	1132
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
11111110 a	Unassigned	1/512
1111 111010	Link local addresses	111024
1111 1110 11	Site local addresses	1/1024
11111111	Multicast addresses	1/256

Logical Addressing

Exercise

1. Find the class of the following IP addresses.

a. 208.34.54.12

b. 238.34.2.1

c. 114.34.2.8

d. 129.14.6.8.

2. Find the class of the following IP addresses.

a. 11110111 11110011 10000111 11011101

b. 10101111 11000000 11110000 00011101

c. 11011111 10110000 00011111 01011101

d. 11101111 11110111 11000111 00011101

Logical Addressing

3. Find the netid and the hostid of the following IP addresses.

a. 114.34.2.8

b. 132.56.8.6

c. 208.34.54.12

4. In a block of addresses, we know the IP address of one host is 25.34.12.56/16. What are the first address (network address) and the last address (limited broadcast address) in this block?

5. An organization is granted the block 16.0.0.0/8. The administrator wants to create 500 fixed-length subnets.

   a. Find the subnet mask.

   b. Find the number of addresses in each subnet.

   c. Find the first and last addresses in subnet 1.

   d. Find the first and last addresses in subnet 500

Logical Addressing

6. Write the following masks in slash notation (/n).

a. 255.255.255.0

b. 255.0.0.0

c. 255.255.224.0

d. 255.255.240.0

7. Find the range of addresses in the following blocks.

a. 123.56.77.32/29

b. 200.17.21.128/27

c. 17.34.16.0/23

d. 180.34.64.64/30

Logical Addressing

8. An ISP is granted a block of addresses starting with 150.80.0.0/16. The ISP wants to distribute these blocks to 2600 customers as follows.

a. The first group has 200 medium-size businesses; each needs 128 addresses.

b. The second group has 400 small businesses; each needs 16 addresses.

c. The third group has 2000 households; each needs 4 addresses.

Design the subblocks and give the slash notation for each subblock. Find out how many addresses are still available after these allocations.

9. Show the shortest form of the following addresses.

a. 2340: lABC:119A:A000:0000:0000:0000:0000

b. 0000:00AA:0000:0000:0000:0000: 119A:A231

c. 2340:0000:0000:0000:0000: 119A:A001:0000

d. 0000:0000:0000:2340:0000:0000:0000:0000

Logical Addressing

10. Show the original (unabbreviated) form of the following addresses.

a. 0::0

b. 0:AA::0

c. 0: 1234::3

d. 123::1:2

11. What is the type of each of the following addresses?

a. FE80::12

b. FEC0: :24A2

c. FF02::0

d. 0::01

12. What is the type of each of the following addresses?

a. 0::0

b. 0: :FFFF:0:0

c. 582F:1234::2222

d. 4821::14:22

e. 54EF::A234:2

Internet Protocol

Internet Protocol

Internet as a Datagram Network

- The Internet has chosen the datagram approach to switching in the network layer.
- It uses the universal addresses defined in the network layer to route packets from the source to the destination.

Internet Protocol

Internet as a Connectionless Network

- Delivery of a packet can be accomplished by using either a connection-oriented or a connectionless network service.
- In a **connection-oriented service**, the source first makes a connection with the destination before sending a packet. When the connection is established, a sequence of packets from the same source to the same destination can be sent one after another. In this case, there is a relationship between packets. They are sent on the same path in sequential order. A packet is logically connected to the packet traveling before it and to the packet traveling after it. When all packets of a message have been delivered, the connection is terminated.
- This type of service is used in a virtual-circuit approach. to packet switching such as in Frame Relay and ATM.

Internet Protocol

In **connectionless service**, the network layer protocol treats each packet independently, with each packet having no relationship to any other packet. The packets in a message mayor may not travel the same path to their destination.

This type of service is used in the datagram approach to packet switching. The **Internet** has chosen this type of service at the network layer.

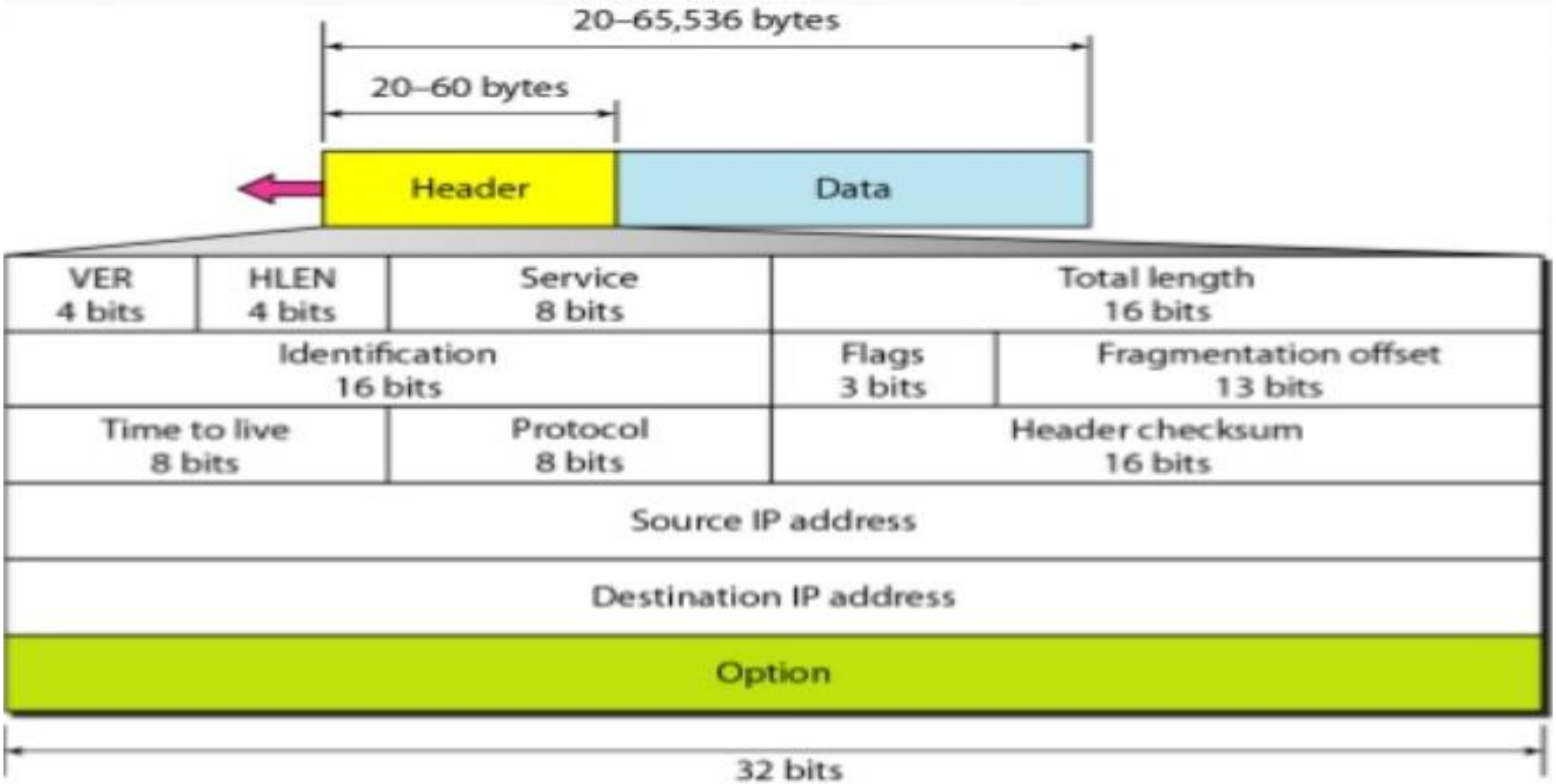
IPv4

- The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.
- IPv4 is an unreliable and connectionless datagram protocol-a best-effort delivery service.
- The term best-effort means that IPv4 provides no error control or flow control (except for error detection on the header).
- If reliability is important, IPv4 must be paired with a reliable protocol such as TCP.

IPv4

Datagram

Packets in the IPv4 layer are called datagrams. Following figure shows the IPv4 datagram format.



20–65,536 bytes

20–60 bytes

Header

Data

VER 4 bits	HLEN 4 bits	Service 8 bits	Total length 16 bits
Identification 16 bits		Flags 3 bits	Fragmentation offset 13 bits
Time to live 8 bits		Protocol 8 bits	Header checksum 16 bits
Source IP address			
Destination IP address			
Option			

32 bits

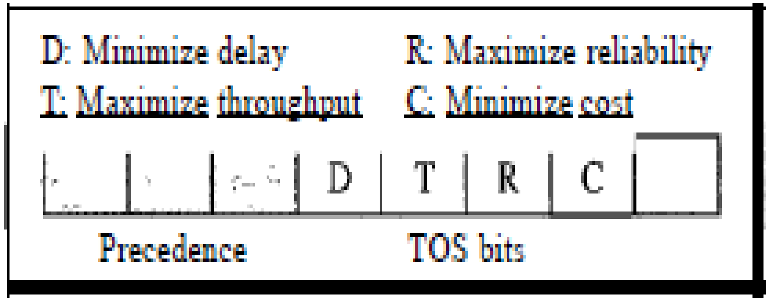
IPv4

- ❖ 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 and delivery.

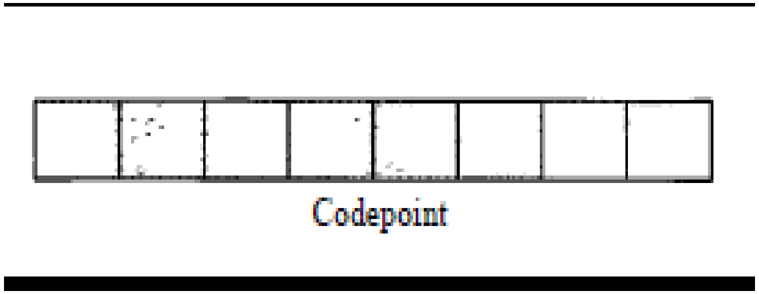
Version (VER): This 4-bit field defines the version of the IPv4 protocol.

Header length (HLEN): This 4-bit field defines the total length of the datagram header in 4-byte words.

Services: This field, previously called service type, is now called differentiated services.



Service type



Differentiated services

IPv4

Service Type

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

a. Precedence is a 3-bit subfield ranging from 0 (000 in binary) to 7 (111 in binary). The precedence defines the priority of the datagram in issues such as congestion. If a router is congested and needs to discard some datagrams, those datagrams with lowest precedence are discarded first.

b. TOS bits is a 4-bit subfield with each bit having a special meaning. Although a bit can be either 0 or 1, one and only one of the bits can have the value of 1 in each datagram.

IPv4

Total length:

- This is a 16-bit field that defines the total length (header plus data) of the IPv4 datagram in bytes.
- To find the length of the data coming from the upper layer, subtract the header length from the total length.
- The header length can be found by multiplying the value in the HLEN field by 4.

Length of data = total length - header length

Identification: This field is used in fragmentation.

Flags: This field is used in fragmentation

Fragmentation offset: This field is used in fragmentation

IPv4

Time to live: A datagram has a limited lifetime in its travel through an internet.

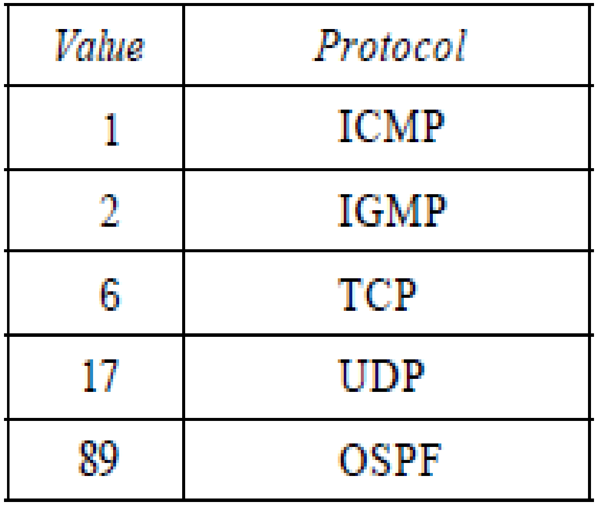
- ❖ This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero.

- ❖ Today, this field is used mostly to control the maximum number of hops (routers) visited by the datagram. When a source host sends the datagram, it stores a number in this field. This value is approximately 2 times the maximum number of routes between any two hosts. Each router that processes the datagram decrements this number by 1. If this value, after being decremented, is zero, the router discards the datagram.

IPv4

Protocol: This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP. This field specifies the final destination protocol to which the IPv4 datagram is delivered.

The value of this field for each higher-level protocol is shown in the following table:-



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

IPv4

Checksum: This field is used to detect error.

Source address: This 32-bit field defines the IPv4 address of the source. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

Destination address: This 32-bit field defines the IPv4 address of the destination. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

IPv4

Example

In an IPv4 packet, the value of HLEN is 1000 in binary. How many bytes of options are being carried by this packet?

Solution

The HLEN value is 8, which means the total number of bytes in the header is 8 x 4, or 32 bytes. The first 20 bytes are the base header, the next 12 bytes are the options.

Example

In an IPv4 packet, the value of HLEN is 5, and the value of the total length field is Ox0028. How many bytes of data are being carried by this packet?

Solution

The HLEN value is 5, which means the total number of bytes in the header is 5 x 4, or 20 bytes (no options). The total length is 40 bytes, which means the packet is carrying 20 bytes of data (40- 20).

IPv4

Example

An IPv4 packet has arrived with the first few hexadecimal digits as shown.

Ox45000028000100000102 ...

How many hops can this packet travel before being dropped? The data belong to what upper-layer protocol?

Solution

To find the time-to-live field, we skip 8 bytes (16 hexadecimal digits). The time-to-live field is the ninth byte, which is 01. This means the packet can travel only one hop. The protocol field is the next byte (02), which means that the upper-layer protocol is IGMP.

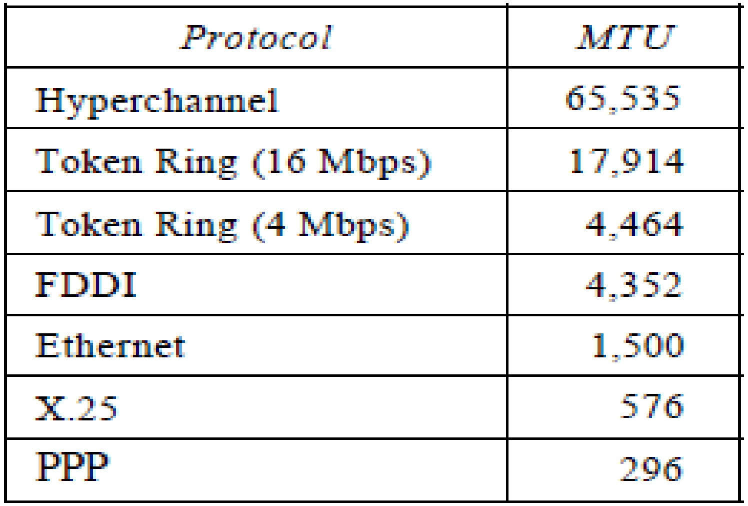
IPv4

Fragmentation

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, which is defined by the restrictions imposed by the hardware and software used in the network.

The value of the MTU depends on the physical network protocol. Following table shows the values for some protocols:-



Protocol	*MTU*
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

IPv4

- If we divide the datagram into fragments to make it possible to pass through the networks, then this process is called the **fragmentation**.
- When a datagram is fragmented, each fragment has its own header with most of the fields repeated, but with some changed. A fragmented datagram may itself be fragmented if it encounters a network with an even smaller MTU. In other words, a datagram can be fragmented several times before it reaches the final destination.

IPv4

Fields Related to Fragmentation

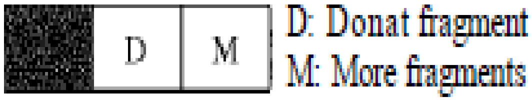
The fields that are related to fragmentation and reassembly of an IPv4 datagram are the identification, flags, and fragmentation offset fields.

Identification:

- ❖ This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host.
- ❖ When a datagram is fragmented, the value in the identification field is copied to all fragments. In other words, all fragments have the same identification number, the same as the original datagram.
- ❖ The identification number helps the destination in reassembling the datagram. It knows that all fragments having the same identification value must be assembled into one datagram.

IPv4

Flags:

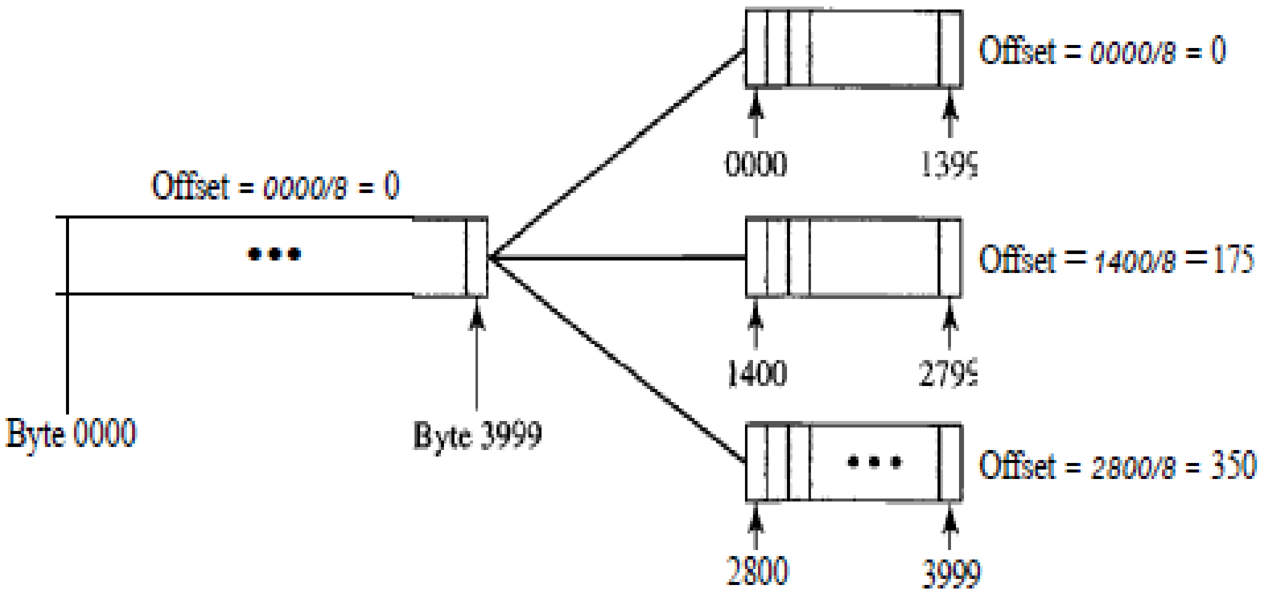


- This is a 3-bit field.
- The first bit is reserved.
- The second bit is called the do not fragment bit. If its value is 1, the machine must not fragment the datagram. If it cannot pass the datagram through any available physical network, it discards the datagram and sends an ICMP error message to the source host. If its value is 0, the datagram can be fragmented if necessary.
- The third bit is called the more fragment bit. If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one. If its value is 0, it means this is the last or only fragment.

IPv4

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

Example: Following figure shows a datagram with a data size of 4000 bytes fragmented into three fragments.

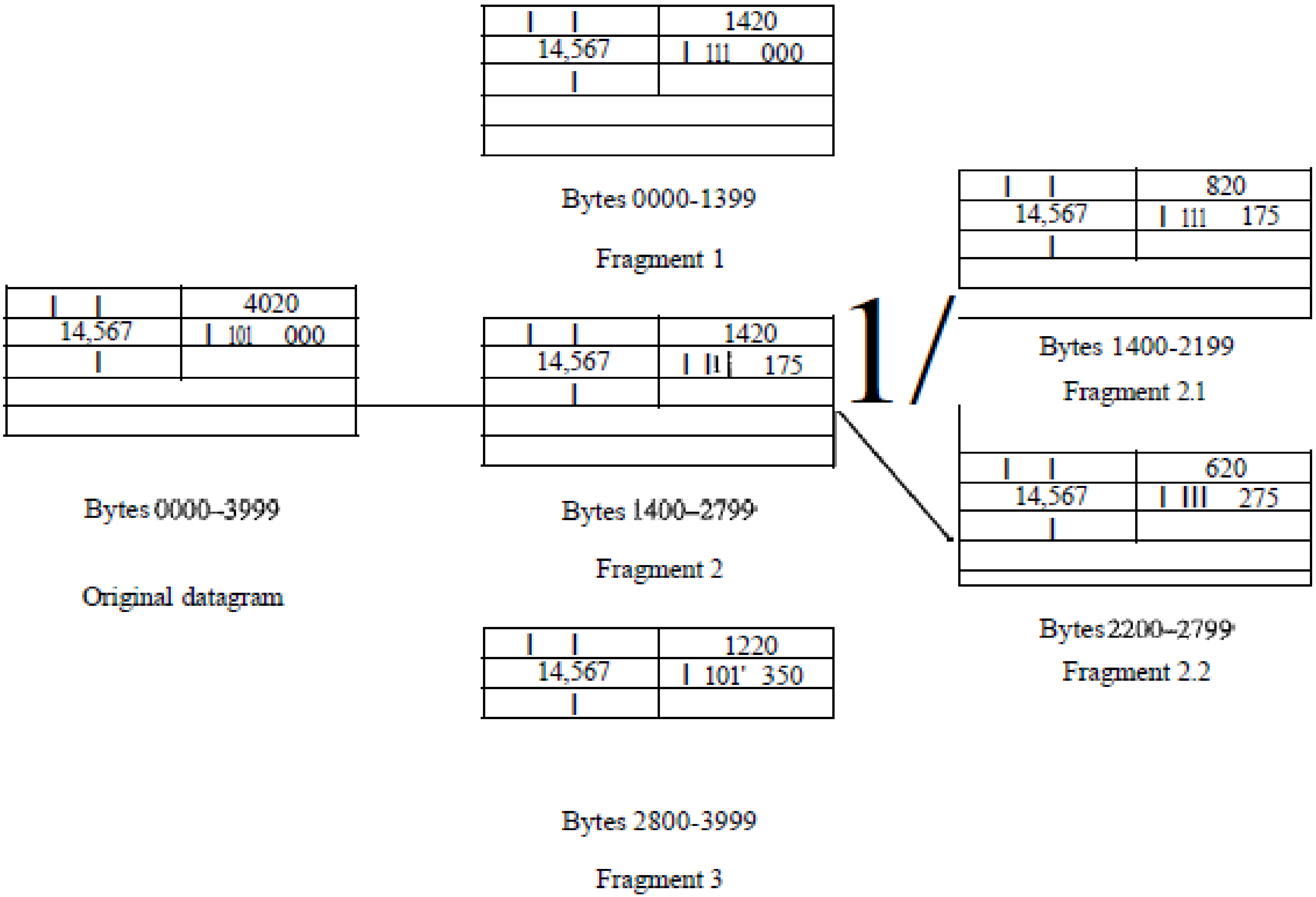


IPv4

The bytes in the original datagram are numbered 0 to 3999.

- The first fragment carries bytes 0 to 1399. The offset for this datagram is 0/8 = 0.
- The second fragment carries bytes 1400 to 2799; the offset value for this fragment is 1400/8 = 175.
- Finally, the third fragment carries bytes 2800 to 3999. The offset value for this fragment is 2800/8 =350.

IPv4



1420
14,567
111 000
Bytes 0000-1399
Fragment 1
4020
14,567
101 000
Bytes 0000–3999
Original datagram
1420
14,567
1 175
Bytes 1400–2799
Fragment 2
820
14,567
111 175
Bytes 1400-2199
Fragment 2.1
620
14,567
111 275
Bytes 2200–2799
Fragment 2.2
1220
14,567
101' 350
Bytes 2800-3999
Fragment 3

IPv4

If each fragment follows a different path and arrives out of order, the final destination host can reassemble the original datagram from the fragments received (if none of them is lost) by using the following strategy:

1. The first fragment has an offset field value of zero.
2. Divide the length of the first fragment by 8. The second fragment has an offset value equal to that result.
3. Divide the total length of the first and second fragments by 8. The third fragment has an offset value equal to that result.
4. Continue the process. The last fragment has a more bit value of 0.

IPv4

Example

A packet has arrived with an M bit value of O. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

Solution

If the M bit is 0, it means that there are no more fragments; the fragment is the last one. However, we cannot say if the original packet was fragmented or not. A non fragmented packet is considered the last fragment.

Example

A packet has arrived with an M bit value of 1. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

Solution

If the M bit is 1, it means that there is at least one more fragment. This fragment can be the first one or a middle one, but not the last one. We don't know if it is the first one or a middle one; we need more information (the value of the fragmentation offset).

IPv4

Example

A packet has arrived with an M bit value of 1 and a fragmentation offset value of 0. Is this the first fragment, the last fragment, or a middle fragment?

Solution

Because the M bit is l, it is either the first fragment or a middle one. Because the offset value is 0, it is the first fragment.

Example

A packet has arrived in which the offset value is 100. What is the number of the first byte? Do we know the number of the last byte?

Solution

To find the number of the first byte, we multiply the offset value by 8. This means that the first byte number is 800. We cannot determine the number of the last byte unless we know the length of the data.

IPv4

Example

A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the total length field is 100. What are the numbers of the first byte and the last byte?

Solution

The first byte number is 100 x 8 = 800. The total length is 100 bytes, and the header length is 20 bytes (5 x 4), which means that there are 80 bytes in this datagram. If the first byte number is 800, the last byte number must be 879.

IPv6

IPv6:

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

- Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.
- 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.
- The Internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

To overcome these deficiencies, IPv6 (Internetworking Protocol, version 6), also known as IPng (Internetworking Protocol, next generation), was proposed.

IPv6

Advantages of IPv6 over IPv4

- **Larger address space:** An IPv6 address is 128 bits long. Compared with the 32-bit address of IPv4, this is a huge (2^{96}) 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.

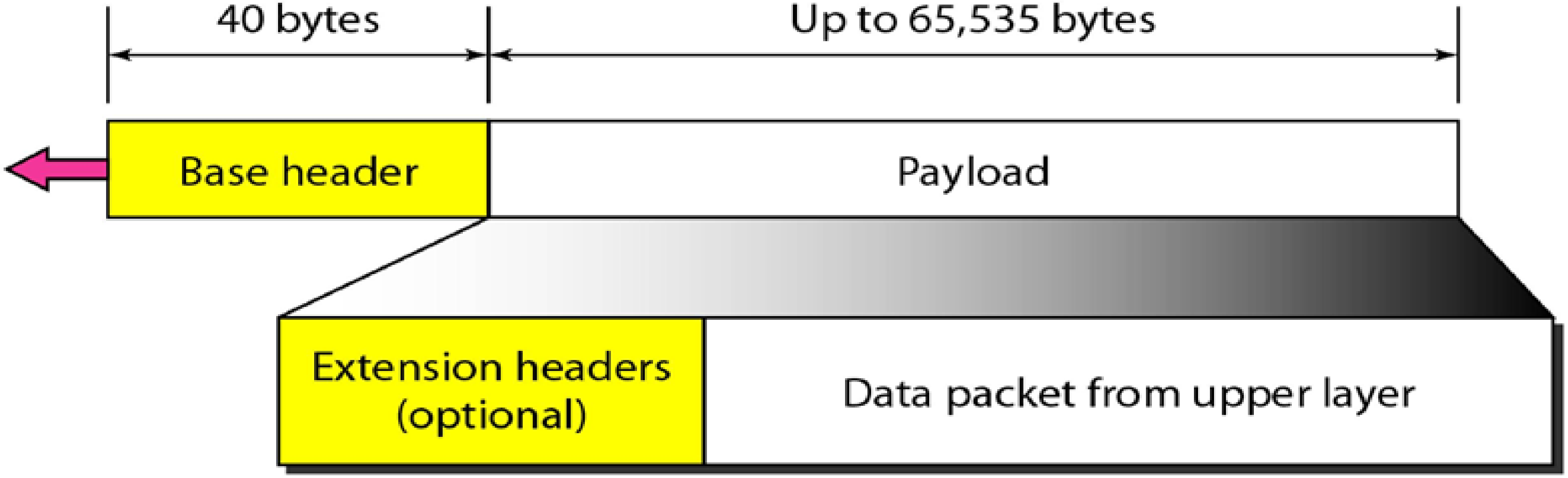
IPv6

- Support for resource allocation. In IPv6, the type-of-service field has been removed, but a mechanism (called Flow label) 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 confidentiality and integrity of the packet.

IPv6

Packet Format

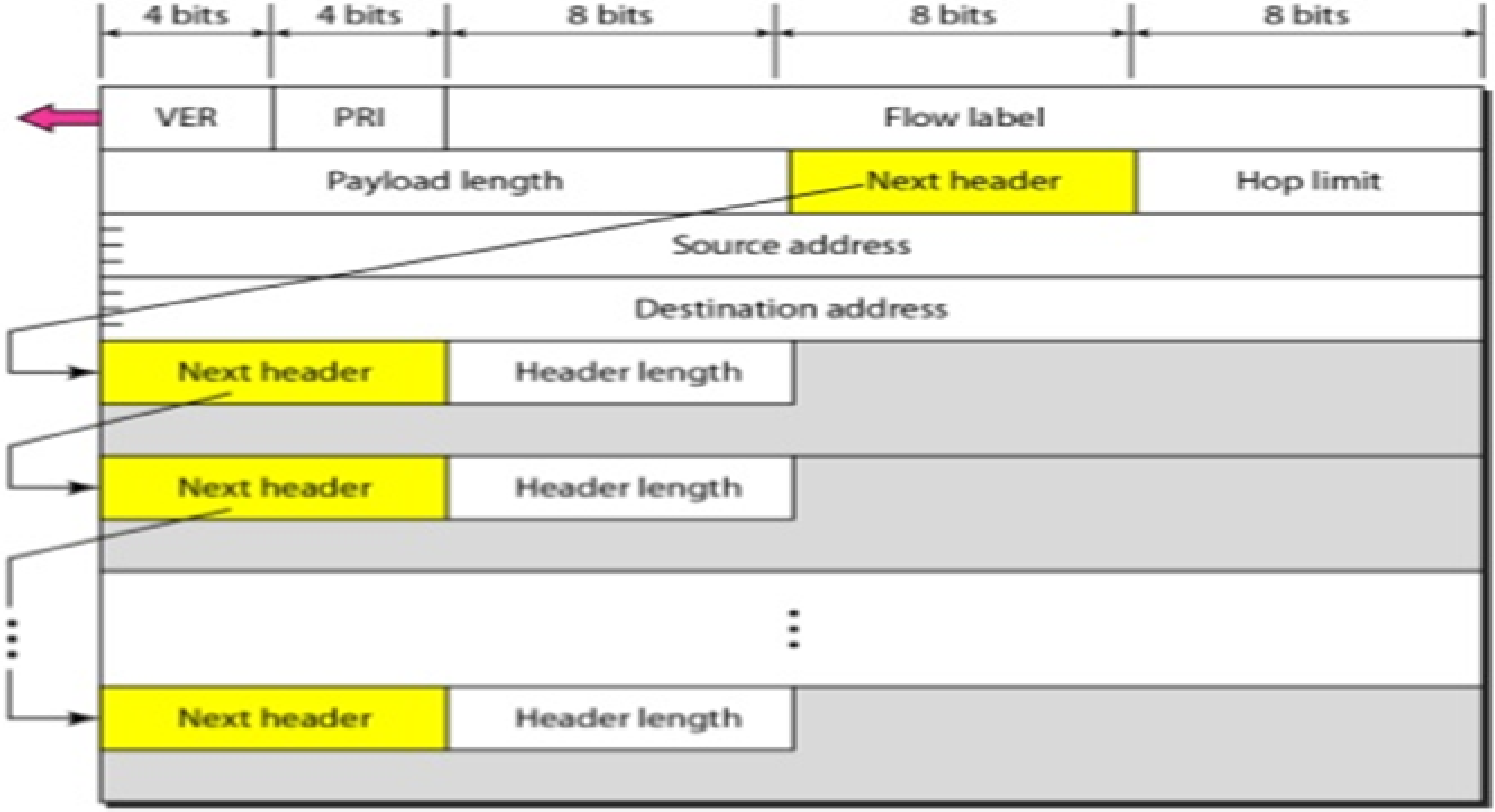
Each packet is composed of a mandatory base header followed by the payload. The payload consists of two parts: optional extension headers and data from an upper layer. The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65,535 bytes of information.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



IPv6 datagram header and payload

IPv6

Format of an IPv6 datagram



IPv6

Base Header

The fields in the base header are as the following:-

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

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

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.

IPv6

Next header: The next header is an 8-bit field defining the header that follows the base header in the datagram. The next header is either one of the optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP. Each extension header also contains this field. Note that this field in version 4 is called the protocol.

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.

Exercise

1. An IPv4 datagram has arrived with the following information in the header (in hexadecimal):

0x45 00 00 54 00 03 58 50 20 06 00 00 7C 4E 03 02 B4 0E 0F 02

a. Is the packet corrupted?

b. Are there any options?

c. Is the packet fragmented?

d. What is the size of the data?

e. How many more routers can the packet travel to?

f. What is the identification number of the packet?

g. What is the type of service?

2. In an IPv4 datagram, the M bit is 0, the value of HLEN is 5, the value of total length is 200, and the offset value is 200. What is the number of the first byte and number of the last byte in this datagram? Is this the last fragment, the first fragment, or a middle fragment?

Delivery, Forwarding, and Routing

Delivery, Forwarding, and Routing

- Delivery refers to the way a packet is handled by the underlying networks under the control of the network layer.
- Forwarding refers to the way a packet is delivered to the next station.
- Routing refers to the way routing tables are created to help in forwarding. Routing protocols are used to continuously update the routing tables that are consulted for forwarding and routing.

Delivery, Forwarding, and Routing

Delivery

The delivery of a packet to its final destination is accomplished by using two different methods of delivery, direct and indirect.

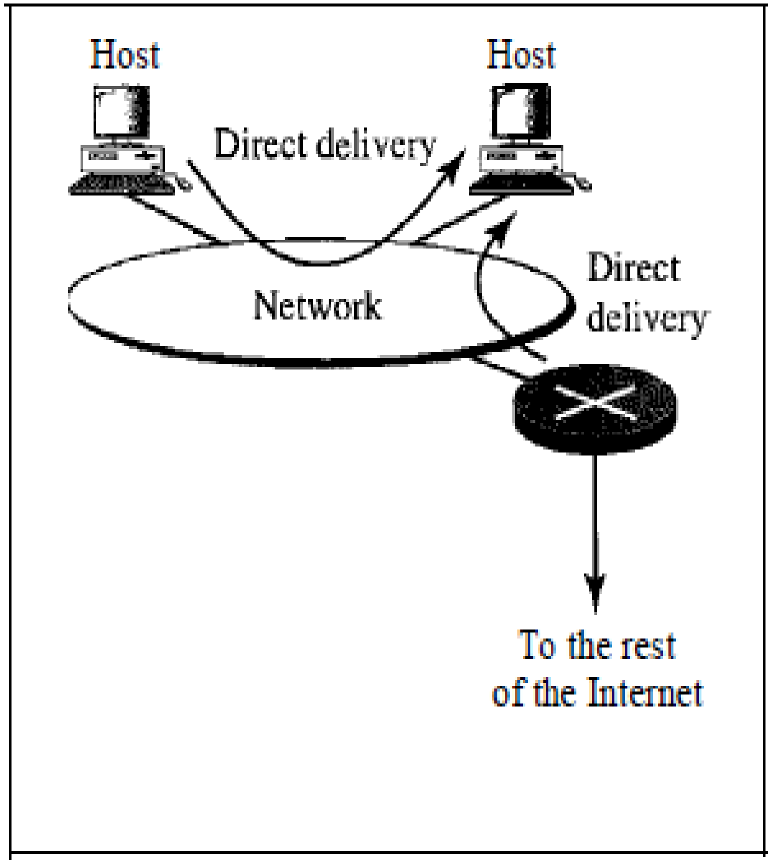
Direct Delivery

- In a direct delivery, the final destination of the packet is a host connected to the same physical network as the deliverer.
- Direct delivery occurs when the source and destination of the packet are located on the same physical network or when the delivery is between the last router and the destination host.

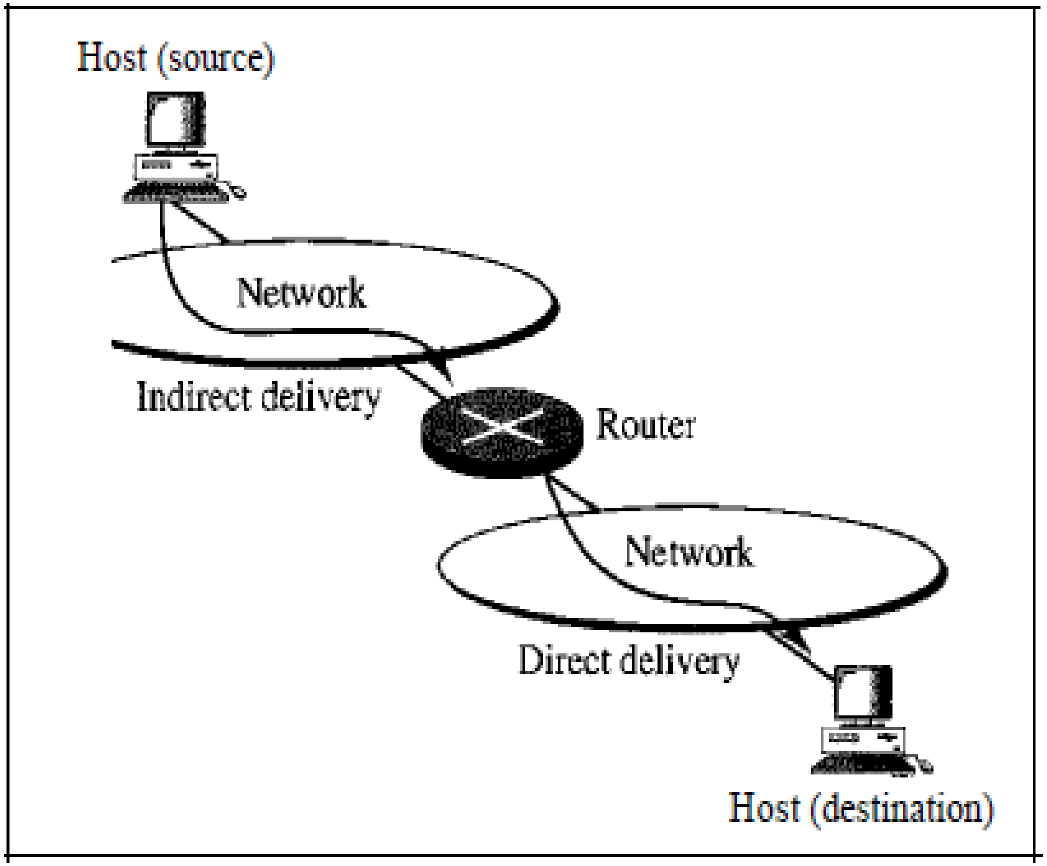
Indirect Delivery

- If the destination host is not on the same network as the deliverer, the packet is delivered indirectly.
- In an indirect delivery, the packet goes from router to router until it reaches the one connected to the same physical network as its final destination.

Delivery, Forwarding, and Routing



a. Direct delivery



b. Indirect and direct delivery

Note: (1) A delivery always involves one direct delivery but zero or more indirect deliveries.

(2) The last delivery is always a direct delivery.

Delivery, Forwarding, and Routing

Forwarding

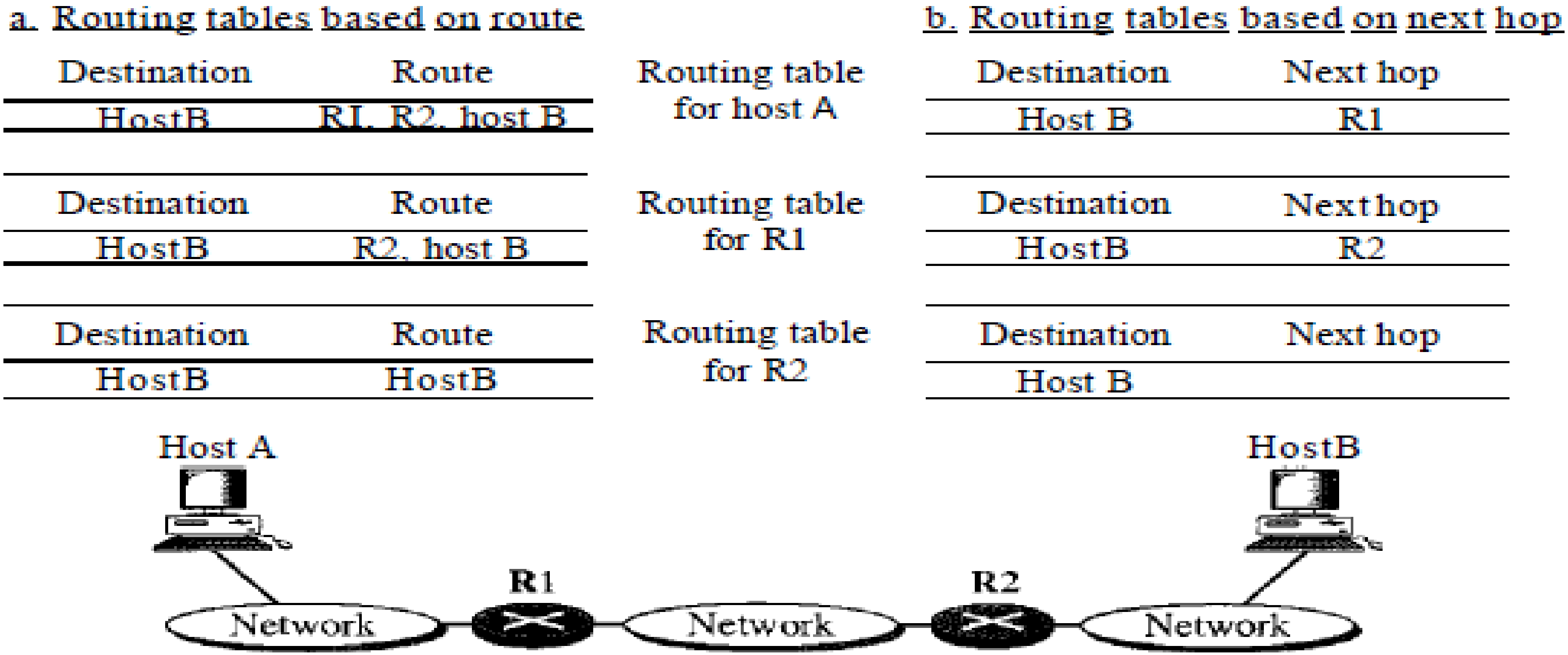
Forwarding means to place the packet in its route to its destination. Forwarding requires a host or a router to have a routing table. When a host has a packet to send or when a router has received a packet to be forwarded, it looks at this table to find the route to the final destination.

There are following methods of forwarding:-

Delivery, Forwarding, and Routing

Next-Hop Method Versus Route Method

One technique to reduce the contents of a routing table is called the next-hop method. In this technique, the routing table holds only the address of the next hop instead of information about the complete route (route method).



a. Routing tables based on route

b. Routing tables based on next hop

Destination	Route	Routing table for host A	Destination	Next hop
HostB	RI. R2. host B		Host B	R1

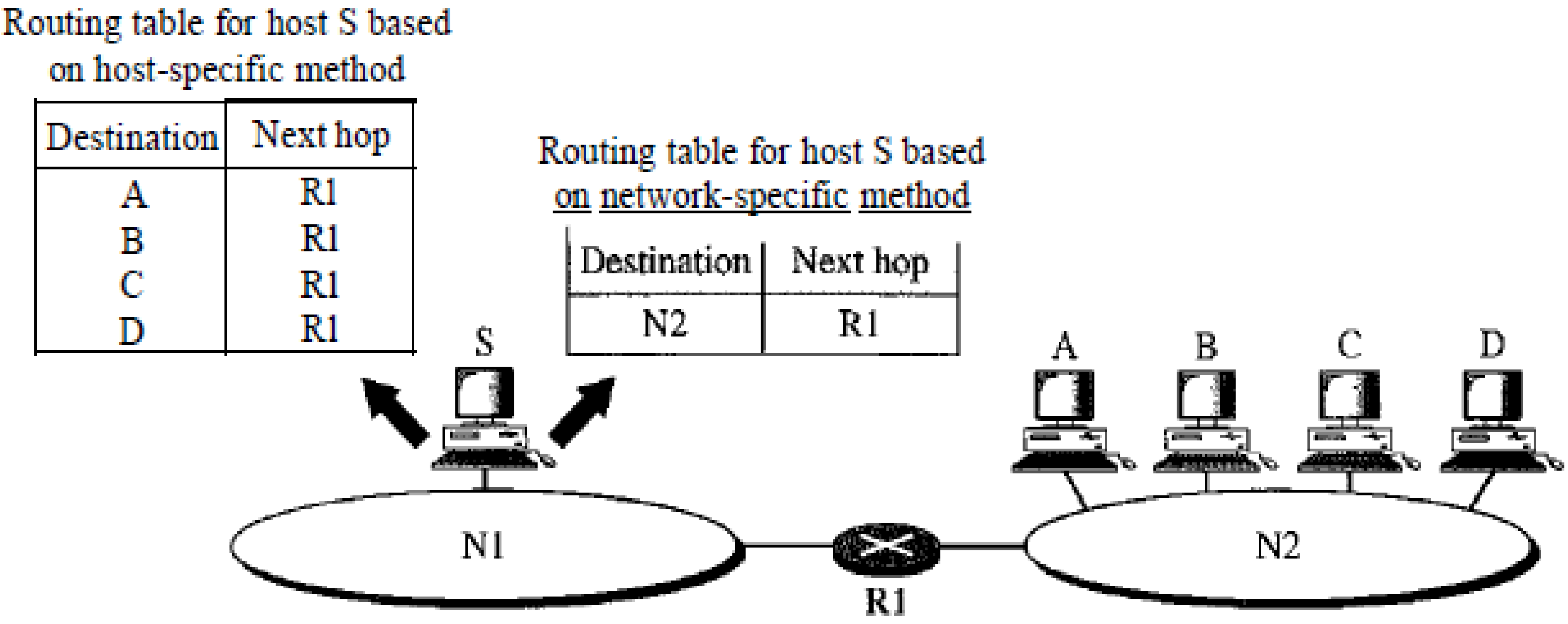
Destination	Route	Routing table for R1	Destination	Next hop
HostB	R2. host B		HostB	R2

Destination	Route	Routing table for R2	Destination	Next hop
HostB	HostB		Host B	

Delivery, Forwarding, and Routing

Network-Specific Method Versus Host-Specific Method

A second technique to reduce the routing table and simplify the searching process is called the network-specific method. Here, instead of having an entry for every destination host connected to the same physical network (host-specific method), we have only one entry that defines the address of the destination network itself. In other words, we treat all hosts connected to the same network as one single entity.



Routing table for host S based on host-specific method

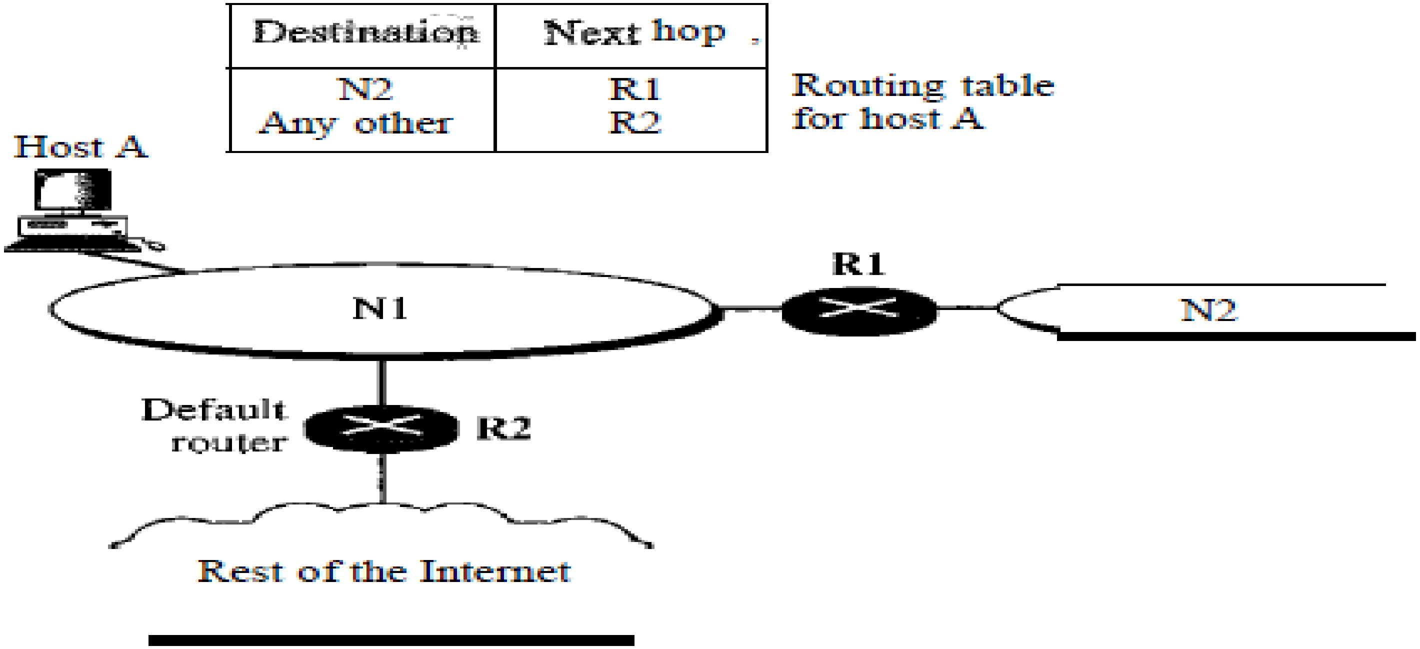
Destination	Next hop
A	R1
B	R1
C	R1
D	R1

Routing table for host S based on network-specific method

Destination	Next hop
N2	R1

Delivery, Forwarding, and Routing

Default Method



Destination	Next hop
N2 Any other	R1 R2

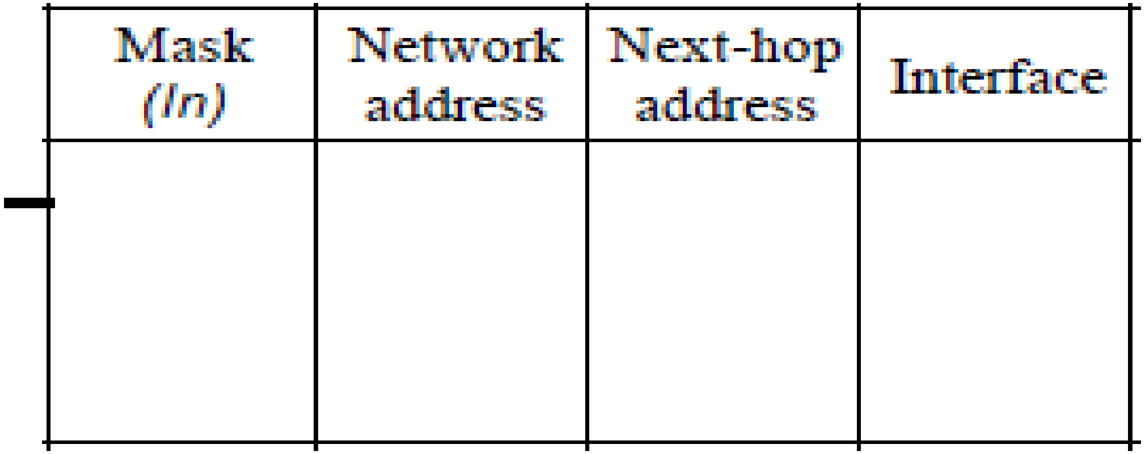
Routing table for host A

In this figure, host A is connected to a network with two routers. Router R1 routes the packets to hosts connected to network N2. However, for the rest of the Internet, router R2 is used. So instead of listing all networks in the entire Internet, host A can just have one entry called the **default (normally defined as network address)**.

Delivery, Forwarding, and Routing

Structure of Routing Table

In classless addressing, we need at least four columns in a routing table.

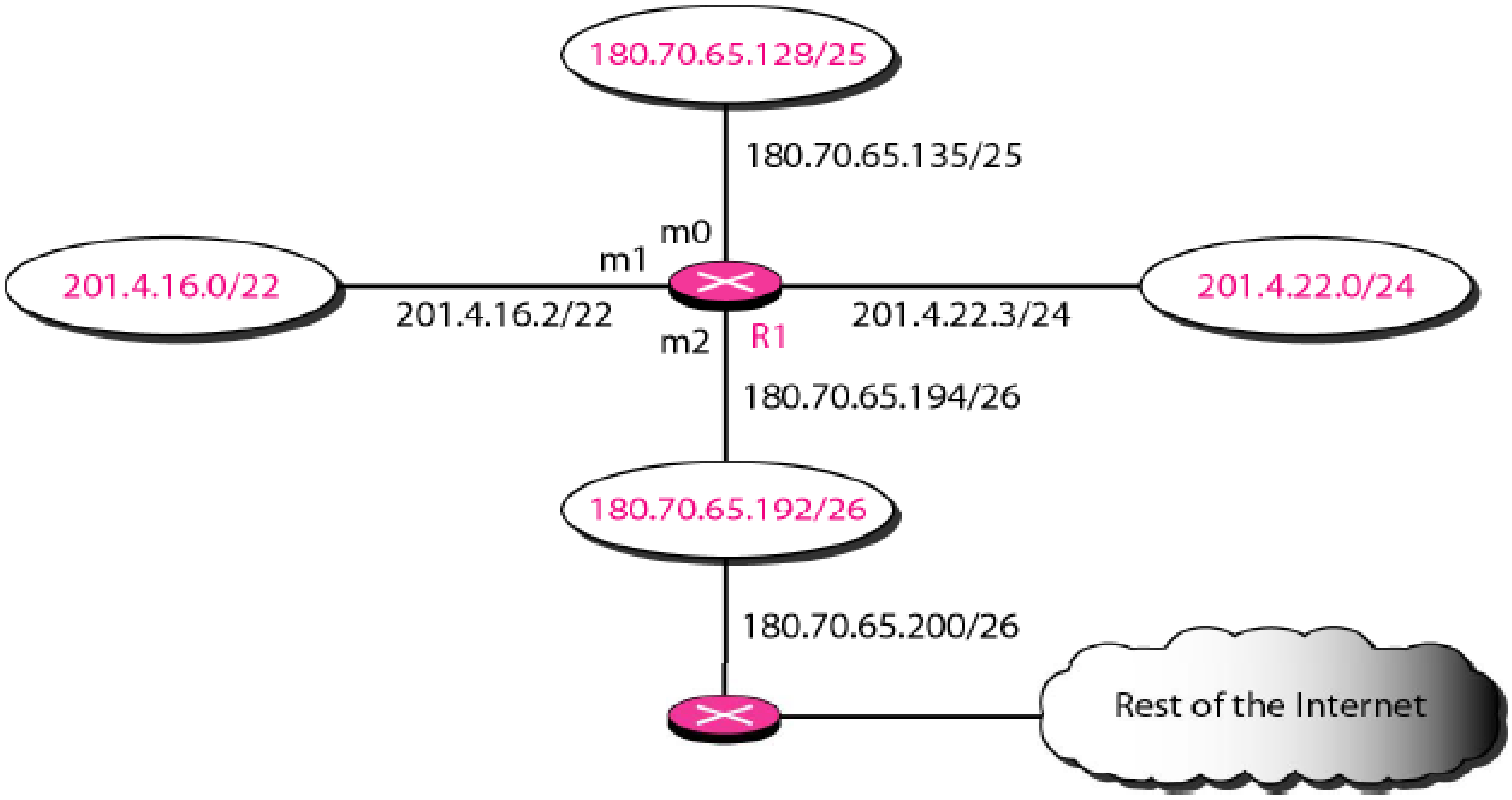


Mask (/n)	Network address	Next-hop address	Interface

Delivery, Forwarding, and Routing

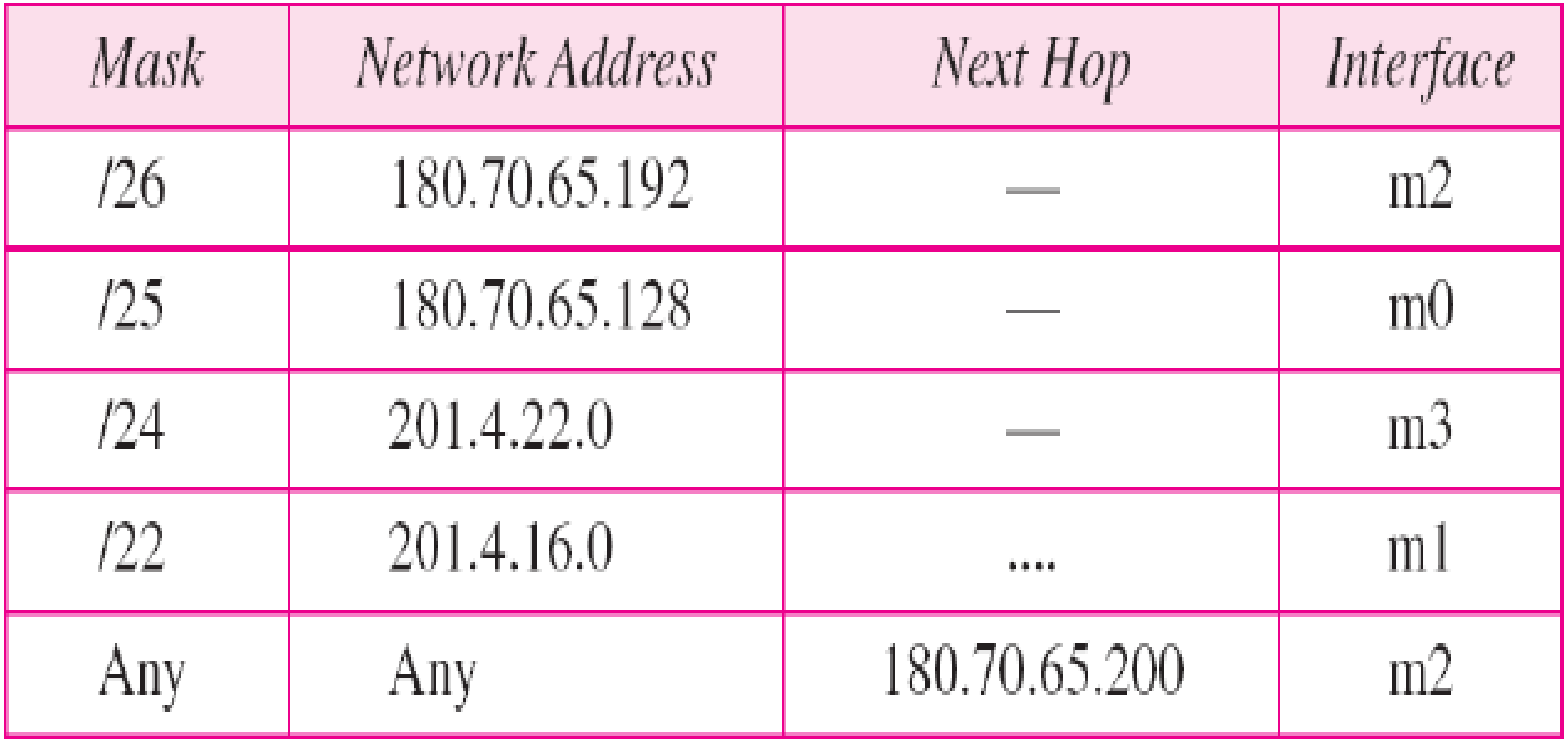
Example

Make a routing table for router R1, using the configuration in shown in following figure:-



Delivery, Forwarding, and Routing

Solution: Routing table corresponding to router R1 is shown in the following figure:-



Mask	*Network Address*	*Next Hop*	*Interface*
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

Delivery, Forwarding, and Routing

Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 180.70.65.140.

Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 180.70.65.128, which does not match the corresponding network address.

2. The second mask (/25) is applied to the destination address. The result is 180.70.65.128, which matches the corresponding network address. Therefore, router forwards the packet at the interface number m0.

Delivery, Forwarding, and Routing

Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 201.4.22.35.

Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 1).

2. The second mask (/25) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 2).

3. The third mask (/24) is applied to the destination address. The result is 201.4.22.0, which matches the corresponding network address. The destination address of the packet and the interface number m3 are passed to ARP.

Delivery, Forwarding, and Routing

Example

Show the forwarding process if a packet arrives at R1 in previous example with the destination address 18.24.32.78.

Solution

This time all masks are applied, one by one, to the destination address, but no matching network address is found. When it reaches the end of the table, the module gives the next-hop address 180.70.65.200 and interface number m2 to ARP. This is probably an outgoing package that needs to be sent, via the default router, to someplace else in the Internet.

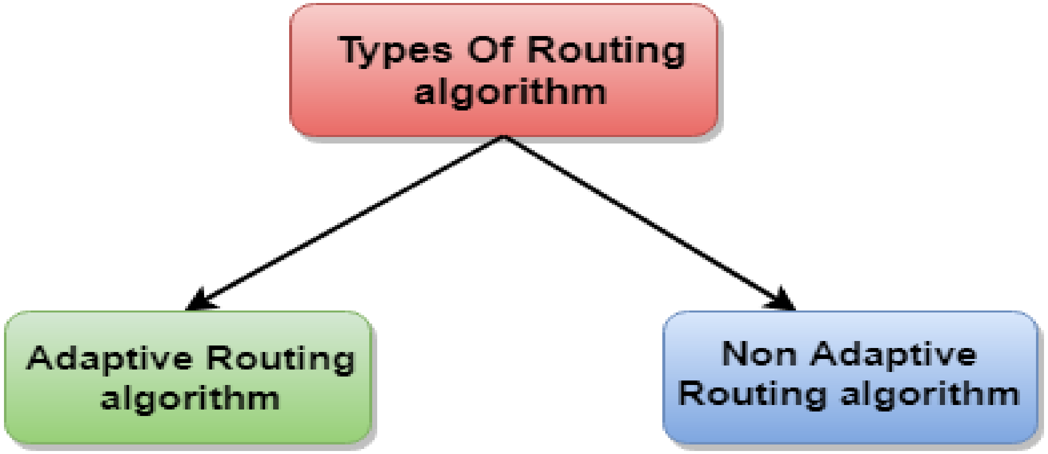
Delivery, Forwarding, and Routing

Routing algorithm

- The routing protocol is a routing algorithm that provides the best path from the source to the destination. The best path is the path that has the "least-cost path" from source to the destination.
- Routing is the process of forwarding the packets from source to the destination but the best route to send the packets is determined by the routing algorithm.

Delivery, Forwarding, and Routing

Classification of a Routing algorithm


Types Of Routing algorithm
Adaptive Routing algorithm
Non Adaptive Routing algorithm

Adaptive Routing algorithm

- An adaptive routing algorithm is also known as dynamic routing algorithm.
- This algorithm makes the routing decisions based on the topology and network traffic.
- The main parameters related to this algorithm are hop count, distance and estimated transit time.

Delivery, Forwarding, and Routing

Non-Adaptive Routing algorithm

- ❖ Non-Adaptive routing algorithm is also known as a static routing algorithm.
- ❖ Non-Adaptive routing algorithms do not take the routing decision based on the network topology or network traffic.

The Non-Adaptive Routing algorithm is of two types:

Flooding: In case of flooding, every incoming packet is sent to all the outgoing links except the one from it has been reached. The disadvantage of flooding is that node may contain several copies of a particular packet.

Random walks: In case of random walks, a packet sent by the node to one of its neighbors randomly. An advantage of using random walks is that it uses the alternative routes very efficiently.

Delivery, Forwarding, and Routing

Unicast Routing Protocols

- A routing table can be either static or dynamic. A static table is one with manual entries.
- A dynamic table, on the other hand, is one that is updated automatically when there is a change somewhere in the internet.
- Today, an internet needs dynamic routing tables.
- A routing protocol is a combination of rules and procedures that lets routers in the internet inform each other of changes. It allows routers to share whatever they know about the internet or their neighborhood.

Delivery, Forwarding, and Routing

Intra domain and Inter domain Routing

- An internet is divided into autonomous systems.
- An autonomous system (AS) is a group of networks and routers under the authority of a single administration.
- Routing inside an autonomous system is referred to as intra domain routing.
- Routing between autonomous systems is referred to as inter domain routing.
- Each autonomous system can choose one or more intra domain routing protocols to handle routing inside the autonomous system. However, only one inter domain routing protocol handles routing between autonomous systems.

Delivery, Forwarding, and Routing

- We are going to discuss two intra domain routing protocols: distance vector and link state and one inter domain routing protocol: Path vector.
- Routing Information Protocol (RIP) is an implementation of the distance vector protocol.
- Open Shortest Path First (OSPF) is an implementation of the link state protocol.
- Border Gateway Protocol (BGP) is an implementation of the path vector protocol.

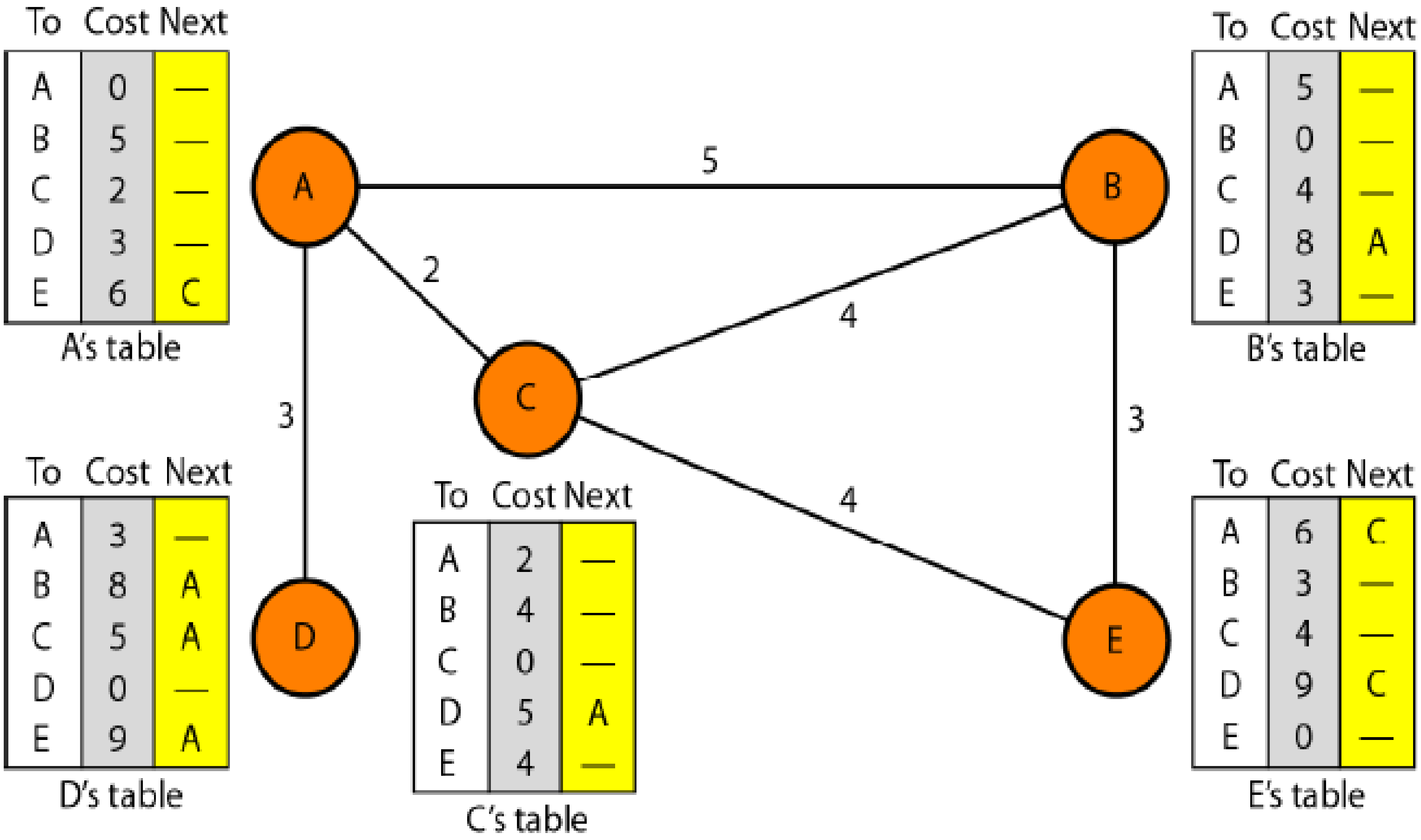
Delivery, Forwarding, and Routing

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 hop in the route (next-hop routing).

Delivery, Forwarding, and Routing

Consider a system of five nodes with their corresponding tables.



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

A's table

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

B's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

D's table

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

C's table

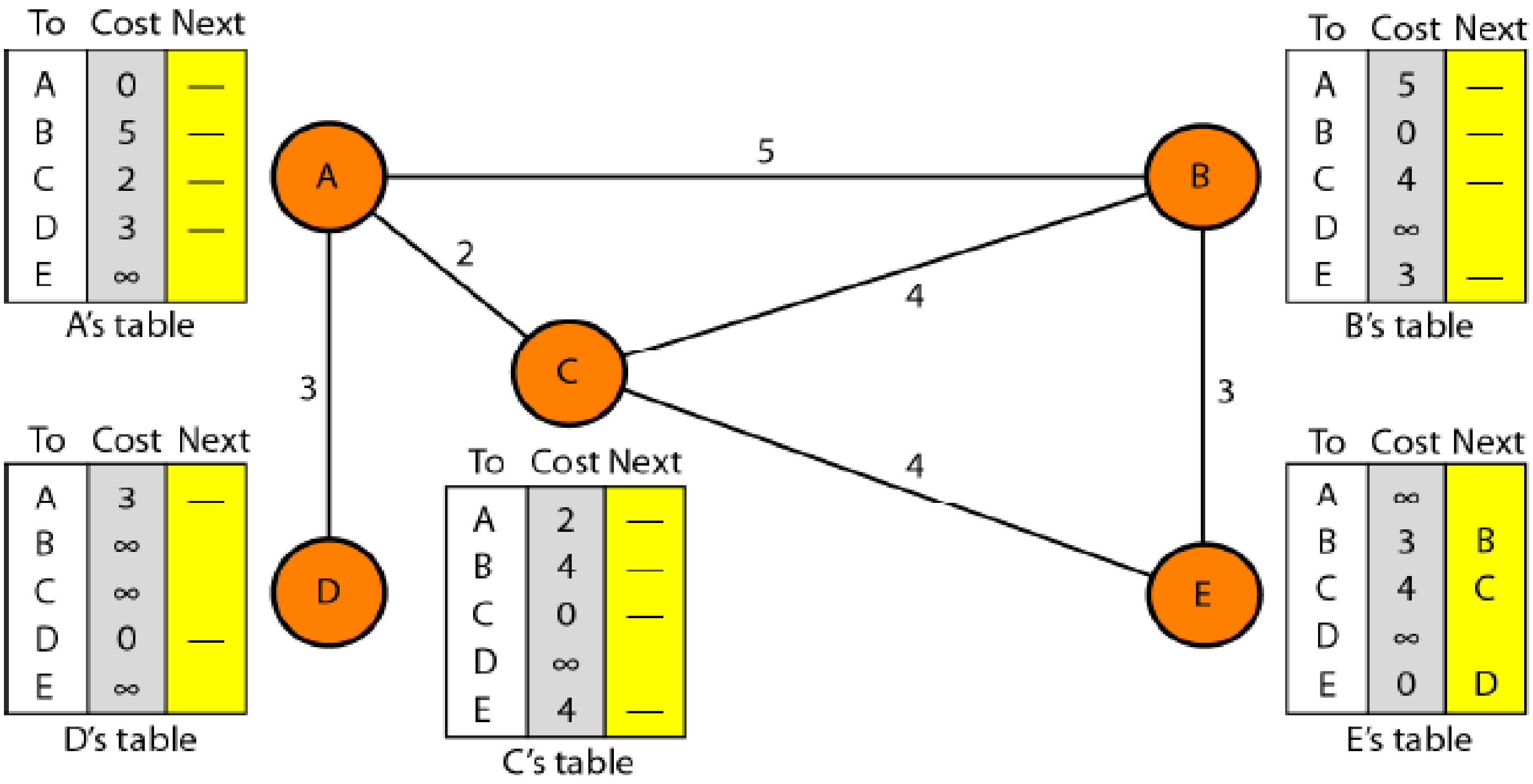
To	Cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—

E's table
]

Delivery, Forwarding, and Routing

Procedure to compute routing table

Initialization



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

A's table

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

B's table

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

D's table

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

C's table

To	Cost	Next
A	∞	
B	3	B
C	4	C
D	∞	
E	0	D

E's table

Delivery, Forwarding, and Routing

Sharing

In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

When the neighbor receives a table, third 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.

Delivery, Forwarding, and Routing

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.
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.

Delivery, Forwarding, and Routing

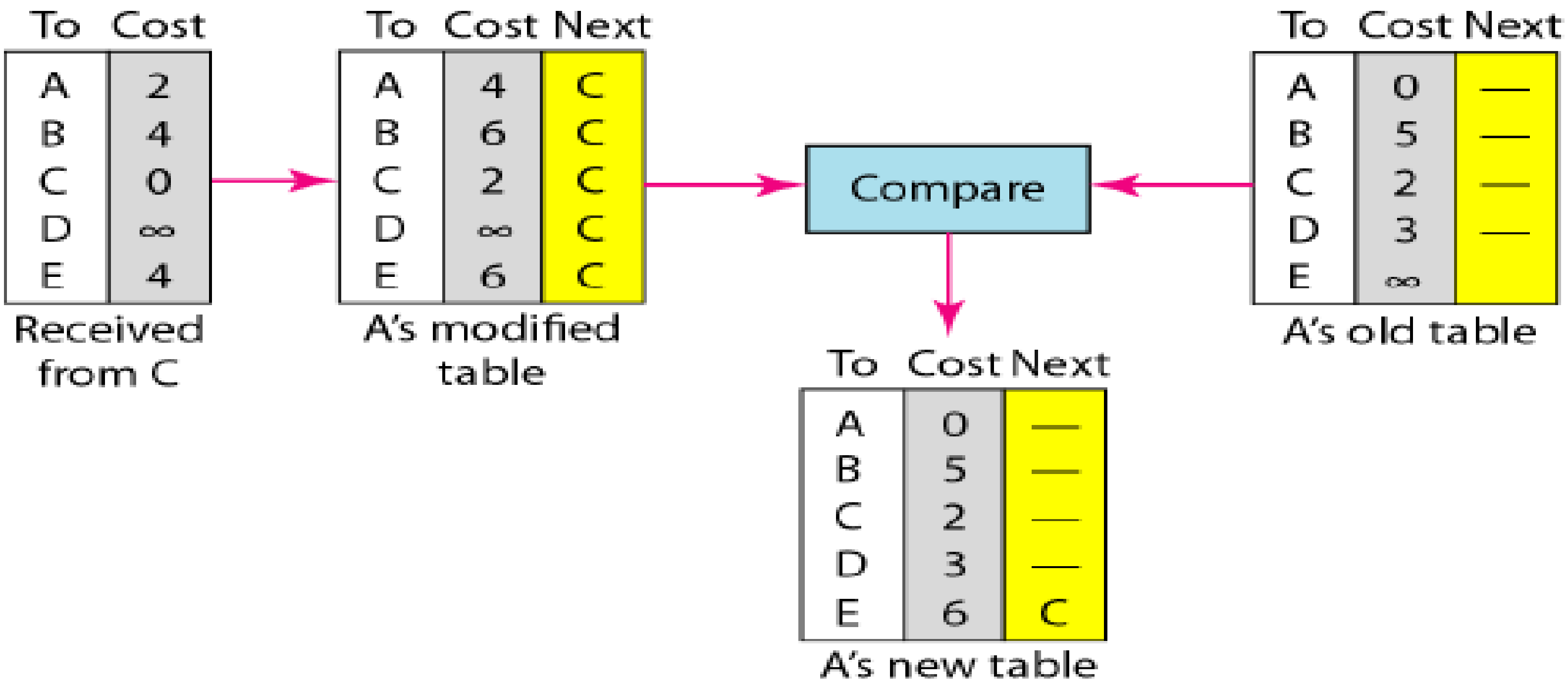
3. The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table.

   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.

Delivery, Forwarding, and Routing

Following figure shows how node A updates its routing table after receiving the partial table from node C.



To	Cost
A	2
B	4
C	0
D	∞
E	4

Received from C

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

A's modified table

Compare

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

Delivery, Forwarding, and Routing

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.

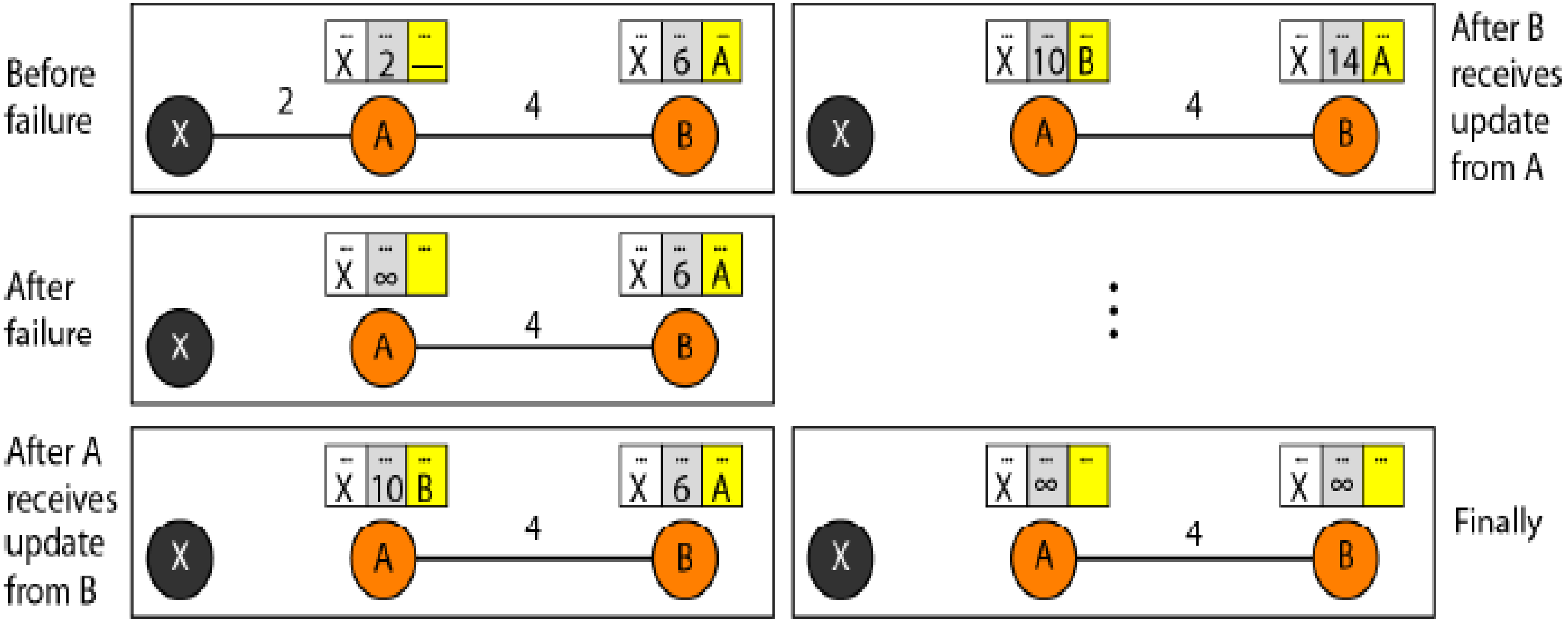
Periodic Update: A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing.

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.

Delivery, Forwarding, and Routing

Two-Node Loop Instability (Count to infinity problem)

A problem with distance vector routing is instability, which means that a network using this protocol can become unstable. To understand the problem, we consider the following figure:-



Delivery, Forwarding, and Routing

- In this figure, at the beginning, both nodes A and B know how to reach node X. But suddenly, the link between A and X fails. Node A changes its table.
- If A can send its table to B immediately, everything is fine.
- However, the system becomes unstable if B sends its routing table to A before receiving A's routing table. Node A receives the update and, assuming that B has found a way to reach X, immediately updates its routing table.
- Based on the triggered update strategy, A sends its new update to B. Now B thinks that something has been changed around A and updates its routing table. The cost of reaching X increases gradually until it reaches infinity. At this moment, both A and B know that X cannot be reached. However, during this time the system is not stable. Node A thinks that the route to X is via B; node B thinks that the route to X is via A. If A receives a packet destined for X, it goes to B and then comes back to A. Similarly, if B receives a packet destined for X, it goes to A and comes back to B. Packets bounce between A and B, creating a two-node loop problem.

Delivery, Forwarding, and Routing

The Routing Information Protocol (RIP)

It is an intra domain routing protocol used inside an autonomous system. It is a very simple protocol based on distance vector routing. RIP implements distance vector routing directly with some considerations:

1. In an autonomous system, we are dealing with routers and networks (links). The routers have routing tables; networks do not.

2. The destination in a routing table is a network, which means the first column defines a network address.

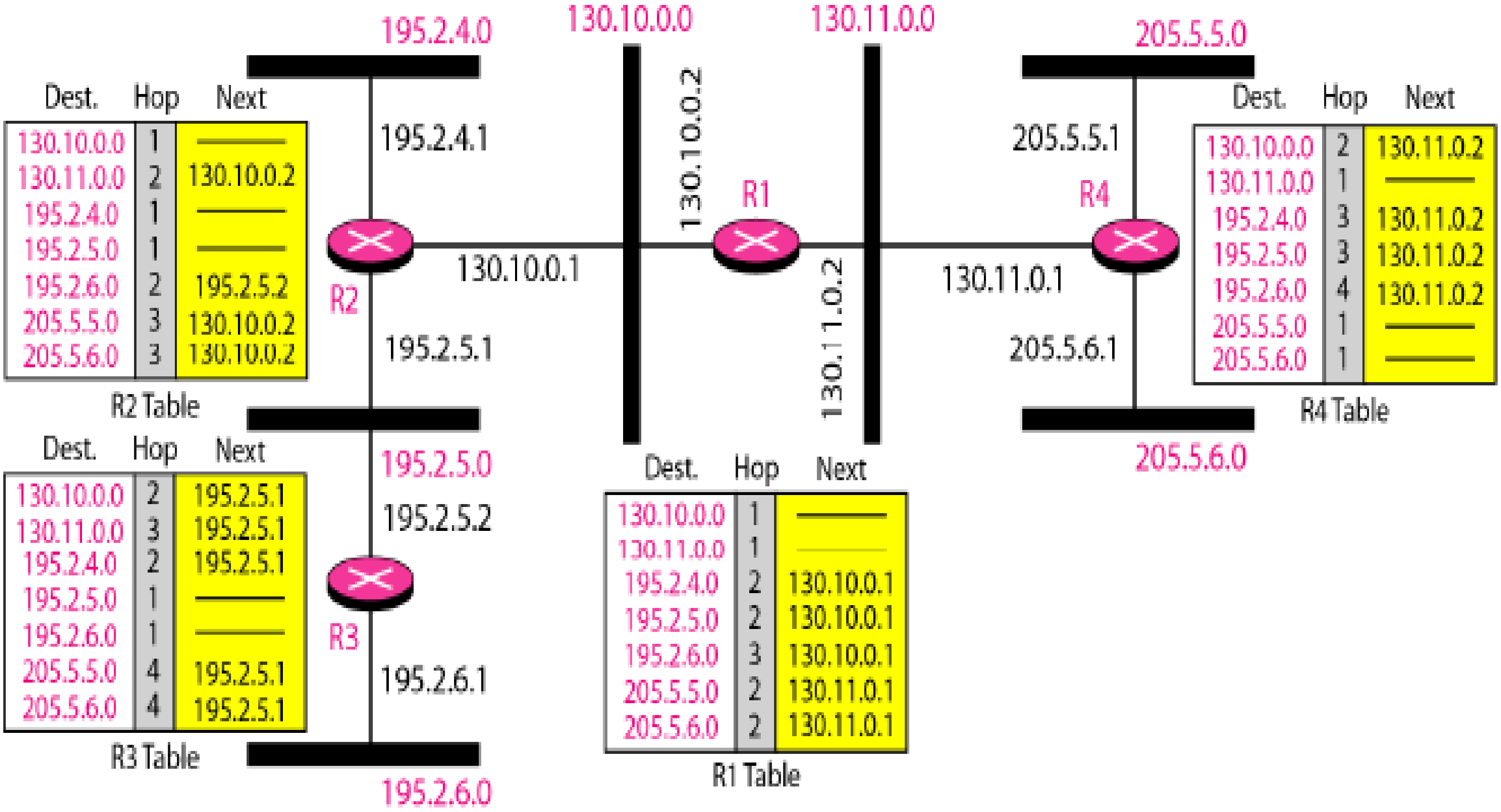
3. The metric used by RIP is very simple; the distance is defined as the number of links (networks) to reach the destination. For this reason, the metric in RIP is called a hop count.

4. Infinity is defined as 16, which means that any route in an autonomous system using RIP cannot have more than 15 hops.

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

Delivery, Forwarding, and Routing

Following figure shows an autonomous system with seven networks and four routers. The table of each router is also shown.



Dest.	Hop	Next
130.10.0.0	1	—
130.11.0.0	2	130.10.0.2
195.2.4.0	1	—
195.2.5.0	1	—
195.2.6.0	2	195.2.5.2
205.5.5.0	3	130.10.0.2
205.5.6.0	3	130.10.0.2

R2 Table

Dest.	Hop	Next
130.10.0.0	2	195.2.5.1
130.11.0.0	3	195.2.5.1
195.2.4.0	2	195.2.5.1
195.2.5.0	1	—
195.2.6.0	1	—
205.5.5.0	4	195.2.5.1
205.5.6.0	4	195.2.5.1

R3 Table

Dest.	Hop	Next
130.10.0.0	1	—
130.11.0.0	1	—
195.2.4.0	2	130.10.0.1
195.2.5.0	2	130.10.0.1
195.2.6.0	3	130.10.0.1
205.5.5.0	2	130.11.0.1
205.5.6.0	2	130.11.0.1

R1 Table

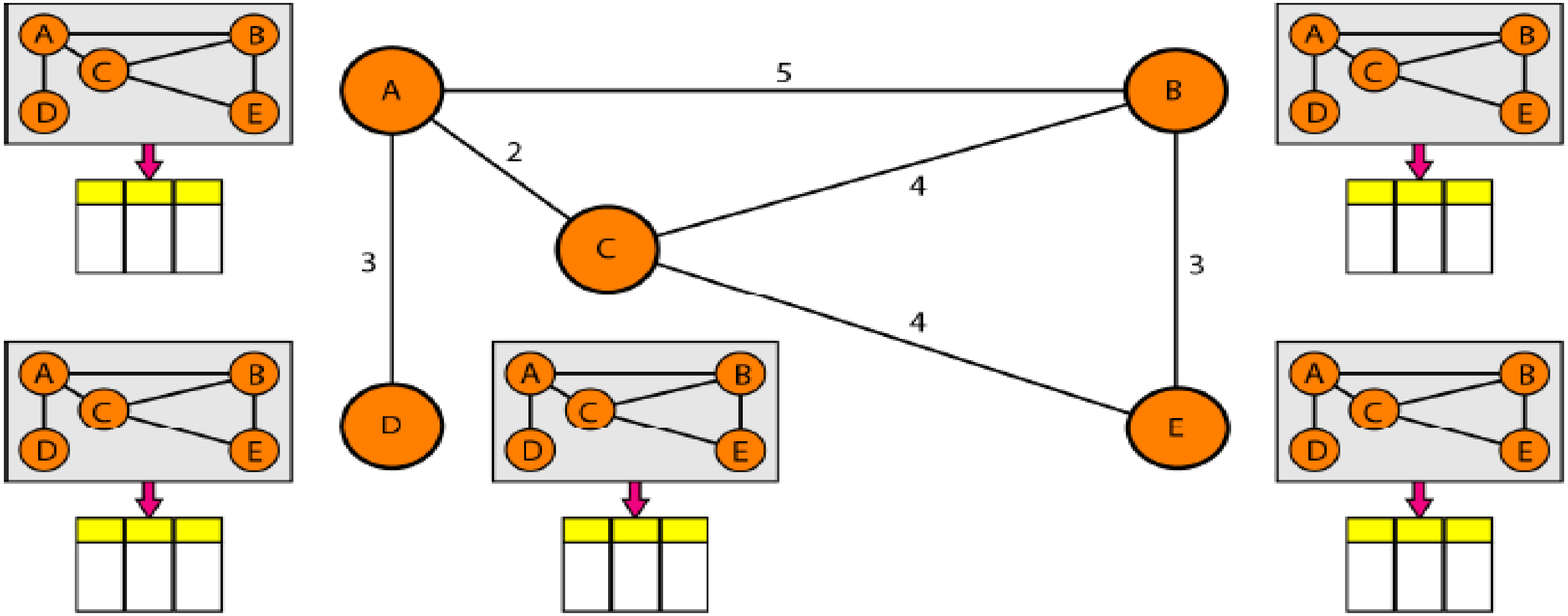
Dest.	Hop	Next
130.10.0.0	2	130.11.0.2
130.11.0.0	1	—
195.2.4.0	3	130.11.0.2
195.2.5.0	3	130.11.0.2
195.2.6.0	4	130.11.0.2
205.5.5.0	1	—
205.5.6.0	1	—

R4 Table

Delivery, Forwarding, and Routing

Link State Routing

In link state routing, each node in the domain has the entire topology of the domain i.e. 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.



Delivery, Forwarding, and Routing

- 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.
- 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.

Delivery, Forwarding, and Routing

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.

Delivery, Forwarding, and Routing

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.
2. On a periodic basis.

Delivery, Forwarding, and Routing

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, it discards the LSP. If it is newer, the node does the following:

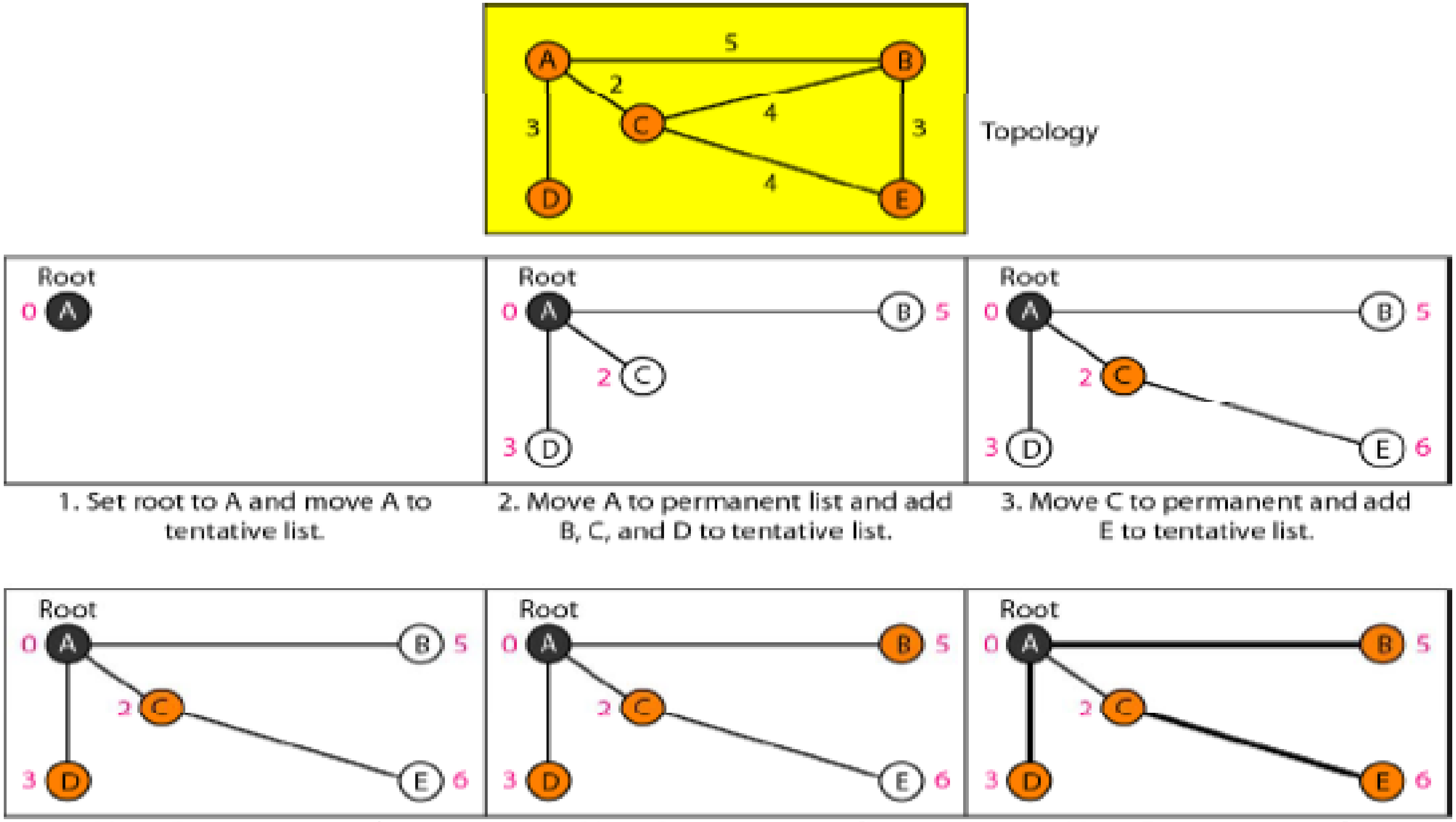
a. It discards the old LSP and keeps the new one.

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).

Delivery, Forwarding, and Routing

Formation of Shortest Path Tree:

After receiving all LSPs, each node will have a copy of the whole topology. Using Dijkstra algorithm, we create shortest path tree at each node.



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.

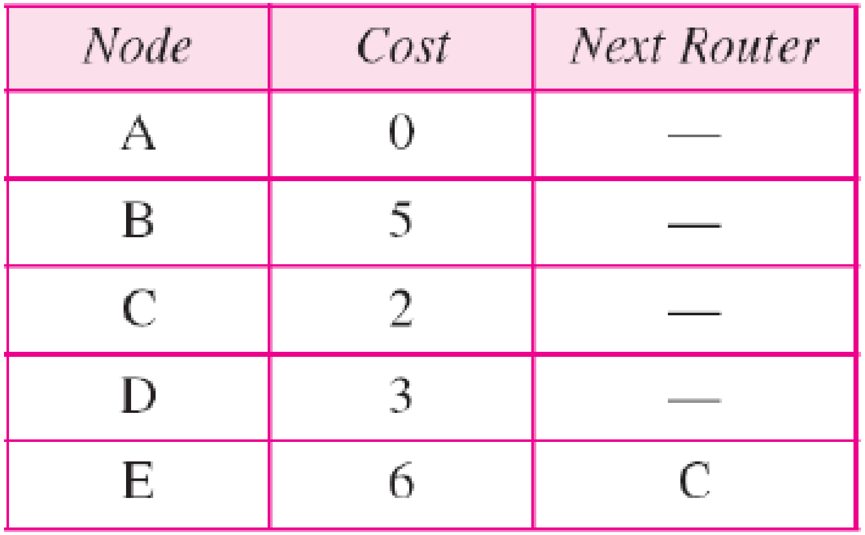
5. Move B to permanent list.

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

Delivery, Forwarding, and Routing

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. Following table shows the routing table for node A.



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

Delivery, Forwarding, and Routing

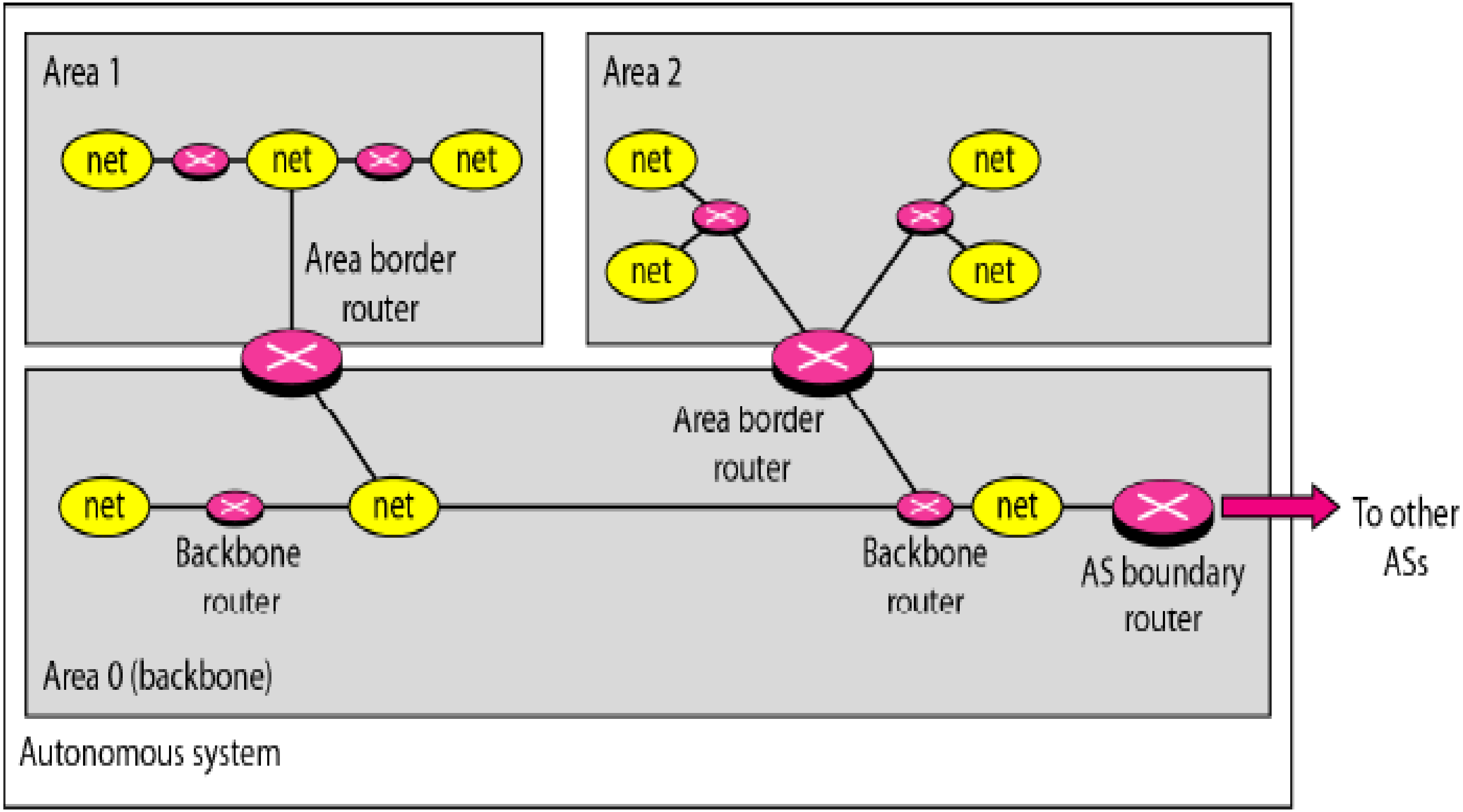
Open Shortest Path First(OSPF) Routing Protocol

- The Open Shortest Path First or OSPF protocol is an intra domain routing protocol based on link state routing. Its domain is also an autonomous system.

- To handle routing efficiently and in a timely manner, OSPF divides an autonomous system into areas. An area is a collection of networks, hosts, and routers all contained within an autonomous system. All networks inside an area must be connected.

Delivery, Forwarding, and Routing

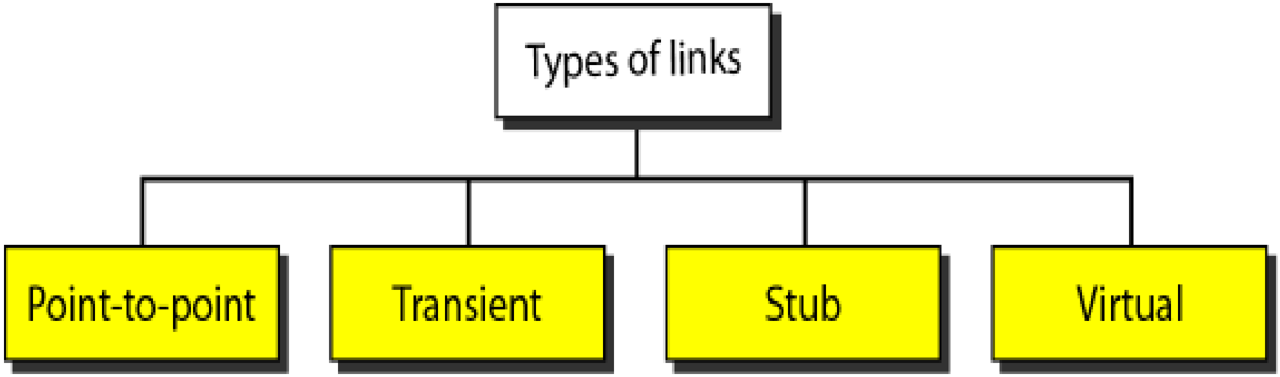
-



Delivery, Forwarding, and Routing

Metric: The OSPF protocol allows the administrator to assign a cost, called the metric, to each route. The metric can be based on a type of service (minimum delay, maximum throughput, and so on).

Types of Links: In OSPF terminology, a connection is called a link. Four types of links have been defined: point-to-point, transient, stub, and virtual.



Delivery, Forwarding, and Routing

Point-to-Point link

A point-to-point link connects two routers without any other host or router in between. There is no need to assign a network address to this type of link.

Transient link

A transient link is a network with several routers attached to it. The data can enter through any of the routers and leave through any router. All LANs and some WANs with two or more routers are of this type. In this case, each router has many neighbors.

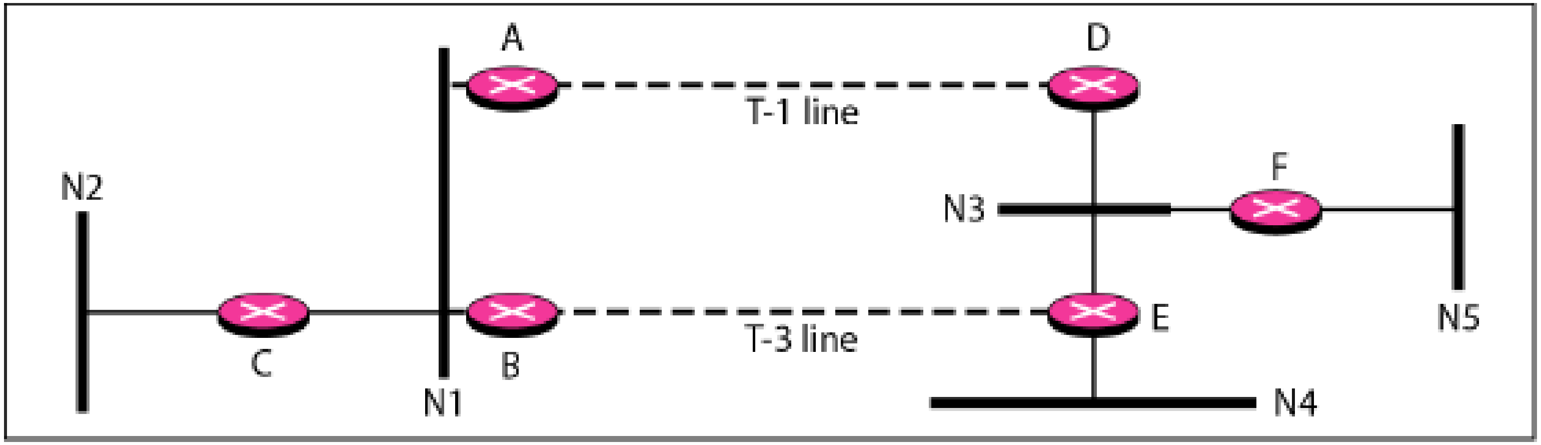
Stub link

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

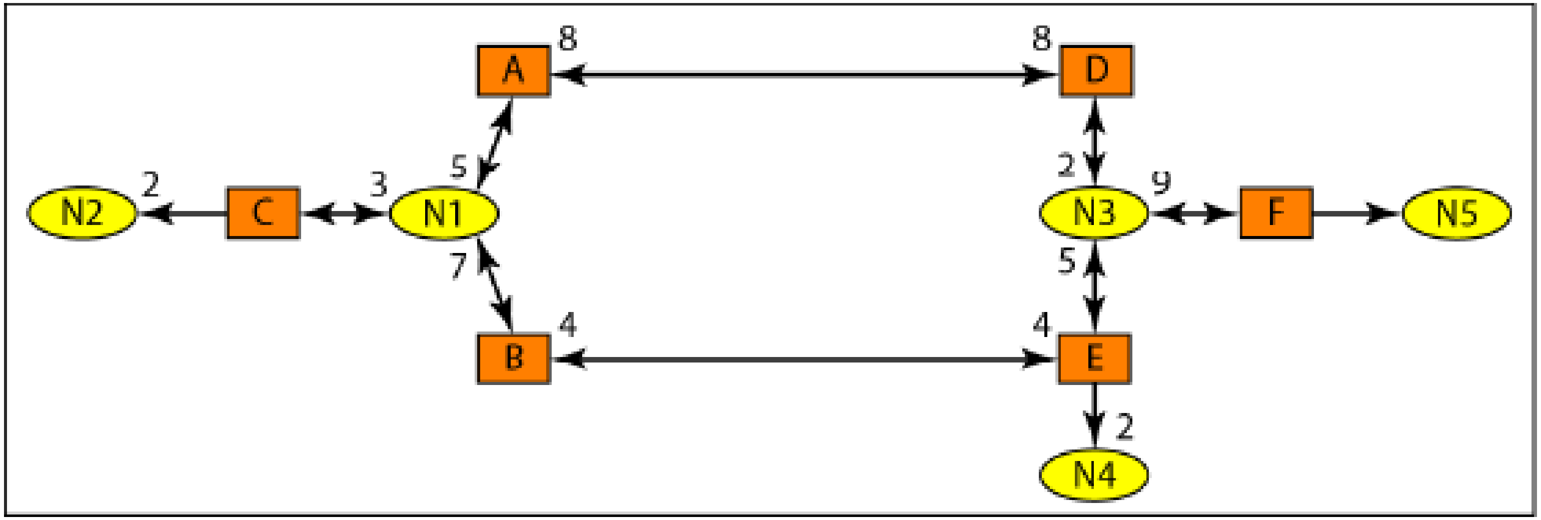
Virtual link

When the link between two routers is broken, the administration may create a virtual link between them, using a longer path that probably goes through several routers.

Example of an AS and its graphical representation in OSPF



a. Autonomous system



b. Graphical representation

AKTU Examination Questions

1. If a class B network on the Internet has a subnet mask of 255.255.248.0, what is the maximum number of hosts per subnet?
2. What is count-to-infinity problem?
3. What is time-to-live or packet lifetime?
4. What is unicast routing? Discuss unicast routing protocols.
5. Write advantages of Next-generation IPV6 over IPV4.
6. The IP network 200.198.160.0 is using subnet mask 255.255.255.224. Design the subnets.

AKTU Examination Questions

7. Write two use of subnet mask.
8. Convert the IPv4 address whose hexadecimal representation is C22F15B2 to dotted decimal notation. What is the class of this address?
9. What do you mean by adaptive and non-adaptive routing algorithm? Discus Distance Vector Routing including count to infinity problem.
10. Sketch the IP header neatly and explain the functions of each field. What are the deficiencies of IPV4 over IPV6?
11. An organization is granted a block 211.17.180.0 /24. The administrator wants to create 32 subnets.

    i) Find the subnet mask.

    ii) Find the number of addresses in each subnet.

    iii) Find the first & last address in subnet 1.

    iv) Find the first & last address in subnet 32.

AKTU Examination Questions

12. Given the IP address 180.25.21.I72 and the subnet mask 255.255.192.0, what is the subnet address?
13. What is IP addressing? How it is classified? How is subnet addressing is performed?
14. What is unicast routing? Discuss unicast routing protocols.
15. With the given IP-address, how will you extract its net-id and host-id?
16. Describe the problem of count to infinity associated with distance vector routing technique.
17. Given the IP address 180.2 5.21 .I72 and the subnet mask 255.255.192.A, what is the subnet address?

AKTU Examination Questions

18. What is the net mask of the gateway interface in a subnetwork where maximum of 25 hosts exist and IP address of one of the hosts is 192.168.1.1 ?

19. Define routing. In what way it is different from switching?

20. What is unicast routing? Discuss unicast routing protocols.

21. Find the class of each address

    (a) 140.213.10.80

    (b) 52.15.150.11

22. What is the type of the following address?

    (a) 4F::A234:2

    (b) 52F::1234:2222