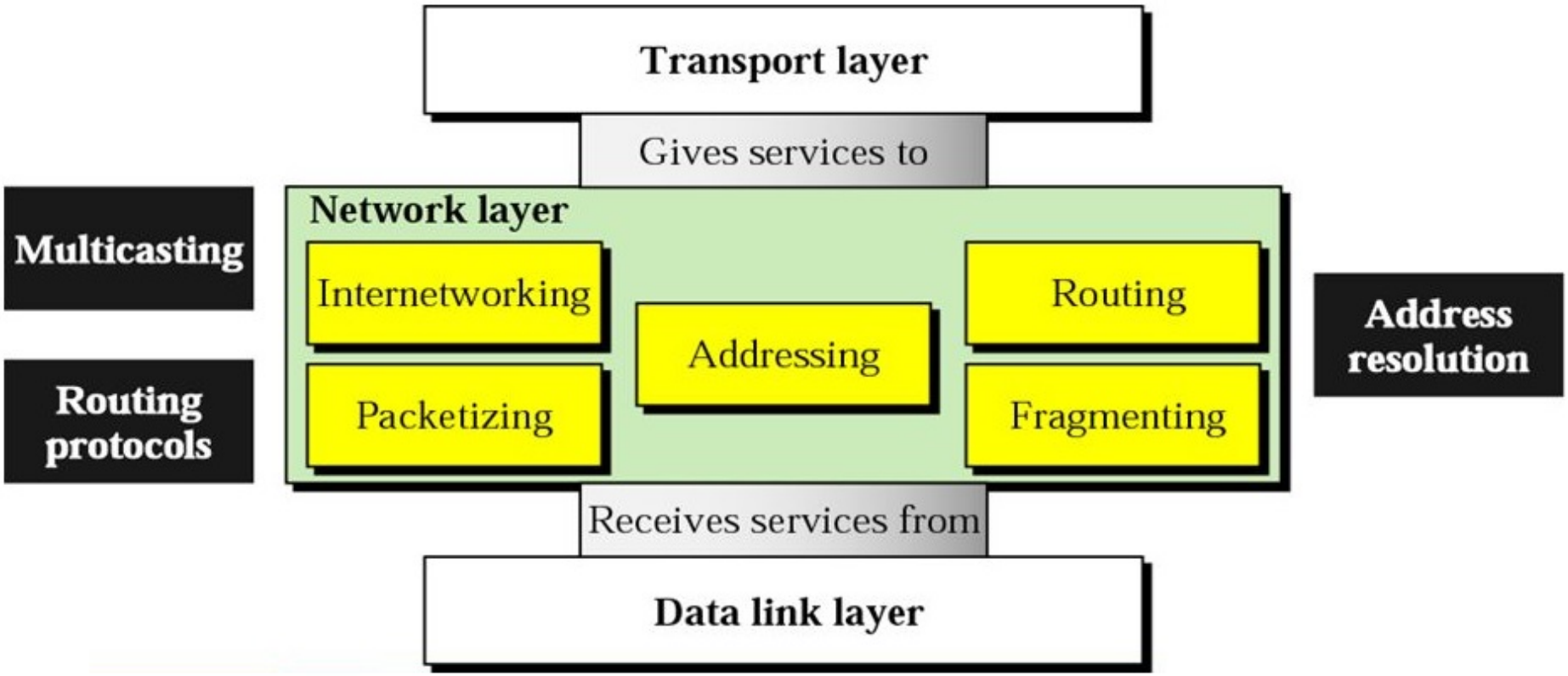
Module 4
Network Layer

Network layer



Network layer

- Packets: It is a protocol data unit utilized in the network layer.
  The source and destination MAC addresses are included in the framing. In contrast, the source and destination IP addresses are included in the packetization process.

Network layer

- Packets: It is a protocol data unit utilized in the network layer.
  The source and destination MAC addresses are included in the framing. In contrast, the source and destination IP addresses are included in the packetization process.
- Source to destination delivery of packets: If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery.

Network layer

- Packets: It is a protocol data unit utilized in the network layer.
  The source and destination MAC addresses are included in the framing. In contrast, the source and destination IP addresses are included in the packetization process.
- Source to destination delivery of packets: If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery.
- Logical addressing: If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, includes the logical addresses of the sender and receiver.

Network layer

- Packets: It is a protocol data unit utilized in the network layer.
  The source and destination MAC addresses are included in the framing. In contrast, the source and destination IP addresses are included in the packetization process.
- Source to destination delivery of packets: If two systems are connected to the same link, there is usually no need for a network layer. However, if the two systems are attached to different networks (links) with connecting devices between the networks (links), there is often a need for the network layer to accomplish source-to-destination delivery.
- Logical addressing: If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems. The network layer adds a header to the packet coming from the upper layer that, includes the logical addresses of the sender and receiver.
- Routing: When independent networks or links are connected to create internetworks (network of networks) or a large network, the connecting devices (called routers or switches) route or switch the packets to their final destination.

Addressing

- Addressing: At the network layer, we need to uniquely identify each device on the internet to allow global communication between all devices.
- The identifier used in the network layer of the Internet model to identify each device connected to the internet is called Internet Address or IP address.
- An IP address is a 32-bit binary address that uniquely and universally defines the connection of a host or a router to the internet.

Addressing

- There are two common notations to show an IP address;
  (1) Binary notation
  (2) Dotted decimal notation

Addressing

- There are two common notations to show an IP address;
  (1) Binary notation
  (2) Dotted decimal notation
- In binary notation, the IP address is displayed as 32 bits.
  It is often referred as a 32-bit address or 4-octet address or 4-byte address.

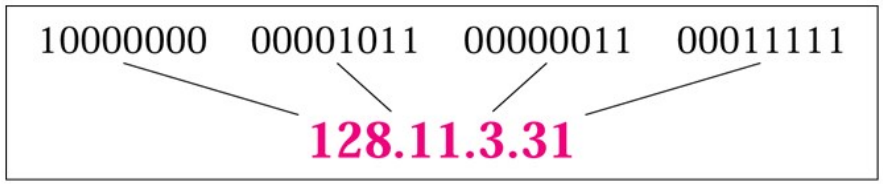
01110101 10010101 00011101 11101010

Addressing

- There are two common notations to show an IP address;
  (1) Binary notation
  (2) Dotted decimal notation
- In binary notation, the IP address is displayed as 32 bits.
  It is often referred as a 32-bit address or 4-octet address or 4-byte address.

01110101 10010101 00011101 11101010

- To make IP address more compact and easier to read, IP addresses are usually written in decimal form.



Note: Since each byte(octet) is only 8 bits, each number in the dotted decimal is between 0 and 255.

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 whole address space.

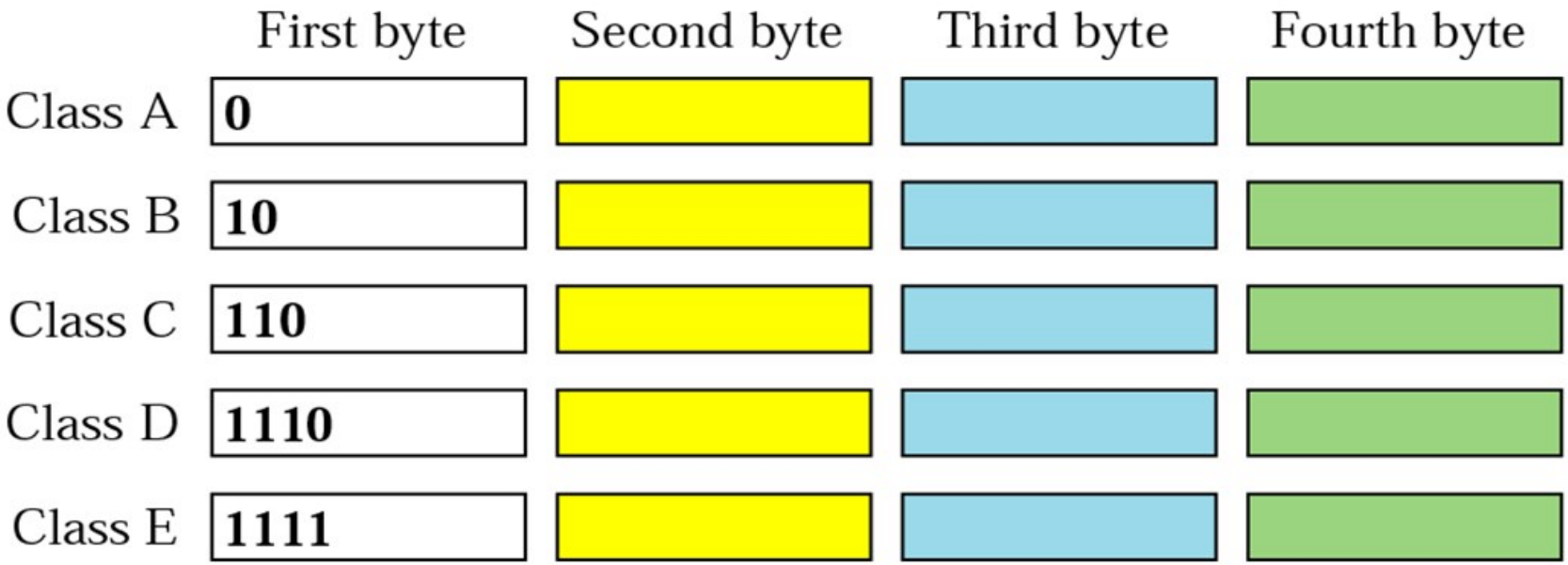


Figure 1 : Finding the class in binary notation

Addressing

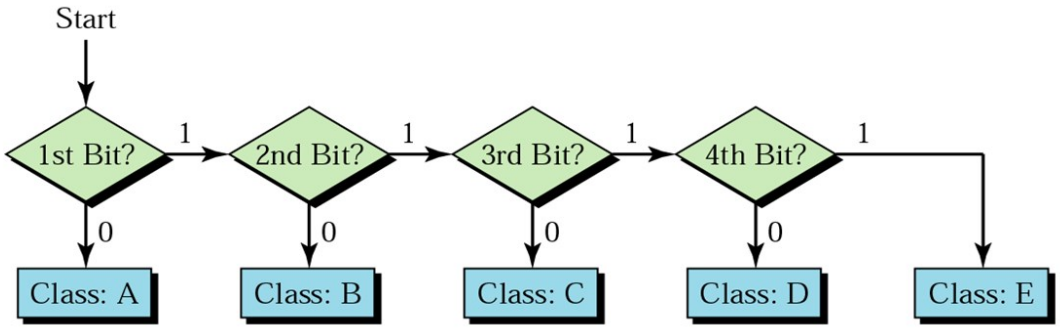


Figure 2 : Finding the address class

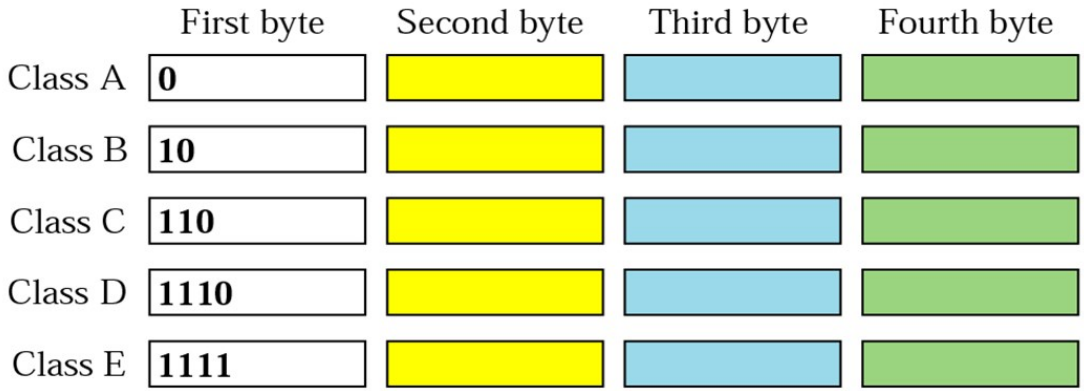


Figure 3 : Finding the class in binary notation

Addressing

- Finding the class in Dotted decimal notation:
  Each class has a specific range of numbers.

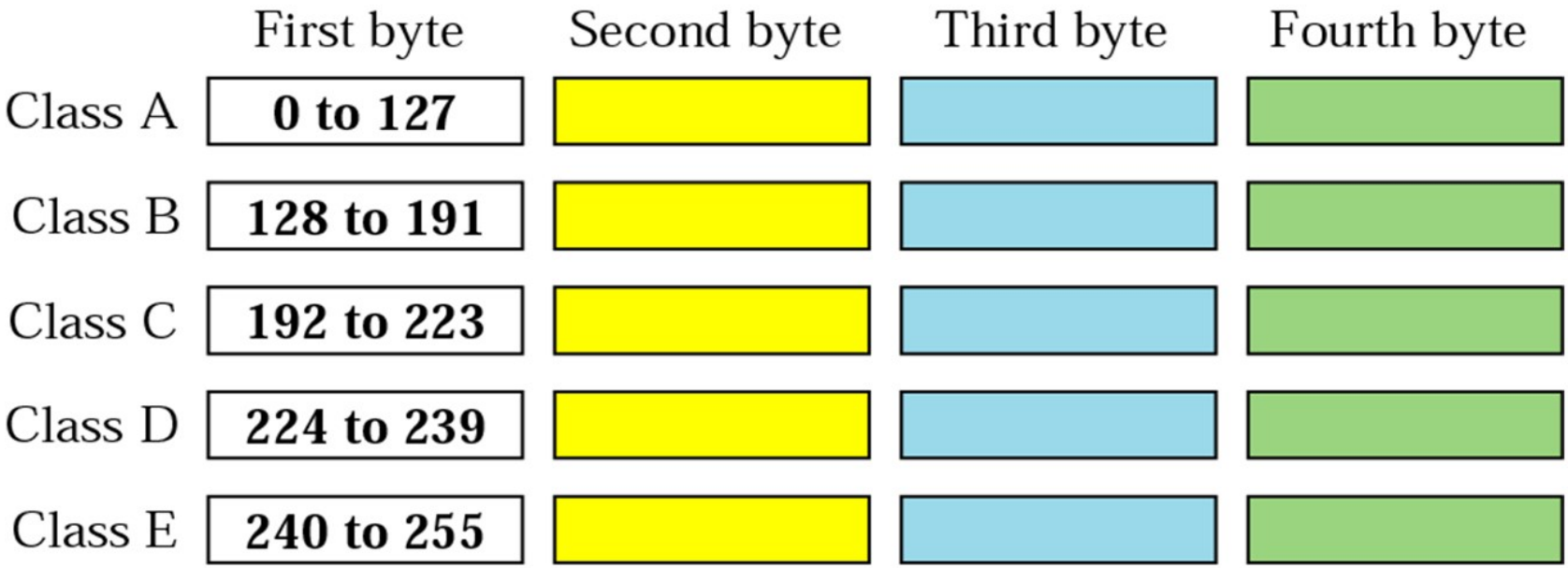


Figure 4 : Finding the class in dotted decimal notation

Addressing

- Unicast, Multicast and Reserved Addresses:
  Addresses in class A, B and C are for unicast communication, from one source to one destination.
  A host needs atleast one unicast address to be able to send or receive packets.
- Addresses in class D are for multicast communication, from one source to group of destinations.
  A multicast address can be used only as a destination address, but never as a source address.
- Addresses in class E are reserved.
  They are used for some special purpose.

Addressing

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

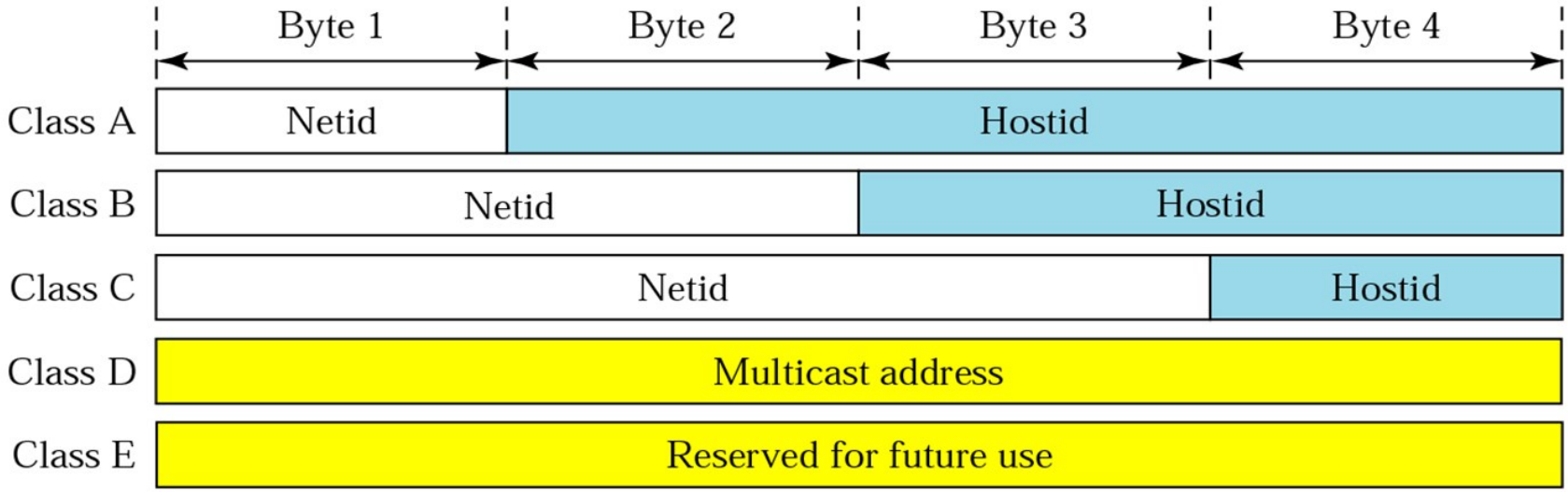


Figure 5 : Netid and Hostid

Addressing

- In class A, one byte defines the netid and three bytes defines the hostid.
- In class B, two byte defines the netid and two bytes defines the hostid.
- In class C, three byte defines the netid and one bytes defines the hostid.
- Classes D and E are not divided into netid and hostid

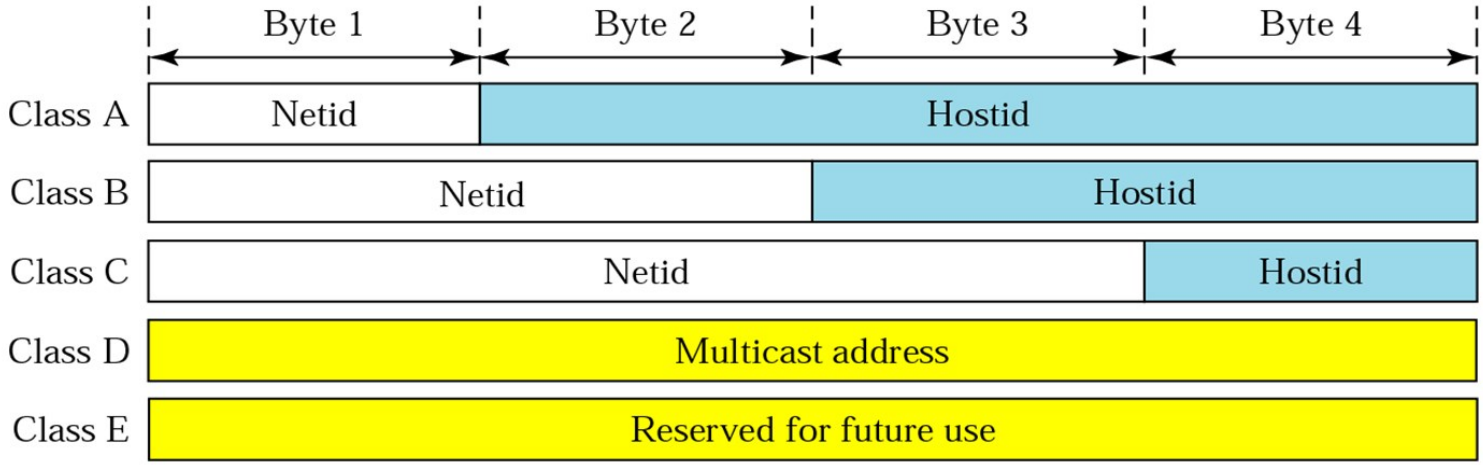


Figure 6 : Netid and Hostid

Addressing

- Exercise1: Change the following IP address from binary notation to dotted decimal notation.
  (a) 10000001 00001011 00001011 11101111
  (b) 11111001 10011011 11111011 00001111

Addressing

- Exercise1: Change the following IP address from binary notation to dotted decimal notation.
  (a) 10000001 00001011 00001011 11101111
  (b) 11111001 10011011 11111011 00001111
  Solution:
  (a) 129.11.11.239
  (b) 249.155.251.15

Addressing

- Exercise2: Change the following IP address from dotted decimal notation to binary notation.
  (a) 111.56.45.78
  (b) 75.45.34.78

Addressing

- Exercise2: Change the following IP address from dotted decimal notation to binary notation.
  (a) 111.56.45.78
  (b) 75.45.34.78
  Solution:
  (a) 01101111 00111000 00101101 01001110
  (b) 01001011 00101101 00100010 01001110

Addressing

- Exercise3: Find the class of each address;
  (a) 00000001 00001011 00001011 11101111
  (b) 11110011 10011011 11111011 00001111

Addressing

- Exercise3: Find the class of each address;
  (a) 00000001 00001011 00001011 11101111
  (b) 11110011 10011011 11111011 00001111
  Solution:
  (a) The first bit is 0; so (a) is class A address
  (b) The first 4 bits are 1; so (b) is class E address

Addressing

- Exercise4: Find the class of each address;
  (a) 227.12.14.87
  (b) 252.5.15.111
  (c) 134.11.78.56

Addressing

- Exercise4: Find the class of each address;
  (a) 227.12.14.87
  (b) 252.5.15.111
  (c) 134.11.78.56
  Solution:
  (a) The first byte is 227 (between 224 and 239); so (a) is class D address
  (b) The first byte is 252 (between 240 and 255); so (b) is class E address
  (c) The first byte is 134 (between 128 and 191); so (c) is class B address

Blocks in Class A, B, C, D and E Addressing

Addressing

- Class A is divided into 128 blocks with each block having a different netid.
- The first block covers addresses from 0.0.0.0 to 0.255.255.255(netid 0)
  The second block covers addresses from 1.0.0.0 to 1.255.255.255(netid 1)
  The last block covers addresses from 127.0.0.0 to 127.255.255.255(netid 127)
- For each block of addresses the first byte(netid) is the same, but the other 3 bytes (hostid) can take any value in the given range.

- Figure7 shows how an organization granted a block with netid 73 uses its addresses.

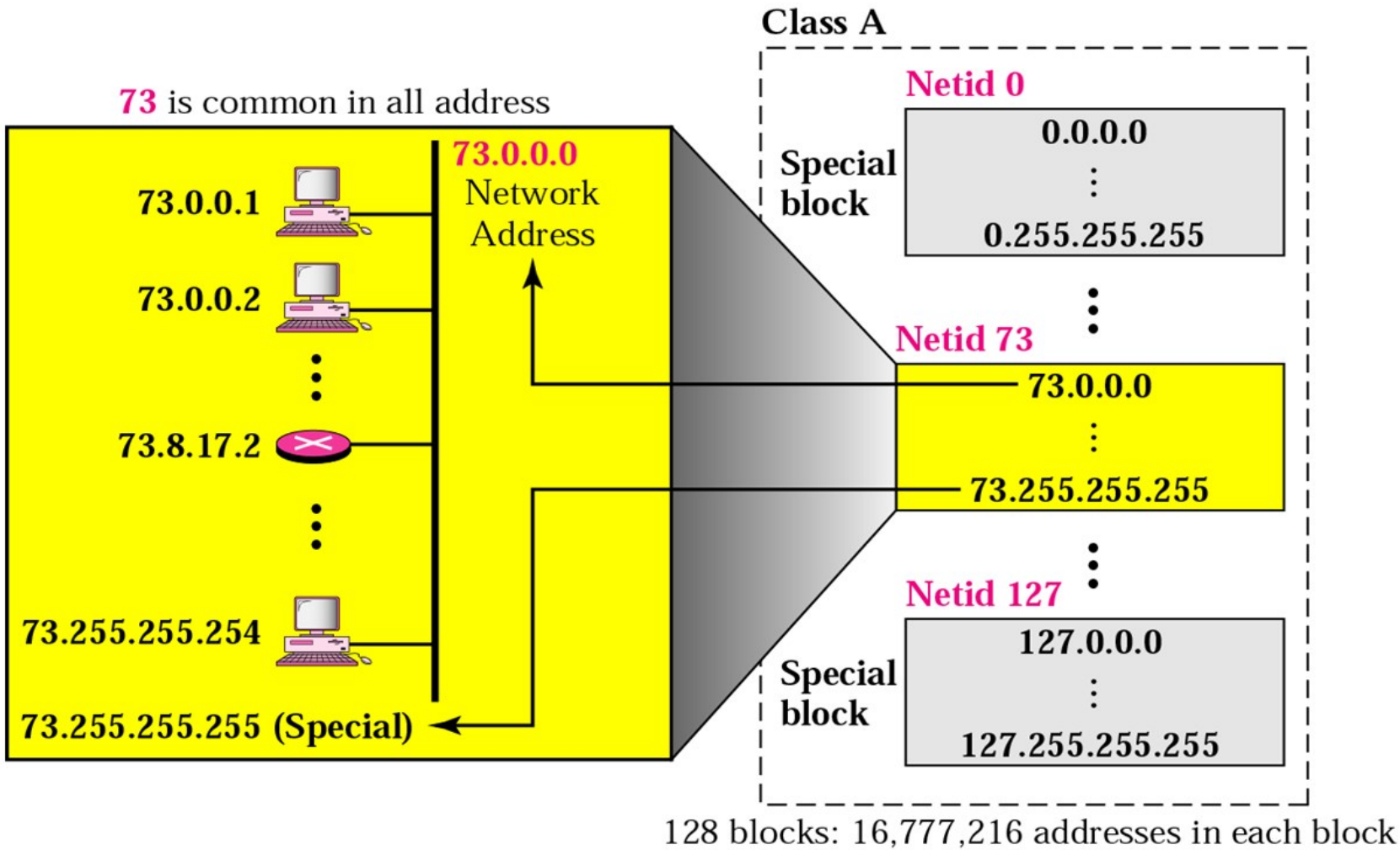


Figure 7 : Blocks in class A

Addressing

- The first address in the block is used to identify the organization to the rest of the internet. This address is called network address.
- The organization is not allowed to use the last address; it is reserved for a special purpose.
- Class A addresses were designed for large organization with a large number of hosts or routers attached to their network.
- The number of addresses in each block is 16777216, is probably larger than the needs of almost all organizations.
  Millions of class A addresses are wasted in this class.

Addressing

- Class B is divided into 16384 blocks with each block having a different netid.
- Sixteen blocks are reserved for private addresses, leaving 16368 blocks for assignment to organizations.
- The first block covers addresses from 128.0.0.0 to 128.0.255.255 (netid 128.0)
  The last block covers addresses from 191.255.0.0 to 191.255.255.255 (netid 191.255)
- Note that, for each block of addresses the first two bytes (netid) is the same, but the other 2 bytes (hostid) can take any value in the given range.

- Figure8 shows the blocks in Class B

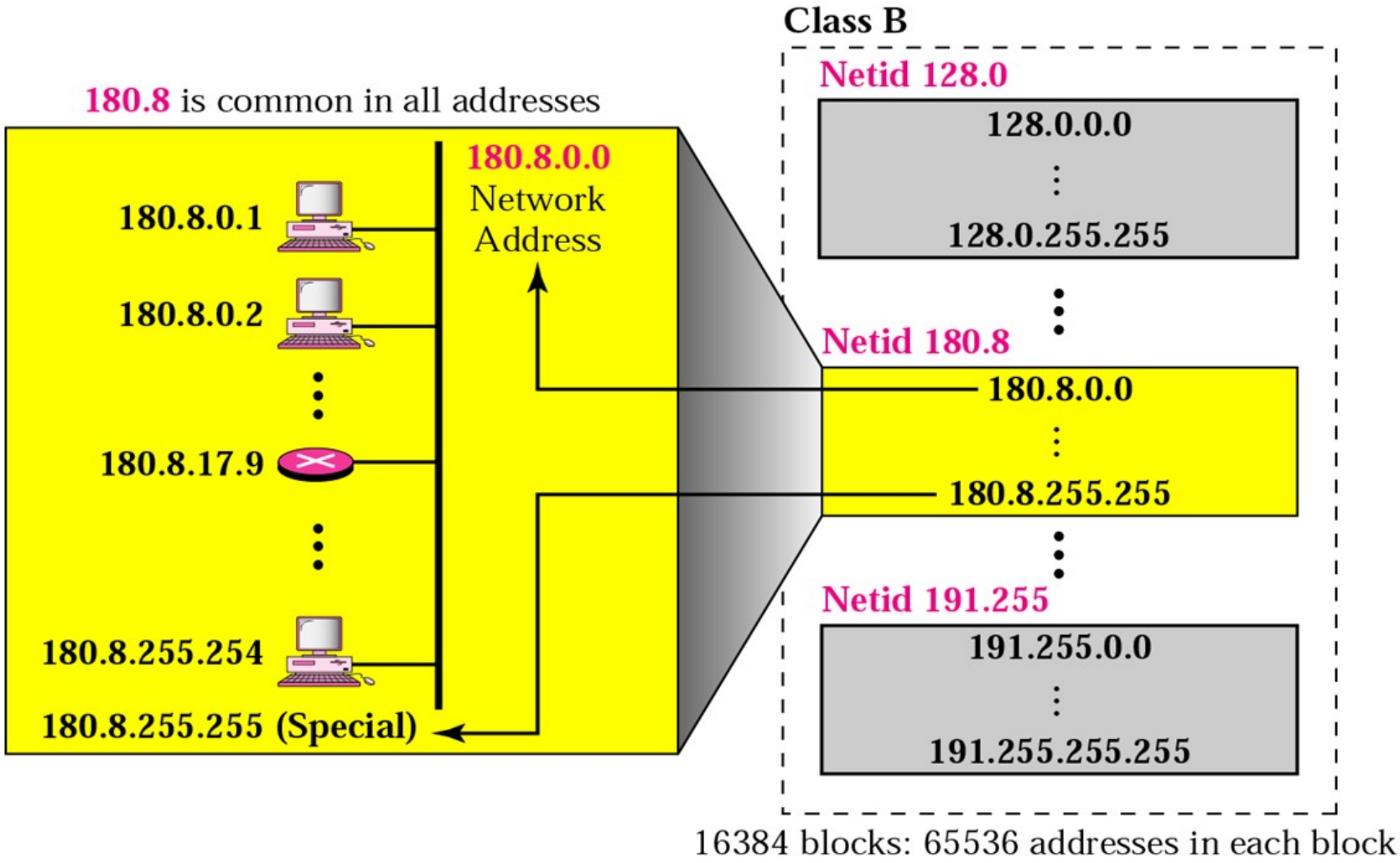


Figure 8 : Blocks in class B

Addressing

- There are 16368 blocks that can be assigned. (i.e) the total number of organizations that can have a class B address is 16368.
- However, each block in this class contains 65536 addresses, the organization should be large enough to use all these addresses.
- Class B addresses were designed for midsize organizations that may have thousands of hosts or routers attached to their networks.
- However, the number of addresses in each block is 65536, larger than the needs of most midsize organizations.
  Many addresses are also wasted in this class.

Addressing

- Class C is divided into 2097152 blocks with each block having a different netid.
- 256 blocks are used for private addresses, leaving 2096896 blocks for assignment to organizations.
- The first block covers addresses from 192.0.0.0 to 192.0.0.255 (netid 192.0.0)
  The last block covers addresses from 223.255.255.0 t0 223.255.255.255 (netid 223.255.255)
- Note that, for each block of addresses the first three bytes (netid) is the same, but the remaining byte (hostid) can take any value in the given range.

- Figure9 shows the blocks in Class C

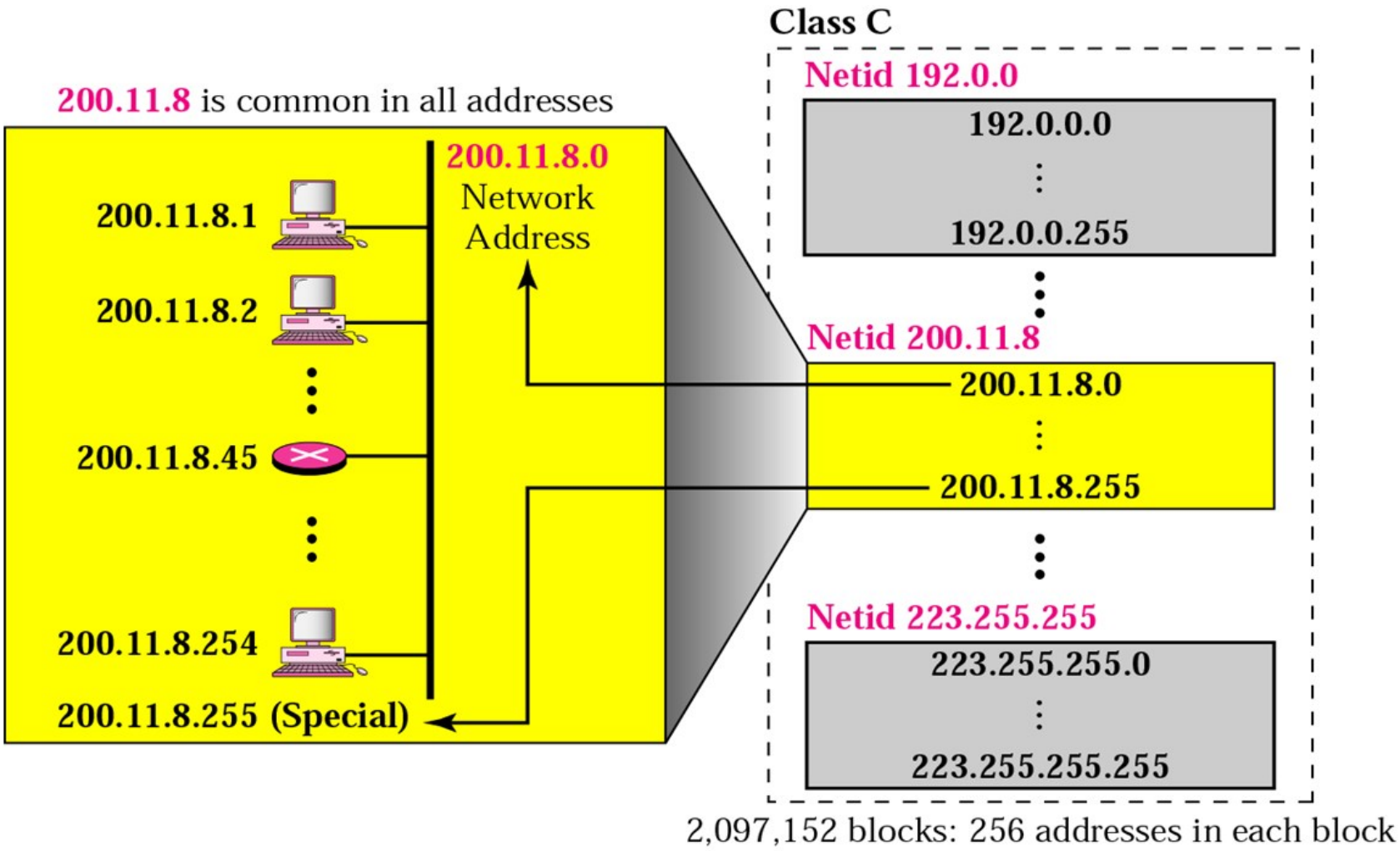


Figure 9 : Blocks in class C

Addressing

- There are 2096902 blocks that can be assigned. (i.e) the total number of organizations that can have a class C address is 2096902.
- However, each block in this class contains 256 addresses, which means the organization should be small enough to need less than 256 addresses.
- Class B addresses were designed for small organizations with a small number of hosts or routers attached to their networks.
- However, the number of addresses in each block is so limited that most organizations do not want a block in this class.
  The number of addresses in class C is smaller than the needs of most organizations.

Addressing

- Class D
  There is one block of class D addresses. It is designed for multicasting.
- Class E
  There is one block of class E addresses. It is designed for use of reserved addresses.

Addressing

- Network Address is an address that defines the network itself; it cannot be assigned to a host.

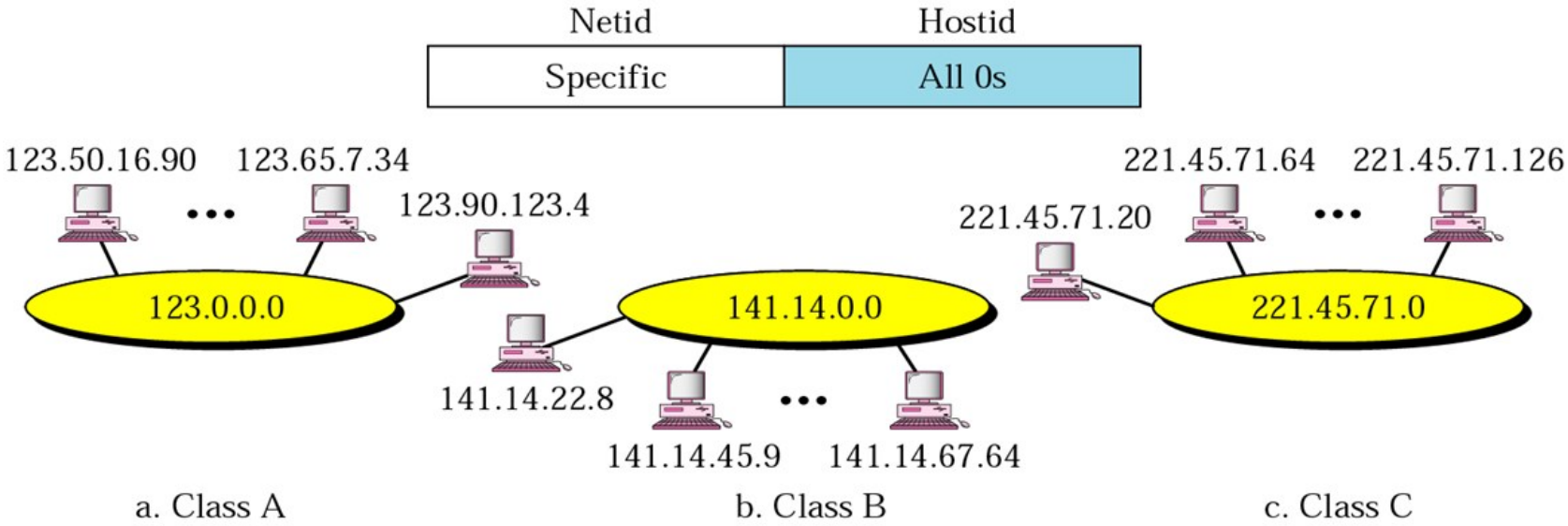


Figure 10 : Network address

- Figure10 shows three examples of network addresses, one for each class

Addressing

- Network Address play a vital role in classful addressing. It has several properties;
  (1) All hostid bytes are 0s
  (2) The network address defines the network to the rest of the internet
  (3) The network address is the first address in the block
  (4) Given the network address, we can find the class of the address

Addressing

- Exercise1: Given the address 23.56.7.91, Find the network address?

Addressing

- Exercise1: Given the address 23.56.7.91, Find the network address?
- Solution:
  The class is A.
  Only the first bytes defines the netid.
  We can find the network address by replacing the hostid bytes 56.7.91 as 0s.
  Therefore, the network address is 23.0.0.0

Addressing

- Exercise2: Given the address 132.6.17.85, Find the network address?

Addressing

- Exercise2: Given the address 132.6.17.85, Find the network address?
- Solution:
  The class is B.
  The first two bytes defines the netid.
  We can find the network address by replacing the hostid bytes 17.85 as 0s.
  Therefore, the network address is 132.6.0.0

Addressing

- Exercise3: Given the network address 17.0.0.0, Find the class?

Addressing

- Exercise3: Given the network address 17.0.0.0, Find the class?
- Solution:
  The class is A, since the netid is only 1 byte.

Addressing

A network address is different from a netid. A network address has both netid and hostid, with 0s for the hostid.

Addressing

- A sample internet with classful addressing:
  Figure11 shows a part of an internet with different networks.
  - An Token Ring LAN with network address 220.3.6.0 (class C)
  - An Ethernet LAN with network address 134.18.0.0 (class B)
  - An Ethernet LAN with network address 124.0.0.0 (class A)
  - A point to point WAN(broken line)
- One router connects the WAN to the Token Ring network. One connects the WAN to one of the Ethernet networks and the other one connects the WAN to the rest of the internet.

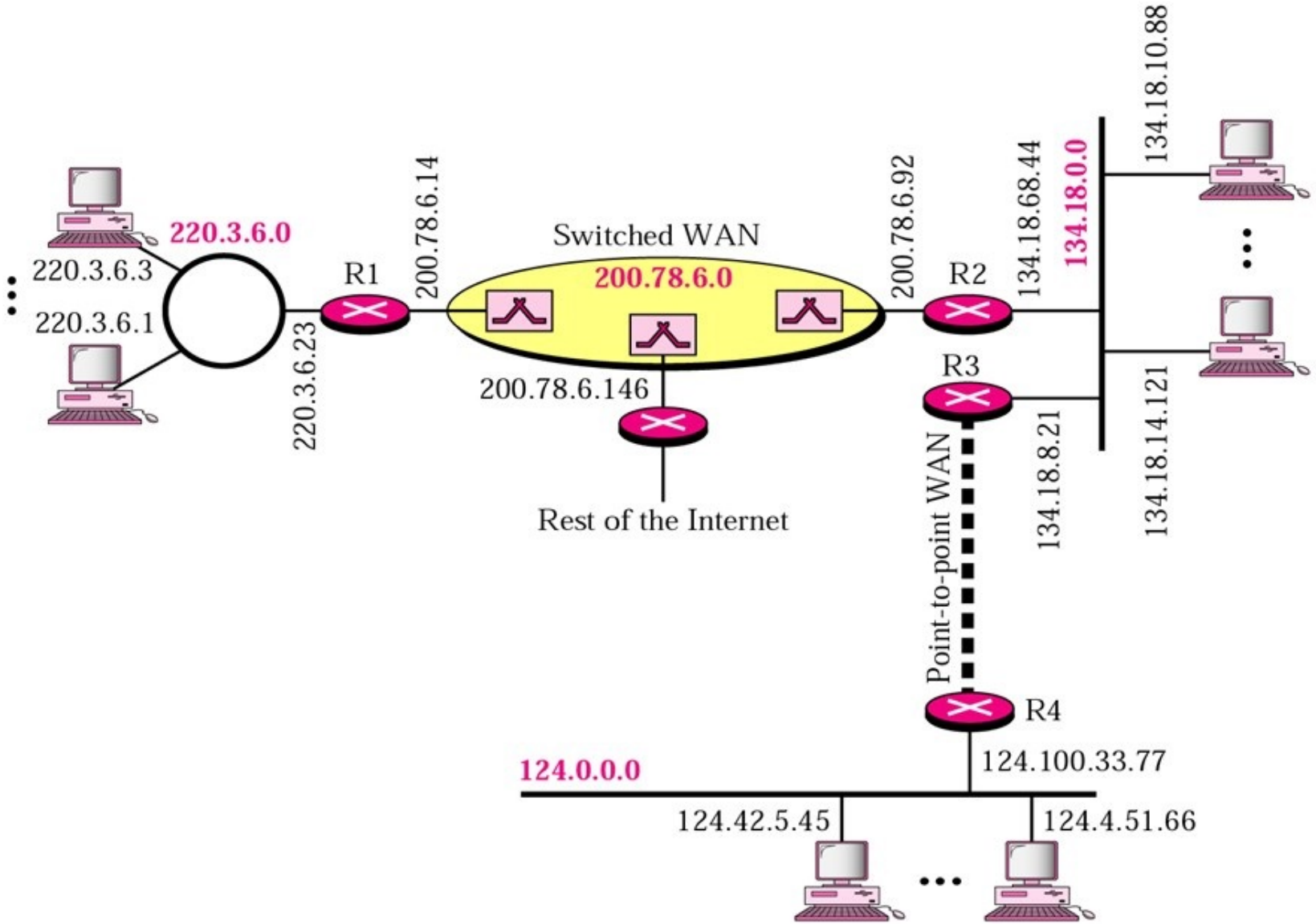


Figure 11 : Sample Internet

Subnetting

Subnetting

- Subnetting: IP addresses are designed with two levels of hierarchy. Figure12 shows the concept.

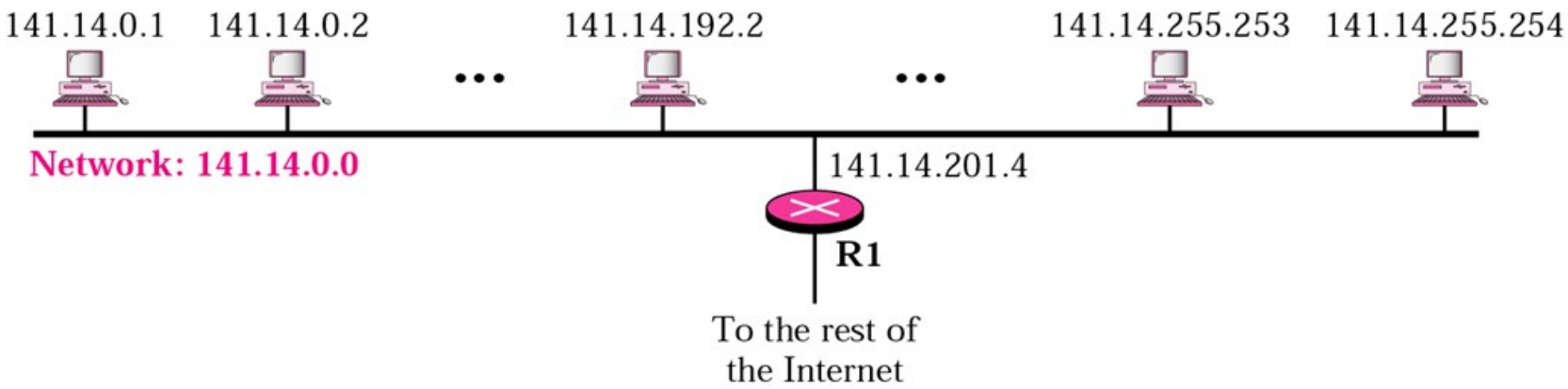


Figure 12 : A network with two levels of hierarchy

- An organization needs to assemble the hosts into groups; the network needs to be divided into several subnetworks(subnets).

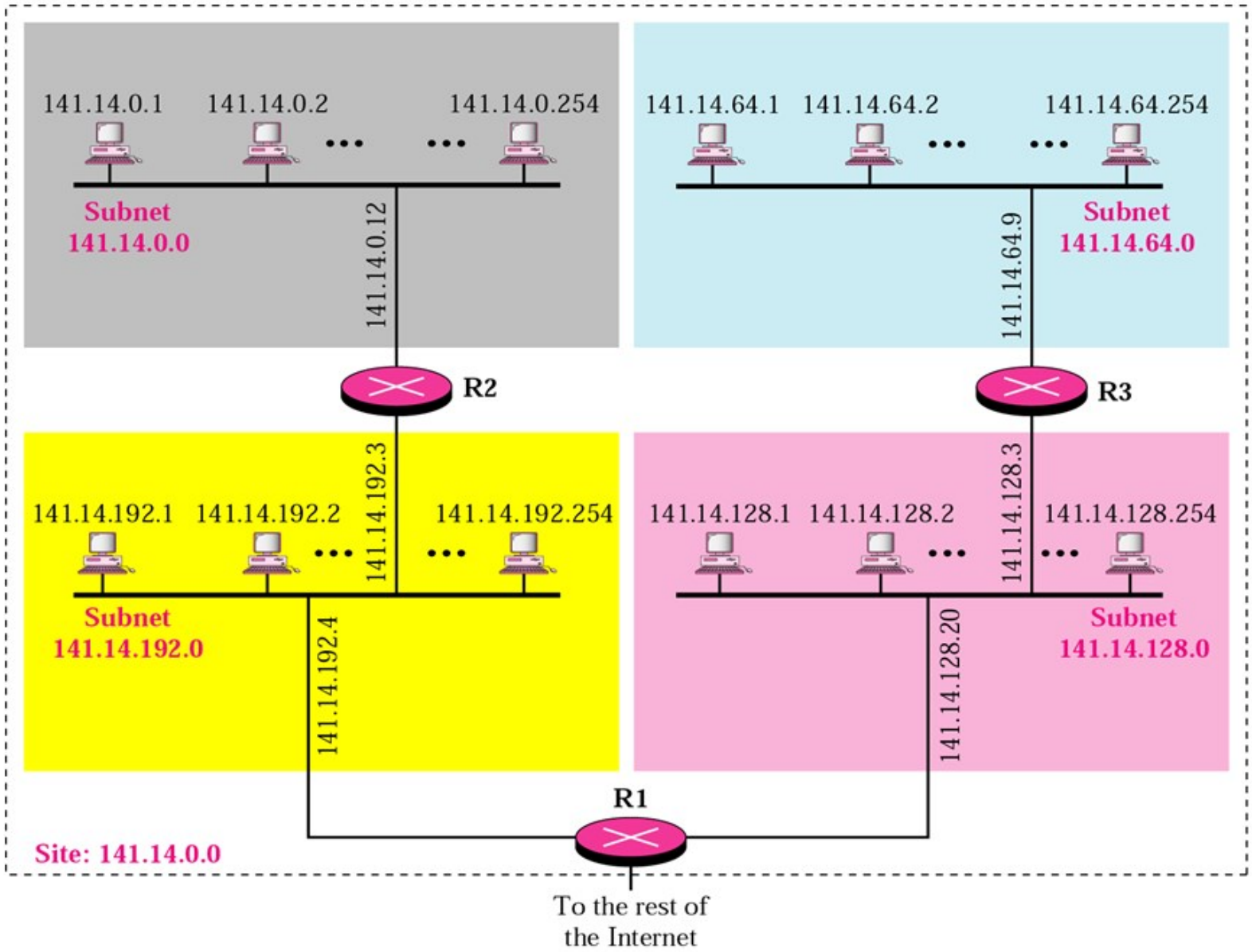


Figure 13 : A network with three levels of hierarchy

Subnetting

- For example, a university may want to group its hosts according to department. In this case, the university has one network address, but needs several subnetwork addresses.
- The outside world knows the organization by its network address.
  Inside the organization, each subnetwork is recognized by its subnetwork address.
- As in Figure13, the rest of the internet is not aware that the network is divided into physical subnetworks.
  The subnetworks still appear as a single network to the rest of the internet.
  A packet destined for host 141.14.192.2 still reaches the router R1.
- However, when the datagram arrives at router R1, the interpretation of the IP address changes.
  Router R1 knows that network 141.14 is physically divided into subnetworks.
  (i.e) it knows that the packet must be delivered to subnet 141.14.192.0

Subnetting

- Adding subnetworks creates an intermediate level of hierarchy in the IP addressing system. The three levels are;
  (1) site
  (2) subnet
  (3) host

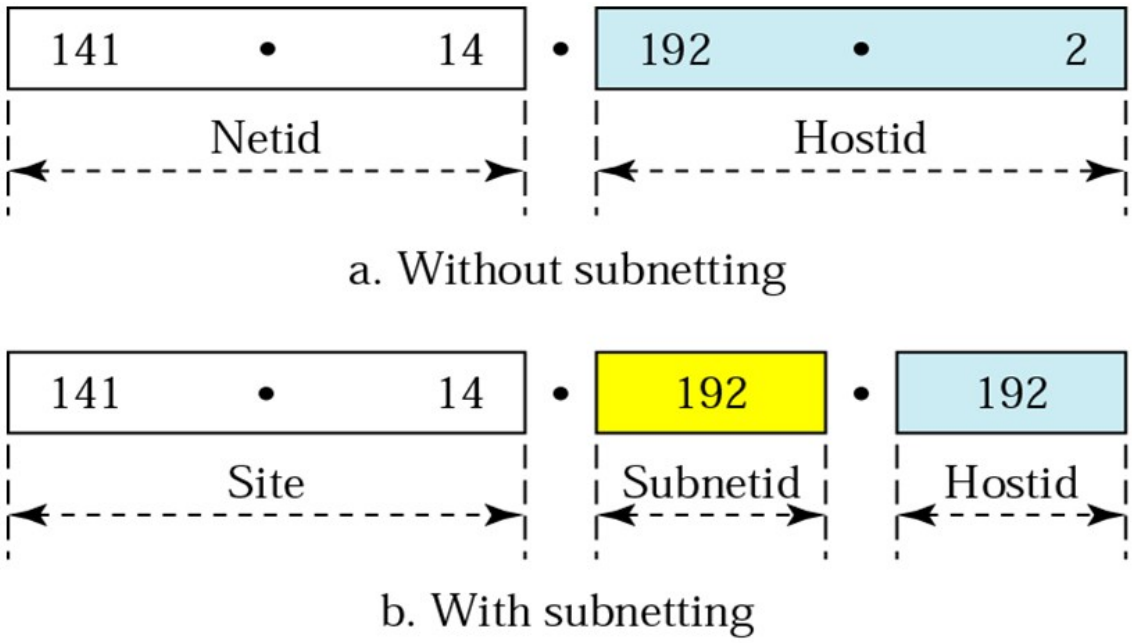


Figure 14 : Addresses in a network with and without subnetting

Subnetting

- Mask: When a router receives a packet with a destination address, it needs to route the packet.
- The routing is based on the network address and subnetwork address.
  The routers outside the organization routes the packet based on the network address;
  the router inside the organization routes the packet based on the subnetwork address.
- The question is, how a router find the network address and subnetwork address?
  A network administrator knows the network address and the subnetwork addresses, but the router does not.
  The router outside the organization has a routing table with one column based on the network addresses;
  the router inside the organization has a routing table based on the subnetwork addresses.
- A 32 bit number called the mask is the key.
  The routers outside the organization use a default mask; the routers inside the organization use a subnet mask.

Subnetting

- A Default Mask is a 32-bit binary number that gives the network address when ANDed with an address in the block.
- AND operation does the following;
  (1) If the bit in the mask is 1, the corresponding bit in the address is retained in the output(no change)
  (2) If the bit in the mask is 0, a 0 bit in the output is the result.

- Figure15 shows the default mask for each class;
  (1) For class A, the mask is eight 1s and twenty four 0s
  (2) For class B, the mask is sixteen 1s and sixteen 0s
  (3) For class C, the mask is twenty four 1s and eight 0s
- An alternative mask notation is a slash followed by the number of 1s. This is called slash notation.

Class	*In Binary*	*In Dotted-Decimal*	*Using Slash*
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 111111111 11111111 00000000	255.255.255.0	/24

Figure 15 : Default masks

- Note that, the number of 1s in each class matches the number of bits in the netid and the number of 0s matches the number of bits in the hostid.
  In other words, when a mask is ANDed with an address, the netid is retained and the hostid is set to 0s.

The network address can be found by applying the default mask to any address in the block (including itself). It retains the netid of the block and sets the hostid to 0s.

- Exercise 1: A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.

- Exercise 1: A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.
- Solution: The router follows three steps;

- Exercise 1: A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.
- Solution: The router follows three steps;
  (1) The router looks at the first byte of the address to find the class. It is class B

- Exercise 1: A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.
- Solution: The router follows three steps;
  (1) The router looks at the first byte of the address to find the class. It is class B
  (2) The default mask for class B is 255.255.0.0. The router ANDs this mask with the address to get 190.240.0.0

- Exercise 1: A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.
- Solution: The router follows three steps;
  (1) The router looks at the first byte of the address to find the class. It is class B
  (2) The default mask for class B is 255.255.0.0. The router ANDs this mask with the address to get 190.240.0.0
  (3) The router looks in its routing table to find out how to route the packet to this destination.

Subnetting

- A Subnet Mask: The number of 1s in a subnet mask is more than the number of 1s in the corresponding default mask.
  In other words, in a subnet mask, we change some of the leftmost 0s in the default mask to make a subnet mask.

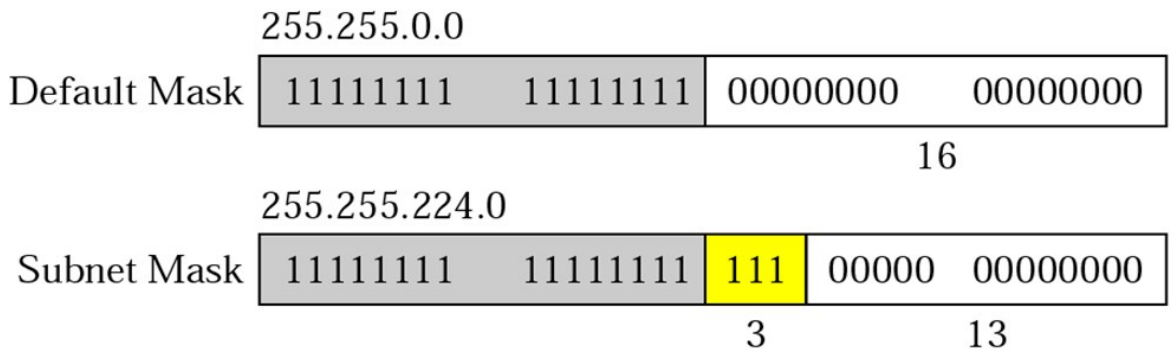
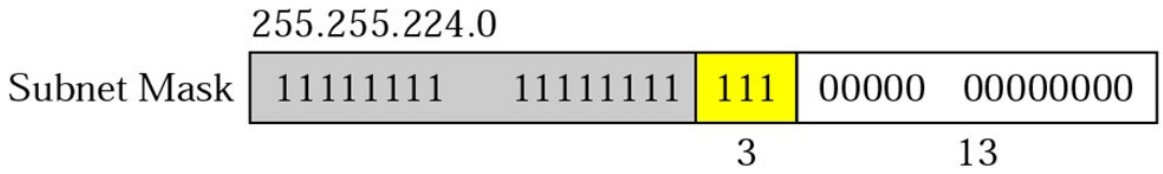


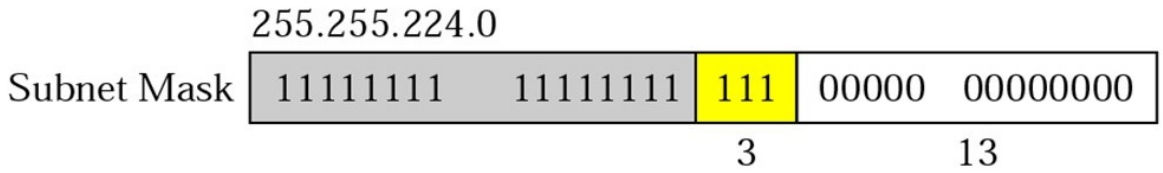
Figure 16 : Subnet mask

- The number of subnets is determined by the number of extra 1s.
  If the number of extra 1s is n, the number of subnets is 2^n
  If the number of subnets is N, the number of extra 1s is $log_2 N$

- Exercise 2: A router inside the organization receives a packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.
  Assume the subnetmask as

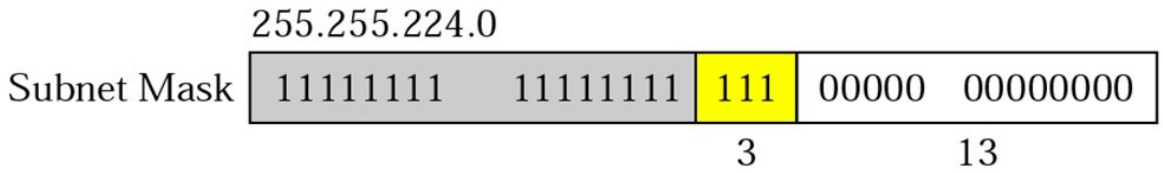


- Exercise 2: A router inside the organization receives a packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.
  Assume the subnetmask as



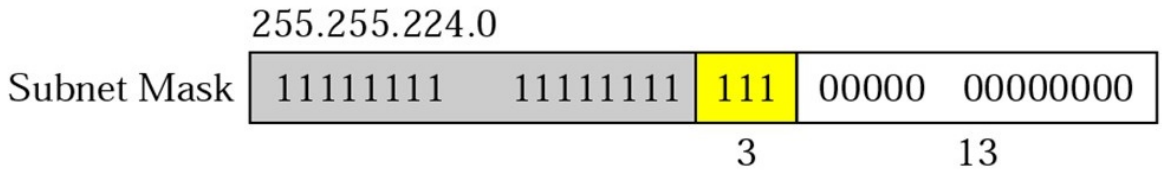
- Solution: The router follows three steps;

- Exercise 2: A router inside the organization receives a packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.
  Assume the subnetmask as



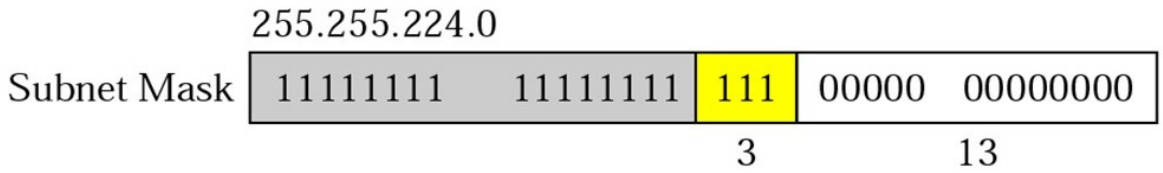
- Solution: The router follows three steps;
  (1) The router must know the mask. It is /19 as shown in Figure

- Exercise 2: A router inside the organization receives a packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.
  Assume the subnetmask as



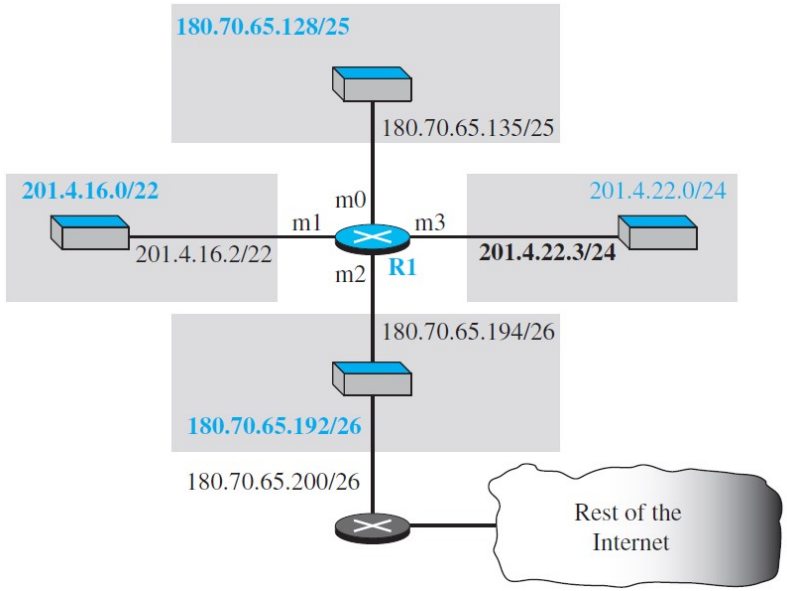
- Solution: The router follows three steps;
  (1) The router must know the mask. It is /19 as shown in Figure
  (2) The router applies the mask to the address, 190.240.33.91. The subnet address is 190.240.32.0

- Exercise 2: A router inside the organization receives a packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.
  Assume the subnetmask as

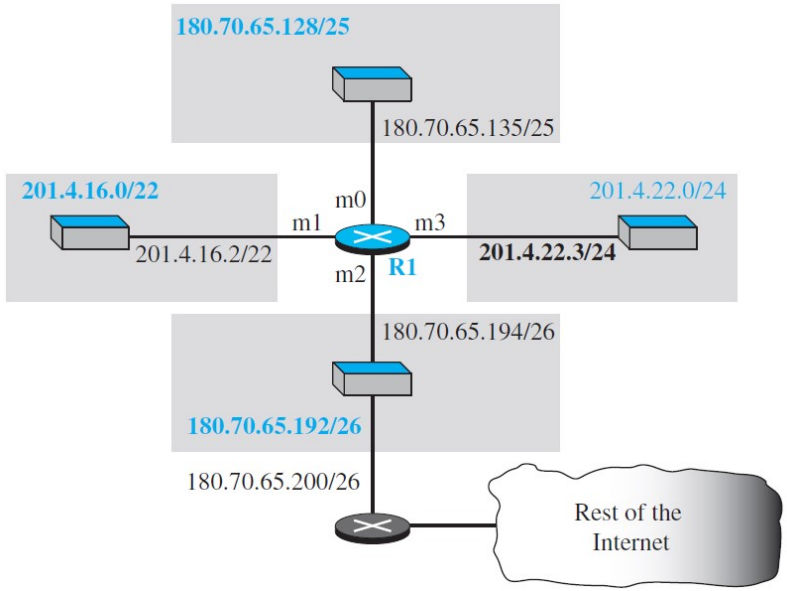


- Solution: The router follows three steps;
  (1) The router must know the mask. It is /19 as shown in Figure
  (2) The router applies the mask to the address, 190.240.33.91. The subnet address is 190.240.32.0
  (3) The router looks in its routing table to find out how to route the packet to this destination.

- Exercise 3: Make a forwarding table for router R1 using the configuration as in Figure



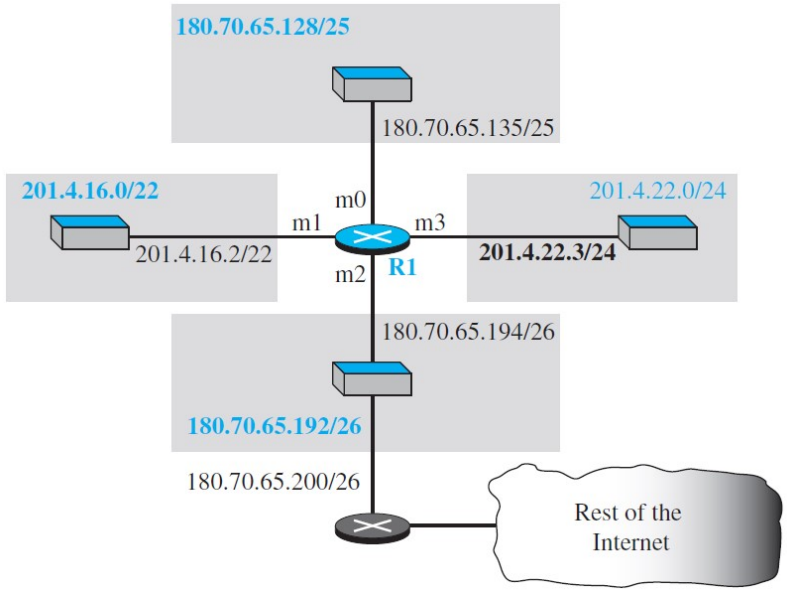
- Exercise 3: Make a forwarding table for router R1 using the configuration as in Figure



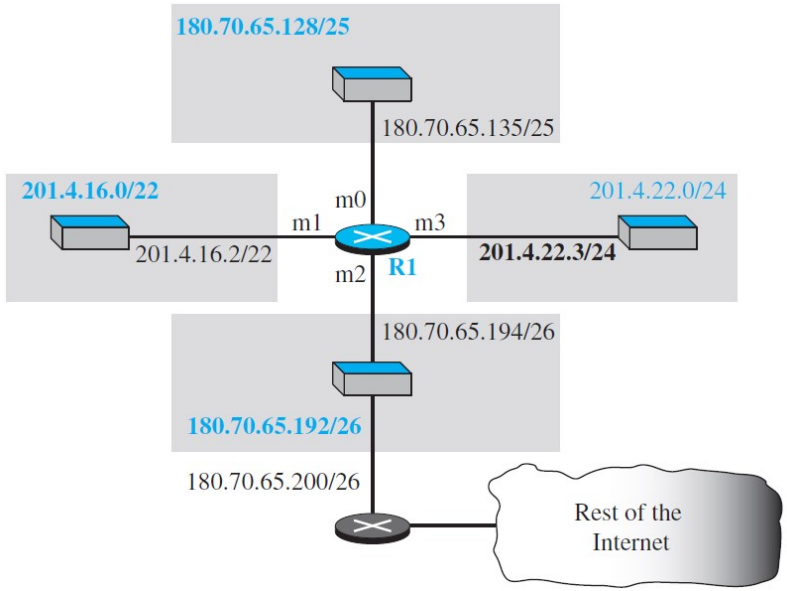
- Solution:

Network address/mask	*Next hop*	*Interface*
180.70.65.192/**26**	—	m2
180.70.65.128/**25**	—	m0
201.4.22.0/**24**	—	m3
201.4.16.0/**22**	—	m1
Default	180.70.65.200	m2

- Exercise 4: Show the forwarding process if a packet arrives at R1 in Figure with the destination address 180.70.65.140.



- Exercise 4: Show the forwarding process if a packet arrives at R1 in Figure 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. The next-hop address and the interface number m0 are extracted for forwarding the packet.

IPv4 Addressing

IPv4 Addressing

- Internet protocol is the host-to-host network layer delivery for the internet.
- IP is an unreliable and connectionless datagram protocol. (i.e.) it uses only an error detection mechanism and discards the packet if it is corrupted.
  If reliability is needed, IP must be paired with a reliable protocol such as TCP(at the transpost layer).

IPv4 Addressing

- Internet protocol is the host-to-host network layer delivery for the internet.
- IP is an unreliable and connectionless datagram protocol. (i.e.) it uses only an error detection mechanism and discards the packet if it is corrupted.
  If reliability is needed, IP must be paired with a reliable protocol such as TCP(at the transpost layer).

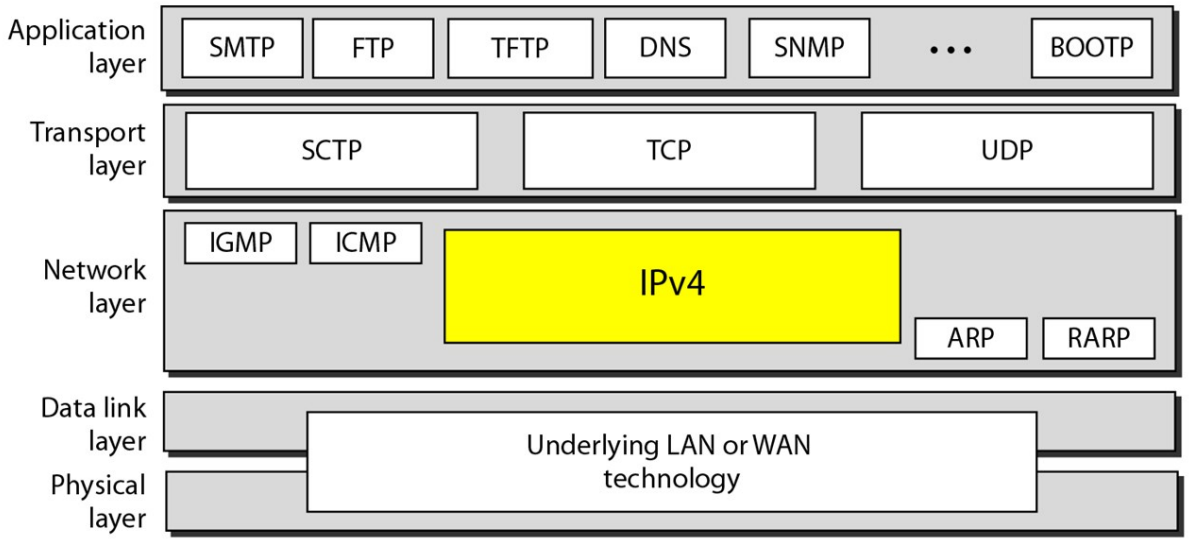


Figure 17 : Position of IPv4 in TCP/IP protocol suite

- Datagram: Packets in the IP layer are called datagrams. Figure18 shows the IP datagram format.
- 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.

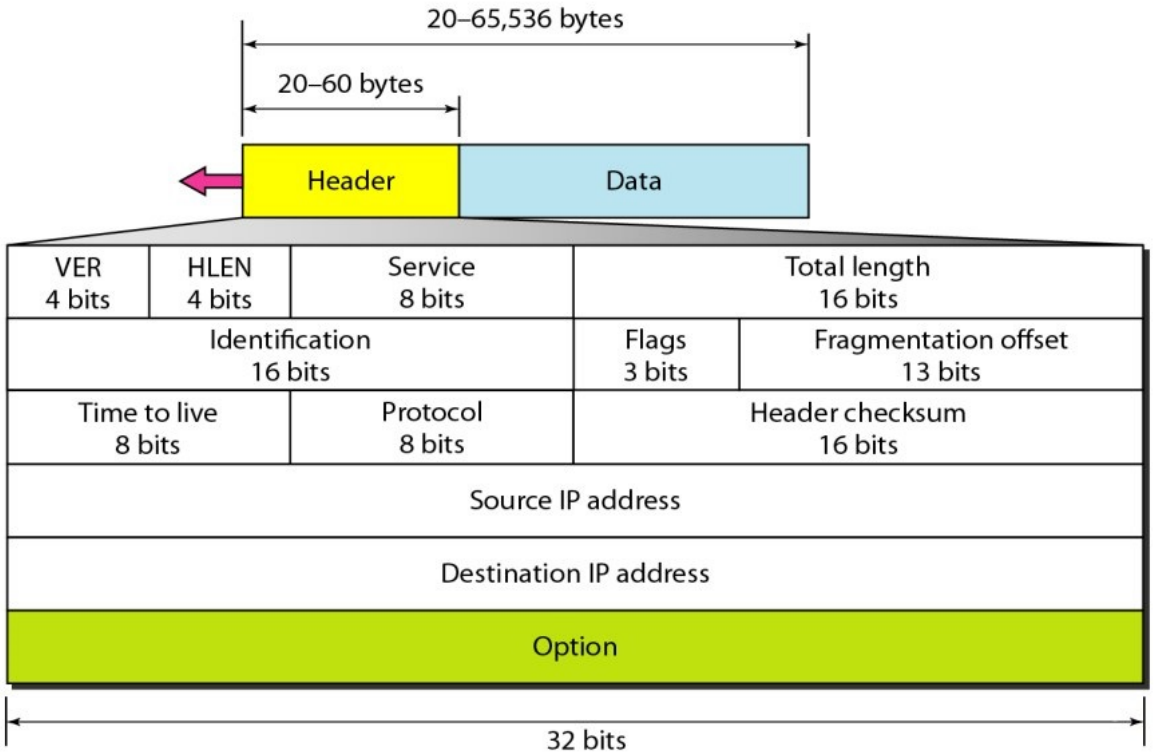


Figure 18 : IP datagram

IPv4 Addressing

- Version(VER): This field defines the version of the IP.

IPv4 Addressing

- Version(VER): This field defines the version of the IP.
- Header Length(HLEN): The length of the header is variable because of option field.
  This field defines the length of the datagram header in 4 byte words.

IPv4 Addressing

- Version(VER): This field defines the version of the IP.
- Header Length(HLEN): The length of the header is variable because of option field.
  This field defines the length of the datagram header in 4 byte words.
- Differentiated Services: This field defines the class of the datagram for Quality of Service purposes.

IPv4 Addressing

- Version(VER): This field defines the version of the IP.
- Header Length(HLEN): The length of the header is variable because of option field.
  This field defines the length of the datagram header in 4 byte words.
- Differentiated Services: This field defines the class of the datagram for Quality of Service purposes.
- Total Length: This field defines the total length of the IP datagram in bytes (header plus data).

$$Length\ of\ data = total\ length - header\ length$$

IPv4 Addressing

- Version(VER): This field defines the version of the IP.
- Header Length(HLEN): The length of the header is variable because of option field.
  This field defines the length of the datagram header in 4 byte words.
- Differentiated Services: This field defines the class of the datagram for Quality of Service purposes.
- Total Length: This field defines the total length of the IP datagram in bytes (header plus data).

$$Length\ of\ data = total\ length - header\ length$$

- Time to Live: A 8-bit field that is used to prevent packets from looping forever within an IP network.
  (i.e) it specifies the maximum number of routers a packet can pass through before being discarded

- Protocol: It contains a number indicating the type of data found in the payload portion of the datagram.
- The most common values are 17 (for UDP) and 6 (for TCP).
- This field provides a demultiplexing feature so that the IP protocol can be used to carry payloads of more than one protocol type.

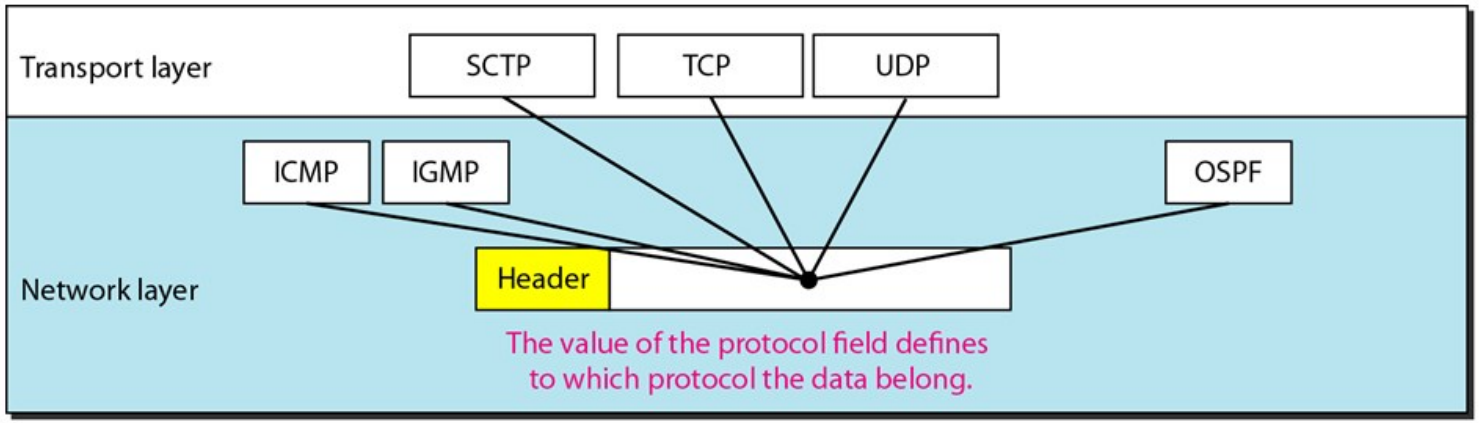


Figure 19 : Multiplexing

IPv4 Addressing

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

Figure 20 : Protocol values

IPv4 Addressing

- Source Address: This field defines the IP address of the source.
  This field remain unchanged during the time the IP datagram travels from the source host to the destination host.

IPv4 Addressing

- Source Address: This field defines the IP address of the source.
  This field remain unchanged during the time the IP datagram travels from the source host to the destination host.
- Destination Address: This field defines the IP address of the destination.
  This field remain unchanged during the time the IP datagram travels from the source host to the destination host.

IPv4 Addressing

- Source Address: This field defines the IP address of the source.
  This field remain unchanged during the time the IP datagram travels from the source host to the destination host.
- Destination Address: This field defines the IP address of the destination.
  This field remain unchanged during the time the IP datagram travels from the source host to the destination host.
- Options: Options are not required for every datagram.
  They are used for network testing and debugging.

IPv4 Addressing

- Fragmentation: A datagram can travel through different networks.
- Each router decapsulates the IP datagram from the frame it receives, processes it and encapsulates it in another frame.
  The format and size of the received frame depend on the protocol used by the physical network through which the frame has just traveled.
- The fields that are related to fragmentation and reassembly of an IP datagram are;
  (i) Identification
  (ii) Flags
  (iii) Fragmentation offset

IPv4 Addressing

- Identification: This field identifies a datagram originating from the source host. When a datagram is fragmented, the value in the identification field is copied into all fragments.
  The identification number helps the destination in reassembling the datagram. It knows that all fragments having the same identification value should be assembled into one datagram.
- Flags: This is a 3-bit field.
  (1) The first bit is reserved
  (2) The second bit is called the do not fragment bit.
  - If its valus is 1, the machine must not fragment the datagram.
  - If its valus is 0, the datagram can be fragmented if necessary.

  (3) The third bit is called the more fragment bit.
  - If its value is 1, it means the datagram is not the last fragment, (i.e) there are more fragments after this one.
  - If its value is 0, it means this is the last or only fragment.

- 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.

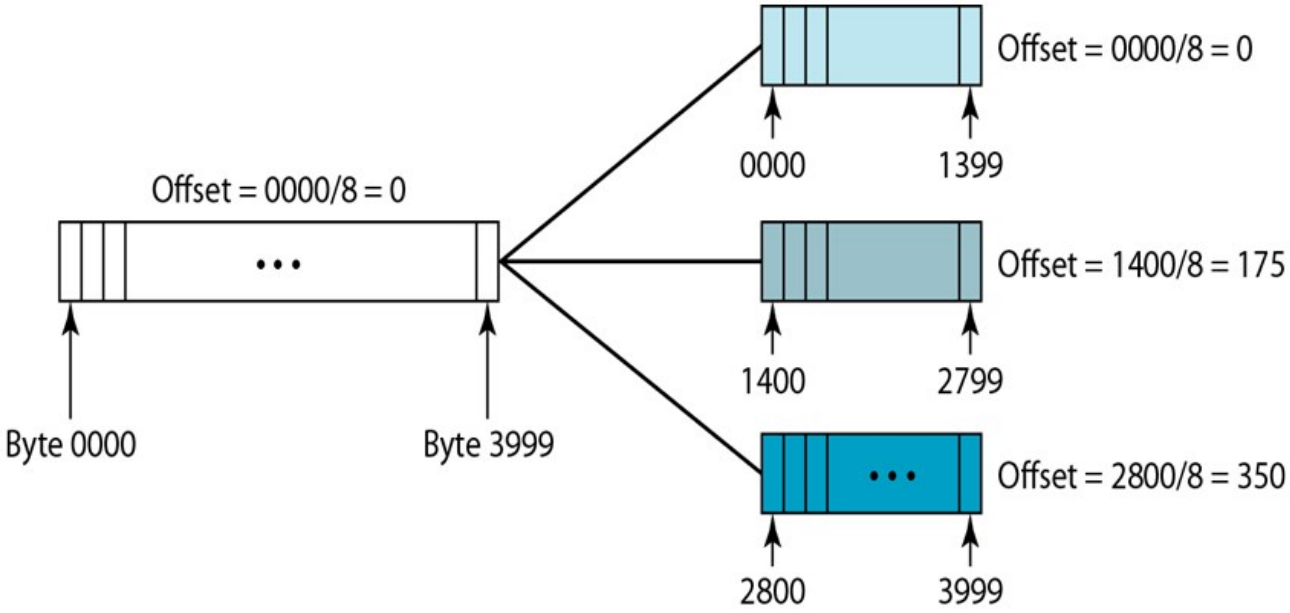
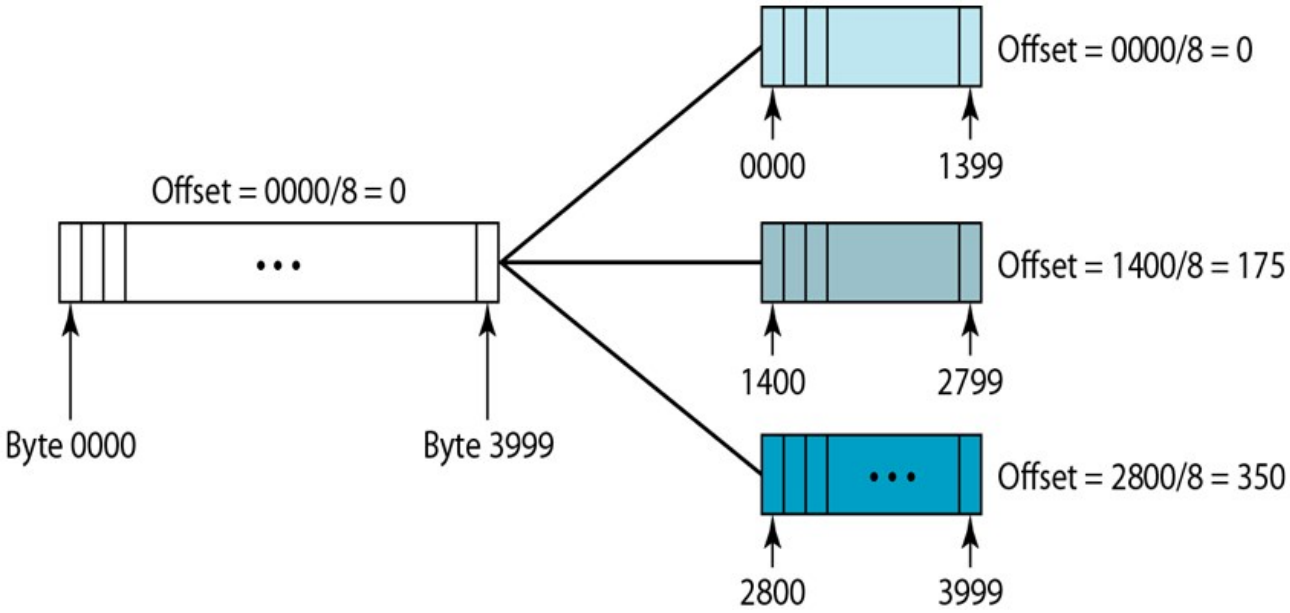
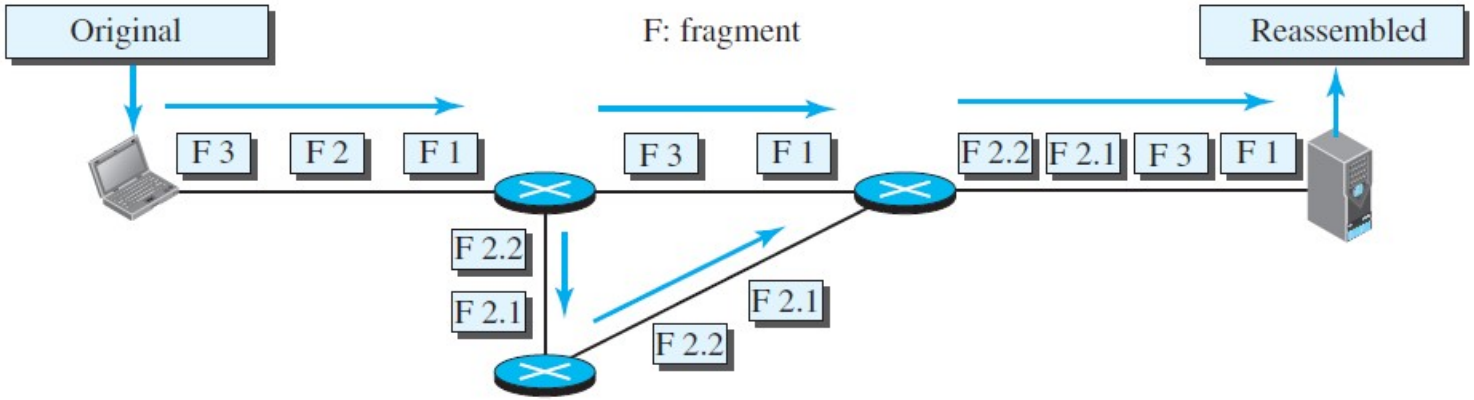


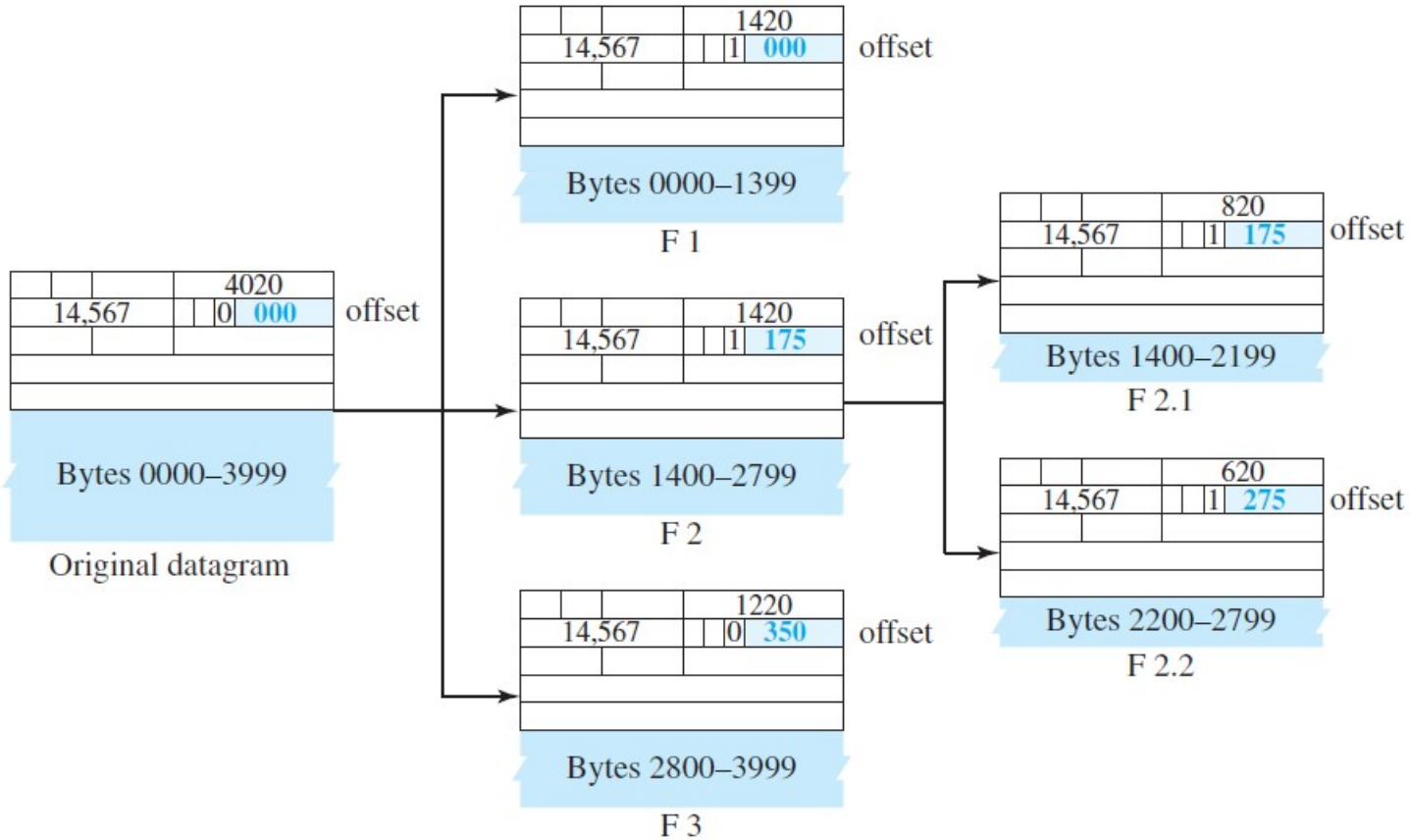
Figure 21 : Fragmentation

- Figure21 shows a datagram with a data size of 4000 bytes fragmented into three parts.

- The bytes in the original datagram are numbered from 0 to 3999.
- The first fragments carries bytes 0 to 1399. The offset for this datagram is 0/8=0.
  The second fragments carries bytes 1400 to 2799. The offset for this fragment is 1400/8=175.
  The third fragments carries bytes 2800 to 3999. The offset for this fragment is 2800/8=350.
- This forces hosts or routers that fragment datagrams to choose the size of each fragment so that the first byte number is divisible by 8.




Original
F: fragment
Reassembled
F 3
F 2
F 1
F 3
F 1
F 2.2
F 2.1
F 3
F 1
F 2.2
F 2.1
F 2.1
F 2.2


4020
14,567
0
000
offset
Bytes 0000–3999
Original datagram
1420
14,567
1
000
offset
Bytes 0000–1399
F 1
1420
14,567
1
175
offset
Bytes 1400–2799
F 2
1220
14,567
0
350
offset
Bytes 2800–3999
F 3
820
14,567
1
175
offset
Bytes 1400–2199
F 2.1
620
14,567
1
275
offset
Bytes 2200–2799
F 2.2

- Checksum: The checksum in the IP packet covers only the header, not the data.
- There are two reasons for this
  (1) All higher level protocols that encapsulate data in the IP datagram have a checksum field that covers the whole packet. Therefore, the checksum for the IP datagram does not have to check the encapsulated data.
  (2) The header of the IP packet changes with each visited router, but the data do not. So, the checksum includes only the part that has changed.
  If the data are included, each router must recalculate the checksum for the whole packet, which means increased processing time for each router.

- Figure23 shows an example of a checksum calculation for an IP header without options.
- The header is divided into 16-bit sections. The valus of the checksum field is set to zero. All the sections are added and the sum is complemented. The result is inserted in the checksum field.

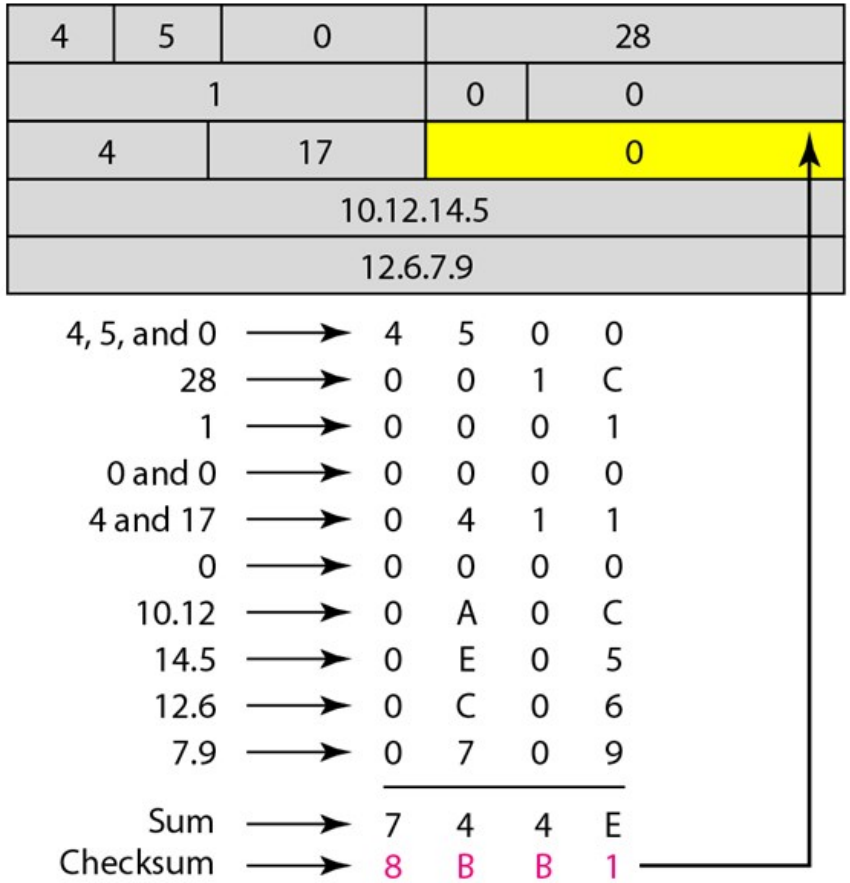


Figure 23 : Checksum Calculation

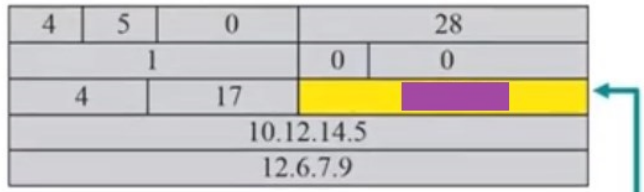
Exercise1

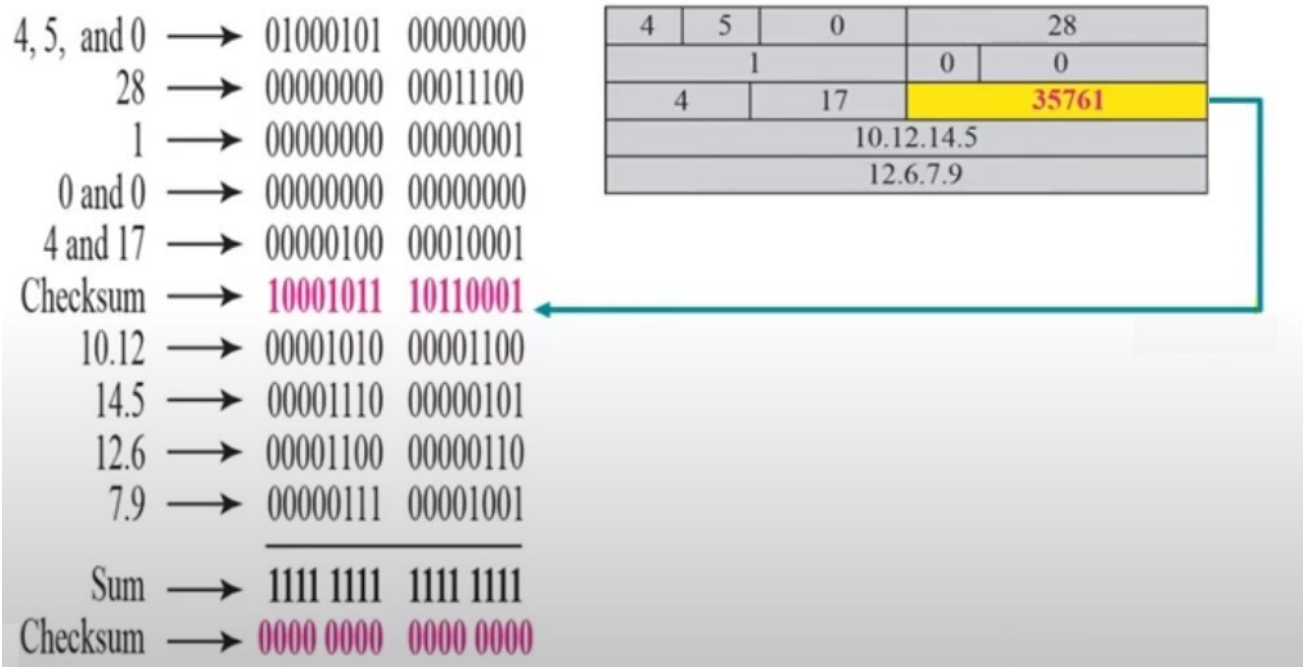
- Calculate the checksum and verify how the datagram is accepted or discarded in IPv4. Assume the values in the datagram field as in Figure23.

Exercise1

- Calculate the checksum and verify how the datagram is accepted or discarded in IPv4. Assume the values in the datagram field as in Figure23.

4, 5, and 0	→	01000101 00000000
28	→	00000000 00011100
1	→	00000000 00000001
0 and 0	→	00000000 00000000
4 and 17	→	00000100 00010001
0	→	00000000 00000000
10.12	→	00001010 00001100
14.5	→	00001110 00000101
12.6	→	00001100 00000110
7.9	→	00000111 00001001
Sum	→	**01110100 01001110**
Checksum	→	**10001011 10110001**




4, 5, and 0 → 01000101 00000000
28 → 00000000 00011100
1 → 00000000 00000001
0 and 0 → 00000000 00000000
4 and 17 → 00000100 00010001
Checksum → 10001011 10110001
10.12 → 00001010 00001100
14.5 → 00001110 00000101
12.6 → 00001100 00000110
7.9 → 00000111 00001001
Sum → 1111 1111 1111 1111
Checksum → 0000 0000 0000 0000
4 5 0 28
1 0 0
4 17 35761
10.12.14.5
12.6.7.9

IPv6 Addressing

IPv6 Addressing

- IPv4 has some deficiencies that make it unsuitable for the fast growing internet, that includes;
  (1) IPv4 has a two level address structure (netid and hostid) categorized into five classess(A, B, C, D and E). The use of address space is inefficient.

IPv6 Addressing

- IPv4 has some deficiencies that make it unsuitable for the fast growing internet, that includes;
  (1) IPv4 has a two level address structure (netid and hostid) categorized into five classess(A, B, C, D and E). The use of address space is inefficient.
  (2) The internet must accomodate real time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.

IPv6 Addressing

- IPv4 has some deficiencies that make it unsuitable for the fast growing internet, that includes;
  (1) IPv4 has a two level address structure (netid and hostid) categorized into five classess(A, B, C, D and E). The use of address space is inefficient.
  (2) The internet must accomodate real time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
  (3) The internet must accomodate encryption and authentication of data for some applications. No security mechanism was provided by IPv4.

IPv6 Addressing

- IPv4 has some deficiencies that make it unsuitable for the fast growing internet, that includes;
  (1) IPv4 has a two level address structure (netid and hostid) categorized into five classess(A, B, C, D and E). The use of address space is inefficient.
  (2) The internet must accomodate real time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
  (3) The internet must accomodate encryption and authentication of data for some applications. No security mechanism was provided by IPv4.
- To overcome these deficiencies, Internet Protocol version6 (IPv6), also called Internetworking Protocol, next generation(IPng) was proposed.
  The format and length of the IP addresses were changed along with the packet format.

IPv6 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.

IPv6 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.
  - Better Header format: IPv6 uses new header format in which options are seperated from the base header and inserted when needed, between the base header and upper layer data.
    This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.

IPv6 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.
  - Better Header format: IPv6 uses new header format in which options are seperated from the base header and inserted when needed, between the base header and upper layer data.
    This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
  - Allowance for extension: IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

IPv6 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.
  - Better Header format: IPv6 uses new header format in which options are seperated from the base header and inserted when needed, between the base header and upper layer data.
    This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
  - 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 flow label is 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.

IPv6 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.
  - Better Header format: IPv6 uses new header format in which options are seperated from the base header and inserted when needed, between the base header and upper layer data.
    This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
  - 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 flow label is 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 Addressing

- IPv6 has some of the advantages over IPv4 that can be summarized as follows;
  - Larger address space: An IPv6 address is 128 bit long.
    Huge increase in address space when compare with 32-bit address of IPv4.
  - Better Header format: IPv6 uses new header format in which options are seperated from the base header and inserted when needed, between the base header and upper layer data.
    This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.
  - 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 flow label is 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.
  - New Options: IPv6 has new options to allow for additional functionalities.

IPv6 Addressing

- An IPv6 address consists of 16 bytes (octets); 128 bit long
- In hexadecimal notation, 128 bits are divided into 8 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 4 digits seperated by a colon.

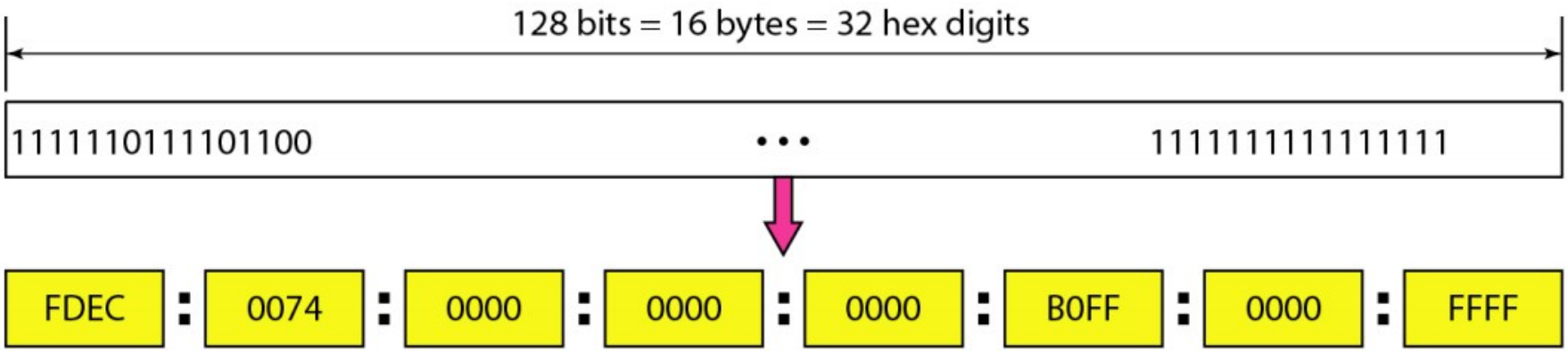
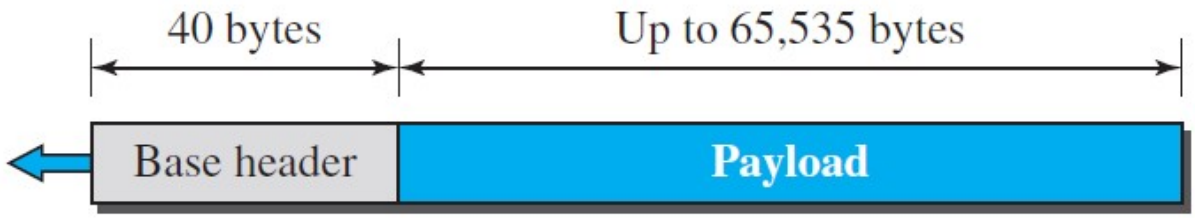
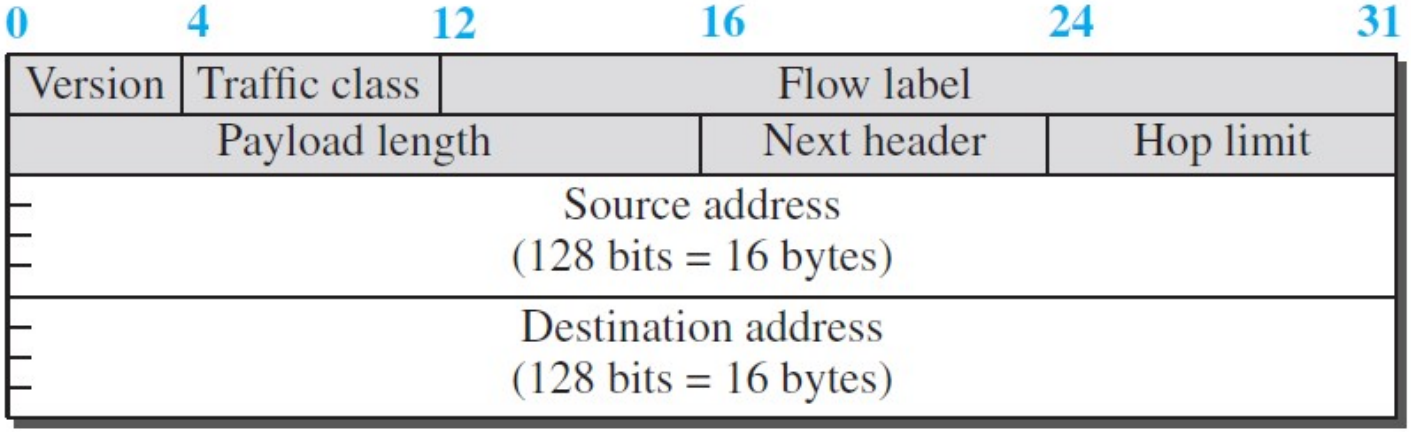


Figure 24 : IPv6 address in binary and hexadecimal colon notation

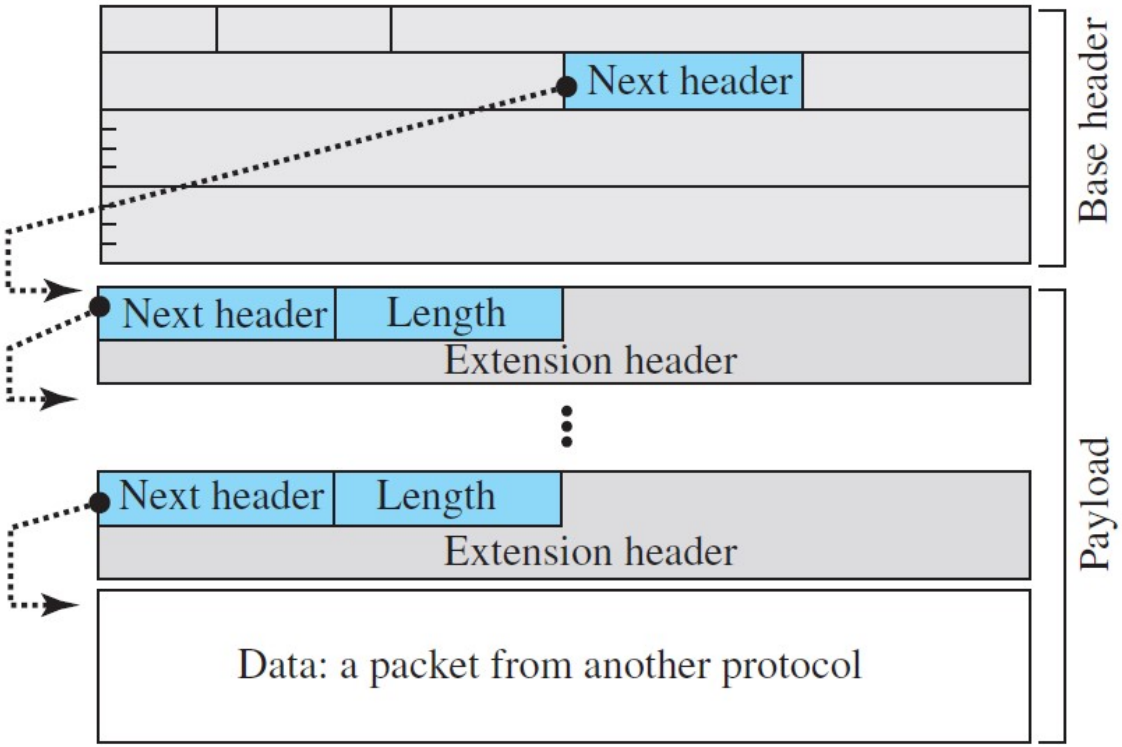


a. IPv6 packet



b. Base header

Figure 25 : IPv6 datagram



Some next-header codes

00: Hop-by-hop option
02: ICMPv6
06: TCP
17: UDP
43: Source-routing option
44: Fragmentation option
50: Encrypted security payload
51:Authentication header
59: Null (no next header)
60: Destination option

Figure 26 : IPv6 datagram format

IPv6 Addressing

- IPv6 packet is shown in Figure26
- Each packet is composed of a mandatory base header followed by the payload. The payload consists of two parts;
  (1) optional extension headers
  (2) data from upper layer
- The base header occupies 40 bytes, whereas the extension headers and data from the upper layer contain up to 65535 bytes of information.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.
  - Payload length: This 2 byte payload length field defines the total length of the IP datagram excluding the base header.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.
  - Payload length: This 2 byte payload length field defines the total length of the IP datagram excluding the base header.
  - Next header: This 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 for an upper layer protocol such as UDP or TCP. Each extension header also contains in this field.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.
  - Payload length: This 2 byte payload length field defines the total length of the IP datagram excluding the base header.
  - Next header: This 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 for an upper layer protocol such as UDP or TCP. Each extension header also contains in this field.
  - Hop limit: This 8-bit hop limit field serves the same purpose as the TTL(Time to Live) field in IPv4.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.
  - Payload length: This 2 byte payload length field defines the total length of the IP datagram excluding the base header.
  - Next header: This 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 for an upper layer protocol such as UDP or TCP. Each extension header also contains in this field.
  - Hop limit: This 8-bit hop limit field serves the same purpose as the TTL(Time to Live) field in IPv4.
  - Source address: It is a 16 byte(128-bit) internet address that identifies the original source of the datagram.

- The base header with eight fields are;
  - Version: This 4 bit field defines the version number of the IP.
  - Priority: This 4 bit priority field defines the priority of the packet with respect to packet congestion.
  - Flow label: This 3 byte(24-bit) field that is designed to provide special handling for a particular flow of data.
  - Payload length: This 2 byte payload length field defines the total length of the IP datagram excluding the base header.
  - Next header: This 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 for an upper layer protocol such as UDP or TCP. Each extension header also contains in this field.
  - Hop limit: This 8-bit hop limit field serves the same purpose as the TTL(Time to Live) field in IPv4.
  - Source address: It is a 16 byte(128-bit) internet address that identifies the original source of the datagram.
  - Destination address: This field is a 16 byte(128-bit) internet address that identifies the final destination of the datagram.
    If the source routing is used, this field contains the address of the next router.

IPv6 Addressing

- An IPv6 packet is made of a base header and some extension headers.
- The length of the base header is fixed at 40 bytes.
  However, to give more functionality to the IP datagram, the base header can be followed by up to six extension headers.

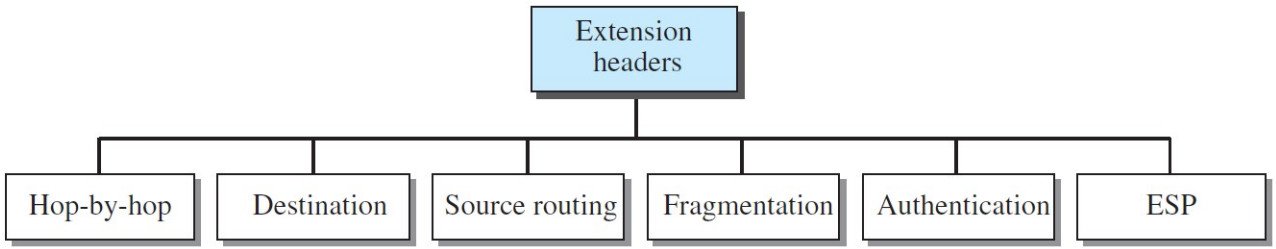


Figure 27 : Extension header types:

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.
  - Destination Option: The destination option is used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.
  - Destination Option: The destination option is used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.
  - Source Routing: The source routing extension header combines the concepts of the strict source route and the loose source route options of IPv4.

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.
  - Destination Option: The destination option is used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.
  - Source Routing: The source routing extension header combines the concepts of the strict source route and the loose source route options of IPv4.
  - Fragmentation: The concept of fragmentation in IPv6 is the same as that in IPv4. However, the place where fragmentation occurs differs.
    In IPv4, the source or a router is required to fragment if the size of the datagram is larger than the MTU of the network over which the datagram travels.
    In IPv6, only the original source can fragment. A source must use a Path MTU Discovery technique to find the smallest MTU supported by any network on the path.

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.
  - Destination Option: The destination option is used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.
  - Source Routing: The source routing extension header combines the concepts of the strict source route and the loose source route options of IPv4.
  - Fragmentation: The concept of fragmentation in IPv6 is the same as that in IPv4. However, the place where fragmentation occurs differs.
    In IPv4, the source or a router is required to fragment if the size of the datagram is larger than the MTU of the network over which the datagram travels.
    In IPv6, only the original source can fragment. A source must use a Path MTU Discovery technique to find the smallest MTU supported by any network on the path.
  - Authentication: The authentication extension header has a dual purpose: it validates the message sender and ensures the integrity of data.

- The extension headers are;
  - Hop-by-Hop Option: The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram.
  - Destination Option: The destination option is used when the source needs to pass information to the destination only. Intermediate routers are not permitted access to this information.
  - Source Routing: The source routing extension header combines the concepts of the strict source route and the loose source route options of IPv4.
  - Fragmentation: The concept of fragmentation in IPv6 is the same as that in IPv4. However, the place where fragmentation occurs differs.
    In IPv4, the source or a router is required to fragment if the size of the datagram is larger than the MTU of the network over which the datagram travels.
    In IPv6, only the original source can fragment. A source must use a Path MTU Discovery technique to find the smallest MTU supported by any network on the path.
  - Authentication: The authentication extension header has a dual purpose: it validates the message sender and ensures the integrity of data.
  - Encrypted Security Payload: The encrypted security payload (ESP) is an extension that provides confidentiality and guards against eavesdropping.

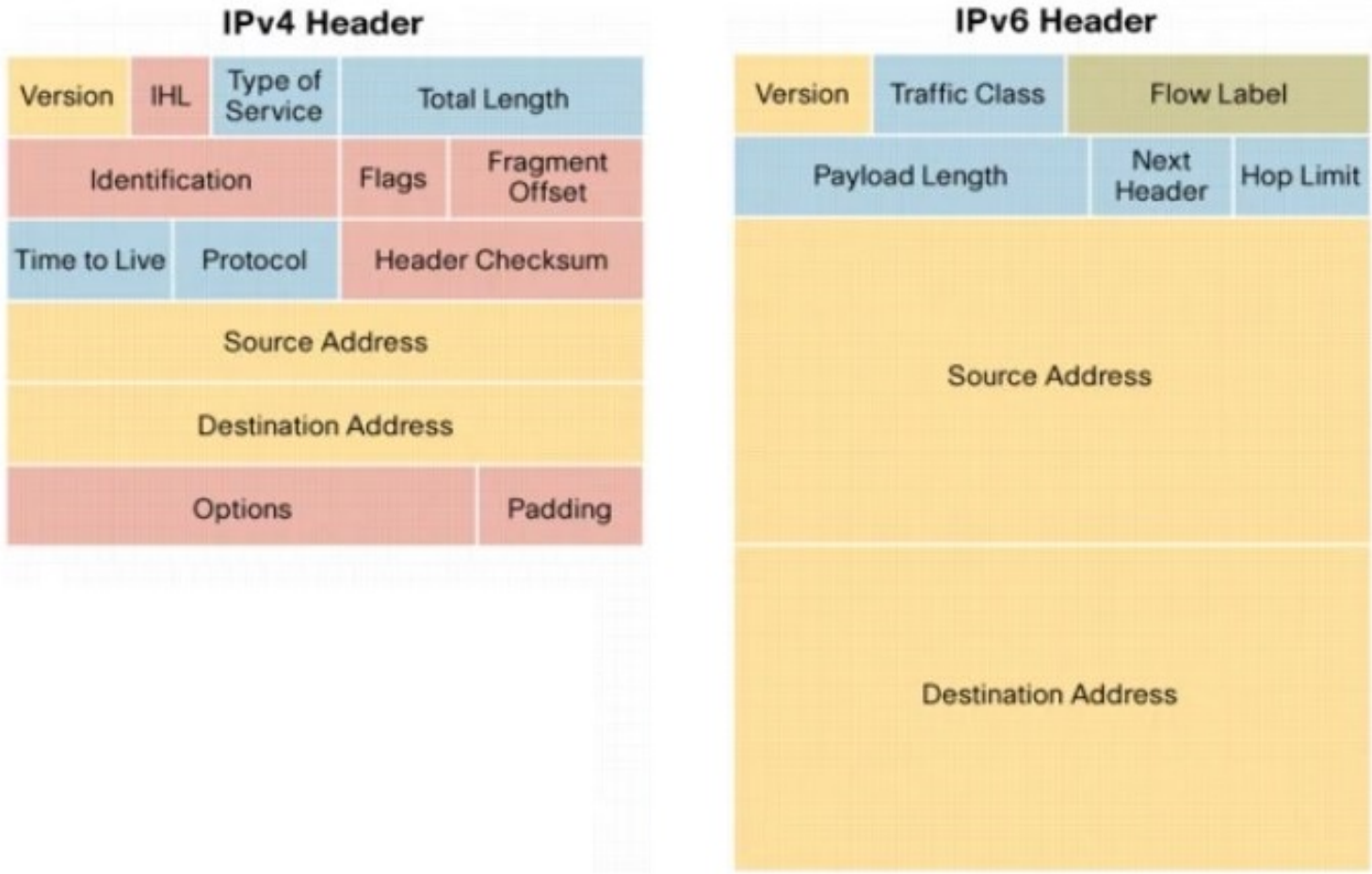


Figure 28 : Comparison of IPv4 and IPv6

Thank you

&

Queries