Chapter 18: Introduction to Network Layer

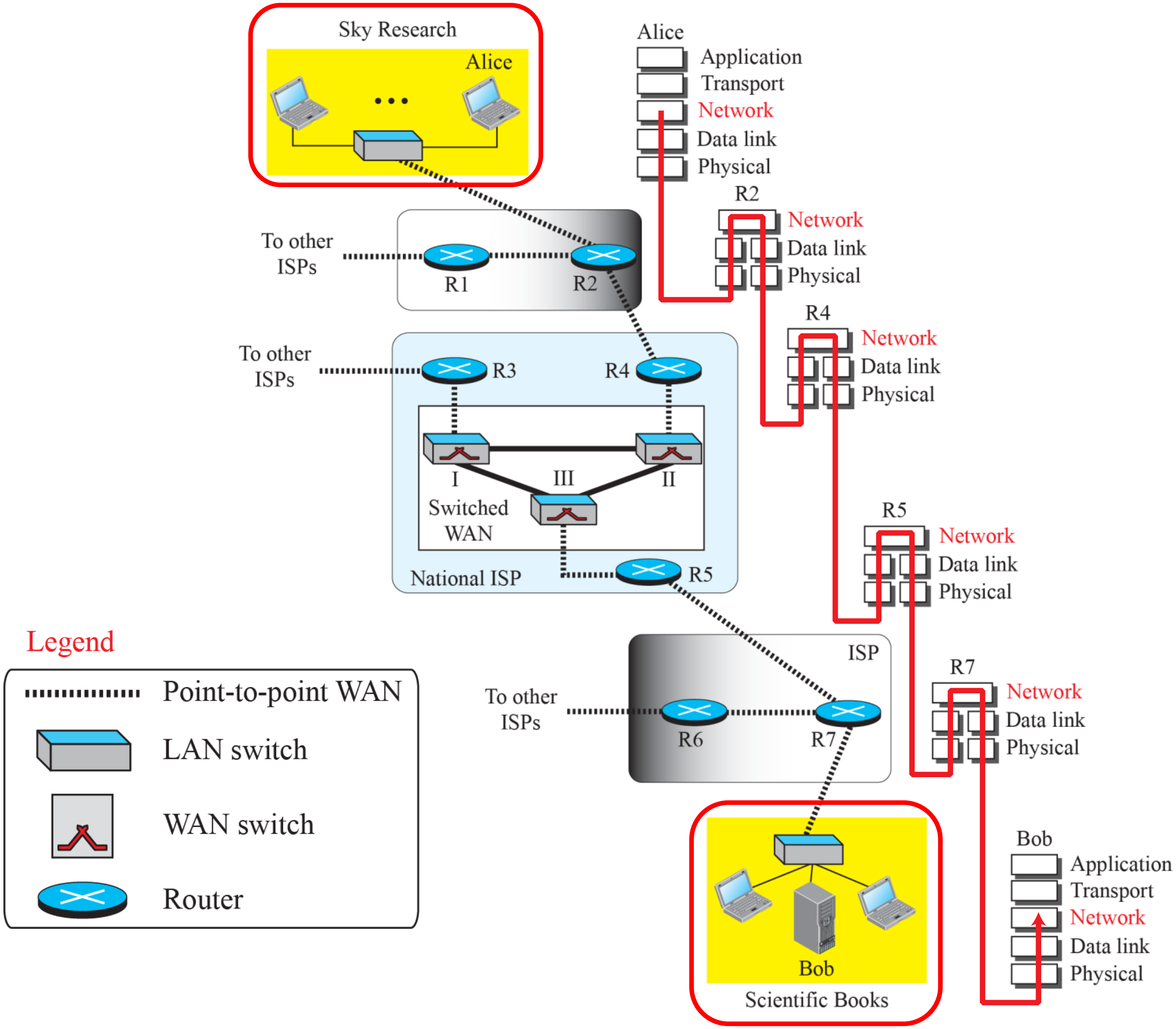
Outline

18.1 NETWORK-LAYER SERVICES

18.2 PACKET SWITCHING

18.4 IPv4 ADDRESSES

Figure 18.1: Communication at the network layer



18.1 Network-Layer Services

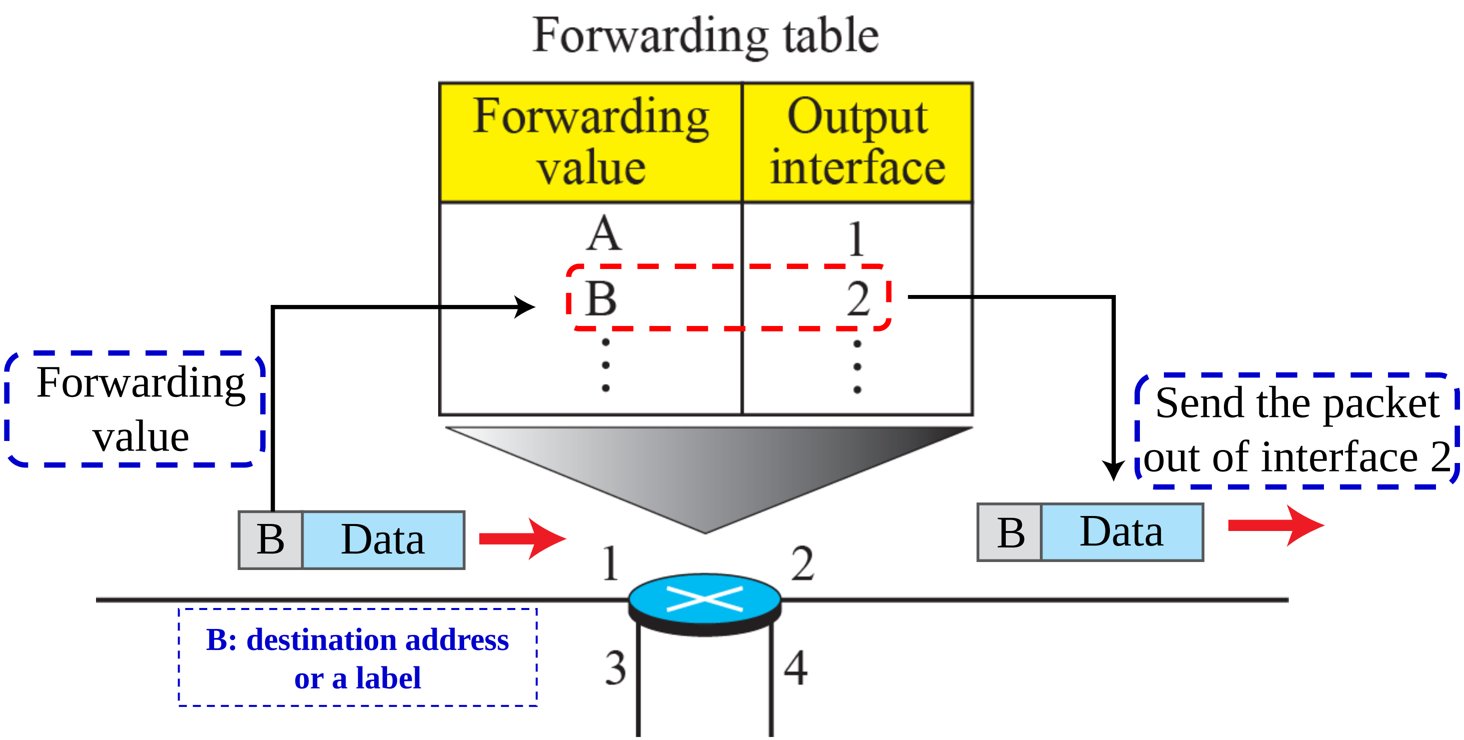
Before discussing the network layer in the Internet today, let's briefly discuss the network-layer services (packetizing, routing, forwarding) that, in general, are expected from a network-layer protocol. In addition, other services (error control, flow control, congestion control, quality of service and security) may also be expected.

Packetizing: encapsulating the payload (data received from upper layer) in a network-layer packet at the source and decapsulating the payload from the network-layer packet at the destination. Note that the network layer carries a payload from the source to the destination without changing or using it.

Routing: there is more than one route from the source to the destination. The network layer is responsible for applying strategies and running routing protocols to find the best one among these possible routes and create routing tables for each router.

18.1 Network-Layer Services

***Forwarding**: is the __action applied__ by each router when __a packet arrives__ at one of its __interfaces__, i.e., to __forward the packet__ to __another__ (unicast) or __some__ (multicast) attached network(s).*

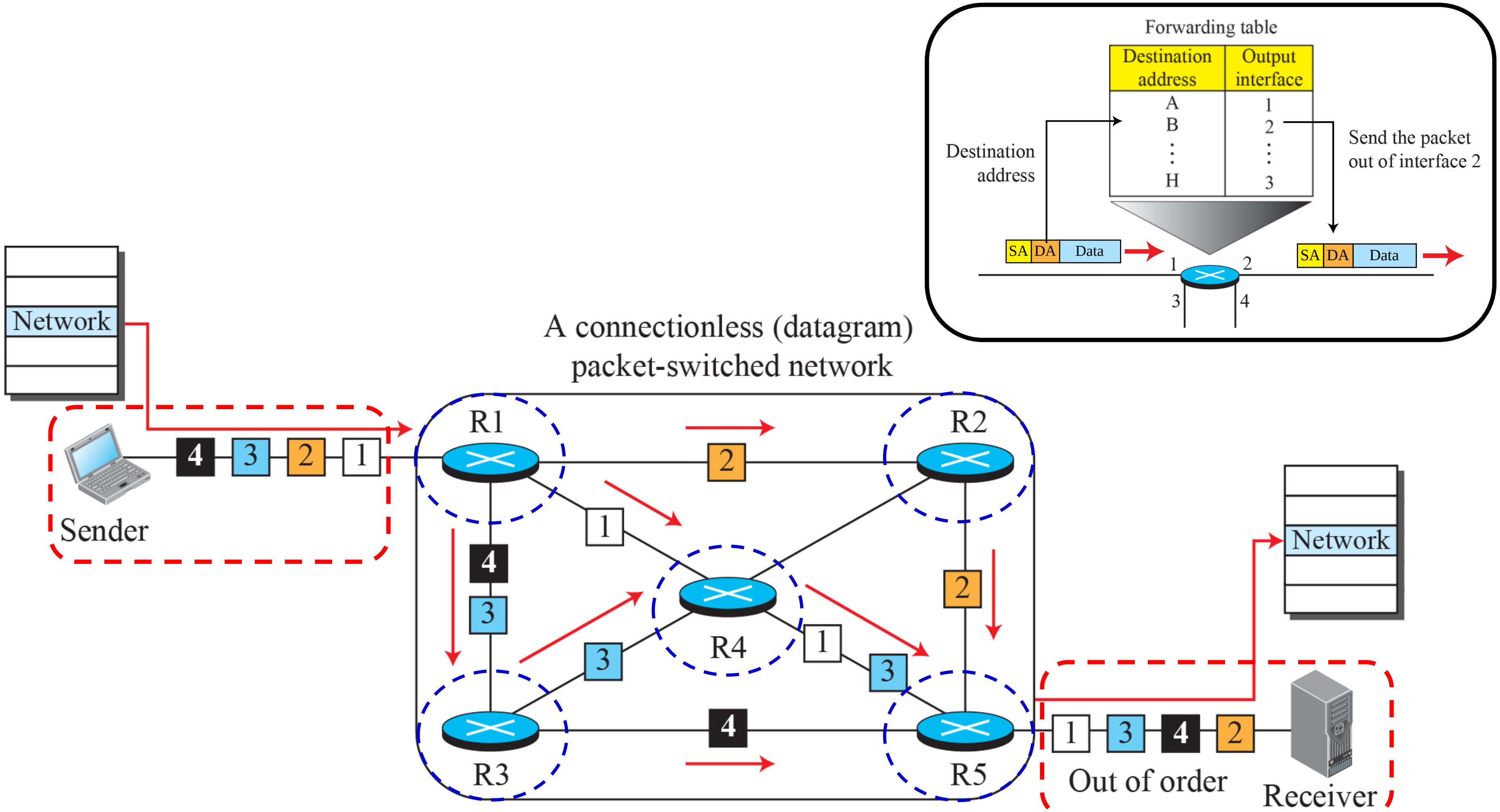


18-2 PACKET SWITCHING

From the discussion of routing and forwarding, we infer that a kind of switching occurs at the network layer.

A <u>router</u>, in fact, is a <u>switch</u> that <u>creates a connection</u> between an <u>input port</u> and an <u>output port</u> (or a set of output ports), just as an electrical switch connects the input to the output to let electricity flow.

Figure 18.3: A connectionless packet-switched network



18-4 IPv4 ADDRESSES

The identifier used in the network layer of the TCP/IP protocol suite to identify the connection of each device to the Internet is called the Internet address or IP address. An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet.

The IP address is the address of the connection, not the host or the router.

18.4.1 Address Space

A protocol like IPv4 that defines addresses has an address space. An address space is the total number of addresses used by the protocol. IPv4 uses 32-bit addresses, which means that the address space is 2^{32}. If there were no restrictions, more than 4 billion devices could be connected to the Internet. A 32-bit IPv4 address can be notated using binary, dotted decimal and hexadecimal.

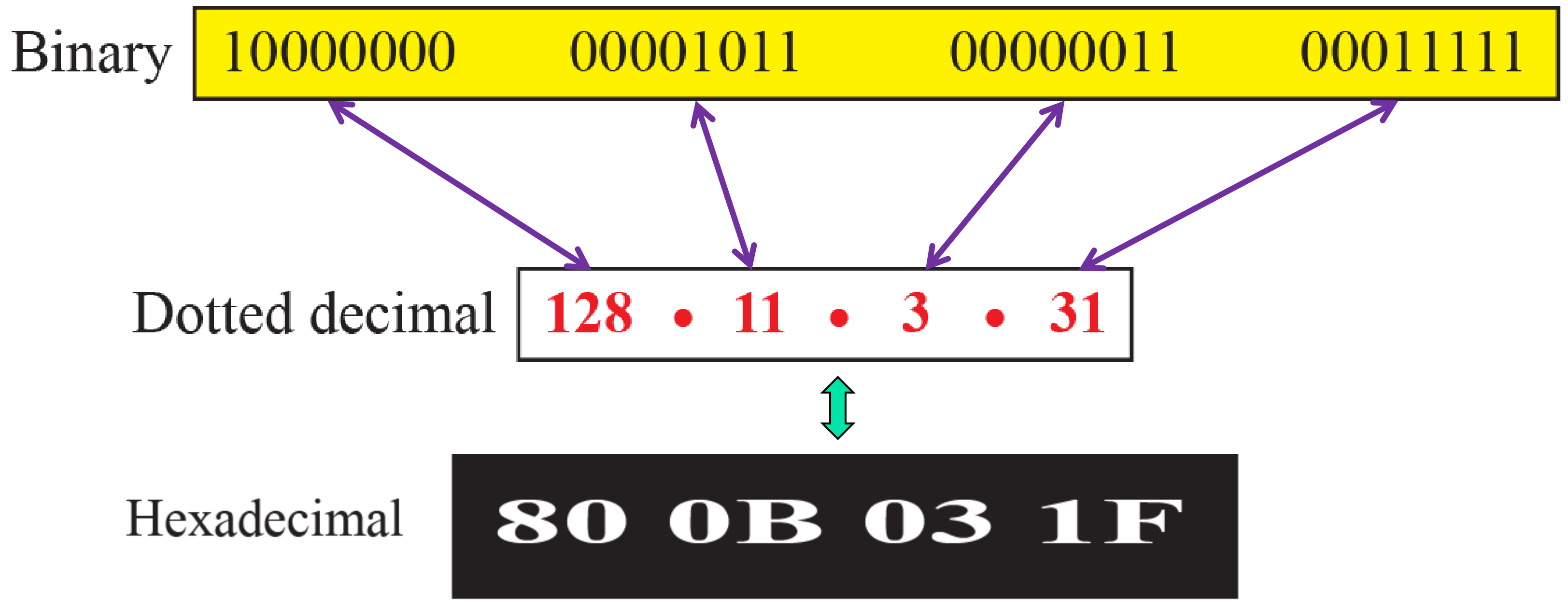
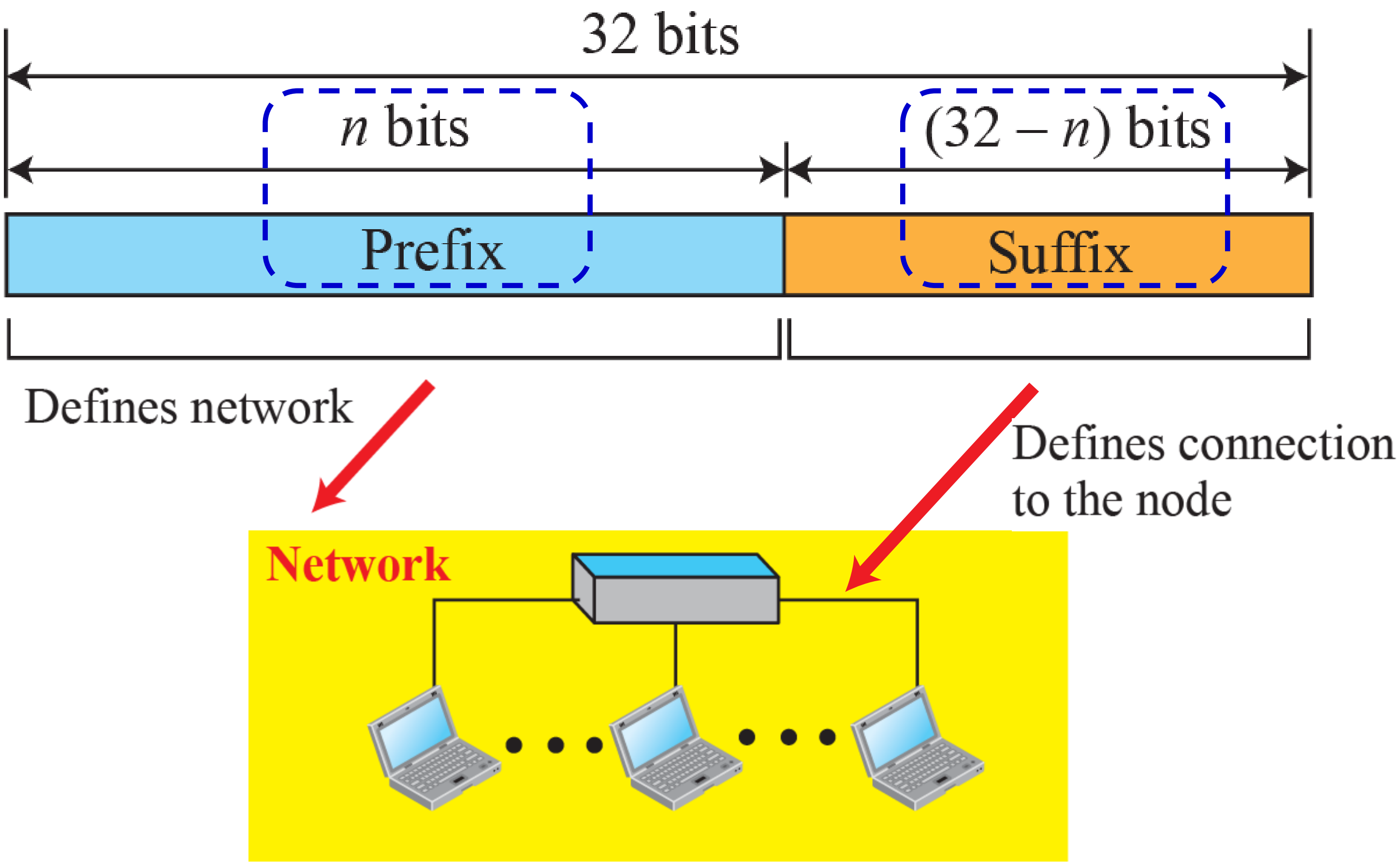


Figure 18.17: Hierarchy in addressing

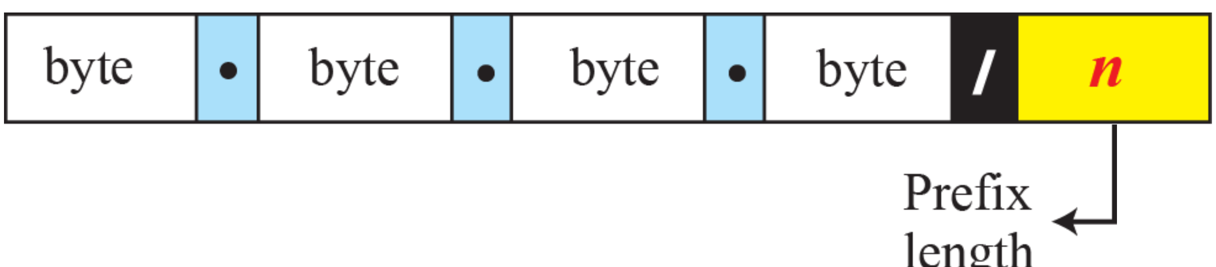
A 32-bit IPv4 address is <u>hierarchical</u> and <u>divided into two parts</u>: the first part of the address is called the <u>prefix</u> (fixed- or variable- length) and <u>defines the network</u>; the second part of the address is called the <u>suffix</u> and <u>defines the connection to the node</u>.



18.4.3 Classless Addressing

With the growth of the Internet, it was clear that a larger address space was needed as a long-term solution. Although the long-term solution has already been devised and is called IPv6 (128-bit addresses with $2^{128} = 340 \times 10^{36}$), a short-term solution was also devised to use the same address space but to change the distribution of addresses (as well as subnetting and supernetting) to provide a fair share to each organization.

The short-term solution still uses IPv4 addresses and is referred to as classless addressing. Note that since the prefix length is not inherent in the address, it is added to the address separated by a slash. The notation is formally know as classless interdomain routing or CIDR.



Examples:
12.24.76.8/**8**
23.14.67.92/**12**
220.8.24.255/**25**

Example

A classless address is given as 167.199.170.82/**27**.

a) How <u>many addresses</u> are there in the network?

b) What is the <u>first address</u> and what is the <u>last address</u>?

Solution:

a) The <u>number of addresses</u> in the network is $2^{32-n} = 2^5 = 32$ addresses.

b) The <u>first address</u> can be found by keeping the first 27 bits and changing the rest of the bits to 0s.

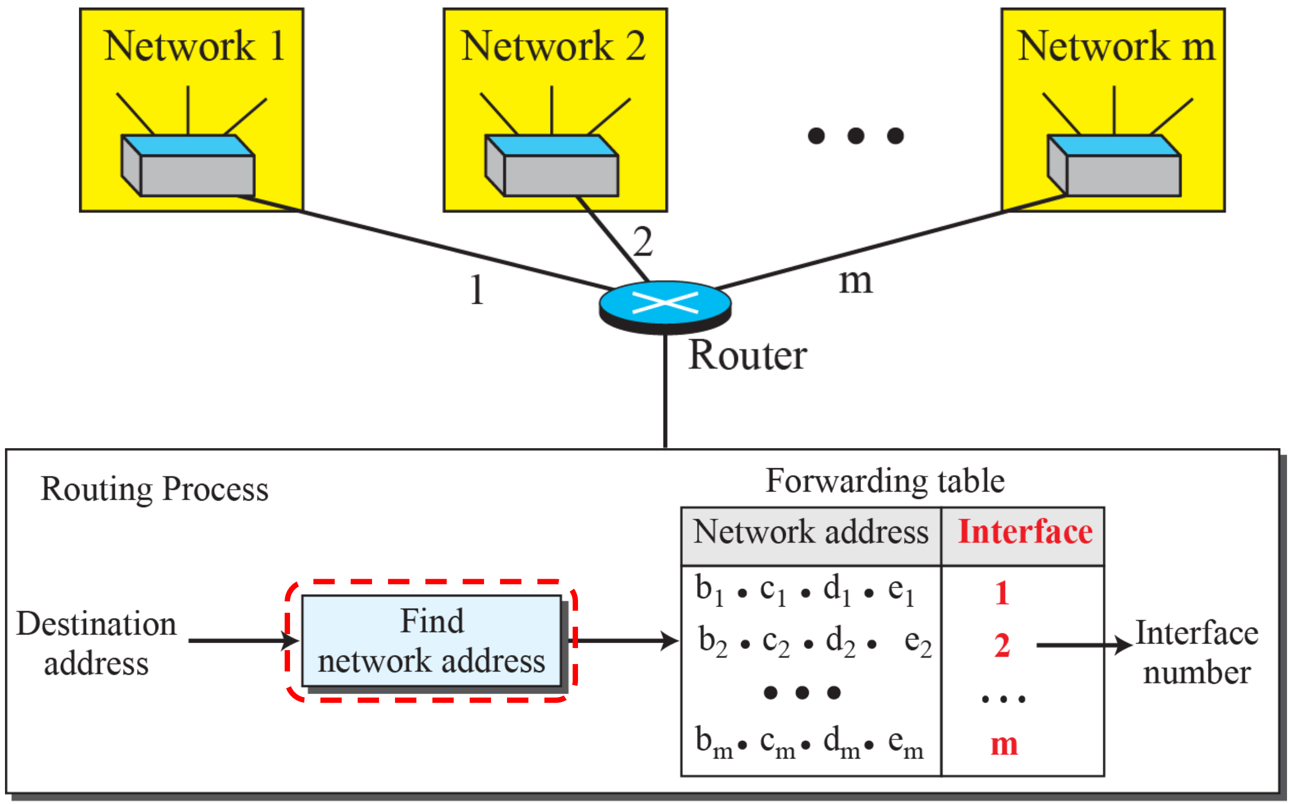
Address: 167.199.170.82/**27**	10100111	11000111	10101010	01010010
First address: 167.199.170.64/**27**	10100111	11000111	10101010	01000000

The <u>last address</u> can be found by keeping the first 27 bits and changing the rest of the bits to 1s.

Last address: 167.199.170.95/**27**	10100111	11000111	10101010	01011111

Figure 18.22: ***Network address***

When a packet arrives at the router from any source host, the router needs to know which interface (i.e., to which network) the packet should be sent out.



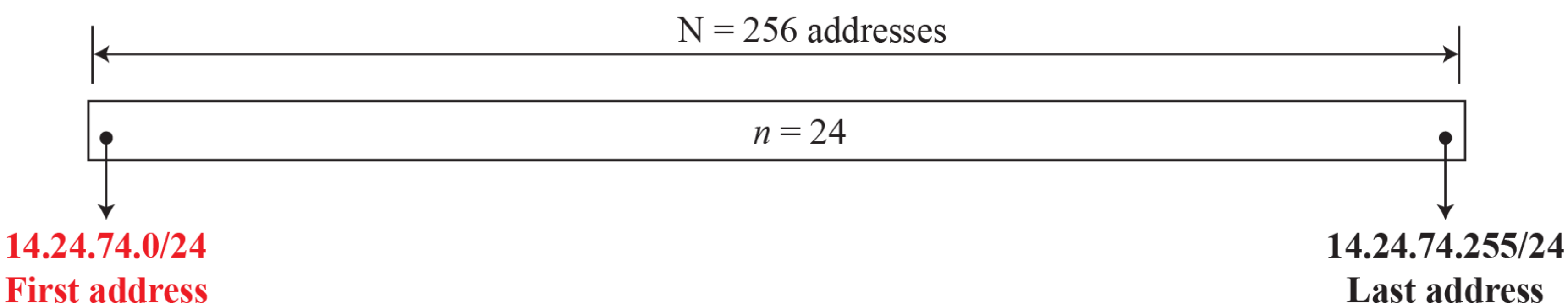
Additional Information: Chapter 18.5 (Forwarding of IP Packets, Longest Mask Matching):
To determine the network address (and hence the corresponding interface to send the packet out), 1) mask the prefix length with the destination address using the logical AND operation and 2) select the interface with the longest mask match. (The prefix length (network part of the address), is indicated by the number of msb 1s in the mask: e.g., "/16" denotes 11111111 11111111 00000000 00000000.)

Problem

An organization is granted a block of addresses with the beginning address 14.24.74.0/**24**. The organization needs to have 3 subblocks of addresses to use in its three **subnets**: one subblock of 10 addresses, one subblock of 60 addresses and one subblock of 120 addresses. Design the subblocks by assigning addresses to subblocks, starting with the largest and ending with the smallest one.

Solution

There are $2^{32-24} = 256$ addresses in this block. The first address is 14.24.74.0/**24**; the last address is 14.24.74.255/**24**.

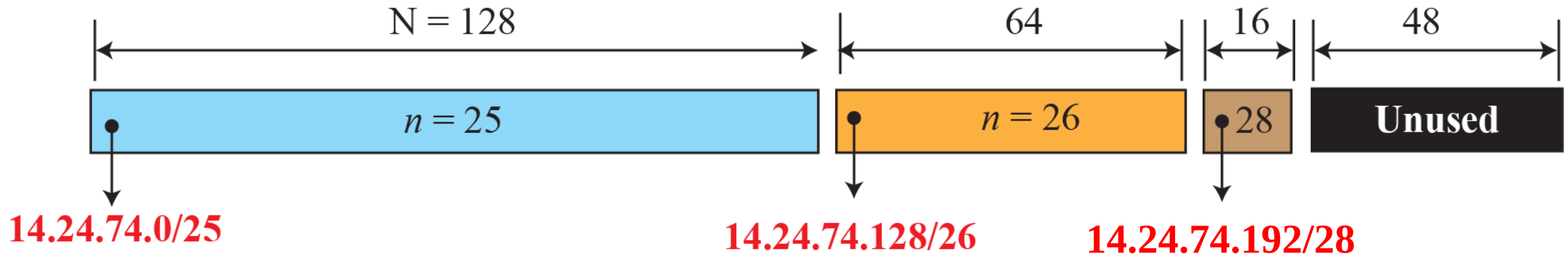


Problem (cont'd)

a. The number of addresses in the largest subblock, which requires 120 addresses, is not a power of 2. We allocate <u>128 addresses</u>. The subnet mask for this subnet can be found as $n_1 = 32 - \log_2 128 = 25$. The first address in this block is 14.24.74.0/**25**; the last address is 14.24.74.127/**25**.

b. The number of addresses in the second largest subblock, which requires 60 addresses, is not a power of 2 either. We allocate <u>64 addresses</u>. The subnet mask for this subnet can be found as $n_2 = 32 - \log_2 64 = 26$. The first address in this block is 14.24.74.128/**26**; the last address is 14.24.74.191/**26**.

c. The number of addresses in the smallest subblock, which requires 10 addresses, is not a power of 2 either. We allocate <u>16 addresses</u>. The subnet mask for this subnet can be found as $n_1 = 32 - \log_2 16 = 28$. The first address in this block is 14.24.74.192/**28**; the last address is 14.24.74.207/**28**.



Chapter 19: Network Layer Protocols

Outline

19.1 IPv4

19.1 NETWORK-LAYER PROTOCOLS

The main protocol in the network layer, <u>Internet Protocol version 4</u> (IPv4), is responsible for <u>packetizing</u>, <u>forwarding</u>, and <u>delivery</u> of a packet. It is an <u>unreliable</u> and a <u>connectionless</u> datagram protocol.

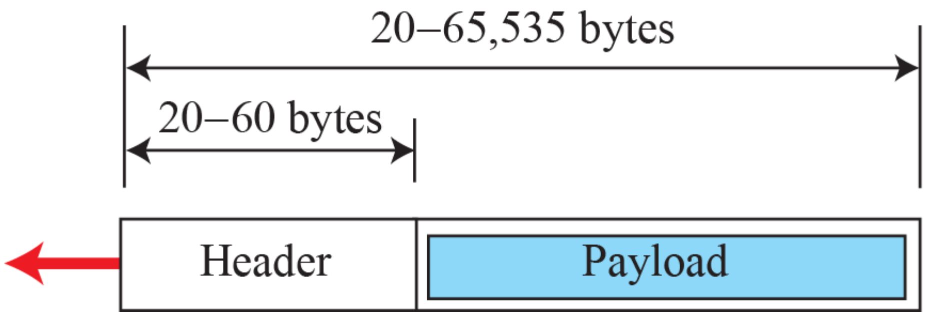
The <u>Internet Contol Message Protocol version 4</u> (ICMPv4), a network layer protocol, is a companion to IPv4 and helps IPv4 to handle some <u>errors</u> that may occur in delivery.

19.1.1 Datagram Format

Packets used by the IP are called <u>datagrams</u>. A datagram is a <u>variable-length</u> packet consisting of two parts: <u>header</u> and <u>payload</u> (data). The <u>header</u> is a <u>minimum of 20 bytes</u> and <u>up to 60 bytes</u> in length and contains information essential to <u>routing</u> and <u>delivery</u>.

It is customary in TCP/IP to show the IP header in <u>32-bit</u> sections.

Figure 19.2: *IP datagram*



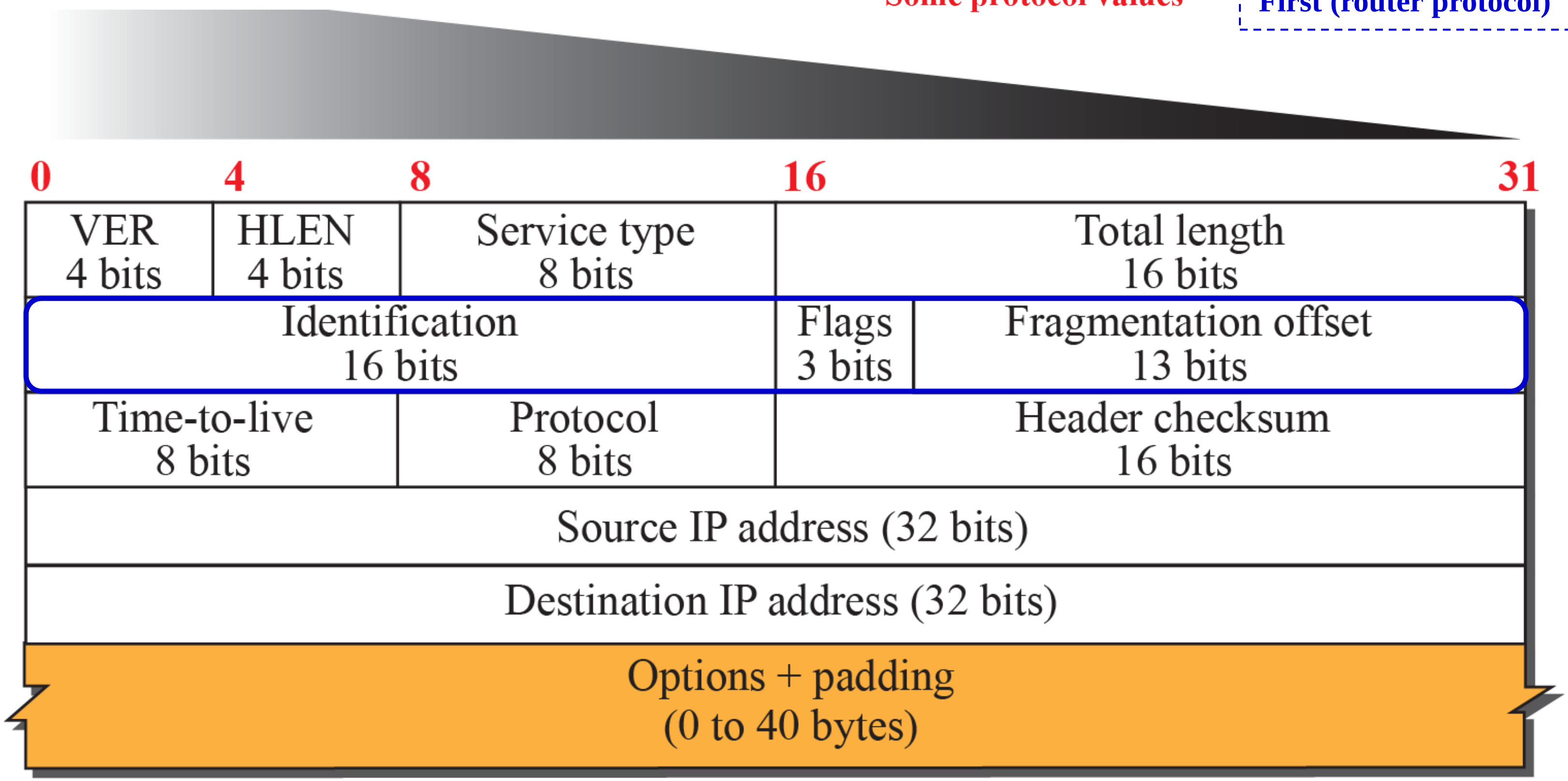
Legend

VER: version number
HLEN: header length
byte: 8 bits

ICMP: 01
IGMP: 02
TCP: 06
UDP: 17
OSPF: 89

Some protocol values

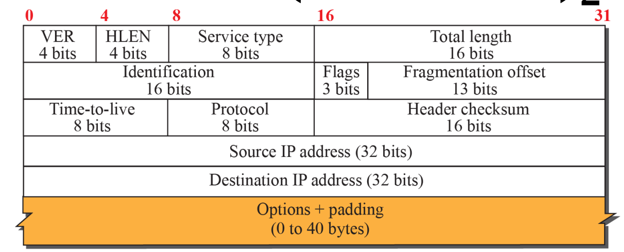
IGMP: Internet Group Management Protocol (IP multicast)
OSPF: Open Shortest Path First (router protocol)



Notes: 1)* HLEN *is in units of* <u>*4-bytes*</u>*. 2)* Total length *includes both the* <u>*header*</u> *and* <u>*payload*</u> *in bytes.

Example

Q) An IPv4 packet has arrived with the first 8 bits as $(01000010)_2$. Why does the receiver discard the packet?



Solution

There is an error in this packet. The 4 leftmost bits $(0100)_2$ show the version, which is correct. The next 4 bits $(0010)_2$ show an invalid header length of 8 bytes (2×4). The minimum number of bytes in the header must be 20 bytes.

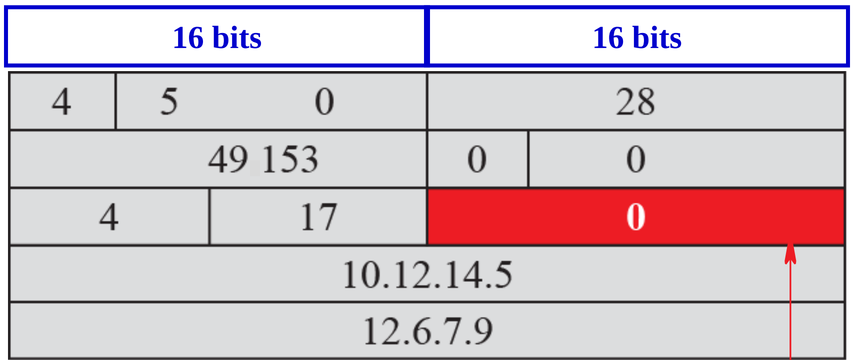
Q) In an IPv4 packet, the value of HLEN is $(1000)_2$. How many bytes of options are being carried by this packet?

Solution

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

Example

An example of a checksum calculation for an IPv4 header without options is shown:



4, 5, and 0	→		4	5	0	0
28	→		0	0	1	C
49153	→		C	0	0	1
0 and 0	→		0	0	0	0
4 and 17	→		0	4	1	1
0	→		0	0	0	0
10.12	→		0	A	0	C
14.5	→		0	E	0	5
12.6	→		0	C	0	6
7.9	→		0	7	0	9
Sum	→	1	3	4	4	E
Wrapped sum	→		3	4	4	F
Checksum	→		**C**	**B**	**B**	**0**

The header is divided into <u>16-bit sections</u>. All the sections are added and the sum is complemented after wrapping the leftmost digit. The result is inserted in the checksum field.

Note that the calculation of wrapped sum and checksum can also be done as follows in hexadecimal:

Wrapped Sum = Sum mod $FFFF_{16}$
Checksum = $FFFF_{16}$ – Wrapped Sum

IPv6

Outline

7.5 IPv6

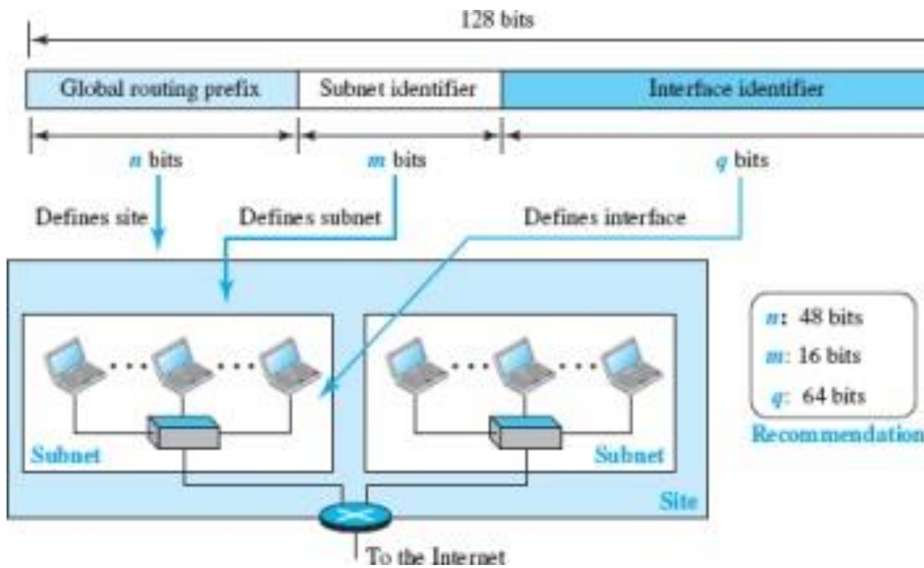
7-5 IPv6

The address depletion and other shortcomings of IPv4 prompted a new version of IP protocol in the 1990s. The new version, called Internet Protocol version 6 (IPv6), was a proposal to augment the address space of IPv4 and at the same time redesign the format of the IP packet.

The main changes in the new protocol were as follows: larger address space, better header format, new options, allowance for extension, support for resource allocation, and support for more security.

Figure 7.41: *Global Unicast address*

Like the address space for IPv4, the <u>address space</u> of <u>IPv6</u> is divided into <u>several blocks</u> of <u>varying size</u> and <u>each block</u> is <u>allocated</u> for a special <u>purpose</u>.



In <u>IPv4 addressing</u>, there is <u>not</u> a <u>specific relation</u> between the <u>hostid</u> (at the IP level) and <u>link-layer address</u> (at the data-link layer). <u>IPv6 addressing</u> allows this relationship, <u>eliminating the mapping process</u>, using two common <u>link-layer addressing</u> schemes: <u>64-bit extended unique identifier</u> (EUI-64) defined by IEEE and <u>48-bit link-layer address</u> defined by Ethernet.

7.5.1 IPv6 Addressing

One of the main reasons for migration from IPv4 to IPv6 is the small size of the address space in IPv4. An <u>IPv6 address</u> is <u>128 bits</u>, <u>4 times the address length of IPv4</u>.

A computer normally stores the address in binary, but 128 bits cannot easily be handled (by humans). In <u>IPv6 addressing</u>, the following <u>notations</u> are used: <u>binary</u> and <u>colon hexadecimal</u> (or colon hex):

Binary (128 bits)	***1111111011110110 1011 … 1111111100000000***
Colon hexadecimal	***FEF6:BA98:7654:3210:ADEF:BBFF:2922:FF00***

<u>Binary notation</u> is used when the addresses are <u>stored</u> in a <u>computer</u>. The <u>colon hexadecimal notation</u> divides the address into <u>eight sections</u>, <u>each made of four hexadecimal digits</u>, <u>separated by colons</u>.

7.5.2 IPv6 Protocol

The change of the IPv6 address size requires the change in the IPv4 packet format. The following describes other changes implemented in the protocol in addition to changing address size and format.

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 data.
(This simplifies and speeds up the routing process because most of the options do not need to be checked by routers)

Support for resource allocation: In IPv6, the type-of-service field in IPv4 has been removed, but two new fields, traffic class and flow label, have 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)

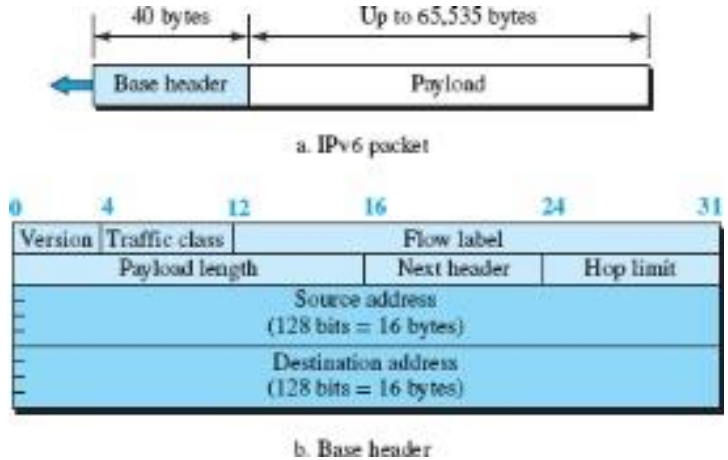
Support for more security: The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

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.

Figure 7.46: IPv6 datagram

The IPv6 datagram is shown below.



Each __packet__ is composed of a __base header__ followed by the __payload__. The __base header__ occupies __40 bytes__, whereas the __payload__ can be up to __65,535 bytes__ of information.

The __description__ of the __fields__ are shown on the next slide.

Figure 7.46: IPv6 datagram

***Version**: The 4-bit version field defines the version number of the IP.*

***Traffic class**: The 8-bit traffic class field is used to distinguish different payloads with different delivery requirements. It replaces the type-of-service field in IPv4.*

***Flow label**: The flow label is a 20-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 header.*

***Next header**: The next header is an 8-bit field defining the type of first extension header (if present) or the type of the data that follows the base header in the datagram. This field is similar to the protocol field in IPv4.*

***Hop limit**: The 8-bit hop limit field serves the same purpose as the TTL field in IPv4.*

***Source and destination address**: The source address field is a 128-bit Internet address that identifies the original source of the datagram. The destination address field is a 128-bit Internet address that identifies the destination of the datagram.*

***Payload**: Compared to IPv4, the payload field in IPv6 has a different format and meaning. The payload in IPv6 means a combination of zero or more extension headers (options) followed by the data from other protocols (UDP, TCP, and so on).*