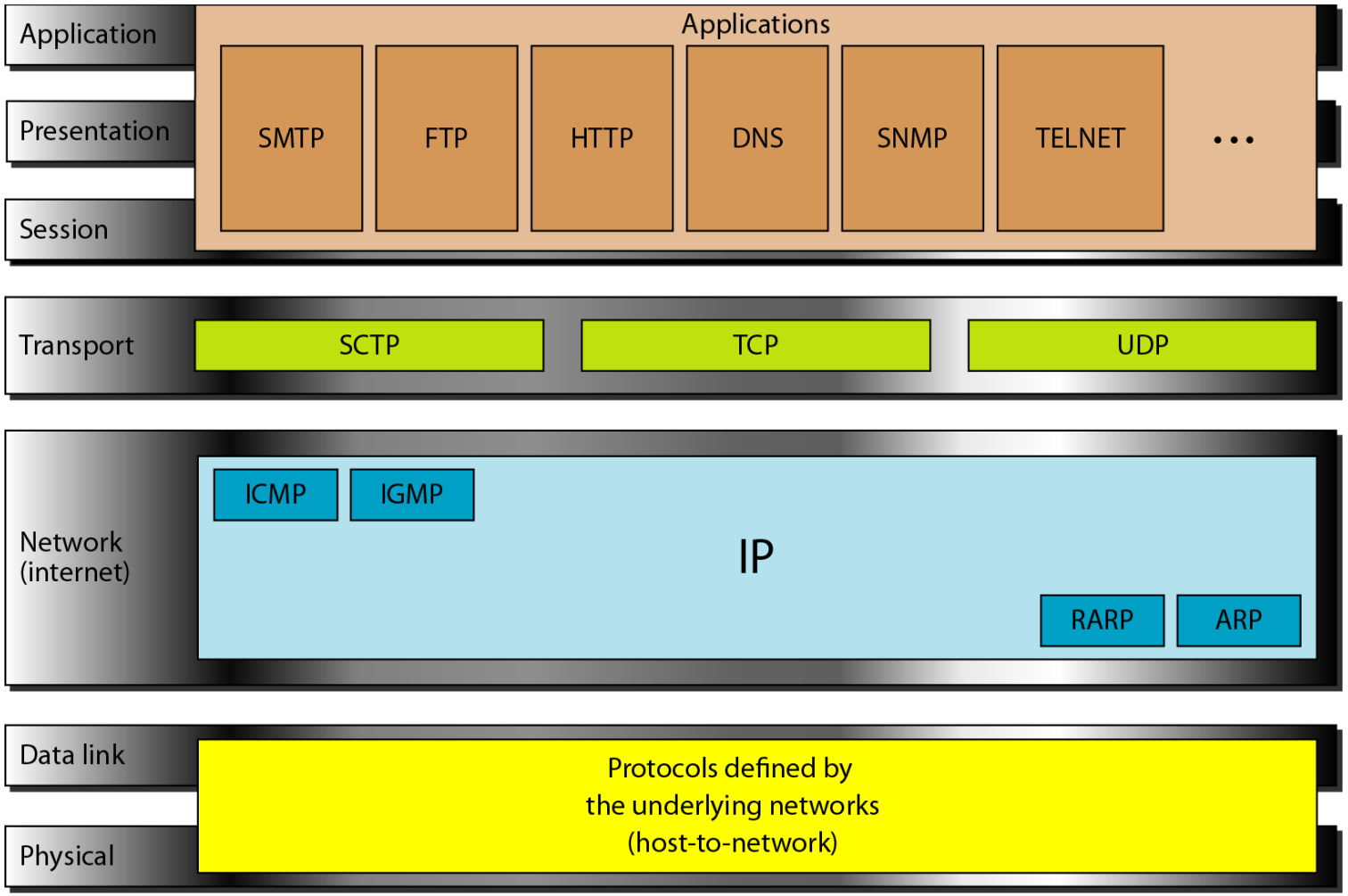
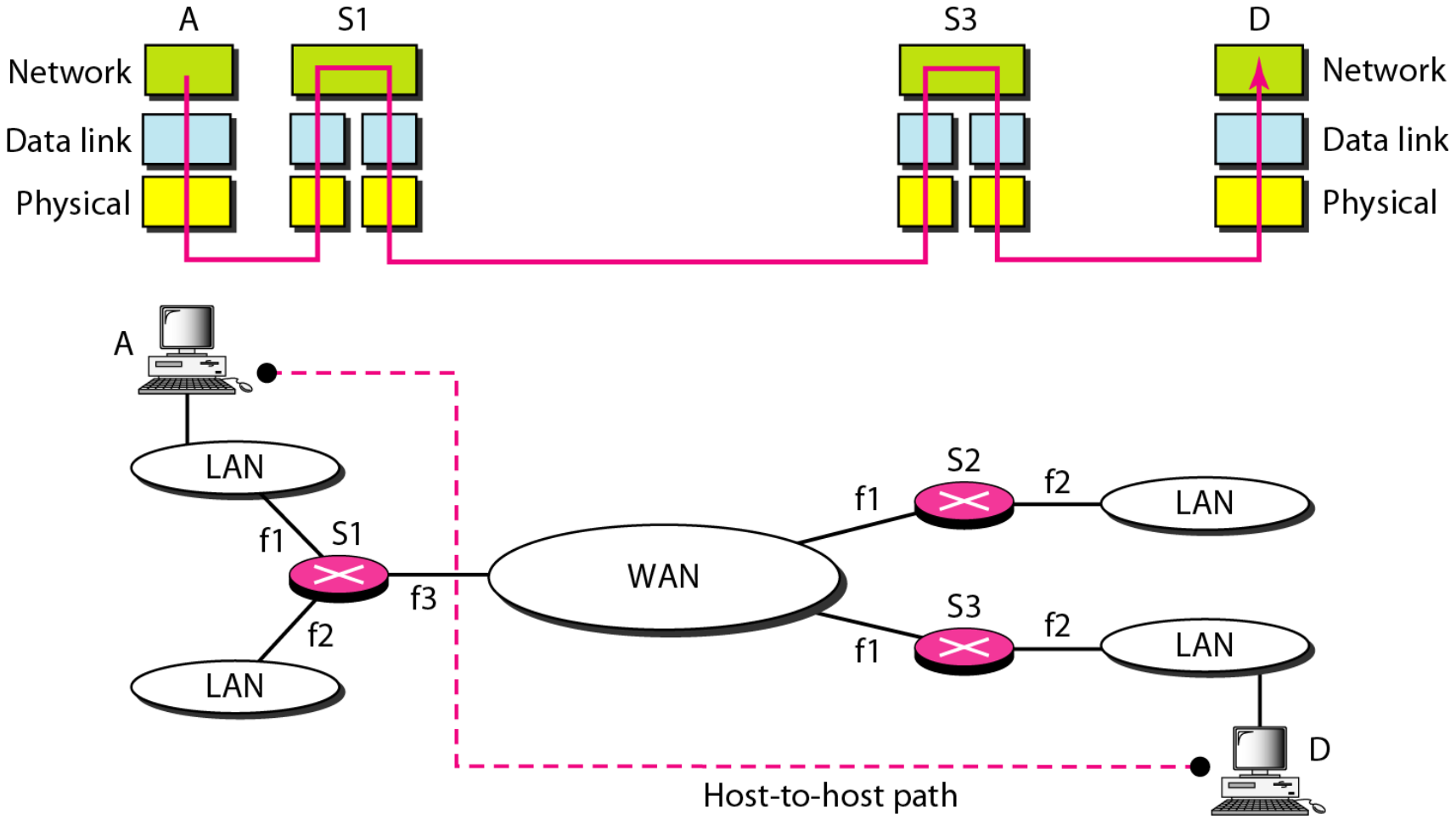
Network Layer: Internet Protocol

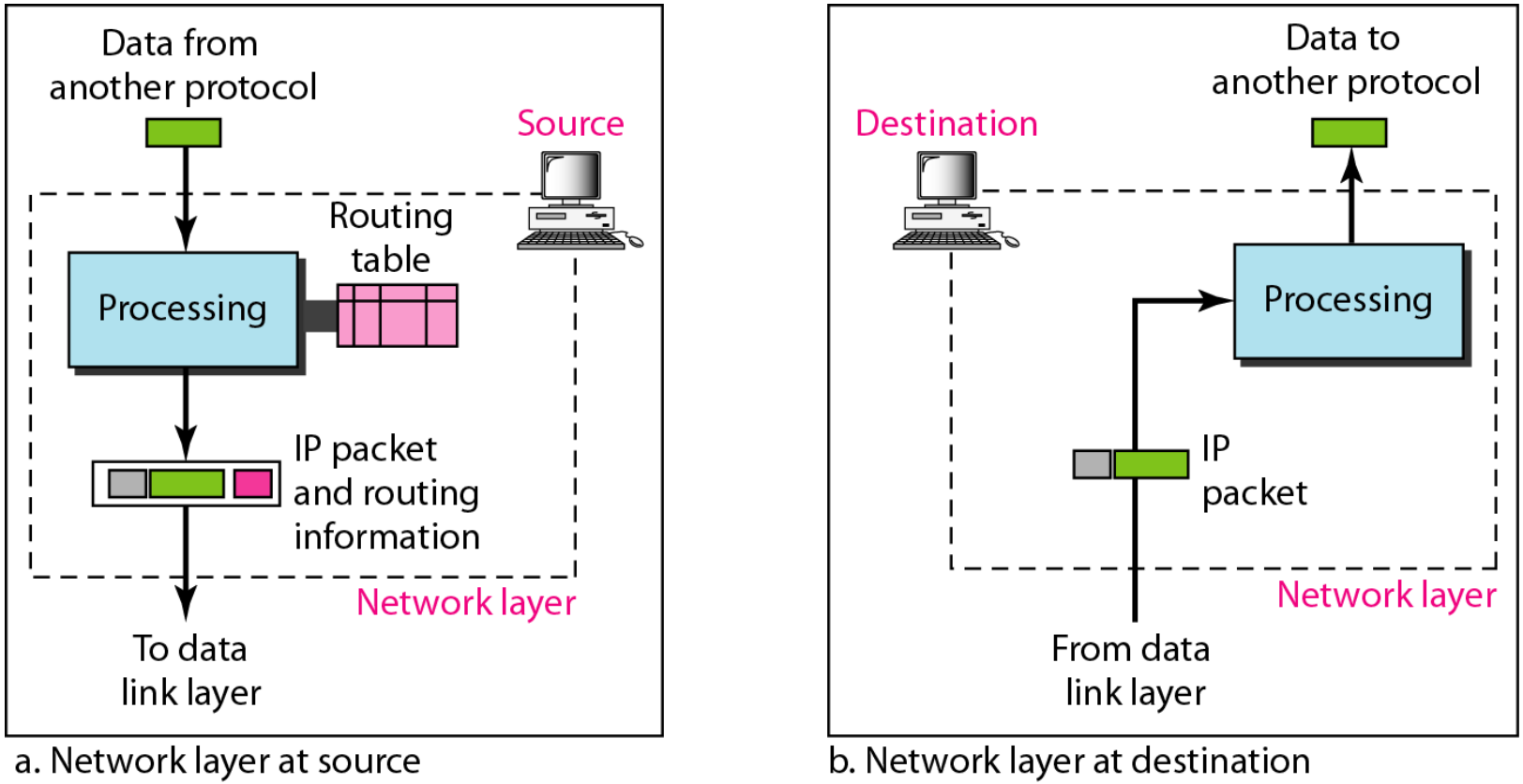
Internet Protocol



Network layer in an internetwork



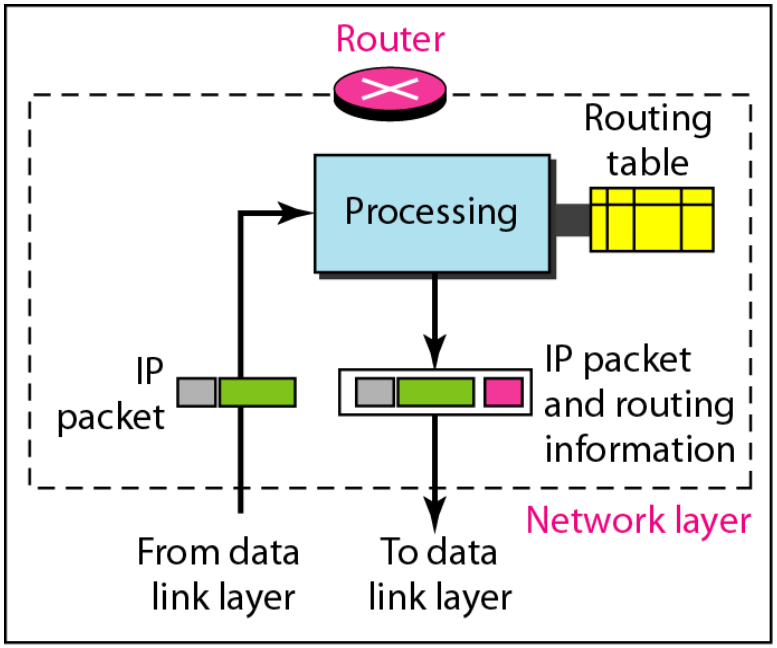
Network layer at the source, router, and destination



a. Network layer at source

b. Network layer at destination

Network layer at the source, router, and destination (continued)



c. Network layer at a router

- When a packet arrives, the router or switch consults its routing table and finds the interface from which the packet must be sent.

- The packet, after some changes in the header, with the routing information is passed to the data link layer again.

Switching Techniques

- In large networks there might be multiple paths linking sender and receiver.
- Information may be switched as it travels through various communication channels.
- There are three typical switching techniques available for digital traffic.
  - **Circuit Switching**
  - **Message Switching**
  - **Packet Switching**

Circuit Switching

- **Circuit switching**
  - Operates almost the same way as the telephone system works.
  - A complete end-to-end path must exist before communication can take place.
- The computer initiating the data transfer must ask for a connection to the destination.
- Once the connection has been initiated and completed to the destination device, the destination device must acknowledge that it is ready and willing to carry on a transfer.

Circuit switching

Advantages:

- The communication channel (once established) is dedicated.

Disadvantages:

- Possible *long wait to establish a connection*, (10 seconds, more on long-distance or international calls.) during which no data can be transmitted.

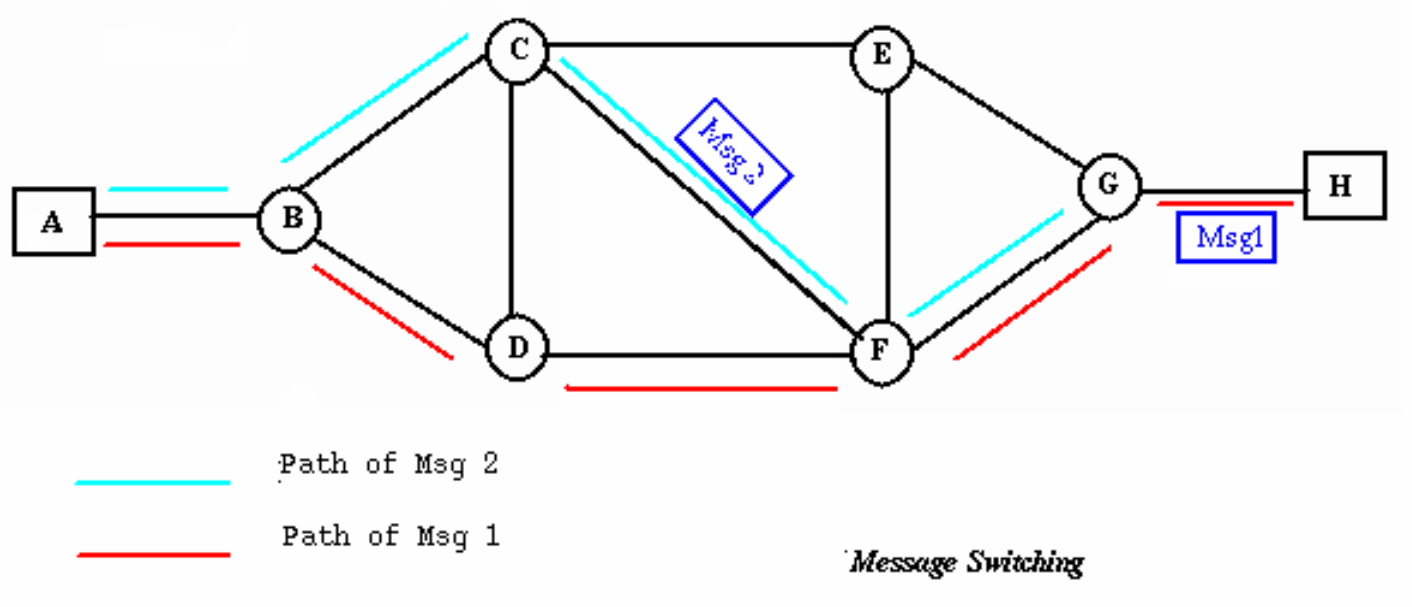
- *More expensive* than any other switching techniques, because a dedicated path is required for each connection.

- *Inefficient use of the communication channel*, because the channel is not used when the connected systems are not using it.

Message Switching

- There is no need to establish a dedicated path between two stations.
- When a station sends a message, the destination address is appended to the message.
- The message is then transmitted through the network, in its entirety, from node to node.
- Each node receives the entire message, stores it in its entirety on disk, and then transmits the message to the next node.
- This type of network is called a ***store-and-forward network***.

Message Switching



- A message-switching node is typically a general-purpose computer.
- The device needs sufficient secondary-storage capacity to store the incoming messages, which could be long.
- A time delay is introduced using this type of scheme due to store-and-forward time, plus the time required to find the next node in the transmission path.

Message Switching

Advantages:

- Channel efficiency can be greater compared to circuit-switched systems, because more devices are sharing the channel.
- Traffic congestion can be reduced, because messages may be temporarily stored in route.
- Message priorities can be established due to store-and-forward technique.
- Message broadcasting can be achieved with the use of broadcast address appended in the message.

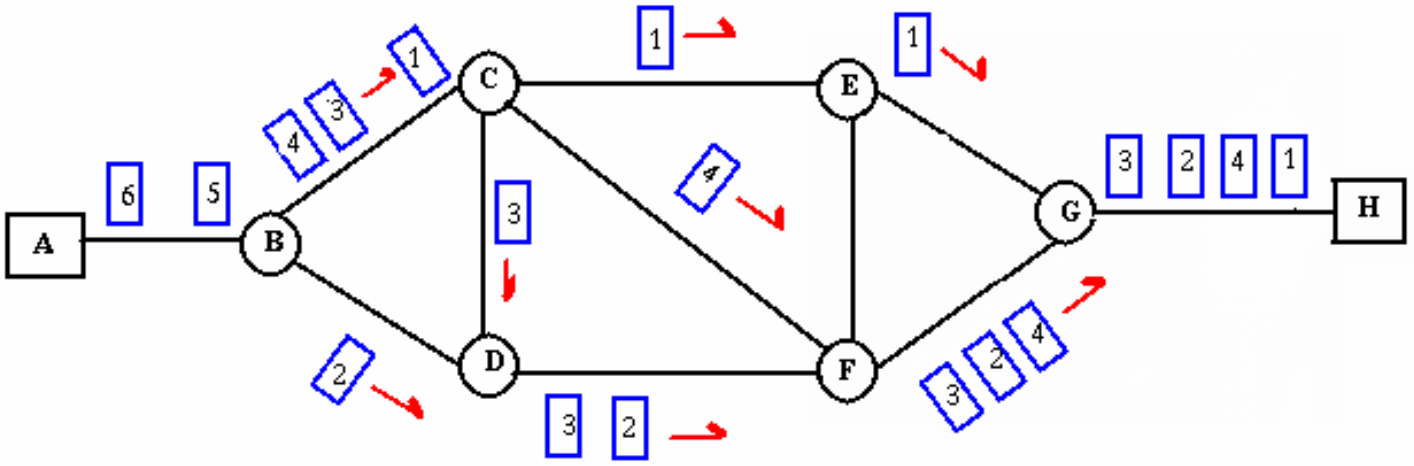
Message Switching

Disadvantages

- Message switching is not compatible with interactive applications.
- Store-and-forward devices are expensive, because they must have large disks to hold potentially long messages.

Packet Switching

- ***Packet switching*** can be seen as a solution that tries to combine the advantages of message and circuit switching and to minimize the disadvantages of both.
- There are two methods of packet switching:
  - **Datagram**
  - **Virtual circuit.**



Packet Switching

Packet Switching: Datagram

- Datagram packet switching is similar to message switching in that each packet is a self-contained unit with complete addressing information attached.

- Allows packets to take a variety of possible paths through the network.

- So the packets, each with the same destination address, do not follow the same route, and they may arrive out of sequence at the exit point node (or the destination).

- Reordering is done at the destination point based on the sequence number of the packets.

- It is possible for a packet to be destroyed if one of the nodes on its way is crashed momentarily. Thus all its queued packets may be lost.

Packet Switching: Virtual Circuit

- In the virtual circuit approach, a preplanned route is established before any data packets are sent.
- A logical connection is established when
  - a sender send a *"call request packet"* to the receiver and
  - the receiver send back an acknowledge packet *"call accepted packet"* to the sender if the receiver agrees on conversational parameters.
- The conversational parameters can be maximum packet sizes, path to be taken, and other variables necessary to establish and maintain the conversation.
- Virtual circuits imply acknowledgements, flow control, and error control, so virtual circuits are reliable.
- That is, they have the capability to inform upper-protocol layers if a transmission problem occurs.

Packet Switching: Virtual Circuit

- In virtual circuit, the route between stations does not mean that this is a dedicated path, as in circuit switching.
- A packet is still buffered at each node and queued for output over a line.
- The difference between virtual circuit and datagram approaches:
  - With virtual circuit, the node does not need to make a routing decision for each packet.
  - It is made only once for all packets using that virtual circuit.

Packet Switching: Virtual Circuit

VC's offer guarantees that

- the packets sent arrive in the order sent
- with no duplicates or omissions
- with no errors (with high probability) regardless of how they are implemented internally.

Advantages of packet switching

Advantages:

- Packet switching is cost effective, because switching devices do *not need massive amount of secondary storage*.
- Packet switching offers *improved delay characteristics*, because there are no long messages in the queue (maximum packet size is fixed).
- Packet can be *rerouted* if there is any problem, such as, busy or disabled links.
- Many *network users can share the same channel* at the same time.
    - Packet switching can *maximize link efficiency* by making optimal use of link bandwidth.

Disadvantages of packet switching

Disadvantages:

- Protocols for packet switching are typically more complex.
- It can add some initial costs in implementation.
- If packet is lost, sender needs to retransmit the data.
- Another disadvantage is that packet-switched systems still can’t deliver the same quality as dedicated circuits in applications requiring very little delay - like voice conversations or moving images.

Note

Switching at the network layer in the Internet uses the datagram approach to packet switching.

Connection Oriented Vs. Connectionless Network

- Delivery of a packet can be accomplished by using either a
  - Connection-oriented
    - the source first makes a connection with the destination before sending a packet.
    - a sequence of packets from the same source to the same destination can be sent one after another.
    - They are sent on the same path in sequential order.
    - When all packets of a message have been delivered, the connection is terminated.

Connection Oriented Vs. Connectionless Network

- Connectionless network 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.
  - 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.**

Internet as a Connectionless Network

- Reasons:
  - The Internet is made of so many heterogeneous networks
  - It is almost impossible to create a connection from the source to the destination without knowing the nature of the networks in advance

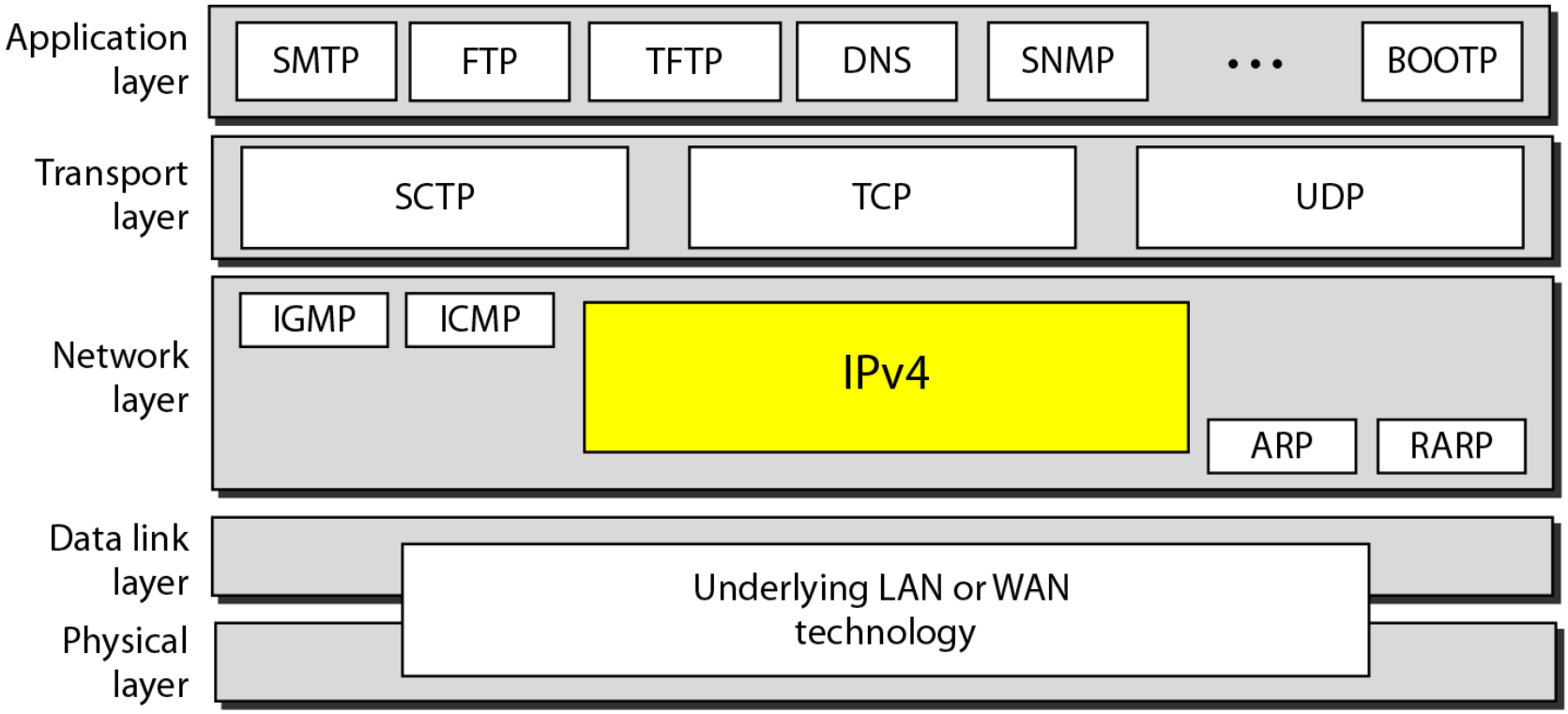
IPv4

- ***The Internet Protocol version 4 (IPv4) is the delivery mechanism used by the TCP/IP protocols.***

- ***IPv4 provides no error control or flow control (except for error detection on the header).***

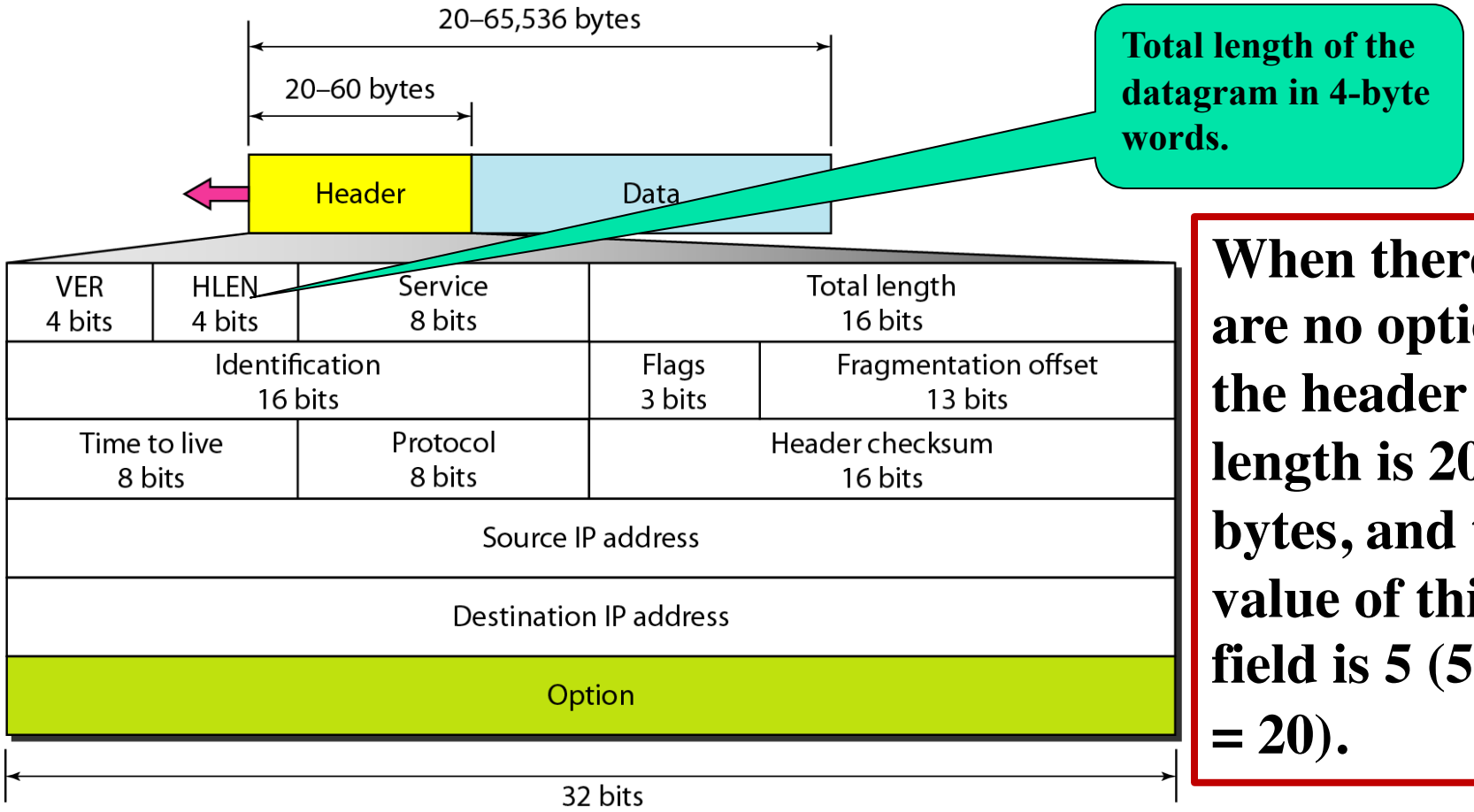
- ***We will know***
  - **Datagram**
  - **Fragmentation**
  - **Checksum**
  - **Options**

Position of IPv4 in TCP/IP protocol suite



IPv4 datagram format

- Packets in the IPv4 layer are called datagrams.
- Figure shows the IPv4 datagram format.



When there are no options, the header length is 20 bytes, and the value of this field is 5 (5 x 4 = 20).

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 × 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 0x0028. 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 × 4, or 20 bytes (no options).***

- ***The total length is 40 bytes, which means the packet is carrying 20 bytes of data (40 – 20).***

Example

An IPv4 packet has arrived with the first 8 bits as shown:

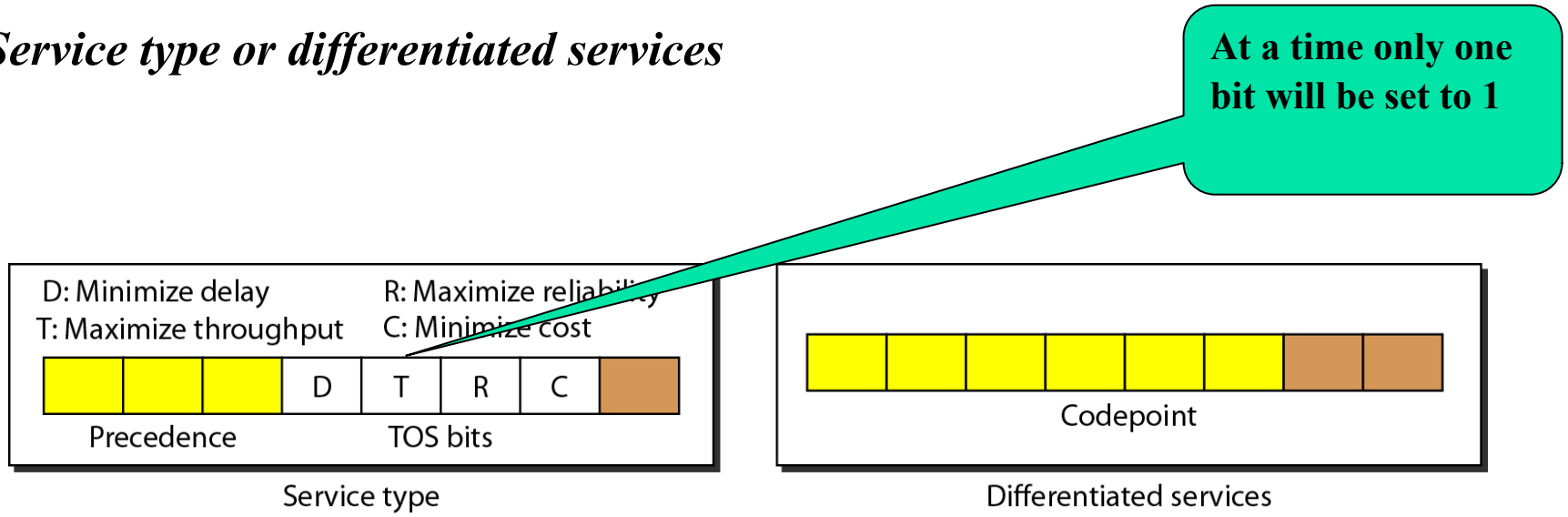
01000010

The receiver discards the packet. Why?

Solution

- *There is an error in this packet.*
- *The 4 leftmost bits (0100) show the version, which is correct.*
- *The next 4 bits (0010) show an invalid header length ($2 \times 4 = 8$).*
- *The minimum number of bytes in the header must be 20.*
- *The packet has been corrupted in transmission.*

Service type or differentiated services



- **The *precedence* defines the priority of the datagram in issues such as congestion.**
  - a datagram used for network management is much more urgent and important than a datagram containing optional information for a group.

Types of service

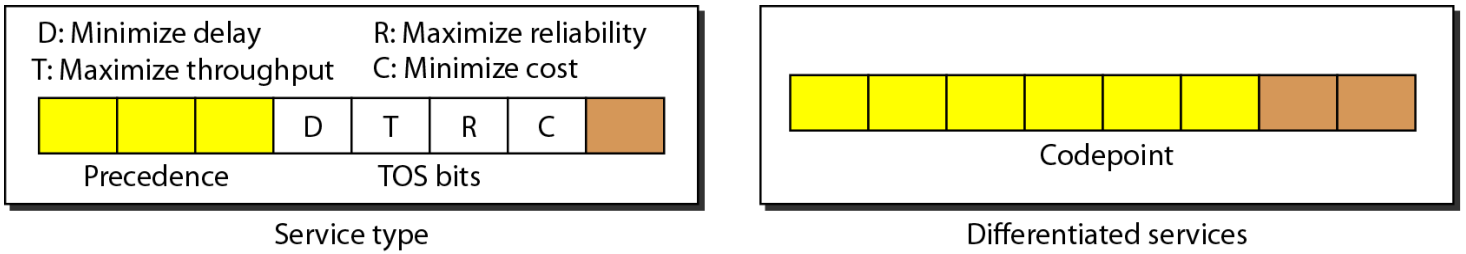
TOS Bits	*Description*
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

Default types of service: Requested by Application Programs

Protocol	*TOS Bits*	*Description*
ICMP	0000	Normal
BOOTP	0000	Normal
NNTP	0001	Minimize cost
IGP	0010	Maximize reliability
SNMP	0010	Maximize reliability
TELNET	1000	Minimize delay
FTP (data)	0100	Maximize throughput
FTP (control)	1000	Minimize delay
TFTP	1000	Minimize delay
SMTP (command)	1000	Minimize delay
SMTP (data)	0100	Maximize throughput
DNS (UDP query)	1000	Minimize delay
DNS (TCP query)	0000	Normal
DNS (zone)	0100	Maximize throughput

- **Interactive activities :**activities requiring immediate attention
- **Activities requiring immediate response** need minimum delay.
- **Those activities that send bulk data require** maximum throughput**.**
- **Management activities need** maximum reliability.
- **Background activities** need minimum cost.

Differentiated Services



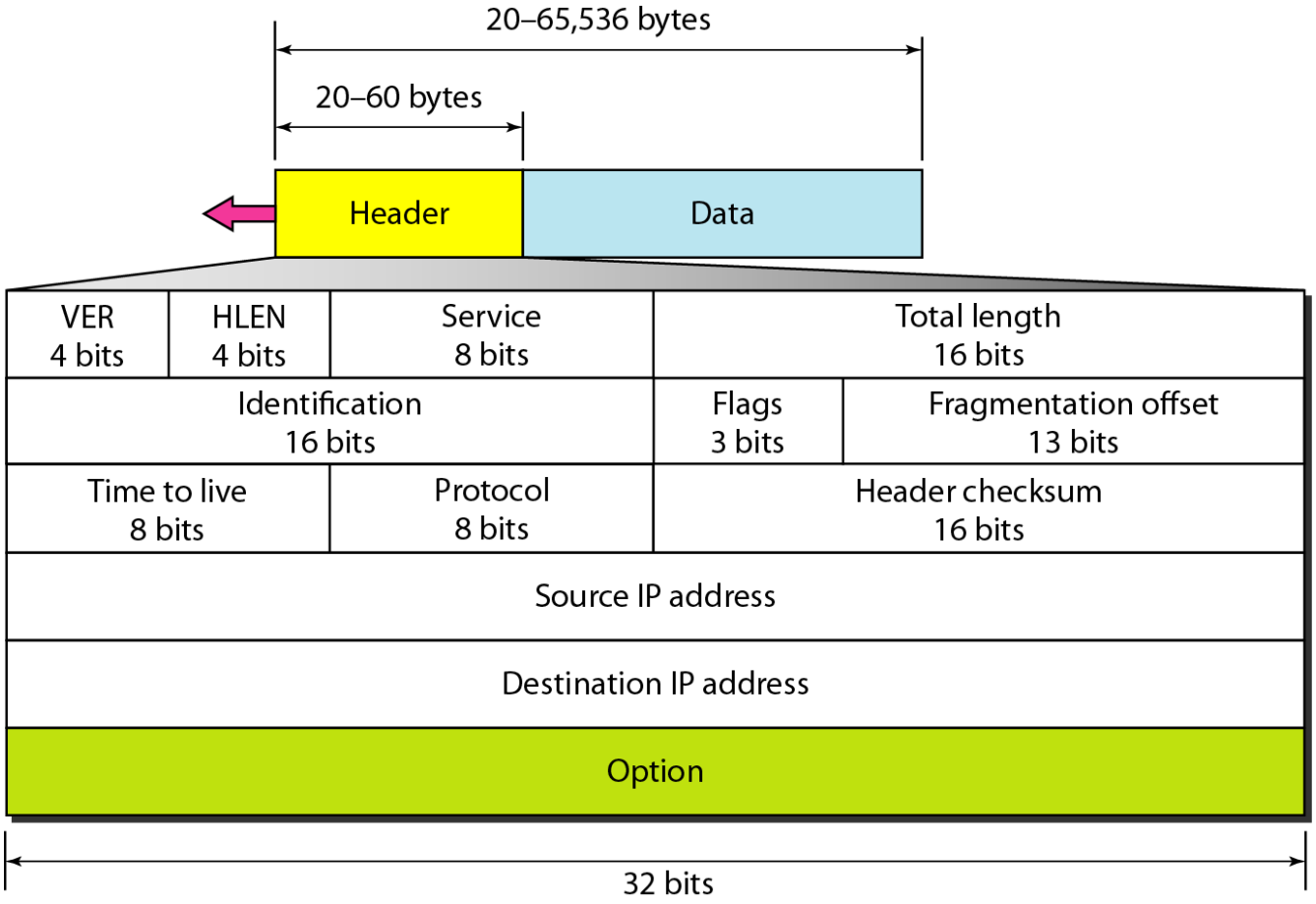
- When the 3 rightmost bits are 0s, the 3 leftmost bits are interpreted the same as the precedence bits in the service type interpretation. So, it is compatible with the old interpretation.

- When the 3 rightmost bits are not all 0s, the ***6 bits define 64 services*** based on the priority assignment by the Internet or local authorities

- The first category contains 32 service types; the second and the third each contain 16. The first category (numbers 0, 2,4, ... ,62) is assigned by the Internet authorities (IETF). The second category (3, 7, 11, 15, , 63) can be used by local authorities (organizations). The third category (1, 5, 9, ,61) is temporary and can be used for experimental purposes.

Category	*Codepoint*	*Assigning Authority*
1	XXXXXO	Internet
2	XXXX11	Local
3	XXXXOI	Temporary or experimental

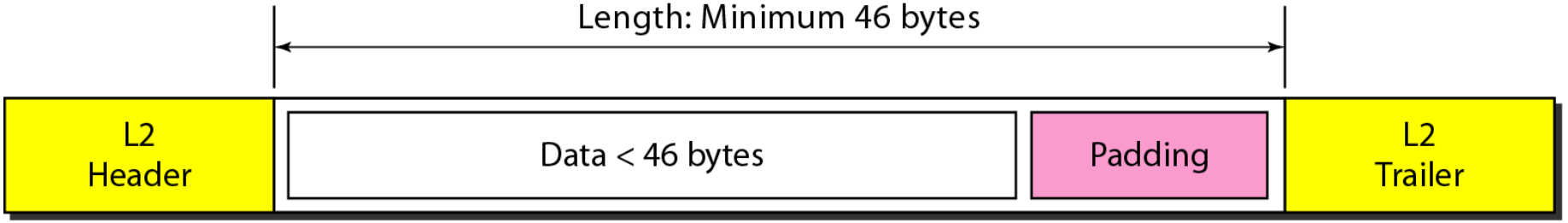
IPv4 datagram format



IPv4 datagram format

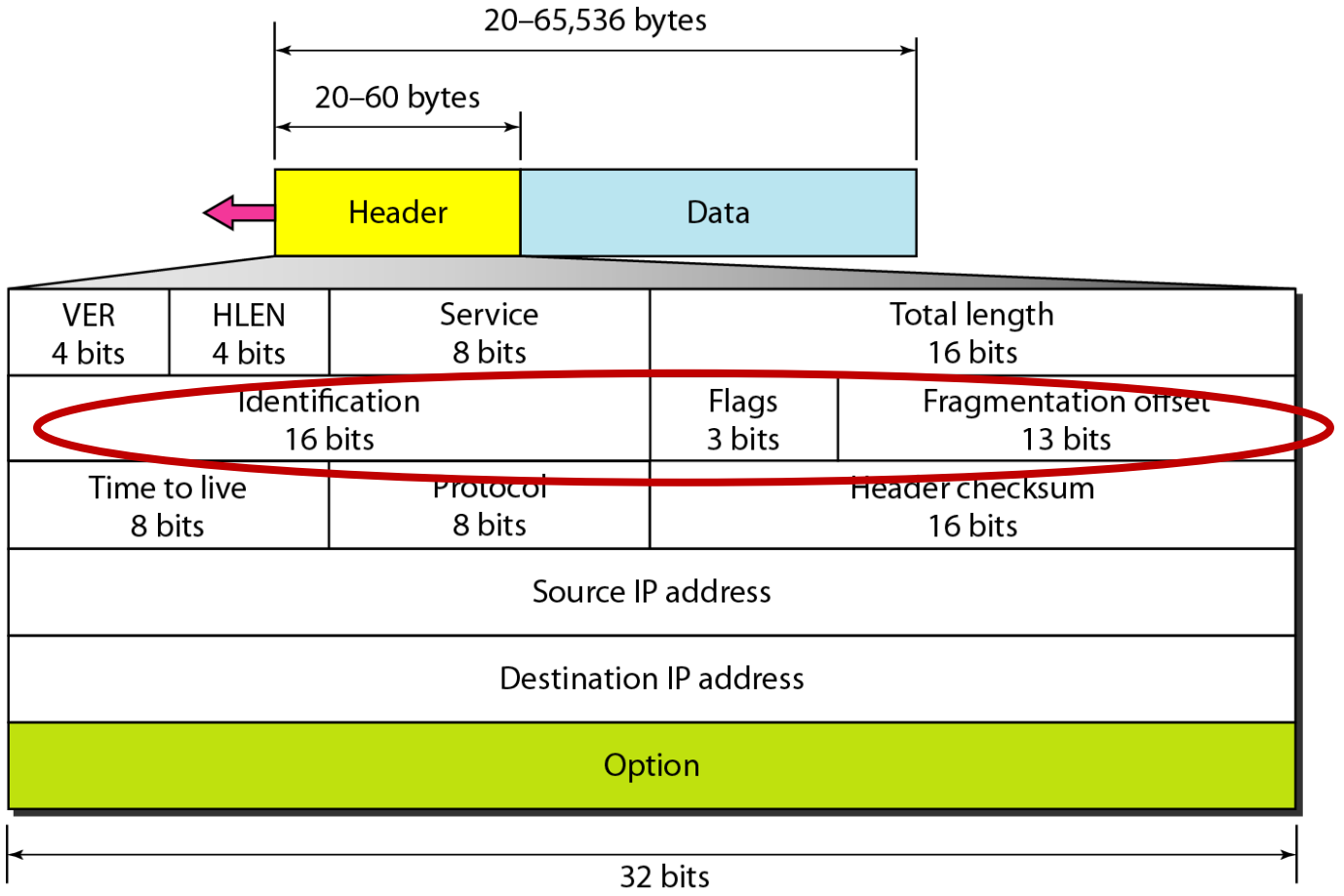
- **Total Length:**
  - This is a In-bit field that defines the total length (header plus data) of the IPv4 datagram in bytes.
  - To find the length of the data coming from the upper layer, subtract the header length from the total length.
  - The header length can be found by multiplying the value in the HLEN field by 4.
  - Though a size of 65,535 bytes might seem large, the size of the IPv4 datagram may increase in the near future as the underlying technologies allow even more throughput (greater bandwidth).

Encapsulation of a small datagram in an Ethernet frame



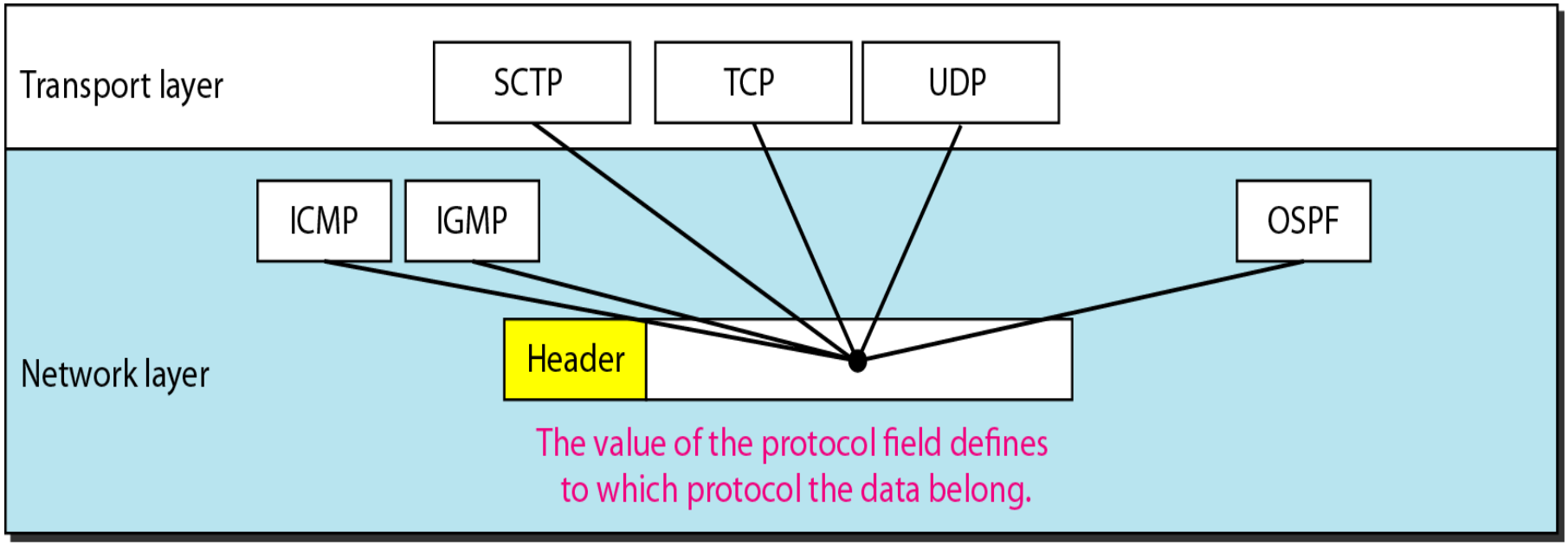
Data or payload length: 46-1500 bytes

Ipv4



- **Time to Live (TTL):**
  - 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.

Protocol field and encapsulated data



Protocol values (in 8 Bits)

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

Example

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

0x45000028000100000102 . . .

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

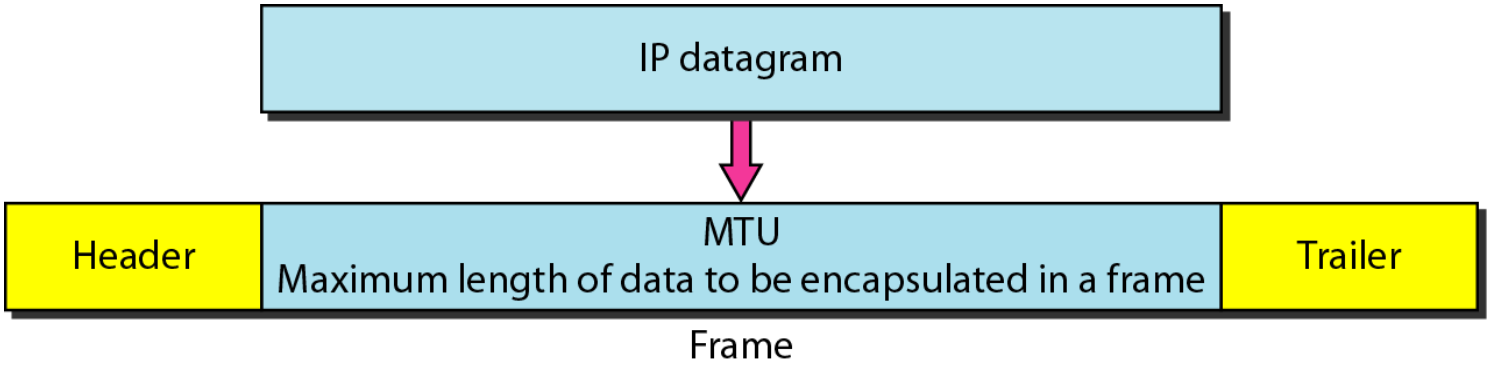
Solution

- ***To find the time-to-live field, we skip 8 bytes.***
- ***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.***

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.
- The value of the MTU depends on the physical network protocol.



MTUs for some networks

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

Fragmentation

- To make the IPv4 protocol independent of the physical network, the designers decided to make the maximum length of the IPv4 datagram equal to 65,535 bytes.
- This makes transmission more efficient if we use a protocol with an MTU of this size.
- However, for other physical networks, we must divide the datagram to make it possible to pass through these networks.
- This is called fragmentation.

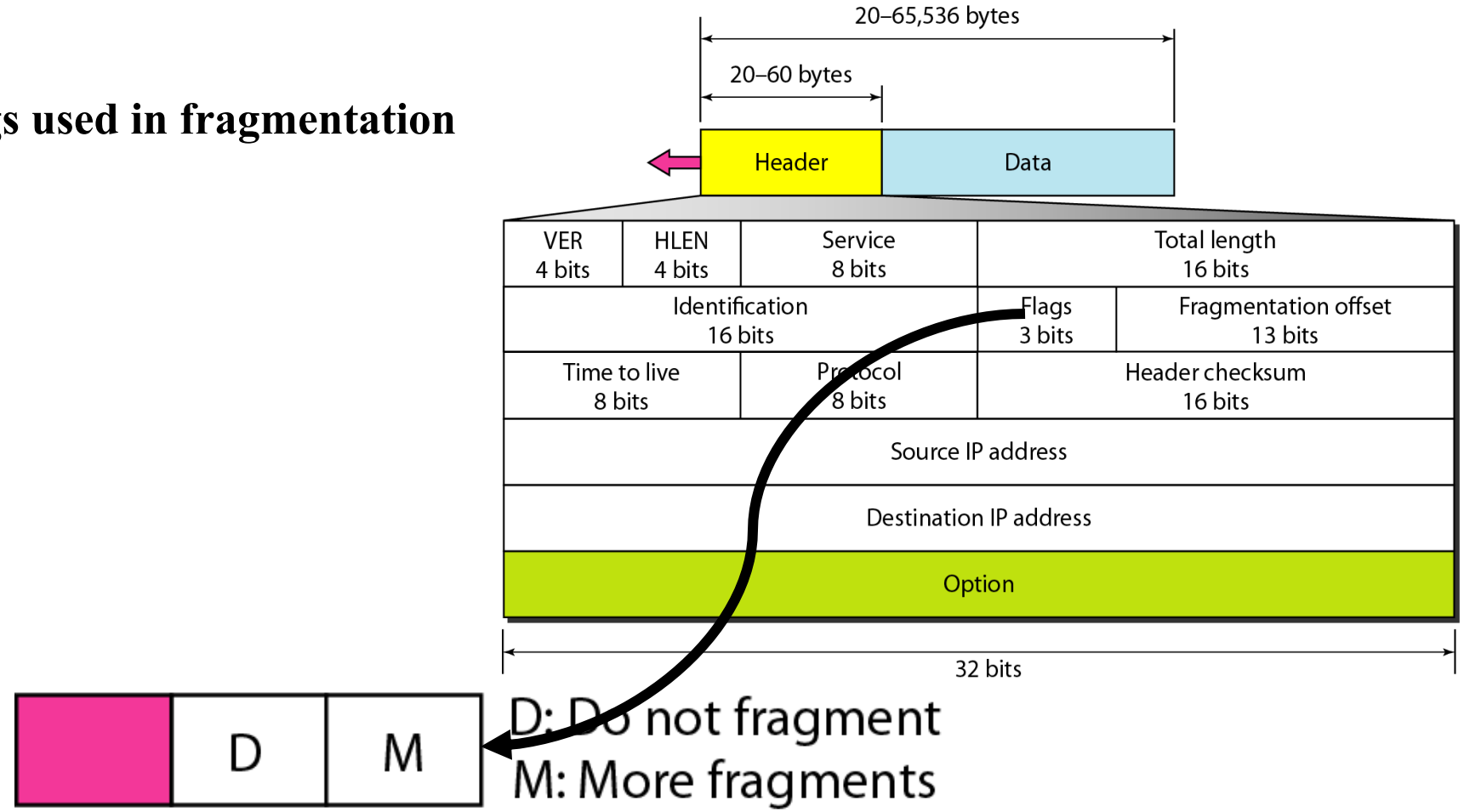
Identification

- 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.
- To guarantee uniqueness, the IPv4 protocol uses a counter to label the datagrams.
- The counter is initialized to a positive number.
- When the IPv4 protocol sends a datagram, it copies the current value of the counter to the identification field and increments the counter by 1.

Identification

- When a datagram is fragmented, the value in the identification field is copied to all fragments.
- The identification number helps the destination in reassembling the datagram.

Flags used in fragmentation



- **If D=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 D=0, the datagram can be fragmented if necessary.**

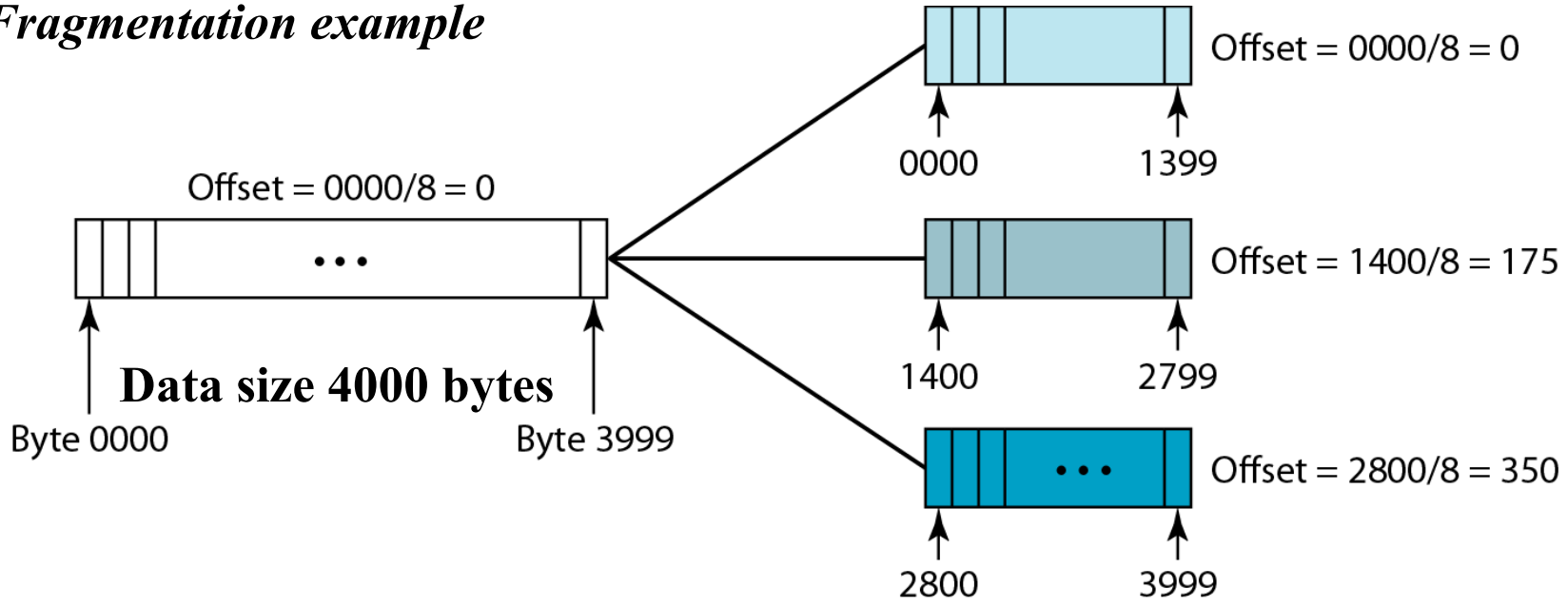
Flags used in fragmentation

- **M 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

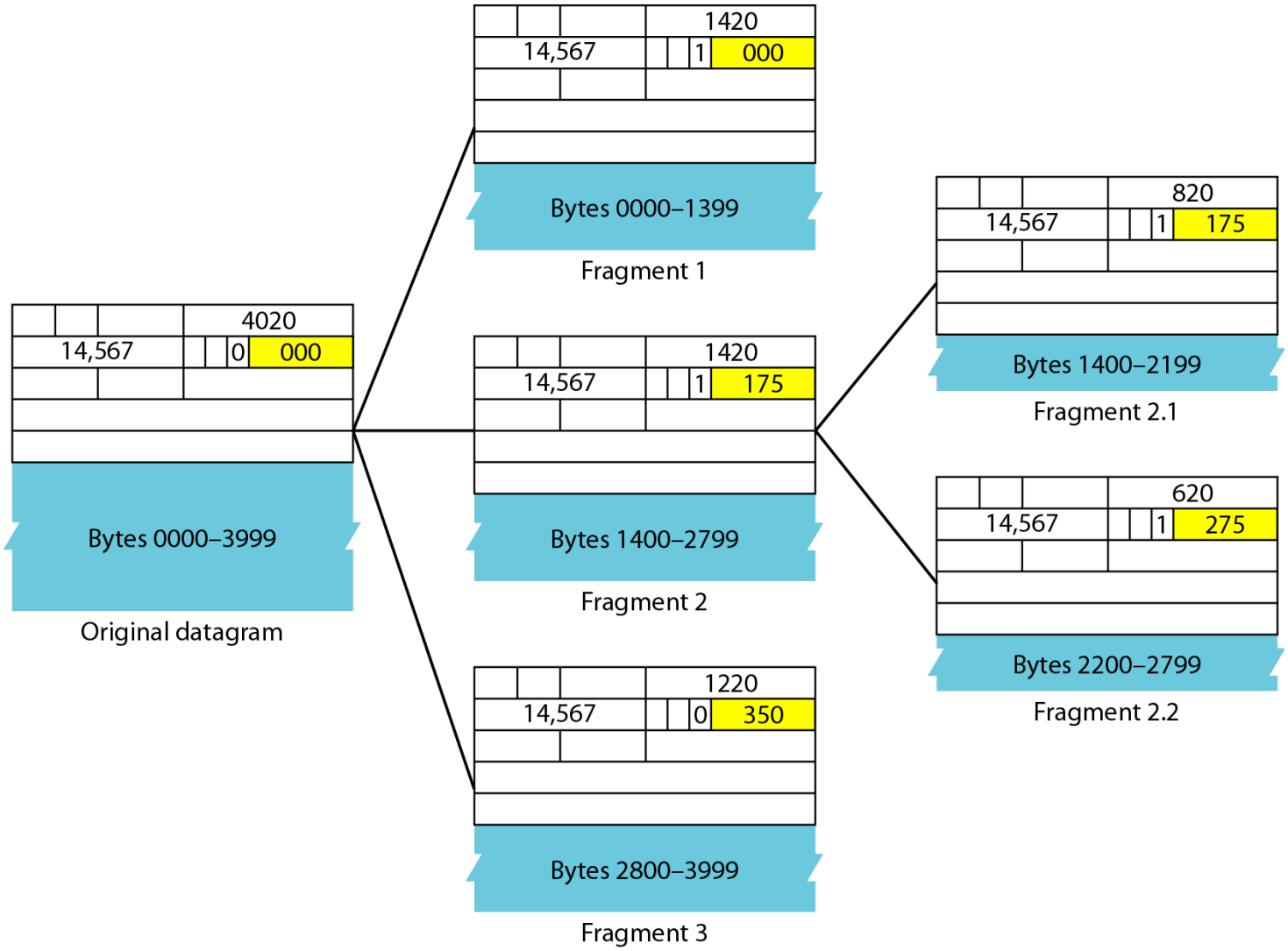
Fragmentation Offset

- 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.

Fragmentation example



Detailed fragmentation example



Example

A packet has arrived with an M bit value of 0. 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).***

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 1, 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.***

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 × 8 = 800.***

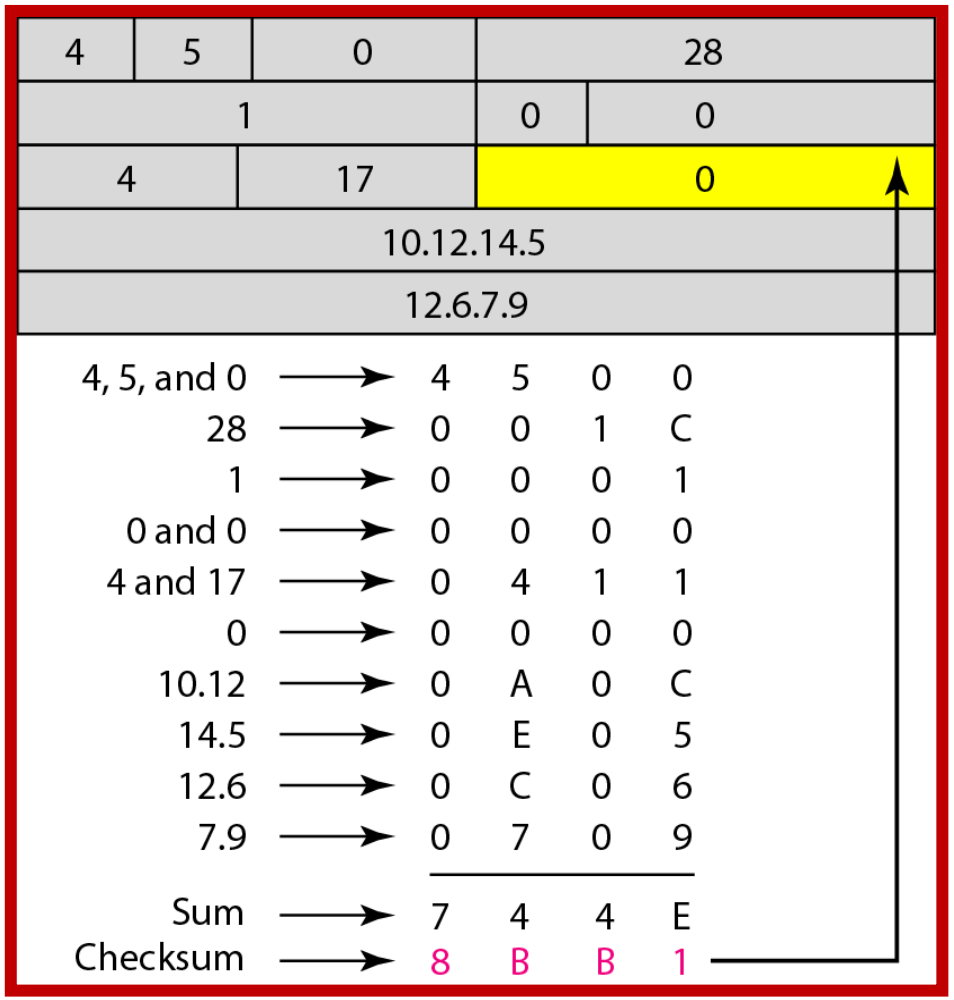
- ***The total length is 100 bytes, and the header length is 20 bytes (5 × 4), which means that there are 80 bytes in this datagram.***

- ***If the first byte number is 800, the last byte number must be 879.***

Checksum

- Figure shows an example of a checksum calculation for an IPv4 header without options.
- The header is divided into 16-bit sections.
- All the sections are added and the sum is complemented. The result is inserted in the checksum field.

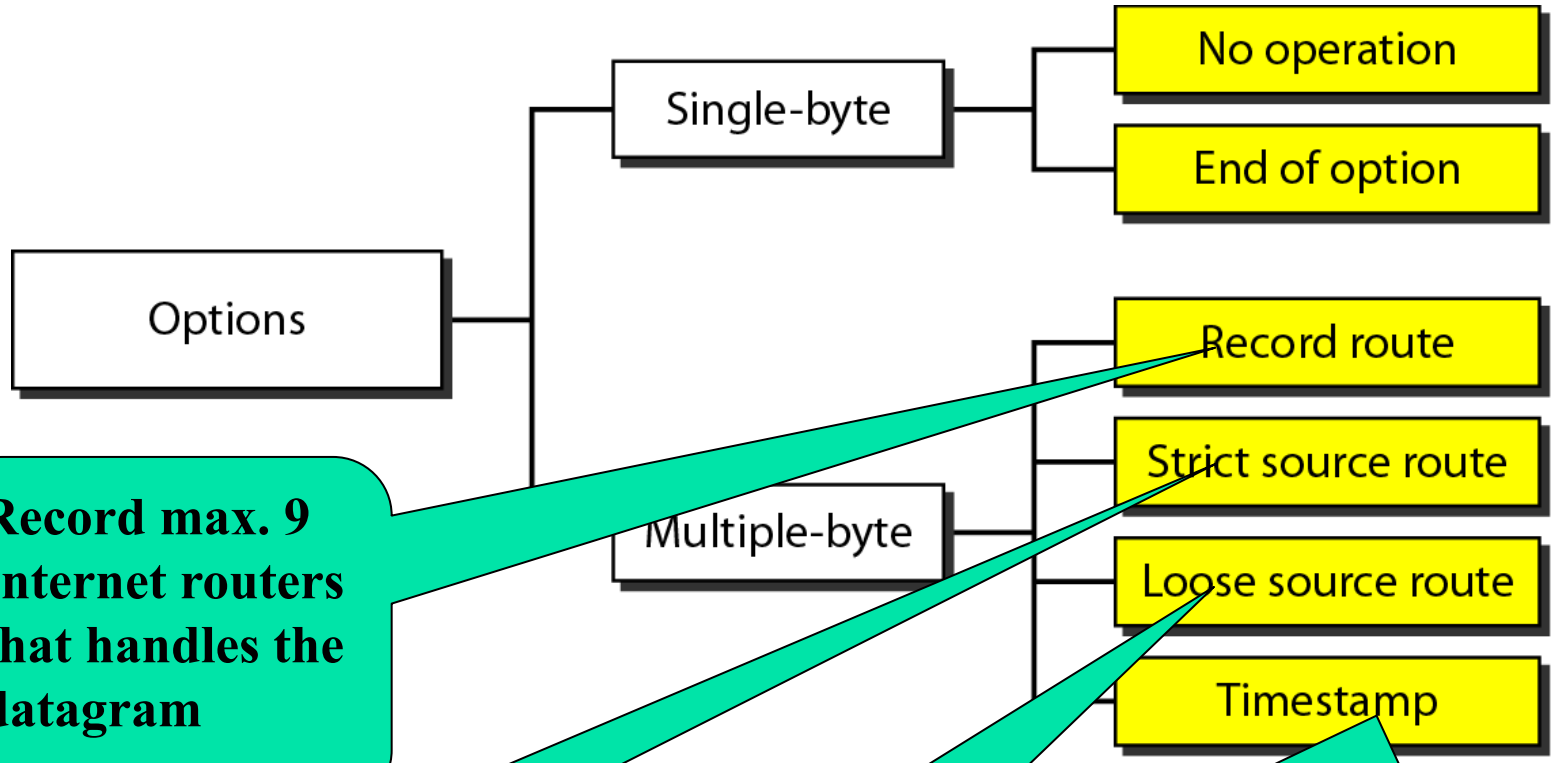
Example of checksum calculation in IPv4



Options

- The header of the IPv4 datagram is made of two parts:
  - a fixed part (20 bytes)
  - a variable part (40 bytes maximum) : Option
- Variable part used for network testing and debugging.

Taxonomy of options in IPv4



1 byte

1 byte

Record max. 9 Internet routers that handles the datagram

Sender can choose a route for a specific type of service to get better throughput

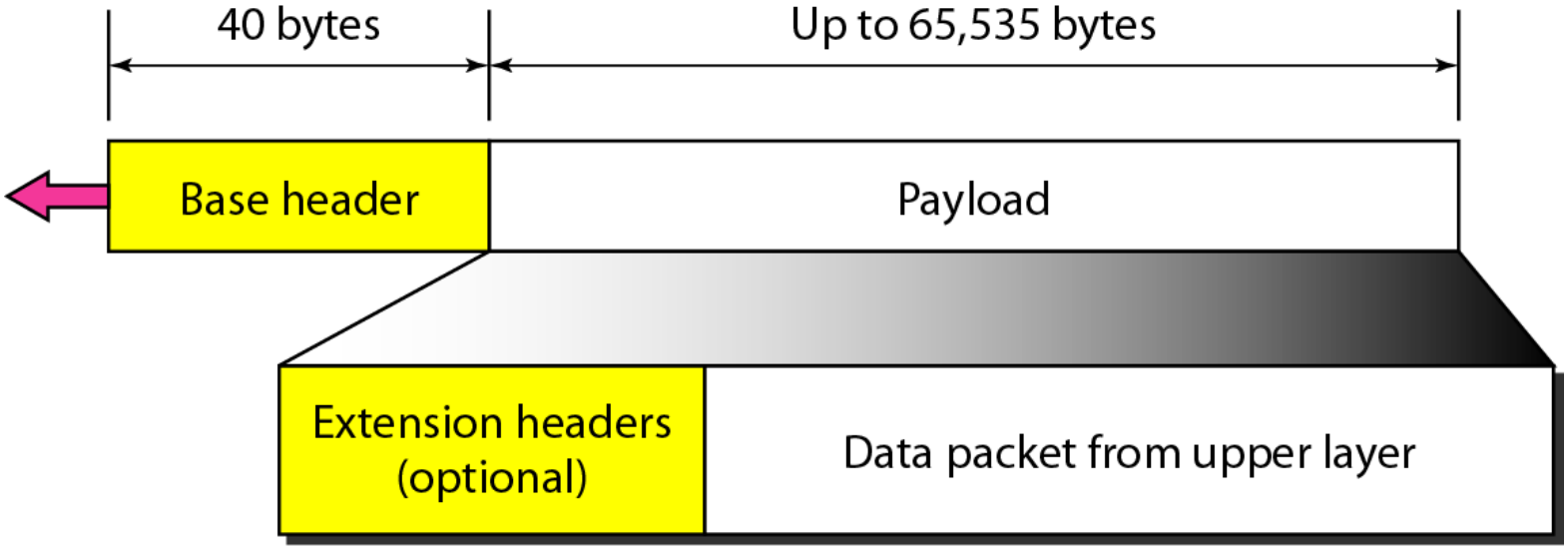
Same as Strict source route but less rigid

It is used top record the time of datagram, processing by the router. This helps to get estimate to required to go from one router to another router.

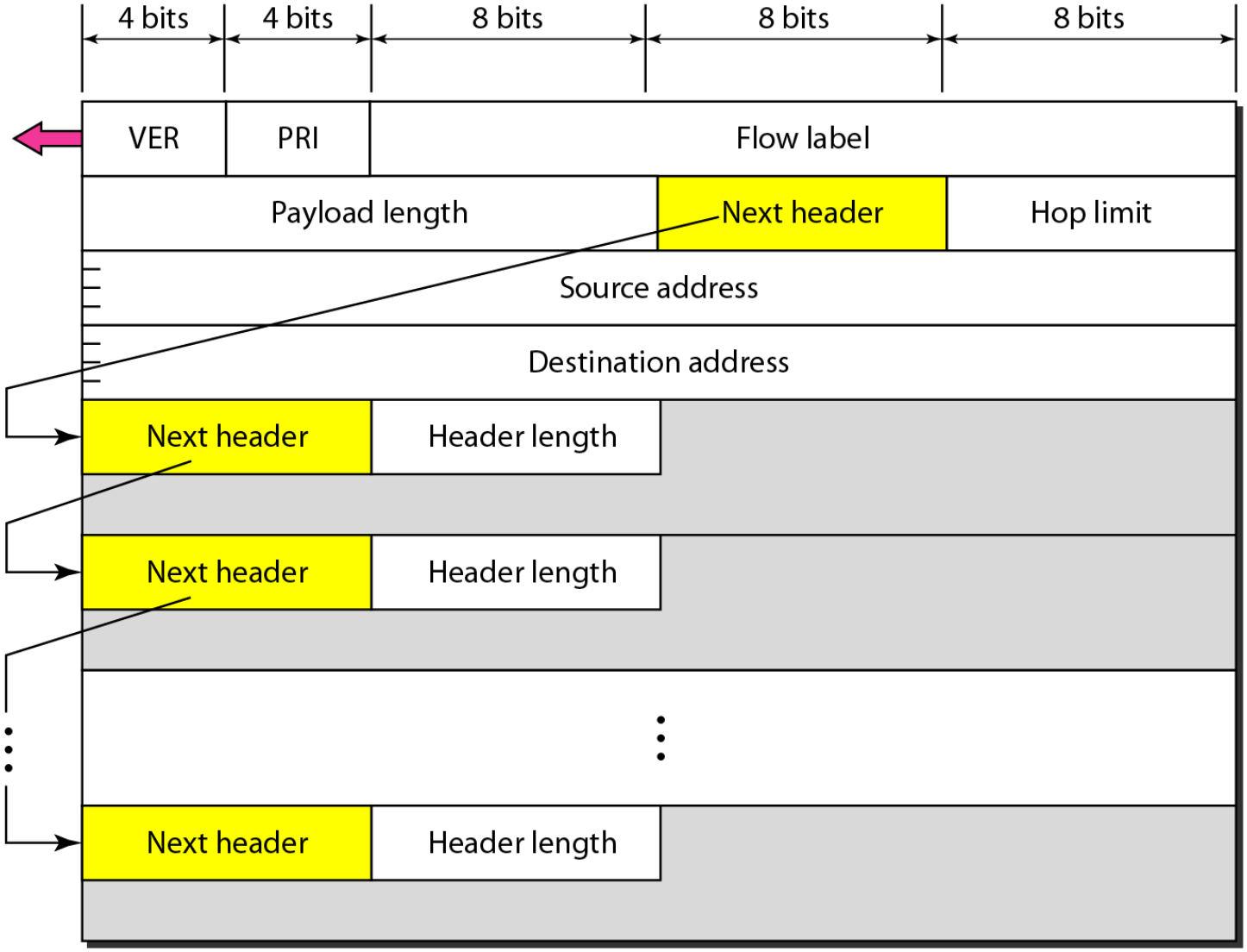
IPv6

- ***The network layer protocol in the TCP/IP protocol suite is currently IPv4.***
- ***Although IPv4 is well designed, data communication has evolved since the inception of IPv4 in the 1970s.***
- ***IPv4 has some deficiencies that make it unsuitable for the fast-growing Internet.***

IPv6 datagram header and payload



Format of an IPv6 datagram



IPv6

- **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 like real time 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 (40 bytes) 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.

Priorities for congestion-controlled traffic

Priority	*Meaning*
0	No specific traffic
1	Background data
2	Unattended data traffic
3	Reserved
4	Attended bulk data traffic
5	Reserved
6	Interactive traffic
7	Control traffic

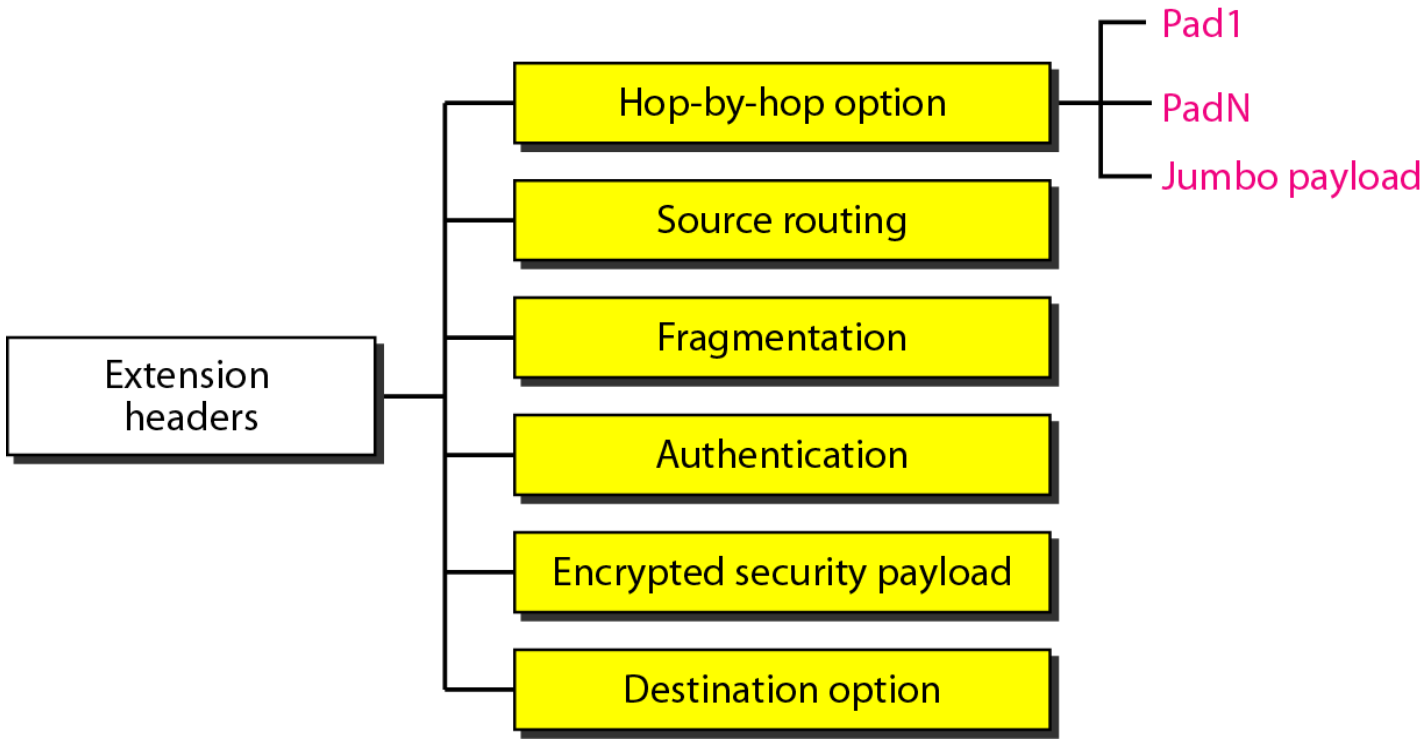
Priorities for noncongestion-controlled traffic

Priority	*Meaning*
8	Data with greatest redundancy
. . .	. . .
15	Data with least redundancy

Next header codes for IPv6

Code	*Next Header*
0	Hop-by-hop option
2	ICMP
6	TCP
17	UDP
43	Source routing
44	Fragmentation
50	Encrypted security payload
51	Authentication
59	Null (no next header)
60	Destination option

Extension header types



Comparison between IPv4 and IPv6 packet headers

Comparison
1. The header length field is eliminated in IPv6 because the length of the header is fixed in this version.
2. The service type field is eliminated in IPv6. The priority and flow label fields together take over the function of the service type field.
3. The total length field is eliminated in IPv6 and replaced by the payload length field.
4. The identification, flag, and offset fields are eliminated from the base header in IPv6. They are included in the fragmentation extension header.
5. The TTL field is called hop limit in IPv6.
6. The protocol field is replaced by the next header field.
7. The header checksum is eliminated because the checksum is provided by upper-layer protocols; it is therefore not needed at this level.
8. The option fields in IPv4 are implemented as extension headers in IPv6.

Comparison between IPv4 options and IPv6 extension headers

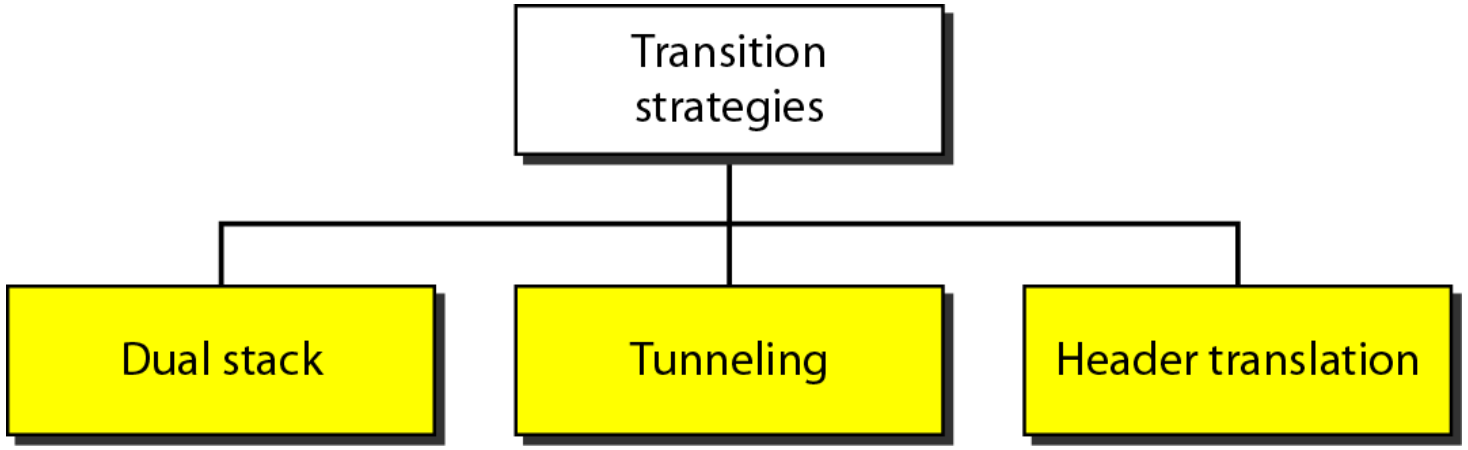
Comparison
1. The no-operation and end-of-option options in IPv4 are replaced by Pad1 and PadN options in IPv6.
2. The record route option is not implemented in IPv6 because it was not used.
3. The timestamp option is not implemented because it was not used.
4. The source route option is called the source route extension header in IPv6.
5. The fragmentation fields in the base header section of IPv4 have moved to the fragmentation extension header in IPv6.
6. The authentication extension header is new in IPv6.
7. The encrypted security payload extension header is new in IPv6.

Transition from IPv4 to IPv6

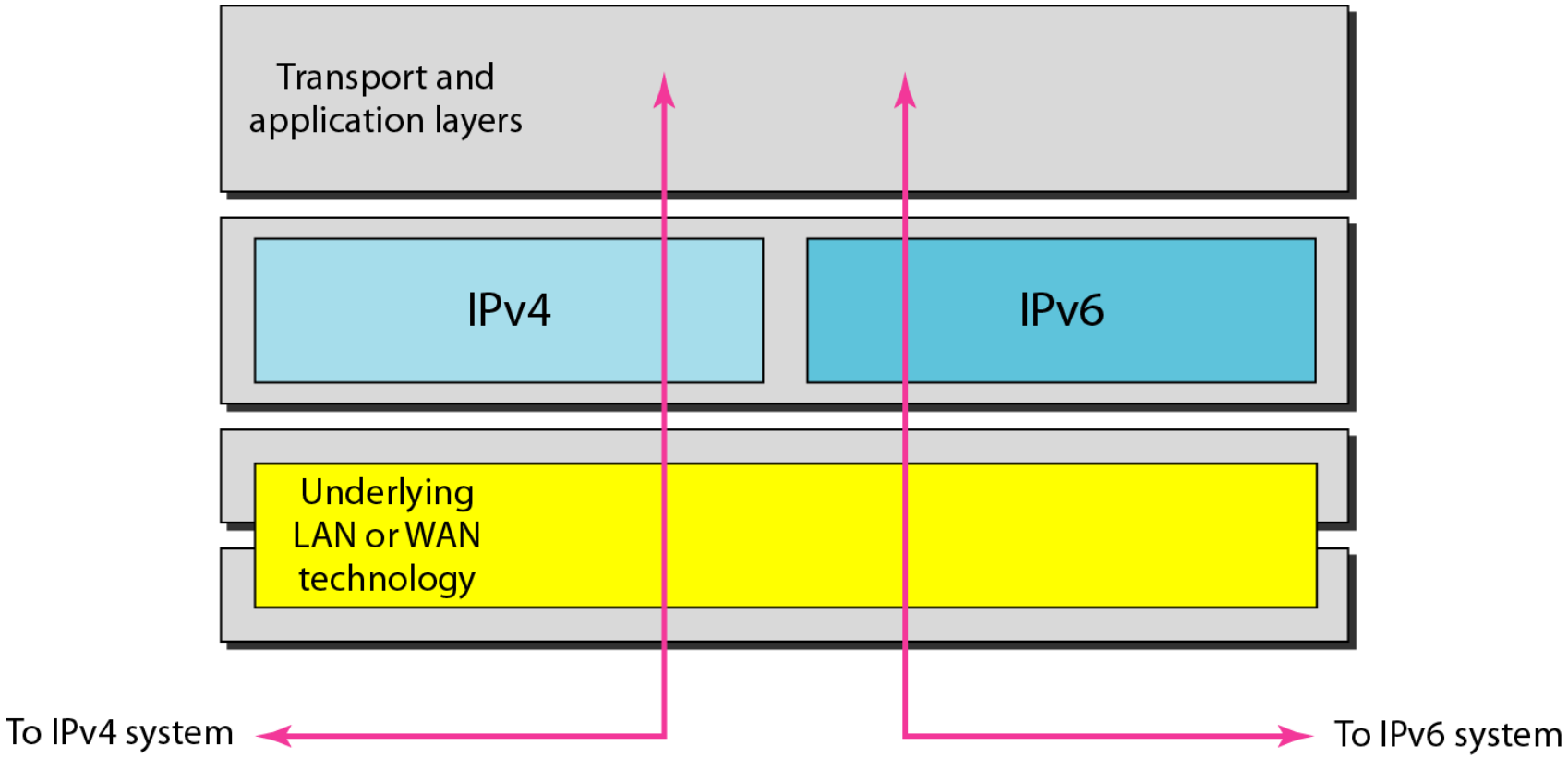
- ***Because of the huge number of systems on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.***
- ***It takes a considerable amount of time before every system in the Internet can move from IPv4 to IPv6.***
- ***The transition must be smooth to prevent any problems between IPv4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

  - **Dual Stack**
  - **Tunneling**
  - **Header Translation**

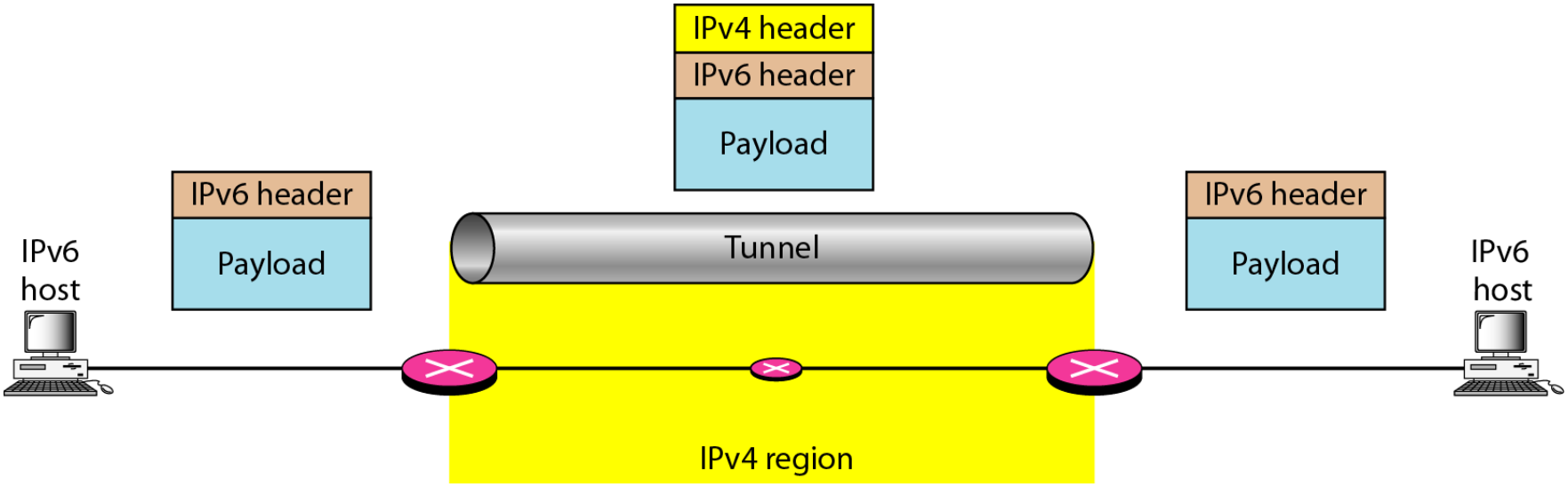
Three transition strategies



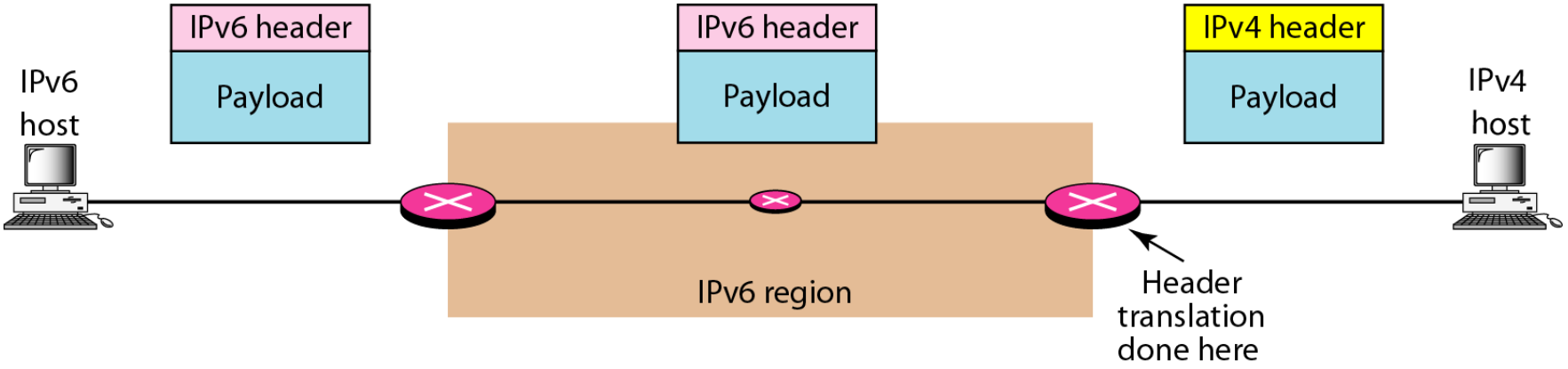
Dual stack


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

Tunneling strategy



Header translation strategy



Header translation

Header Translation Procedure
1. The IPv6 mapped address is changed to an IPv4 address by extracting the rightmost 32 bits.
2. The value of the IPv6 priority field is discarded.
3. The type of service field in IPv4 is set to zero.
4. The checksum for IPv4 is calculated and inserted in the corresponding field.
5. The IPv6 flow label is ignored.
6. Compatible extension headers are converted to options and inserted in the IPv4 header. Some may have to be dropped.
7. The length of IPv4 header is calculated and inserted into the corresponding field.
8. The total length of the IPv4 packet is calculated and inserted in the corresponding field.