Computer Networks

By

Lt Col Ishtiaq Kiani

(10 Sep 12 to 12 Jan 13)

PART 4

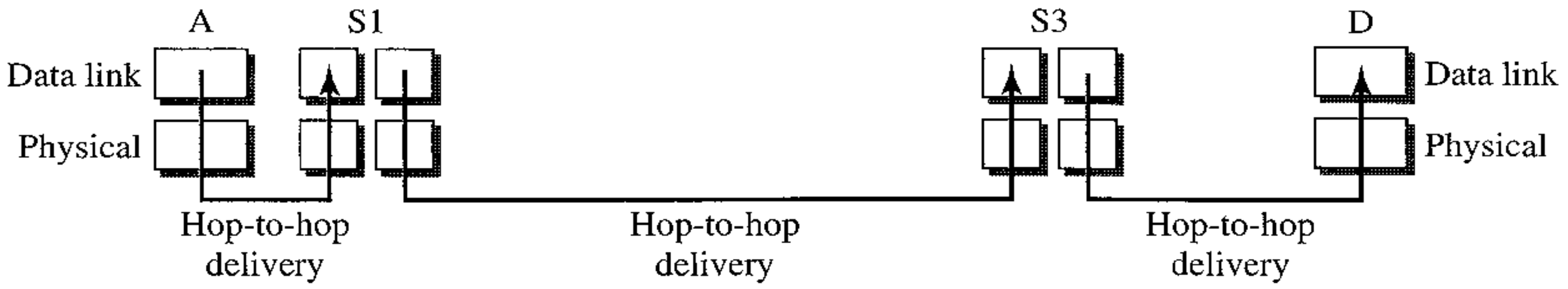
Network Layer

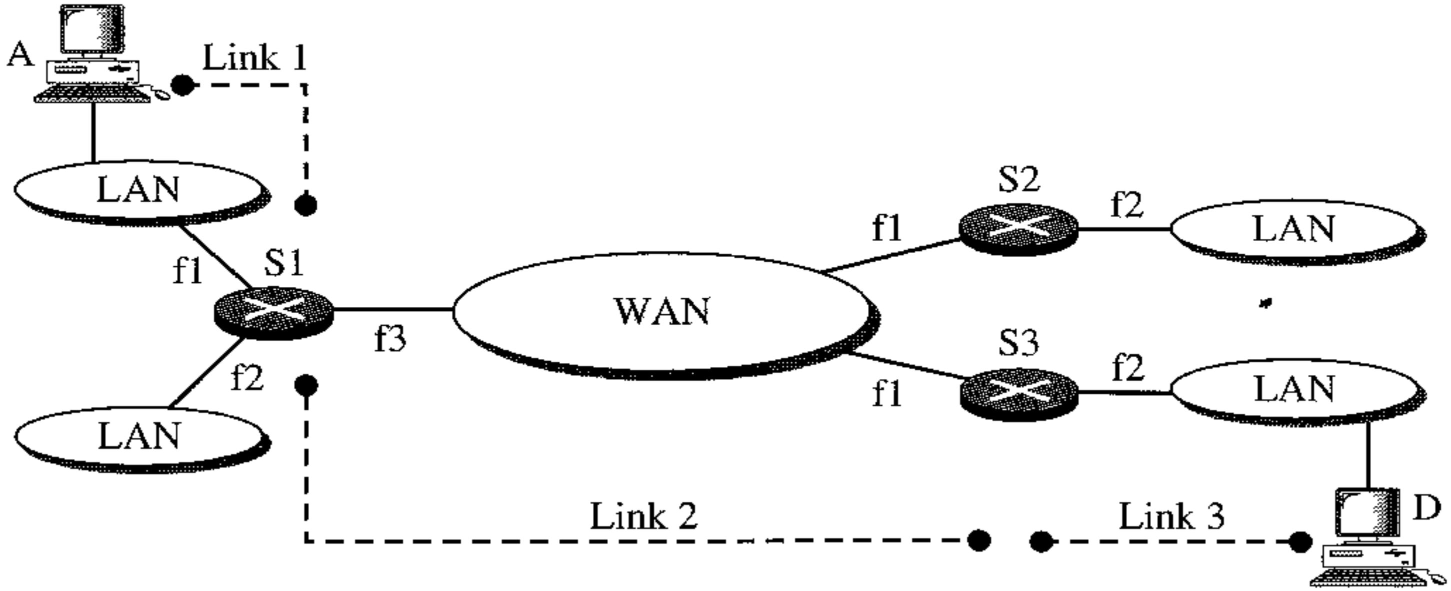
Chapter 20
Network Layer – Internet Protocol

- **Internetworking**
- **IPv4**
- **IPv6**
- **Transition from IPv4 to IPv6**

Internetworking

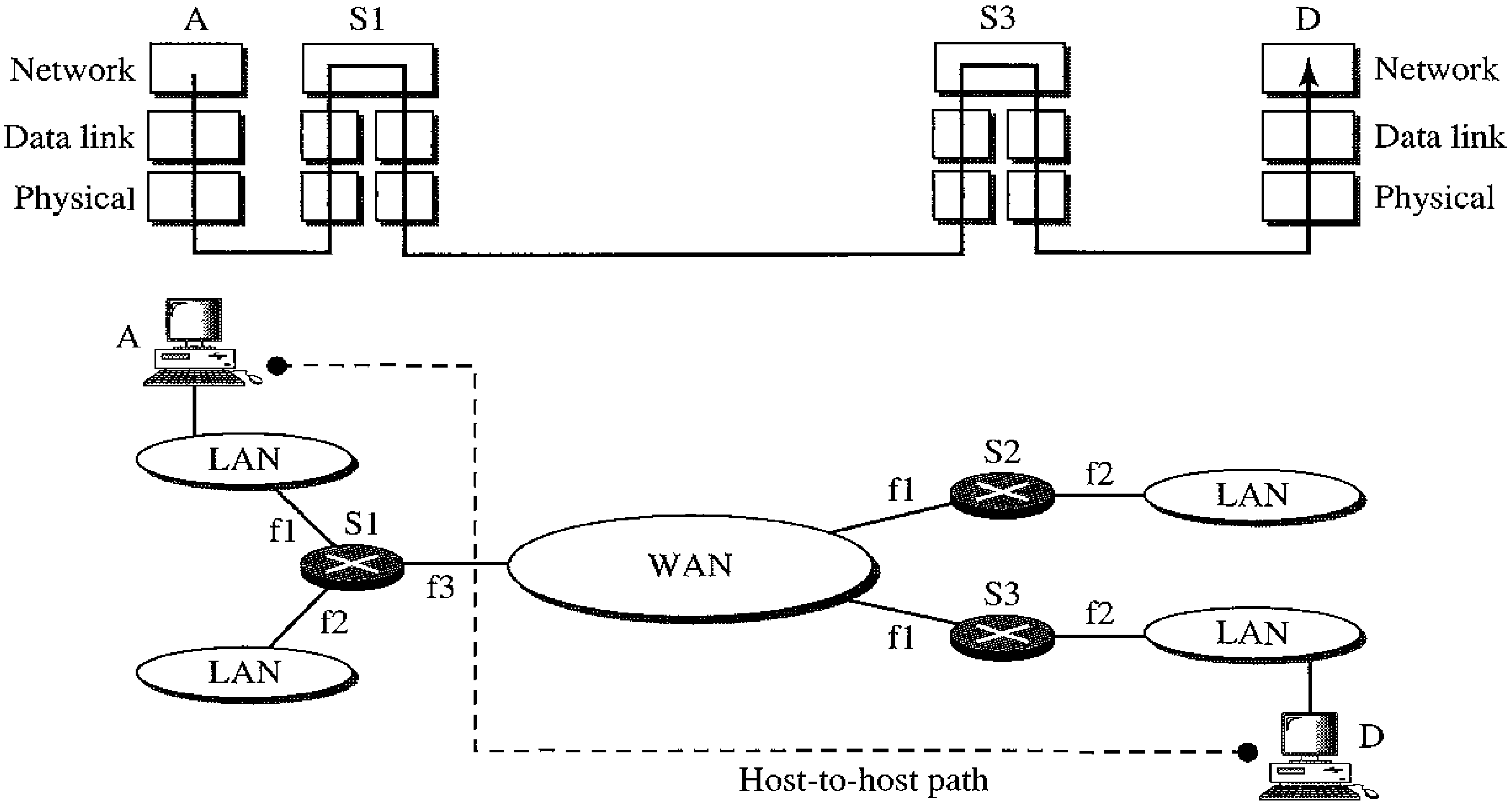
Need for Network Layer





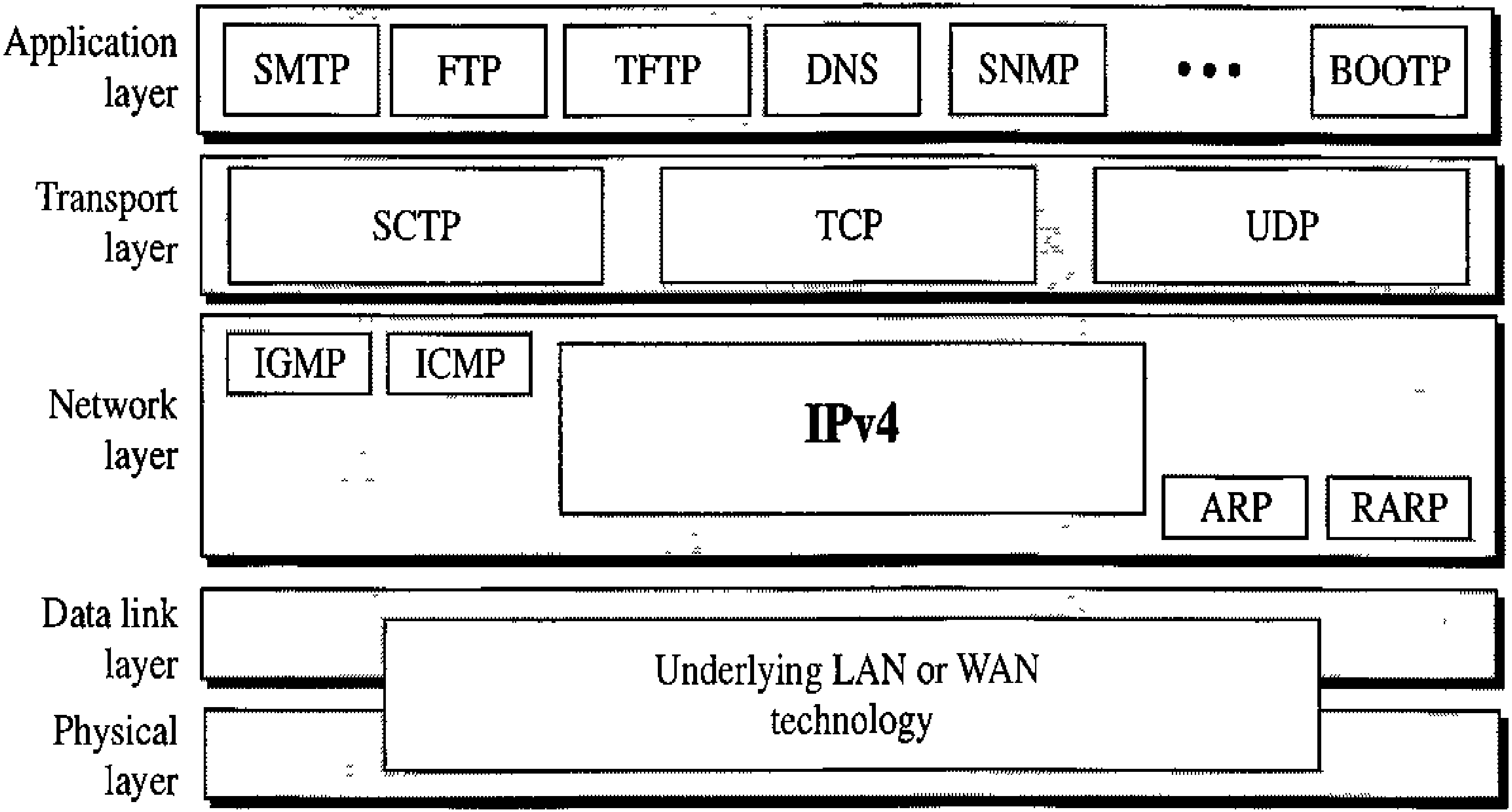
Internetworking

Need for Network Layer



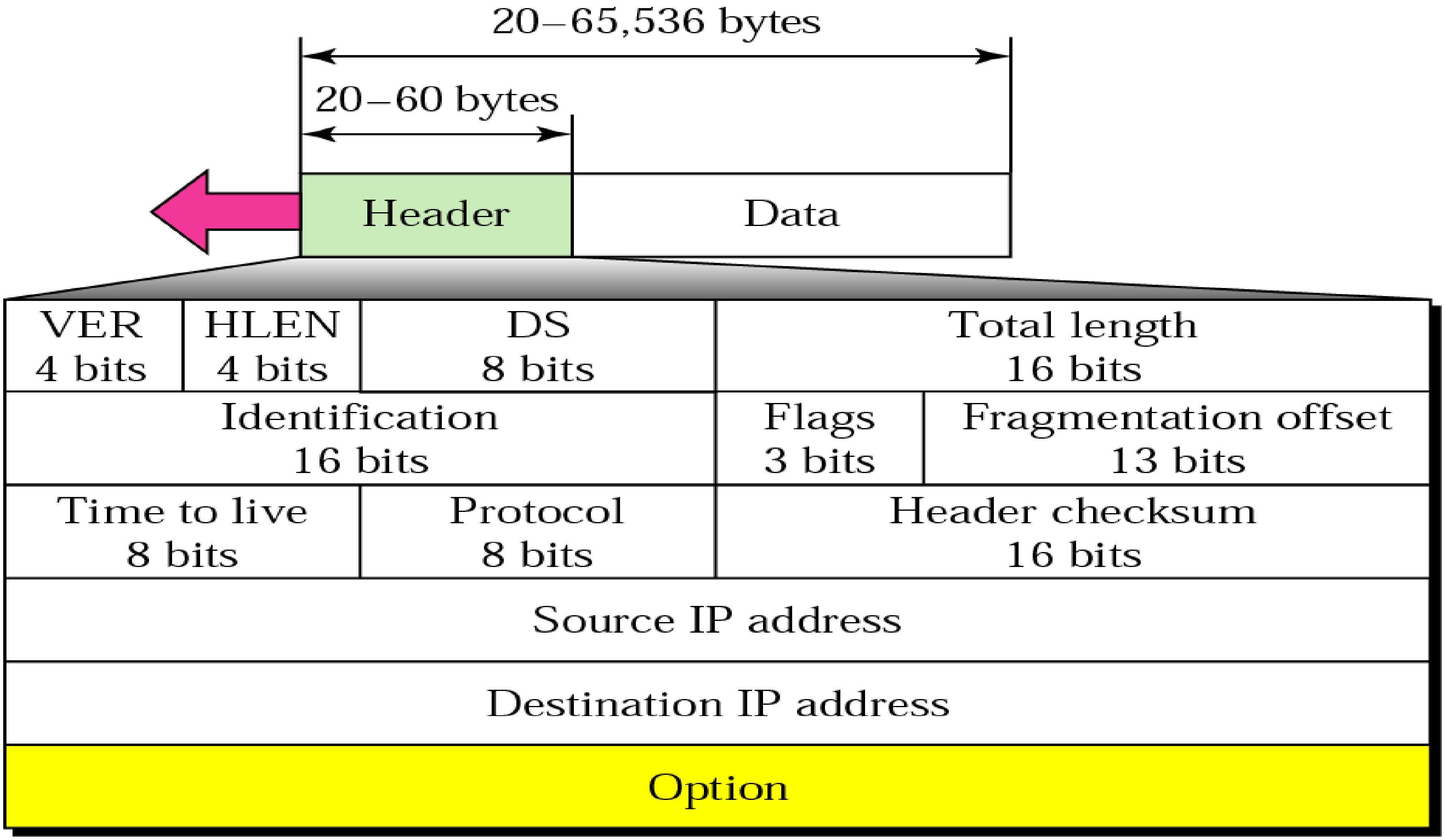
IPv4

Position of IPv4



IPv4

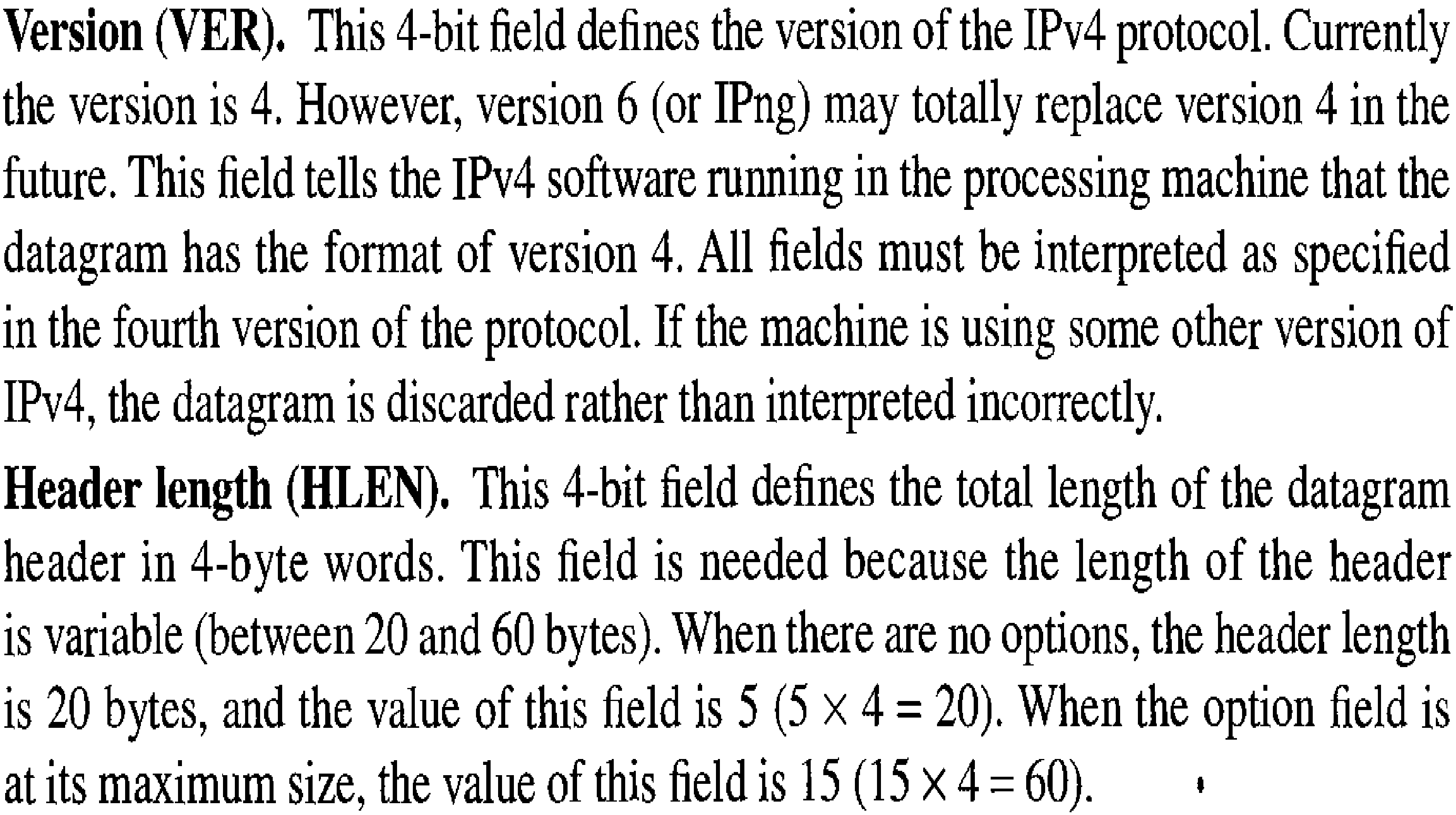
IPv4 Datagram



<table>
<tr><td colspan="4">VER
4 bits</td><td colspan="4">HLEN
4 bits</td><td colspan="8">DS
8 bits</td><td colspan="16">Total length
16 bits</td></tr>
<tr><td colspan="16">Identification
16 bits</td><td colspan="3">Flags
3 bits</td><td colspan="13">Fragmentation offset
13 bits</td></tr>
<tr><td colspan="8">Time to live
8 bits</td><td colspan="8">Protocol
8 bits</td><td colspan="16">Header checksum
16 bits</td></tr>
<tr><td colspan="32">Source IP address</td></tr>
<tr><td colspan="32">Destination IP address</td></tr>
<tr><td colspan="32">Option</td></tr>
</table>

IPv4

IPv4 Datagram – VER and HLEN

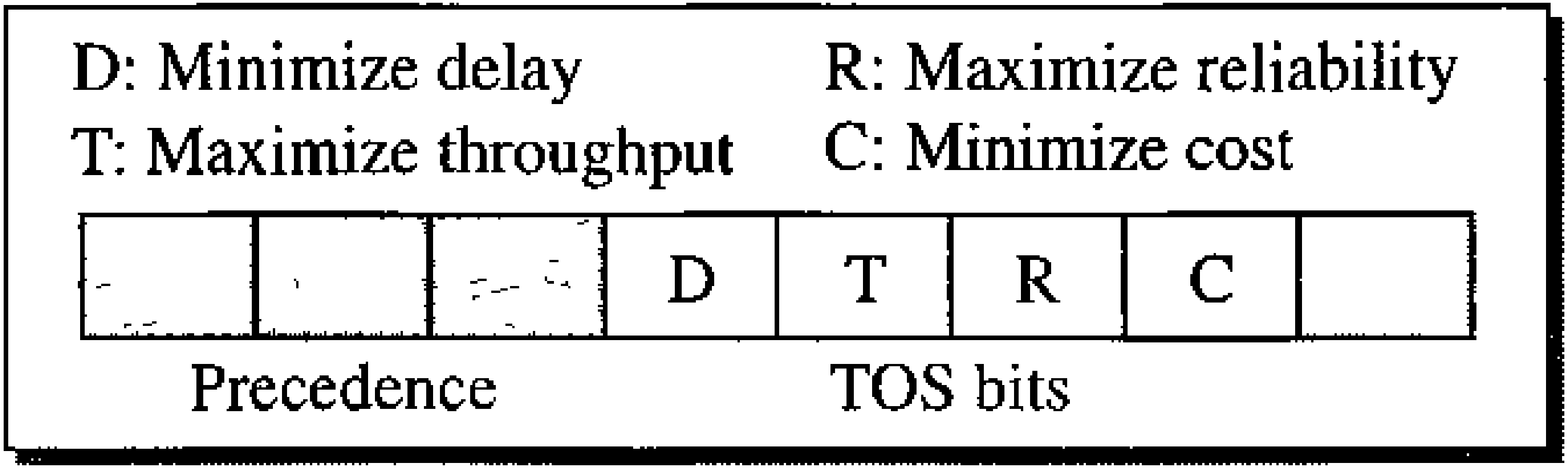


Version (VER). This 4-bit field defines the version of the IPv4 protocol. Currently the version is 4. However, version 6 (or IPng) may totally replace version 4 in the future. This field tells the IPv4 software running in the processing machine that the datagram has the format of version 4. All fields must be interpreted as specified in the fourth version of the protocol. If the machine is using some other version of IPv4, the datagram is discarded rather than interpreted incorrectly.

Header length (HLEN). This 4-bit field defines the total length of the datagram header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes). When there are no options, the header length is 20 bytes, and the value of this field is 5 (5 × 4 = 20). When the option field is at its maximum size, the value of this field is 15 (15 × 4 = 60).

IPv4

IPv4 Datagram – DS Field



D: Minimize delay

T: Maximize throughput

R: Maximize reliability

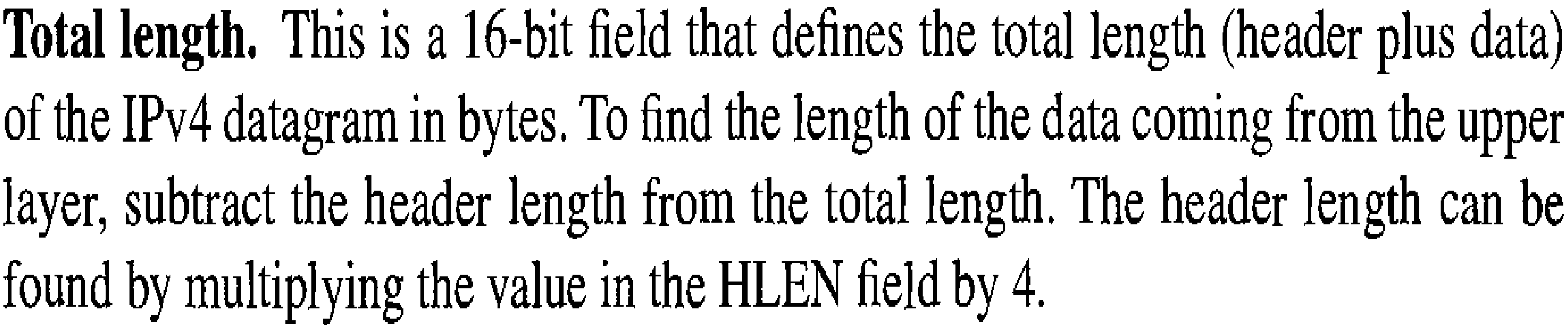
C: Minimize cost

Service type

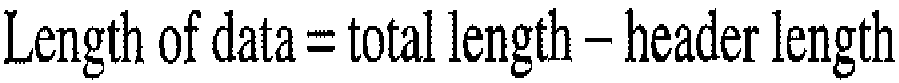
In this interpretation, the first 3 bits are called precedence bits. The next 4 bits are called **type of service (TOS)** bits, and the last bit is not used.

IPv4

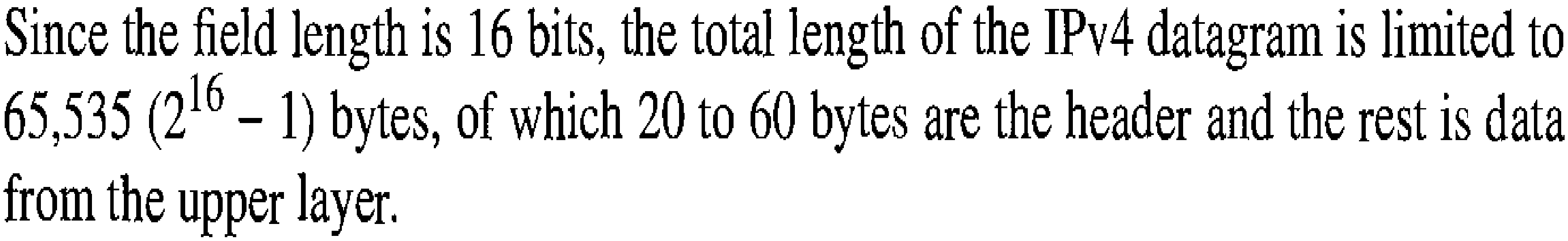
IPv4 Datagram – Total Length



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



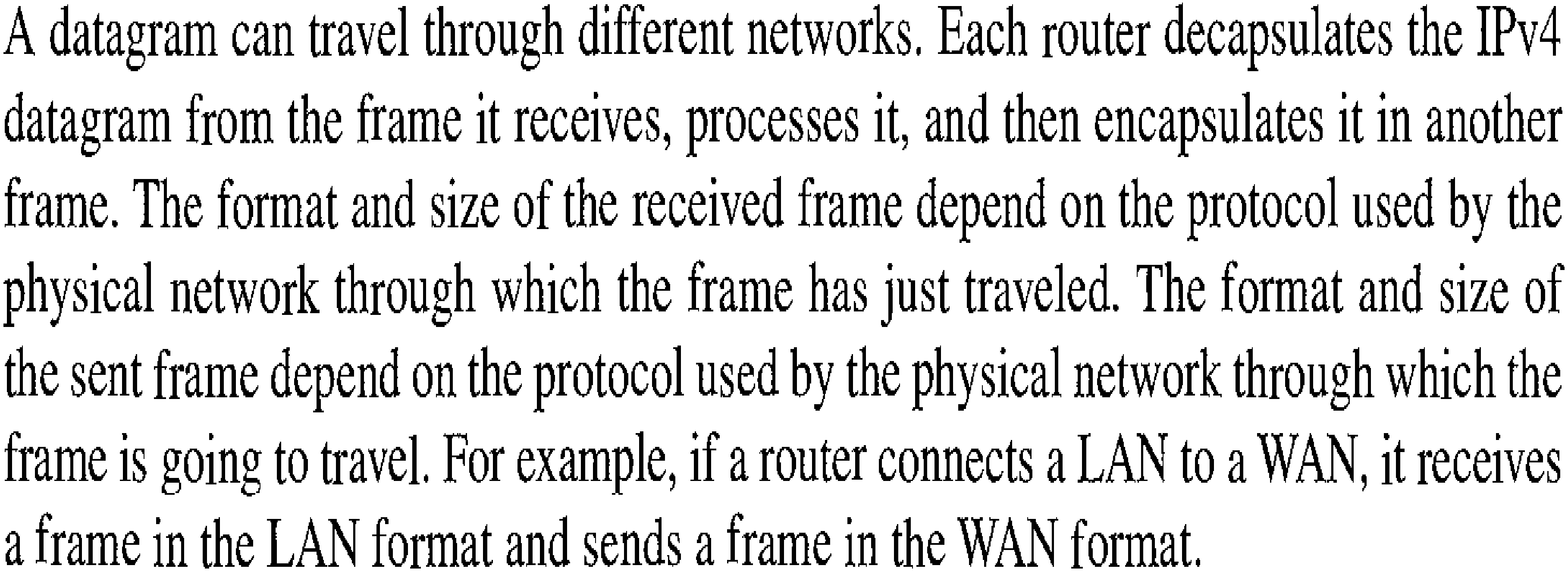
Length of data = total length – header length



Since the field length is 16 bits, the total length of the IPv4 datagram is limited to 65,535 ($2^{16} - 1$) bytes, of which 20 to 60 bytes are the header and the rest is data from the upper layer.

IPv4

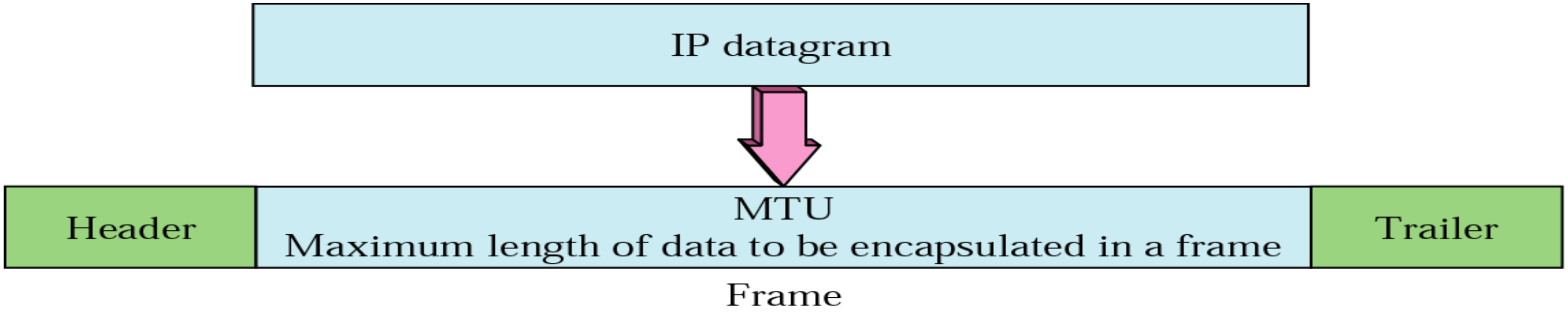
IPv4 Datagram – Fragmentation



A datagram can travel through different networks. Each router decapsulates the IPv4 datagram from the frame it receives, processes it, and then 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 format and size of the sent frame depend on the protocol used by the physical network through which the frame is going to travel. For example, if a router connects a LAN to a WAN, it receives a frame in the LAN format and sends a frame in the WAN format.

IPv4

IPv4 Datagram – Fragmentation



IP datagram

Header

MTU

Maximum length of data to be encapsulated in a frame

Trailer

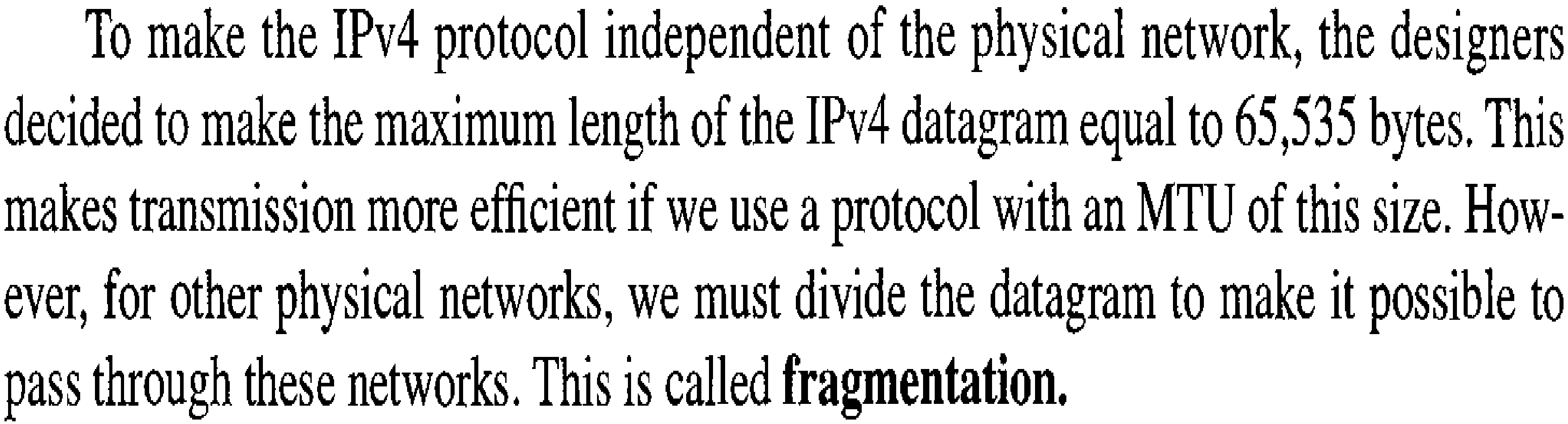
Frame

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

Value of Max Tfr Unit depends on Physical Network Protocols

IPv4

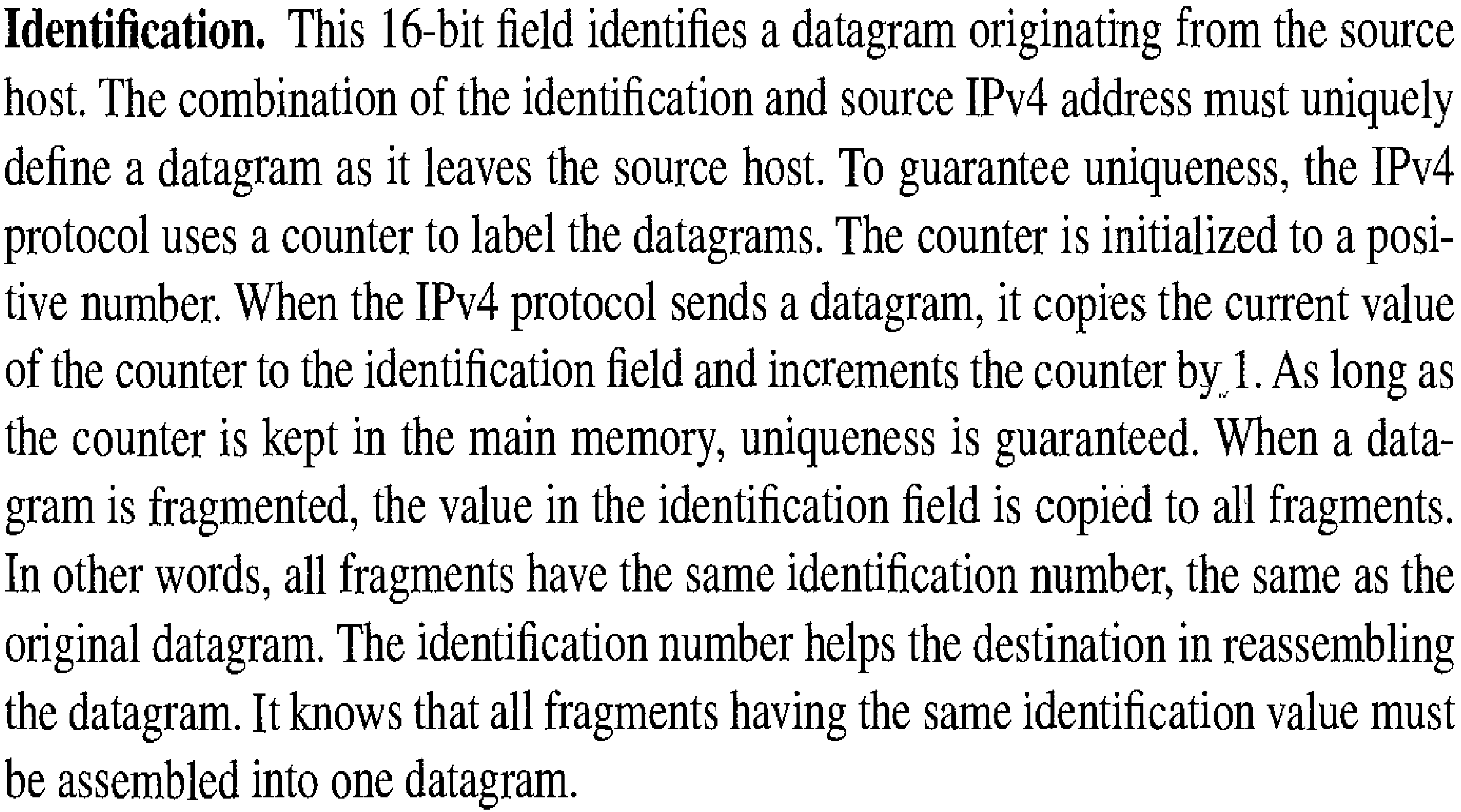
IPv4 Datagram – Fragmentation



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.**

IPv4

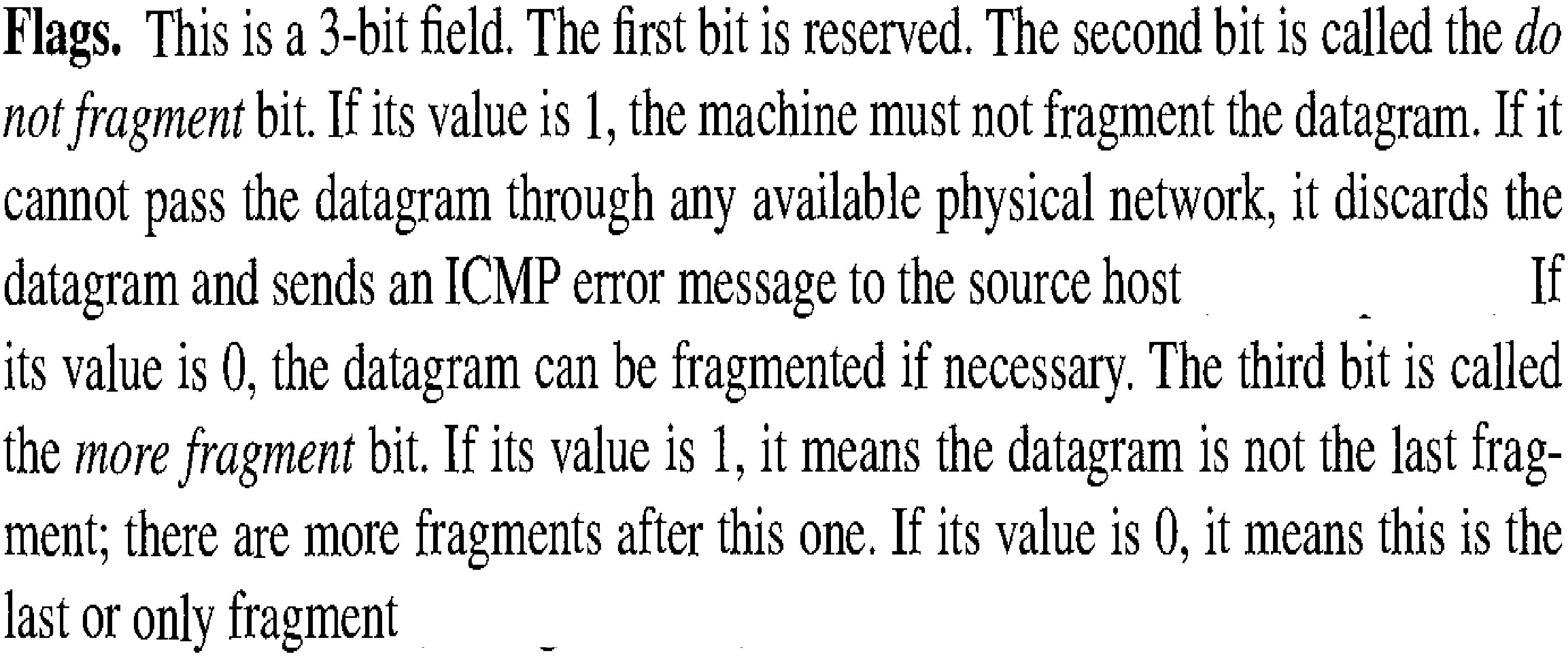
IPv4 Datagram – Identification



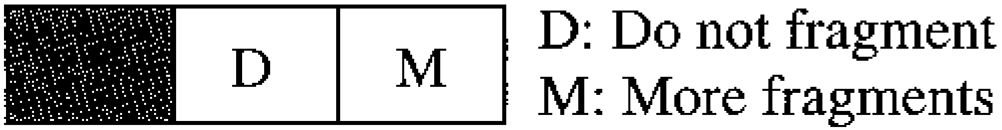
Identification. This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IPv4 address must uniquely define a datagram as it leaves the source host. To guarantee uniqueness, the IPv4 protocol uses a counter to label the datagrams. The counter 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. As long as the counter is kept in the main memory, uniqueness is guaranteed. When a datagram is fragmented, the value in the identification field is copied to all fragments. In other words, all fragments have the same identification number, the same as the original datagram. The identification number helps the destination in reassembling the datagram. It knows that all fragments having the same identification value must be assembled into one datagram.

IPv4

IPv4 Datagram – Flags



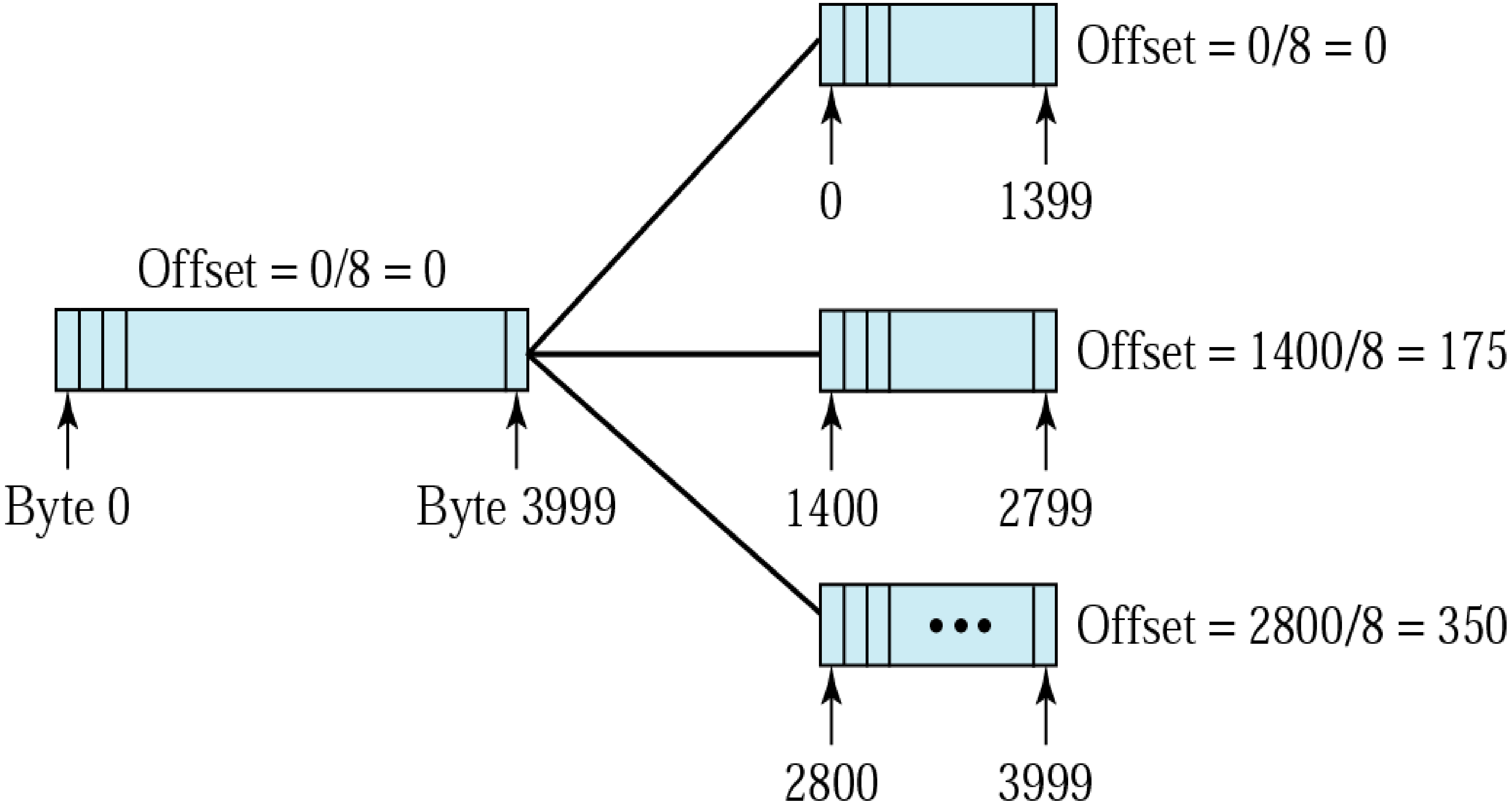
Flags. This is a 3-bit field. The first bit is reserved. The second bit is called the *do not fragment* bit. If its value is 1, the machine must not fragment the datagram. If it cannot pass the datagram through any available physical network, it discards the datagram and sends an ICMP error message to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



D: Do not fragment
M: More fragments

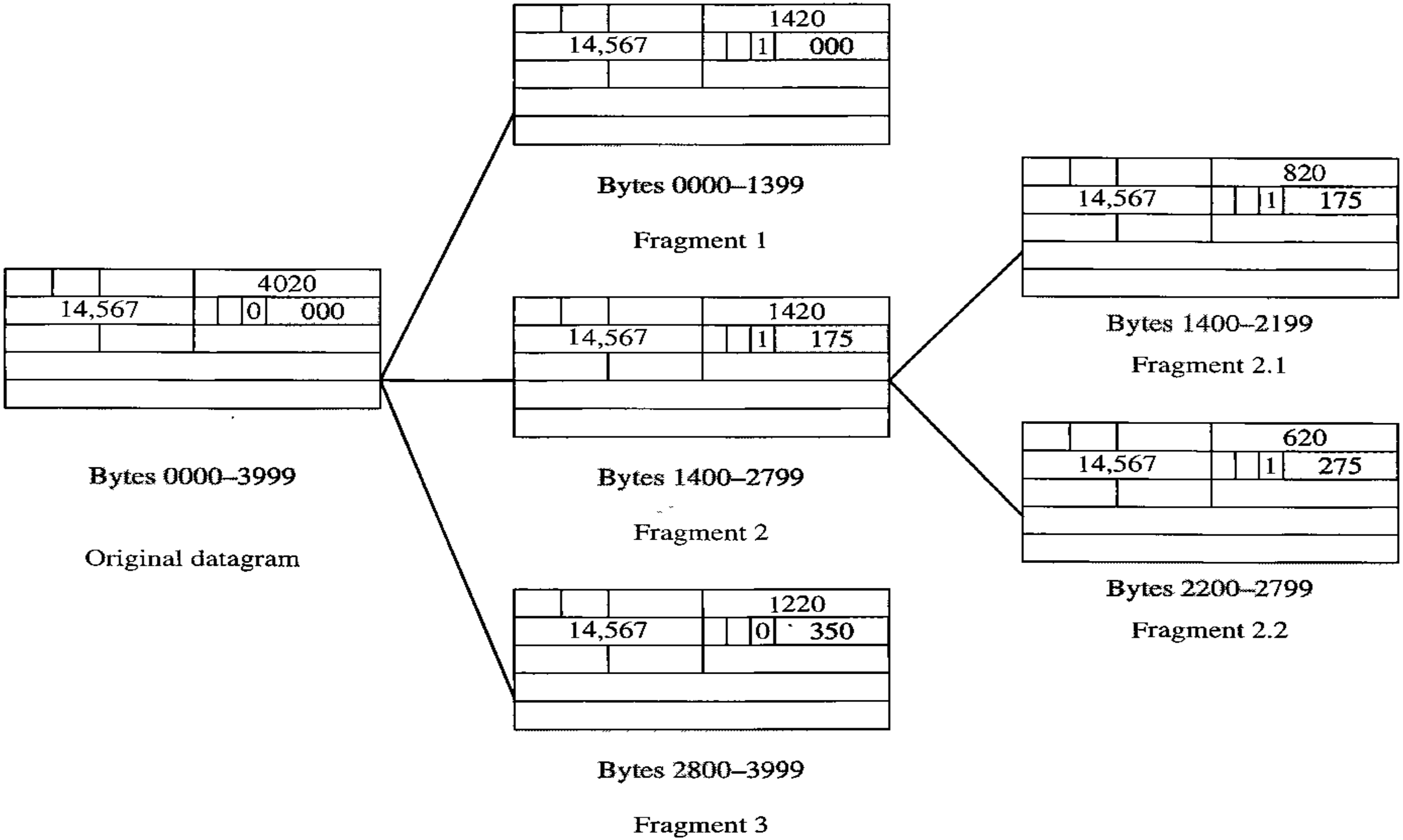
IPv4

IPv4 Datagram – Fragmentation Off set



IPv4

IPv4 Datagram – Fragmentation Off set



Bytes 0000–3999

Original datagram

Bytes 0000–1399

Fragment 1

Bytes 1400–2799

Fragment 2

Bytes 2800–3999

Fragment 3

Bytes 1400–2199

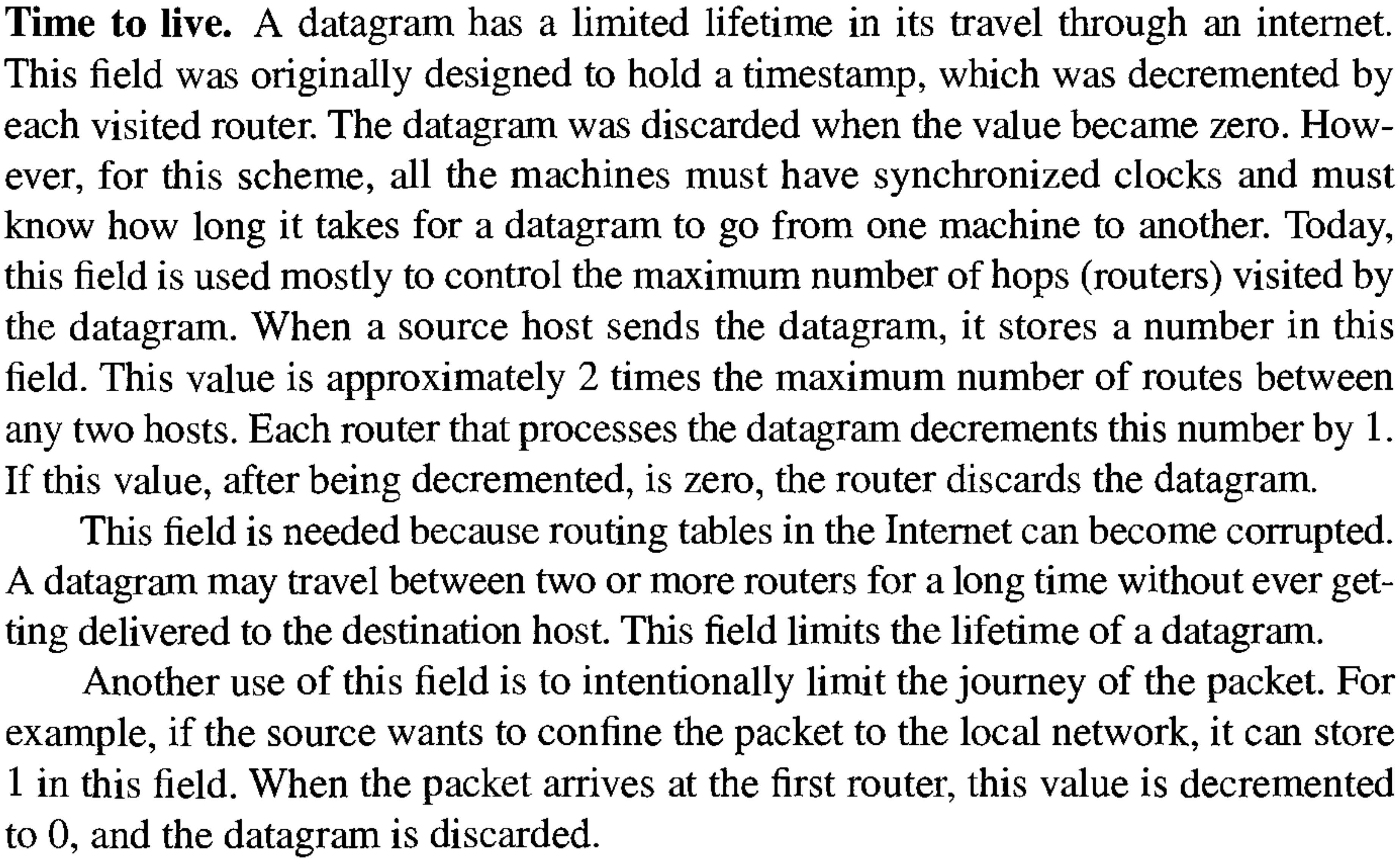
Fragment 2.1

Bytes 2200–2799

Fragment 2.2

IPv4

IPv4 Datagram – Time to Live



Time to live. A datagram has a limited lifetime in its travel through an internet. This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero. However, for this scheme, all the machines must have synchronized clocks and must know how long it takes for a datagram to go from one machine to another. Today, this field is used mostly to control the maximum number of hops (routers) visited by the datagram. When a source host sends the datagram, it stores a number in this field. This value is approximately 2 times the maximum number of routes between any two hosts. Each router that processes the datagram decrements this number by 1. If this value, after being decremented, is zero, the router discards the datagram.

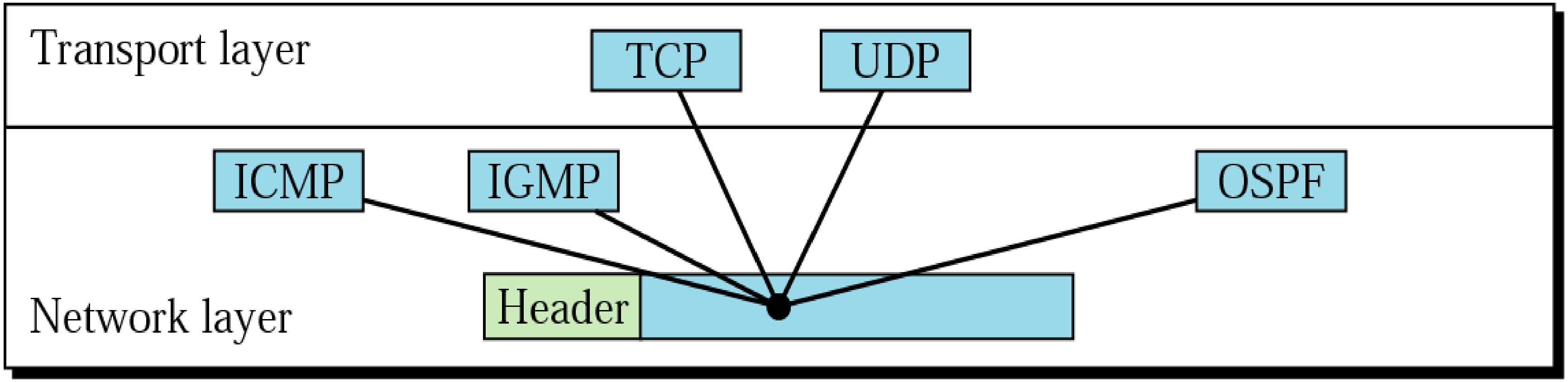
This field is needed because routing tables in the Internet can become corrupted. A datagram may travel between two or more routers for a long time without ever getting delivered to the destination host. This field limits the lifetime of a datagram.

Another use of this field is to intentionally limit the journey of the packet. For example, if the source wants to confine the packet to the local network, it can store 1 in this field. When the packet arrives at the first router, this value is decremented to 0, and the datagram is discarded.

IPv4

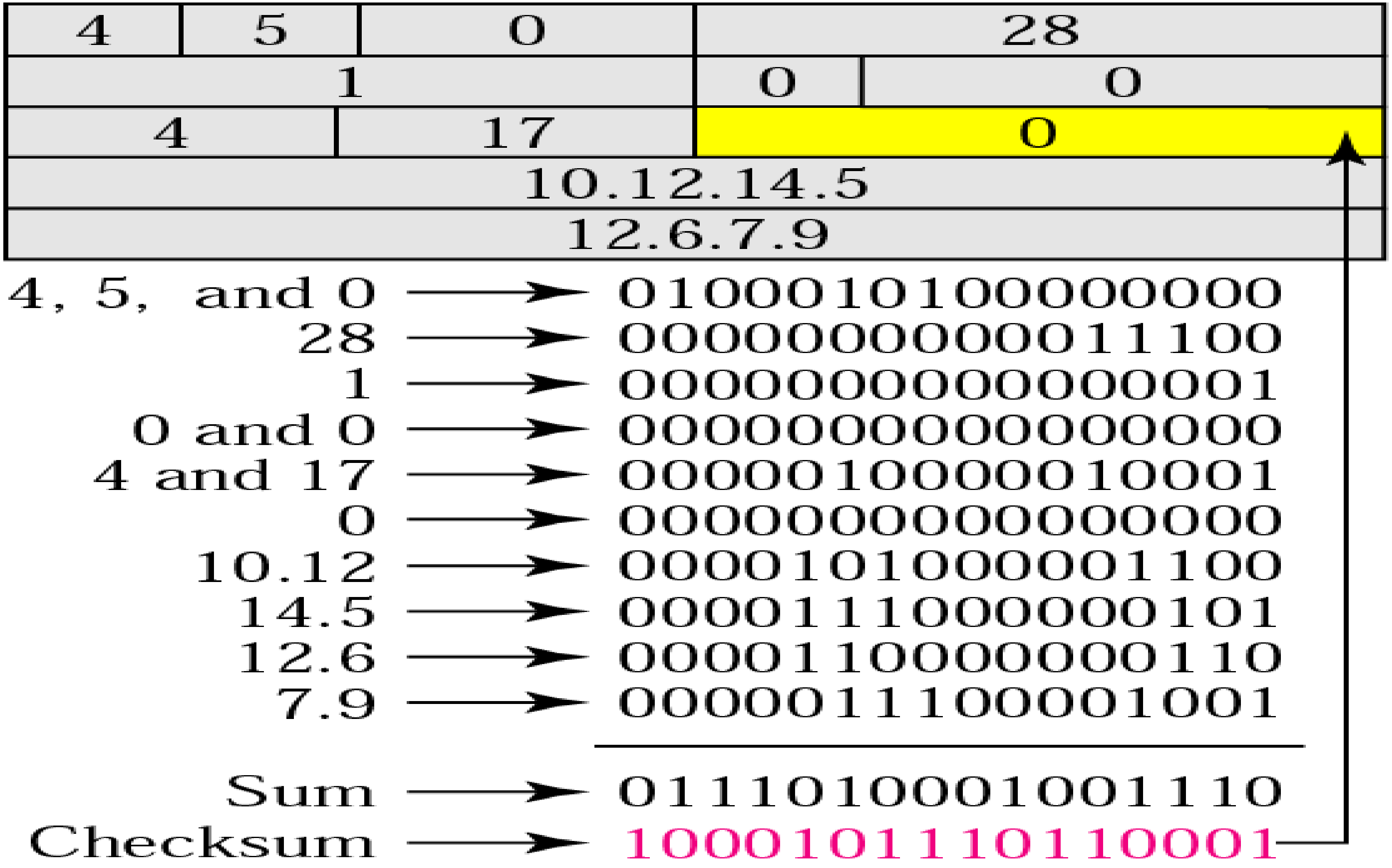
IPv4 Datagram – Protocol

Protocol. This 8-bit field defines the higher-level protocol that uses the services of the IPv4 layer. An IPv4 datagram can encapsulate data from several higher-level protocols such as TCP, UDP, ICMP, and IGMP. This field specifies the final destination protocol to which the IPv4 datagram is delivered. In other words, since the IPv4 protocol carries data from different other protocols, the value of this field helps the receiving network layer know to which protocol the data belong (see



IPv4

IPv4 Datagram – Checksum

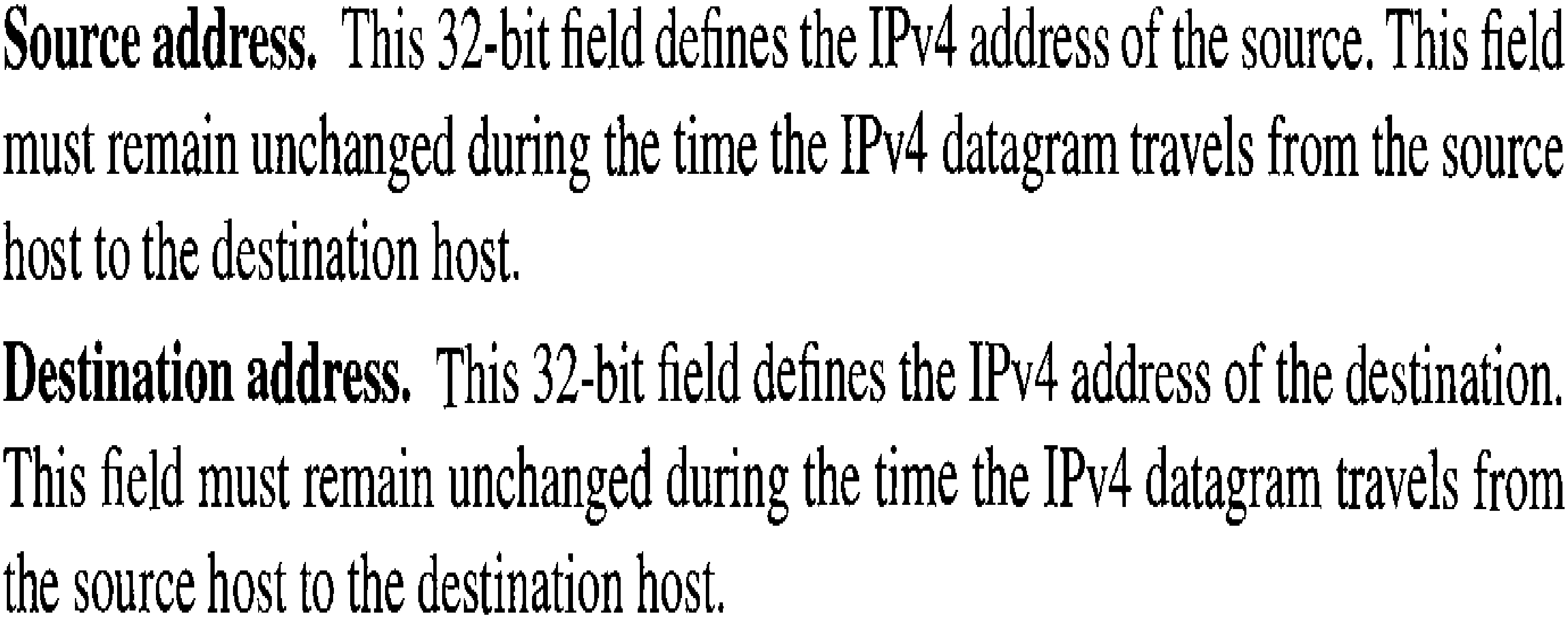


<table>
<tr><td>4</td><td>5</td><td>0</td><td colspan="2">28</td></tr>
<tr><td colspan="3">1</td><td>0</td><td>0</td></tr>
<tr><td colspan="2">4</td><td>17</td><td colspan="2">0</td></tr>
<tr><td colspan="5">10.12.14.5</td></tr>
<tr><td colspan="5">12.6.7.9</td></tr>
</table>

```
4, 5, and 0  ──→  0100010100000000
         28  ──→  0000000000011100
          1  ──→  0000000000000001
    0 and 0  ──→  0000000000000000
   4 and 17  ──→  0000010000010001
          0  ──→  0000000000000000
      10.12  ──→  0000101000001100
       14.5  ──→  0000111000000101
       12.6  ──→  0000110000000110
        7.9  ──→  0000011100001001
                  ────────────────
        Sum  ──→  0111010001001110
   Checksum  ──→  1000101110110001
```

IPv4

IPv4 Datagram – Source and Destination IP Addresses

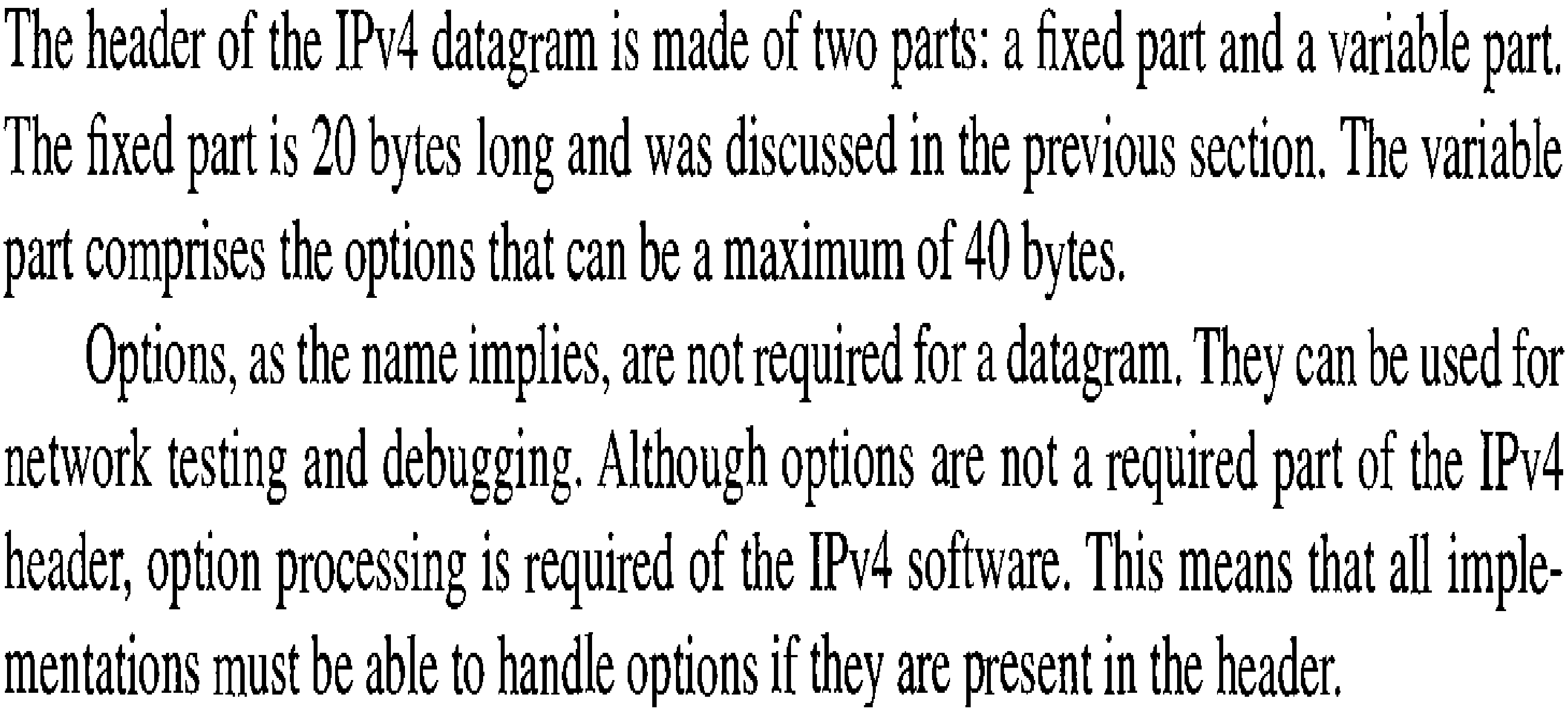


Source address. This 32-bit field defines the IPv4 address of the source. This field must remain unchanged during the time the IPv4 datagram travels from the source host to the destination host.

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

IPv4

IPv4 Datagram – Options

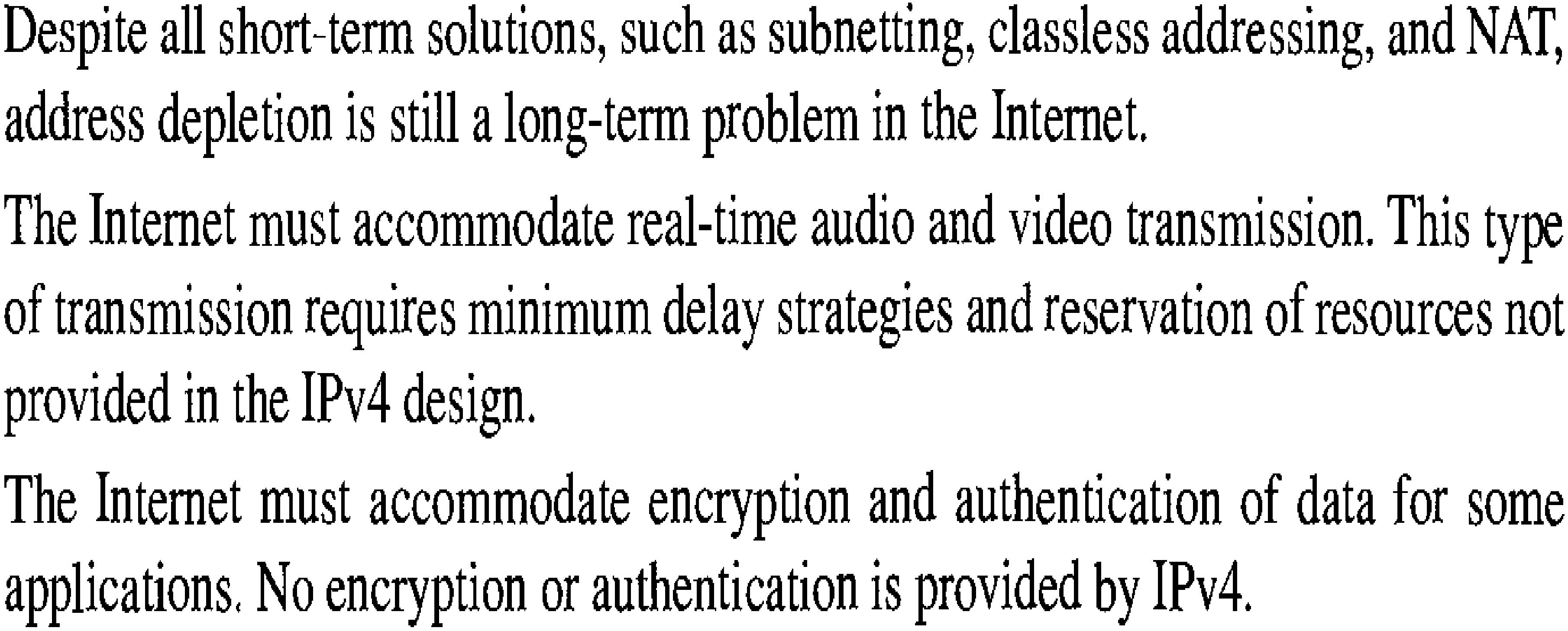


The header of the IPv4 datagram is made of two parts: a fixed part and a variable part. The fixed part is 20 bytes long and was discussed in the previous section. The variable part comprises the options that can be a maximum of 40 bytes.

Options, as the name implies, are not required for a datagram. They can be used for network testing and debugging. Although options are not a required part of the IPv4 header, option processing is required of the IPv4 software. This means that all implementations must be able to handle options if they are present in the header.

IPv6

Deficiencies in IPv4



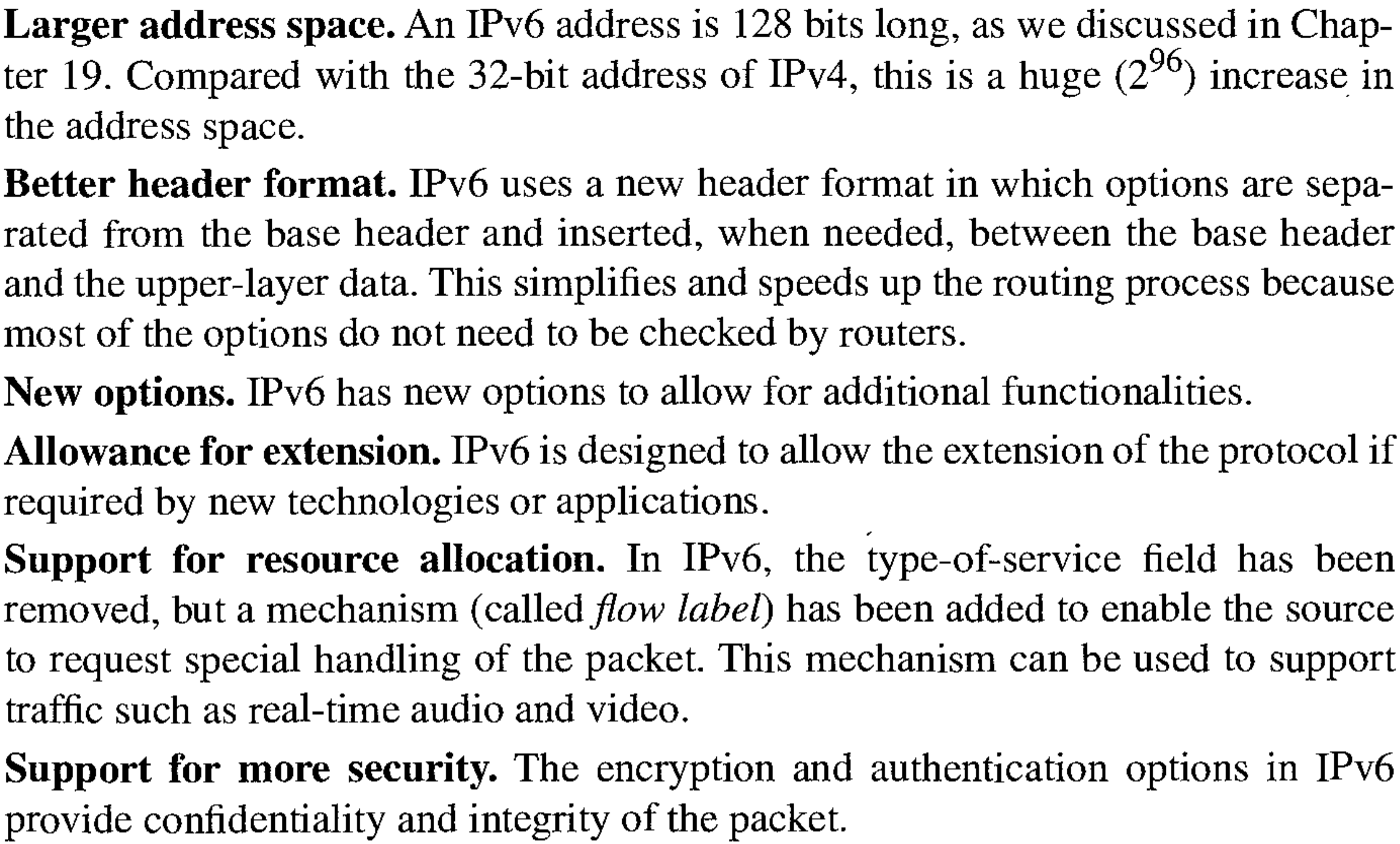
Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.

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

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

IPv6

Advantages



Larger address space. An IPv6 address is 128 bits long, as we discussed in Chapter 19. Compared with the 32-bit address of IPv4, this is a huge (2^{96}) increase in the address space.

Better header format. IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper-layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.

New options. IPv6 has new options to allow for additional functionalities.

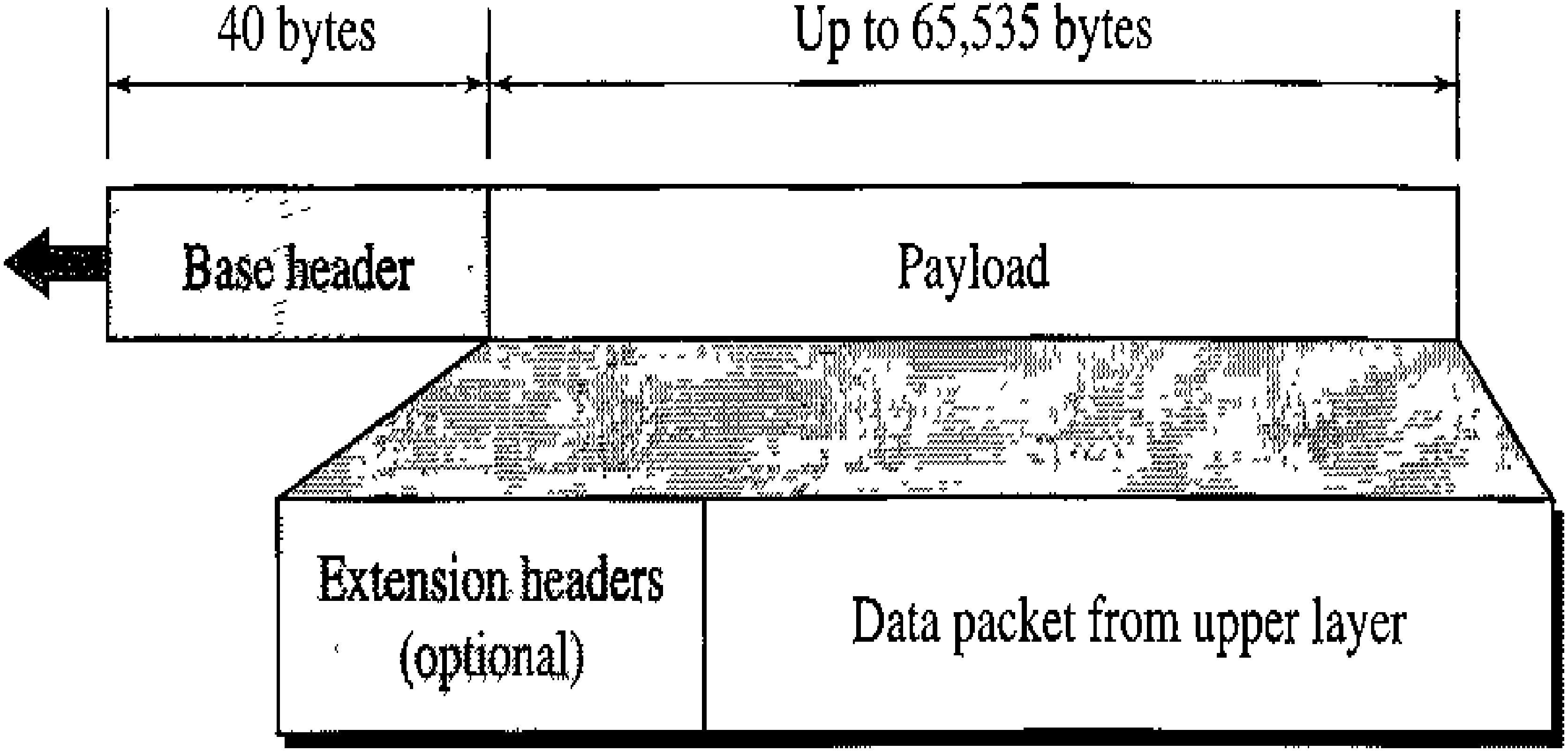
Allowance for extension. IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

Support for resource allocation. In IPv6, the type-of-service field has been removed, but a mechanism (called *flow label*) has been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.

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

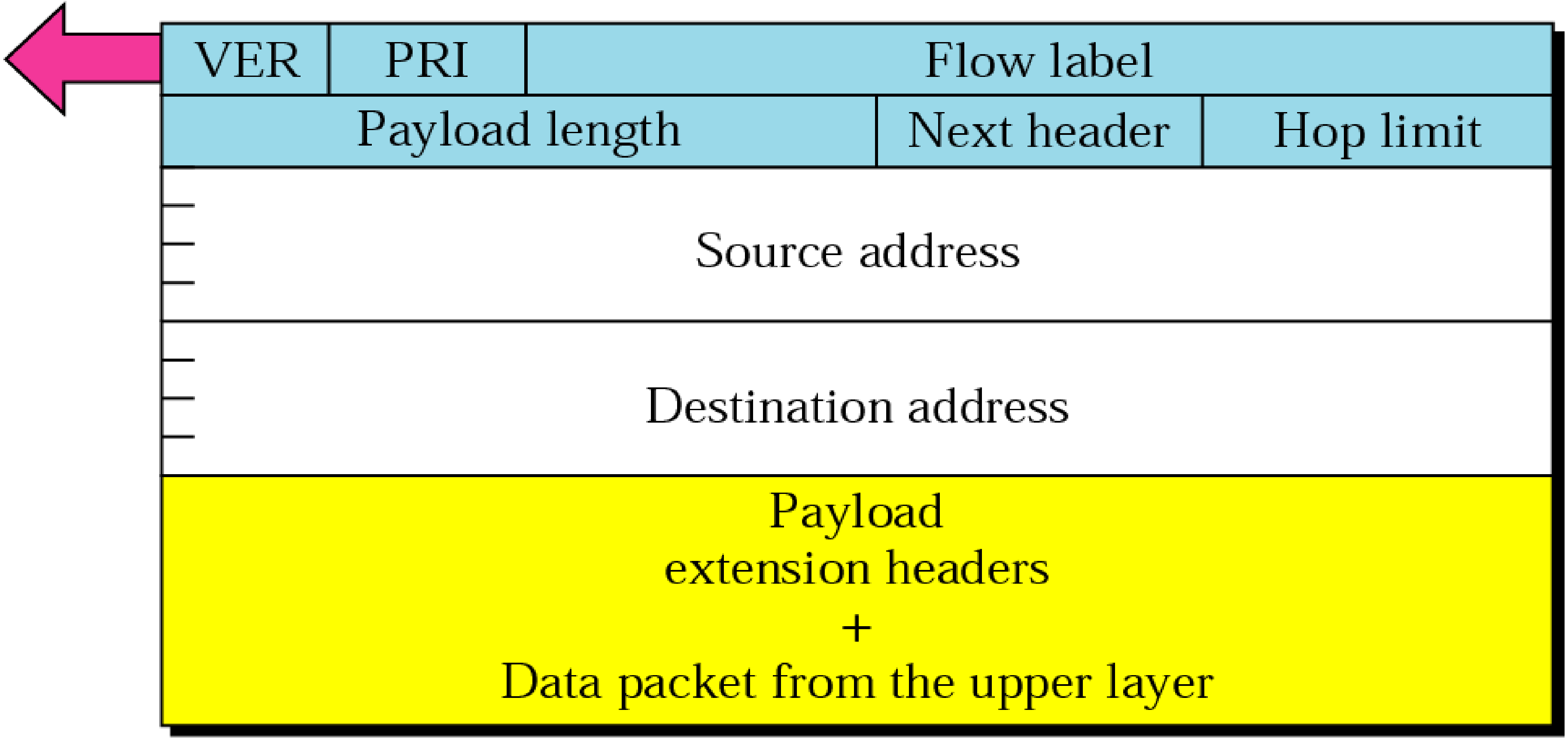
IPv6

Format of IPv6 Datagram



IPv6

Format of IPv6 Datagram



VER

PRI

Flow label

Payload length

Next header

Hop limit

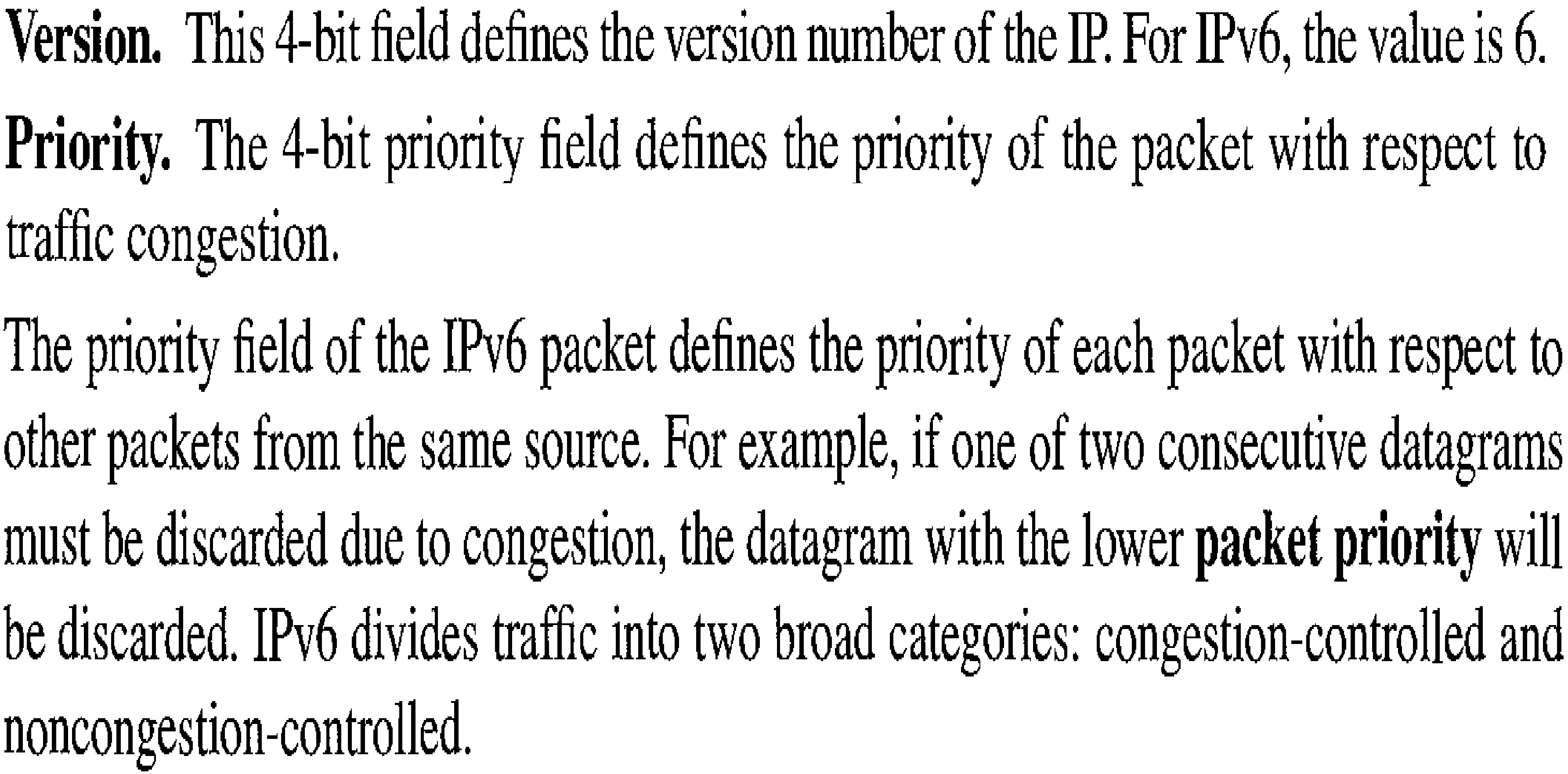
Source address

Destination address

Payload
extension headers
+
Data packet from the upper layer

IPv6

Format of IPv6 Datagram



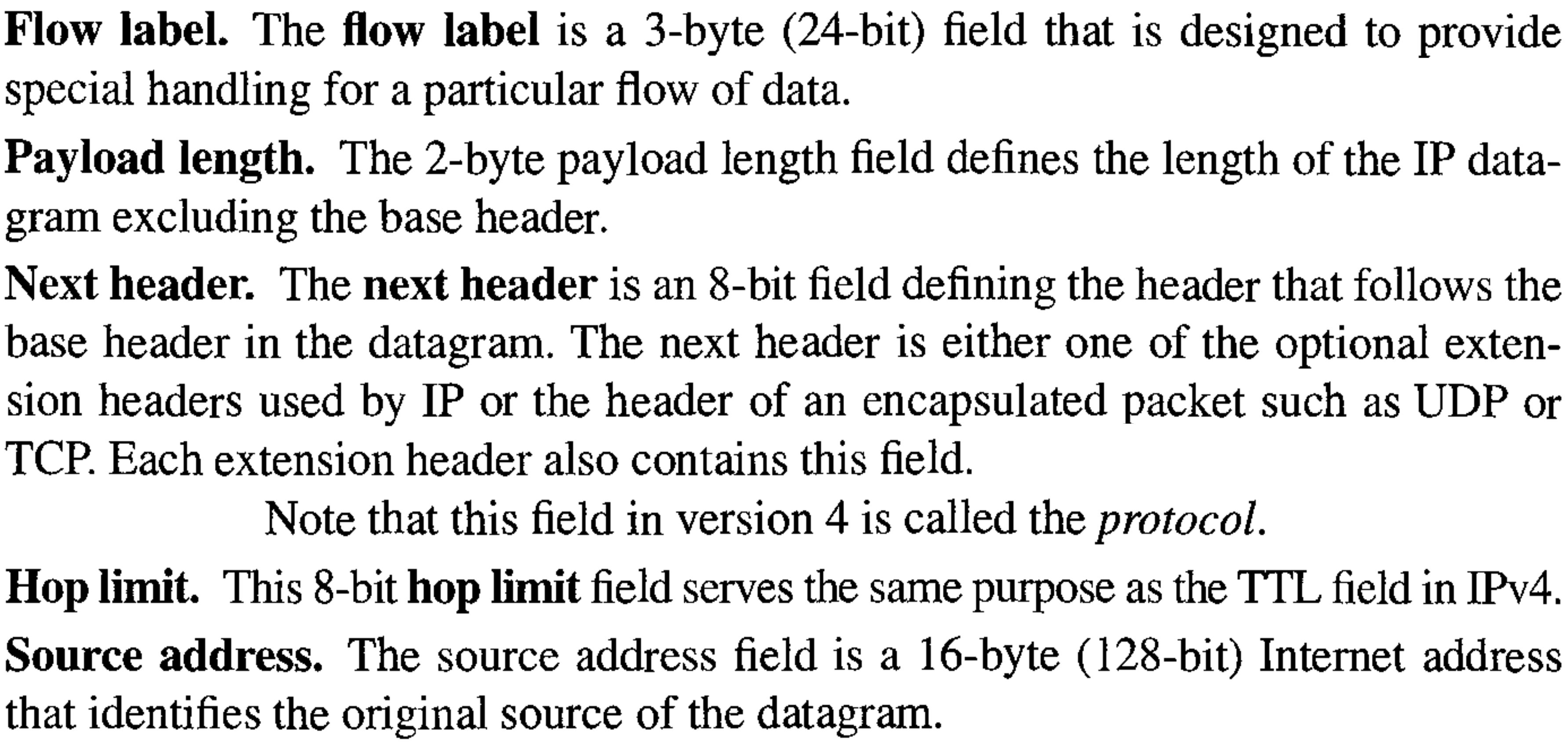
Version. This 4-bit field defines the version number of the IP. For IPv6, the value is 6.

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

The priority field of the IPv6 packet defines the priority of each packet with respect to other packets from the same source. For example, if one of two consecutive datagrams must be discarded due to congestion, the datagram with the lower **packet priority** will be discarded. IPv6 divides traffic into two broad categories: congestion-controlled and noncongestion-controlled.

IPv6

Format of IPv6 Datagram



Flow label. The **flow label** is a 3-byte (24-bit) field that is designed to provide special handling for a particular flow of data.

Payload length. The 2-byte payload length field defines the length of the IP datagram excluding the base header.

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

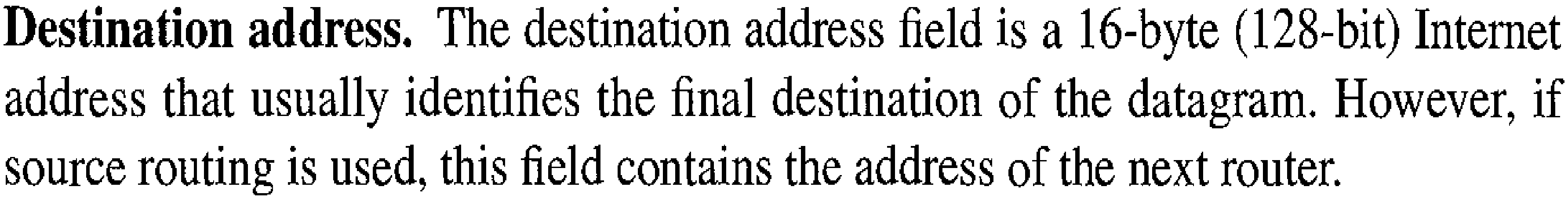
Note that this field in version 4 is called the *protocol*.

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

Source address. The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram.

IPv6

Format of IPv6 Datagram



Destination address. The destination address field is a 16-byte (128-bit) Internet address that usually identifies the final destination of the datagram. However, if source routing is used, this field contains the address of the next router.

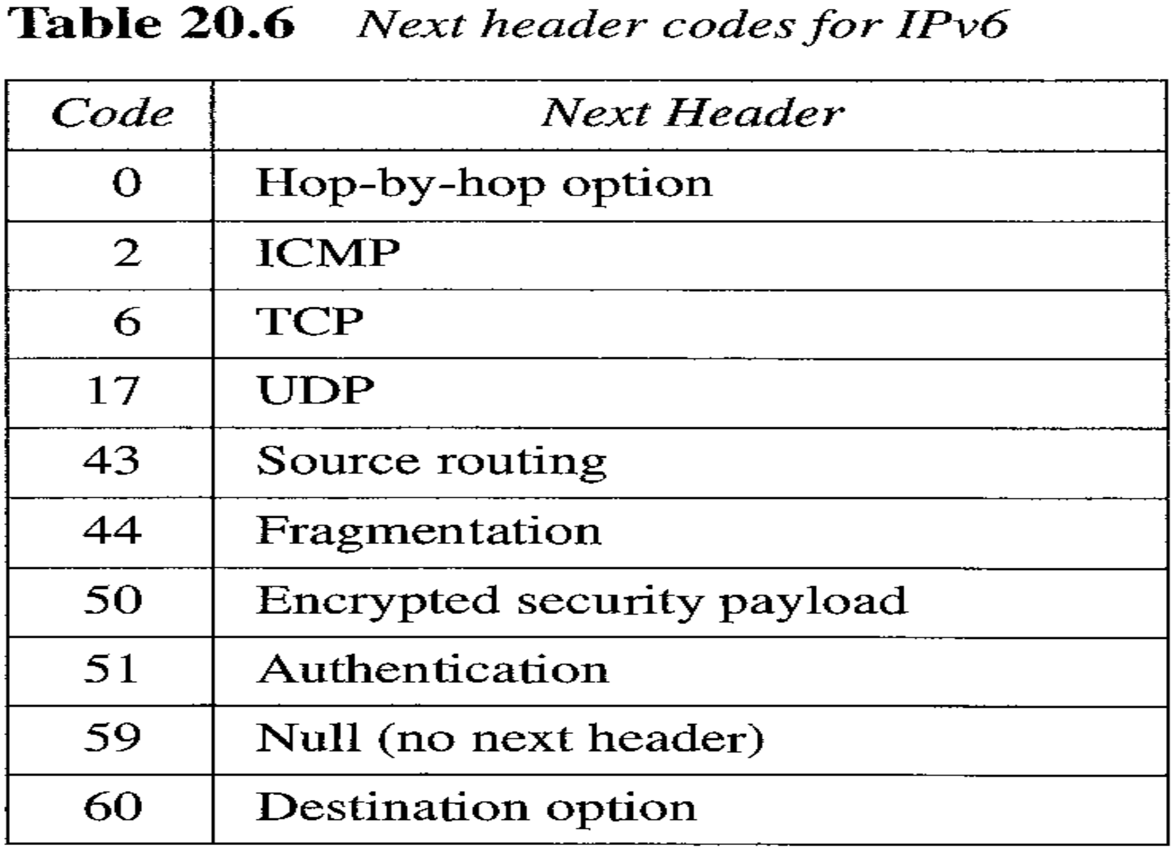
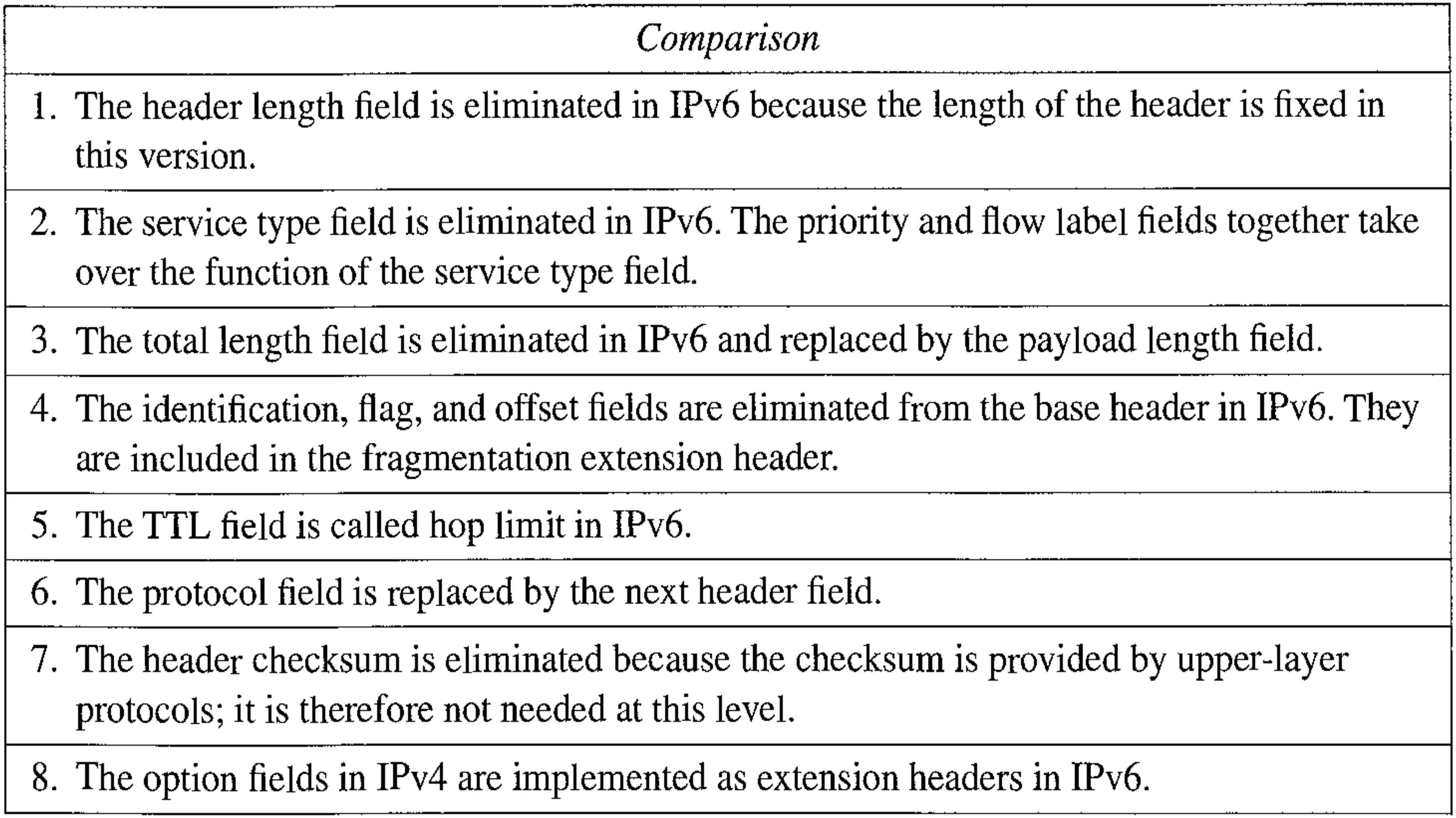


Table 20.6 *Next header codes for IPv6*

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

IPv6

Comparison of IPv4 and IPv6 Header

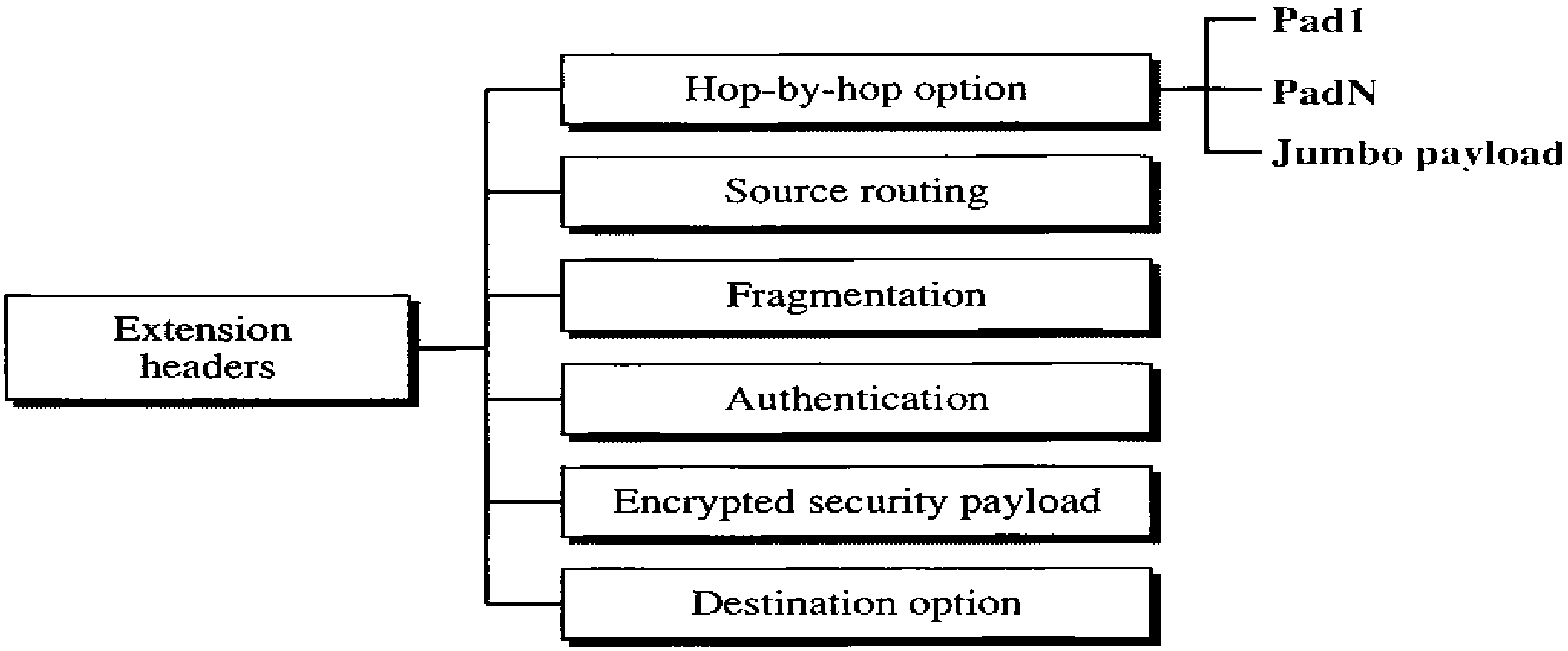


Comparison
1. The header length field is eliminated in IPv6 because the length of the header is fixed in this version.
2. The service type field is eliminated in IPv6. The priority and flow label fields together take over the function of the service type field.
3. The total length field is eliminated 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.

IPv6

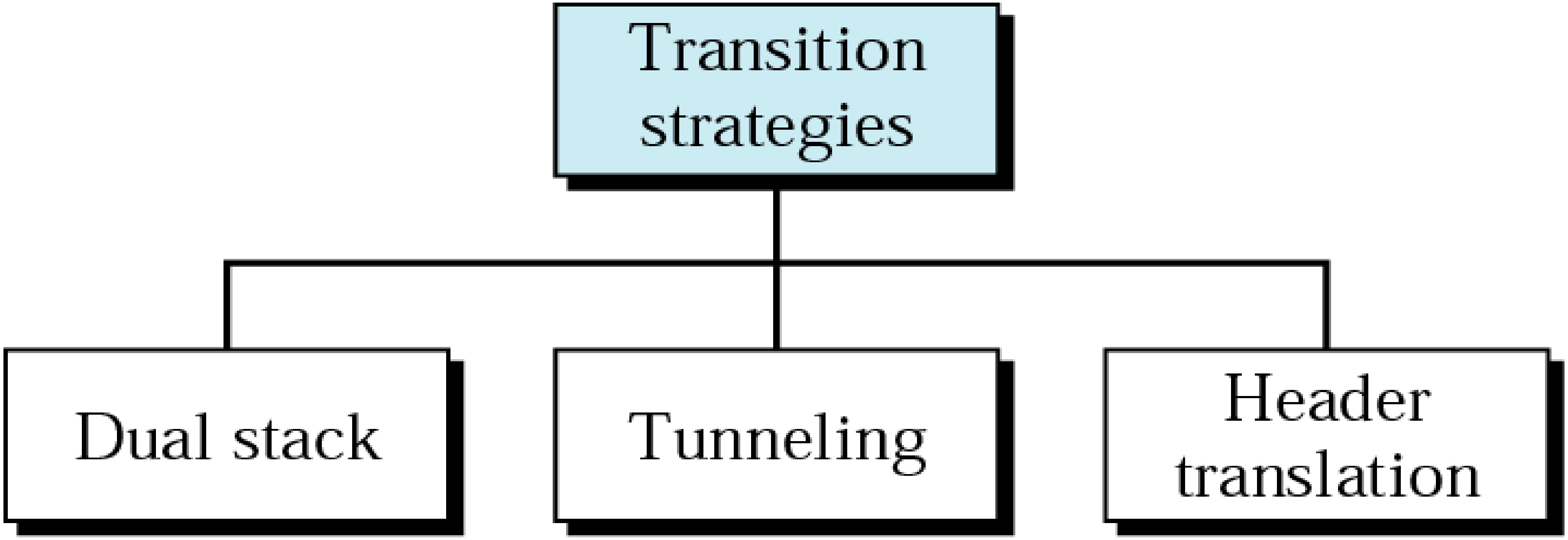
Extension Header

The length of the base header is fixed at 40 bytes. However, to give greater functionality to the IP datagram, the base header can be followed by up to six **extension headers.** Many of these headers are options in IPv4. Six types of extension headers have been defined



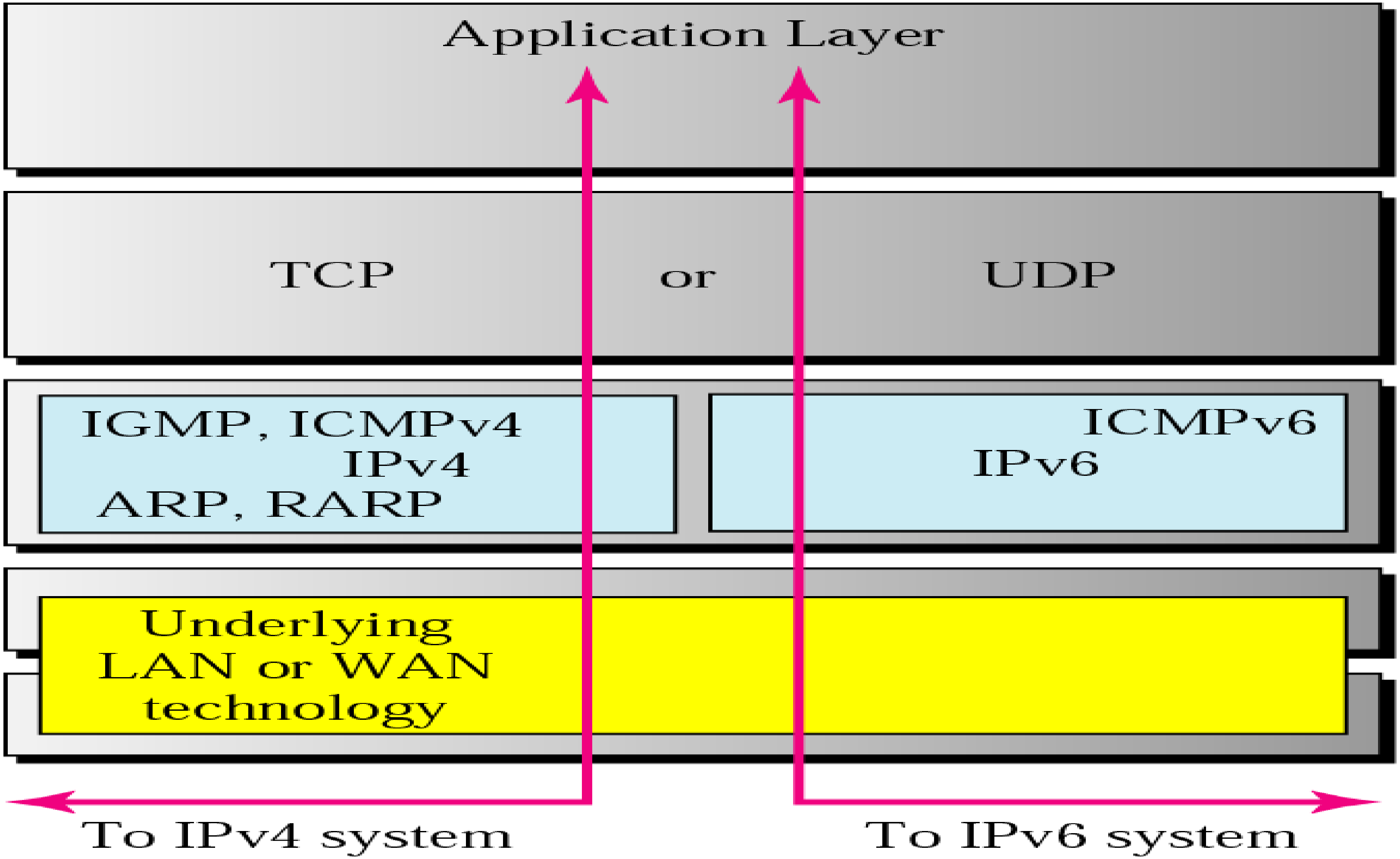
Transition from IPv4 to IPv6

Transition Strategies



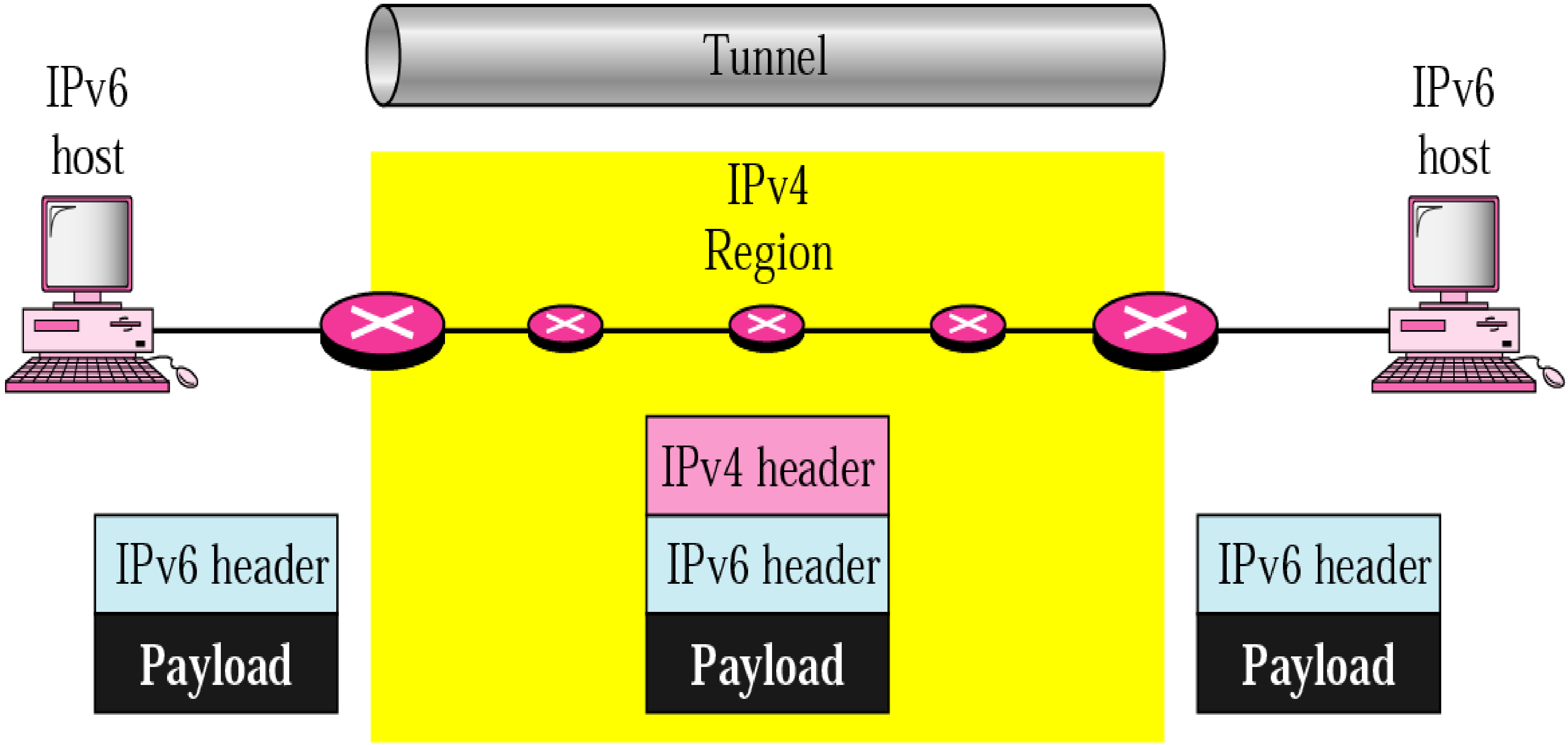
Transition from IPv4 to IPv6

Dual Stack



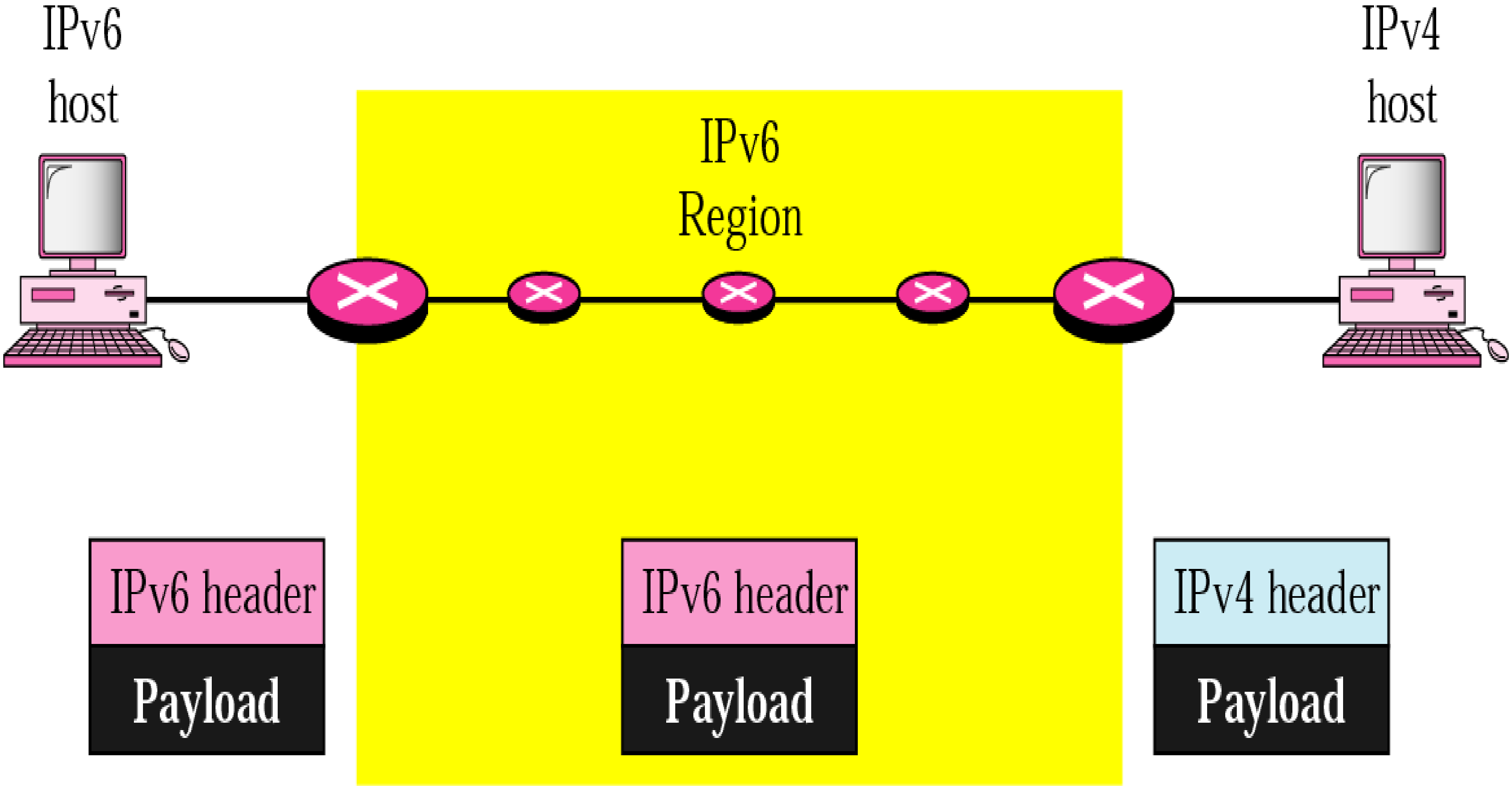
Transition from IPv4 to IPv6

Tunneling



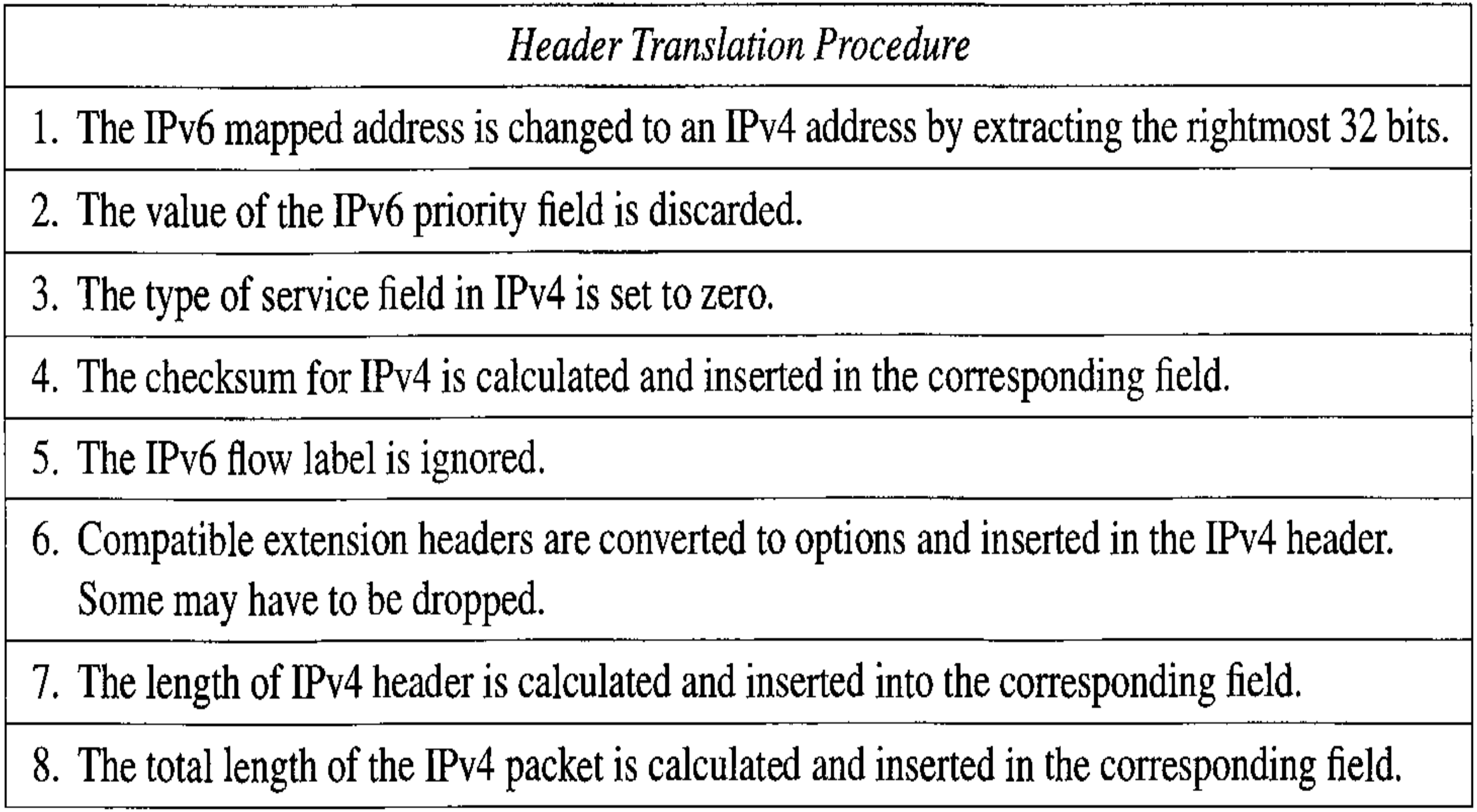
Transition from IPv4 to IPv6

Header Translation



Transition from IPv4 to IPv6

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.