Unit 3 (Part-2)

Network Layer: Logical Addressing

IPv4 Addresses

- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.
- The address space of IPv4 is 2^{32} or 4,294,967,296.

IPv4 Addresses

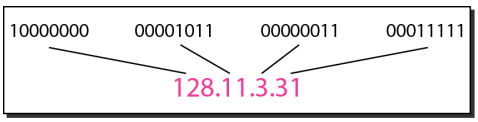


Figure 1: Dotted-decimal notation and binary notation for an IPv4 address

Example 1

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

a. 10000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation.

a. 129.11.11.239
b. 193.131.27.255

Example 2

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

a. 111.56.45.78
b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent (see Appendix B).

a. 01101111 00111000 00101101 01001110
b. 11011101 00100010 00000111 01010010

Example 3

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

a. 111.56.045.78
b. 221.34.7.8.20
c. 75.45.301.14
d. 11100010.23.14.67

Solution

a. There must be no leading zero (045).
b. There can be no more than four numbers.
c. Each number needs to be less than or equal to 255.
d. A mixture of binary notation and dotted-decimal notation is not allowed.

Classful Addressing

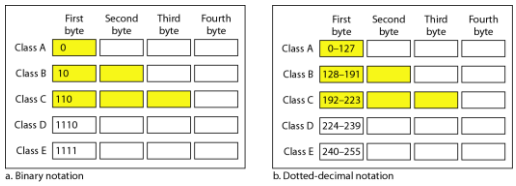
- In classful addressing, the address space is divided into five classes:

  A, B, C, D, and E.

- Each class occupies some part of the address space.

Classful Addressing

Figure 2: Finding the classes in binary and dotted-decimal notation



Example 4

Find the class of each address.

a. 00000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 14.23.120.8
d. 252.5.15.111

Solution

a. The first bit is 0. This is a class A address.
b. The first 2 bits are 1; the third bit is 0. This is a class C address.
c. The first byte is 14; the class is A.
d. The first byte is 252; the class is E.

Hostid & Netid

- Netid determines the network address
- Hostid determines the host connected to that network.

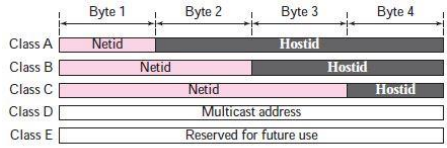


Fig1: Netid and hostid

Disadvantage of Classful Addressing

- Each class has fixed-size block of addresses.
- A large part of the available addresses were wasted.

Table 1 ***Number of blocks and block size in classful IPv4 addressing***

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

Classless Addressing

- Subnetting
  - It could divide the addresses into several smaller groups called subnets.
- Supernetting
  - Class C does not satisfy the need of midsize organizations.
  - Combine several class C blocks to form large network called supernet

Mask

- Mask is 32 bits number that defines the block of addresses.

Table 2 *Default masks for classful addressing*

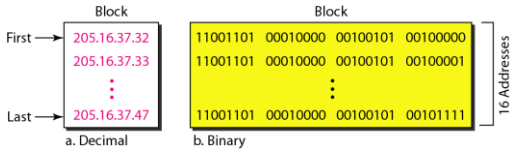
Class	Binary	Dotted-Decimal	CIDR
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

Classless Addressing

- Classful addressing, which is almost obsolete, is replaced with classless addressing.
- Restrictions on classless addressing:
  - IP addresses must be contiguous
  - Number of addresses must be power of 2 (1,2,4,8…)
  - First address must be evenly divisible by number of addresses.

Classless Addressing

Figure 3 *A block of 16 addresses granted to a small organization*



The decimal number coversion of first address is 3,440,387,360.

Classless Addressing

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

$$x.y.z.t\ /n$$

Where x.y.z.t defines one of the addresses and the /*n* defines the mask.

Classless Addressing

- The first address in the block can be found by setting the rightmost 32 – *n* bits to 0s.
- The last address in the block can be found by setting the rightmost 32 – n bits to 1s.
- The number of addresses in the block can be found by using the formula 2^{32-n}.

Example 6

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32−28 rightmost bits to 0, we get

11001101 00010000 00100101 0010000

or

205.16.37.32.

Example 7

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the last address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32 − 28 rightmost bits to 1, we get

11001101 00010000 00100101 00101111

or

205.16.37.47

Example 8

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. Find the number of addresses in the block.

Solution

The value of n is 28, which means that number of addresses is 2^{32-28} or 16.

Classless Addressing

Another way to find the first address, the last address, and the number of addresses is to represent the mask as a 32-bit binary (or 8-digit hexadecimal) number. This is particularly useful when we are writing a program to find these pieces of information.

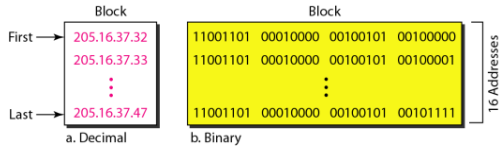
a. The first address can be found by ANDing the given addresses with the mask.

b. The last address can be found by ORing the given addresses with the complement of the mask.

c. The number of addresses can be found by complementing the mask.

Classless Addressing

Figure 4 *A network configuration for the block 205.16.37.32/28*



Example 9

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. Find

a. The first address in the block.

b. The last address in the block.

c. The number of address in the block.

Example 9 (continued)

Solution

a. The first address can be found by ANDing the given addresses with the mask. ANDing here is done bit by bit. The result of ANDing 2 bits is 1 if both bits are 1s; the result is 0 otherwise.

Address:	11001101	00010000	00100101	00100111
Mask:	**11111111**	**11111111**	**11111111**	**11110000**
First address:	11001101	00010000	00100101	00100000

Example 9 (continued)

b. ***The last address can be found by ORing the given addresses with the complement of the mask. ORing here is done bit by bit. The result of ORing 2 bits is 0 if both bits are 0s; the result is 1 otherwise. The complement of a number is found by changing each 1 to 0 and each 0 to 1.***

Address:	11001101 00010000 00100101 00100111
Mask complement:	00000000 00000000 00000000 00001111
Last address:	11001101 00010000 00100101 00101111

Example 9 (continued)

c. ***The number of addresses can be found by complementing the mask, interpreting it as a decimal number, and adding 1 to it.***

Mask complement:	00000000 00000000 00000000 00001111
Number of addresses:	15 + 1 = 16

Classless Addressing

- The first address in a block is normally not assigned to any device; it is used as the network address that represents the organization to the rest of the world.
- Each address in the block can be considered as a two-level hierarchical structure: the leftmost *n* bits (prefix) define the network; the rightmost 32 – n bits define the host.

Figure 6 *A frame in a character-oriented protocol*

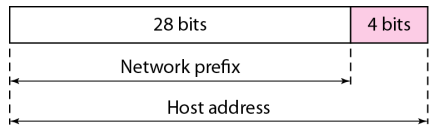
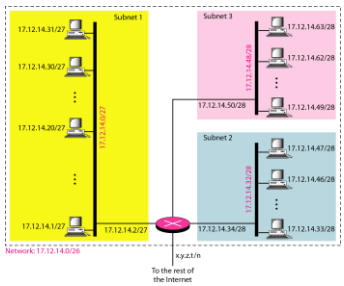


Figure 7 *Configuration and addresses in a subnetted network*



Example 10

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

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

b. ***The second group has 128 customers; each needs128 addresses.***

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

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

Example 10 (continued)

Solution

Figure 9 shows the situation.

Group 1

For this group, each customer needs 256 addresses. This means that 8 (log2 256) bits are needed to define each host. The prefix length is then 32 − 8 = 24. The addresses are

1st Customer:	190.100.0.0/24	190.100.0.255/24
2nd Customer:	190.100.1.0/24	190.100.1.255/24
. . .		
64th Customer:	190.100.63.0/24	190.100.63.255/24
Total = 64 × 256 = 16,384		

Example 10 (continued)

Group 2

For this group, each customer needs 128 addresses. This means that 7 (log2 128) bits are needed to define each host. The prefix length is then 32 − 7 = 25. The addresses are

1st Customer:	190.100.64.0/25	190.100.64.127/25
2nd Customer:	190.100.64.128/25	190.100.64.255/25
. . .		
128th Customer:	190.100.127.128/25	190.100.127.255/25
Total = 128 × 128 = 16,384		

Example 10 (continued)

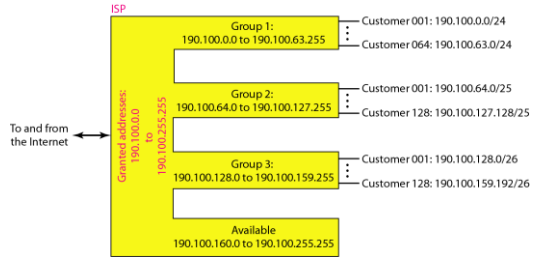
Group 3

For this group, each customer needs 64 addresses. This means that 6 ($log_2 64$) bits are needed to each host. The prefix length is then 32 − 6 = 26. The addresses are

1st Customer:	190.100.128.0/26	190.100.128.63/26
2nd Customer:	190.100.128.64/26	190.100.128.127/26
. . .		
128th Customer:	190.100.159.192/26	190.100.159.255/26
Total = 128 × 64 = 8192		

Number of granted addresses to the ISP: 65,536
Number of allocated addresses by the ISP: 40,960
Number of available addresses: 24,576

Figure 9 *An example of address allocation and distribution by an ISP*



Private IP Addresses

***NAT** enables the users to have large a set of addresses internally and a small set of addresses externally.*

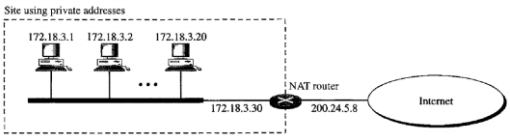


Figure 10 *A NAT implementation*

Private IP Addresses

Table 3 *Addresses for private networks*

Range			Total
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

IPv6 ADDRESSES

Despite all short-term solutions, address depletion is still a long-term problem for the Internet. This and other problems in the IP protocol itself have been the motivation for IPv6.

An IPv6 address is 128 bits long.

Topics discussed in this section:

Structure

Figure 14 *IPv6 address in binary and hexadecimal colon notation*

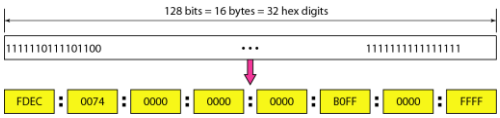
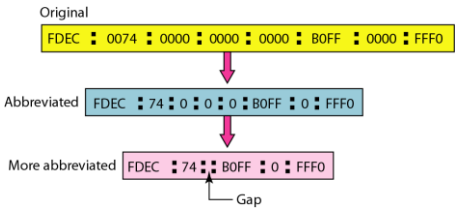


Figure 15 *Abbreviated IPv6 addresses*

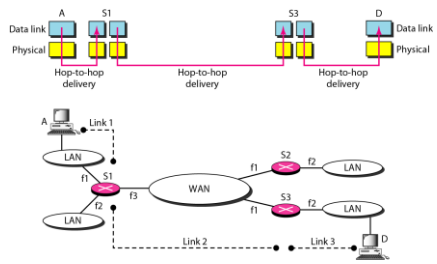


NETWORK LAYER: INTERNET PROTOCOL

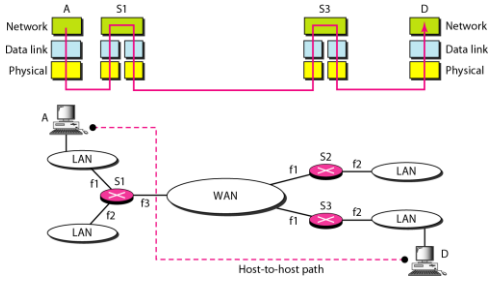
Network Layer

- Need
  - A frame has no routing info.
  - Only physical and data link layers involve.
  - How to deliver packet to the outside world
- Responsibility
  - Host-to-host delivery
  - For routing packets through the router and switches.

Links between two hosts



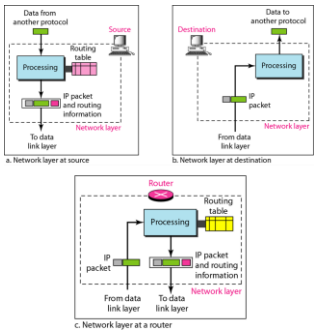
Network layer in an internetwork



Network Layer

- Source
  - Creating a packet from the upper layer.
    - The header contains source and destination IP addresses.
  - Checking the routing table to find the routing info (eg. Outgoing interface, or machine address of the next hop)
  - If the packet is larger than MTU, fragment it.
    - Note that it is different from L4 segmentation/reassembly
- Router
  - Routing the packet by consulting the routing table for each incoming packet and find the interface from which the packet must be sent.
- Destination
  - Address verification.
  - For fragmented frames, wait for all fragmentations then reassemble them before delivering the packet to the upper layer.

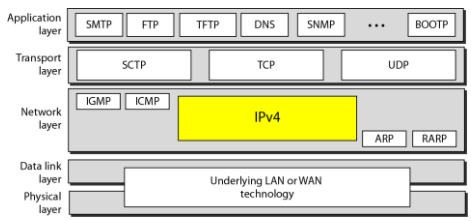
Network layer at the source, router, and destination



Type of Network Services

- Delivery of packet can be accomplished by using either.
  - Connection-oriented service
  - Connection-less service
- In connection-oriented service
  - Each packet follow the same path.
  - A packet is logically connected to the packet traveling before it.
- In connection-less service
  - Each packet is treated independently by the intermediate routers.
  - Packets in a message may travel through different paths.
- Why does Internet at Network Layer use connection less service?

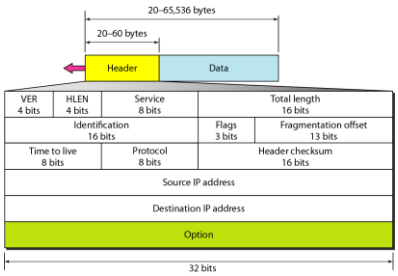
IPv4 in TCP/IP protocol suite



IPv4

- Unreliable and connection-less datagram protocol
- Best-effort delivery
  - No error control
  - No flow control
- If reliability important:
  - It must be paired with reliable protocol such as TCP.
- Packets in the IPv4 are called datagrams.

IPv4 datagram format



IPv4 Header

- Variable length: 20-60 byte (multiple of 4)
- Contains routing information

IPv4 Format

- Version (4-bit): currently 4.
- Header length (4-bit): the length of the IP header in 4-byte unit.
- Type of Services(TOS):
  - First 3 bits are precedence bits
  - Next 4 bits are for TOS and rightmost bit is unused bit.

- Total length
  - 16 bit field that defines total length of the datagram including the header in bytes.
  - 16-bit number, the maximum IP size is limited to 2^{16} bytes, or 65,535 bytes.

IPv4 Format

- Identification
  - A source node gives a unique ID to each packet.
  - Identification, Flags, Fragmentation offset fields are used for fragmentation.
- Time to Live (TTL)
  - A packet has a limited lifetime in the network to avoid zombie packets.
  - Designed to hold a timestamp, and decreased by each router. A packet is discarded by a router if TTL is zero.
  - Revised to hold the maximum number of hops the packet can travel through the network. Each router decrements it by one.

IPv4 Format

- Protocol
  - To define payload protocol type
  - 1 for ICMP
  - 2 for IGMP
  - 6 for TCP
  - 17 for UDP
  - 89 for OSPF

IPv4 Format

- Header checksum
  - An IP header is slightly modified by each router. At least TTL field.
  - The checksum must be re-calculated by routers which is a kind of general computers with more than one network interface.
  - The checksum must be efficiently calculated with no need of special hardware.
- Source IP address and Destination IP address
- Options
  - Variable length
  - For new protocols
- Padding
  - To make the header a multiple of 32-bit words

Example 1

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 × 4 = 8). The minimum number of bytes in the header must be 20. The packet has been corrupted in transmission.

Example 2

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 3

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 4

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.

Example 5

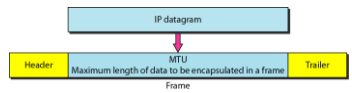
An IPv4 packet has arrived with the first few hexadecimal digits as shown.

0x4B00040000100000806xxxx0A0C0E050C060709

Create a header format of the IP datagram.
What is the value of HLEN and option?
How many hops can this packet travel before being dropped?
The data belong to what upper-layer protocol?

Fragmentation

- A IP packet can travel through many different networks using different Data Link layers.
- The source node has no idea of the path and data link layer its packets will travel.
- MTU
  - Each DL has its own frame format and limitation.
  - One of such limitation is the maximum size of the frame, which is imposed by software, hardware, performance, and standards.



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

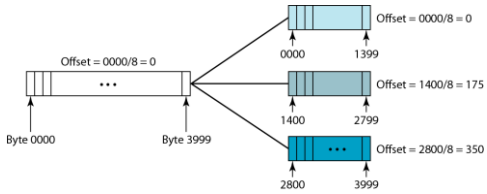
Fragmentation of IP

- The source node usually does not fragment the packet. Instead, L4 will segment the data into a size that can fit into L3 and L2 of the source.
- But, there is a possibility that a packet travel through a link whose MTU is smaller than one of the source node.
  - Then, the packet must be fragmented to go forward the next hop.
  - Each fragment has its own header mostly repeated from the original packet.
  - A fragmented packet can be further fragmented into even smaller packet.
  - Fragmented packets will be re-assembled only by the final destination. Why?

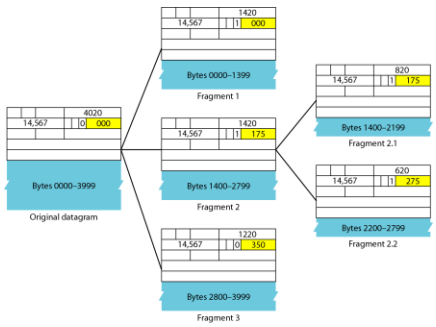
Fields for Fragmentation

- Identification
  - The source host generates the unique ID
- Flags (3-bits)
  - Unused bit
  - DF bit (Don't Fragment)
  - 1 – force the router not to fragment the packet. If the packet length is greater than the MTU, the router will discard the packet and send an error message to the source
  - MF bit (More Fragment)
  - 1 – tell the destination whether or not more fragments follow
- Offset
  - Unit of 8-byte (why?)
  - Between the beginning of the packet to be fragmented and the beginning of the fragment
- Intelligent Router (Switch) uses these fields for efficiency.
  - PPD (Partial Packet Discard)
  - EPD (Early Packet Discard)

Fragmentation example



Detailed fragmentation example



Example 6

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 7

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 8

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 9

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 10

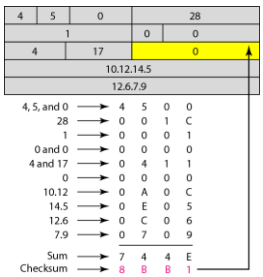
A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the total length field is 100. What are the numbers of the first byte and the last byte?

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.

Example 11

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.



Deficiencies in IPv4

- Addresses depletion
- Real time audio and video transmission
- No encryption and authentication of data for some application

Why IPv6?

- Larger address space
- Better header format
- New options
- Allowance for extension
- Support for resource allocation
- Support for more security

IPv6 Datagram

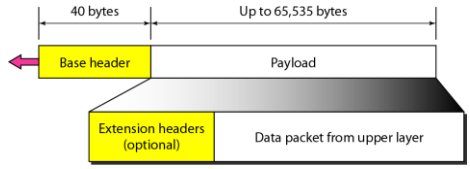
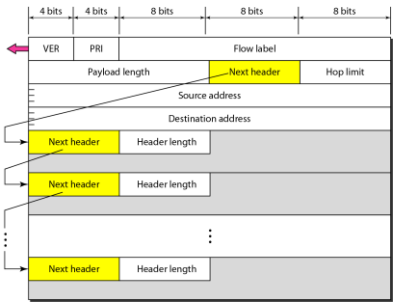


Figure *IPv6 datagram header and payload*

IPv6 Datagram Format



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

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

IPv4 Vs 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.

Thank You

Any Query?