The Network Layer

- ✓ The **network layer** is responsible for the source-to-destination delivery of a packet (called IP packet), possibly across multiple networks (homogeneous or heterogeneous) networks called *internetworking*.

- ✓ In case of homogeneous networks (each network use same protocol) packet is passed through router and that of through heterogeneous networks (dissimilar protocol is used by network, for example 4G LTE network to Internet) the job is done through gateway router.

- ✓ Whereas the **data link layer** oversees the delivery of the packet between two systems on the same network.

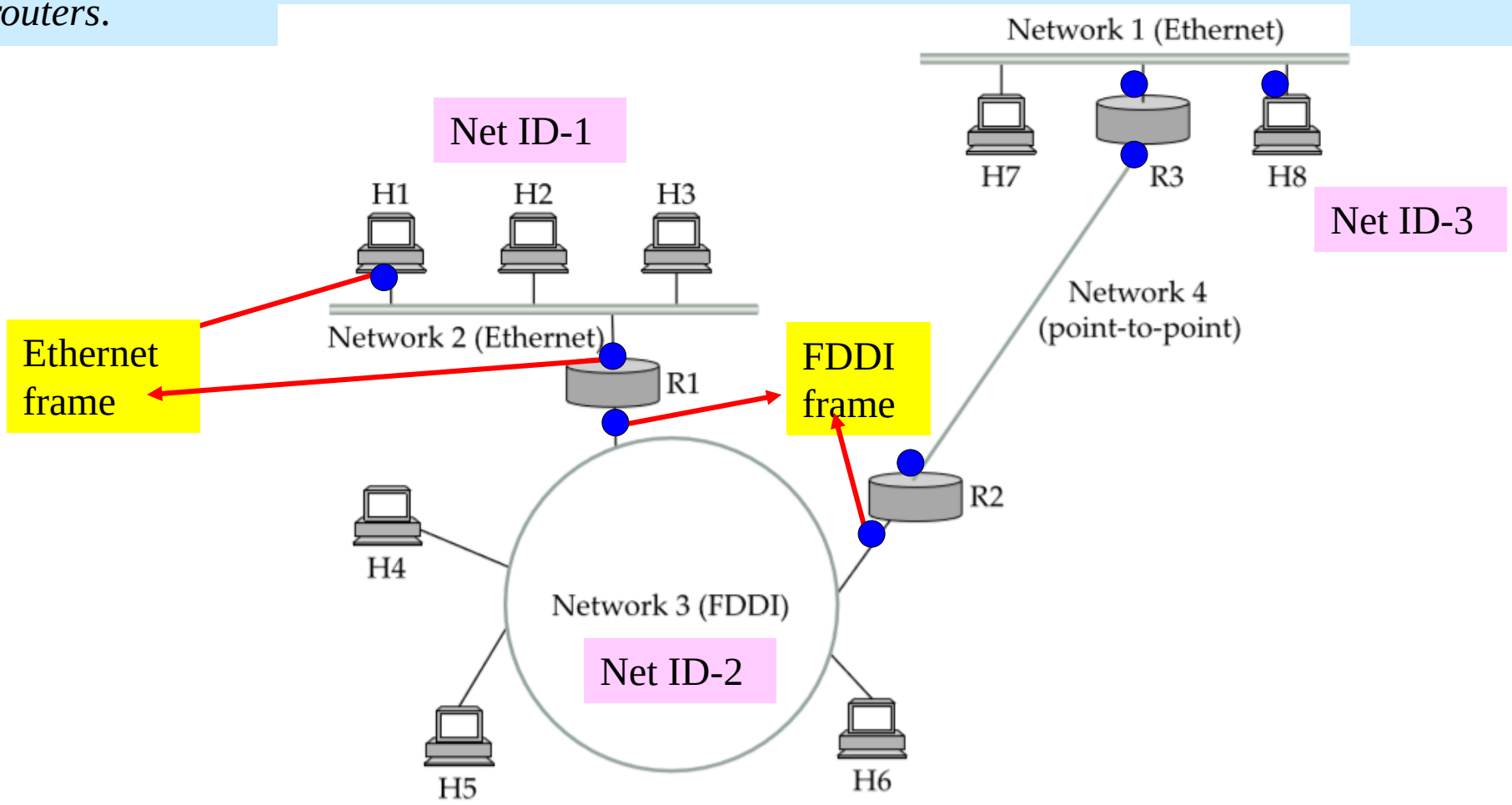
✓We use the term "*internetwork*," or sometimes just "*internet*" with a lowercase *i*, to refer to an **arbitrary collection of networks** interconnected to provide some sort of host-to-host packet delivery service.

✓For example, a corporation with many sites (or several branch offices) might construct a private *internetwork* by interconnecting the LANs at their different sites (or branch office) with point-to-point links leased from the phone company.

✓When we are talking about the widely used, global *internetwork* to which a large percentage of networks are now connected, we call it the "**Internet**" with a capital *I* .

✓The **Internet Protocol** is the key tool used today to build scalable, heterogeneous internetworks.

Figure below shows an example *internetwork*. An *internetwork* is often referred to as a network of networks because it is made up of lots of smaller networks. In this figure, we see Ethernets, an FDDI (Fibre Distributed Data Interface) ring, and a point-to-point link. Each of these is a single technology network. The nodes that interconnect the networks are called *routers*.



A simple internetwork: H*n* = host; R*n* = router

Figure below shows how hosts H1 and H8 are logically connected by the internet in including the protocol graph running on each node. Let H1 transmits a TCP packet to node H8.

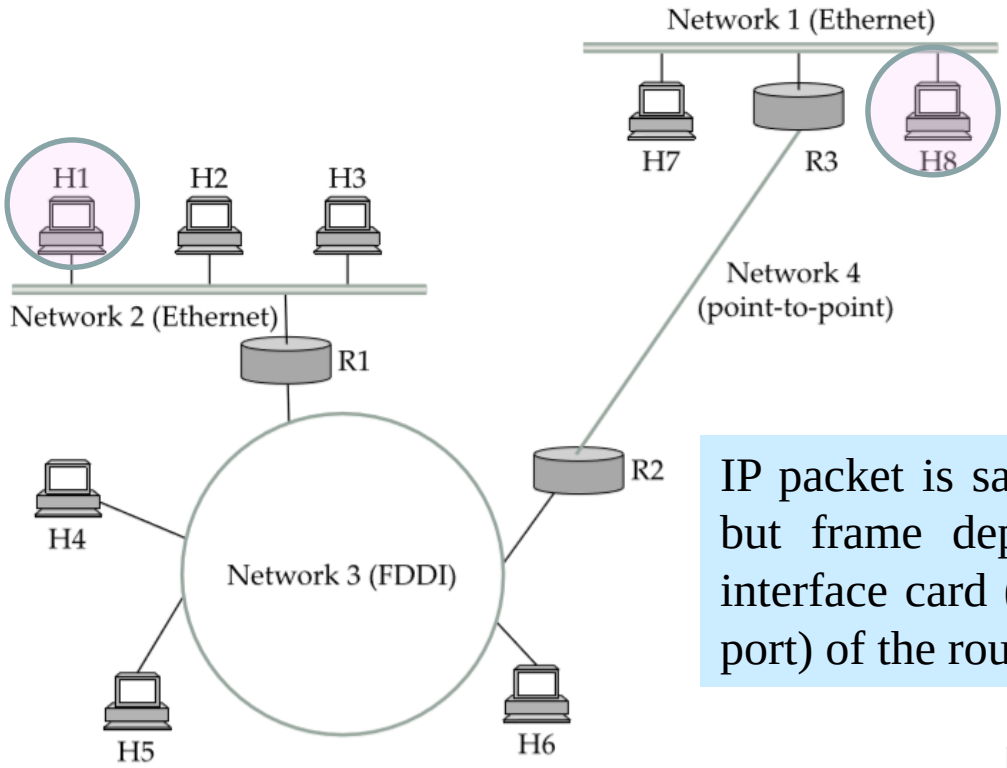


Figure (a)

IP packet is same for all the network but frame depends on the network interface card (Ethernet port or FDDI port) of the router

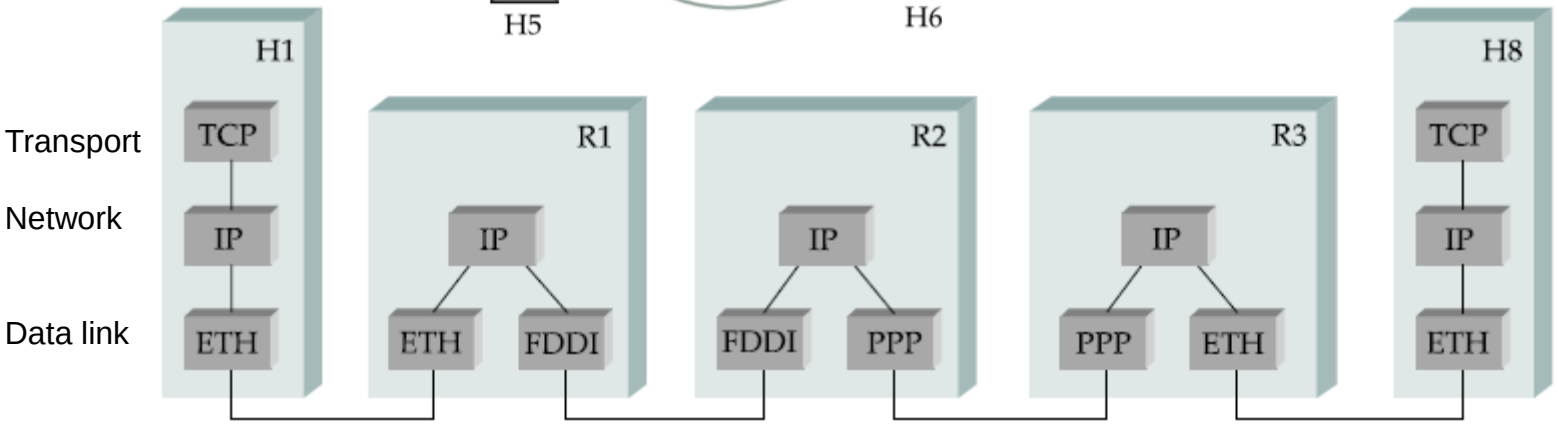


Figure (b)

IP Packet

- ✓ We need a global addressing scheme to route packet through interconnected network; we called this logical addressing as IP address under TCP/IP protocol suite.

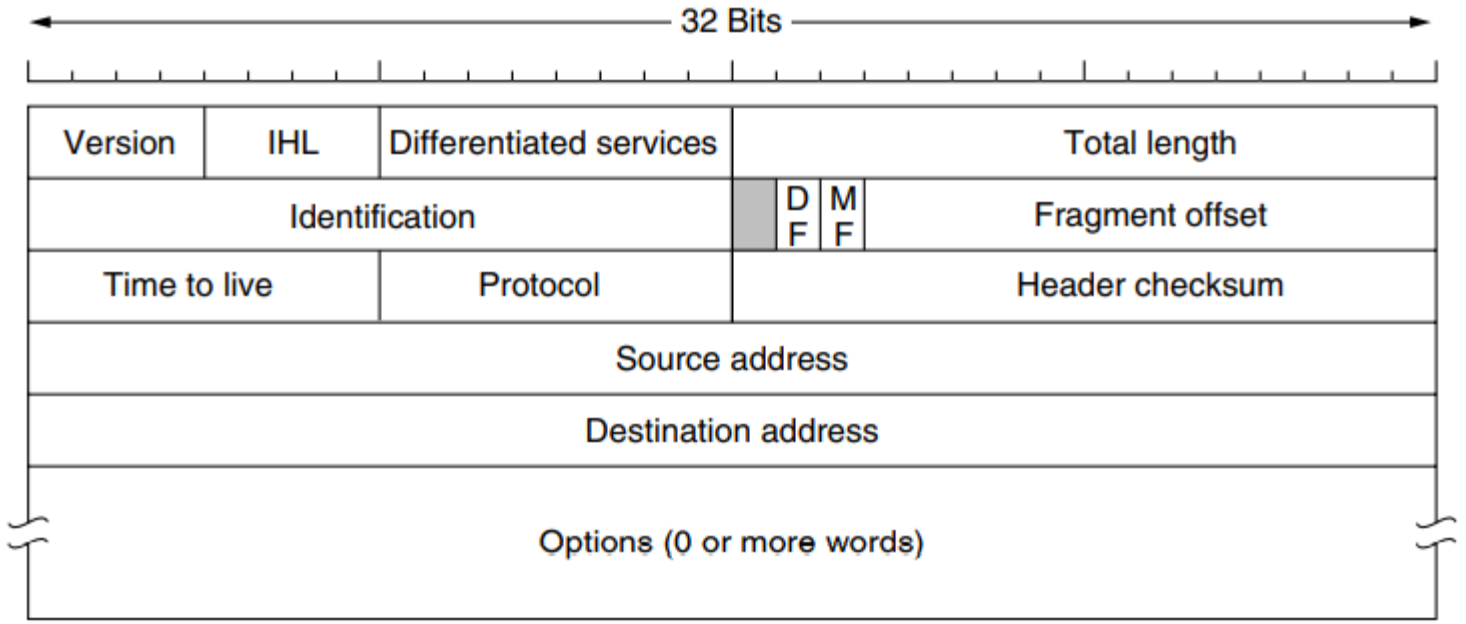
- ✓ In IPv4 (IP version 4) the length of the address is 32 bits and for new generation of IP or IPv6 (IP version 6) the length of address is 128 bits. This section deals with different fields of IP packet.

- ✓ The IP packet, like most packets, consists of a header followed by a number of bytes of data.

IPv4 packet:

Version number (4 bits)
Indicates the version of the IP protocol
Typically “4” (for IPv4), and sometimes “6” (for IPv6)

Internet Header Length (IHL)
The header length is not constant, IHL field indicates the number of 32 bit words in the header

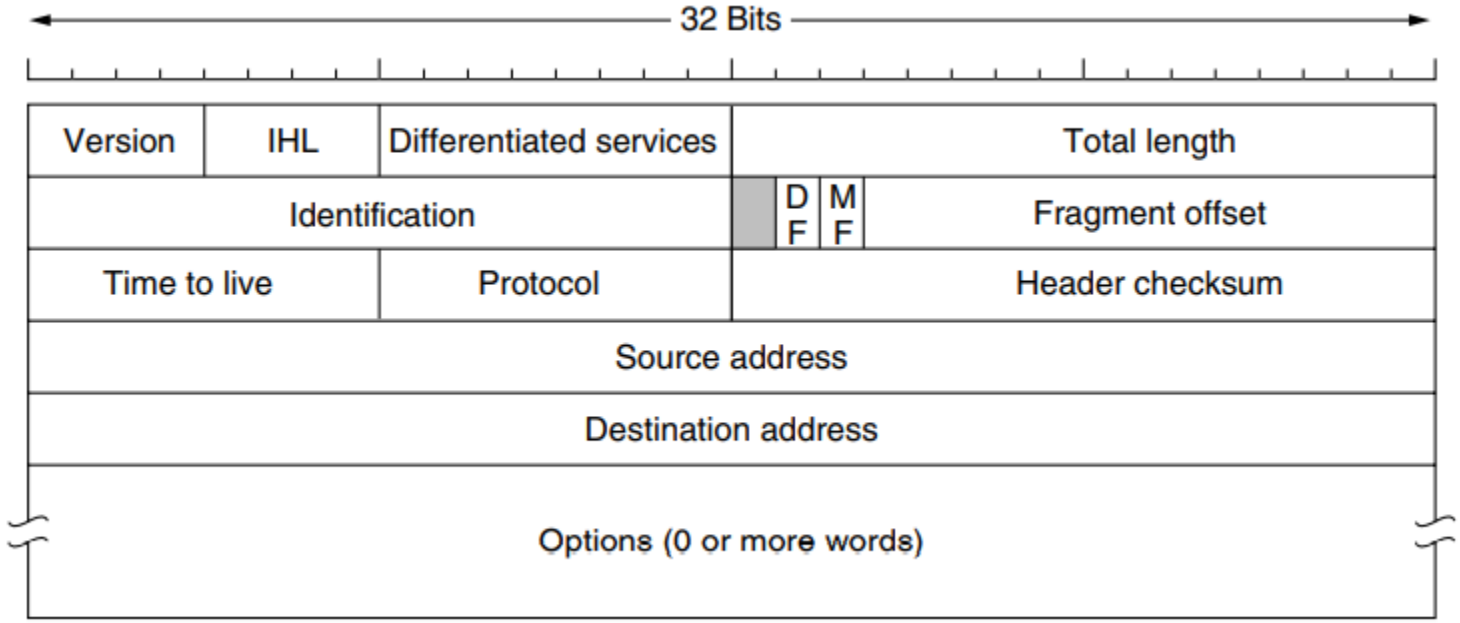


IP Packet Structure (IPv4)

Differentiated Services (8 bits)

❖It is intended to distinguish between different classes of service. Various combinations of reliability and speed are possible. For digitized voice, fast delivery is more essential than accurate delivery. For file transfer, error-free transmission is more important than fast transmission.

❖This field is related to three parameters: {Delay, Throughput, Reliability} in communication.



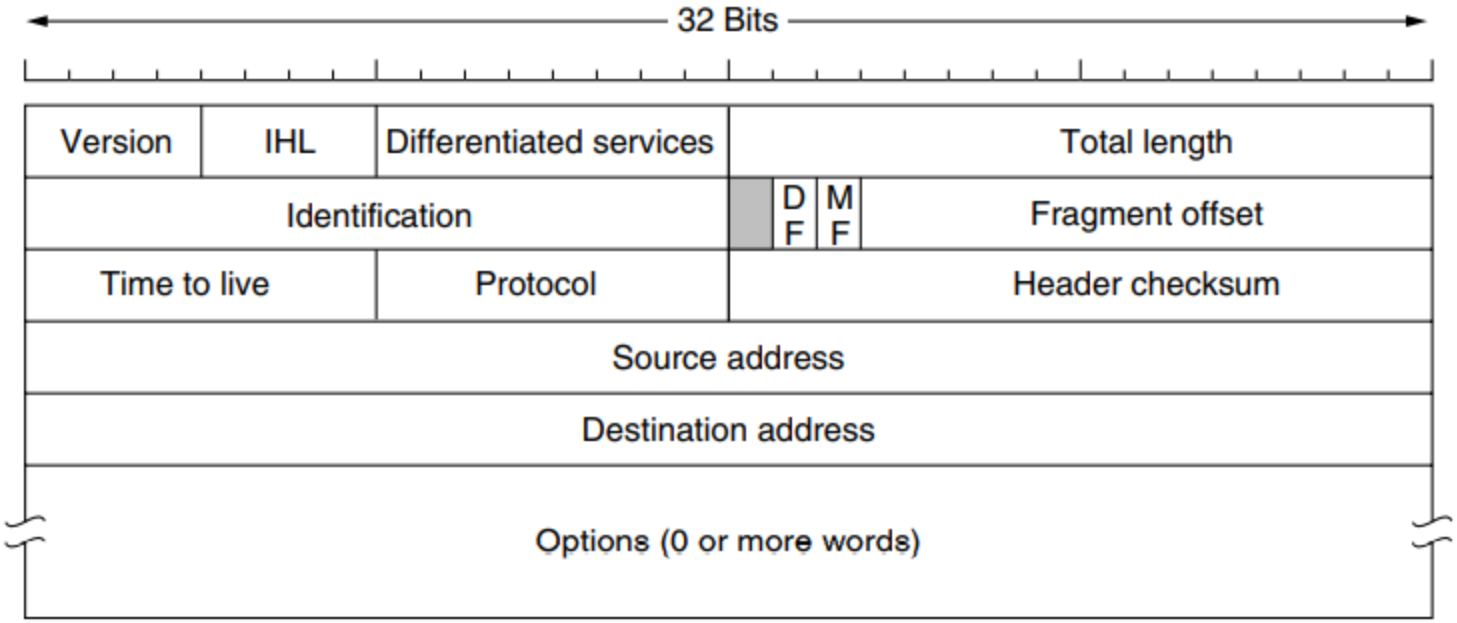
Total length (16 bits)

– Number of bytes in the packet

– Maximum size is 65,535 bytes (2^{16} -1)

Identification

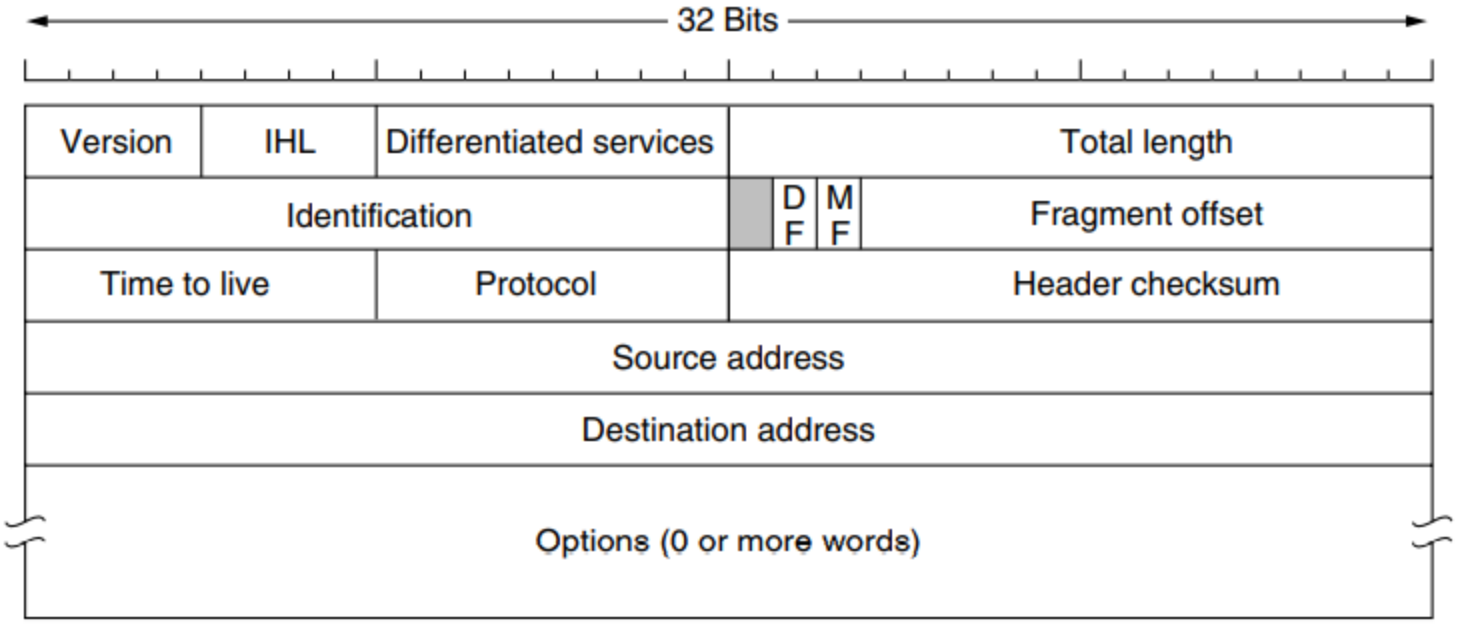
The Identification field is needed to allow the destination host to determine which datagram a newly arrived fragment belongs to. All the fragments of a datagram contain the same Identification value.



Flags :A three-bit field follows and is used to control or identify fragments. They are (in order, from high order to low order):

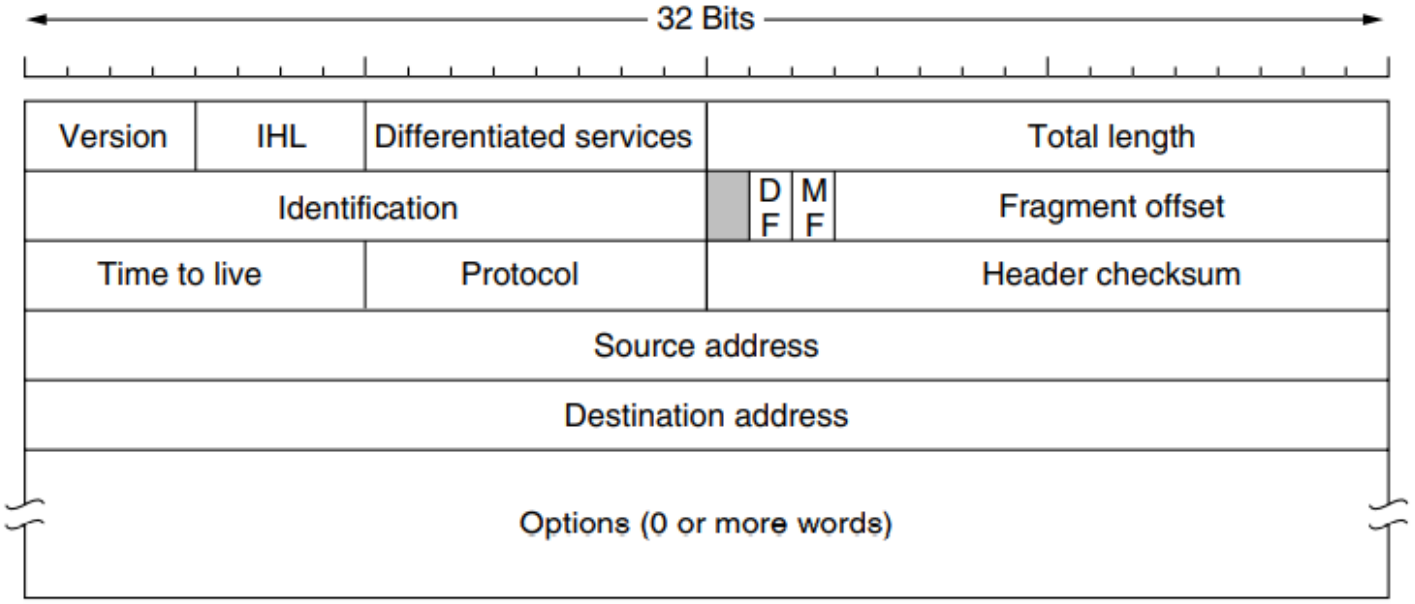
bit 0: Reserved; must be zero.
bit 1: Don't Fragment (DF)
bit 2: More Fragments (MF)

Fragment Offset The Fragment offset tells where in the current packet this fragment belongs. All fragments except the last one in a datagram must be a multiple of 8 bytes, the elementary fragment unit. Since 13 bits are provided, there is a maximum of $2^{13} = 8192$ fragments per datagram.



Time-To-Live (8 bits)

- ✓ The TTL (Time to live) field is a counter used to limit packet lifetimes. It was originally supposed to count time in seconds, allowing a maximum lifetime of 255 sec. It must be decremented on each hop and is supposed to be decremented multiple times when a packet is queued for a long time in a router.
- ✓ In practice, it just counts hops. When it hits zero, the packet is discarded and a warning packet is sent back to the source host. This feature prevents packets from wandering around forever, which happens if the routing tables ever become corrupted.



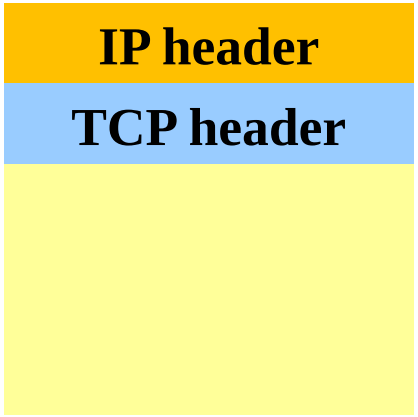
Protocol (8 bits)

The Protocol field tells it which transport process to give the packet to. TCP is one possibility, but so are UDP and some others.

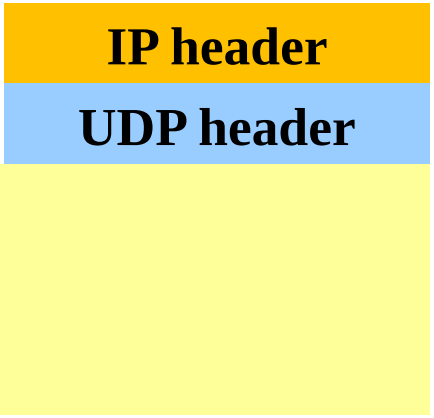
Identifies the higher-level protocol

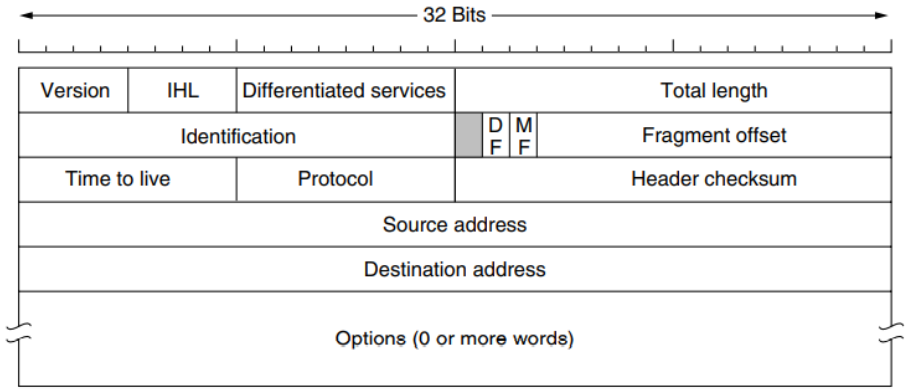
- "6" for the Transmission Control Protocol (TCP)
- "17" for the User Datagram Protocol (UDP)

protocol=6



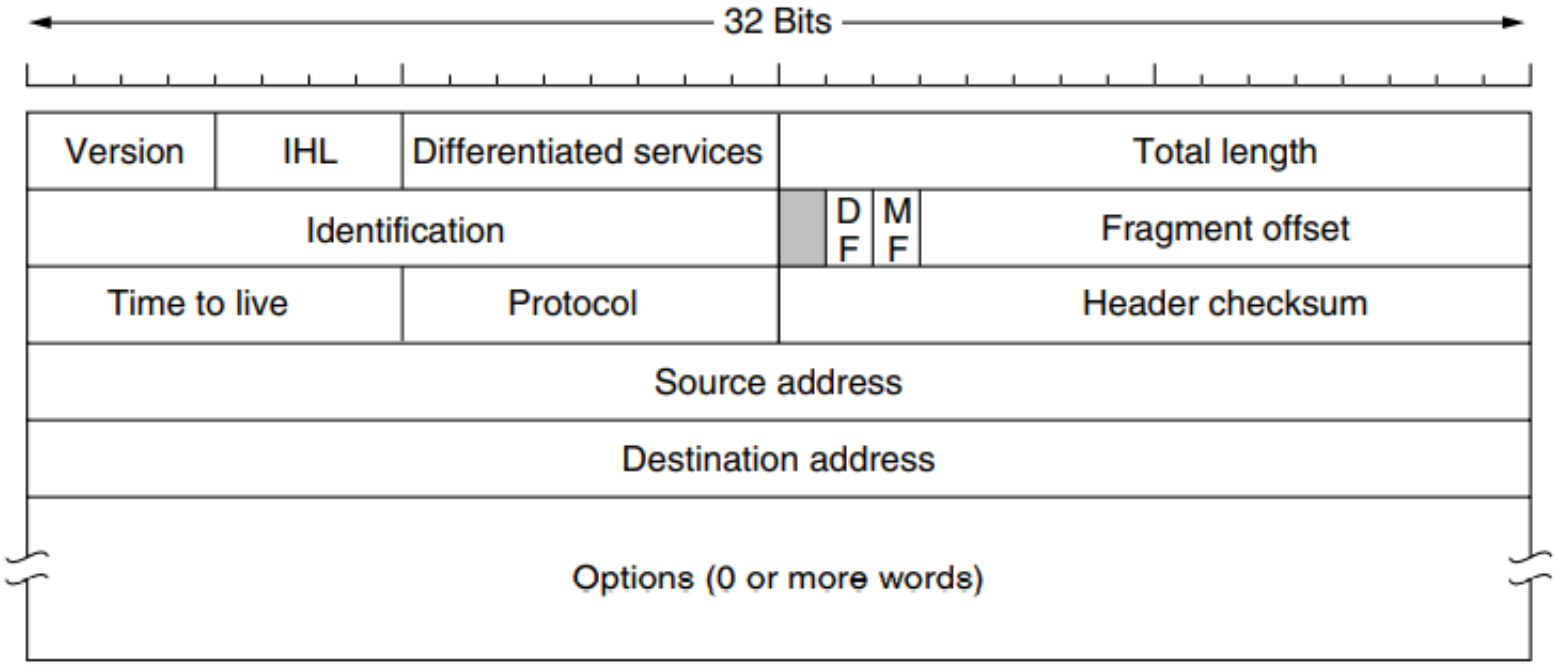
protocol=17





Checksum (16 bits)

- Sum of all 16-bit words in the IP packet header
- If any bits of the header are corrupted in transit
  … the checksum won't match at receiving host
- Receiving host discards corrupted packets
  - Sending host will retransmit the packet, if needed

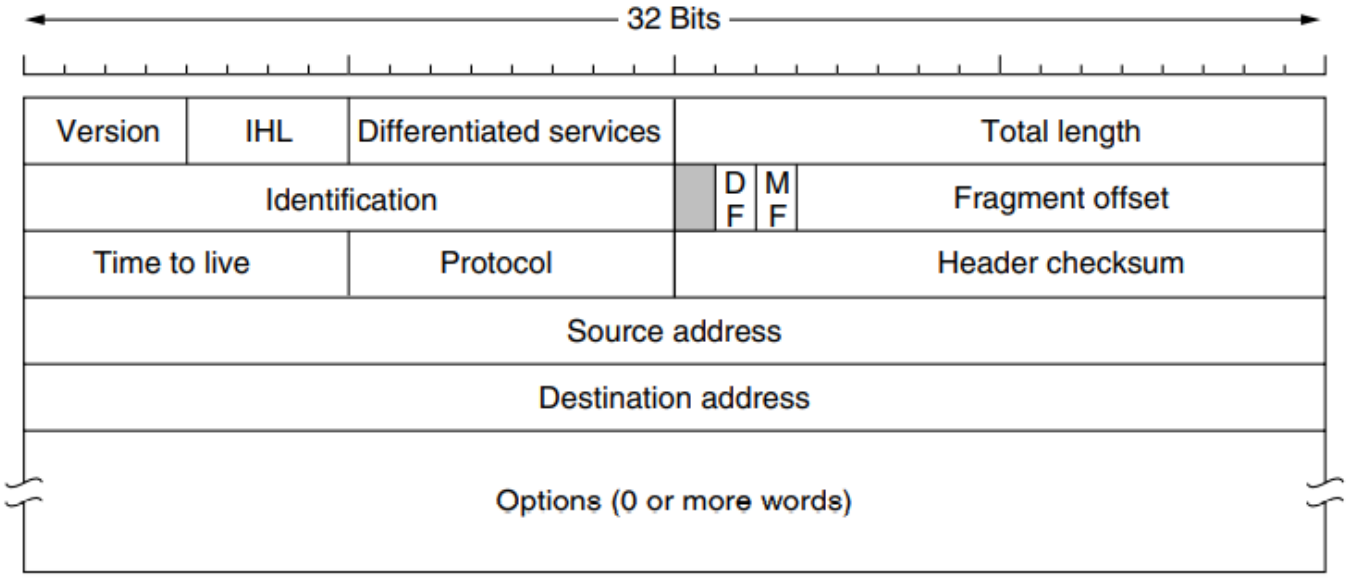


Example: Calculating a checksum

❑The header is shown in red and the checksum is underlined.
4500 0073 0000 4000 4011 b861 c0a8 0001 c0a8 00c7 0035 e97c 005f 279f 1e4b 8180

❑To calculate the checksum, we can first calculate the sum of each 16 bit value within the header, skipping only the checksum field itself. Note that the values are in hexadecimal notation.

4500 + 0073 + 0000 + 4000 + 4011 + c0a8 + 0001 + c0a8 + 00c7
= 2479C (equivalent to 149,404 in decimal).



- Next, we convert the value 2479C to binary:

0010 0100 0111 1001 1100

The first 4 bits are the carry and will be added to the rest of the value:

0010 + 0100 0111 1001 1100 = 0100 0111 1001 1110

Next, we flip every bit (1's complement) in that value, to obtain the checksum:

0100 0111 1001 1110 becomes:

1011 1000 0110 0001

This is equal to **B861** in hexadecimal, as shown underlined in the original IP packet header.

❑ Verifying a checksum

When verifying a checksum, the same procedure is used as above, except that the original header checksum is not omitted.

4500 + 0073 + 0000 + 4000 + 4011 + **b861** + c0a8 + 0001 + c0a8 + 00c7 = 2fffd

Add the carry bits:

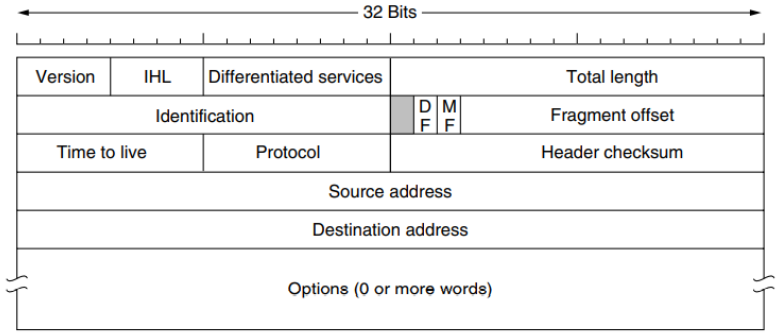
fffd + 2 = ffff

Taking the 1's complement (flipping every bit) yields 0000, which indicates that no error is detected. IP header checksum does not check for the correct order of 16 bit values within the header.

- Two IP addresses
  - Source IP address (32 bits)
  - Destination IP address (32 bits)
- Destination address
  - Unique identifier for the receiving host
  - Allows each node to make forwarding decisions
- Source address
  - Unique identifier for the sending host
  - Recipient can decide whether to accept packet
  - Enables recipient to send a reply back to source

Options

The Options field was designed to provide an opportunity to allow subsequent versions of the protocol to include information not present in the original design and to permit experimenters to try out new ideas.

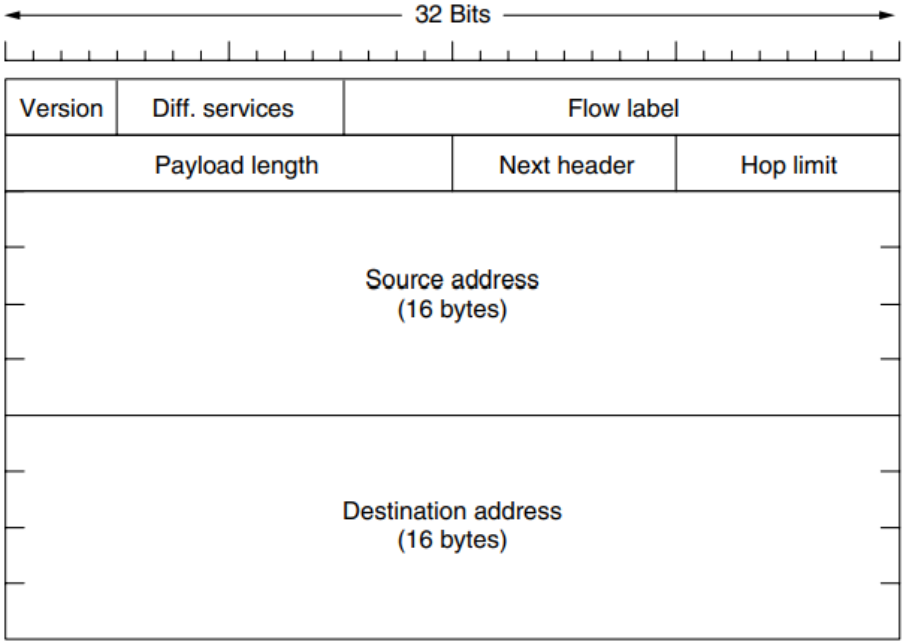


Some of the IP options

Option	Description
Security	Security Specifies how secret the datagram is
Strict source routing	Gives the complete path to be followed
Loose source routing	Gives a list of routers not to be missed
Record route	Makes each router append its IP address
Timestamp	Makes each router append its address and timestamp

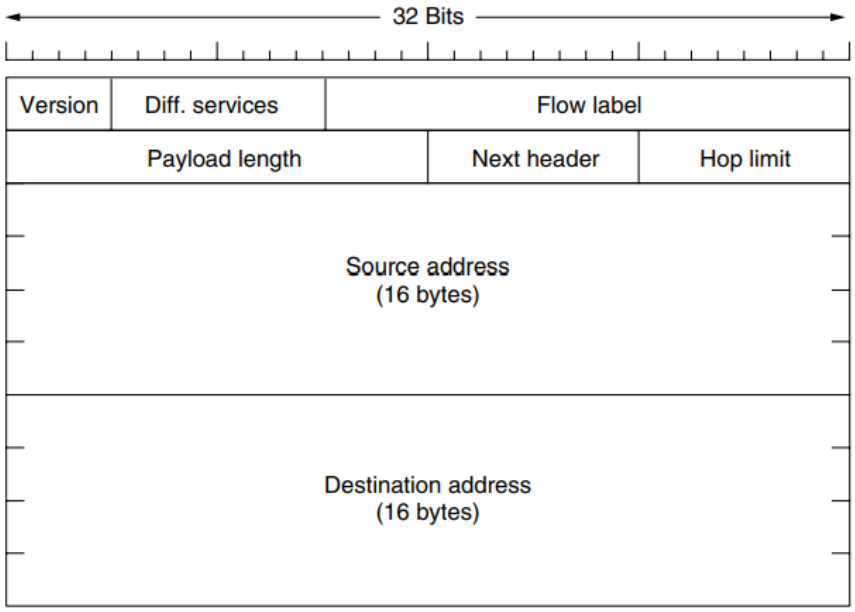
IPv6 Header

- ✓ The **Flow label** field provides a way for a source and destination to mark groups of packets that have the same requirements and should be treated in the same way by the network.



- ✓ For example, a stream of packets from one process on a certain source host to a process on a specific destination host might have stringent (strict condition) delay requirements and thus need reserved bandwidth.
- ✓ When a packet with a nonzero Flow label shows up, all the routers can look it up in internal tables to see what kind of special treatment it requires.

The **Payload length** field tells how many bytes follow the 40-byte header i.e. it measures the length of message field of Fig. below.



Next header field tells which transport protocol (e.g., TCP, UDP) is passed to the packet.

The **Hop limit** field is used to keep packets from living forever. It is, in practice, the same as the Time to live field in IPv4, namely, a field that is decremented on each hop.