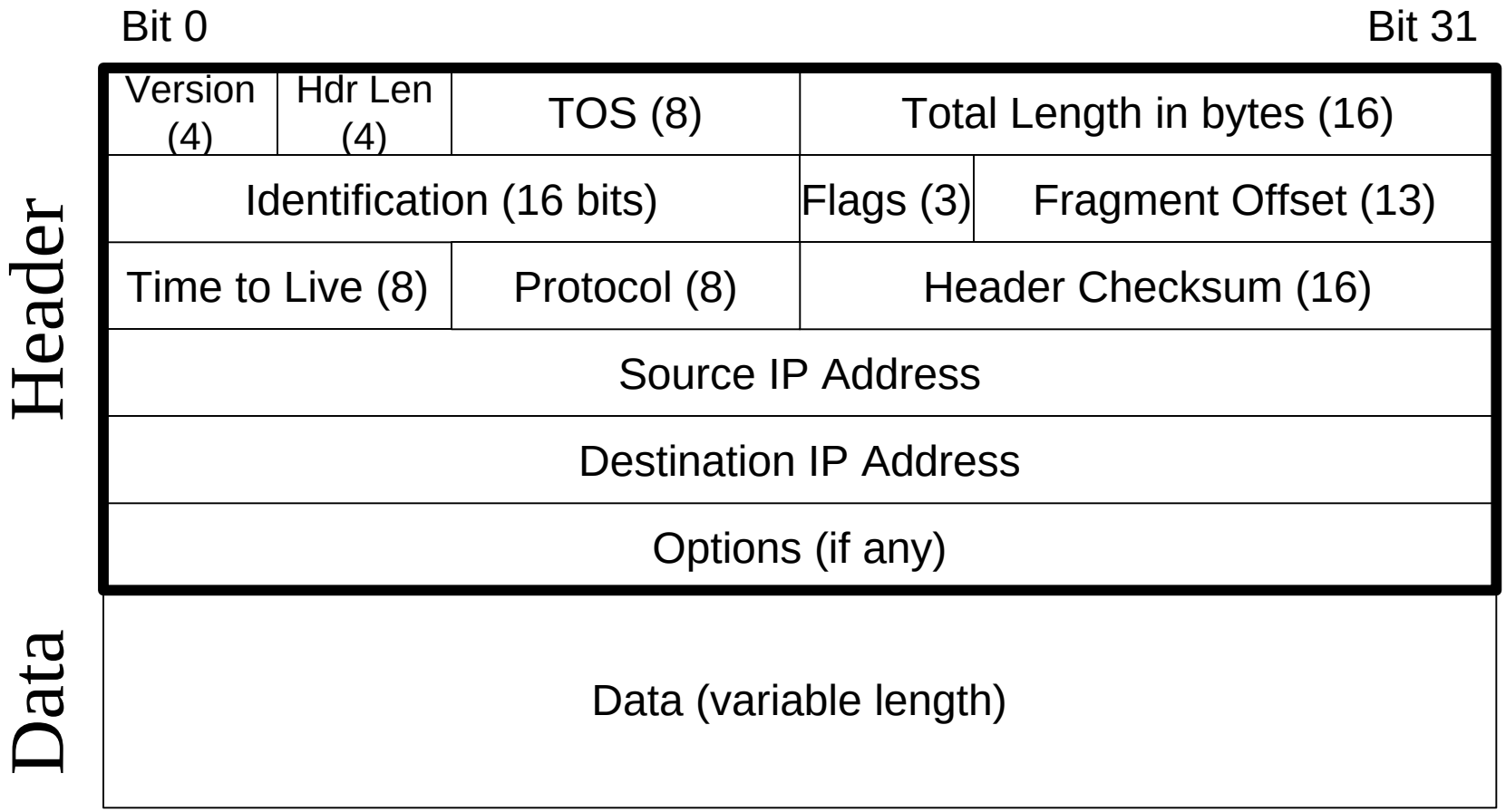
Internet Protocol

IP Datagram, Fragmentation and Reassembly

IP Datagram


Bit 0
Bit 31
Header
Version (4)
Hdr Len (4)
TOS (8)
Total Length in bytes (16)
Identification (16 bits)
Flags (3)
Fragment Offset (13)
Time to Live (8)
Protocol (8)
Header Checksum (16)
Source IP Address
Destination IP Address
Options (if any)
Data
Data (variable length)

IP Packet Header

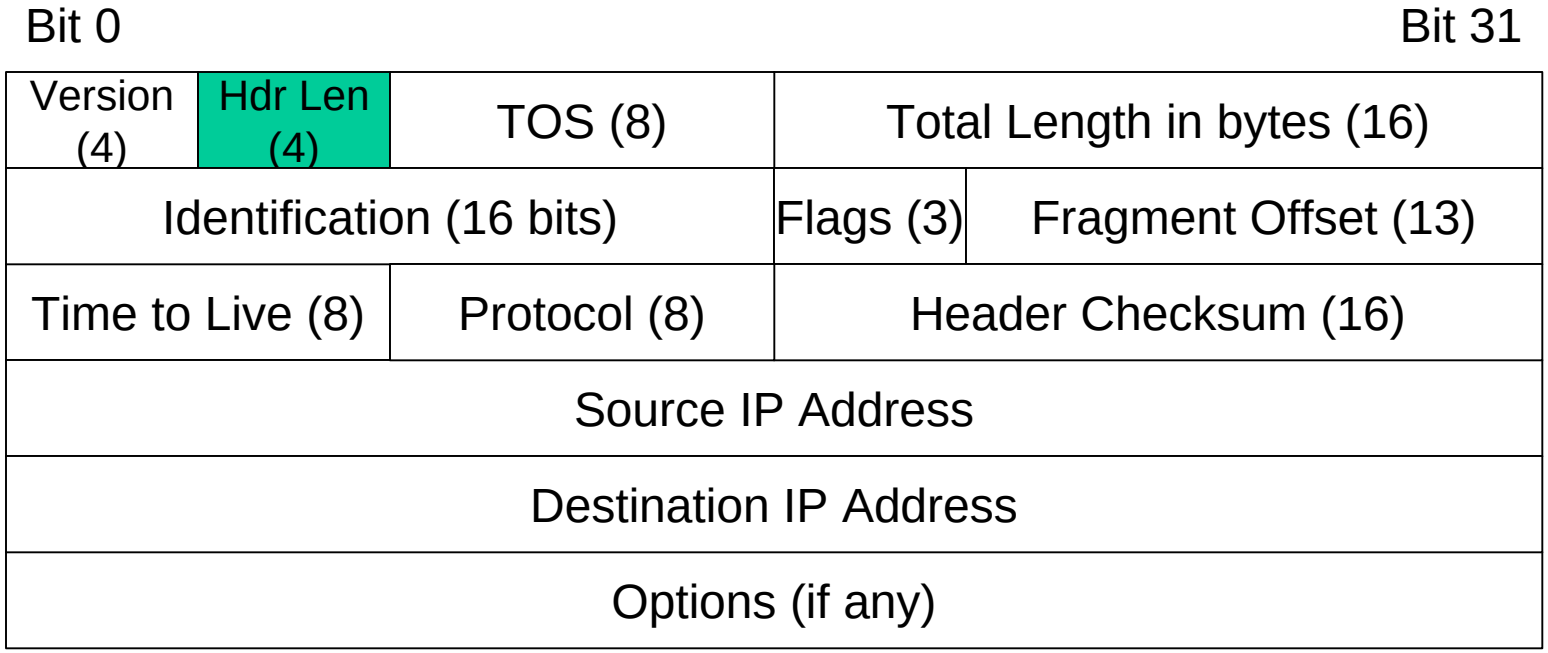
- Version
  - Version number of IP protocol
  - Current version is Version 4
  - Version 6 has different header format

Bit 0 Bit 31

Version (4)	Hdr Len (4)	TOS (8)	Total Length in bytes (16)	
Identification (16 bits)			Flags (3)	Fragment Offset (13)
Time to Live (8)		Protocol (8)	Header Checksum (16)	
Source IP Address				
Destination IP Address				
Options (if any)				

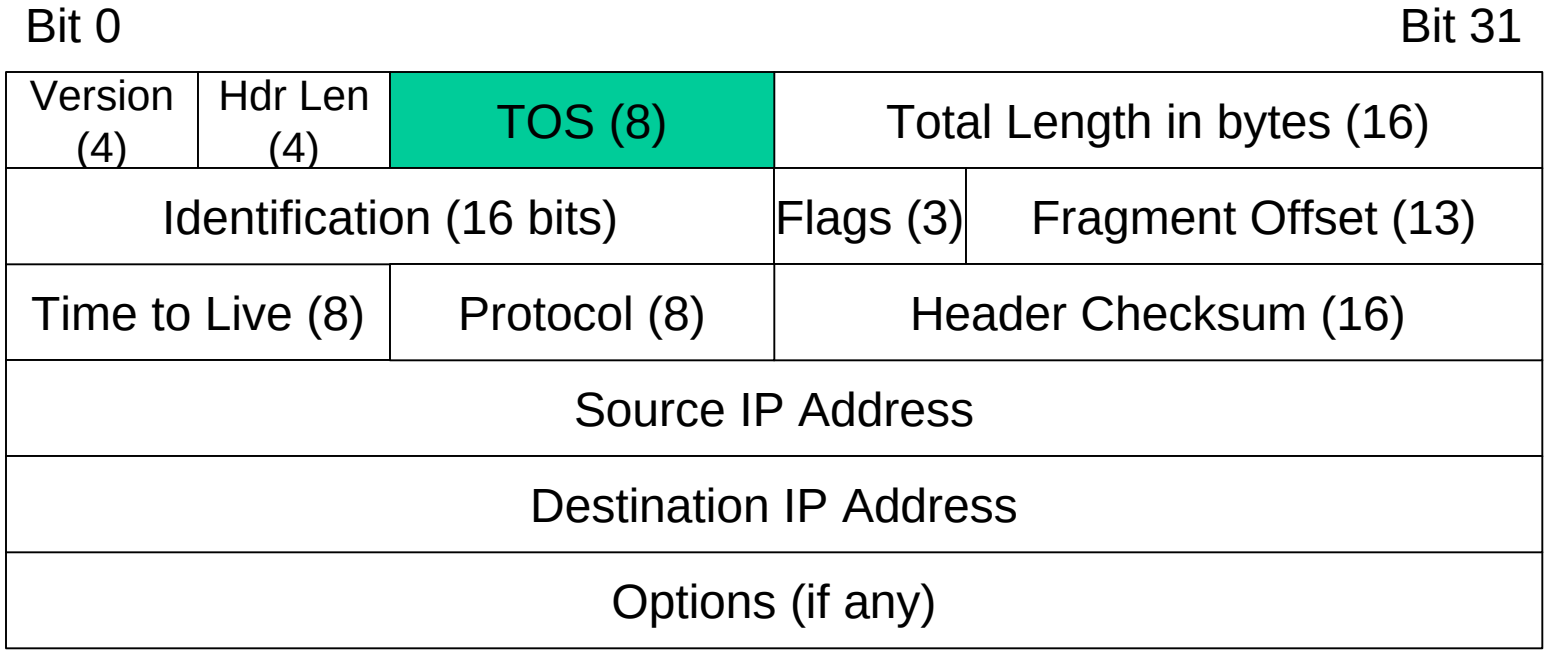
IP Packet Header

- Header Length (in 32 bit words)
  - Indicates end of header and beginning of payload
  - If no options, Header length = 5



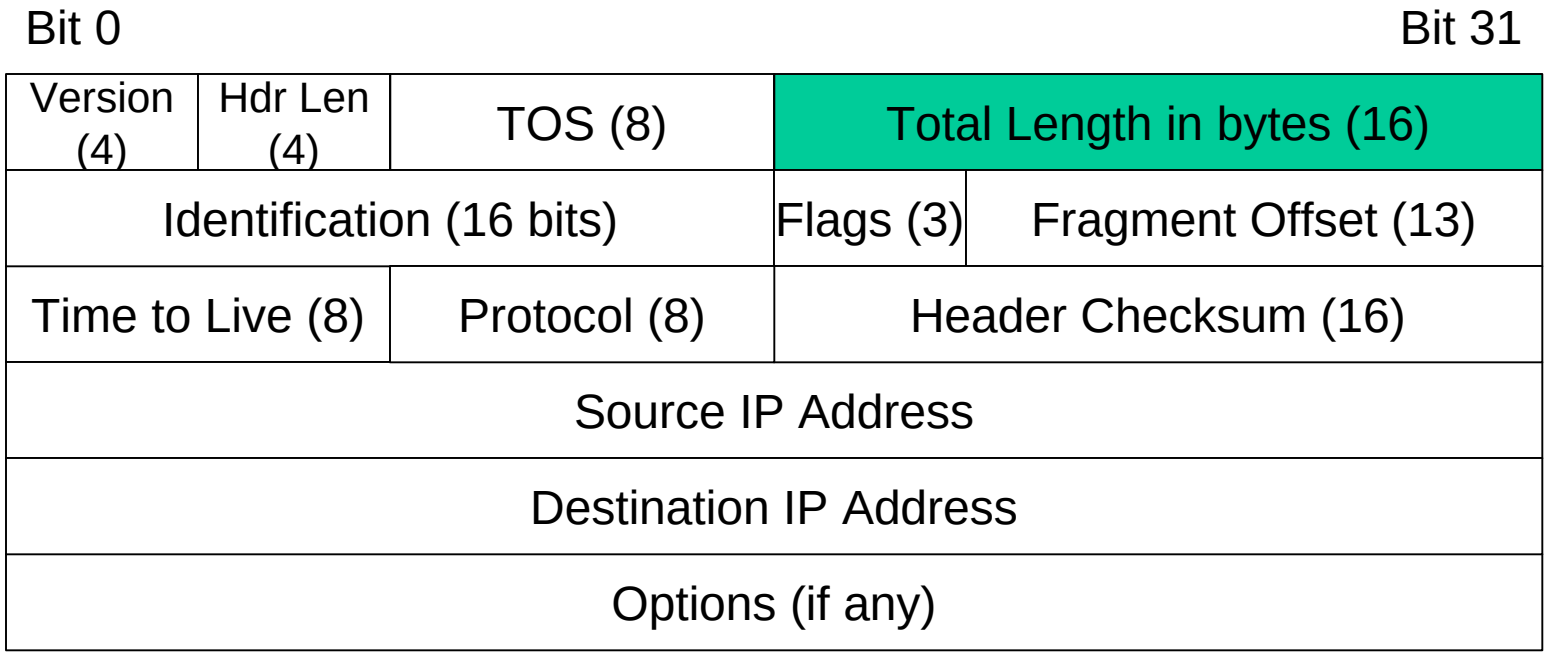
IP Packet Header

- Type of Service (TOS)
  - Allows different types of service to be requested
  - Initially, meaning was not well defined
  - Currently being defined (diffserv)



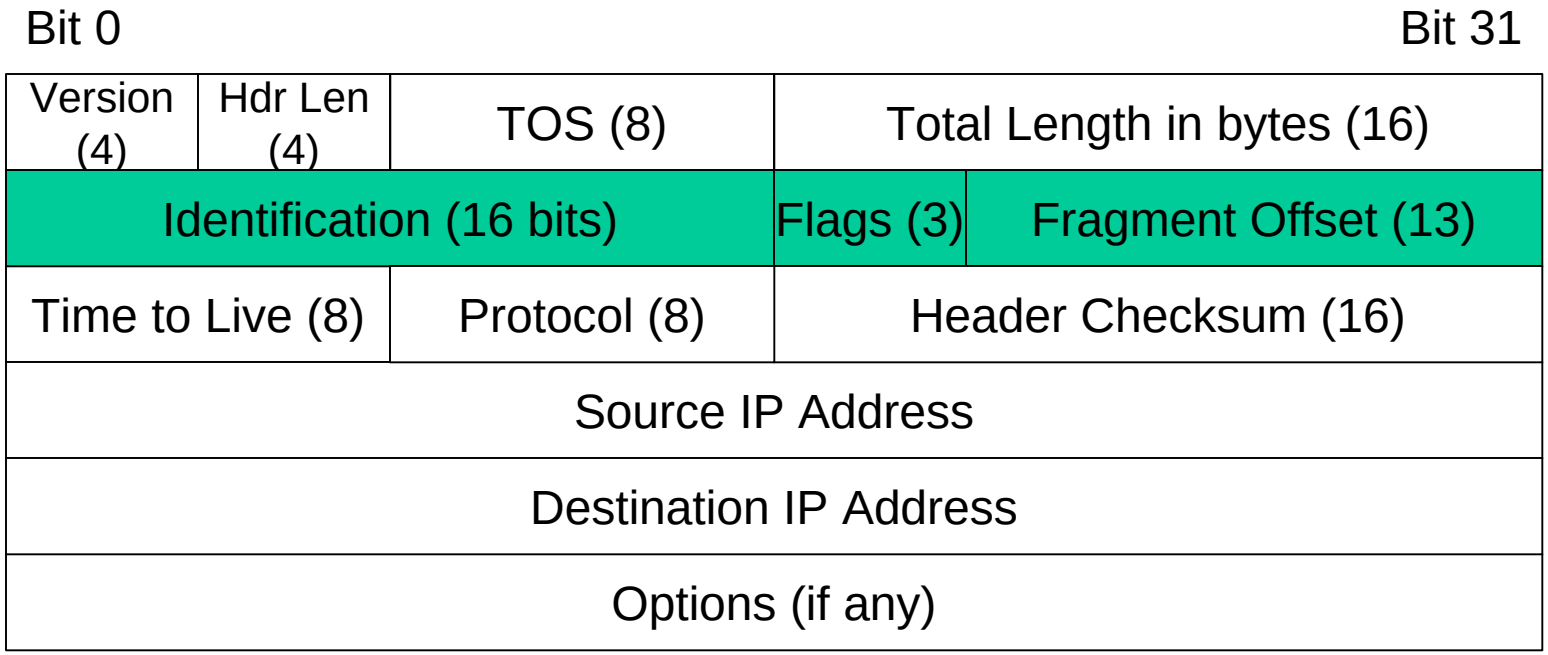
IP Packet Header

- Packet Length (in Bytes)
  - Unambiguously specify end of packet
  - Max packet size = 2^{16} = 65,535 Bytes



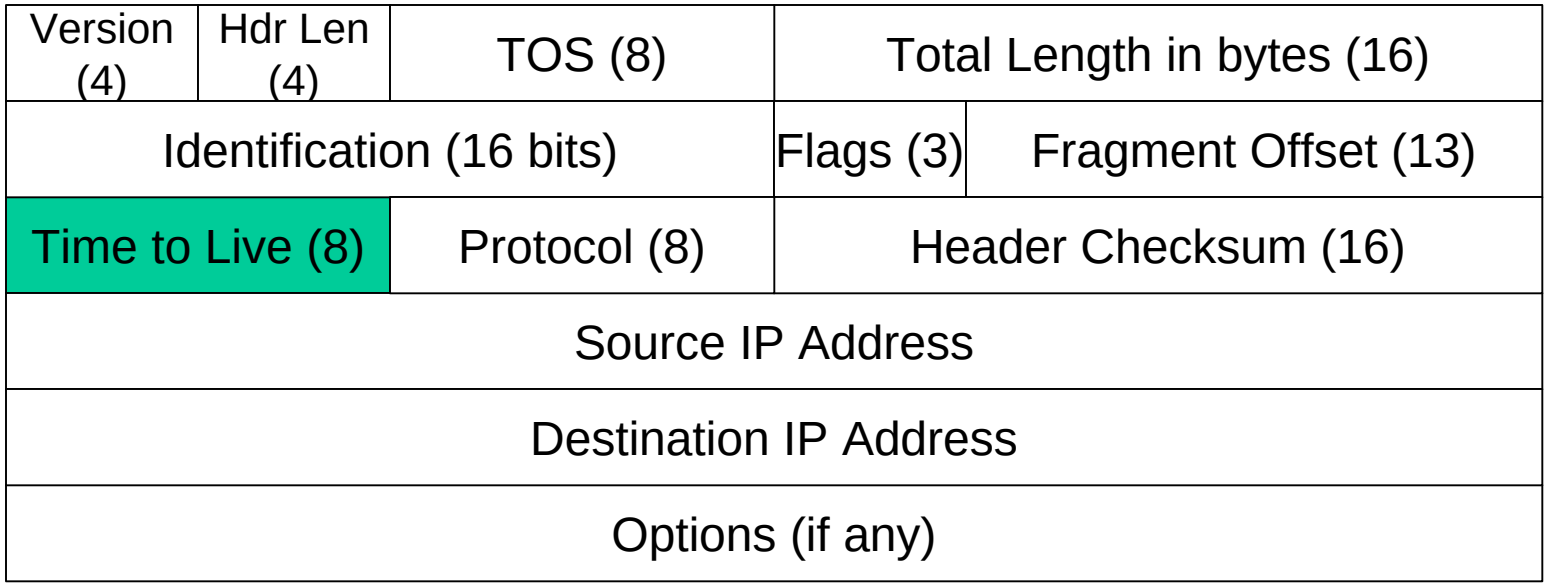
IP Packet Header

- These three fields for Fragmentation Control (will come back to them later)



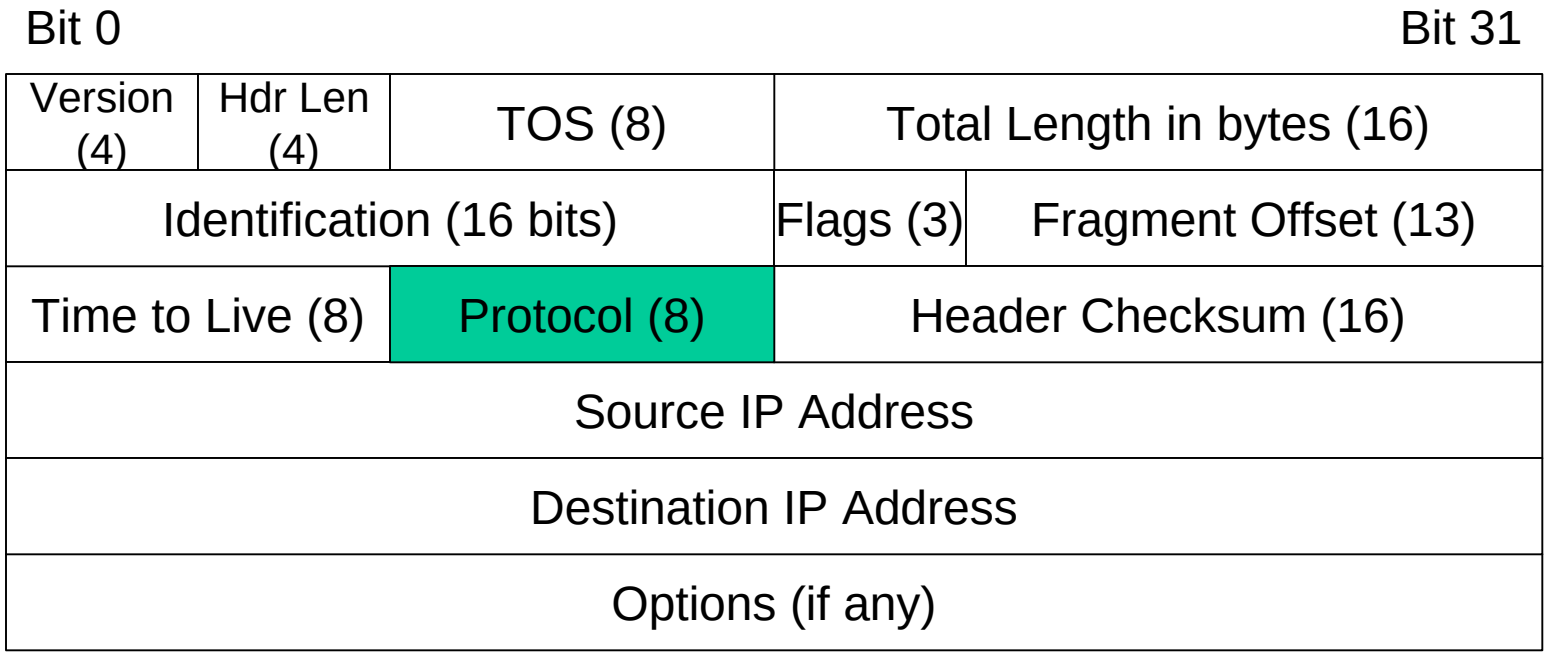
IP Packet Header

- Time to Live
  - Initially set by sender (up to 255)
  - Decremented by each router
  - Discard when TTL = 0 to avoid infinite routing loops



IP Packet Header

- Protocol
  - Value indicates what is in the data field
  - Example: TCP or UDP



IP Packet Header

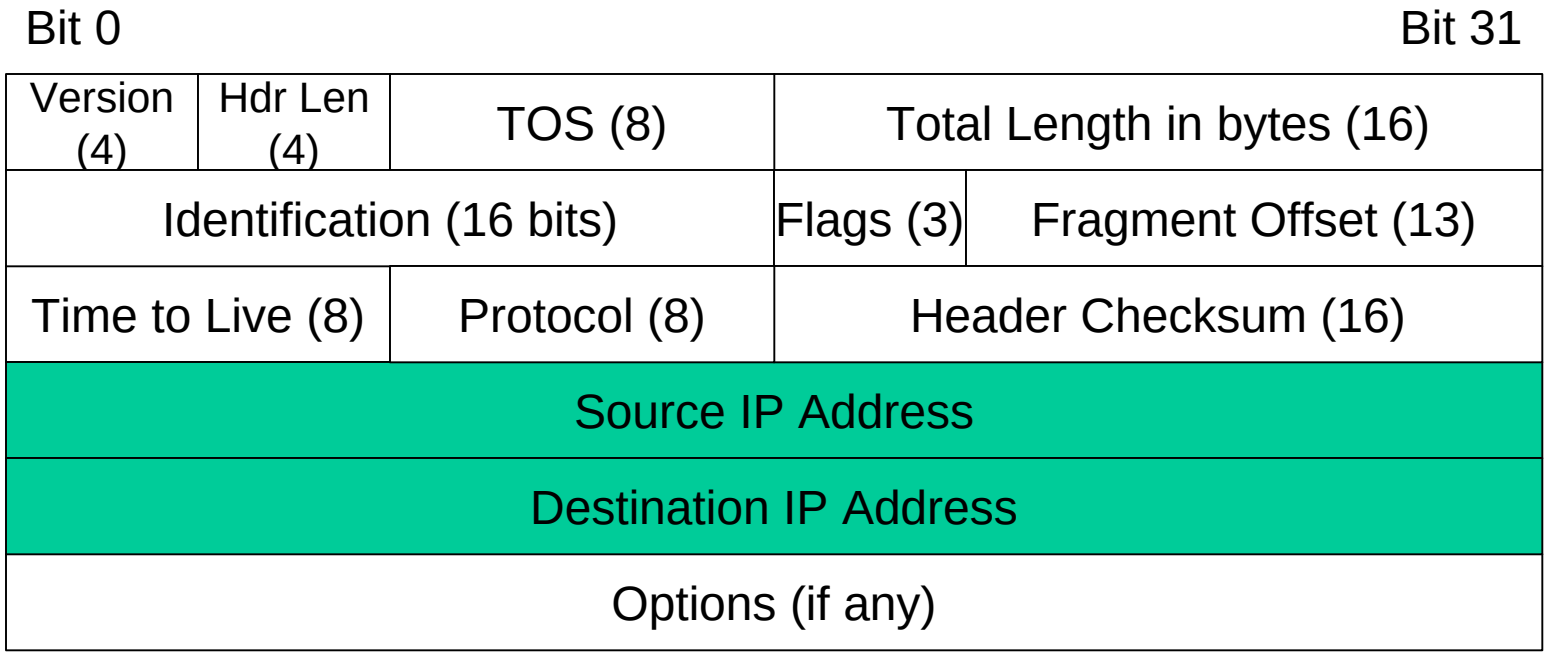
- Header Checksum
  - Checks for error in the header only
  - Bad headers can harm the network
  - If error found, packet is simply discarded

Bit 31

Version (4)	Hdr Len (4)	TOS (8)	Total Length in bytes (16)	
Identification (16 bits)			Flags (3)	Fragment Offset (13)
Time to Live (8)		Protocol (8)	Header Checksum (16)	
Source IP Address				
Destination IP Address				
Options (if any)				

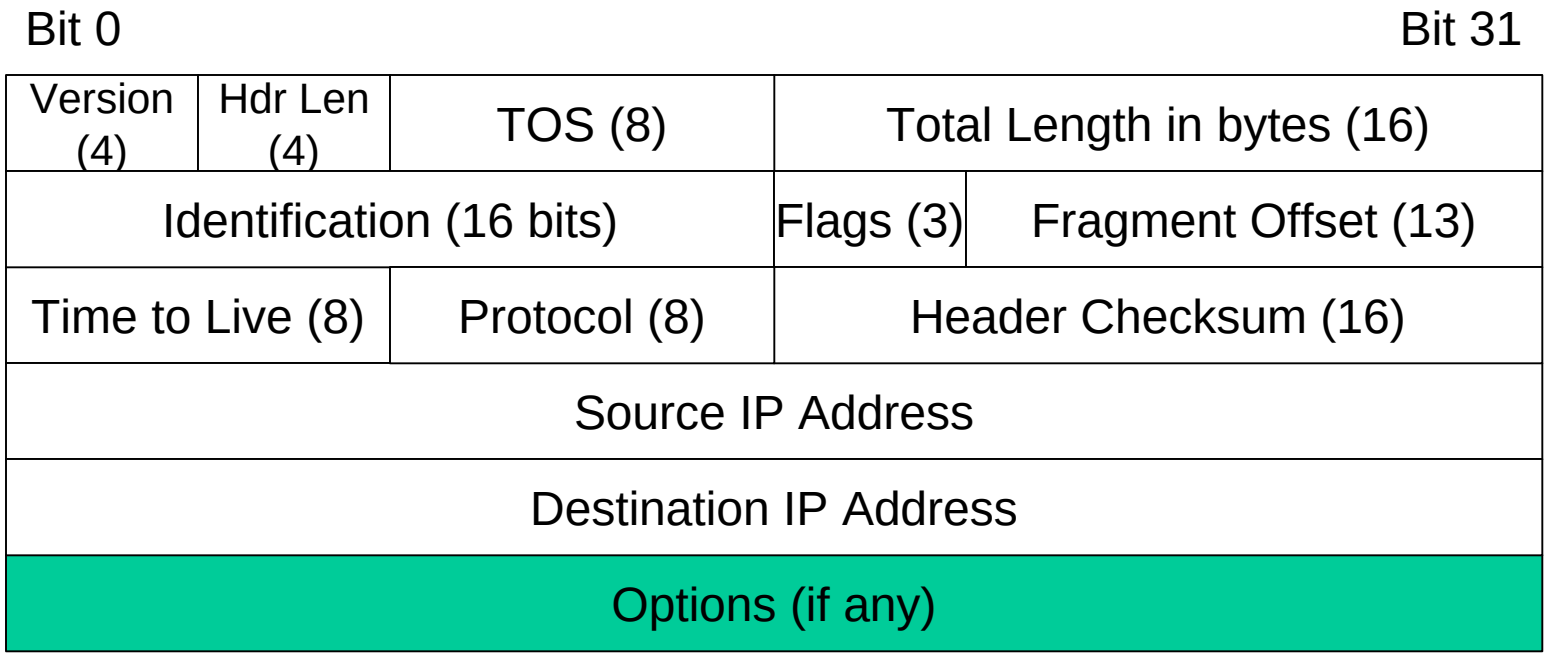
IP Packet Header

- Source and Destination IP Addresses
  - Strings of 32 ones and zeros



IP Packet Header

- Options
  - Example: timestamp, record route, source route



IP Fragmentation & Reassembly

- Maximum Transmission Unit (MTU)
  - Largest IP packet a network will accept
  - Arriving IP packet may be larger (max IP packet size = 65,535 bytes)
- Sender or router will split the packet into multiple fragments
- Destination will reassemble the packet
- IP header fields used to identify and order related fragments

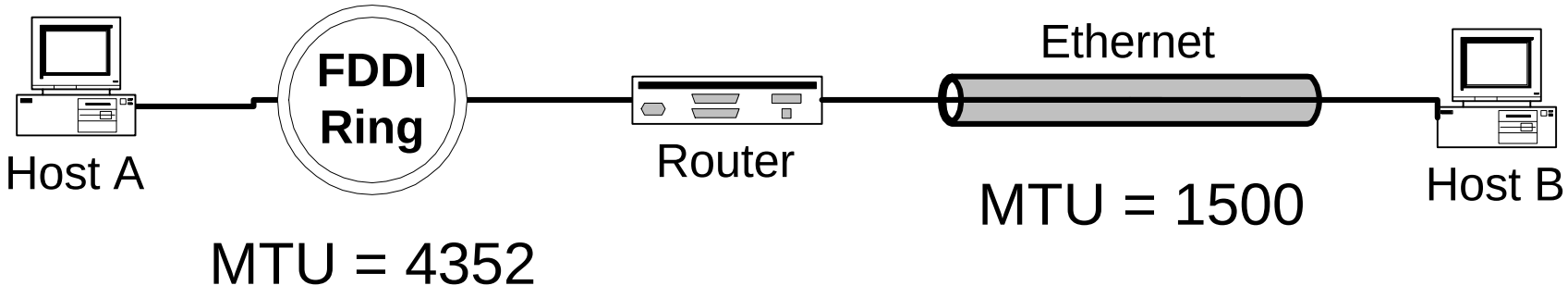
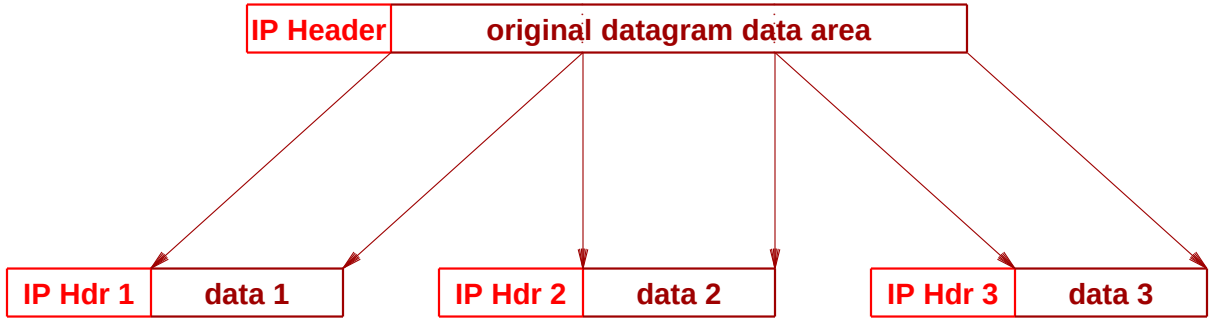


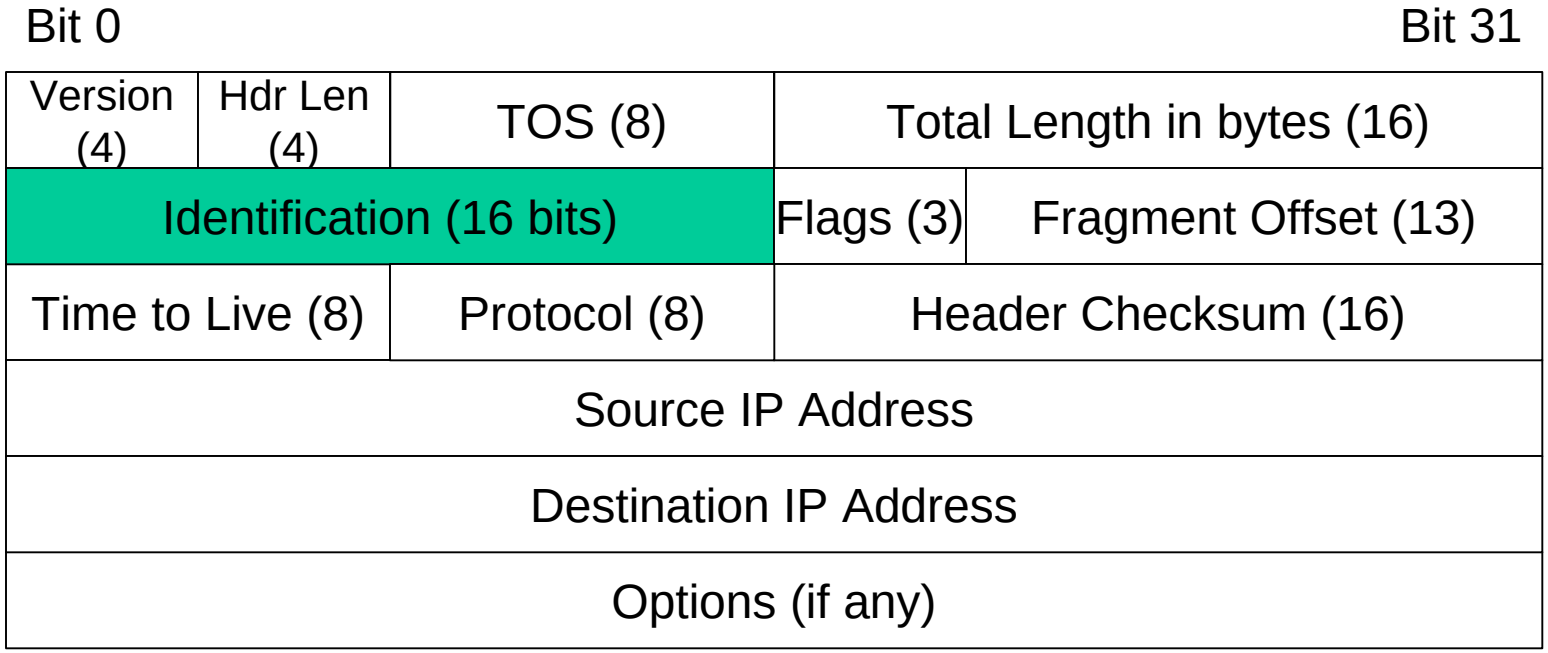
Illustration Of Datagram Fragmentation



- Each fragment has IP datagram header
- Header fields
  - Identify original datagram
  - Indicate where fragment fits

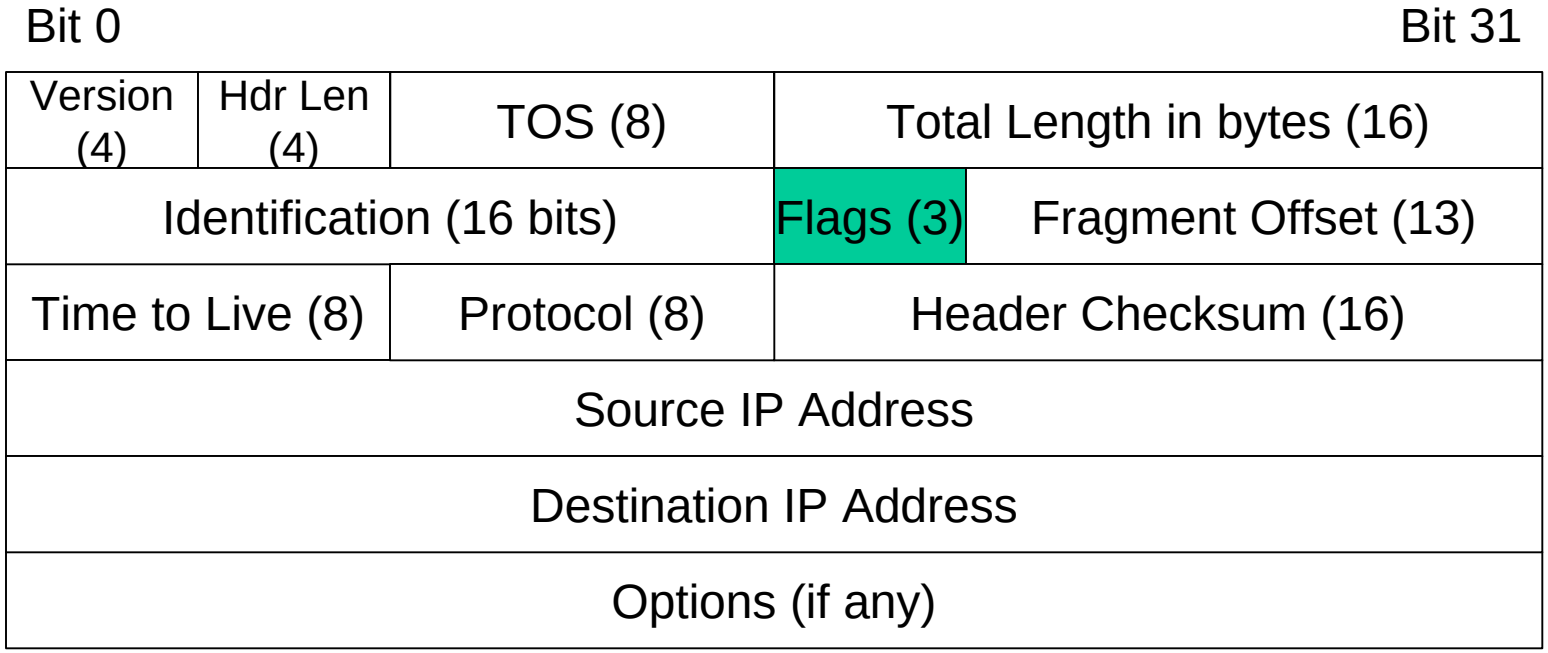
IP Packet Header

- Identification
  - All fragments of a single datagram have the same identification number



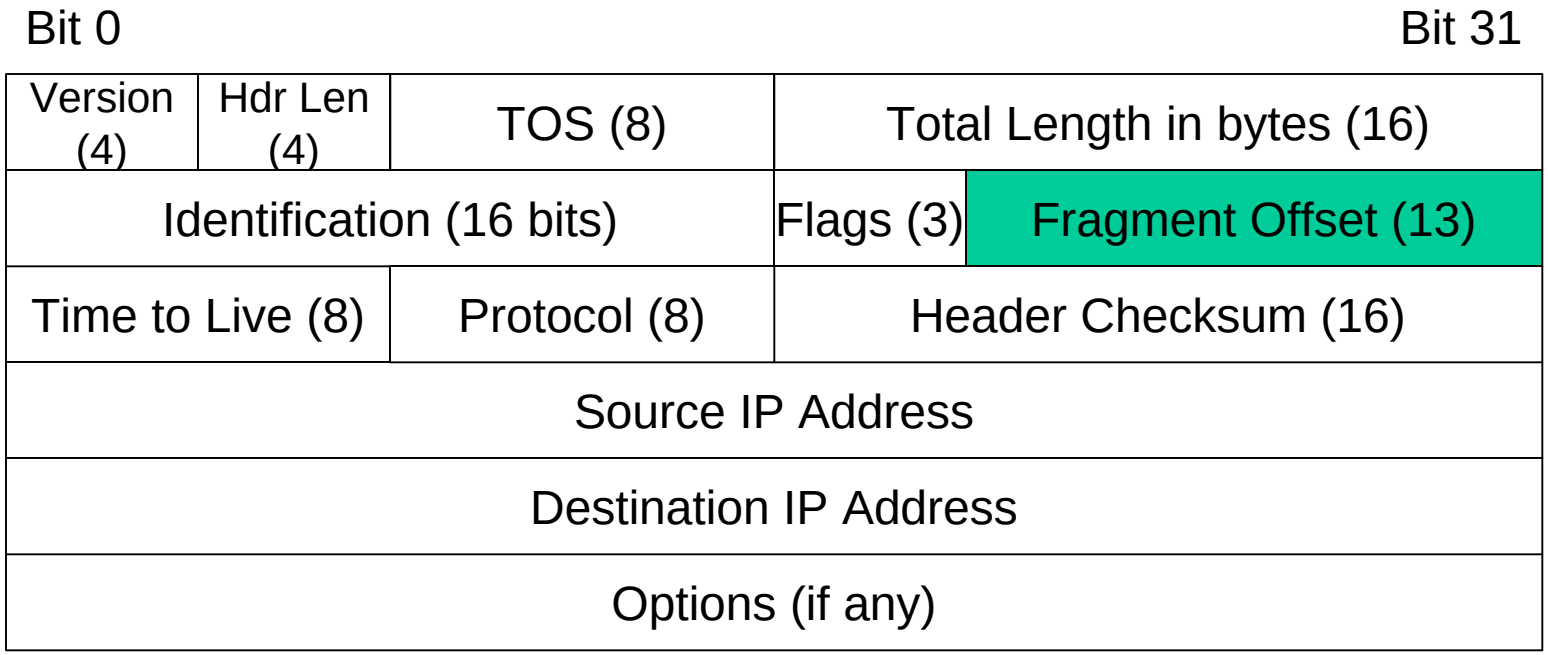
IP Packet Header

- Flags:
  - 1st bit: reserved, must be zero
  - 2nd bit: DF -- Do Not Fragment
  - 3rd bit: MF -- More Fragments



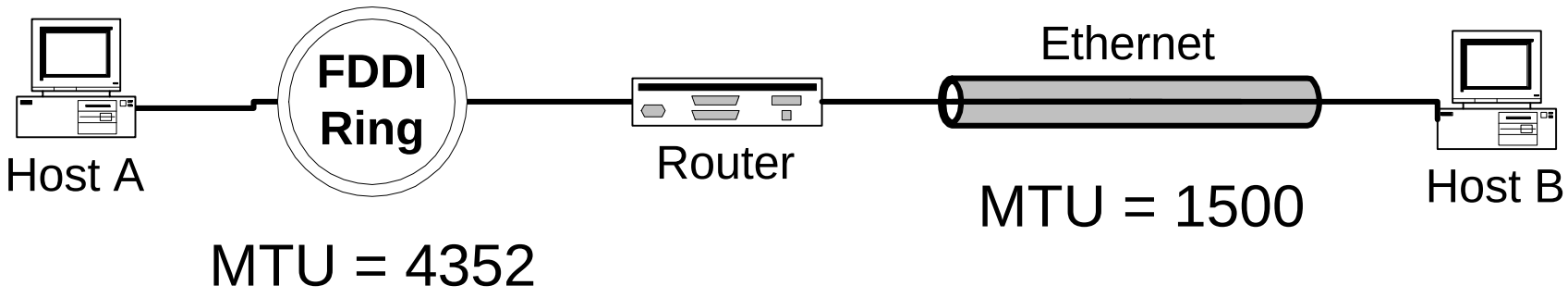
IP Packet Header

- Fragment Offset (in units of 8 bytes)
  - Used for reassembly of packet
  - 1st fragment has offset = 0

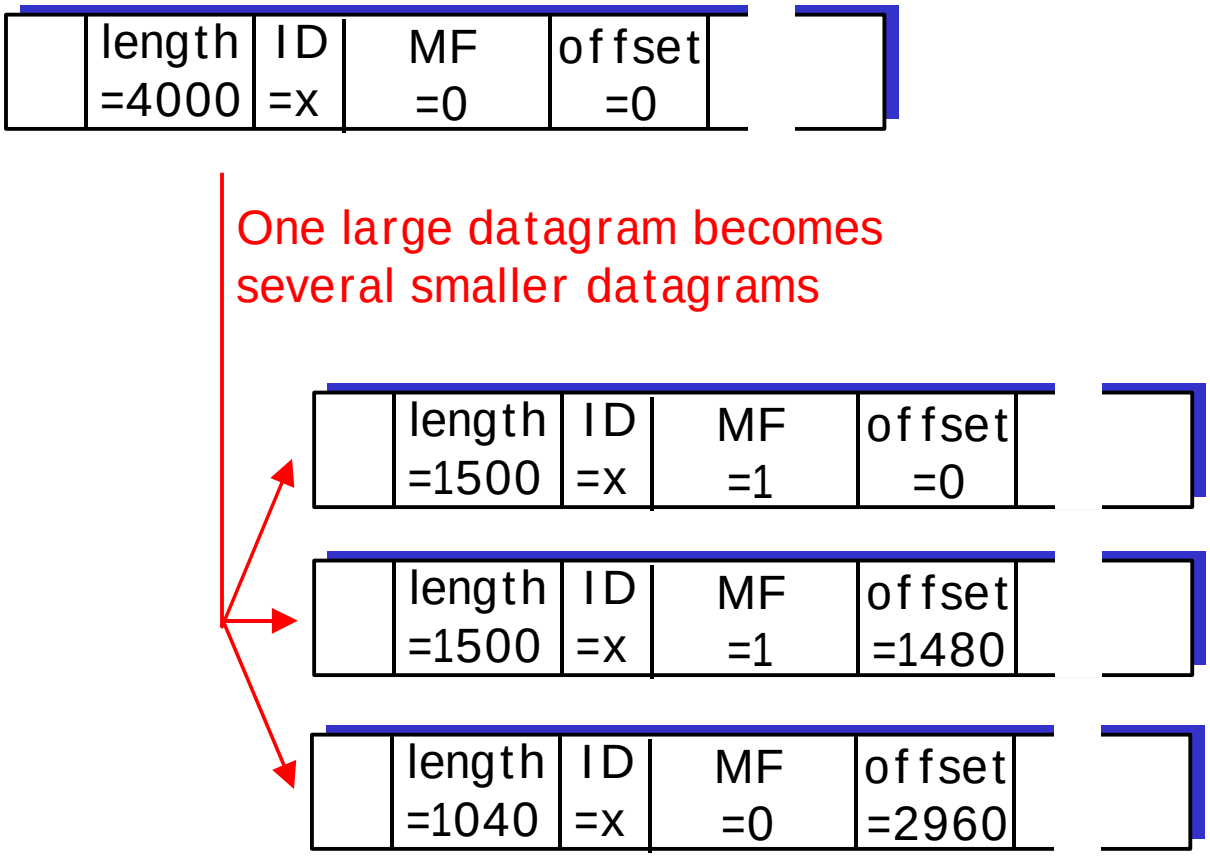


IP Fragmentation Example

- Host A wants to send to Host B an IP datagram of size = 4000 Bytes



IP Fragmentation Example



Multiple Fragmenting Points

- Let MTUs along internet path be
  - 1500
  - 1500
  - 1000
  - 1500
  - 576
  - 1500
- Result: fragmentation can occur twice

Fragmenting A Fragment

- Needed when fragment too large for network MTU
- Arbitrary subfragmentation possible
- Router divides fragments into smaller pieces
- All fragments at same ‘‘level’’
  - Offset given with respect to original datagram
  - Destination cannot distinguish subfragments

Fragment Loss

- Receiver
  - Collects incoming fragments
  - Reassembles when all fragments arrive
  - Does not know identity of router that did fragmentation
  - Cannot request missing pieces
- Consequence: Loss of one fragment means entire datagram lost