Internet Protocol (IP)

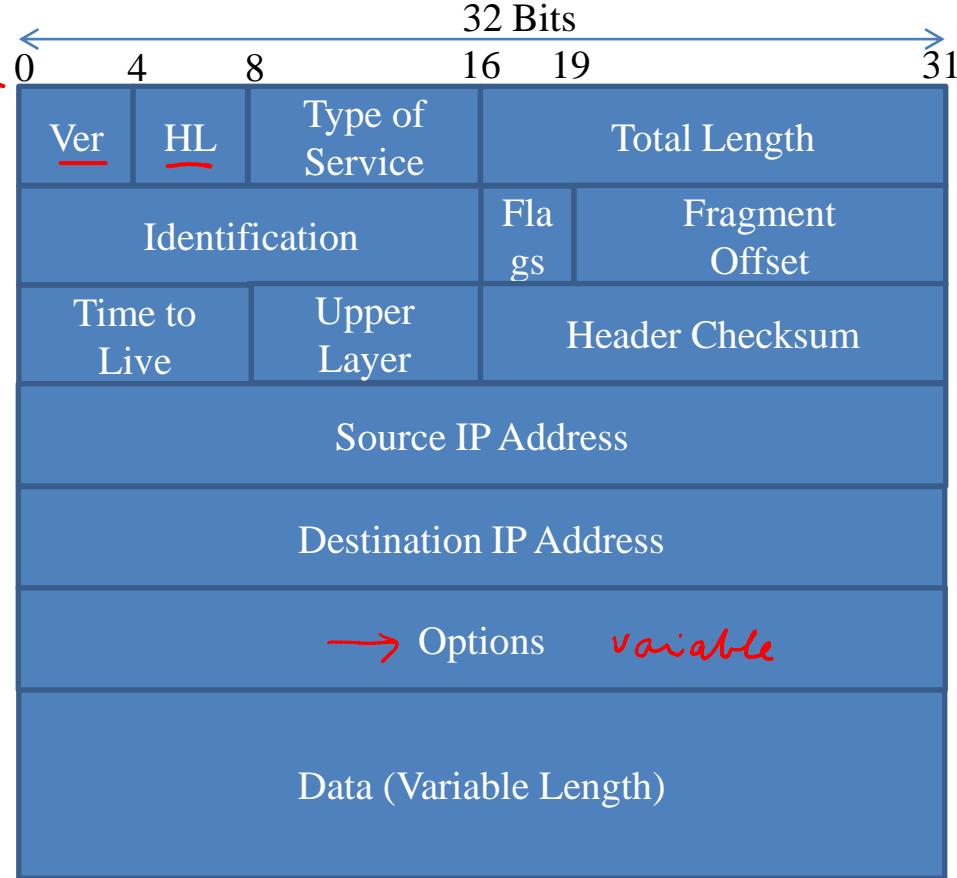Kameswari Chebrolu

Recap

- IP Protocol: Needed functionality

  - IP Protocol: Packet format, addressing

  - Forwarding

  - Routing

  - Error reporting and host signaling

- Focus: Packet Format and Fragmentation and re-assembly

Packet Format

- Version: Specifies the version of the protocol
  - IPv4, IPv6

- Header Length: Specifies the header in 32-bit words
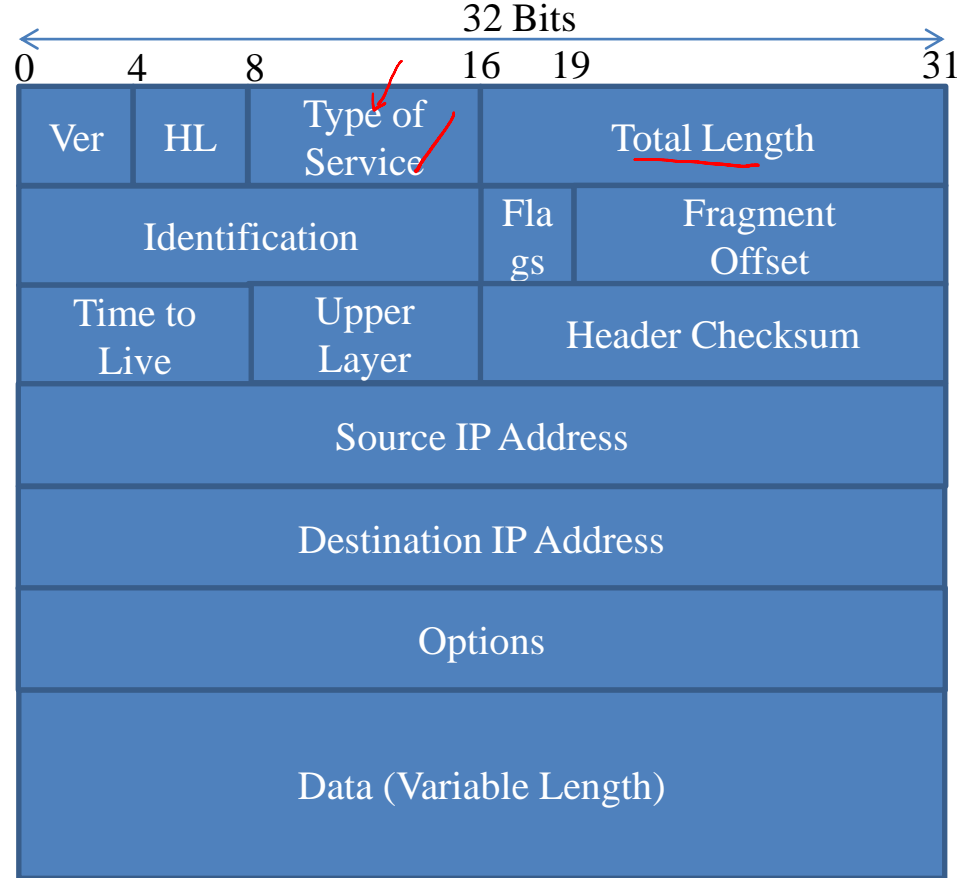 - 5 words (without options)

HL = 5

5 × 32
= 160 bits

32 Bits				
Ver	HL	Type of Service	Total Length	
Identification			Flags	Fragment Offset
Time to Live	Upper Layer		Header Checksum	
Source IP Address				
Destination IP Address				
Options				variable
Data (Variable Length)				

0   4   8   16   19   31

Packet Format

- Type of Service: Permits packets to be treated differently

  – Research Focus

- Total Length: Specifies the length of the datagram (in bytes) including header
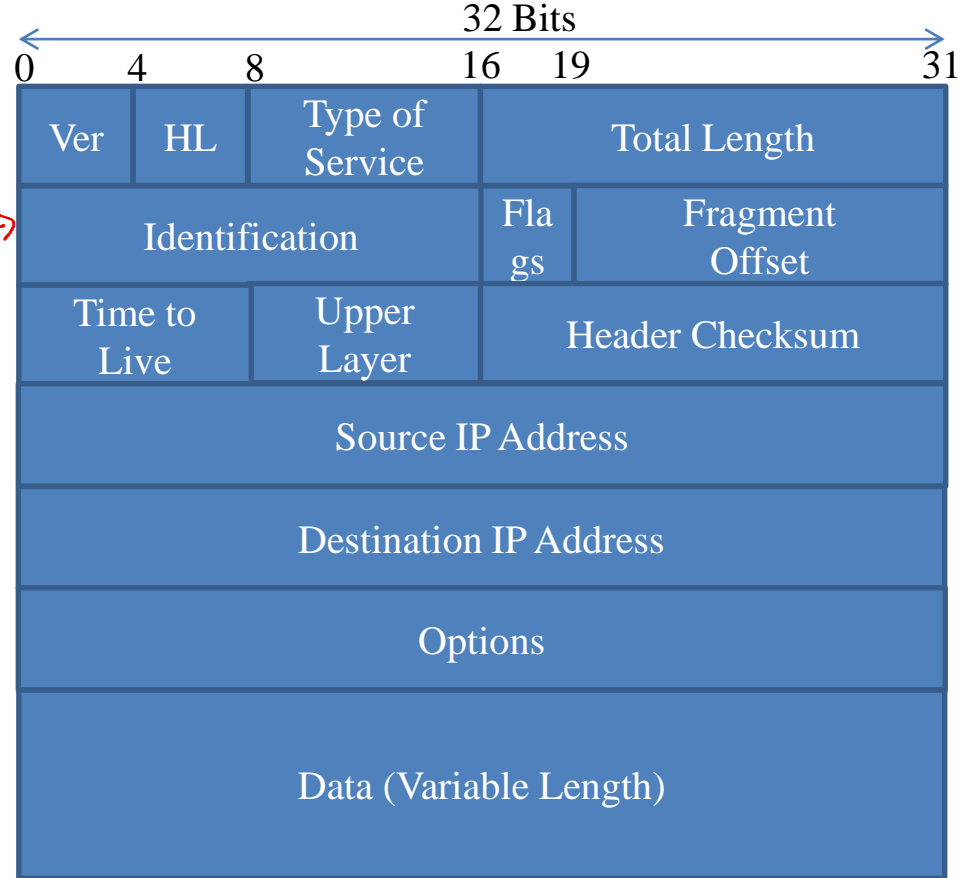
32 Bits

0	4	8	16	19	31
Ver	HL	Type of Service		Total Length	
Identification			Flags	Fragment Offset	
Time to Live		Upper Layer	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Packet Format

$(2^{14}-1)$ bytes
16

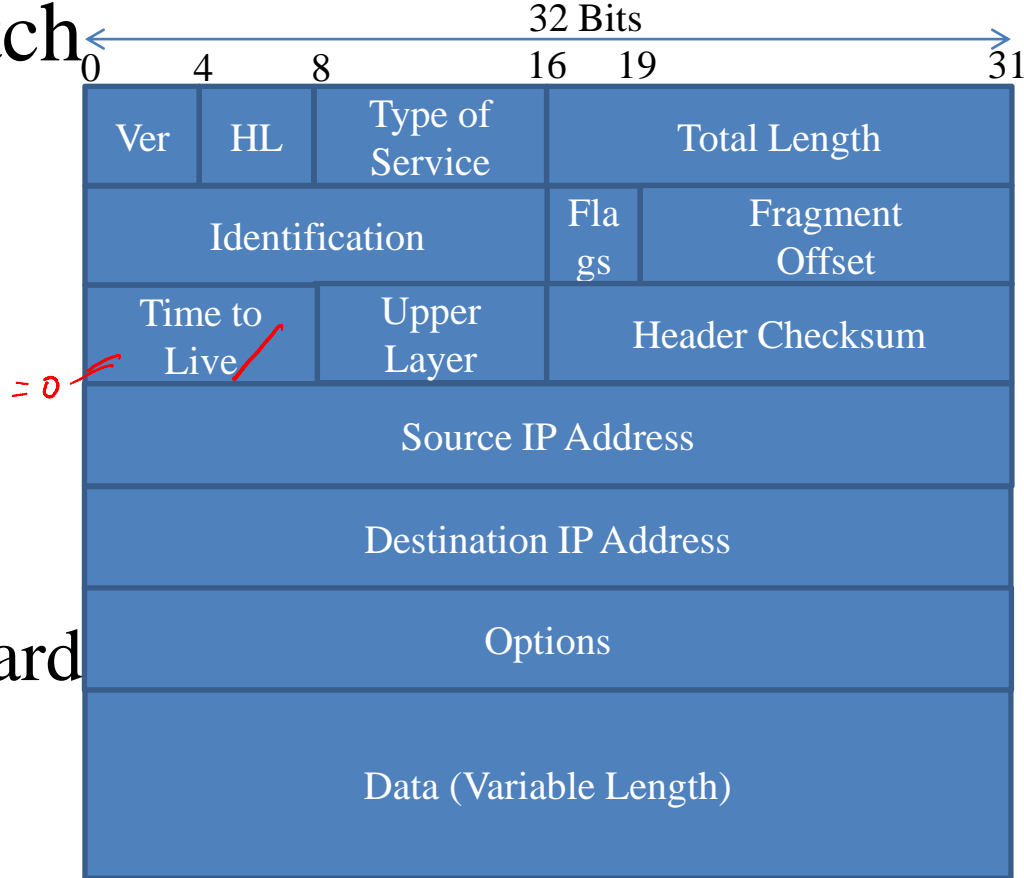- Identification/Flags/ Fragment Offset:

  - Max size of IP packet is 65535 Bytes

  1500 bytes

  - Physical Networks may not support large packets
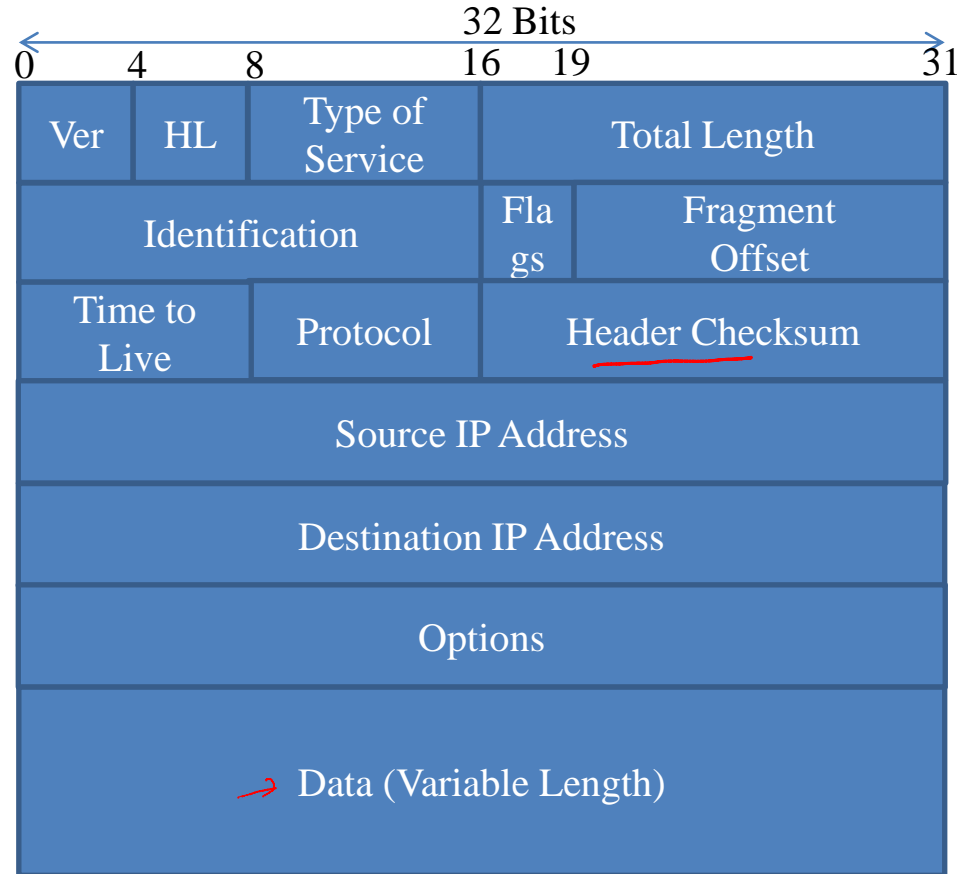
  - Need Fragmentation and reassembly (more on it soon)

32 Bits

0	4	8	16	19	31

Ver	HL	Type of Service		Total Length	
Identification			Flags	Fragment Offset	
Time to Live		Upper Layer	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Packet Format

- Time to Live: Helps catch packets doing rounds

  expiry date

  - Not really time but ~~hop~~ *x sec* count

  - Routers decrement the field by one before forwarding; if zero discard

  - Default value = 64

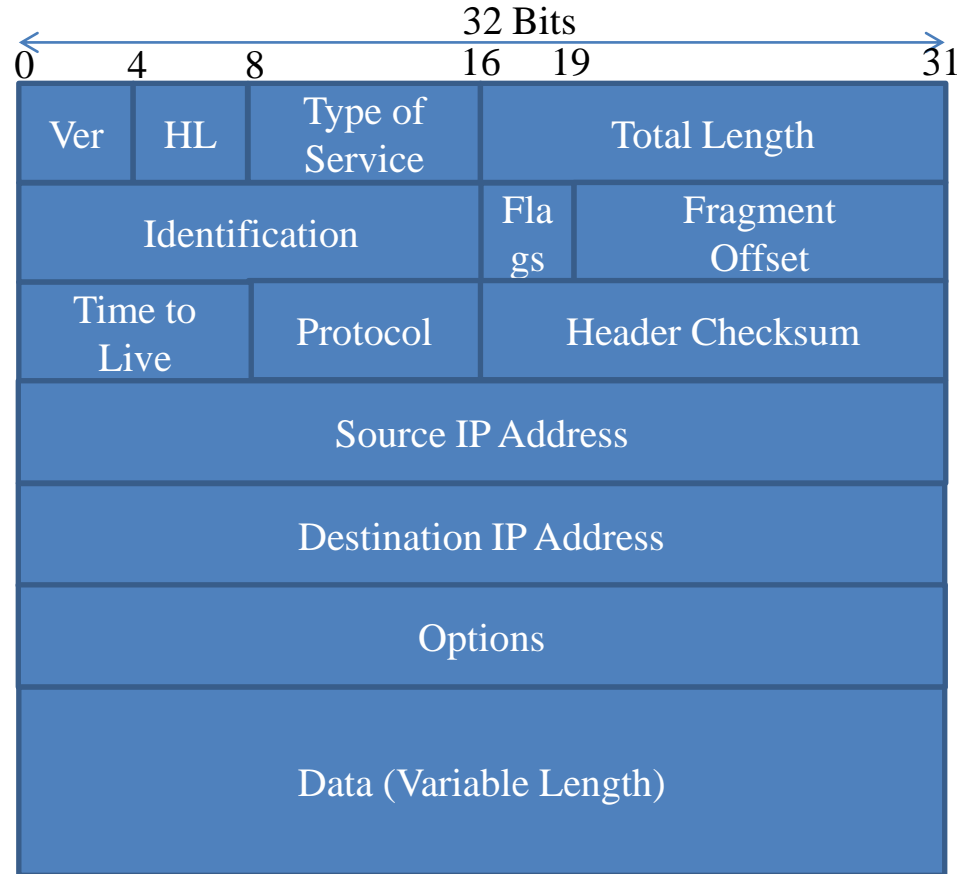| 32 Bits						
0	4	8	16	19		31
Ver	HL	Type of Service	Total Length			
Identification			Flags	Fragment Offset		
Time to Live *= 0*		Upper Layer	Header Checksum			
Source IP Address						
Destination IP Address						
Options						
Data (Variable Length)						

Packet Format

- Protocol: Demux key that identifies higher layer protocol
  - TCP: 6, UDP: 17
- Checksum (Internet): Detects errors in header

			32 Bits		
0	4	8	16	19	31

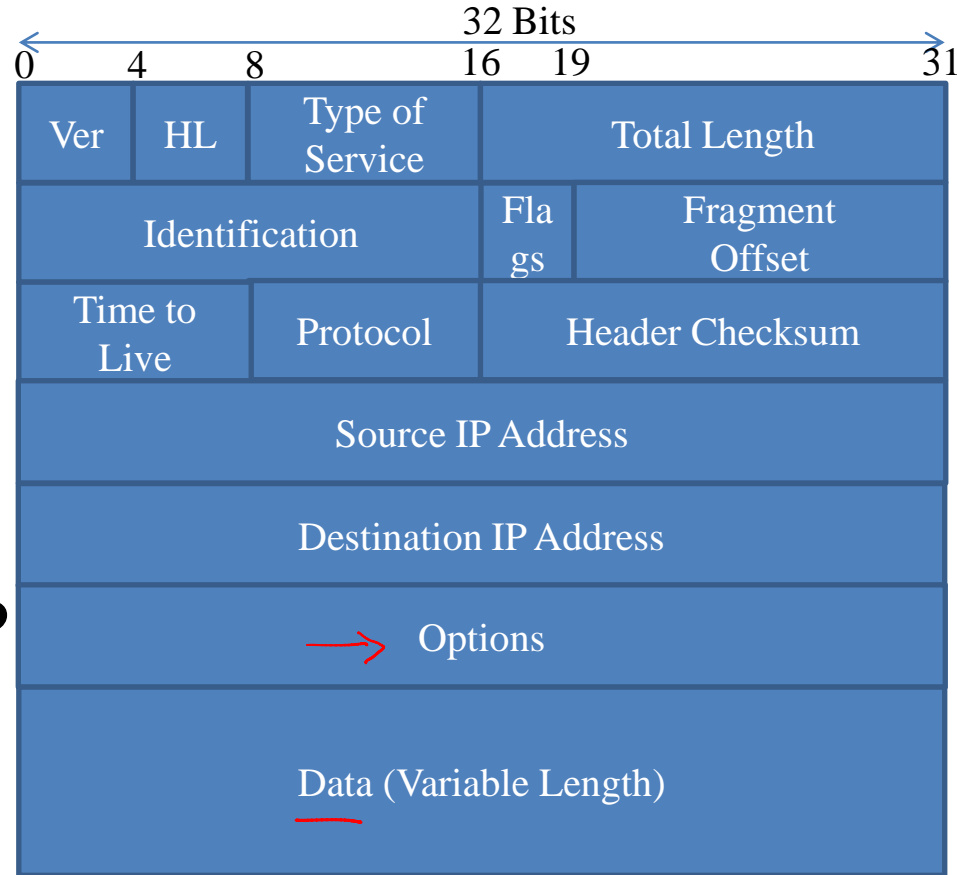| Ver | HL | Type of Service | Total Length | | |
Identification			Flags	Fragment Offset	
Time to Live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Packet Format

- Source/Destination IP address: 32-bit

  - Destination key to forwarding

  - Source for replying back

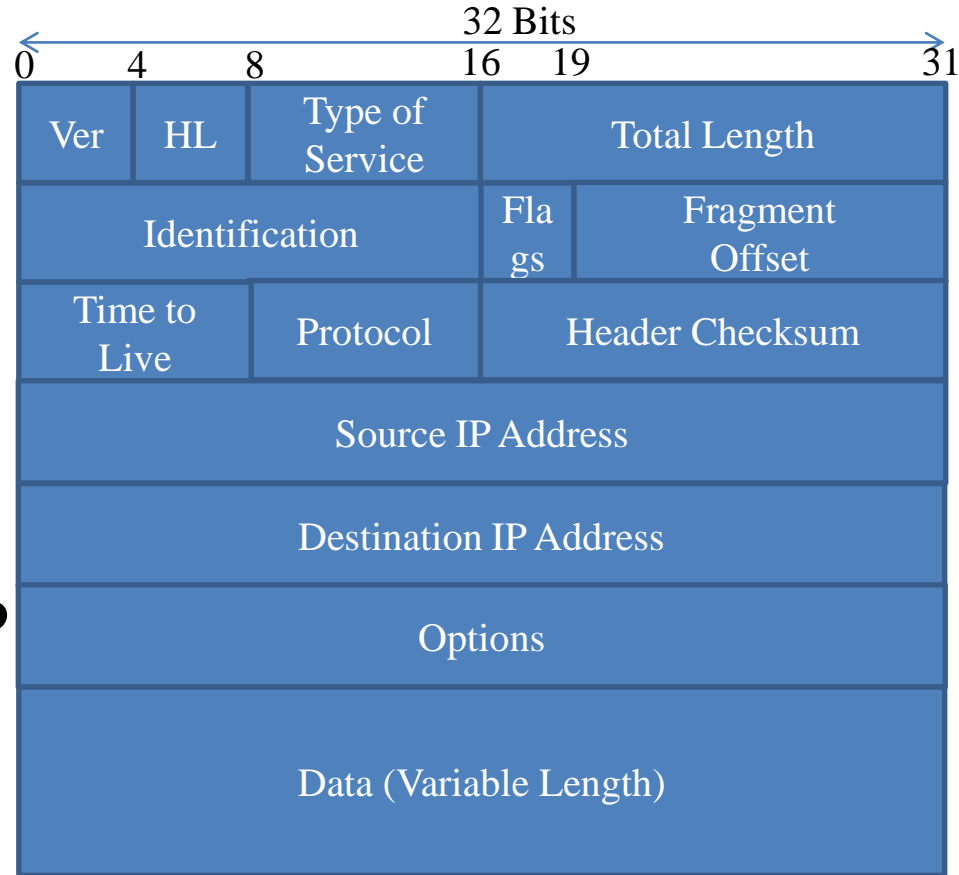 - Global address space, independent of physical network address (MAC)

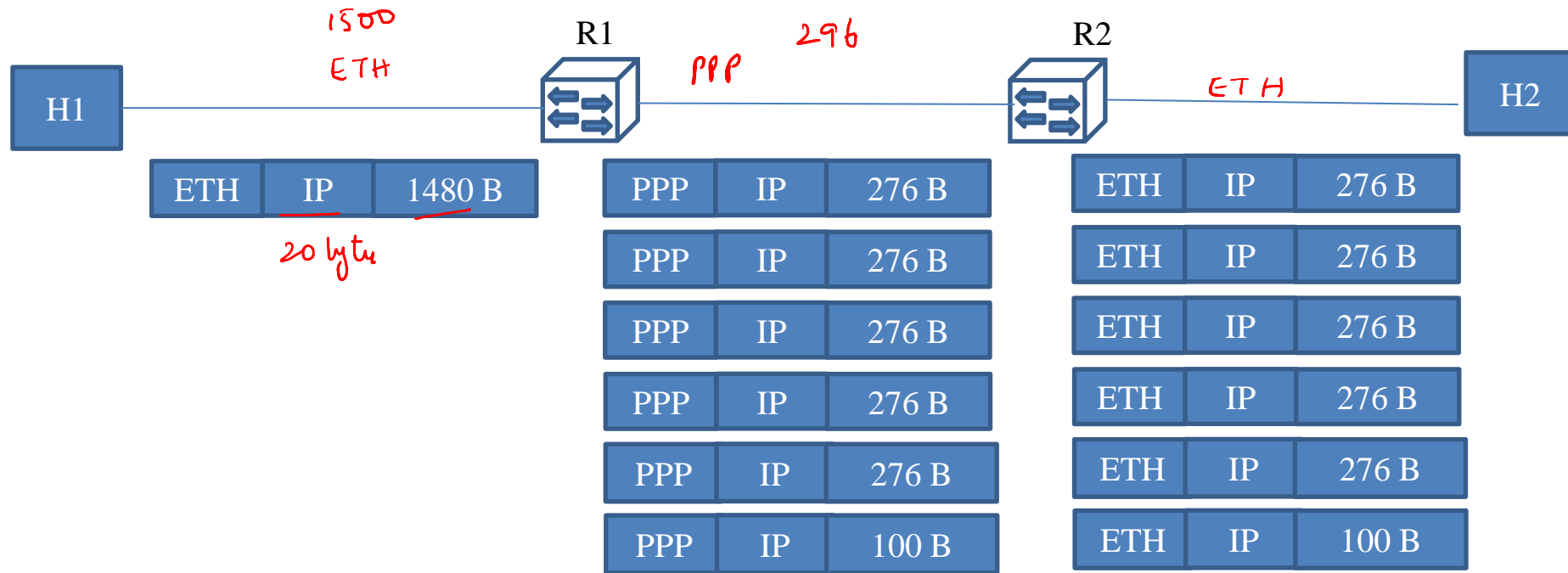| 32 Bits					
Ver	HL	Type of Service	Total Length		
Identification			Flags	Fragment Offset	
Time to Live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Packet Format

- Options: Rarely used
  - Record Time stamp
  - Record route taken
  - Specify source route
- Data/Payload: Higher Layer Data (TCP or UDP segment)

32 Bits

0	4	8	16	19	31

Ver	HL	Type of Service	Total Length		
Identification			Flags	Fragment Offset	
Time to Live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Packet Format

- Options: Rarely used
  - Record Time stamp
  - Record route taken
  - Specify source route

- Data/Payload: Higher Layer Data (TCP or UDP segment)

32 Bits					
Ver	HL	Type of Service	Total Length		
Identification			Flags	Fragment Offset	
Time to Live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
Options					
Data (Variable Length)					

Fragmentation and Reassembly

- Goal: Interconnect heterogeneous networks

- Problem: Each technology has different Maximum Transmission Unit (MTU) size

  - MTU: Largest IP datagram that can be carried in a frame

  - E.g. Ethernet: 1500, FDDI: 4352, PPP: 296 (Negotiable), WiFi: 7981

- At host: Select MTU of link it is connected to

- At intermediate router: Forward datagram on a network with smaller MTU

  - Need to fragment the datagram

- Where to reassemble?

  - Next hop router? Increases overhead and datagram may again be fragmented

  - Destination is the best place

ETH	IP	1480 B

1500
ETH
20 bytes

PPP	IP	276 B
PPP	IP	276 B
PPP	IP	276 B
PPP	IP	276 B
PPP	IP	276 B
PPP	IP	100 B

296
PPP

ETH	IP	276 B
ETH	IP	276 B
ETH	IP	276 B
ETH	IP	276 B
ETH	IP	276 B
ETH	IP	100 B

ETH

Note: Above values not true in practice due to Offset field having to be a multiple of 8

Fragmentation Fields

- Identification: Helps identify a datagram

  – All fragments carry same identification

- Flags: 3 bits

Identification	Flags	Fragment Offset

  – bit 0: Reserved, set to zero

  – bit 1: Don't Fragment (DF); Useful for path MTU discovery

  – bit 2: More Fragments (MF); Set to one to indicate more fragments to follow

Fields

- Fragmentation Offset: 13 bits long

  - Measures data/payload in units of eight-byte blocks

  - For a particular fragment, offset specifies start of data relative to the beginning of the original un-fragmented IP datagram.

    - E.g first fragment would have an offset of zero

Identification	Flags	Fragment Offset

Example

8 bytes

- Original datagram: 1500B
  - Data within is <u>1480B</u>

- MTU: <u>296B</u>   276
  - Max Data within is 276B
  - Offset has to be multiple of 8 → Data within 272B

→ TCP/UDP   20B

272 x 5 = 1360   1480 - 1360 = 120

Original Datagram

Length=1500	ID=x	Fragflag=0	Offset=0

Fragmented Datagrams

Length=292	ID=x	Fragflag=1	Offset=0
Length=292	ID=x	Fragflag=1	Offset=34
Length=292	ID=x	Fragflag=1	Offset=68
Length=292	ID=x	Fragflag=1	Offset=102
Length=292	ID=x	Fragflag=1	Offset=136
Length=140	ID=x	Fragflag=0	Offset=170

20

272 +20

272/8

Summary

- Looked at IP packet format

- Looked at fragmentation and reassembly

- Next: Addressing and Forwarding