פרק 9 חלק א' – TCP/IP

- Definitions
- Network Layer (IP) services – 3rd Layer
- IPv4 Address Structure
- Private versus Global IP Addresses
- NAT- Network Address Translation
- IP Header Structure
- MTU and Fragmentation process
- ARP – Address Resolution Protocol
- DHCP –Dynamic Host Configuration Protocol
- Transpiration Layer: UDP, TCP, SCTP

Definitions

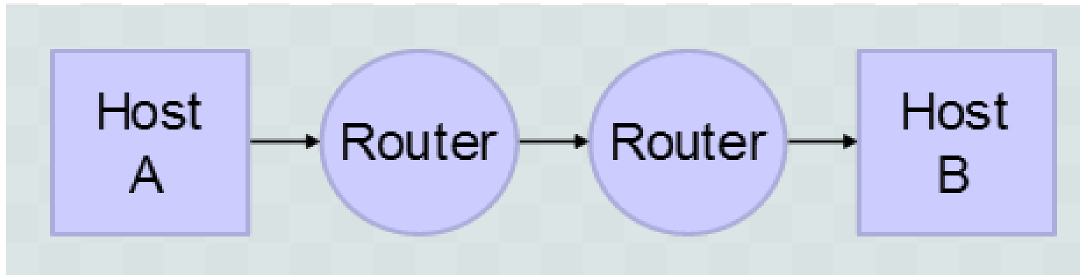
- The term internet is short for “internetworking”
  - interconnection of networks with different network access mechanisms, addressing, different routing techniques, etc
- An internet
  - Collection of communications networks interconnected by layer 3 switches and/or routers
- IP (Internet Protocol)
  - A set of rules to send and receive messages at the Internet address level
  - Most widely used internetworking protocol
  - IP provides connectionless (datagram) service
  - Each packet treated separately
  - Network layer protocol

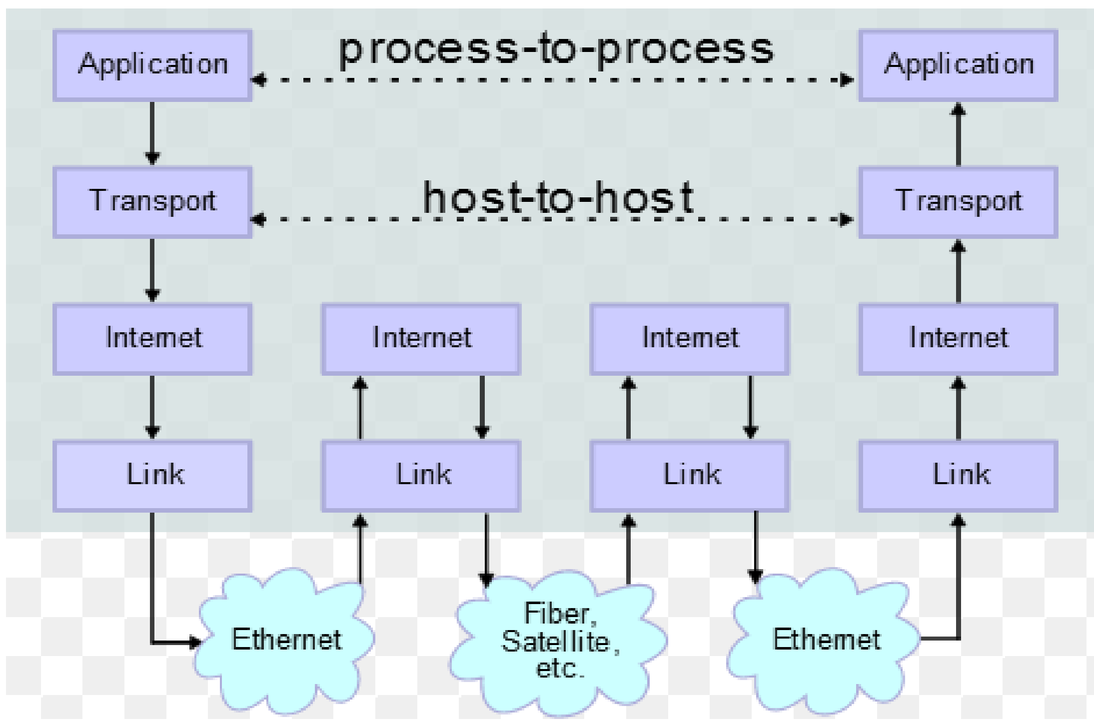
Connectionless Internetworking

- Advantages
  - Flexible and robust
    - In case of congestion or node failure, packets find their way easier than connection-oriented services
  - No unnecessary overhead for connection setup
  - Can work with different network types

- Disadvantage: Unreliable
  - No guarantee of delivery
  - Not guarantee of packets order
    - Packets can take different routes
  - Next up layer is responsible for Reliability (for example:TCP)

IP Communication – Example



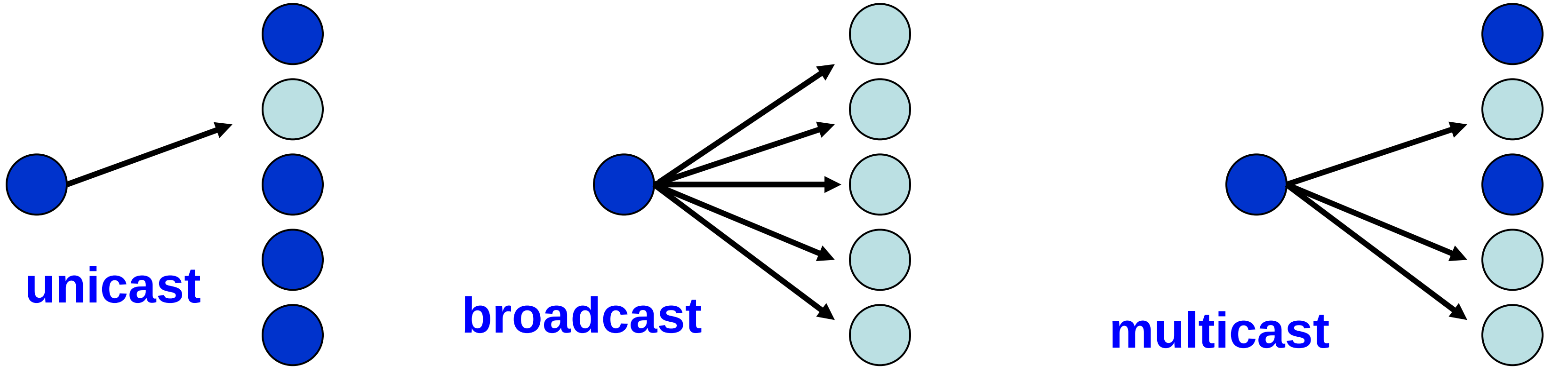


IP Services

- IP provides an unreliable connectionless best effort service (also called: “datagram service”)
  - Unreliable: IP does not make an attempt to recover lost packets
  - Connectionless: Each packet (“datagram”) is handled independently. IP is not aware that packets between hosts may be sent in a logical sequence
  - Best effort: IP does not make guarantees on the service (no throughput guarantee, no delay guarantee,…)
- Consequences
  - Higher layer protocols have to deal with losses or with duplicate packets
  - Packets may be delivered out-of-sequence

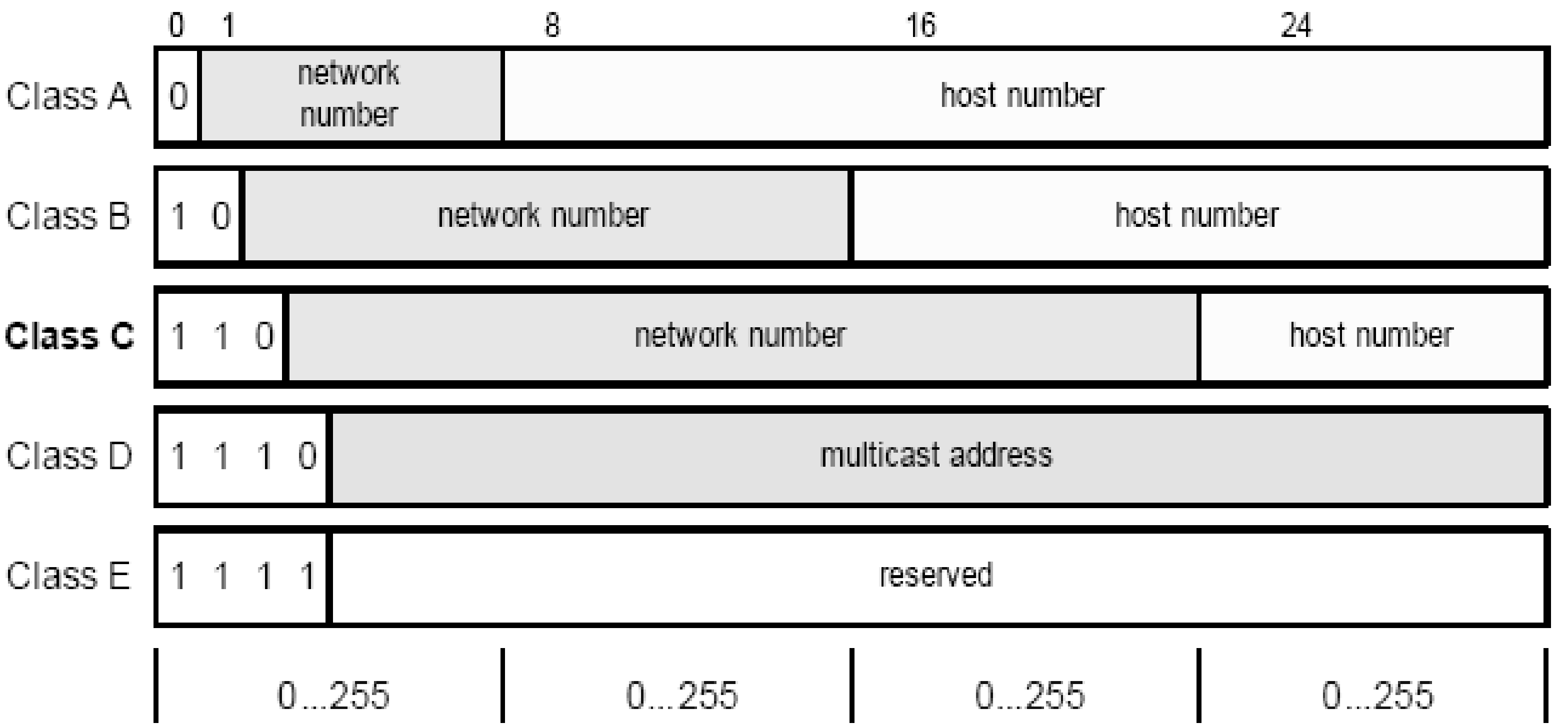
IP Services (Cont.)

- IP supports the following services:
  - one-to-one (**unicast**)
  - one-to-all (**broadcast**)
  - one-to-several (**multicast**)



IPv4 Address Structure

IP Address = <network number><host number>
Initially, 5 Classes of IP addresses where defined



IPv4 Address Classes

Address Class	1st octet range (decimal)	1st octet bits	Network(N) and Host(H) parts of address	Default subnet mask (decimal and binary)	Number of possible networks and hosts per network
A	1-127**	00000000-01111111	N.H.H.H	255.0.0.0	128 nets (2^7) 16,777,214 hosts per net (2^24-2)
B	128-191	10000000-10111111	N.N.H.H	255.255.0.0	16,384 nets (2^14) 65,534 hosts per net (2^16-2)
C	192-223	11000000-11011111	N.N.N.H	255.255.255.0	2,097,150 nets (2^21) 254 hosts per net (2^8-2)
D	224-239	11100000-11101111	NA (multicast)		
E	240-255	11110000-11111111	NA (experimental)		

Green bits do not change

IPv4 Address Structure

125.31.137.33

01111101 00011111 10001001 00100001

32 bits - Decimal Notation

Each node has at least one IP address on each one of its interfaces

NAT- Network Address Translation (RFC 2663)

- A method of remapping one IP address space into another by modifying network address information

- Challenge - IPv4 address exhaustion
  - Solution - sharing one Internet-routable IP address of a NAT gateway for an entire private network

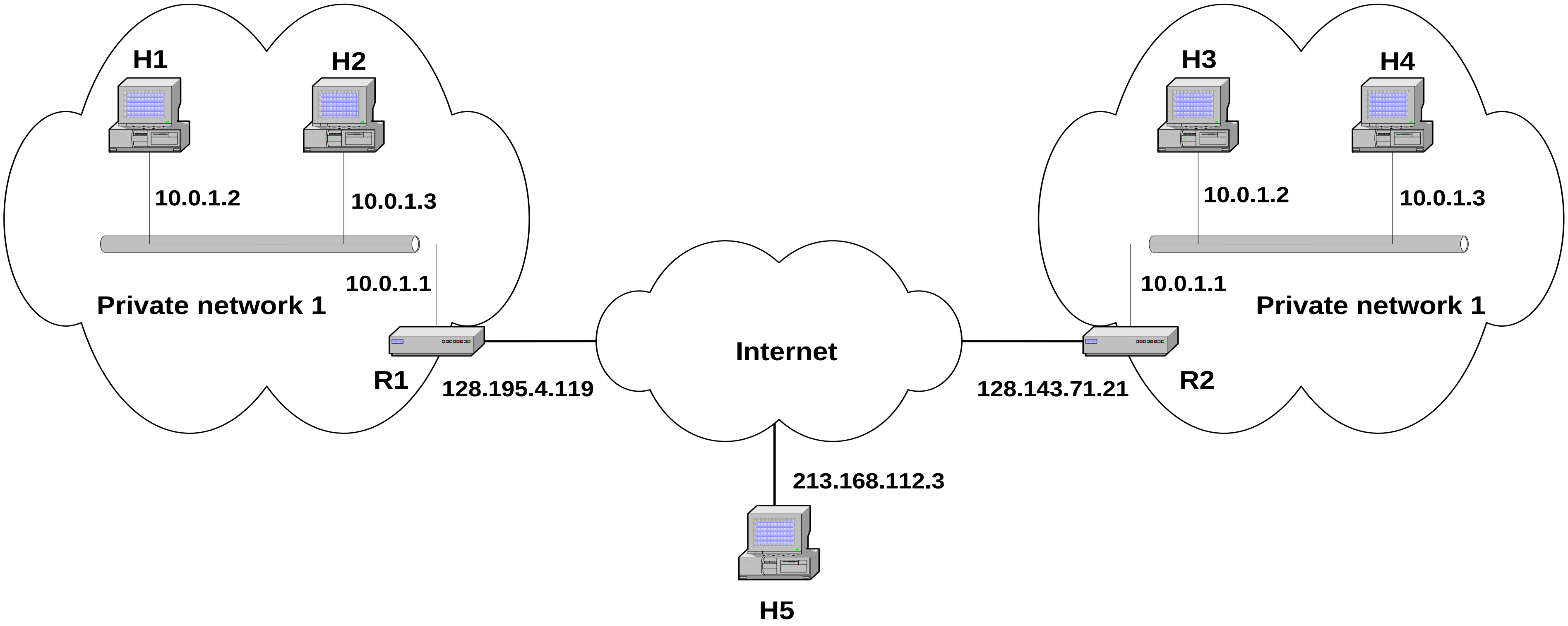
- Privacy - Hides an entire IP address space (private IP addresses) behind a single global IP address

The NAT Concept

- Internal **to** external hosts
  - The NAT replaces Internal Port to External Port and the internal IP address with the external IP address (of the NAT device)
  - Generates an entry in a translation table containing the internal IP address, original source port, and the translated source port
  - Subsequent packets from the same connection are translated to the same port number

- External **to** Internal hosts
  - Mapping to a corresponding internal IP address and port number from the translation table, replacing the external IP address and port number in the incoming packet header
  - The packet is then forwarded over the inside network
  - If the destination port number of the incoming packet is not found in the translation table, the packet is dropped

Private versus Global IP Addresses

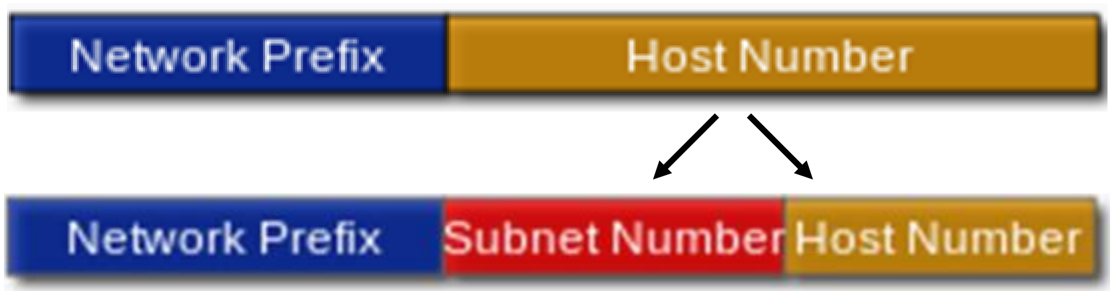


Private Networks IP Addresses

- Private IP network is an IP network that is not Directly Connected to the Internet
- Global IP Address – Unique ; Private IP Address - Not unique IP
- Private networks address ranges
  - Class A: 10.0.0.0 – 10.255.255.255 **10.x.x.x**
  - Class B: 172.16.0.0 – 172.31.255.255 **172.16.x.x-172.31.x.x**
  - Class C: 192.168.0.0 – 192.168.255.255 **192.168.x.x**

Subnet Mask

- An IP address has two components :the network address and the host address



- A subnet mask separates the IP address into the network and host addresses (<network><host>)
  - The network bits are represented by the 1's in the mask, and the host bits are represented by 0's

Subnet Mask

- A Subnet mask is a 32-bit number that masks an IP address
- Subnetting an IP network is to separate a big network into smaller multiple networks for reorganization and security purposes
- Performing a bitwise logical AND operation on the IP address with the subnet mask produces the network address

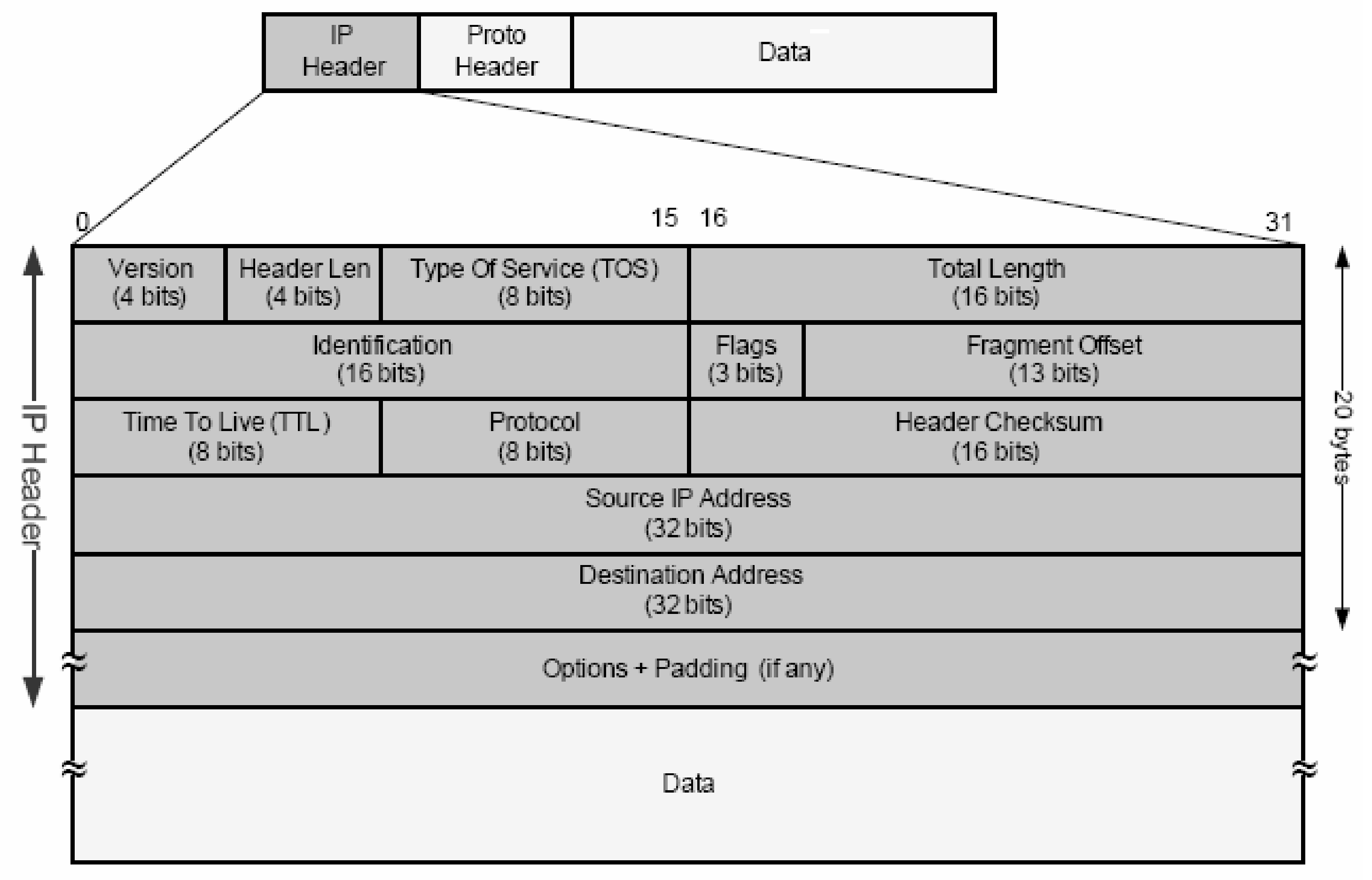
```
IP:   1101 1000 . 0000 0011 . 1000 0000 . 0000 1100   (216.003.128.012)
Mask: 1111 1111 . 1111 1111 . 1111 1111 . 0000 0000   (255.255.255.000)
      ---------------------------------------------
      1101 1000 . 0000 0011 . 1000 0000 . 0000 0000   (216.003.128.000)
```

Special IP Addresses

Prefix (network)	Suffix (host)	Type & Meaning
all zeros	all zeros	this computer (used during bootstrap)
network address	all zeros	identifies network
network address	all ones	broadcast on the specified network
all ones	all ones	broadcast on local network
127	any	loop back (for testing purposes)

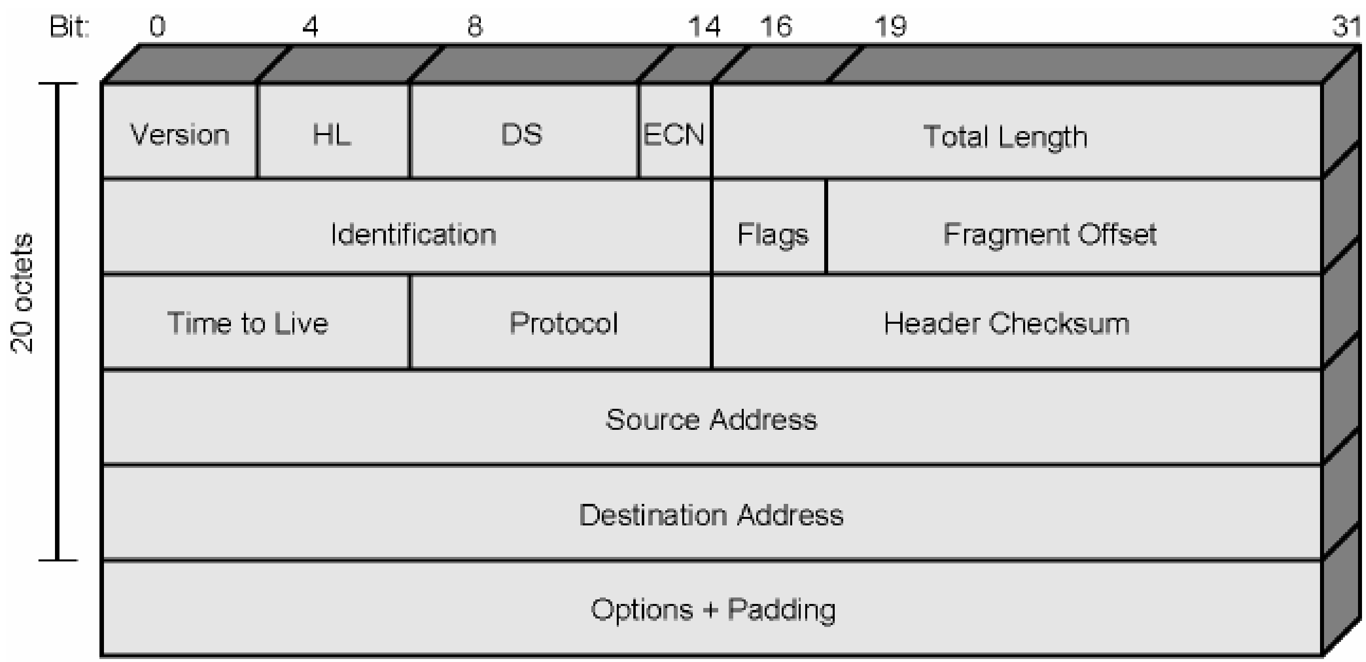
IP Header Structure (RFC 791)

RFC 791 : https://www.ietf.org/rfc/rfc791.txt



IP Header Structure (RFC 2474)

RFC 2474: https://tools.ietf.org/html/rfc2474



**** DS (Differentiated Services) and ECN (Explicit Congestion Notification)**

IP Header parameters

- Version
  - IP version 4
- Header length
  - Including options
- Type of Service (RFC 791)
  - Specify treatment of data unit during transmission through networks
- DS (Differentiated Services) and ECN (Explicit Congestion Notification) - RFC 2474
  - previously used for "Type of Service"
  - now used by (interpreted as) DS and ECN
  - DS is for QoS support

IP Header parameters (Cont.)

- Total length
  - Datagram (header + data), in octets

- Identification
  - Sequence number
  - Used with addresses and user protocol to identify datagram uniquely
  - Used for Fragmentation and Re-assembly

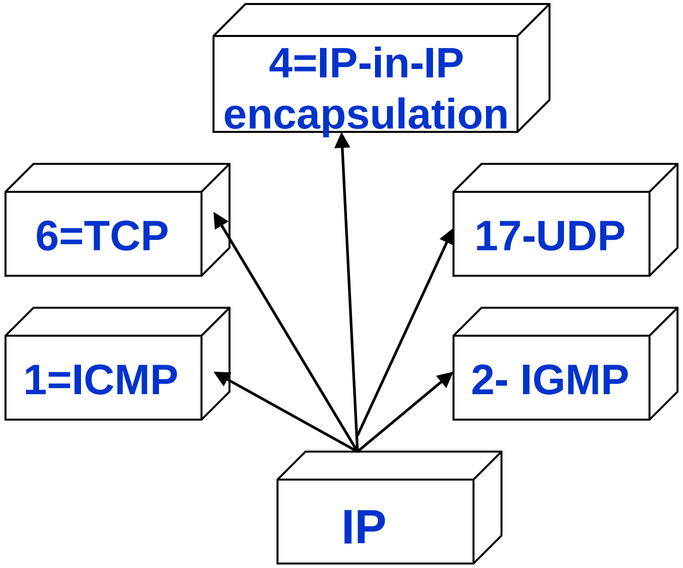
- Flags (3 bits)
  - First bit always set to 0
  - DF bit (Do not fragment)
  - MF bit (More fragments)

IP Header parameters (Cont.)

- Fragmentation offset

- Time To Live (TTL) (1 byte)
  - Ensure that packet is eventually dropped when a routing loop occurs
  - Used as follows
    - Sender sets the value (e.g., 64)
    - Each router decrements the value by 1
    - When the value reaches 0, the datagram is dropped

- Protocol (1 byte)
    - Specifies the higher-layer protocol



Datagram Lifetime

- Datagrams could loop indefinitely
  - Unnecessary resource consumption
  - Transport protocol needs upper bound on datagram life

- Datagram marked with lifetime
  - Time To Live (TTL) field in IP
  - Once lifetime expires, datagram discarded
  - Hop count
    - Decrement time to live on passing through each router

IP Header parameters (Cont.)

- Header checksum (2 bytes)
  - 16-bit long checksum which is computed for the header of the datagram
  - Verified and recomputed at each router
- Source address
- Destination address
- Options
    - Security restrictions
    - Record Route: each router that processes the packet adds its IP address to the header
    - Timestamp: each router that processes the packet adds its IP address and time to the header

IP Header parameters (Cont.)

- Options (cont.)
    - (loose) Source Routing: specifies a list of routers that must be traversed
    - (strict) Source Routing: specifies a list of the only routers that can be traversed
- Padding
  - Padding bytes are added to ensure that header ends on a 4-byte boundary (32 bits long)
- Data
  - User (upper layer) data
  - any octet length is OK
    - But max length of IP datagram (header plus data) is 65,535 octets

Maximum Transmission Unit

- Maximum size of IP datagram is 65535, but the data link layer protocol generally imposes a limit that is much smaller
  - Ethernet frames have a maximum payload of 1500 bytes → IP datagrams encapsulated in Ethernet frame cannot be longer than 1500 bytes

- The limit on the maximum IP datagram size, imposed by the data link protocol is called **Maximum Transmission Unit (MTU)**

- MTU for various data link protocols:
  - Ethernet: 1500
  - 802.3: 1492
  - 802.5: 4464
  - FDDI: 4352
  - ATM AAL5: 9180
  - PPP: negotiated

FDDI - Fiber Distributed Data Interface

Fragmentation and Re-assembly

- Different maximum packet sizes for different networks
  - routers may need to split the datagrams into smaller fragments

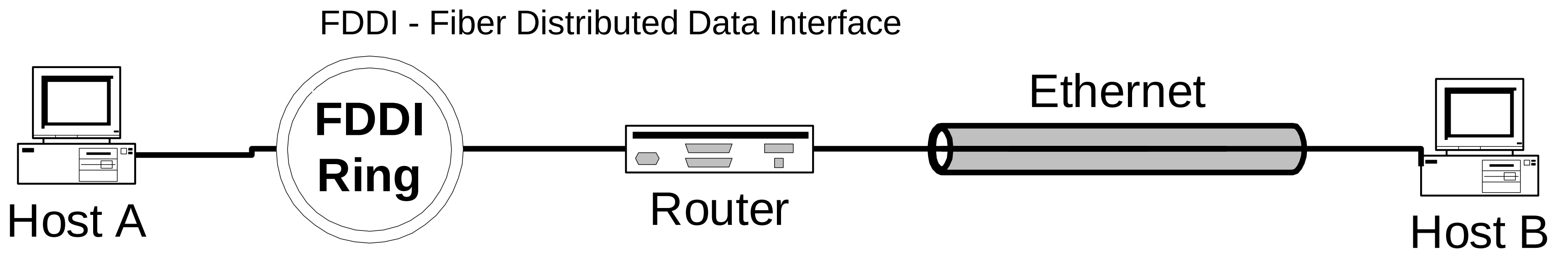
- When to re-assemble
  - At destination
    - Packets get smaller as data travel
      - inefficiency due to headers

  - Intermediate reassembly
    - Need large buffers at routers
    - All fragments must go through same router
      - Inhibits dynamic routing

IP Fragmentation

- What if the size of an IP datagram exceeds the MTU?
  - IP datagram is fragmented into smaller units

- What if the route contains networks with different MTUs?

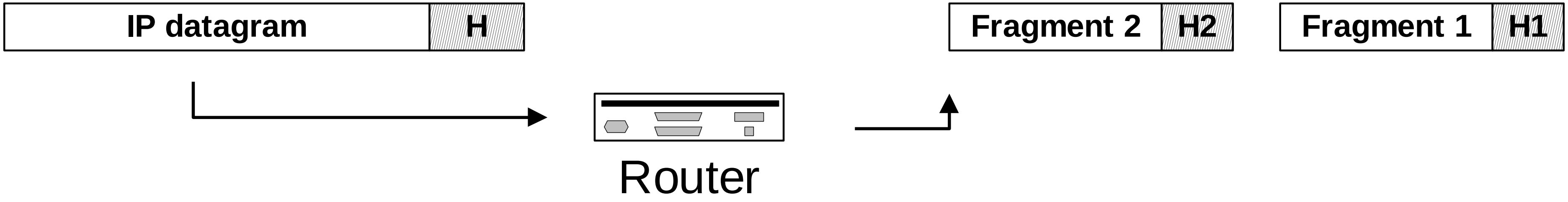


MTUs: FDDI: 4352 Ethernet: 1500

- **Fragmentation**:
  - IP router splits the datagram into several datagram
  - Fragments are reassembled at receiver

Where is Fragmentation done?

- Fragmentation can be done at the sender or at intermediate routers
- The same datagram can be fragmented several times
- Reassembly of original datagram is only done at destination hosts



What’s involved in Fragmentation?

<table>
<tr><td>version</td><td>header length</td><td>DS</td><td>ECN</td><td colspan="4">total length (in bytes)</td></tr>
<tr><td colspan="4">Identification</td><td>0</td><td>DF</td><td>MF</td><td>Fragment offset</td></tr>
<tr><td colspan="2">time-to-live (TTL)</td><td colspan="2">protocol</td><td colspan="4">header checksum</td></tr>
</table>

- Total length: Length of user data in octets (if fragment, length of fragment data) including the header
- Identification: uniquely identify datagram. all fragments that belong to a datagram share the same identifier
- Flags
  - DF (Do not fragment) bit is set: Datagram cannot be fragmented and must be discarded if MTU is too small
  - MF (More fragments) bit set: This datagram is part of a fragment and an additional fragment follows this one (not the last fragment)

What’s involved in Fragmentation?

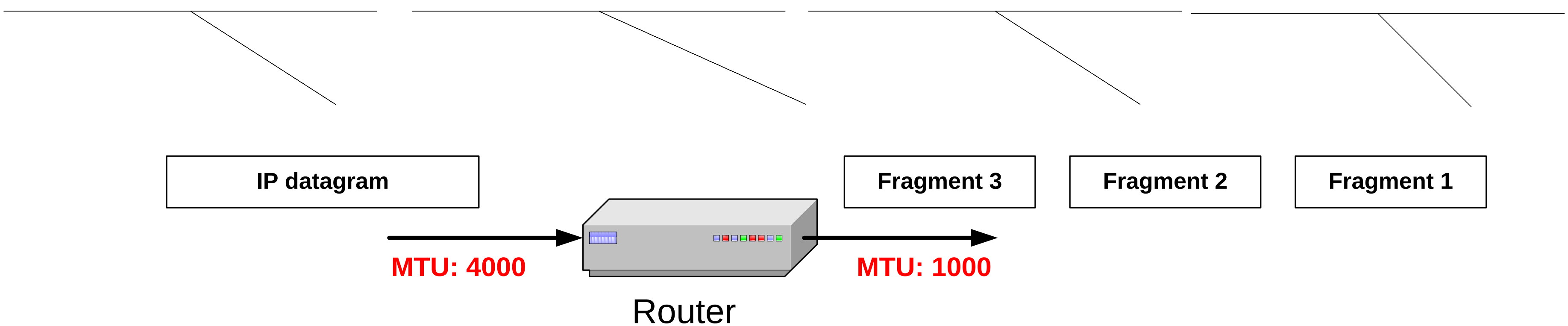
<table>
<tr><td>version</td><td>header length</td><td>DS</td><td>ECN</td><td colspan="4">total length (in bytes)</td></tr>
<tr><td colspan="4">Identification</td><td>0</td><td>D F</td><td>M F</td><td>Fragment offset</td></tr>
<tr><td colspan="2">time-to-live (TTL)</td><td colspan="2">protocol</td><td colspan="4">header checksum</td></tr>
</table>

- Fragment offset
    - Offset of the payload of the current fragment in the original datagram
    - Position of fragment of user data in original datagram In multiples of 64 bits (8 octets)

Fragmentation Example

- A datagram with size 2400 bytes must be fragmented according to an MTU limit of 1000 bytes

Header length: 20	Header length: 20	Header length: 20	Header length: 20
Total length: 2400	Total length: 448	Total length: 996	Total length: 996
Identification: 0xa428	Identification: 0xa428	Identification: 0xa428	Identification: 0xa428
DF flag: 0	DF flag: 0	DF flag: 0	DF flag: 0
MF flag: 0	MF flag: 0	MF flag: 1	MF flag: 1
Fragment offset: 0	Fragment offset: 244	Fragment offset: 122	fragment offset: 0



ARP – Address Resolution Protocol

- ARP was defined by RFC 826 in 1982
- Resolution of Internet layer addresses into link layer addresses
  - ARP is used for mapping a network address to a physical address like an Ethernet address (MAC address)
  - A request and response protocol whose messages are encapsulated by a link layer protocol

ARP – Address Resolution Protocol (2)

- Usage of a simple message format containing one address resolution request or response

- The size of the ARP message depends on the upper layer and lower layer address sizes

- The message header is completed with the operation code for request (1) and reply (2)

- The payload of the packet consists of four addresses, the hardware and protocol address of the sender and receiver hosts

ARP Packet Structure

0	Hardware type (HTYPE)	
2	Protocol type (PTYPE)	
4	Hardware address length (HLEN)	Protocol address length (PLEN)
6	Operation (OPER)	
8	Sender hardware address (SHA) (first 2 bytes)	
10	(next 2 bytes)	
12	(last 2 bytes)	
14	Sender protocol address (SPA) (first 2 bytes)	
16	(last 2 bytes)	
18	Target hardware address (THA) (first 2 bytes)	
20	(next 2 bytes)	
22	(last 2 bytes)	
24	Target protocol address (TPA) (first 2 bytes)	
26	(last 2 bytes)	

ARP Packet Structure

- Hardware type (HTYPE) - specifies the network protocol type. Example: Ethernet is 1
- Protocol type (PTYPE) - specifies the Internetwork protocol for which the ARP request is intended
  - For IPv4, this has the value 0x0800
- Hardware length (HLEN) - Length (in Bytes) of a hardware address
  - Ethernet addresses size is 6
- Protocol length (PLEN) - Length (in Bytes) of addresses used in the upper layer protocol.
  - IPv4 address size is 4
- Operation - Specifies the operation that the sender is performing: 1 for request, 2 for reply

ARP Packet Structure

- Sender hardware address (SHA) –
  - ARP request : to indicate the address of the host sending the request
  - In an ARP reply this field is used to indicate the address of the host that the request was looking for
- Sender protocol address (SPA) - Internetwork address of the sender
- Target hardware address (THA) - Media address of the intended receiver
  - In an ARP request this field is ignored
  - In an ARP reply this field is used to indicate the address of the host that originated the ARP request
- Target protocol address (TPA) - Internetwork address of the intended receiver

IP Addressing – Issues and solutions

- IP Addresses: Collisions and duplication
  - Must be unique in the LAN
  - Default GW and subnet must allocation
  - Interconnection between computers sharing the same LAN
  - Dynamic IP address allocation

DHCP –Dynamic Host Configuration Protocol

- Controlled by a DHCP server that dynamically distributes network configuration parameters, such as IP addresses, for interfaces and services

- A Router can be enabled to act as a DHCP server

- A DHCP server enables computers to request IP addresses and networking parameters automatically
  - reducing the need for a network administrator or a user to configure these settings manually
  - In the absence of a DHCP server, each computer or other device (e.g., a printer) on the network needs to be statically (i.e., manually) assigned to an IP address

DHCP

- Most residential network routers receive a globally unique IP address within the provider network
- Within a local network, a DHCP server assigns a local IP address to each device connected to the network
- The DHCP operates based on the client–server model
- When a computer or other device connects to a network, the DHCP client software sends a broadcast query requesting the necessary information
- Any DHCP server on the network may service the request

DHCP

- The DHCP server manages a pool of IP addresses and information about client configuration parameters
  - default gateway, domain name, the name servers, and time servers

- On receiving a request, the server may respond with specific information for each client
  - as previously configured by an administrator
  - or with a specific address and any other information valid for the entire network and for the time period for which the allocation (*lease*) is valid

DHCP

- A client typically queries for this information immediately after booting
  - And Periodically before the expiration of the information

- When a DHCP client refreshes an assignment, it initially requests the same parameter values
  - The DHCP server may assign a new address based on the assignment policies set by administrators

DHCP methods of allocating IP addresses

- Dynamic allocation
  - A network administrator reserves a range of IP addresses for DHCP
    - each DHCP client on the LAN is configured to request an IP address from the DHCP server during network initialization
    - The request-and-grant process uses a lease concept with a controllable time period, allowing the DHCP server to reclaim (and then reallocate) IP addresses that are not renewed

- Automatic allocation
  - The DHCP server permanently assigns an IP address to a requesting client from the range defined by the administrator
  - Like dynamic allocation, but the DHCP server keeps a table of past IP address assignments, so that it can preferentially assign to a client the same IP address that the client previously had

DHCP methods of allocating IP addresses

- Manual allocation (commonly called static allocation)
  - The DHCP server issues a private IP address dependent upon each client's MAC address, based on a predefined mapping by the administrator

DHCP Messages

- The DHCP employs a connectionless service model, using UDP
  - Two UDP port numbers for its operations which are the same as for the BOOTP protocol
  - UDP port number 67 is the destination port of a server, and UDP port number 68 is used by the client
- DHCP operations fall into four phases: server discovery, IP lease offer, IP lease request, and IP lease acknowledgement
  - These stages are often abbreviated as DORA for discovery, offer, request, and acknowledgement
- The DHCP operation begins with clients broadcasting a request

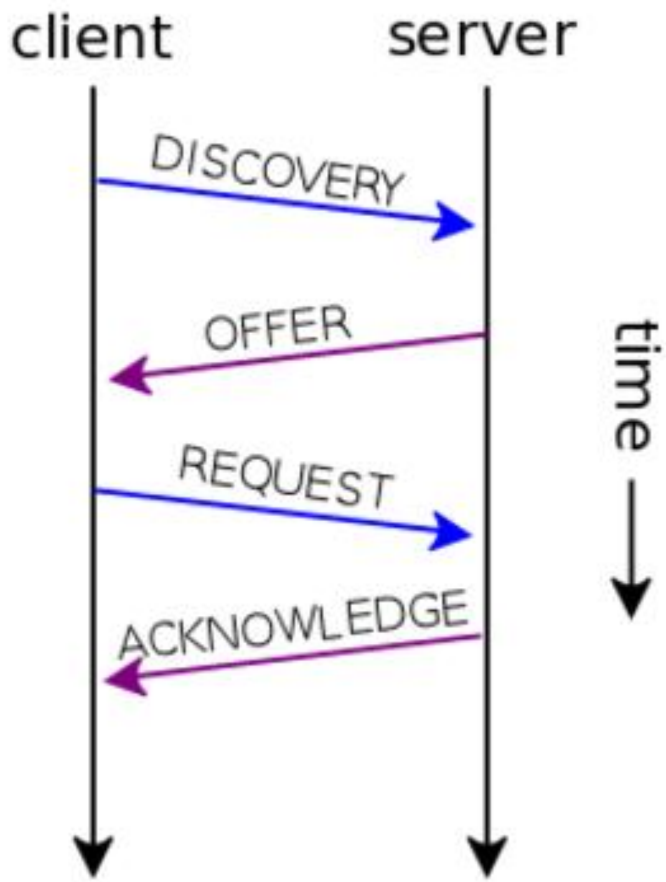
DHCP Messages

- Clients requesting renewal of an existing lease may communicate directly via UDP unicast, since the client already has an established IP address at that point

- BOOTP flag - the client can use to indicate in which way (broadcast or unicast) it can receive the DHCPOFFER: 0x8000 for broadcast, 0x0000 for unicast

- Only hosts with preconfigured IP addresses can receive unicast packets so in the usual use case clients in discovery phase should set BOOTP flag to 0x8000 (broadcast)

DHCP Messages Flow



DHCP Messages

- The client broadcasts messages on the network subnet using the destination address 255.255.255.255 or the specific subnet broadcast address

- A DHCP client may also request its last-known IP address
  - If the client remains connected to the same network, the server may grant the request
  - Otherwise, it depends whether the server is set up as authoritative or not
  - An authoritative server denies the request, causing the client to issue a new request
  - A non-authoritative server simply ignores the request, leading to an implementation-dependent timeout for the client to expire the request and ask for a new IP address

DHCP Security issues

- The base DHCP does not include any mechanism for authentication
  - It is vulnerable to a variety of attacks:
    - Unauthorized DHCP servers providing false information to clients
    - Unauthorized clients gaining access to resources
    - Resource exhaustion attacks from malicious DHCP clients

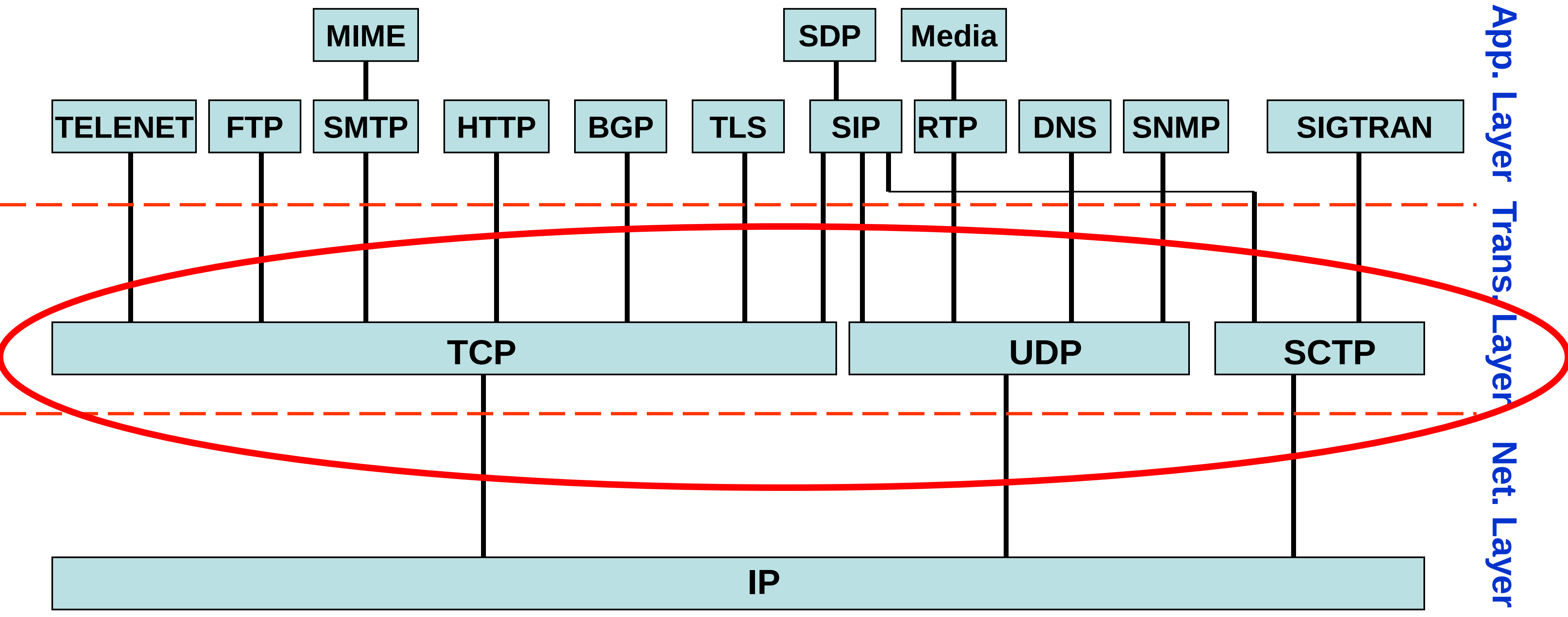
- No way for the Client to validate the identity of a DHCP server
  - Unauthorized DHCP servers can be operated on networks, providing incorrect information to DHCP clients
  - Two attack types
    - Denial-of-service attack, preventing the client from gaining access to network connectivity
    - A man-in-the-middle attack

DHCP Security issues

- DHCP server provides the DHCP client with server IP addresses, such as the IP address of one or more DNS servers
  - an attacker can convince a DHCP client to do its DNS lookups through its own DNS server, and can therefore provide its own answers to DNS queries from the client
  - Allows the attacker to redirect network traffic through itself, allowing it to eavesdrop
  -
- DHCP server has no secure mechanism for authenticating the client
  - clients can gain unauthorized access to IP addresses by presenting credentials, such as client identifiers, that belong to other DHCP clients
  - This also allows DHCP clients to exhaust the DHCP server's store of IP addresses—DoS Attack

4th Layer - Transport Layer

- The Transport Layer provides transparent transfer of data between end users, providing services to the upper layers



UDP

- Thin layer on top of IP
- Adds packet length + checksum
  - Guard against corrupted packets
- Also source and destination ports
  - Ports are used to associate a packet with a specific application at each end
- Still unreliable:
  - Duplication, loss, out-of-order possible

UDP datagram

0	16	31
Source Port	Destination Port	
Length	Checksum	
Application data		

Field	Purpose
Source Port	16-bit port number identifying originating application
Destination Port	16-bit port number identifying destination application
Length	Length of UDP datagram (UDP header + data)
Checksum	Checksum of UDP header and data

UDP Usage by typical Applications

- Reliability and QoS – responsibility of upper layer: The Application Layer

- Efficiency and small overhead are required
  - VoIP signalling
  - SNMP
  - DNS
  - RTP
  - Most games

TCP - Transmission Control Protocol (RFC: 793)

- Connection-oriented protocol
  - Connection is established and maintained until the application programs at each end have finished exchanging messages

- It determines how to break application data into packets that networks can deliver

- Sends packets to and accepts packets from the network layer

- Manages flow control
- Provides error-free data transmission
  - Handles retransmission of dropped or corrupted
  - Provides acknowledgement of all packets that arrive

TCP Reliability

- Reliable, full-duplex, connection-oriented, stream delivery
  - Data is guaranteed to arrive, and in the correct order without duplications
- Disadvantages
  - Imposes significant overheads
  - Not suitable for Real-Time Applications that are time-sensitive

TCP Implementation

- Connections are established using a three-way handshake
- Packets are numbered. Received packets are acknowledged
- Timers for retransmission process (by the operating system)
- Connections are explicitly closed
  - May abnormally terminate

TCP Timers

- RTO (Retransmission Time-Out)
  - When a segment is sent, a retransmission timer is started
  - If the segment is acknowledged before the timer expires, the timer is stopped
  - If the timer goes off before the acknowledgement comes in, the segment is retransmitted
  - The retransmission timer is also held to a minimum of 1 second, regardless of the estimates - to prevent spurious retransmissions based on measurements

TCP Timers

- Persistence timer
  - To prevent deadlock situations
  - The receiver sends an acknowledgement with a window size of 0, telling the sender to wait
  - Later, the receiver updates the window, but the packet with the update is lost
  - Now the sender and the receiver are each waiting for the other to do something
  - When the persistence timer goes off, the sender transmits a probe to the receiver
  - The response to the probe gives the window size
  - If it is still 0, the persistence timer is set again and the cycle repeats
  - If it is nonzero, data can now be sent

TCP Timers

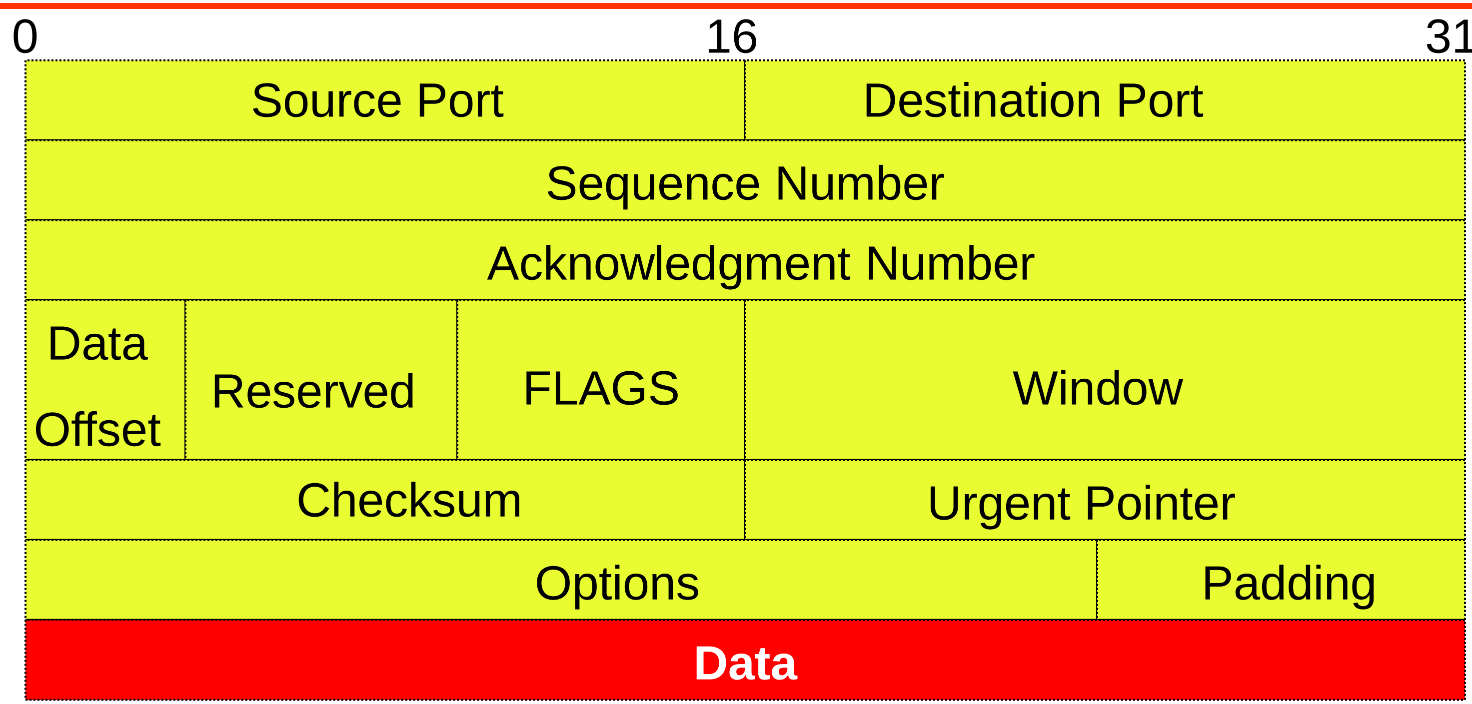
- Keep-alive timer
  - When a connection has been idle for a long time, the keep-alive timer may go off to cause one side to check whether the other side is still there
  - If it fails to respond, the connection is terminated

- TIME WAIT state while closing
  - It runs for twice the maximum packet lifetime to make sure that when a connection is closed; all packets created by it have died off

TCP Packets

- Source + destination ports
- Sequence number (used to order packets)
- Acknowledgement number (used to verify packets are received)

TCP Structure



Field	Purpose
Source Port	Identifies originating application
Destination Port	Identifies destination application
Sequence Number	Sequence number of first octet in the segment
Acknowledgment #	Sequence number of the next expected octet (if ACK flag set)
Len/Data Offset	Length of TCP header in 4 octet units
Flags	TCP flags: SYN, FIN, RST, PSH, ACK, URG
Window	Number of octets from ACK that sender will accept
Checksum	Checksum of TCP header + data
Urgent Pointer	Pointer to end of “urgent data”
Options	Special TCP options

TCP Structure

- Source Port (16 bits)
- Destination Port (16 bits)
- Sequence Number (32 bits) - The sequence number of the first data octet in this segment
- Acknowledgment Number (32 bits) - If the ACK Flag bit is set, then this field contains the value of the next sequence number expecting to receive
  - Once a connection is established this is always sent
- Data Offset (4 bits)- Indicates where the data begins
- Reserved (6 bits) Reserved for future use
  - Must be zero

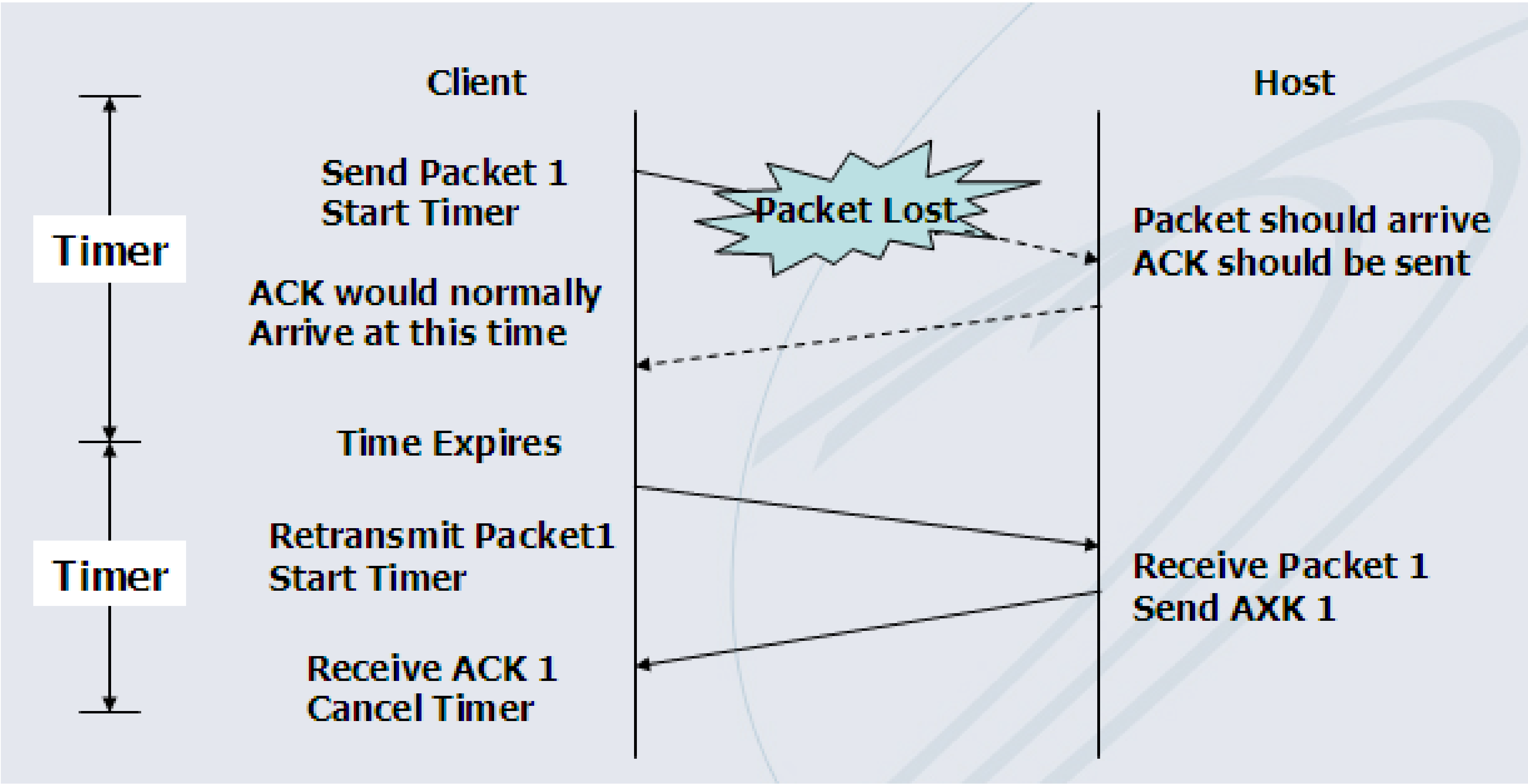
TCP Structure

- Flags/Control Bits (6 bits)
  - URG: Urgent Pointer field significant
  - ACK: Acknowledgment field significant
  - PSH: Push Function
  - RST: Reset the connection
  - SYN: Synchronize sequence numbers
  - FIN: No more data from sender
- Window (16 bits) - The number of data octets which the sender of this segment is willing to accept
- Checksum (16 bits)

TCP Structure

- Urgent Pointer (16 bits) - Points to the sequence number of the octet following the urgent data
  - This field is only be interpreted in segments if the URG Flag bit set
- Options (variable)
- Padding (variable) - To ensure that the TCP header ends and data begins on a 32 bit boundary
  - The padding is composed of zeros

TCP : Data transfer



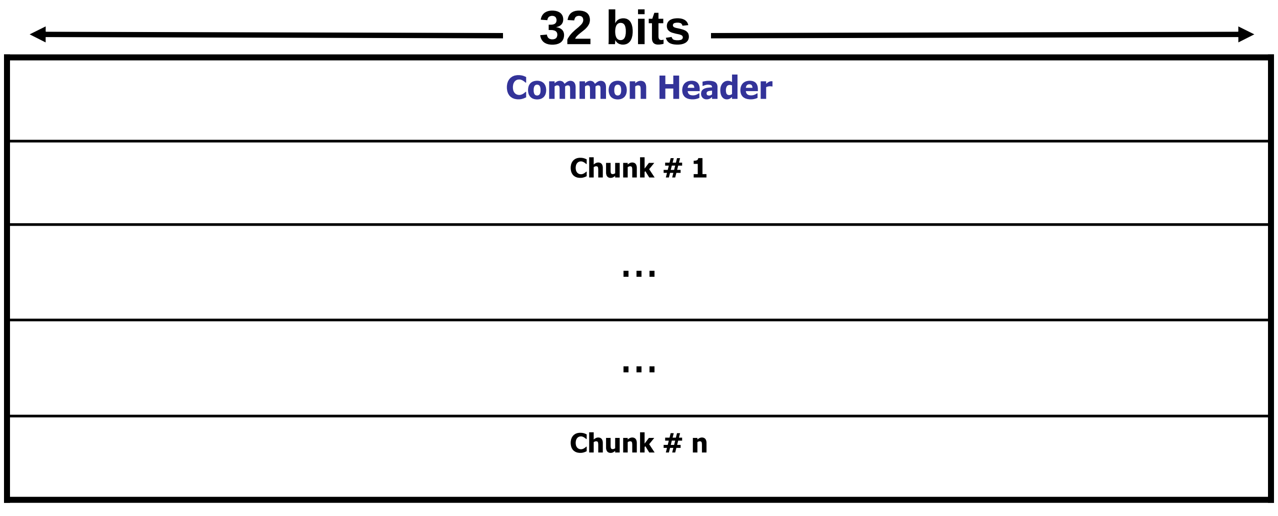
What is SCTP

(Stream Control Transmission Protocol- RFC 2960)

- Supports PSTN signaling messages over IP Networks
- Reliable transport protocol over connectionless network
- Provides partially ordered delivery
- Reduces user-perceived latency and improve throughput

SCTP Structure

- 4th Layer. Used for Real-time sensitive applications require reliability
  - SS7, SIP, H.323, etc.
- An SCTP packet is composed of a common header and chunks
- A chunk contains either control information or user data



- Multiple chunks can be bundled into one SCTP packet
  up to the MTU Size*
- If a user data message doesn't fit into one SCTP packet
  -it can be fragmented into multiple chunks

* except for the INIT, INIT ACK, and SHUTDOWN COMPLETE chunks

SCTP Common Header Format

2 bytes	2 bytes
Source Port Number	Destination Port Number
Verification Tag	
Adler 32 Checksum	

- Source/Destination Port Number Field: 16 Bits
  - Indicates the SCTP sender's/destination's port number

- Verification Tag Field: 32 Bits
  - The receiver of this packet uses the verification tag to validate the sender of this SCTP packet

- Checksum Field: 32 Bits (Header +Chunks)

Payload Data (chunk ID=0)

<table>
<tr><th>1 Byte</th><th colspan="4">1 Byte</th><th>2 Bytes</th></tr>
<tr><td>Type = 0</td><td>Reserved</td><td>U</td><td>B</td><td>E</td><td>Length</td></tr>
<tr><td colspan="6">TSN</td></tr>
<tr><td colspan="5">Stream Identifier S</td><td>Stream Sequence Number n</td></tr>
<tr><td colspan="6">Payload Protocol Identifier (32 bits)</td></tr>
<tr><td colspan="6">User Data (Seq n of Stream S)</td></tr>
</table>

SCTP Benchmark

Protocol	Loss	File 1	File 2	File 3	File 4	File 5	File 6	File 7
TCP	0%	0.679	0.768	3.873	3.910	3.942	4.243	4.273
SCTP	0%	0.802	0.888	4.468	4.507	4.607	4.834	4.878
TCP	1%	4.930	5.595	29.598	31.047	31.924	33.460	34.333
SCTP	1%	4.299	4.775	24.132	24.536	25.106	26.678	27.143
TCP	2%	5.983	6.725	35.361	37.232	38.509	40.681	42.568
SCTP	2%	5.506	6.098	31.539	32.164	32.692	33.117	33.981

Latency of each file in multiple file transfer test, B/W=10Mbps. Values in **red** are higher. All times are in seconds

SCTP versus TCP

- Major Differences
  - Message oriented (instead of packet oriented)
  - Fragmentation process
  - Multiple streams support
  - Multiple homing support

- Major Similarities
  - Similar Flow Control
  - Congestion Avoidance
  - Fast Retransmit