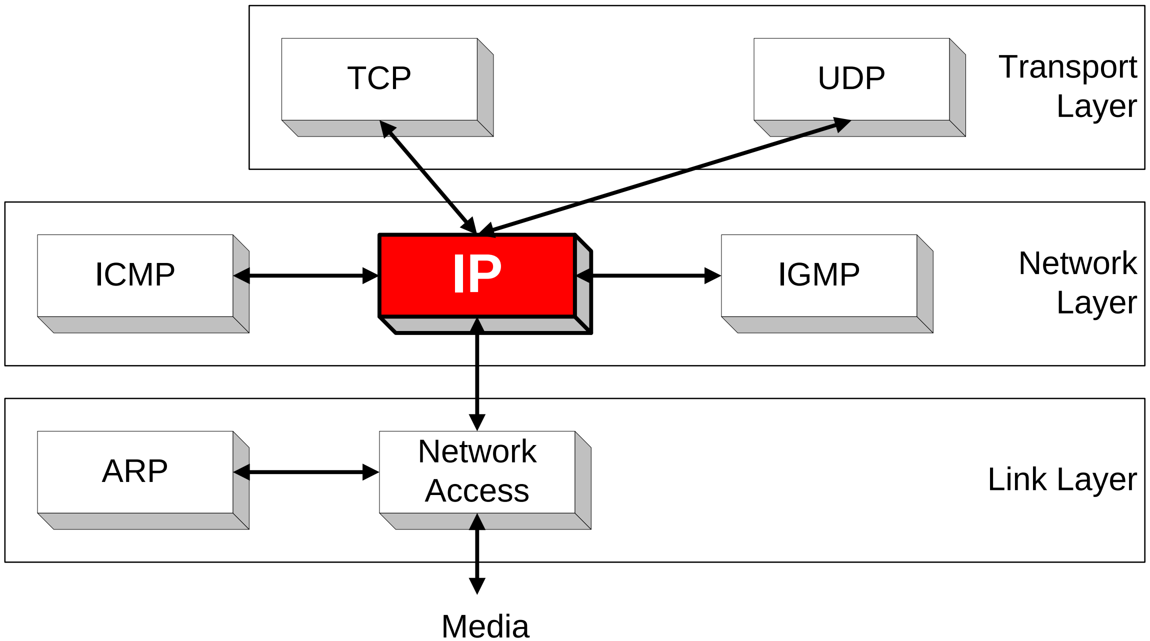
IP - The Internet Protocol

Based on the slides of Dr. Jorg Liebeherr, University of Virginia

Orientation

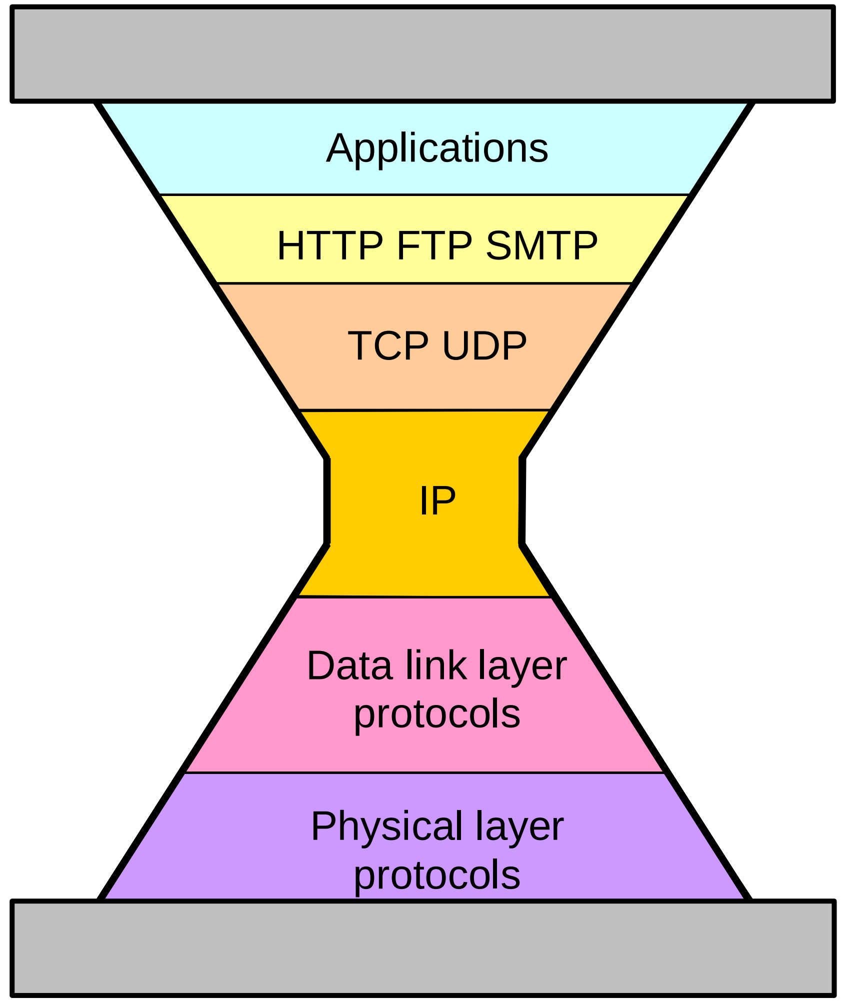
- IP (Internet Protocol) is a Network Layer Protocol.



- IP's current version is Version 4 (IPv4). It is specified in RFC 891.

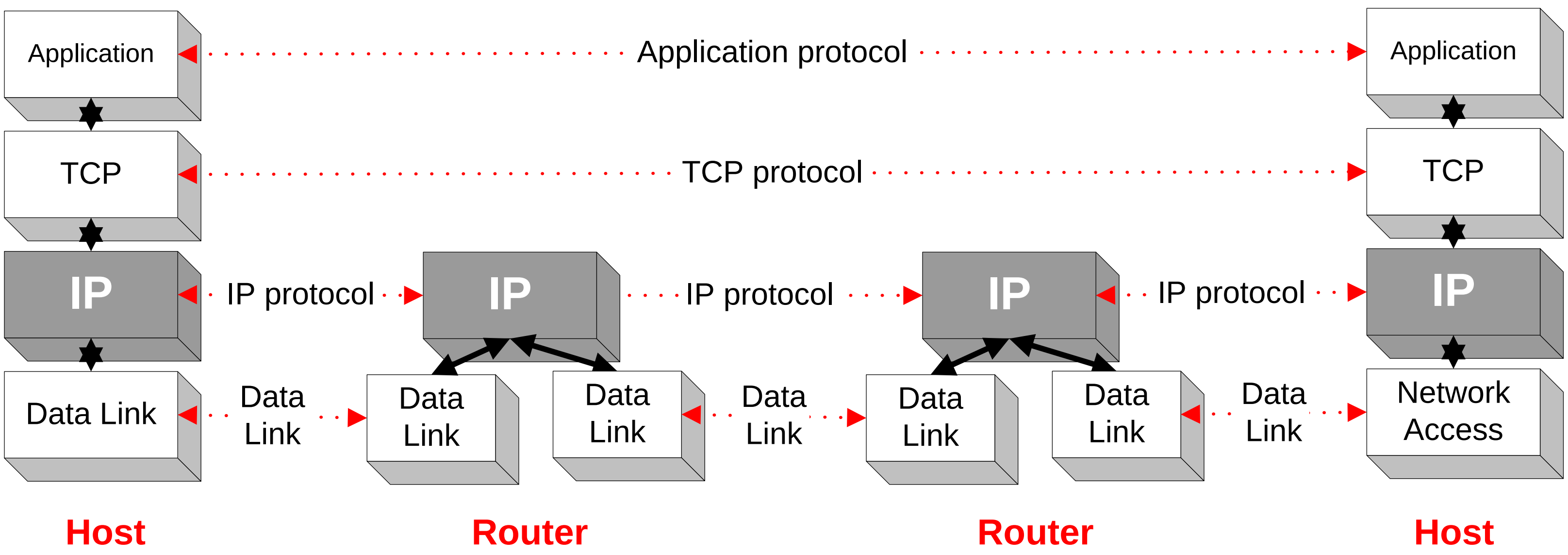
IP: The waist of the hourglass

- **IP is the waist of the hourglass of the Internet protocol architecture**
- Multiple higher-layer protocols
- Multiple lower-layer protocols
- Only one protocol at the network layer.



Network Layer Protocol

- IP is the highest layer protocol which is implemented at both routers and hosts



IP Service

- Delivery service of IP is minimal

- IP provide 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,…)

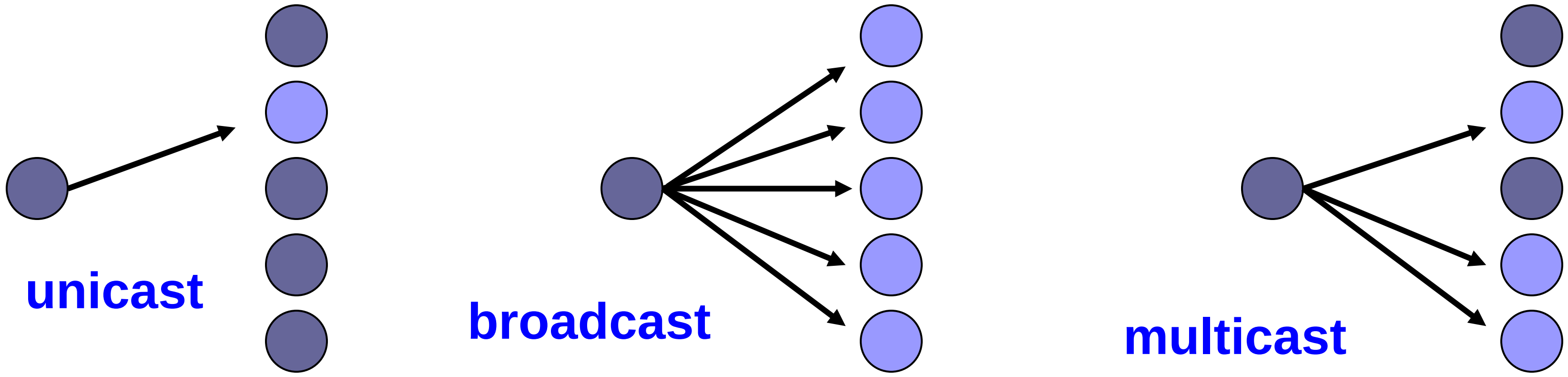
- Consequences:

- Higher layer protocols have to deal with losses or with duplicate packets

- Packets may be delivered out-of-sequence

IP Service

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



- IP multicast also supports a many-to-many service.
- IP multicast requires support of other protocols (IGMP, multicast routing)

IP Addresses

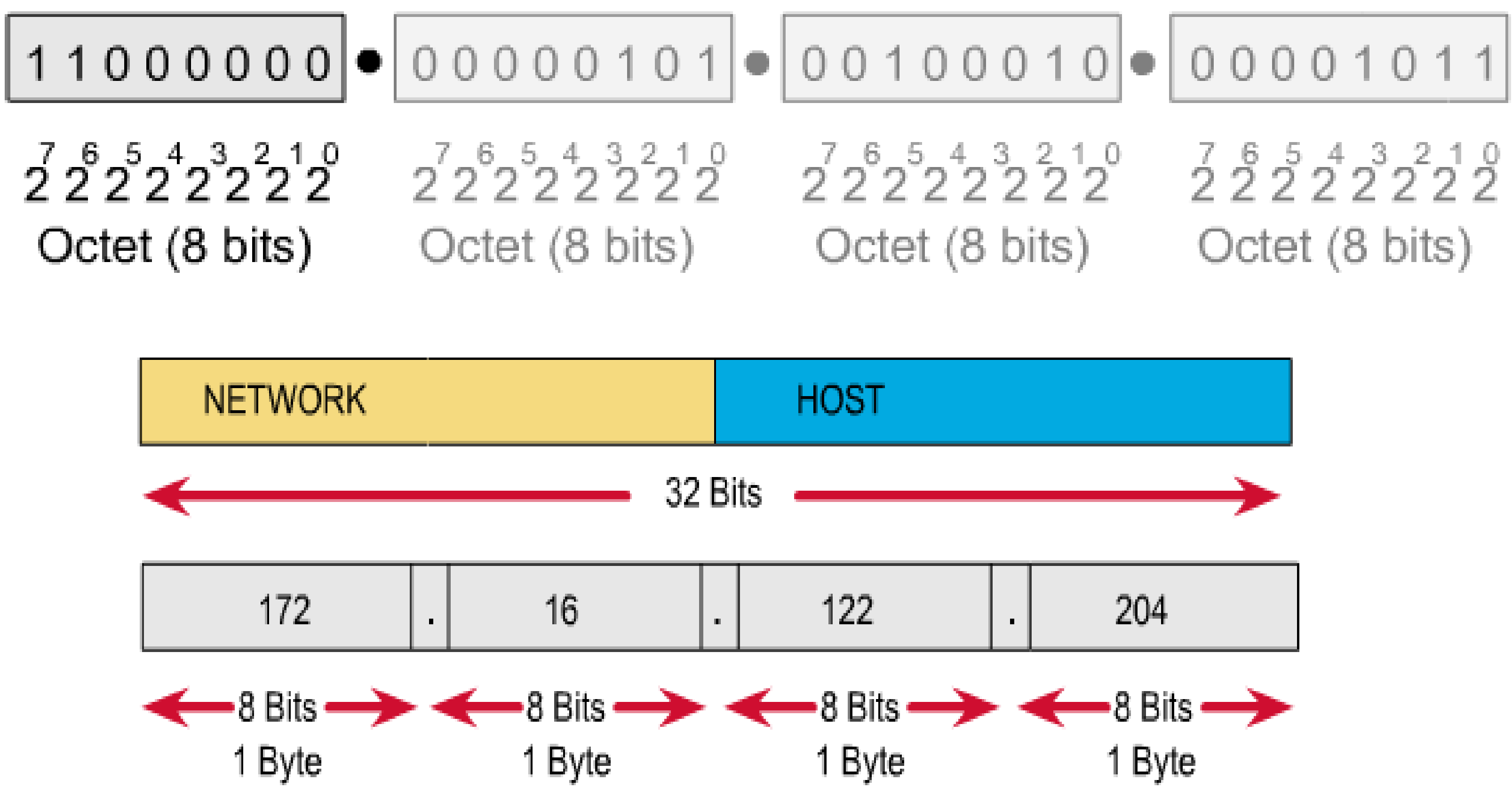
- IP is a network layer - it must be capable of providing communication between hosts on different kinds of networks (different data-link implementations).
- The address must include information about what *network* the receiving host is on. This is what makes routing feasible.

IP Addresses

- IP addresses are *logical* addresses (not physical)
- 32 bits. IPv4 *(version 4)*
- Includes a network ID and a host ID.
- Every host must have a unique IP address.
- IP addresses are assigned by ICANN

(*Internet Corporation for Assigned Names and Numbers*).

IP Address as a 32-Bit Binary Number (four numbers separated by periods)

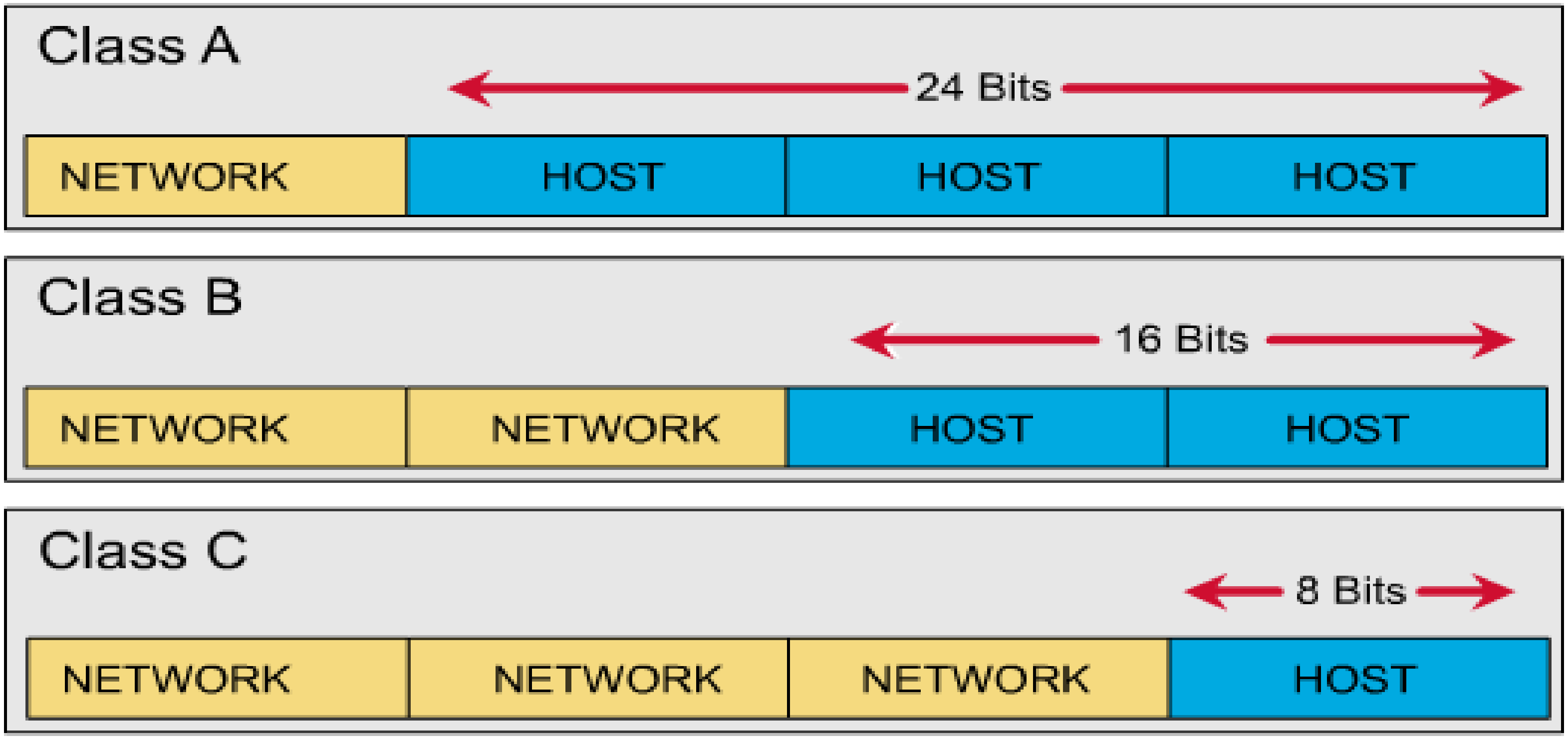


Binary and Decimal Conversion

$2^{(7)}$	$2^{(6)}$	$2^{(5)}$	$2^{(4)}$	$2^{(3)}$	$2^{(2)}$	$2^{(1)}$	$2^{(0)}$
128	64	32	16	8	4	2	1

192.57.30.224
11000000.00111001.00011110.11100000

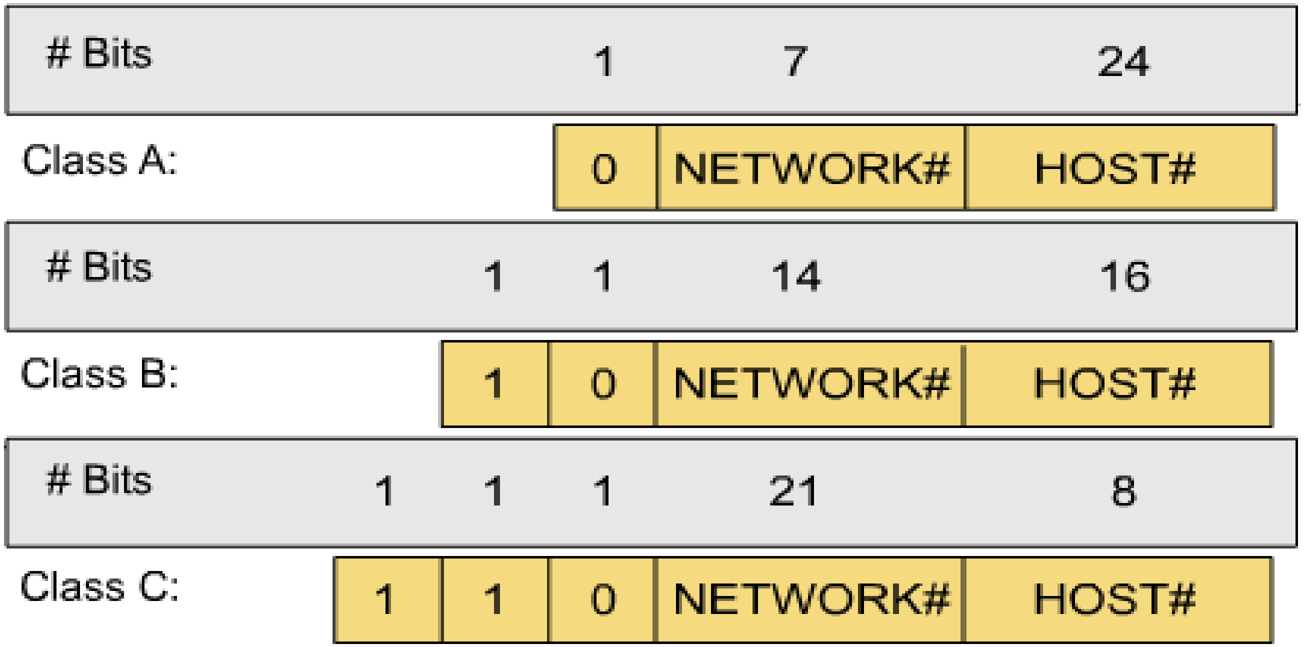
Classes of Network IP Addresses



Class D IP addresses are reserved for so-called multicast addresses (leading bits of 1 1 1 0)

Class E IP addresses are reserved for experimental use (leading bits of 1 1 1 1)

IP Addresses as Decimal Numbers



Class A : we can have up to 2^7 networks (the first of the eight bits is fixed as 0) and and 2^{24} Interfaces (hosts)

Class B : we can have up to 2^{14} networks, and up to 2^{16} interfaces

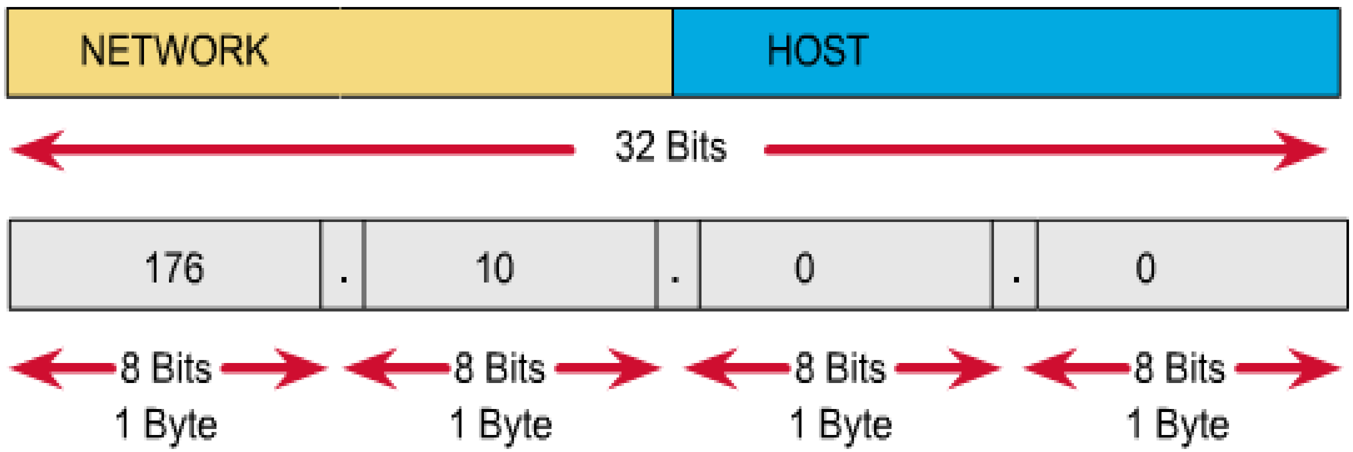
Class C : we can have up to 2^{21} networks, and up to 2^8 interfaces

IP Addresses

	First byte	Second byte	Third byte	Fourth byte
Class A	0 to 127			
Class B	128 to 191			
Class C	192 to 223			
Class D	224 to 239			
Class E	240 to 255			

Network IDs and Broadcast Addresses

An IP address such as 176.10.0.0 that has all binary 0s in the host bit positions is reserved for the <u>network address</u>.



An IP address such as 176.10.255.255 that has all binary 1s in the host bit positions is reserved for the <u>broadcast address.</u>

Private Addresses

The following ranges are available for private addressing

10.0.0.0 - 10.255.255.255

172.16.0.0 - 172.31.255.255

192.168.0.0 - 192.168.255.255

Subnet Mask

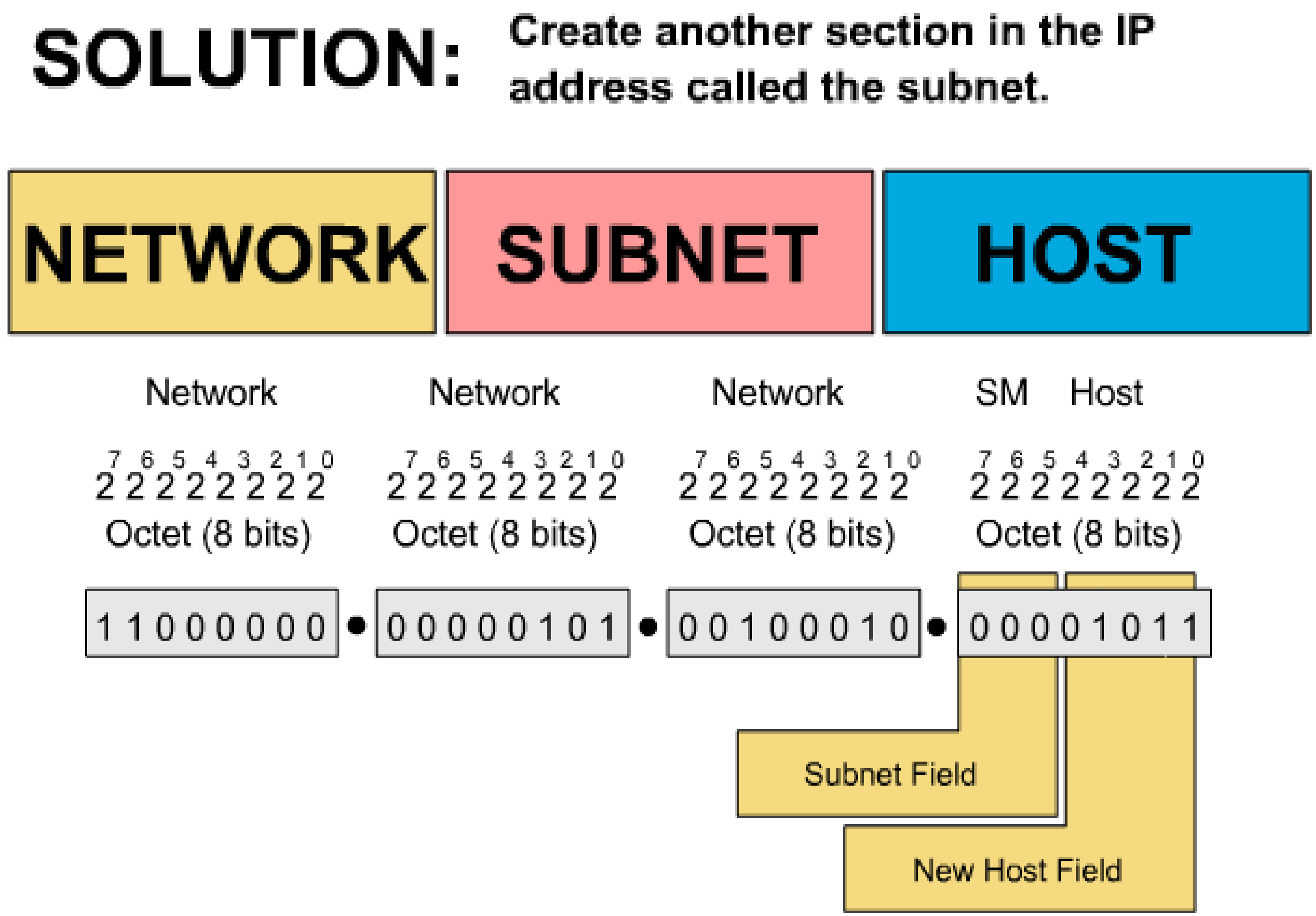
- A subnet mask is a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s.
- Subnet masks are used to distinguish between hosts, networks, and subnetworks
- The result of the bitwise AND operation of IP address and the subnet mask is the network prefix (ex: 192.0.2.130) ==== (192.0.2.0)
- The host part, which is 130, is derived by the bitwise AND operation of IP address and the one's complement of the subnet mask, (0.0.0.130)
- 192.0.2.130 class C, Subnet mask (255.255.255.0) ======== 192.0.2.130/24
- All IP’s (192.0.2.0) to (192.0.2.255)

	Binary form	Dot-decimal notation
IP address	11000000.00000000.00000010.10000010	192.0.2.130
Subnet mask	11111111.11111111.11111111.00000000	255.255.255.0
Network prefix	11000000.00000000.00000010.00000000	192.0.2.0
Host identifier	00000000.00000000.00000000.10000010	0.0.0.130

Subnetworks

To create a subnet address, a network administrator borrows bits from the original host portion and designates them as the subnet field.

Subnetworks



Subnetworks

Subnetting is the process of designating some high-order bits from the host part as part of the network prefix and adjusting the subnet mask appropriately. This divides a network into smaller subnets.

The following diagram modifies the previous example by moving 2 bits from the host part to the network prefix to form four (2^2) smaller subnets each one quarter of the previous size 256/4=64 (2^6). (192.0.2.130/26)

(192.0.2.0) to (192.0.2.63) , (192.0.2.64) to (192.0.2.127) , (192.0.2.128) to (192.0.2.191) , (192.0.2.192) to (192.0.2.255)

	Binary form	Dot-decimal notation
IP address	11000000.00000000.00000010.10000010	192.0.2.130
Subnet mask	11111111.11111111.11111111.**11**000000	255.255.255.192
Network prefix (subnet)	11000000.00000000.00000010.10000000	192.0.2.128
Host part	00000000.00000000.00000000.00000010	0.0.0.2

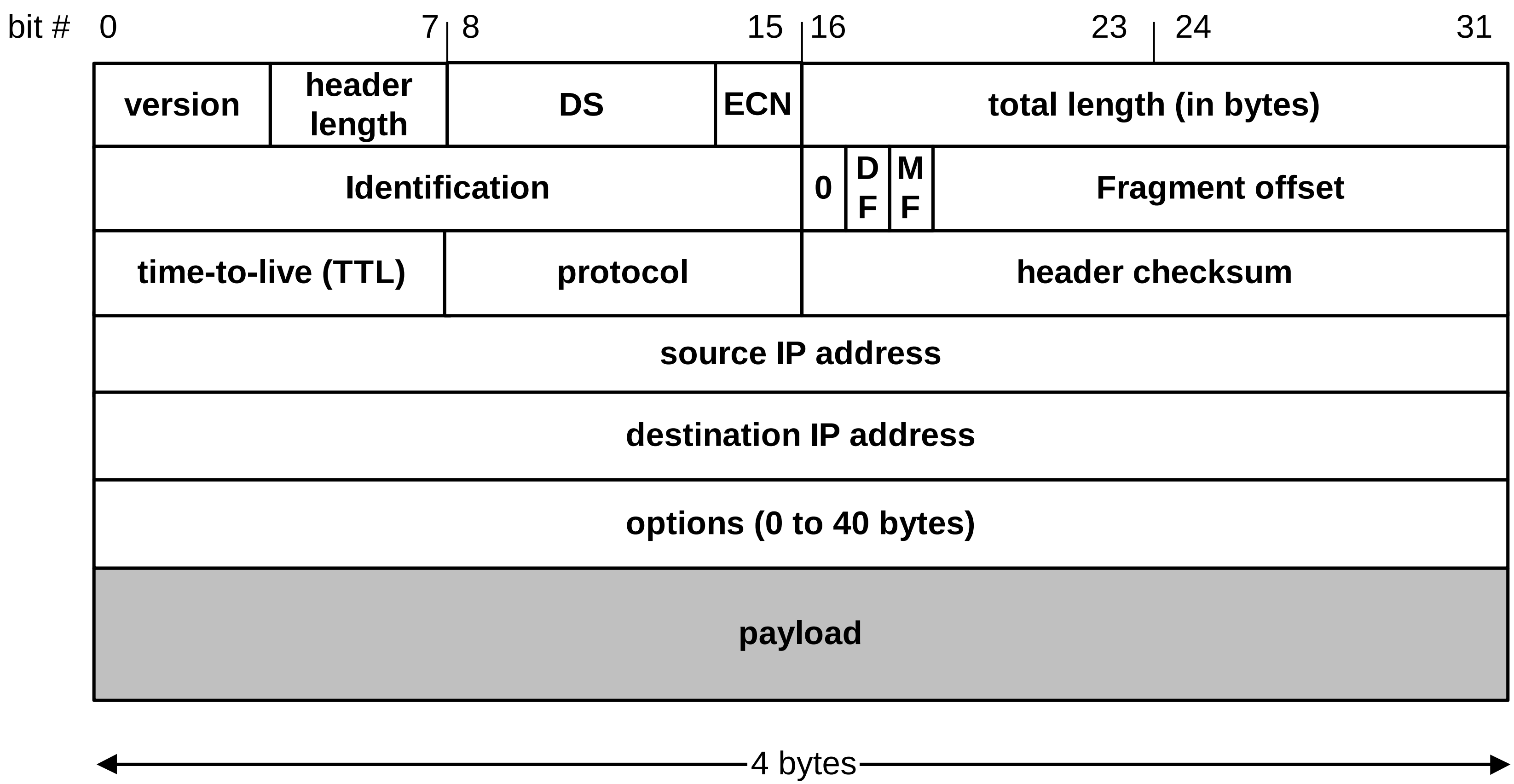
Subnetworks

Network	Network (binary)	Broadcast address
192.168.5.0/26	11000000.10101000.00000101.00000000	192.168.5.63
192.168.5.64/26	11000000.10101000.00000101.01000000	192.168.5.127
192.168.5.128/26	11000000.10101000.00000101.10000000	192.168.5.191
192.168.5.192/26	11000000.10101000.00000101.11000000	192.168.5.255

- The first subnet obtained from subnetting has all bits in the subnet bit group set to zero (0). It is therefore called subnet zero. The last subnet obtained from subnetting has all bits in the subnet bit group set to one (1). It is therefore called the all-ones subnet.

- 2 of which can actually be assigned to a device since host ids of all zeros or all ones are not allowed (i.e avoiding subnet zero and the all-ones subnet)

IP Datagram Format



- 20 bytes ≤ Header Size < 2^4 x 4 bytes = 60 bytes
- 20 bytes ≤ Total Length < 2^{16} bytes = 65536 bytes

IP Datagram Format

- **Question:** In which order are the bytes of an IP datagram transmitted?
- **Answer:**
  - Transmission is row by row
  - For each row:
    1. First transmit bits 0-7
    2. Then transmit bits 8-15
    3. Then transmit bits 16-23
    4. Then transmit bits 24-31
- This is called **network byte** order or **big endian** byte ordering.

- **Note:** Many computers (incl. Intel processors) store 32-bit words in little endian format. Others (incl. Motorola processors) use big endian.

Big endian vs. small endian

- Conventions to store a multibyte work
- Example: a 4 byte Long Integer **`Byte3 Byte2 Byte1 Byte0`**

Little Endian

- Stores the low-order byte at the lowest address and the highest order byte in the highest address.

```
Base Address+0 Byte0
Base Address+1 Byte1
Base Address+2 Byte2
Base Address+3 Byte3
```

- Intel processors use this order

Big Endian

- Stores the high-order byte at the lowest address, and the low-order byte at the highest address.

```
Base Address+0 Byte3
Base Address+1 Byte2
Base Address+2 Byte1
Base Address+3 Byte0
```

Motorola processors use big endian.

Fields of the IP Header

- **Version (4 bits)**: current version is 4, next version will be 6.
- **Header length (4 bits)**: length of IP header, in multiples of 4 bytes
- **DS/ECN field (1 byte)**
  - This field was previously called as Type-of-Service (TOS) field. The role of this field has been re-defined, but is "backwards compatible" to TOS interpretation
  - Differentiated Service (DS) (6 bits):
    - Used to specify service level (currently not supported in the Internet)
  - Explicit Congestion Notification (ECN) (2 bits):
    - New feedback mechanism used by TCP

Fields of the IP Header

- **Identification (16 bits):** Unique identification of a datagram from a host. Incremented whenever a datagram is transmitted

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

  Will be explained later→ Fragmentation

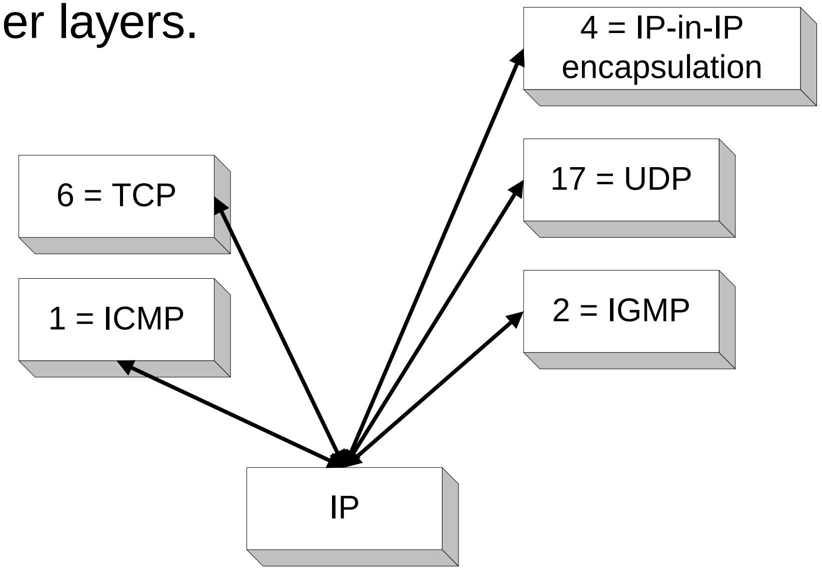
Fields of the IP Header

- **Time To Live (TTL) (1 byte):**
  - Specifies longest paths before datagram is dropped
  - Role of TTL field: 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

Fields of the IP Header

- **Protocol (1 byte):**
  - Specifies the higher-layer protocol.
  - Used for demultiplexing to higher layers.



- **Header checksum (2 bytes):** A simple 16-bit long checksum which is computed for the header of the datagram.

Fields of the IP Header

- **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.
  - (loose) Source Routing: specifies a list of routers that must be traversed. (but the datagram can also pass through other routers between any two addresses in the list.)
  - (strict) Source Routing: specifies a list of the only routers that can be traversed. (If a router encounters a next hop in the source route that isn't on a directly connected network, an ICMP "source route failed" error is returned.)

- **Padding:** Padding bytes are added to ensure that header ends on a 4-byte boundary

Maximum Transmission Unit

- Maximum size of IP datagram is 65535, but the data link layer protocol generally imposes a limit that is much smaller

- Example:
  - 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)**

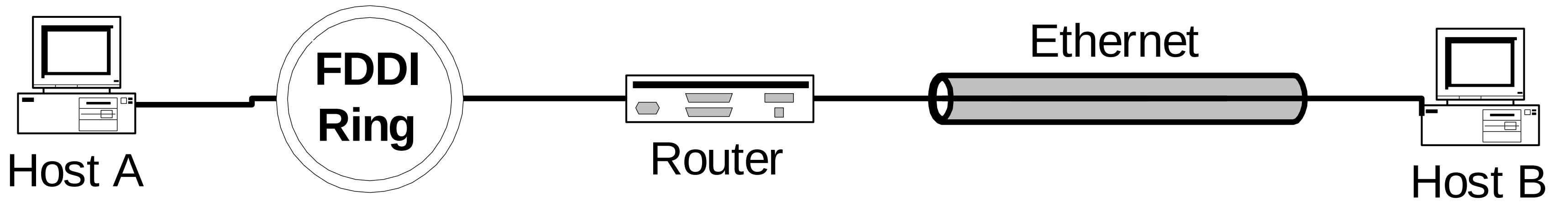
- MTUs for various data link protocols:

Ethernet:	1500	FDDI (Fiber Distributed Data Interface)	:4352
802.3 (CSMA/CD) :	1492	ATM (Asynchronous Transfer Mode) AAL5	: 9180
802.5 (token ring):	4464	PPP	: negotiated

Note: IEEE standards 802 are used for controlling the Local Area Network and Metropolitan Area Network

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?

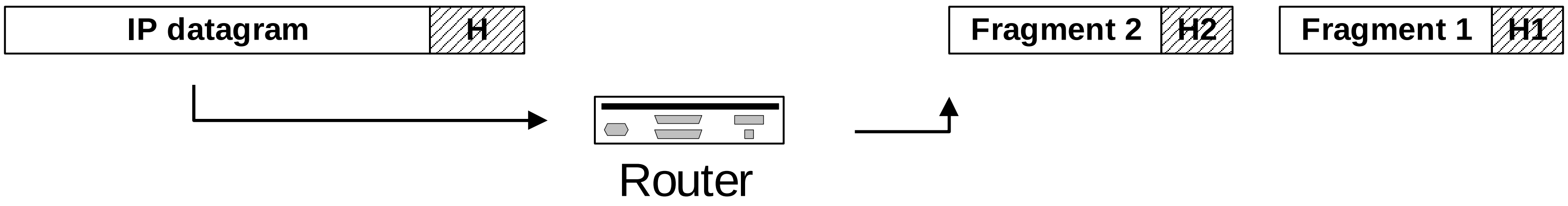


MTUs: FDDI: 4352 Ethernet: 1500

- **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?

- The following fields in the IP header are involved:

<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>

Identification — When a datagram is fragmented, the identification is the same in all fragments

Flags

DF bit is set: Datagram cannot be fragmented and must be discarded if MTU is too small

MF bit set: This datagram is part of a fragment and an additional fragment follows this one

What's involved in Fragmentation?

- The following fields in the IP header are involved:

version	header length	DS	ECN	total length (in bytes)			
Identification				0	DF	MF	Fragment offset
time-to-live (TTL)		protocol		header checksum			

Fragment offset	Offset of the payload of the current fragment in the original datagram
Total length	Total length of the current fragment

Example of Fragmentation

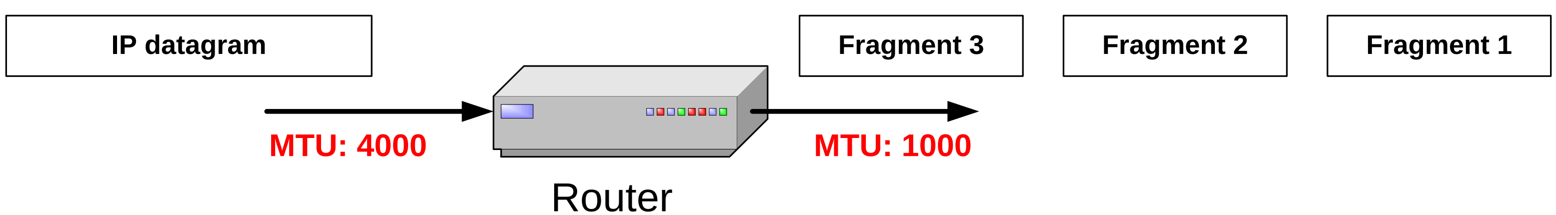
- Example 1: A datagram with size 2400 bytes must be fragmented according to an MTU limit of 1000 bytes

Header length:	20
Total length:	2400
Identification:	0xa428
DF flag:	0
MF flag:	0
Fragment offset:	0

Header length:	20
Total length:	448
Identification:	0xa428
DF flag:	0
MF flag:	0
Fragment offset:	244

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

Header length:	20
Total length:	996
Identification:	0xa428
DF flag:	0
MF flag:	1
fragment offset:	0



Example of Fragmentation Example 1:

IP datagram = 2400 bytes , IP header = 20 bytes , MTU = 1000 bytes

1. Actual data to be transmitted= datagram- IP header = segment=2400-20=2380
2. Maximum possible data length per fragment = MTU – IP header = 1000 – 20 = 980 bytes.
3. The number of required fragments are 2380/980=2.4=3
4. The data length of each fragment must be a multiple of eight bytes; therefore the maximum number of data bytes that can be carried per fragment is 122*8=976. (980/8=122.5)----------- (976 /8=122)
5. The payload for the first fragment is 976 and has bytes 0 ~ 975 of the original IP datagram. The offset is 0/8=0.
6. The payload for the second fragment is 976 and has bytes 976 ~ 1951 of the original IP datagram. The offset is 976 / 8 = 122.
7. The pay load of the last fragment is 2380 – 976 * 2 = 428 bytes and has bytes 1952 ~ 2379 of the original IP datagram. The offset is 1952/8= 244.
8. Total length of three fragments: 996 + 996 + 448 = 2440 > 2400

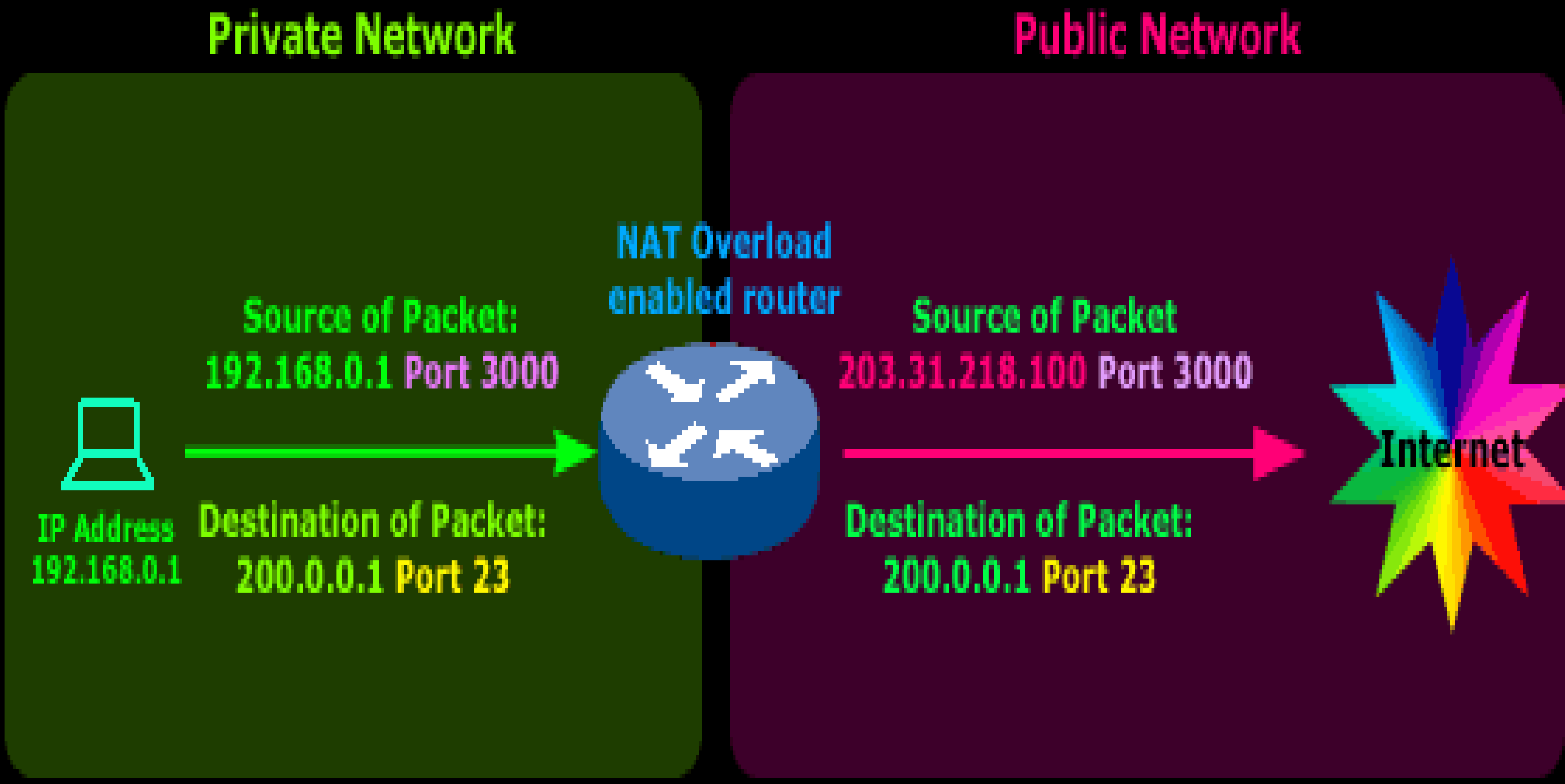
Example of Fragmentation
Example 2:

- For a data packet of 4000 bytes and MTU of 1500 bytes, we have actual data of 3980 bytes that is to be transmitted and 1480 bytes is the maximum data size that is permissible to be sent. So, there would be 3 fragments:
- For the first fragment, data size = 1480 bytes (0-1479), offset = 0 and MF flag = 1
- For the second fragment, data size = 1480 bytes (1480-2959), offset = 185(1480/ 8) and MF flag = 1
- For the third fragment, data size = 3980-1480*2=1020 bytes (2960-3979), offset = 370 (2960/8) and MF flag = 0

Network Address Translation

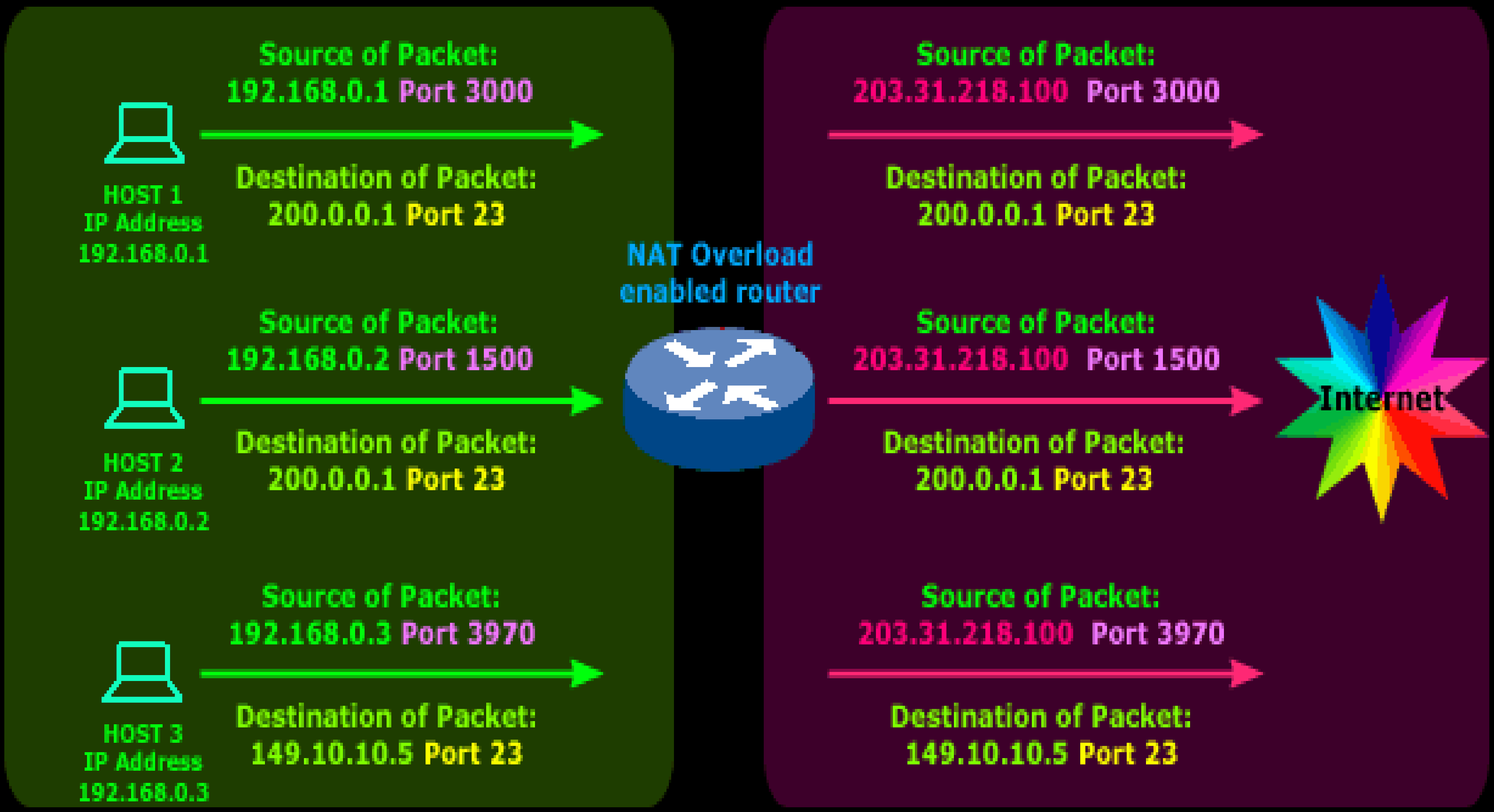
WWW Resource: http://www.firewall.cx/modules.php?name=Alternative_Menu

Understanding How NAT Overload Works



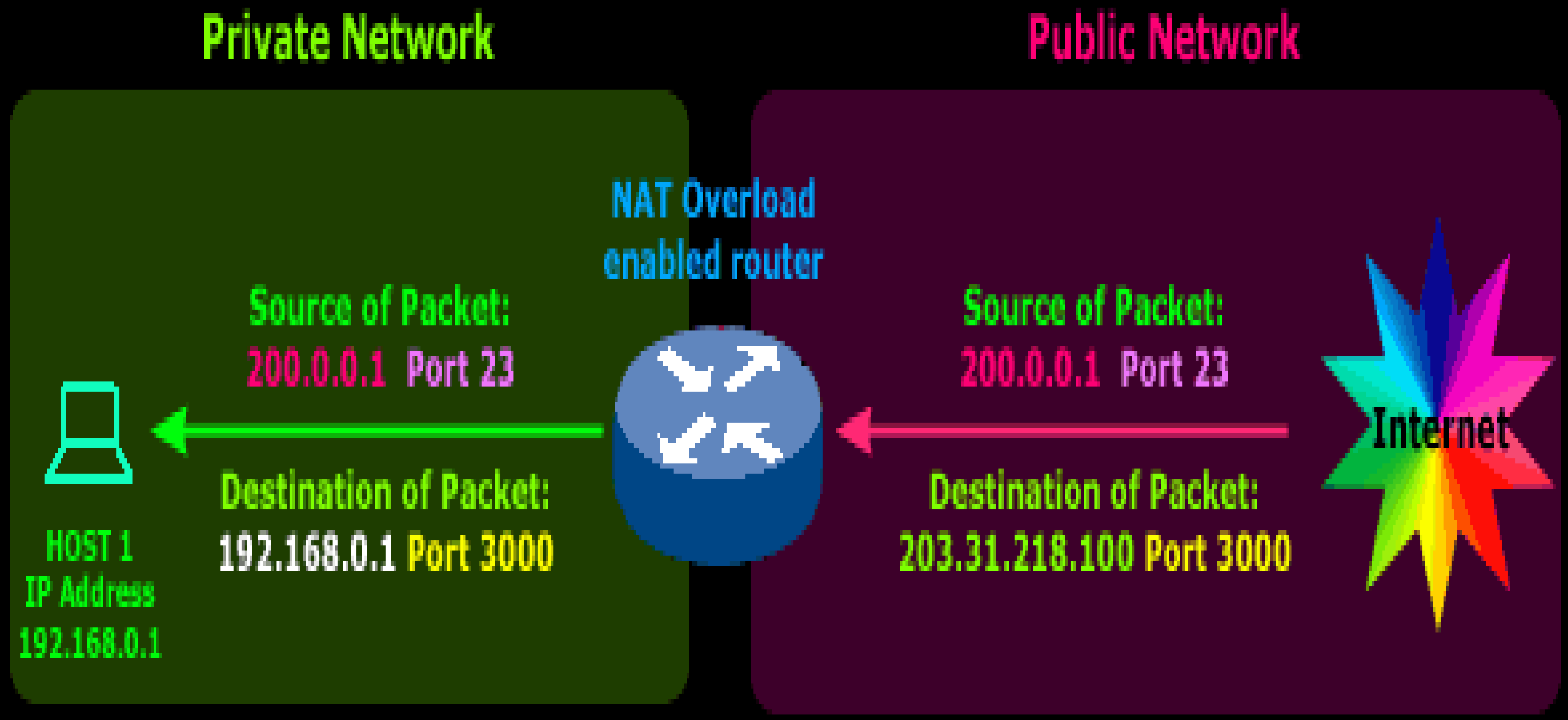
This diagram shows what happens to a packet that traverses a NAT Overload enabled router.

Unleashing the Power of NAT Overload



Here we see how the NAT Overload router deals with multiple packets from 3 different hosts on the private network. Notice that only the Source IP Address field is changed as the packets traverse the router.

Unleashing the Power of NAT Overload



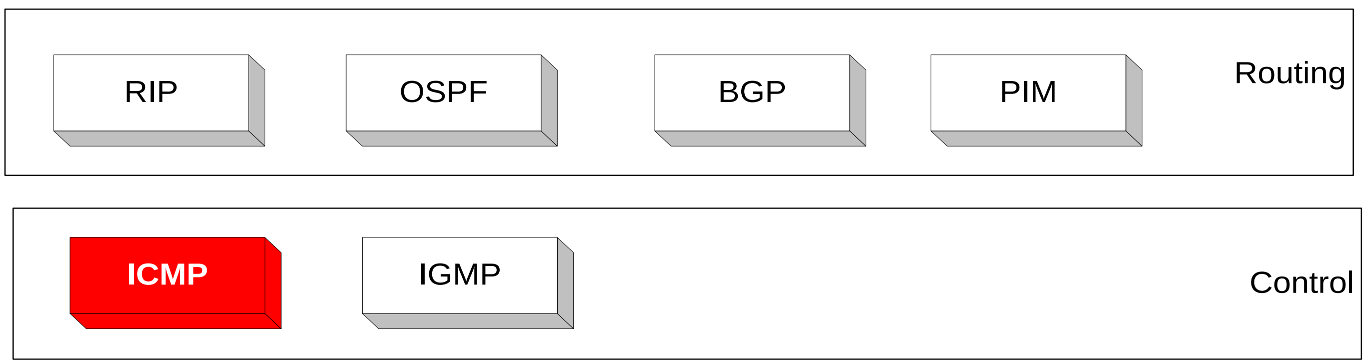
The reply from the Internet server arrives at our router's public interface. The router accepts the packet and, after checking where it's come from and its destination port, which is 3000, it recognises this as a reply to the packet Host 1 sent earlier. The router modifies the Destination IP Address to that of Host 1's IP Address and forwards the packet.

Internet Control Message Protocol (ICMP)

Based on the slides of Dr. Jorg Liebeherr, University of Virginia

Overview

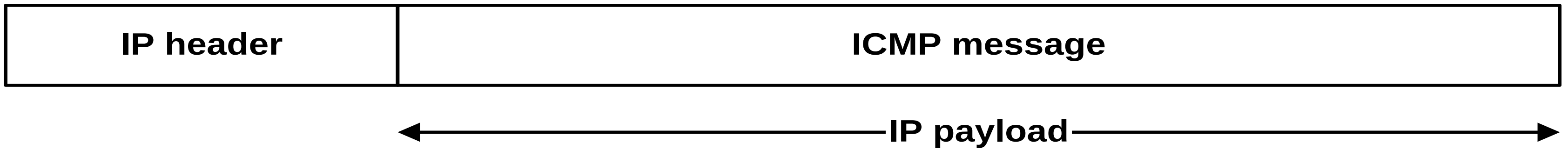
- The IP (Internet Protocol) relies on several other protocols to perform necessary control and routing functions:
  - Control functions (ICMP)
  - Multicast signaling (IGMP)
  - Setting up routing tables (RIP, OSPF, BGP, PIM, …)



Internet Control Message Protocol (ICMP)
Internet Group Management Protocol (IGMP)
Routing Information Protocol (RIP)
Open Shortest Path First (OSPF)
Border Gateway Protocol (BGP)
Protocol-Independent Multicast (PIM)

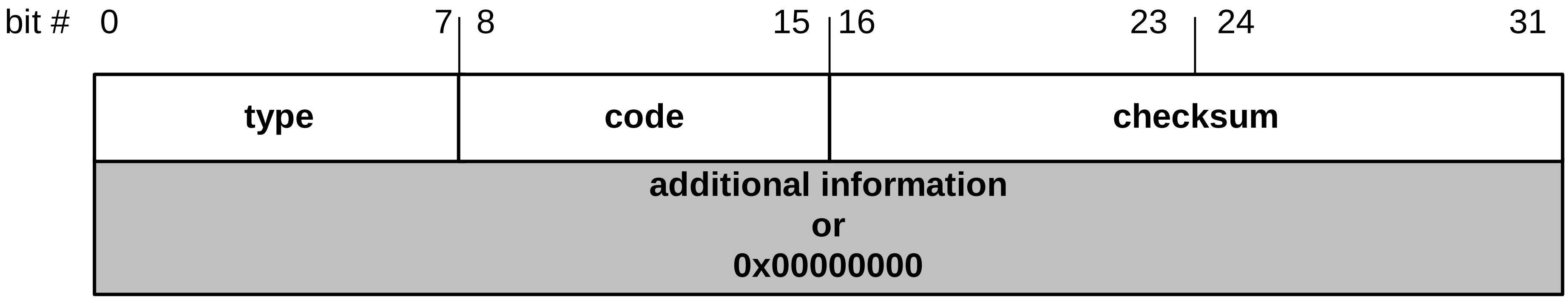
Overview

- The **Internet Control Message Protocol (ICMP)** is a helper protocol that supports IP with facility for
  - Error reporting
  - Simple queries



- ICMP messages are encapsulated as IP datagrams:

ICMP message format



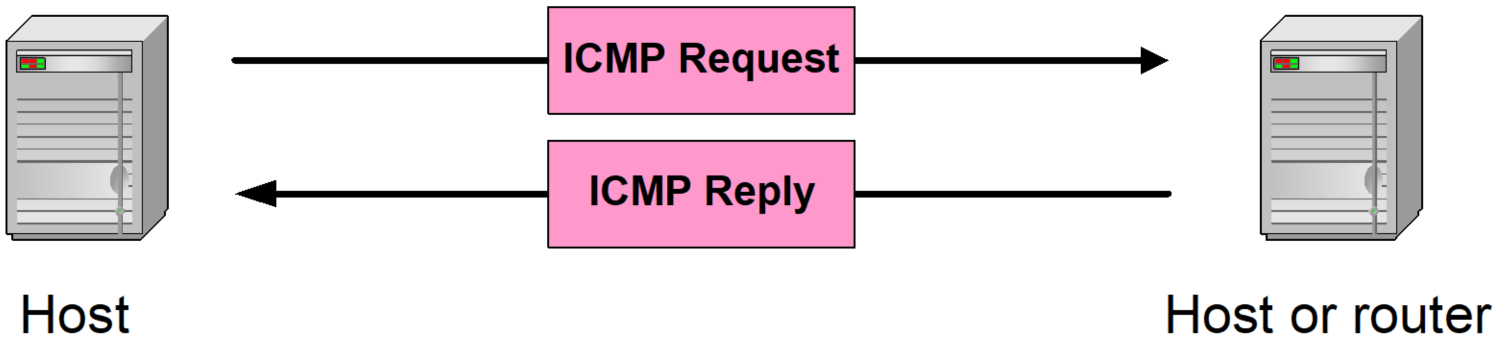
4 byte header:

- Type (1 byte): type of ICMP message
- Code (1 byte): subtype of ICMP message
- Checksum (2 bytes): similar to IP header checksum. Checksum is calculated over entire ICMP message

If there is no additional data, there are 4 bytes set to zero.

→ each ICMP messages is at least 8 bytes long

ICMP Query message



ICMP query:

- Request sent by host to a router or host
- Reply sent back to querying host

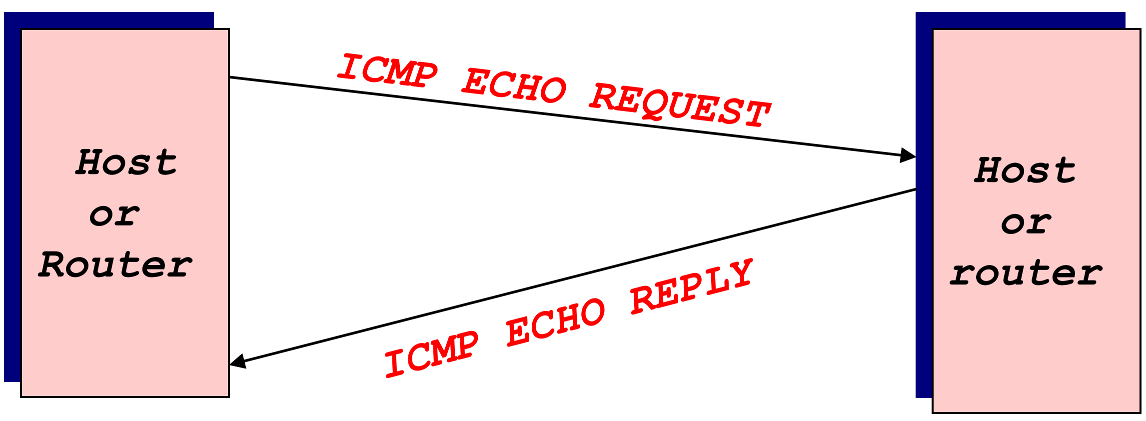
Example of ICMP Queries

Type/Code:	Description
8/0	Echo Request
0/0	Echo Reply
13/0	Timestamp Request
14/0	Timestamp Reply
10/0	Router Solicitation (حث)
9/0	Router Advertisement

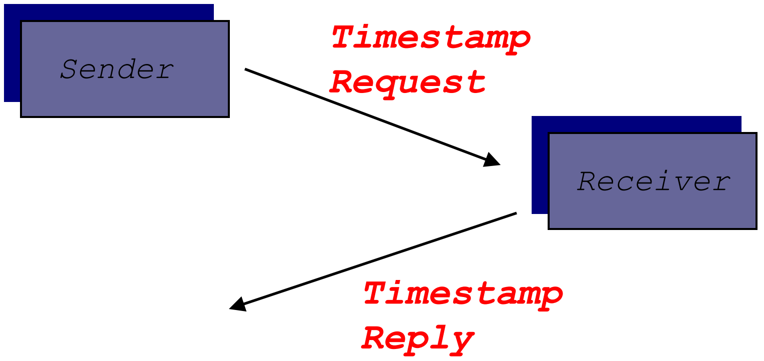
The ping command uses Echo Request/ Echo Reply

Example of a Query: Echo Request and Reply

- Ping's are handled directly by the kernel
- Each Ping is translated into an ICMP Echo Request
- The Ping'ed host responds with an ICMP Echo Reply



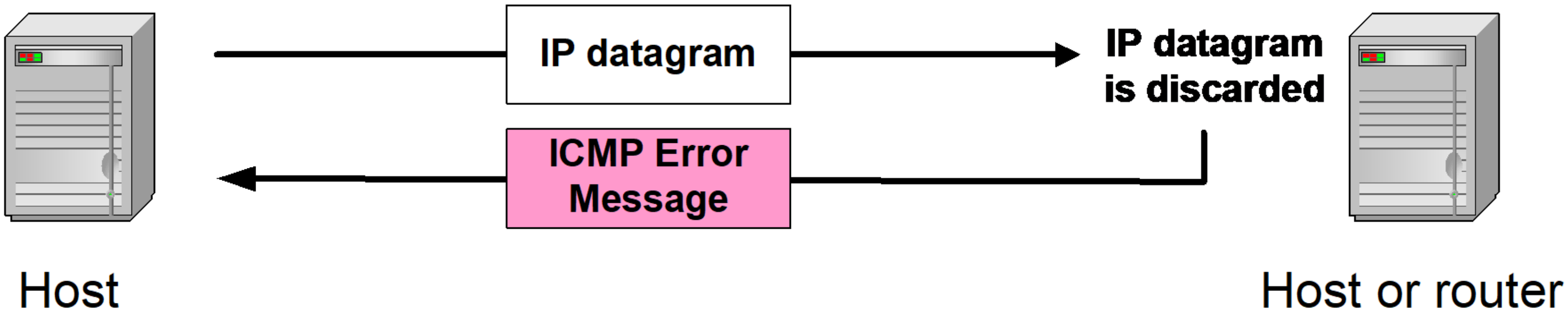
Example of a Query: ICMP Timestamp



- A system (host or router) asks another system for the current time.
- Time is measured in milliseconds after midnight UTC (Universal Coordinated Time) of the current day
- Sender sends a request, receiver responds with reply

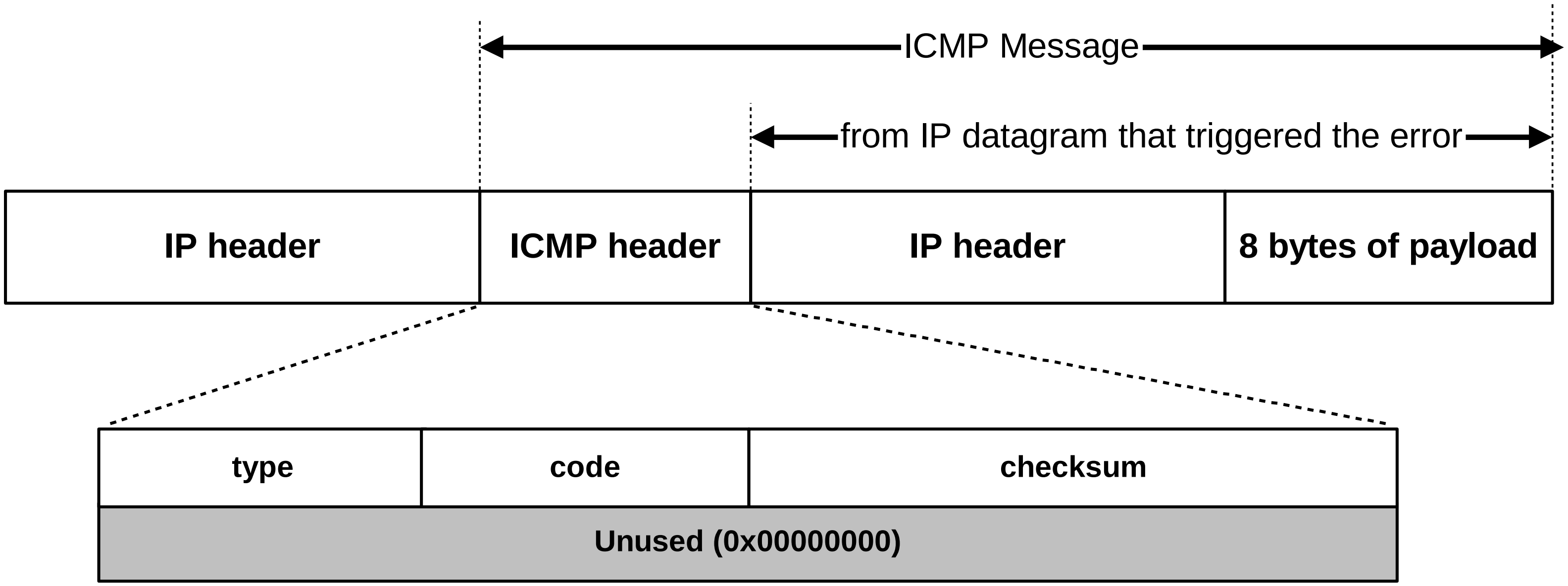
Type (= 17 or 18)	Code (=0)	Checksum
identifier		sequence number
32-bit sender timestamp		
32-bit receive timestamp		
32-bit transmit timestamp		

ICMP Error message



- **ICMP error messages report error conditions**
- **Typically sent when a datagram is discarded**
- **Error message is often passed from ICMP to the application program**

ICMP Error message



- ICMP error messages include the complete IP header and the first 8 bytes of the payload (typically: UDP, TCP)

Frequent ICMP Error message

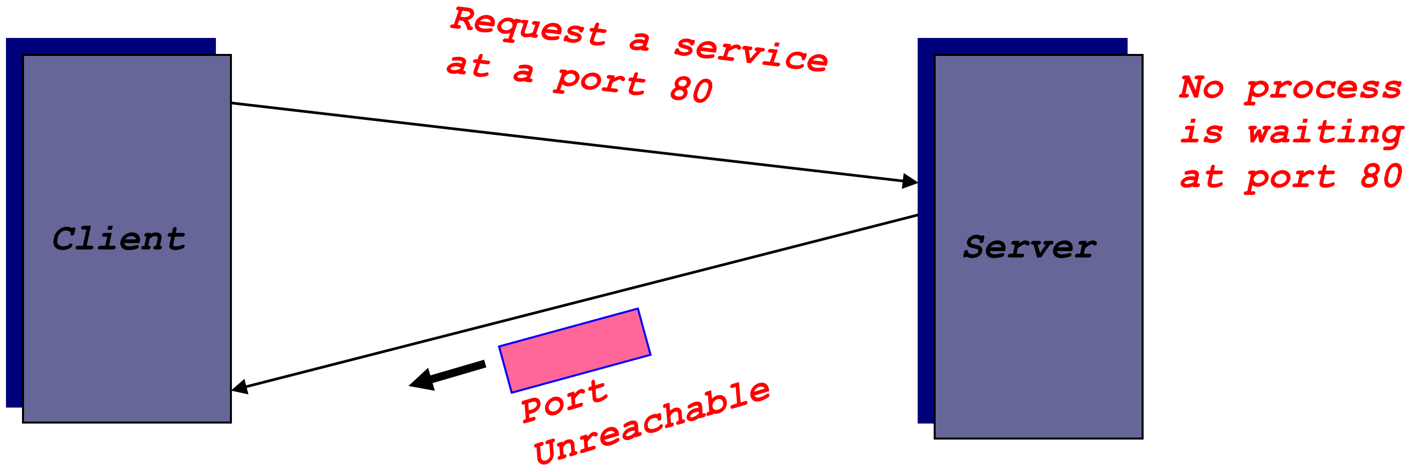
Type	Code	Description	
3	0–15	Destination unreachable	Notification that an IP datagram could not be forwarded and was dropped. The code field contains an explanation.
5	0–3	Redirect	Informs about an alternative route for the datagram and should result in a routing table update. The code field explains the reason for the route change.
11	0, 1	Time exceeded	Sent when the TTL field has reached zero (Code 0) or when there is a timeout for the reassembly of segments (Code 1)
12	0, 1	Parameter problem	Sent when the IP header is invalid (Code 0) or when an IP header option is missing (Code 1)

Some subtypes of the "Destination Unreachable"

Code	Description	Reason for Sending
0	Network Unreachable	No routing table entry is available for the destination network.
1	Host Unreachable	Destination host should be directly reachable, but does not respond to ARP Requests.
2	Protocol Unreachable	The protocol in the protocol field of the IP header is not supported at the destination.
3	Port Unreachable	The transport protocol at the destination host cannot pass the datagram to an application.
4	Fragmentation Needed and DF Bit Set	IP datagram must be fragmented, but the DF bit in the IP header is set.

Example: ICMP Port Unreachable

- RFC 792: If, in the destination host, the IP module cannot deliver the datagram because the indicated protocol module or process port is not active, the destination host may send a destination unreachable message to the source host.
- Scenario:

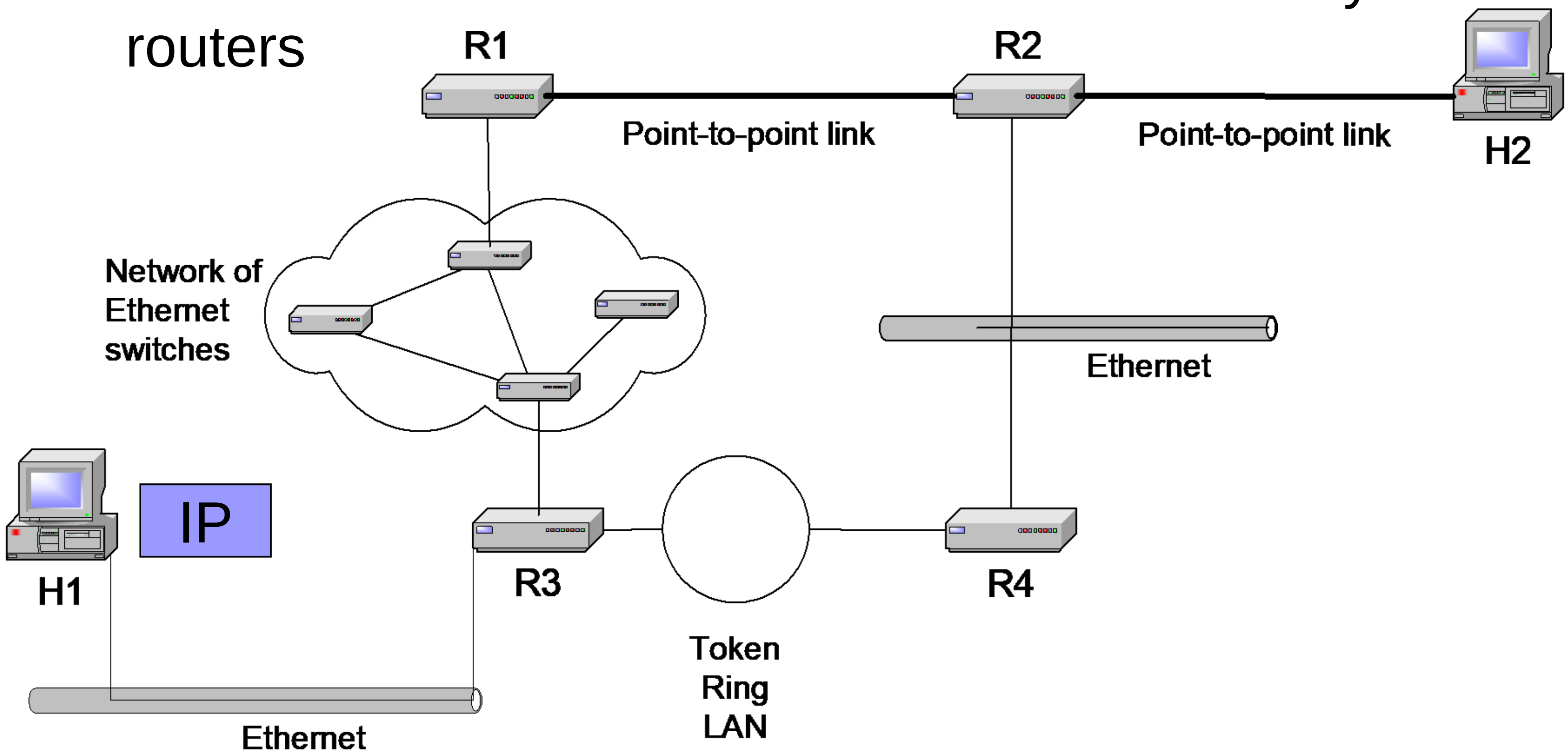


IP Forwarding

Based on the slides of Dr. Jorg Liebeherr, University of Virginia

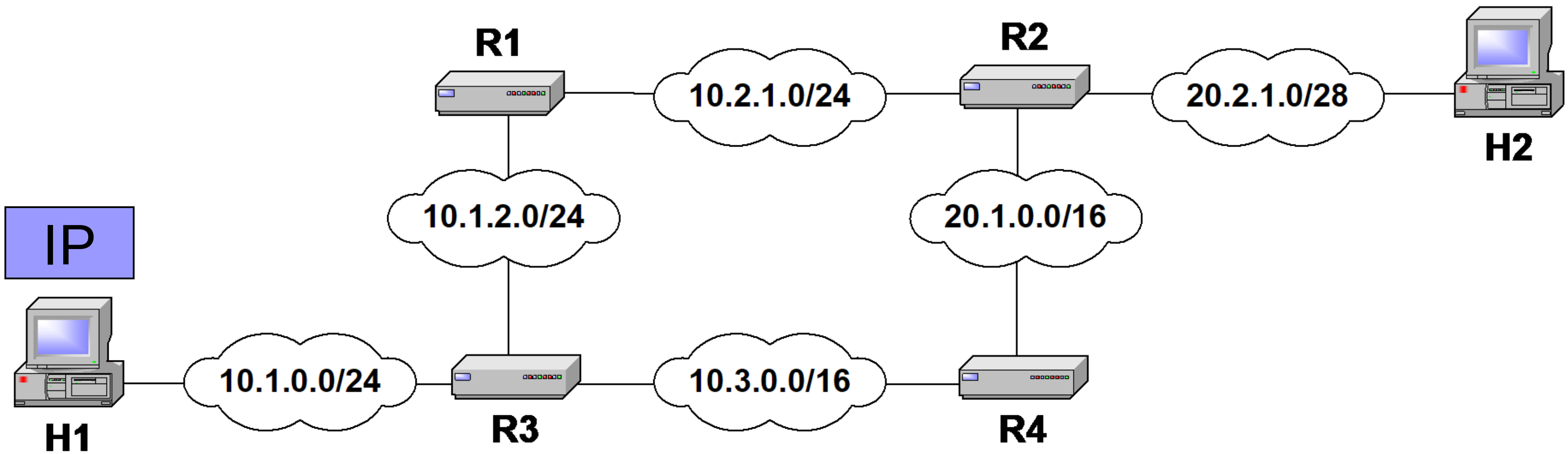
Delivery of an IP datagram

- View at the data link layer:
  - Internetwork is a collection of LANs or point-to-point links or switched networks that are connected by routers



Delivery of an IP datagram

- View at the IP layer:
  - An IP network is a logical entity with a network number
  - We represent an IP network as a "cloud"
  - The IP delivery service takes the view of clouds, and ignores the data link layer view



Tenets of end-to-end delivery of datagrams

The following conditions must hold so that an IP datagram can be successfully delivered

1. The network prefix of an IP destination address must correspond to a unique data link layer network (=LAN or point-to-point link or switched network).
   (The reverse need not be true!)
2. Routers and hosts that have a common network prefix must be able to exchange IP dagrams using a data link protocol (e.g., Ethernet, PPP)
3. Every data link layer network must be connected to at least one other data link layer network via a router.

Routing tables

- Each router and each host keeps a **routing table** which tells the router how to process an outgoing packet
- Main columns:
    1. **Destination address:** where is the IP datagram going to?
    2. **Next hop:** how to send the IP datagram?
    3. **Interface:** what is the output port?
- Next hop and interface column can often be summarized as one column
- Routing tables are set so that datagrams gets closer to the its destination

Routing table of a host or router

IP datagrams can be directly delivered ("direct") or is sent to a router ("R4")

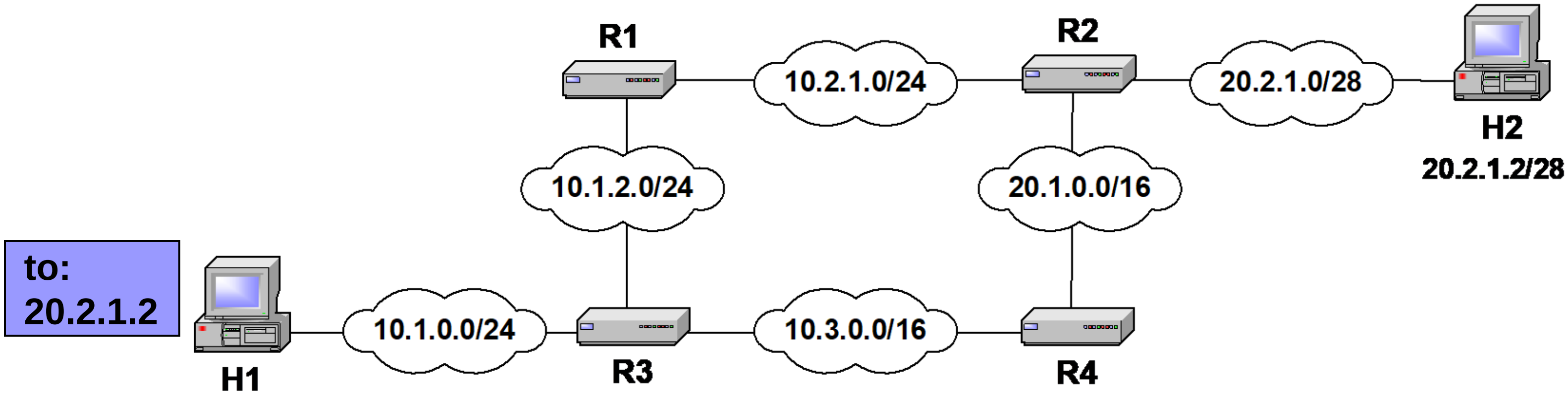
Destination	Next Hop	interface
10.1.0.0/24	direct	eth0
10.1.2.0/24	direct	eth0
10.2.1.0/24	R4	serial0
10.3.1.0/24	direct	eth1
20.1.0.0/16	R4	eth0
20.2.1.0/28	R4	eth0

Delivery with routing tables

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
30.1.1.0/28	R2

Destination	Next Hop
10.1.0.0/24	R1
10.1.2.0/24	R1
10.2.1.0/24	direct
10.3.1.0/24	R4
20.1.0.0/16	direct
20.2.1.0/28	direct

Destination	Next Hop
10.1.0.0/24	R2
10.1.2.0/24	R2
10.2.1.0/24	R2
10.3.1.0/24	R2
20.1.0.0/16	R2
20.2.1.0/28	direct



Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	R3
10.2.1.0/24	R3
10.3.1.0/24	R3
20.1.0.0/16	R3
20.2.1.0/28	R3

Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	direct
10.2.1.0/24	R4
10.3.1.0/24	direct
20.1.0.0/16	R4
20.2.1.0/28	R4

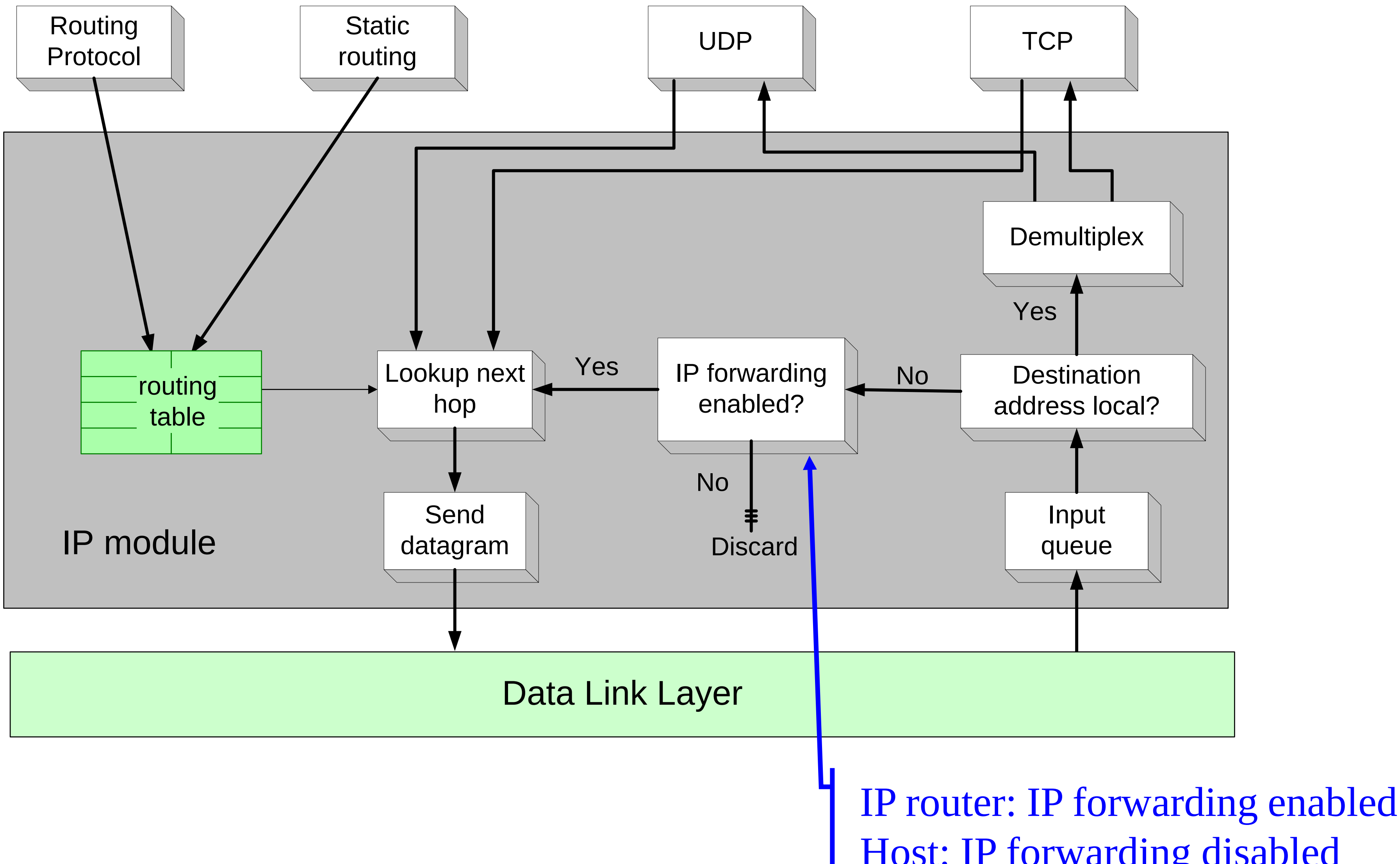
Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	R3
10.2.1.0/24	R2
10.3.1.0/24	direct
20.1.0.0/16	direct
20.2.1.0/28	R2

Delivery of IP datagrams

- There are two distinct processes to delivering IP datagrams:
  1. **Forwarding:** How to pass a packet from an input interface to the output interface?
  2. **Routing:** How to find and setup the routing tables?

- Forwarding must be done as fast as possible:
  - on routers, is often done with support of hardware
  - on PCs, is done in kernel of the operating system
- Routing is less time-critical
  - On a PC, routing is done as a background process

Processing of an IP datagram in IP



Processing of an IP datagram in IP

- Processing of IP datagrams is very similar on an IP router and a host
- **Main difference:**
  "IP forwarding" is enabled on router and disabled on host

- **IP forwarding enabled**
  → if a datagram is received, but it is not for the local system, the datagram will be sent to a different system
- **IP forwarding disabled**
  → if a datagram is received, but it is not for the local system, the datagram will be dropped

Processing of an IP datagram at a router

Receive an IP datagram

1. IP header validation
2. Process options in IP header
3. Parsing the destination IP address
4. Routing table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop
9. Send ICMP packet (if necessary)

Routing table lookup

- When a router or host need to transmit an IP datagram, it performs a routing table lookup

- **Routing table lookup:** Use the IP destination address as a key to search the routing table.

- Result of the lookup is the IP address of a next hop router, and/or the name of a network interface

Destination address	**Next hop/ interface**
network prefix *or* host IP address *or* loopback address *or* default route	IP address of next hop router *or* Name of a network interface

Loopback: 127.0.0.0 to 127.255.255.255, enabling devices to transmit and receive the data packets. The loopback address 127.0.0.1 is generally known as localhost.

Type of routing table entries

- **Network route**
  - Destination addresses is a network address (e.g., 10.0.2.0/24)
  - Most entries are network routes
- **Host route**
  - Destination address is an interface address (e.g., 10.0.1.2/32)
  - Used to specify a separate route for certain hosts
- **Default route**
  - Used when no network or host route matches
  - The router that is listed as the next hop of the default route is the **default gateway (for Cisco: "gateway of last resort ملتجأ)**
- **Loopback address**
  - Routing table for the loopback address (127.0.0.1)
  - The next hop lists the loopback (lo0) interface as outgoing interface

Routing table lookup: Longest Prefix Match

- **Longest Prefix Match:** Search for the routing table entry that has the longest match with the prefix of the destination IP address

1. Search for a match on all 32 bits
2. Search for a match for 31 bits

   …..

32. Search for a match on 0 bits

Host route, loopback entry
→ 32-bit prefix match
Default route is represented as 0.0.0.0/0
→ 0-bit prefix match

128.143.71.21 received by the router

Destination address	Next hop
10.0.0.0/8	R1
128.143.0.0/16	R2
128.143.64.0/20	R3
128.143.192.0/20	R3
128.143.71.0/24	R4
128.143.71.55/32	R3
default	R5

The longest prefix match for 128.143.71.21 is for 24 bits with entry 128.143.71.0/24

Datagram will be sent to R4

Longest Prefix Match

- The router receives a packet with a destination IP address of 192.168.1.33.
- The routing table contains the following possible matches:
  192.168.1.32/28
  192.168.1.0/24
  192.168.0.0/16

- To determine the longest match, it's easiest to convert the IP addresses to binary and compare them

Address	Converted Binary Address
192.168.1.33 (destination IP address)	11000000.10101000.00000001.00100001
192.168.1.32/28	**11000000.10101000.00000001.001**00000
192.168.1.0/24	**11000000.10101000.00000001.**00000000
192.168.0.0/16	**11000000.10101000.**00000000.00000000

192.168.1.32/28 Best match

Route Aggregation (تجميع)

- Longest prefix match algorithm permits to aggregate prefixes with identical next hop address to a single entry
- This contributes significantly to reducing the size of routing tables of Internet routers

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
20.1.1.0/28	R2

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.0.0.0/8	R2

How do routing tables get updated?

- Adding an interface:
  - Configuring an interface eth2 with 10.0.2.3/24 adds a routing table entry:

Destination	Next Hop/ interface
10.0.2.0/24	eth2

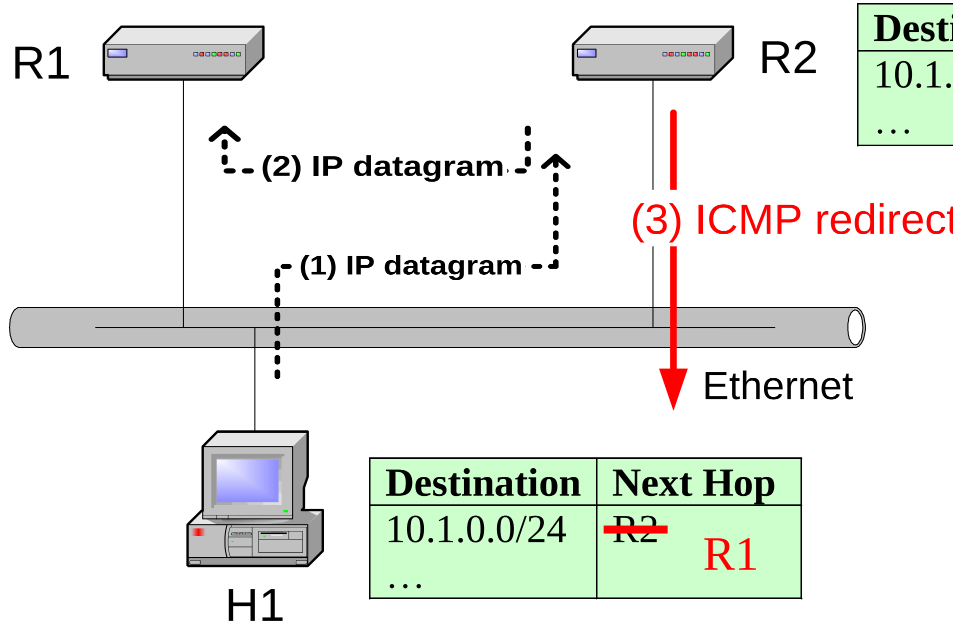
- Adding a default gateway:
  - Configuring 10.0.2.1 as the default gateway adds the entry:

Destination	Next Hop/ interface
0.0.0.0/0	10.0.2.1

- Static configuration of network routes or host routes
- Update of routing tables through routing protocols
- ICMP messages

Routing table manipulations with ICMP

- When a router detects that an IP datagram should have gone to a different router, the router (here R2)
  - forwards the IP datagram to the correct router
  - sends an ICMP redirect message to the host
- Host uses ICMP message to update its routing table

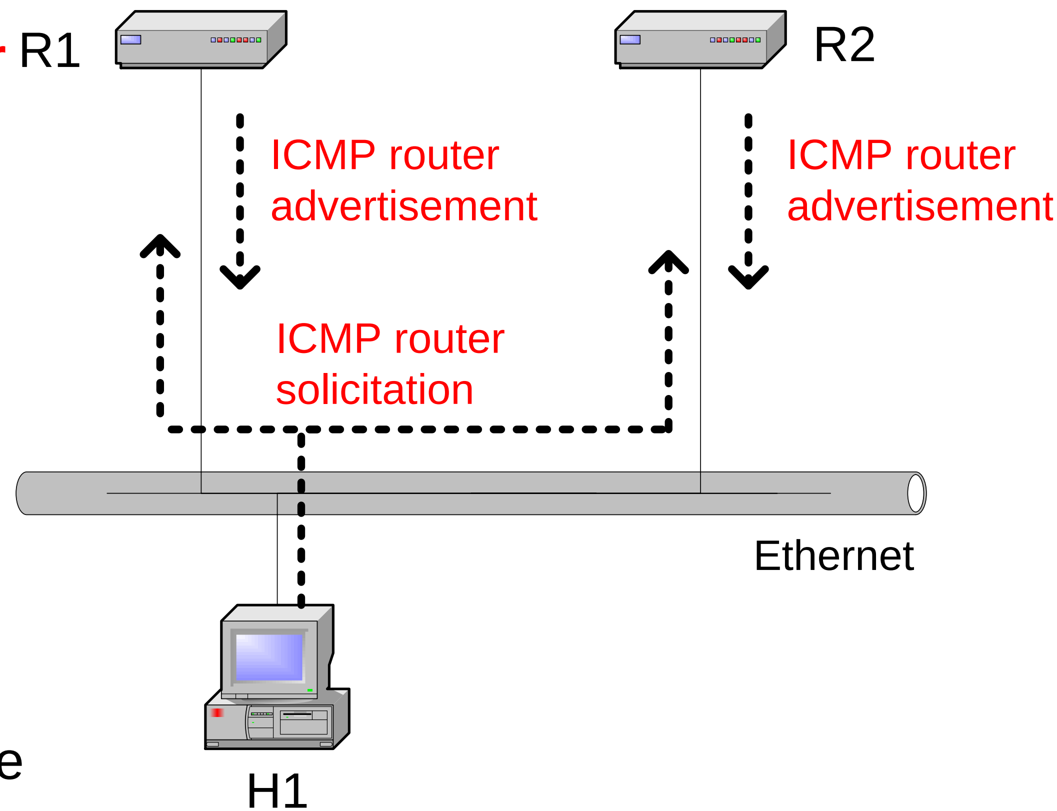


Destination	Next Hop
10.1.0.0/24	R1
…	

Destination	Next Hop
10.1.0.0/24	~~R2~~ R1
…	

ICMP Router Solicitation
ICMP Router Advertisement

- After bootstrapping a host broadcasts an **ICMP router solicitation.**
- In response, routers send an **ICMP router advertisement** message
- Also, routers periodically broadcast **ICMP router advertisement**



This is sometimes called the [Router Discovery Protocol]