Network Layer

Rick Graziani

Thanks to Brad Smith for his contributions

Spring 2018

Next week in CE 151

- Sign up for Piazza (piazza.com)
- Sunday
  - First lab due.
  - First lab section is half of the lab grade.
  - 85 points for the lab
  - 85 points for attending first lab section
  - VM Problems
- Tuesday
  - Link layer lecture
- Thursday
  - Read “End-to-End Argument” paper
  - IPv4/layer exercise
  - IPv4 quiz… review **orange** slides

Today

- Too many slides, but will go fast over some and slower on others
- Evolution of TCP/IP - Overview

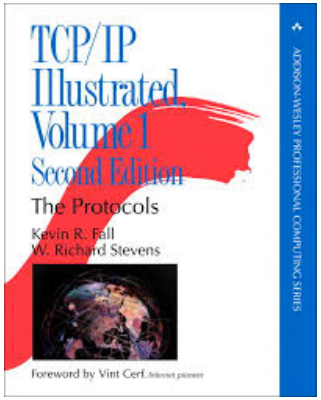
Evolution of TCP/IP (Overview)

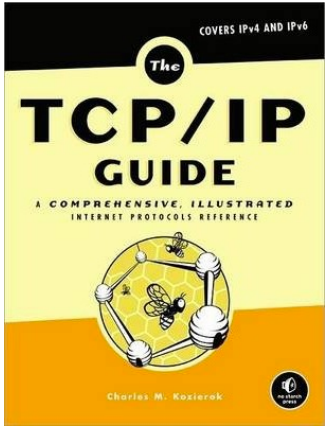
The OSI Model

7	Application	DHCP, DNS, FTP, HTTP, HTTPS, POP, SMTP, SSH, etc…
6	Presentation	
5	Session	
4	Transport	Segment: TCP UDP
3	Network	Datagram: IP Address: IPv4, IPv6
2	Link	Frame: MAC Address
1	Phisical	Ethernet cable, fibre, wireless, coax, etc…

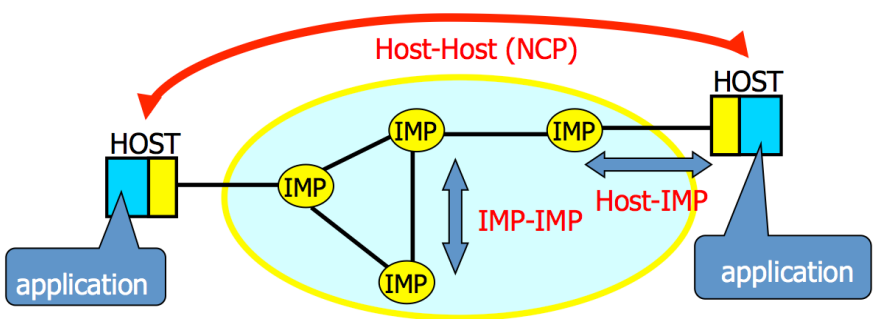
The TCP/IP Model

Application
Transport
Internet
Network Access

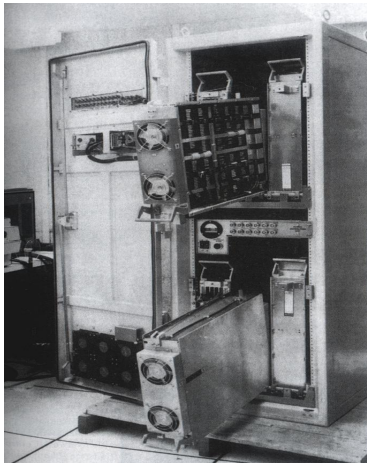




Internet Background

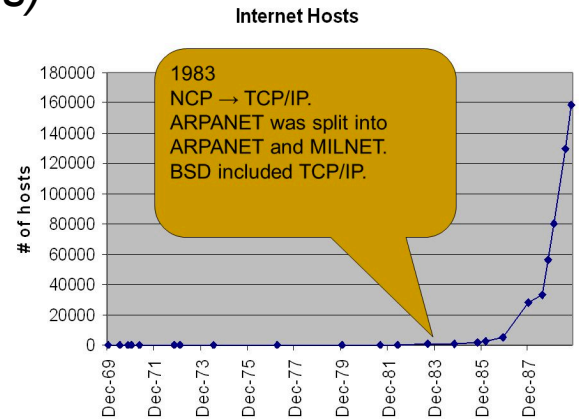


- The ARPAnet was the first **packet-switched** network (1969)
- It's success led to the Internet
- The ARPAnet architecture
  - Homogeneous (same) switches
    - Interface Message Processors (IMPs) – Honeywell 316/516 minicomputers (maps to "physical, data link and network layers")
  - Heterogeneous (different) hosts
  - The **Network Control Protocol or Program** (**NCP**) implemented conservative error, flow, and congestion control (early TCP)
    - Stop and wait
    - Hop by hop

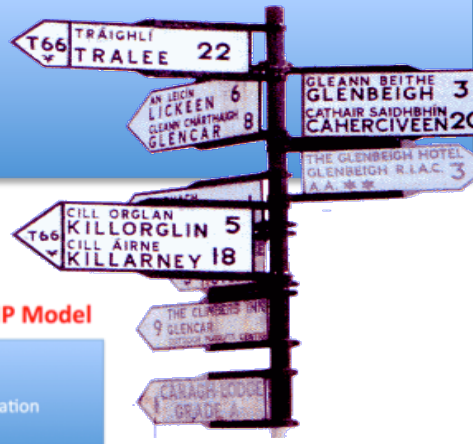


Evolution of TCP/IP (Overview)

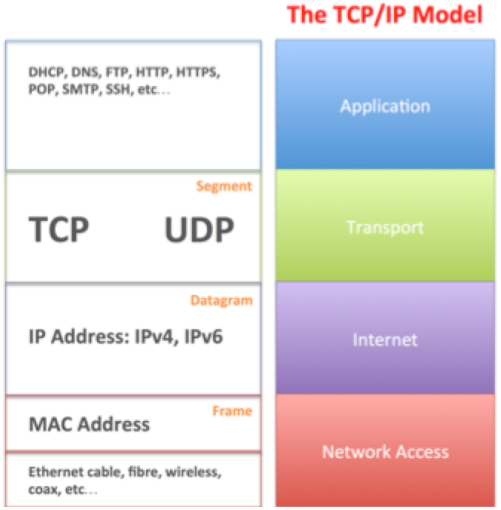
- NCP provided the transport layer:
  - ARPANet Host-to-Host Protocol (AHHP)
  - Initial Connection Protocol (ICP)
- Move from smart network to smart hosts … "*End-to-end Principle" (Work done by Paul Baran and Donald Davies in the 1960s)*
  - Reliable delivery a host-host issue
- TCP split into TCP/IP (Early 1980s)
  - Internet Protocol (IP)
    - Best-effort routing
    - Internet level addressing
  - Transmission Control Protocol (TCP)
    - New, end-to-end error, flow, and congestion control

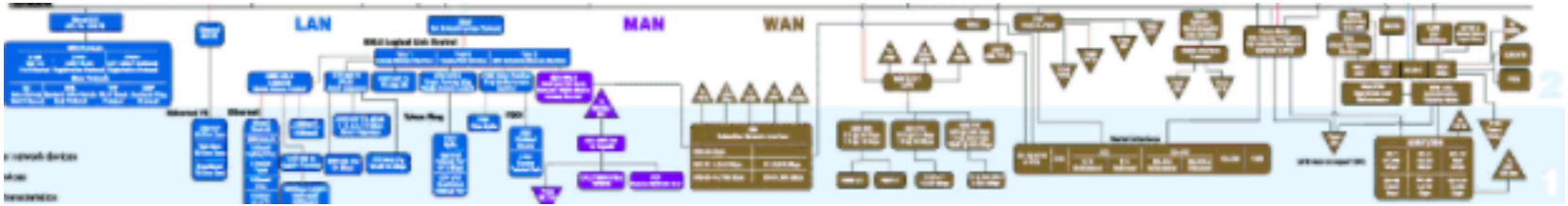


Internet Background



- New technical challenges
  - Routing
  - Error, flow, and congestion control across a network
- ***Link-layer (layer 2) network from today's perspective***
  - IP over anything





Network Working Group
Request for Comments: 1149

D. Waitzman
BBN STC
1 April 1990

A Standard for the Transmission of IP Datagrams on Avian Carriers

Status of this Memo

This memo describes an experimental method for the encapsulation of IP datagrams in avian carriers. This specification is primarily useful in Metropolitan Area Networks. This is an experimental, not recommended standard. Distribution of this memo is unlimited.

Overview and Rational

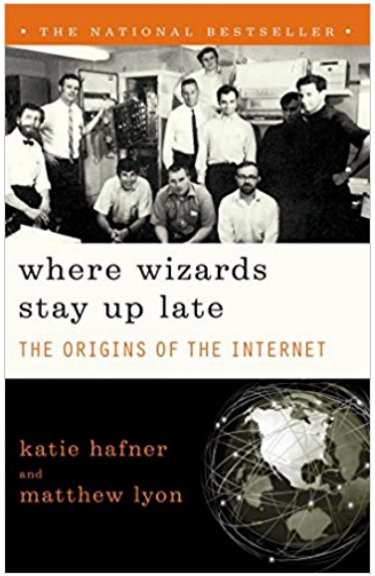
Avian carriers can provide high delay, low throughput, and low altitude service. The connection topology is limited to a single point-to-point path for each carrier, used with standard carriers, but many carriers can be used without significant interference with each other, outside of early spring. This is because of the 3D ether space available to the carriers, in contrast to the 1D ether used by IEEE802.3. The carriers have an intrinsic collision avoidance system, which increases availability. Unlike some network technologies, such as packet radio, communication is not limited to line-of-sight distance. Connection oriented service is available in some cities, usually based upon a central hub topology.

Frame Format

The IP datagram is printed, on a small scroll of paper, in hexadecimal, with each octet separated by whitestuff and blackstuff. The scroll of paper is wrapped around one leg of the avian carrier. A band of duct tape is used to secure the datagram's edges. The bandwidth is limited to the leg length. The MTU is variable, and paradoxically, generally increases with increased carrier age. A typical MTU is 256 milligrams. Some datagram padding may be needed.

Goals of the Internet

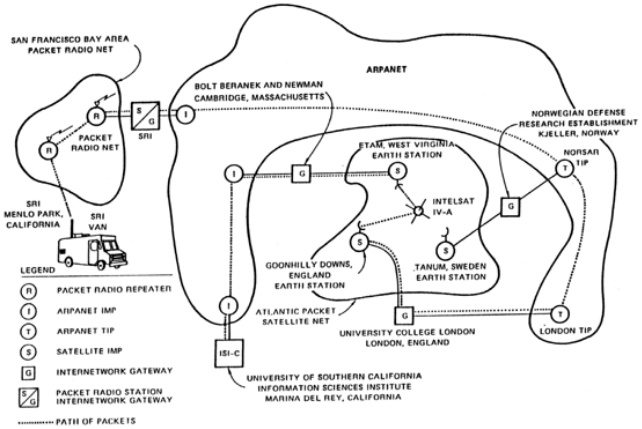
- New, diverse network technologies
  - LANs (e.g Ethernet)
  - DARPA packet radio network (PRNET)
  - DARPA satellite network (SATNET)
- Internet goal
  - Interconnection of diverse networks, which we call ***subnets***
- Minimal-service network model – Best effort
- Vint Cerf and Robert Kahn - “A Protocol for Packet Network Interconnection” (May, 1974)



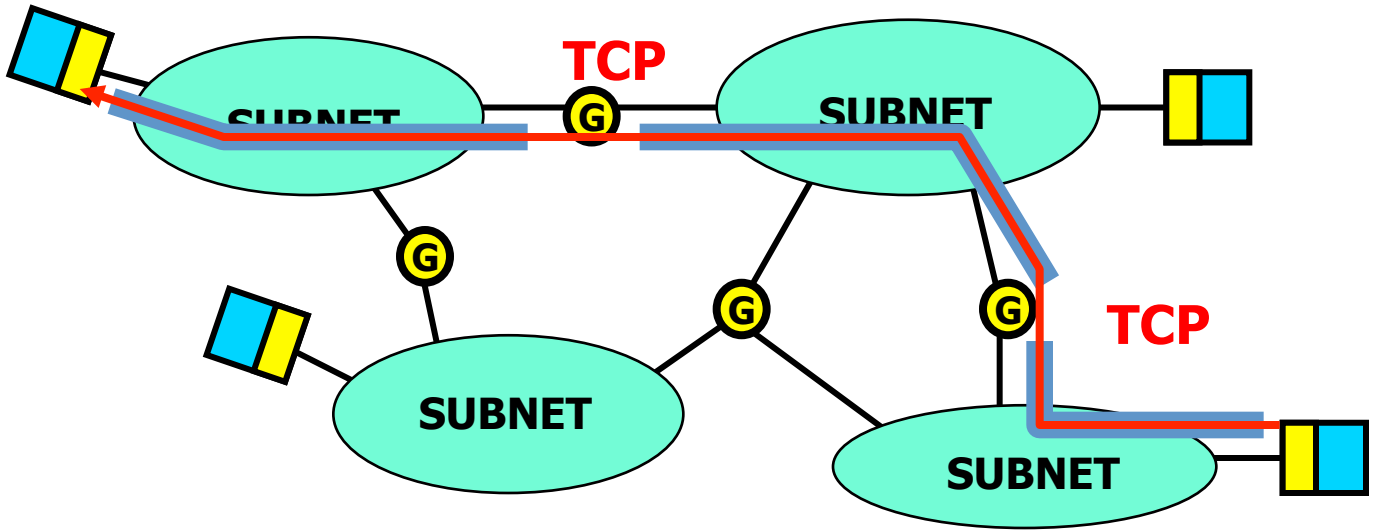
Minimal-Service Network Model

- New concepts
  - “Gateway” interconnects dissimilar networks
  - Internet-level host addressing

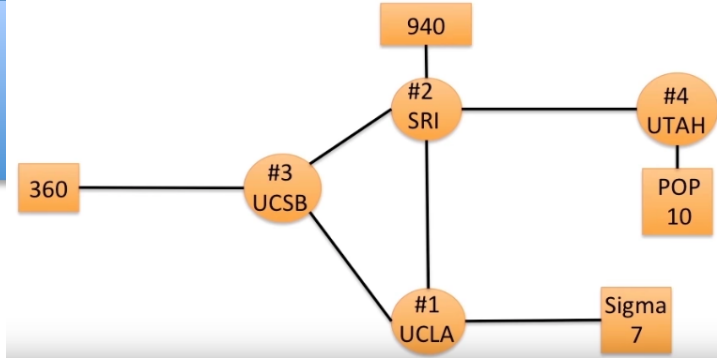
- New functionality
  - Best-effort routing
  - Performance – just switch packets
  - NCP transitions to Transmission Control Protocol (TCP) in 1983
    - End-to-end error control, flow control, sliding windows, and congestion control



The Internet Architecture



Timeline: NCP to TCP



- 1967 – 1969: BBN (Bolt Beranek and Newman) built the ARPANet
- 1969: First node (IMP#1 at UCLA) to a Sigma 7 system
- 1969: NCP (Network Control Protocol or Program)
- 1974: First specification of TCP (included transport and network layers)
- 1978: TCP split into TCP and IP
- 1981: RFC 801 NCP-TCP transition plan
- Reliable delivery a host-host issue
- Move from smart network to smart hosts … “End-to-end Principle”
- Early 1980s: TCP/IP becomes the defacto protocol suite
- 1996: Decided TCP/IP would be the protocol of the Internet (ISO not happy)

Evolution from IPv4 to IPv6

- See slides at the end of this presentation
- ISO's OSI protocols suite, not the TCP/IP protocol suite was suppose to be Internet suite "of the future"
- A lot of protocol politics took place in the mid 1990s

End-to-End Principle

"If a function can completely and correctly be implemented only with the knowledge and help of the application end-points of a communication system, then the function should not be implemented in the communication system itself (although sometimes it may be useful to implement an incomplete version of the function in the communication system as a performance enhancement)."

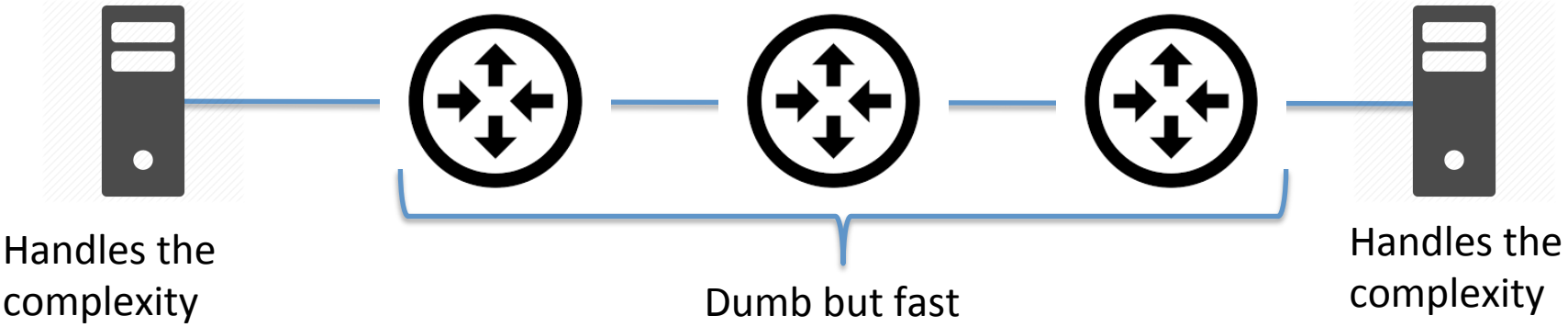
"End-to-End Arguments in System Design"
by Saltzer, Reed, and Clark ('84)

End-to-End Principle

In other words…

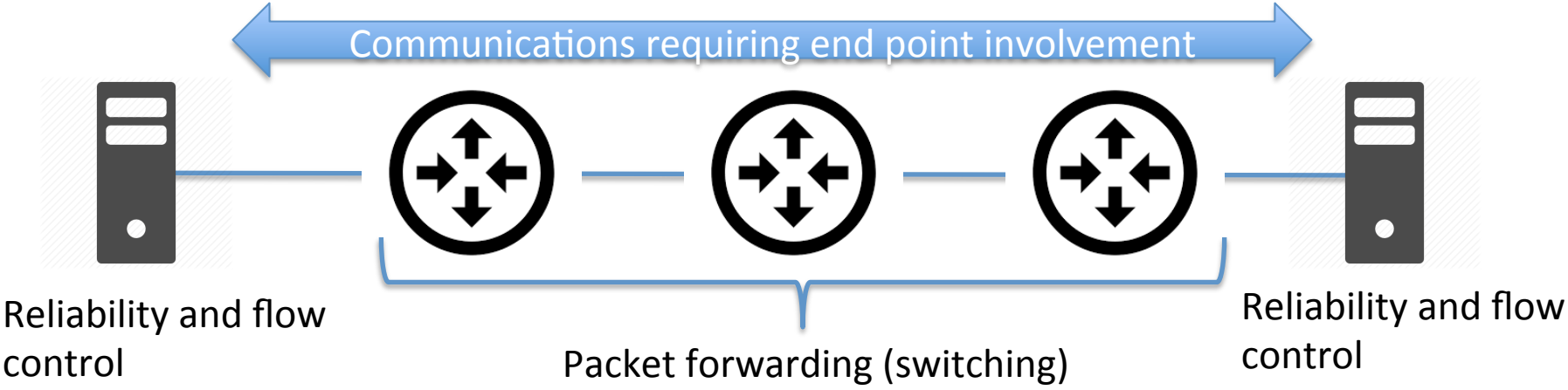
- If a function requires end-point involvement to implement the function completely and correctly, then only implement it in the end points.
- It's all about performance in the network – moving packets as quickly as possible

End-to-End Principle



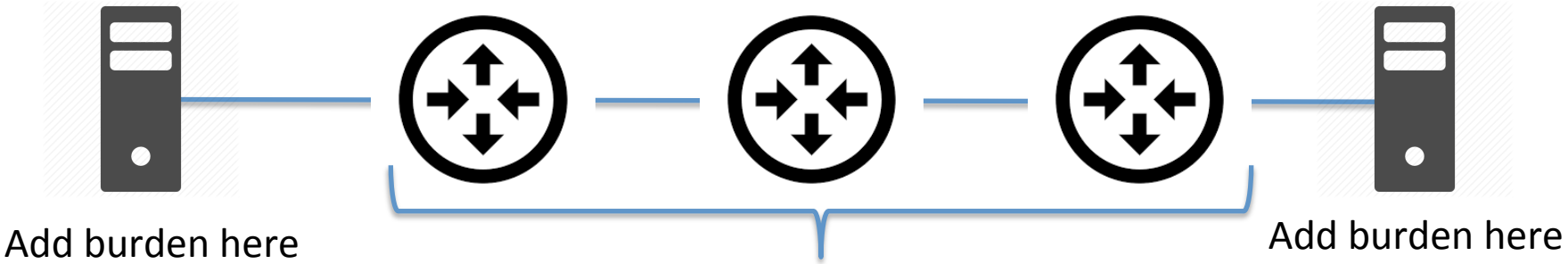
- Avoids impacting applications that don't need this function.
- Reduces complexity of the network…
  - Dumb network, smart end-points… reverse of the telephone system!
- Overall, improves efficiency and reliability of the network.

End-to-End Principle



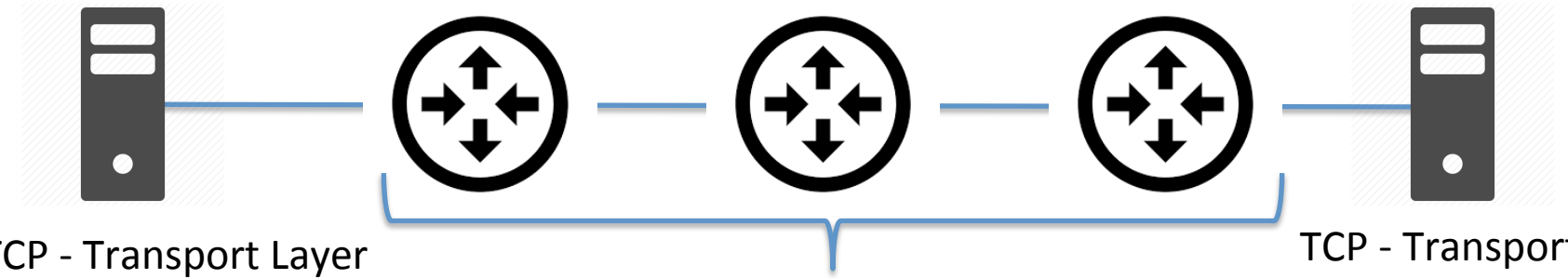
- Transport functions…
  - End-points must be involved to ensure data makes it to the process (corruption in the end-system)
  - Therefore no transport functionality in the network.

End-to-End Principle



- Reasonable interpretation… for functions requiring end-point involvement
  - Think twice before implementing in the network
  - Must be justified as a performance enhancement
  - Doesn't need to be heavy-weight… can depend on end-to-end mechanisms

End-to-End Principle



- This concept led to TCP split into TCP and IP
  - **Internet Protocol (IP)**
    - Best-effort routing
    - Internet level addressing
  - **Transmission Control Protocol (TCP)**
    - New, end-to-end error, flow, and congestion control

Review

- IP goals
  - interconnect diverse network technologies, making minimal assumptions of the underlying networks
  - implement the minimal set of functionality needed to construct an internet… in this sense IP is the waist of the network protocol stack hourglass
- IP implements a datagram, packet-switched model of communications.
- Packet-switch communication involves transmission of digital data
  - in packets
  - no resource reservation... use statistical multiplexing to share a channel

Review

- End-to-End Principle
  - If a function requires end-point involvement to implement the function completely and correctly, then only implement in the end-points!
    - Avoids impacting network applications that don't need this function
    - Reduces complexity of the network... dumb network, smart end-points. Reverse of the telephone system!
    - ***Overall, improves efficiency and reliability of the network.***
    - ***Performance***
  - Reasonable interpretation... if a function requires involvement of end-points...
    - Think twice before implementing it in the network
    - Only justification is as a performance enhancement

Reading Review

- "End-to-End Arguments in System Design" – Salzer, Reed, Clark
  - How does the reliable file transfer problem motivate the end-to-end principle?
  - What is the primary reason for exceptions to the end-to-end principle?
  - What is a common problem with implementing a performance-enhancement in the communication channel?
  - Why is it often acceptable to implement a weak version of end-to-end services as an exception to the end-to-end principle (i.e. outside the end-nodes)?

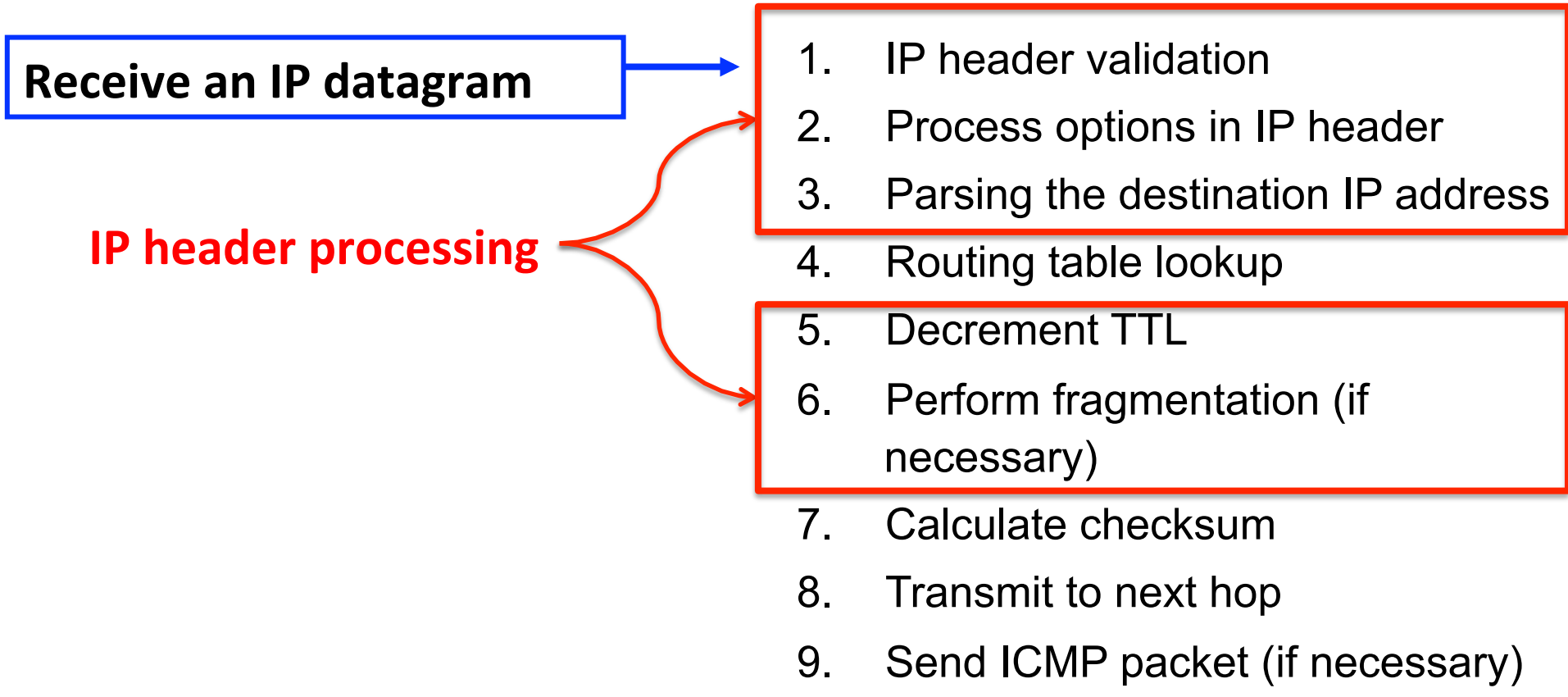
IP Datagram Processing

Receive an IP datagram

Check Version, IHL, Total length, etc.

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)

IP Datagram Processing



IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

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)

IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

Address Resolution Protocol

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)

IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

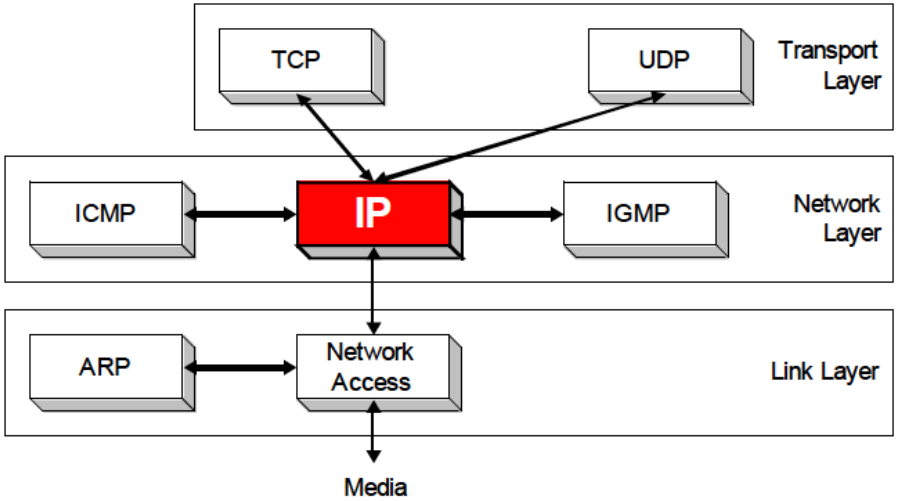
Address Resolution Protocol

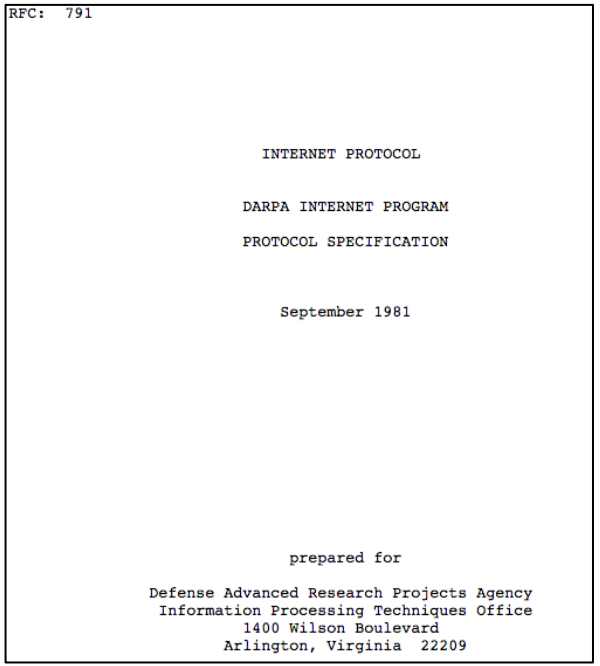
Internet Control Message Protocol

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)

IP - Orientation

- IP (Internet Protocol) is a Network Layer Protocol
- IP's current version is Version 4 (IPv4)
- It is specified in RFC 791
- IPv6 is being deployed now…





RFC: 791

INTERNET PROTOCOL

DARPA INTERNET PROGRAM

PROTOCOL SPECIFICATION

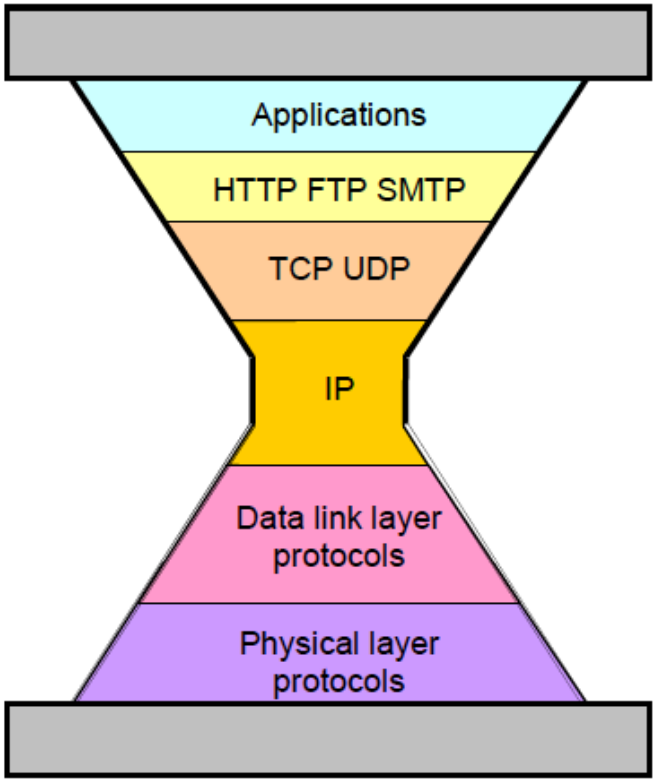
September 1981

prepared for

Defense Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Virginia 22209

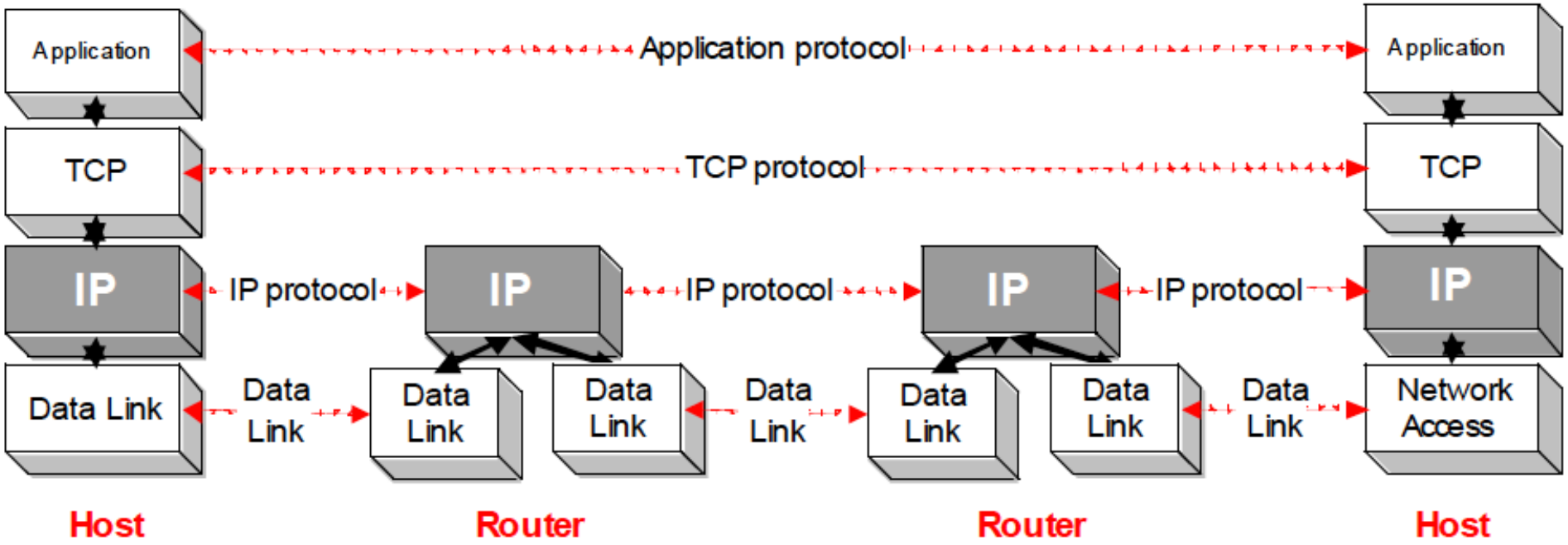
IP: The waist of the hourglass

- IP is the waist of the hourglass of the Internet protocol architecture
- Multiple higher-layer protocols
- Multiple lower-layer protocols
- Only one protocol (two versions) at the network layer.
- Minimum functionality to construct an internet



Highest Layer Hop-by-Hop Protocol

- IP is the highest layer protocol which is implemented at both routers and hosts (hop-by-hop)

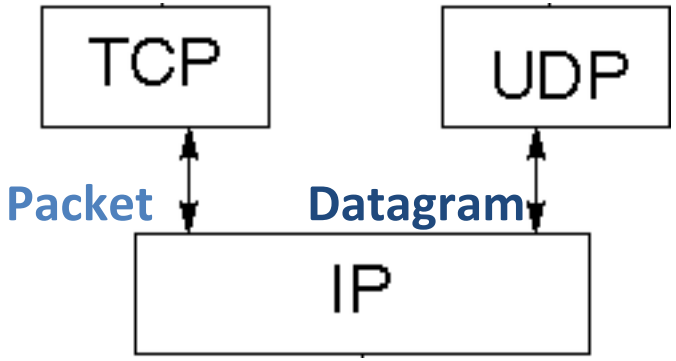


IP Service

- Delivery service of IP is minimal… ***packet-switched*** communication model
  - ***Data sent in packets***
  - ***Best-effort***… (statistically multiplexed) packets can be dropped or delivered out of order)
- IP implements ***datagram*** (packet) flavor of packet-switching
- Distinguishing characteristic of **datagram** is it is **connectionless**
  - Routes computed on an event-driven basis (topology changes)
  - Forwarding decisions done per packet (per flow not required)
  - Different packets in the same flow may follow different paths
  - No per-flow state is "required"
  - Think **telegram**
- ***What is the other form of packet-switching?***

Datagram or Packet

- We use the term ***packet*** when it comes to TCP, connection oriented.
- Whereas, ***datagram*** is a synonym for packets and used in UDP, connectionless.
- ***IP datagrams*** are also referred to as ***IP packets*** by many.



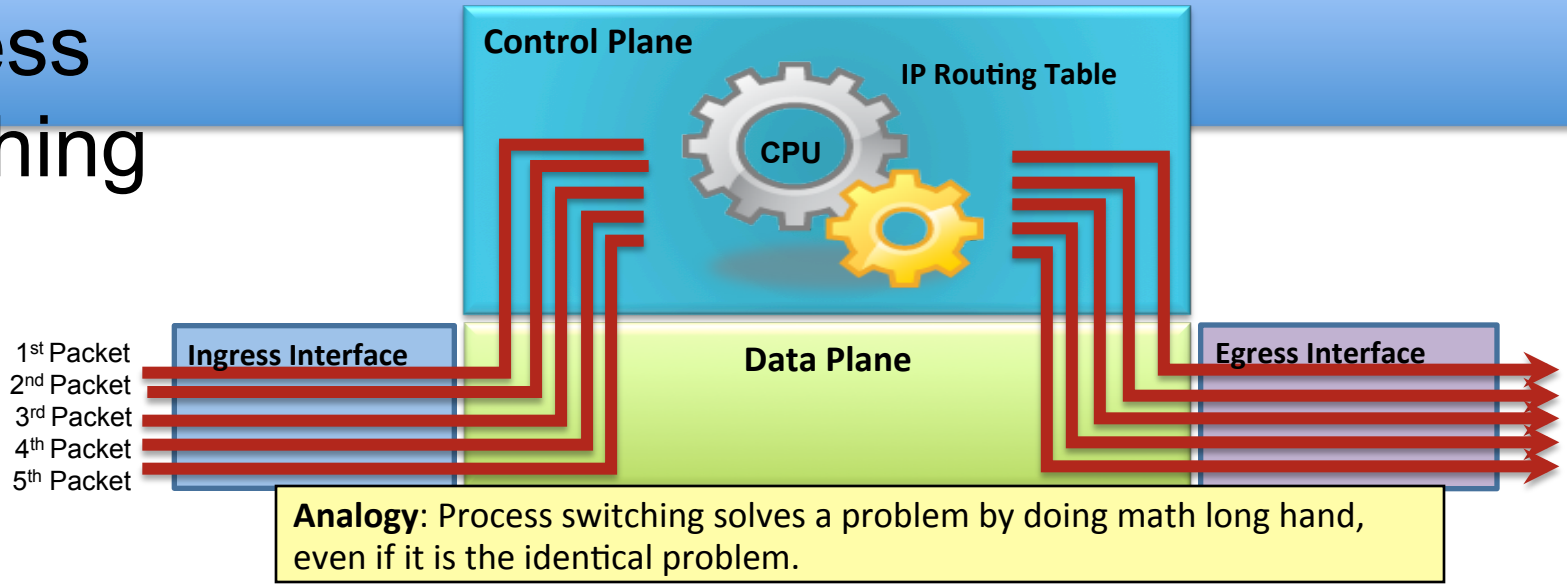
IP Service

- ***Virtual-circuit***
- ***How is virtual-circuit different from datagram in terms of packet switched networks? (Not to be confused with connectionless UDP vs connection oriented TCP)***
- Distinguishing characteristic of **virtual-circuit** is it is **connection-oriented IP service**
  - Route computation and forwarding decisions done ***once per flow***
  - Requires per-flow state
  - Think **telephone-call** without bandwidth reservations
- Consequences of ***datagram*** model packet switching (***routing per packet***)
  - ***Higher layer protocols have to deal with losses or with duplicate packets***
  - ***Packets may be delivered out-of-sequence***

Best Path Decisions

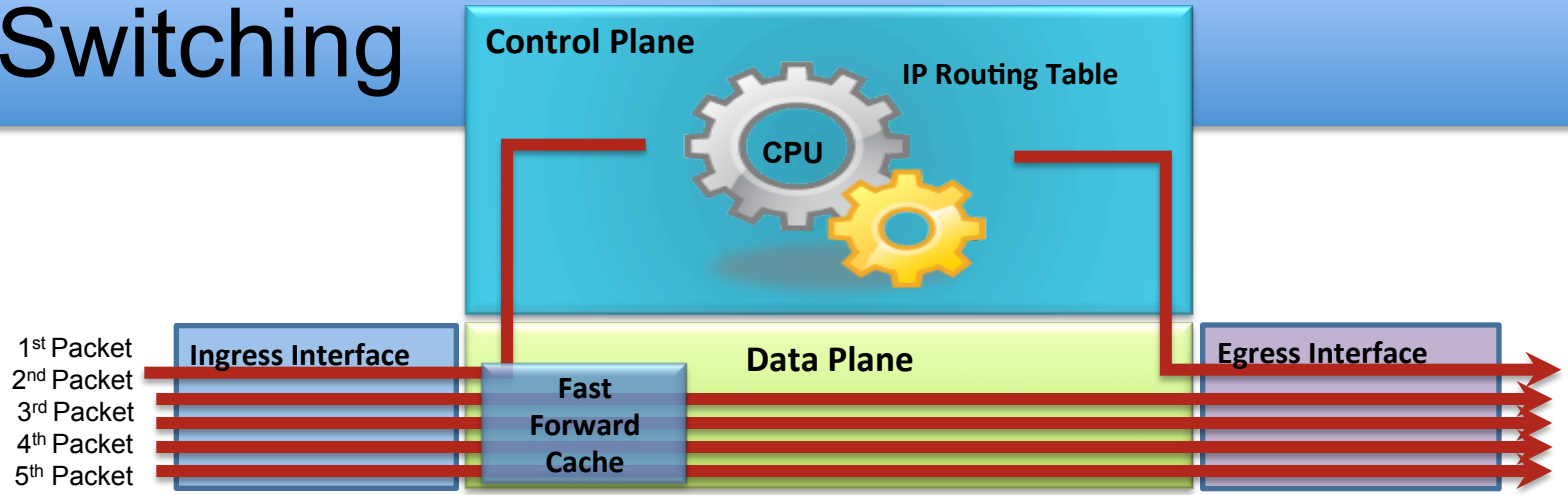
- The **router** uses its routing table to determine the best path to forward the packet.
  - When the **router** receives a packet, it examines its destination IP address and searches for the best network address match in the **routing table**.
  - The **routing table** entries also includes the interface to be used to forward the packet.
  - Once a match is found, the **router** encapsulates the IP packet into the data link frame of the outgoing or exit interface.
  - The **packet** is then forwarded toward its destination.
- Routers support **three packet-forwarding mechanisms** (routers and multilayer switches):
  - **Process switching**
  - **Fast Switching**
  - **Cisco Express Forwarding (CEF)**

Process Switching



- Earliest switching method.
- This is an older packet forwarding mechanism.
  - When a packet arrives on an interface, it is forwarded to the control plane where the CPU examines the routing table, determines the exit interface and forwards the packet.
  - It does this for every packet, even if the destination is the same for a stream of packets.

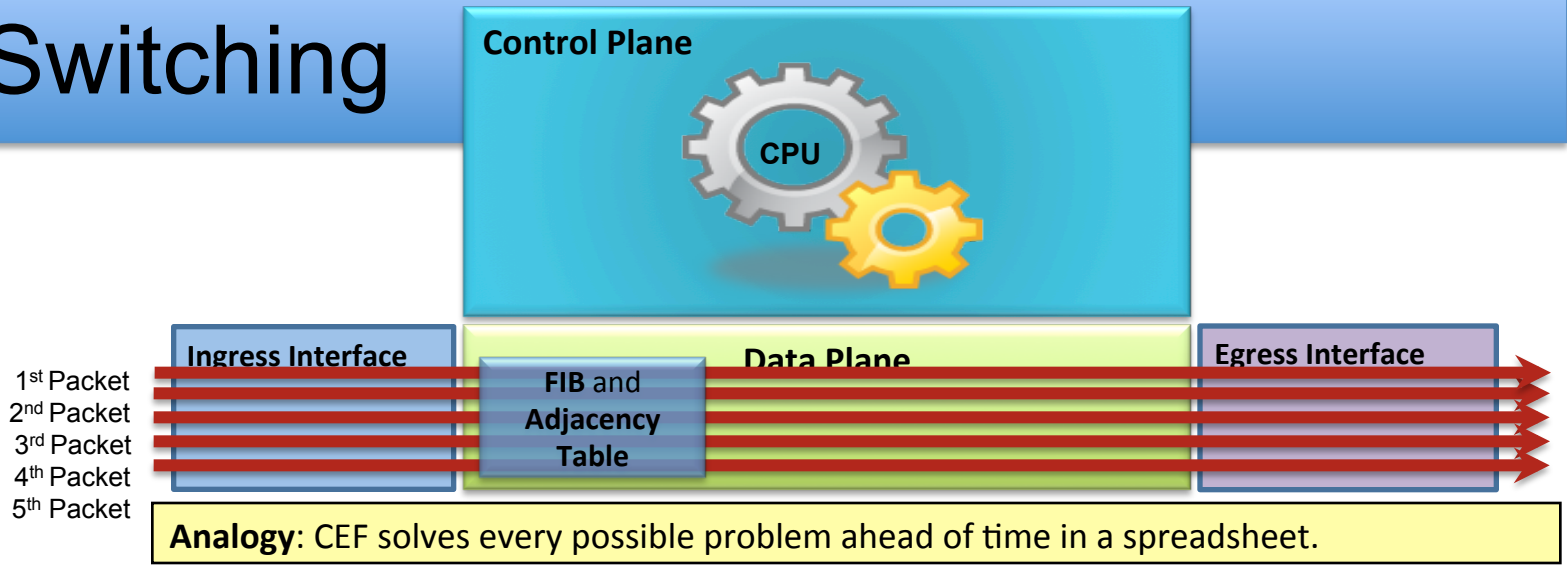
Fast Switching



Analogy: Fast switching solves a problem by doing math long hand one time and remembering the answer for subsequent identical problems.

- As routers had to process more packets, it was determined process switching was not fast enough.
- Next evolution in packet switching was Fast Switching.
  - The first packet is process-switched (CPU + routing table) but it also uses a fast-switching cache to store next-hop information of the flow.
  - The next packets in the flow are forwarded using the cache and without CPU intervention.

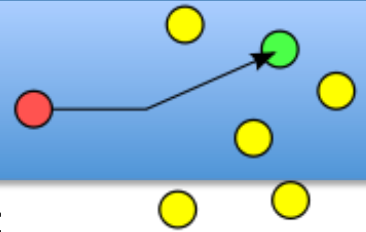
CEF Switching

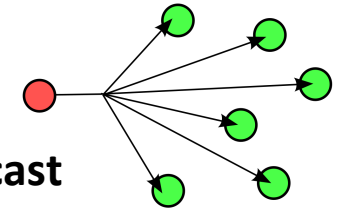


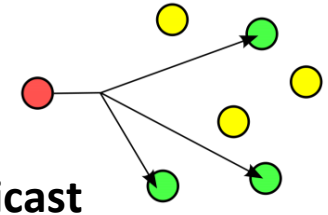
- Preferred and default Cisco IOS packet-forwarding mechanism
  - CEF copies the routing table to the Forwarding Information Base (FIB)
  - CEF creates an adjacency table which contains all the layer 2 information a router would have to consider when forwarding a packet such as Ethernet destination MAC address.
  - The adjacency table is created from the ARP table.
  - CEF is discussed in more detail in CIS 187 CCNP SWITCH.

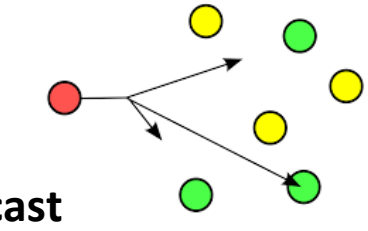
IP Service

- 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)
- ***Anycast***? One-to any









Review

- Distinguishing characteristics of packet-switched communication
  - Data transmitted in ***packets***
  - Statistical multiplexing (best-effort - packets can be dropped or delivered out of order)
- Packet-switching comes in two flavors
  - virtual-circuit - connection oriented
    - route once/flow
    - per-flow forwarding state
  - datagram – connectionless
    - routes computed on event driven-basis
    - per-destination forwarding state
- Consequences of packet-switching
  - higher layer protocols have to deal with losses or with duplicate packets
  - with datagram model, packets may be delivered out of sequence

Review

- 4 classes of services
  - Unicast
  - Broadcast
  - Multicast
  - Anycast

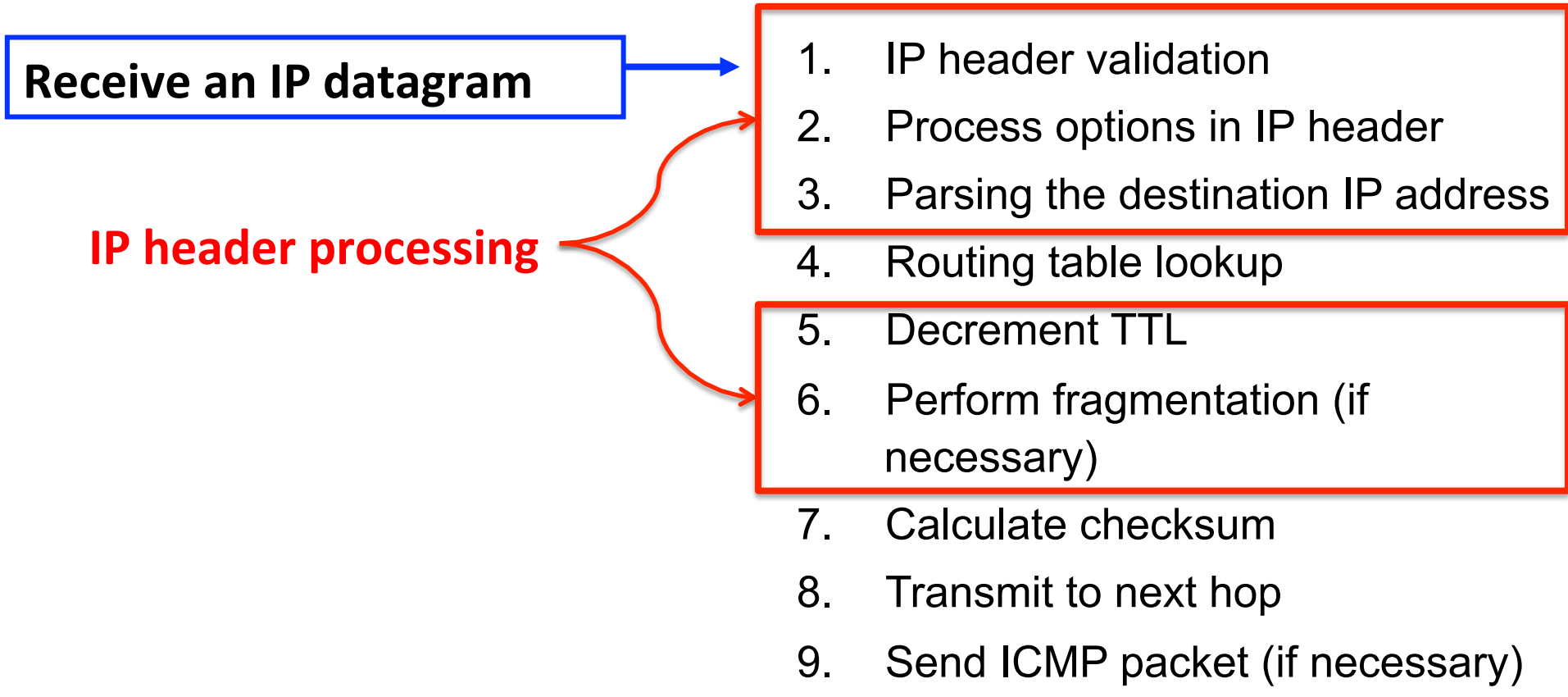
Review

- IP goals
  - interconnect diverse network technologies, making minimal assumptions of the underlying networks
  - implement the minimal set of functionality needed to construct an internet… in this sense IP is the waist of the network protocol stack hourglass
- IP implements a datagram, packet-switched model of communications.
- Packet-switch communication involves transmission of digital data
  - in packets
  - no resource reservation... use statistical multiplexing to share a channel
    - best-effort - packets can be dropped or delivered out of order

Review

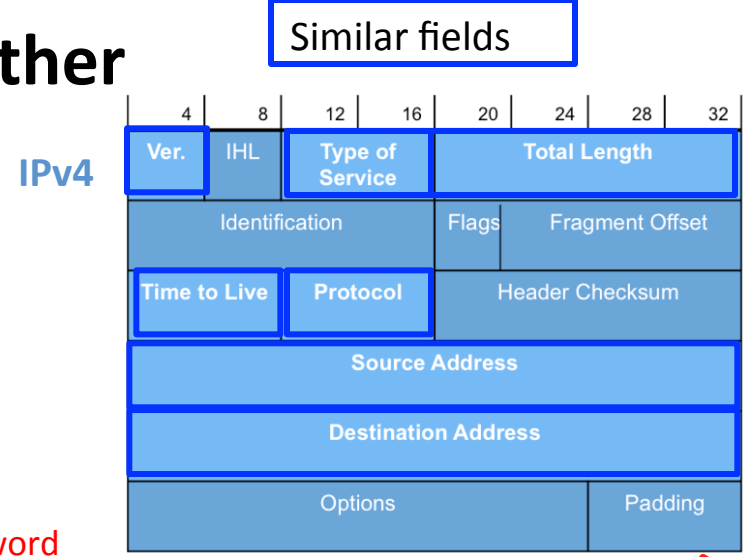
- Consequences of packet-switching
  - higher layer protocols have to deal with losses or with duplicate packets
  - with datagram model, packets may be delivered out of sequence
- End-to-end Principal
- Packet-switching comes in two flavors
  - virtual-circuit - connection oriented
    - route once/flow
    - per-flow forwarding state
  - datagram – connectionless
    - routes computed on event driven-basis
    - per-destination forwarding state
- 4 classes of services: Unicast, Broadcast, Multicast, Anycast

IP Datagram Processing



Understanding IPv4 and IPv6 together

- Several differences between IPv4 and IPv6 headers.



- Simpler IPv6 header.
- Fixed 40 byte IPv6 header.
- Lets look at the differences…

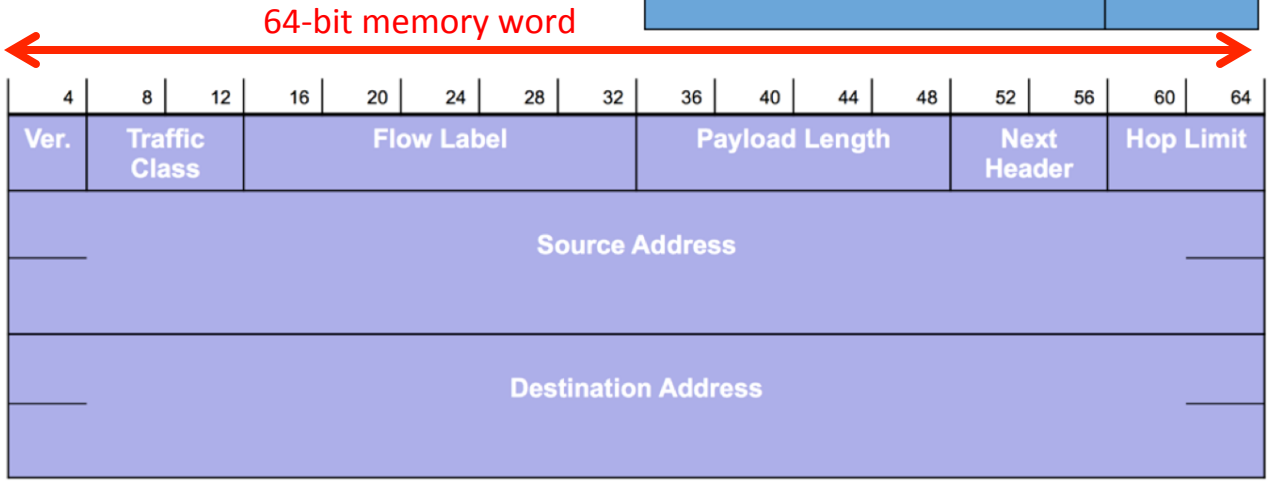
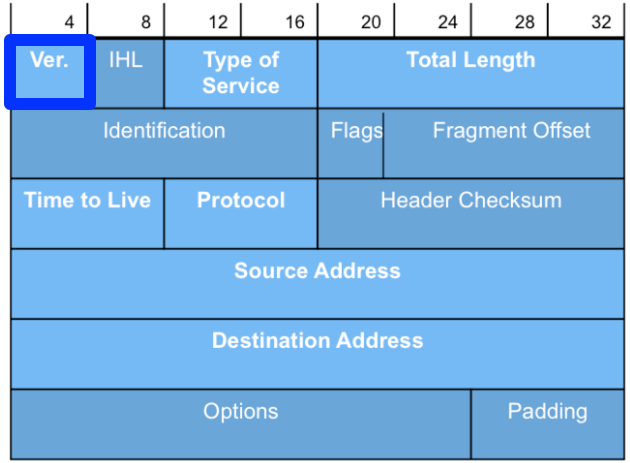


Figure 3-1 – IPv4 Header and Figure 3-2 – IPv6 Header

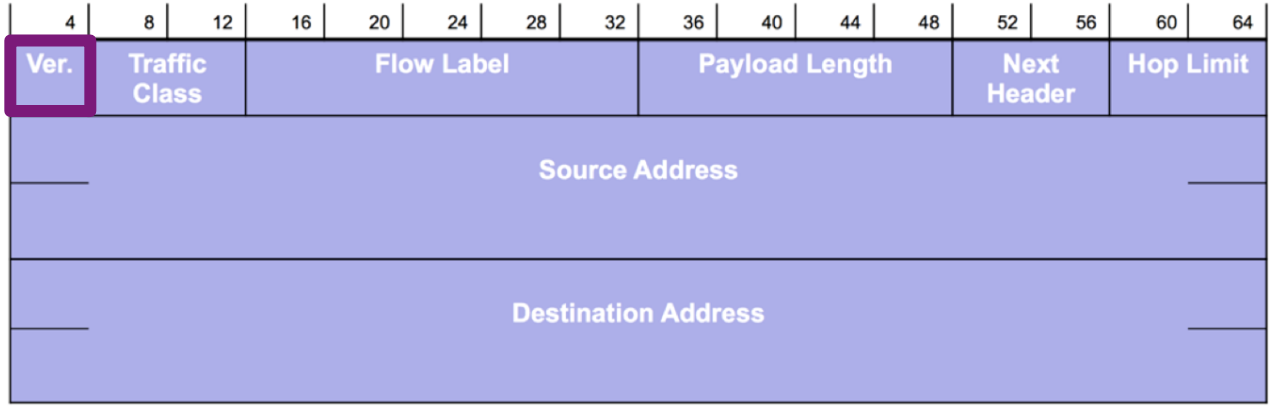
Version

- **IPv4 Version** contains 4.
- **IPv6 Version** contains 6.
- Version 5?
- Internet Stream Protocol (ST2)

IPv4

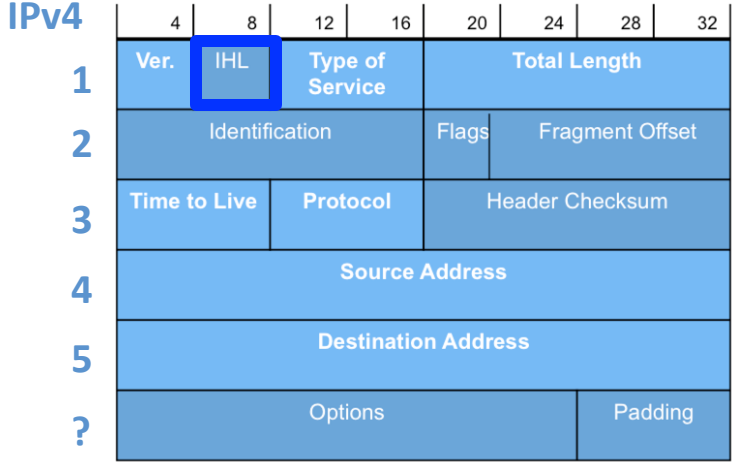


IPv6



IPv4 Internet Header Length

- **IPv4 Internet Header Length (IHL)**
  - Length of IPv4 header in 32-bit words including any Options or Padding.
- **IPv6**
  - IHL for IPv6 is not needed.
  - IPv6 header is fixed at 40 bytes.



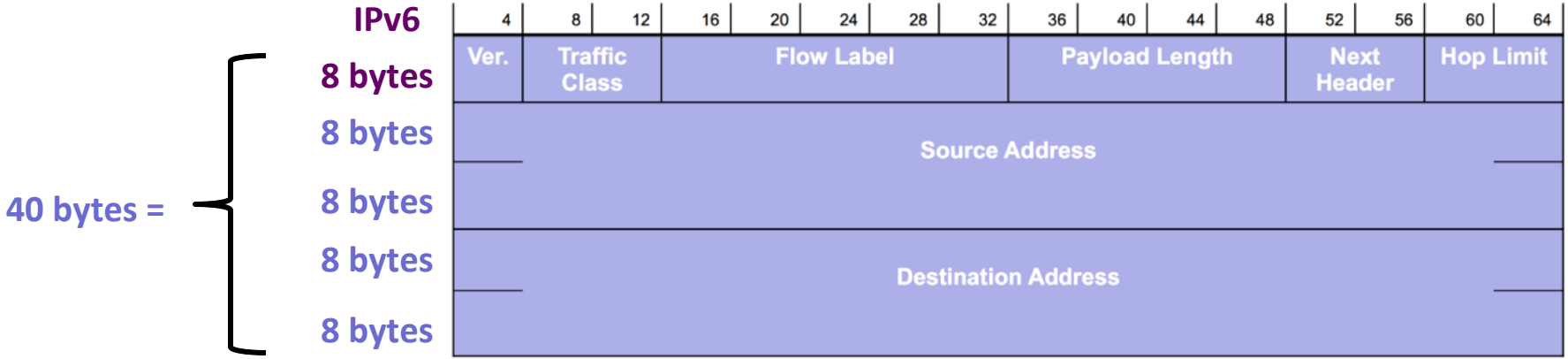


Figure 3-3 – IPv4 Internet Header Length (IHL)

IPv6 Traffic Class

- **IPv4 Type of Service**
- **IPv6 Traffic Class**
  - Not mandated by any IPv6 RFCs.
  - Same functionality as IPv4.
  - Uses same Differentiated Services technique (RFC 2474) as IPv4.

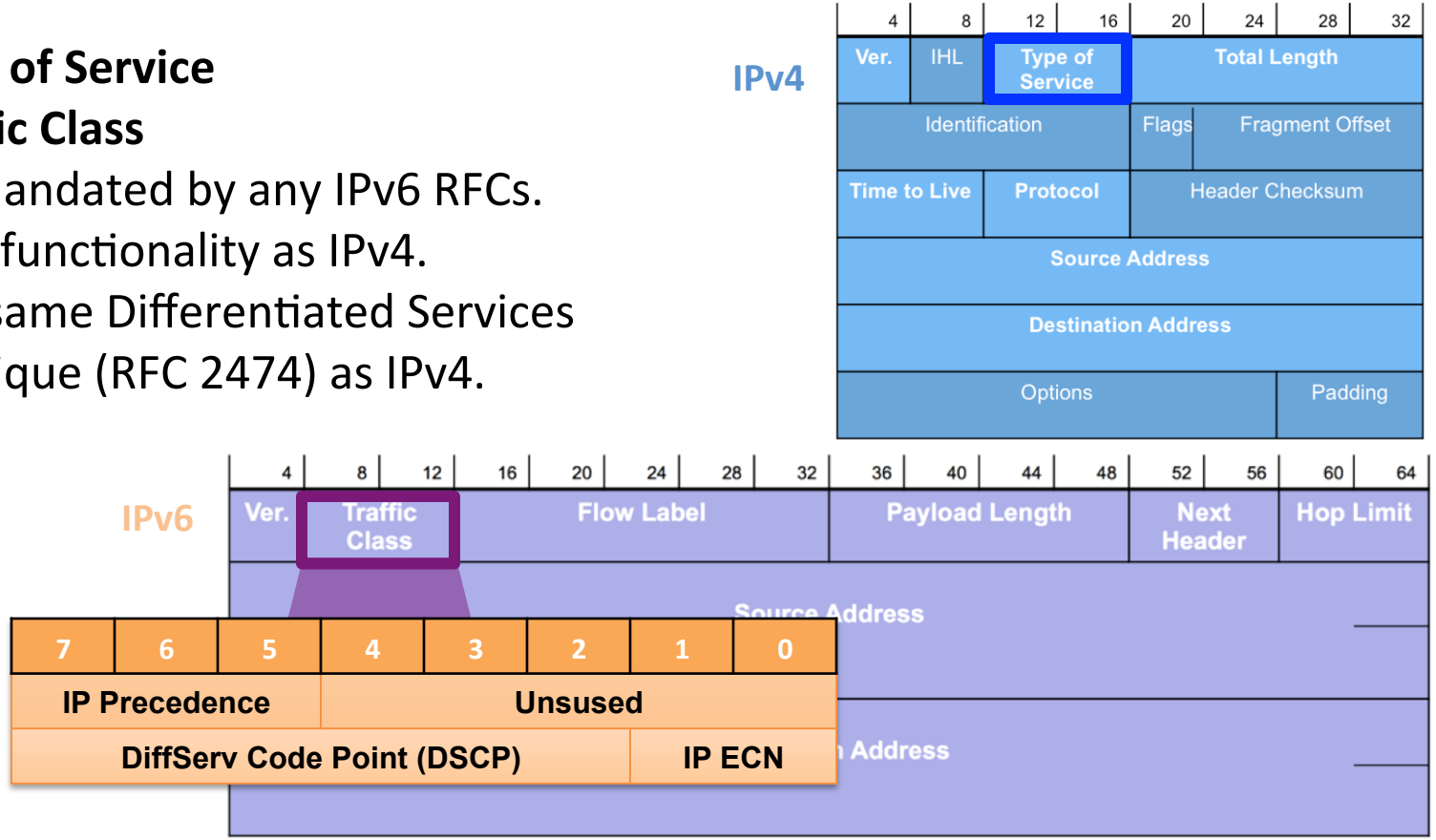


Figure 3-4 – IPv4 IPv4 Type of Service (ToS) Field and IPv6 Traffic Class Field

IPv6 Flow Label

- New field in IPv6 – not part of IPv4.
- Flow label is used to identify the packets in a common st
- Traffic from source to destination share a common flow l
- RFC 6437 IPv6 Flow Label Specification
- Flow label 0 means traffic is not associated with any flow
- Can request spcial handling by IPv6 routers for “real-time

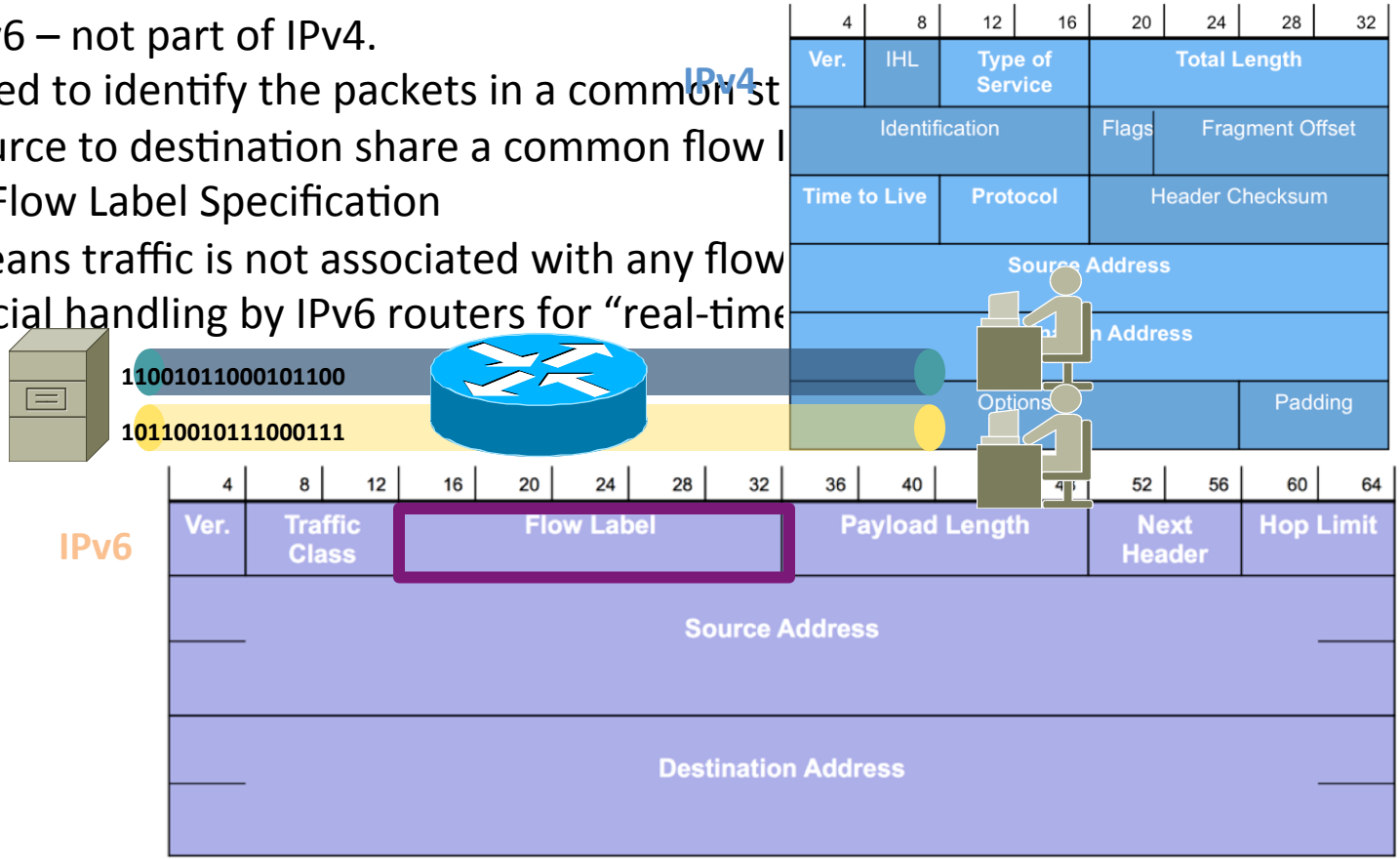
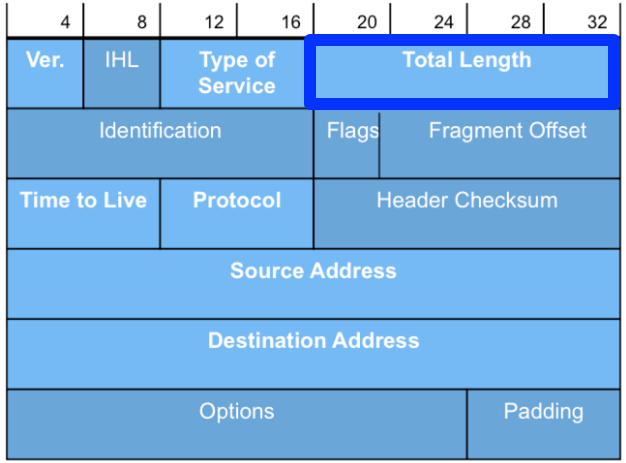


Figure 3-5 – IPv6 Flow Label Field

IPv6 Payload Length

- **IPv4 Total Length** – Number of bytes of the IPv4 header (options) + data.
- Length of data = total length - IHL
- **IPv6 Payload Length** – Number of bytes of the payload.
  - Does not include the main IPv6 header.
  - Includes extension headers + data



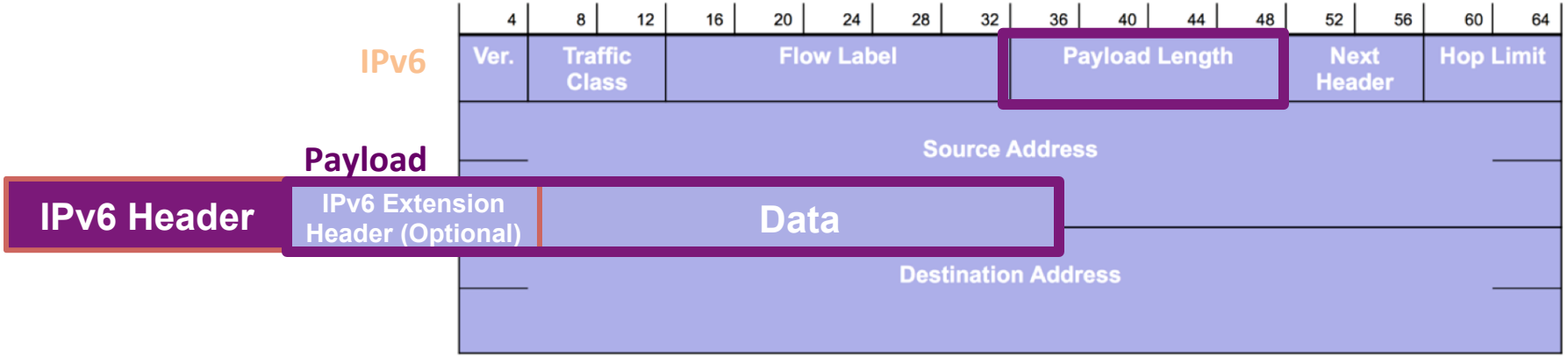
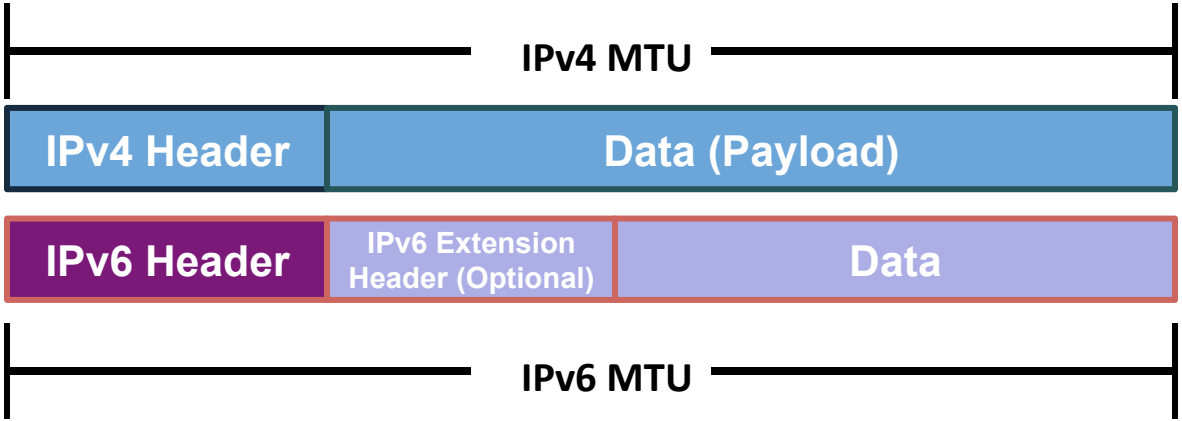


Figure 3-6 –IPv4 Total Field and Figure 3-7 – IPv6 Payload Length Field

IPv4 and IPv6 MTUs



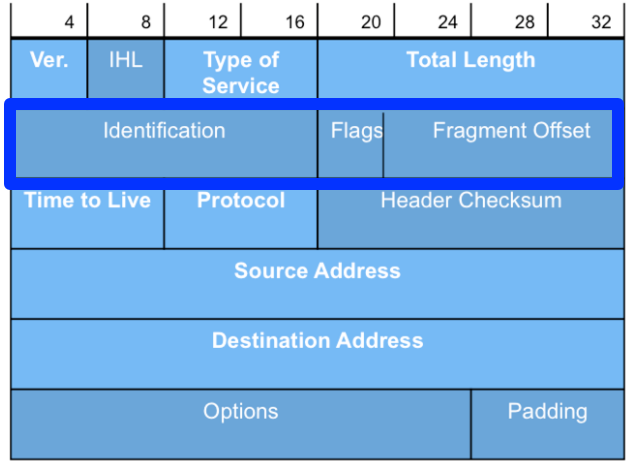
- **IPv4 Total Length** – Number of bytes of the IPv4 header (options) + data.
- **IPv6 Payload Length** – Number of bytes of the payload.
  - Does not include the main IPv6 header.
  - Includes extension headers + data

Figure 3-8 – IPv4 and IPv6 MTUs

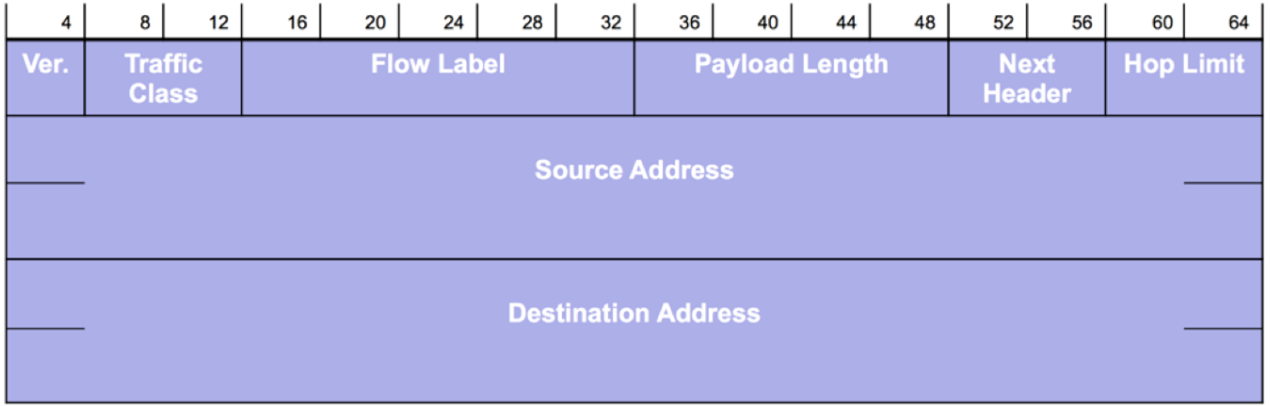
IPv4 Fragmentation

- IPv4 fields used for fragmentation and reassembly.
- Intermediate devices such as IPv6 routers do not perform fragmentation.
- Any fragmentation needed will be handled by the source using an extension header.
- Details in extra slides

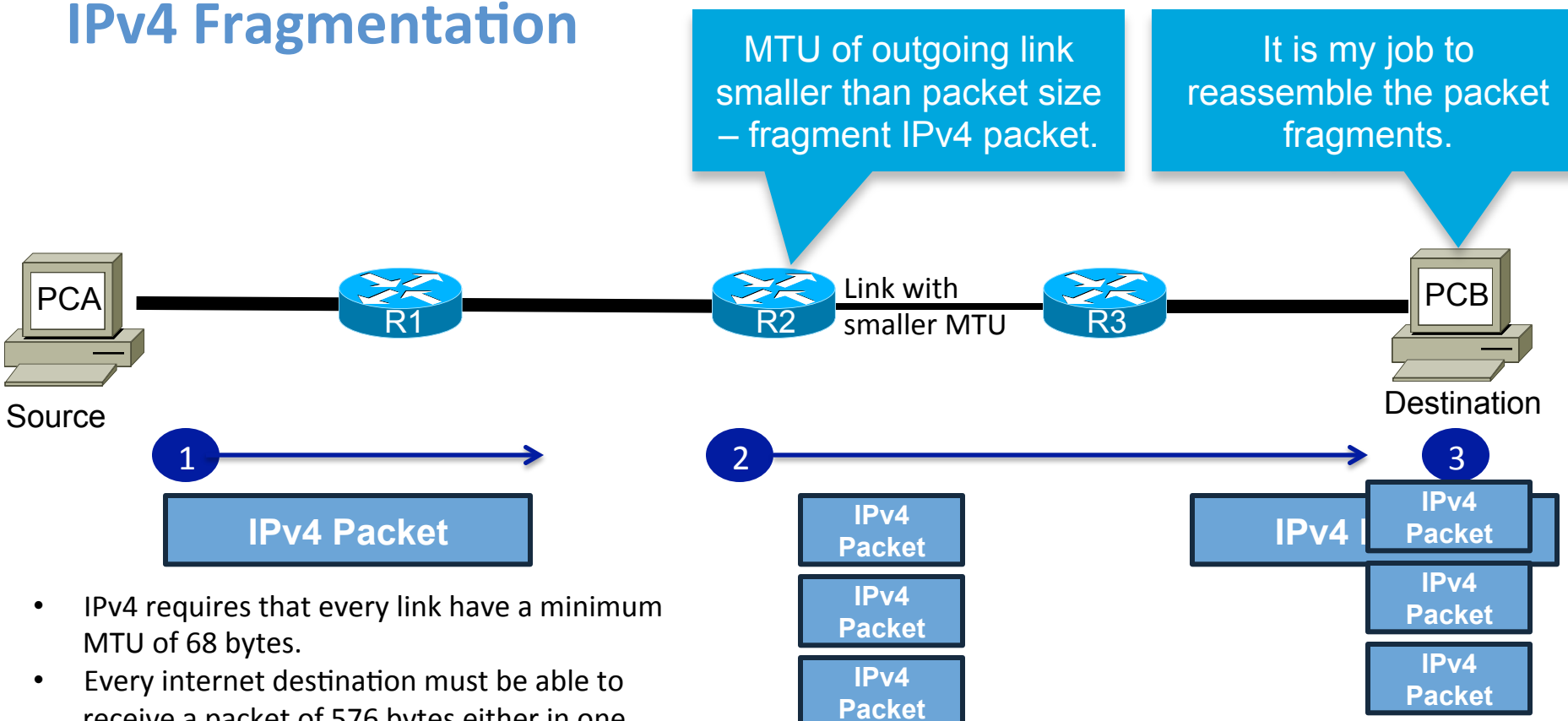
IPv4



IPv6



IPv4 Fragmentation

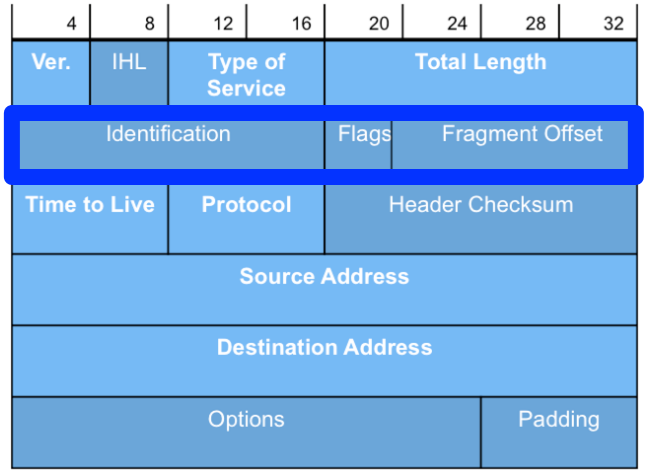


- IPv4 requires that every link have a minimum MTU of 68 bytes.
- Every internet destination must be able to receive a packet of 576 bytes either in one piece or in fragments to be reassembled.

Figure 3-9

IPv6 No Fragmentation

IPv4



IPv6

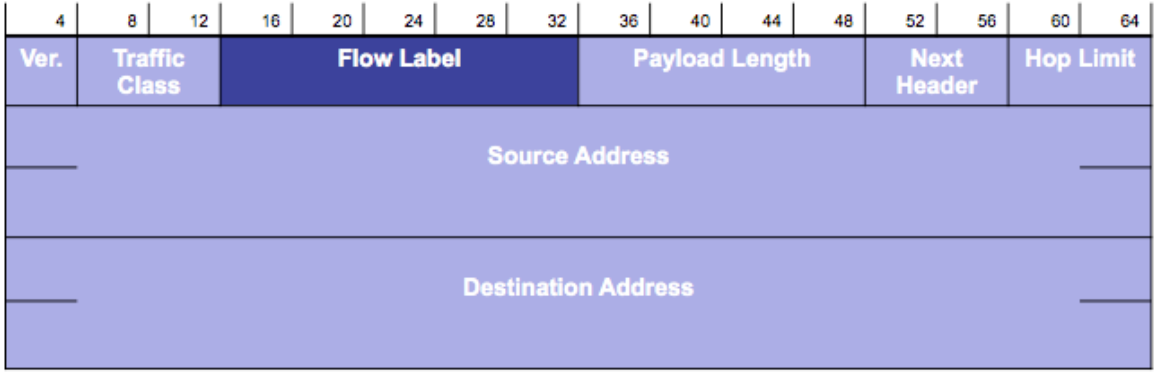


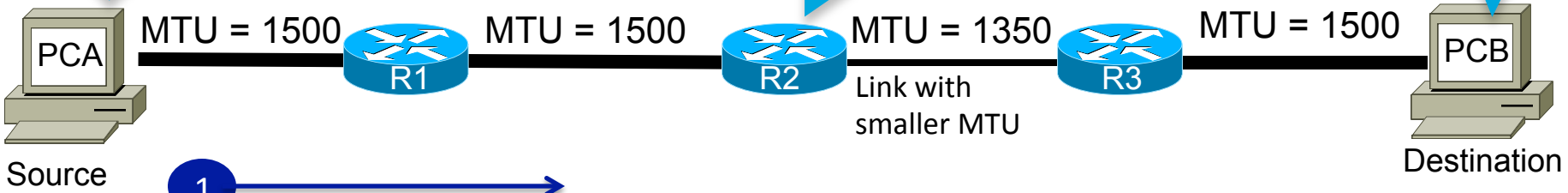
Figure 3-10

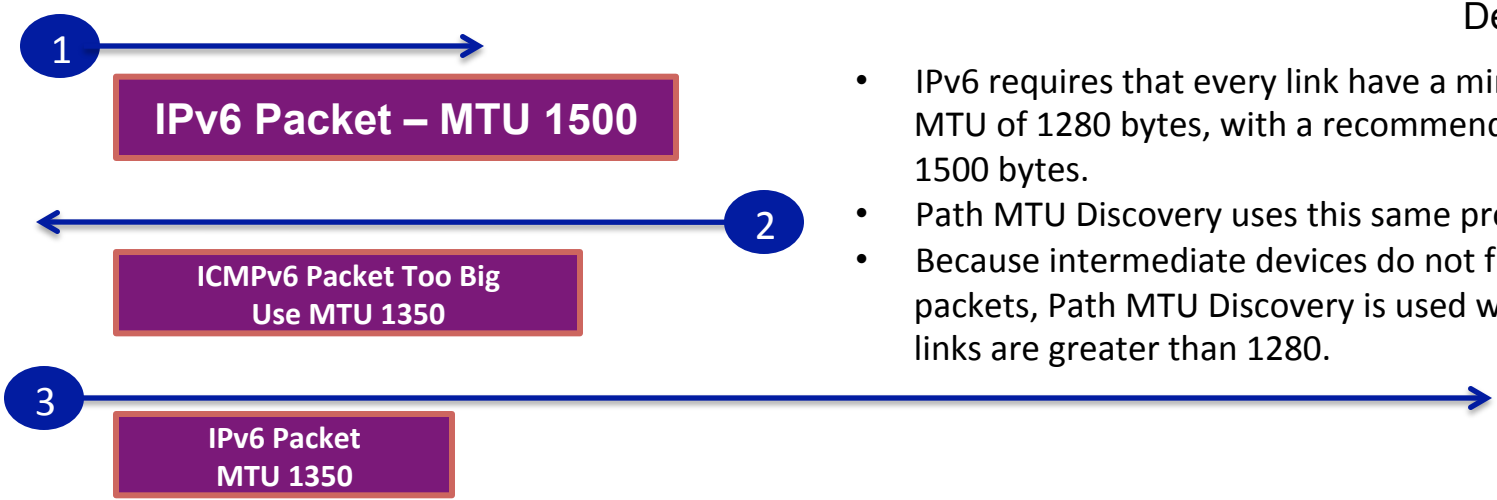
IPv6 No Fragmentation

I will use MTU of the interface.

MTU of outgoing link smaller than packet size. Drop packet. Send ICMPv6 Packet Too Big message, use MTU 1350.

Packet received. No reassembly required.

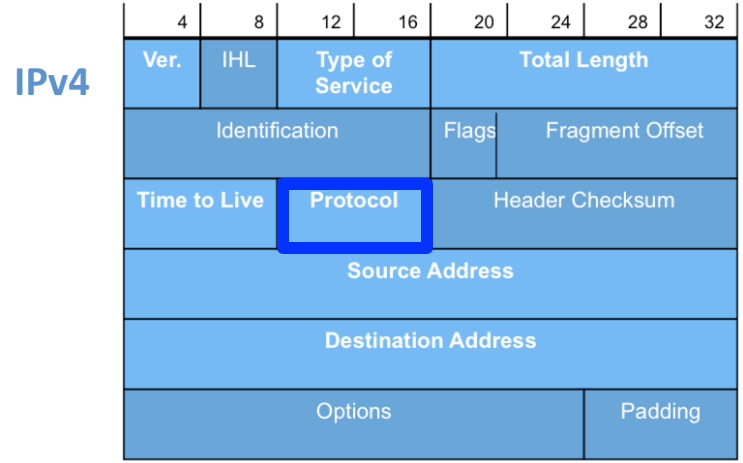




- IPv6 requires that every link have a minimum MTU of 1280 bytes, with a recommended MTU of 1500 bytes.
- Path MTU Discovery uses this same process.
- Because intermediate devices do not fragment packets, Path MTU Discovery is used when their links are greater than 1280.

IPv6 Next Header

- **IPv4 Protocol**
- **IPv6 Next Header**
- For both protocols, the field indicates the type of header following the IP header.



- Common values:
- 6 = TCP
- 17 = UDP
- 58 = ICMPv6
- 88 = EIGRP
- 89 = OSPF

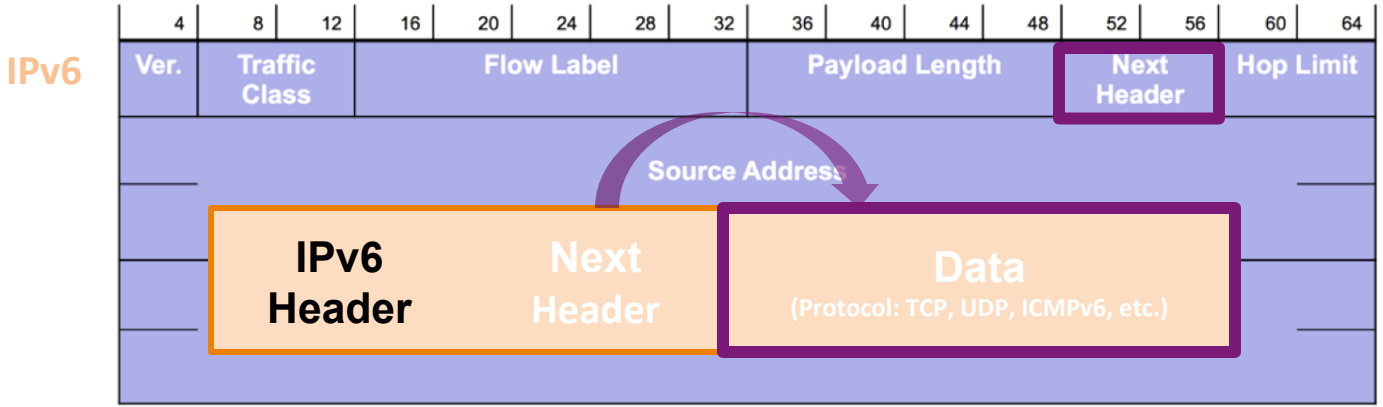


Figure 3-11 – IPv4 Protocol and IPv6 Next Header Fields

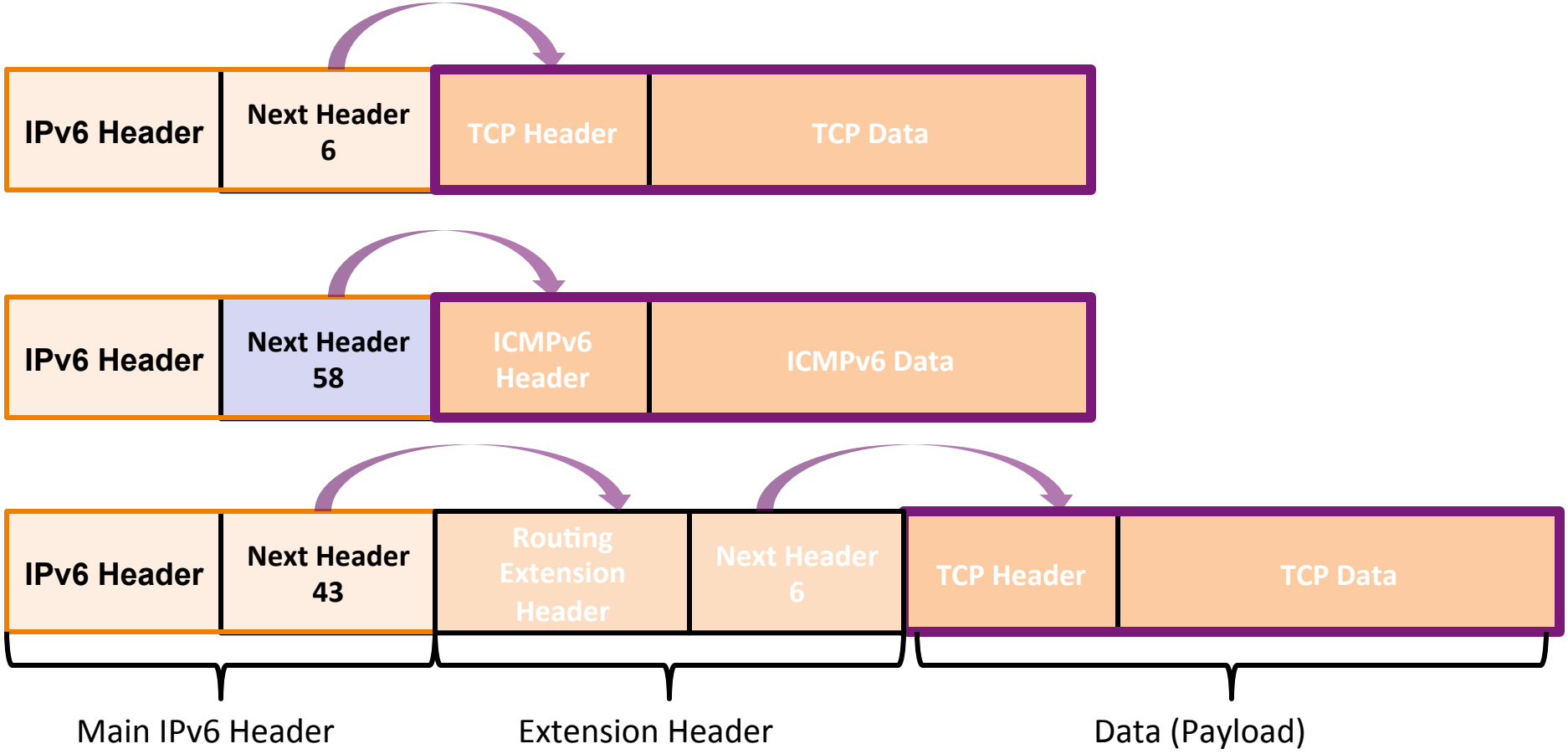


Figure 3-12 –IPv6 Next Header Examples

IPv6 Hop Limit

- **IPv4 TTL (Time to Live)**
- **IPv6 Hop Limit**
- Renamed to more accurately reflect process.
- Set by source, every router in path decrements hop limit by 1.
- When 0, drop packet.
- No TTL in Ethernet… why do you think they didn't include one?

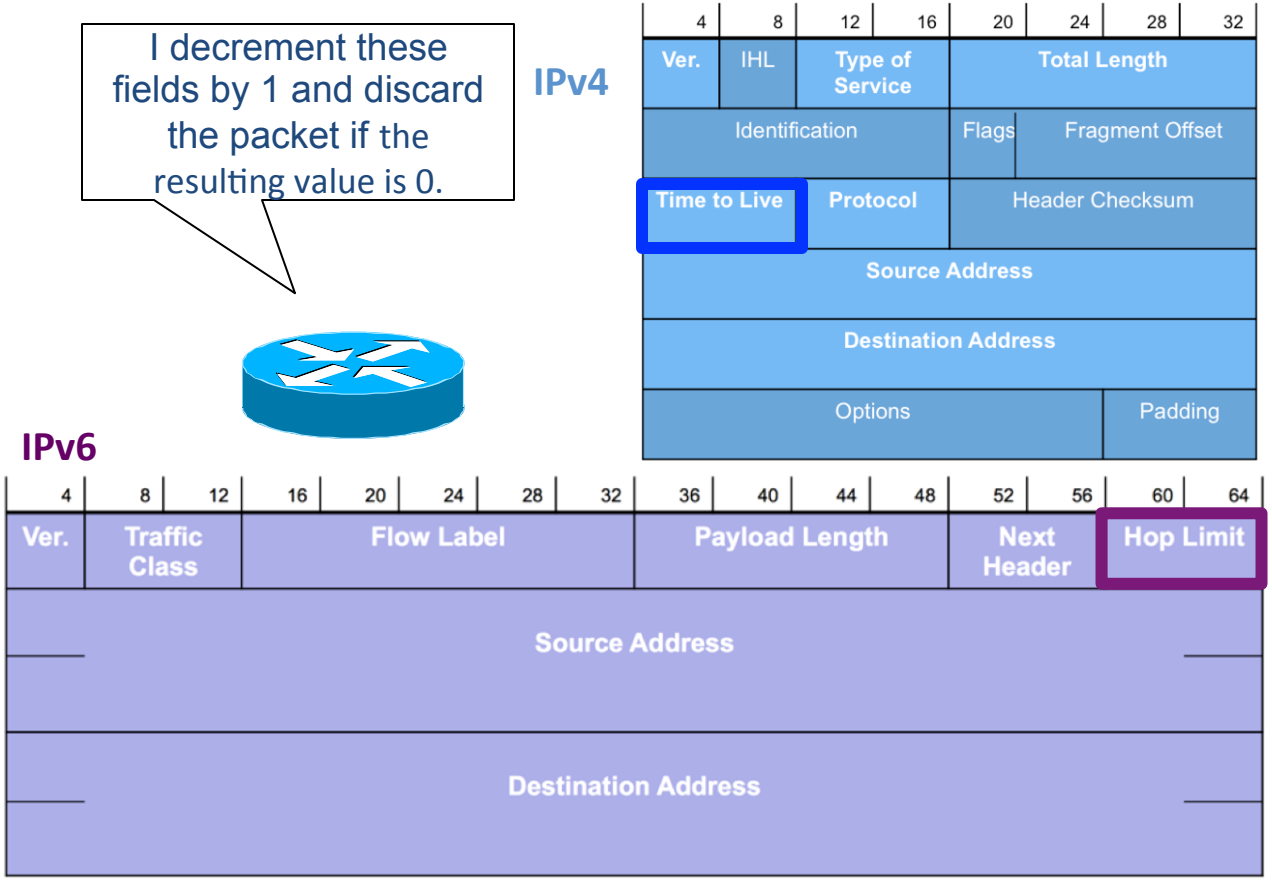
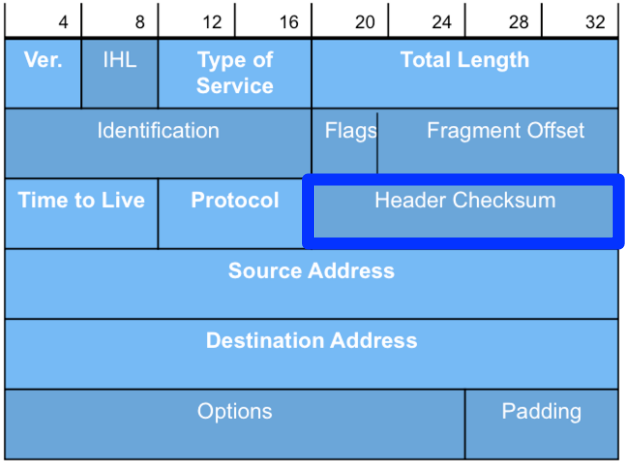


Figure 3-13 – IPv4 TTL and IPv6 Hop Limit Fields

IPv4 Header Checksum

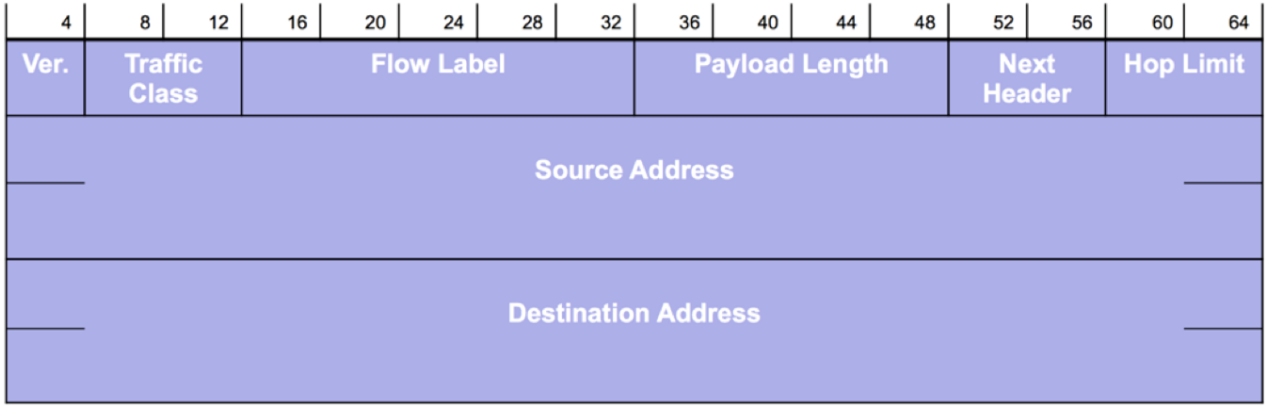
- **IPv4 Header Checksum**
- Not used in IPv6.
- Upper-layer protocols generally have a checksum (UDP and TCP).
- So, in IPv4 the UDP checksum is optional.

IPv4



- Because it's not in IPv6, the UDP checksum is now mandatory.

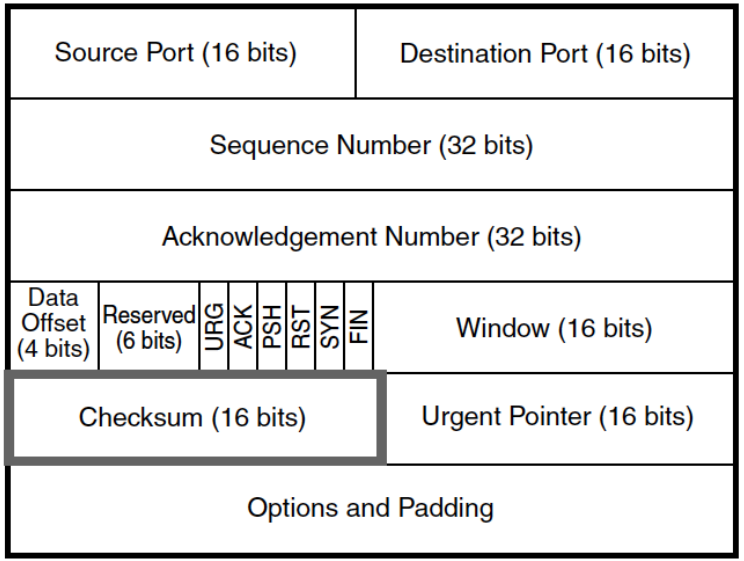
IPv6



IPv4: TCP and UDP Checksums

- **UDP checksum**, which is ***optional in IPv4***, is therefore ***mandatory in IPv6***.
- The designers of IPv6 did not include a Checksum field because Layer 2 data link technologies such as Ethernet perform their own checksum and error control.

TCP



UDP

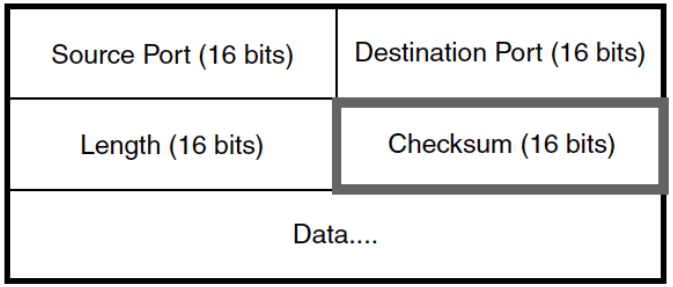
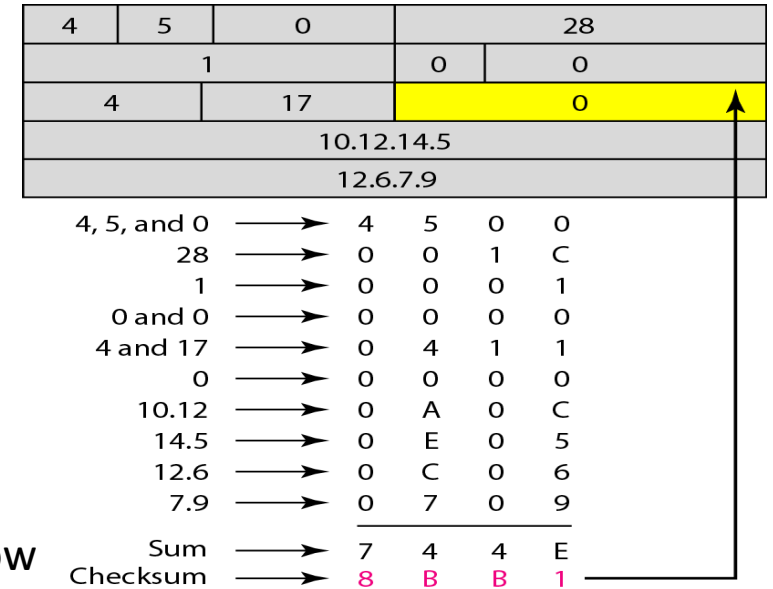


Figure 3-15 *TCP and UDP*

Checksum

- **IPv4 Header checksum** (2 bytes): Simple 16-bit long checksum covers only header.
- Upper layer protocols cover data
- IP is highest hop-by-hop protocol; need to minimize processing

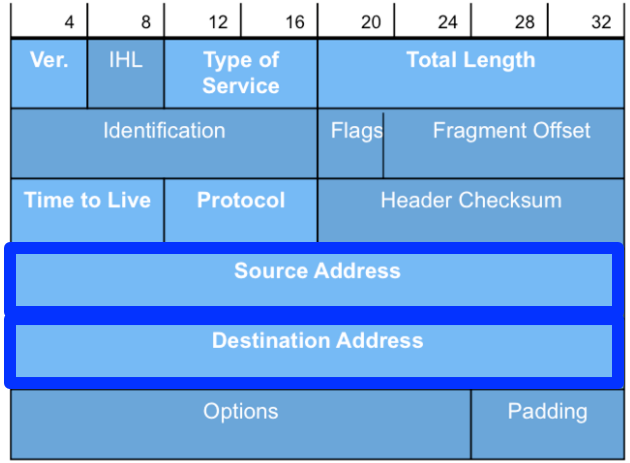
How Checksum is calculated:
https://www.youtube.com/watch?v=dXartoyj2ow



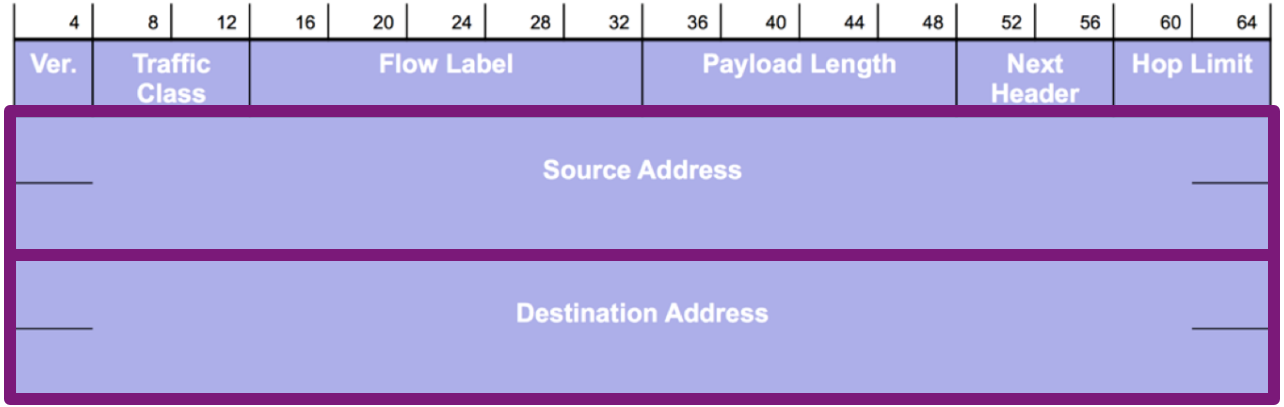
IPv6 Source and Destination Addresses

- **IPv6 Source** and **Destination** addresses have the same basic functionality as IPv4.
- IPv4 – 32-bit addresses.
- IPv6 – 128-bit addresses.
- Some significant changes in IPv6.

IPv4



IPv6



IPv4 Options and Padding

- **IPv4 Options** and **Padding**
- Not used in IPv6.
- Variable length, optional.
- **IPv4 Options** are handled using extension headers in IPv6.
- **Padding** makes sure IPv4 options fall on a 32-bit boundary.
- IPv6 header is fixed at 40 bytes.

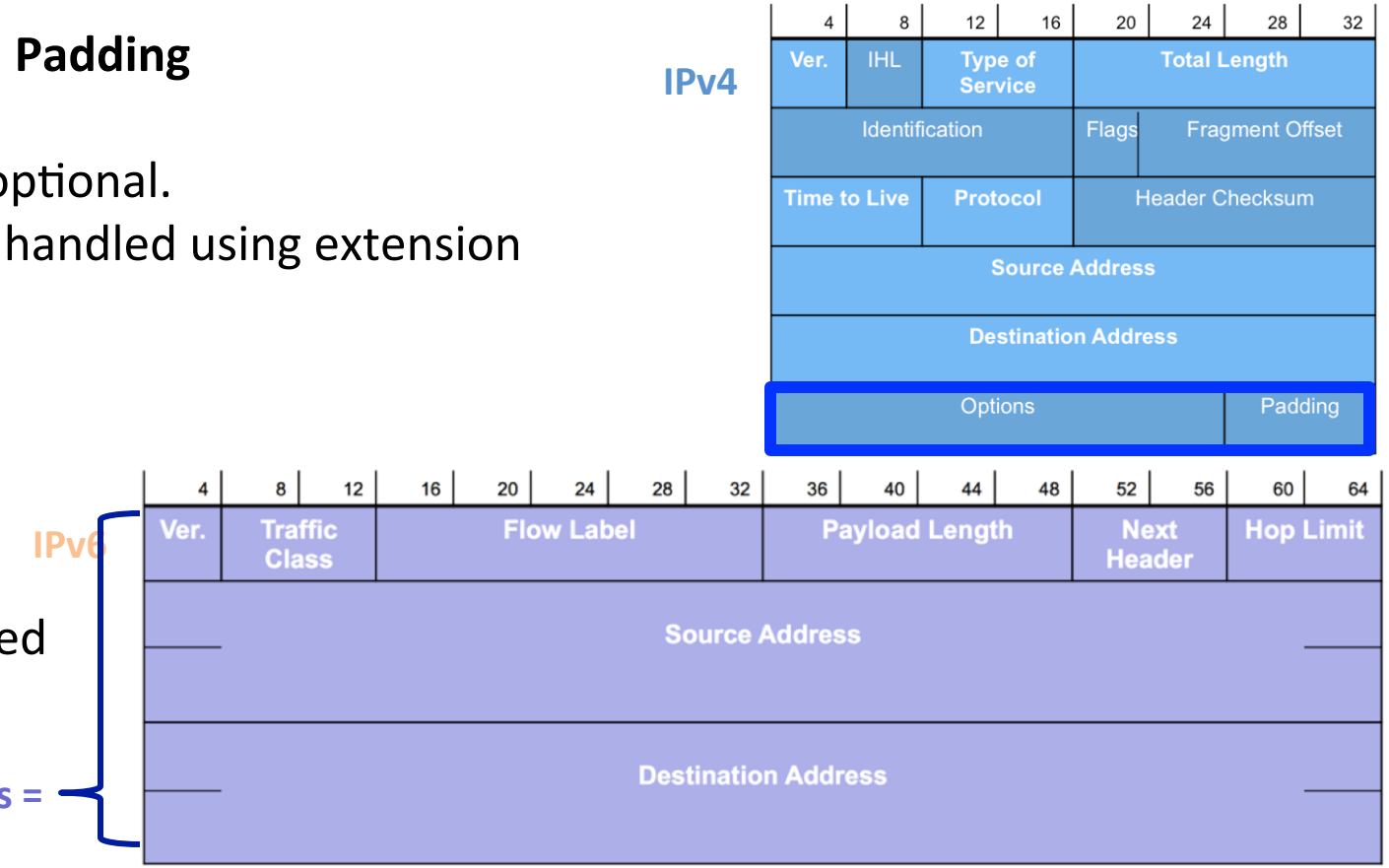
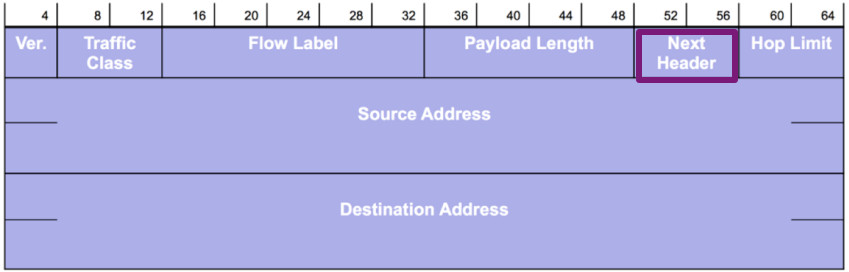
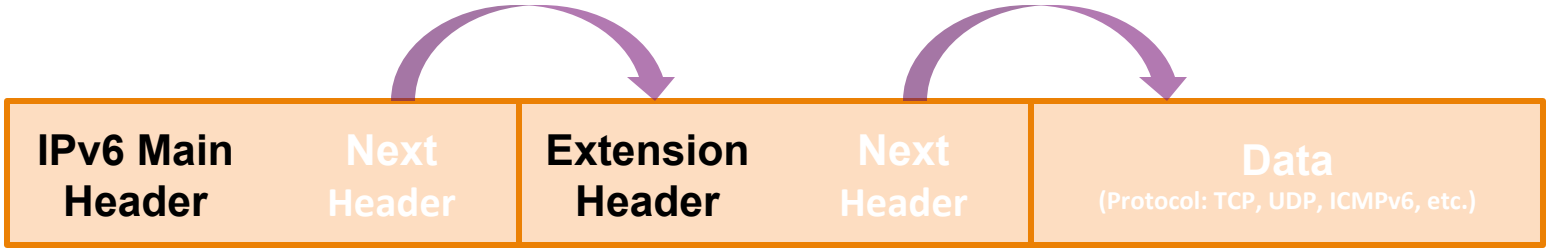


Figure 3-16 – IPv4 Options Field and Padding Field

IPv6 Extension Header



- **Next Header** identifies:
  - The protocol carried in the data portion of the packet.
  - The presence of an extension header.
- **Extension headers** are optional and follow the main IPv6 header.
- Provide flexibility and features to the main IPv6 header for future enhancements without having to redesign the entire protocol.
- Allows the main IPv6 header to have a fixed size for more efficient processing.



Review

- IP headers…
  - Checksum only covers header to minimize hop-by-hop processing
    - Assumes upper layer protocols cover data
  - Checksum must be recalculated at each hop
    - IP header fields change… TTL, Fragmentation information, Header length
  - Fragmentation and reassembly
    - Used to match packets to link MTUs
    - Fragmentation can be done at any hop on the path
    - Reassembly can only be done at the destination

IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

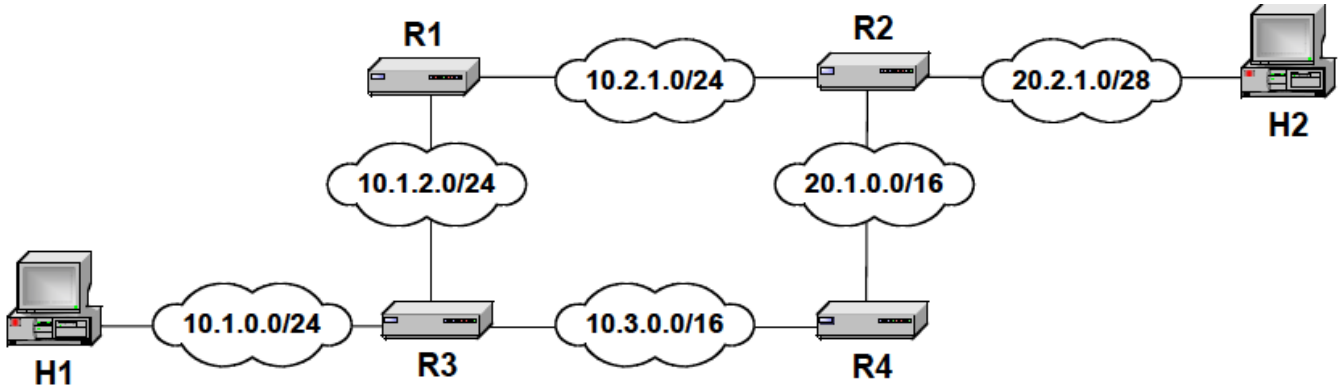
1. IP header validation
2. Process options in IP header
3. Parsing the destination IP address
4. Routing table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop
9. Send ICMP packet (if necessary)

Forwarding vs. Routing (urg)

- There are two distinct processes to delivering IP packets (datagrams):
  - **Routing**:
    - Finding a suitable path for a packet from sender to destination
    - Finding the path
  - **Forwarding**:
    - The action of delivering the packet one hop closer to destination based on the routing information.
- And there is also:
  - **Switching**
    - Refers to moving a packet from one interface to another within a device

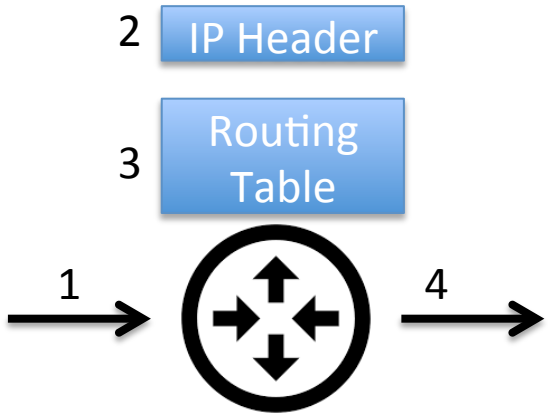
Forwarding

- An internet is a collection of subnets
- IP implements hop-by-hop delivery of packets between hosts in an internet
- Routers
  - Interconnect subnets (broadcast domains)
  - forward packets across an internetwork for subnets (even the Internet)



Routers

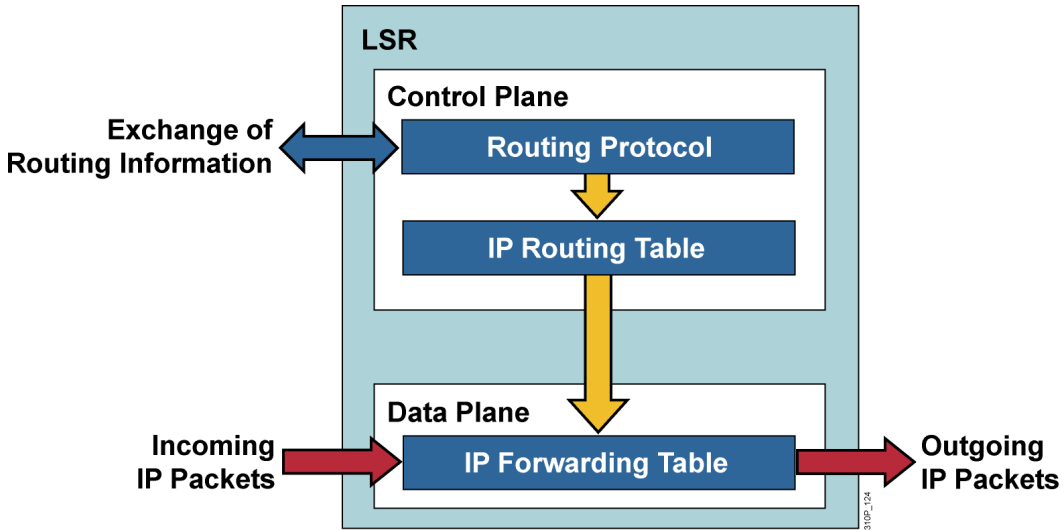
- Routers implement
  - Forwarding process
  - Routing process
- Routers connect multiple subnets
  - Have interfaces on multiple subnets
  - Forward packets between subnets
    1. Receive on one interface
    2. Process IP header
    3. Determine next hop
    4. Send out next hop interface

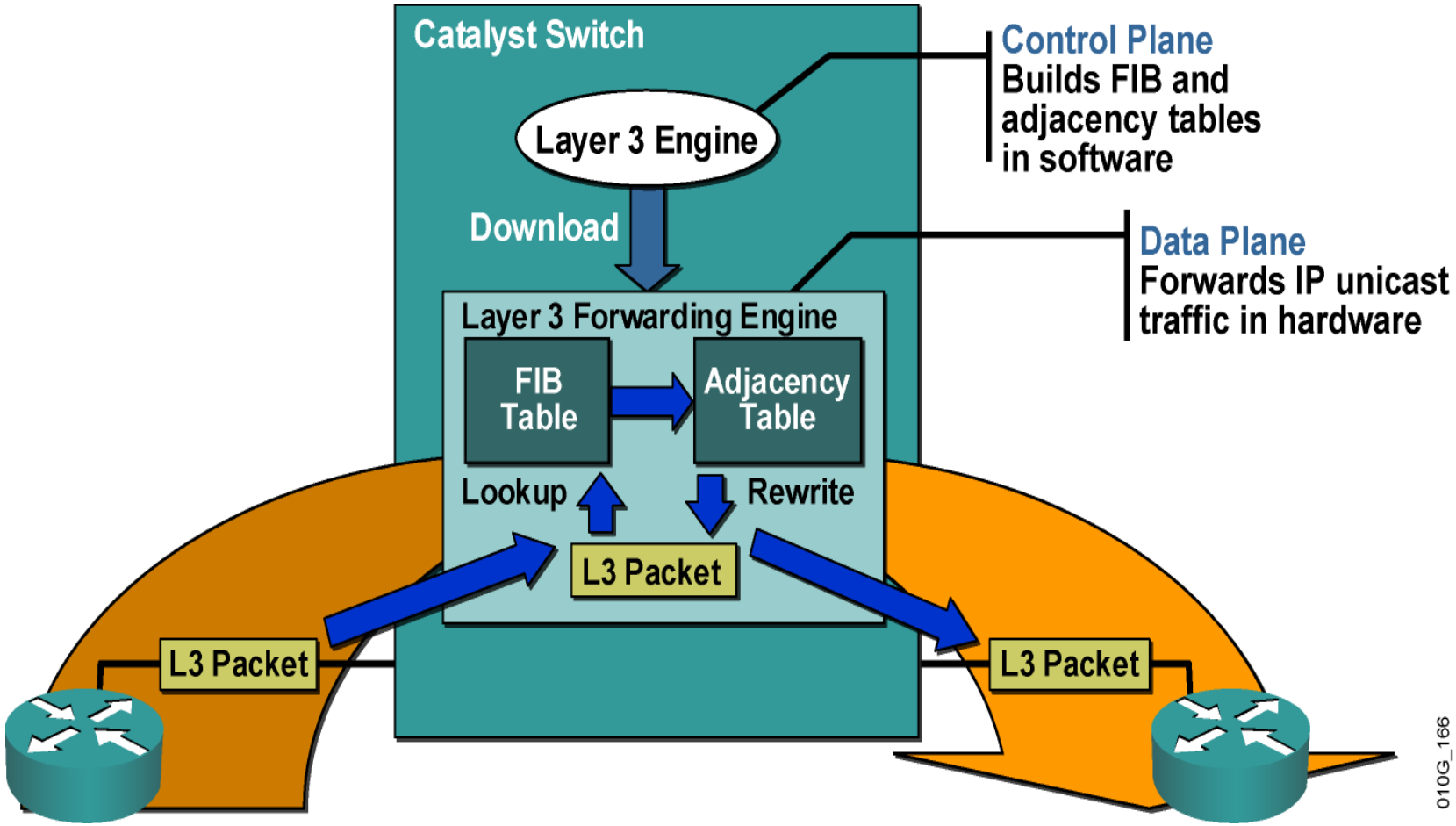


Routing Table

- **Forwarding table or routing table** is the interface between the routing and forwarding processes
  - Simple mechanism
  - Implements (potentially) complex policies
- Maps destination address to next hop towards destination

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




Catalyst Switch
Layer 3 Engine
Download
Layer 3 Forwarding Engine
FIB Table
Adjacency Table
Lookup
Rewrite
L3 Packet
L3 Packet
L3 Packet
Control Plane
Builds FIB and adjacency tables in software
Data Plane
Forwards IP unicast traffic in hardware
010G_166

Routing and Forwarding

Routing functions include:

- **Route calculation**: Determine best paths
- **Maintenance of the routing table**: Topology changes
- **Execution of routing protocols** (Route calculation and maintenance)

- On commercial routers handled by a single general purpose processor, called route processor
- On high-end commercial routers, IP forwarding is distributed
- Most work is done on the interface cards

Forwarding Tables

- Each router and each host maintains a **forwarding table (or routing table)** which tells the router how to process an outgoing packet
- Main columns:
  - **Destination address**: where is the IP datagram going to?
  - **Next hop or interface**: how to forward the IP datagram?
- Routing tables are set so that a packet gets closer to the its destination every hop

Routing table of a host or router

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

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

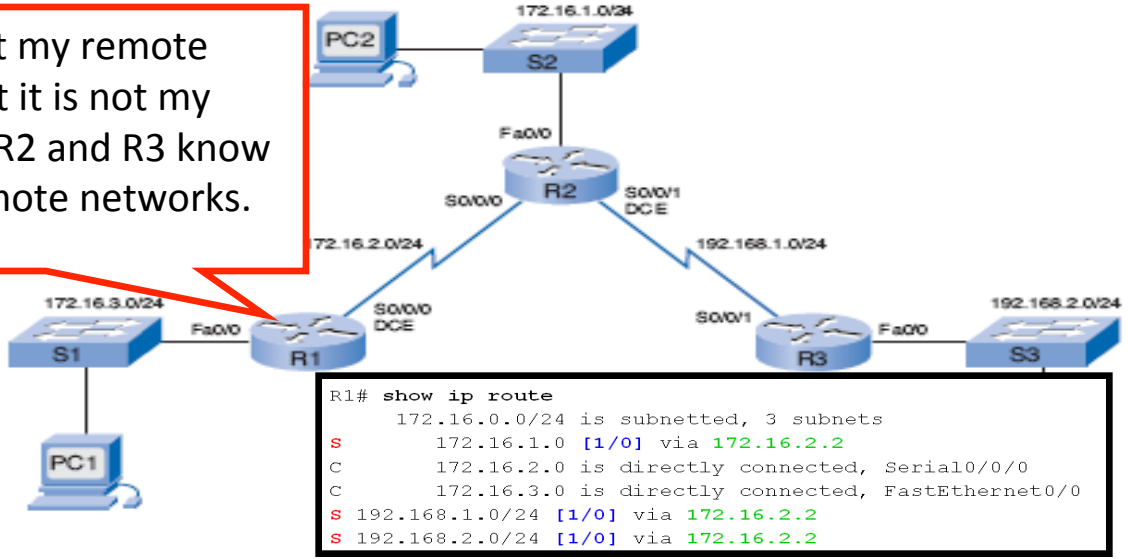
Forwarding (Routing) Table Lookup

- When a router or host needs to transmit an IP packet, it performs a ***forwarding table lookup***
- **Forwarding 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 exit interface

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

Alex Zinin's Routing Table Principles

I know about my remote networks but it is not my responsibility if R2 and R3 know about their remote networks.

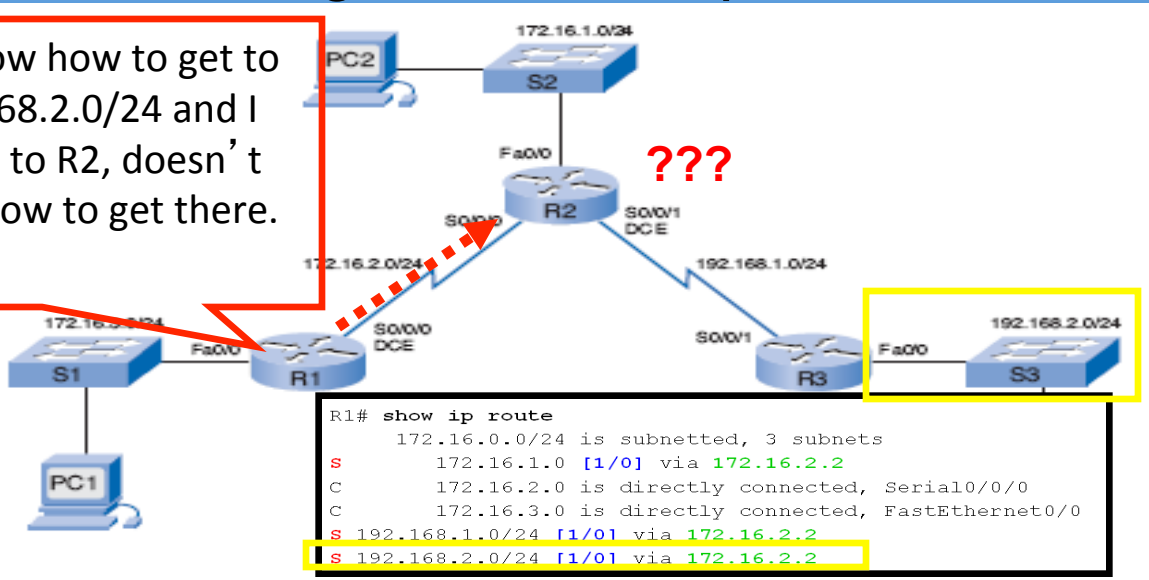



```
R1# show ip route
     172.16.0.0/24 is subnetted, 3 subnets
S       172.16.1.0 [1/0] via 172.16.2.2
C       172.16.2.0 is directly connected, Serial0/0/0
C       172.16.3.0 is directly connected, FastEthernet0/0
S 192.168.1.0/24 [1/0] via 172.16.2.2
S 192.168.2.0/24 [1/0] via 172.16.2.2
```

- **Principle 1: Every router makes its decision alone, based on the information it has in its own routing table.**
- R1 makes forwarding decisions based solely on the information in the routing table.
- R1 does not consult the routing tables in any other routers.
- Making each router aware of remote networks is the responsibility of the network administrator.

Alex Zinin's Routing Table Principles

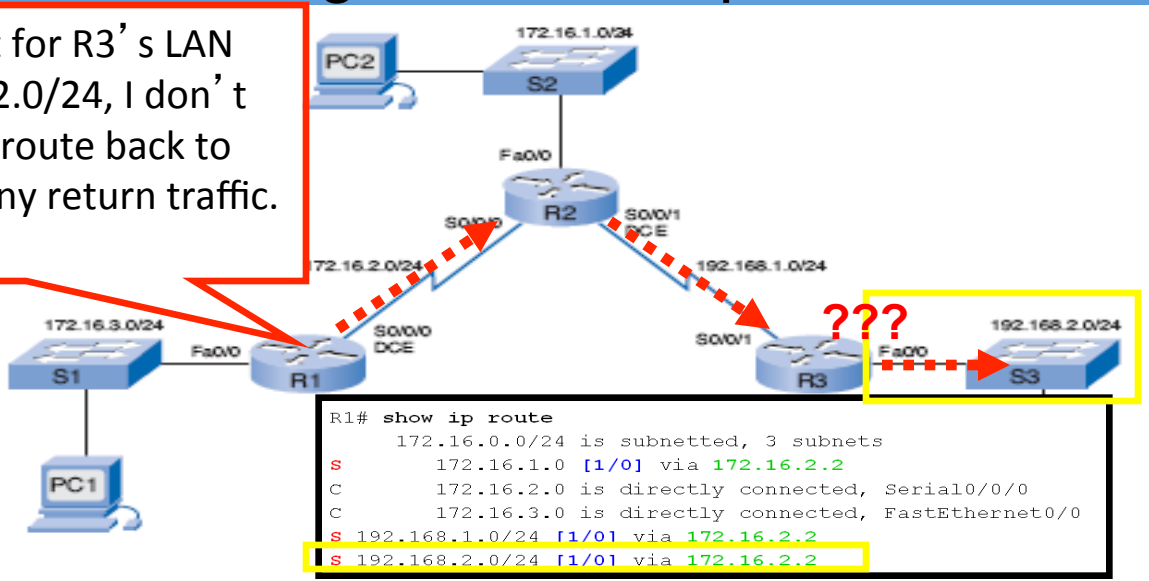
Just because I know how to get to R3's LAN, 192.168.2.0/24 and I send that packet to R2, doesn't mean R2 knows how to get there.



- **Principle 2: The fact that one router has certain information in its routing table does not mean that other routers have the same information.**

Alex Zinin's Routing Table Principles

And if the packet for R3's LAN reaches 192.168.2.0/24, I don't know if R3 has a route back to 172.16.3.0/24 for any return traffic.



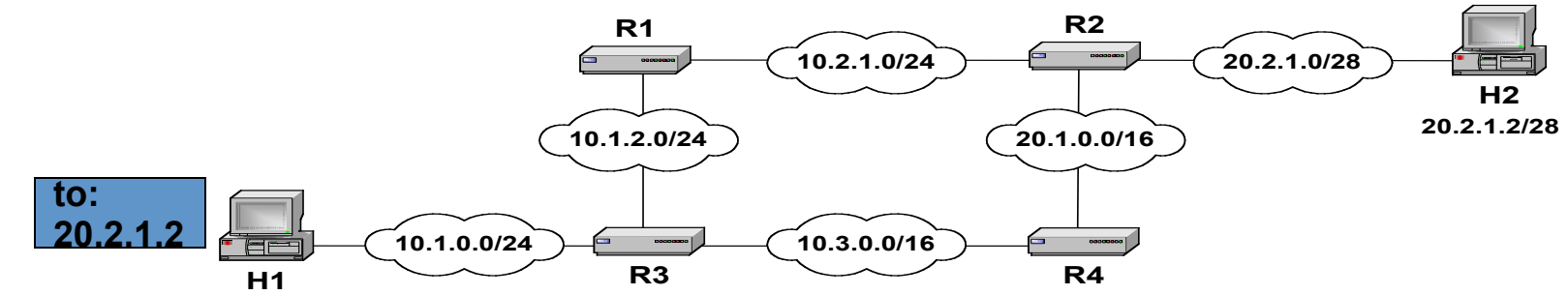
- **Principle 3: Routing information about a path from one network to another does not provide routing information about the reverse, or return, path.**

Delivery with Forwarding 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

Forwarding Table Match

- Forwarding table entries composed of: **<IP Address>/<network mask>**
  - E.g. 128.114.48.128/26
- Mask defines the network part of an address
  - **128.114.48.128/26 = 10000000 01110010 00110000 10xxxxxx**
  - ”x” = don’t care (host part of address)
- Forwarding table match occurs when…
  - Both the routing entry and IP addresses have the same network part…
  - …given the route’s network mask
- Example: Is 128.114.48.0/17 a matching route for 128.114.122.5?
  - 128.114.48.0/17 = **10000000 01110010 0**xxxxxxx xxxxxxxx
  - 128.114.122.5 = **10000000 01110010 0**~~1111010 00000101~~
  - **Yes.**

128.114.128.5

Routing Table
128.114.0.0/16
128.114.48.0/17

- What order used in considering forwarding table entries?

- **128.114.128.5 matches 128.114.0.0/16**
  - 128.114.0.0/16 = 10000000 01110010 xxxxxxxx xxxxxxxx
  - 128.114.128.5 = 10000000 01110010 ~~10000000 00000101~~
- **128.114.128.5 *also* matches 128.114.0.0/17**
  - 128.114.128.0/17 = 10000000 01110010 0xxxxxxx xxxxxxxx
  - 128.114.128.5 = 10000000 01110010 ~~00000000 00000101~~

- Answer is ***longest prefix match***
- ***Can have both a less specific and more specific route in routing table***
- ***Allows for route aggregation (summarization)***

Types of Forwarding Table Entries

- **Network route**
  - Destination address with 0 < prefix length < 32 (e.g., 10.0.2.0/24)
  - Most entries are network routes
- **Host route**
  - Destination address with prefix length = 32 (e.g., 10.0.1.2/32)
  - Used to specify a separate route for certain hosts
- **Default route**
  - Destination address with prefix length = 0 (i.e. 0.0.0.0/0)
  - Matches all destinations
  - Commonly use to connect a company's edge router to the ISP network.

Tools: ipcalc

- IMO, subnets are best thought of as address ranges
- `ipcalc` calculates these ranges for you…

```
% ipcalc 128.114.48.0/17
Address:   128.114.48.0          10000000.01110010.0 0110000.00000000
Netmask:   255.255.128.0 = 17    11111111.11111111.1 0000000.00000000
Wildcard:  0.0.127.255           00000000.00000000.0 1111111.11111111
=>
Network:   128.114.0.0/17        10000000.01110010.0 0000000.00000000
HostMin:   128.114.0.1           10000000.01110010.0 0000000.00000001
HostMax:   128.114.127.254       10000000.01110010.0 1111111.11111110
Broadcast: 128.114.127.255       10000000.01110010.0 1111111.11111111
Hosts/Net: 32766                  Class B
```

- …address range of 128.114.0.1 – 127.254
- http://jodies.de/ipcalc (available as command line tool)

Tools: Rick's Python

```
Part 1: Major Network
----------------------

Enter an IPv4 address: 200.10.10.17
Enter the current subnet mask: 255.255.255.0

        IPv4 address in binary:          11001000 00001010 00001010 00010001
        Subnet mask in binary:           11111111 11111111 11111111 00000000

This is a HOST address. There are not all 0s nor all 1s in the host portion.
Press enter to continue...

IPv4 address:    11001000 00001010 00001010 00010001
Subnet mask:     11111111 11111111 11111111 00000000
                 ----------------------------------    Copy the network bits:

Network:         11001000 00001010 00001010 00000000    Network + Host (all 0s)
First host:      11001000 00001010 00001010 00000001    Network + Host (all 0s + 1)
Last host:       11001000 00001010 00001010 11111110    Network + Host (all 1s + 0)
Broadcast:       11001000 00001010 00001010 11111111    Network + Host (all 1s)

Number of network bits:  24 bits or /24
Number of host bits:  8 bits
Number of hosts :  2^8 - 2 = 254 hosts

Addresses in dotted decimal notation:
Network address is: 200.10.10.0
First host is: 200.10.10.1
Last host is: 200.10.10.254
Broadcast address is: 200.10.10.255


Generate a random IPv4 host address from subnet? (y/n)
```

Route Aggregation

- Longest prefix match algorithm allows aggregation of prefixes with identical next hop address to a single entry
- Benefits
  - Reduces size of routing tables
  - More efficient routing table lookups

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

Calculating a Route Summary

Step 1: List networks in binary format.

172.20.0.0	10101100 . 00010100 . 00000000 . 00000000
172.21.0.0	10101100 . 00010101 . 00000000 . 00000000
172.22.0.0	10101100 . 00010110 . 00000000 . 00000000
172.23.0.0	10101100 . 00010111 . 00000000 . 00000000

Calculating a Route Summary

Step 1: List networks in binary format.

172.20.0.0	10101100 . 00010100 . 00000000 . 00000000
172.21.0.0	10101100 . 00010101 . 00000000 . 00000000
172.22.0.0	10101100 . 00010110 . 00000000 . 00000000
172.23.0.0	10101100 . 00010111 . 00000000 . 00000000

Step 2: Count the number of left-most matching bits to determine the mask.

Answer: 14 matching bits = **/14** or **255.252.0.0**

Calculating a Route Summary

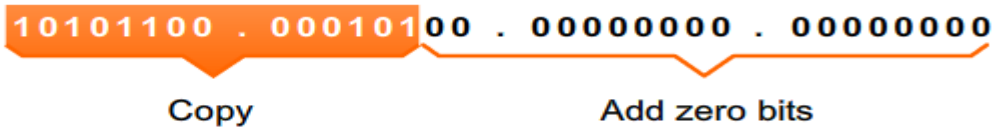
Step 1: List networks in binary format.

172.20.0.0	10101100 . 00010100 . 00000000 . 00000000
172.21.0.0	10101100 . 00010101 . 00000000 . 00000000
172.22.0.0	10101100 . 00010110 . 00000000 . 00000000
172.23.0.0	10101100 . 00010111 . 00000000 . 00000000

Step 2: Count the number of left-most matching bits to determine the mask.

Answer: 14 matching bits = **/14** or **255.252.0.0**

Step 3: Copy the matching bits and add zero bits to determine the network address.



Answer: 172.20.0.0

Review

- An internet is a collection of subnets.
- A subnet is defined by an IP prefix (using address/mask notation)
- IP implements hop-by-hop delivery of packets between hosts in an internet
- Routers connect subnets and forward packets across an internet
  - Forwarding: selection of packet's next hop, data plane, fast (in hardware)
  - Routing: computing forwarding state, signaling plane, not as time critical
- The forwarding table is the interface between the routing and forwarding processes
  - Destination
  - Next hop
- Forwarding table lookups are done using Longest Prefix Match

IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

Address Resolution Protocol

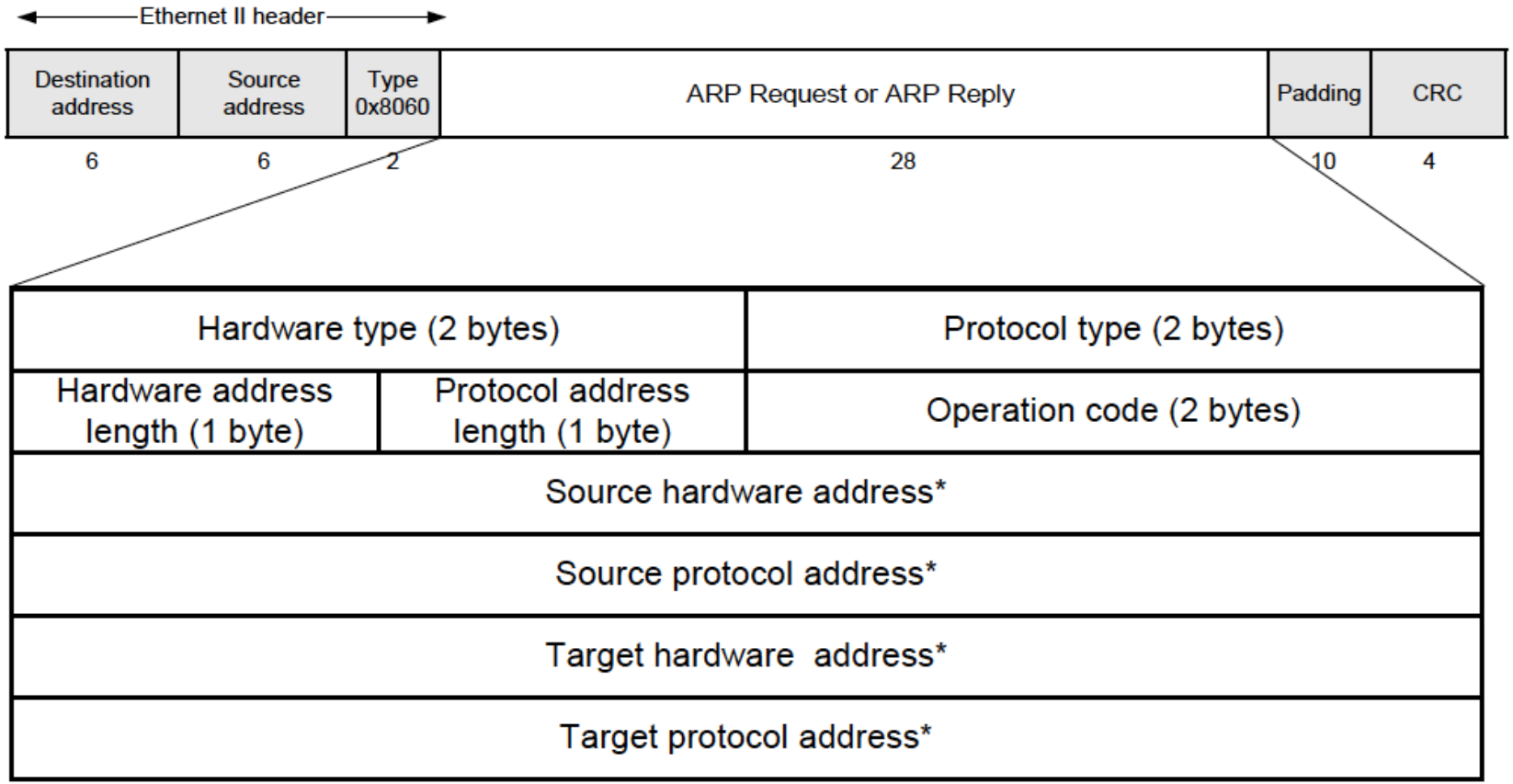
1. IP header validation
2. Process options in IP header
3. Parsing the destination IP address
4. Routing table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop
9. Send ICMP packet (if necessary)

Address Resolution Protocol (ARP)

Address Resolution Protocol (ARP)

- The Internet is based on IP addresses
- Data link protocols (Ethernet, FDDI, ATM) may have different (MAC) addresses
- ***The ARP (and RARP) protocols perform the translation between IP addresses and MAC layer addresses***
- We will discuss ARP for broadcast LANs, particularly Ethernet LANs
- Devices with IPv4 addresses and Ethernet NICs maintain an ARP cache or ARP table

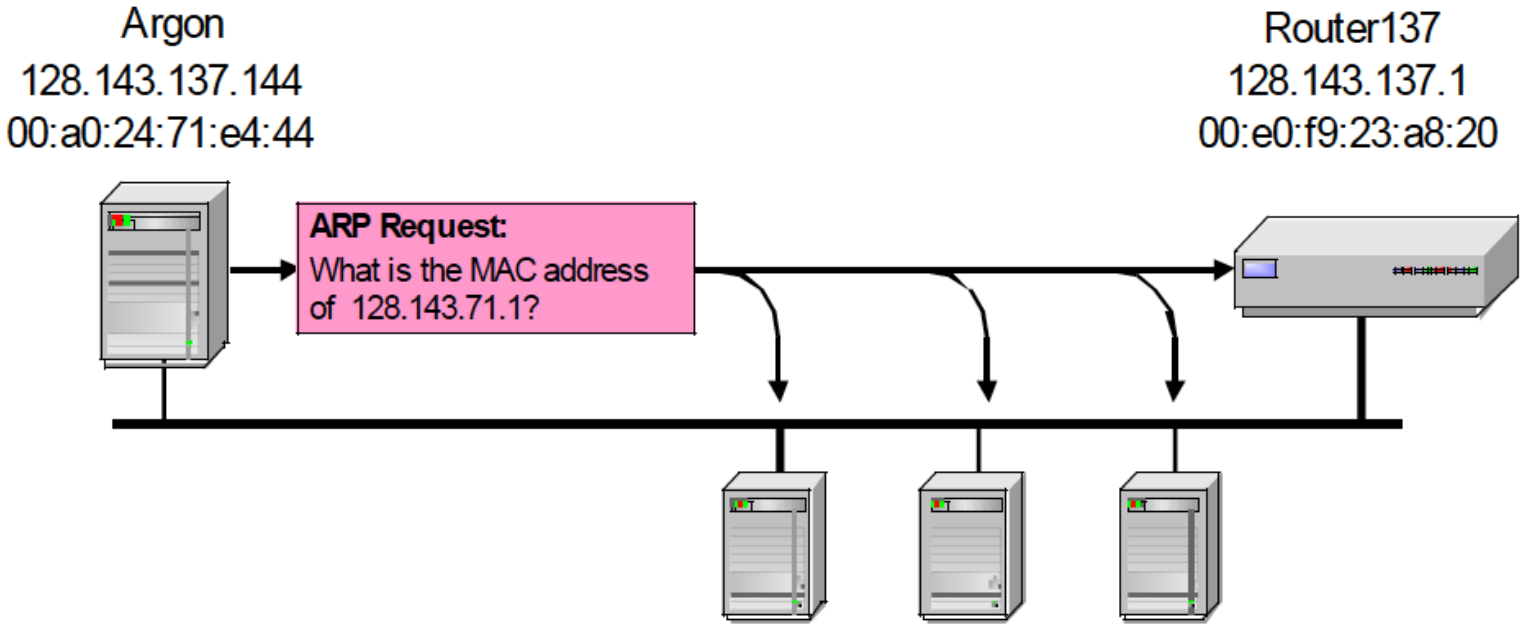
ARP Packet Format



Address Translation with ARP

ARP Request (after checking ARP cache):

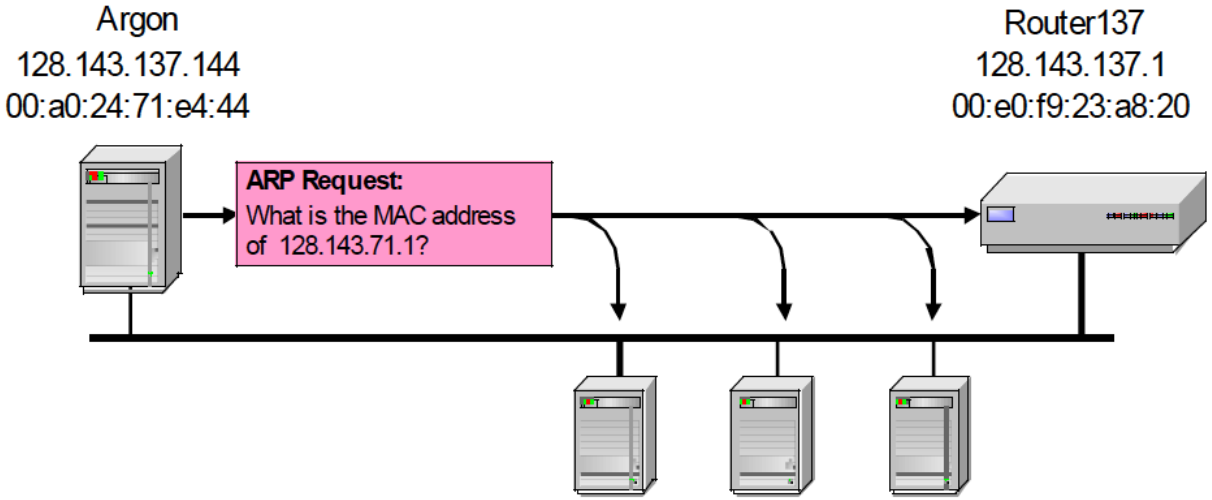
Argon **broadcasts (Ethernet)** an ARP request to all stations on the network: "**What is the hardware address of Router137?**"



Example

- ***ARP Request from Argon***:

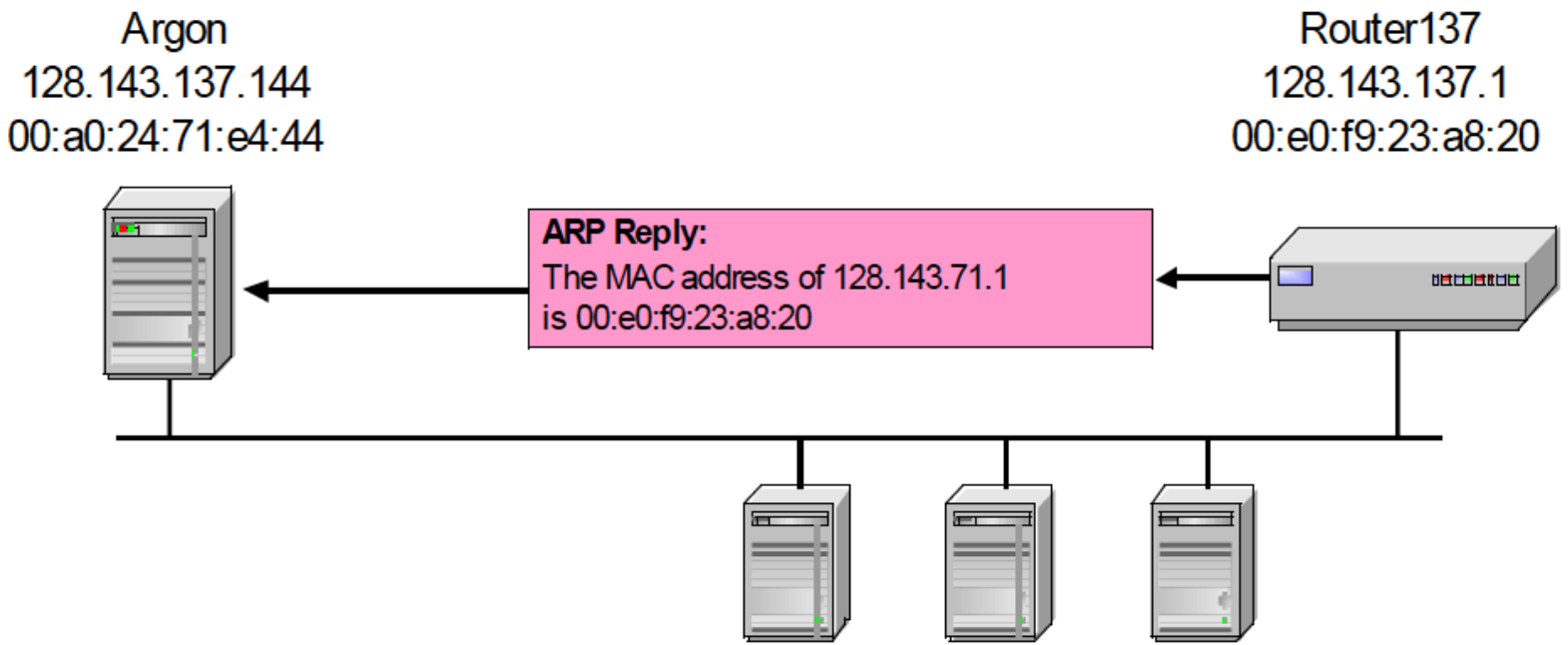
  Source hardware address: `00:a0:24:71:e4:44`
  Source protocol address: 128.143.137.144
  Target hardware address: `00:00:00:00:00:00`
  Target protocol address: 128.143.137.1



Address Translation with ARP

ARP Reply:

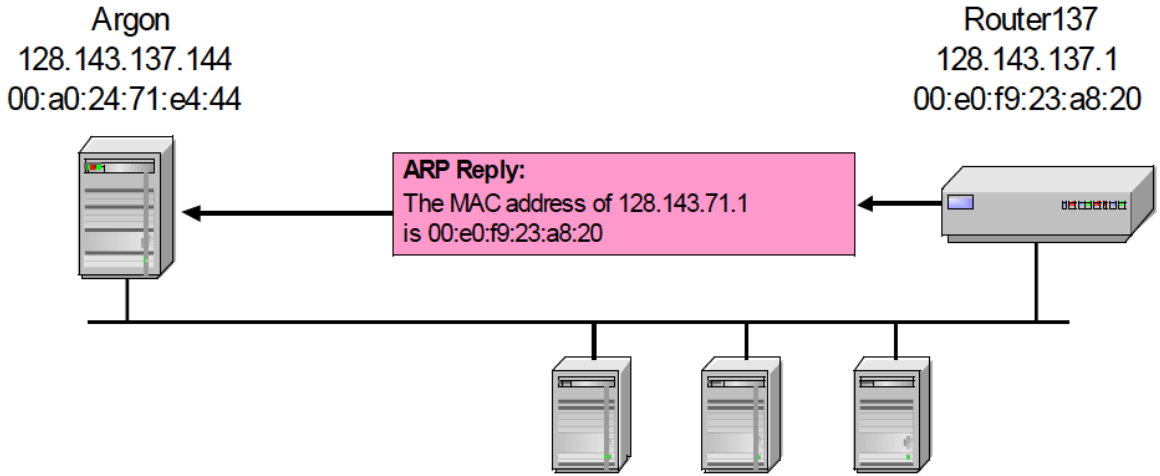
Router 137 **unicasts** an ARP reply to with its hardware address.



Example

- ***ARP Reply from Router137***:

  Source hardware address: `00:e0:f9:23:a8:20`
  Source protocol address: 128.143.137.1
  Target hardware address: `00:a0:24:71:e4:44`
  Target protocol address: 128.143.137.144



ARP Cache

- Since sending an ARP request/reply for each IP packet is inefficient, devices maintain a cache (ARP Cache) of current entries.
- The entries expire after 20 minutes (linux).
- Contents of the ARP Cache ("arp –a"):

(128.143.71.37) at 00:10:4B:C5:D1:15 [ether] on eth0
(128.143.71.36) at 00:B0:D0:E1:17:D5 [ether] on eth0
(128.143.71.35) at 00:B0:D0:DE:70:E6 [ether] on eth0
(128.143.136.90) at 00:05:3C:06:27:35 [ether] on eth1
(128.143.71.34) at 00:B0:D0:E1:17:DB [ether] on eth0
(128.143.71.33) at 00:B0:D0:E1:17:DF [ether] on eth0

Other ARP Uses

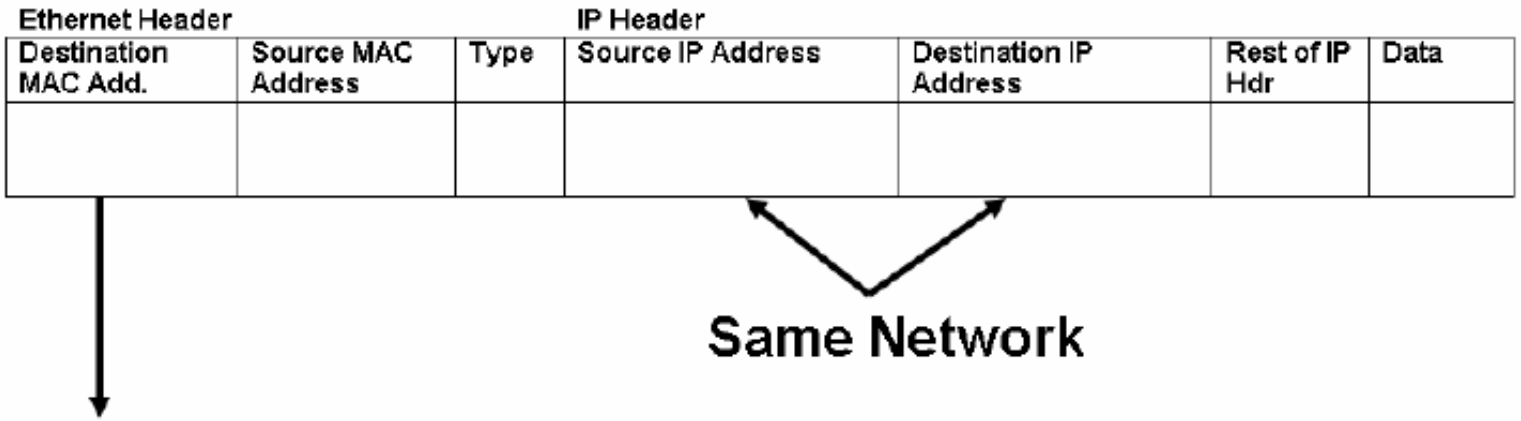
- What happens if an ARP Request is made for a non-existing host?

  Several ARP requests are made with increasing time intervals between requests. Eventually, ARP gives up.

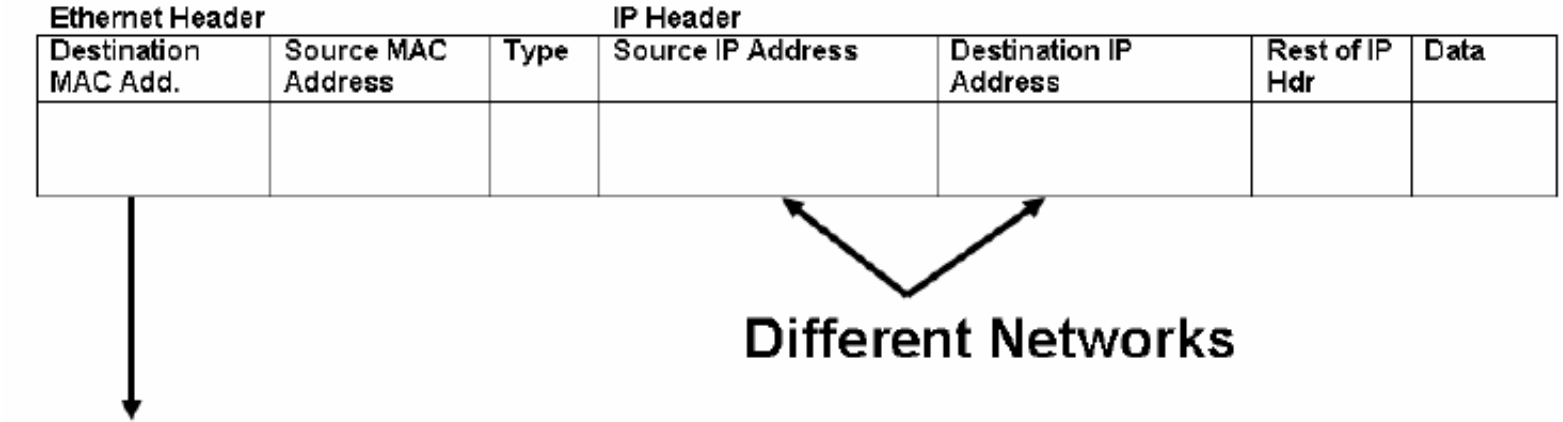
- What if a host sends an ARP Request for its own IP address?

  The other machines respond (**gratuitous ARP**) as if it was a normal ARP request.

  This is useful or detecting if an IP address has already been assigned.

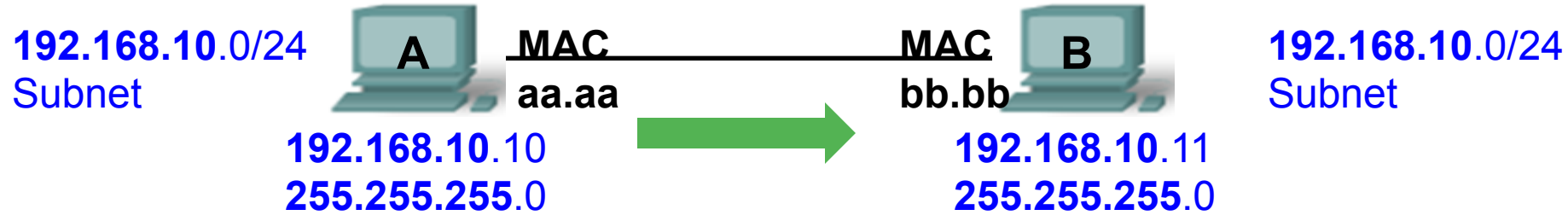


Destination MAC is the MAC Address of the host with the Destination IP Address in the IP packet.



Destination MAC is the MAC Address of the Default Gateway (Router).

Understanding IP communications



Destination Address	Source Address	Type	IP	FCS
bb.bb	aa.aa		DA 192.168.10.11	

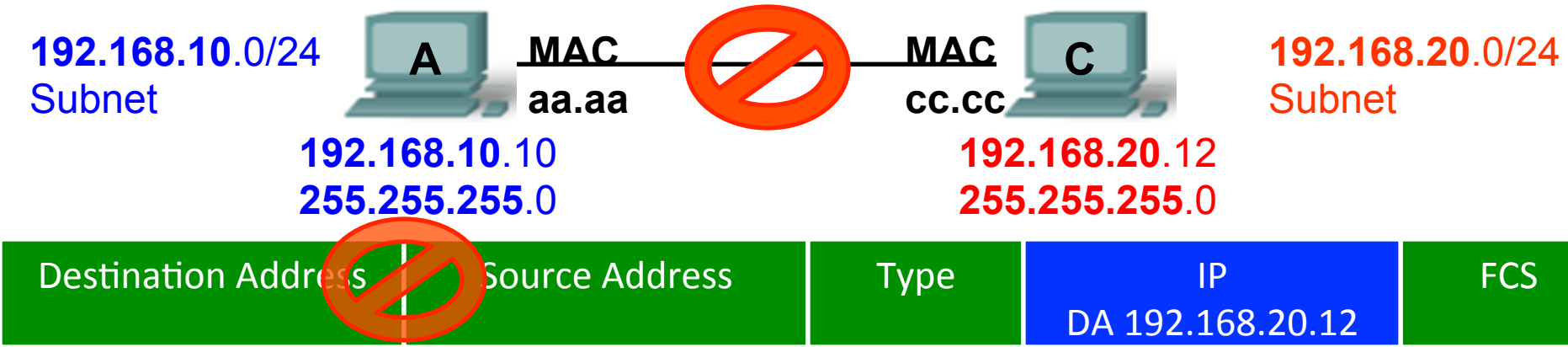
- Devices can only communicate with other devices on the same subnet
- A knows that it is on the 192.168.10.0/24 subnet (AND operation with its IP address and subnet mask). (Same subnet = Same subnet mask)
- A knows that B (192.168.1.11) is on its **same subnet** (AND operation with B's IP address and A's subnet mask)

```
A        192.168.10.10
   AND   255.255.255.0
  ---------------------
         192.168.10.0
```

SAME Subnet
A can reach B directly without going through a router

```
B        192.168.10.11
   AND   255.255.255.0
  ---------------------
         192.168.10.0
```

Understanding IP communications



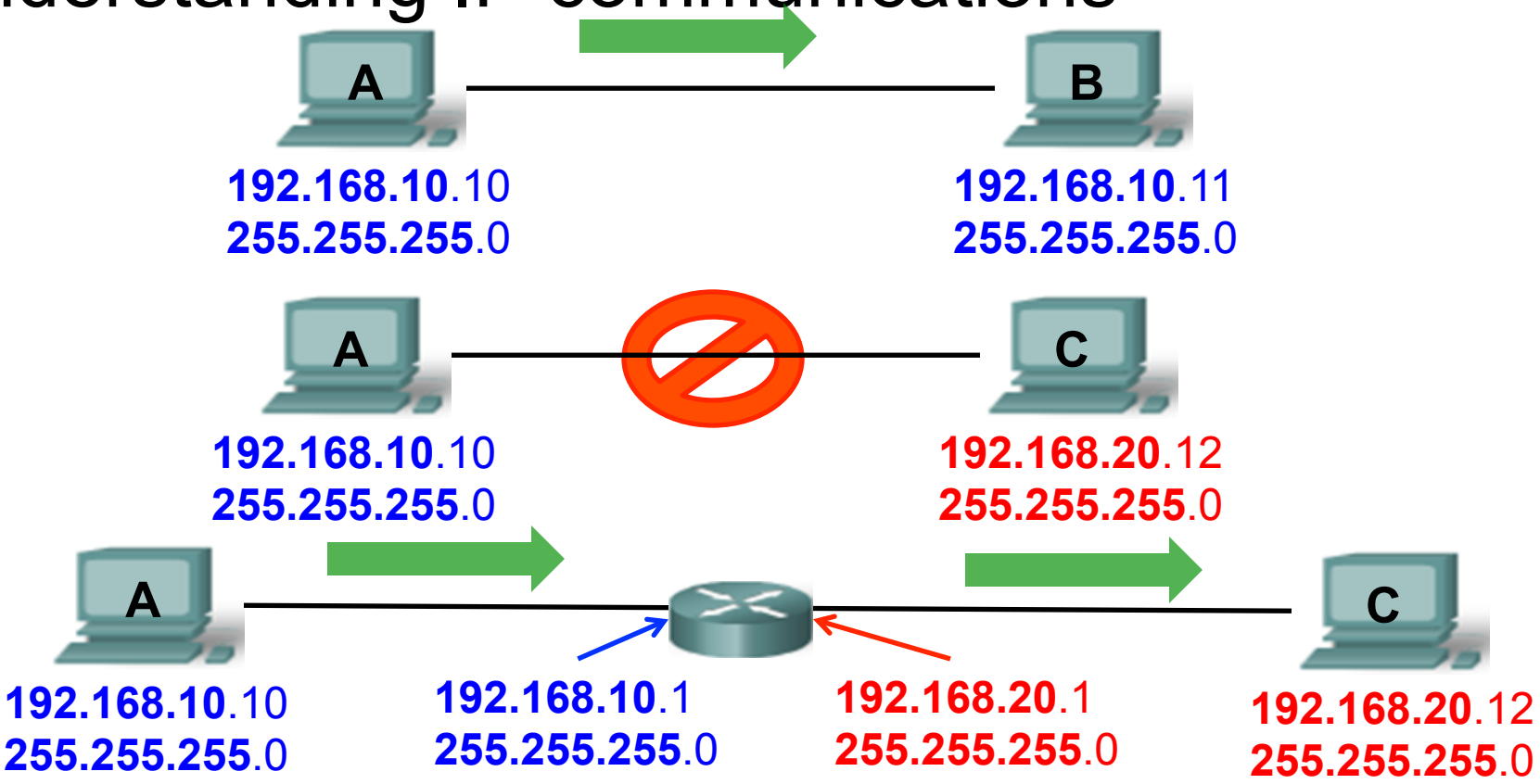
- Devices can only communicate with other devices on the same subnet
- A knows that it is on the 192.168.10.0/24 subnet (AND operation with its IP address and subnet mask) (Same subnet = Same subnet mask)
- A knows that C (192.168.20.12) is on a **different subnet** (AND operation with B's IP address and A's subnet mask) – ***Can't get there directly!***

```
A        192.168.10.10
    AND  255.255.255.0
  ------------------
         192.168.10.0
```

DIFFERENT Subnets
A can NOT reach B directly. Must go through a router

```
B        192.168.20.12
    AND  255.255.255.0
  ------------------
         192.168.20.0
```

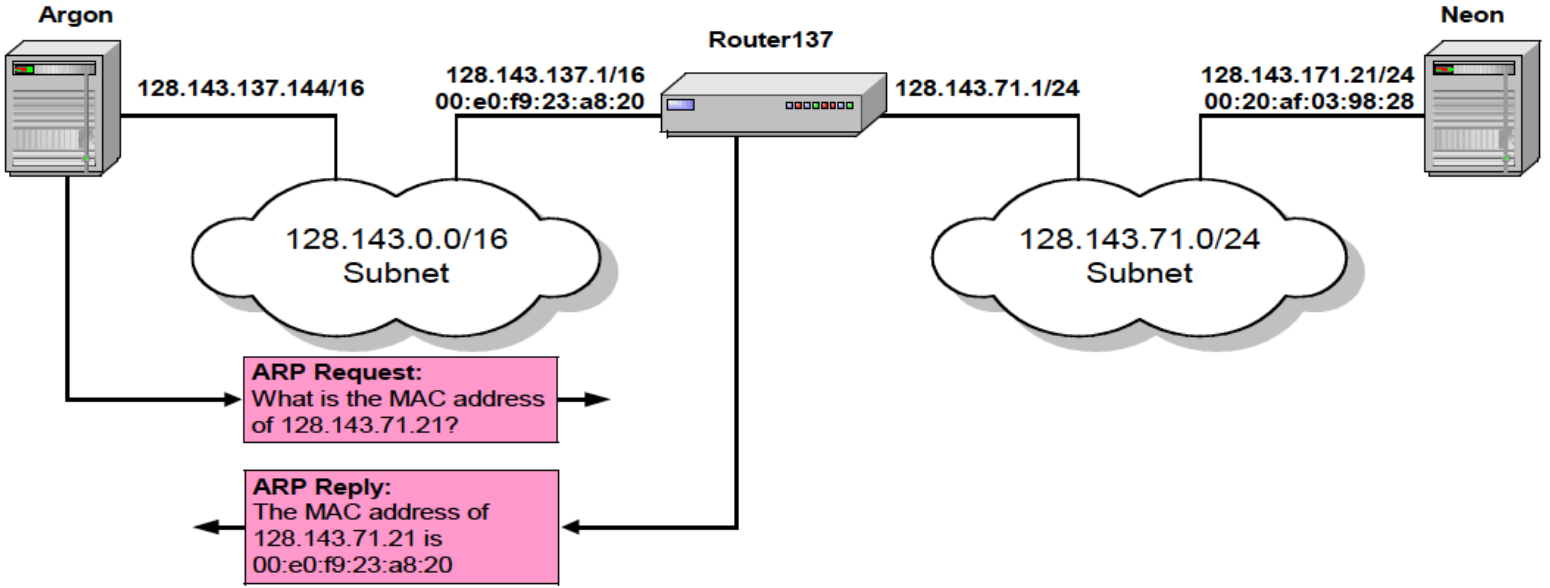
Understanding IP communications



- Devices can only communicate with other devices on the same subnet
- Otherwise, they must go through a router, that is on its same subnet

Proxy ARP (Don't use!)

- **Proxy ARP**: Host or router responds to ARP Request that arrives from one of its connected networks for a host that is on another of its connected networks.



Review

- The Address Resolution Protocol translates from IP to MAC addresses.
- ARP works by
  - Broadcasting ARP Requests for an IP address
  - Unicasting an ARP Reply with the MAC address to the requestor.
  - ARP Requests are repeated until a Reply is received or ARP times out.
- Hosts maintain an ARP cache to limit the need for ARP queries for every packet sent
- Gratuitous ARP are ARP Request/Replies that are issued for other than standard ARP purposes
  - Gratuitous ARP Requests detect if an IP address is in use
- Routers can be configured to issue Proxy ARP Replies to ARP Requests on one of its interfaces for hosts on another interface…but don't

IP Datagram Processing

Receive an IP datagram

IP header processing

Routing

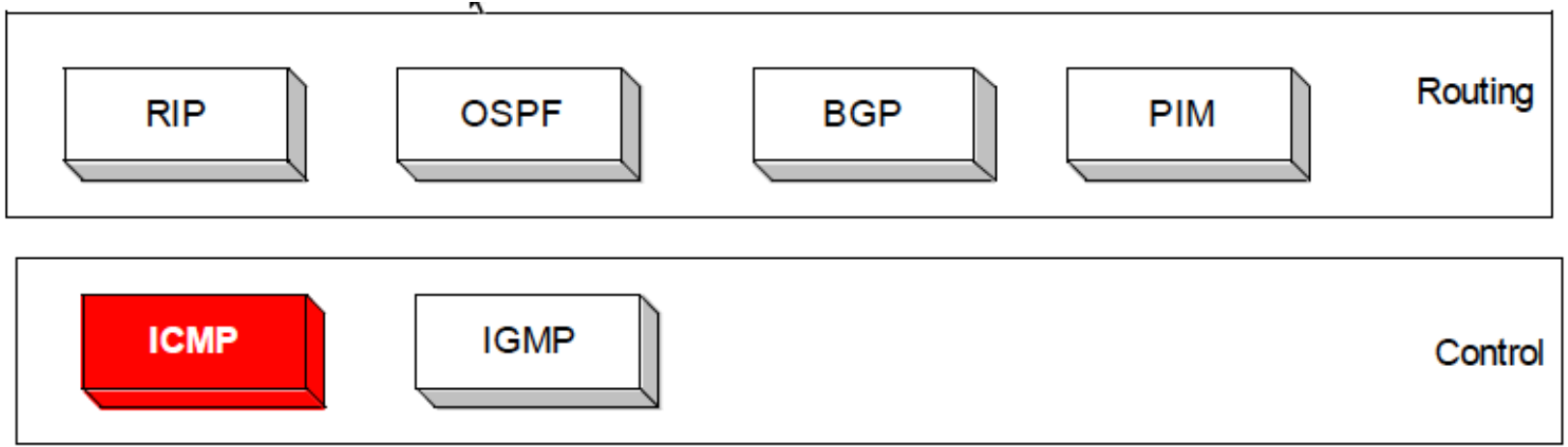
Address Resolution Protocol

Internet Control Message Protocol

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)

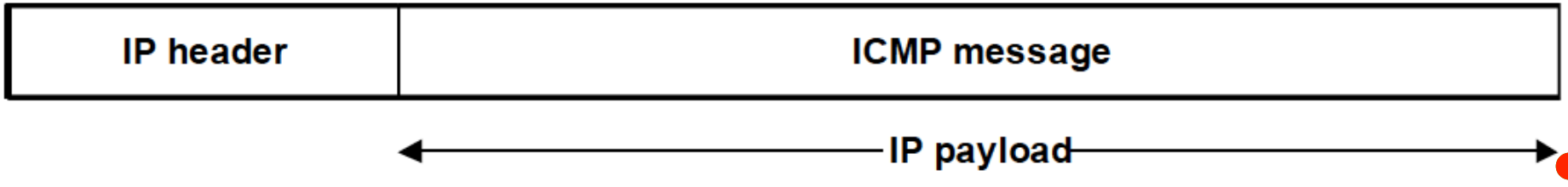
Overview

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



Overview

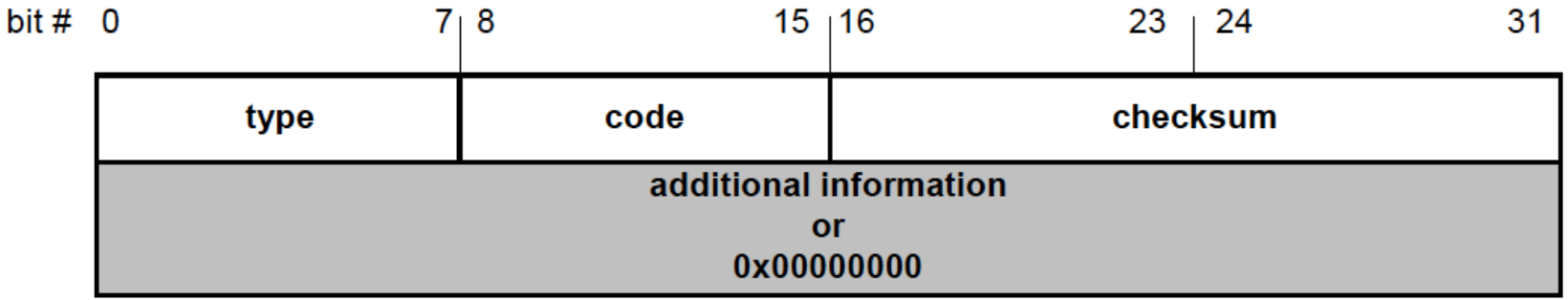
- The **Internet Control Message Protocol** (**ICMP**) is a helper protocol that supports IP with facility for
  - ***Simple queries***
  - ***Error reporting***
- Defined in RFC 792.
- Conceptually ICMP is a part of IP…
- …however is implemented “on top” of IP
- ICMP messages are encapsulated in IP datagrams:

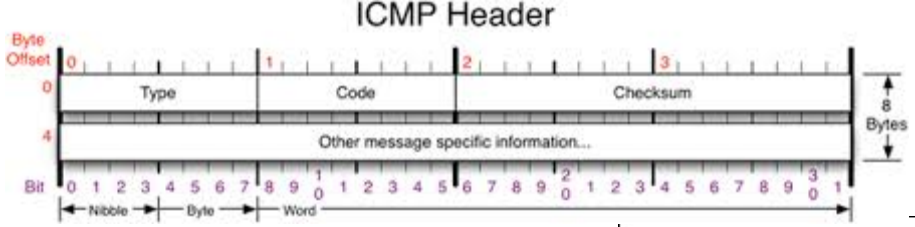


ICMP message format

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





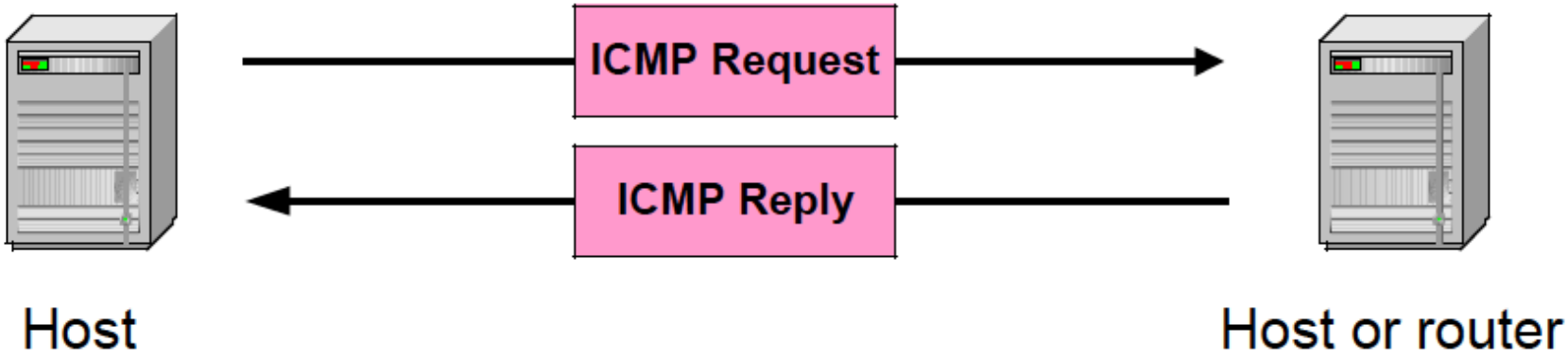
- The ICMP packets are identified by **type** and **code** fields.

Type	Code	Description	RFC
0	0	Ping, Echo reply	792
3	0	Net Unreachable	792
	1	Host Unreachable	
	2	Protocol Unreachable	
	3	Port Unreachable	
	4	Fragmentation Needed and Don't Fragment was Set	
	5	Source Route Failed	
	6	Destination Network Unknown	
	7	Destination Host Unknown	
	8	Source Host Isolated	
	9	Communication with Destination Network is Administratively Prohibited	
	10	Communication with Destination Host is Administratively Prohibited	
	11	Destination Network Unreachable for Type of Service	
	12	Destination Host Unreachable for Type of Service	
	13	Communication Administratively Prohibited	
4	0	Source Quench (Подавление источника)	792
5	0-3	ICMP Redirects	792
8	0	Ping, Echo request	792
9	0	Router Advertisement	1256
10	0	Router Discovery	1256
11	0	TTL Exceeded	792
	1	Fragment Reassembly Time Exceeded	
12	0-2	Bad IP Header	792
13	0	Timestamp Request	792
14	0	Timestamp Reply	792
30		Traceroute	1939
33		IPv6 Where-Are-You	-
34		IPv6 I-Am-Here	-

ICMP Query messages

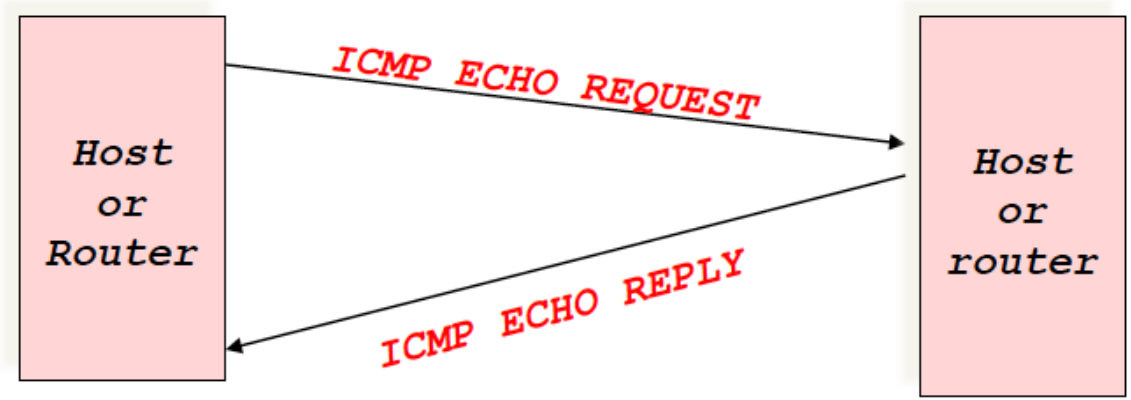
ICMP query:

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



Example of a Query: “ping”

- Each Ping is translated into an **ICMP Echo Request**
- The Ping’ed host responds with an **ICMP Echo Reply**
- **Additional information**: Identifier, Sequence #, Data



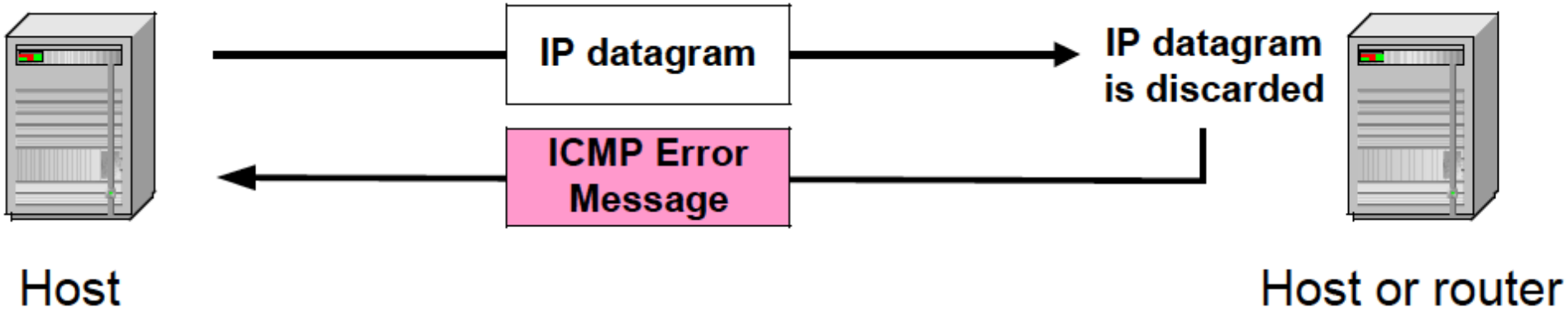
Example of ICMP Queries

Type/Code	Description
8/0	**Echo Request**
0/0	**Echo Reply**
13/0	**Timestamp Request**
14/0	**Timestamp Reply**
10/0	**Router Solicitation**
9/0	**Router Advertisement**

} The ping command uses Echo Request/ Echo Reply

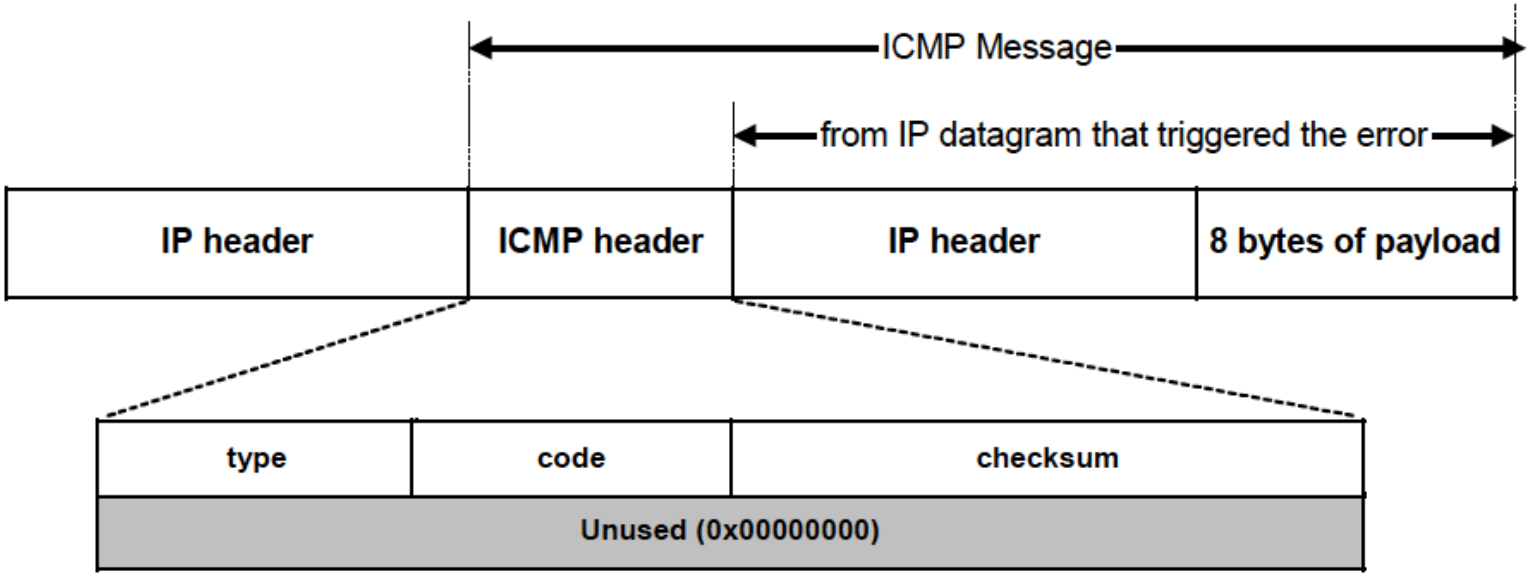
ICMP Error messages

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



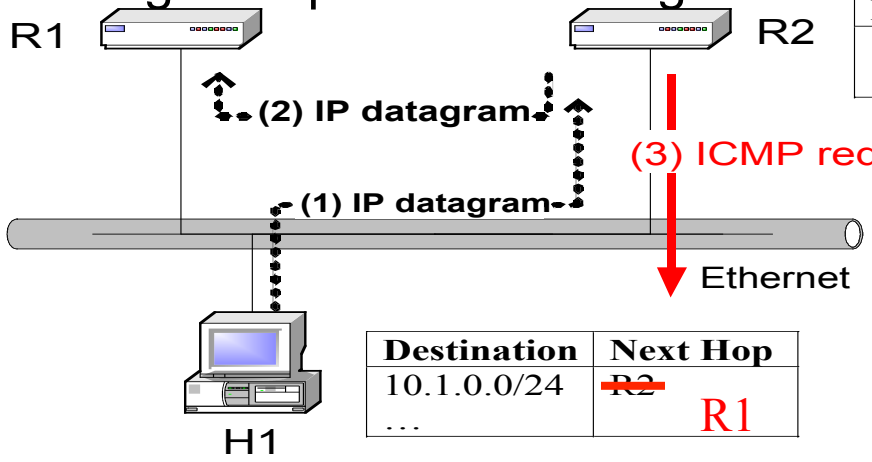
ICMP Error messages

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



ICMP Redirect Message

- 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

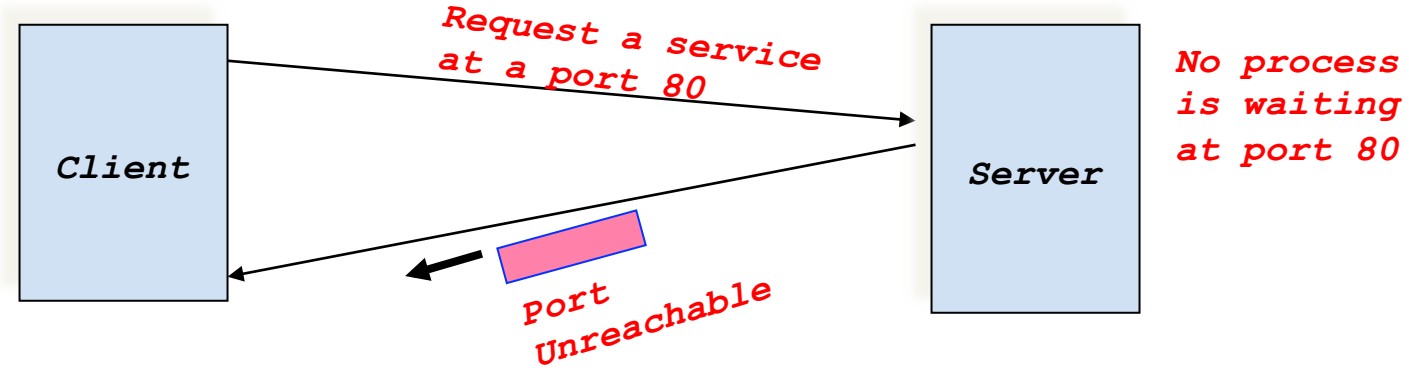


Destination	Next Hop
10.1.0.0/24	R1
…	

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

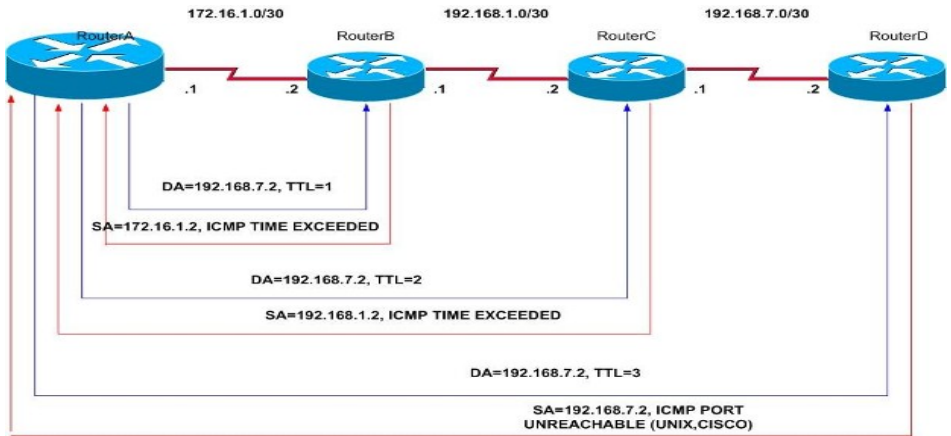
Example: ICMP Port Unreachable

- 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:



Example of an Error: traceroute

- Send UDP datagram to destination with IP TTL of 1.
- Wait for ICMP Time Exceeded message to get IP address of router (source).
- Increase TTL and repeat.
- Destination identified by use of high UDP port resulting in ICMP Port Unreachable message.
- Additional information (for both messages):
  - Internet Header
  - 64 bits of original datagram



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)

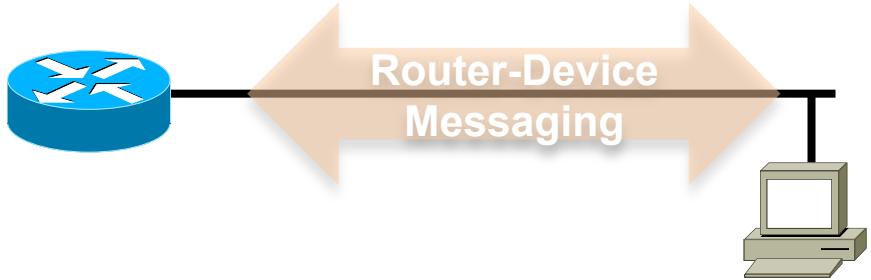
Some subtypes of the “Destination Unreachable”

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.

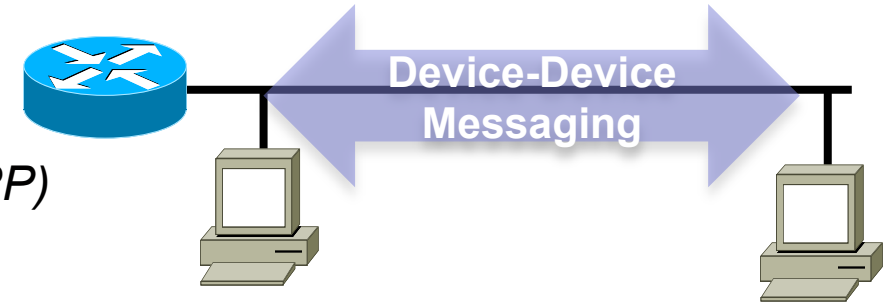
ICMPv6 Neighbor Discovery

ICMPv6 Neighbor Discovery defines 5 different packet types:

- **Router Solicitation Message**
- **Router Advertisement Message**
  Used with dynamic address allocation
  IRDP in IPv4 – never implemented



- **Neighbor Solicitation Message**
- **Neighbor Advertisement Message**
  Used with address resolution (IPv4 ARP)



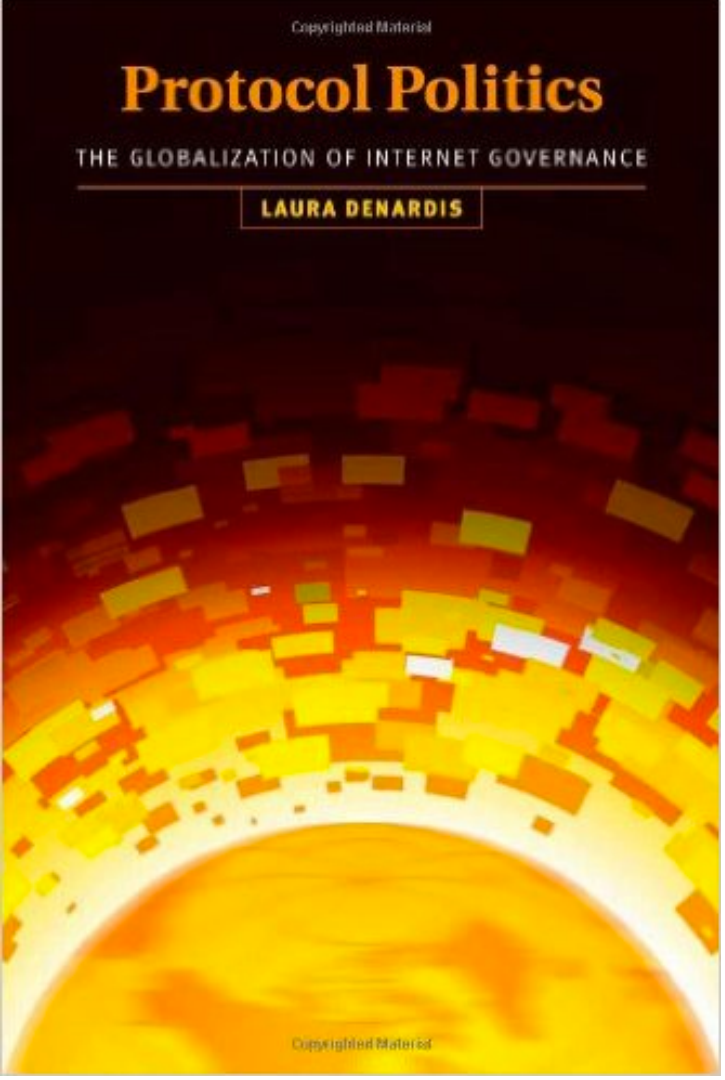
- **Redirect Message**
  Similar to ICMPv4 redirect message
  Router-to-Device messaging

Review

- ICMP provides two basic services:
  - Network queries
  - Error reporting
- Function of an ICMP message determined by Type and Code fields.
- For network queries
  - Type field defines matching Request/Reply types
  - Code field is 0
  - Additional information field used for parameters
- For error reporting
  - Type field identifies general class of errors
  - Code field identifies specific error
  - Typically sent when a datagram is discarded
  - Include IP header and first 8 bytes of payload (UDP or TCP data)

Extra Slides

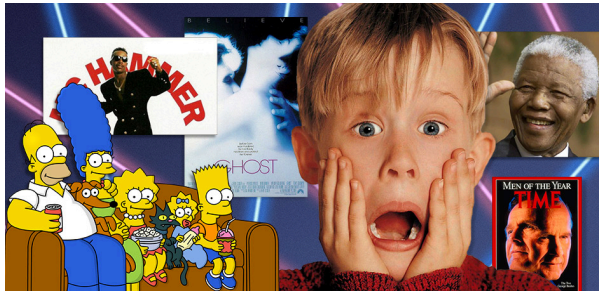
IPv4 to IPv6



Material from:

- **Protocol Politics: The Globalization of Internet Governance** (Information Revolution and Global Politics) *by Laura DeNardis*

The year was 1990 – A US-centric Internet



World Wide Web

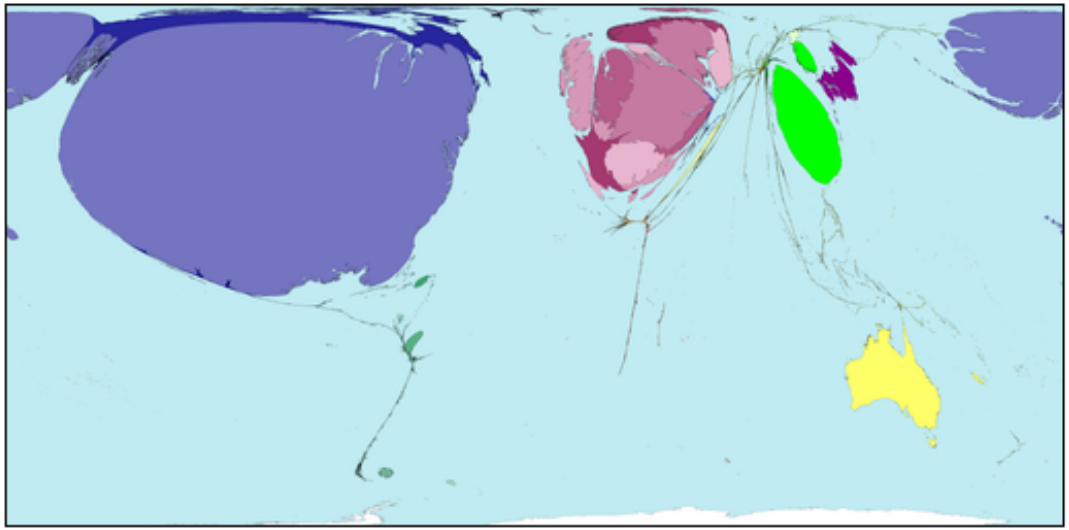
The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists , Policy , November's W3 news , Frequently Asked Questions .

What's out there?
Pointers to the world's online information, subjects , W3 servers, etc.
Help
on the browser you are using
Software Products
A list of W3 project components and their current state. (e.g. Line Mode ,X11 Viola , NeXTStep , Servers , Tools , Mail robot , Library)
Technical
Details of protocols, formats, program internals etc
Bibliography
Paper documentation on W3 and references.
People
A list of some people involved in the project.
History
A summary of the history of the project.
How can I help ?
If you would like to support the web..
Getting code
Getting the code by anonymous FTP , etc.

The first website on the World Wide Web went live in August, 1990.

- **Web did not exist** …. ***End of 1990, the first Web page*** was served on the open internet
- Prior to the web - no Amazon, Netscape or Yahoo.
- ***Americans the predominant users of the Internet – 73%***
  - Internet population about **3 million**
  - Most popular application: **text-based email**
  - **No images, video or voice**



"I sit down comfortably in the university's modern computer lab and take advantage of the technology available to enter the digital world. The digital world!" Gabriela Tôrres Barbosa, 2006

In 1990 the Internet had existed for only 7 years; just 3 million people had access to it worldwide. 73% of these people were living in the United States, 15% were in Western Europe

Internet users in 1990 were recorded in just a few other territories. Outside Western Europe and the United States, most users lived in Canada, followed by Australia, then Japan, the Republic of Korea and Israel. In 1990 there was practically no access elsewhere.

Switzerland is home to the European Organisation for Nuclear Research (CERN) where the World Wide Web was developed. In 1990, 5.8 people per thousand in Switzerland used the Internet.

Territory size shows the proportion of worldwide Internet users who lived there in 1990.

IAB and Vint Cerf

Network Working Group
Request for Comments: 1120

V. Cerf
NRI
September 1989

The Internet Activities Board

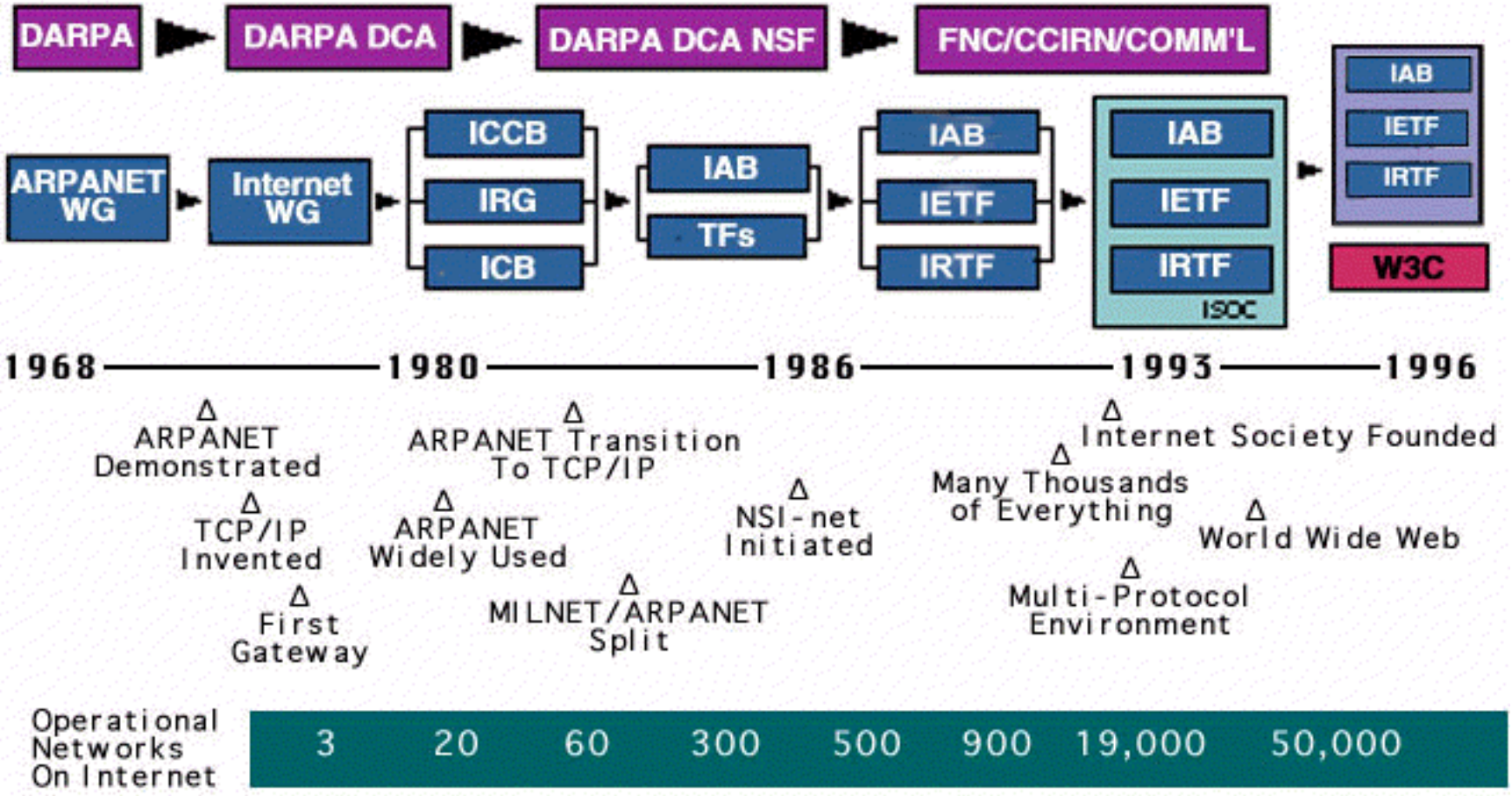
Status of this Memo

This RFC provides a history and description of the Internet Activities Board (IAB) and its subsidiary organizations. This memo is for informational use and does not constitute a standard. Distribution of this memo is unlimited.

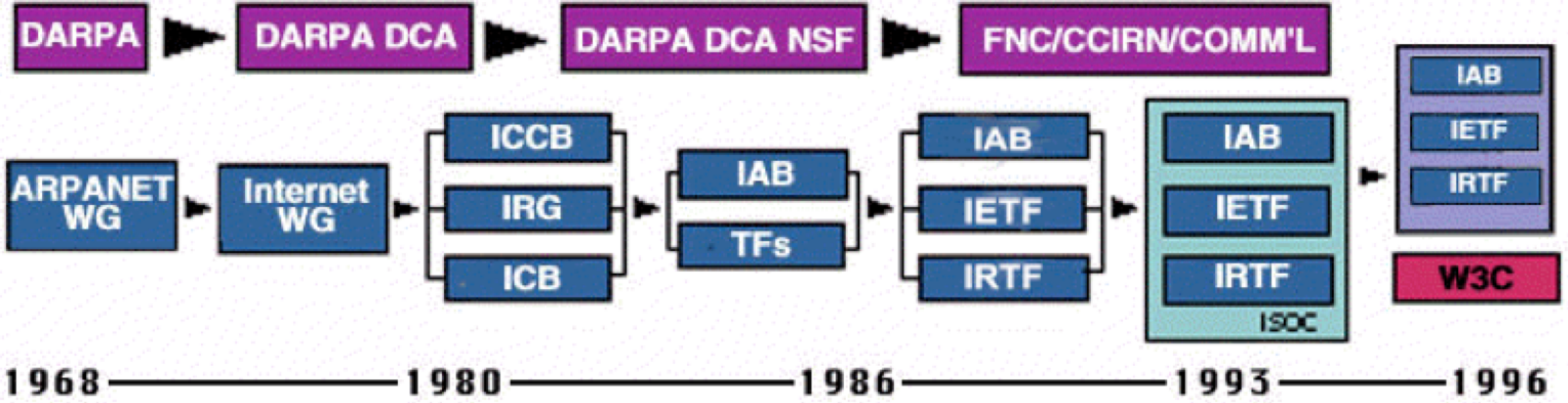
1. Introduction

In 1968, the U.S. Defense Advanced Research Projects Agency (DARPA) initiated an effort to develop a technology which is now known as packet switching. This technology had its roots in message switching methods, but was strongly influenced by the development of low-cost minicomputers and digital telecommunications techniques during the mid-1960's [BARAN 64, ROBERTS 70, HEART 70, ROBERTS 78]. A very useful survey of this technology can be found in [IEEE 78].

- **IAB** (***Internet Activities Board***, later named to ***Internet Architecture Board***)
- Responsibilities:
  - 1. **Ultimate responsibility** for the ***direction of the Internet***
    - IAB chair (1990) – **Vinton Cerf**
  - 2. **Managing RFC process – oversee IETF**
    - The late **John Postel** – collector, editor and archivist until his death in 1998.
    - Originally, *paper documents – later mailing list*
    - RFC's standards track:
      - ***Proposed standard > Draft standard > Standard***
      - Also, **Current best practices**, **Informational**, **Experimental**
- The Internet's core philosophy was that of "***working code and rough consensus***."

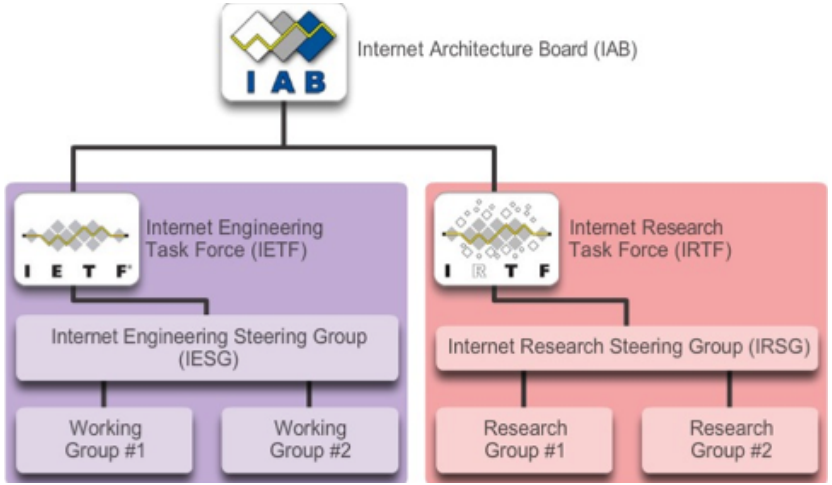

DARPA
DARPA DCA
DARPA DCA NSF
FNC/CCIRN/COMM'L
ARPANET WG
Internet WG
ICCB
IRG
ICB
IAB
TFs
IAB
IETF
IRTF
IAB
IETF
IRTF
ISOC
IAB
IETF
IRTF
W3C
1968
1980
1986
1993
1996
ARPANET Demonstrated
TCP/IP Invented
First Gateway
ARPANET Transition To TCP/IP
ARPANET Widely Used
MILNET/ARPANET Split
NSI-net Initiated
Internet Society Founded
Many Thousands of Everything
World Wide Web
Multi-Protocol Environment
Operational Networks On Internet
3
20
60
300
500
900
19,000
50,000

IAB (Internet Activities Board)

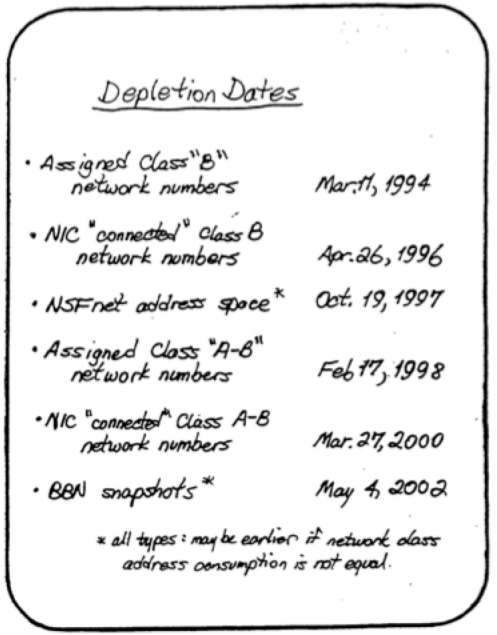


- **11 individuals**
- **Mostly Americans** working at corporations, universities, and research *institutions.*
- **Appointed** by the chair of the IAB
- Quarterly meetings
- Formalized in **1983**, origins in late 1970s, ICCB (Internet Configurations Control Board)
- Responsibility for **Internet protocol architecture**
  - ***Ultimate responsibility for approving protocols***

IETF (Internet Engineering Task Force)



Frank Solensky's Report on Address Depletion, Proceedings of IETF 18, Vancouver, August 1990 (http://www.ietf.org/proceedings/18.pdf)



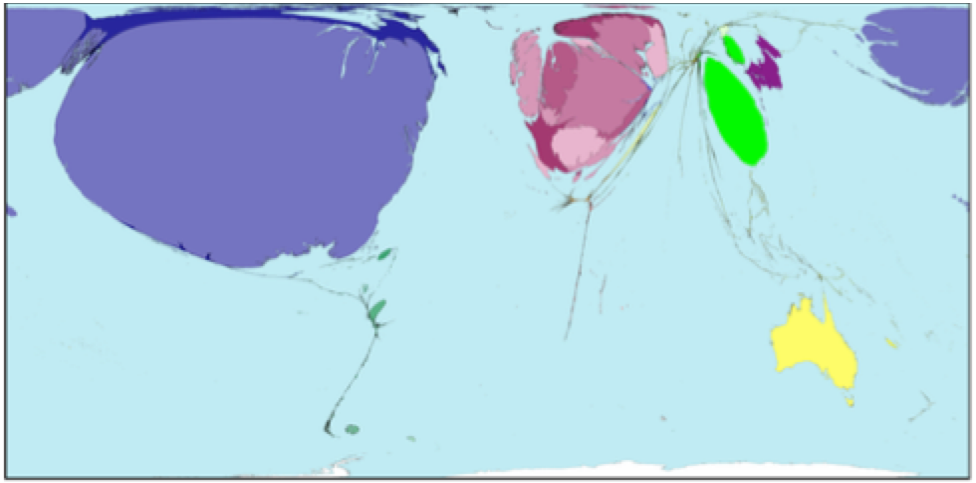
Depletion Dates

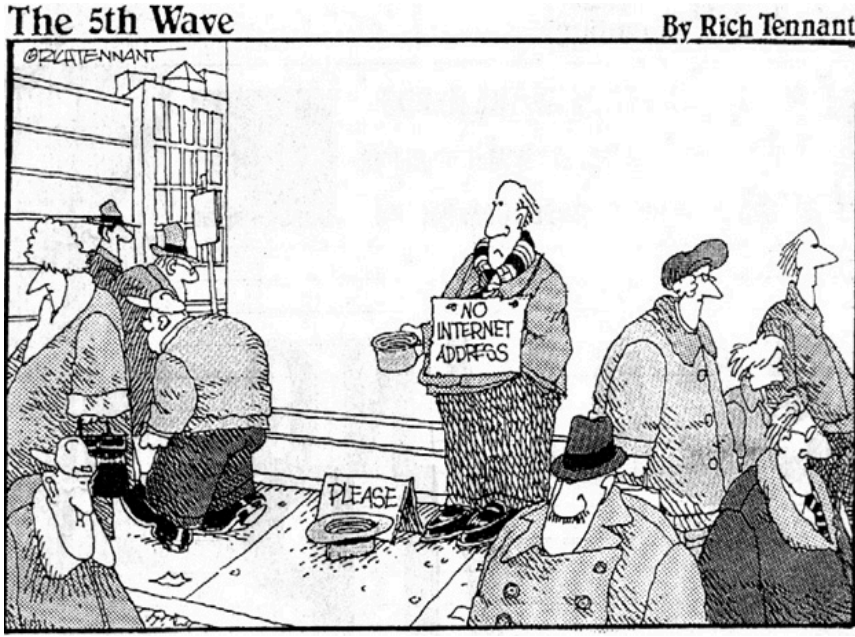
• Assigned Class "B" network numbers	Mar. 11, 1994
• NIC "connected" Class B network numbers	Apr. 26, 1996
• NSFnet address space*	Oct. 19, 1997
• Assigned Class "A-B" network numbers	Feb 17, 1998
• NIC "connected" Class A-B network numbers	Mar. 27, 2000
• BBN snapshots*	May 4, 2002

* all types: may be earlier if network class address consumption is not equal.

- **1986**, IAB established the IETF (Internet Engineering Task Force)
- *No formal membership, volunteers* – Mostly Americans
- Triennial informal meetings
- **1990** – Discussions about *shortage of IP addresses*
- **August 1990 IETF Vancouver meeting** – Current IP address assignment rate would deplete IP address space by March 1994
- ***IAB also concerned***.

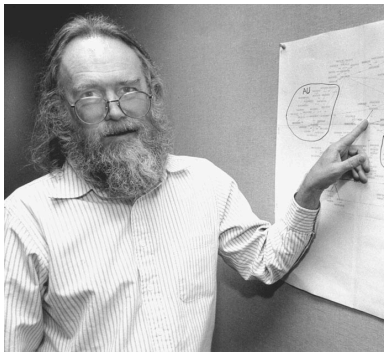
IAB Concerns



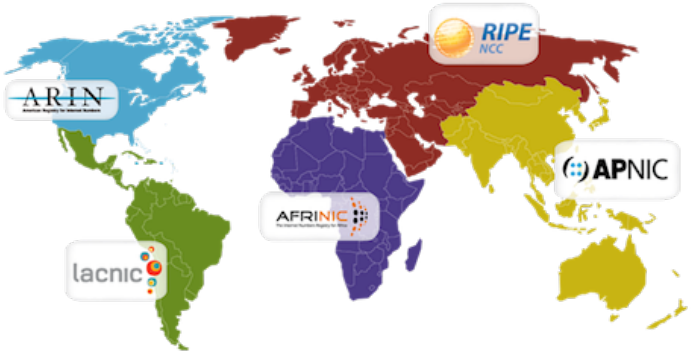


- **IAB** acknowledged, "***rapidly growing concern internationally***" that American institutions controlled the distribution of Internet resources (***IAB mostly if not all US***)
- Remember, US had 73% of Internet users, ARPANET was created in the US
- **IPv4 addresses** scarce and finite resource
- 32 bits = 4.3 billion addresses but still finite
- Requires some conservation and control.

IP Address Allocation in 1991



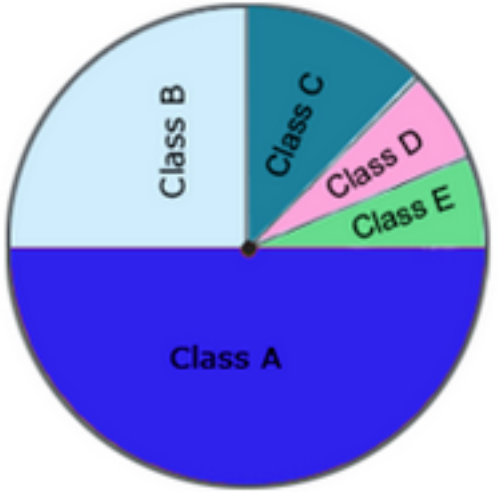
iana
Internet Assigned Numbers Authority



- **Early days, IP addresses allocated by John Postel**
- John Postel **worked** at the University of Southern California's Information Science Institute (ISI)
- **Funded** by the US Department of Defense (DoD)
- **Later**, **IANA** (Internet Assigned Numbers Authority), at USC (ISI) with John Postel playing a central role.
- **By 1990**, **IANA** had delegated the process to SRI International's Networking Information Center (DDN-NIC), funded by the US DoD, in Marina Del Ray, California, USA
- *Regional Internet Registries (RIRs) not created yet*

IAB Concerns and the Need to be More International

Internet Assigned Numbers Authority



- **IAB had concerns**:
  - **Address assignment** should be more internationally distributed
  - **Not controlled** by US-centric institution funded by US DoD
- **Cerf** Issued a recommendation to the Federal Networking Council (FNC), then the US government's coordinating body for agencies supporting the Internet:
  - **IP addresses** should be allocated by international organizations
  - **IANA** still retaining centralized control
- **Class A and B** addresses becoming scarce

OSI

TCP/IP	ISO	AppleTalk	Novell Netware
HTTP DNS DHCP FTP	ACSE ROSE TRSE SESE	AFP	NDS
TCP UDP	TP0 TP1 TP2 TP3 TP4	ATP AEP NBP RTMP	SPX
IPv4 IPv6 ICMPv4 ICMPv6	CONP/CMNS CLNP/CLNS	AARP	IPX
Ethernet PPP Frame Relay ATM WLAN			

- **OSI protocols** advanced by ISO (International Organization for Standardization)
  - **Not** US-centric IETF
- **In contention** to become the global interoperability standard
- *Standardization needed amongst products*
- **OSI is international**, sanctioned by numerous governments, particularly western Europe
- **1990**, **US government** mandated that its government procured products conform to OSI standard
- **US DoD,** original proponent of TCP/IP, knew this was inevitable.
- Despite government endorsement of OSI, there was competition between the underlying Internet (TCP/IP) and OSI.

Popular standards

- ISO 9000 Quality management
- ISO 14000 Environmental management
- ISO 3166 Country codes
- ISO 26000 Social responsibility
- ISO 50001 Energy management
- ISO 31000 Risk management
- ISO 22000 Food safety management
- ISO 27001 Information security management
- ISO 45001 Occupational health and safety

We're ISO, the International Organization for Standardization. We develop and publish International Standards.

Are you looking to **buy an ISO standard?**

Visit the ISO Store

Preview ISO standards

Preview content before you buy, search within documents and keep up to date with changes using our Online Browsing Platform.

Looking to get certified?

ISO doesn't provide certification or conformity assessment. You'll need to contact an external certification body for that.

Common questions

- What is a standard?
- How are standards developed?
- What are the benefits of standards?

Popular standards

ISO 9000 - Quality management

Make sure your products and services meet customers' needs with this family of standards.

Learn more about ISO 9000

ISO 14000 - Environmental management

Improve your environmental performance with this family of standards.

Learn more about ISO 14000

ISO 3166 - Country codes

Avoid confusion when referring to countries and their subdivisions with this standard.

Learn more about ISO 3166

ISO 22000 - Food safety management

Inspire confidence in your food products with this family of standards.

Learn more about ISO 22000

ISO 26000 - Social responsibility

Help your organization to operate in a socially responsible way with this standard.

Learn more about ISO 26000

ISO 50001 - Energy management

Make energy savings and help make your organization more efficient with this standard.

Learn more about ISO 50001

ISO 31000 - Risk management

Manage risks that could be negative for your company's performance with this standard.

Learn more about ISO 31000

ISO 4217 - Currency codes

Avoid confusion when referring to world currencies with this standard.

Learn more about ISO 4217

ISO 639 - Language codes

Describe languages in an internationally accepted way with this standard.

Learn more about ISO 639

ISO 20121 - Sustainable events

Manage the social, economic and environmental impacts of your event with this standard.

ISO 27001 - Information security

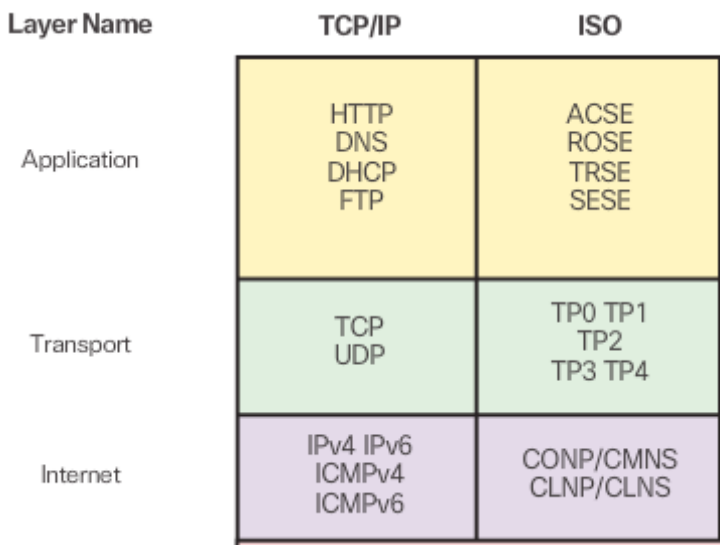
Ensure your organization's information is secure with this family of standards.

ISO 45001 - Occupational Health and Safety

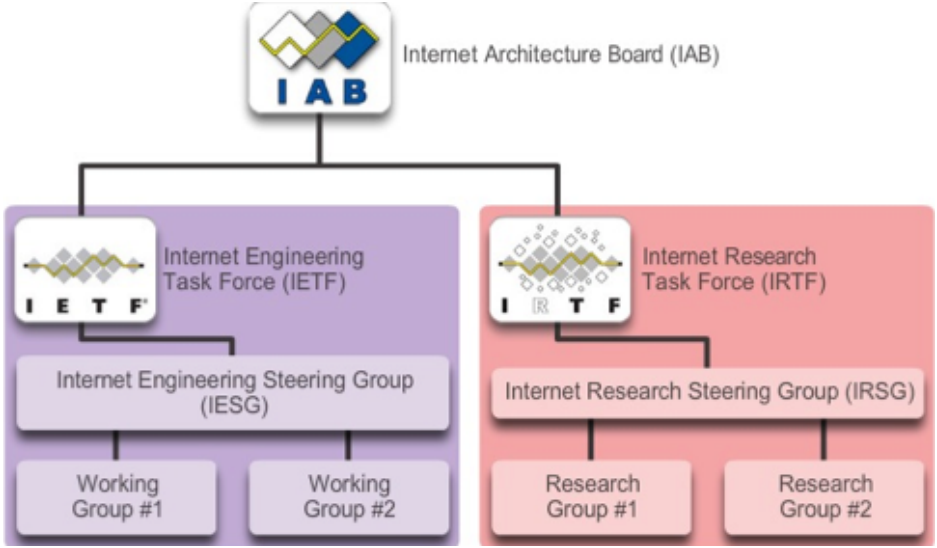
Reduce workplace risks and create safer working environments.

TCP/IP (IETF) versus OSI (ISO)

- **1990**: Not clear which family of protocols would be the dominant vendor-neutral suite of protocols for the Internet
- **OSI**
  - **Limited deployment**
  - **Backing of** international governments and the US National Institute of Standards (NIST)
  - **Increasing investment** by large networking computing vendors like Digital Equipment Corporation (DEC)
- **TCP/IP**
  - **Working set and dominant protocol suite**
  - Increased presence in private corporations
  - **Backing of Internet's technical community (IETF)**
  - **Well documented** (RFCs)



IAB Meeting – January 1991, USC-ISI, California



- **IAB and IESG** two-day "soul searching" meeting.
- **Topics**
  - **Export restrictions** in encryption products
  - **Divisive issue of OSI** (Open Systems Interconnection)
- **Interoperability** between vendor products an issue
  - Did not always use common network protocols

IAB Meeting – January 1991, USC-ISI, California

- **Meeting addressed six problem areas:**
  1. **The multiprotocol Internet**
  2. Routing and addressing
  3. Getting big
  4. Dealing with divestiture
  5. New services (e.g. video)
  6. Security

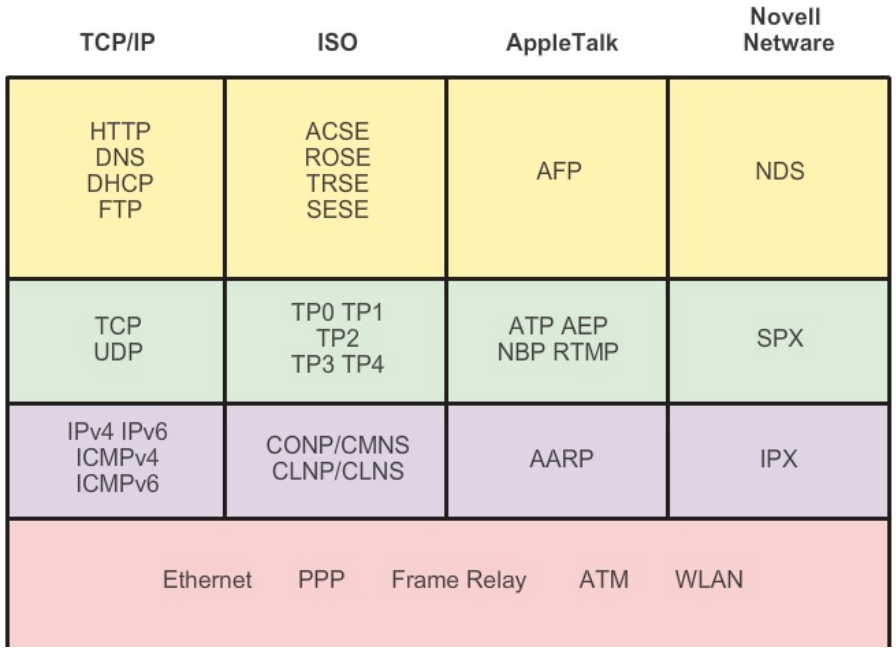
- **First area addressed four alternatives**:
  1. **OSI and TCP/IP** coexist indefinitely
  2. **TCP/IP be replaced by OSI**
  3. **OSI could fade** and **TCP/IP remain** the protocol suite of the Internet
  4. **Next-generation protocol suite** replace both TCP/IP and OSI
- Meeting described as "***spirited, provocative, and at times controversial***"
- *Almost everyone backed the continued concurrent development of* ***both TCP/IP and OSI.***

IAB/IESG Retreat – June 1991, USC-ISI, California

- **Decided** needed an **additional 3-day "architecture retreat"**
- **June 1991, 3-day retreat** (5 subgroups)
  - **32 IAB and IESG members**, and some guests
  - Corporations, universities and research institutions
  - **Outcome published as RFC 1287**, Towards the Future of Internet Architecture
  - **ROAD** (ROuting and ADdressing) **Subgroup** dealt with: Address space exhaustion – replacing IPv4
  - ROAD made recommendations to IAB (coming)
- **One alternative:**
  - **Retain 32-bit address format**
    - **Eliminate** need for uniqueness
    - **Internet regions** would be globally unique and translate addresses
  - **Another alternative** – expand from 32-bit to 64-bit address
  - **Under international pressure to adopt OSI protocols**
  - **US government** supported OSI through GOSIP standard (Government Open Systems Interconnection Protocol)

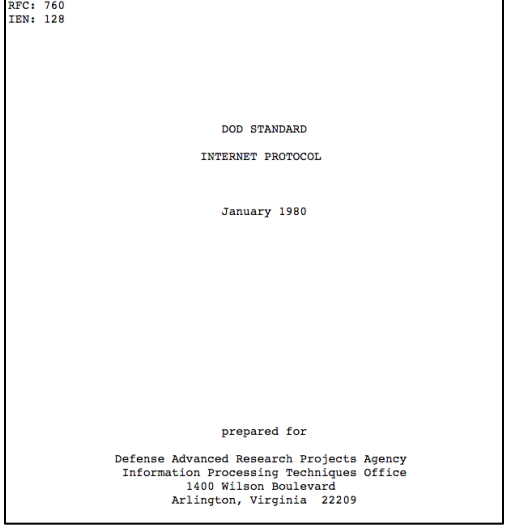
IAB/IESG Retreat – June 1991, USC-ISI, California

- **US corporation networking environments** were multiprotocol.
- **Also** included IBM's SNA (Systems Network Architecture) and DEC's DECnet.
- **LANs were typically isolated** technical islands with Novell IPX/SPX and Apple AppleTalk.
- **Question** was, ***"Would the universal standard be OSI, TCP/IP or something totally new?"***
- June meeting members, acknowledged that there were "powerful political and market forces" behind OSI.
- **Believed** decision should assess protocol alternatives based on technology, not on culture or politics.



- Group acknowledged that IP connectivity historically defined Internet connectivity.
- Reachable via **PING**

IPv4



RFC: 760
IEN: 128

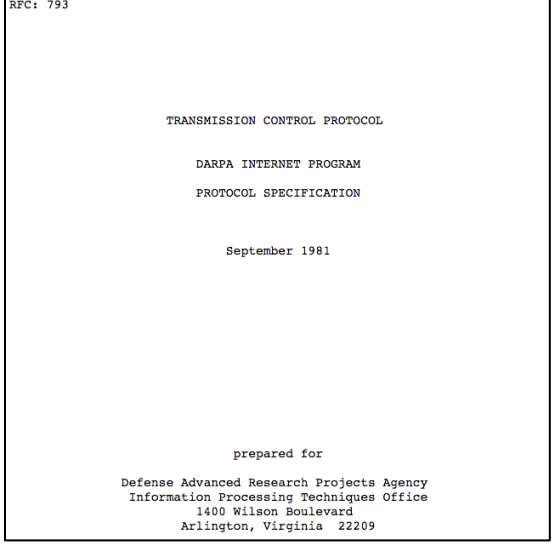
DOD STANDARD

INTERNET PROTOCOL

January 1980

prepared for

Defense Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Virginia 22209



RFC: 793

TRANSMISSION CONTROL PROTOCOL

DARPA INTERNET PROGRAM

PROTOCOL SPECIFICATION

September 1981

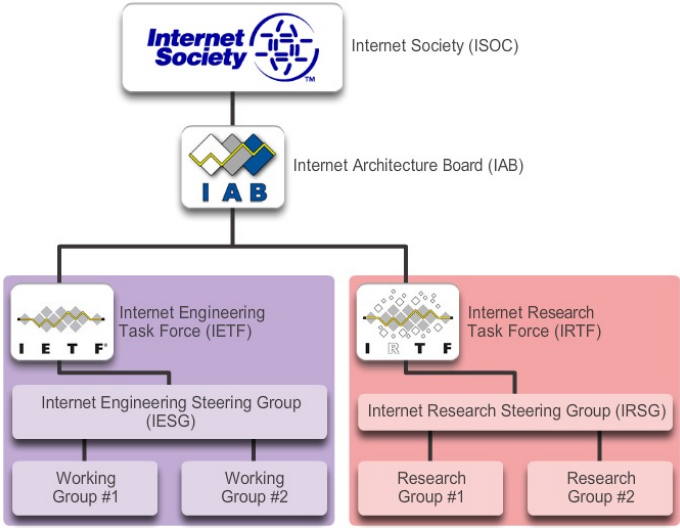
prepared for

Defense Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Virginia 22209

- **1991**: IPv4 now 10 years old
  - **RFC 760** Jan 1980 obsoleted by RFC 791 Sep 1981
  - The same year as the IBM PC was introduced
- **Why version 4?**
  - There was no official predecessor to IPv4
  - Its **function branched off from TCP,** which previously had 3 versions
  - Four versions were developed: TCP v1, TCP v2, TCP v3 and IP v3, and TCP/IP with IPv4.
  - See: https://en.wikipedia.org/wiki/Internet_protocol_suite

ISOC – Internet Society

- **IAB seeking** greater "internationalization of IAB and its activities"
- **IETF** was mostly Americans
- **1992**, *group led by Vint Cerf* **established Internet Society (ISOC).**
- **Possible issue of liability** and questions whether IETF members might face **lawsuits** from those who thought Internet standards did them harm.
- **Also**, decline in US government funding of Internet standards activities and increased commercialization.
- **ISOC** would provide more legitimacy and legal protection
- **14 members** with greater international representation.
- **IAB renamed** to Internet Architecture Board.



June 1992: IAB and the New Internet Protocol!

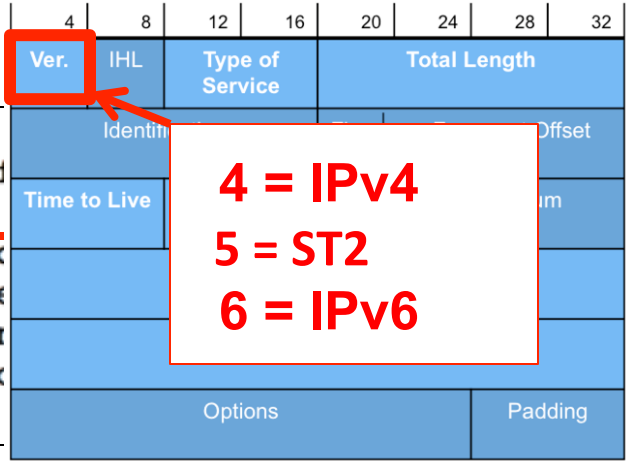
IPv7

- **June 1992, IAB meeting in Kobe, Japan**
- **ROAD subgroup** (1991 retreat) and IESG both recommended to "aggressively pursue" a new version of IP
- **IAB**, **without referring to an IETF** working group (as normal) uncustomary proposed a top-down edict, **proposing** a specific **protocol to replace IPv4**:
  - **OSI's CLNP** (ConnectionLess Network Protocol)
  - CLNP-based proposal called **TCP and UDP with Bigger Addresses (TUBA)**
  - **Replace IP with CLNP** and its variable length address of 20 bytes (160 bits)
  - CLNP already existed – **OSI protocol suite**
  - **Use as much of the proven** TCP/IP suite but over CLNP (not IP)
  - **CLNP** dubbed **IP Version 7** (Kind of blew it and erroneously skipped version 6. *IPv5 had been taken by ST, an experimental streaming protocol over IP.*

RFC 1190

What About IPv5?

The ST packet header is not constrained to be compatible
IP packet header, except for the IP Version Number (the
bits) that is used to distinguish ST packets (IP Version
IP packets (IP Version 4). The ST packets, or protocol
(PDUs), can be encapsulated in IP either to provide conne
(possibly with degraded service) across portions of an i
that do not provide support for ST, or to allow access t
such as security that are not provided directly by ST.



- In the late 1970s, a family of experimental protocols was developed intended to provide quality of service (QoS) for real-time multimedia applications such video and voice.
- Known as Internet Stream Protocol (ST) and later ST2 – (RFC 1190 and RFC 1819).
- Although it was never known as IPv5, when encapsulated in IP, ST uses IP Protocol version 5.

IETF and Internet Community Backlash

- **IETF participants expressed outrage over IAB's suggestion to replace IP with ISO's CLNP.**
- Outrage expressed over Internet mailing lists:
  - *"shocked disbelief"*
  - *"fails on both technical and political grounds"*

© Can Stock Photo - csp23950343

  - *"I view this idea of adopting CLNP as IPv7 as a disastrous idea."*
  - *"Adopting CLNP means buying into ISP standards process." (Not IETF, the current standards committee)*
  - *"As such, we have to face the painful reality that any future changes that Internet community wishes to see in the network layer will require ISO approval too."*

  - *"Do you want to see the political equation? IPv7 = DECNET Phase 5"*

  - *"For decisions this big, I'm shocked to see that IAB made the move without holding an open hearing period for opinions from the Internet community." (Which was the norm.)*

  - Etc. etc. etc.

IETF and Internet Community Backlash

The IETF Standards Process

The basic definition of the IETF standards process is in RFC 2026 (BCP 9). However, this document has been amended several times. The intellectual property rules are now separate, in RFC 5378 (BCP 78) (rights in contributions) and RFC 3979 (BCP 79) (rights in technology). Another update is RFC 3932 (BCP 92) (independent submissions to the RFC Editor). An overview of many process documents is available in The IETF Process: An Informal Guide.

From RFC 2026, section 1.2:

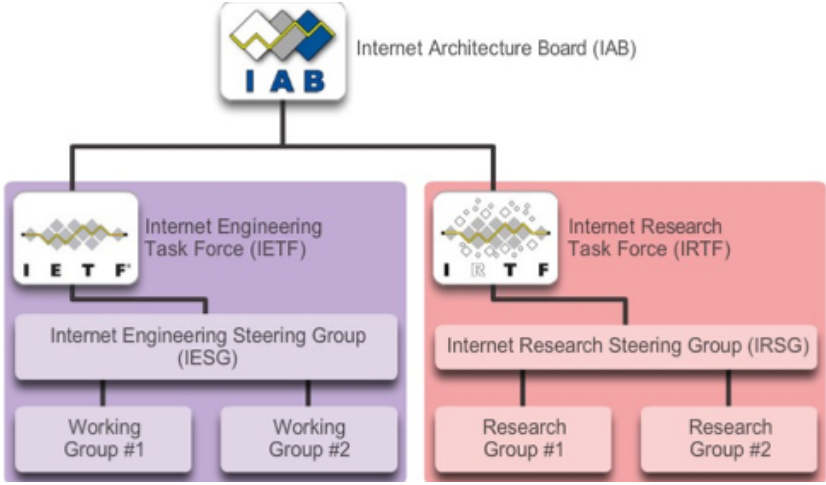
> In outline, the process of creating an Internet Standard is straightforward: a specification undergoes a period of development and several iterations of review by the Internet community and revision based upon experience, is adopted as a Standard by the appropriate body... and is published. In practice, the process is more complicated, due to (1) the difficulty of creating specifications of high technical quality; (2) the need to consider the interests of all of the affected parties; (3) the importance of establishing widespread community consensus; and (4) the difficulty of evaluating the utility of a particular specification for the Internet community.
>
> The goals of the Internet Standards Process are:
>
> - technical excellence;
> - prior implementation and testing;
> - clear, concise, and easily understood documentation;
> - openness and fairness; and
> - timeliness.
>
> ... The goal of technical competence, the requirement for prior implementation and testing, and the need to allow all interested parties to comment all require significant time and effort. On the other hand, today's rapid development of networking technology demands timely development of standards. The Internet Standards Process is intended to balance these conflicting goals. The process is believed to be as short and simple as possible without sacrificing technical excellence, thorough testing before adoption of a standard, or openness and fairness.

- **Greatest concern** related to **competition between IETF and ISO** standards bodies.
- Who would have power and control over the standards?
- **IETF feared** if they would have "change control" and would be subjected to **ISO's** lengthy, top-down, and complex standards development process.
- **IESG recommended** that the community examine other alternatives.

July 1992: IETF's 24th Annual Meeting at MIT

- **MIT's David Clark's, plenary session** addressed:
  1. New Internet services such as real-time video
  2. Emerging services such as ATM, SMDS and B-ISDN
  3. Cyber-terrorism
  4. ***"US: We have met the enemy and he is …."***
- **Clark**:
  - **IAB** is "sort of like the House of Lords"
  - "We reject kings, presidents, and voting."
  - We believe in: "rough consensus and running code."
  - The standards community (**IETF**) had traditionally succeeded by adopting working, tested code rather than proposing top-down standards and making them work.
- The **message was clear**…. **REJECT** IAB's mandate for this new protocol.
- The **conference proposed two alternatives**, PIP (P Internet Protocol) and IPAE (IP Address Encapsulation).

IAB Gets the Message



- **IAB was "fried"** and they got the message.
- **IAB formally withdrew its draft proposal** at the IETF conference.
- **IAB concluded** with IETF would continue pursuing alternative proposals, IPng, IP next generation.
- Also, changes made to selection process for IAB and IESG, not just appointed.
- Talk **IPng** dominated IETF mailing lists

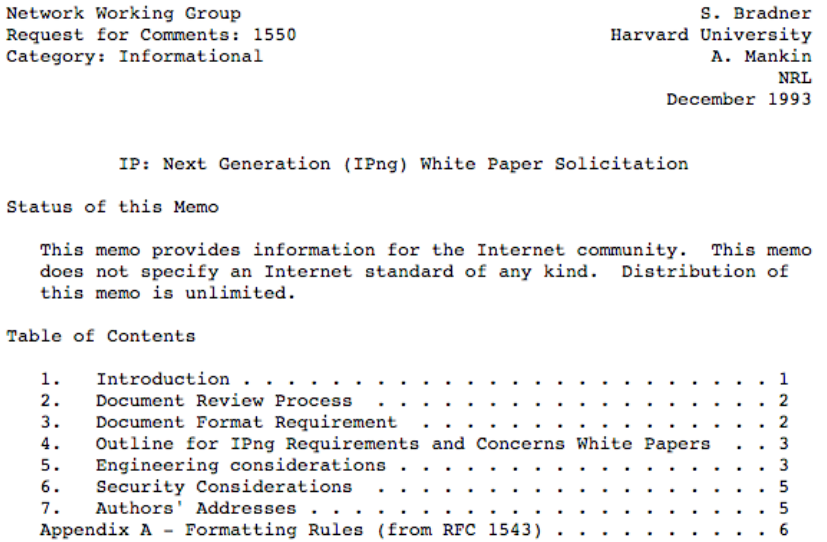
IETF and IPng

- **November, 1992: IETF meeting**
  - Discussed four proposals including TUBA.
- **July 1993: IETF meeting held in Amsterdam**, first ever outside of North America
  - **Formed an IPng Decision BOF** (Birds Of a Feather) to discuss the decision process for IPng
  - **IETF would decide**, not the market
- **IESG** created an ad hoc working group to select the IPng
- **Tension and conflict still existed** between ISO and IETF
- **Some suggested that TUBA faced** 'Not Invented Here' prejudice
- Some feared potential **legal difficulties if IETF protocol proposals were eliminated on arbitrary grounds.**
- **Six page solicitation invited interested parties** to submit documents detailing requirements for IPng.

Corporate Response

- **Corporate response:**
- **Boeing Corporation's** response:
  - "Large corporate users view IPng with disfavor."
  - "… a threat rather than an opportunity."
- **Early 1990's**, US corporations had a mixed protocol environment.
- **Boeing** used at least 16 distinct sets of protocols, which was typical.
- **Trend** was to reduce this number.
- **IBM**, "IPv4 users won't upgrade to IPng without a compelling reason."
- **Cable companies** envisioned opportunities to become providers of converged services, TV, Internet, VoD, phone.

Call for White Papers



Network Working Group
Request for Comments: 1550
Category: Informational

S. Bradner
Harvard University
A. Mankin
NRL
December 1993

IP: Next Generation (IPng) White Paper Solicitation

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Table of Contents

- **1993**, IETF announced a call for white papers with RFC 1550 *IP: Next Generation (IPng) White Paper Solicitation*.
- **Selection committee not diverse**:
  - Software companies (Sun, Microsoft, Novell, Lotus)
  - Hardware vendors (Cisco, IBM, Wellfleet, DEC, Xerox PARC)
  - Service providers (AT&T, NEARNET, NTT, Ameritech)
  - David Clark, MIT
  - Corporations – Mostly US-based
  - One person representing corporate Internet users

Finalists

	CATNIP (Combo)	SIPP (IP)	TUBA (ISO)
Formal name	Common Architecture for the Internet	Simple Internet Protocol Plus	TCP/UDP with Bigger Addresses
Working Group Chairs	Vladimir Sukonnik	Steve Deering, Paul Francis, Robert Hinden, Dave Crocker, Christian Huitema	Mark Knopper, Peter Ford
Protocol approach	New network protocol integrating IP, OSI, and Novell	Evolution of IPv4	Replace IPv4 with OSI CLNP
Address Format	160-bit address	64-bit address	160-bit address

- IPng Directorate considered CATNIP not adequately specified.
- Criteria specified in RFC 1726

Finalists

	CATNIP (Combo)	SIPP (IP)	TUBA (ISO)
Formal name	Common Architecture for the Internet	Simple Internet Protocol Plus	TCP/UDP with Bigger Addresses
Working Group Chairs	Vladimir Sukonnik	Steve Deering, Paul Francis, Robert Hinden, Dave Crocker, Christian Huitema	Mark Knopper, Peter Ford
Protocol approach	New network protocol integrating IP, OSI, and Novell	Evolution of IPv4	Replace IPv4 with OSI CLNP
Address Format	160-bit address	64-bit address	160-bit address

- Still issues on who controls the standards, IETF or ISO.
- CATNIP proposal included, *"The argument that the IETF need not (or should not) follow existing ISO standards will not hold. The ISO is the legal standards organization for the planet. Every other industry develops and follows ISO standards. ISO convergence is both necessary and sufficient to gain international acceptance and deployment of IPng."*

Announcement of IPv6

- **30th IETF meeting in Toronto, Canada**, July 1994, directorate **announced their recommendation that SIPP, with modifications** (128-bit address), would become the basis for **IPng**.
- **IETF** would preserve control of the new standard.
- **IANA** formally announced that the new version would be **version 6**, since 5 was taken by ST.

RFC 1752: The Recommendation for IPng

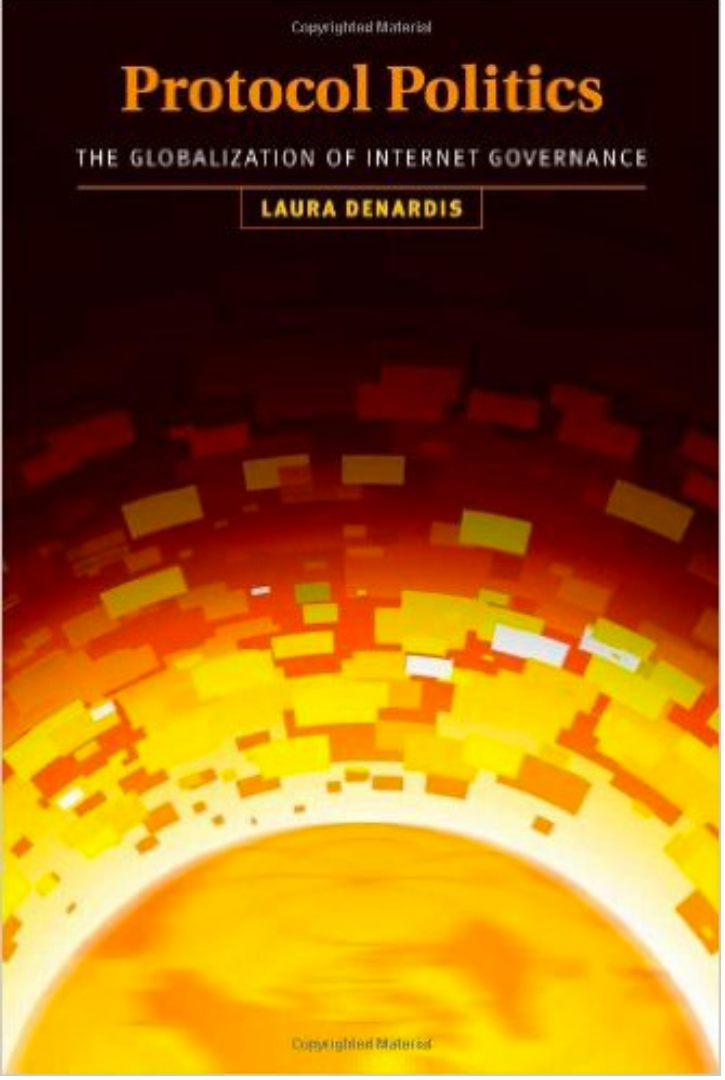
Did the reviewers feel the proposal met the specific criterion?

	CATNIP	SIPP	TUBA
complete spec	no	yes	mostly
simplicity	no	no	no
scale	yes	yes	yes
topological flex	yes	yes	yes
performance	mixed	mixed	mixed
robust service	mixed	mixed	yes
transition	mixed	no	mixed
media indepdnt	yes	yes	yes
datagram	yes	yes	yes
config. ease	unknown	mixed	mixed
security	unknown	mixed	mixed
unique names	mixed	mixed	mixed
access to stds	yes	yes	mixed
multicast	unknown	yes	mixed
extensibility	unknown	mixed	mixed
service classes	unknown	yes	mixed
mobility	unknown	mixed	mixed
control proto	unknown	yes	mixed
tunneling	unknown	yes	mixed

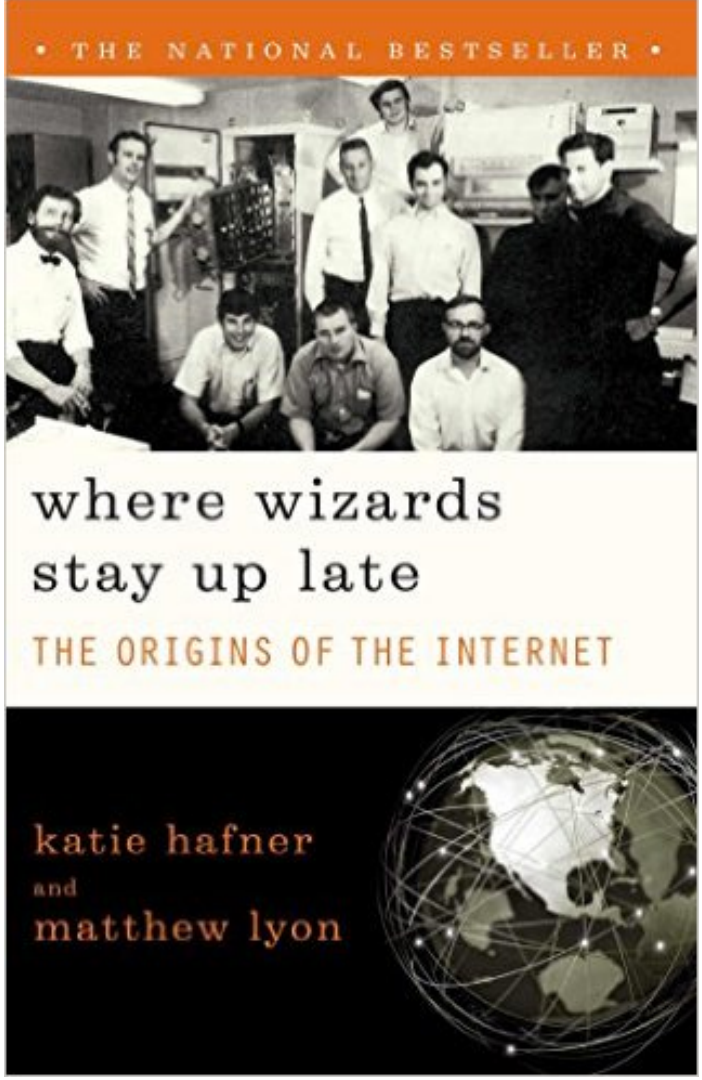
IPv6

- **If a different protocol had been selected**, control of the Internet's technical direction could have shifted from IETF to ISO.
- **At the time**, the majority of the people involved in the IETF were American.
- **In contrast**, ISO was a more international organization.
- **IAB and IETF** became more international in the later years.
- **Regional Internet Registries (RIRs) were created in 1992** to allocate IP addresses at a regional level.
- **Examining IPv6 against other alternatives** demonstrated both institutional tensions (IETF, ISO, IAB) and conflicts among dominant vendors like DEC versus newer entrants like SUN Microsystems; grassroots rank-and-file (IETF) versus newer institutional formations like ISOC.
- *"Despite the Internet standards community's strategy of eliminating the influence of sociological factors on its architectural decisions, the history of IPv6 indicates that the definition of the Internet, ultimately, includes people."*

  Laura Denardis, Protocol Politics, The Globalization of Internet Governance



- Material from:
- Protocol Politics: The Globalization of Internet Governance (Information Revolution and Global Politics) by Laura DeNardis

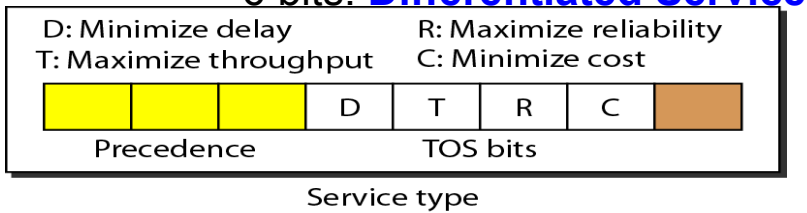


- Where Wizards Stay Up Late: The Origins Of The Internet First Paperback Edition Edition
- by Katie Hafner

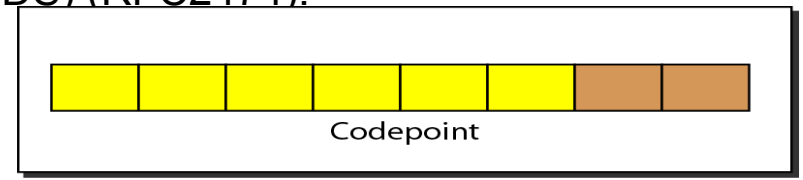
And we're back!

IP Header Fields

- **Version** (4 bits): current version is 4.
- **Header length** (4 bits):
  - Length of IP header, in multiples of 4 bytes
  - 20 bytes ≤ Header Length ≤ $(2^4-1) * 4 = 60$ bytes
- **Service field** (1 byte)
  - If first three bits are 0, interpreted as original **Type-of-Service** (TOS).
  - Otherwise
    - 6 bits: **Differentiated Service** (DS) (RFC2474):



Service type



Differentiated services

IP Header Fields

- **Total length** (16 bits):
  - Total length of IPv4 datagram, in bytes.
  - 20 bytes ≤ Total Length ≤ 2^{16} - 1 = 65535 bytes
  - Length of data = total length - header length
- **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)

  For **Fragmentation**… will be explained later

IP Header Fields

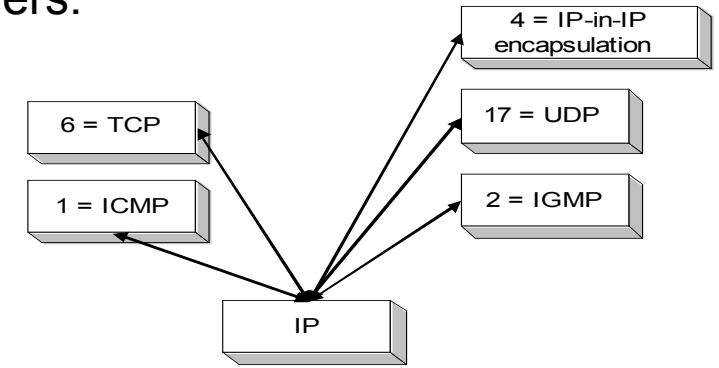
- **Time To Live** (TTL) (1 byte):
  - Specifies longest paths before datagram is dropped
  - Role of TTL field: Ensure that packet is eventually dropped when a routing loop occurs

  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

- ***No TTL in Ethernet… why do you think they didn't include one?***

IP Header Fields

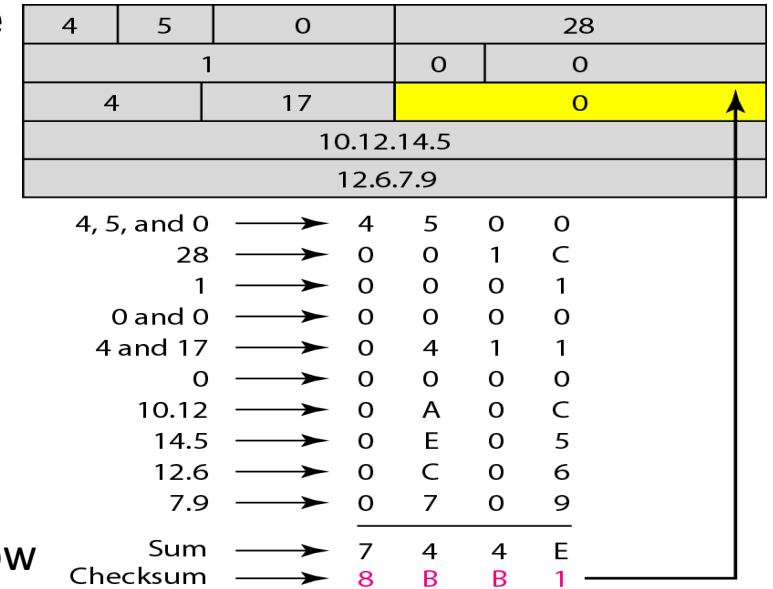
- **Protocol** (1 byte):
  - Specifies the higher-layer protocol.
  - Used for demultiplexing to higher layers.



IP Header Fields

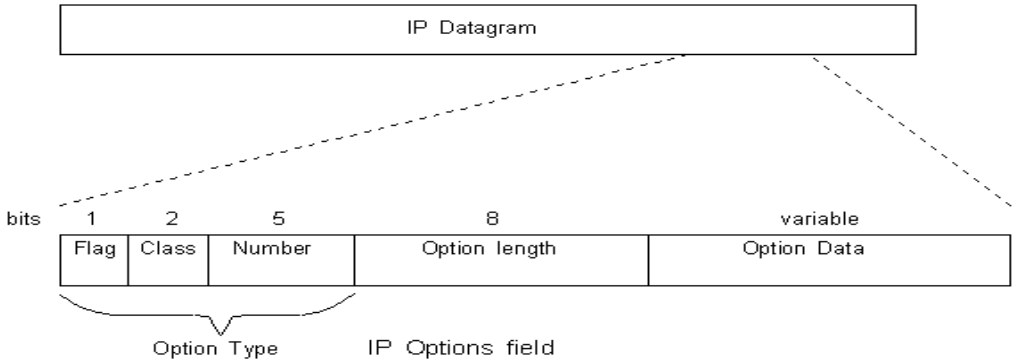
- **Header checksum** (2 bytes): Simple 16-bit long checksum covers only header.
- Upper layer protocols cover data
- IP is highest hop-by-hop protocol; need to minimize processing

How Checksum is calculated:
https://www.youtube.com/watch?v=dXartoyj2ow



IP Header Fields

- **Option Type**
  - Copy flag: Indicates if option to be copied to fragments.
  - Option class: 0 = Control, 2 = Debug/Measurement, rest “Reserved”.
  - Option number: identifies option
- **Option length**: not present for Noop and End of Options
- **Option da**



Fragmentation

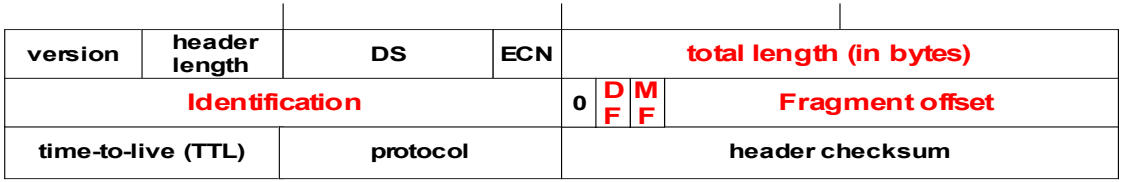
- Maximum size of IP datagram is 65535…
  - …but link-layer payload limits typically much smaller
- Called the **Maximum Transmission Unit** (MTU).
- Example MTUs:

Ethernet:	1500	FDDI:	4352
802.3:	1492	ATM AAl5:	9180
802.5:	4464	PPP:	Negotiated

- **Fragment** IP datagrams larger than MTU of a link.
- Issues
  - How communicate fragmentation among hops in a path?
  - How handle paths containing networks with different MTUs?
  - Where is fragmentation done?

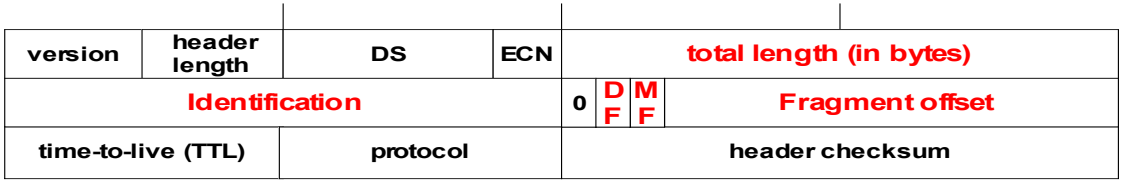
Fragmentation - How?

- Involves following fields (plus checksum)



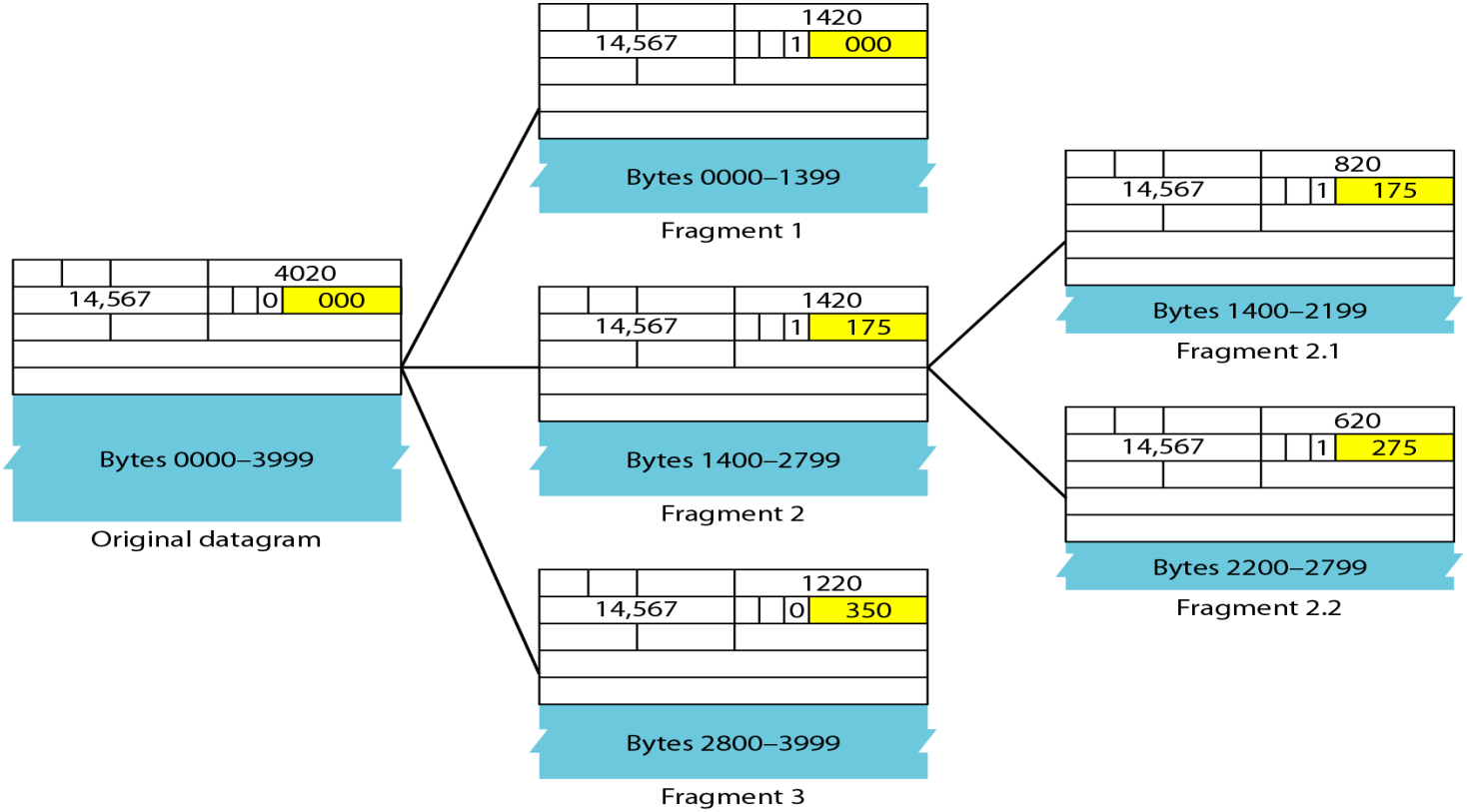
- **Identification**: ID and source IP address uniquely identify datagram.
- **Flags**
  - **DF**: “Don’t fragment.” Discard and send error.
  - **MF**: “More fragments.” More fragments follow.
- **Fragment Offset**: Offset of current payload in original datagram.
  - Only 13 bit field - gives offset in units of 8 bytes
  - Number of first byte in payload is FO * 8.
  - Size of all fragments, but last, must be multiple of 8.

Fragmentation - How?



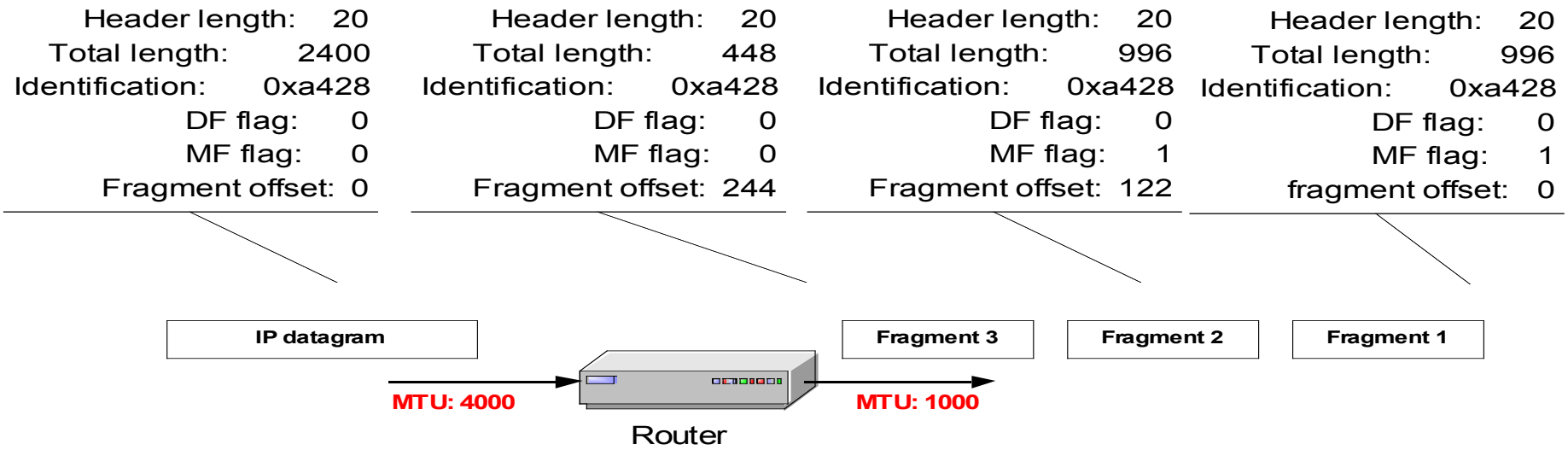
- **Total length**: Total length of the current fragment.
- Constraints of fragmentation
  - Fragmentation can be done at the sender or at intermediate routers
  - The same datagram can be fragmented several times.
  - Reassembly of original datagram is only done at destination hosts!!
    - ***Why?***
- **How determine if a datagram is a fragment?**
  - FO ≠ 0 or…
  - MF flag is set
    - Explanation
      - What does FO = 0 mean? *This is the first* fragment.
      - What does MF = 0 mean? *This is the last* fragment.
      - What does ((FO = 0) **and** (MF = 0)) mean? *This is both the first and last fragment -> this is the only fragment -> this is the original packet (i.e. not a fragment). NOTE: this is the only way to not be a fragment.*
      - So a packet is a fragment if !((FO = 0) **and** (MF = 0)) <-> ((FO ≠ 0) **or** (MF ≠ 0))

Fragmentation Example



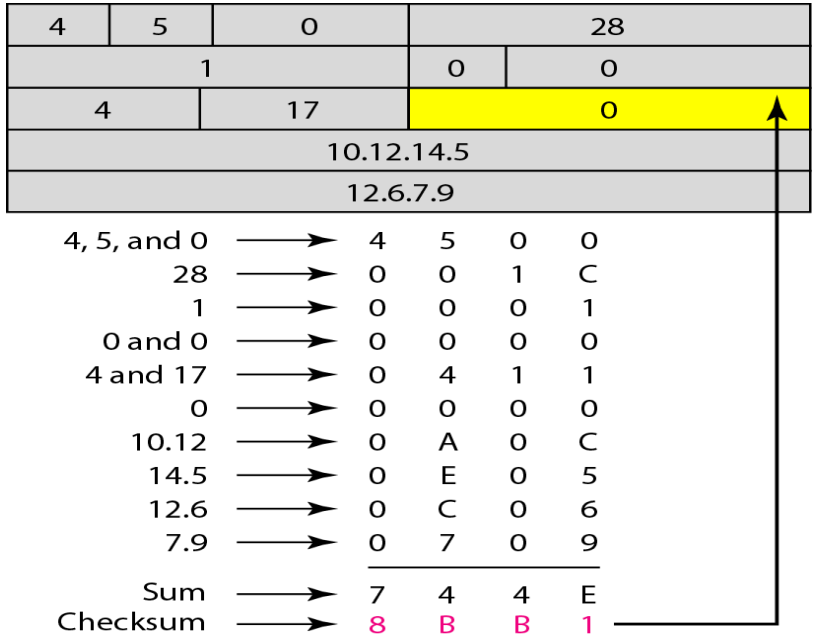
Fragmentation Example

- Example where last fragment is not multiple of 8.



Why calculate checksum @ each hop?

- Fields will change…
  - TTL
  - Fragmentation information
  - Header length
  - Others..?



Longest Prefix Match

- Given forwarding table
  - (1) 128.114.48.0/17
  - (2) 128.114.48.0/20
  - (3) 128.114.48.0/22
  - (4) 0/0
- Which entry is chosen for destination: 128.114.122.5
- Answer: #1
- Explanation (remember bit values: 128 64 32 16 8 4 2 1)
  - `10000000 01110010 0xxxxxxx xxxxxxxx` = 128.114.48.0/17 (0.0 – 127.255)
  - `10000000 01110010 0011xxxx xxxxxxxx` = 128.114.48.0/20 (48.0 – 63.255)
  - `10000000 01110010 001100xx xxxxxxxx` = 128.114.48.0/22 (48.0 – 51.255)
  - `10000000 01110010 01111010 00000101` = 128.114.122.5

Longest Prefix Match

- Given forwarding table
  - (1) 128.114.48.0/17
  - (2) 128.114.48.0/20
  - (3) 128.114.48.0/22
  - (4) 0/0
- Which entry is chosen for destination: 128.114.50.2
- Answer: **??**
- Explanation (remember bit values: 128 64 32 16 8 4 2 1)
  - `10000000 01110010 0xxxxxxx xxxxxxxx` = 128.114.48.0/17 (0.0 – 127.255)
  - `10000000 01110010 0011xxxx xxxxxxxx` = 128.114.48.0/20 (48.0 – 63.255)
  - `10000000 01110010 001100xx xxxxxxxx` = 128.114.48.0/22 (48.0 – 51.255)

Longest Prefix Match

- Given forwarding table
  - (1) 128.114.48.0/17
  - (2) 128.114.48.0/20
  - (3) 128.114.48.0/22
  - (4) 0/0
- Which entry is chosen for destination: 128.114.50.2
- Answer: #3
- Explanation (remember bit values: 128 64 32 16 8 4 2 1)
  - `10000000 01110010 0xxxxxxx xxxxxxxx` = 128.114.48.0/17 (0.0 – 127.255)
  - `10000000 01110010 0011xxxx xxxxxxxx` = 128.114.48.0/20 (48.0 – 63.255)
  - `10000000 01110010 001100xx xxxxxxxx` = 128.114.48.0/22 (48.0 – 51.255)
  - `10000000 01110010 00110010 00000000` = 128.114.50.2

Longest Prefix Match

- Given forwarding table
  - (1) 128.114.48.0/17
  - (2) 128.114.48.0/20
  - (3) 128.114.48.0/22
  - (4) 0/0
- Which entry is chosen for destination: 128.114.52.10
- Answer: #2
- Explanation (remember bit values: 128 64 32 16 8 4 2 1)
  - `10000000 01110010 0xxxxxxx xxxxxxxx` = 128.114.48.0/17 (0.0 – 127.255)
  - `10000000 01110010 0011xxxx xxxxxxxx` = 128.114.48.0/20 (48.0 – 63.255)
  - `10000000 01110010 001100xx xxxxxxxx` = 128.114.48.0/22 (48.0 – 51.255)
  - `10000000 01110010 00110100 00001010` = 128.114.52.10
- *Give an example of an address that would use (4)… the default route.*