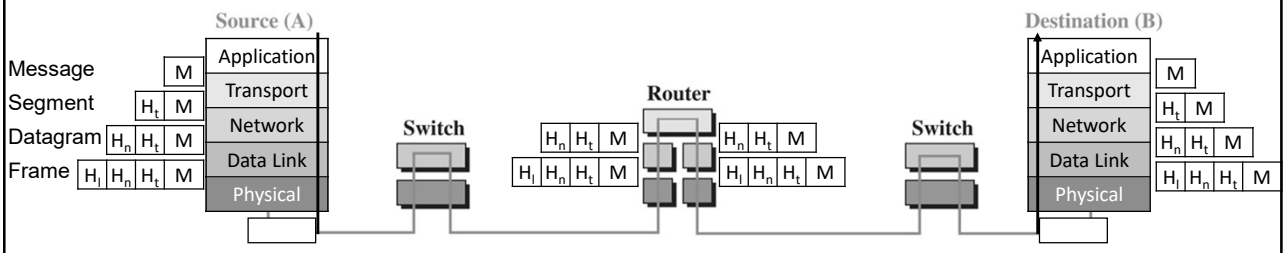
Welcome!

ELEC 8560 – Computer Networks

Network Layer: Data Plane

1

Recall: End-to-End Communication via Internet



2

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
- Sidebar: Network Neutrality

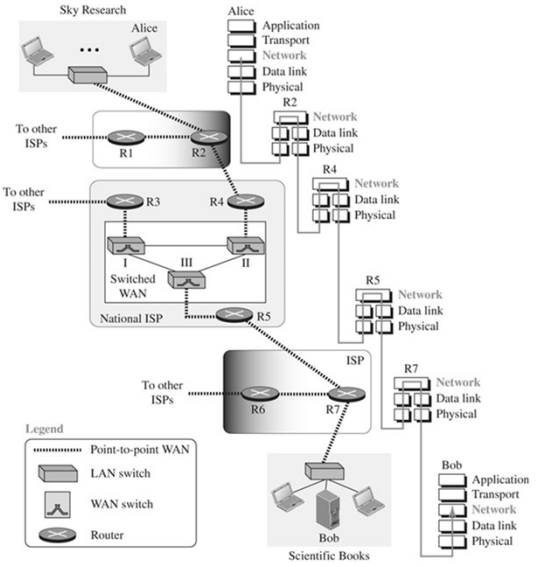
- Recommended reading: Forouzan – Chapter 7

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
- Sidebar: Network Neutrality

Communication at the Network Layer

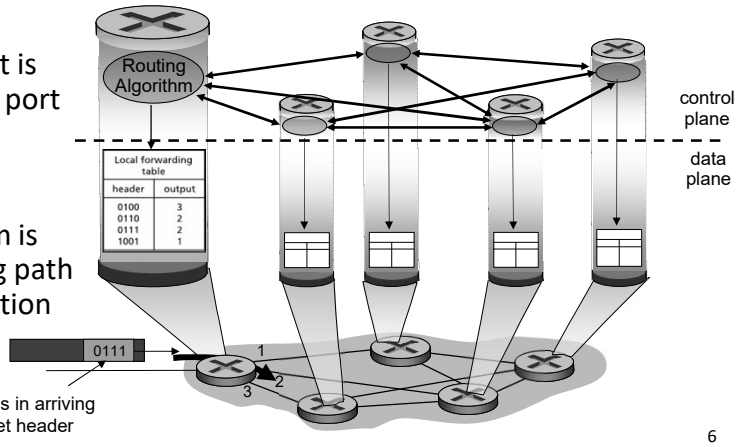
- Network layer provides a logical connection between hosts
  - Transport segments from source to destination
  - Sender: encapsulates segment from transport layer into datagrams, passes to data link layer (called packetization)
  - Receiver: reassembles datagrams, delivers segments to transport layer



Data Flow Diagram

Network Layer: Data Plane and Control Plane

- Network layer has two key functions: forwarding and routing
- Data plane
  - Local, per-router function
  - Determines how datagram arriving on router input port is forwarded to router output port
- Control plane
  - Network-wide logic
  - Determines how a datagram is routed among routers along path from source host to destination host (routing algorithms)



Outline

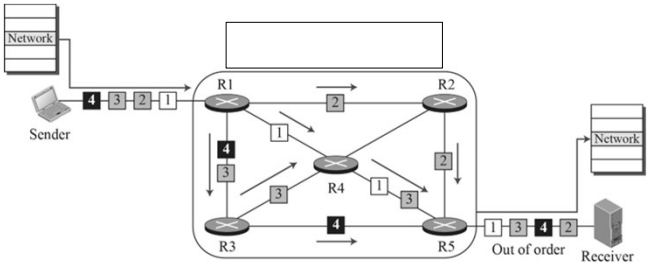
- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
- Sidebar: Network Neutrality

Packet Switching

- Routers
  - Examine header fields in all IP datagrams passing through it
  - Moves datagrams from input ports to output ports to transfer datagrams along end-to-end path
- A router creates a connection between an input port and an output port (or a set of output ports)
- Two approaches for packet switching:
  - Datagram approach (connectionless service)
  - Virtual-circuit approach (connection-oriented service)

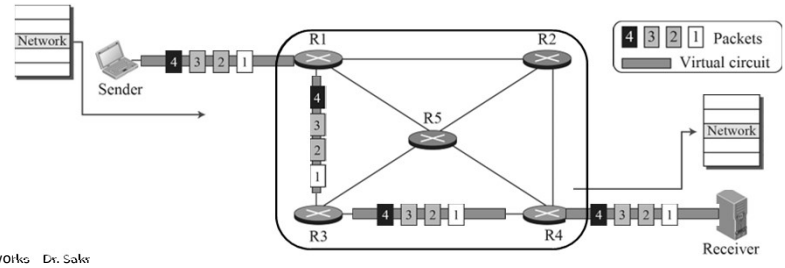
Connectionless Packet-Switched Network

- Datagram approach
  - Earlier when Internet started, to make it simple, network layer was designed to provide a connectionless service (treats each packet independently, with no relationship to any other packet)
  - Network layer is only responsible for delivery of packets from source to destination
  - Packets in a message may or may not travel the same path to their destination



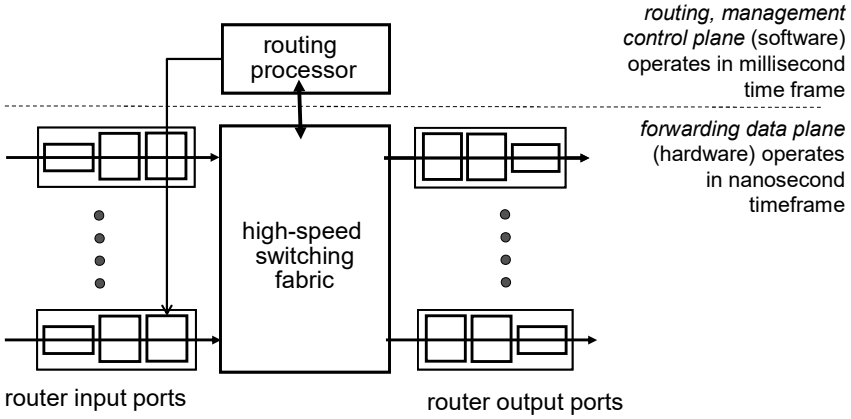
Connection-Oriented Packet-Switched Network

- Virtual-circuit approach
  - Service considers the relationship between all packets belonging to a message
  - Before all datagrams in a message are sent, a virtual connection should be set up to define the path for the datagrams
  - After connection setup, datagrams can all follow the same path
  - Packet must contain: (i) source and destination addresses, and (ii) a flow label, a virtual circuit identifier that defines the virtual path packet should follow

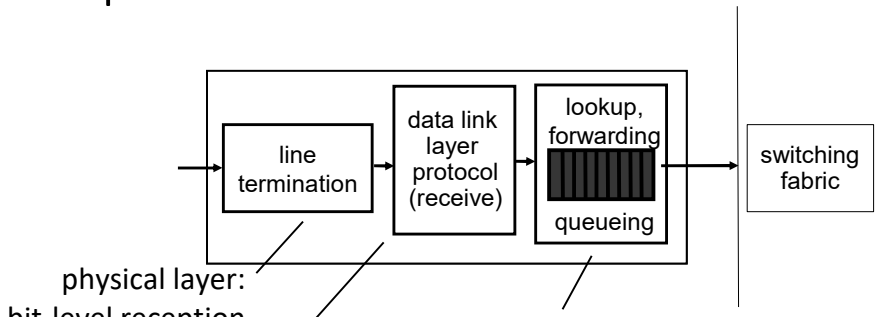


What is Inside a Router?

- High-level view of generic router architecture:
  - Input and output ports
  - Switching fabric: transfer packet from input link to appropriate output link



Input Port Functions



physical layer: bit-level reception

data link layer: e.g., Ethernet

Decentralized switching*:*

- Using header field values, lookup output port using forwarding table in input port memory
- Need to complete input port processing at 'line speed'
- Forward based on header field values
- Input port queuing: if datagrams arrive faster than forwarding rate into switch fabric

Performance of Network Layer

- Performance of a network can be measured in terms of
  - Bandwidth: How much data could theoretically be transferred (bps)
  - Throughput: How much data was actually transferred (bps)
  - Latency (or Delay): How long it takes for an entire message to arrive
    - Four types: processing, queueing, transmission, and propagation
  - Packet loss: When a queue is full, next packets dropped
    - For example, arrival rate > transmission rate
  - Bandwidth-delay product
  - Jitter
  - and more ..

Outline

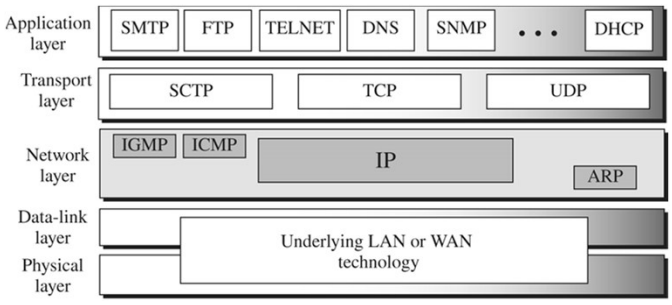
- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
- Sidebar: Network Neutrality

Network Layer in the Internet

- Network layer in the Internet has gone through several versions, but only two versions have survived:
  - Internet Protocol Version 4 (IPv4)
  - Internet Protocol Version 6 (IPv6)

Four Related Protocols

- Network layer can be thought of as one main protocol and three auxiliary ones
  - IP: main protocol, responsible for packetizing, forwarding, and delivery of a packet
  - ICMP: helps IP to handle some errors that may occur in delivery
  - IGMP: helps IP in multicasting
  - ARP: used in address mapping

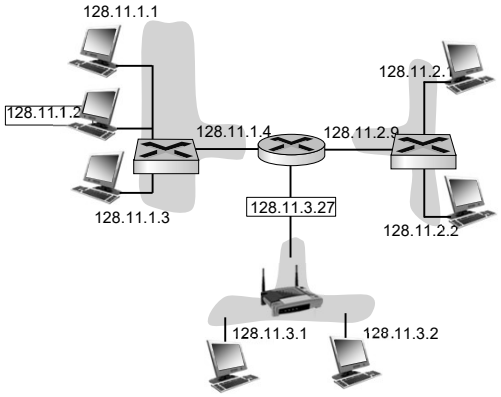


Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - IPv4
- Sidebar: Network Neutrality

IPv4 Addressing

- IP address:
  - 32-bit identifier used in the IP layer of the TCP/IP protocol suite
  - Identify the connection of each device (interface) to the Internet
    - A router typically has multiple interfaces
    - A Host typically has one or two interfaces (e.g., wired Ethernet, Wi-Fi 802.11)
  - IP address is the address of the connection, not the host or the router
  - If the device is moved to another network, the IP address may be changed



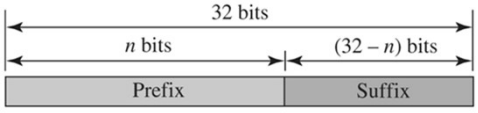
Notation	IP Address
Dotted-decimal	128.11.3.31
Binary	10000000 00001011 00000011 00011111
Hexadecimal	800B031F

Address Space

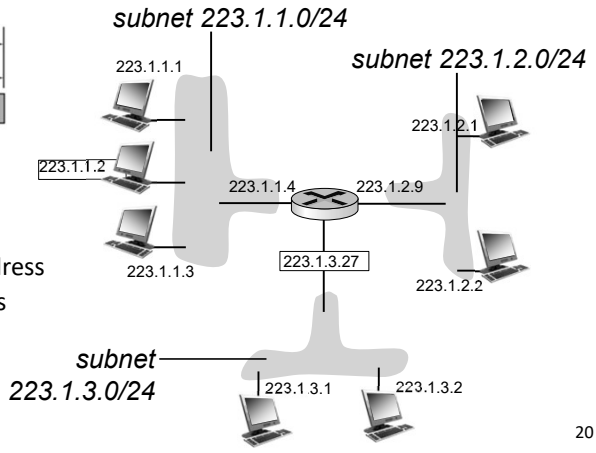
- Address space is the total number of addresses used by the protocol
- IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than four billion)
- If there were no restrictions, more than 4 billion devices could be connected to the Internet

Subnets

- IPv4 divided into two parts (i.e., hierarchy):
  - Prefix, high order bits to define the network (subnet)
  - Suffix, low order bits to define the connection to node (host)



- Example:
  - Subnet mask: /24
    - High order 24 bits: subnet part of IP address
    - Low order 8 bits: node part of IP address
    - Each isolated network is called a subnet

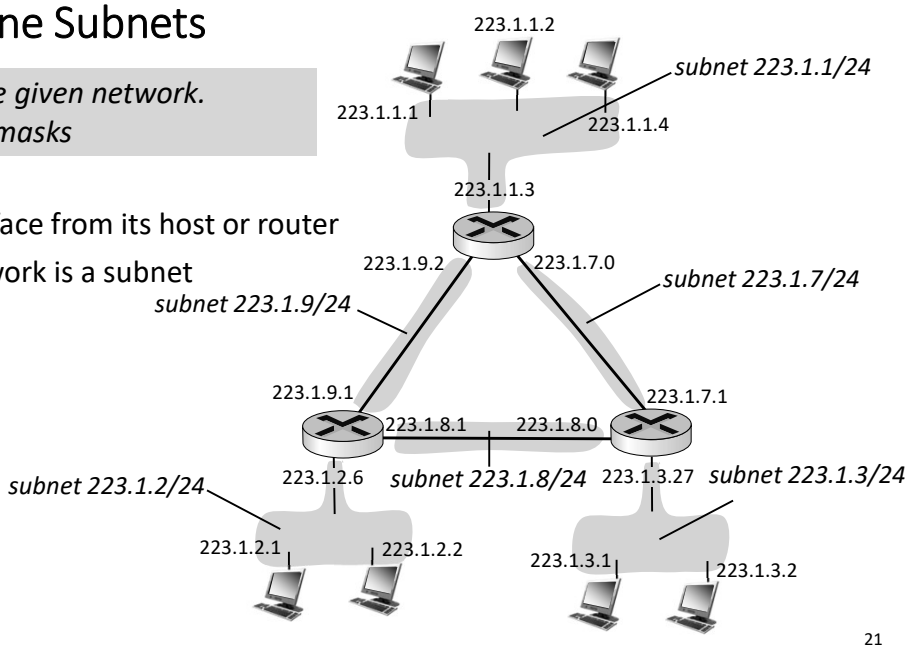


Example: Define Subnets

List all subnets in the given network.
Assume /24 subnet masks

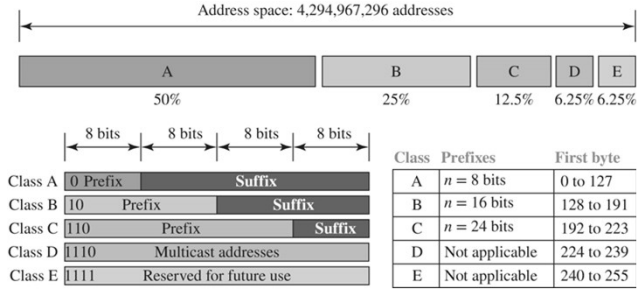
Solution:

- Detach each interface from its host or router
- Each isolated network is a subnet



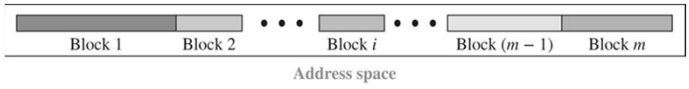
Subnet Addressing

- Classful Addressing:
  - When the Internet started, IPv4 address space was divided into five classes with fixed-length prefix



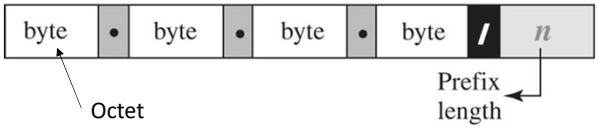
Class	Prefixes	First byte
A	$n = 8$ bits	0 to 127
B	$n = 16$ bits	128 to 191
C	$n = 24$ bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

- Classless Addressing:
  - With the growth of the Internet, larger address space was needed
  - Use variable-length blocks that belong to no class (instead of only 5 classes)



Slash Notation (CIDR) for Classless Addressing

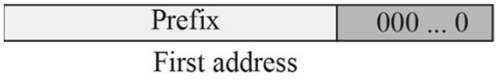
- CIDR: Classless Inter-Domain Routing (pronounced “cider”)
  - Subnet portion of address of variable length
  - Address format: a.b.c.d/n, where n is number of bits in subnet portion of address



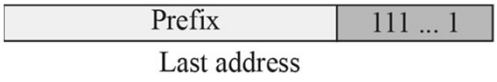
  - Examples:
    - 12.24.76.5/8
    - 23.14.67.92/12
    - 220.8.24.255/25

Slash Notation (CIDR) for Classless Addressing (cont.)

- Given a.b.c.d/n:
  - Number of address in the block = 2^{32-n}
  - First address: keep *n* leftmost bits, set (32-*n*) rightmost bits all to 0s



  - Last address: keep *n* leftmost bits, set (32-*n*) rightmost bits all to 1s



- Note:
  - First and last address are not usable (network address and broadcasting)

Example: CIDR

A classless address is given as 167.199.170.82/27. Find the number of addresses in the block, the first address, and the last address.

Solution:

Number of address in the block	$= 2^{32-27} = 32$			
Address: 167.199.170.82/**27**	10100111	11000111	10101010	01010010
First address: 167.199.170.64/**27**	10100111	11000111	10101010	01000000
Last address: 167.199.170.95/**27**	10100111	11000111	10101010	01011111

First address is also called the network address

Last address is also called the broadcast address

Subnet Mask

- Another way for classless addressing is to use the subnet mask
  - A 32-bit number with *n* leftmost bits set to 1s and the rest of (32-*n*) set to 0s
- Example:
  - /27 is equivalent to 11111111 11111111 11111111 1110000 = 255.255.255.224
- Given subnet mask:
  - Number of address in the block = $2^{\text{no. of 0s in subnet mask}}$
  - First address = (Any address in the block) AND (Mask)
  - Last address = (Any address in the block) OR (NOT (Mask))

Example: Subnet Mask

A classless address is given as 167.199.170.82 and subnet mask 255.255.255.224. Find the number of addresses in the block, the first address, and the last address.

Solution:

Mask:	11111111 11111111 11111111 11100000
Number of addresses in the block	= 2^5 = *32* addresses
First address:	= (address) AND (mask) = 10100111 11000111 10101010 01010010 AND 11111111 11111111 11111111 11100000 = 10100111 11000111 10101010 01000000 = 167.199.170.64
Last address:	= (address) OR (NOT mask) = 10100111 11000111 10101010 01010010 OR 00000000 00000000 00000000 00011111 = 10100111 11000111 10101010 01011111 = 167.199.170.95

Example: Note on Classless Addressing

Note that an address cannot define the block it belongs to. For example, show that the address 230.8.24.56 can belong to many blocks.

Solution:

Prefix length:16	→	Block:	230.8.0.0	to	230.8.255.255
Prefix length:20	→	Block:	230.8.16.0	to	230.8.31.255
Prefix length:26	→	Block:	230.8.24.0	to	230.8.24.63
Prefix length:27	→	Block:	230.8.24.32	to	230.8.24.63
Prefix length:29	→	Block:	230.8.24.56	to	230.8.24.63
Prefix length:31	→	Block:	230.8.24.56	to	230.8.24.57

Network Address

- Internet Corporation for Assigned Names and Numbers (ICANN) assigns a large block of addresses to an ISP
- Network gets allocated portion of its provider ISP address space
- ISP can then allocate out its address space for smaller subnets
  - e.g., 1 block, 2 blocks, 4 blocks, etc.

Network Address (cont.)

- Example:
  - ISP's block 11001000 00010111 00010000 00000000 200.23.16.0/20
  - ISP can then allocate out its address space in 1 subnet of 2^{12} hosts, 2 subnets of 2^{11} hosts, 4 subnets of 2^{10} hosts, etc.
  - Assume we need 8 blocks of 2^9 addresses:

Subnet mask	11111111 11111111 11111110 00000000	/23
Organization 0	11001000 00010111 00010000 00000000	200.23.16.0/23
Organization 1	11001000 00010111 00010010 00000000	200.23.18.0/23
Organization 2	11001000 00010111 00010100 00000000	200.23.20.0/23
...	....	....
Organization 7	11001000 00010111 00011110 00000000	200.23.30.0/23

Designing a Subnet

- Number of addresses N_{sub} in any subnetwork must be a power of 2
- Prefix length n_{sub} of a subnetwork = $32 - \log_2 N_{sub}$
- Start by assigning address to larger subnetworks

Example: Designing a Subnet

An organization is granted a block of addresses with the beginning address 14.24.74.0/24. The organization needs to have 3 subblocks of addresses to use in its three subnets: one subblock of 10 addresses, one subblock of 60 addresses, and one subblock of 120 addresses. Design the subblocks.

Solution:

- There are $2^{32-24} = 256$ addresses in this block
  - first address is 14.24.74.0/24; last address is 14.24.74.255/24
- We assign addresses to subblocks, starting with the largest

a. Number of addresses in the largest subblock, which requires 120 addresses, is not a power of 2, we allocate 128 addresses

- subnet mask for this subnet can be found as $n_1 = 32 - \log_2 128 = 25$
- first address in this block is 14.24.74.0/25; the last address is 14.24.74.127/25

Example: Designing a Subnet (cont.)

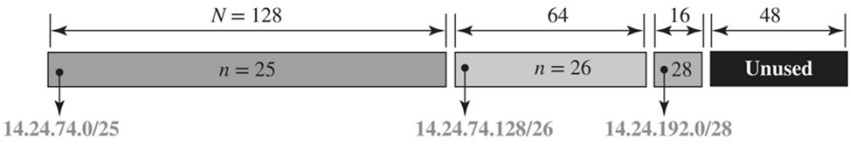
b. Number of addresses in the second largest subblock, which requires 60 addresses, is not a power of 2, we allocate 64 addresses

- subnet mask for this subnet can be found as $n_2 = 32 - \log_2 64 = 26$
- first address in this block is 14.24.74.128/26; last address is 14.24.74.191/26

c. Number of addresses in the smallest subblock, which requires 10 addresses, is not a power of 2, we allocate 16 addresses

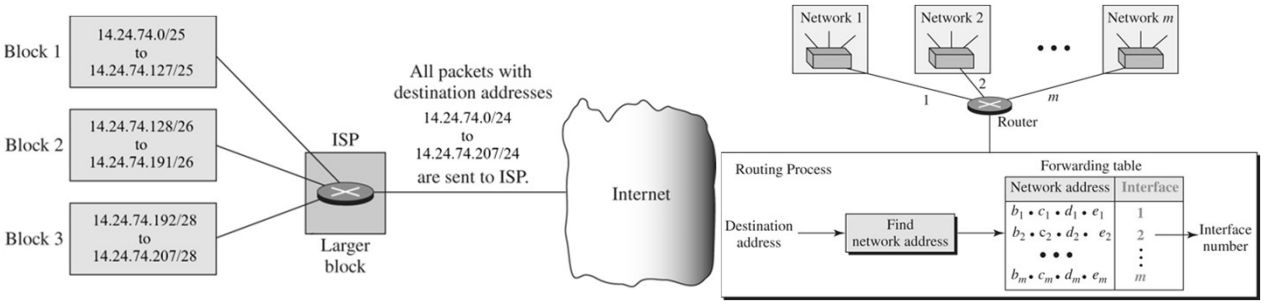
- subnet mask for this subnet can be found as $n_1 = 32 - \log_2 16 = 28$
- first address in this block is 14.24.74.192/28; the last address is 14.24.74.207/28

- We only allocated 208 addresses, which means 48 addresses are left in reserve
  - first address in this range is 14.24.74.208.; last address is 14.24.74.255



Address Aggregation

- After assigning small blocks of addresses to organizations by the ISP:
  - ISP combines subblocks into one single block and advertises the larger block to the rest of the world
  - Any packet destined for this larger block should be sent to this ISP
  - ISP to forward the packet to the appropriate organization



Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - DHCP
- Sidebar: Network Neutrality

How to Get an IP Address?

- After a block of addresses are assigned to an organization, the questions is:
  - How does a host get IP address within its network (host part of address)?
- Automatic Solution:
  - Dynamic Host Configuration Protocol (DHCP) server
- Manual Solution:
  - Network administration can also manually assign addresses to individual hosts or routers

Dynamic Host Configuration Protocol (DHCP)

- DHCP is an application-layer protocol
- When a host joins network, DHCP server automatically assigns IP address (and other communication parameters) to the host using a client-server architecture
  - Host broadcasts DHCP discover msg [optional]
  - DHCP server responds with DHCP offer msg [optional]
  - Host requests IP address: DHCP request msg
  - DHCP server sends address: DHCP ack msg
- Typically, DHCP server will be co-located in router, serving all subnets to which router is attached
  - Allows reuse of addresses (only hold address while host is connected)
  - Support for mobile users who join/leave network

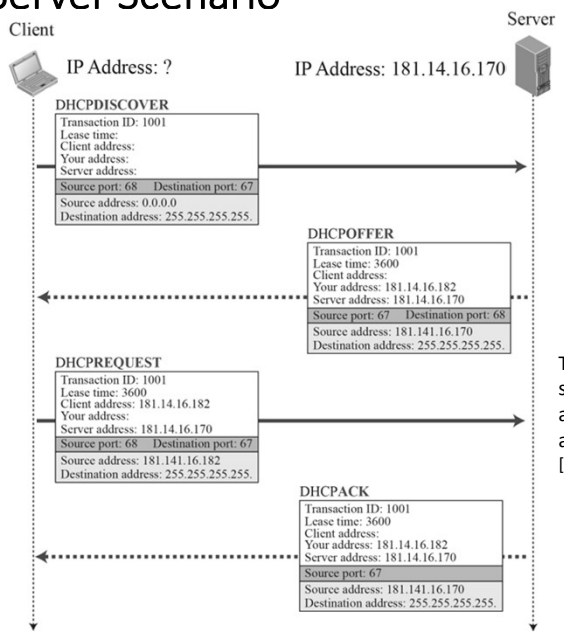
Example: DHCP

Use ipconfig in Windows and show if DHCP is enables.

Solution:

```
C:\Users\admin>ipconfig /all

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : home
   Description . . . . . . . . . . . : Killer(R) Wi-Fi 6 AX1650s 160MHz Wireless Network Adapter
(201D2W)
   Physical Address. . . . . . . . . : 2C-6D-B2-65-36-8C
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv4 Address. . . . . . . . . . . : 192.168.2.24(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Tuesday, August 23, 2022 9:50:17 PM
   Lease Expires . . . . . . . . . . : Tuesday, August 30, 2022 12:43:37 PM
```

DHCP Client-Server Scenario



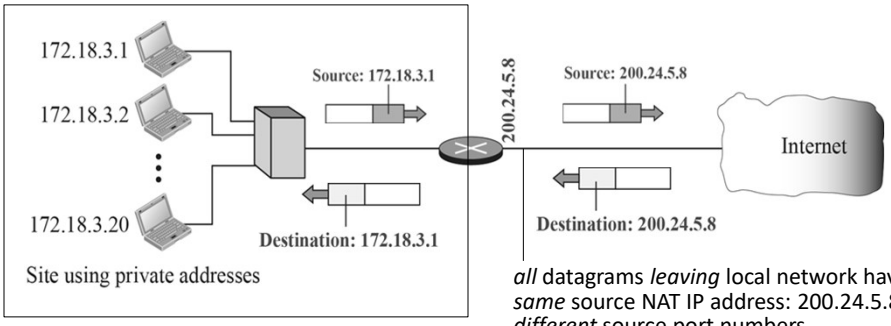
The two steps above can be skipped "if a client remembers and wishes to reuse a previously allocated network address" [RFC 2131]

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - NAT
- Sidebar: Network Neutrality

Network Address Translation (NAT)

- In most situations, only a portion of computers in a small network need access to the Internet simultaneously
- NAT allows all devices in a local network to
  - Share one (or few) IPv4 address for communication with the rest of the world
  - Use a set of private addresses for internal communication



all datagrams *leaving* local network have *same* source NAT IP address: 200.24.5.8, but *different* source port numbers

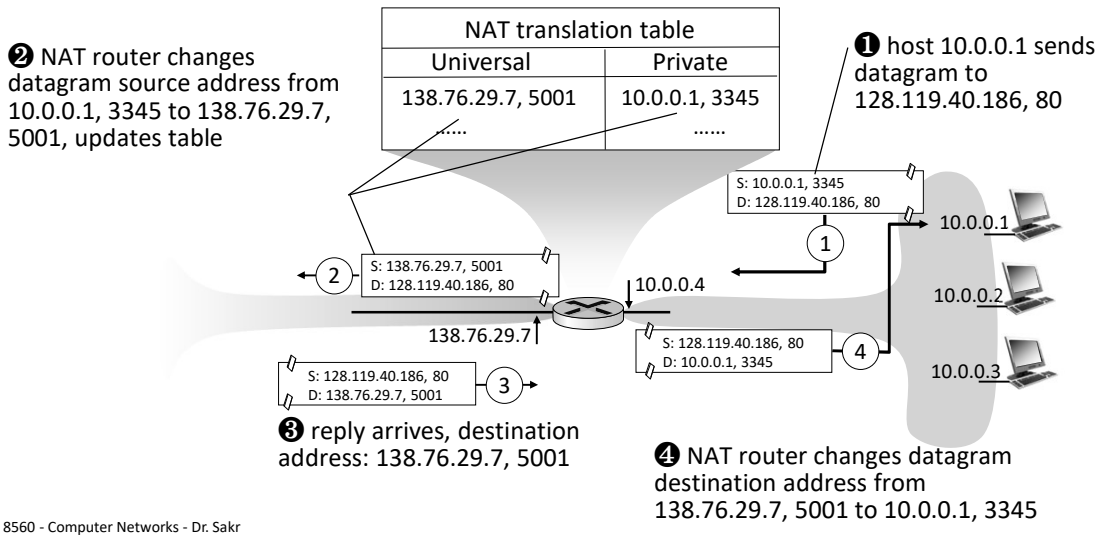
NAT (cont.)

- All devices in local network have 32-bit addresses in a private IP address space (10/8, 172.16/12, 192.168/16 prefixes) that can only be used in local network
- Advantages:
  - Just one IP address needed from provider ISP for all devices
    - Helps ipv4 address space exhaustion
  - Can change addresses of host in local network without notifying outside world
  - Can change ISP without changing addresses of devices in local network
  - Security: devices inside local network not directly addressable or visible by outside world

Address Translation

- NAT router must (transparently):
  - Outgoing datagrams:
    - Replace (source IP address, port no.) of every outgoing datagram to (NAT IP address, new port no.)
    - Remote clients/servers will respond using (NAT IP address, new port no.) as destination address
  - Remember (in NAT translation table) every (source IP address, port no.) to (NAT IP address, new port no.) translation pair
  - Incoming datagrams:
    - Replace (NAT IP address, new port no.) in destination fields of every incoming datagram with corresponding (source IP address, port no.) stored in NAT table

Example: Address Translation

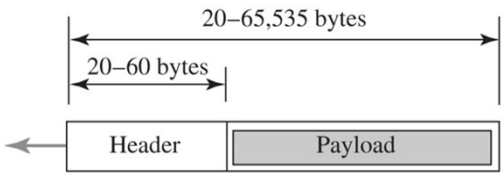


Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - IPv4 datagram format
- Sidebar: Network Neutrality

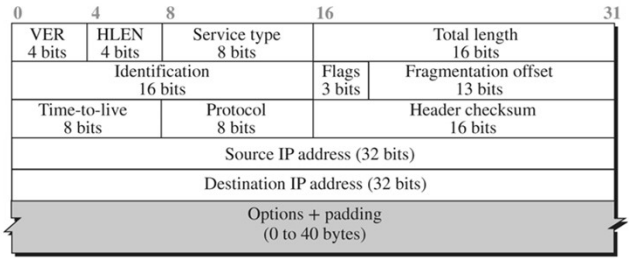
IPv4 Datagram Format

- Packets used by the IP are called datagrams
  - Maximum length is 65,535 bytes
- Variable-length packet with two parts:
  - Header: 20-60 bytes in length
    - 20 bytes contain information essential to routing and delivery
    - Up to 40 bytes for options and padding [optional]
  - Payload (data)



IPv4 Datagram Header

- Customary to be shown in 4-byte sections
  - VER: version number (4 for IPv4)
  - HLEN: header length in 4-byte words (i.e., length in bytes divided by 4)
  - Total length: header + data in bytes (i.e., data length = Total length - HLEN × 4)
  - Identification, flags, fragmentation offset: related to fragmentation (later)
  - Time-to-live: max no. of hops (routers) visited by the datagram, then discard
  - Protocol: protocol using payload
    - e.g., ICMP: 01, UDP: 17, TCP: 06, IGMP:02, OSPF: 89, etc.
  - Checksum: for error detection
  - IP addresses
  - Options: for testing, debugging, etc.



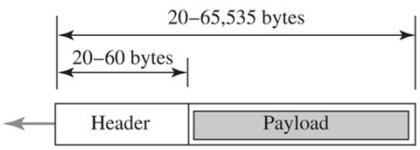
b. Header format

Example: IP Packet

An IPv4 packet has arrived with the first 8 bits as $(01000010)_2$. The receiver discards the packet. Why?

Solution:

- The 4 leftmost bits $(0100)_2$ show the version, which is correct
- The next 4 bits $(0010)_2$ show an invalid header length (2 × 4 = 8 bytes)
  - The minimum number of bytes in the header must be 20
- The packet has been corrupted in transmission

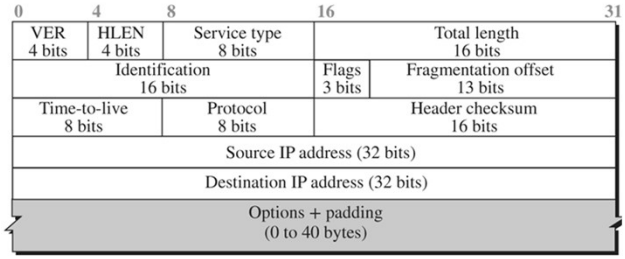


Example: Bytes of Option

In an IPv4 packet, the value of HLEN is $(1000)_2$. How many bytes of options are being carried by this packet?

Solution:

- The HLEN value is 8, which means the total number of bytes in the header is 8 × 4, or 32 bytes
- The first 20 bytes are the base header, the next 12 bytes are the options

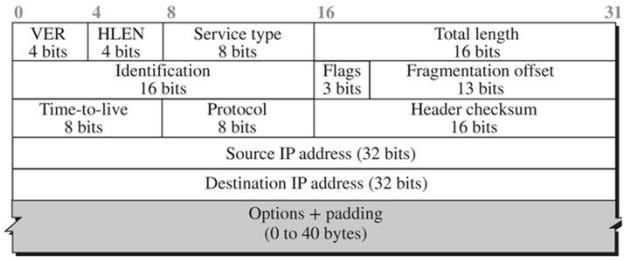


Example: Bytes of Data

In an IPv4 packet, the value of HLEN is 5, and the value of Total Length field is $(0028)_{16}$. How many bytes of data are being carried by this packet?

Solution:

- The HLEN value is 5, which means the total number of bytes in the header is 5 × 4, or 20 bytes (i.e., no options)
- The total length is $(0028)_{16}$ or 40 bytes, which means the packet is carrying 40 – 20 = 20 bytes of data



Example: TTL and Protocol

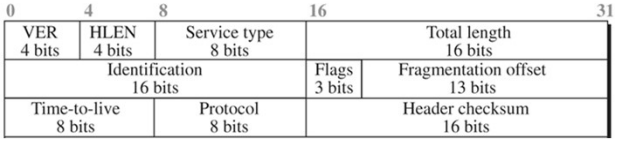
An IPv4 packet has arrived with the first few hexadecimal digits as shown.

$$(45000028000100000102\ldots)_{16}$$

How many hops can this packet travel before being dropped? The data belong to what upper-layer protocol?

Solution:

- To find the time-to-live field, we skip 8 bytes (16 hexadecimal digits)
  - Time-to-live field is the ninth byte, which is $(01)_{16}$
- This means the packet can travel only one hop
- Protocol field is the next byte $(02)_{16}$, which means that the upper-layer protocol is IGMP

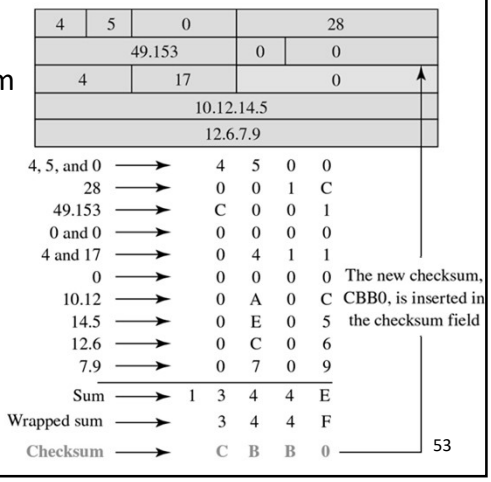


Concept of Checksum

- Checksum is used in the Internet by several protocols for error checking
- Main idea:
  - In addition to sending our data, e.g., the set of numbers is (7, 11, 12, 0, 6), we send (7, 11, 12, 0, 6, 36), where 36 is the sum of the original numbers
  - The receiver adds the five numbers and compares the result with the sum
    - If the two are the same, assume no error, accept the five numbers, and discard the sum
    - Otherwise, there is an error somewhere and the data are not accepted
  - To make it easier for the receiver, we can send the negative (complement) of the sum, called the checksum, e.g., we send (7, 11, 12, 0, 6, −36)
    - The receiver can add all the numbers received (including the checksum)
    - If the result is 0, it assumes no error; otherwise, there is an error

Header Checksum Calculation

- Checksum is calculated over the header only
- Example: checksum calculation for an IPv4 header without options
  - Header is divided into 16-bit sections
  - Sections are added
  - If a carry (leftmost digit) occurs, add carry to sum
  - Sum is complemented
  - Result is inserted in the checksum field

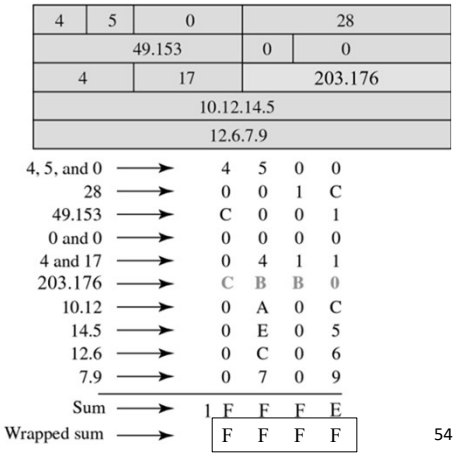


- Note:
  - F = $(1111)_2$ → complement = $(0000)_2$ = 0
  - 4 = $(0100)_2$ → complement = $(1011)_2$ = B
  - 3 = $(0011)_2$ → complement = $(1100)_2$ = C

Header Checksum Verification

- To verify a checksum at the receiver, same procedure is used including header checksum
  - Should yield all 1's if no error is detected

- Note:
  - IP is not a reliable protocol, does not check if payload is corrupted during transmission
  - Only header added by IP is checked
  - Other protocols are responsible for their own data

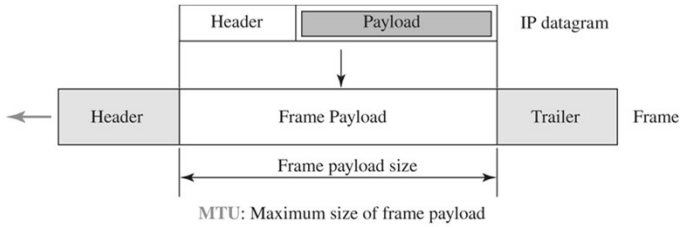


Fragmentation

- A datagram can travel through different networks
- Each router decapsulates the IP datagram from the frame it receives, processes it, and then encapsulates it in another frame
  - Format and size of the received frame depend on the protocol used by the physical network through which the frame has just traveled
  - Format and size of the sent frame depend on the protocol used by the physical network through which the frame is going to travel
- For example, if a router connects a LAN to a WAN, it receives a frame in the LAN format and sends a frame in the WAN format

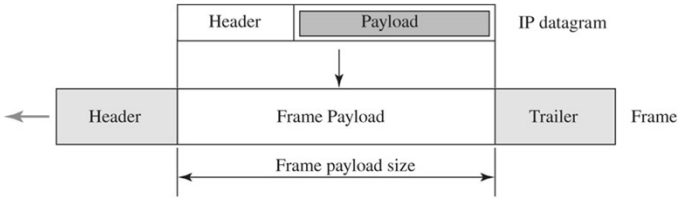
Fragmentation (cont.)

- Each link-layer protocol has its own frame format
- Each frame format has maximum size of the payload that can be encapsulated in a frame
  - Total size of the datagram must be less than the maximum size (called maximum transfer unit)
  - For current technology, this size is much less than 65,535 bytes



Fragmentation (cont.)

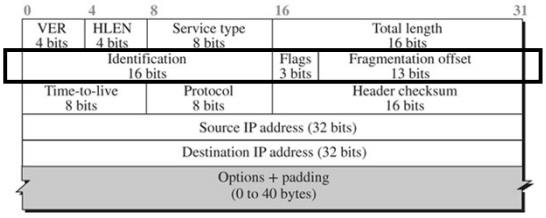
- Fragmentation is dividing the payload of the IP datagrams to make it possible to pass through networks
  - Datagram can be fragmented several times before reaching destination
  - Can be fragmented by source host or any router in the path
  - Needs to be reassembled at the destination host

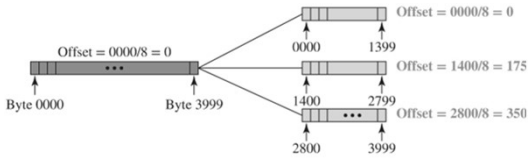


MTU: Maximum size of frame payload

Fragmentation (cont.)

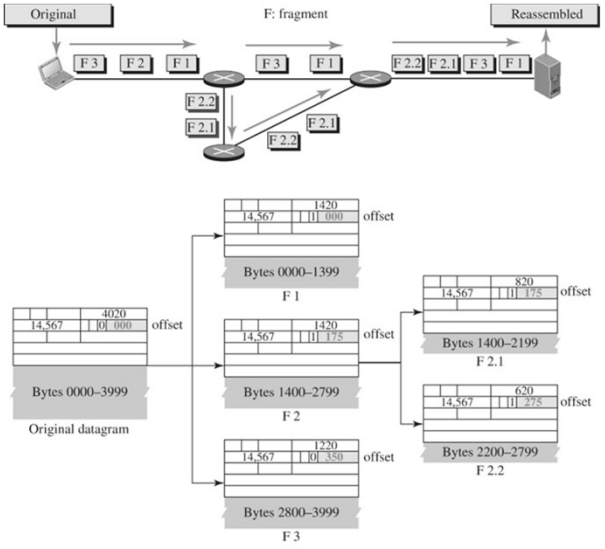
- Fields in IP datagram related to fragmentation:
  - Identification: identifies datagram originating from the source host (counter)
  - Flags field: defines three flags
    - Leftmost bit is reserved
    - D: 1 means do not fragment, 0 means fragment if necessary
    - M: 1 means more fragments coming, 0 means last (or only) fragment
  - Fragmentation offset field: relative position of this fragment with respect to the whole datagram (in 8 bytes)





Example: Fragmentation

- Remarks:
  - Fragments can be fragmented
  - Identification field is the same for all fragments belonging to same datagram
  - M bit of Flags field is 1 except for last fragment
  - Offset is always relative to the original datagram
  - Destination can reassemble even though fragments are out of order



Example 1: Flags

A packet has arrived with an M bit value of 0. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

Solution:

- If the M bit is 0, it means that there are no more fragments; the fragment is the last one
- We cannot say if the original packet was fragmented or not
- A non-fragmented packet is considered the last fragment

Example 2: Flags

A packet has arrived with an M bit value of 1. Is this the first fragment, the last fragment, or a middle fragment? Do we know if the packet was fragmented?

Solution:

- If the M bit is 1, it means that there is at least one more fragment
- This fragment can be the first one or a middle one, but not the last one
- We do not know if it is the first one or a middle one; we need more information (e.g., the value of the fragmentation offset)

Example 1: Offset

A packet has arrived with an M bit value of 1 and a fragmentation offset value of 0. Is this the first fragment, the last fragment, or a middle fragment?

Solution:

- Because the M bit is 1, it is either the first fragment or a middle one
- Because the offset value is 0, it is the first fragment

Example 2: Offset

A packet has arrived in which the offset value is 100. What is the number of the first byte? Do we know the number of the last byte?

Solution:

- To find the number of the first byte, we multiply the offset value by 8
- This means that the first byte number is 800
- We cannot determine the number of the last byte unless we know the length of the data

Example 3: Offset

A packet has arrived in which the offset value is 100, the value of HLEN is 5, and the value of the Total Length field is 100. What are the numbers of the first byte and the last byte?

Solution:

- To find the number of the first byte, we multiply the offset value by 8
- This means that the first byte number is 800
- The total length is 100 bytes, and the header length is 20 bytes (5 × 4), which means that there are 80 bytes in this datagram
- If the first byte number is 800, the last byte number must be 879

Options

- The header of the IPv4 datagram is made of two parts:
  - Fixed part: 20 bytes long
  - Variable part: options that can be a maximum of 40 bytes (in multiples of 4 bytes) to preserve the boundary of the header
- Examples:
  - Record Route: record addresses of routers that handle the datagram
    - Useful for debugging and management
  - Strict Source Route: predetermine a route for the datagram to travel through
    - All routers defined must be visited by the datagram, no other router not on the list
  - Timestamp: record time of datagram processing by a router
    - Useful to track behavior of different routers
  - End-of-Option: 1-byte option used for padding at the end of the option field

Security of IPv4 Datagrams

- Some security issues in IP protocol:
  - Packet Sniffing: An intruder may intercept an IP packet and make a copy of it
  - Packet Modification: The attacker intercepts the packet, changes its contents, and sends the new packet to the receiver
  - IP Spoofing: An attacker can masquerade as somebody else and create an IP packet that carries the source address of another computer
- IP Security (IPSec):
  - IPSec protocol can protect IP packets from these attacks
  - Creates a connection-oriented service between two entities in which they can exchange IP packets without worrying about theses three attacks

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - ICMPv4
- Sidebar: Network Neutrality

Internet Control Message Protocol version 4 (ICMPv4)

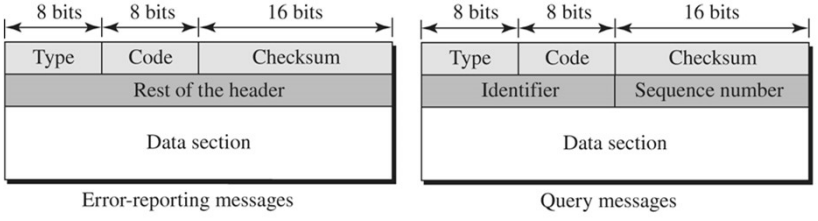
- IPv4 has no error-reporting or error-correcting mechanism
  - e.g. a router cannot find destination, time-to-live is 0, a fragment is missing at receiver, a datagram has to be discarded, etc.
  - IP is an unreliable protocol
- It also lacks a mechanism for host and management queries
  - e.g., check if a router or a host is alive

Internet Control Message Protocol version 4 (ICMPv4)

- ICMPv4 is designed to compensate for the above two deficiencies
  - ICMP messages carried in IP datagrams (i.e., Protocol field is set to 1)
  - Used by hosts and routers to communicate network-level information
- ICMP messages divided into two broad categories:
  - Error-reporting messages: report problems that a router or a host (destination) may encounter when it processes an IP packet
  - Query messages: help a host or a network manager get specific information from a router or another host
    - Nodes can discover their neighbor, hosts can discover and learn about routers on their network, and routers can help a node redirect its messages

ICMP Messages Format

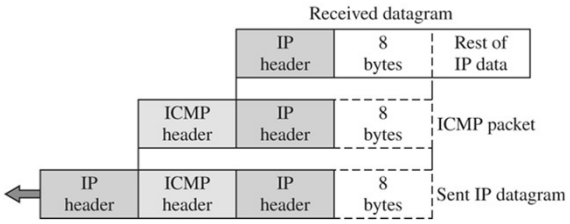
- ICMP messages has an 8-byte header and a variable-size data section
  - First 4 bytes are common: type, code, and checksum
    - For all types and codes, check ICMP Parameters (iana.org)
  - Data section:
    - Carries information to find problematic packet in error reporting
    - Carries extra information based on type of query



Type	Code	Description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

Error-Reporting Messages

- IP is an unreliable protocol
- ICMP reports errors to the original source (does not correct errors)
- ICMP forms an error packet, which is then encapsulated in an IP datagram
  - Example: access a web page and server is down → send Destination Unreachable (Type 3) and Host Unreachable (Code 1) ICMP message
  - Data section is IP header + first 8 bytes of datagram causing error



Query Messages

- Used to test liveliness of hosts or routers, find one-way or round-trip time between two devices, etc.
- Query messages come in pairs: request and reply
  - Examples: ICMP echo request (Type 8) and ICMP echo reply (Type 0)
- There are several debugging tools that use query messages in the Internet
  - ping program is used to find if a host is alive and is responding
    - Source sends echo-request message, if destination alive, respond with echo-reply message
  - traceroute (or tracert in Windows) can be used to trace the path of a packet from a source to the destination

Example: ping

Send a ping message to the ahmedsakr.com site and show output.

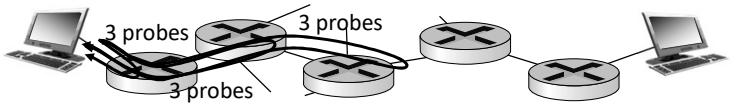
Solution:

```
C:\Users\admin> ping ahmedsakr.com

Pinging ahmedsakr.com [185.199.108.153] with 32 bytes of data:
Reply from 185.199.108.153: bytes=32 time=16ms TTL=56
Reply from 185.199.108.153: bytes=32 time=10ms TTL=56
Reply from 185.199.108.153: bytes=32 time=9ms TTL=56
Reply from 185.199.108.153: bytes=32 time=9ms TTL=56

Ping statistics for 185.199.108.153:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 16ms, Average = 11ms
```

- Note:
  - TTL is the max no. of routers a packet can go through before being discarded (default 64)
  - Ping calculates the round-trip time (rrt): insert sending time in data section of the message, when packet arrives, subtract arrival time from departure time to get rtt

Traceroute or Tracert

- A program provides delay measurement from source to router along end-end Internet path towards destination
- Find the IP addresses of all the routers that are visited along the path
- For all R_i:
  - sends three packets that will reach router *i* on path towards destination (with time-to-live field value of R_i)
  - router R_i will return packets to sender
  - sender measures time interval between transmission and reply



Example: tracert

Use tracert in Windows and show output for eurecom.fr site.

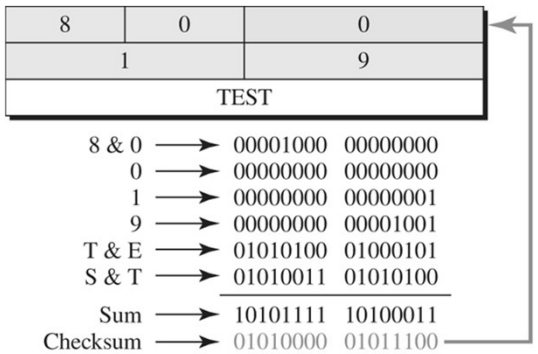
Solution:

```
C:\Users\admin>tracert eurecom.fr

Tracing route to eurecom.fr [193.55.113.222]
over a maximum of 30 hops:

  6    22 ms    22 ms    22 ms  tcore4-toronto12_39.net.bell.ca [64.230.52.178]
  7     *        *        *     Request timed out.
  8    27 ms    23 ms    22 ms  tcore4-chicagocp-bundle-ether15.net.bell.ca [142.124.127.174]
  9    20 ms    20 ms    21 ms  bx10-chicagodt_ae1.net.bell.ca [64.230.78.175]
 10    28 ms    24 ms    25 ms  bx10-chicagodt_et-8/1/2_ae8.net.bell.ca [184.150.181.36]
 11   116 ms   113 ms   115 ms  et-3-3-0.cr2-par7.ip4.gtt.net [213.200.119.214]
 12   113 ms   113 ms   114 ms  renater-gw-th2.gtt.net [77.67.123.210]
 13   119 ms   120 ms   119 ms  te-0-1-0-14-ren-nr-lyon2-rtr-091.noc.renater.fr [193.51.180.55]
 14   120 ms   120 ms   120 ms  xe-0-0-14-marseille2-rtr-131.noc.renater.fr [193.51.180.105]
 15   118 ms   117 ms   117 ms  xe-1-0-10-marseille1-rtr-131.noc.renater.fr [193.51.180.121]
 16   122 ms   124 ms   122 ms  te0-2-0-0-ren-nr-sophia-rtr-091.noc.renater.fr [193.51.177.21]
 17   122 ms   122 ms   122 ms  eurocom-valbonne-gi9-7-sophia-rtr-021.noc.renater.fr [193.51.187.17]
```

ic link

ICMP Checksum

- Checksum is calculated over the entire message (header and data)
- Example: checksum calculation for a simple echo-request message
  - Message is divided into 16-bit (2-byte) words
  - Words are added and the sum is complemented
  - Sender puts value in checksum field



Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - Mobile IP
- Sidebar: Network Neutrality

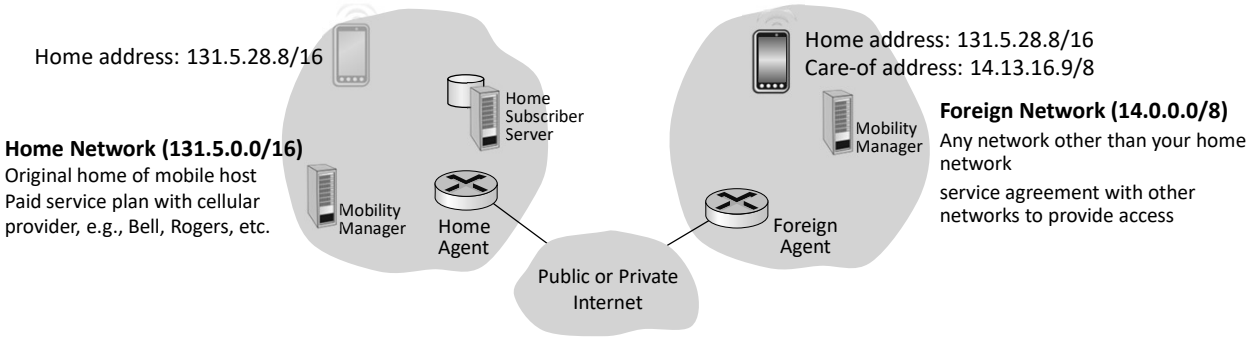
RFC 2002, 3344

Mobile IP

- Extension of IP protocol to allow mobile hosts to be connected to the Internet at any location where the connection is possible
- Addressing is the main problem that must be solved in providing mobile communication using the IP protocol
  - IP addresses were designed to work with stationary hosts because part of address defines the network to which the host is attached

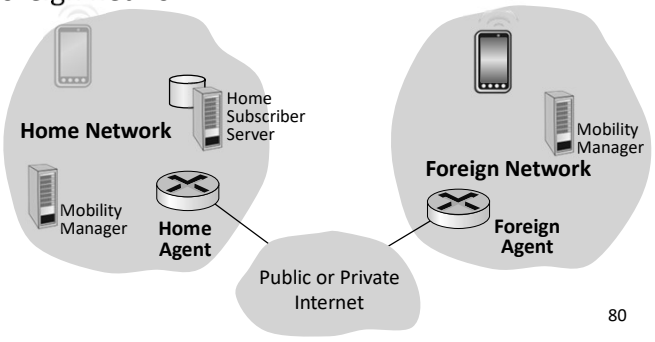
Mobile IP (cont.)

- For mobile hosts, the solution is to use two IP addresses:
  - Home address: permanent, associates host to home network
  - Care-of address: temporary, changes when a host moves from a foreign network to a foreign network



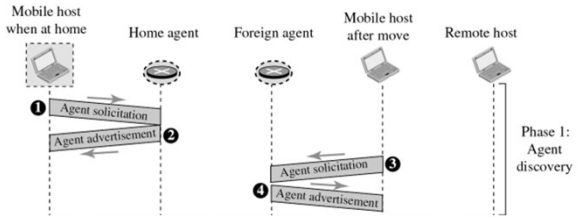
Mobile IP (cont.)

- To make change of address transparent to the Internet requires
  - Home agent: router attached to home network
    - Stores information about mobile hosts whose permanent home address is in the home agent's network
    - Acts on behalf of the mobile host when a remote host sends a packet to the mobile host
    - Receives the packet and send it to the foreign agent
  - Foreign agent: router attached to foreign network
    - Stores information about mobile nodes visiting its network
    - Receives and delivers packets sent by the home agent to the mobile host



Remote Host and Mobile Host Communication

- To communicate with a remote host, a mobile host goes through three phases:
  - Agent discovery
    - Host learns the address of home agent before leaving home network
    - Host learns the address of foreign agent and care-of address after moving to a foreign network

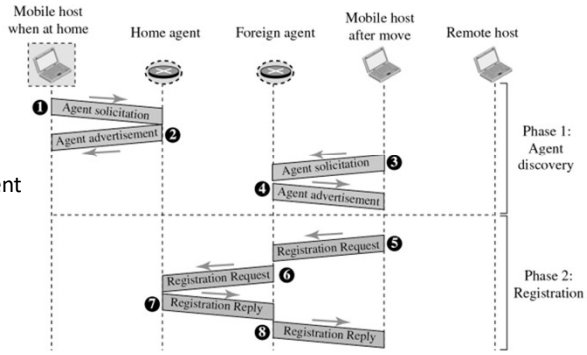


Note:

- Home and foreign agents advertise their presence on the network to which they are attached
- If a mobile host does not receive this ad, it can initiate an agent solicitation
- Mobile IP uses the existing ICMP router advertisement (Type 9) and router solicitation (Type 10) messages

Remote Host and Mobile Host Communication (cont.)

- To communicate with a remote host, a mobile host goes through three phases:
  - Registration
    - After host moved to a foreign network and discovered the foreign agent
    - Mobile host must:
      - register its care-of address with the foreign agent and announce home address and agent
      - register itself with its home agent
      - renew registration if it has expired
      - cancel its registration when it returns

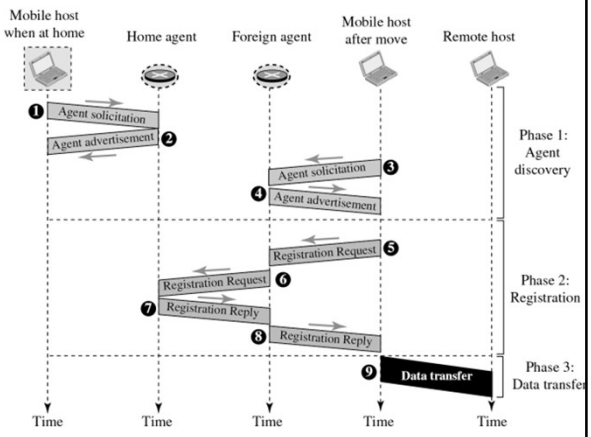


Note:

- Mobile host uses registration request and reply
- Foreign agent acts on behalf of mobile host to register with its home agent

Remote Host and Mobile Host Communication (cont.)

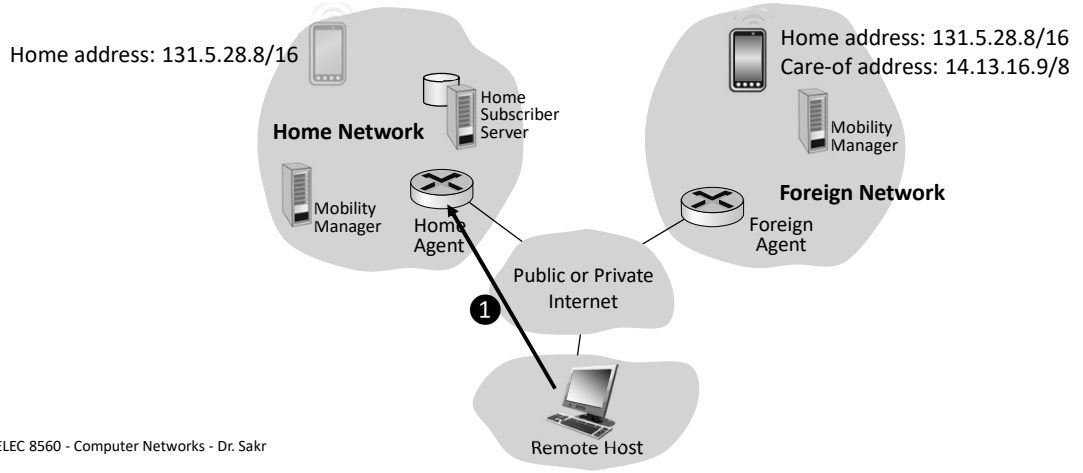
- To communicate with a remote host, a mobile host goes through three phases:
  - Data transfer
    - Movement of mobile host is transparent to the rest of the Internet (i.e., unaware)
      - Remote hosts send packets using the home address of the mobile host as destination
      - Remote hosts receive packets with the home address of the mobile host as source



Data Transfer

❶ From remote host to home agent

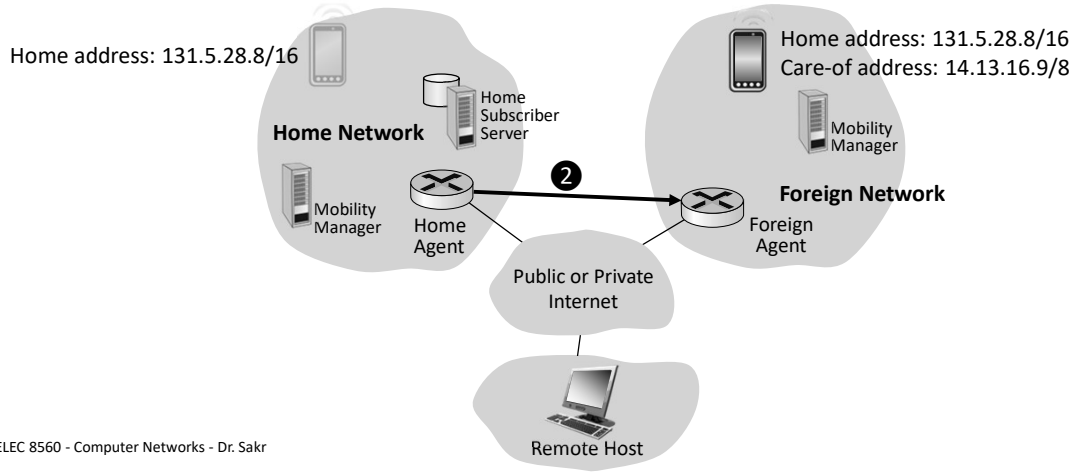
- Remote host uses home address of mobile host as the destination address to send packets
- Packets intercepted by the home agent



Data Transfer

❷ From home agent to foreign agent

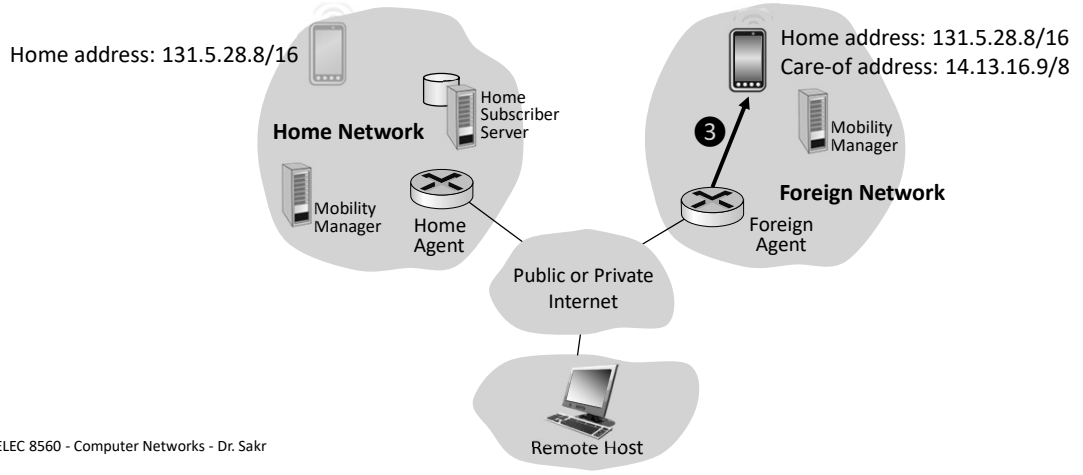
- Home agent redirects (tunnel) packets to foreign agent encapsulated in a new IP packet



Data Transfer

❸ From foreign agent to mobile host

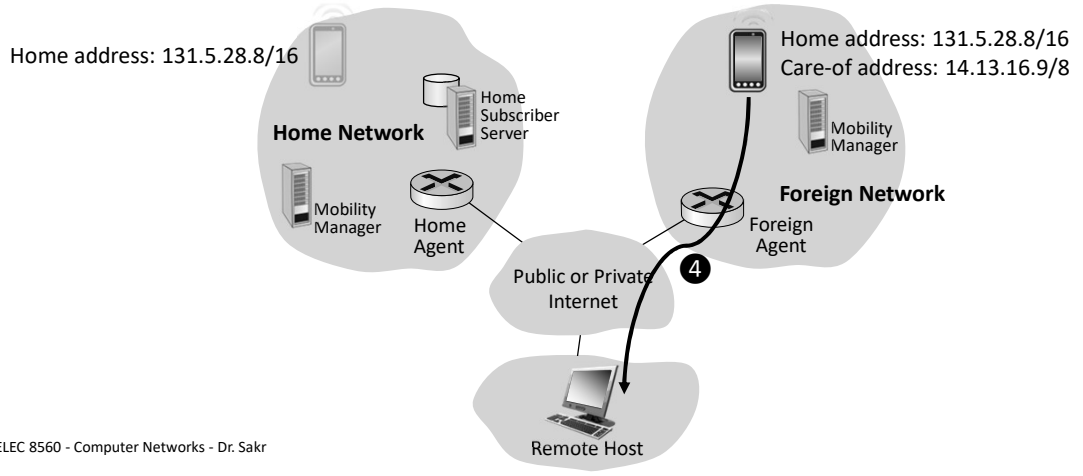
- Foreign agent extracts original packet and changes destination address to the care-of address



Data Transfer

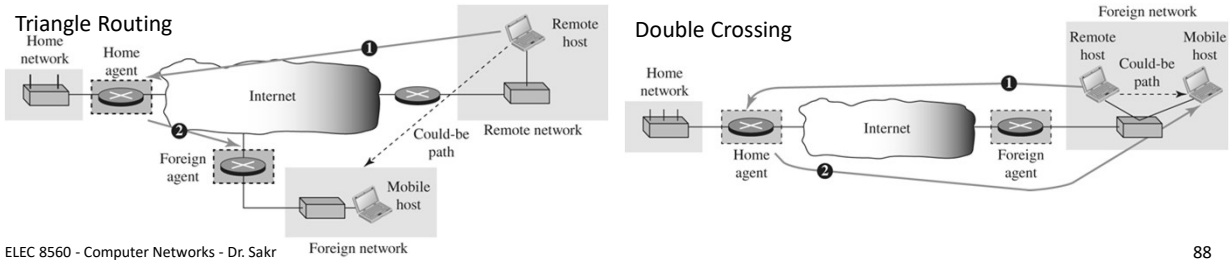
❹ From mobile host to remote host

- Mobile host uses its home address as the source address to send packets directly to remote host



Communication Inefficiency

- Communication involving mobile IP can be inefficient, for example,
  - Triangle routing (or dog-leg routing): moderate case
    - When remote host sends packets to a mobile host that moved to another network
      - Home agent can solve by sending updates to remote hosts with the care-of addresses
  - Double crossing (or 2X): severe case
    - When remote host sends packets to a mobile host that moved to the same network
      - Packets travel the Internet twice

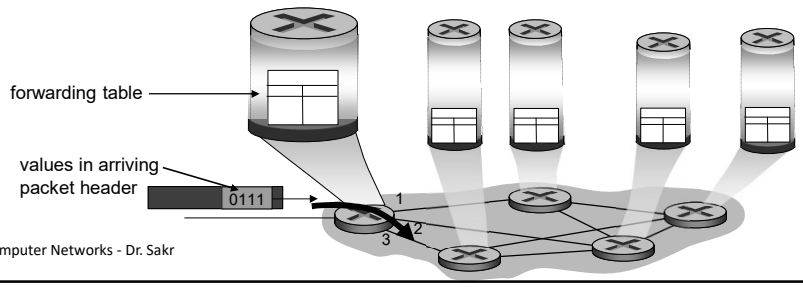


Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - Forwarding of IP packets
- Sidebar: Network Neutrality

Forwarding of IP Packets

- As discussed, forwarding means to deliver packet to the next hop
  - Can be the final destination or an intermediate connecting device
- Forwarding can be
  - Destination-based forwarding: based of destination IP address, connectionless
  - Generalized forwarding: other header fields can determine an action, connection-oriented

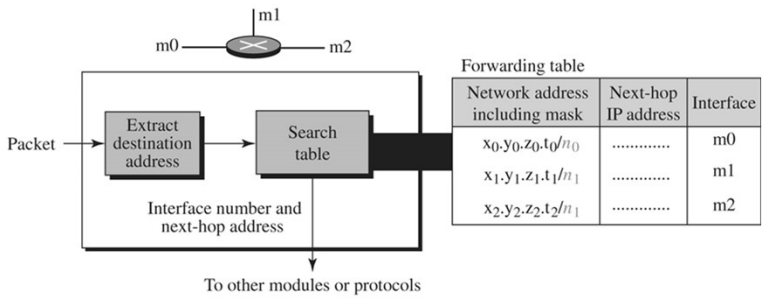


Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - Destination-Based Forwarding
- Sidebar: Network Neutrality

Forwarding Based on Destination Address

- A traditional approach which is prevalent today
- Forwarding requires a host or a router to have a forwarding table
- When a host has a packet to send or when a router has received a packet to be forwarded, it looks at the table to find the next hop to deliver the packet to

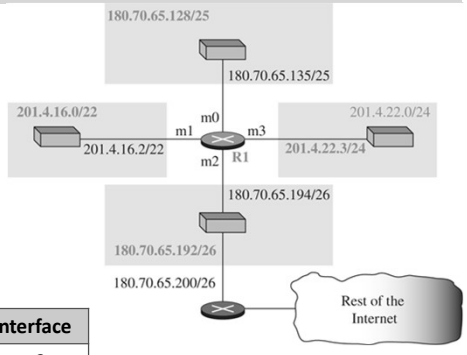


Example: Destination-Based Forwarding Table

Make a forwarding table for router R1 using the configuration below.

Solution:

Network address/mask	Next hop	Interface
180.70.65.192/26	—	m2
180.70.65.128/25	—	m0
201.4.22.0/24	—	m3
201.4.16.0/22	—	m1
Default	180.70.65.200	m2



Alternatively, R1 can use prefix (leftmost) bits

Leftmost bits in the destination address	Next hop	Interface
10110100 01000110 01000001 11	—	m2
10110100 01000110 01000001 1	—	m0
11001001 00000100 00011100	—	m3
11001001 00000100 000100	—	m1
Default	180.70.65.200	m2

Forwarding Table Search Algorithm

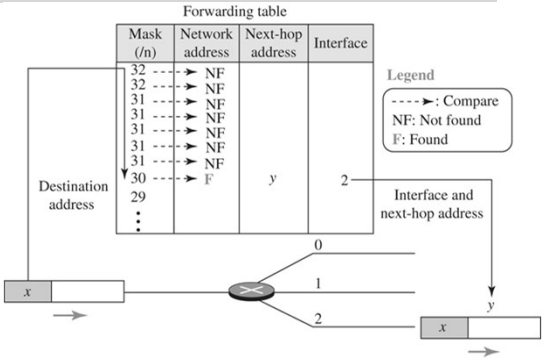
- Simple, but not very efficient, search method is called the longest prefix (or mask) match
  - Forwarding table is divided into buckets, one for each prefix
  - Router first tries the longest prefix
  - If the destination address is found in this bucket, the search is complete
  - If the address is not found, the next prefix is searched, and so on

Example: Longest Prefix Match Algorithm

Figure below shows a simple example of searching in a forwarding table using the longest mask algorithm.

Solution:

- Forwarding algorithm gets destination address of the packet
- Search the mask column for each entry by apply the mask to find destination network address
- Then, check network addresses in the table until it finds the match
- Router then extracts the next-hop address and the interface number to be delivered to the data-link layer



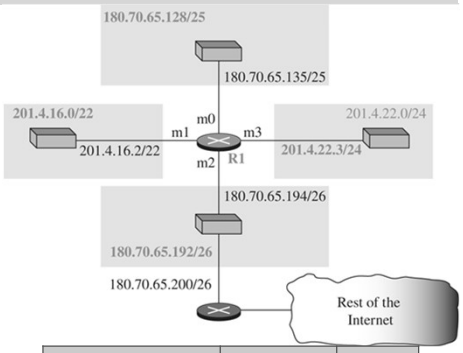
Example: Destination-Based Forwarding

Show the forwarding process if a packet arrives at R1 with the destination address 180.70.65.140.

Solution:

The router performs the following steps:

- First mask (/26) is applied to the destination address
  - result is 180.70.65.128, does not match the corresponding network address
- Second mask (/25) is applied to the destination address
  - result is 180.70.65.128, matches the corresponding network address
- The next-hop address and the interface number m0 are extracted for forwarding the packet



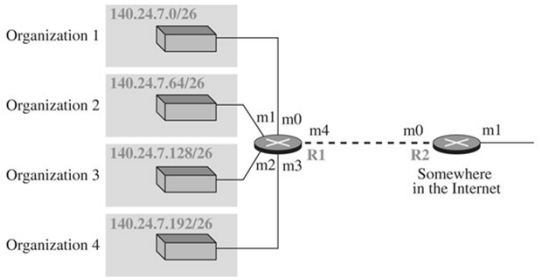
Network address/mask	Next hop	Interface
180.70.65.192/26	—	m2
180.70.65.128/25	—	m0
201.4.22.0/24	—	m3
201.4.16.0/22	—	m1
Default	180.70.65.200	m2

Address Aggregation

- In classful addressing, there is only one entry in the forwarding table for each site outside the organization
  - When a packet arrives at the router, the router checks the corresponding entry and forwards the packet accordingly
- In classless addressing, it is likely that the number of forwarding table entries (and search time) will increase
  - Due to dividing the address space into smaller blocks
- To alleviate the problem, the idea of address aggregation was designed

Address Aggregation (cont.)

- R1 connected to 4 organizations with 64 addresses each
  - R1 has a longer forwarding table compared to R2
- For R2, packets with destination 140.24.7.0-140.24.7.255 go to m0, regardless of organization number
- This is called address aggregation
  - Blocks of four organizations are aggregated into one larger block



Forwarding table for R1

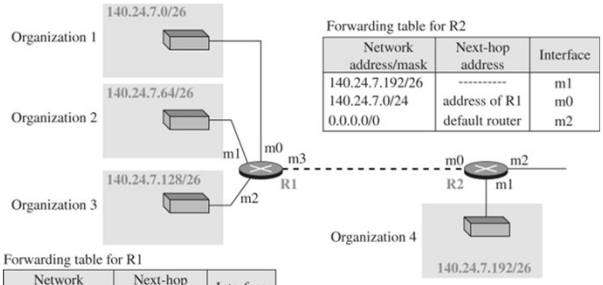
Network address/mask	Next-hop address	Interface
140.24.7.0/26	----------	m0
140.24.7.64/26	----------	m1
140.24.7.128/26	----------	m2
140.24.7.192/26	----------	m3
0.0.0.0/0	address of R2	m4

Forwarding table for R2

Network address/mask	Next-hop address	Interface
140.24.7.0/24	----------	m0
0.0.0.0/0	default router	m1

Longest Mask Addressing

- What happens if Organization 4 cannot be connected to R1 (e.g., not geographically close)?
  - We can still use address aggregation with other three organizations
- Longest mask addressing states that forwarding tables must be sorted from the longest mask to the shortest mask
  - Say packet destined to 140.24.7.200
  - This way R2 will apply first mask and forward data to Organization 4 on m1
  - Applying /24 mask first would have resulted in incorrect routing to R1



Forwarding table for R2

Network address/mask	Next-hop address	Interface
140.24.7.192/26	----------	m1
140.24.7.0/24	address of R1	m0
0.0.0.0/0	default router	m2

Forwarding table for R1

Network address/mask	Next-hop address	Interface
140.24.7.0/26	----------	m0
140.24.7.64/26	----------	m1
140.24.7.128/26	----------	m2
0.0.0.0/0	default router	m3

Hierarchical Routing

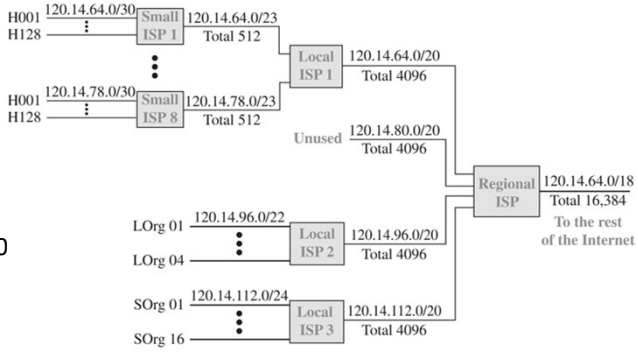
- Creating a hierarchy can also solve the problem of gigantic forwarding tables
- For example, regional ISP is granted addresses → divides into subblocks and assigns to local ISPs → divides into subblocks and assigns to organization → subnetting → …
- This way the rest of the Internet does not have to be aware of this division
  - All customers of the local ISP are defined as a.b.c.d/n to the regional ISP
  - All customers of the regional ISP are defined as e.f.g.h/m to the rest of Internet

Example: Hierarchical Routing with ISPs

A regional ISP is granted 16,384 addresses starting from 120.14.64.0. The regional ISP divided this block into 4 subblocks, each with 4096 addresses. Three subblocks are assigned to three local ISPs, the second subblock is reserved, and the first subblock is assigned into 8 smaller blocks with 128 households. Show the hierarchy.

Solution:

- *16,384 = 2^{14} → 120.14.64.0*/18
- Note:
  - Mask of each block of local ISPs is /20 because the original block with mask /18 is divided into 4 blocks
  - Mask of each block of small ISPs is /23 because the local ISP block with mask /20 is divided into 8 blocks
  - Mask of each household is /30 because the small ISP block with mask /23 is divided into 128 blocks



Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - Label-Based Forwarding
- Sidebar: Network Neutrality

Forwarding Based on Label

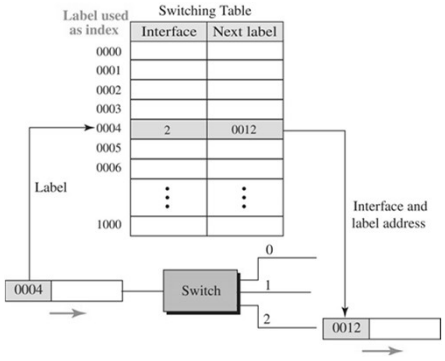
- In a connectionless network (datagram approach), a router forwards a packet based on the destination address in the header of the packet
- In a connection-oriented network (virtual-circuit approach), a switch forwards a packet based on the label attached to the packet
- Routing is normally based on searching the contents of a table
- Switching can be done by accessing a table using an index (from label)

Example: Label-Based Forwarding

Figure below a simple example of using a label to access a switching table. Since the labels are used as the index to the table, finding information in table is immediate.

Solution:

- Forwarding algorithm gets destination address of the packet
- Use label attached to the packet to access the table
- Router then extracts the next-hop label and the interface number to be delivered to the data-link layer

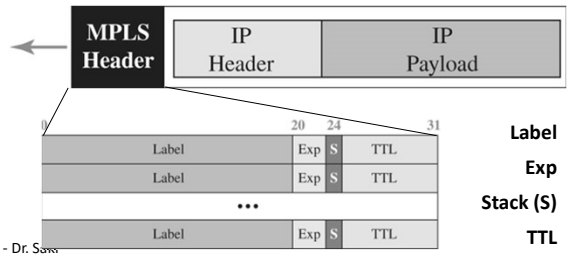


Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - MPLS
- Sidebar: Network Neutrality

Multiprotocol Label Switching (MPLS)

- MPLS-capable routers can behave like a router and a switch
  - Router: forward packets based on destination address (datagram approach)
  - Switch: forward packets based on label (virtual-circuit approach)
- This leads to much faster lookup using fixed-length identifier
- MPLS header:
  - Encapsulates the IPv4 packet in an MPLS packet
  - Header is a stack of sub-headers used for multilevel hierarchical switching



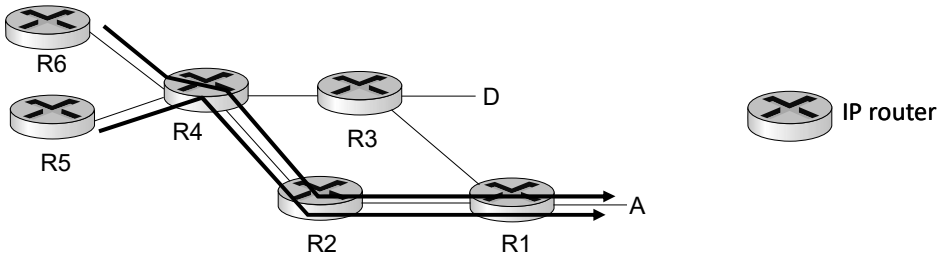
Label Used to index forwarding table
Exp Reserved for experimental purposes
Stack (S) 1 means header is last in stack
TTL Each visited router decrements by 1

MPLS-capable Routers

- Sometimes called label-switched router
- Forward packets to outgoing interface based only on label value
  - Do not inspect IP address
  - MPLS forwarding table distinct from IP forwarding tables
- Flexibility: MPLS forwarding decisions can differ from those of IP
  - Use destination and source addresses to route flows to same destination differently (traffic engineering)
  - Re-route flows quickly if link fails: pre-computed backup paths

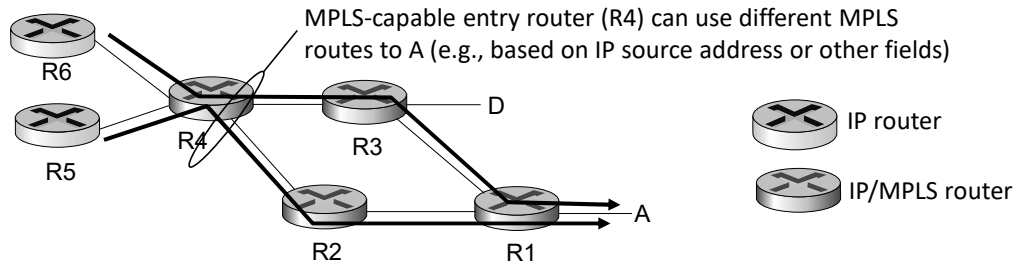
MPLS vs. IP Paths

- IP routing:
  - path to destination determined by destination address alone



MPLS vs. IP Paths

- IP routing:
  - path to destination determined by destination address alone
- MPLS routing:
  - path to destination can be based on source and destination address
  - flavor of generalized forwarding
  - fast re-route: precompute backup routes in case of link failure



Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - IPv6
- Sidebar: Network Neutrality

Internet Protocol version 6 (IPv6)

- Sometimes called IP new generation (IPng)
- Main motivations:
  - Address depletion of the 32-bit IPv4 addresses
  - Fast processing/forwarding (fixed-length header)

IPv6 Addressing

- An IPv6 address is 128 bits or 16 bytes, four times the address length in IPv4
- Representation

Notation	IP Address
Binary	1111111011101101011 … 1111111100000000
Colon Hexadecimal	FEF6:BA98:7654:3210:ADEF:BBFF:2922:FF00

- Abbreviations: remove leading 0s and consecutive 0s (once)

2001:0db8:0000:0000:0001:0000:0000:0001→ 2001:db8::1:0:0:1

- Address space of IPv6 contains 2^{128} addresses
  - 2^{96} times the IPv4 address – definitely no address depletion

2^{128} = 340,282,366,920,938,463,374,607,431,768,211,456 = **340+ undecillion***!*

Address Space Allocation

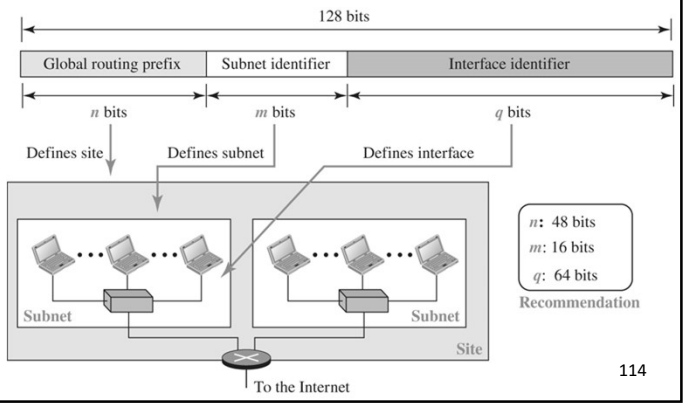
- Similar to IPv4, address space of IPv6 is divided into several blocks of varying size and each block is allocated for a special purpose
- Most of the blocks are still unassigned and have been set aside for future use

Block prefix	*CIDR*	*Block assignment*	*Fraction of space*
0000 0000	0000::/8	Special addresses	1/256
001	2000::/3	Global unicast	1/8
1111 110	FC00::/7	Unique local unicast	1/128
1111 1110 10	FE80::/10	Link local addresses	1/1024
1111 1111	FF00::/8	Multicast addresses	1/256

Prefixes for assigned IPv6 addresses

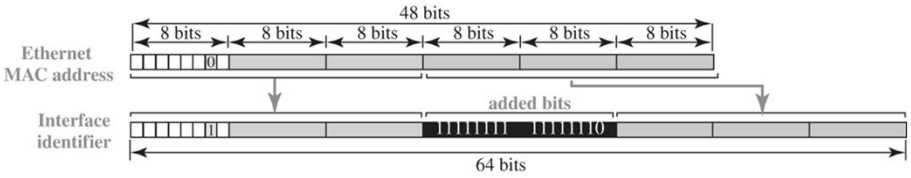
Global Unicast Address

- Block used for one-to-one communication between two hosts
  - CIDR is 2000::/3 which means prefix 001 is same for all addresses
  - Size is 2^{125} which is mote than enough for Internet many years to come
- Each address is divided into three parts:
  - Global routing prefix (*n* bits)
    - Site: organization or ISP
  - Subnet identifier (*m* bits)
    - Up to 65,536 subnets
  - Interface identifier (*q* bits)
    - Not host identifier
    - Can be mapped to MAC address



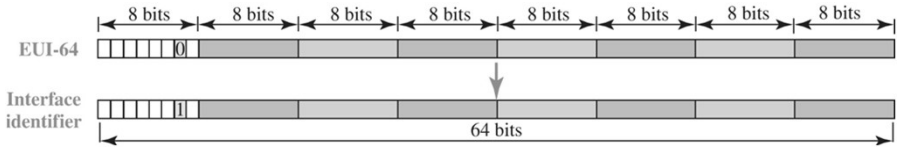
Ethernet MAC Address Mapping

- Defined by Ethernet
- Link layer addressing scheme for mapping the 48-bit MAC address into 64-bit interface identifier
  - Split MAC address into two pieces
  - Insert 16 bits (15 ones + 1 zero or $FFFE_{16}$) in between
  - Invert local/global (7th leftmost) bit from 0 to 1 (local to global)



Extended Unique Identifier (EUI-64) Mapping

- IEEE created EUI-64 standard for 64-bit MAC addresses
- Link layer addressing scheme for mapping the 64-bit physical address into 64-bit interface identifier
  - Invert local/global (7th leftmost) bit from 0 to 1 (local to global)



Example: Subnet Identifier

An organization is assigned the block 2000:1456:2474/48. What is the CIDR notation for the blocks in the first and second subnets in this organization.

Solution:

- Assume *n=48*, *m=16*, and *q=64* bits
- Assume subnet identifiers $(0001)_{16}$ and $(0002)_{16}$
- The blocks are
  - 2000:1456:2474:0001/64
  - 2000:1456:2474:0002/64

Example 1: Interface Identifier

Find the interface identifier if the physical address in the EUI is
$(F5–A9–23–EF–07–14–7A–D2)_{16}$
using the format we defined for Ethernet addresses.

Solution:

- We only need to change the seventh bit of the first octet from 0 to 1 and change the format to colon hex notation
- The result is F7A9:23EF:0714:7AD2

EUI-64 Address of the Host

F5–A9–23–EF–07–14–7A–D2

11110101

11110111

F7A9:23EF:0714:7AD2

EUI-64 Interface ID of the Host

Example 2: Interface Identifier

Find the interface identifier if the physical MAC address is
$$(F5{-}A9{-}23{-}14{-}7A{-}D2)_{16}$$
using the format we defined for Ethernet addresses.

Solution:

- We need to change the seventh bit of the first octet from 0 to 1, insert $FFFE_{16}$, and change the format to colon hex notation
- The result is F7A9:23FF:FE14:7AD2

MAC Address of the Host

F5−A9−23−14−7A−D2

F5−A9−23−FF−FE−14−7A−D2

11110101

11110111

F7A9:23FF:FE14:7AD2

EUI-64 Interface ID of the Host

Example 3: Interface Identifier

An organization is assigned the block 2000:1456:2474/48. What is the IPv6 address of an interface in the third subnet if the IEEE physical address of the computer is
$$(F5{-}A9{-}23{-}14{-}7A{-}D2)_{16}$$

Solution:

- The interface identifier is F7A9:23FF:FE14:7AD2
- Adding global prefix and subnet identifier $(0003)_{16}$, we get

2000:1456:2474:0003:F7A9:23FF:FE14:7AD2/128

Autoconfiguration

- In IPv4, hosts and routers can originally be configured manually and DHCP allocates IPv4 addresses to hosts that join the network
- In IPv6, DHCP protocol can still be used to allocate an IPv6 address to a host, but a host can also configure itself
  - First, host creates a link local address for itself
    - 128 bits: 10-bit link local prefix (1111 1110 10) + 54 zeroes + 64-bit interface identifier
  - Host test the link local address to see if it is unique (not used by other hosts)
  - If unique, host stores this address as its link local address, and sends a router solicitation message to get global unicast prefix
  - Host receives router advertisement message that announces the combination of global unicast prefix and subnet identifier
  - Host appends its interface identifier to this prefix to find and store its global unicast address

Example: Autoconfiguration

Assume a host with Ethernet address $(F5{-}A9{-}23{-}11{-}9B{-}E2)_{16}$ has joined the network. What would be its global unicast address if the global unicast prefix of the organization is 3A21:1216:2165 and the subnet identifier is A245:1232.

Solution:

- The interface identifier is F7A9:23FF:FE11:9BE2
- The host then creates its link-local address as FE80::F7A9:23FF:FE11:9BE2
- Assume the address is unique, the host sends a router solicitation message and receives the router advertisement message that announces the combination of global unicast prefix and the subnet identifier as

  3A21:1216:2165:A245:1232
- The host then appends its interface identifier to this prefix to find and store its global unicast address as

  3A21:1216:2165:A245:1232:F7A9:23FF:FE11:9BE2

Renumbering

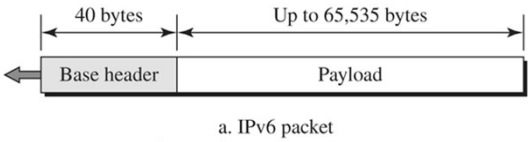
- Each site is given a prefix by ISP to which it is connected
- To allow sites to change ISP, renumbering of the address prefix was built into IPv6 addressing
  - If the site changes the provider, the address prefix needs to be changed
- A router to which the site is connected can advertise a new prefix and let the site use the old prefix for a short time before disabling it
  - During the transition period, a site has two prefixes

Outline

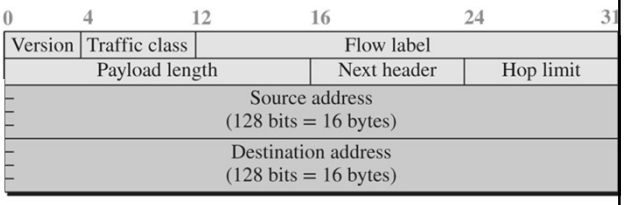
- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - IPv6 datagram format
- Sidebar: Network Neutrality

IPv6 Datagram Format

- Each packet is composed of a base header followed by the payload
  - Version: version number (6 for IPv6)
  - Traffic class: priority among datagrams
  - Flow label: identify datagrams in the same "flow"
  - Payload length: datagram length (excluding header) in bytes
  - Next header: type of next extension header or Protocol in IPv4
  - Hop limit: similar to TTL in IPv4
  - IP addresses



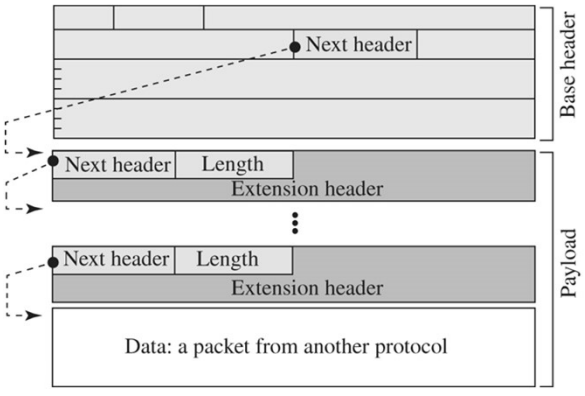
a. IPv6 packet



b. Base header

Payload in an IPv6 datagram

- Payload is a combination of zero or more extension headers (i.e., options) followed by the data from upper layers
  - Next header field value (code) defines type of next header



Some next-header codes

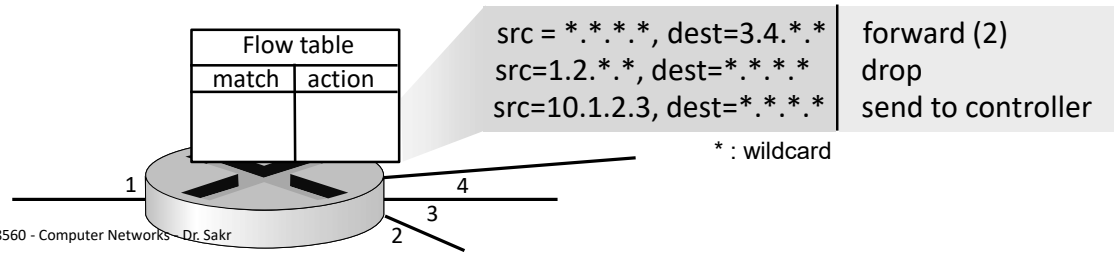
00: Hop-by-hop option
02: ICMPv6
06: TCP
17: UDP
43: Source-routing option
44: Fragmentation option
50: Encrypted security payload
51:Authentication header
59: Null (no next header)
60: Destination option

Concept of Flow and Priority in IPv6

- IP protocol was originally designed as a connectionless protocol
- However, the tendency is to use is as a connection-oriented protocol
  - e.g., MPLS technology described earlier allows us to encapsulate an IPv4 packet in an MPLS header using a label field
- In version 6, the flow label has been directly added to the format of the IPv6 datagram to allow us to use IPv6 as a connection-oriented protocol
- Flow is a sequence of packets that share the same characteristics
  - e.g., travelling same path, same kind of security, etc.
- A router uses a flow label table to route packets based on flow label

Flow Table Abstraction

- Flow is defined by flow label field values
- Generalized forwarding: simple packet-handling rules
  - match: pattern values in packet header fields
  - actions: for matched packet: drop, forward, modify, matched packet or send matched packet to controller
  - priority: disambiguate overlapping patterns
  - counters: bytes and packets



Fragmentation and Reassembly

- There is still fragmentation and reassembly of datagrams in the IPv6 protocol, but there is a major difference in this respect
- IPv6 datagrams can be fragmented only by the source and the reassembly takes place at the destination
  - Routers are not allowed to perform fragmentation to speed up processing
  - If a packet it too long (i.e., larger than MTU), drop and send a *packet-too-big* ICMPv6 error message

Extension Header

- An IPv6 packet is made of a base header and some extension headers
  - Length of the base header is fixed at 40 bytes
- To give more functionality, base header can be followed by up to six extension headers
  - Many of these headers are options in IPv4
  - Six types of extension headers have been defined: hop-by-hop option, source routing, fragmentation, authentication, encrypted security payload, and destination option

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - ICMPv6
- Sidebar: Network Neutrality

The ICMPv6 Protocol

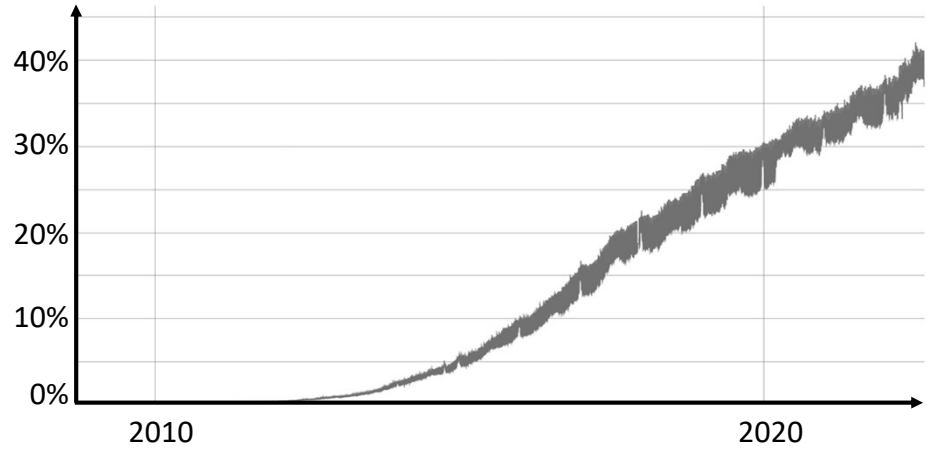
- ICMP has been modified in version 6 of the TCP/IP protocol suite
- ICMPv6 is more complicated than ICMPv4:
  - Some protocols that were independent in version 4 are now part of ICMPv6
  - Some new messages have been added to make it more useful:
    - Error-Reporting Messages: e.g., destination unreachable, packet too big, time exceeded, and parameter problems
    - Neighbor-Discovery Messages
    - Group Membership Messages

Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
  - Transition from IPv4 To IPv6
- Sidebar: Network Neutrality

IPv6 Adoption

- Percentage of users that access Google over IPv6

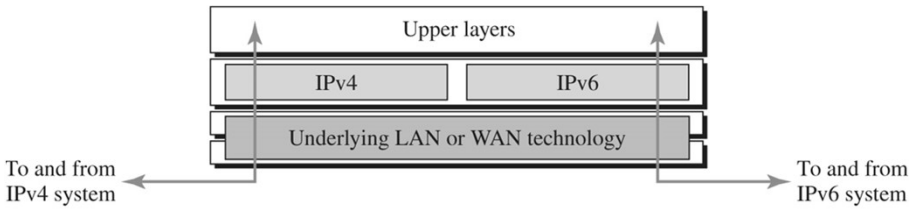


Transition from IPv4 To IPv6

- Transition must be smooth to prevent any problems between IPv4 and IPv6 systems
  - Not all routers can be upgraded simultaneously
- Three strategies have been devised for transition:
  - Dual stack
  - Tunneling
  - Header translation
- One or all of these three strategies can be implemented during the transition period

Dual Stack

- It is recommended that all hosts, before migrating completely to version 6, have a dual stack of protocols during the transition
- In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6



Example: Dual Stack

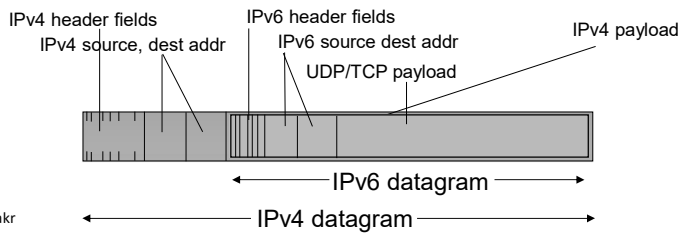
Use ipconfig in Windows and show your IPv4 and IPv6 link control addresses.

Solution:

```
C:\Users\admin>ipconfig /all

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : home
   Description . . . . . . . . . . . : Killer(R) Wi-Fi 6 AX1650s 160MHz Wireless Network Adapter
(201D2W)
   Physical Address. . . . . . . . . : 2C-6D-B2-65-18-8C
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : FE80::F7A9:23FF:FE11:9264%13(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.2.24(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Tuesday, August 23, 2022 9:50:17 PM
   Lease Expires . . . . . . . . . . : Tuesday, August 30, 2022 12:43:37 PM
```

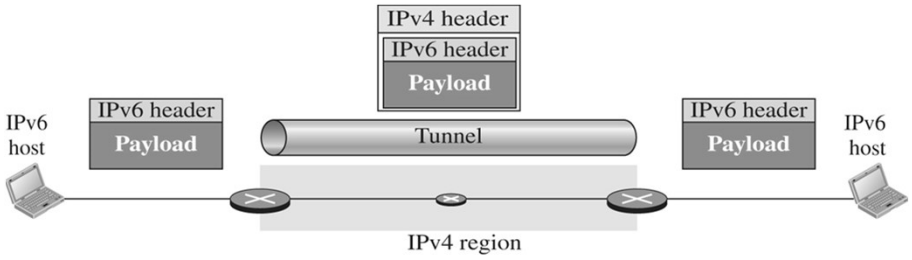

Tunneling

- Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4
- To pass through this region, the packet must have an IPv4 address
- So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region
  - IPv6 datagram carried as payload in IPv4 datagram among IPv4 routers



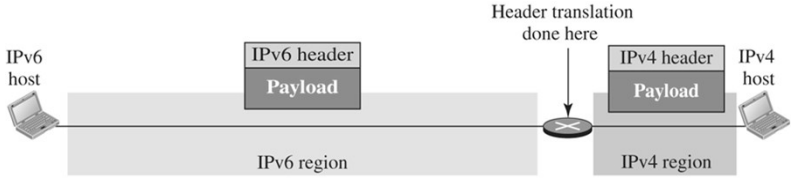
Tunneling (cont.)

- IPv6 packet goes through a tunnel at one end and emerges at the other end
- To make it clear that the IPv4 packet is carrying an IPv6 packet as data, the protocol value is set to 41
- Tunneling used extensively in other contexts (4G/5G)



Header Translation

- Header translation is converting header of an IPv6 packet to an IPv4 header
- Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4
  - The sender wants to use IPv6, but the receiver does not understand IPv6
- Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver
- In this case, the header format must be totally changed through header translation



Outline

- Communication at the network layer
- Packet-switched networks
- Internet Protocol (IP)
- Sidebar: Network Neutrality

Sidebar: Network Neutrality

- What is network neutrality?
  - ISP has to provide access to all sites, content, and applications at the same speed, under the same conditions without blocking or giving preference to any content [Wikipedia]
    - protects free speech
    - encourages innovation and competition
- Clear, Bright-Line Rules: [Order on Protecting and Promoting an Open Internet by FCC]
  - no blocking … *shall not block lawful content, applications, services, or non-harmful devices, subject to reasonable network management*
  - no throttling … *shall not impair or degrade lawful Internet traffic on the basis of Internet content, application, or service, or use of a non-harmful device, subject to reasonable network management*
  - no paid prioritization … *shall not engage in paid prioritization*

Summary

- We covered:
  - Packet-switched networks
  - IPv4, IPv6, DHCP, NAT, ICMP
  - Forwarding of IP packets