

# BUSINESS DATA COMMUNICATIONS & NETWORKING

## Chapter 5

### Network and Transport Layers

FitzGerald • Dennis • Durcikova

# Outline

- Transport Layer Protocols
- Network Layer Protocols
- Transport Layer Functions
  - Linking to the application layer
  - Segmenting
  - Session Management
- Network Layer Functions
  - Addressing
  - Routing
- TCP/IP Examples
- Implications for Management

# Network and Transport Layers

- Transport Layer
  - Layer 4 in the Internet model
  - Links application and network layers
  - Responsible for segmentation and reassembly
  - Session management
  - Responsible for end-to-end delivery of messages
- Network Layer
  - Layer 3 in the Internet model
  - Responsible for addressing and routing of messages

## Internet Model

Application

Transport

Network

Data Link

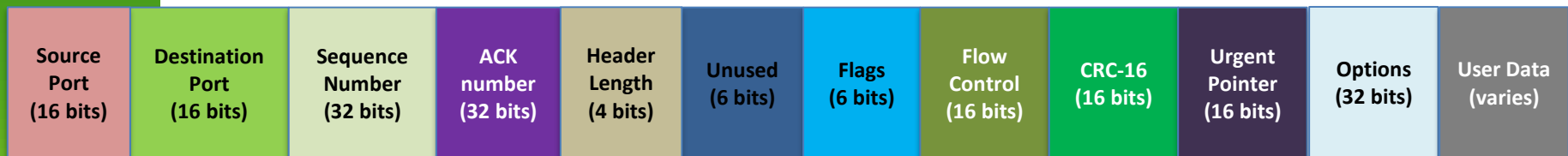
Physical

# Protocols

- TCP/IP
  - Originally developed as a single internetworking protocol by Vint Cerf and Bob Kahn in 1974
  - Later divided into the TCP and IP protocols
  - Most common protocols of the Internet and in LANs, WANs, and backbone networks

# Transport Layer Protocols

- Transmission Control Protocol (TCP)
  - Most common transport layer protocol
  - PDU called a segment
  - Used for reliable transmission of data
  - 160 - 192 bits (20 -24 bytes) of overhead
    - Options field is not required



# Transport Layer Protocols

- User Datagram Protocol (UDP)
  - Operates at the transport layer
  - PDU called a segment
  - Used in time-sensitive situations, for control messages, or when reliability is handled by the application layer
  - 32-64 bits (4-8 bytes) of overhead
    - Source port is optional in IPv4 and IPv6, Checksum is optional in IPv4



# Network Layer Protocols

- Internet Protocol (IP)
  - IP version 4 (IPv4)
    - Most common version of IP used
    - 32-bit addresses ( $2^{32}$  or ~4.29 billion possible)
    - Exhaustion of address space
  - IP version 6 (IPv6)
    - 128-bit addresses ( $2^{128}$  or  $\sim 3.4 \times 10^{38}$  possible)
    - Slowly being adopted due to IPv4 exhaustion

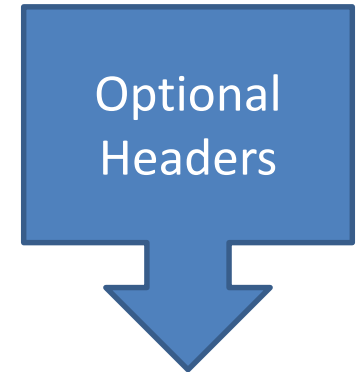
# Network Protocols

- IPv4 Packet
  - 160-192 bits (20-24 bytes) of overhead
  - Options field rarely used

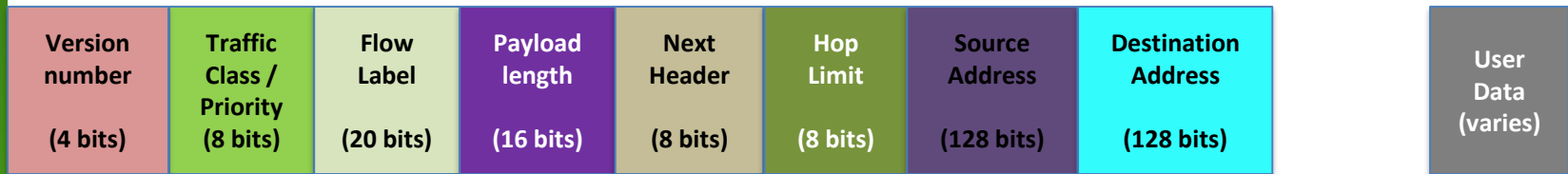
|                            |                           |                             |                           |                  |                   |                            |                                      |                      |                     |                             |                                  |                      |                       |
|----------------------------|---------------------------|-----------------------------|---------------------------|------------------|-------------------|----------------------------|--------------------------------------|----------------------|---------------------|-----------------------------|----------------------------------|----------------------|-----------------------|
| Version number<br>(4 bits) | Header length<br>(4 bits) | Type of service<br>(8 bits) | Total length<br>(16 bits) | IDs<br>(16 bits) | Flags<br>(3 bits) | Packet Offset<br>(13 bits) | Time to Live / Hop Limit<br>(8 bits) | Protocol<br>(8 bits) | CRC-16<br>(16 bits) | Source Address<br>(32 bits) | Destination Address<br>(32 bits) | Options<br>(32 bits) | User Data<br>(varies) |
|----------------------------|---------------------------|-----------------------------|---------------------------|------------------|-------------------|----------------------------|--------------------------------------|----------------------|---------------------|-----------------------------|----------------------------------|----------------------|-----------------------|



# Network Protocols



- IPv6 Packet
  - Fixed Header
  - 320 bits (40 bytes) of overhead



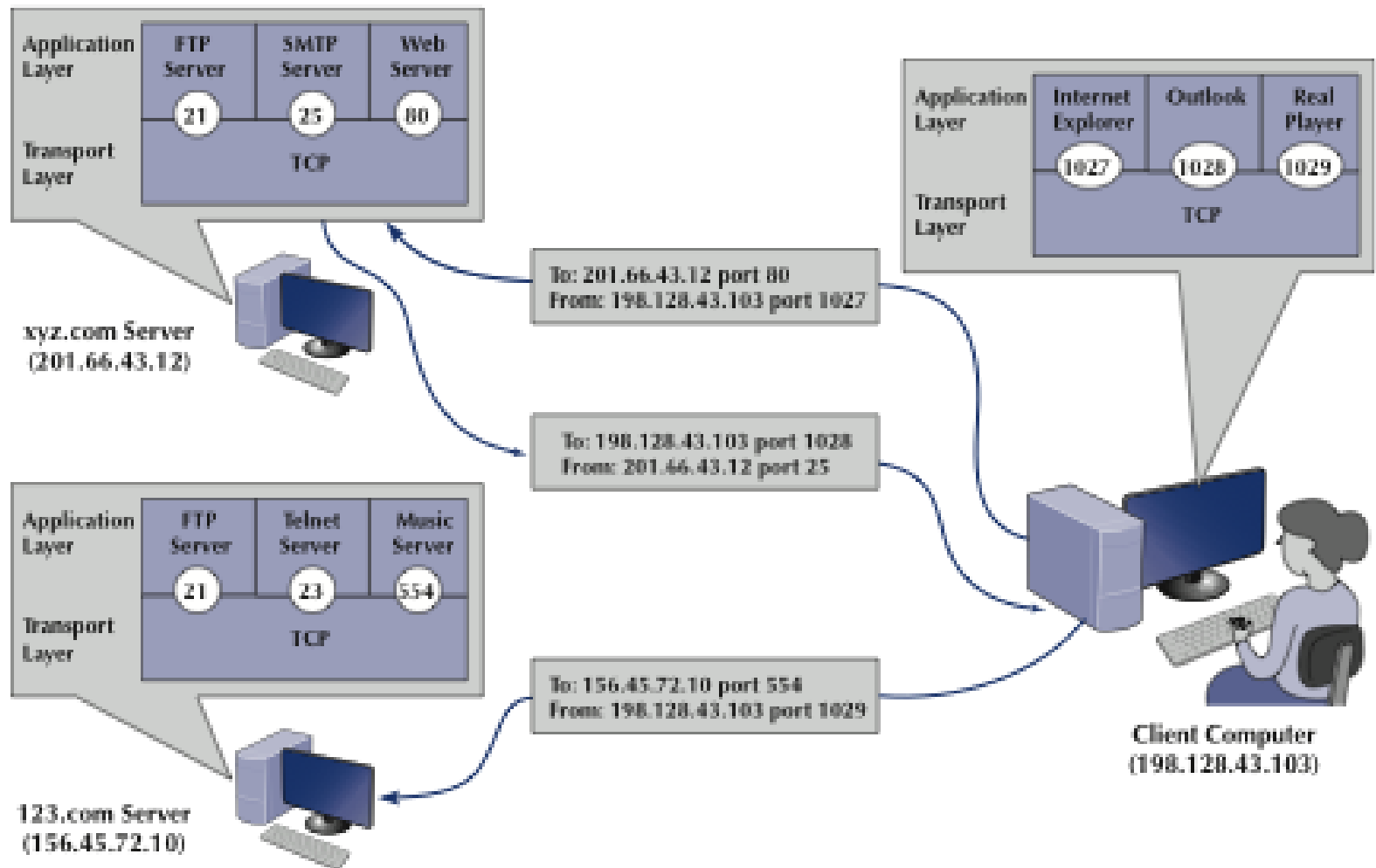
## Optional Headers

- Hop-by hop options
- Destination options (with routing options)
- Routing
- Fragment
- Authentication
- Encapsulation Security Payload
- Destination options
- Mobility

# Transport Layer Functions

1. Linking to the application layer
  - TCP/UDP may serve multiple application layer protocols
  - **Ports** used to identify application (2-byte numbers)
  - Many source/destination ports follow standards
  - Common port standards
    - HTTP: TCP port 80
    - HTTPS: TCP port 443
    - FTP: TCP ports 20 and 21
    - SMTP: TCP port 25
    - IMAP: TCP port 143
    - POP3: TCP port 110 (more commonly TCP port 995 secure version)
    - DNS: TCP or UDP port 53 (most commonly UDP)

# Transport Layer Functions



**FIGURE 5-5** Linking to application layer services

# Transport Layer Functions

## 2. Segmenting

- Breaking up large files into smaller segments (and putting them back together)
- Segments may be passed individually to application layer or after reassembly
- How large are the segments?
  - Size depends on the network and data link layer protocols
  - Maximum Segment Size (MSS) is negotiated during TCP handshake
  - e.g., if the maximum size of the data in an Ethernet frame is 1,500 bytes and TCP and IP use 20 byte headers, the maximum segment size is 1460 bytes

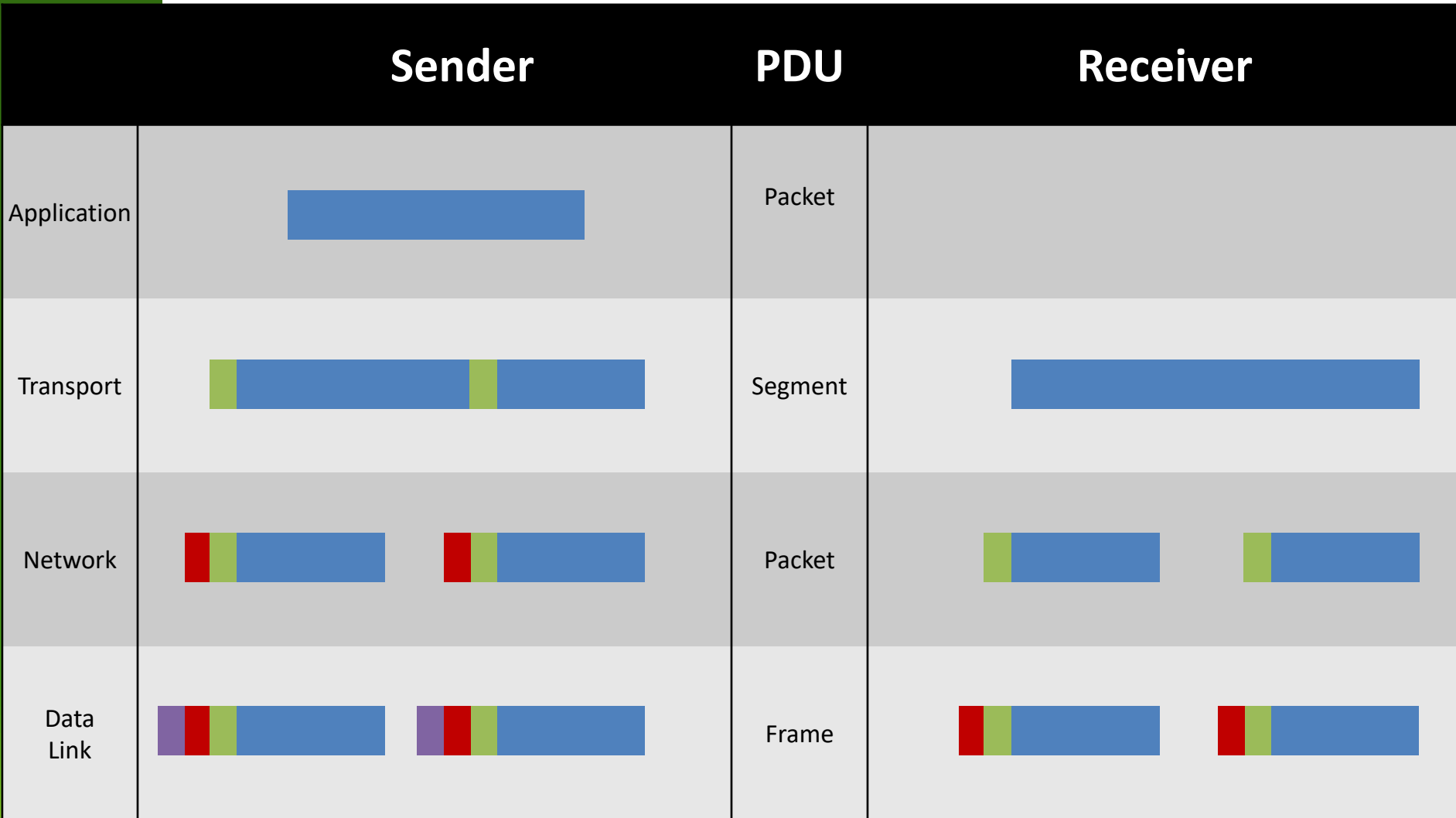
**Ethernet Frame Data Size** →  $1500 - 20 - 20 = 1460$  bytes

↑ **TCP header**

↓ **IPv4 header**

The diagram illustrates the calculation of the Maximum Segment Size (MSS). It shows the equation  $1500 - 20 - 20 = 1460$  bytes. An arrow points from the text 'Ethernet Frame Data Size' to the number 1500. Another arrow points from the text 'TCP header' to the first 20. A third arrow points from the text 'IPv4 header' to the second 20.

# Transport Layer Functions



# Transport Layer Functions

## 3. Session management

- A session can be thought of as a conversation between two computers or creating a virtual circuit
- Using a session to send data is also called **connection-oriented** messaging (TCP)
- Sending messages without establishing a session is **connectionless** messaging (UDP)
- TCP connections are opened using a three-way handshake
  - SYN
  - SYN-ACK
  - ACK
- Sessions provide reliable end-to-end connections

# Network Layer Functions

- Addressing
  - Used to direct messages from source to destination
  - Addresses are assigned in various ways (e.g., by system administrators, ICANN, hardware vendors, etc.)
  - Addresses exist at different layers
  - Addresses may be translated (resolved) from one layer to another (e.g., DNS, ARP)

| Address Type      | Example                        | Example Address             |
|-------------------|--------------------------------|-----------------------------|
| Application layer | Uniform Resource Locator (URL) | www.indiana.edu             |
| Network layer     | IP address                     | 129.79.78.193 (4 bytes)     |
| Data link layer   | MAC address                    | 1C-6F-65-F8-33-8A (6 bytes) |

# Network Layer Functions

- Addressing
  - IPv4 addresses are 32 bits
  - Most common way to write is using dot-decimal notation
    - Easier for people to read and remember
    - Breaks the address into four bytes and writes each byte in decimal notation instead of binary
    - Example: 129.79.78.193

10000001

01001111

01001110

11000001



# Network Layer Functions

- Addressing
  - A portion of an IP address represents the network and the rest identifies the host
  - Classful addressing
    - Uses the first bits to determine number of hosts
    - Discontinued, but nomenclature still used
  - Classless Inter-Domain Routing (CIDR)
    - Uses subnet masks to more flexibly divide address space into subnets
      - IP address: **129.79.78.193**
      - Subnet Mask: **255.255.255.0**

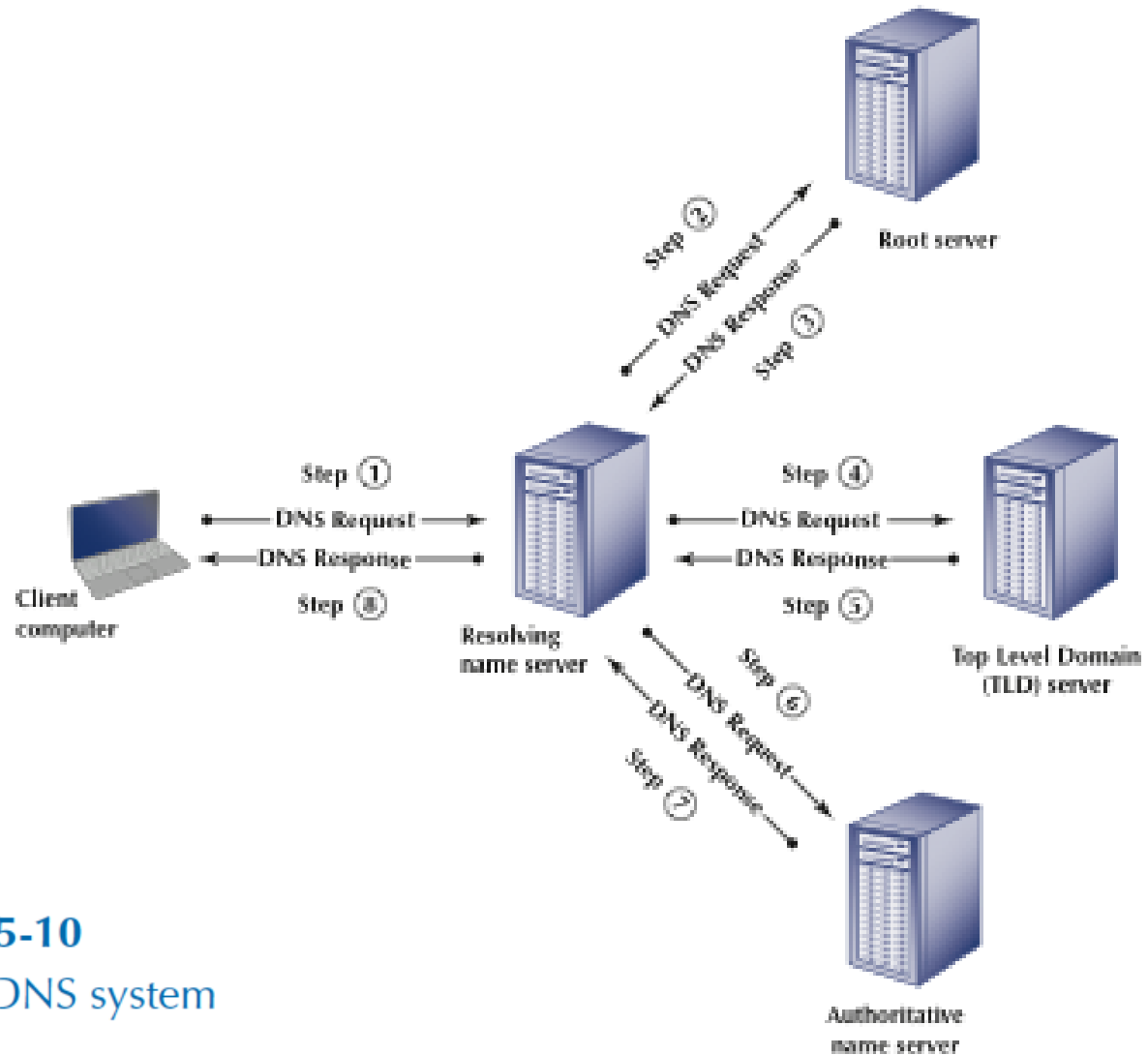
# Network Layer Functions

- Dynamic addressing
  - Configuring each device manually is time consuming
  - Assigning addresses permanently can be inefficient when devices are not connected to network
  - A server can supply IP addresses automatically
  - Dynamic Host Configuration Protocol (**DHCP**)
    - Most common protocol for dynamic addressing
    - Device sends out broadcast message
    - DHCP responds with IP settings
    - Addresses are “leased” for a length of time

# Network Layer Functions

- Address resolution
  - Host (server) name resolution
    - Translate host name to IP address
    - e.g., `www.indiana.edu` → `129.79.78.193`
    - Domain Name Service (**DNS**)
  - MAC address resolution
    - Identify MAC address of the next device in the circuit
    - Address Resolution Protocol (**ARP**)

# Network Layer Functions



**FIGURE 5-10**  
How the DNS system works

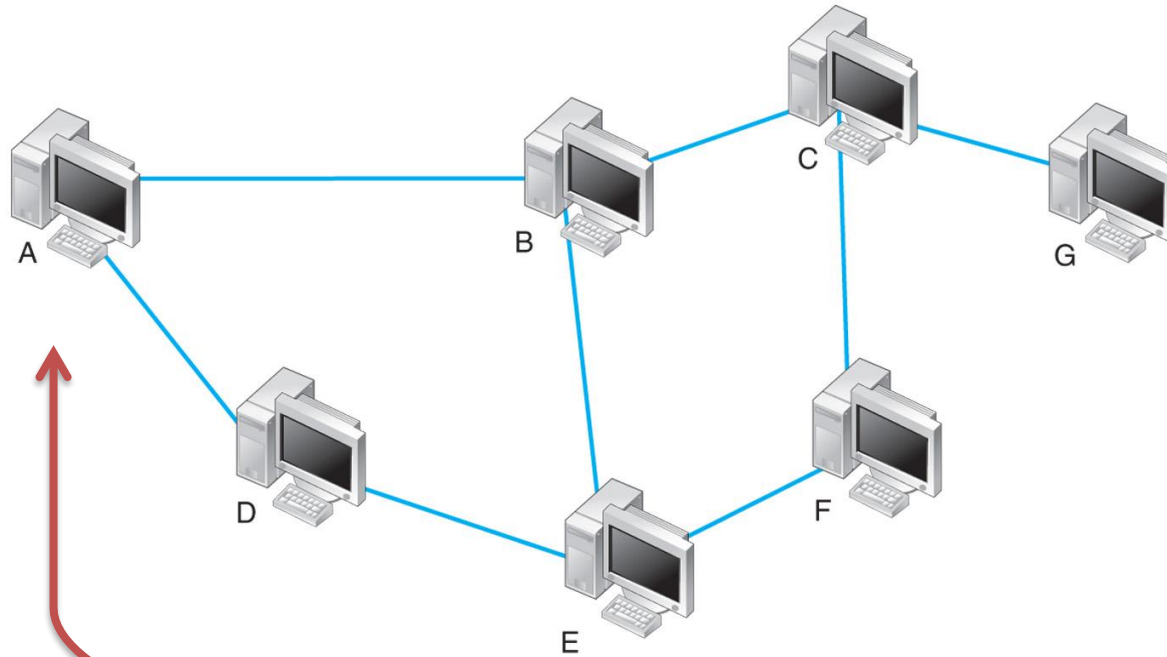
# Network Layer Functions

- Routing
  - Process of identifying what path to have a packet take through a network from sender to receiver
  - Routing Tables
    - Used to make routing decisions
    - Shows which path to send packets on to reach a given destination
    - Kept by computers making routing decisions
  - Routers
    - Special purpose devices used to handle routing decisions on the Internet
    - Maintain their own routing tables

| Dest. | Next |
|-------|------|
| B     | B    |
| C     | B    |
| D     | D    |
| E     | D    |
| F     | D    |
| G     | B    |

# Routing

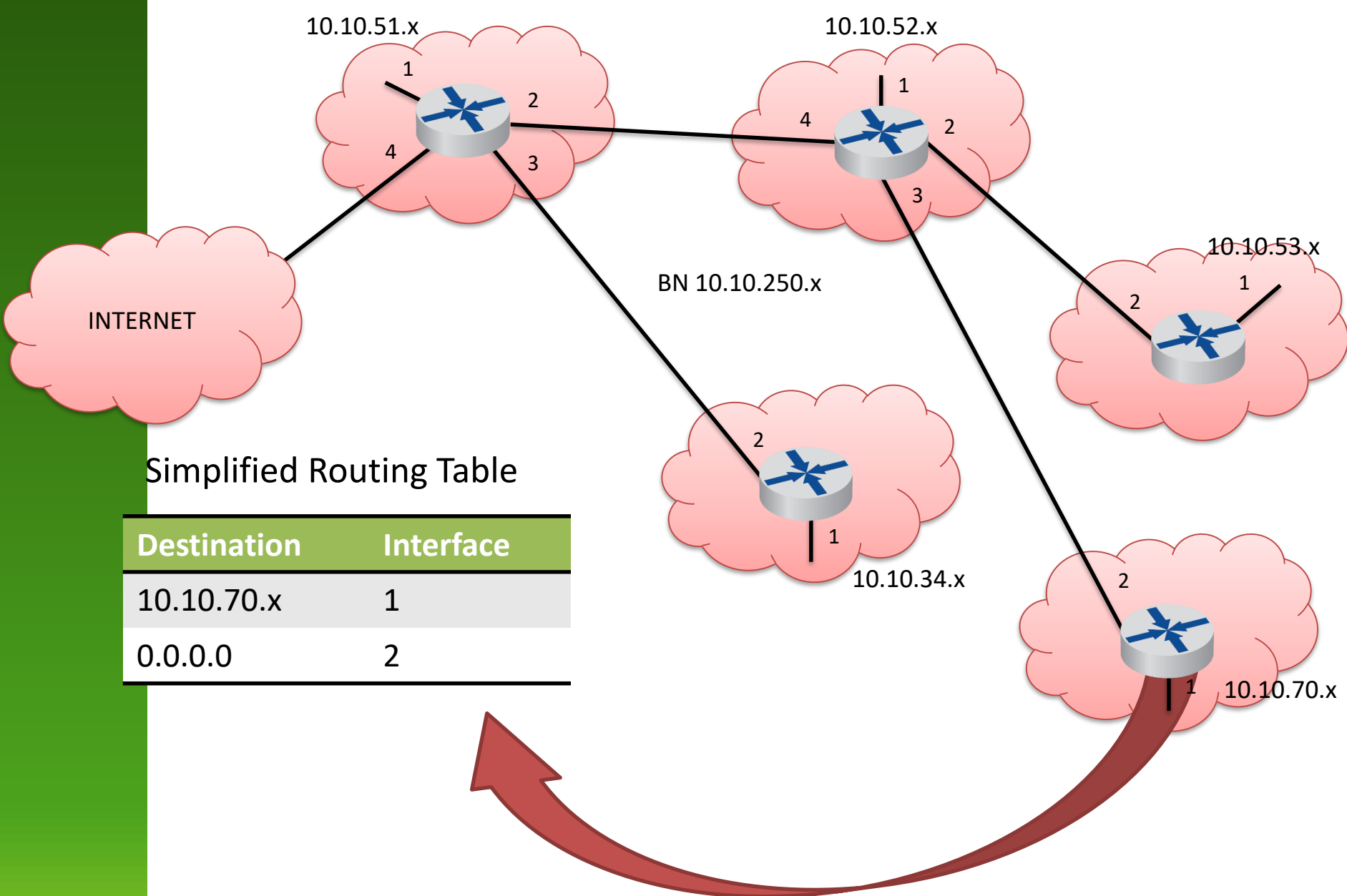
What are the possible paths from A to G?



- **ABCG**
- **ABEFCG**
- **ADEFCG**
- **ADEBCG**

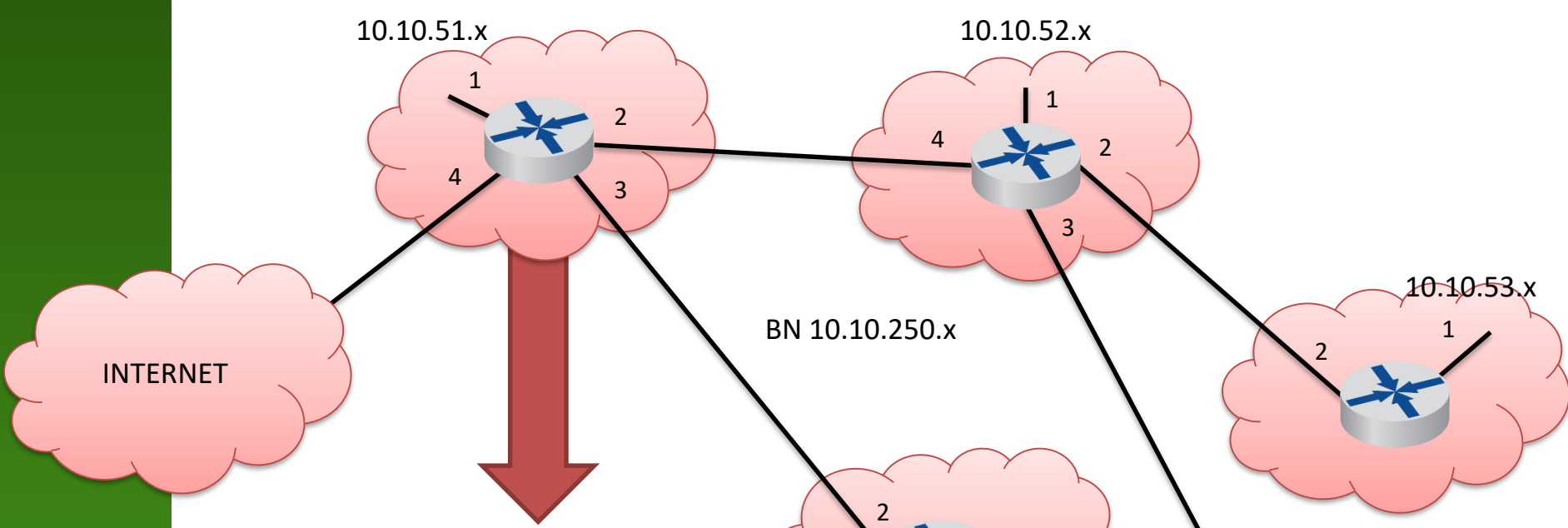
## Simplified Routing Table for A

| Dest. | Next |
|-------|------|
| B     | B    |
| C     | B    |
| D     | D    |
| E     | D    |
| F     | D    |
| G     | B    |



Simplified Routing Table

| Destination | Interface |
|-------------|-----------|
| 10.10.70.x  | 1         |
| 0.0.0.0     | 2         |



**Simplified Routing Table**

| Destination  | Interface |
|--------------|-----------|
| 10.10.51.x   | 1         |
| 10.10.52.x   | 2         |
| 10.10.34.x   | 3         |
| 10.10.53.x   | 2         |
| 10.10.70.x   | 2         |
| 10.10.250.34 | 3         |
| 10.10.250.x  | 2         |
| 0.0.0.0      | 4         |



# Routing

- **Centralized Routing**
  - Routing decisions made by one computer
  - Not common anymore
- **Decentralized Routing**
  - Decisions made by each node independently of one another
  - Information needs to be exchanged to prepare routing tables
  - Used by the Internet

# Routing

- **Static**
  - Fixed routing tables
  - Manually configured by network managers
  - Local adjustments when computers added or removed
- **Dynamic**
  - Routing tables updated periodically
  - Routers exchange information using **protocols** to update tables

# Routing

- Dynamic routing algorithms
  - **Distance vector:** based on the number of “hops” between two devices
  - **Link state:** based on the number of hops, circuit speed, and traffic congestion
    - Provides more reliable, up to date paths to destinations

# Routing Protocols

- **Routing Information Protocol (RIP)**
  - Dynamic distance vector protocol used for interior routing
  - Operation
    - Network manager builds the routing table
    - Routing tables broadcast periodically (e.g., every minute or so)
    - When new computers are added, router counts “hops” and selects the shortest route
  - Useful in smaller, less complex networks

# Routing Protocols

- **Open Shortest Path First (OSPF)**
  - Dynamic link state protocol used for interior routing
  - Most widely used interior routing protocol on large enterprise networks
  - More reliable paths
  - Less burdensome to the network because only updates sent

# Routing Protocols

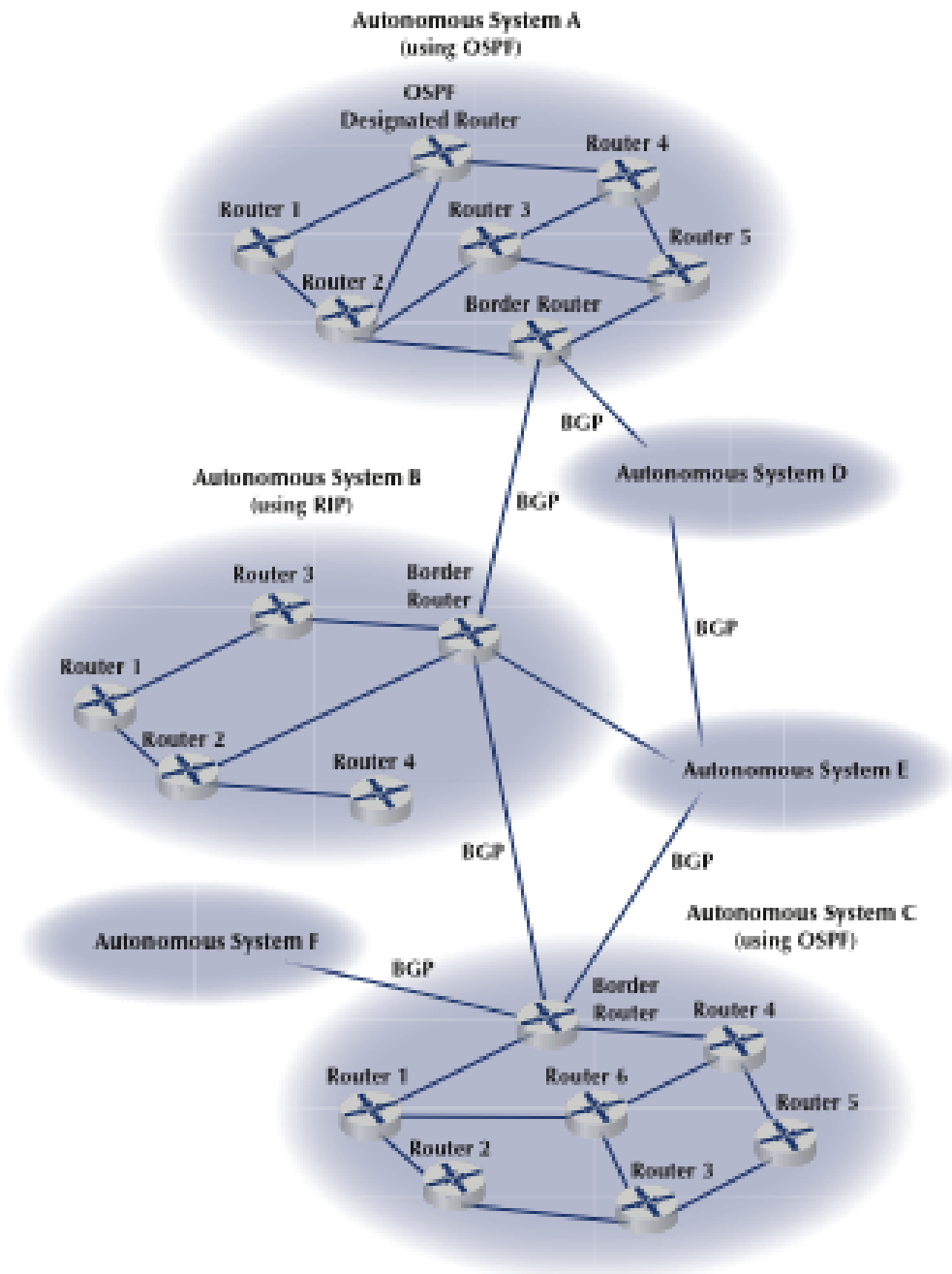
- **Enhanced Interior Gateway Routing Protocol (EIGRP)**
  - A dynamic link state protocol (developed by Cisco)
  - Records transmission capacity, delay time, reliability and load for all paths
  - Keeps the routing tables for its neighbors and uses this information in its routing decisions as well

# Routing Protocols

- If each network uses a different protocol internally, how are they able to communicate?
- **Border Gateway Protocol (BGP)**
  - Dynamic distance vector protocol used for exterior routing
  - Far more complex than interior routing protocols
  - Provide routing info only on selected routes (e.g., preferred or best route)

**FIGURE 5-13**

Routing on the Internet with Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), and Routing Information Protocol (RIP)





# Multicasting

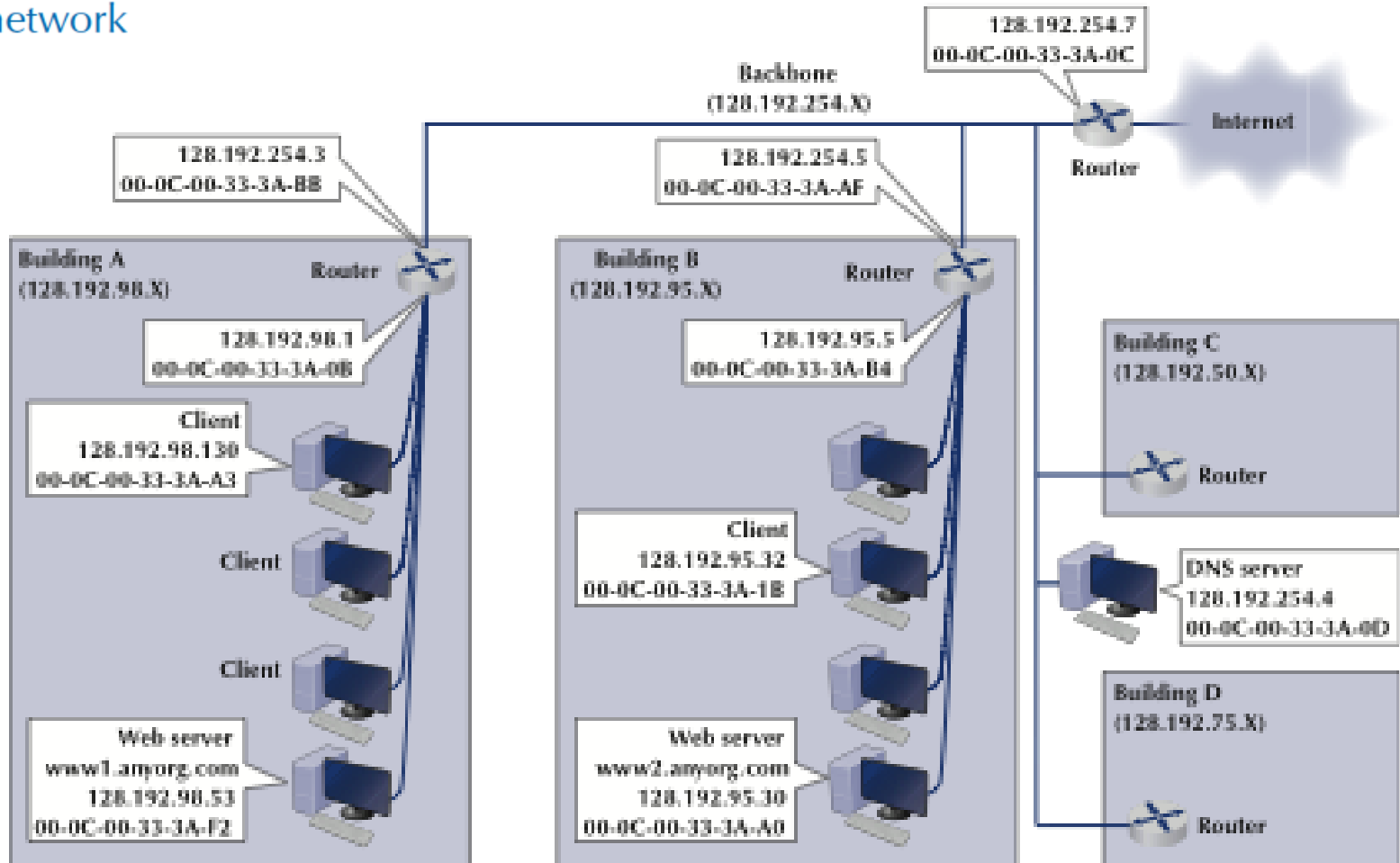
- **Unicast** - one computer to another computer
- **Broadcast** - one computer to all computers in the network
- **Multicast** - one computer to a group of computers (e.g., videoconference)
  - Same data needs to reach multiple receivers and avoid transmitting it once for each receiver
    - Particularly useful if access link has bandwidth limitations
    - Many implementations at different layers
    - In IP multicast, hosts dynamically join and leave multicast groups using Internet Group Management Protocol (IGMP)

# TCP/IP Example

- Required network addressing information:
  1. Device's own IP address
  2. Subnet mask
  3. IP address of default gateway (most commonly the router)
  4. IP address of at least one DNS server
- Obtained from a configuration file or DHCP

# Known Addresses, Same Subnet

**FIGURE 5-15** Example Transmission Control Protocol/Internet Protocol (TCP/IP) network



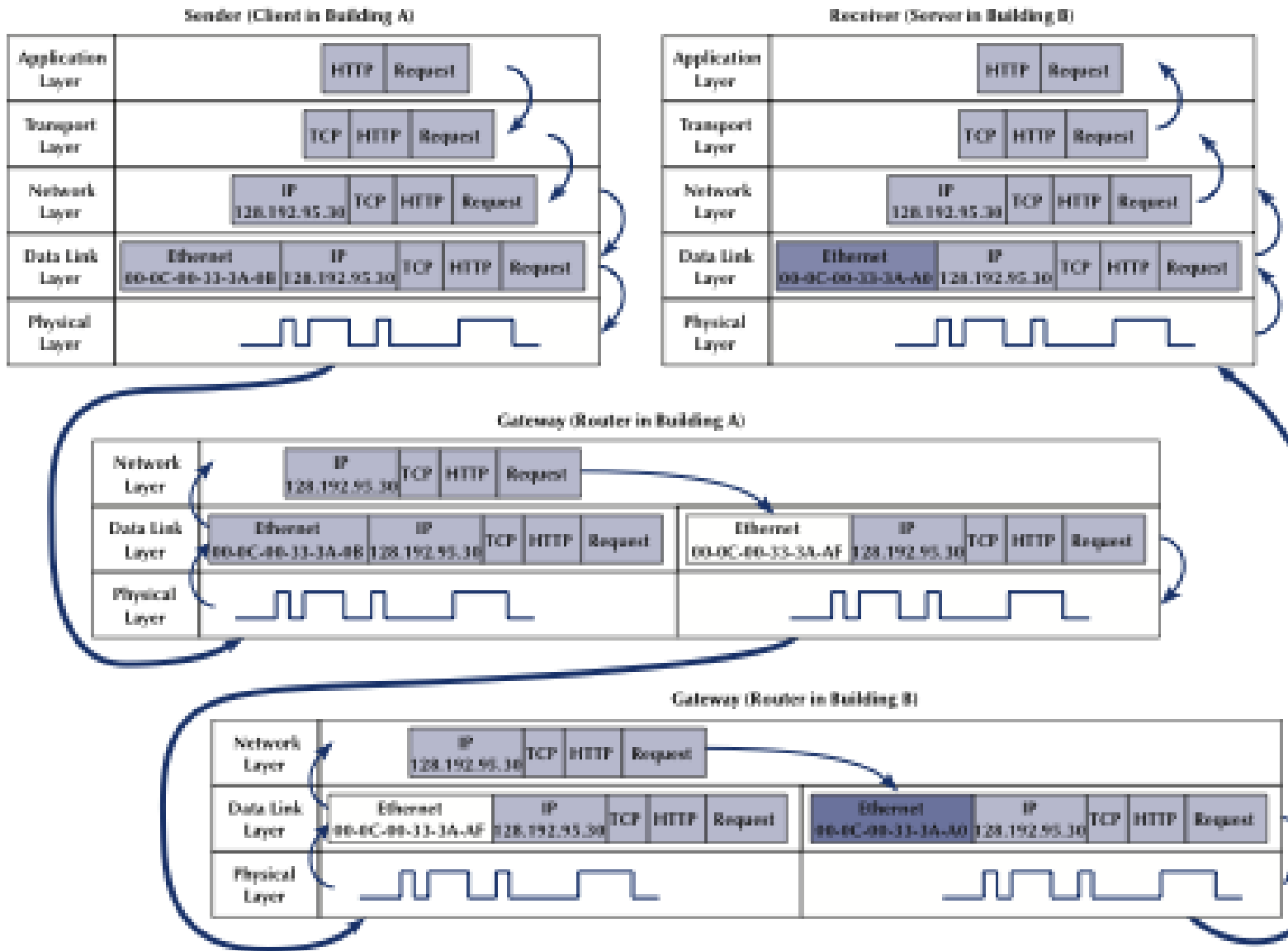
- Suppose we have an HTTP request from Client in building A to Server in building B.

# TCP/IP Examples

1. A Client (128.192.98.130) requests a Web page from a server (www1.anyorg.com)
  - Client knows the server's IP
2. A Client (128.192.98.130) requests a Web page from a server (www2.anyorg.com) on a different subnet
  - Client knows the server's IP
3. A Client (128.192.98.130) requests a Web page from a server (www1.anyorg.com)
  - Client does not know server's IP

# TCP/IP and Layers

- Host Computers
  - Packets move through all layers
- Gateways, Routers
  - Packet moves from Physical layer to Data Link Layer through the network Layer
- At each stop along the way
  - Ethernet packets is removed and a new one is created for the next node
  - IP and above packets never change in transit (created by the original sender and destroyed by the final receiver)



**FIGURE 5-18** How messages move through the network layers.  
*Note:* The addresses in this example are destination addresses

# Implications for Management

- Organizations standardizing on TCP/IP
  - Decreases costs of equipment and training
  - Network providers are also moving towards standardization
- Slow transition to IPv6