Business Data Communications and Networking

11th Edition

Jerry Fitzgerald and Alan Dennis

John Wiley & Sons, Inc

Dwayne Whitten, D.B.A
Mays Business School
Texas A&M University

Copyright 2011 John Wiley & Sons, Inc

Chapter 5

Network and Transport Layers

Copyright 2011 John Wiley & Sons, Inc

Chapter 5 Outline

5.1 - Introduction

5.2 - Transport & Network Layer Protocols

- **TCP/IP**

5.3 Transport Layer Functions

- **Linking to the Application Layer**
- **Segmenting**
- **Session management**

5.4 - Addressing

- **Assigning addresses and address resolution**

5.5 - Routing

- **Types of routing, routing protocols, multicasting, and router anatomy**

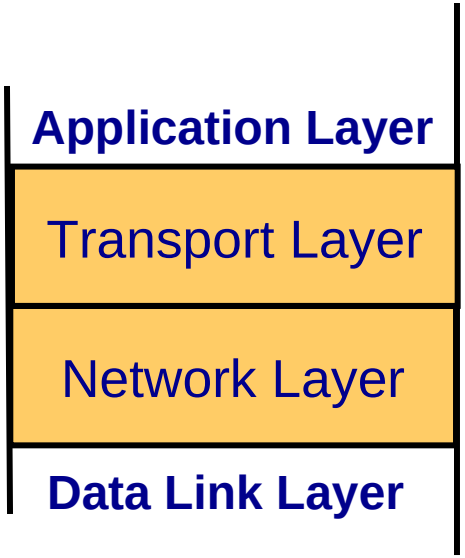
5.6 - TCP/IP Example

5.7 – Implications for Management

Copyright 2011 John Wiley & Sons, Inc

5.1 Introduction

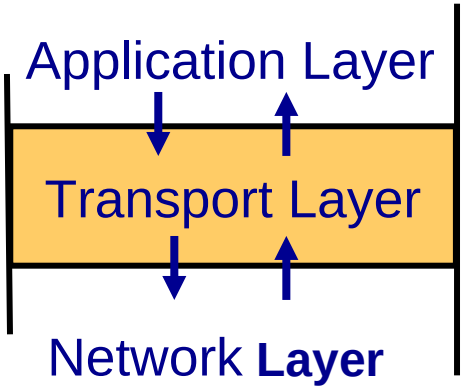
- **Transport and Network layers**
  - **Responsible for moving messages from end-to-end in a network**
  - **Closely tied together**
  - **TCP/IP: most commonly used protocol**
    - **Used in Internet**
    - **Compatible with a variety of Application Layer protocols as well as with many Data Link Layer protocols**
      - **Email, ftp etc**



Copyright 2011 John Wiley & Sons, Inc

Introduction - Transport layer

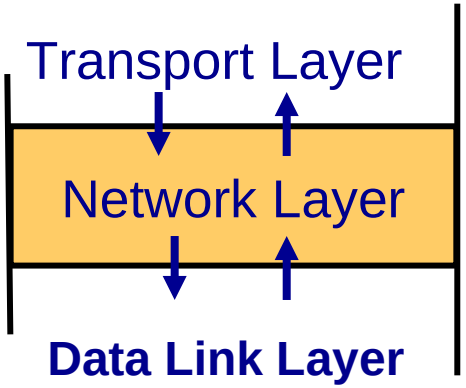
- **Responsible for end-to-end delivery of messages**
  - **Sets up virtual circuits (when needed)**
- **Responsible for segmentation and reassembly**
  - **Breaking the message into several smaller pieces at the sending end**
  - **Reconstructing the original message into a single whole at the receiving end**
- **Interacts with Application Layer**



Copyright 2011 John Wiley & Sons, Inc

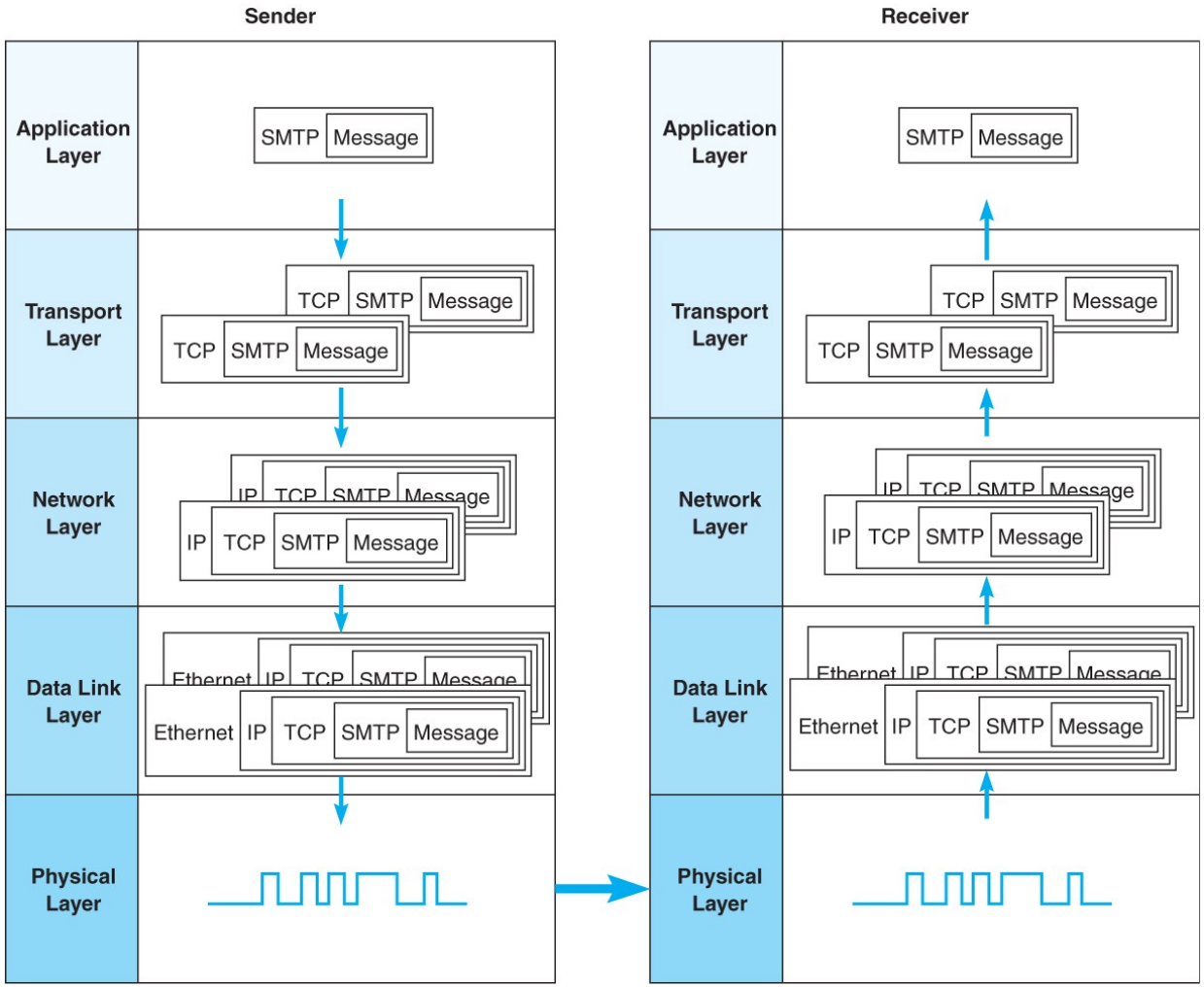
Introduction – Network Layer

- **Responsible for addressing and routing of messages**
  - **Selects the best path from computer computer until the message reaches destination**
- **Performs encapsulation on sending end**
  - **Adds network layer header to message segments**
- **Performs decapsulation on receiving end**
  - **Removes the network layer header at receiving end and passes them up to the transport layer**



Copyright 2011 John Wiley & Sons, Inc

TCP/IP's 5-Layer Network Model



Copyright 2011 John Wiley & Sons, Inc

5.2 Transport/Network Layer Protocols

- TCP/IP (Transmission Control Protocol / Internet Protocol)
  - Most common, used by all Internet equipment
- Developed in 1974 by V. Cerf and B. Kahn
  - As part of Arpanet (U.S. Department of Defense)
- Most common protocol suite
  - Used by the Internet
  - Largest percentage of all backbone, metropolitan, and wide area networks use TCP/IP
  - Most commonly used protocol on LANs
- Reasonably efficient and error free transmission
  - Performs error checking
  - Transmits large files with end-to-end delivery assurance
  - Compatible with a variety of data link layer protocols

Copyright 2011 John Wiley & Sons, Inc

Transmission Control Protocol

- **Links the application layer to the network layer**
- **Performs packetization and reassembly**
  - **Breaks up a large message into smaller packets**
  - **Numbers the packets**
  - **Reassembles the packets at the destination end**
- **Ensures reliable delivery of packets**

Source port	Destination port	Sequence number	ACK number	Header length	Unused	Flags	Flow control	CRC–16	Urgent pointer	Options	User data
16 bits	16 bits	32 bits	32 bits	4 bits	6 bits	6 bits	16 bits	16 bits	16 bits	32 bits	Varies

Transmission Control Protocol

Sequence Number: 32 bits

•The client on either side of a TCP session maintains a 32-bit *sequence number* it uses to keep track of how much data it has sent.

•This sequence number is included on each transmitted packet, and acknowledged by the opposite host as an *acknowledgement number* to inform the sending host that the transmitted data was received successfully.

•When a host initiates a TCP session, its initial sequence number is effectively random; it may be any value between 0 and 4,294,967,295, inclusive.

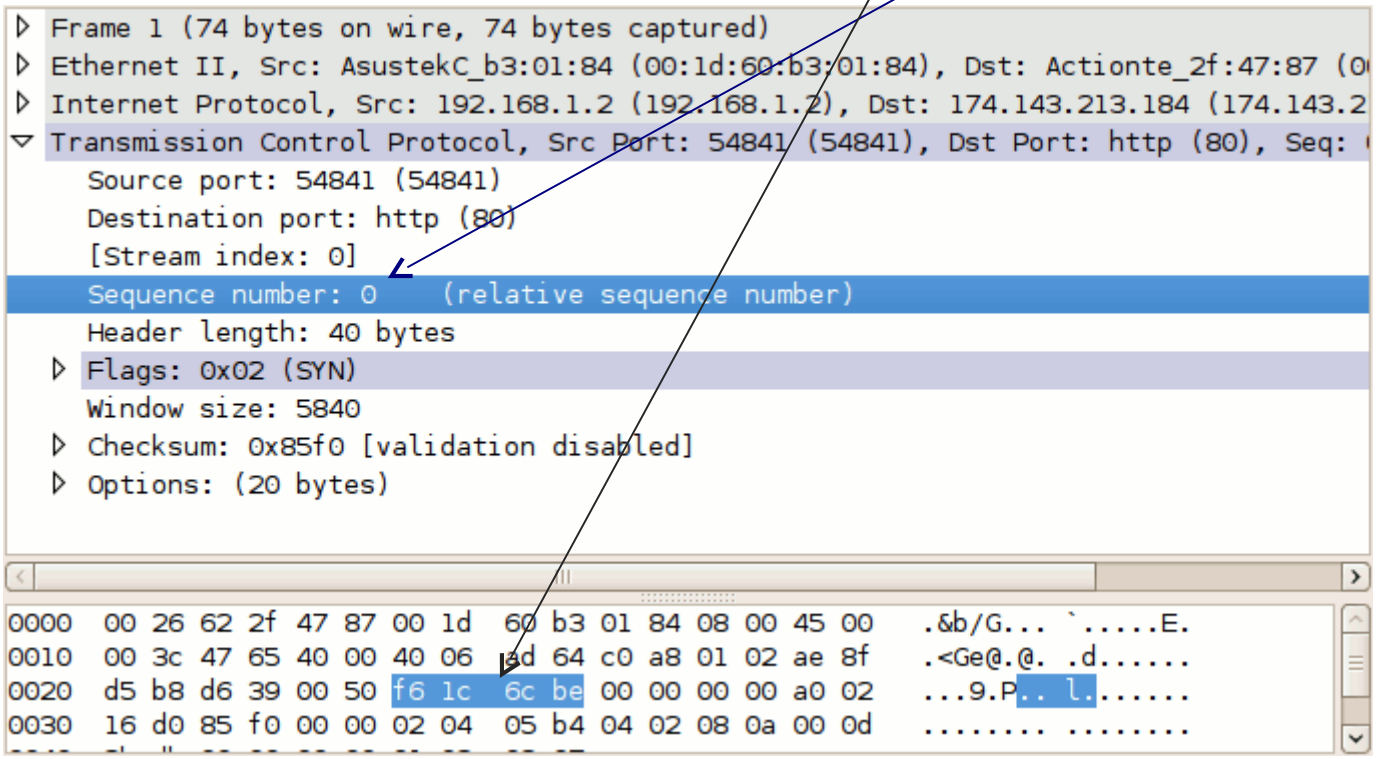
•Protocol analyzers ie. Wireshark will display *relative* sequence and ackt number in place of the field's actual value.

- These values are relative to the initial sequence no. of that stream.
- So it is much easier to keep track of relatively small, predictable numbers rather than the actual numbers sent on the wire.

For example, the initial relative sequence number shown in packet 1 is 0, the ASCII sequence number is 0xf61c6cbe, or 4129057982 decimal.

Transmission Control Protocol

(captured using wireshark) For example, the initial relative sequence number shown in packet #1 is 0 (naturally), while the ASCII decode in the third pane shows that the actual sequence number is 0xf61c6cbe, or 4129057982 decimal.



Transmission Control Protocol

- **Acknowledgment Number: 32 bits**
  If the ACK control bit is set this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established this is always sent.
- **Header: 4 bits**
  The number of 32 bit words in the TCP Header. This indicates where the data begins. The TCP header (even one including options) is an integral number of 32 bits long.
- **Unused: 6 bits**
  Reserved for future use. Must be zero.
- **Control Bits: 6 bits (from left to right):**
  URG: Urgent field; ACK: Acknowledgment field significant;
  PSH*: Push Function; RST: Reset the connection; SYN: Synchronize sequence no.; FIN: No more data sent
- **Window: 16 bits**
  The number of data octets the sender of this segment is willing to accept.
- **CRC: 16 bits**
- **Urgent Pointer**
  The Urgent Pointer is used when some information has to reach the server ASAP. ie. usually used in Telnet, where an immediate response (e.g. the echoing of characters) is desirable.

Note*: **Push function**: Asks to **push** the buffered data to the receiving application.

Transmission Control Protocol

- Options

When establishing a connection the following sequence shows the process of a TCP connection being established:

Frame 1:
As you see in the first frame, the client, NTW3, sends a SYN segment (TCP ....S.).

- This is a request to the server to synchronize the sequence numbers.
- It specifies its initial sequence number (ISN), which is incremented by 1, 8221821+1=8221822, and that is sent to the server.
- To initialize a connection, the client and server must synchronize each other's sequence numbers.
- There is also an **option** for the Maximum Segment Size (MSS) to be set, which is defined by the length (len: 4). This option communicates the maximum segment size the sender wants to receive. The Acknowledgement field (ack: 0) is set to zero because this is the first part of the three-way handshake.

```
1     2.0785 NTW3 --> BDC3 TCP ....S., len: 4, seq: 8221822-8221825, ack: 0,
win: 8192, src: 1037  dst:  139 (NBT Session)  NTW3 -->  BDC3 IP

TCP: ....S., len: 4, seq: 8221822-8221825, ack: 0, win: 8192, src: 1037
dst:  139 (NBT Session)

    TCP: Source Port = 0x040D
    TCP: Destination Port = NETBIOS Session Service
    TCP: Sequence Number = 8221822 (0x7D747E)
    TCP: Acknowledgement Number = 0 (0x0)
    TCP: Data Offset = 24 (0x18)
    TCP: Reserved = 0 (0x0000)
    TCP: Flags = 0x02 : ....S.

       TCP: ..0..... = No urgent data
       TCP: ...0.... = Acknowledgement field not significant
       TCP: ....0... = No Push function
       TCP: .....0.. = No Reset
       TCP: ......1. = Synchronize sequence numbers
       TCP: .......0 = No Fin

    TCP: Window = 8192 (0x2000)
    TCP: Checksum = 0xF213
    TCP: Urgent Pointer = 0 (0x0)
    TCP: Options

          TCP: Option Kind (Maximum Segment Size) = 2 (0x2)
          TCP: Option Length = 4 (0x4)
          TCP: Option Value = 1460 (0x5B4)

    TCP: Frame Padding

00000:  02 60 8C 9E 18 8B 02 60 8C 3B 85 C1 08 00 45 00   .`.....`.;....E.
00010:  00 2C 0D 01 40 00 80 06 E1 4B 83 6B 02 D6 83 6B   .,..@....K.k...k
00020:  02 D3 04 0D 00 8B 00 7D 74 7E 00 00 00 00 60 02   .......}t~....`.
00030:  20 00 F2 13 00 00 02 04 05 B4 20 20               ..........
```

Client NW3

request to the server to synchronize the sequence numbers

Options

Synchronization
Session establishing

FRAME

Last four bytes of the packet hold the TCP Options. The first byte of the options is the option code, the second byte is length which includes the option byte in it's calculation. The last two bytes comprise the data. The option 2, which we put in the first packet we send to a server, means Maximum Segment Size. The two bytes of data hold the size of the largest segment we can handle without fragmentation.

Lets assume we've got this data to send across to the guy at the other end.

A B C D E F G H I J

Now for some reason or another, we're going to send the bytes across only four bytes at a time. In other words, this data will flow across in thee packets.

The First Packet

A B C D

The Second Packet

E F G H

The Third Packet

I J

Each alphabet is given a number and this will be it's sequence number.

A	B	C	D	E	F	G	H	I	J
1	2	3	4	5	6	7	8	9	10

In the very first packet we send across we set the four byte sequence number to 1 i.e. the number of the first byte in the packet and the acknowledgement number as 0. The acknowledgement number is 0 because we haven't received any bytes, so there so data to acknowledge.

A B C D

1 2 3 4

The computer across the wire will respond with an ACK packet (an acknowledgement packet with the ACK flag on in the TCP header) holding an acknowledgement number of 5. This means that the computer at the other end has received all the bytes till byte 4 and it's now ready for data from byte 5 onwards. Since he hasn't sent us any data yet, his sequence number will be a random value. The sequence and acknowledgement numbers are extremely important and are the only way TCP maintains a reliable connection. If these bytes are omitted, the connection will not work.

The next packet we send will have a sequence number of 5 i.e. the number of the first byte in the packet relative to the start of the data stream. The acknowledment number will be the other guys sequence number + 1.

E F G H

5 6 7 8

We will then receive an ACK with the acknowledgement number set to 9; the byte we have to start our next packet with.

We then shot off the last two bytes and wait for the ACK and when that comes, we klnow that all the bytes we've sent across has reached the computer at the other end.

I J

9 10

The Three Way Handshake

Internet Protocol (IP)

- **Responsible for addressing and routing of packets**
- **Two versions in current in use**
  - **IPv4: a 192 bit (24 byte) header, uses 32 bit addresses.**
  - **IPv6: Mainly developed to increase IP address space due to the huge growth in Internet usage (128 bit addresses)**
- **Both versions have a variable length data field**
  - **Max size depends on the data link layer protocol.**
  - **e.g., Ethernet's max message size is 1,492 bytes, so max size of TCP message field:**

1492 – 24 – 24 = 1444 bytes

TCP header **IPv4 header**

Copyright 2011 John Wiley & Sons, Inc

IP Packet Formats

IPv4 Header: 192 bits (24 bytes)

Version number	Header length	Type of service	Total length	Identifiers	Flags	Packet offset	Hop limit	Protocol	CRC 16	Source address	Destination address	Options	User data
4 bits	4 bits	8 bits	16 bits	16 bits	3 bits	13 bits	8 bits	8 bits	16 bits	32 bits	32 bits	32 bits	Varies

IPv6 Header: 320 bits (40 bytes)

Version number	Priority	Flow name	Total length	Next header	Hop limit	Source address	Destination address	User data
4 bits	4 bits	24 bits	16 bits	8 bits	8 bits	128 bits	128 bits	Varies



HOP LIMIT: The number of network segments on which the packet is allowed to travel

Copyright 2011 John Wiley & Sons, Inc

5.3 Transport Layer Functions

- **Linking to the Application Layer**
- **Segmenting**
- **Session management**
  - **Connection-oriented**
  - **Connectionless**
  - **Quality of Service (QoS)**

Copyright 2011 John Wiley & Sons, Inc

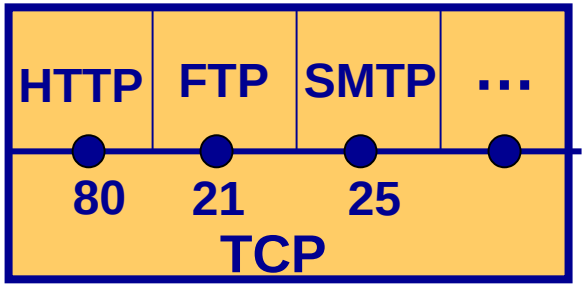
Linking to Application Layer

- **TCP may serve several Application Layer protocols at the same time**
  - **Application layer program to send a message to is determined by Port numbers located in TCP header fields; 2-byte each (source, destination)**

LINK TO PORT NUMBERS:

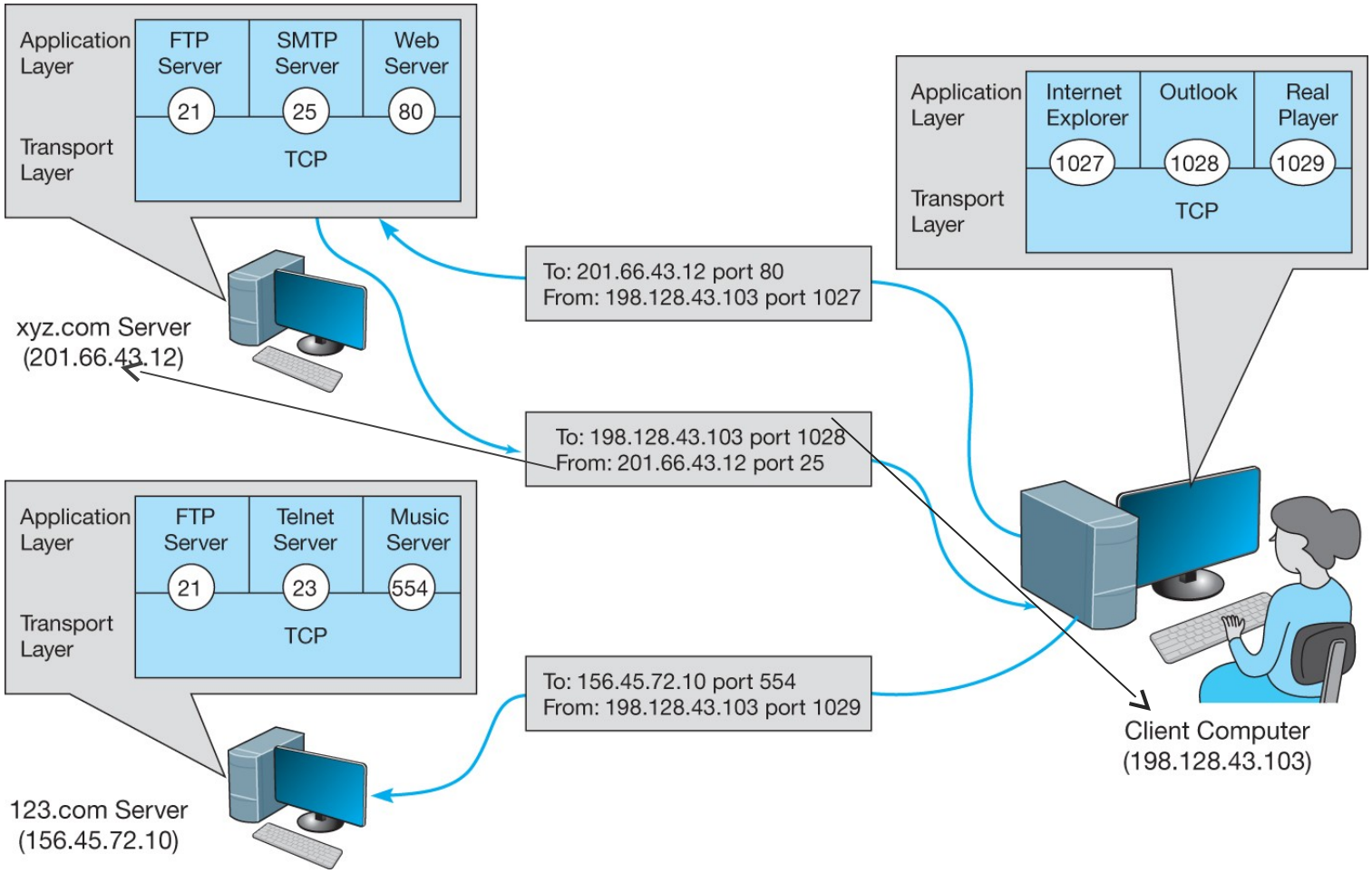
http://www.flashcardexchange.com/cards/application-layer-protocols-and-port-numbers-643487

- **Standard port numbers**
  - **Usual practice numbers**
- **Nonstandard port numbers**
  - **Possible, but requires configuration of TCP**
  - **Can be used to enhance security from commonly known ports**



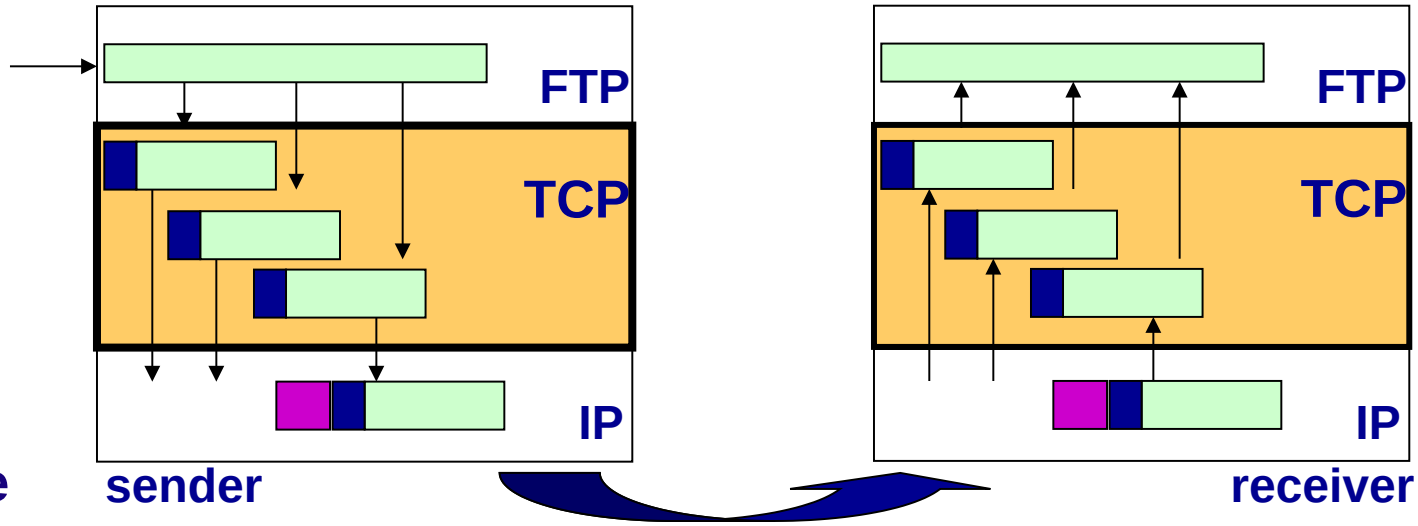
Copyright 2011 John Wiley & Sons, Inc

Application Layer Services



Packetization and Reassembly

Application layer sees message as a single block of data



TCP breaks a large message into smaller pieces (packetization)

What size packet to use? Done through negotiations

TCP puts packets back together at the destination (reassembly)

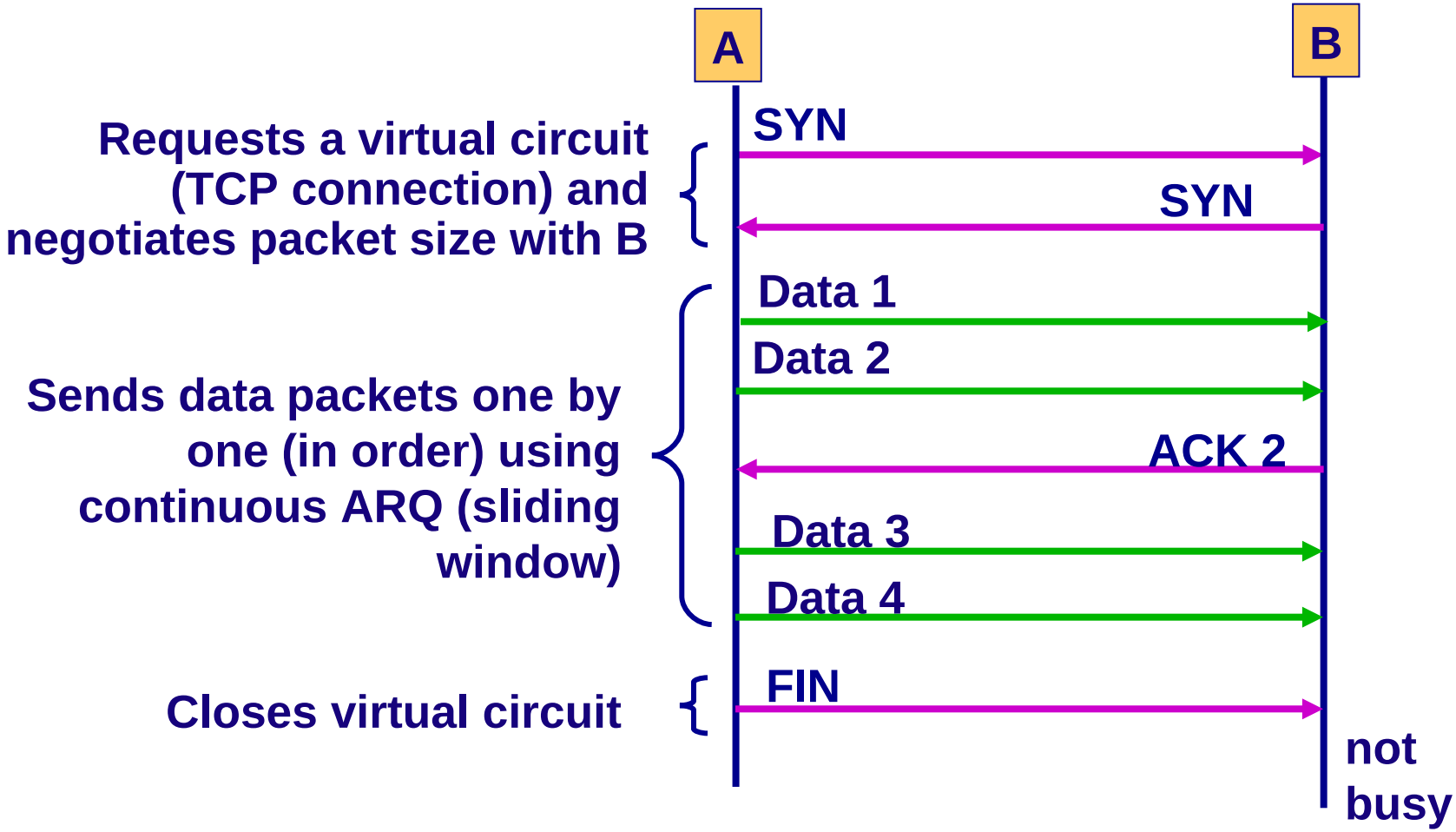
Delivers incoming packets as they arrive (e.g., Web pages) or to wait until entire message arrives (e.g., e-mail)

Copyright 2011 John Wiley & Sons, Inc

Session Management

- ***Connection Oriented*** **is provided by Transmission Control Protocol (TCP)**
  - **Setting up a virtual circuit, or a TCP connection**
    - **TCP asks IP to route all packets in a message by using the same path (from source to destination)**
    - **Packet deliveries are acknowledged**
    - **Used by HTTP, SMTP, FTP**
- ***Connectionless Routing*** **is provided by User Datagram Protocol (UDP)**
  - **Sending packets individually without a virtual circuit**
  - **Each packet is sent independently of one another, and will be routed separately, following different routes and arriving at different times (used in Small control messages ie DHCP , Some multimedia appl)**
- **QoS Routing (provided by RTP)**
  - **A special kind connection oriented routing with priorities**

Setting up Virtual Connections



Copyright 2011 John Wiley & Sons, Inc

UDP - User Datagram Protocol

- **Protocol used for connectionless routing in TCP/IP suite that uses no acks, no flow control**
- **Uses only a small packet header**
  - **Only 8 bytes containing only 4 fields:**
    - **Source port**
    - **Destination port**
    - **Message length**
    - **Header checksum**
- **Commonly used for control messages that are usually small, such as DNS (Domain Name System), DHCP (Dynamic Host Configuration Protocol), RIP (Routing Information Protocol_ and SNMP (Simple Network Management Protocol).**
- **Can also be used for applications where a packet can be lost, such as information rich video**

UDP - User Datagram Protocol

(applications that a packet can be lost)

- **Some audio and video traffic is generally transmitted using UDP**
- **Real-time video and audio streaming protocols are designed to handle occasional lost packets**
- **Only slight degradation in quality occurs, rather than large delays if lost packets were retransmitted**
- **When TCP detects packet loss, it will throttle back its data rate usage**

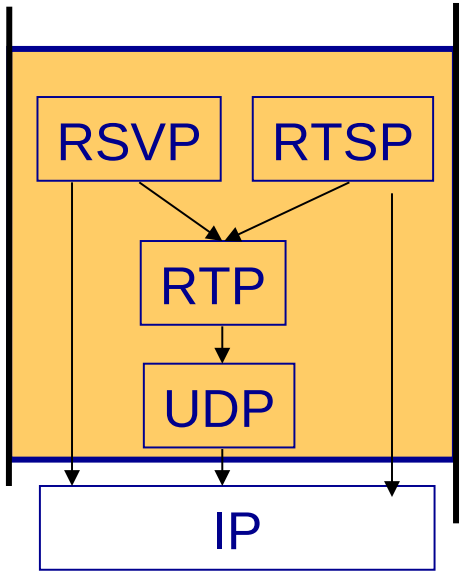
QoS – Quality of Service

- **QoS defines and assigns priorities to "classes of service"**
- **Timeliness - timely delivery of packets**
  - **Packets be delivered within a certain period of time (to produce a smooth, continuous output**
  - **Required by some applications, especially real time applications (e.g., voice and video frames)**
  - **(e-mail doesn't require this)**
- **QoS routing**
  - **Defines classes of service, each with a different priority:**
    - **Real-time applications such as VoIP- highest**
    - **A graphical file for a Web page - a lower priority**
    - **E-mail - lowest (can wait a long time before delivery)**

Copyright 2011 John Wiley & Sons, Inc

Protocols Supporting QoS

- **Asynchronous Transfer Mode (ATM)**
  - **A high-speed data link layer protocol**
- **TCP/IP protocol suite**
  - **Resource Reservation Protocol (RSVP)**
    - **Sets up virtual circuits for general purpose real-time applications**
  - **Real-Time Streaming Protocol (RTSP)**
    - **Sets up virtual circuits for audio-video applications**
  - **Real-Time Transport Protocol (RTP)**
    - **Used after a virtual connection setup by RSVP or RTSP**
    - **Adds a sequence number and a timestamp for helping applications to synchronize delivery**
    - **Uses UDP (because of its small header) as transport**



Copyright 2011 John Wiley & Sons, Inc

Network Layer Functions

- **Addressing**
  - **Each device on the path between source and destination must have an address**
    - **Internet Addresses**
    - **Address Assignment**
    - **Translation between network layer addresses and other addresses (address resolution)**
- **Routing**
  - **Process of deciding what path a packet must take to reach destination**
  - **Routing protocols**

Copyright 2011 John Wiley & Sons, Inc

5.4 Addressing

Copyright 2011 John Wiley & Sons, Inc

Types of Addresses

Address Type	Example	Example Address	Analogy
Application Layer	URL	www.manhattan.edu	Name
Network Layer	IP address	149.61.10.22 (4 bytes)	Zip code
Data Link Layer	MAC address	00-0C-00-F5-03-5A (6 bytes)	Street addr.

- These addresses must be translated from one type to another for a message to travel from sender to receiver.
- This translation process is called address resolution.
- It is like sending a letter via postal mail to John Smith. The zip code (IP address) is used to get the letter to the city (LAN), and then the street address (MAC) is used to get it to the specific house (computer).

Copyright 2011 John Wiley & Sons, Inc

Assignment of Addresses

- **Application Layer address (URL)**
  - **For servers only (clients don't need it)**
  - **Assigned by network managers and placed in configuration files.**
  - **Some servers may have several application layer addresses**
- **Network Layer Address (IP address)**
  - **Assigned by network managers, or by programs such as DHCP, and placed in configuration files**
  - **Every network on the Internet is assigned a range of possible IP addresses for use on its network**
- **Data Link Layer Address (MAC address)**
  - **Unique hardware addresses placed on network interface cards (NIC) by their manufacturers (based on a standardized scheme)**
- **Servers have permanent addresses, clients usually do not**

Copyright 2011 John Wiley & Sons, Inc

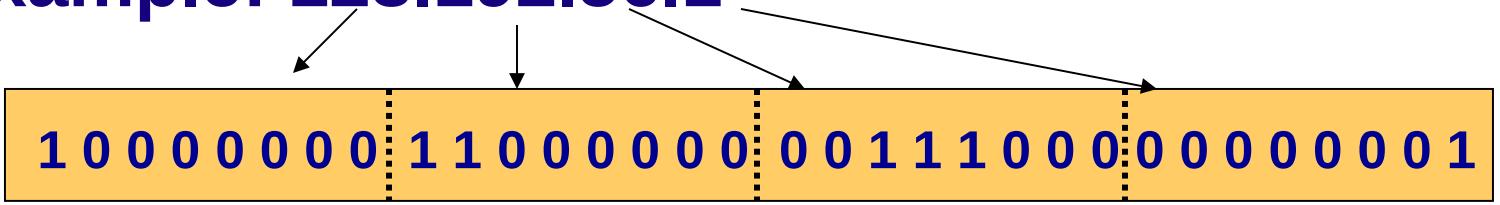
Internet Addresses

- **Managed by ICANN**
  - **Internet Corporation for Assigned Names and Numbers**
  - **Manages the assignment of both IP and application layer name space (domain names)**
    - **Both assigned at the same time and in groups**
    - **Manages some domains directly (e.g., .com, .org, .net) and**
    - **Authorizes private companies to become domain name registrars as well**
- **Example: Indiana University**
  - **URLs that end in .indiana.edu and iu.edu**
  - **IP addresses in the 129.79.x.x range (where x is any number between 0 and 255)**

Copyright 2011 John Wiley & Sons, Inc

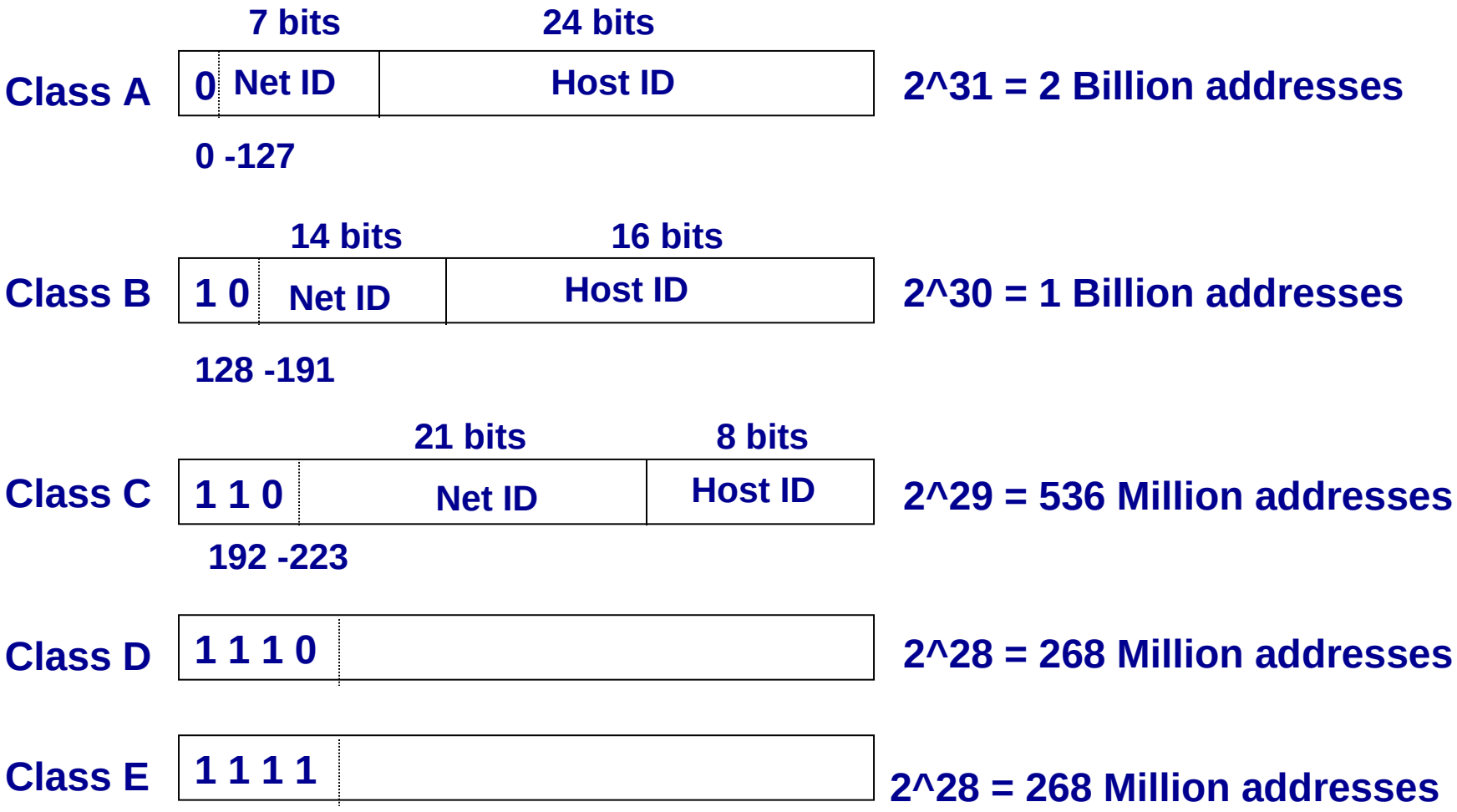
IPv4 Addresses

- **4 byte (32 bit) addresses**
  - **Strings of 32 binary bits**
- **Dotted decimal notation**
  - **Used to make IP addresses easier to understand for human readers**
  - **Breaks the address into four bytes and writes the digital equivalent for each byte**
- **Example: 128.192.56.1**



Copyright 2010 John Wiley & Sons, Inc

Classfull Addressing



A, B and C are routed IPs (may be PUBLIC or PRIVATE). Class D addresses are used for Multi-casting protocols exclusively, for example, OSPF, IGMP etc. E reserved for testing

IPv6 Addressing

- **Need**
  - **IPv4 uses 4 byte addresses:**
    - **Total of one billion possible addresses**
  - **IP addresses often assigned in (large) groups**
    - **Giving out many numbers at a time**
    - **→ IPv4 address space has been used up quickly**
    - **e.g., Indiana University: uses a Class A IP address space (65,000 addresses; many more than needed)**
- **IPv6 uses 16 byte addresses:**
  - **3.2×10^{38} addresses, a very large number**
  - **Little chance this address space will ever be used up**

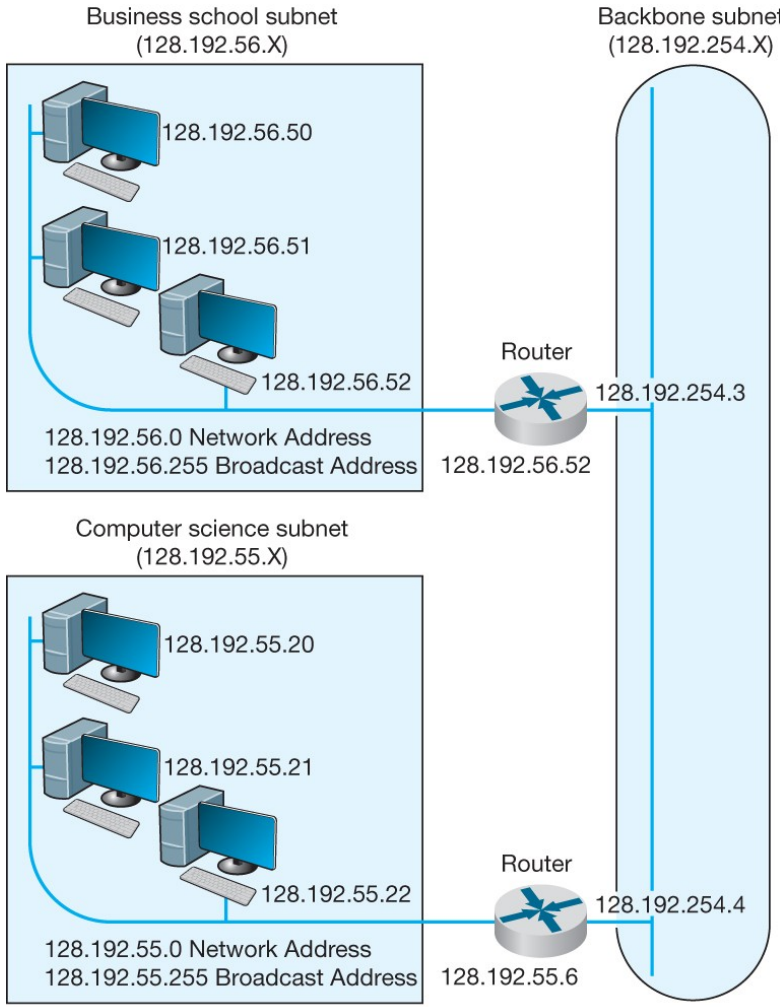
Copyright 2011 John Wiley & Sons, Inc

Subnets

- **Group of computers on the same LAN with IP numbers using the same prefix**
- **Assigned addresses that are 8 bits in length**
  - **For example:**
    - **Subnet 149.61.10.x**
      - **Computers in Business (x is between 0 & 255)**
    - **Subnet 149.61.15.x**
      - **Computers in CS department**
- **Assigned addresses could be more or less than eight bits in length**
  - **For example: If 7 bits used for a subnet**
    - **Subnet 1: 149.61.10.1-128**
    - **Subnet 2: 149.61.10.129-255**

Copyright 2011 John Wiley & Sons, Inc

Subnets: Example



Copyright 2011 John Wiley & Sons, Inc

Subnet Masks

- **Used to make it easier to separate the subnet part of the address from the host part.**
- **Example**
  - **Subnet: 149.61.10.x**
  - **Subnet mask: 255.255.255.000 or in binary**

    11111111.11111111.11111111.00000000
- **Example**
  - **Subnets: 149.61.10.1-128,**
  - **Subnet mask 255.255.255.128 or, in binary:**

    11111111.11111111.11111111.10000000

Copyright 2011 John Wiley & Sons, Inc

Dynamic Addressing

- **Giving addresses to clients (automatically) only when they are logged in to a network**
  - **Eliminates permanent addresses to clients**
  - **When the computer is moved to another location, its new IP address is assigned automatically**
  - **Makes efficient use of IP address space**
  - **Example:**
    - **A small ISP with several thousands subscribers**
    - **Might only need to assign 500 IP addresses to clients at any one time**
- **Uses a server to supply IP addresses to computers whenever the computers connect to network**

Copyright 2011 John Wiley & Sons, Inc

Programs for Dynamic Addressing

- **Bootstrap Protocol (bootp)**
- **Dynamic Host Control Protocol (DHCP)**
- **Different approaches, but same basic operations:**
  - **A program residing in a client establishes connection to bootp or DHCP server**
  - **A client broadcasts a message requesting an IP address (when it is turned on and connected)**
  - **Server (maintaining IP address pool) responds with a message containing IP address (and its subnet mask)**
  - **IP addresses can also be assigned with a time limit (leased IP addresses)**
    - **When expires, client must send a new request**

Copyright 2011 John Wiley & Sons, Inc

Address Resolution

- **Server Name Resolution**
  - **Translating destination host's domain name to its corresponding IP address**
  - **www.yahoo.com is resolved to → 204.71.200.74**
  - **Uses one or more Domain Name Service (DNS) servers to resolve the address**
- **Data Link Layer Address Resolution**
  - **Identifying the MAC address of the next node (that packet must be forwarded)**
  - **Uses Address Resolution Protocol (ARP)**

Copyright 2011 John Wiley & Sons, Inc

DNS – Domain Name Service

- **Used to determine IP address for a given URL**
- **Provided through a group of name servers**
  - **Databases containing directories of domain names and their corresponding IP addresses**
- **Large organizations maintain their own name servers**
  - **smaller organizations rely on name servers provided by their ISPs**
- **When a domain name is registered, IP address of the DNS server must be provided to registrar for all URLs in this domain**
  - **Example: Domain name: *indiana.edu***

    URLs: *www.indiana.edu, www.kelly.indiana.edu, abc.indiana.edu*

Copyright 2011 John Wiley & Sons, Inc

How DNS Works

- **If the desired URL is in the client's address table:**
  - **Use the corresponding IP address**
  - **Each client maintains a server address table**
    - **containing URLs used and corresponding IP addresses**
- **If the desired URL is not in the client's address table:**
  - **Use DNS to resolve the address**
  - **Sends a DNS request packet to its local DNS server**
  - **URL in Local DNS server responds by sending a DNS response packet back to the client**

Copyright 2011 John Wiley & Sons, Inc

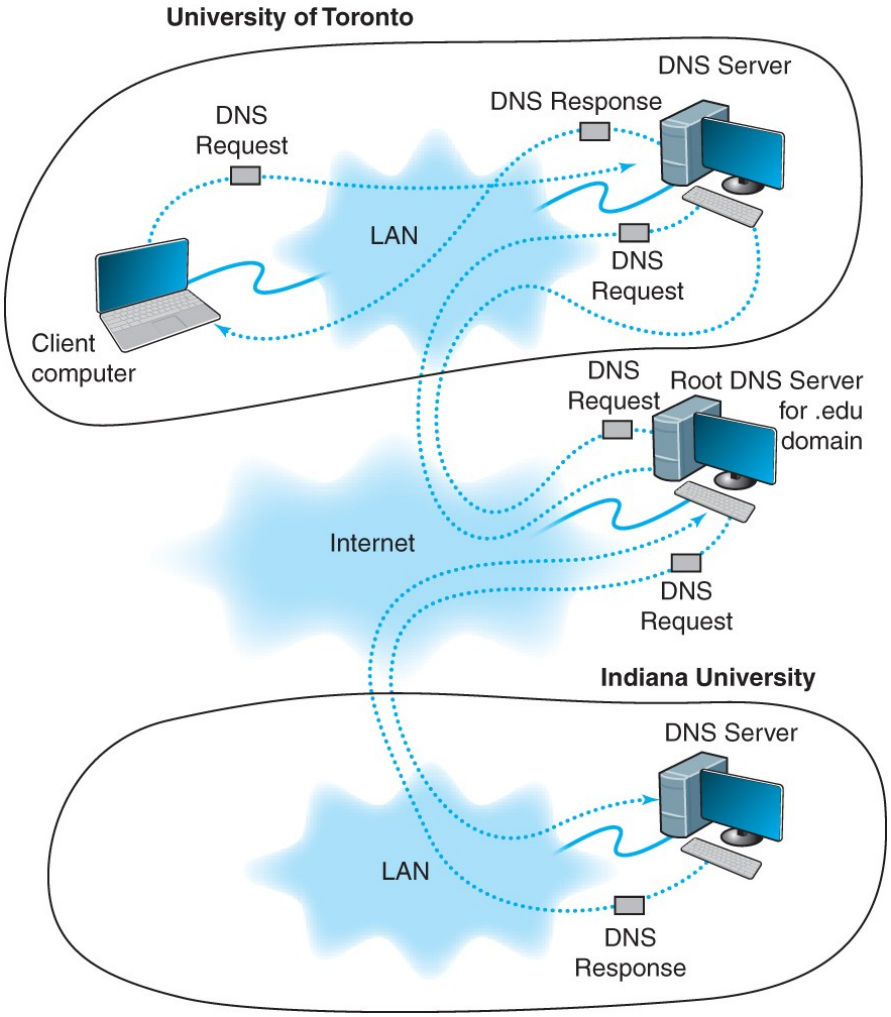
How DNS Works (Cont.)

- **If the URL is NOT in the local DNS server**
  - **Sends DNS request packet to the next highest name server in the DNS hierarchy**
  - **Usually the DNS server at the top level domain (such as the DNS server for all .edu domains)**
  - **If the URL is NOT in the name server**
    - **Sends DNS request packet ahead to name server at the next lower level of the DNS hierarchy**

Copyright 2011 John Wiley & Sons, Inc

How DNS Works

If client at Toronto asks for a web page on Indiana University's server:



MAC Address Resolution

- **Use Address Resolution Protocol (ARP) to find**
  - **Unknown MAC address of the next node (whose IP address is known)**
- **Operation**
  - **Broadcast an ARP message to all nodes on a LAN asking which node has a certain IP address**
  - **Host with that IP address then responds by sending back its MAC address**
  - **Store this MAC address in its address table**
  - **Send the message to the destination**

Copyright 2011 John Wiley & Sons, Inc

Warriors of the Net Video

http://www.youtube.com/watch?v=PBWhzz_Gn10

Copyright 2011 John Wiley & Sons, Inc

5.5 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**

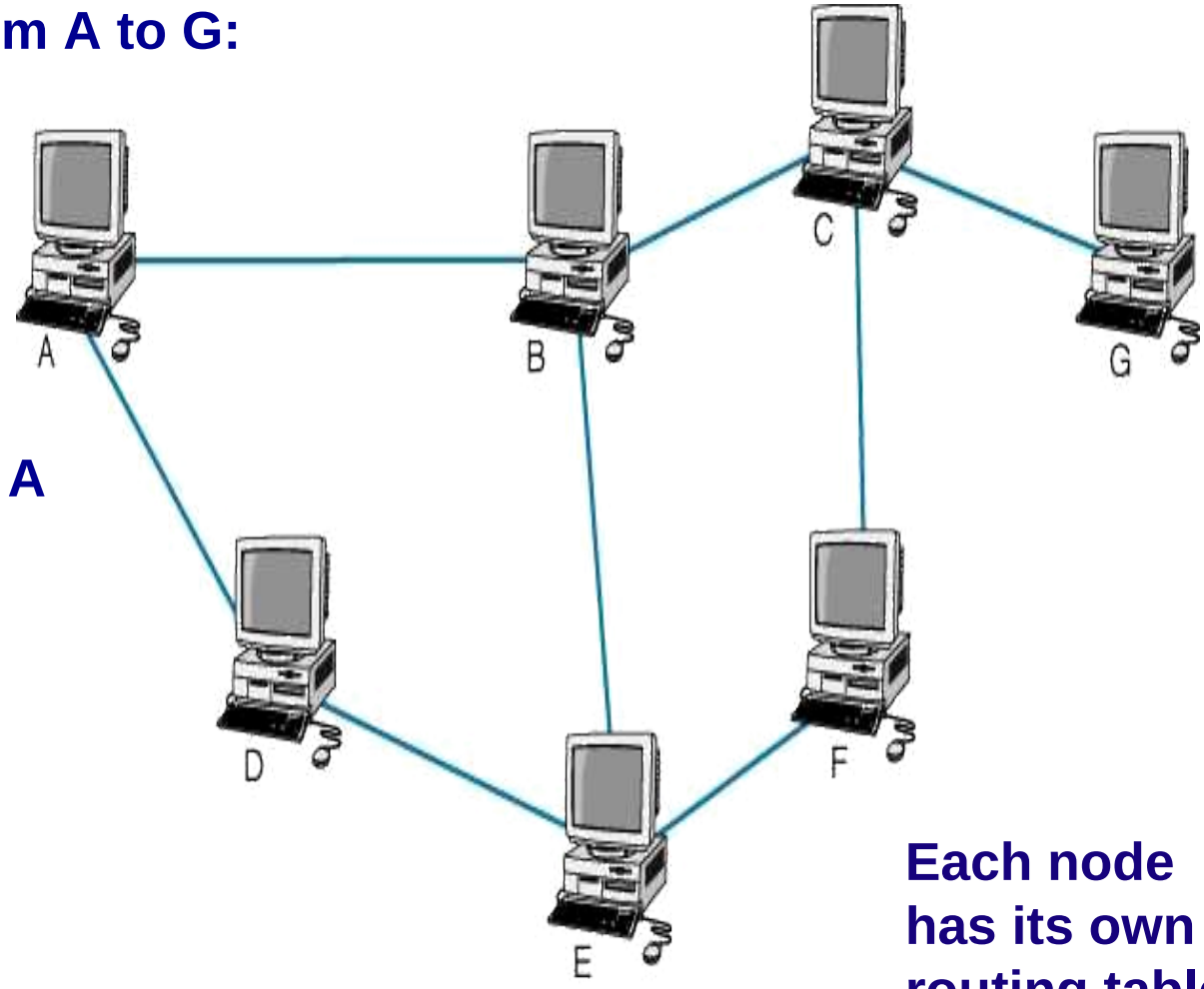
Dest.	Next
B	B
C	B
D	D
E	D
F	D
G	B

Copyright 2011 John Wiley & Sons, Inc

Routing Example

Possible paths from A to G:

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



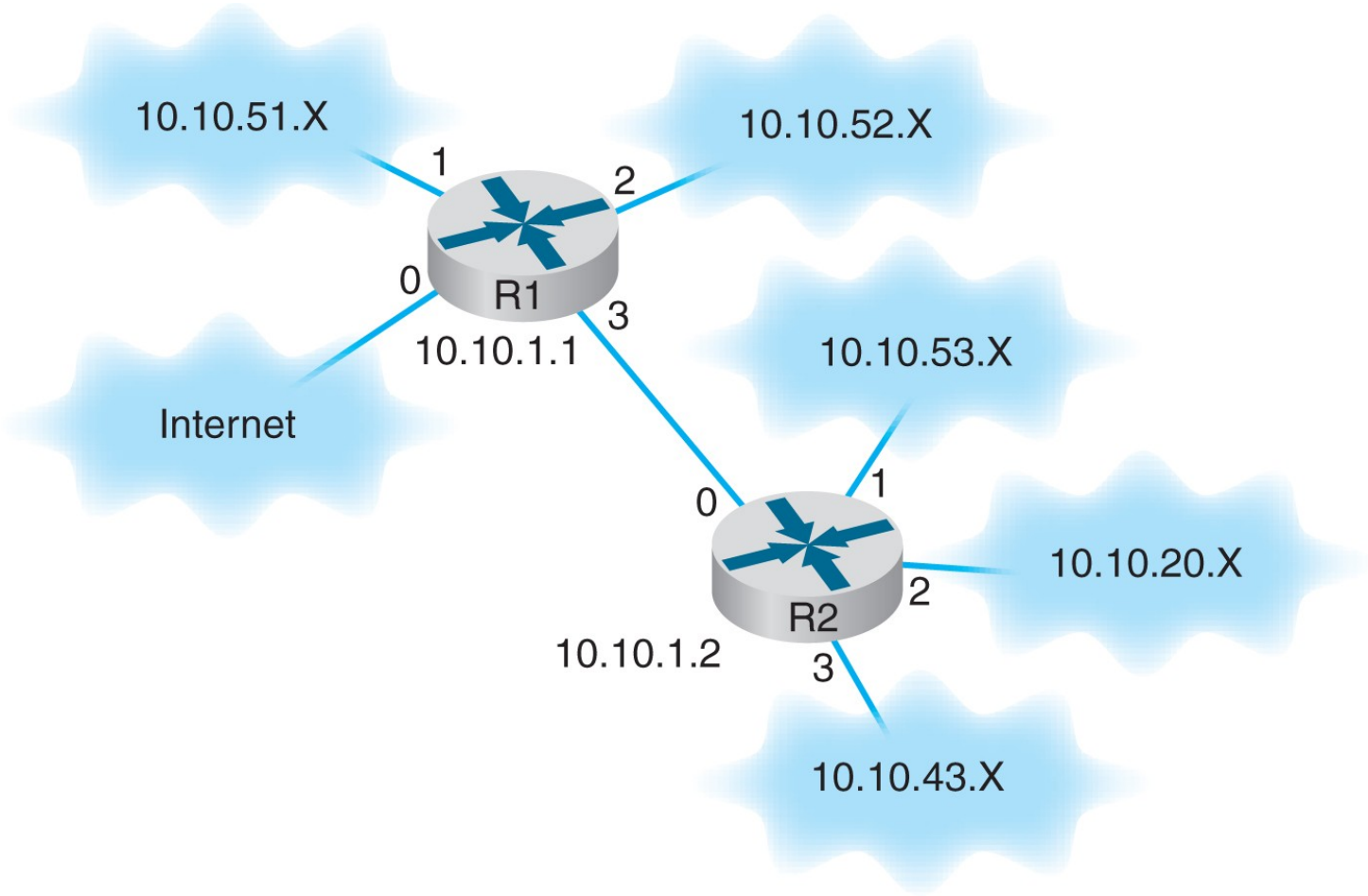
Routing Table for A

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

Each node has its own routing table

Copyright 2011 John Wiley & Sons, Inc

Routing



Copyright 2011 John Wiley & Sons, Inc

Routing

Router R1's Routing Table

Network Address	Interface
10.10.51.0 to 10.10.51.255	1
10.10.52.0 to 10.10.52.255	2
10.10.53.0 to 10.10.53.255	3
10.10.20.0 to 10.10.20.255	3
10.10.43.0 to 10.10.43.255	3
10.10.1.2	3
All other addresses	0

Router R2's Routing Table

Network Address	Interface
10.10.1.1	0
10.10.53.0 to 10.10.53.255	1
10.10.20.0 to 10.10.20.255	2
10.10.43.0 to 10.10.43.255	3
All other addresses	0

Copyright 2011 John Wiley & Sons, Inc

Types of Routing

- **Centralized routing**
  - **Decisions made by one central computer**
  - **Used on small, mainframe-based networks**
- **Decentralized routing**
  - **Decisions made by each node independently of one another**
  - **Information needs to be exchanged to prepare routing tables**
  - **Used by the Internet**

Copyright 2011 John Wiley & Sons, Inc

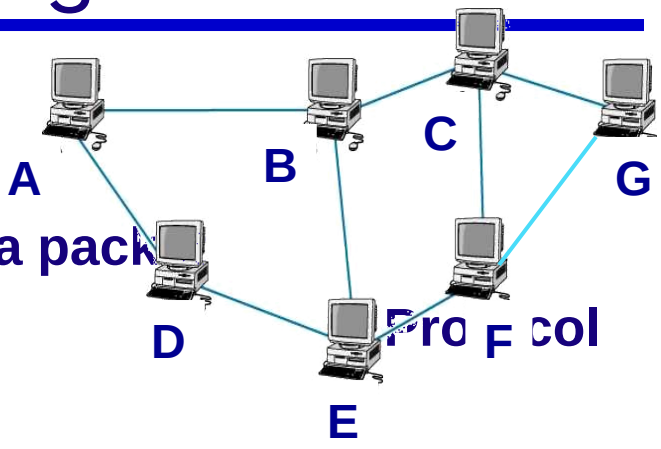
Types of Decentralized Routing

- **Static routing:**
  - **Uses fixed routing tables developed by network managers**
    - **Each node has its own routing table**
    - **Changes when computers added or removed**
  - **Used on relatively simple networks with few routing options that rarely change**
- **Dynamic routing or Adaptive routing:**
  - **Uses routing tables at each node that are updated dynamically**
  - **Based on routing condition information exchanged between routing devices (ie. Number of hops)**

Copyright 2011 John Wiley & Sons, Inc

Dynamic Routing Algorithms

- **Distance Vector**
  - **Uses the least number of hops to decide how to route a pack**
  - **Used by Routing Information Protocol (RIP)**



Ex: From A to G → ABCG

- **Link State**
  - **Uses a variety of information types to decide how to route a packet (more sophisticated)**
    - **e.g., number of hops, congestion, speed of circuit**
  - **Links state info exchanged periodically by each node to keep every node in the network up to date**
  - **Provides more reliable, up to date paths to destinations**
  - **Used by Open Shortest Path First (OSPF)**

Copyright 2011 John Wiley & Sons, Inc

Routing Protocols

- **Used to exchange info among nodes for building and maintaining routing tables**
- **Autonomous System (AS)**

  Is a network operated by an organization (e.g., Indiana U.)

  –Protocols classified are based on autonomous systems

    - **AS being a collection of connected Internet Protocol (IP) routing prefixes under the control of one or more network operators that presents a common, clearly defined routing policy to the Interne**

- **Types of Routing Protocols**

  –Interior routing protocols (RIP, OSPF, EIGRP, ICMP)

    - **Operate within a network (autonomous system)**
    - **Provide detailed info about each node and paths**

  –Exterior routing protocols (BGP)

    - **Operate between networks (autonomous systems)**

Routing Information Protocol (RIP)

- **A dynamic distance vector interior routing protocol**
- **Once popular on Internet; now used on simple networks**
- **Operations:**
  - **Manager builds a routing table**
  - **Routing tables broadcast periodically (every minute or so) by all nodes**
  - **When a new node added, RIP counts number of hops between computers and updates routing tables**

Copyright 2011 John Wiley & Sons, Inc

Open Shortest Path First (OSPF)

- **A dynamic link state interior routing protocol**
  - **Only the updates sent (not entire routing tables) and only to other routers (no broadcasting)**

Copyright 2011 John Wiley & Sons, Inc

Other Interior Routing Protocols

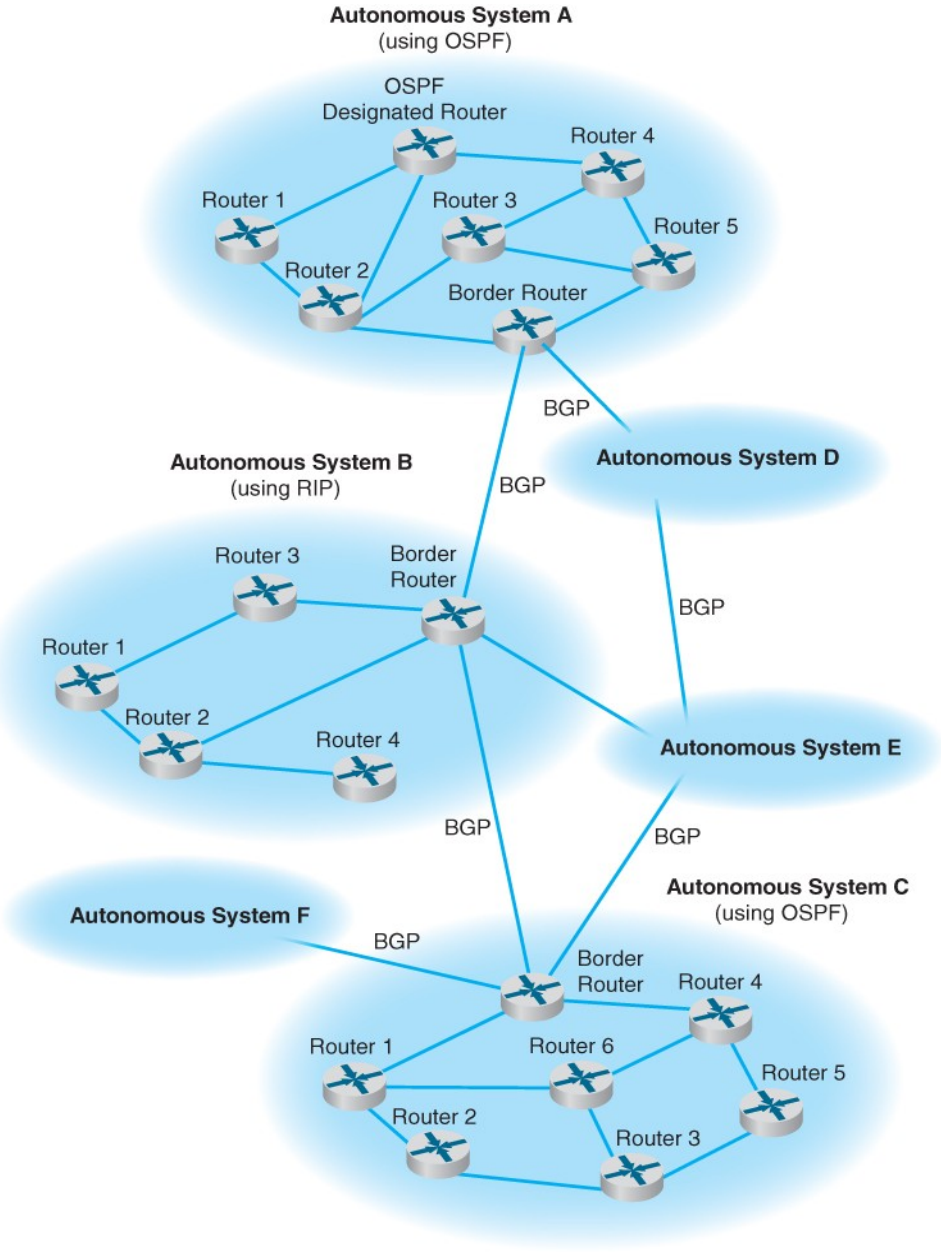
- **Internet Control Message Protocol (ICMP)**
  - **Simplest and most basic**
  - **An error reporting protocol (report routing errors to message senders)**
  - **Limited ability to update routing tables**
- **Enhanced Interior Gateway Routing Protocol (EIGRP)**
  - **A dynamic link state protocol (developed 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**

Copyright 2011 John Wiley & Sons, Inc

Exterior Routing Protocols

- **Border Gateway Protocol (BGP)**
  - **Used to exchange routing info between autonomous systems**
  - **Based on a dynamic distance vector algorithm**
  - **Far more complex than interior routing protocols**
  - **Provide routing info only on selected routes (e.g., preferred or best route)**
    - **Privacy concern**
    - **Too many routes; can't maintain tables of every single rout**

Copyright 2011 John Wiley & Sons, Inc



Internet Routing using BGP, OSPF and RIP

Copyright 2011 John Wiley & Sons, Inc

Multicasting

- **Casting**
  - ***Unicast* message: one computer → another computer**
  - ***Broadcast* message: one computer → all computers in the network**
  - ***Multicast* message: one computer → a group of computers (e.g., videoconference)**
- **Internet Group Management Protocol (IGMP)**
  - **Assists computer to report multicast group membership to adjacent routers**
  - **A special IP address assigned to identify the group**
  - **Routing node sets MAC address to a matching MAC address**
  - **When multicast session ends, IGMP sends a message to the organizing computer(or router) to remove multicast group**

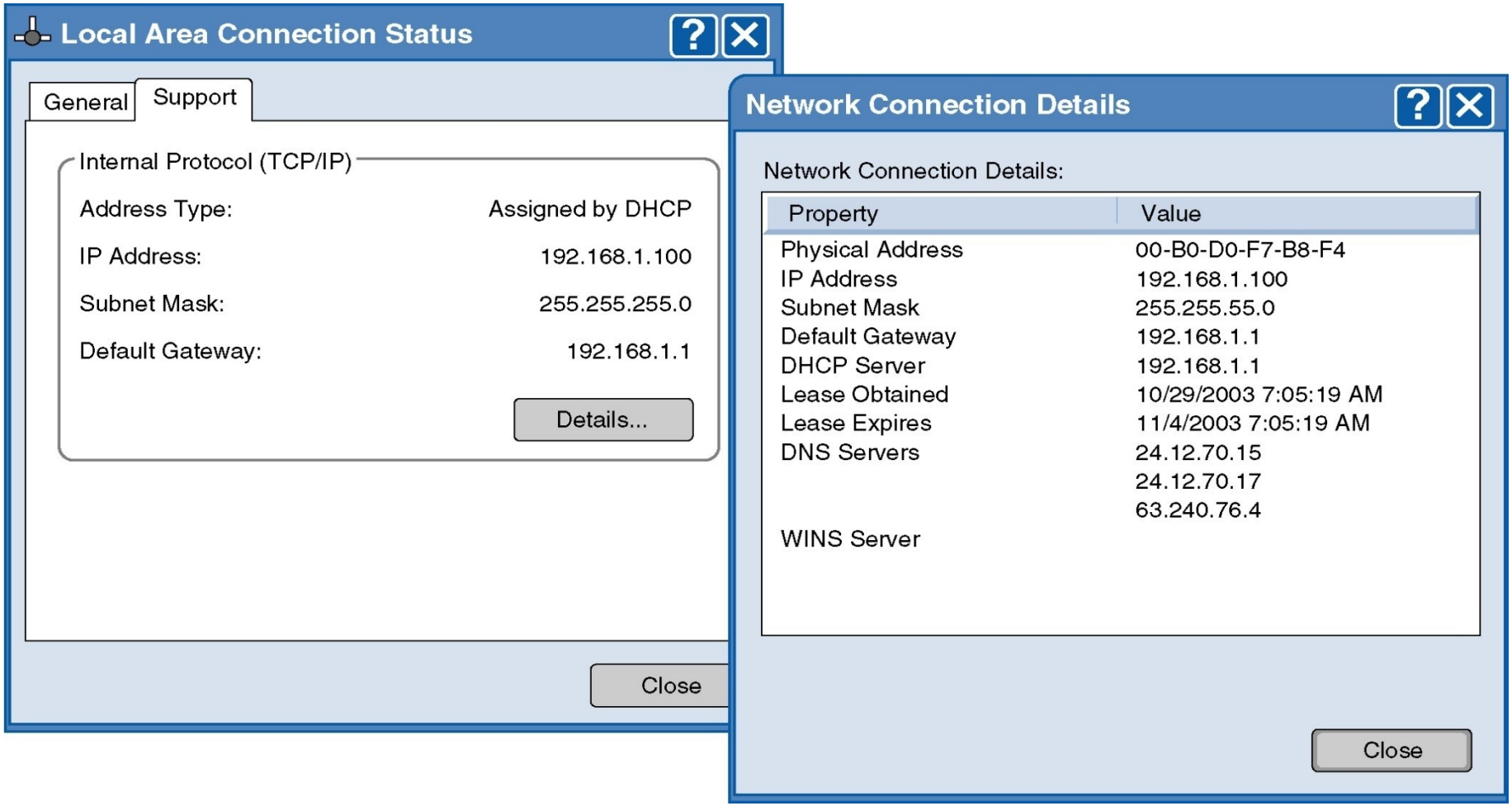
Copyright 2011 John Wiley & Sons, Inc

Sending Messages using TCP/IP

- **Required Network layer addressing information**
  - **Computer's own IP address**
  - **Its subnet mask**
    - **To determine what addresses are part of its subnet**
  - **Local DNS server's IP address**
    - **To translate URLs into IP addresses**
  - **IP address of the router (gateway) on its subnet**
    - **To route messages going outside of its subnet**
- **Address information is obtained from a configuration file or provided by a DHCP server**
  - **Servers also need to know their own application layer addresses (domain names)**

Copyright 2011 John Wiley & Sons, Inc

TCP/IP Configuration Information



Copyright 2011 John Wiley & Sons, Inc

Routers

- **Functions:**
  - **Determine path**
  - **Transmit packets**
  - **Support communication between variety of devices and protocols**
- **Contain:**
  - **CPU, memory, ports/interfaces, OS**

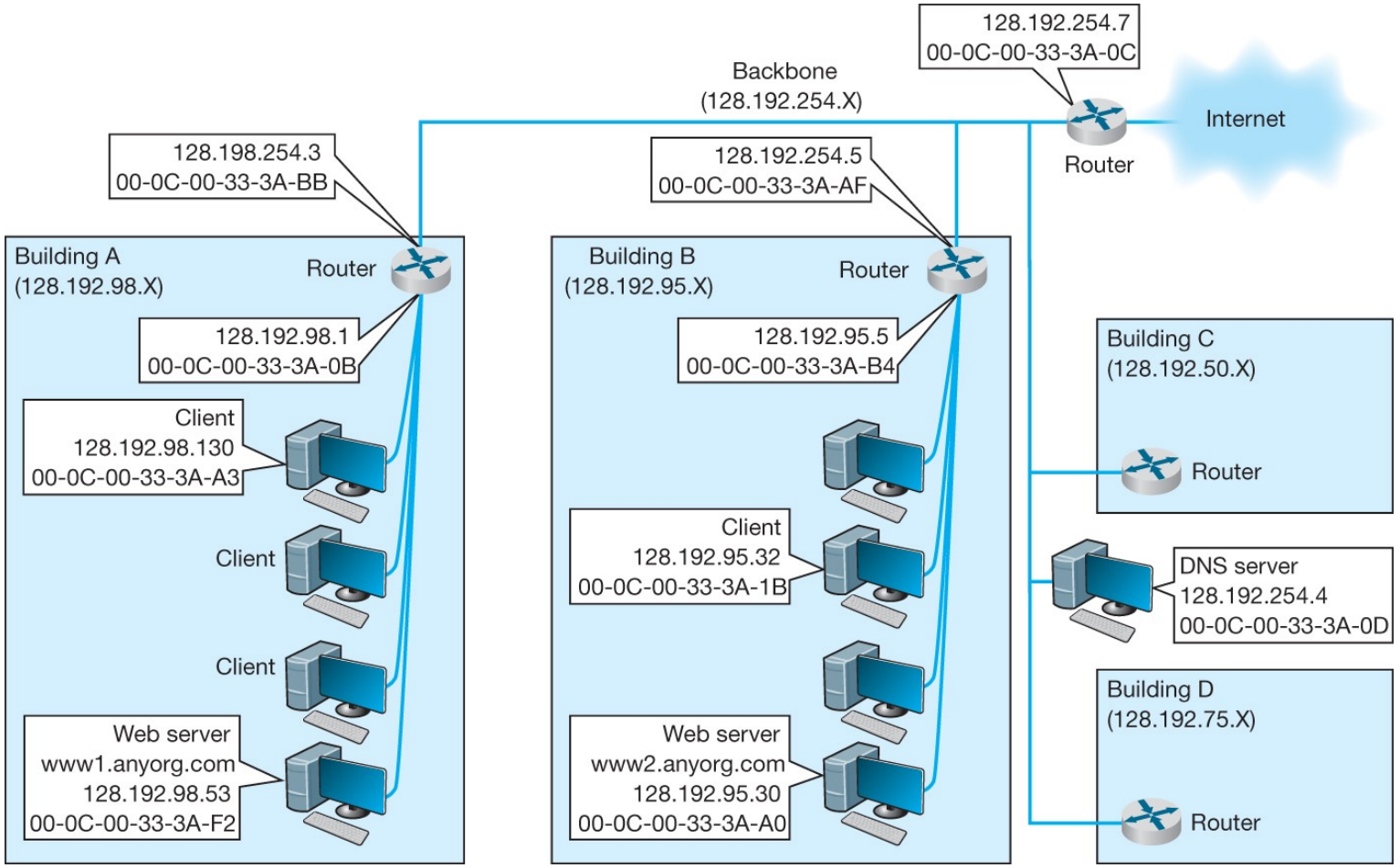
Copyright 2011 John Wiley & Sons, Inc

Routers

- **Connecting to a router:**
  - **Console port**
  - **Network interface port**
  - **Auxiliary port**
- **ACL**
  - **Access Control List**

Copyright 2011 John Wiley & Sons, Inc

5.6 TCP/IP Example



Copyright 2011 John Wiley & Sons, Inc

Case 1a: Known Address, Same Subnet

- **Case:**
  - **A Client (128.192.98.130) requests a Web page from a server (www1.anyorg.com)**
  - **Client knows the server's IP and Ethernet addresses**
- **Operations (performed by the client)**
  - **Prepare HTTP packet and send it to TCP**
  - **Place HTTP packet into a TCP packet and sent it to IP**
  - **Place TCP packet into an IP packet, add destination IP address, 128.192.98.53**
  - **Use its subnet mask to see that the destination is on the same subnet as itself**
  - **Add server's Ethernet address into its destination address field, and send the frame to the Web server**

Copyright 2011 John Wiley & Sons, Inc

Case 1b: HTTP response to client

- **Operations (performed by the server)**
  - **Receive Ethernet frame, perform error checking and send back an ACK**
  - **Process incoming frame successively up the layers (data link, network, transport and application) until the HTTP request emerges**
  - **Process HTTP request and sends back an HTTP response (with requested Web page)**
  - **Process outgoing HTTP response successively down the layers until an Ethernet frame is created**
  - **Send Ethernet frame to the client**
- **Operations (performed by the client)**
  - **Receive Ethernet frame and process it successively up the layers until the HTTP response emerges at browser**

Copyright 2011 John Wiley & Sons, Inc

Case 2: Known Address, Different Subnet

- **Similar to Case 1a**
- **Differences**
  - **Use subnet mask to determine that the destination is NOT on the same subnet**
  - **Send outgoing frames to the local subnet's GW**
  - **Local gateway operations**
    - **Receive the frame and remove the Ethernet header**
    - **Determine the next node (via Router Table)**
    - **Make a new frame and send it to the destination GW**
  - **Destination gateway operations**
    - **Remove the header, determine the destination (by destination IP address)**
    - **Place the IP packet in a new Ethernet frame and send it to its final destination.**

Copyright 2011 John Wiley & Sons, Inc

Case 3: Unknown Address

- **Operations (by the host)**
  - **Determine the destination IP address**
    - **Send a UDP packet to the local DNS server**
    - **Local DNS server knows the destination host's IP address**
      - **Sends a DNS response back to the sending host**
    - **Local DNS server does not know the destination IP address**
      - **Send a second UDP packet to the next highest DNS host, and so on, until the destination host's IP address is determined**
  - **Follow steps in Case 2**

Copyright 2011 John Wiley & Sons, Inc

TCP Connections

- **Before any data packet is sent, a connection is established**
  - **Use SYN packet to establish connection**
  - **Use FIN packet to close the connection**
- **Handling of HTTP packets**
  - **Old version:**
    - **a separate TCP connection for each HTTP Request**
  - **New version:**
    - **Open a connection when a request (first HTTPP Request) send to the server**
    - **Leave the connection open for all subsequent HTTP requests to the same server**
    - **Close the connection when the session ends**

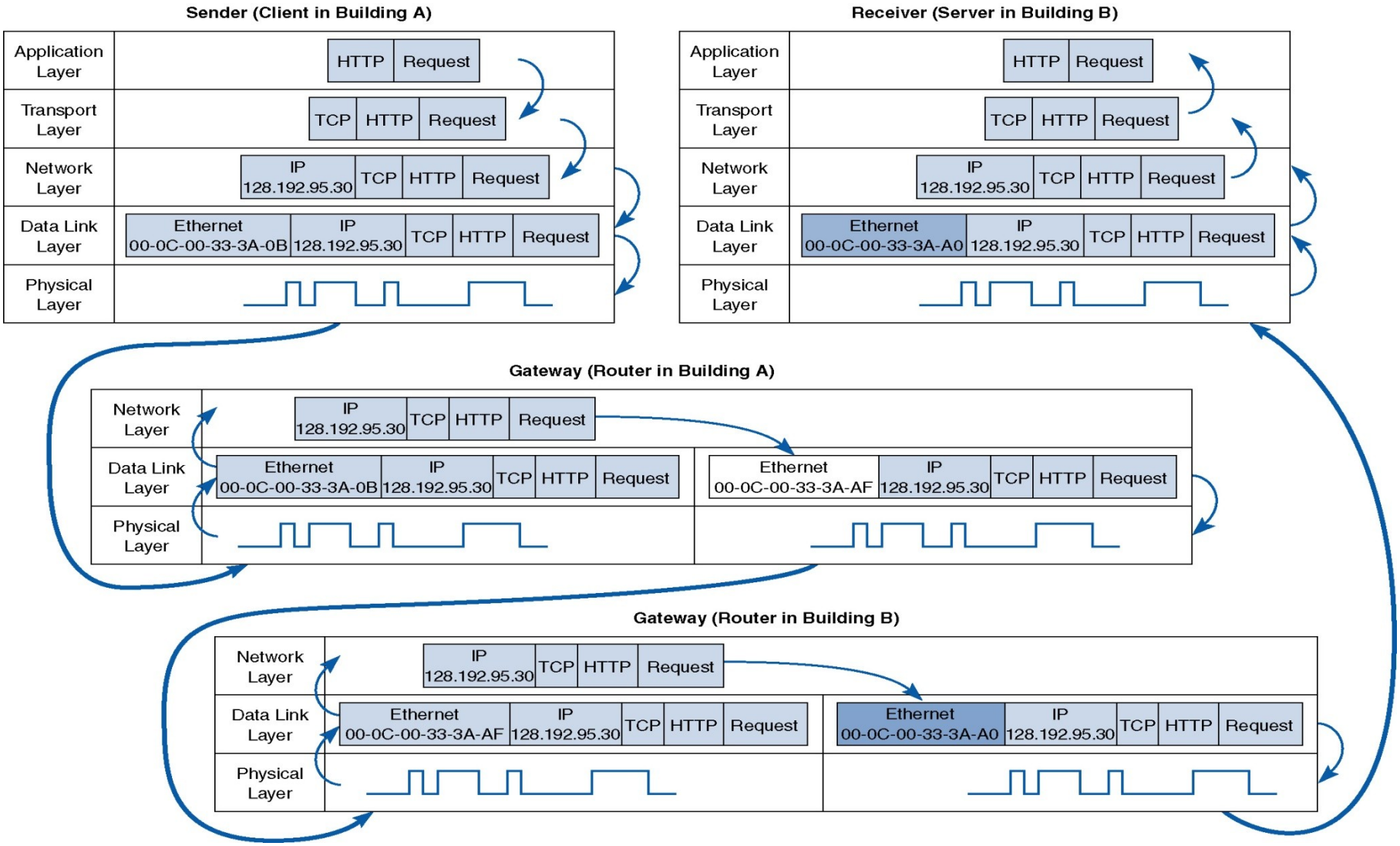
Copyright 2011 John Wiley & Sons, Inc

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)**

Copyright 2011 John Wiley & Sons, Inc

Message Moving Through Layers



5.7 Implications for Management

- **Most organizations moving toward a single standard based on TCP/IP**
  - **Decreased cost of buying and maintaining network equipment**
  - **Decreased cost of training networking staff**
- **Telephone companies with non-TCP/IP networks are also moving toward TCP/IP**
  - **Significant financial implications for telcos**
  - **Significant financial implications for networking equipment manufacturers**

Copyright 2011 John Wiley & Sons, Inc

Copyright 2011 John Wiley & Sons, Inc.

All rights reserved. Reproduction or translation of this work beyond that permitted in section 117 of the 1976 United States Copyright Act without express permission of the copyright owner is unlawful. Request for further information should be addressed to the Permissions Department, John Wiley & Sons, Inc. The purchaser may make back-up copies for his/her own use only and not for distribution or resale. The Publisher assumes no responsibility for errors, omissions, or damages caused by the use of these programs or from the use of the information herein.