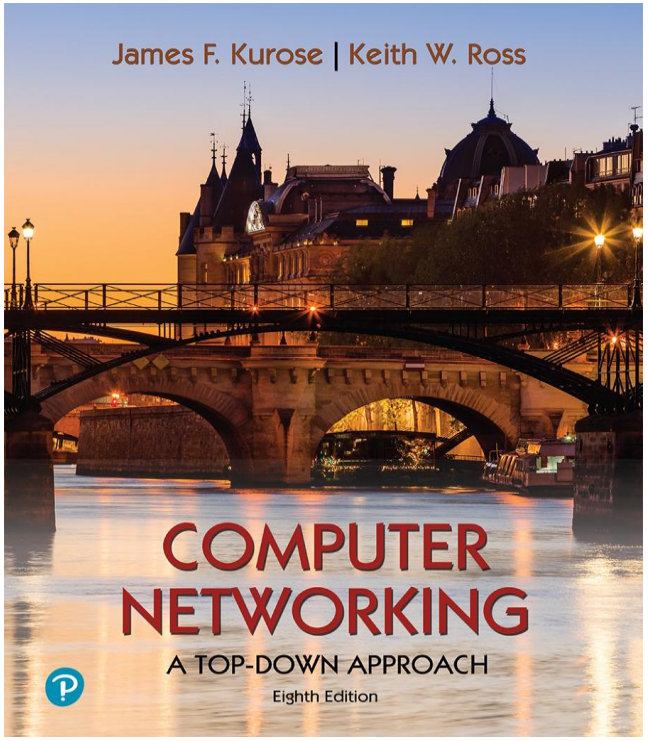
CS509

Computer Networks

Lecture One

Course Outline

1. Computer Networks and the Internet (Chapter 1)
2. TCP/IP Protocol
   - A. Application Layer (Chapter 2)
   - B. Transport Layer (Chapter 3)
   - C. Network Layer
     - I. Data Plane (Chapter 4)
     - II. Control Plane (Chapter 5)
   - D. Link Layer and LANs (Chapter 6)
3. Wireless and Mobile Networks (Chapter 7)



Computer Networking: A Top-Down Approach

8th edition
Jim Kurose, Keith Ross
Pearson, 2020

Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

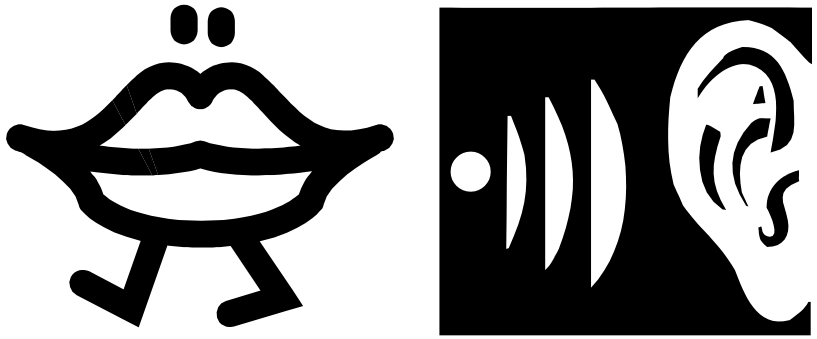
1.3 Network Core: packet/circuit switching, internet structure

1.4 Performance: delay, loss and throughput

1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

Human Communication



- A **transmitter**: mouth
- A **receiver**: ear
- The **media**: air
- The **protocol**: a common human language

Communication Model

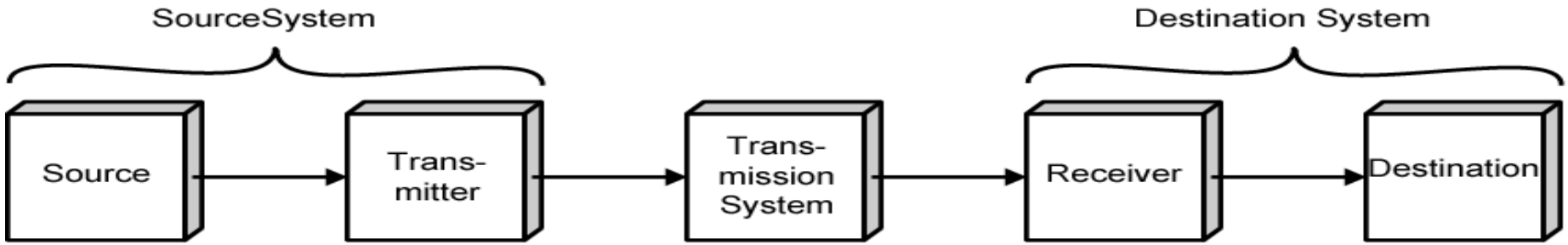
- What is the purpose of communication?
  - **Exchange of information between two parties**

- Key elements
  - **Source:** Generates data to be transmitted. E.g., telephones, PCs.
  - **Transmitter:** A transmitter transforms and encodes the information in such a way as to produce electromagnetic signals that can be transmitted across some sort of transmission system.
  - **Transmission System:** It can be a single transmission line or a complex network connecting source and destination.

Communication Model (2)

- **Receiver:** The receiver accepts the signal from the transmission system and converts it into a form that can be handled by the destination device.

- **Destination:** Takes incoming data from the receiver.



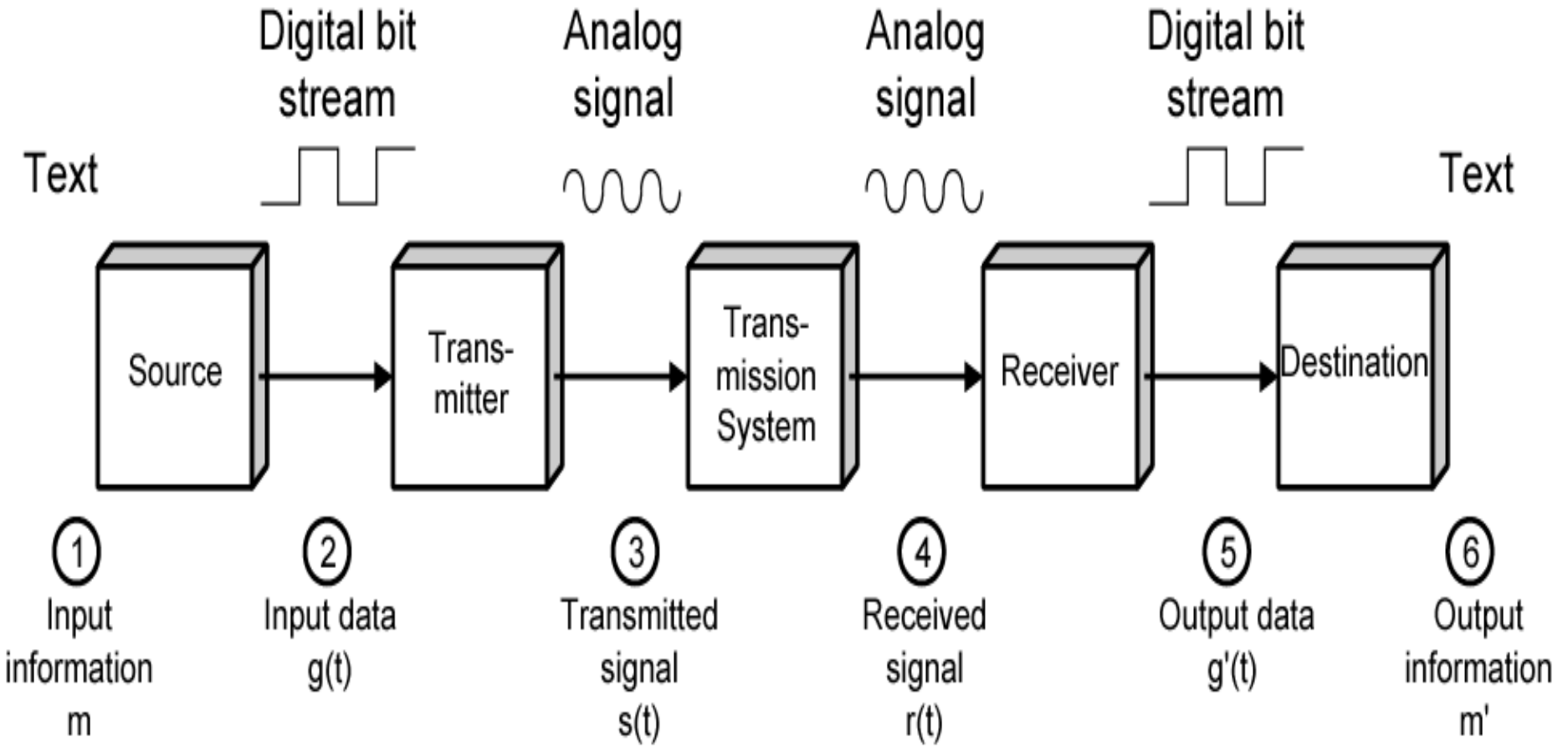
(a) General block diagram

(b) Example

Communication Model (3)

Assume the source and destination are PCs.

The source wishes to send a message ***m*** to the destination.



The Internet: a "nuts and bolts" view

Billions of connected computing *devices*:

- *hosts = end systems*
- running *network apps* at Internet's "edge"

Packet switches: forward packets (chunks of data)

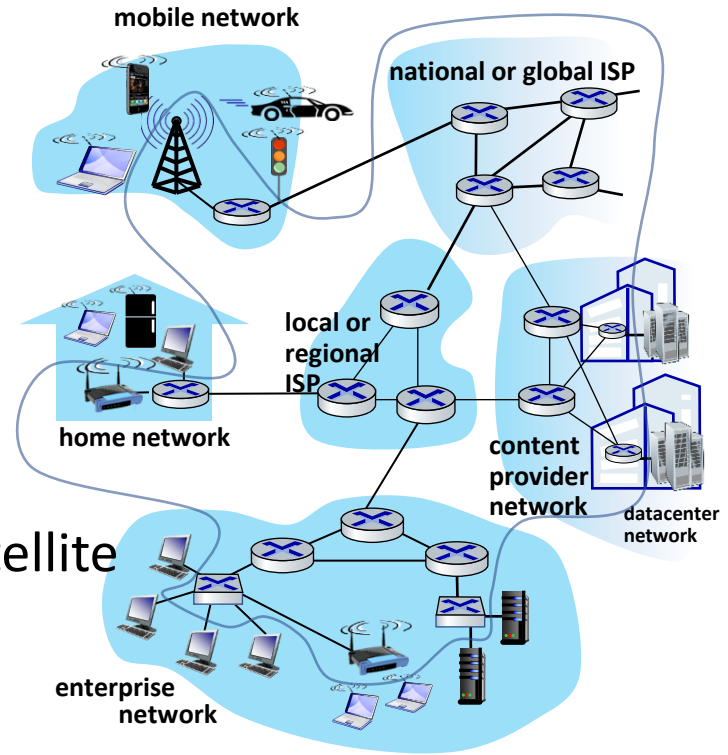
- *routers, switches*

Communication links

- fiber, copper, radio, satellite
- transmission rate: *bandwidth*

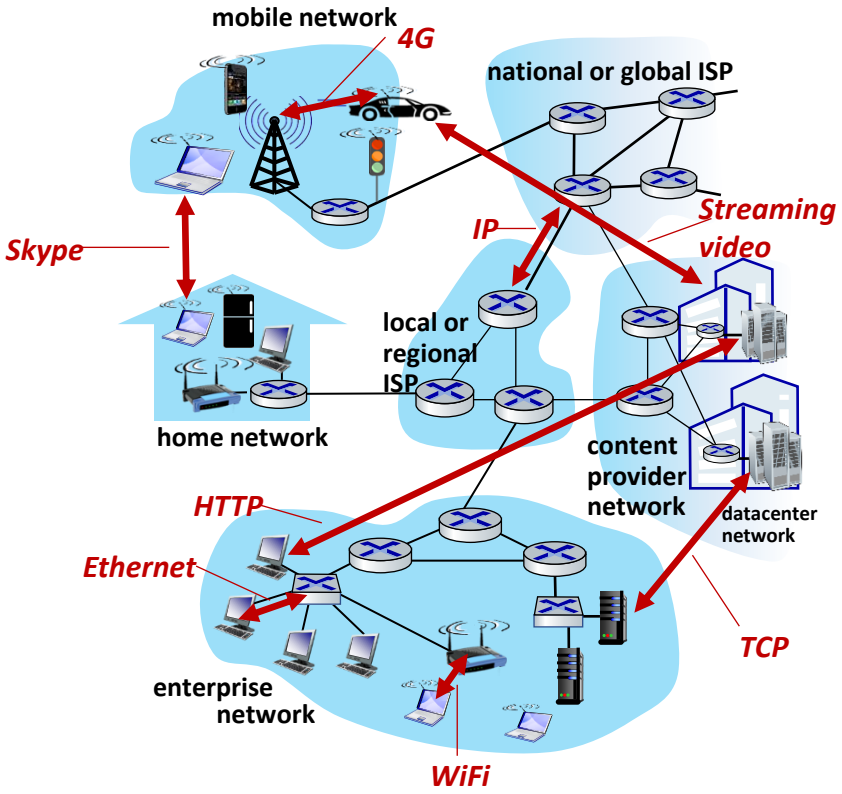
Networks

- collection of devices, routers, links: managed by an organization



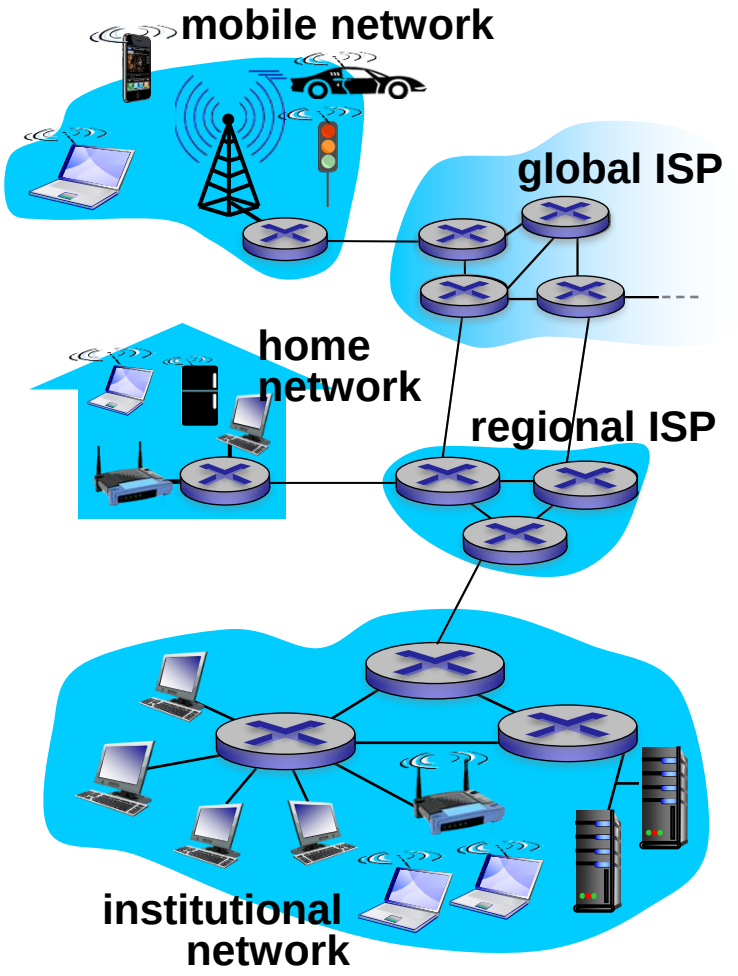
The Internet: a "nuts and bolts" view (2)

- *Internet:* "network of networks"
  - Interconnected ISPs
- *Protocols* are *everywhere*
  - control sending, receiving of messages
  - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 4G, Ethernet
- *Internet standards*
  - RFC: Request for Comments
  - IETF: Internet Engineering Task Force



What's the Internet: a service view

- *infrastructure that provides services to applications:*
  - Web, VoIP, email, games, e-commerce, social nets, …
- *provides programming interface to apps*
  - hooks that allow sending and receiving app programs to "connect" to Internet
  - provides service options, analogous to postal service



What's a Protocol?

human protocols:

- "what's the time?"
- "I have a question"

… specific messages sent

… specific actions taken when messages received, or other events

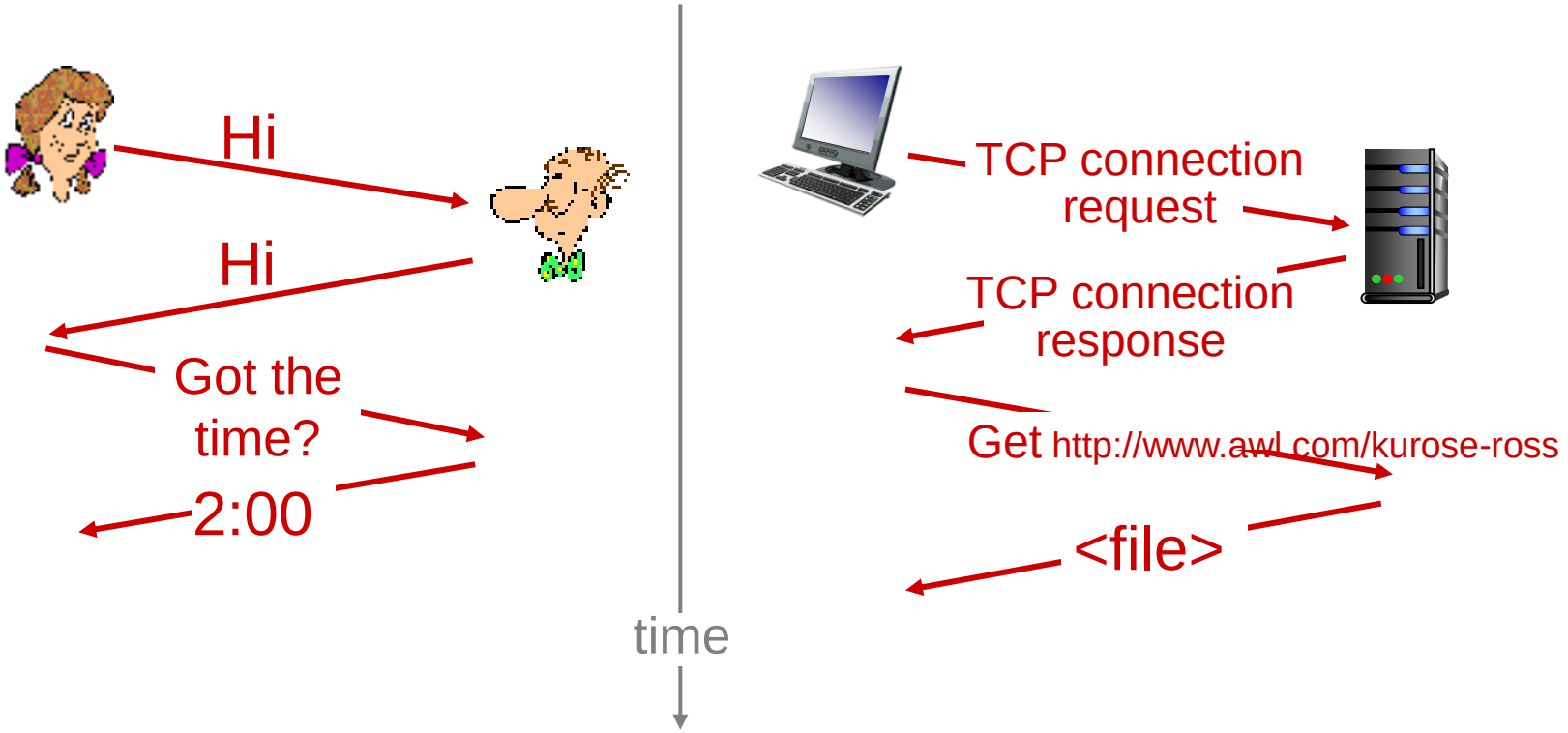
network protocols:

- machines rather than humans
- all communication activity in Internet governed by protocols

protocols define format, order of messages sent and received among network entities, and actions taken on message transmission, receipt

What's a Protocol? (2)

a human protocol and a computer network protocol:



Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

1.3 Network Core: packet/circuit switching, internet structure

1.4 Performance: delay, loss and throughput

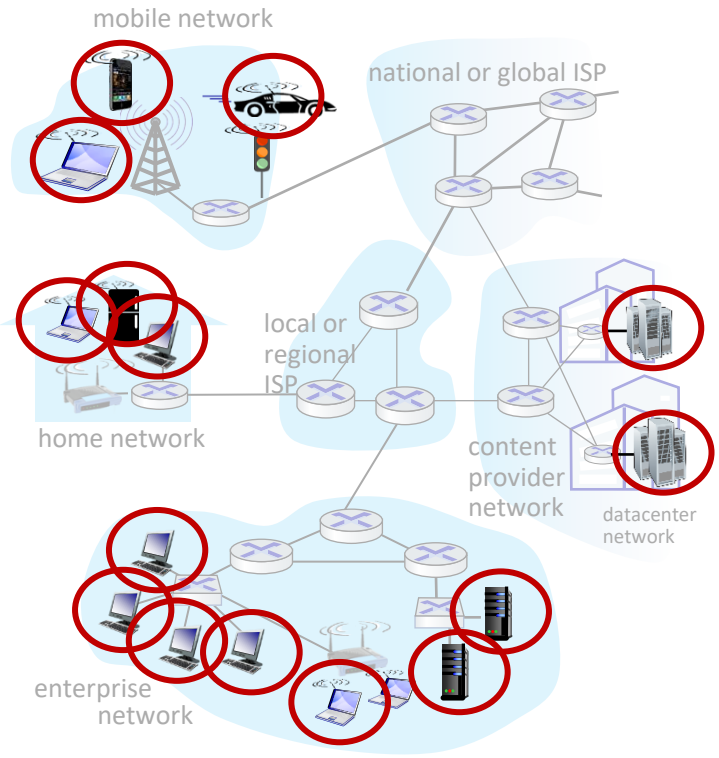
1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

A closer look at Internet structure

Network edge:

- hosts: clients and servers
- servers often in data centers



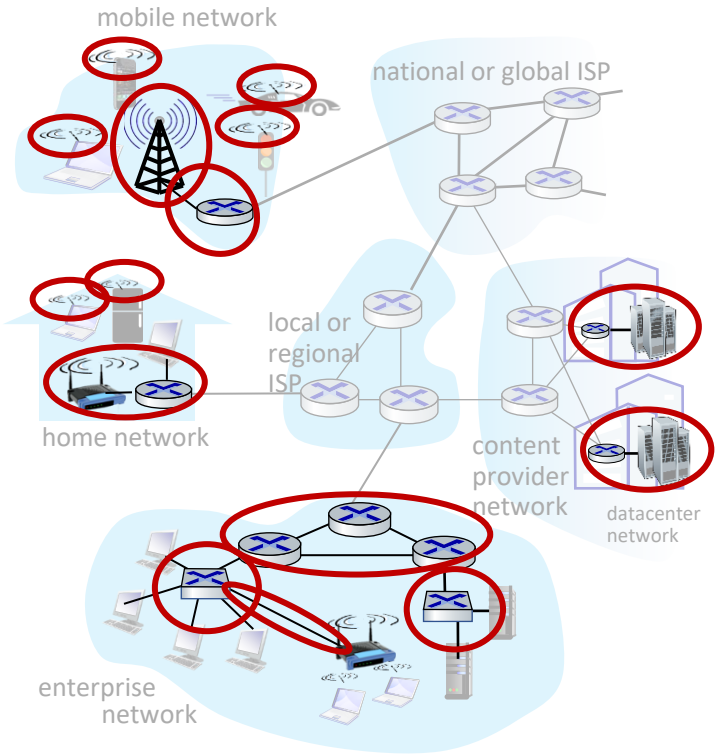
A closer look at Internet structure (2)

Network edge:

- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links



A closer look at Internet structure (3)

Network edge:

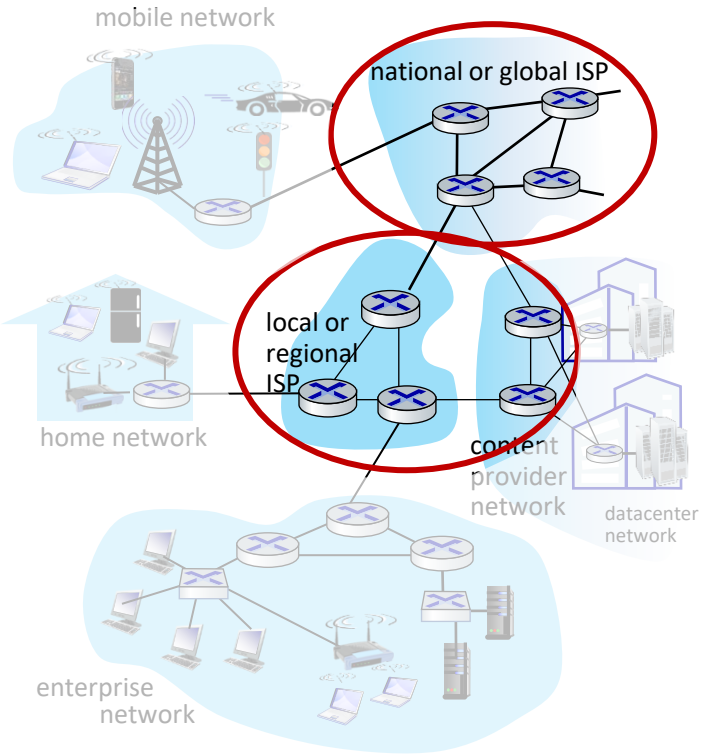
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

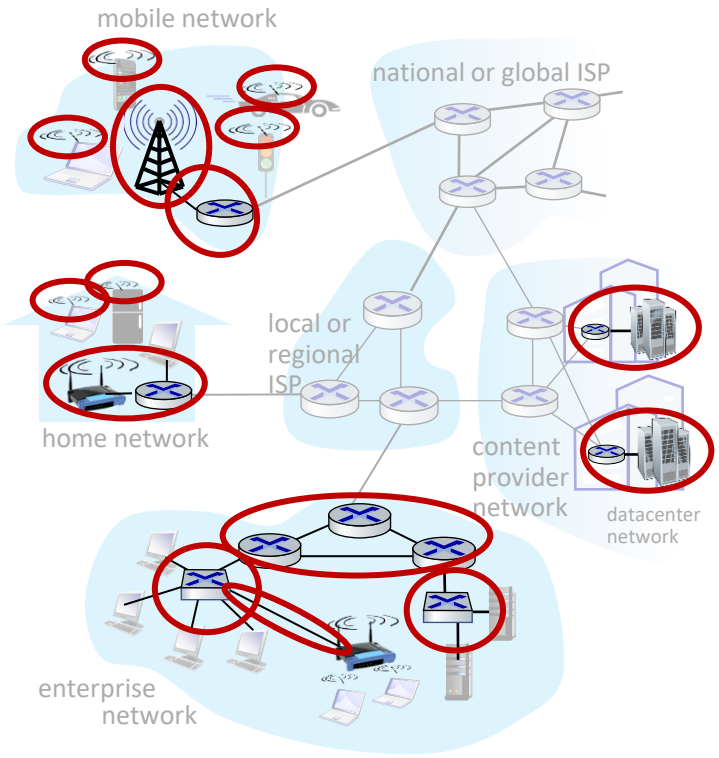
- interconnected routers
- network of networks



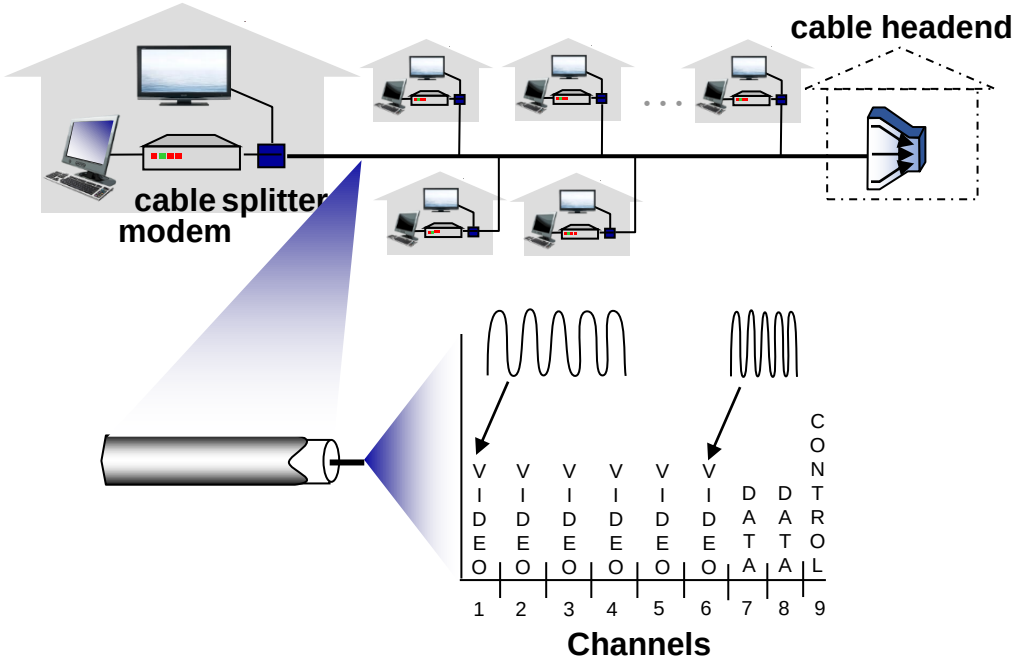
Access networks and physical media

Q: How to connect end systems to edge router?

- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)

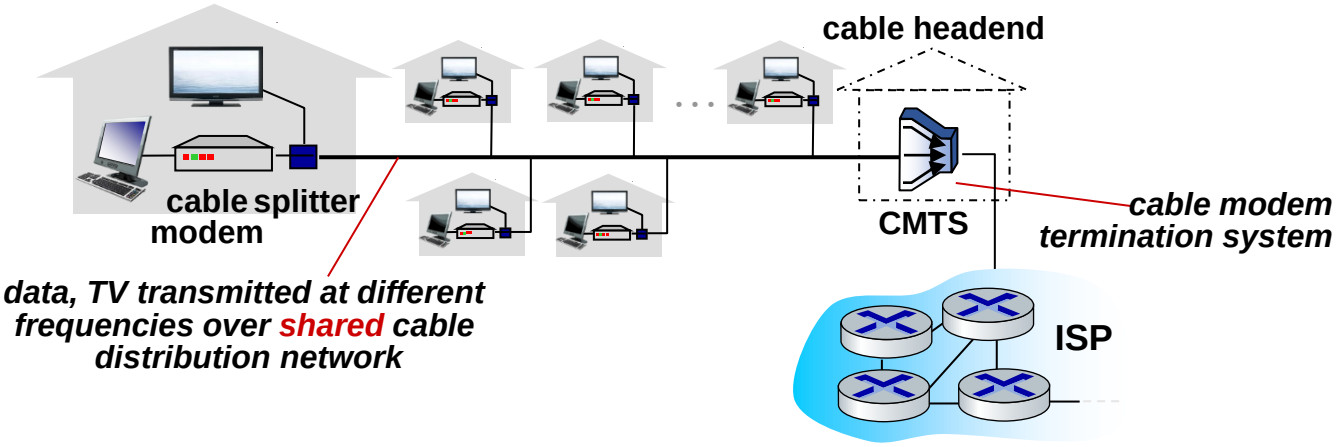


Access networks: cable-based access



frequency division multiplexing (FDM): different channels transmitted in different frequency bands

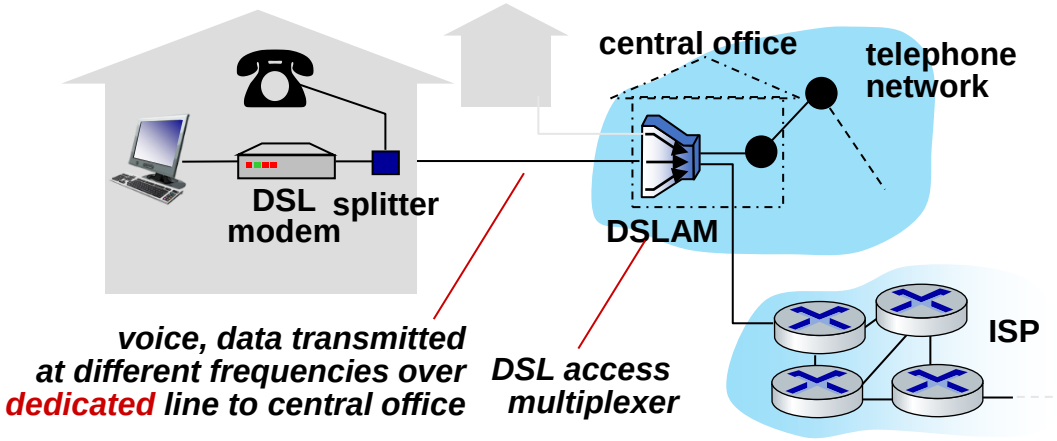
Access networks: cable-based access (2)



- HFC (Hybrid Fiber Coax)
  - asymmetric: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate

- Network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend

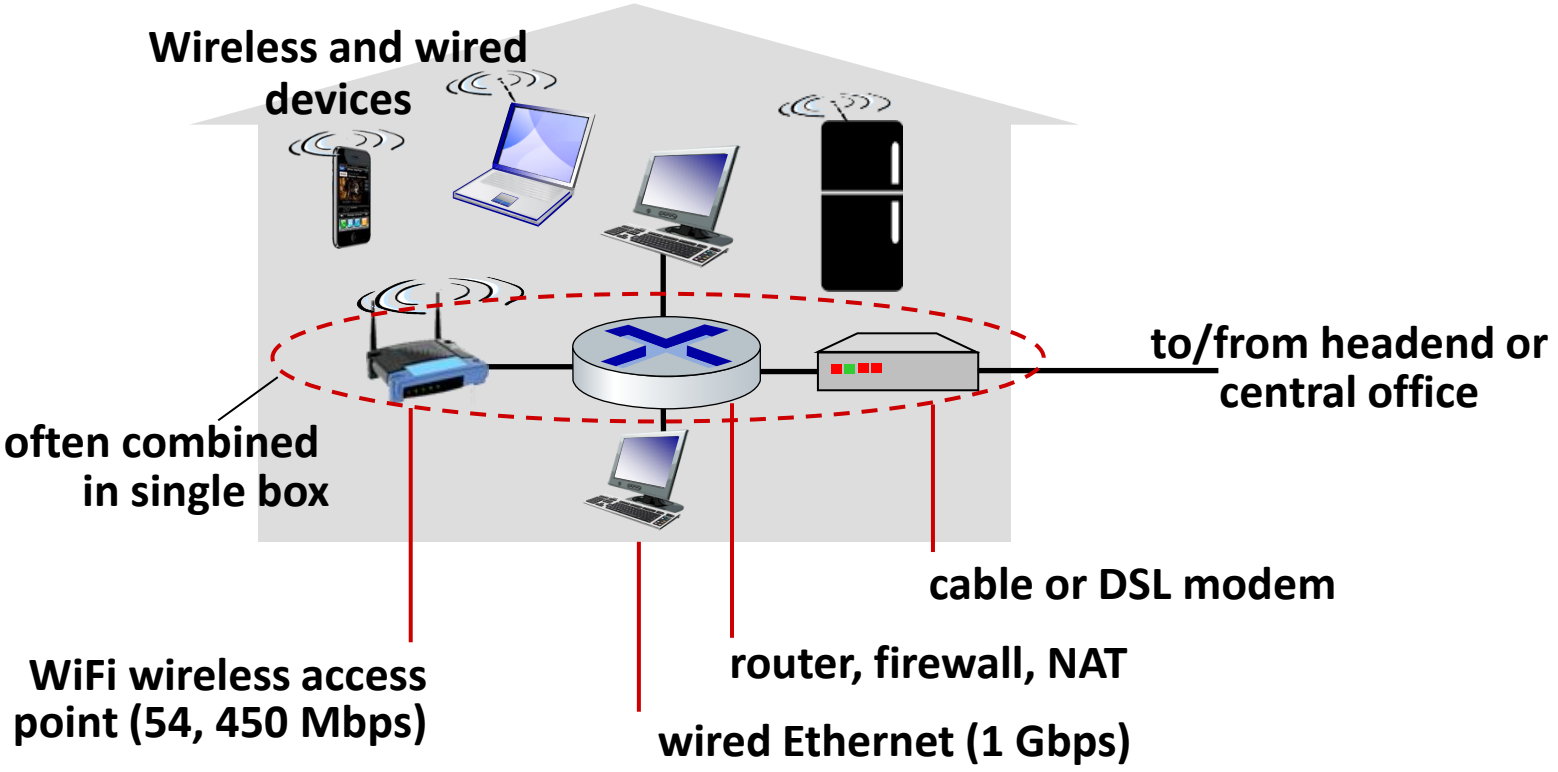
Access networks: digital subscriber line (DSL)



- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net

- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate

Access networks: home networks



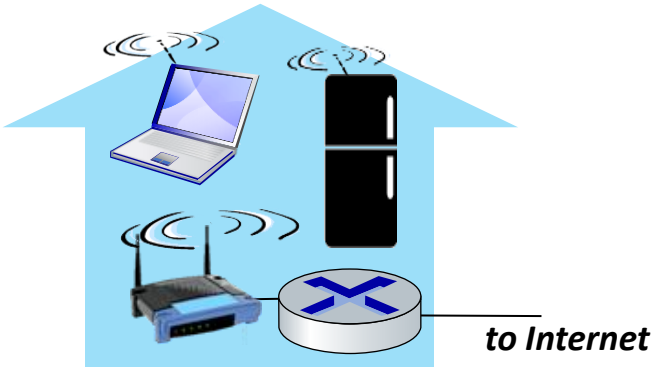
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka “access point”

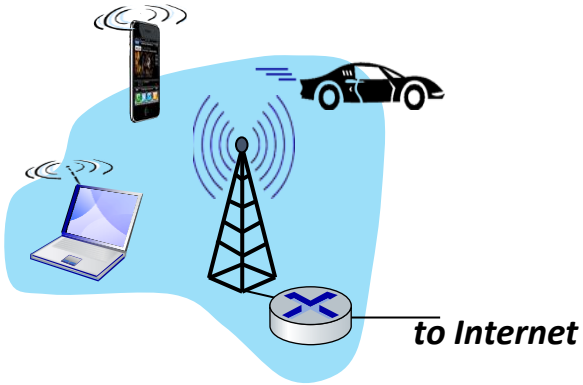
Wireless local area networks (WLANs)

- typically within or around building (~100 ft)
- 802.11b/g/n (WiFi): 11, 54, 450 Mbps transmission rate

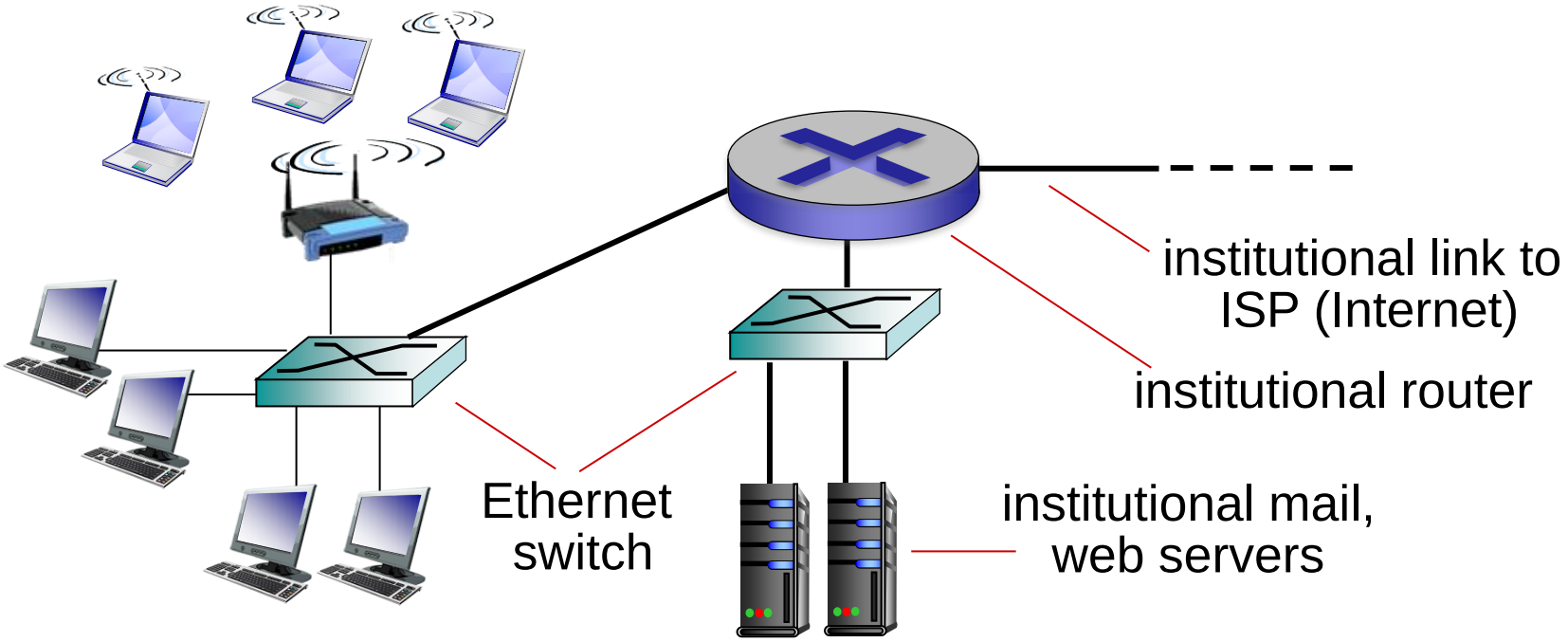


Wide-area cellular access networks

- provided by mobile, cellular network operator (10’s km)
- 10’s Mbps
- 4G & 5G cellular networks



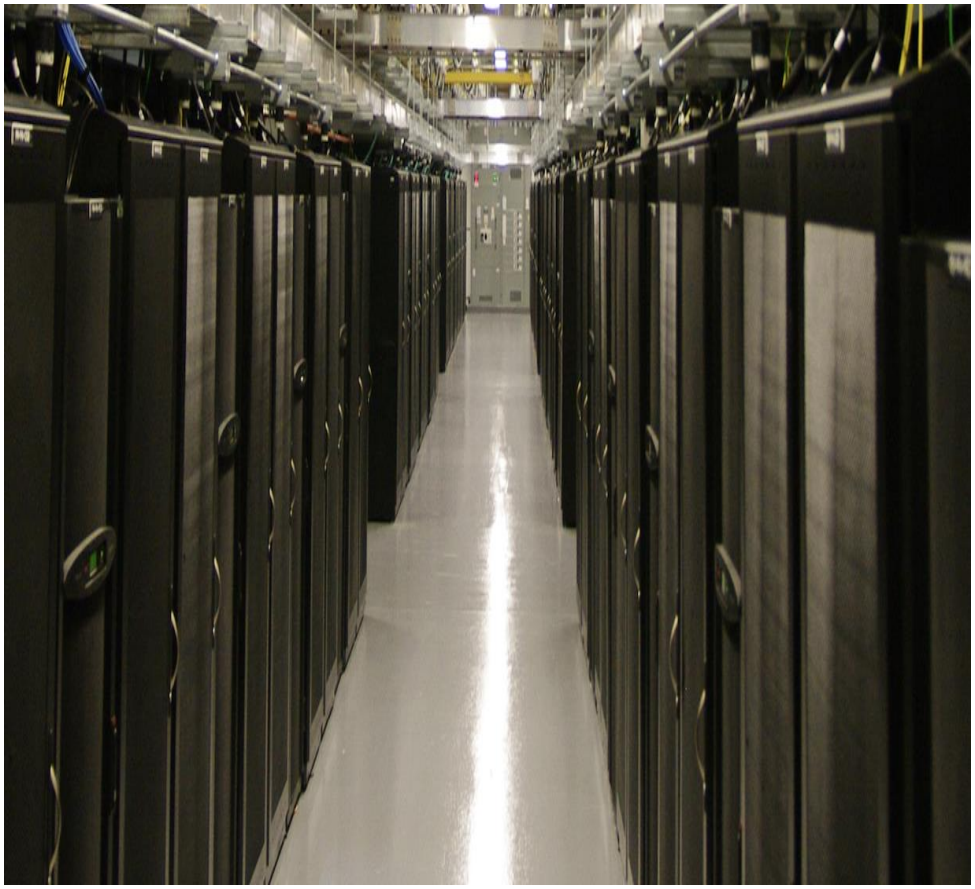
Enterprise access networks (Ethernet)



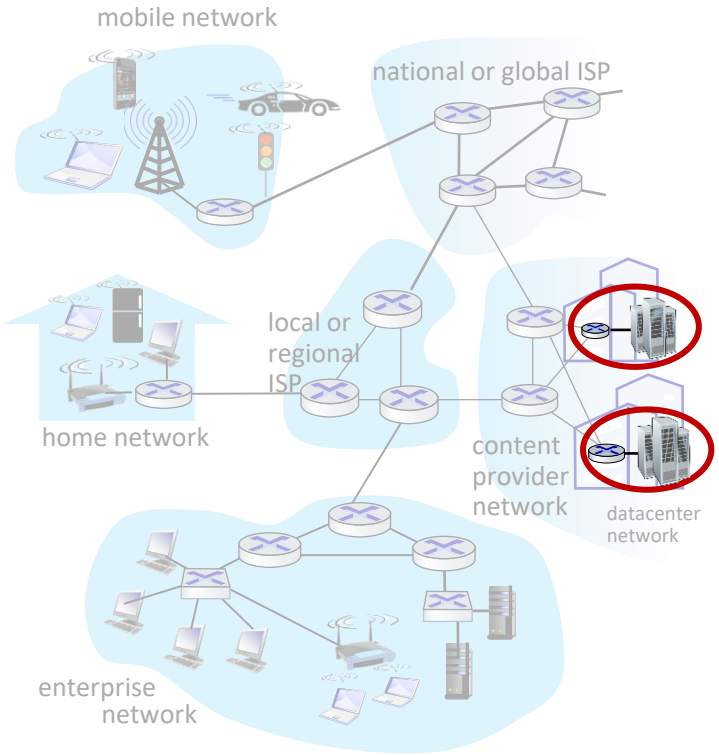
- typically used in companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

Access networks: data center networks

- high-bandwidth links (10s to 100s Gbps) connect hundreds to thousands of servers together, and to Internet



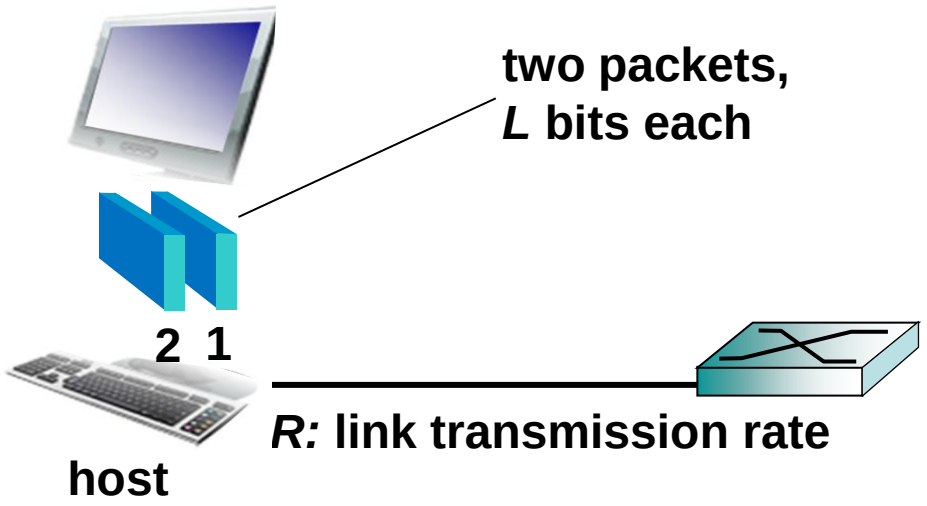
Courtesy: **Massachusetts Green High Performance Computing Center (mghpcc.org)**



Host: sends *packets* of data

host sending function:

- takes application message
- breaks into smaller chunks, known as *packets*, of length *L* bits
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*



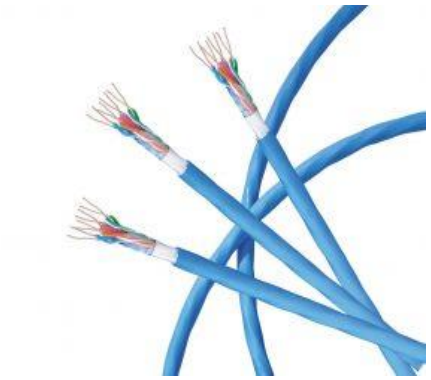
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

Links: physical media

- bit: propagates between transmitter/receiver pairs
- physical link: what lies between transmitter & receiver
- guided media:
  - signals propagate in solid media: copper, fiber, coax
- unguided media:
  - signals propagate freely, e.g., radio

Twisted Pair (TP)

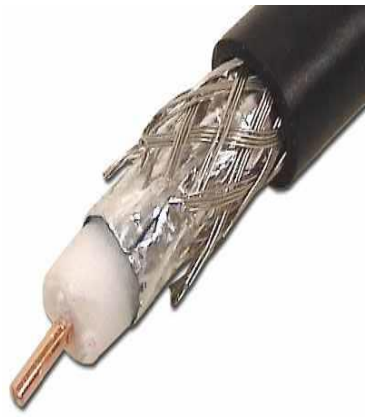
- two insulated copper wires
  - Category 5: 100Mbps, 1Gbps Ethernet
  - Category 6: 10Gbps Ethernet



Links: physical media (2)

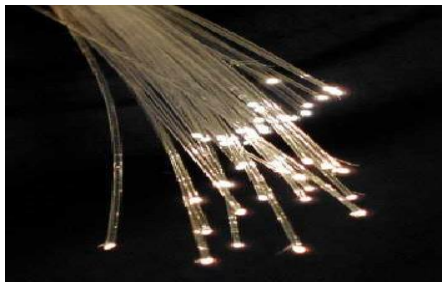
coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple frequency channels on cable
  - 100's Mbps per channel



fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (10's-100's Gbps)
- low error rate:
  - repeaters spaced far apart
  - immune to electromagnetic noise



Links: physical media (3)

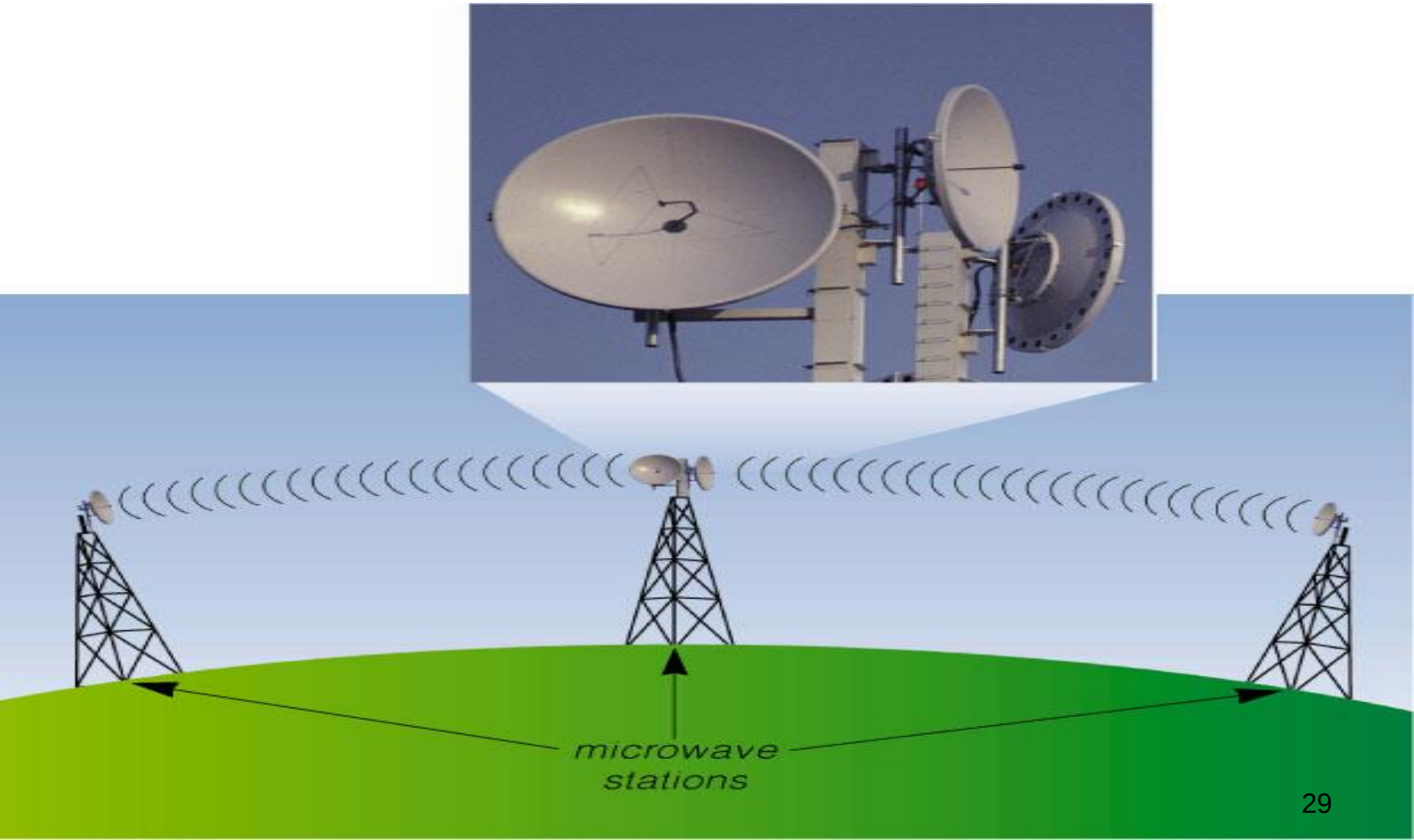
Wireless radio

- signal carried in various "bands" in electromagnetic spectrum
- no physical "wire"
- broadcast, "half-duplex" (sender to receiver)
- propagation environment effects:
  - reflection
  - obstruction by objects
  - Interference/noise

Radio link types:

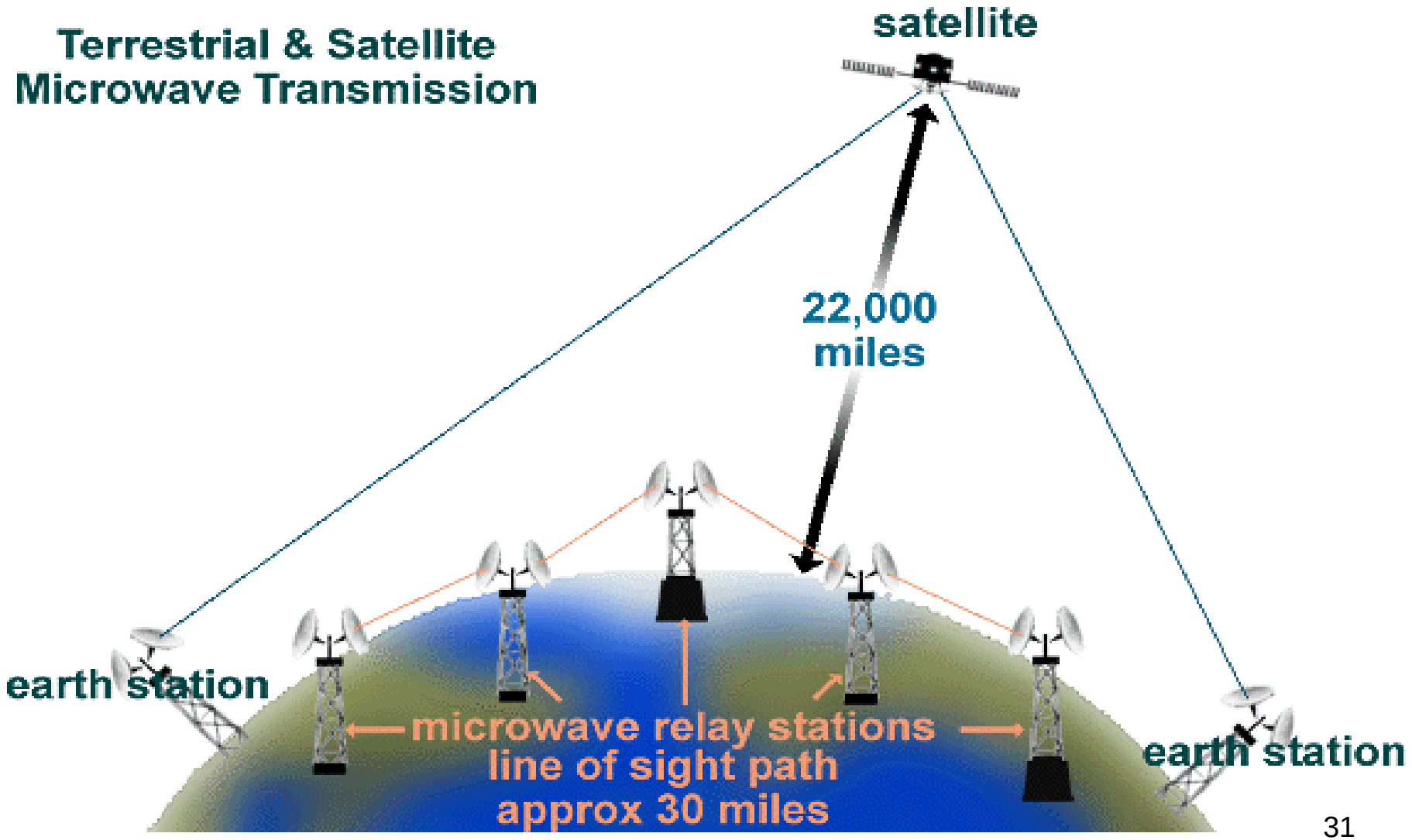
- **Wireless LAN** (WiFi)
  - 10-100's Mbps; 10's of meters
- **Wide-Area** (e.g., 4G cellular)
  - 10's Mbps over ~10 Km
- **Bluetooth:** cable replacement
  - short distances, limited rates
- **Terrestrial Microwave**
  - point-to-point; 45 Mbps channels
- **Satellite**
  - up to 45 Mbps per channel
  - 270 msec end-end delay

Links: physical media (4)
Terrestrial Microwave



Links: physical media (5)
Satellite

Links: physical media (6)
Terrestrial Microwave & Satellite



Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

1.3 Network Core: packet/circuit switching, internet structure

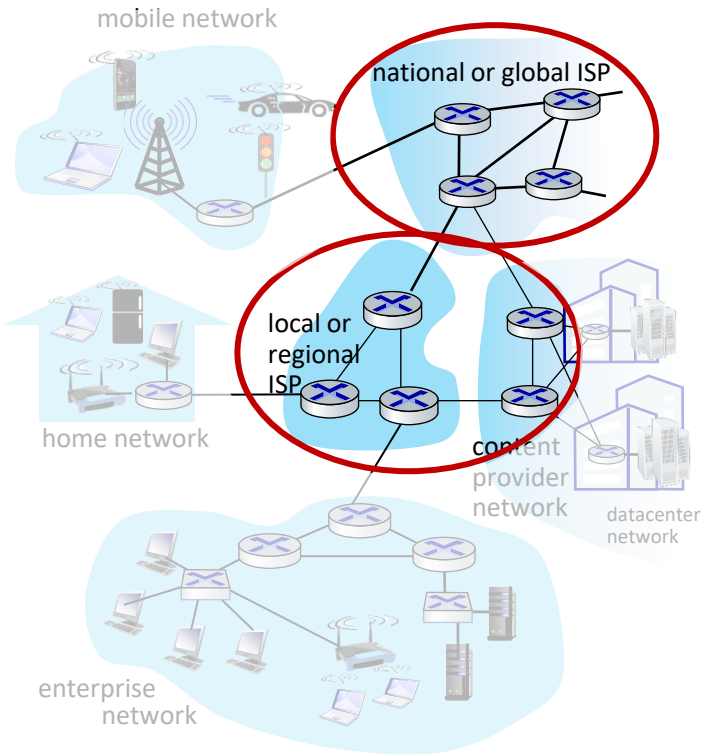
1.4 Performance: delay, loss and throughput

1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

The network core

- mesh of interconnected routers
- packet-switching: hosts break application-layer messages into *packets*
  - network forwards packets from one router to the next, across links on path from source to destination



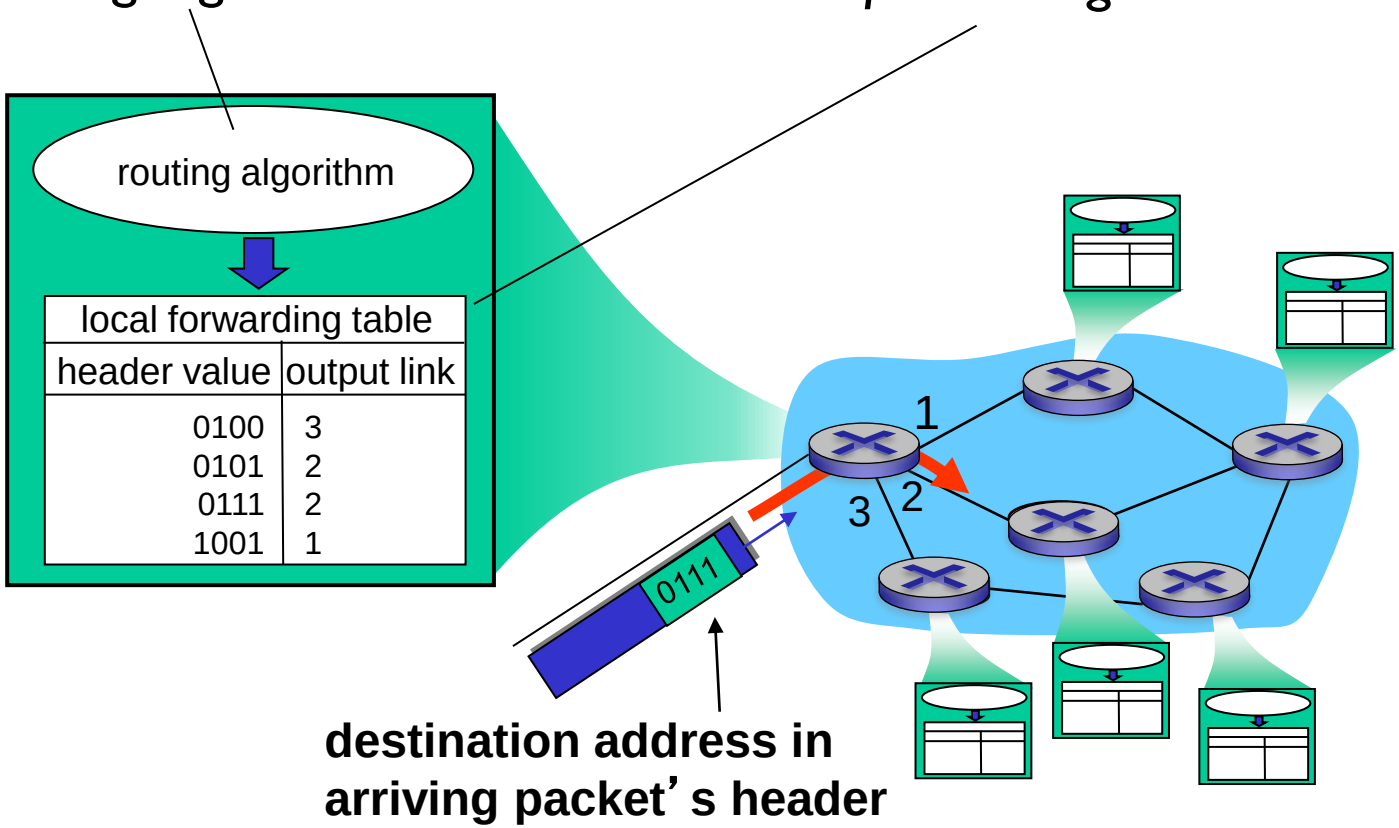
Two key network-core functions

routing: *global* action determines source-destination route (paths) taken by packets

- *routing algorithms*

forwarding: *local* action move packets from router's input to appropriate router output

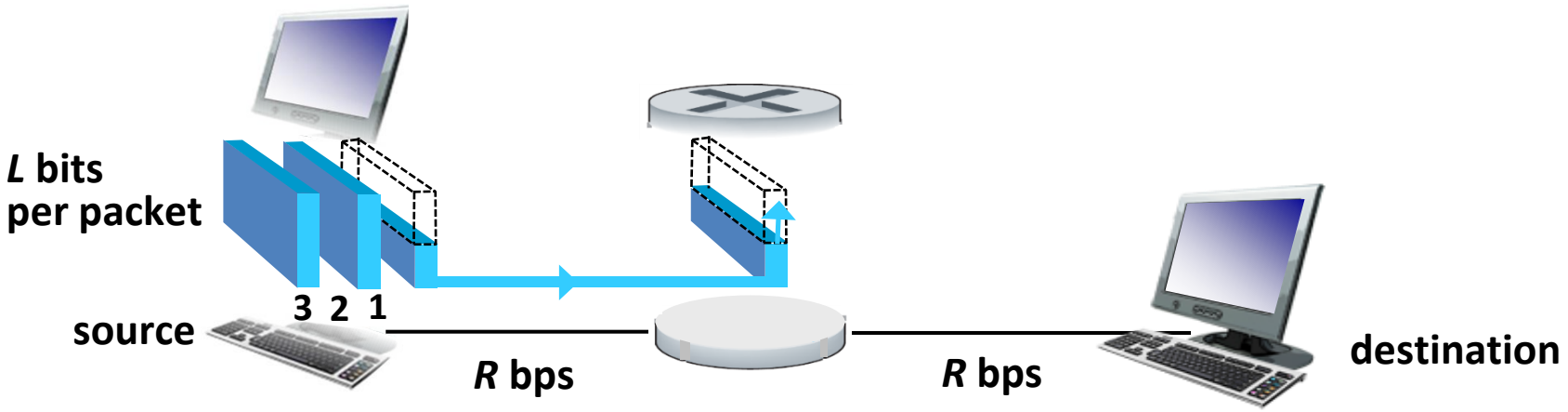
- *forwarding table*




routing
45 h
3,053 miles
48 h
3,309 miles
San Jose
Northampton
United States
Victoria
Seattle
WASHINGTON
Portland
OREGON
NEVADA
CALIFORNIA
Sacramento
San Francisco
Las Vegas
Los Angeles
San Diego
ARIZONA
Phoenix
Tucson
El Paso
BAJA CALIFORNIA
SONORA
CHIHUAHUA
NEW MEXICO
COLORADO
UTAH
IDAHO
MONTANA
WYOMING
NORTH DAKOTA
SOUTH DAKOTA
MINNESOTA
Minneapolis
IOWA
WISCONSIN
MICHIGAN
Chicago
Detroit
Toronto
Ottawa
Montreal
Quebec City
MAINE
VERMONT
NEW HAMPSHIRE
CT
RI
PENNSYLVANIA
New York
Philadelphia
NJ
MARYLAND
DE
OHIO
INDIANA
ILLINOIS
Indianapolis
Kansas City
MISSOURI
KANSAS
OKLAHOMA
Dallas
TEXAS
Austin
Houston
San Antonio
COAHUILA
LOUISIANA
New Orleans
ARKANSAS
MISSISSIPPI
ALABAMA
GEORGIA
Atlanta
Nashville
TENNESSEE
KENTUCKY
WEST VIRGINIA
VIRGINIA
NORTH CAROLINA
Charlotte
SOUTH CAROLINA
Jacksonville
Orlando
Tampa
FLORIDA


forwarding
forwarding
45 h
3,053 miles
48 h
3,309 miles
United States
Victoria
Seattle
WASHINGTON
Portland
NORTH DAKOTA
MINNESOTA
Minneapolis
SOUTH DAKOTA
WISCONSIN
MICHIGAN
Toronto
Quebec City
Montreal
Ottawa
MAINE
VERMONT
NEW HAMPSHIRE
Northampton
WYOMING
IOWA
Chicago
Detroit
ILLINOIS
INDIANA
OHIO
Indianapolis
Kansas City
COLORADO
KANSAS
MISSOURI
San Francisco
San Jose
CALIFORNIA
Las Vegas
KENTUCKY
Nashville
TENNESSEE
OKLAHOMA
ARKANSAS
Atlanta
Los Angeles
ARIZONA
NEW MEXICO
Dallas
MISSISSIPPI
ALABAMA
San Diego
Phoenix
Tucson
El Paso
TEXAS
BAJA CALIFORNIA
SONORA
CHIHUAHUA
Austin
Houston
LOUISIANA
New Orleans
San Antonio
COAHUILA
Tampa
FLORIDA
Cleveland

Packet-switching: store-and-forward

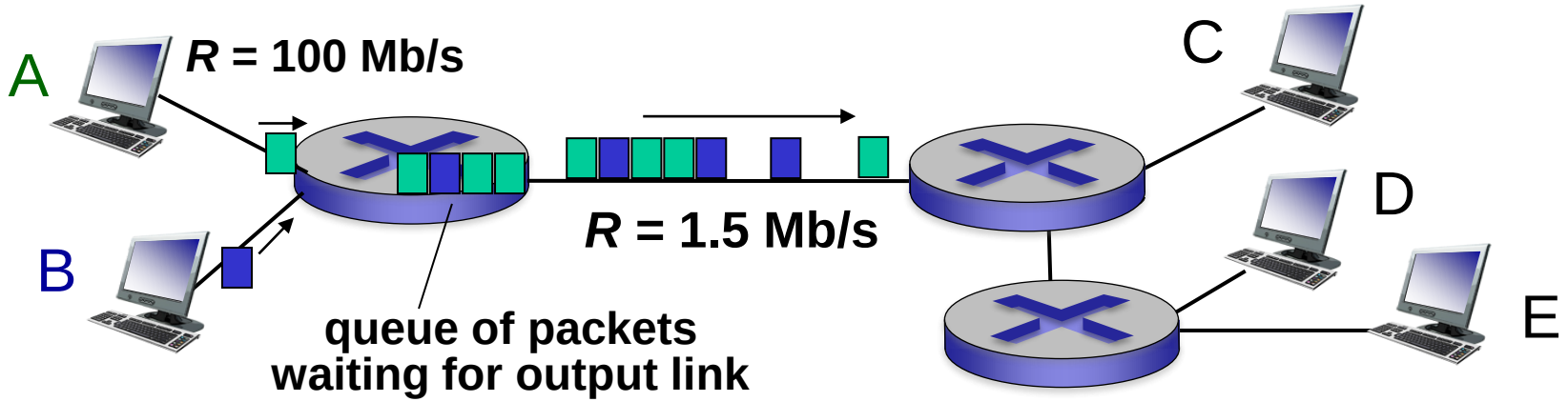


- takes *L*/*R* seconds to transmit (push out) *L*-bit packet into link at *R* bps
- *store and forward:* entire packet must arrive at router before it can be transmitted on next link

one-hop numerical example:

- *L* = 10 Kbits
- *R* = 100 Mbps
- one-hop transmission time = 0.1 msec

Packet Switching: queueing delay, loss



queuing and loss:

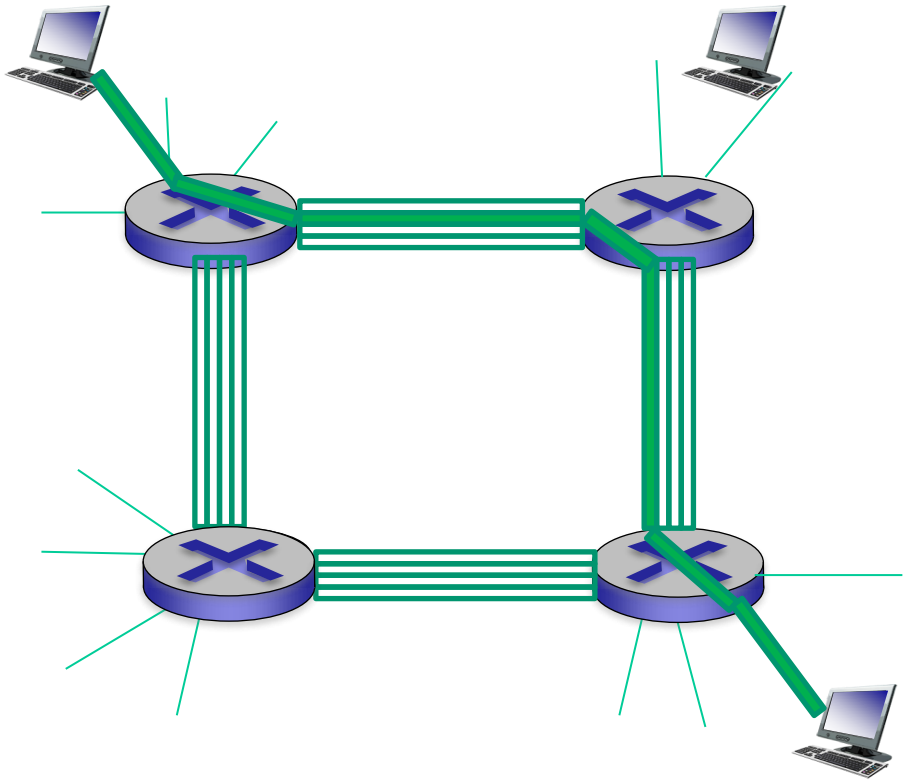
- if arrival rate (in bits) to link exceeds transmission rate (bps) of link for a period of time:
  - packets will queue, wait to be transmitted on link
  - packets can be dropped (lost) if memory (buffer) fills up

Queueing occurs when work arrives faster than it can be serviced

Alternative core: circuit switching

end-end resources allocated to, reserved for "call" between source & destination:

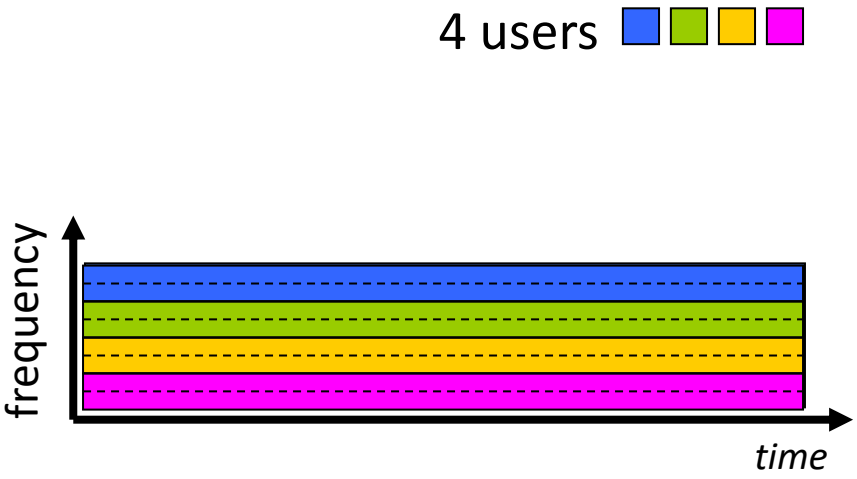
- in diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call *(no sharing)*
- commonly used in traditional telephone networks



Circuit switching: FDM and TDM

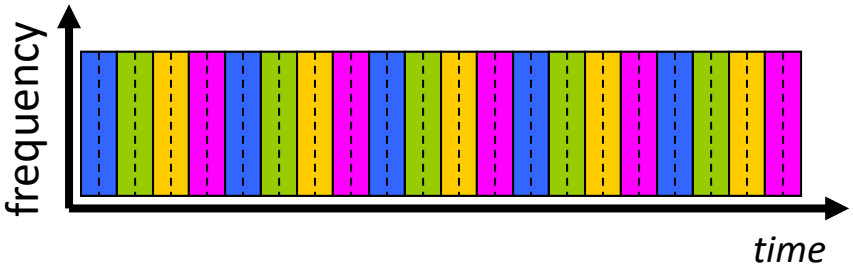
Frequency Division Multiplexing (FDM)

- optical, electromagnetic frequencies divided into (narrow) frequency bands
- each call allocated its own band, can transmit at max rate of that narrow band



Time Division Multiplexing (TDM)

- time divided into slots
- each call allocated periodic slot(s), can transmit at maximum rate of (wider) frequency band (only) during its time slot(s)

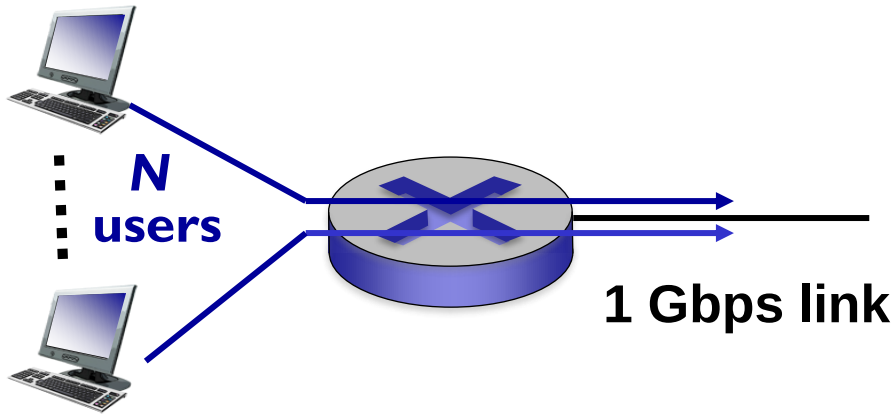


Packet switching versus circuit switching

packet switching allows more users to use network!

example:

- 1Gb/s link
- each user:
  - 100 Mb/s when "active"
  - active 10% of time



- *circuit-switching:*
  - 10 users

- *packet switching:*
  - with 35 users, probability > 10 active at same time is less than .0004 *

Q: how many users can use this network under circuit-switching and packet switching?

Q: what happens if > 35 users ?

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

- great for bursty data
  - resource sharing
  - simpler, no call setup

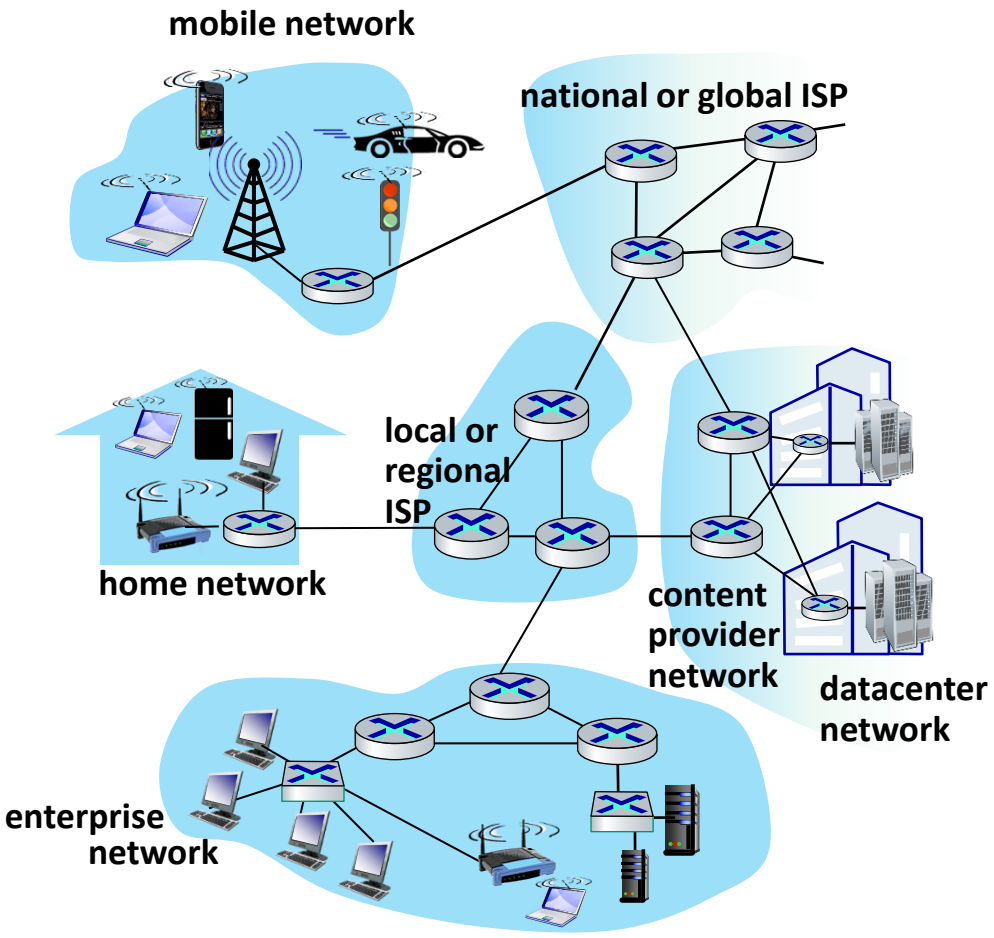
- excessive congestion possible: packet delay and loss
  - protocols needed for reliable data transfer, congestion control

- *Q:* How to provide circuit-like behavior?
  - bandwidth guarantees needed for audio/video apps
  - "It's complicated." We'll study various techniques that try to make packet switching as "circuit-like" as possible.

Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?

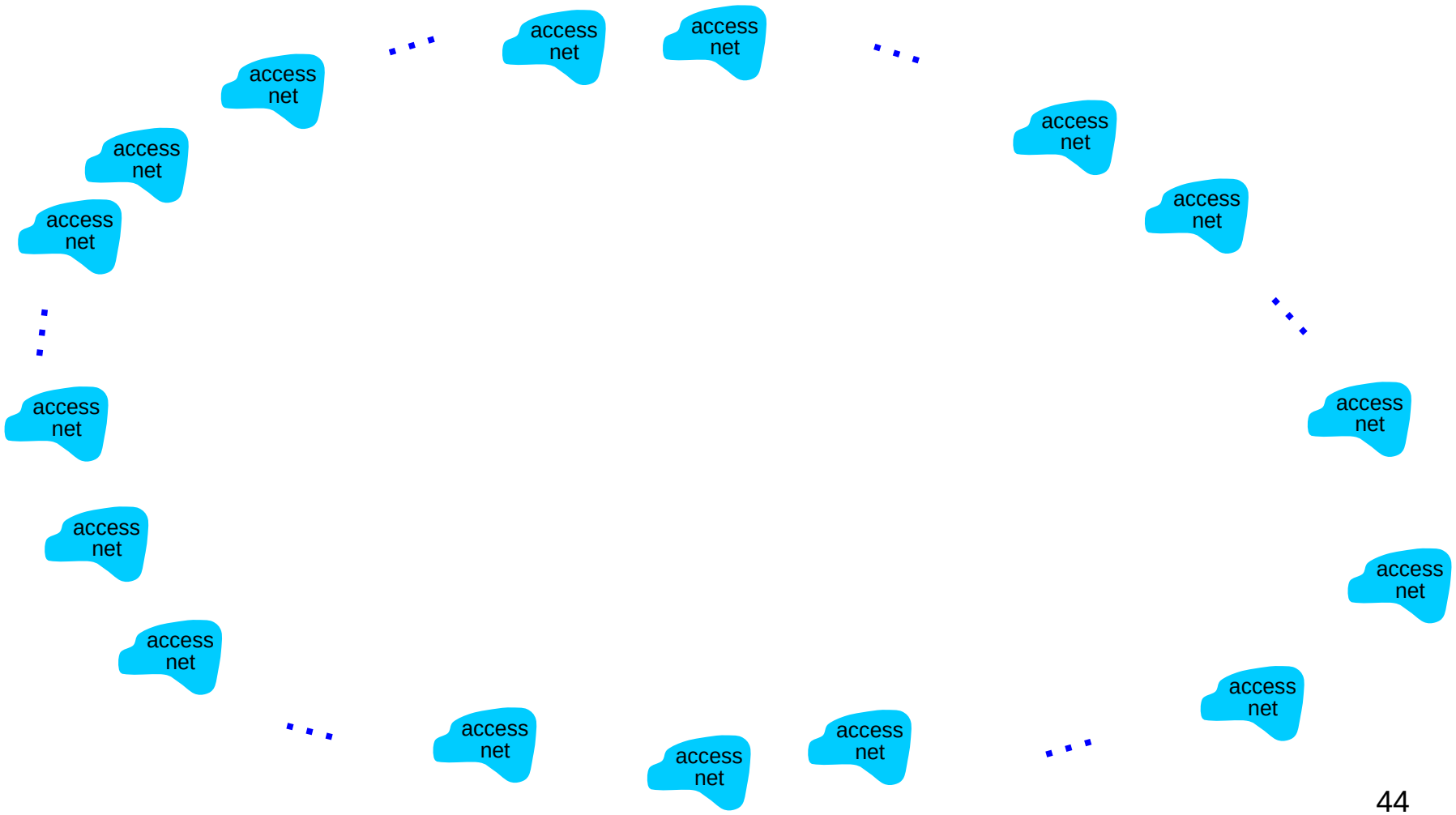
Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
- Access ISPs in turn must be interconnected.
  - so that any two hosts can send packets to each other
- Resulting network of networks is very complex
  - evolution was driven by economics and national policies



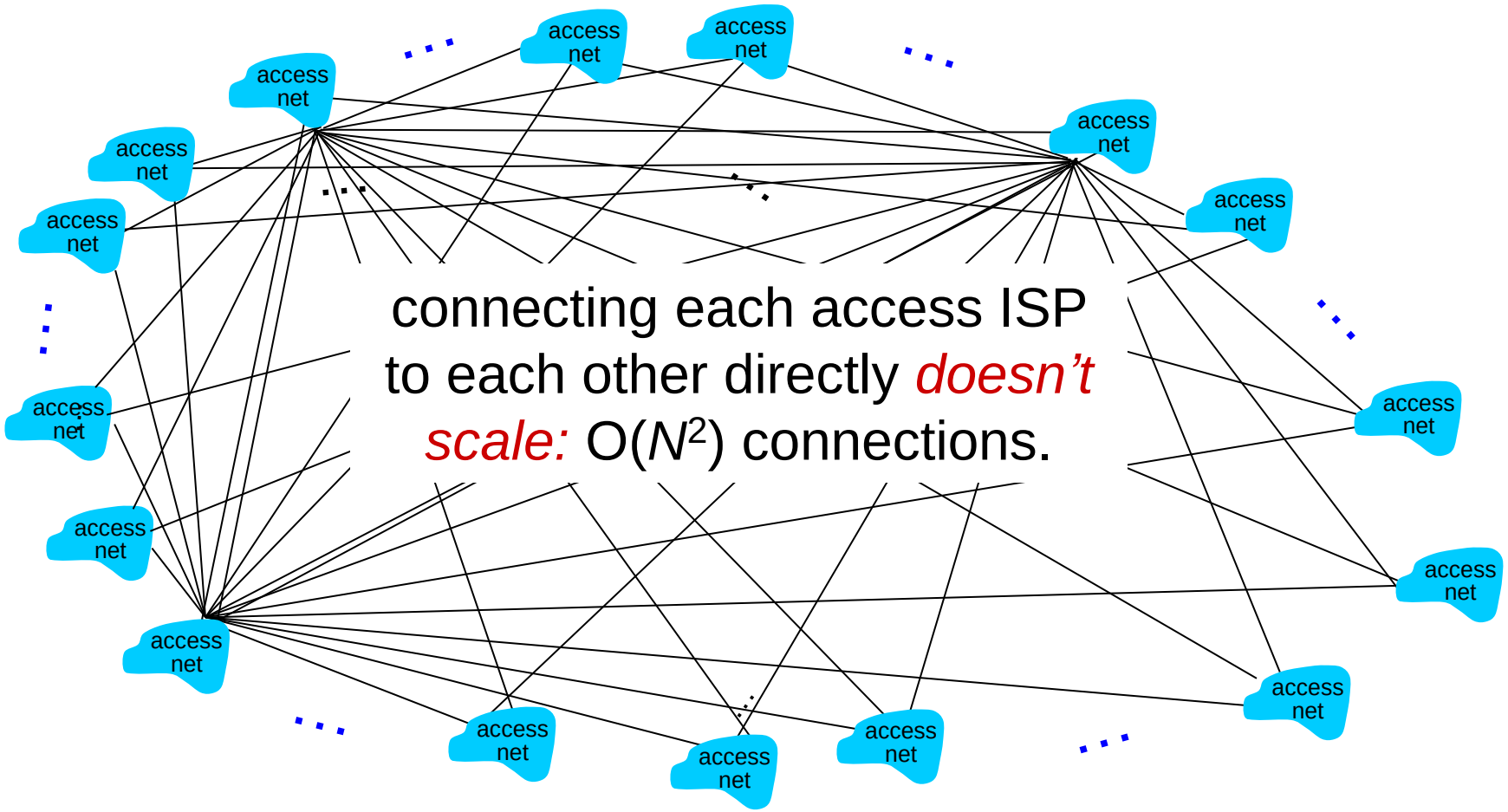
Internet structure: network of networks (2)

Question: given *millions* of access ISPs, how to connect them together?



Internet structure: network of networks (3)

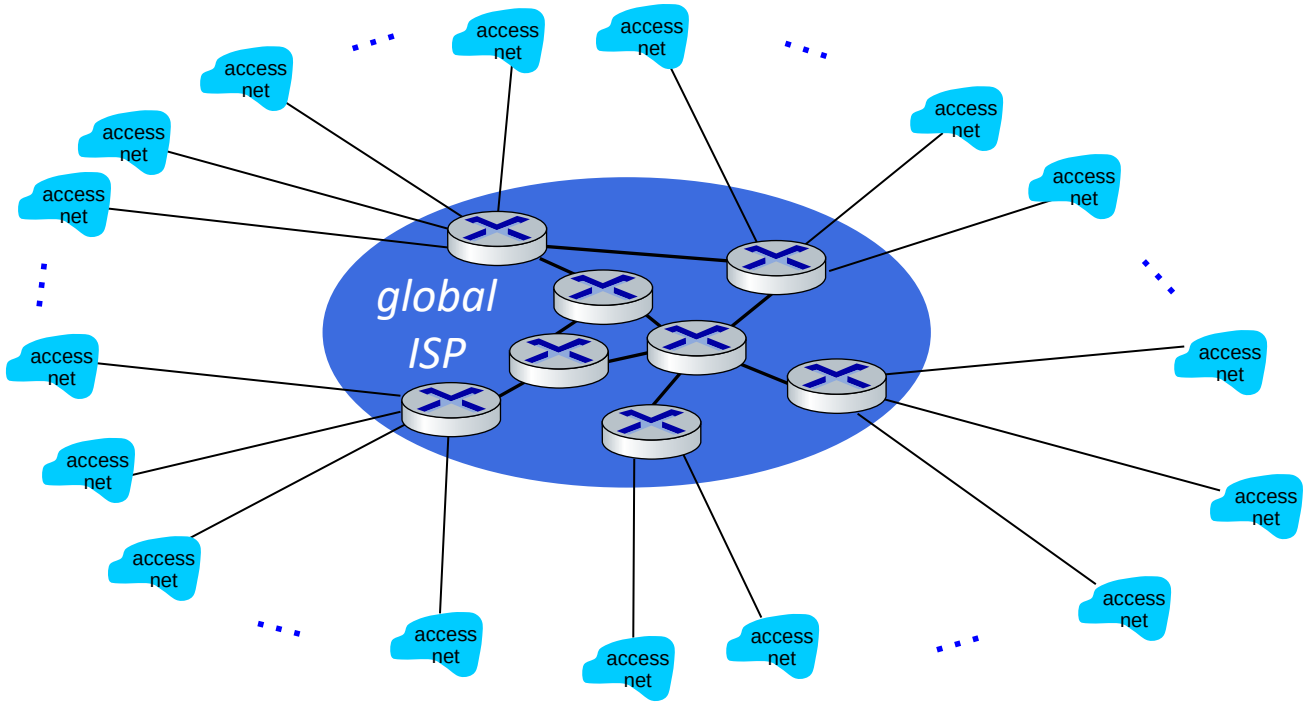
Option: connect each access ISP to every other access ISP?



Internet structure: network of networks (4)

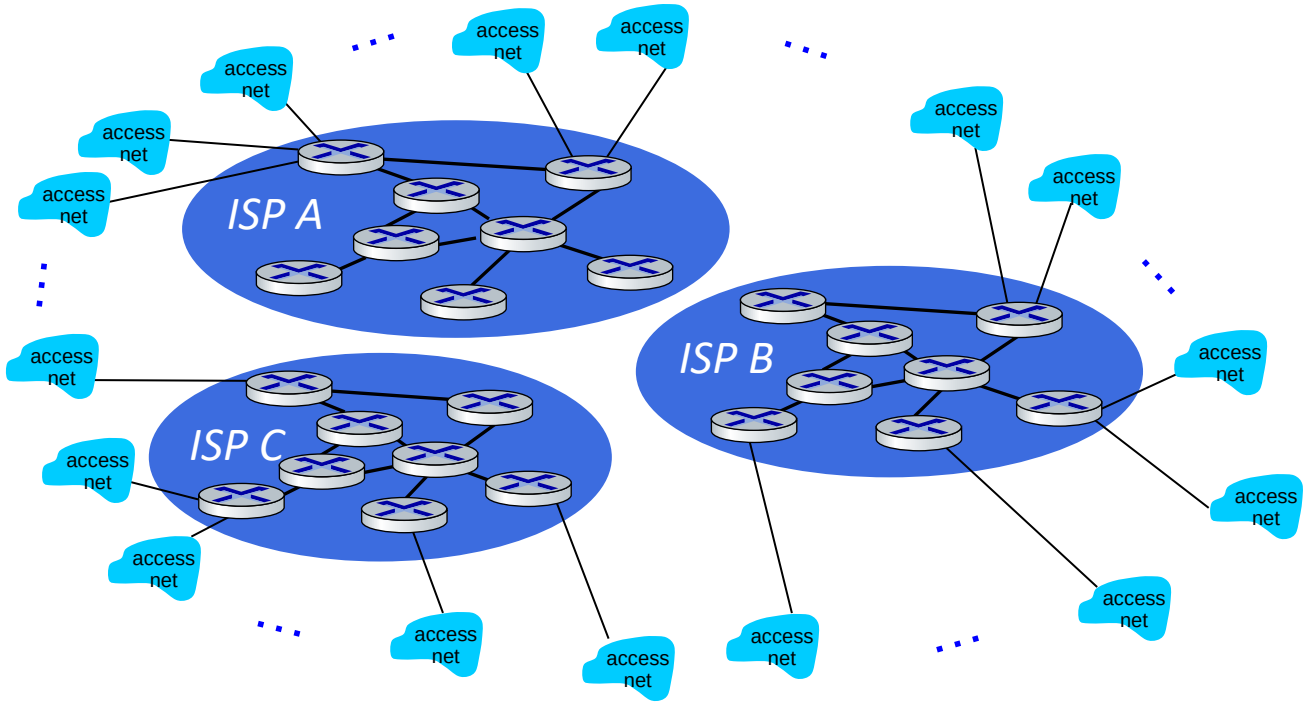
Option: connect each access ISP to one global transit ISP?

Customer and provider ISPs have economic agreement.



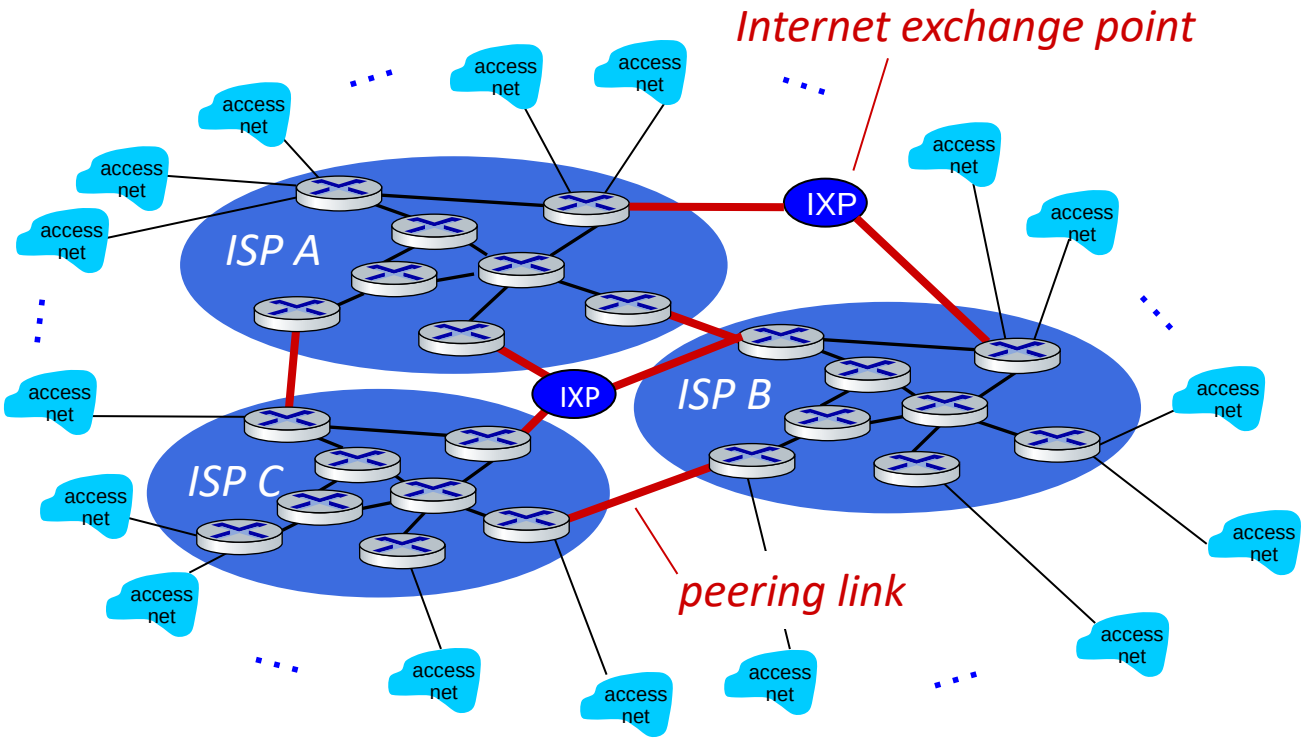
Internet structure: network of networks (5)

But if one global ISP is viable business, there will be competitors ….



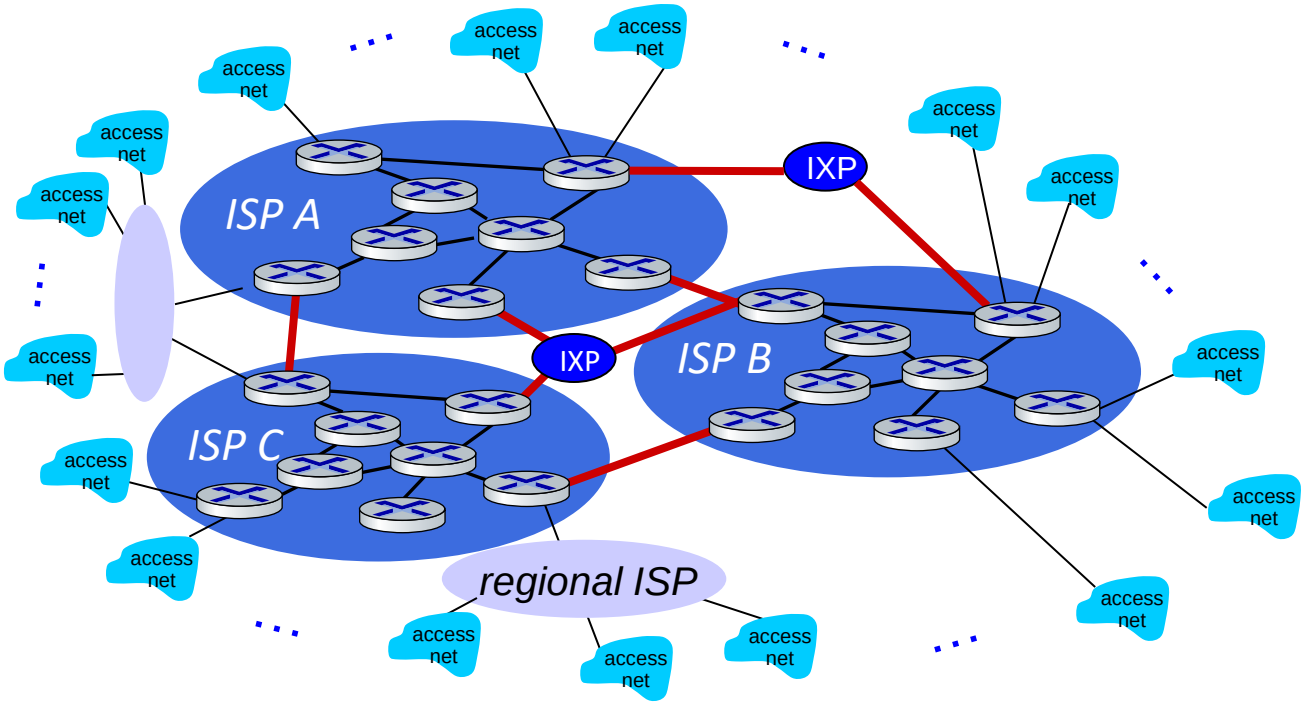
Internet structure: network of networks (6)

But if one global ISP is viable business, there will be competitors …. which must be interconnected



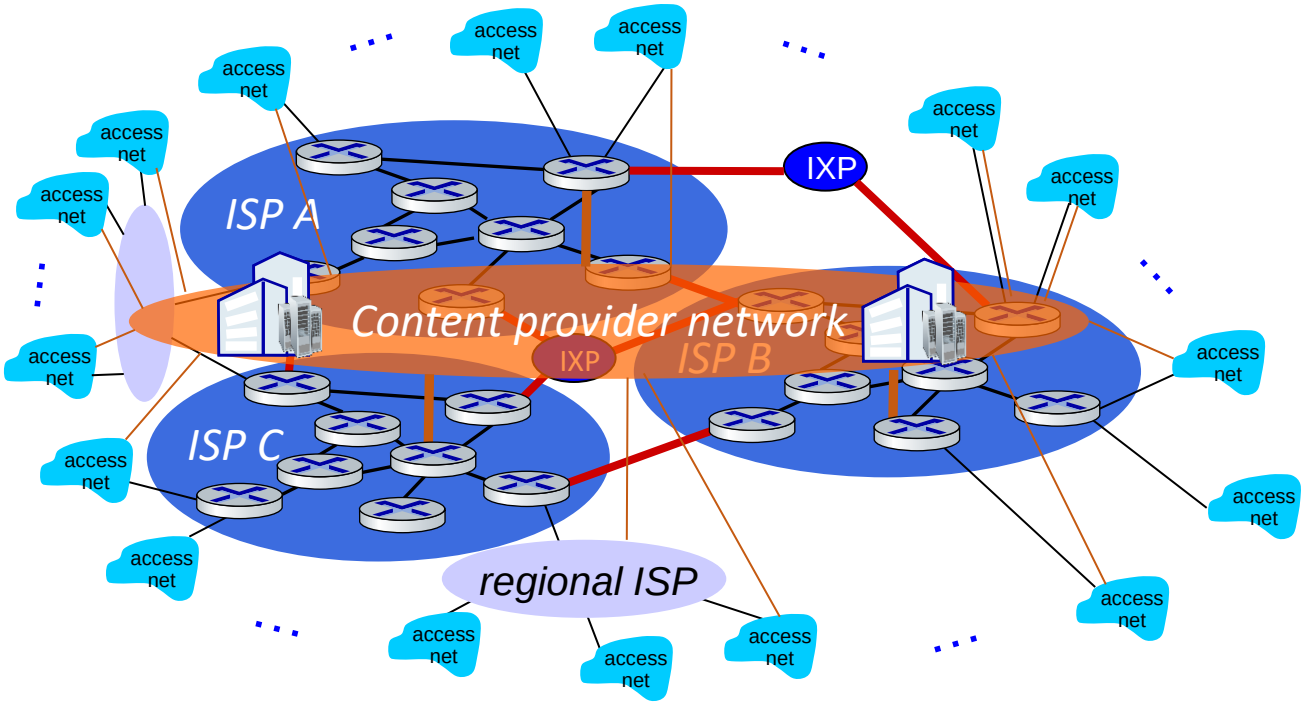
Internet structure: network of networks (7)

… and regional networks may arise to connect access nets to ISPs

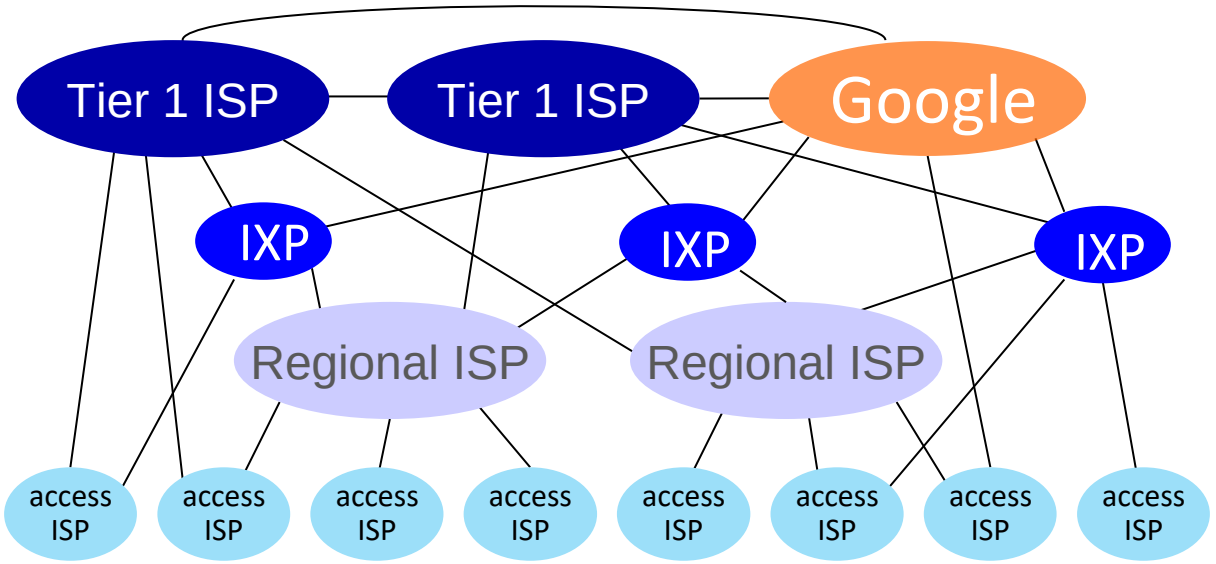


Internet structure: network of networks (8)

… and content provider networks (e.g., Google, Microsoft) may run their own network, to bring services and content close to end users



Internet structure: network of networks (9)



At "center": small # of well-connected large networks

- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage

- content provider networks (e.g., Google, Facebook): private network that connects its data centers to Internet, often bypassing tier-1, regional ISPs

Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

1.3 Network Core: packet/circuit switching, internet structure

1.4 Performance: delay, loss and throughput

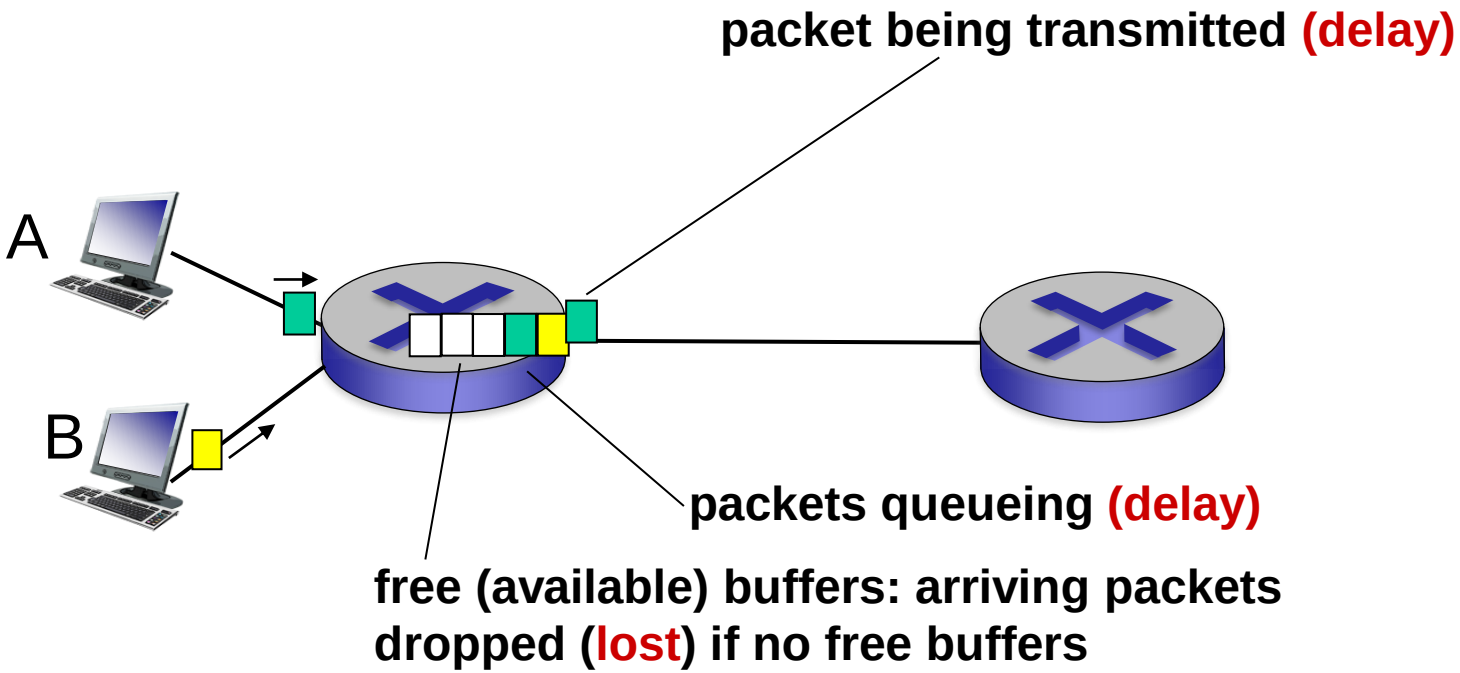
1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

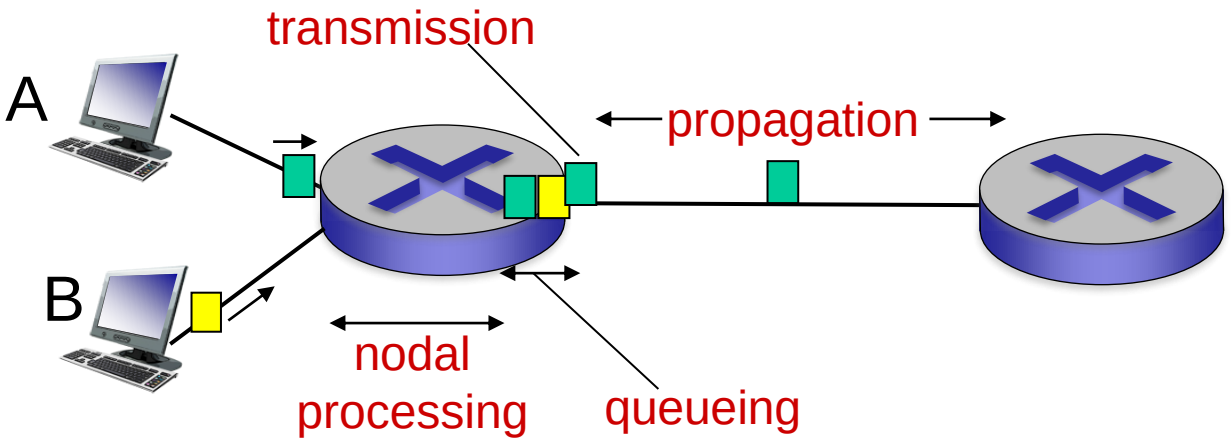
How do delay and loss occur in packet switching?

Packets *queue* in router buffers when:

- Packet arrival rate to a link (temporarily) exceeds the output capacity of this link
- Then, packets queue, wait for turn



Four sources of packet delay



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

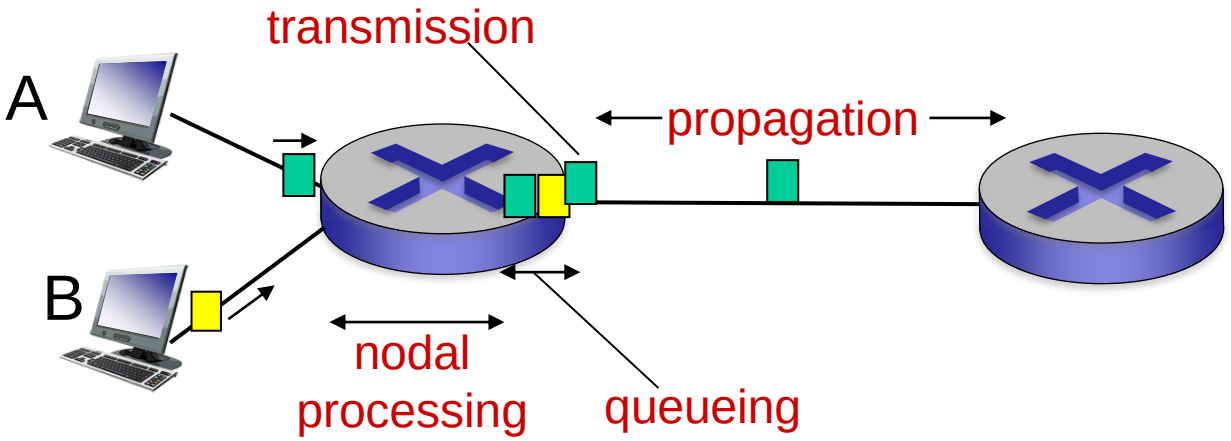
d_{proc}: nodal processing

- check bit errors
- determine output link
- typically < microseconds

d_{queue}: queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

Four sources of packet delay (2)



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

d_{trans}: transmission delay:

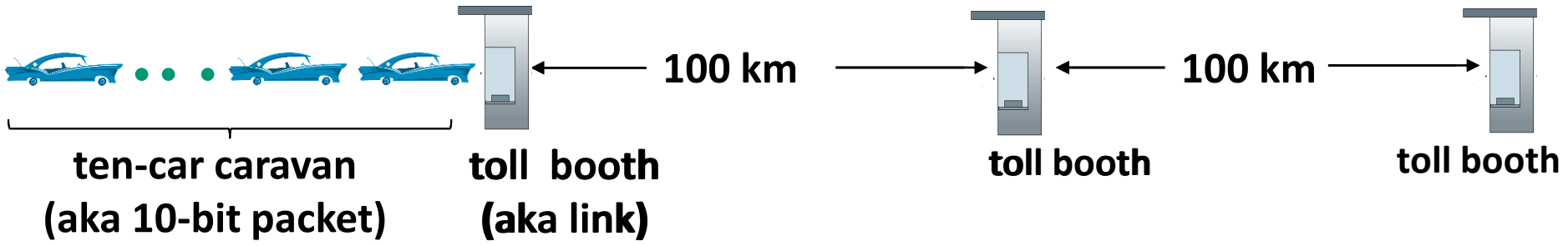
- L: packet length (bits)
- R: link *bandwidth (bps)*
- $d_{trans} = L/R$

d_{trans} and d_{prop} *very* different

d_{prop}: propagation delay:

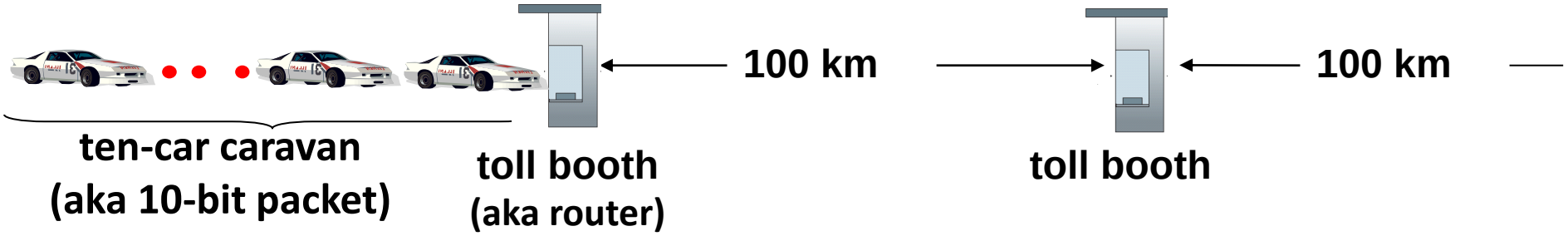
- d: length of physical link
- s: propagation speed (~2×10^8 m/sec)
- $d_{prop} = d/s$

Caravan analogy



- car ~ bit; caravan ~ packet; toll service ~ link transmission
- toll booth takes 12 sec to service car (bit transmission time)
- cars "propagate" (travel) at 100 km/hr
- *Q:* How long until caravan is lined up before 2nd toll booth?
- time to "push" entire caravan through toll booth onto highway = 10 * 12 = 120 sec (2 min)
- time for last car to propagate from 1st to 2nd toll booth: 100km / (100km/hr) = 1 hr
- *A:* 62 minutes

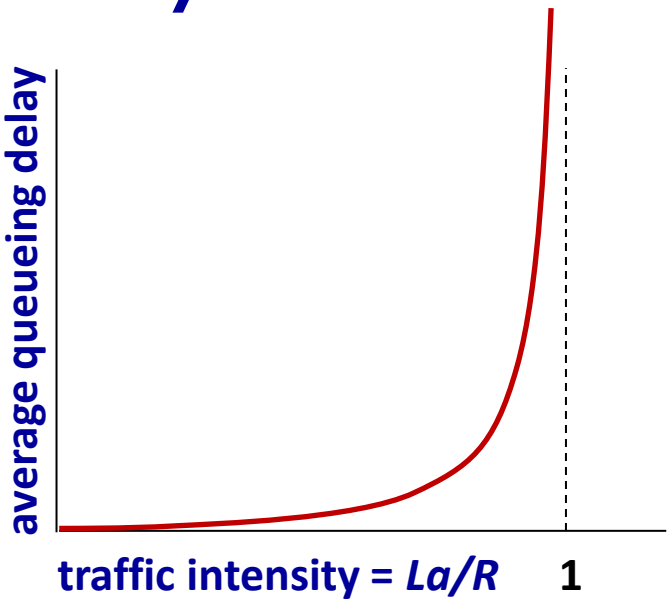
Caravan analogy (2)



- suppose cars now "propagate" (travel) at 1000 km/hr
- suppose toll booth now takes 1 min to service a car
- *Q:* Will cars arrive to 2nd booth before all cars serviced at first booth?
  - time to "push" entire caravan through toll booth onto highway = 10 * 1 = 10 min
  - time for a car to propagate from 1st to 2nd toll booth:
    100km / (1000km/hr) = 0.1 hr = (6 min)
- *A: Yes!* after 6 min, the first car arrives at the 2nd toll booth; while there are three cars still at 1st toll booth

Packet queueing delay

- *R:* link transmission rate (bps)
- *L:* packet length (bits)
- a: average packet arrival rate at the router
- *La: average bits arrival rate at the router*



Assume queue length is infinity (not practical)

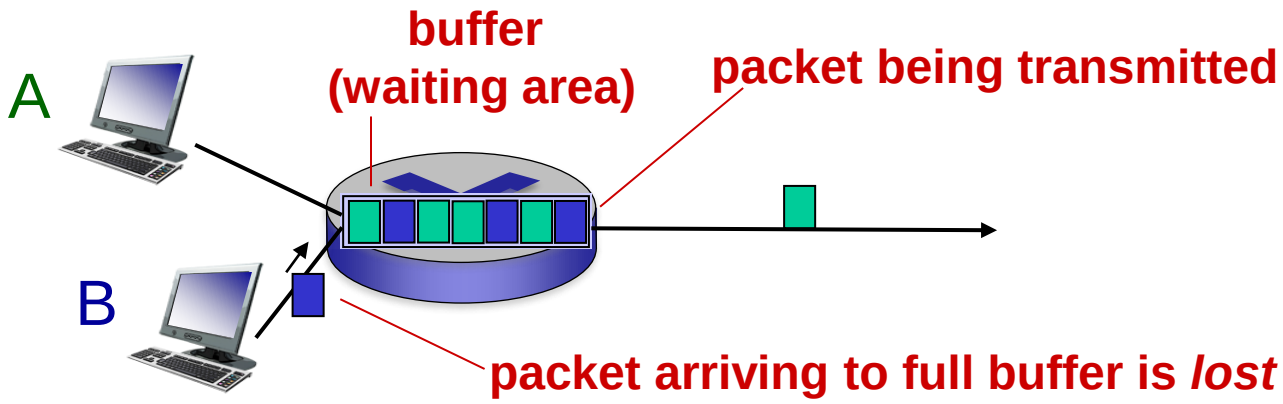
- *La/R* ≈ 0: avg. queueing delay small
- *La/R* ≥ 1: avg. queueing delay large
- *La/R* > 1: more "packets" arriving than can be serviced, average delay infinite!

***La/R* ≈ 0**

La/R* ≥ *1

Packet loss

- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



End-to-End delay

- is the delay from source to destination
- suppose there are *N*-1 routers between the source host and the destination host
- also suppose for the moment that the network is uncongested (so that queuing delays, d_{queue} , are negligible)
- The nodal delays accumulate and give an end-to-end delay:

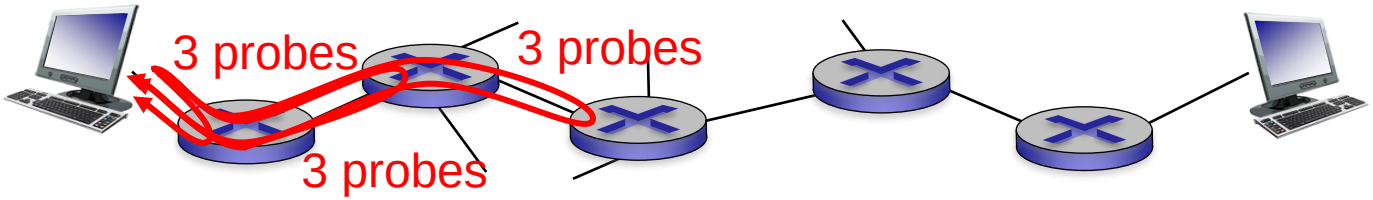
$$d_{end\text{-}end} = N\,(d_{proc} + d_{trans} + d_{prop})$$

"Real" Internet delays and routes

- what do "real" Internet delay & loss look like?

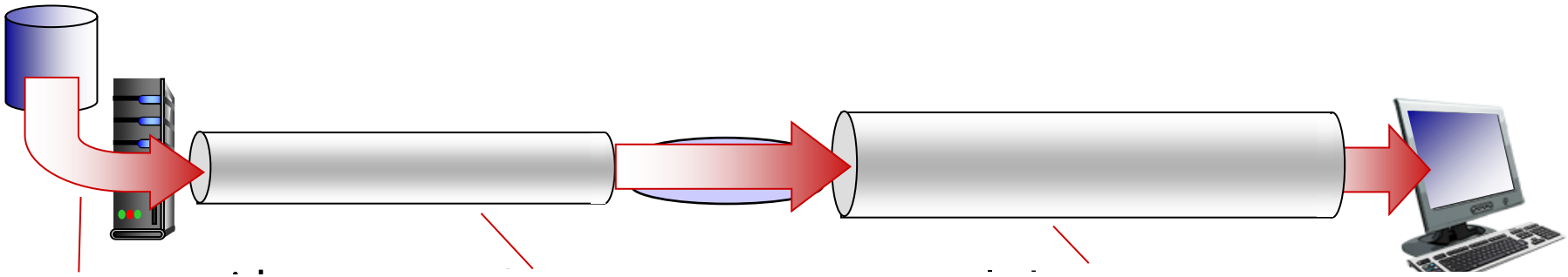
- `traceroute` program: provides delay measurement and route reconstruction from source to router along end-end Internet path towards destination.

- For all *i*:
  - sends 3 packets that will reach router *i* on path towards destination
  - router *i* will return packets to sender
  - sender times interval between transmission and reply.

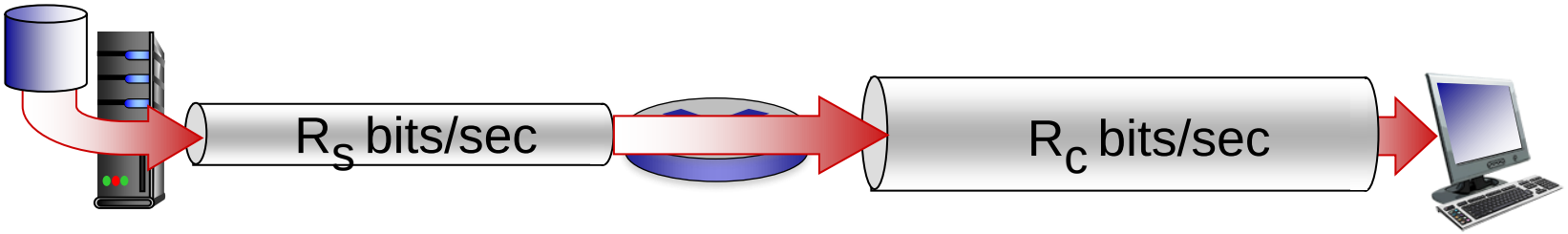


Throughput

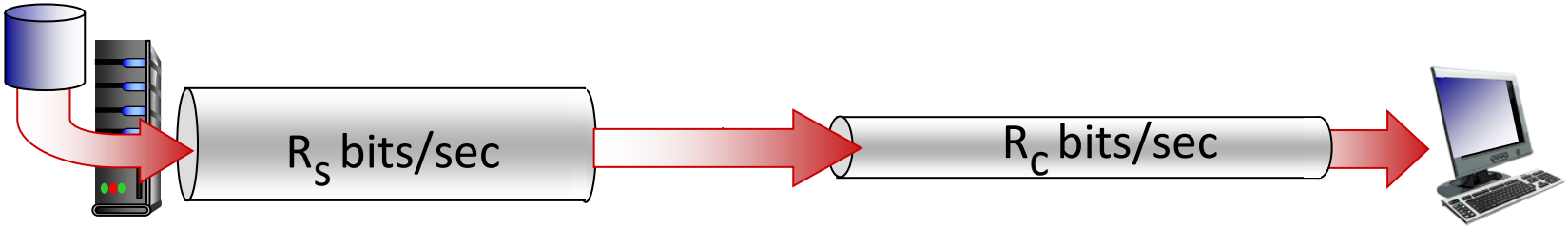
- *throughput:* rate (bits/time unit) at which bits transferred between sender/receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time
- consider transferring a large file from Host A to Host B across a computer network
- if the file consists of *F* bits and the transfer takes *T* seconds for Host B to receive all *F* bits, then the average throughput of the file transfer is *F/T* bits/sec



Throughput (2)

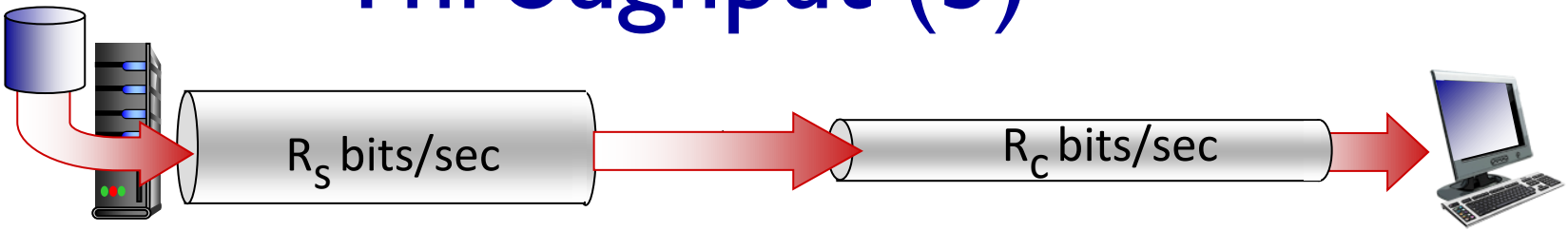


- $R_s < R_c$ What is average end-end throughput? Answer: R_s

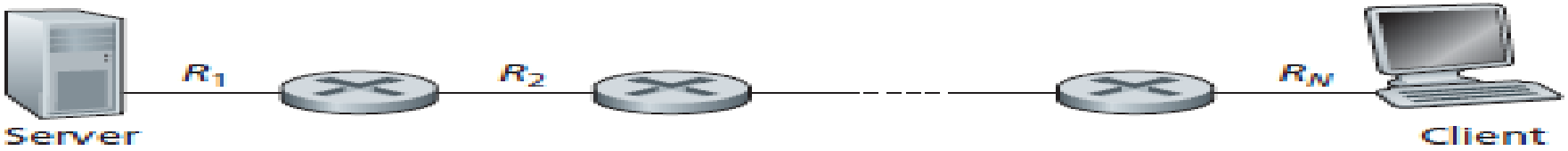


- $R_s > R_c$ What is average end-end throughput? Answer: R_c
- Throughput is min{Rs,Rc} = the bottleneck link transmission rate
- Bottleneck link is link on end-end path that constrains end-end throughput
- Having determined the throughput, we can now approximate the time it takes to transfer a large file of ***F*** bits from server to client as $F \div min\{Rs, Rc\}$

Throughput (3)



- Suppose downloading an MP3 file of F = 32 Mbits, the server has a transmission rate of R_s = 2 Mbps, and have an access link of R_c = 1 Mbps, so the time needed to transfer the file is then 32 seconds.

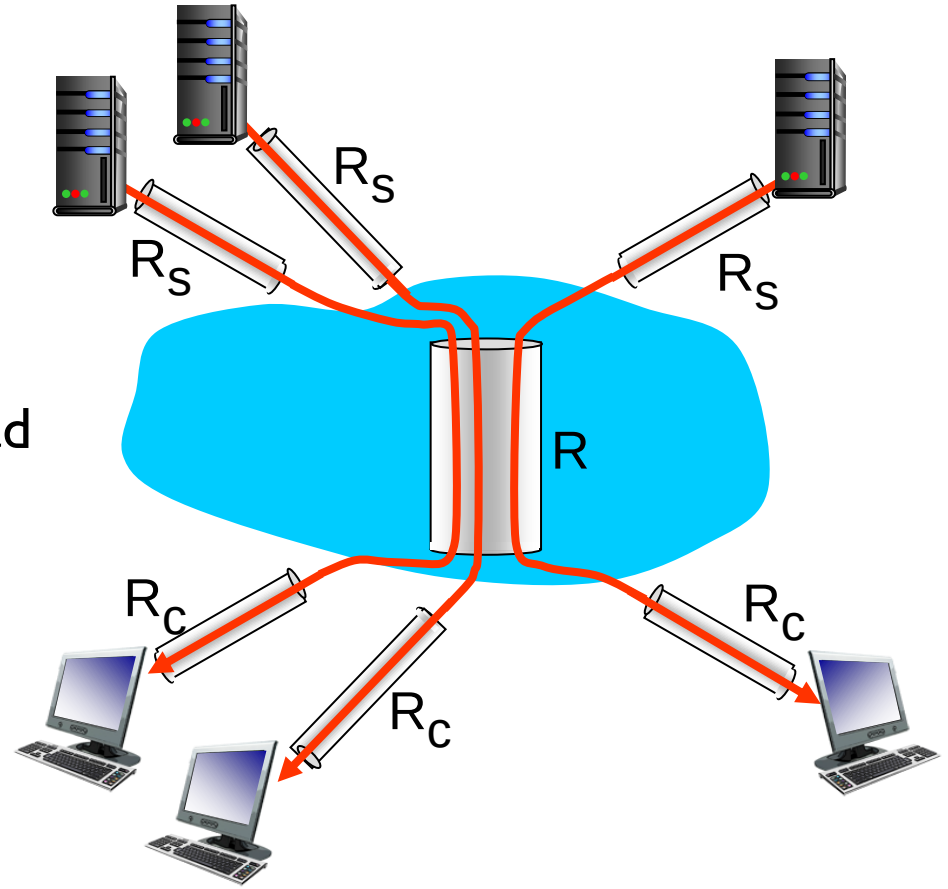


- if network with N links between the server and the client, with the transmission rates of the N links being $R_1, R_2,..., R_N$

- Applying the same analysis as for the two-link network, we find that the throughput for a file transfer from server to client is min{$R_1, R_2,..., R_N$}, which is once again the transmission rate of the bottleneck link along the path between server and client.

Throughput: Internet scenario

- suppose that:
  - all servers' access links have the same rate R_s,
  - all clients' access links have the same rate R_c
- if the rate of the common link, R, is larger than both R_s and R_c, then end-to-end throughput for each download will again be $\min\{R_s, R_c\}$
- else per-connection end-end throughput: $min(R_c, R_s, R/10)$
- E.g., R_s= 2 Mbps, R_c= 1 Mbps, R = 5 Mbps, and a common link divides its transmission rate equally among the 10 downloads, then each download with 500 kbps of throughput



10 connections simultaneously and fairly share backbone bottleneck link of R bits/sec transmission rate

Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

1.3 Network Core: packet/circuit switching, internet structure

1.4 Performance: delay, loss and throughput

1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

Protocol layers and service models

Networks are complex, with many "pieces":

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

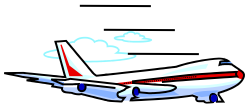
Question:

is there any hope of *organizing* structure of network?

…. or at least our discussion of networks?

Protocol layers and service models (2)

Example: organization of air travel



end-to-end transfer of person plus baggage

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing

airplane routing

How would you *define/discuss* the *system* of airline travel?

- a series of steps, involving many services

Protocol layers and service models (3)

Example: organization of air travel functionality (layering)

Layers:

- each layer implements a service to the layer above via its own internal-layer actions
- relying on services provided by the layer below

Why layering?

Approach to design/discuss complex systems:

- explicit structure allows identification of relationships among complex system's pieces
- modularization eases maintenance and updating of system
  - change of implementation of layer's service is transparent to rest of system
  - e.g., change in gate procedure doesn't affect the rest of air travel system

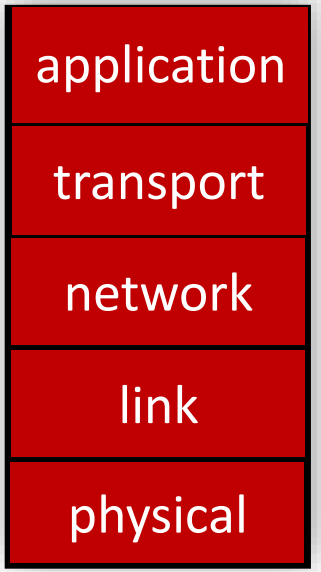
Internet protocol layering?

- network designers organize protocols—and the network hardware and software that implement the protocols— in layers.
- the Internet protocol stack consists of five layers: the physical, link, network, transport, and application layers
- when taken together, the protocols of the various layers are called the protocol stack.
- layer-*n* protocol is distributed among the end systems, packet switches, and other components that make up the network, that is, there's often a piece of a layer-n protocol in each of these network components
- A protocol layer can be implemented in software, in hardware, or in a combination of the two

Internet protocol layering? (2)

Internet protocol stack

- *application:* supporting network applications
  - HTTP, IMAP, SMTP, DNS
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi), PPP
- *physical:* bits "on the wire"

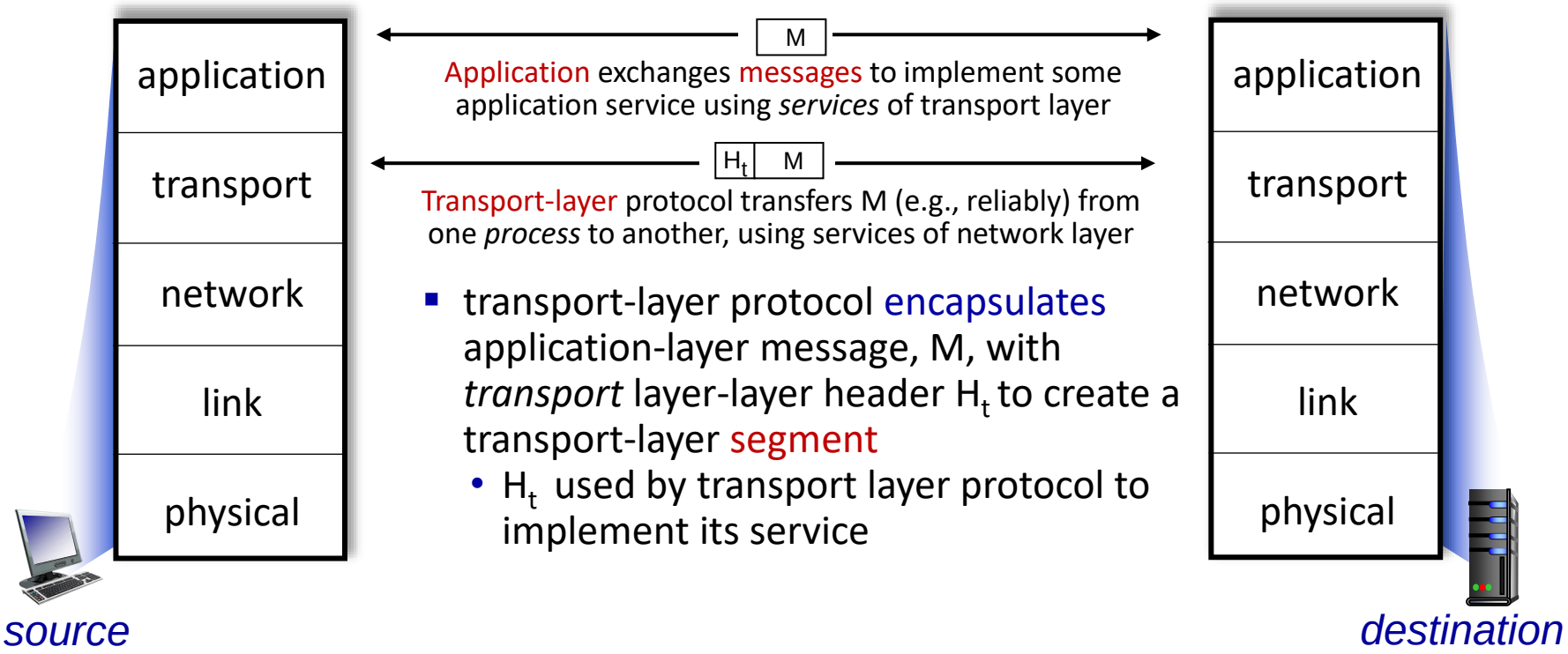


Internet protocol layering? (3)

- Application-layer and transport-layer protocols are almost always implemented in software in the end systems
- network layer is often a mixed implementation of hardware and software
- physical layer and data link layers are responsible for handling communication over a specific link, so they are typically implemented in a network interface card associated with a given link
- one potential drawback of layering is that one layer may duplicate lower-layer functionality. For example, many protocol stacks provide error recovery on both a per-link basis and an end-to-end basis
- A second potential drawback is that functionality at one layer may need information that is present only in another layer; this violates the goal of separation of layers

Internet protocol layering? (4)

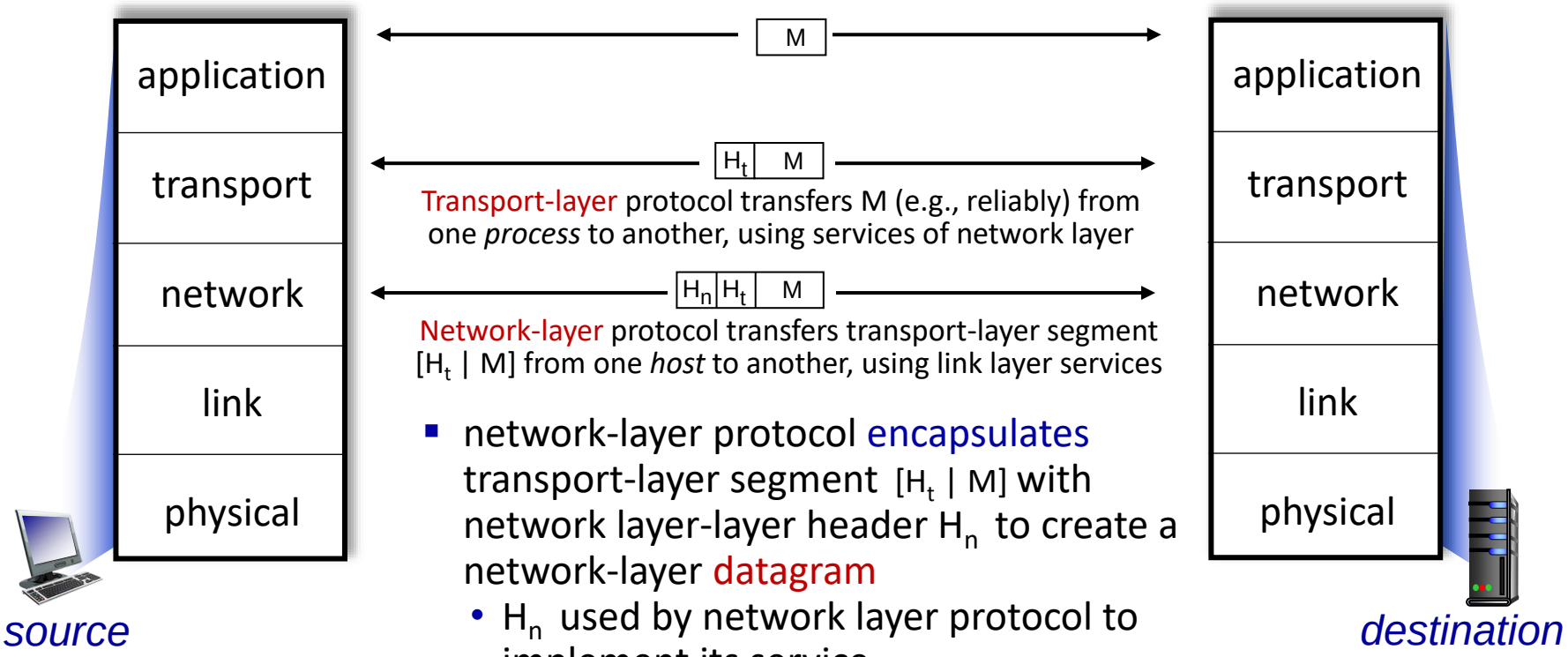
Services, Layering and Encapsulation



- transport-layer protocol encapsulates application-layer message, M, with *transport* layer-layer header H_t to create a transport-layer segment
  - H_t used by transport layer protocol to implement its service

Internet protocol layering? (5)

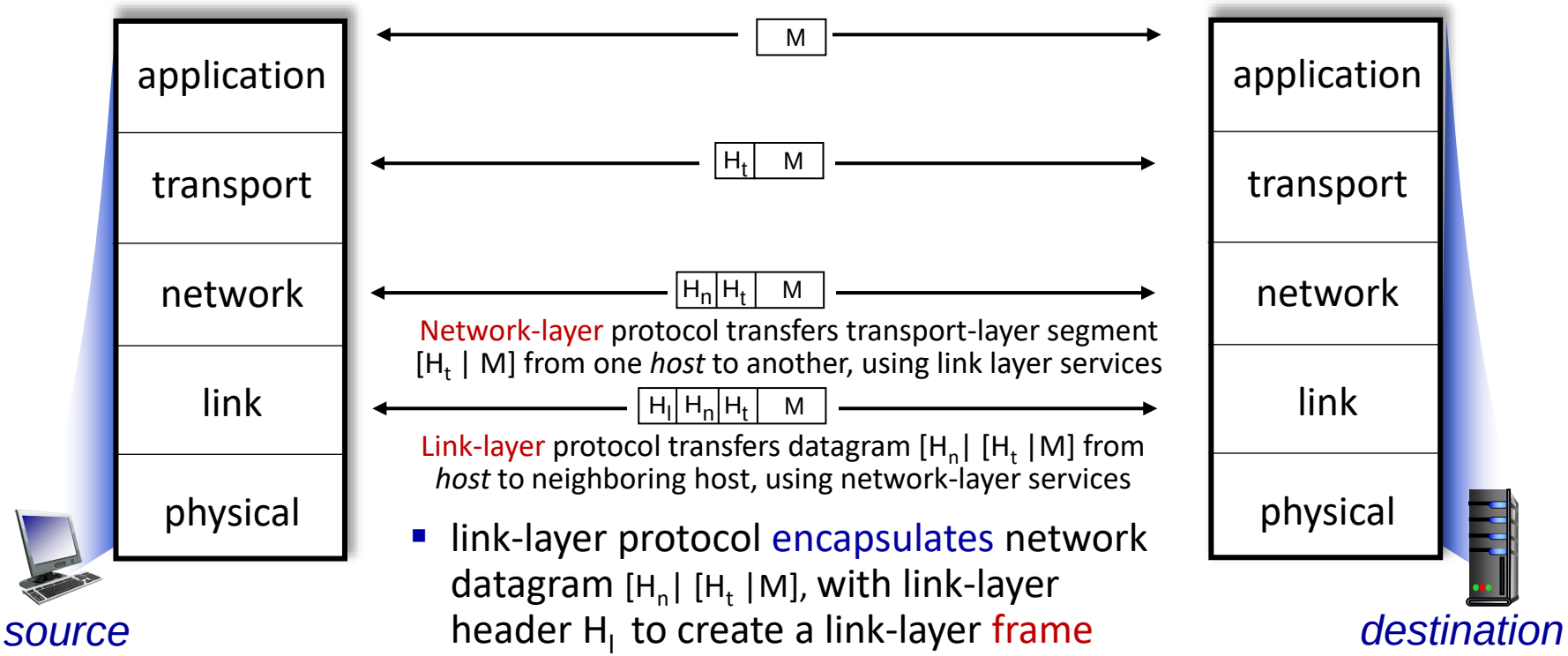
Services, Layering and Encapsulation



- network-layer protocol encapsulates transport-layer segment [H_t | M] with network layer-layer header H_n to create a network-layer datagram
  - H_n used by network layer protocol to implement its service

Internet protocol layering? (6)

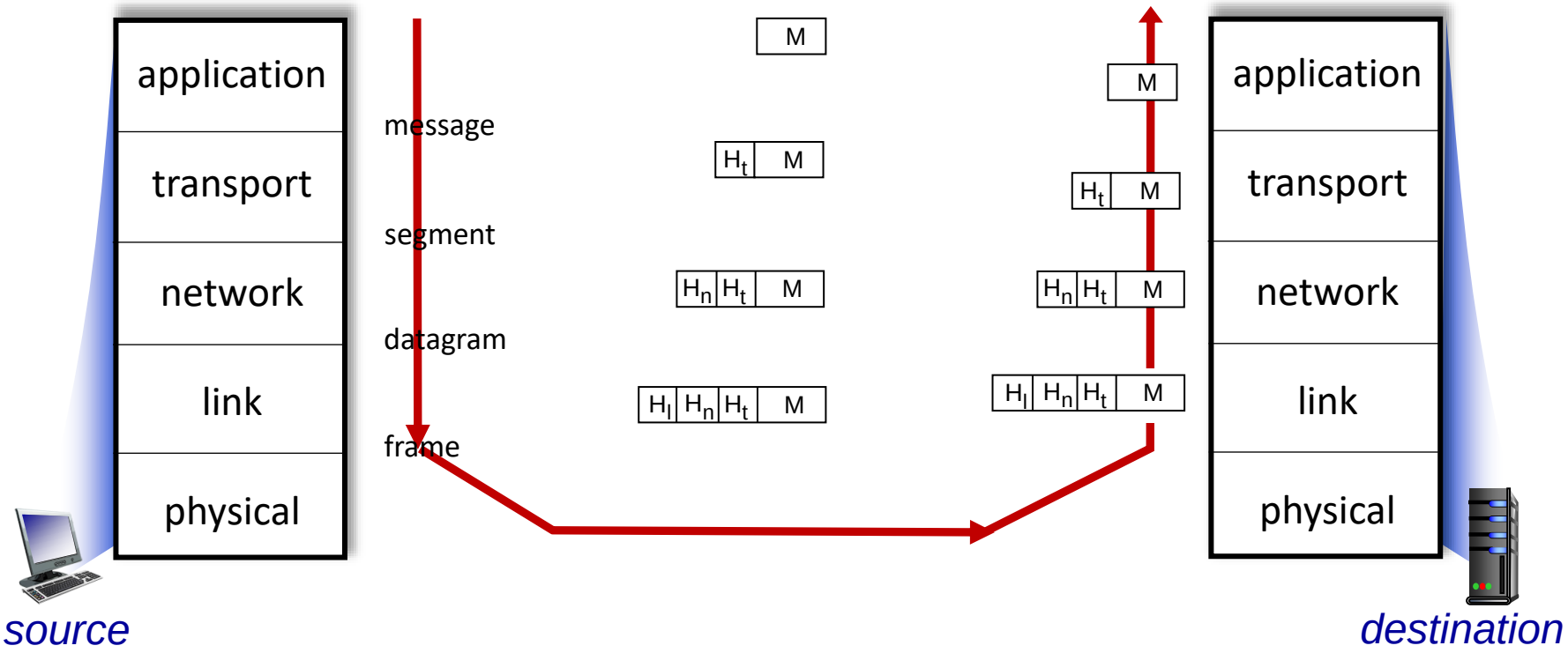
Services, Layering and Encapsulation



- link-layer protocol encapsulates network datagram [H_n| [H_t |M], with link-layer header H_l to create a link-layer frame

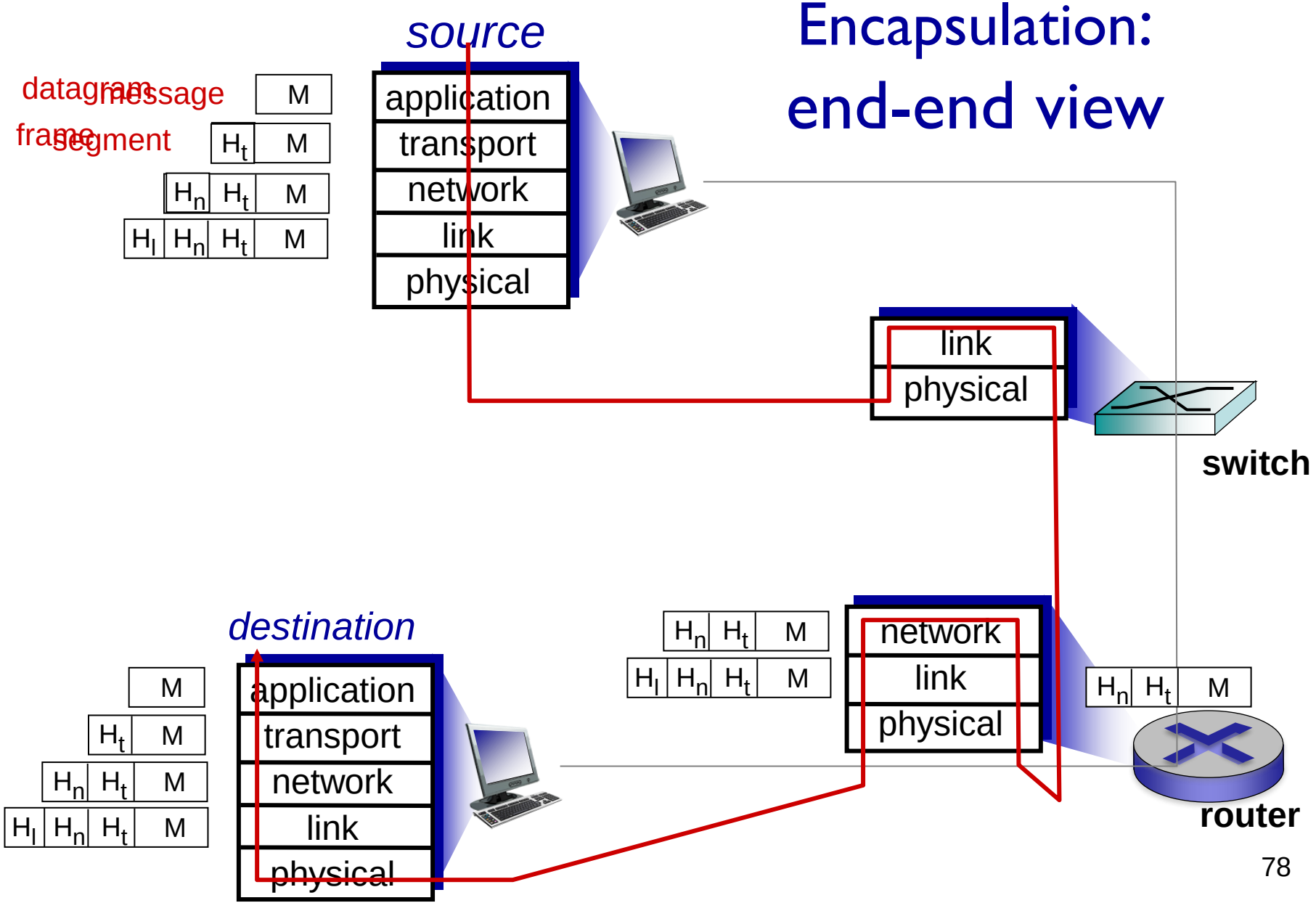
Internet protocol layering? (7)

Services, Layering and Encapsulation



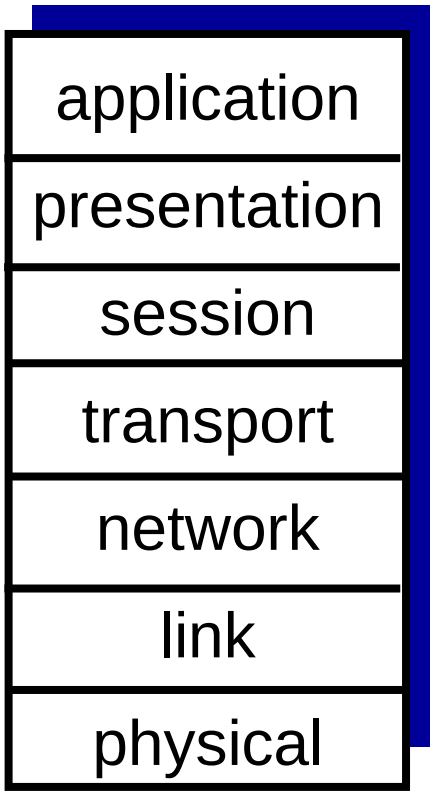
Internet protocol layering? (8)

Encapsulation: end-end view



ISO/OSI reference model

- the international organization for standardization (ISO) proposed that computer networks be organized theoretically around seven layers, called the Open Systems Interconnection (OSI) model



- *presentation:* allow applications to interpret meaning of exchanged data, e.g., encryption, compression, and description (frees the applications from having to worry about the internal format in which data are represented/stored—formats that may differ from one computer to another)

- *session:* delimiting, synchronization, checkpointing, and recovery of data exchange

- Internet protocol stack "missing" these two layers!
  - these services, if needed, must be implemented in by developers in internet applications

Lecture One Outline

- Chapter 1: roadmap

1.1 What is the Internet? What is a Protocol?

1.2 Network Edge: hosts, access network, physical media

1.3 Network Core: packet/circuit switching, internet structure

1.4 Performance: delay, loss and throughput

1.5 Protocol Layers and Service Models

1.6 Security: networks under attack

Network security

- field of network security:
  - how bad guys can attack computer networks
  - how we can defend networks against attacks
  - how to design architectures that are immune to attacks
- Internet not originally designed with (much) security in mind
  - *original vision:* "a group of mutually trusting users attached to a transparent network" ☺
  - Internet protocol designers playing "catch-up" security considerations in all layers!

Bad guys: put malware into hosts via Internet

- malware can get in host from:
  - virus: self-replicating infection requires some form of user interaction to infect the user's device by receiving and executing an infected code (e.g., e-mail attachment)
  - worm: self-replicating infection without any explicit user interaction by passively running a vulnerable network application to which an attacker can send malware
- self-replicating means that once it infects one host, from that host it seeks entry into other hosts over the Internet, and from the newly infected hosts, it seeks entry into yet more hosts
- spyware malware: can record keystrokes, visited web sites, social security numbers, or passwords and upload that info to collection site
- infected host can be enrolled in botnet (network of thousands of similarly compromised hosts) used for distribution of spam emails distribution and denial-of-service (DoS) attacks

Bad guys: attack server, network infrastructure

- *Denial of Service (DoS):* attackers make resources (server, bandwidth) unavailable to legal traffic by overwhelming resources with bogus traffic

- most Internet DoS attacks fall into one of three categories:

1. Vulnerability attack: involves sending a few well-crafted messages to a vulnerable application/operating system running on a targeted host. If the right sequence of packets is sent to a vulnerable application/operating system, the service can stop or, worse, the host can crash.

2. Bandwidth flooding: the attacker sends a flood of packets to the targeted host—so many packets that the target's access link becomes clogged, preventing legal packets from reaching the server.

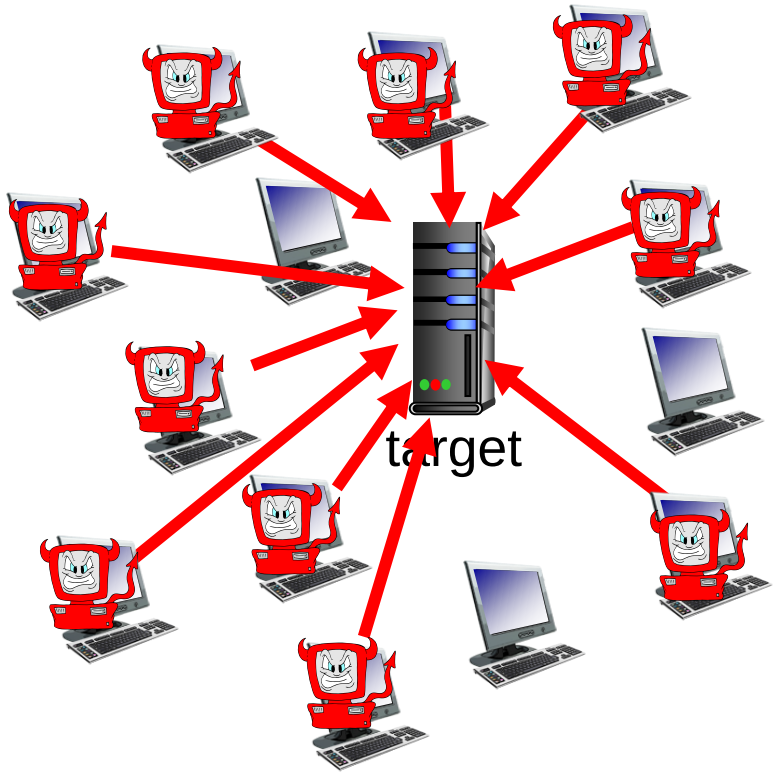
3. *Connection flooding*: the attacker establishes a large number of fully open TCP connections at the target host. The host can become so bogged down with these bogus connections that it stops accepting legal connections.

Bad guys: attack server, network infrastructure-2

- regarding the bandwidth-flooding DoS attack:
  - if the server has an access rate of *R* bps, then the attacker will need to send traffic at a rate of approximately *R* bps to cause damage.
  - if all the traffic emanates from a single source, an upstream router may be able to detect the attack and block all traffic from that source before the traffic gets near the server. Thus a single attack source may not be able to generate enough traffic to harm the server
  - so the attacker controls multiple sources, and has each source blast traffic at the target. With this approach, the aggregate traffic rate across all the controlled sources needs to be approximately *R* to cripple the server
  - this is called distributed DoS (DDoS) attack which is much harder to detect and defend against than a DoS attack from a single host

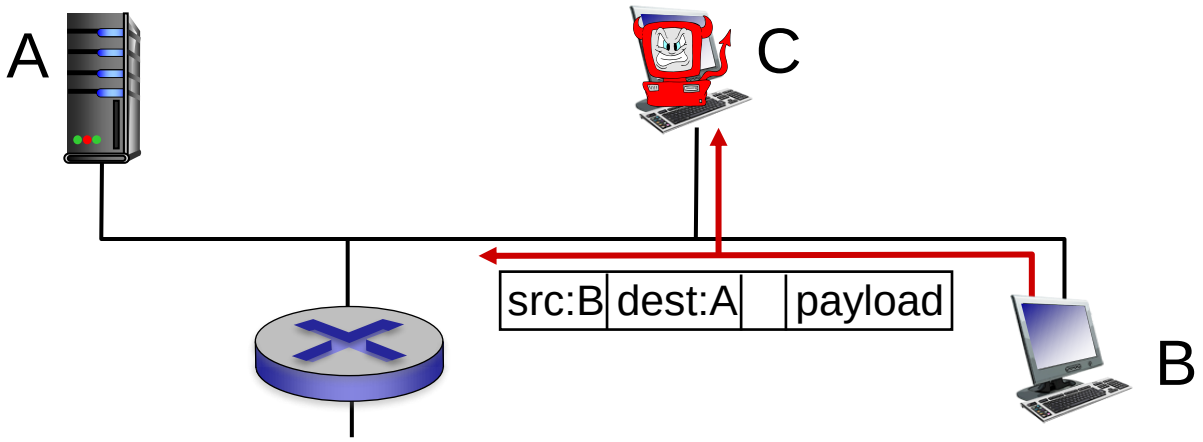
Bad guys: attack server, network infrastructure-3

- DDos attack steps:
  1. select target
  2. break into hosts around the network (botnet)
  3. send packets to target from compromised hosts



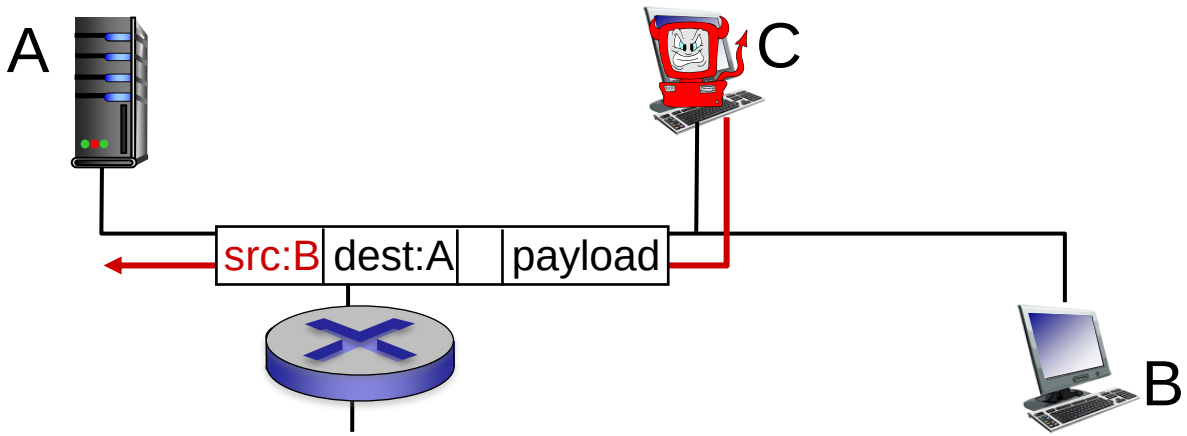
Bad guys can intercept packets

- *packet "sniffing":*
  - occurs in broadcast media (shared Ethernet, wireless)
  - Illegal network interface reads/records all packets (e.g., including sensitive information!) passing by
  - Sniffed packets can then be analyzed offline for sensitive information
  - Because packet sniffers are passive—that is, they do not inject packets into the channel—they are difficult to detect
  - some of the best defenses against packet sniffing involve cryptography



Bad guys can use fake addresses

- *IP spoofing:* send packet with false source address masquerading as truthful users
  - imagine the unsuspecting receiver (say an Internet router) who receives such a packet, takes the (false) source address as being truthful, and then performs some command embedded in the packet's contents (say modifies its forwarding table)
  - to solve this problem, we will need end-point authentication, that is, a mechanism that will allow us to determine with certainty if a message originates from where we think it does.



Lines of defense

- authentication: proving you are who you say you are
  - cellular networks provides hardware identity via SIM card; no such hardware assist in traditional Internet
- confidentiality: via encryption
- integrity checks: digital signatures prevent/detect tampering
- access restrictions: password-protected VPNs
- firewalls: specialized "middleboxes" in access and core networks:
  - off-by-default: filter incoming packets to restrict senders, receivers, applications
  - detecting/reacting to DOS attacks

… lots more on security (throughout, Chapter 8)

The End