CS 4390: Computer Networks

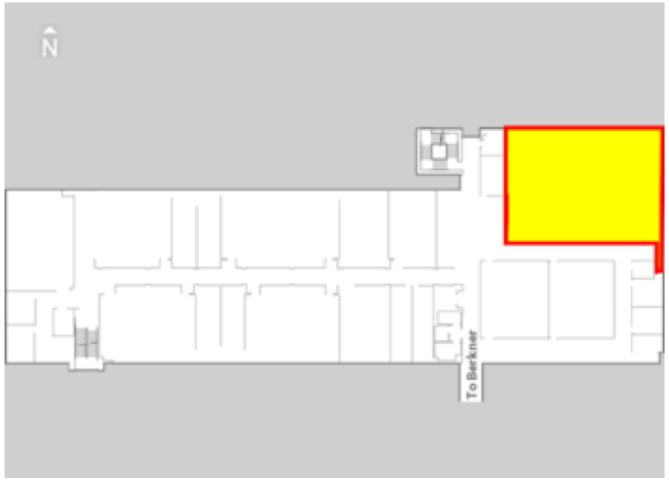
Shuang Hao
University of Texas at Dallas
Fall 2023

CS 4390 Info

- Class time and location
  - 5:30-6:45 pm Monday/Wednesday, SLC 2.303

- Office hour
  - TBA
  - Email: shao@utdallas.edu

- TA: TBA



What You Will Learn in This Class?

- The general goal is to gain a solid knowledge about computer networks

- Goals and skills
  - Understand TCP/IP network models
  - Understand Internet services, such as HTTP and DNS
  - Understand and design routing protocols
  - Carry out network programming

Course Style

- The course is taught in lectures

- Contents and information will be distributed via eLearning
  - Slides
  - Assignments (including submission and grading)
  - Announcements (such as TAs, exam rules)

Course Style

- Textbook
  - "Computer Networking – a Top-Down Approach, 6th Edition", Jim F. Kurose and Keith W. Ross

- Requirements
  - Have taken Data Structure class, and C programming

Grading Policy

- 5% Class participation
  - Checking attendance
- 25% Homework assignments
  - 3-4 assignments, written solutions and programming
- 30% Midterm exam
  - One-page cheat sheet, topics taught till midterm
- 40% Final exam
  - Two-page cheat sheet, all topics

Final Grade

- Final grades based on the percentile
  - 10% A+
  - 25% A
  - 35% A-
  - 50% B+
  - 70% B
  - Rest B-

    C+, lower than (mean – 2 x std), bounded by Chebyshev's inequality

    C, lower than (mean – 3 x std)

    and so on….
    - Missing exam/assignment, deduct half grade (e.g., A- → B+)

Grading Policy

- Late policy for assignment
  - 1 point deduction every 24 hours
- Missing attendance checking
  - 1 point deduction each time
- No one is exempt from the exams
  - Makeup exam only with a legitimate reason in advance (and evidence provided)

Grading Policy

- Exams taken at testing center
  - Need to register exams in advance
  - Please make sure available during the exam time NOW
- Midterm date
  - 60 mins, time window 10/4/2023 Wed 4:30pm-8pm
- Final exam date
  - 75 mins, time window 12/11/2023 Mon 4:30pm-8:30pm

Classroom Policy

- Masks are encouraged to wear in the classroom
- No food is allowed in the classroom
- No laptop is allowed to use in the class
- Cell phones shall be turned mute during the class and shall not be used in the class
- No talking is allowed during the class, and raising hands for questions.

Other Policy

- Cheating policy
  - Follow the university policy on cheating and plagiarism
- Ethics
  - No disruption of other networks or machines

What's the Internet

IP picture frame
http://www.ceiva.com/

Web-enabled toaster +
weather forecaster

Tweet-a-watt:
monitor energy use

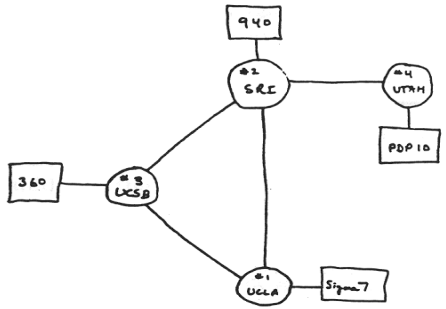
Internet
refrigerator

Slingbox: watch,
control cable TV remotely

Internet phones

Internet History

- 1961-1972: Early packet-switching principles
  - 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
  - 1964: Baran - packet-switching in military nets
  - 1967: ARPANET conceived by Advanced Research Projects Agency
  - 1969: first ARPANET node operational
  - 1972:
    - ARPANET public demo
    - NCP (Network Control Protocol) first host-host protocol
    - first e-mail program
    - ARPANET has 15 nodes



THE ARPA NETWORK

Internet History

- ARPANET node at UCLA
  - Photos from 2011

Internet History

- 1972-1980: Internetworking, new and proprietary nets

- 1970: ALOHAnet satellite network in Hawaii
- 1974: Cerf and Kahn - architecture for interconnecting networks
- 1976: Ethernet at Xerox PARC
- late70's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPANET has 200 nodes

Cerf and Kahn's internetworking principles:

- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today's Internet architecture

Internet History

- 1980-1990: new protocols, a proliferation of networks

- 1983: deployment of TCP/IP
- 1982: SMTP e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: FTP protocol defined
- 1988: TCP congestion control

- new national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

Internet History

- 1990-2000: commercialization, the Web, new apps

- early 1990's: ARPANET decommissioned
- 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- early 1990s: Web
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990's: commercialization of the Web

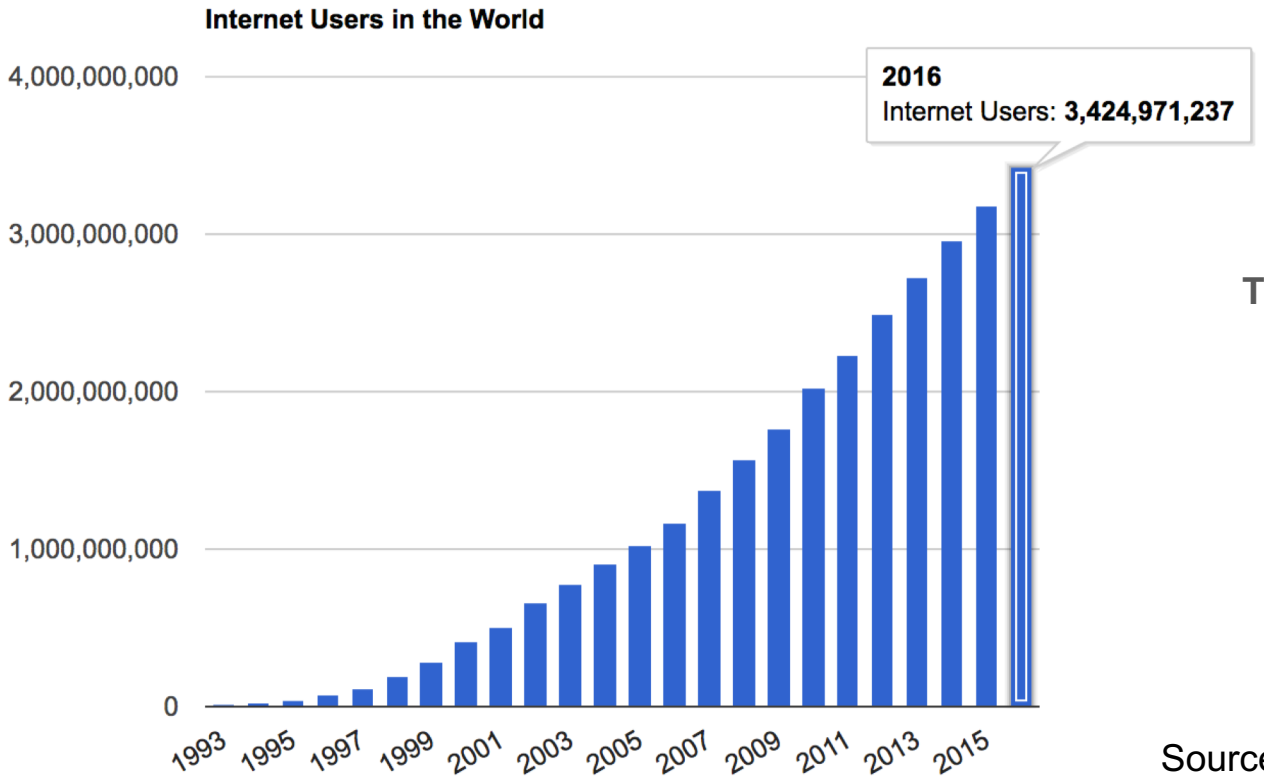
late 1990's – 2000's:

- More apps: instant messaging, P2P file sharing
- network security to forefront
- est. 50 million host, 100 million+ users
- backbone links running at Gbps

Internet History

- 2005-present
  - Smartphones and tablets
  - Aggressive deployment of broadband access
  - Increasing ubiquity of high-speed wireless access
  - Emergence of online social networks:
    - Facebook: soon one billion users
  - Service providers (Google, Microsoft) create their own networks
    - Bypass Internet, providing “instantaneous” access to search, emai, etc.
  - E-commerce, universities, enterprises running their services in “cloud” (eg, Amazon EC2)

Trend of Internet Users and Traffic



1,116,091,120

Total number of websites

179,719,165,683

Emails sent today

Source: Internet Live Stats

Recap Where We're At

- Class Overview and Policies
- Internet History

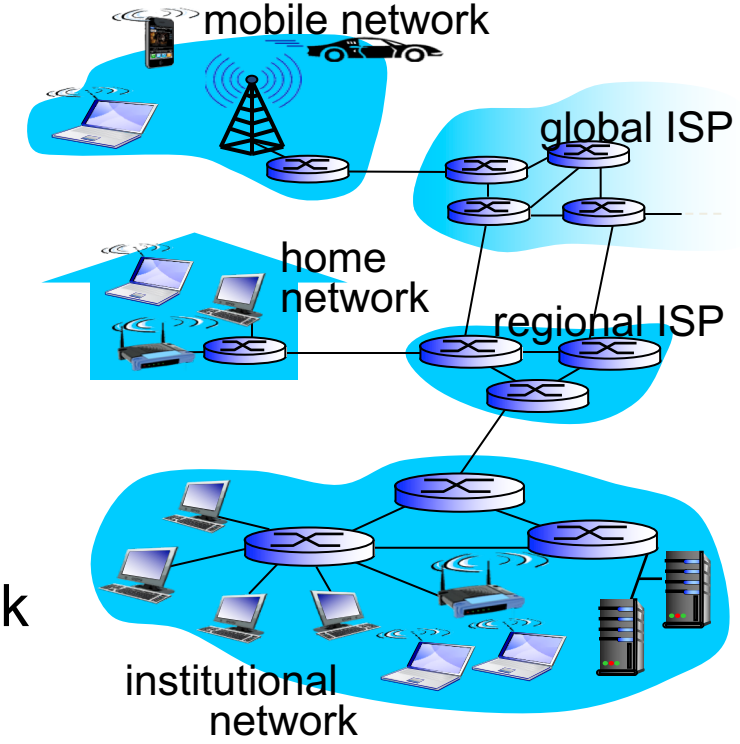
Outline

- Structure of the Internet

- Networks under Attack: Security

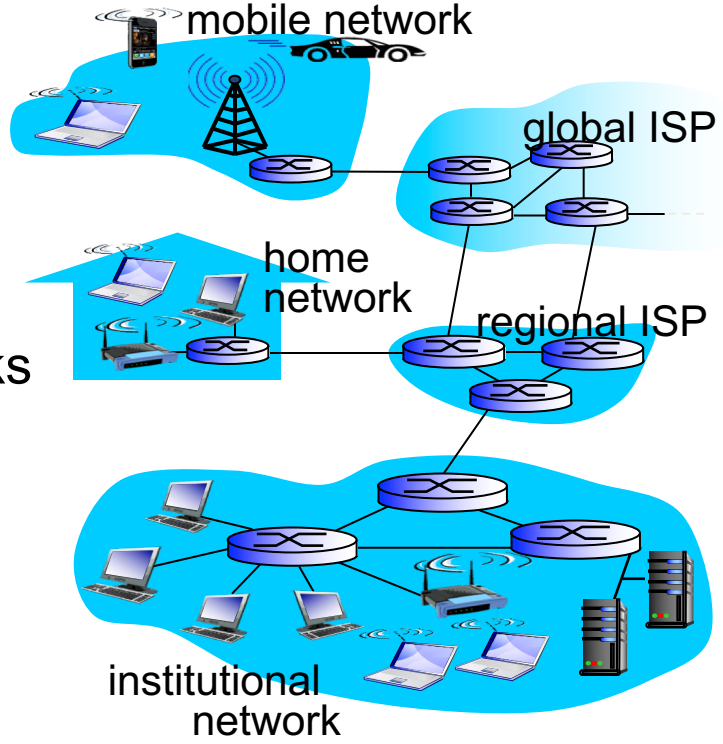
What's the Internet

- Internet: "network of networks"
- Protocols control sending, receiving of messages
  - e.g., TCP, IP, HTTP, Skype
- Internet standards
  - RFC: Request for comments
    - https://tools.ietf.org/html/
  - IETF: Internet Engineering Task Force



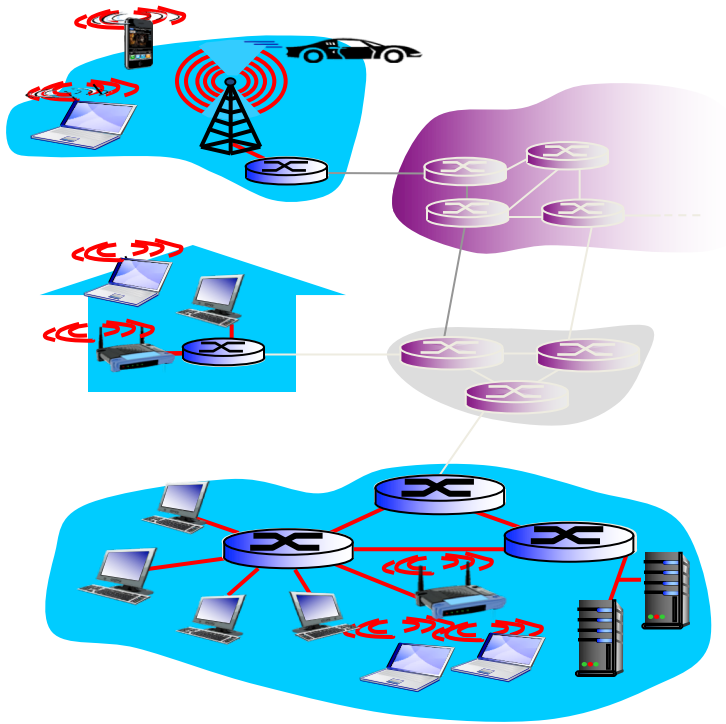
A Closer Look at Network Structure

- Network edge
  - hosts: clients and servers
  - servers often in data centers
- Access networks, physical media
  - wired, wireless communication links
- Network core
  - interconnected routers
  - Internet Service Providers (ISPs)



Access Networks and Physical Media

- How to connect end systems to edge router?
  - residential access nets
  - institutional access networks (school, company)
  - mobile access networks

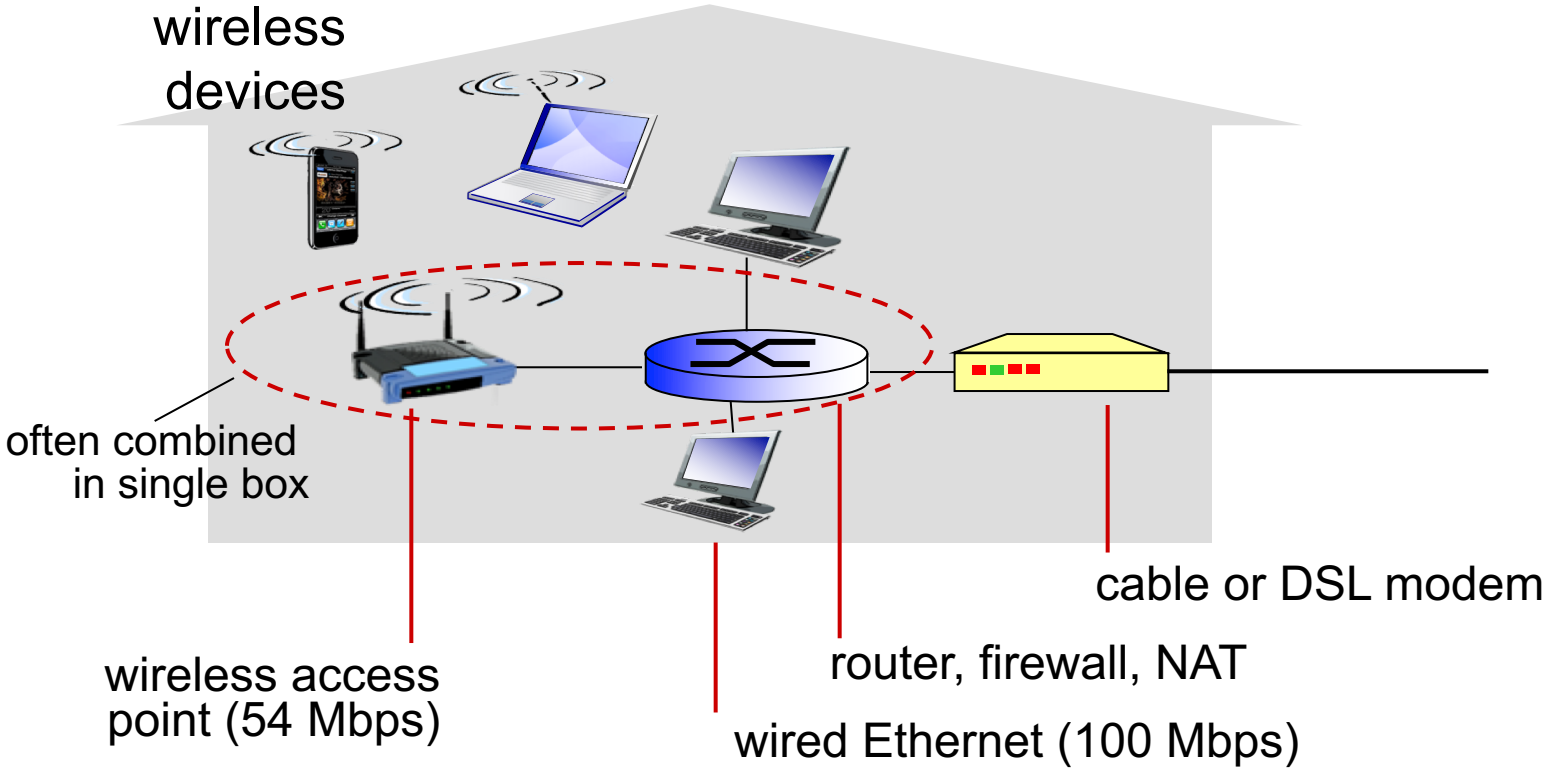


Access Networks

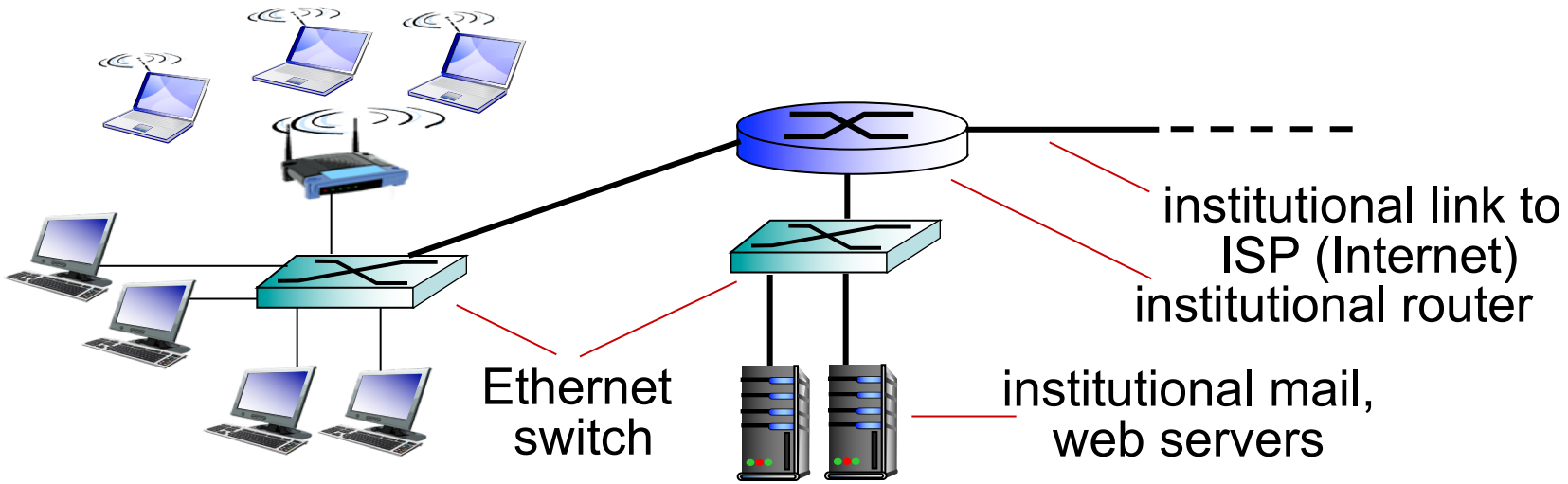
- Digital Subscriber Line (DSL)
  - Use existing telephone line
  - Data over DSL phone line goes to Internet
  - < 24 Mbps downstream transmission rate

- Cable
  - Hybrid Fiber Coax (HFC)
  - Need cable modem termination system
  - High downstream transmission rate

Home Access Networks



Enterprise Access Networks (Ethernet)



- Typically used in companies, universities, etc
- 10 Mbps, 100Mbps, 1Gbps, 10Gbps transmission rates

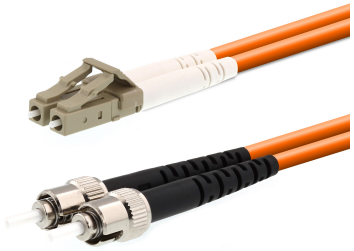
Physical Media

- Twisted pair (TP)
  - Two insulated copper wires
  - Support 100 Mbps to 1 Gbps Ethernet
  - Connected to machines/desktops

- Coaxial cable
  - two concentric copper conductors
  - Broadband, HFC

Physical Media

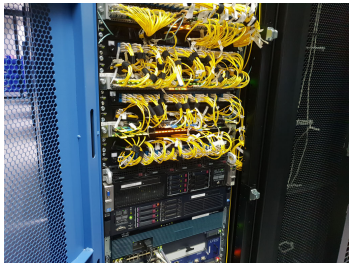
- Fiber optic cable
  - glass fiber carrying light pulses, each pulse a bit
  - high-speed operation:
    - high-speed point-to-point transmission (e.g., 10's-100's Gbps transmission rate)
  - low error rate:
    - repeaters spaced far apart
    - immune to electromagnetic noise



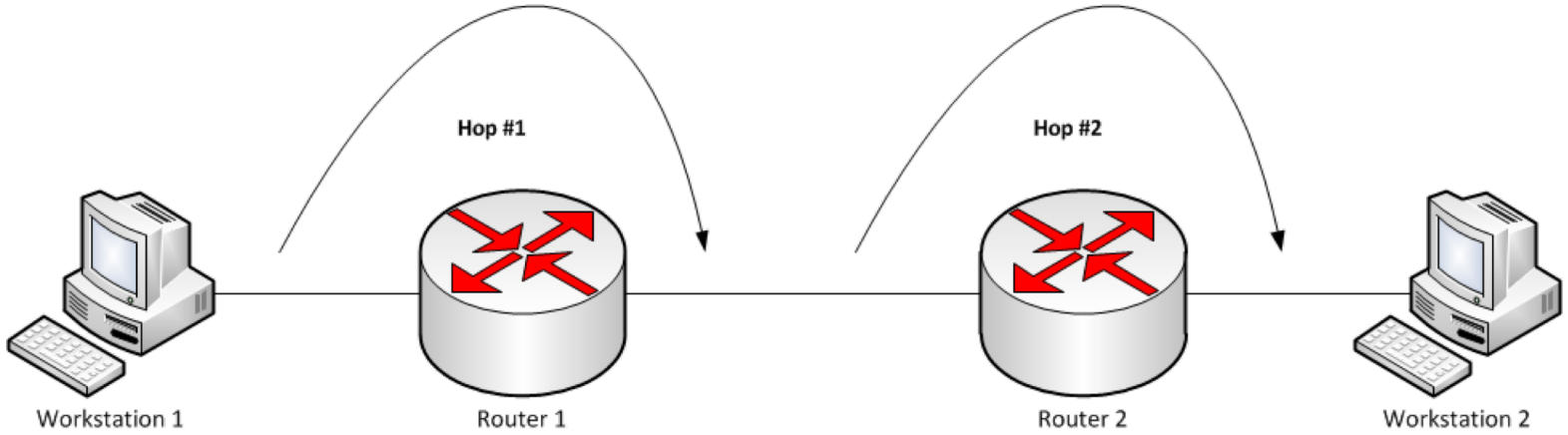
Network Devices

- Hub
  - All traffic is broadcasted to all ports

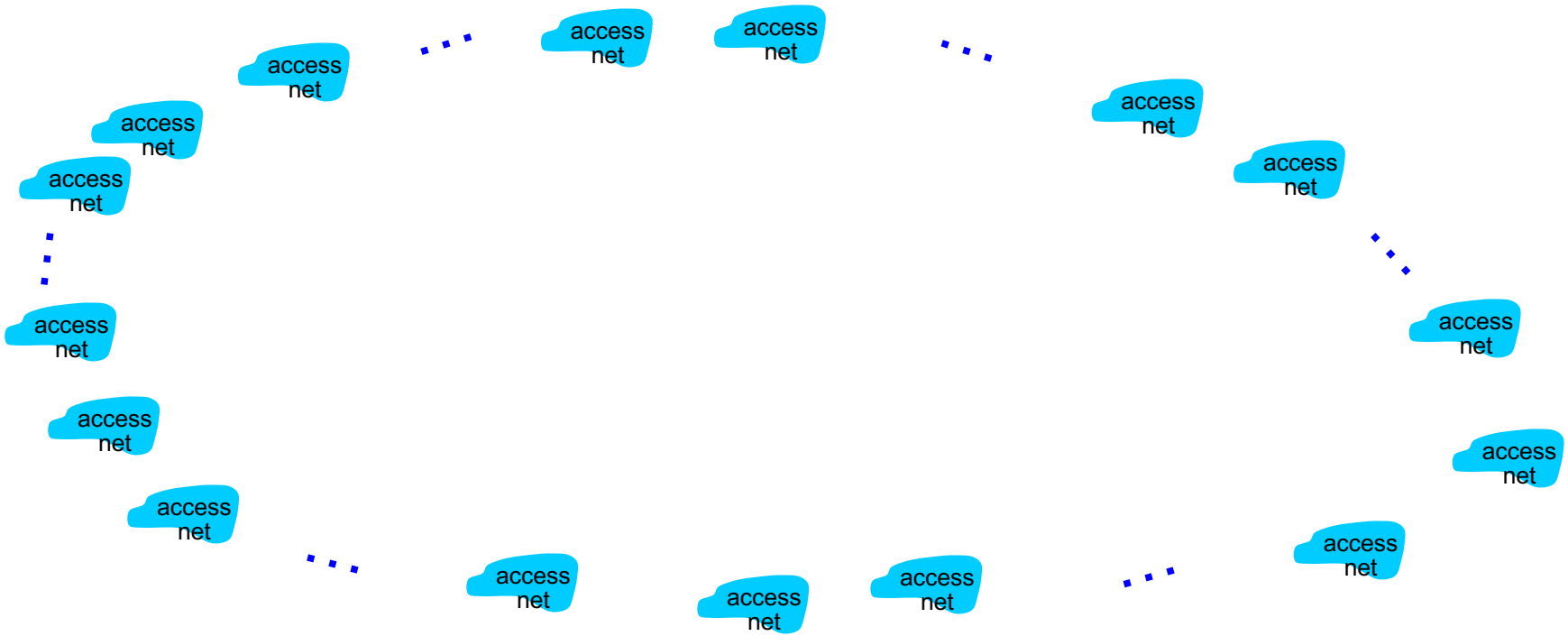
- Switch
  - All directed traffic is sent to the ports associated with the referenced hardware address
- Router
  - Forward traffic towards its destination



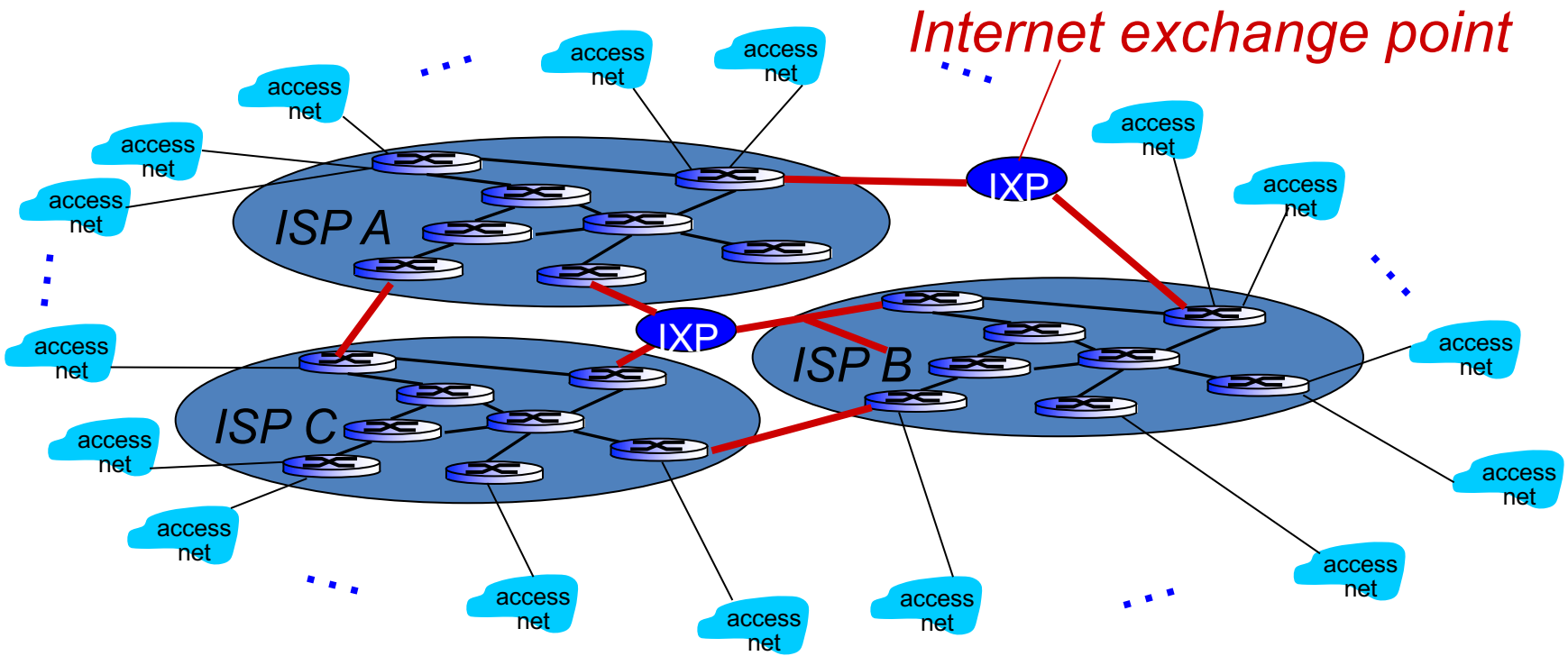
Internet Structure



Internet Structure



Internet Structure



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”

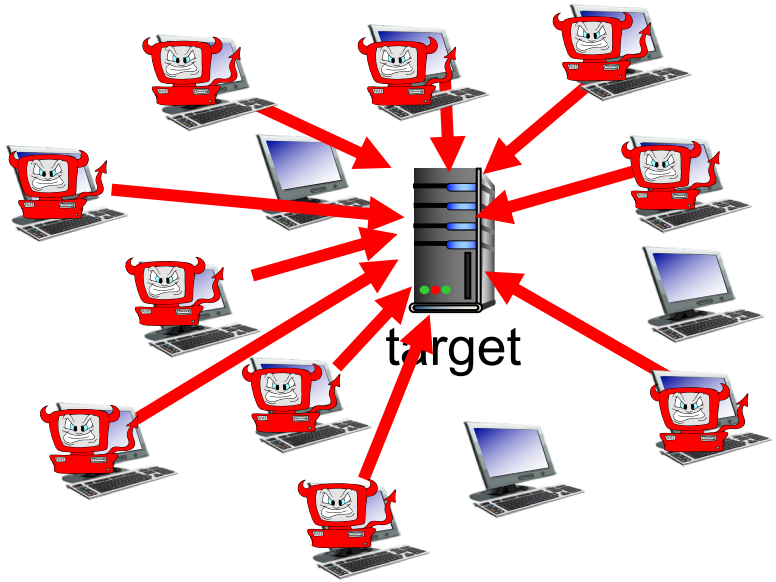
Bad Guys: Put Malware into Hosts

- Malware can get in host from:
  - virus: self-replicating infection by receiving/executing object (e.g., e-mail attachment)
  - worm: self-replicating infection by passively receiving object that gets itself executed
- spyware malware can record keystrokes, web sites visited, upload info to collection site
- Infected host can be enrolled in botnet, used for spam. Denial-of-Service attacks

Bad Guys: Attack Server, Network Infrastructure

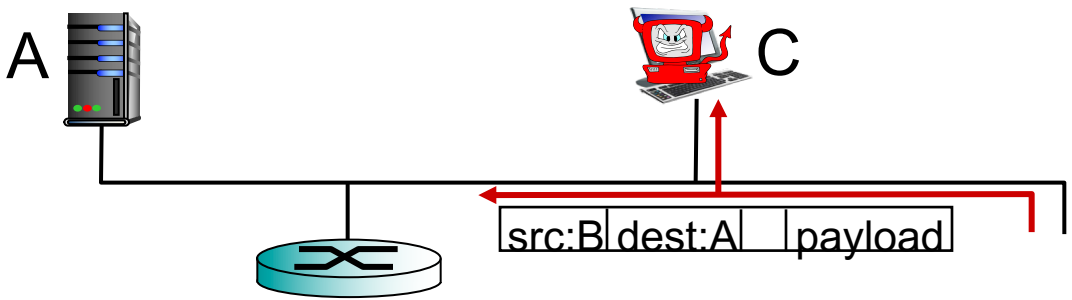
- Denial of Service (DoS)
  - Attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic

1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



Bad Guys can Sniff Traffic

- Traffic “sniffing”
  - broadcast media (shared ethernet, wireless)
  - promiscuous network interface reads/records all traffic (e.g., including passwords!) passing by



Recap Where We're At

- Structure of the Internet

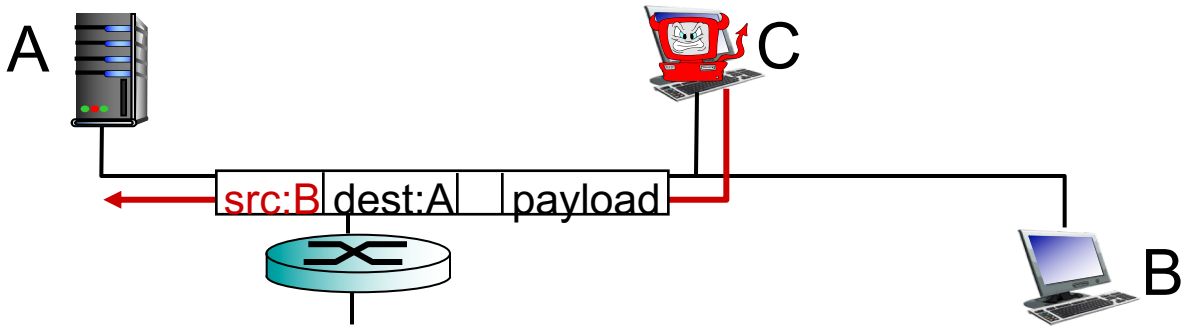
- Networks under Attack: Security

Outline

- Networks under Attack: Security (Continue)
- Packet Switching
- Delay, Loss, Throughput in Networks

Bad Guys Can Use Fake Addresses

- Address spoofing
  - Send packet with false source address



Security Concepts

- Communication security
  - Confidentiality: preventing adversaries from reading our private data (message or document)
  - Integrity: preventing attackers from altering the data (Data itself might or might not be private)
  - Authentication: determining who created a given message or document
  - Non repudiation: associating actions or changes to a unique individual

Infamous Incidents

- Morris worm (1988): Robert Morris, Jr, Cornell U. grad student "accidentally" released the 1988 worm crashing thousands of Internet-connected computers
  - Robert was sentenced to three years' probation, a $10,000 fine, and 400 hours of community service

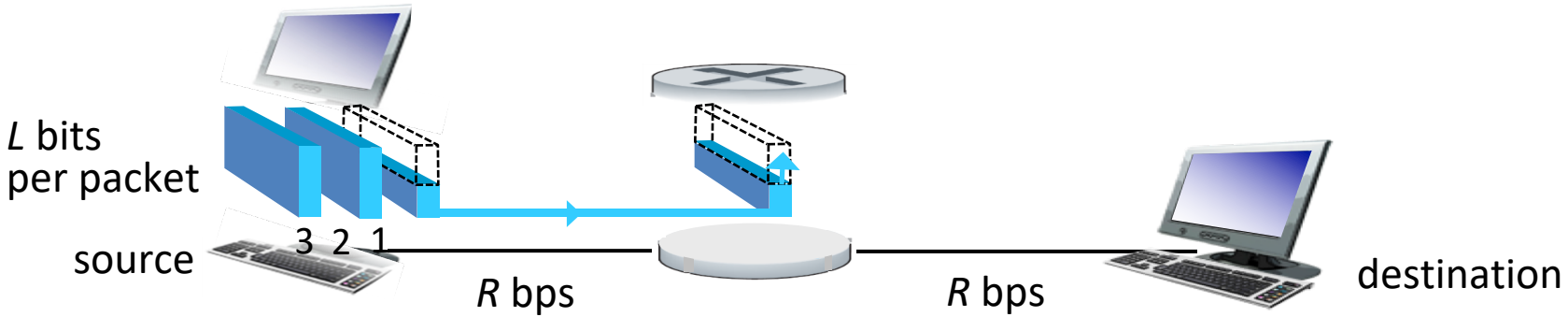
- Albert Gonzales' case: stole millions of credit card numbers from hacking into TJX companies (T.J. Maxx, and other retail companies)
  - On March 25, 2010 he was sentenced to 20 years in federal prison

Packet Switching v.s. Circuit Switching

- Packet switching
  - Break messages into packets, allocate transmission resources as needed
  - Used in computer networks

- Circuit switching
  - End-end resources allocated to, reserved for “call” between source & destination
  - Used in telephone networks

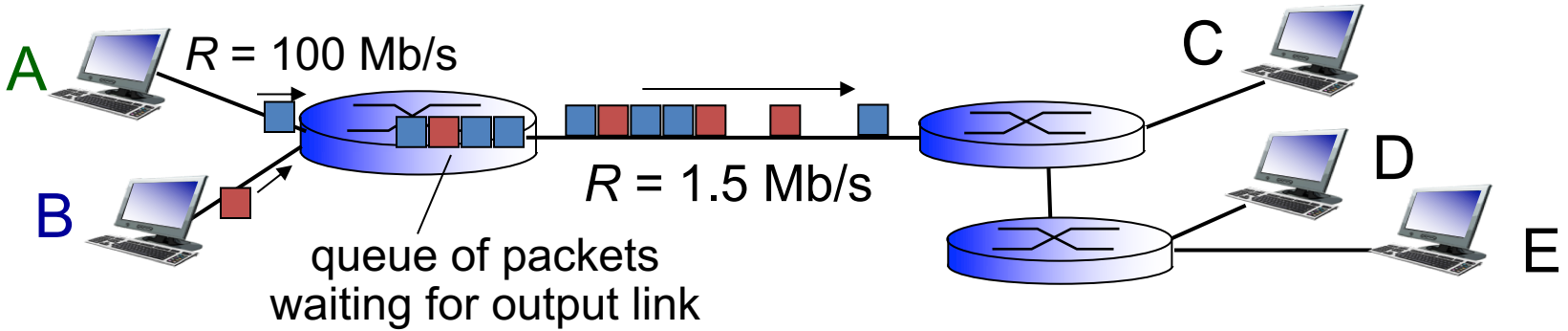
Packet Switching: Store-and-Forward



- Take 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
- End-end delay
  - 2L/R (assuming zero propagation delay)

- one-hop numerical example:
  - L = 7.5 Mbits
  - R = 1.5 Mbps
  - one-hop transmission delay = 5 sec

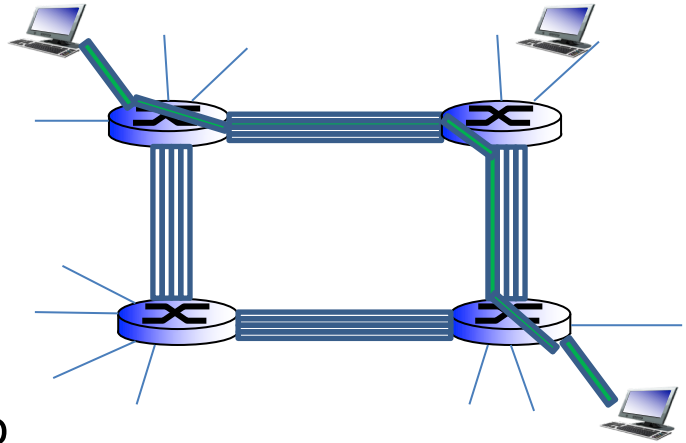
Packet Switching: Queueing Delay, Loss



- If arrival rate (in bits) to link exceeds transmission rate 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

Alternative: Circuit Switching

- End-end resources allocated to, reserved for “call” between source & destination
- In the example 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



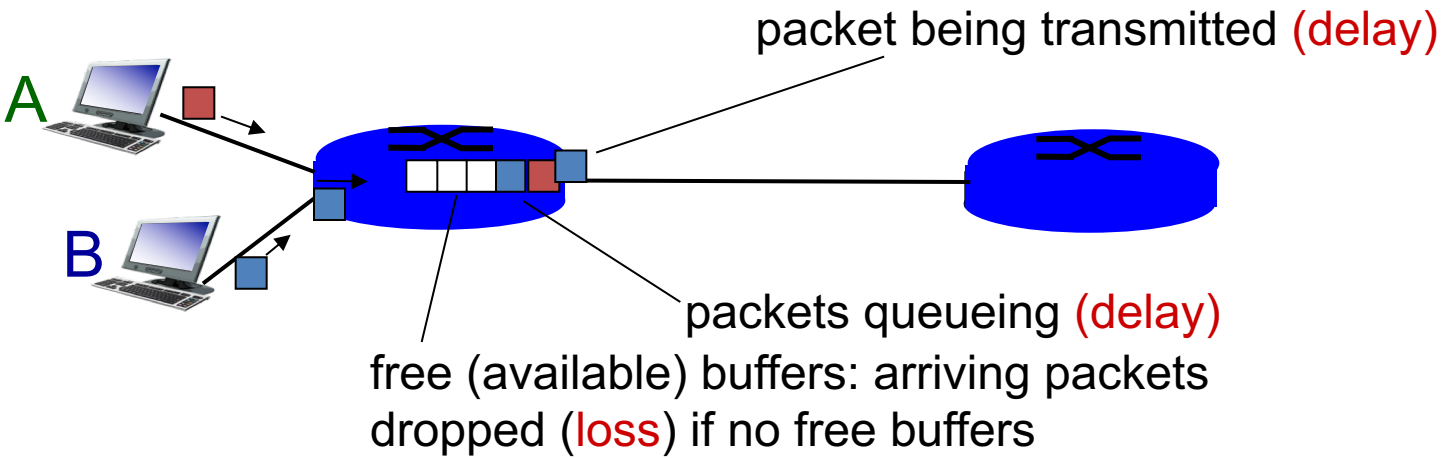
Packet Switching v.s. Circuit Switching

- Pros of packet switching
  - Great for bursty data
  - Resource sharing
  - Simpler, no call setup

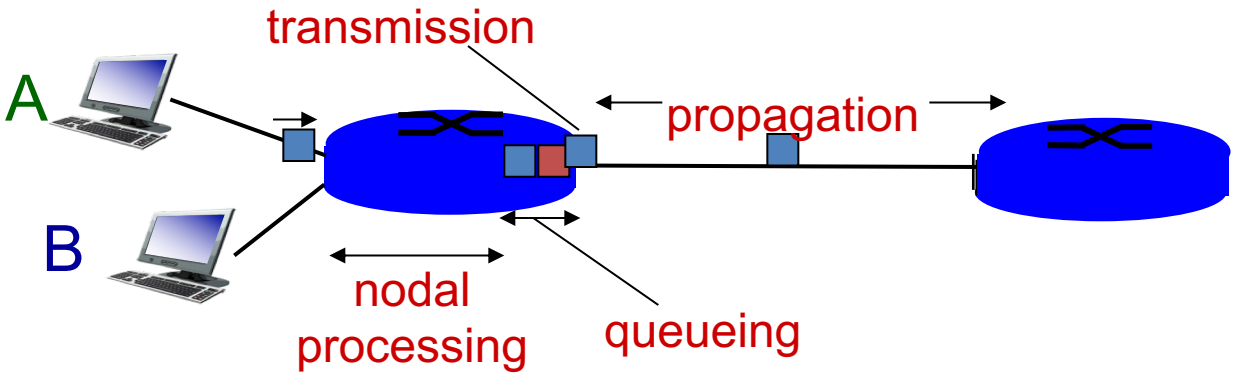
- Cons of packet switching
  - excessive congestion possible: packet delay and loss
  - protocols needed for reliable data transfer, congestion control

How Do Loss and Delay Occur?

- Packets queue in router buffers
  - packet arrival rate to link (temporarily) exceeds output link capacity
  - packets queue, wait for turn



Four Sources of Packet Delay



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

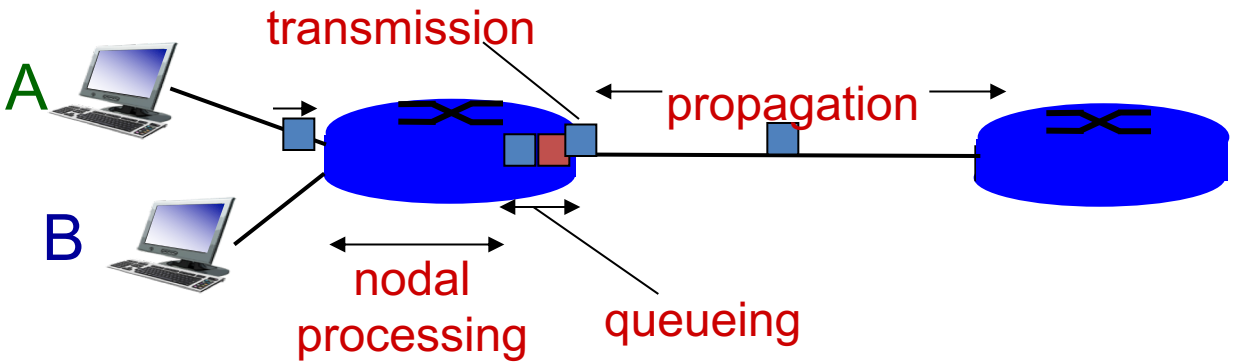
d_{proc}: nodal processing

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

d_{queue}: queueing delay

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

Four Sources of Packet Delay



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

d_{trans}: transmission delay:

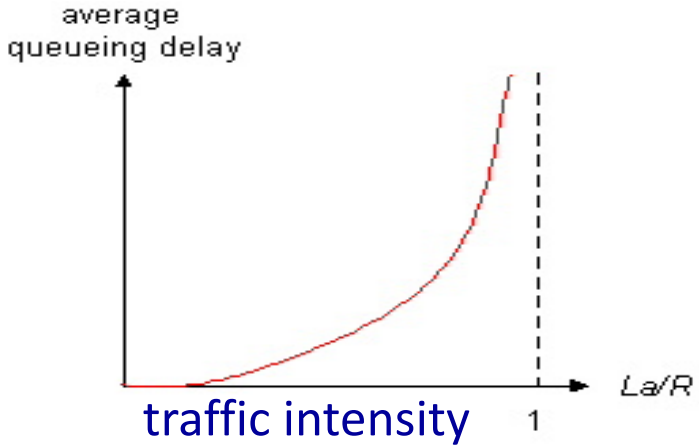
- L: packet length (bits)
- R: link *bandwidth (bps)*
- $d_{trans} = L/R$

d_{prop}: propagation delay:

- d: length of physical link
- s: propagation speed in medium ($\sim 2 \times 10^8$ m/sec)
- $d_{prop} = d/s$

Queuing Delay (Revisit)

- *R*: link bandwidth (bps)
- *L*: packet length (bits)
- a: average packet arrival rate



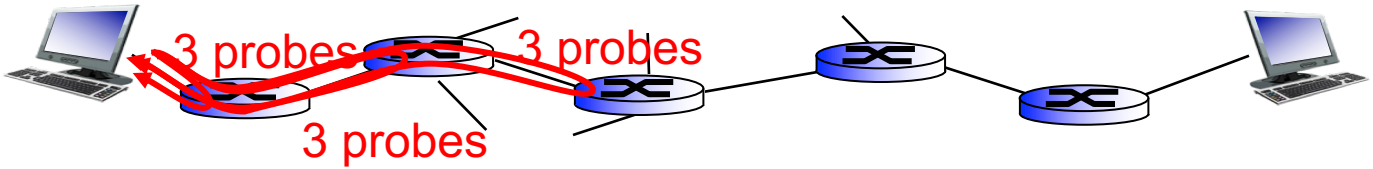
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

“Real” Internet Delays and Routes

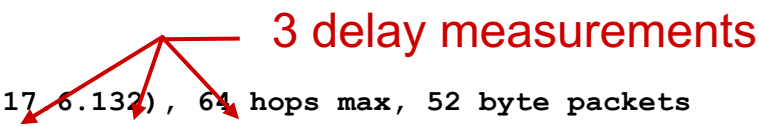
- What do “real” Internet delay & loss look like?
- traceroute program: provides delay measurement from source to router along end-end Internet path towards destination. For all *i*:
  - sends three packets that will reach router *i* on path towards destination
  - router *i* will return packets to sender
  - sender times interval between transmission and reply.



“Real” Internet Delays, Routes

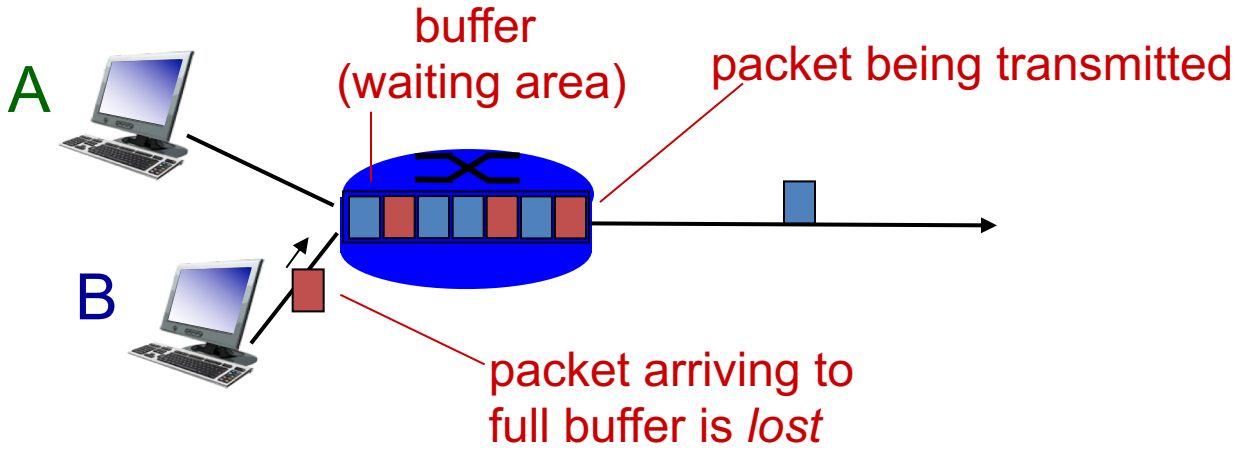
```
$ traceroute www.google.com

traceroute to www.google.com (172.217.6.132), 64 hops max, 52 byte packets
 1  cometnet-gw (10.21.0.1)  2.263 ms  2.043 ms  2.169 ms
 2  corefo-m2-po3 (129.110.83.81)  1.441 ms
    corephy-m2-po6 (129.110.83.156)  1.633 ms  1.494 ms
 3  utd9-v40 (129.110.5.75)  2.757 ms  2.305 ms  2.761 ms
 4  74.200.189.108 (74.200.189.108)  3.413 ms  3.660 ms  3.526 ms
 5  74.200.189.251 (74.200.189.251)  4.010 ms  3.896 ms  4.819 ms
 6  72.14.215.83 (72.14.215.83)  6.638 ms  3.896 ms  3.718 ms
 7  * * *
 8  72.14.232.167 (72.14.232.167)  3.837 ms  3.722 ms
    72.14.234.61 (72.14.234.61)  3.985 ms
 9  dfw25s16-in-f4.1e100.net (172.217.6.132)  3.754 ms  4.383 ms  3.698 ms
```



Packet loss

- Queue (aka buffer) preceding link in buffer has finite capacity
- Packet arriving to full queue dropped (aka lost)
- Lost packet may be retransmitted by previous node, or by source end system



Packet loss

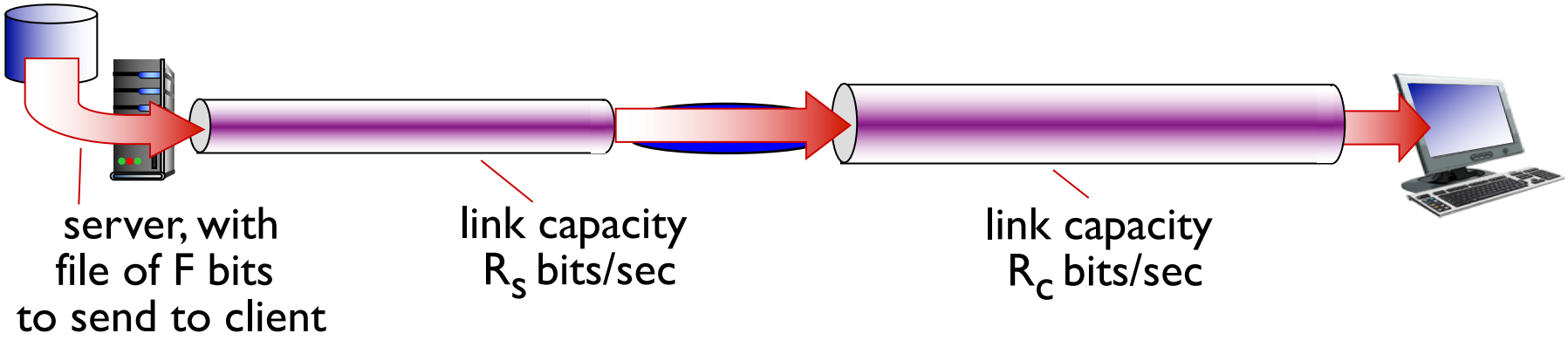
```
$ ping -c 5 www.google.com

PING www.google.com (172.217.4.164) 56(84) bytes of data.
64 bytes from lax28s01-in-f164.1e100.net (172.217.4.164): icmp_seq=1 ttl=55 time=3.49 ms
64 bytes from lax28s01-in-f164.1e100.net (172.217.4.164): icmp_seq=2 ttl=55 time=3.21 ms
64 bytes from lax28s01-in-f164.1e100.net (172.217.4.164): icmp_seq=3 ttl=55 time=3.64 ms
64 bytes from lax28s01-in-f164.1e100.net (172.217.4.164): icmp_seq=4 ttl=55 time=3.37 ms
64 bytes from lax28s01-in-f164.1e100.net (172.217.4.164): icmp_seq=5 ttl=55 time=3.22 ms

--- www.google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 3.217/3.391/3.641/0.177 ms
```

Throughput

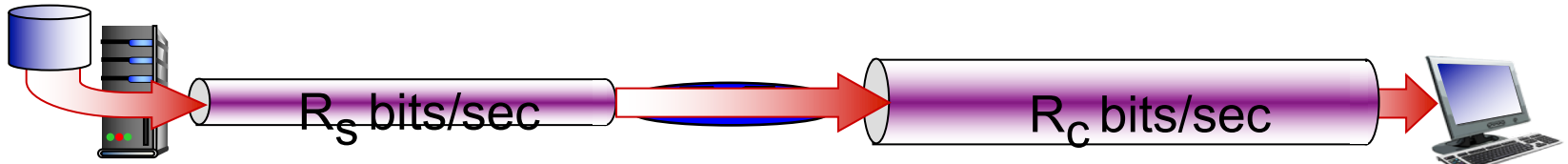
- Throughput: rate (bits/time unit) at which bits transferred between sender/receiver
  - Instantaneous: rate at given point in time
  - Average: rate over longer period of time



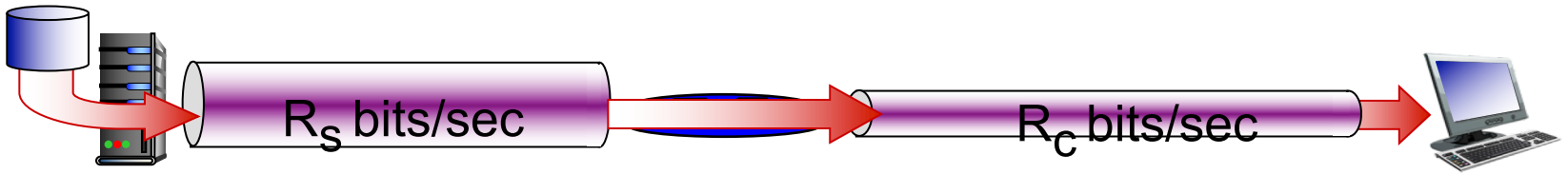
Throughput

- What is average end-end throughput?

$R_s < R_c$



$R_s > R_c$



- Bottleneck link
  - Link on end-end path that constrains end-end throughput

Recap Where We're At

- Networks under Attack: Security
- Packet Switching
- Delay, Loss, Throughput in Networks

Outline

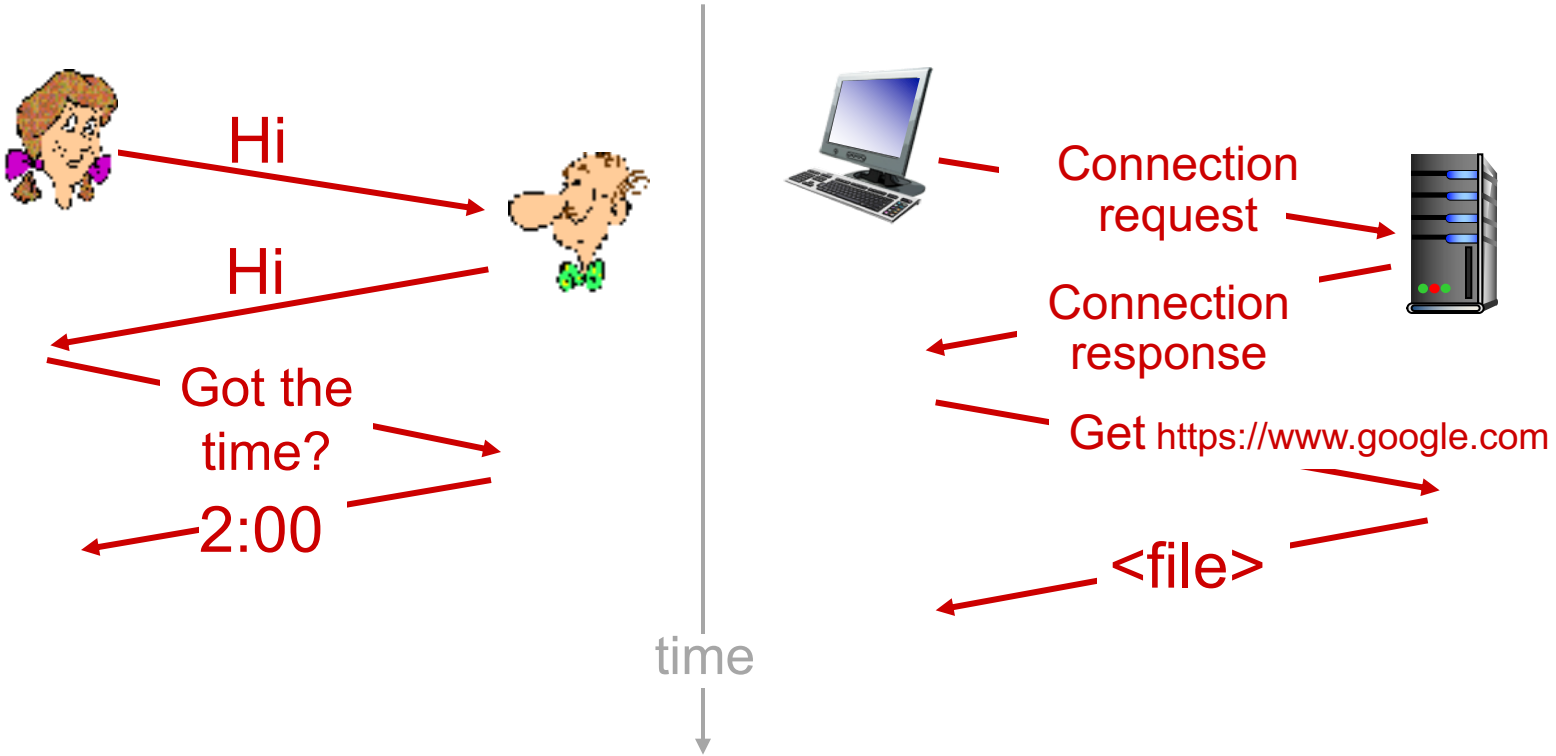
- Protocol Layers

What is a Protocol?

- Human protocols:
  - "what's the time?"
  - "I have a question"
  - introductions
- Network protocols:
  - Machines rather than humans
  - All communication activity in Internet governed by protocols
- Protocols define format, order of msgs sent and received among network entities, and actions taken on msg transmission, receipt

What is a Protocol?

- A human protocol and a computer network protocol



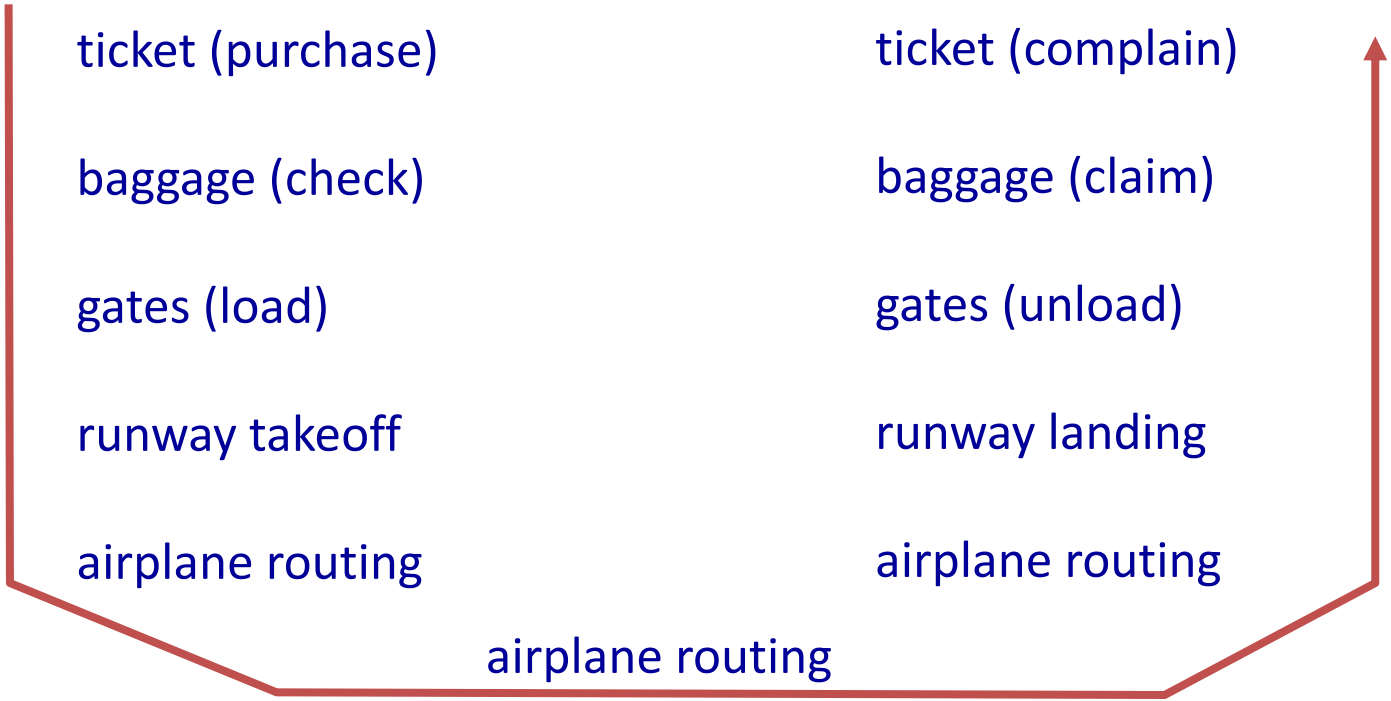
Protocol Layers

- Networks are complex, with many “pieces”
  - hosts
  - routers
  - links of various media
  - applications
  - protocols
  - hardware, software

- Question
  - Is there any hope of organizing and discussing functionalities and structure of networks?
- Answer
  - The key is layering!
  - It uses the power of abstraction

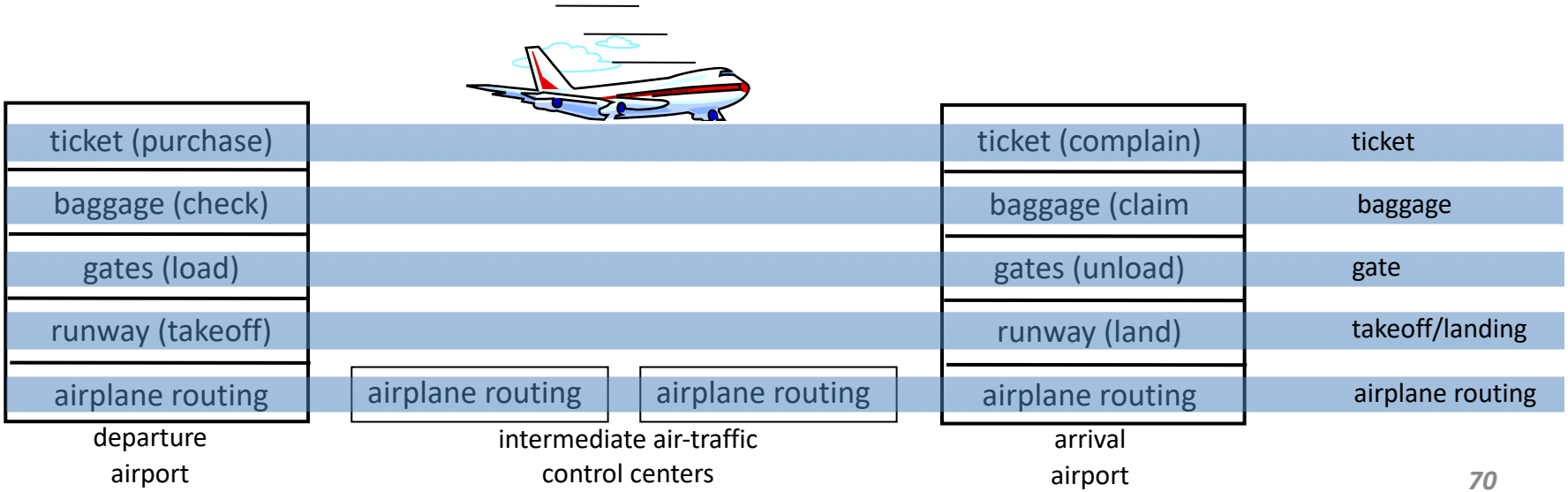
Analogy: Organization of Air Travel

- A series of steps



Layering of Airline Functionality

- Layers: each layer implements a service
  - via its own internal-layer actions
  - relying on services provided by layer below

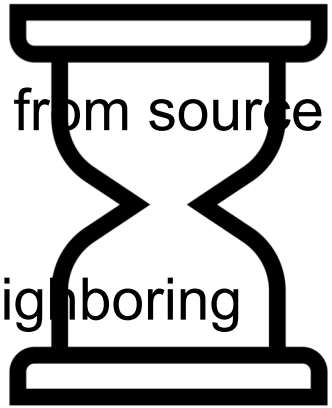


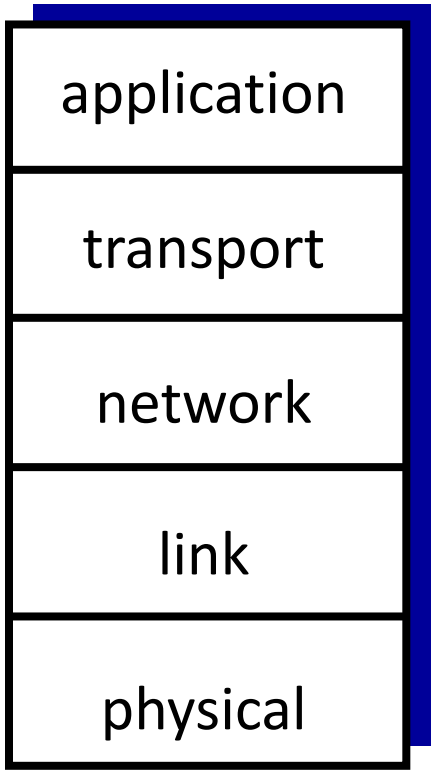
Layering and Power of Abstraction

- Divide the (complex) functionality of a system into manageable chunks (layers) – divide-and-conquer principle!
  - Each layer relies on functions of the layer below
  - Each layer exports functions to the one above
  - A layer ‘sees’ the immediate lower layer as an abstraction
- Power of abstraction
  - Implementation details are hidden
  - Functions in a layer can be changed without disturbing other layers
- Interface between layers defines interaction between them

Internet (TCP/IP) Protocol Stack

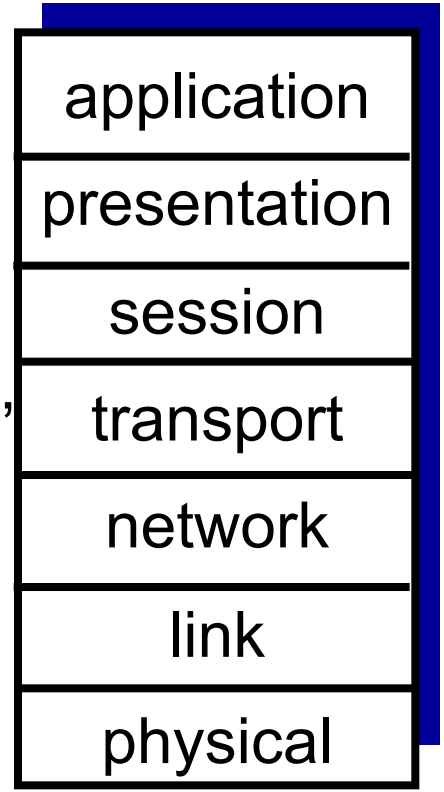
- Application: supporting network applications
  - FTP, SMTP, HTTP
- Transport: process-process data transfer
  - TCP, UDP
- Network: routing of datagrams from source to destination
  - IP, routing protocols
- Link: data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi)
- Physical: bits “on the wire”





ISO/OSI reference model

- Presentation: allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- Session: synchronization, checkpointing, recovery of data exchange
- Internet stack “missing” these layers!
  - These services, if needed, must be implemented in application



Encapsulation