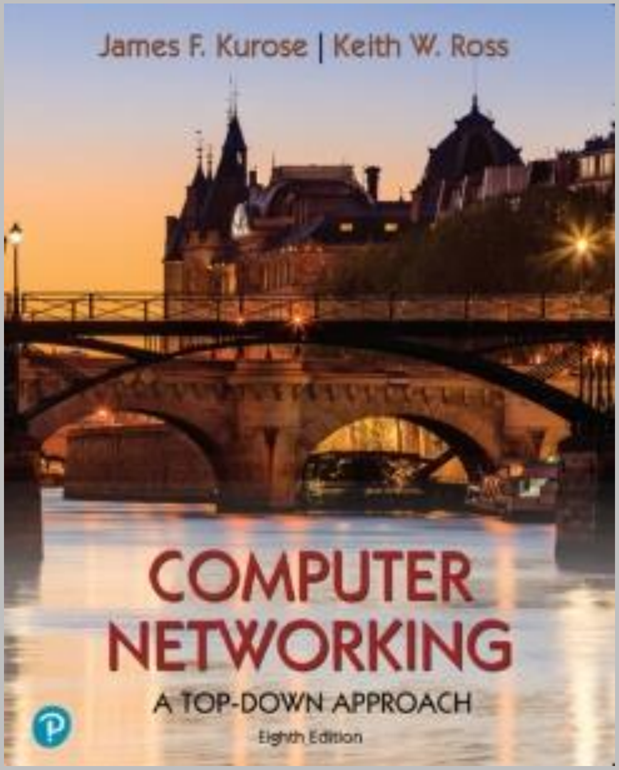
# CS 313: Networks




*The slides are adapted from Computer Networking: A Top-Down Approach, 8th edition, Jim Kurose, Keith Ross, Pearson, 2020.*

## Chapter 1

1.4 Delay, Loss, and Throughput

1.5 Protocol Layers

1.6 Networks Under Attack

Qian Mao | September 19 2023 | Whitworth University

# Self-Checking Questions

- R1. What is the difference between a host and end system? List several different types of end systems. Is a Web server an end system?
- There is no difference. Throughout this text, the words “host” and “end system” are used interchangeably. End systems include PCs, workstations, Web servers, mail servers, PDAs, Internet-connected game consoles, etc.

# Self-Checking Questions

- R3. Why are standards important for protocols?
- Standards are important for protocols so that people can create networking systems and products that interoperate.

# Self-Checking Questions

- R4. List four access technologies. Classify each one as home access, enterprise access, or wide-area wireless access.
- Cable to HFC: home;
- Fiber to the Home (FTTH); home networks
- Ethernet; enterprise
- Local area network (WLAN)
  - Wifi (802.11)
- 3G, 4G and 5G; wide-area wireless
  - AT&T: 5G networks: 850MHz, average speed: 59.3 Mbps
  - 5G speed by country: https://www.speedcheck.org/5g-index/

# Self-Checking Questions

- R6. List the available residential access technologies in your city. For each type of access, provide the advertised downstream rate, upstream rate, and monthly price.
- In most American cities, the current possibilities include: dial-up; DSL; cable modem; fiber-to-the-home.

# Self-Checking Questions

- R9. HFC, DSL, and FTTH are all used for residential access. For each of these access technologies, provide a range of transmission rates and comment on whether the transmission rate is shared or dedicated.
- DSL: up to 24 Mbps downstream and 2.5 Mbps upstream, bandwidth is dedicated;
- HFC, rates up to 42.8 Mbps and upstream rates of up to 30.7 Mbps, bandwidth is shared.
- FTTH: 2-10Mbps upload; 10-20 Mbps download; bandwidth is not shared.

# Self-Checking Questions

- R10. Describe the most popular wireless Internet access technologies today. Compare and contrast them.
- There are two popular wireless Internet access technologies today:
  - a. Wifi (802.11) In a wireless LAN, wireless users transmit/receive packets to/from an base station (i.e., wireless access point) within a radius of few tens of meters. The base station is typically connected to the wired Internet and thus serves to connect wireless users to the wired network.
  - b. 3G and 4G wide-area wireless access networks. In these systems, packets are transmitted over the same wireless infrastructure used for cellular telephony, with the base station thus being managed by a telecommunications provider. This provides wireless access to users within a radius of tens of kilometers of the base station.

# Self-Checking Questions

- R11. Suppose there is exactly one packet switch between a sending host and a receiving host. The transmission rates between the sending host and the switch and between the switch and the receiving host are R1 and R2, respectively. Assuming that the switch uses store-and-forward packet switching, what is the total end-to-end delay to send a packet of length L? (Ignore queuing, propagation delay, and processing delay.)
- The end-to-end delay is $L/R1 + L/R2$

# Self-Checking Questions

- R14. Why will two ISPs at the same level of the hierarchy often peer with each other? How does an IXP earn money?
- If the two ISPs do not peer with each other, then when they send traffic to each other they have to send the traffic through a provider ISP (intermediary), to which they have to pay for carrying the traffic. By peering with each other directly, the two ISPs can reduce their payments to their provider ISPs. An Internet Exchange Points (IXP) (typically in a standalone building with its own switches) is a meeting point where multiple ISPs can connect and/or peer together. An ISP earns its money by charging each of the ISPs that connect to the IXP a relatively small fee, which may depend on the amount of traffic sent to or received from the IXP.

# Chapter 1: roadmap

- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History

# Discussion

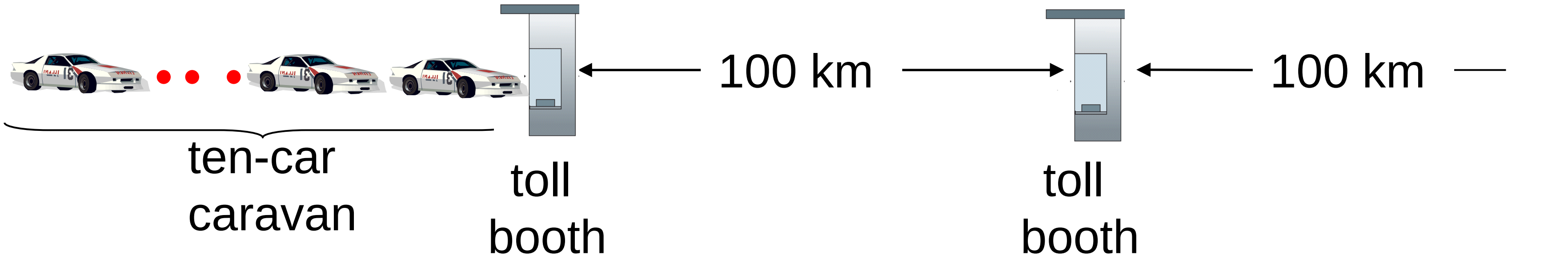
- If we are going to visit our friend, Mario, in caravan, what factors impact the travelling time?



- ✓ Speed of your vehicle
- ✓ Distance
- ✓ Transmission speed
- ✓ Traffic
- ✓ Service speed

https://www.youtube.com/watch?v=HHKwnUa3txo

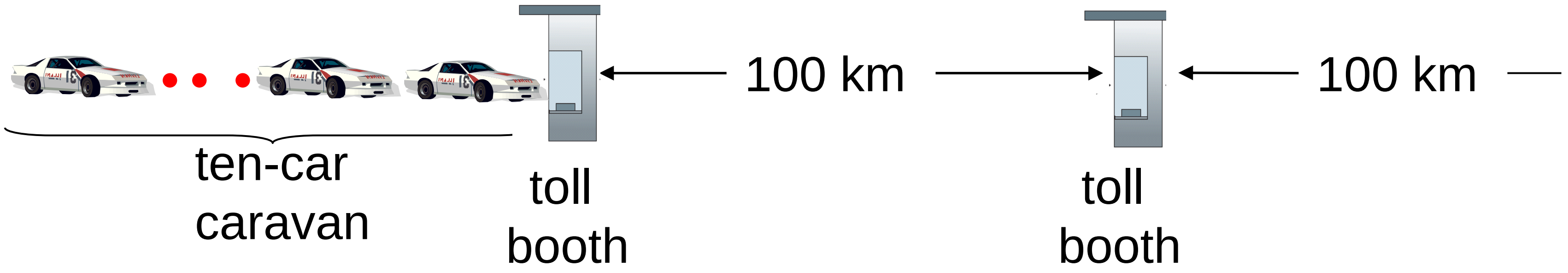
# Caravan analogy




- car ~ bit; caravan ~ packet
- cars "propagate" at 100 km/hr
- toll booth takes 12 sec to service car (bit transmission time)
- Q: How long until the entire caravan is lined up before 2nd toll booth?

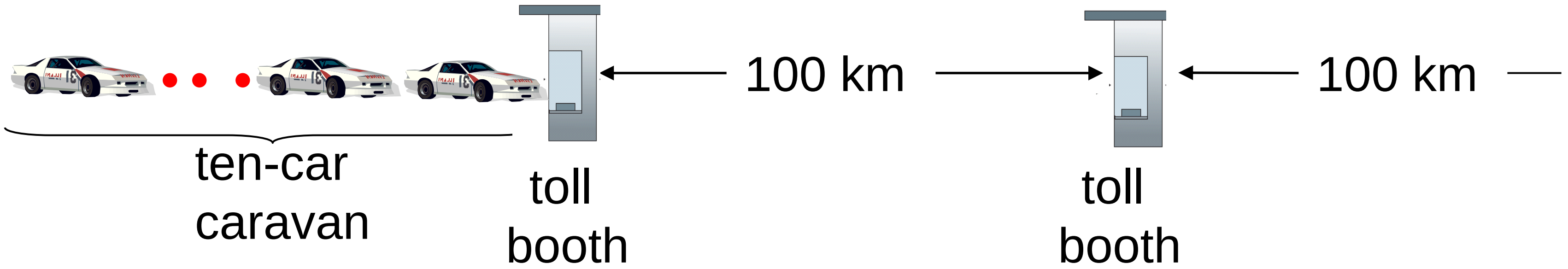
- time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- *Answer:* 62 minutes

# Caravan analogy (more)



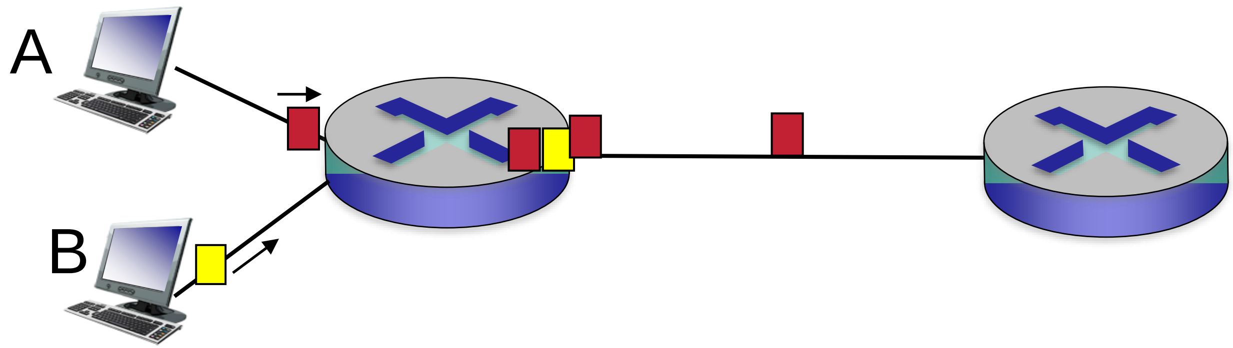

- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one min to service a car
- <u>Q:</u> Will cars arrive to 2nd booth before all cars serviced at first booth?

# Caravan analogy (more)



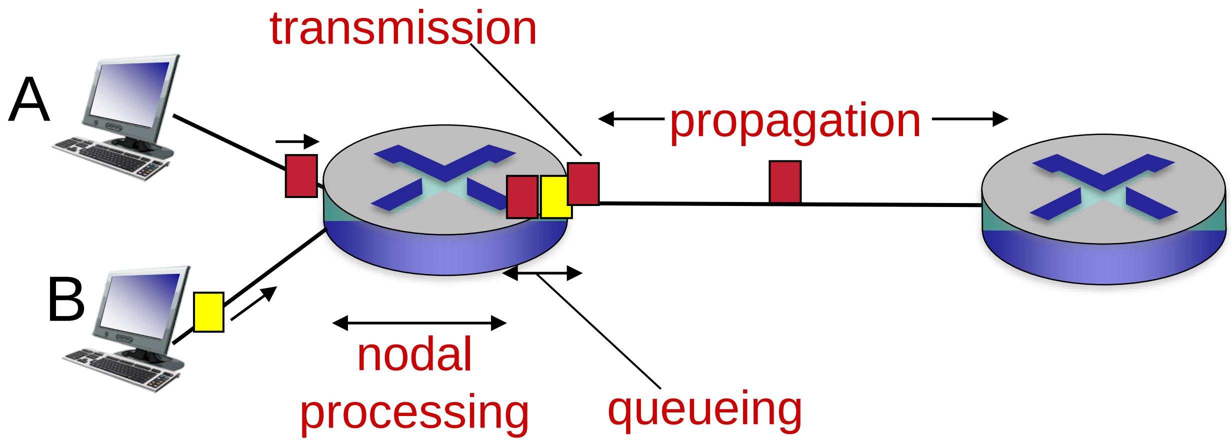

- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one min to service a car
- Q: Will cars arrive to 2nd booth before all cars serviced at first booth?
  - *Yes!* after 7 min, first car arrives at second booth; three cars still at first booth

# Discussion




- When a packet is transmitted from a source node to a destination node, what reasons cause packet delay? (the amount of time it takes for the packet to travel from the source to the destination)

# Four sources of packet delay




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

$d_{trans}$: transmission delay:

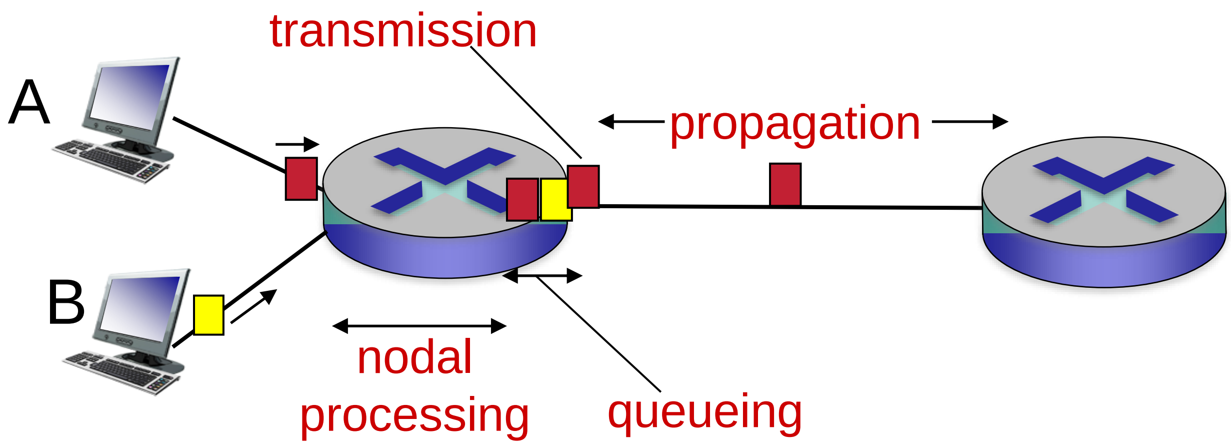
- $L$: packet length (bits)
- $R$: link *bandwidth (bps)*
- $d_{trans} = L/R$

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

$d_{prop}$: propagation delay:

- $d$: length of physical link
- $s$: propagation speed (~$2\times10^8$ m/sec)
- $d_{prop} = d/s$

# 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

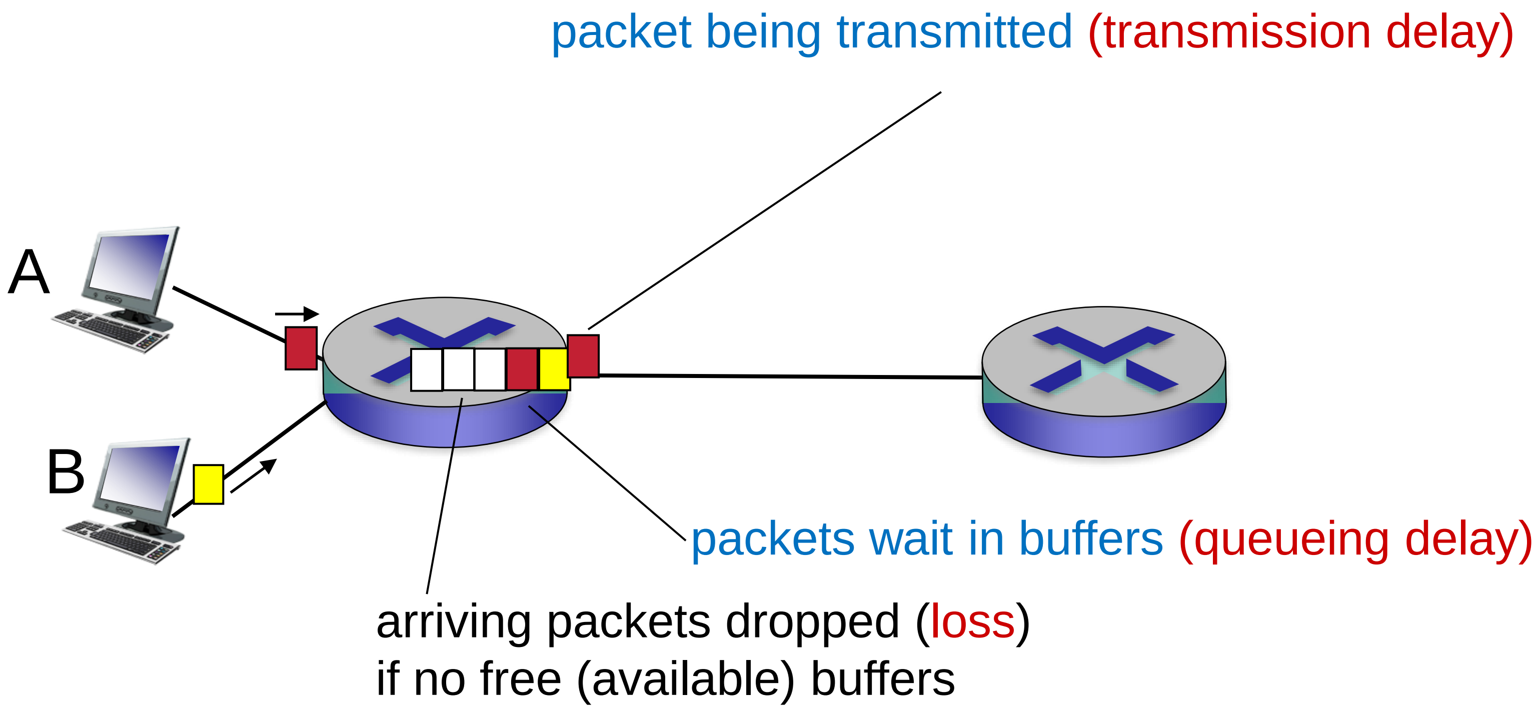
## $d_{queue}$: queueing delay

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

# packet queue in a router

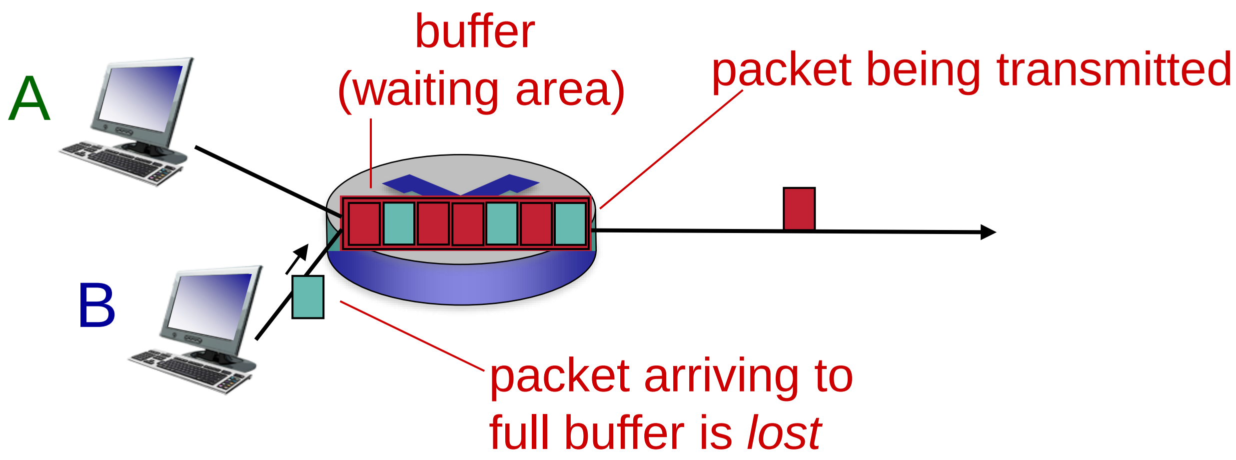
packets *queue* in router buffers, wait for turn

- packet arrival rate to link (temporarily) exceeds output link capacity: Packet Loss



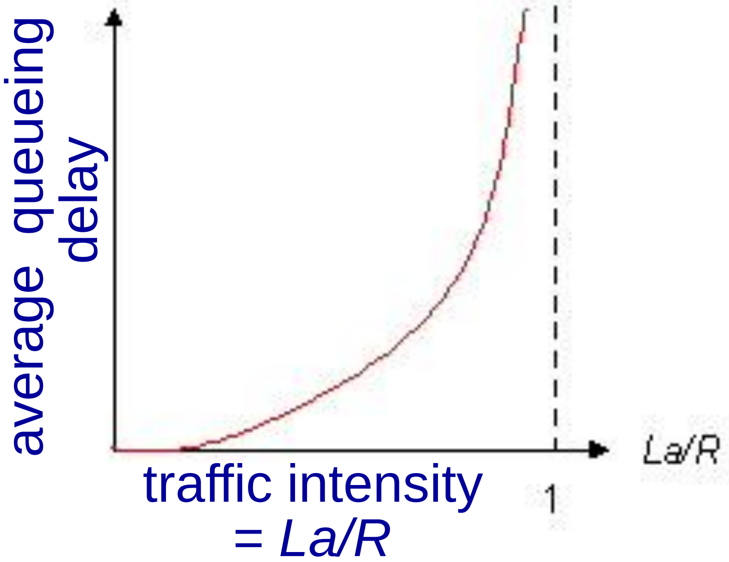
# Packet loss

- 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, by source end system, or not at all



# Queueing delay

- *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

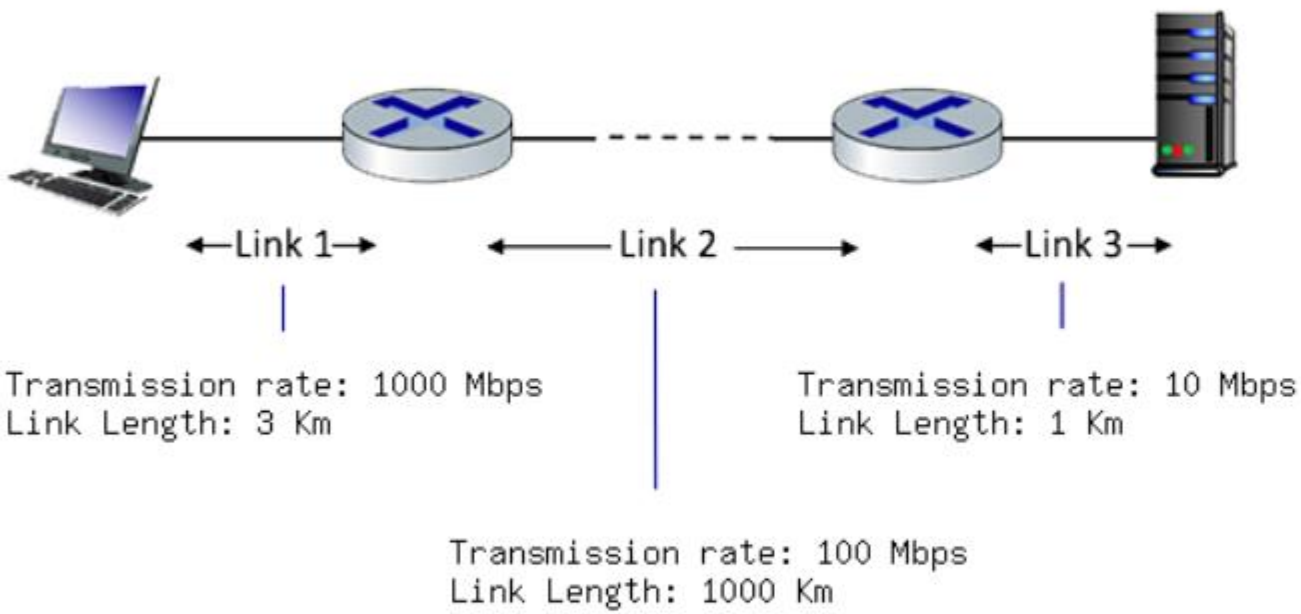
*La/R* -> 1

# Question

- How do we know the value of each delay?
  - transmission delay
  - propagation delay
  - nodal processing delay
  - queueing delay

# Exercise

- Consider the figure below, with three links, each with the specified transmission rate and link length.
- The length of a packet is 12000 bits. The speed of light propagation delay on each link is 3x10^8 m/sec.
- Assume the processing delay and the queueing delay are zero.
- Calculate the end-end delay in second. Round the result to six decimals.



# Answer

- Link 1:
  - Transmission Delay = $12000 * 10^{-6} / 1000 = 12 * 10^{-6}$ s
  - Propagation Delay = $3 * 10^{3} / * (3 * 10^{8}) / = 10^{-5}$ s
- Link 2:
  - Transmission Delay = $12000 * 10^{-6} / 100 = 12 * 10^{-5}$ s
  - Propagation Delay = $1000 * 10^{3} / * (3*10^{8}) / = (1/3) * 10^{-2}$ s
- Link 3:
  - Transmission Delay = $12000 * 10^{-6} / 10 = 12 * 10^{-4}$ s
  - Propagation Delay = $1 * 10^{3} / * (3 * 10^{8}) / = (1/3) * 10^{-5}$ s

- End-End Delay = $12 * 10^{-6} + 10^{-5}$

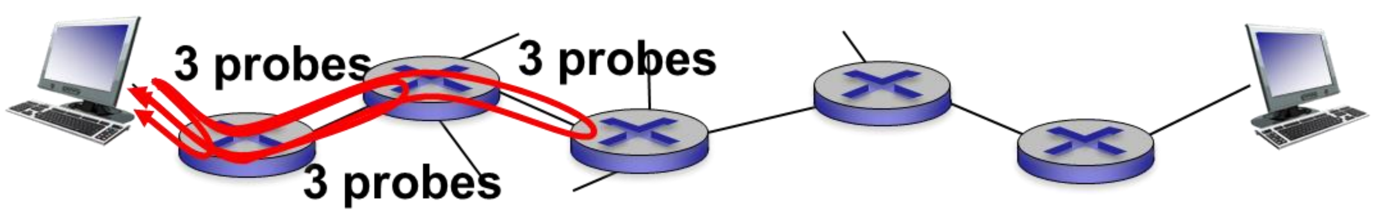
  $+ 12 * 10^{-5} + (1/3) * 10^{-2}$

  $+ 12 * 10^{-4} + (1/3) * 10^{-5}$

  $= 0.004678$ s $= 4.678$ ms

# "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.



# Try it

- In command window, type:

  tracert www.google.com

```
H:\>tracert www.google.com

Tracing route to www.google.com [142.251.33.68]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.21.34.254
  2    <1 ms    <1 ms    <1 ms  10.19.21.2
  3     1 ms     1 ms    <1 ms  198.29.2.2
  4     3 ms     1 ms     1 ms  209.210.59.170
  5    13 ms    15 ms    13 ms  209.63.101.6
  6    13 ms    13 ms    14 ms  209.63.101.6
  7    26 ms    14 ms    16 ms  209.85.173.42
  8    13 ms    13 ms    14 ms  108.170.245.113
  9    17 ms    17 ms    17 ms  142.251.50.245
 10    13 ms    21 ms    16 ms  sea09s28-in-f4.1e100.net [142.251.33.68]

Trace complete.
```

# Try it

- What's going to happen if we try

  tracert www.whitworth.edu

```
H:\>tracert www.whitworth.edu

Tracing route to www.whitworth.edu [198.29.3.9]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.21.34.254
  2    <1 ms    <1 ms    <1 ms  10.19.21.2
  3     1 ms     1 ms     1 ms  web1.whitworth.edu [198.29.3.9]

Trace complete.
```

# Try it

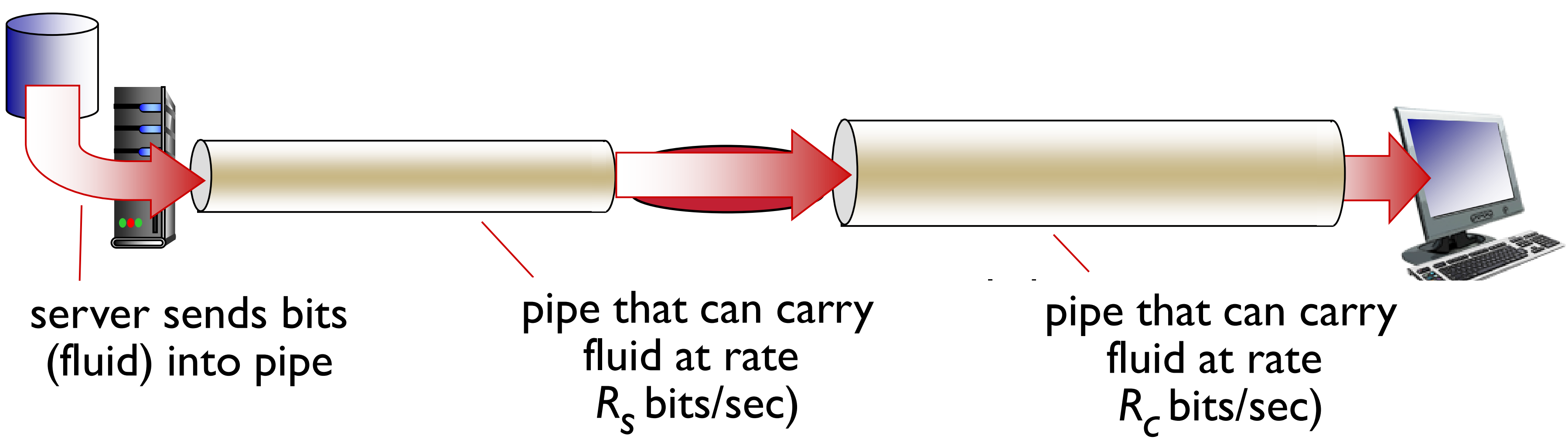
- Traceroute a server abroad?

  tracert www.amazon.in

```
H:\>tracert www.amazon.in

Tracing route to d1elgm1ww0d6wo.cloudfront.net [13.224.30.165]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.21.34.254
  2    <1 ms    <1 ms    <1 ms  10.19.21.2
  3    <1 ms    <1 ms    <1 ms  198.29.2.2
  4     2 ms     3 ms     2 ms  209.210.59.170
  5    13 ms    13 ms    13 ms  209.63.101.6
  6    15 ms    13 ms    13 ms  209.63.101.6
  7    13 ms    13 ms    15 ms  ae23.mpr1.sea1.us.zip.zayo.com [64.125.12.69]
  8    17 ms    16 ms    16 ms  ae28.cs2.sea1.us.eth.zayo.com [64.125.29.104]
  9    16 ms    16 ms    16 ms  ae27.mpr2.sea1.us.zip.zayo.com [64.125.29.3]
 10    13 ms    13 ms    13 ms  99.82.182.102
 11     *       14 ms     *     150.222.136.59
 12    13 ms    13 ms    13 ms  52.95.53.145
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15    17 ms    17 ms    17 ms  205.251.225.27
 16     *        *        *     Request timed out.
 17     *        *        *     Request timed out.
 18     *        *        *     Request timed out.
 19     *        *        *     Request timed out.
 20     *        *        *     Request timed out.
 21    17 ms    17 ms    17 ms  server-13-224-30-165.sea19.r.cloudfront.net [13.224.30.165]

Trace complete.
```

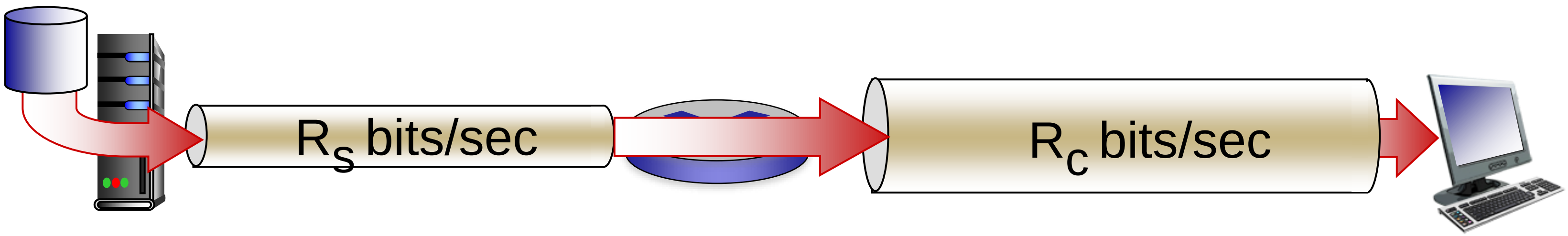
# Throughput

- *throughput:* rate (bits/time unit) at which bits are being sent from sender to receiver

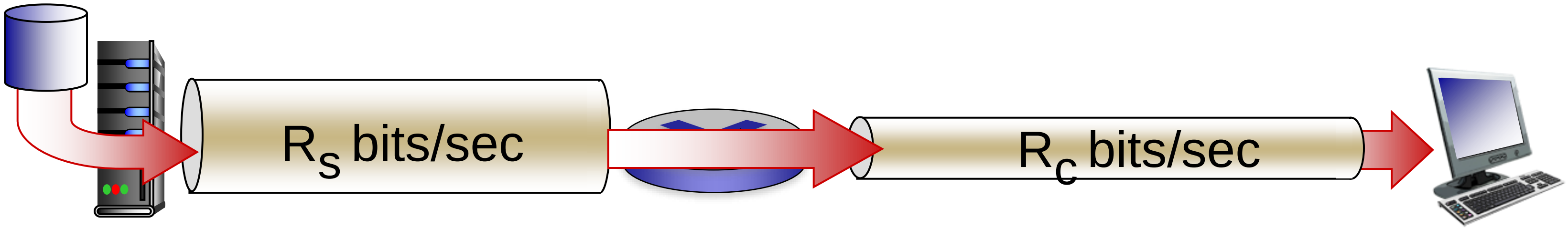


# Throughput (more)

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




- $R_s > R_c$ What is average end-end throughput?

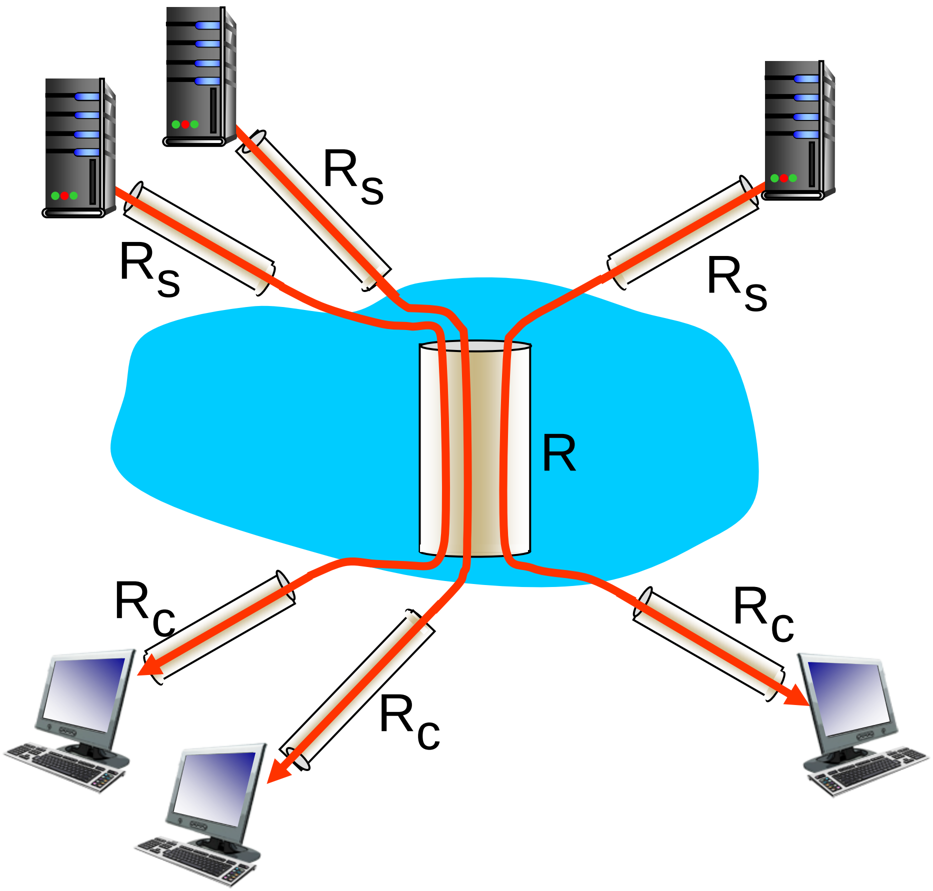



*bottleneck link*

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

# Throughput: Internet scenario

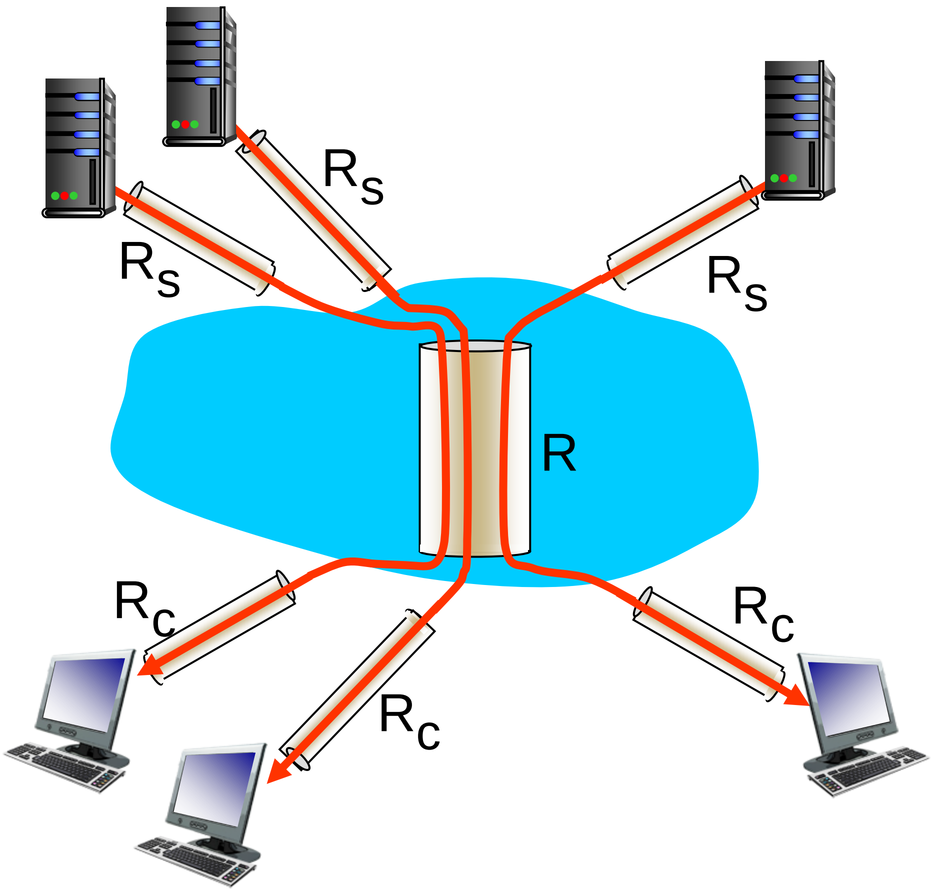
- per-connection end-end throughput:




10 connections (fairly) share backbone bottleneck link *R* bits/sec

# Throughput: Internet scenario

- per-connection end-end throughput: $min(R_c,R_s,R/10)$
- in practice: $R_c$ or $R_s$ is often bottleneck



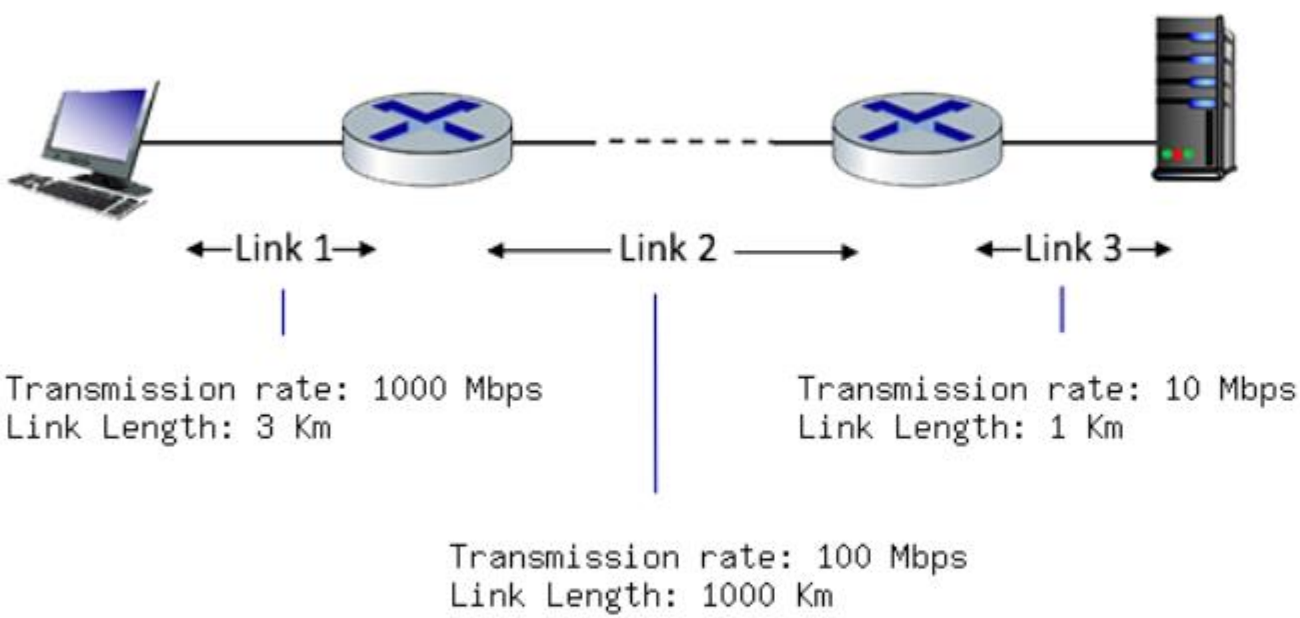

10 connections (fairly) share backbone bottleneck link *R* bits/sec

# Calculate Throughput

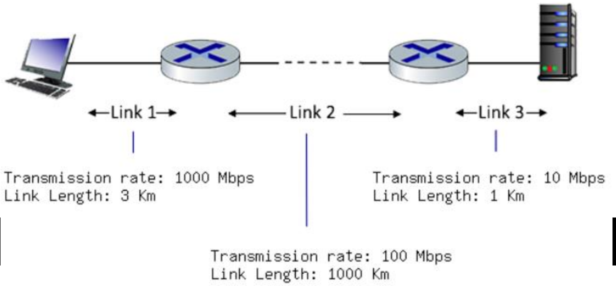
- In the two-router exercise, what's the throughput?

# Calculate Throughput

- In the two-router exercise, what's the throughput?
  - If there is only one packet being sent out from the source, the throughput = 12000 / 0.004678 = 2.565 Mbit/s

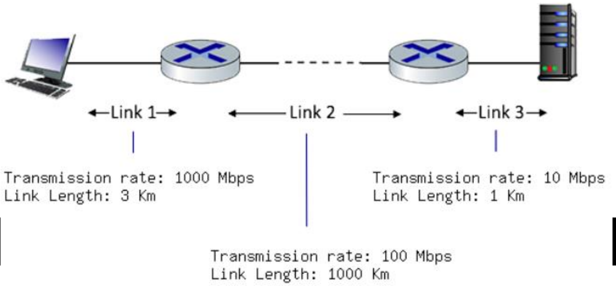


# Calculate Throughput



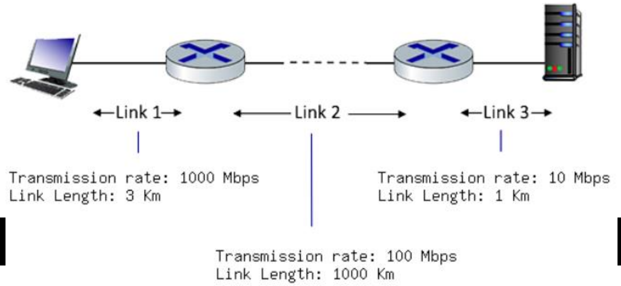

- In the two-router exercise, what's the throughput?
  - If there is only one packet being sent out from the source, the throughput = 12000 / 0.004678 = 2.565 Mbit/s
  - Why it is much lower than the minimum of link capacities?

# Calculate Throughput



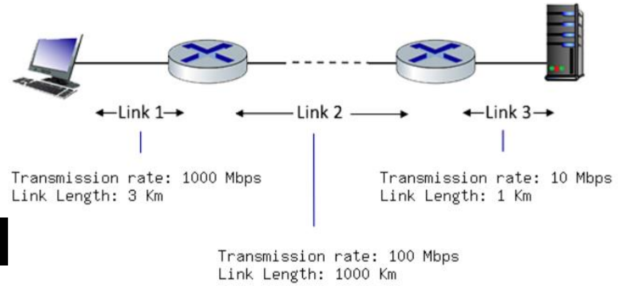

- In the two-router exercise, what's the throughput?
  - If there is only one packet being sent out from the source, the throughput = 12000 / 0.004678 = 2.565 Mbit/s
  - Why it is much lower than the minimum of link capacities?
    - Most likely, the next packet will be sent out by the source BEFORE the first packet reaches the destination, therefore, the throughput would be higher.

# Calculate Throughput




- In the two-router exercise, what's the throughput?
  - If there is only one packet being sent out from the source, the throughput = 12000 / 0.004678 = 2.565 Mbit/s
  - Why it is much lower than the minimum of link capacities?
    - Most likely, the next packet will be sent out by the source BEFORE the first packet reaches the destination, therefore, the throughput would be higher.
  - How to get a more accurate calculation of throughput?

# Calculate Throughput




- In the two-router exercise, what's the throughput?
  - If there is only one packet being sent out from the source, the throughput = 12000 / 0.004678 = 2.565 Mbit/s
  - Why it is much lower than the minimum of link capacities?
    - Most likely, the next packet will be sent out by the source BEFORE the first packet reaches the destination, therefore, the throughput would be higher.
  - How to get a more accurate calculation of throughput?
  - Should the real throughput be smaller/greater/equal to the min(R1, R2, R3)? (R is the transmission rate.)

# Chapter 1: roadmap

- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History

# Protocol "layers"

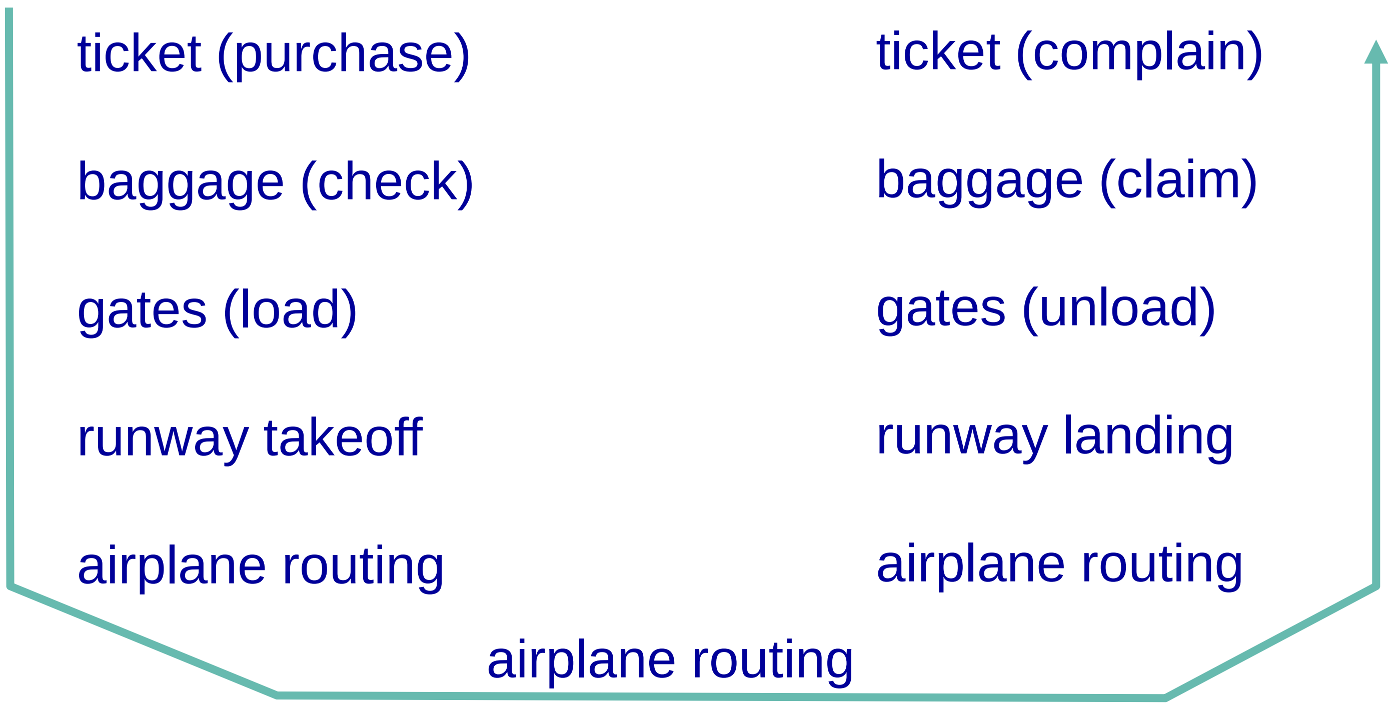
*Networks are complex, with many "pieces":*

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

*Question:*

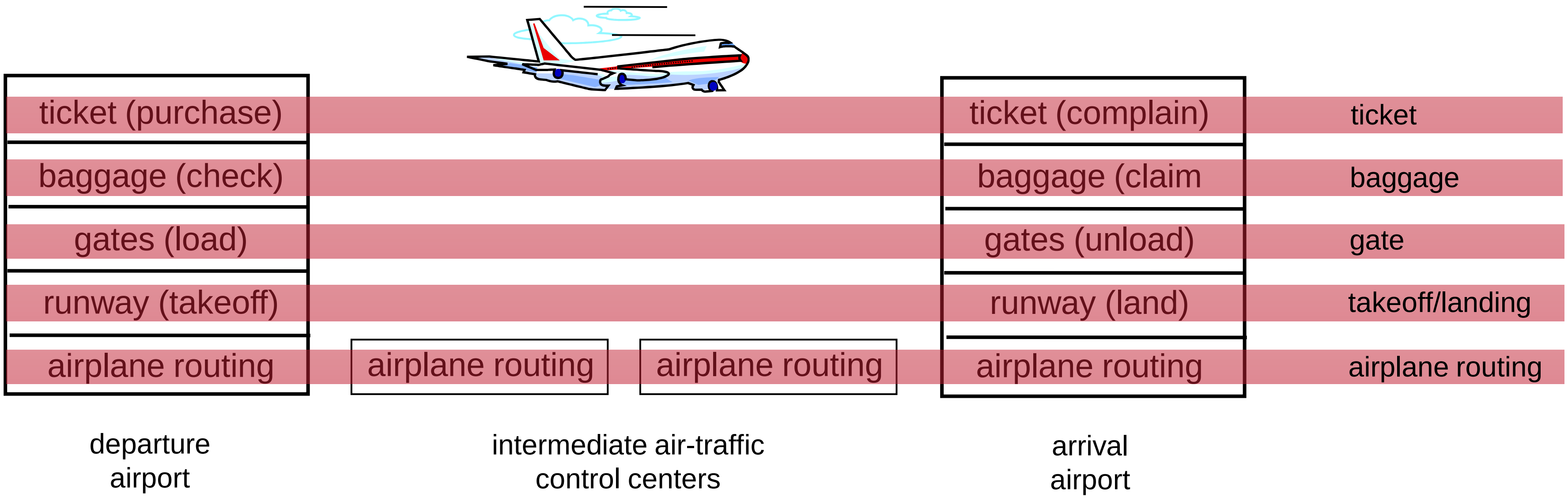
Is there any hope of *organizing* structure of network?

# 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

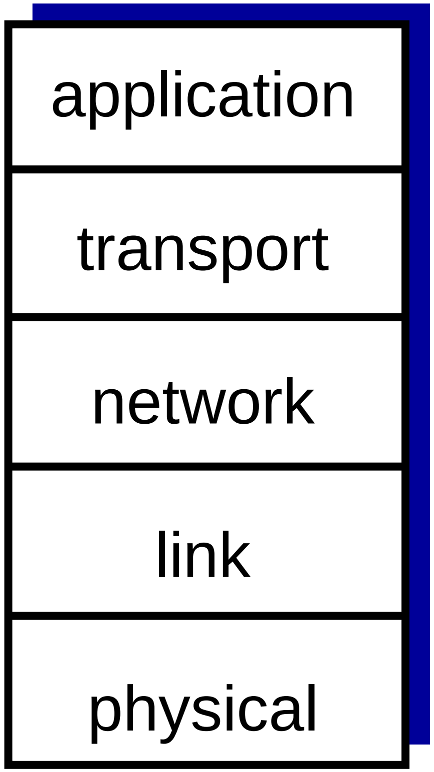
# Why layering?

dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
    - e.g., change in gate procedure doesn't affect rest of system
- What are the other layered models in our life?
- layering considered harmful?

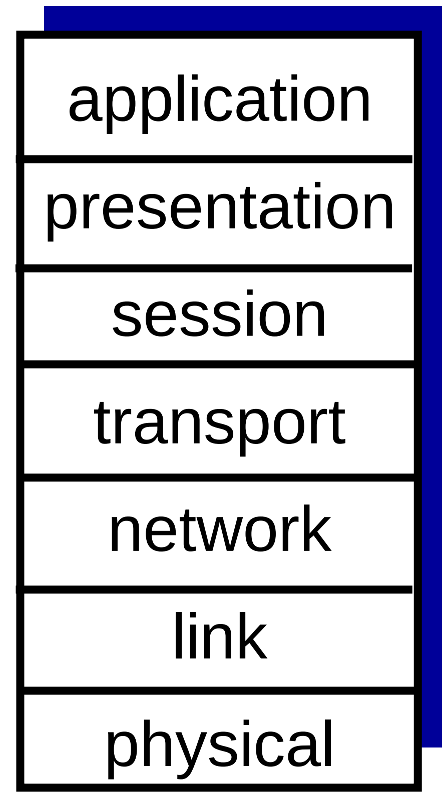
# Internet protocol stack

- *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, 802.111 (WiFi), PPP
- *physical:* bits "on the wire"

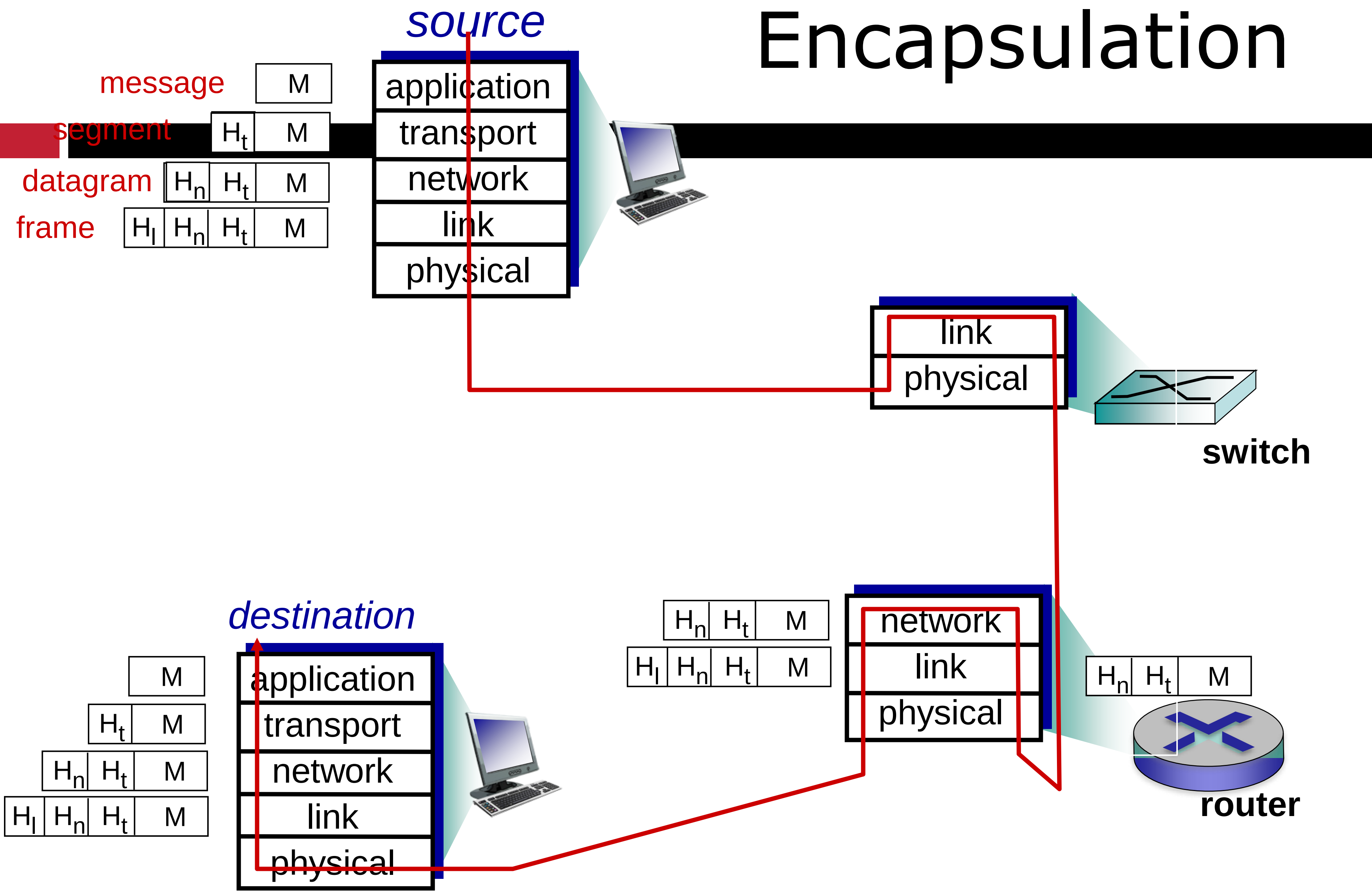


# OSI reference model (Open System Interconnection )

- *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



# Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

# 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" ☺

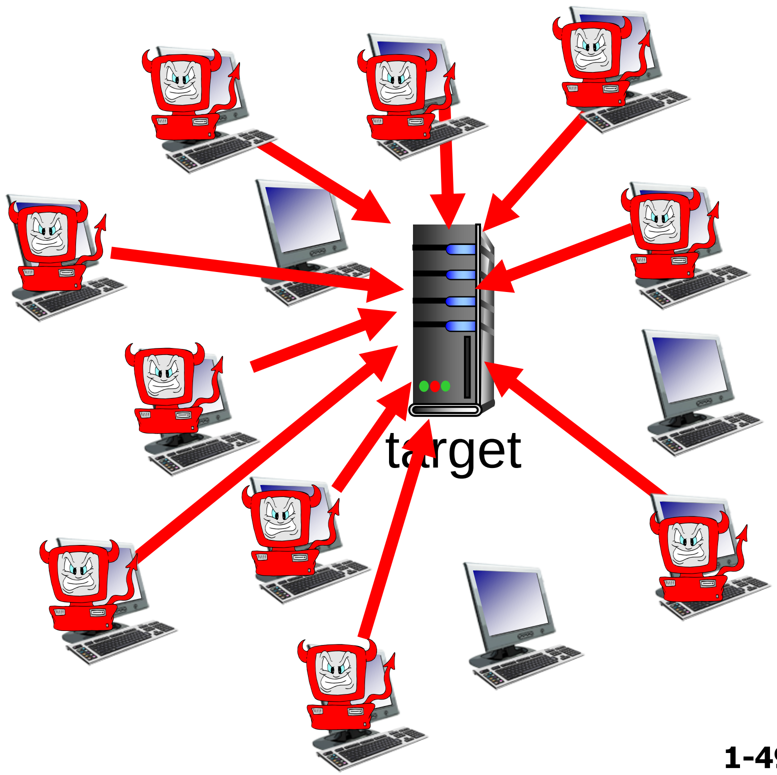
# Bad guys: put malware into hosts via Internet

- 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

# 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

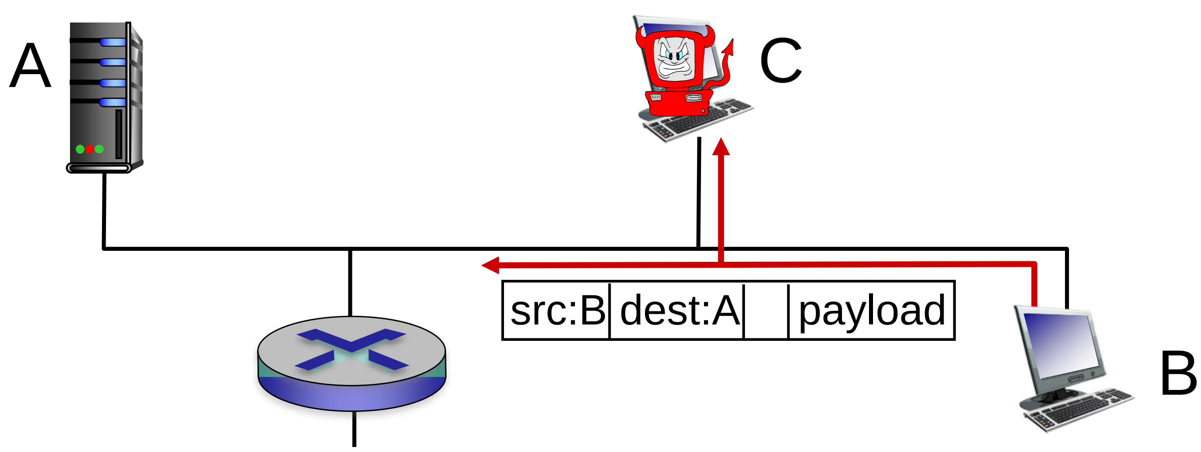
1. select target
2. break into hosts around the network
3. send packets to target from compromised hosts



# Bad guys can sniff packets

*packet "sniffing":*

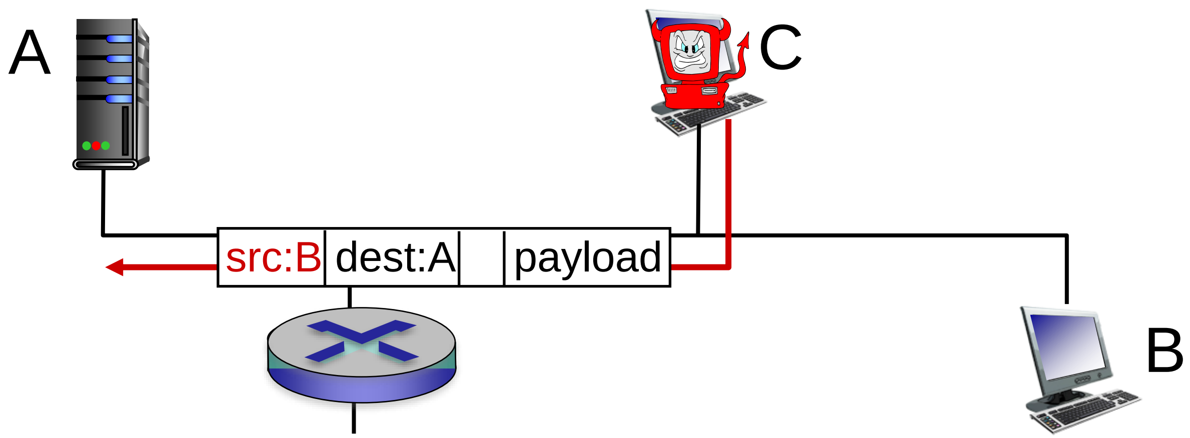
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by




- wireshark software is a (free) packet-sniffer

# Bad guys can use fake addresses

*IP spoofing:* send packet with false source address




*… lots more on security in Chapter 8*

# Introduction: summary

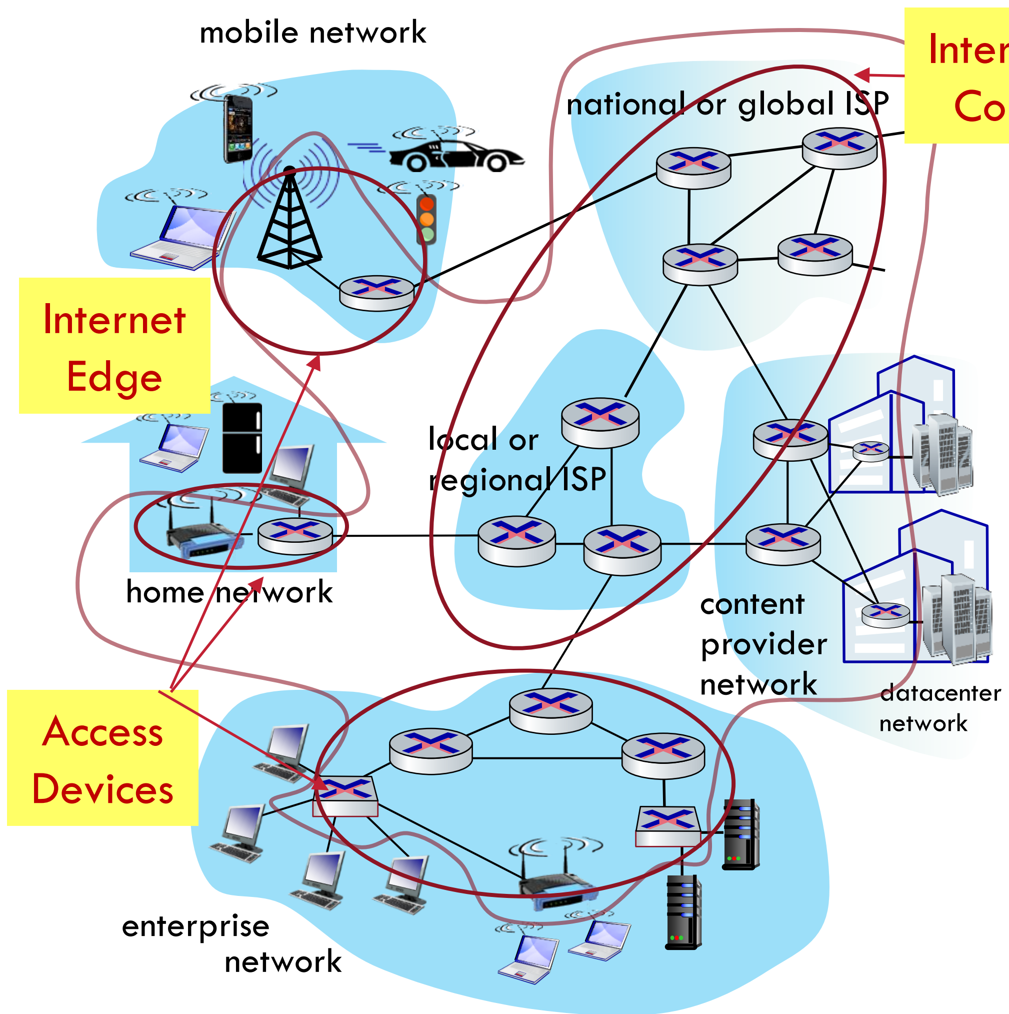
*covered a "ton" of material!*

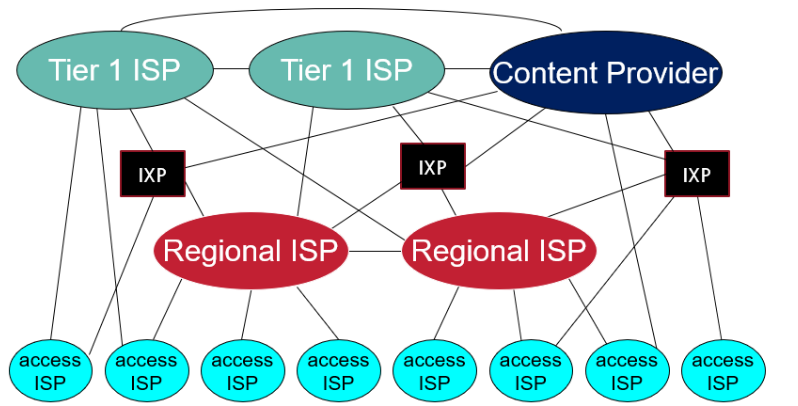
- Internet overview
- what's a protocol?
- network edge, core, access network
  - Internet structure
- performance: loss, delay, throughput
- layering, service models
- security
- history

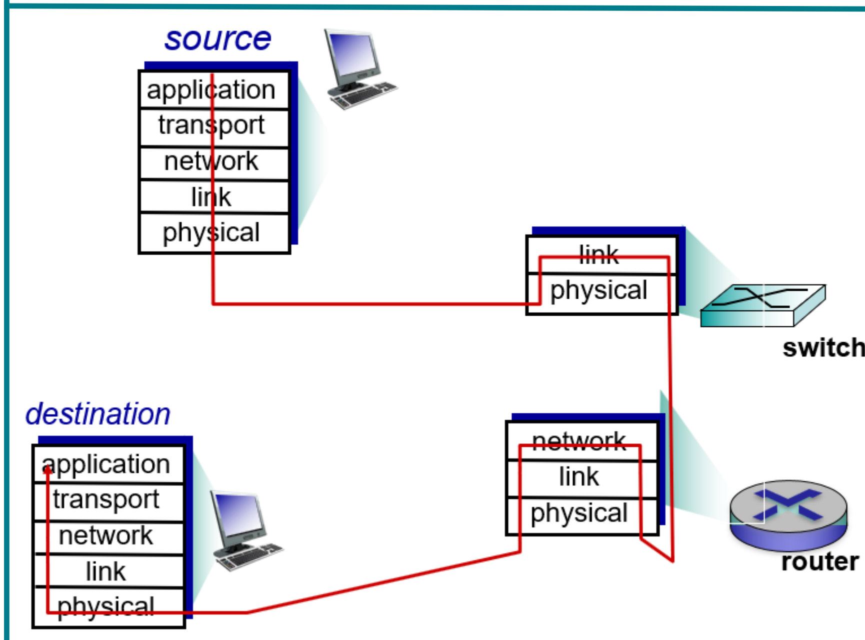
*you now have:*

- context, overview, "feel" of networking
- more depth, detail *to follow!*

# Introduction: summary



# Self-Checking Questions

- Textbook page 67-68
  - R16, R18, R19
  - R23
  - R28

# Moving Forward

- Tuesday
  - Quiz
  - Chapter 2
  - LM1 due
- Reading: 1.4,1.5, 1.6