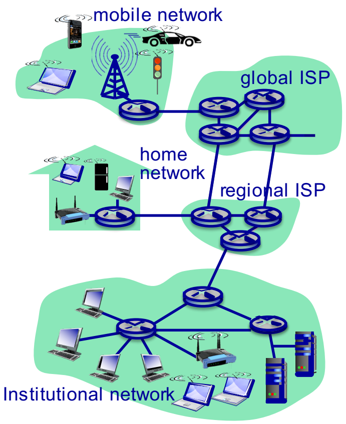
# A Whirlwind Introduction to the Internet
## Overview

- What's the Internet
- Network core
- Network edge
- Access nets, physical media
- Internet Structure & ISPs
- Performance: loss, delay
- Security
- Protocol layers, service models

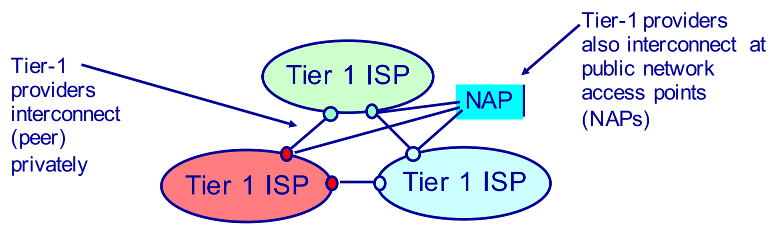





Internet is a network of networks

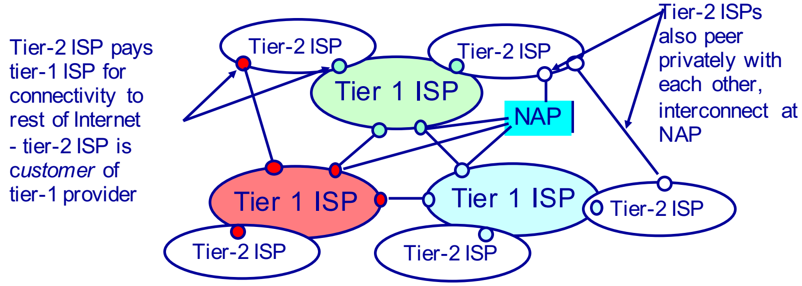
# Internet Structure: Network of Networks

- Roughly Hierarchical
- At center: "tier-1" ISPs (e.g., UUNet, Level 3, Sprint, AT&T), national/international coverage
  - » treat each other as equals



# Internet Structure: Network of Networks

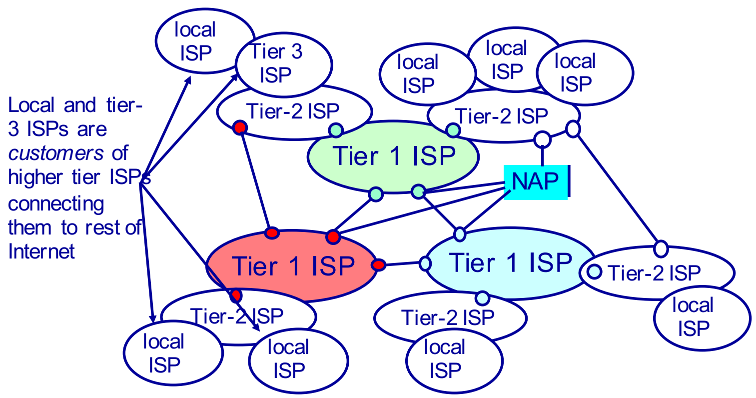
- "Tier-2" ISPs: smaller (often regional) ISPs
  - » Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs





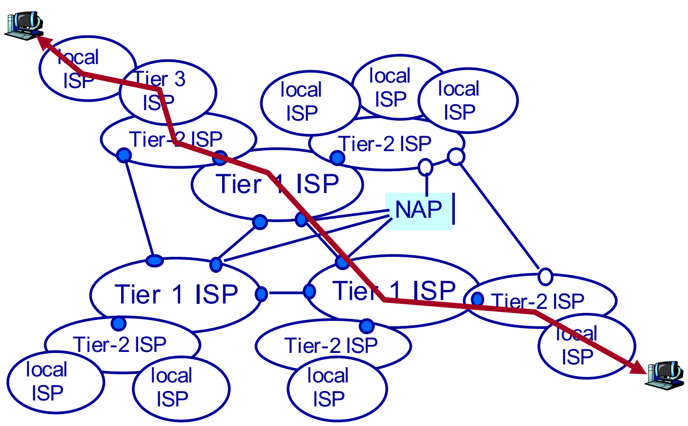

# Internet Structure: Network of Networks

- "Tier-3" ISPs and local ISPs
  - » last hop ("access") network (closest to end systems)



## Internet Structure: Network of Networks

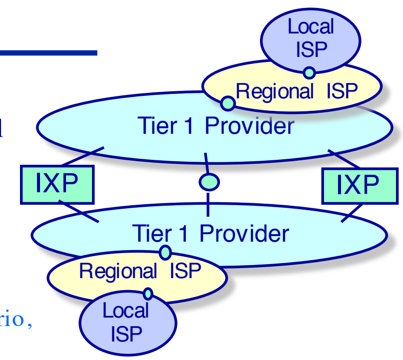
- A packet passes through many networks!






## Internet Architecture

### Summary




- The Internet is roughly hierarchical
- National/international backbone providers (“Tier 1”) at “the root”
  - » AT&T, Verizon, Sprint (Softbank Broadband), Century Link (Qwest), Level 3 (Global Crossing), NTT/Verio, Cogen
- Tier 1 providers interconnect (“peer”) with each other privately, or at a public Internet exchange/peering point (IXP)
- Regional ISPs connect into Tier 1 provider’s network
- Local ISPs connect into regional ISPs

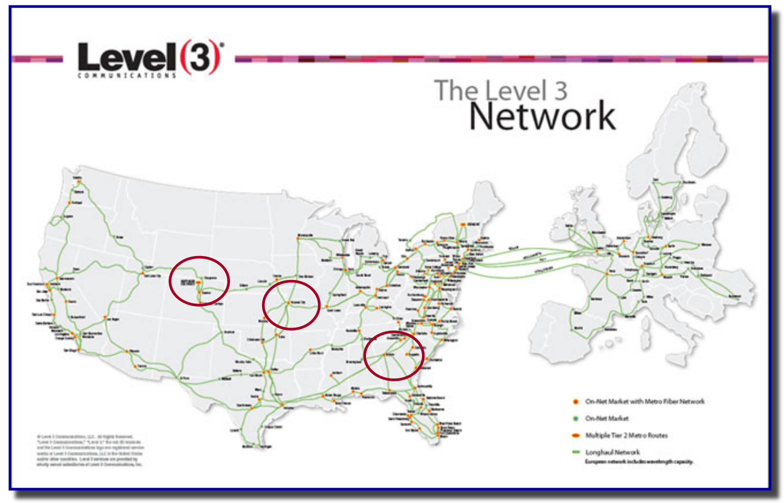
## Network Maps

Just how big are Tier-1 ISPs…?



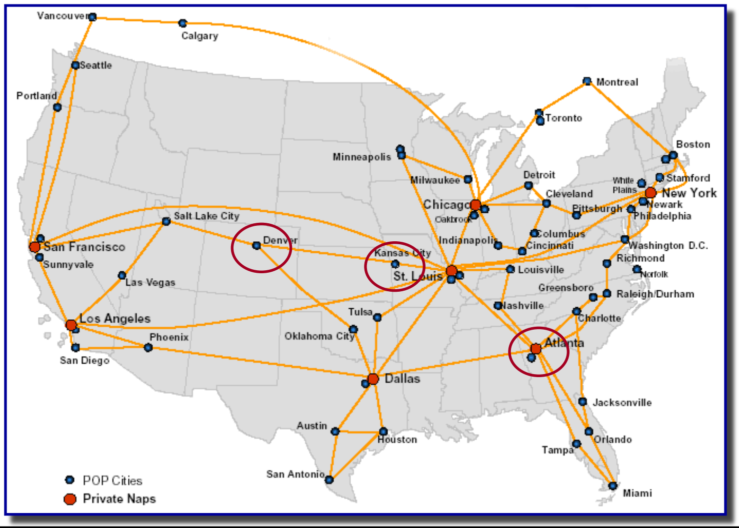
## The Network Core

### A map of the Internet (Level 3)…



# The Network Core

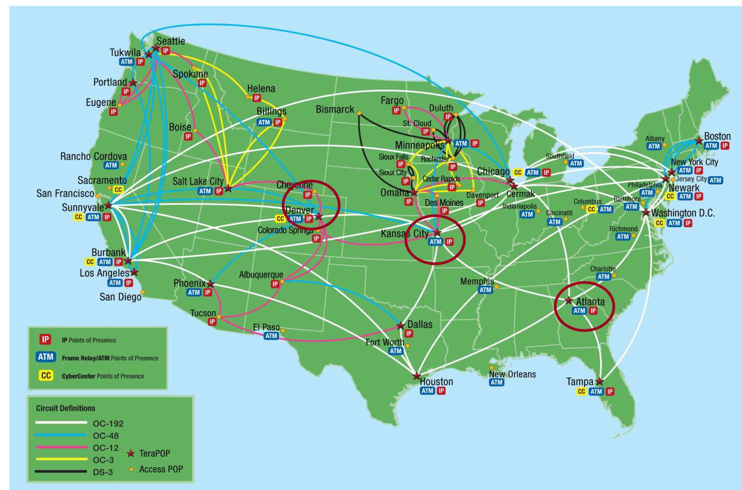
## A map of the Internet (Savvis)…





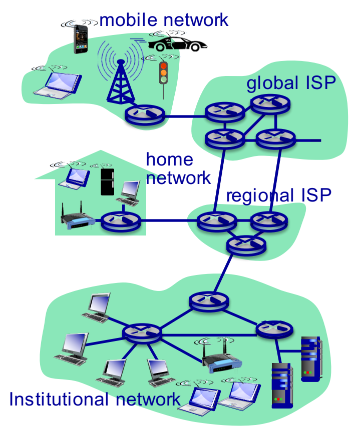

# The Network Core

## A map of the Internet (Qwest in 2006)…



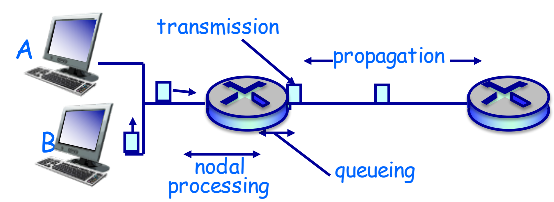
# A Whirlwind Introduction to the Internet
## Overview

- What's the Internet
- Network core
- Network edge
- Access nets, physical media
- Internet Structure & ISPs
- Performance: loss, delay
- Security
- Protocol layers, service models






# Understanding the Performance of the Internet
## Delay in packet-switched networks




- Packets experience variable delays along path from source to destination
- Four sources of delay at *each* hop
  - » Nodal processing:
    - ❖ Check for bit errors
    - ❖ Determine the output interface to forward packet on
  - » Queuing:
    - ❖ Time spent waiting at outbound interface for transmission
    - ❖ Duration depends on the level of congestion at the interface
  - » Transmission
  - » Propagation

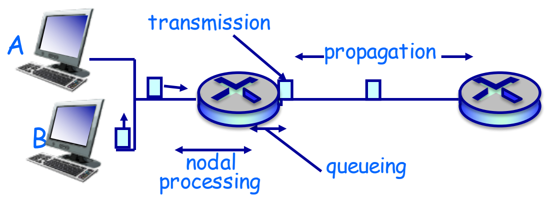


Packet Switch Network is a store and switch network: every node that you visit will be doing nodal processing

router determines which outgoing link the packet should go on

# Understanding the Performance of the Internet

## Delay in packet-switched networks




Beware: $s$ and $R$ are *very* different quantities!

- Transmission delay = time to "put bits onto the link" = $L/R$
  - » $R$ = link bandwidth (bps)
  - » $L$ = packet length (bits)
- Propagation delay = $d/s$
  - » $d$ = length of physical link
  - » $s$ = signal propagation speed in medium (~$2\text{x}10^8$ $m/sec$)

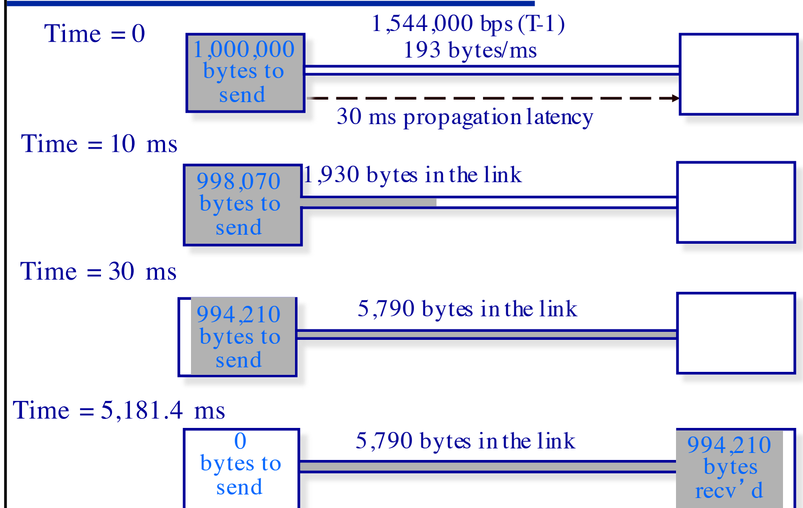


Transmission delay: time to encode bits into electromagnetic wave and put it on the wire

Propagation delay:
- depends on the speed of light or the speed of electromagnetic waves on the medium;
- pretty small in most cases

# Transmission & Propagation Example

## Transmission on a "slow" link




Time to encode + time to transfer across wire

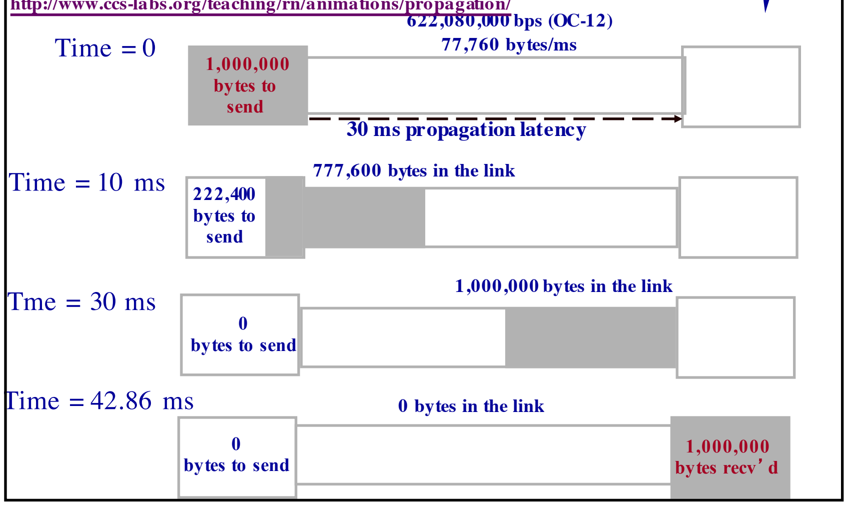
# Transmission & Propagation Example

## Transmission on a “fast” link

Animation

https://wps.pearsoned.com/ecs_kurose_compnetw_6/216/55463/14198702.cw/index.html

http://www.ccs-labs.org/teaching/rn/animations/propagation/




T

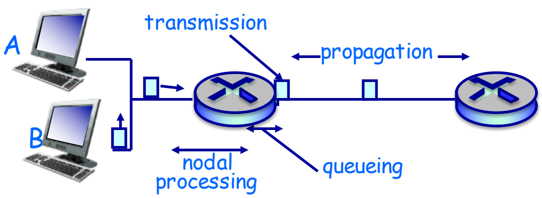
# Transmission Delay

## Telecommunications transmission speed alphabet soup

- DS-1/T-1 = 1.544 Mbps
- DS-3/T-3 = 44.736 Mbps
- OC-1 = 51.84 Mbps
- OC-*n* = *n* × OC-1
  - » OC-3 = 3 × OC-1 (155.52 Mbps)
  - » OC-12 = 12 × OC-1 (622.08 Mbps)
  - » OC-48 = 48 × OC-1 (2,488.32 Mbps or “2.5 Gbps”)
  - » OC-192 = 192 × OC-1 (9,953.28 Mbps or “10 Gbps”)
  - » OC-768 = 768 × OC-1 (39,813.12 Mbps or “40 Gbps”)

# Understanding the Performance of the Internet

## Delay in packet-switched networks




- Typical transmission delay: 120 $\mu s$
  - » 1,500 byte packet on a 100 Mbps Ethernet
- Typical propagation delay:
  - » $\leq 1$ $\mu s$ on a small campus
  - » $\approx$ 25-30 $ms$ to the West coast (and back)
- Typical processing delay:
  - » ??

Very low

- Typical queuing delay:
  - » ??



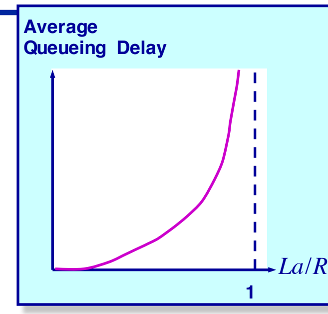
for 10mbs

(1500*8)/10^7

# Delay in Packet-Switched Networks

## Queuing delay & traffic intensity

- Understand queuing delay in terms of *traffic intensity* $La/R$
  - » $R$ = link transmission speed (bps)
  - » $L$ = packet length (bits/packet)
  - » $a$ = average packet arrival rate (packets/second)




- If $La/R \sim 0$: Average queuing delay small
- As $La/R \Rightarrow 1$: Delays become large
- If $La/R > 1$: Work arrives faster than it can be serviced
  - » Average delay goes to infinity (with infinite buffers)!
  - » With finite buffers???



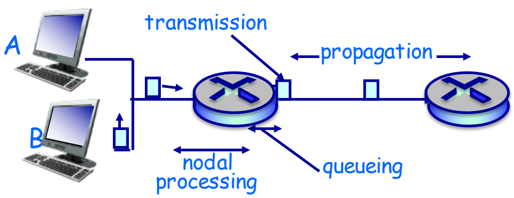
Traffic intensity: how does the load on the link compare to its transmission capacity

R: rate you can send out
L: load

Queing delay is going to increase exponentially as traffic delay gets closer to 1

When queuing delay is high packets will get dropped

## Understanding the Performance of the Internet
### Delay in packet-switched networks




Transmission delay = 120 $\mu s$
Propagation delay = 10 $\mu s$
Queuing delay = $k$ x 120 $\mu s$
Processing delay = 100 $\mu s$

What is $k$?

- What dominates end-to-end delay?
- Note that processing, transmission, and queuing delays are encountered at each hop
  - » End-to-end delay is largely a function of the number of routers encountered along the path from source to destination

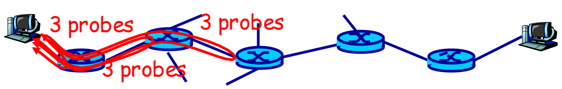


Queuing delay is proportional to the number of packets
Transmission, propagation, and processing delay are all fixed

Now have everything needed to do HW2

## "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.






Traceroute exploits ttl going to 0

Send ttl of 1 that exploit a counter
when the ttl reaches a router it will be decremented, if the
ttl gets to 0 it will send an error back to the computer
running the tracerout program

## Seeing Paths and Delays in the Internet

- www.traceroute.org



## Understanding the Performance of the Internet

### Example: What is the delay to cs.utexas.edu?

```
>>> traceroute cs.utexas.edu

traceroute: Warning: cs.utexas.edu has multiple addresses; using
128.83.139.9
traceroute to cs.utexas.edu (128.83.139.9), 30 hops max, 38 byte packets
 1  ciscokid-cs.net.unc.edu  (152.2.31.1)  0.418 ms  0.355 ms  0.356 ms
 2  unc7600.internet.unc.edu  (128.109.36.254)  0.412 ms  0.495 ms  0.473
ms
 3  rtp7600-gw-to-unc7600-gw.ncren.net  (128.109.70.33)  0.908 ms  0.941
ms  0.849 ms
 4  nlr-atl-to-rtp7600.ncren.net  (128.109.70.106)  10.669 ms  10.381 ms
10.273 ms
 5  hous-atla-70.layer3.nlr.net  (216.24.186.8)  34.444 ms  34.269 ms
34.280 ms
 6  192.124.229.6  (192.124.229.6)  33.767 ms  33.835 ms  33.815 ms
 7  192.124.229.10  (192.124.229.10)  36.995 ms  36.962 ms  37.005 ms
 8  192.124.229.82  (192.124.229.82)  37.149 ms  36.948 ms  37.146 ms
 9  ser9-v703.gw.utexas.edu  (128.83.9.1)  37.112 ms  37.016 ms  37.124 ms
10  128.83.37.42  (128.83.37.42)  37.093 ms  37.113 ms  37.147 ms
11  cs.utexas.edu  (128.83.139.9)  37.390 ms  37.245 ms  37.330 ms
```

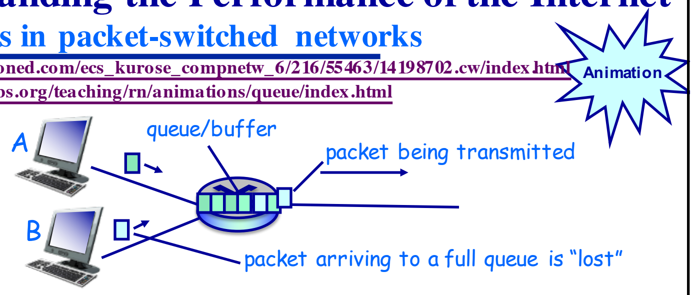
Could be related to priority or a congestion issue

Most probable is that the priority was lower since it was essentially a diagnostic test

# Understanding the Performance of the Internet

## Packet loss in packet-switched networks

https://wps.pearsoned.com/ecs_kurose_compnetw_6/216/55463/14198702.cw/index.html
http://www.ccs-labs.org/teaching/rn/animations/queue/index.html

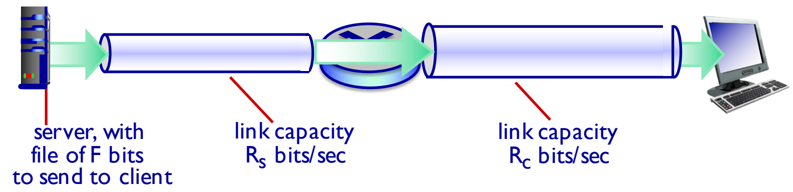



- Queues (*a.k.a.* "buffer") in routers and switches have finite capacity
- Because of "fan-in," packets can arrive faster than they can de transmitted
  - » Queues grow in length when this occurs
- Packets arriving to full queue are "dropped" ("lost")
- Lost packets may be retransmitted by the previous node, by source end system, or not at all



# Understanding the Performance of the Internet

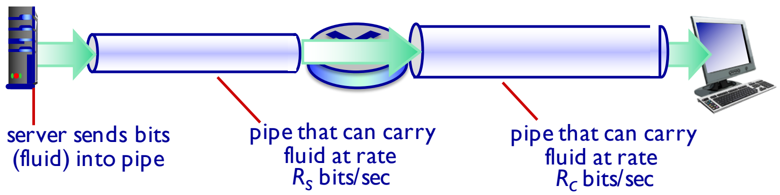
## Throughput in packet-switched networks




- Throughput is the *rate* (bits/time unit) at which bits are transferred between sender and receiver?
  - » Instantaneous throughput: rate measured at a given *point* in time
  - » Average throughput: rate measured over some period of time
    - ❖ The average of a series of measurements

## Understanding the Performance of the Internet

### Throughput in packet-switched networks



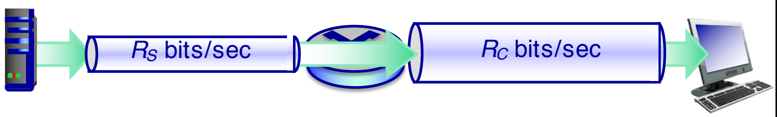

- Throughput is the *rate* (bits/time unit) at which bits are transferred between sender and receiver?
  - » Instantaneous throughput: rate measured at a given *point* in time
  - » Average throughput: rate measured over some period of time
    - ❖ The average of a series of measurements



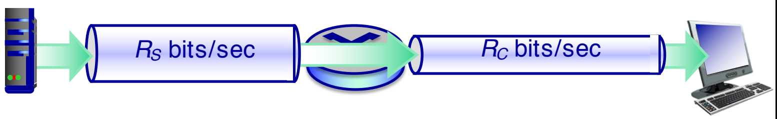
## Understanding the Performance of the Internet

### Throughput in packet-switched networks

- If $R_S < R_C$ then what is the average end-to-end throughput?




- If $R_S > R_C$ then what is the average end-to-end throughput?

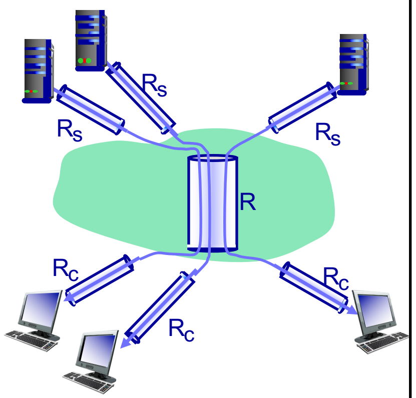



The *bottleneck link* is the link on the end-to-end path that constrains end-to-end throughput

# Understanding the Performance of the Internet

## Throughput on the Internet

- Throughput on the Internet is significantly more complex
- Generally, the per-connection end-to-end throughput is min($R_S$, $R_C$, $R/10$)
  - » In practice either $R_S$ or $R_C$ is the bottleneck and most commonly it's $R_C$



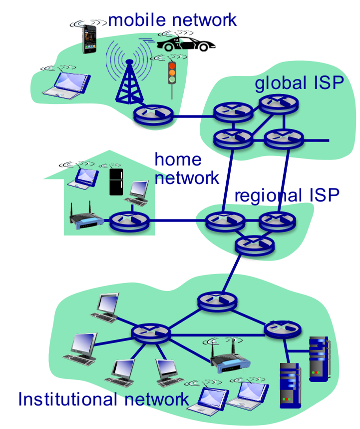

10 connections (fairly) share backbone bottleneck link $R$ bits/sec



# A Whirlwind Introduction to the Internet

## Overview

- What's the Internet
- Network core
- Network edge
- Access nets, physical media
- Internet Structure & ISPs
- Performance: loss, delay
- Security
- Protocol layers, service models



# Network Security

## Sometimes the Internet is not your friend

- 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 largely playing "catch-up"
  - » Efforts are underway to integrate security considerations into all layers of the network
- Issues in 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?



# Network Security

## The malware epidemic


- The Internet delivers malware to our computers on a daily basis
  - » 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: records keystrokes, web sites visited, upload info to collection site
  - » Ransomware: encrypts your disk until you pay a fee to get the decryption key
- Infected host can be enrolled in botnet, used for spam, DDoS attacks
  - » Infected PCs are a commodity: Infected hosts often sold to others!

# Network Security

## The

THE WALL STREET JOURNAL.

TECH

### In the Bitcoin Era, Ransomware Attacks Surge

Digital currency, better software help hackers hold data hostage

By **ROBERT MCMILLAN**

Updated Aug. 19, 2016 11:59 p.m. ET

One evening in April, Dave Winston stood in a convenience store in suburban Charlotte, N.C., uneasily shoving $20 bills into a slim automated-teller machine unlike any he had ever seen. He was buying bitcoin, a digital currency unknown to him a few hours earlier, before hackers took over his computer.

Mr. Winston, crew chief with the Circle Sport-Leavine Family Nascar race team, is among a growing number of victims of a pernicious type of malicious software called ransomware, which has earned millions of dollars for cybercriminals by encrypting computer files and holding them hostage.

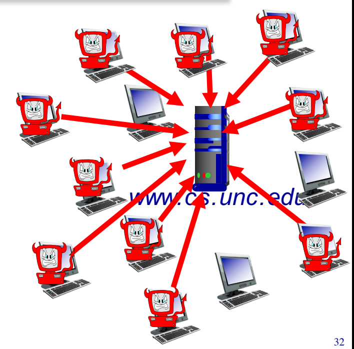


# Network Security

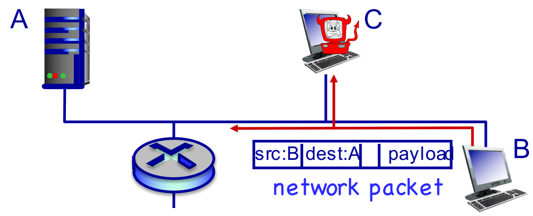
## Attacking the network itself

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

- Taking down X in three easy steps!
  - » Select your target
  - » Break into hosts to create a botnet
  - » Send packets/requests to X from compromised hosts
- Some resource (router buffers, server sockets) is exhausted and legitimate traffic/requests dropped with a high probability



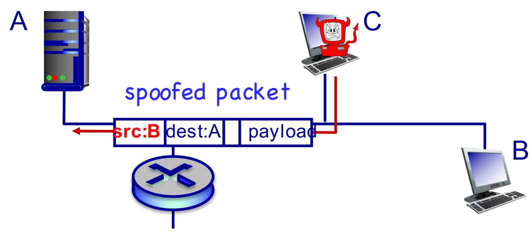
## Network Security
### Compromises based on eavesdropping




- Certain media use a broadcast means to transmit data
  - » "Shared" Ethernet (Ethernet over coax, wireless)
- A network interface in "promiscuous mode" reads/records all packets (*e.g.*, passwords!) passing by
  - » Part of the Ethernet standard!
- Wireshark is a (free) packet-sniffer
  - » We'll use this software used for end-of-chapter labs



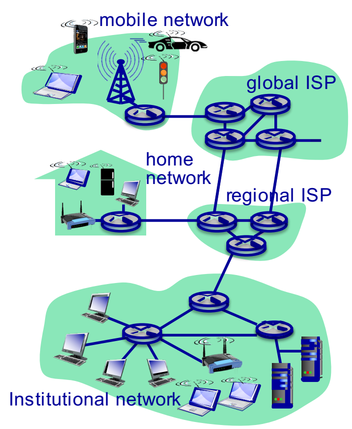
## Network Security
### Impersonation




- IP "spoofing" is used to send packet with a fake source address
  - » At the "packet level," there's no way a receiver can detect fake packets
  - » Higher layer protocols have to deal with this problem

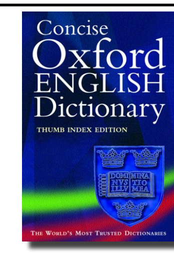
# A Whirlwind Introduction to the Internet
## Overview

- What's the Internet
- Network core
- Network edge
- Access nets, physical media
- Internet Structure & ISPs
- Performance: loss, delay
- Security
- Protocol layers, service models






# The Nuts & Bolts View
## What is a protocol?




Main Entry: pro-to-col

1: An original draft, minute, or record of a document or transaction

2a: A preliminary memorandum often formulated and signed by diplomatic negotiators as a basis for a final convention or treaty

b: The records or minutes of a diplomatic conference or congress that show officially the agreements arrived at by the negotiators

3a: A code prescribing strict adherence to correct etiquette and precedence (as in diplomatic exchange and in the military services)

b: A set of conventions governing the treatment and especially the formatting of data in an electronic communications system

4: A detailed plan of a scientific or medical experiment, treatment, or procedure



IP: Internet Protocol

Protocol: set of conventions governing the treatment and especially the formation of data in an electronic communications system

# The Nuts & Bolts View

## What is a protocol?

- Human protocols:
  - » "Do you have the time?"
  - » "I have a question"
  - » Introductions
- Network protocols:
  - » Machines rather than humans
  - » All communication activity in Internet governed by protocols
- Both:
  - » Specific messages sent
  - » Specific actions taken when messages (or other events) received

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

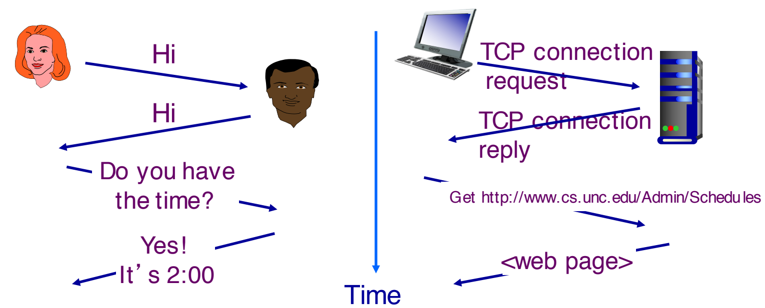


In the Internet it is possible for two machines that have never talked to each other before and need to talk to each other now, these machines need to be able to do that

# What is a protocol?

## A specification for a set of message exchanges

- Example:
  - » Human protocols: Get the time from a stranger
  - » Computer protocols: Get the class time from a web server



# A Whirlwind Introduction to the Internet

## Protocol "Layers"

- Networks are complex!
  - » Composed of many "pieces"
    - ❖ Hosts, routers, links of various media, applications, protocols, hardware, software
- Is there any hope of organizing the structure of the network?
  - » Or at least organizing our discussion of networks?
- Solution!
  - » Decompose functions into a "stack" of function "layers"
  - » Each layer provides well-defined "services" to the layer above it in the stack…
  - » …and uses the services provided by the layer below it
- ***Each layer can treat everything below it in the stack as a "black box"***

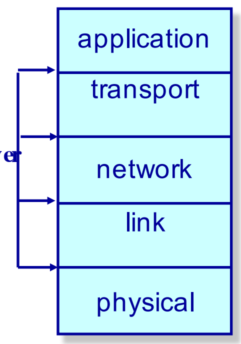


# Protocol Layering in the Internet

## Internet protocol layers ("stack")

- Application layer
  - » Supporting network applications
    - ❖ *ftp*, SMTP, HTTP
- Transport layer
  - » Process-process data transfer
    - ❖ TCP, UDP
- Network layer
  - » Routing of packets from source to destination
    - ❖ IP, routing protocols
- Link layer
  - » Data transfer between directly connected network elements
    - ❖ Ethernet, 802.11, SONET, …
- Physical layer
  - » The insertion of individual bits "on the wire"

**Different services specified at each layer interface**






Layers in the network stack

Highest is application layer and lowest is physical

Applications can use the services from transport layer

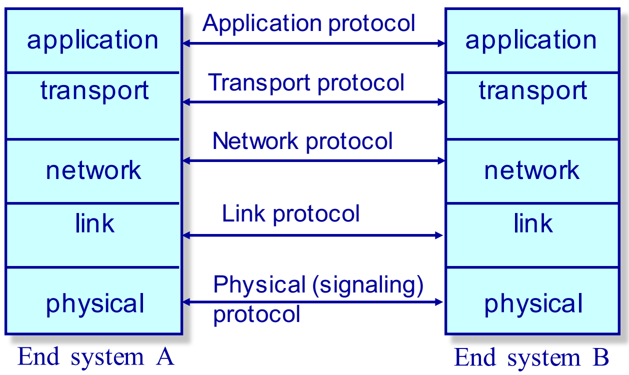
Link layer: transfer from one machine to another
Physical: take 0s and 1s, converts to electromagnetic, and put them on a wire

IP only runs on the network layer

# Protocol Layering in the Internet

## Internet protocol layers ("stack")

Each layer implements a protocol with its peer layer in a distributed system

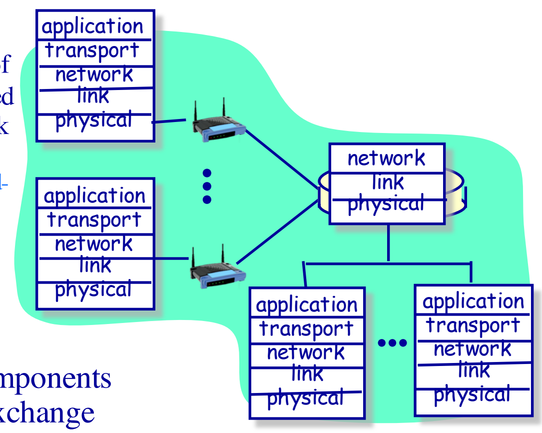





# Protocol Layering in the Internet

## Logical communication

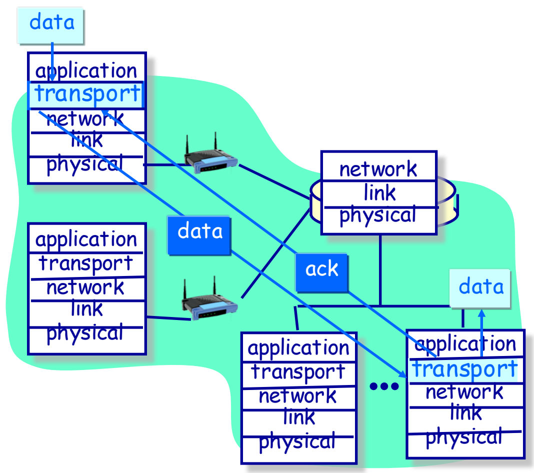
- The implementation of each layer is distributed throughout the network
  - » Some layers just distributed on the end-systems
- The distributed components perform actions, exchange messages with peers



# Logical Communication Example

## The transport layer

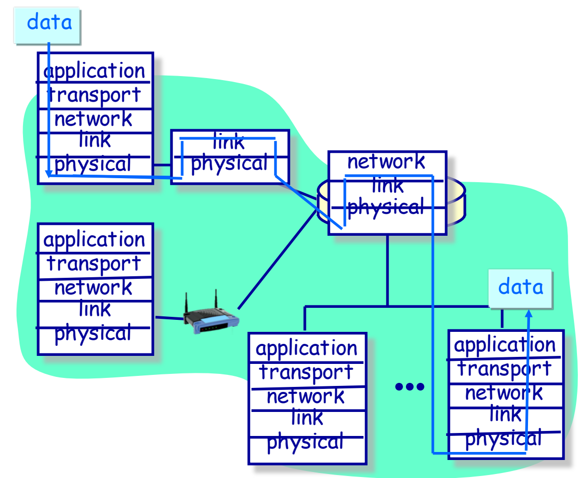
- Receive data from application
- Add transport-layer protocol information
- Send to peer transport layer
- Wait for peer transport layer to respond
- Peer transport delivers data to its application layer






# Protocol Layering in the Internet

## Data flow through protocol layers



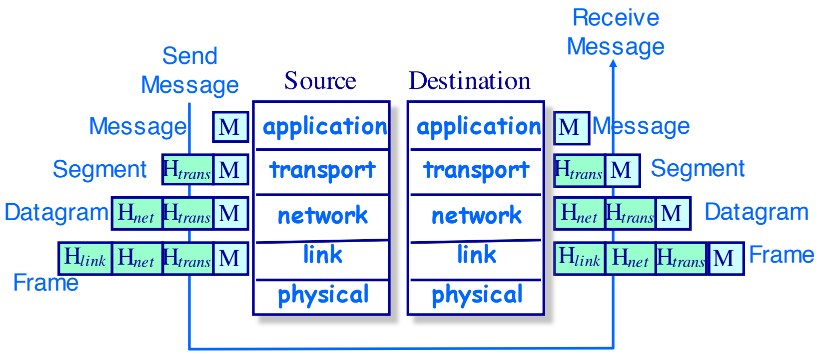



Router is any machine running the network layer

# Protocol Layering in the Internet

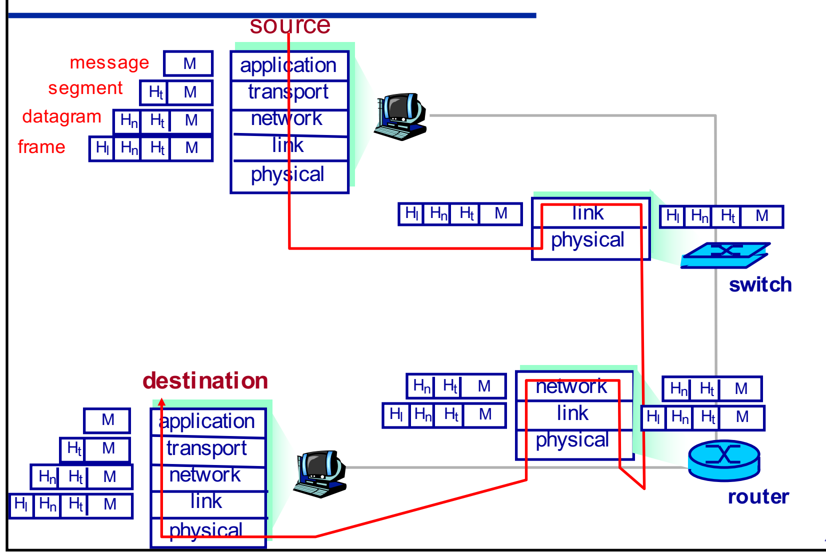
## Protocol layering and data formats

- At sender, each layer takes data from above
  - » Adds header information to create new data unit
  - » Passes new data unit to layer below
- Process reversed at receiver






# Encapsulation Flow in Network Layers



# Protocol Layering in the Internet

## Common logical functions in most layers

- Error control
  - » Make the logical channel between layers reliable (or simply more reliable)
- Flow control
  - » Avoid overwhelming a peer with data
- Segmentation and reassembly
  - » Partitioning large messages into smaller ones at the sender and reassembling them at the receiver
- Multiplexing
  - » Allowing several higher-level sessions to share a single lower-level connection
- Connection setup
  - » Handshaking with a peer

| |
|---|
| application |
| transport |
| network |
| link |
| physical |



Most layers have built in error control

Flow control: sender is asked to slow down so that the receiver is not overwhelmed

# Why Layering?

## Dealing with complex systems

- Explicit structure allows identification, relationship of complex system's pieces
  - » Layered reference model for discussion
- Modularization eases maintenance, updating of system
  - » Change of implementation of layer's service transparent to rest of system
- Layering considered harmful?

  - Functions get duplicated across layers
  - Some optimizations are no longer possible because everything on other layers is hidden

# A Whirlwind Introduction to the Internet

## Summary

- Covered a "ton" of material
  - Internet overview
  - What's a protocol?
  - Network edge, core, access network
  - ISPs
  - Performance: loss, delay
  - Layering and service models
- You now hopefully have:
  - Context, overview, "feel" of networking
  - More depth, detail *later* in course
- Something dangerous to mumble at parties!



End of CH 1