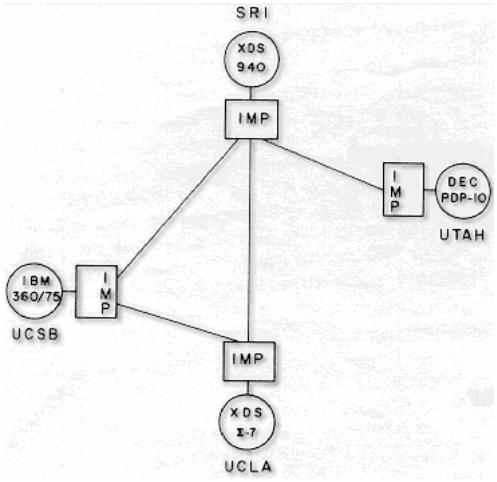
Networking and the Internet

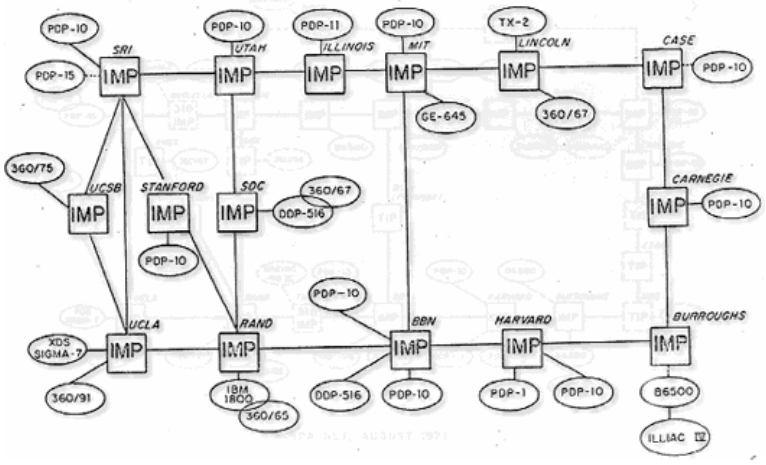
Introduction: "Everything is Connected to Everything"

- Seeds of Networking
  - 1966: ARPA (Advanced Research Projects Agency) State Defense Department's research organization.
    - Focused major development effort on computer networking.
    - ARPA's Goal: To promote research in advanced future technologies by funding university and industry research proposals.
      - Distributed communication system
      - Enable research communication
      - Enable dissimilar computers to share information
      - Reroute information automatically
      - Act as a network of networks; internetworking
  - Result: Thousands of databases became available to researchers

Original ARPANet Sites



ARPANet 1972



The Internet: Struggling to Maturity

- ARPA intended to sell off the ARPAnet to an academic or corporate consortium.
  - Before the sale, federal rules required the Defense Department to determine if ARPAnet was needed for national defense.
  - ARPAnet was transferred to the Defense Communications Agency in 1975.
  - Only about 15 universities were given access to the network.
- 1980: National Science Foundation started CSnet
  - Purpose: To provide a resource-sharing network opportunity to computer science research at all universities.
  - Used TCP/IP protocol.

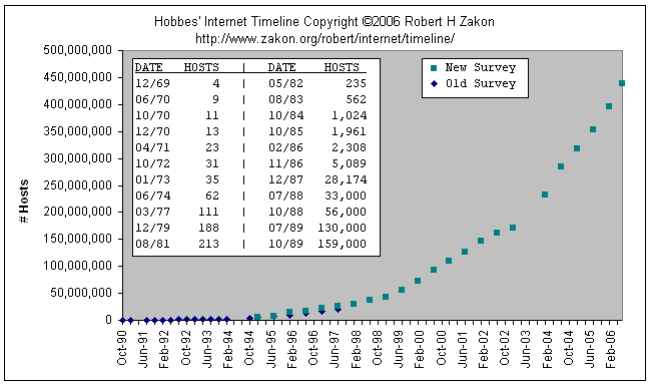
The Internet: Struggling to Maturity

- CSnet fueled interest in creating a more comprehensive network to link all scientific communities (not just CS)
  - NSF couldn't fund such an expensive project.
    - Backbone: NSF built a very fast connection between 5 supercomputing centers linking them all together.
    - Each region surrounding each center would develop its own community network.
    - NSF allowed the regional community networks exclusive use of the backbone.

The Internet: Struggling to Maturity

- 1983: ARPAnet split.
  - Converted to TCP/IP protocol.
  - Part remained ARPAnet: universities, research institutes.
  - Part became Milnet: non-classified military information.
- 1989: majority of ARPAnet switched to NSF's backbone.
  - ARPAnet sites were connected to the NSF backbone through the regional community networks.
- NSFnet became what is known today as the **Internet**.

Growth of the Internet



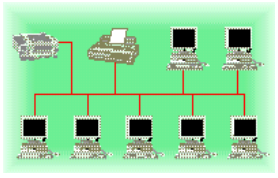
The Physical Organization of Networks

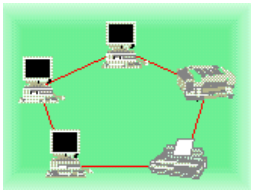
- **LAN** (Local Area Network)
  - A collection of nodes (i.e. computing devices) within a small area.
  - The nodes are linked in a bus, ring, star, tree, or fully connected topology network configuration.
  - Benefits of LANs:
    - Sharing of hardware resources.
    - Sharing of software and data.
    - More efficient person-to-person communication.

The Physical Organization of Networks

- The **bus network** -
  - A continuous coaxial cable to which all the devices are attached.
  - All nodes can detect all messages sent along the bus.
- The **ring network** -
  - Nodes linked together to form a circle.
  - A message sent out from one node is passed along to each node in between until the target node receives the message.

Linking nodes:



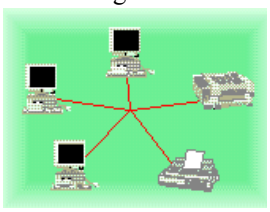


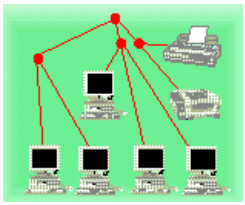
Both of these topologies are uncommon today

The Physical Organization of Networks

- The **star network** -
  - Each node is linked to a central node.
  - All messages are routed through the central node, who delivers it to the proper node.
  - Most common today
- The **tree network** - (**hierarchical network**)
  - Looks like an upside-down tree where end nodes are linked to interior nodes that allow linking through to another end node.

Linking nodes:

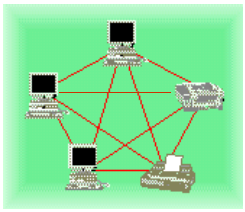




The Physical Organization of Networks

- The **fully connected network** -
  - All nodes are connected to all other nodes.
  - Typically too expensive to implement, need n(n-1)/2 connections for n nodes
- **Internetworking -**
  - Connecting together any number of direct-connected networks.
  - The largest: Internet.
  - Some collection of networks: internet
  - Terms intranet and extranet also used

Linking nodes:



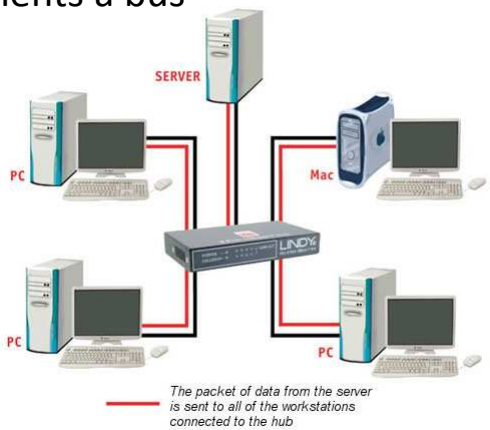
Hardware Architecture of Networks

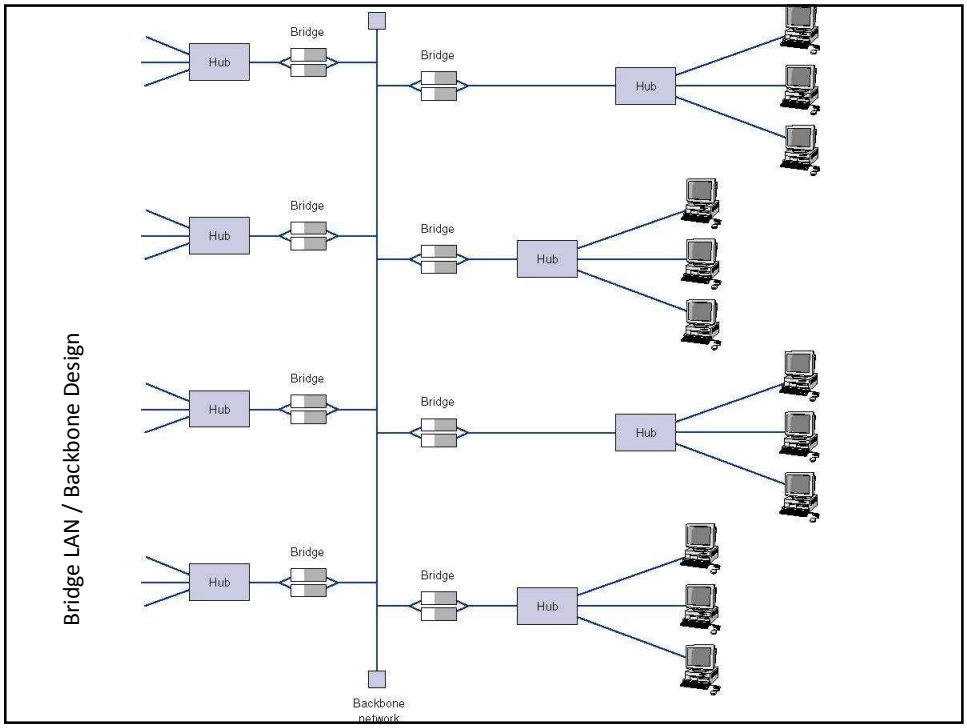
- Types of hardware used to create networks:

Hub	A device that repeats or broadcasts the network stream of information to individual nodes (usually personal computers)
Switch	A device that receives packets from its input link, and then sorts them and transmits them over the proper link that connects to the node addressed.
Bridge	Transparently links two local networks that have identical rules of communication.
Gateway	A link between two different networks that have different rules of communication.
Router	A node that sends network packets in one of many possible directions to get them to their destination.

Hub

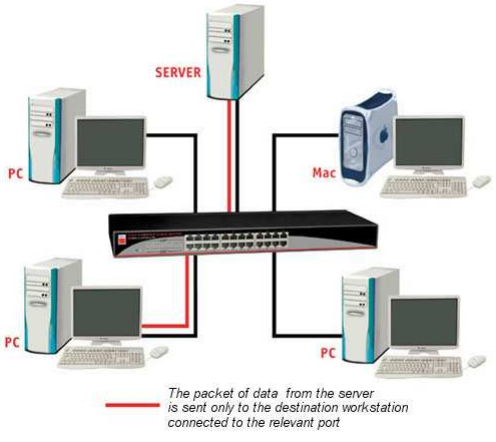
- A hub looks like a star configuration but really implements a bus



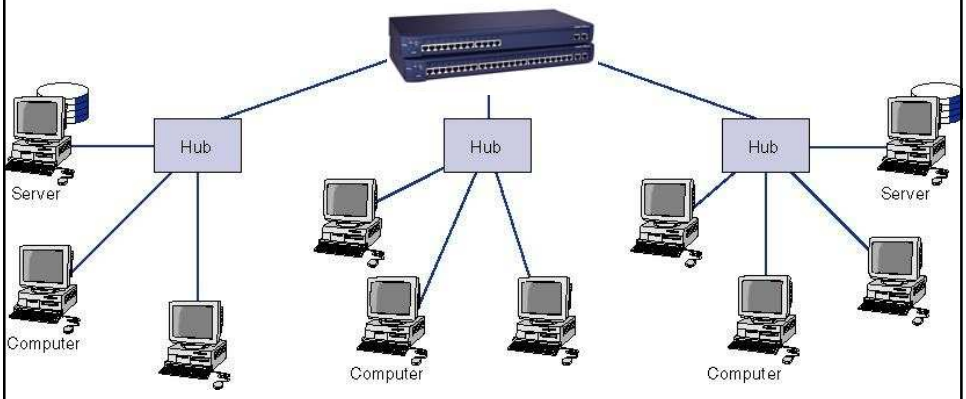


Switch

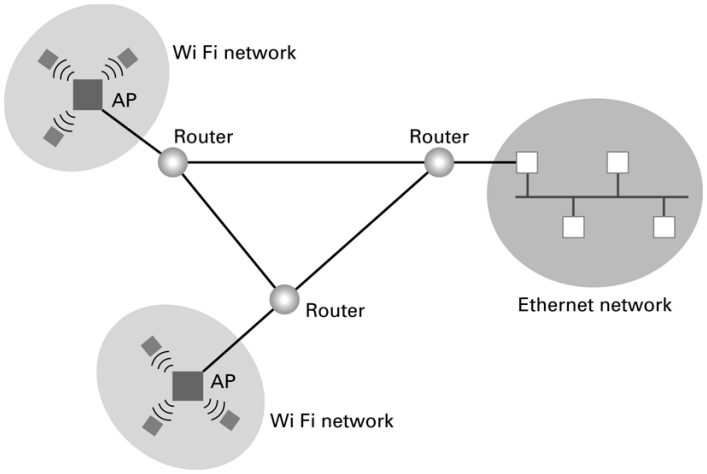
- A switch is a true star configuration
- Consumer devices today are switches, not hubs



Switches



Routers connecting two WiFi networks and an Ethernet network to form an internet



4-18

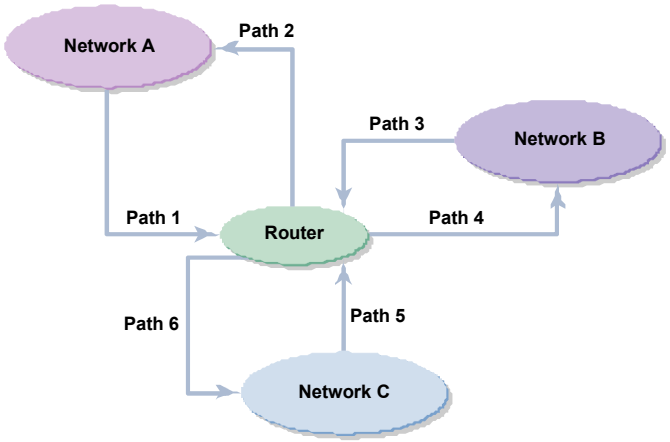
The Physical Organization of Networks

- **MAN** (Metropolitan Area Network)
  - Consists of many local area networks linked together.
  - Span the distance of several few miles.
- **WAN** (Wide Area Network)
  - Consists of a number of computer networks including LANs.
  - Connected by many types of links.

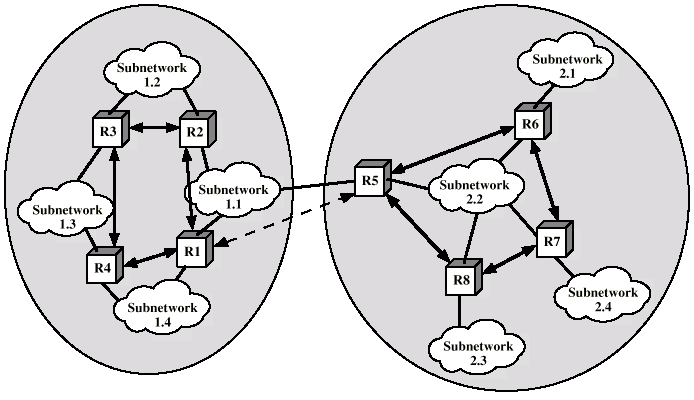
Connecting Independent Networks

- Router: fundamental building block of the Internet
  - Has a processor, memory, and network interface, its own specialized software
  - Connects LANs to backbone WANs
  - Forwards data from one network to another
  - Determines best routes for data to travel

Routers Enable Different Paths between Networks



Routers – Connect Networks



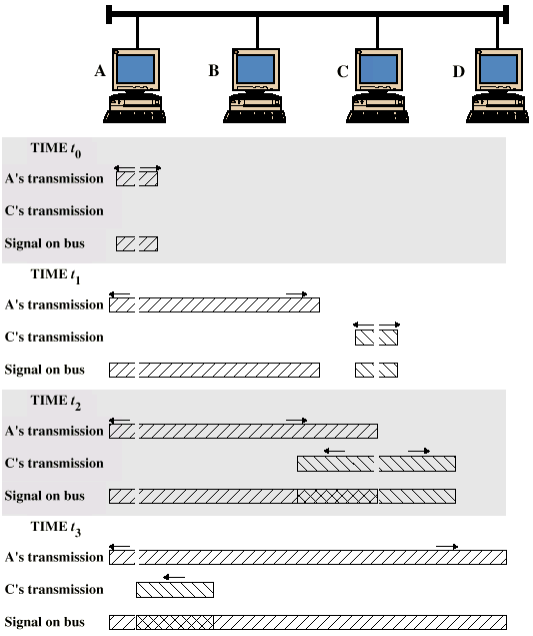
Software Architecture of Networks

- Problem:
  - Connect several different machines running different operating systems (Windows, OS/2, MacOS, UNIX, VMS...)
  - Now, try to: send email, data or files between them.
- Solution:
  - Create a standardized set of rules, or **protocols**, that, when followed, will allow an orderly exchange of information.
  - A collection of these programs is called a **protocol suite**.
    - Must be on all computers or nodes in the network.
    - In order to send data over the network, the necessary programs must be executed.
  - **Network's architecture**: The protocol suite and the general scheme that guides the network's rules.

Software Architecture of Networks

- Problem: Collisions of information are caused by two computers simultaneously attempting to send information to the network.
- Solution: Different networks have different protocol suites:
  - Apple Computer's **LocalTalk** Protocol - Permission must be granted before information can be sent along the network.
  - **Token-Ring** Protocol (IBM and others) - A token is "picked up" by a node signifying that a message is about to be sent, the computer sends the message, then, replaces the token so that others can use the network.
  - **Ethernet** Protocol (Xerox and others) – CSMA/CD
    - Carrier Sense Multiple Access / Collision Detection
    - "Listen" for quiet line; then send message
    - Collision occurs with simultaneous messages
    - If detected, send jam signal, wait random time, and resend
    - Distributed Fairness

CSMA/CD Operation



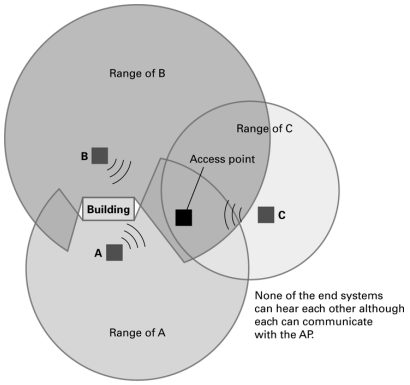
Frames must be long enough to detect collision during transmission!

Wireless LANs

- IEEE 802.11
- Basic service set (cell)
  - One or more stations using same protocol, together with an **access point** or **base station**
  - Typically some radius (~200 ft) from the access point
  - Competing to access shared medium
  - May connect to backbone via access point (bridge)
- Extended service set
  - Two or more BSS connected by distributed system
  - Appears as single logic LAN to Data Link layer

Wireless – 802.11

- CSMA/CD approach doesn't work because of the Hidden Terminal Problem



802.11 Media Access

- If media is idle for some time then it is allowed to transmit
- If media is busy
  - Wait until media is free
  - Then wait an additional random time (less greedy than 802.3)
- Data is sent with a checksum
  - A checksum is a simple check to see if the data was received properly
  - E.g. if sending the numbers 10, 20, and 30 then the checksum might be the sum of all numbers (60), so 10,20,30 and 60 are sent. The receiver verifies that all numbers received add up to 60.
- If the checksum matches, the receiver sends an ACK to the sender
- Receiver might get a collision, but no detection for one – instead the checksum will (hopefully) fail
  - If sender doesn't receive ACK in some time period, resends

802.11 Media Access

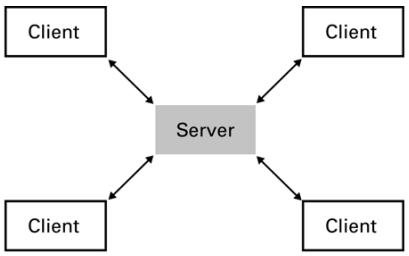
- Also supports a way to reserve channel access, avoids hidden station problem
- Sender sends short Request To Send (RTS)
- Receiver sends short Clear To Send (CTS) with permission to the Sender
  - All other stations also hear the CTS, and have to withhold sending until transmission is complete
- Sender sends data
- Recipient sends ACK if successful
  - Other stations also hear the ACK, can now issue their own RTS
- Might have collisions with RTS or CTS but not with Data or ACK. If collisions, stations requesting to send will not get a CTS and will have to wait random time to re-submit the RTS

Inter-Program Communication

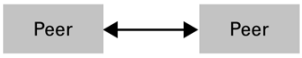
- Client-server
  - One server, many clients
  - Server must execute continuously
  - Client initiates communication
  - E.g. web browser asks a web server for a web page
- Peer-to-peer (P2P)
  - Two processes communicating as equals
  - Peer processes can be short-lived
  - E.g. copying music or video directly from another computer
    - Legal issues

4-30

The client/server model compared to the peer-to-peer model



a. Server must be prepared to serve multiple clients at any time.



b. Peers communicate as equals on a one-to-one basis.

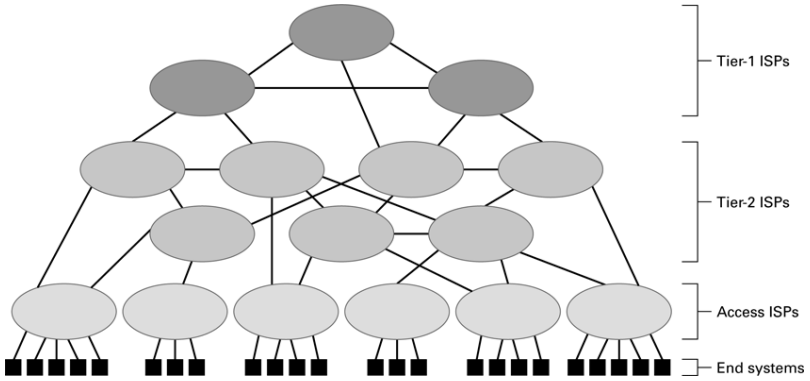
4-31

Internet Architecture

- Internet Service Provider (ISP)
  - Tier-1
    - Major communication companies
  - Tier-2
- Access ISP: Provides connectivity to the Internet
  - Traditional telephone (dial up connection)
  - Cable connections
  - DSL
  - Wireless

4-32

Internet Composition



4-33

Internet Addressing

- IP address: pattern of 32 or 128 bits often represented in dotted decimal notation
  - IP 4: 32 bits
    - Four groups of 8 bits; each 8 bit group can be represented by a value from 0 to 255
    - 0.0.0.0 to 255.255.255.255
    - 2^{32} or about 4.3 billion addresses theoretically possible, but not divided equally; exhaustion predicted 2010-2012
  - IP 6: 128 bits
  - Addresses often assigned to clients via DHCP – Dynamic Host Configuration Protocol
- Domain name system (DNS)
  - Provides a way to assign a more meaningful name to the IP address
  - Name servers
  - DNS lookup

4-34

Internet Corporation for Assigned Names & Numbers (ICANN)

- Allocates IP addresses to ISPs who then assign those addresses within their regions.
- Oversees the registration of domains and domain names.
- Names of computers use the following convention for domains
  - hostname.subdomain.domain.type
    - type : Organization status
    - domain : Registered ICANN name
    - Subdomain : optional hostname
    - Hostname : Name of the machine
  - Organizations determine own structure at the hostname and subdomain levels

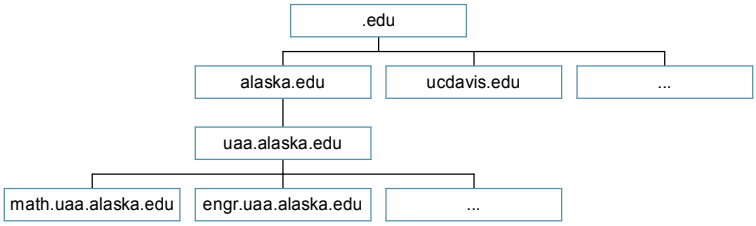
4-35

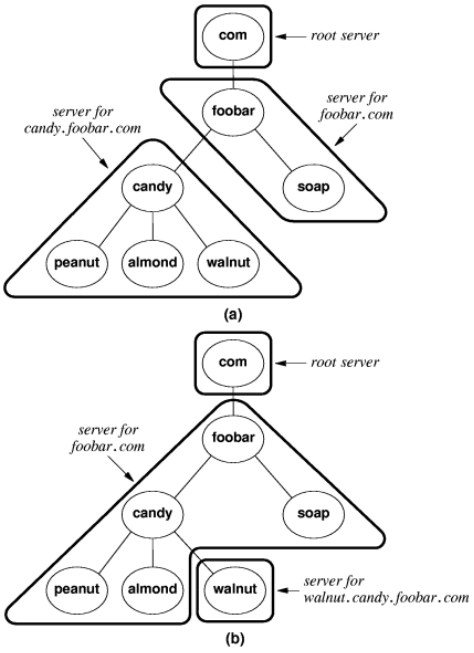
Common Types

- .edu - Educational
- .gov - Government
- .org - Nonprofit organizations
- .net - Internet service providers, backbone
- .country - Two letter country code, e.g. .jp for Japan, .us for United States, .to for Togo
- Latest top level domains approved:
  - .aero, .biz, .museum, .info, .jobs, .travel
  - .xxx rejected in 2006

Hierarchy of Names - DNS

- Routers are based on IP address (e.g. 134.114.140.34), not the English-like domain name!
- DNS = Domain Name Server system
  - Way to translate from domain name to IP address
- Tree-based hierarchy, with .org, .com, .edu, and .gov at the top of the tree





DNS Client-Server

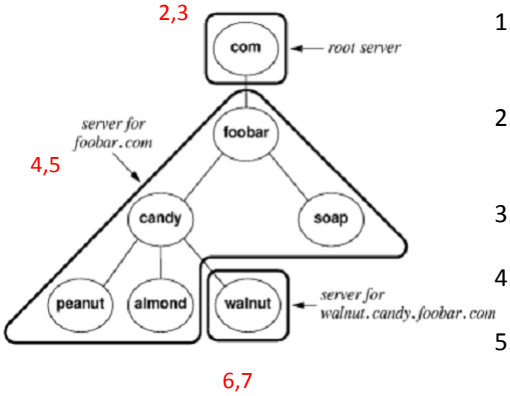
- DNS names are managed by a hierarchy of DNS servers
- Hierarchy is related to DNS domain hierarchy
- Root server at top of tree knows about next level servers
  - Next level servers, in turn, know about lower level servers

DNS Servers

- Each DNS server is the *authoritative server* for the names it manages
- If request contains name managed by receiving server, that server replies directly
- Otherwise, request must be forwarded to the appropriate authoritative server

- Process:
  - Client contacts local DNS server, L
  - If L knows the requested IP or is the authority, return the IP
  - Otherwise, contact the root server
    - Root server returns to L the authoritative server for the domain
    - L contacts this server
    - Process may repeat until we find the authoritative server

DNS Lookup Example



1. Computer requests IP for comp.walnut.candy.foobar.com from local DNS
2. Not found in local DNS, local DNS becomes a client and contacts root server
3. Root server returns server below, for foobar.com
4. Local DNS contacts server at foobar.com
5. Foobar.com server returns server below, for walnut.foobar.com
6. Local DNS contacts server at walnut.foobar.com
7. This is the authority, returns IP
8. Local DNS returns IP to Computer

Iterative Resolution

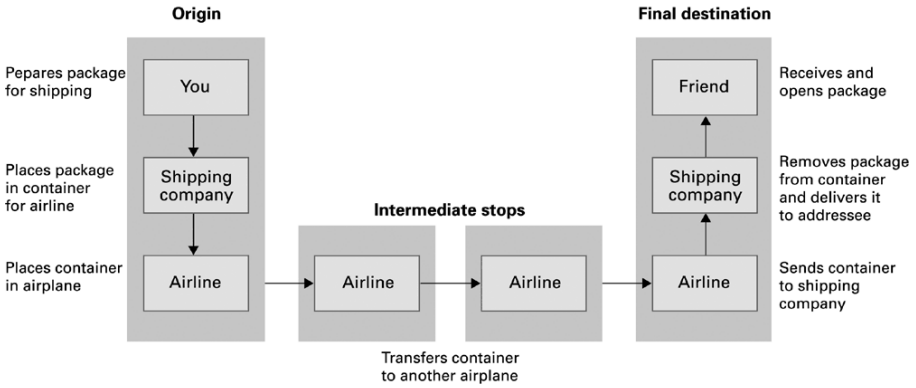
DNS Efficiencies

- DNS resolution can be **very** inefficient
  - Every host referenced by name triggers a DNS request
  - Every DNS request for the address of a host in a different organization goes through the root server
- Servers and hosts use *caching* to reduce the number of DNS requests
  - Cache is a list of recently resolved names and IP addresses
- Servers use *replication* to decrease the load on root servers

Protocol of the Internet: TCP/IP

- Data is sent on the Internet using TCP/IP
- Transmission Control Protocol
  - Breaks information into data packets
  - Reassembles packets when received
  - Checks for lost packets
- Internet Protocol
  - Addressing using IP addresses
- First, an analogy with shipping packages

Package-shipping example

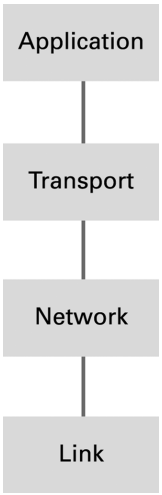


4-43

High Level Internet Architecture

- Different **Layers** in the "Stack"
  - Application
    - Programs like web browsers, servers,
    - Protocols such as ftp, ssh, httpd, html
  - Transport
    - Keeps track of a session, breaks data into packets, error checking
    - Protocols such as TCP or UDP
  - Internet
    - Routes messages to their destination
    - Protocols such as IP
  - Data Access or Data Link
    - Controls physical hardware
    - Protocols such as Ethernet, PPP
  - Physical
    - Physical medium, twisted pair, fiber, radio, etc.

Abstract Internet software layers



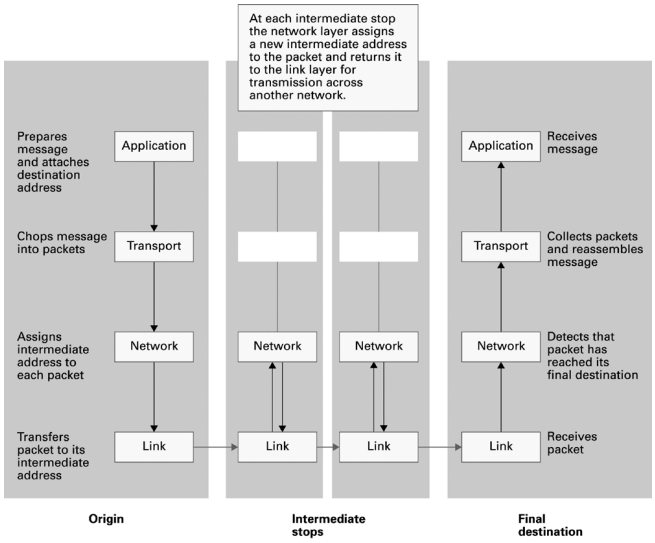
4-45

Basics of Internet Communication

At the Transport Layer, messages broken into *datagrams*, also known as *packets*, and addressed at the IP layer:

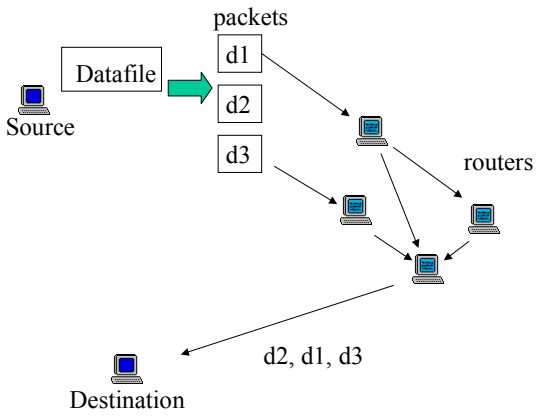
Sequence Number	34
Source Address	128.120.56.214
Destination Address	199.237.80.7
Data Size	7
(Other fields)	…
Data	hi mom

Following a message through the Internet



4-47

Packet Routing



Packets may travel different routes to reach their destination
Sequence number used to put data back in the original order

TCP/IP Protocol Suite

- Transport Layer
  - TCP
    - Transmission Control Protocol
    - Reliable but overhead to get the reliability
    - Used with email, retrieving web pages, etc.
  - UDP
    - User Datagram Protocol
    - Unreliable but less overhead / more efficient
    - Used for streaming video/audio, DNS requests
- Network Layer
  - IP (IPv4 and IPv6)

4-49

Network Example: Traceroute

- Traceroute: A program that allows the the tracing of packets over the Internet or any network using TCP/IP protocol.
  - Uses a special number - TTL (Time to Live) - contained in a place at the beginning of each packet sent over the network.
    - The number is originally set to 255.
    - Each time it is received by a router, it decrements by 1.
    - If the TTL number becomes 0 before reaching its destination, the router where this happened sends back an error message (time exceeded) with the address of the router.
      - Stops messages from circulating forever.

Tracing the router hops…

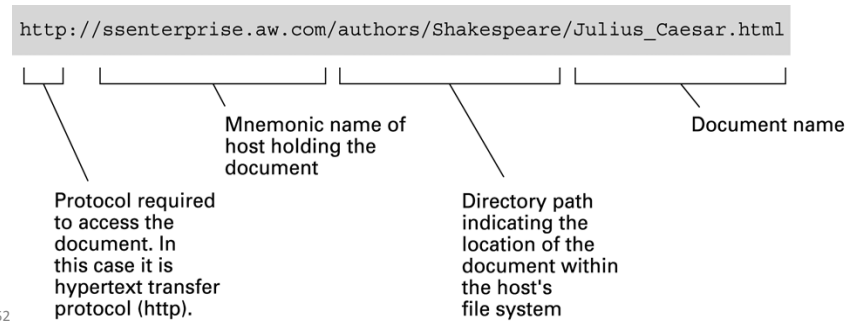
- On Windows: tracert <dest> This is an OLD traceroute:

```
Tracing route to www.alaska.net [209.112.131.196] over a maximum of 30 hops:
  1  431 ms  440 ms  421 ms  uaa-du-02.alaska.edu [137.229.98.66]
  2  350 ms  341 ms  300 ms  r98-99-e1.alaska.edu [137.229.98.99]
  3  361 ms  340 ms  381 ms  swf-7507-1 [137.229.254.21]
  4  300 ms  341 ms  300 ms  m40 [137.229.2.1]
  5  290 ms  321 ms  320 ms  uacore1-ge-0-0-0-0.pnw-gigapop.net [198.32.40.129]
  6  411 ms  401 ms  390 ms  westincore1-so-0-1-0-0.pnw-gigapop.net [198.48.91.33]
  7  431 ms  380 ms  441 ms  westinnsp2-GE1-0.pnw-gigapop.net [198.32.170.17]
  8  631 ms  440 ms  461 ms  p1-1-2-2.a07.sttlwa01.us.ra.verio.net [204.203.3.1]
  9  591 ms  640 ms  461 ms  ge-6-2-0.r03.sttlwa01.us.bb.verio.net [129.250.28.1]
 10  521 ms  500 ms  421 ms  p4-5-0-0.r06.plalca01.us.bb.verio.net [129.250.3.89]
 11  461 ms  450 ms  511 ms  p4-6-0-0.r01.snjsca03.us.bb.verio.net [129.250.2.198]
 12  621 ms  681 ms  620 ms  p4-1-0-0.r00.lsanca01.us.bb.verio.net [129.250.2.114]
 13  611 ms  661 ms  621 ms  p1.att.r00.lsanca01.us.bb.verio.net [129.250.9.34]
 14  601 ms  541 ms  531 ms  gbr3-p50.la2ca.ip.att.net [12.123.28.130]
 15  481 ms  421 ms  560 ms  gbr4-p20.sffca.ip.att.net [12.122.2.69]
 16  371 ms  420 ms  401 ms  gbr3-p30.st6wa.ip.att.net [12.122.2.198]
 17  400 ms  481 ms  481 ms  gbr2-p10.st6wa.ip.att.net [12.122.5.166]
 18  421 ms  421 ms  410 ms  gar1-p370.st6wa.ip.att.net [12.123.44.62]
 19  521 ms  561 ms  540 ms  12.123.203.1
 20  440 ms  441 ms  501 ms  12.124.174.6
 21  561 ms  440 ms  461 ms  www.alaska.net [209.112.131.196]
```

21 hops from Anchorage to Anchorage!

World Wide Web

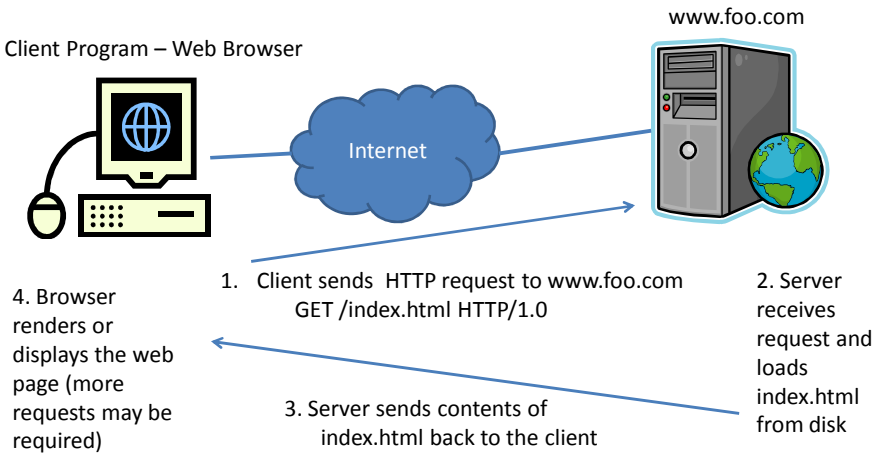
- Example Internet Program: Web Browser
- Hypertext and HTTP
- Browser gets documents from Web server
- Documents identified by URLs



4-52

Web Browser Request

Client navigates to http://www.foo.com/index.html



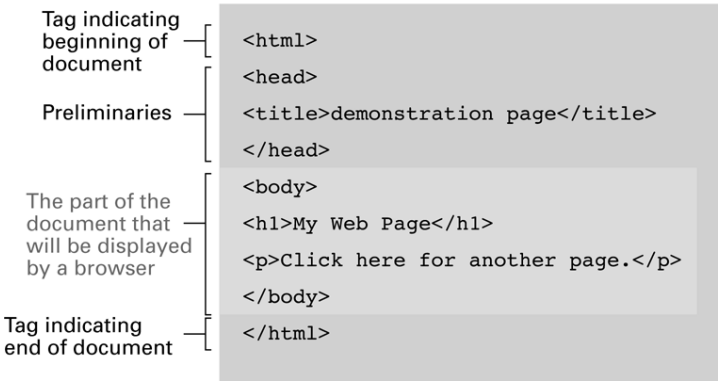
Hypertext Document Format

- Encoded as text file
- Contains tags to communicate with browser
  - Appearance
    - <h1> to start a level one heading
    - <p> to start a new paragraph
  - Links to other documents and content
    - Click Me
  - Insert images
    - <img src = "mypic.jpg">

4-54

A simple Web page

a. The page encoded using HTML.



4-55

A simple Web page (continued)

b. The page as it would appear on a computer screen.

4-56

An enhanced simple Web page

a. The page encoded using HTML.

```
<html>
<head>
<title>demonstration page</title>
</head>
<body>
<h1>My Web Page</h1>
<p>Click
   <a href="http://crafty.com/demo.html">
   here
   </a>
   for another page.</p>
</body>
</html>
```

Anchor tag containing parameter

Closing anchor tag

4-57

An enhanced simple Web page (continued)

b. The page as it would appear on a computer screen.

4-58

Programs can run on the Client Side or the Server Side

- Client-side activities
  - Examples: java applets, javascript, Macromedia Flash
- Server-side activities
  - Common Gateway Interface (CGI)
  - Servlets
  - PHP

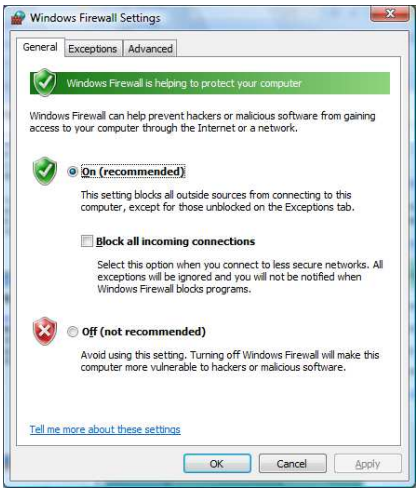
4-59

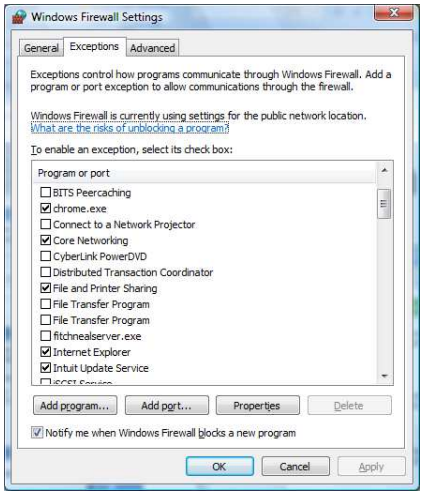
Security

- Attacks
  - Malware (viruses, worms, Trojan horses, spyware, phishing software)
    - Example: Erin Andrews peephole video http://www.youtube.com/watch?v=H4qbLKy32rI
  - Denial of service
  - Spam
- Protection
  - Firewalls
    - Can operate at the IP or Application Layers
  - Spam filters
  - Antivirus software

4-60

Example: Windows Firewall





Encryption

- Data sent over a network may be vulnerable to eavesdropping
  - All clients on a bus or wireless coverage receive broadcast data
  - Routers along the path could piece together individual packets
  - Routers along the path could change or mess with data
- FTPS, HTTPS, SSL
- Public-key Encryption
  - Public key: Used to encrypt messages
  - Private key: Used to decrypt messages
- Certificates and Digital Signatures

Issues

- Privacy
  - Message is secret
- Authentication
  - Recipient knows the message is not a forgery
- Integrity
  - Message was not tampered with in transit
- Nonrepudiation
  - Author can't later deny sending the message

Simple Encryption

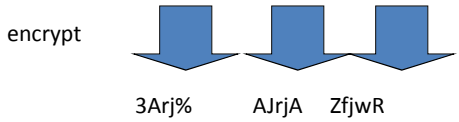
- Substitution cipher, assign different letter to each letter; Y="A", E="Z", S="B" so to encrypt "YES" this becomes "AZB"
  - Both sender and receiver share a **secret key** – the proper substitutions
- Can use statistical properties to help deduce guesses
- Fairly easy to break using brute force of the computer to try all possible assignments

Block Cipher

- Takes a fixed-length block of plaintext, perhaps 64 bits, and encrypts it

Plaintext: ATTACK AT DAWN

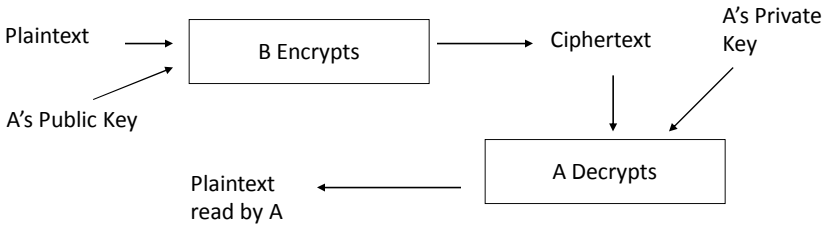
Using blocks of 4 chars: ATTA, CK A, TDA,WN



Each block is usually treated as a number.
Most schemes use block ciphers.
There are some modifications to prevent repeating blocks so someone couldn't insert them and confuse the data.

Public Key Crypto

- Each participant gets two keys
  - Public key is made available to anyone
  - Private key is kept secret
- Like a safe with a slot in the top – anyone can put information in, but only the person with the combo can get it out



Authentication

- Problem with the previous – anyone could have sent the message!
- Solution : double encryption. Design the public and private keys so that they can encrypt or decrypt each other:
  - M = Plaintext Message
  - P = Public Key
  - S = Secret Key
  - Then: M = P(S(M)) and M=S(P(M))
- That is, if we encrypt with the public key, we can decrypt it with the private key. Similarly if we encrypt with the private key, we can decrypt it with the public key.

Crypto Example

- Bob encrypts the message M using his private key to get C1
- Bob encrypts C1 using Alice's public key to get C2 and sends it to Alice
- Alice decrypts C2 using her private key to get C1
- Alice decrypts C1 using Bob's public key
  - Bob sends: AlicePublic(BobPrivate(M))
  - Alice decrypts: BobPublic(AlicePrivate(M))
- If this all works, only Alice can read M and only Bob could have sent it! (idea of digital signature)
- How is this done? Lots of math and number theory; RSA uses the idea that it is computationally hard to factor large numbers.

Summary

- Network Fundamentals
- The Internet
- The World Wide Web
- Internet Protocols
- Security