Chapter 1: Introduction

**Overview:**
- What's the Internet?
- What's 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

**Goal:**
- Get "feel" and terminology
- More depth, detail later in course
- Approach: use Internet as example

Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
- end systems, access networks, links
1.3 Network core
- circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

What's the Internet: "Nuts and Bolts" view

- Millions of connected computing devices: hosts = end systems
- running network apps
- Communication links
- Fiber, copper, radio, satellite
- Transmission rate = bandwidth
- Routers forward packets (chunks of data)

"Cool" Internet Appliances

- IP picture frame
  - http://www.cavia.com/
- World's smallest Web server
- Web-enabled toaster + weather forecaster
- Internet phone
What's the Internet: "nuts and bolts" view

- Protocols control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, Skype, Ethernet
- Internet: "network of networks"
  - loosely hierarchical
- Internet standards
  - Developed by IETF: Internet Engineering Task Force
  - IETF produces RFCs: Request for comments
  - More than 5000
  - IEEE for links (e.g., 802.11)

What's the Internet: a Service View

- Communication infrastructure enables distributed applications:
  - Web, VoIP, email, games, e-commerce, file sharing
- Communication services provided to apps:
  - Reliable data delivery from source to destination
  - "best effort" (unreliable) data delivery

What's a Protocol?

Human protocols:
- "What's the time?"
  - Say "yes" first
- "I have a question"
  - Raise hand first

Network protocols:
- Machines rather than humans
- All communication activity in Internet governed by protocols

Protocols: 1) define format, 2) order of msgs sent and received among network entities, and 3) actions taken on msg transmission, receipt

Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
  - end systems, access networks, links
1.3 Network core
  - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

A Closer Look at Network Structure

- Network edge: applications and hosts
- Access networks, physical media: wired, wireless communication links
- Network core
  - interconnected routers
  - network of networks
The Network Edge

- End systems (hosts):
  - run application programs
  - e.g. Web, email
  - at "edge of network"

- Client/server model
  - client host requests, receives service from always-on server
  - e.g. Web browser/server, email client/server
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent

- Peer-peer model:
  - no "edge of network"

Client/Server Model

- Client host requests, receives service from always-on server
- e.g. Web browser/server, email client/server

- Peer-peer model:
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent

Access Networks and Physical Media

Q: How to connect end systems to edge router?

- residential access nets
- institutional access networks (school, company)
- mobile access networks

Keep in mind:
- bandwidth (bits per second) of access network?
- shared or dedicated?

Dial-up Modem

- Uses existing telephony infrastructure
- Home is connected to central office
- up to 56Kbps direct access to router (often less)
- Can’t surf and phone at same time: not “always on”

Digital Subscriber Line (DSL)

- Also uses existing telephone infrastructure
- up to 1 Mbps upstream (today typically < 256 kbps)
- up to 8 Mbps downstream (today typically < 1 Mbps)
- dedicated physical line to telephone central office

Cable Modems

- Does not use telephone infrastructure
- HFC: hybrid fiber coax
  - asymmetric: up to 30 Mbps downstream, 2 Mbps upstream
- Network of cable and fiber attaches homes to ISP router
- Homes share access to router (500 to 5,000 homes)
- Unlike DSL, which has dedicated access

Cable Network Architecture: Overview

Typically 500 to 5,000 homes
Cable Network Architecture: Overview

FDM (more shortly):

- Optical links from central office to the home
- Much higher Internet rates; fiber also carries television and phone services

Ethernet Internet access

- Typically used in companies, universities, etc
- 10 Mbps, 100 Mbps, 1 Gbps, 10 Gbps Ethernet
- Today, end-systems typically connect into Ethernet switch

Wireless Access Networks

- **Shared wireless access**
  - network connects end system to router
  - via base station, aka "access point" (AP)

- **Wireless LANs:**
  - 802.11b/g (WiFi): 11 or 54 Mbps

- **Wider-area wireless access**
  - Provided by telco operator
  - ~1 Mbps over cell phone system (EVDO, HSDPA)
  - next up (?): WiMAX (10's Mbps) over wide area
Home Networks

Typical home network components:
- DSL or Cable modem
- Router/firewall/NAT
- Ethernet
- Wireless access point

Physical Media: Twisted Pair

- Bit: propagates between transmitter/rcvr pairs
- Physical link: what lies between transmitter & receiver
- Guided media:
  - signals propagate in solid media: copper, fiber, coax
- Unguided media:
  - signals propagate freely, e.g., radio

Physical Media: Coax, Fiber

Coaxial cable:
- Two concentric copper conductors
- Bidirectional
- Baseband:
  - single channel on cable
  - legacy Ethernet
- Broadband:
  - multiple channels on cable
  - HFC

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

Physical Media: Radio

Radio link types:
- terrestrial microwave
  - up to 45 Mbps channels
- LAN (e.g., Wifi)
  - 802.11b: 11 Mbps
  - 802.11g: 54 Mbps
- Wide-area (e.g., cellular)
  - 3G cellular: ~ 1 Mbps
- Satellite
  - Kbps to 45 Mbps channel (or multiple smaller channels)
  - 270 msec end-end delay

Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
  - end systems, access networks, links
1.3 Network core
  - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

The Network Core

- Mesh of interconnected routers
- The fundamental question: how is data transferred through net?
  - circuit switching:
    - dedicated circuit per call
    - e.g., telephone
  - packet-switching:
    - data sent thru net in discrete "chunks"
    - e.g., postal mail
- human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?
Network Core: Circuit Switching

End-end resources reserved for "call"
- link bandwidth, switch capacity
- dedicated resources: no sharing
- circuit-like (guaranteed) performance
- call setup required

- network resources (e.g., bandwidth) divided into "pieces"
- pieces allocated to calls
- resource piece idle if not used by owning call (no sharing)

- dividing link bandwidth into "pieces"
- frequency division multiplexing (FDM)
- time division multiplexing (TDM)

Circuit Switching: FDM and TDM

Example:
- 4 users

FDM

frequency

TDM

frequency

Numerical Example

- How long does it take to send a file of 80 Kbytes from host A to host B over a circuit-switched network?
  - All links are 1.536 Mbps
  - Each link uses TDM with 24 slots/sec
  - 500 msec to establish end-to-end circuit

Let's work it out!

Numerical Example: Solution

- 80 Kbytes is 640,000 Kbits
  - NOTE: networks in bits, end systems in bytes
  - NOTE: 1 Kbyte = 1024 bytes, 1Kbit = 1000 bits
- Each circuit has a rate of 1.536 / 24 = 64 Kbps
- So, it takes 640,000 bits / 64 Kbps = 10 seconds to transmit the file
- Need to add the circuit establishment time (½ second)
- So, 10.5 seconds

Network Core: Packet Switching

Each end-end data stream divided into packets
- user A, B packets share network resources
- each packet uses full link bandwidth
- resources used as needed

Resource contention:
- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - Node receives complete packet before forwarding
Packet Switching: Statistical Multiplexing

- 100 Mb/s Ethernet
- Packet switching can allow more users to use network!
  - 1 Mb/s link
  - Each user:
    - 100 kb/s when "active"
    - Active 10% of time
  - Circuit-switching:
    - 10 users
  - Packet switching:
    - With 35 users, probability 10+ active at same time is less than 0.0004

Packet Switching versus Circuit Switching

Packet switching can give individual users better performance!

- Consider:
  - 3 users,
  - TDM with 1000 bit slot, 1 slot per 10 msec
- Users quiet, then one user 1000 1kbit packets
- With TDM, will take 10 seconds to transmit
- With packet switch, user can take all (1 Mbps) and transmit in about 1 second

Packet Switching versus Circuit Switching

Is packet switching a "slam dunk" winner?

- Great for bursty data
  - Resource sharing
  - Simpler, no call setup
- But... can have excessive congestion: packet delay and loss
  - Protocols needed for reliable data transfer, congestion control
- Q: How to provide circuit-like behavior?
  - Bandwidth guarantees needed for audio/video apps
  - Still an unsolved problem (chapter 7)

Packet-switching: Store-and-Forward

- Takes L/R seconds to transmit (push out) packet of L bits on to link at R bps
- Sequence of A & B packets does not have fixed pattern, bandwidth shared on demand → statistical multiplexing.

Example:

- 3 hops (end plus 2 routers)
- L = 7.5 Mbites
- R = 1.5 Mbps
- Transmission delay = 3 * 7.5 / 1.5 = 15 sec

Packet Switching versus Circuit Switching

- Consider:
  - 3 users,
  - TDM with 1000 bit slot, 1 slot per 10 msec
- Users quiet, then one user 1000 1kbit packets
- With TDM, will take 10 seconds to transmit
- With packet switch, user can take all (1 Mbps) and transmit in about 1 second

Internet structure: Network of Networks

- Roughly hierarchical
- At center: "Tier-1" ISPs (e.g., Verizon, Sprint, AT&T, Cable and Wireless), national/international coverage
  - Treat each other as equals

Tier-1 providers interconnect (peer) privately
Tier-1 ISP: e.g., Sprint

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

  Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet
  Tier-2 ISP is customer of Tier-1 provider

  Tier-2 ISPs also peer privately with each other.

Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
  - end systems, access networks, links
1.3 Network core
  - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

How do Loss and Delay Occur?

Packets queue in router buffers
- packet arrival rate to link exceeds output link capacity
- packets queue, wait for turn (delay)
- Queue is full, packets are dropped (loss)

packet being transmitted (delay)
packets queueing (delay)
free (available) buffers: arriving packets dropped (loss) if no free buffers
Four Sources of Packet Delay

1. Nodal processing:
   - check bit errors
   - determine output link

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

3. Transmission delay:
   - $R$ = link bandwidth (bps)
   - $L$ = packet length (bits)
   - time to send bits into link = $L/R$

4. Propagation delay:
   - $d$ = length of physical link
   - $s$ = propagation speed in medium (~$2 \times 10^8$ m/sec)
   - propagation delay = $d/s$

Queueing Delay (revisited)

- $R$ = link bandwidth (bps)
- $L$ = packet length (bits)
- $a$ = average packet arrival rate
- traffic intensity = $La/R$

$La/R \sim 0$: average queuing delay small

$La/R \rightarrow 1$: delays become large

$La/R > 1$: more "work" arriving than can be serviced, average delay infinite

Note: $s$ and $R$ are very different quantities!
"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 on path towards destination
  - router i will return packets to sender
  - sender times interval between transmission and reply.

<table>
<thead>
<tr>
<th>Router</th>
<th>Delay 1</th>
<th>Delay 2</th>
<th>Delay 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>cs-gw</td>
<td>1 ms</td>
<td>1 ms</td>
<td>2 ms</td>
</tr>
<tr>
<td>border1-rt-fa5-1-0.gw.umass.edu</td>
<td>1 ms</td>
<td>1 ms</td>
<td>2 ms</td>
</tr>
<tr>
<td>cht-vbns.gw.umass.edu</td>
<td>6 ms</td>
<td>5 ms</td>
<td>5 ms</td>
</tr>
<tr>
<td>jn1-at1-0-0-19.wor.vbns.net</td>
<td>16 ms</td>
<td>11 ms</td>
<td>13 ms</td>
</tr>
<tr>
<td>jn1-so7-0-0-0.wae.vbns.net</td>
<td>21 ms</td>
<td>18 ms</td>
<td>18 ms</td>
</tr>
<tr>
<td>abilene vbns abilene ucaid edu</td>
<td>22 ms</td>
<td>18 ms</td>
<td>22 ms</td>
</tr>
<tr>
<td>nycm-wash.abilene.ucaid.edu</td>
<td>22 ms</td>
<td>22 ms</td>
<td>22 ms</td>
</tr>
<tr>
<td>62.40.103.253</td>
<td>104 ms</td>
<td>109 ms</td>
<td>106 ms</td>
</tr>
<tr>
<td>de2-1.de1.de.geant.net</td>
<td>109 ms</td>
<td>102 ms</td>
<td>104 ms</td>
</tr>
<tr>
<td>de.fr1.fr.geant.net</td>
<td>113 ms</td>
<td>121 ms</td>
<td>114 ms</td>
</tr>
<tr>
<td>renater-gw.fr1.fr.geant.net</td>
<td>112 ms</td>
<td>114 ms</td>
<td>112 ms</td>
</tr>
<tr>
<td>nio-n2.cssi.renater.fr</td>
<td>111 ms</td>
<td>114 ms</td>
<td>116 ms</td>
</tr>
<tr>
<td>nice.cssi.renater.fr</td>
<td>123 ms</td>
<td>125 ms</td>
<td>124 ms</td>
</tr>
<tr>
<td>r3t2-nice.cssi.renater.fr</td>
<td>126 ms</td>
<td>126 ms</td>
<td>124 ms</td>
</tr>
<tr>
<td>eurecom-valbonne.r3t2.ft.net</td>
<td>135 ms</td>
<td>128 ms</td>
<td>133 ms</td>
</tr>
<tr>
<td>194.214.211.25</td>
<td>126 ms</td>
<td>128 ms</td>
<td>126 ms</td>
</tr>
<tr>
<td>fantasia.eurecom.fr</td>
<td>132 ms</td>
<td>128 ms</td>
<td>136 ms</td>
</tr>
</tbody>
</table>

* means no response (probe lost, router not replying)

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

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 (more)

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

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

Throughput: Internet Scenario

• Per-connection end-end throughput: $\min(R_c,R_s,R/10)$
• In practice: $R_s$ or $R_c$ is often bottleneck
  - "last mile" connection

10 connections (fairly) share backbone bottleneck link $R_{bits/sec}$
Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
   - end systems, access networks, links
1.3 Network core
   - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

Protocol "Layers"

Networks are complex!

- Many "pieces":
  - hosts
  - routers
  - links of various media
  - applications
  - protocols
  - hardware, software

Question: Is there any hope of organizing structure of network?

Or at least our discussion of networks?

Organization of air travel

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

• 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
- layered reference model for discussion
- 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
- layering considered harmful?
  - May need info from other layer (e.g. rate)
  - May be redundant functions (e.g. error check)

Internet Protocol Stack

- application: supporting network applications
  - FTP, SMTP, HTTP
- transport: process-process data transfer (segment)
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements (frames)
  - PPP, Ethernet
- 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
  - needed?

Encapsulation

Chapter 1: Roadmap

1.1 What is the Internet?
1.2 Network edge
  - end systems, access networks, links
1.3 Network core
  - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History

Network Security

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

Bad guys can put malware into hosts via Internet

- Malware can get in host from a virus, worm, or trojan horse.
- Spyware malware can record keystrokes, Web sites visited, upload info to collection site
- Infected host can be enrolled in a botnet, used for spam and DDoS attacks
- Malware is often self-replicating: from an infected host, seeks entry into other hosts

Bad guys can put malware into hosts via Internet

- Trojan horse
  - Hidden part of some otherwise useful software
  - Today often on a Web page (Active-X plugin)
- Virus
  - Infection by receiving object (e.g., e-mail attachment), actively executing
  - self-replicating: propagate itself to other hosts, users
- Worm: infection by passively receiving object that gets itself executed
  - self-replicating: propagates to other hosts, users

Sapphire Worm aggregate scan rate in the first 5 minutes of outbreak (CAIDA, UWisc data)
Bad guys can attack servers and 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 toward target from compromised hosts

The 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 used for end-of-chapter labs is a (free) packet-sniffer

The bad guys can use false source addresses

IP spoofing: send packet with false source address

The bad guys can record and playback

record-and-playback: sniff sensitive info (e.g., password), and use later
- password holder is that user from system point of view

Network Security

- A bit in this course
- Chapter 8: focus on security

Chapter 1: roadmap

1.1 What is the Internet?
1.2 Network edge
   - end systems, access networks, links
1.3 Network core
   - circuit switching, packet switching, network structure
1.4 Delay, loss and throughput in packet-switched networks
1.5 Protocol layers, service models
1.6 Networks under attack: security
1.7 History
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 demonstration
- NCP (Network Control Protocol)
- first e-mail program
- ARPAnet has 15 nodes

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
- 1980's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

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: Cnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

1990, 2000's: commercialization, the Web, new apps
- Early 1990's: ARPAnet decommissioned
- early 1990's: 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 killer apps: instant messaging, P2P file sharing
  - network security to forefront
  - est. 50 million host, 100 million+ users
  - backbone links running at Gbps

2007:
- ~500 million hosts
- Voice, Video over IP
- P2P applications: BitTorrent (file sharing) Skype (VoIP), PPLive (video)
- more applications: YouTube, gaming
- wireless, mobility

Introduction: Summary

Covered a "ton" of material
- Internet overview
- what's a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
- 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

CS 3516 - Computer Networks