Outline

- Administrative trivia’s
- What Is the Internet?
- Access Networks
Overview

- Instructor: Zizhan Zheng
  - Research Interests: Networks, clouds, cybersecurity, machine learning
  - Office: Stanley Thomas 307B
  - Email: zzheng3@tulane.edu

- Office hours: Tu 3:30-4:30pm, Wed 2-3pm (starting from the second week)

- Course Webpage
  - Used to post weekly schedule, homework assignments, lecture slides, reading assignment, etc.
Course Overview

- Objective: understand the key concepts and fundamental principles in the design and analysis of computer network

- Our Approaches
  - Using the Internet as the paradigm
  - Combine system and analytic approaches (graduate level)
  - A Top-Down Approach
Textbooks

James Kurose and Keith Ross

Communication Networks: An Optimization, Control, and Stochastic Networks Perspective
R. Srikant and Lei Ying
References

- Communication Networks: A Concise Introduction, Jean Walrand and Shyam Parekh

- TCP/IP Illustrated, Volume 1: The Protocols (2nd Edition), Kevin R. Fall and W. Richard Stevens
Homework and Labs

- Homework assignments (3-4)
  - Written problem sets
  - Graduate level students will be given extra questions that require advanced algorithmic and/or analytic techniques.

- Labs: programming assignments (2-3)
  - Python, C/C++, JAVA

- Requests for a homework/lab assignment extension (with a valid reason) must be given to the instructor before the assignment is due.
Exams

- One midterm and one final exam
  - closed-book and closed-notes
  - a cheat sheet allowed: one letter page single-sided

- The midterm will be non-comprehensive, but the final exam will be comprehensive.

- Missing an exam will result in a grade of zero. A request for a make-up exam must be given to the instructor prior to the exam date (documentation may be required).
Attendance

- Attendance in class is required.

- Students are fully responsible for all material presented or assigned in class.
Grading

- Homework - 25%
- Labs - 25%
- Midterm - 20%
- Final Exam - 30%

- All grades will be posted on Canvas.
Academy Integrity

- You are required to adhere to Tulane’s Code of Academic Conduct.

- You may discuss homework problems with your classmates. However, what you turn in must be your own. You may not read another classmate’s solutions or copy a solution from the web.

- Every cheating will be reported to the Associate Dean of Newcomb-Tulane College. If two people are caught sharing solutions then both the copier and copiee will be held equally responsible. Cheating on an exam will result in failing the course.
Outline

- Administrative trivia’s
- What Is the Internet?
- Access Networks
A Nuts-and-Bolts View of the Internet

- **Hosts = end systems**
  - Running network apps
  - Billions of connected computing devices

- **Communication links**
  - copper, cables, fiber, radio, satellite
  - transmission rate (bit/sec), maximum distance

- **Packet switches**: forward packets
  - Routers and link-layer switches
  - ISP: a network of packet switches

- **Internet**: “network of networks”
A Service View of the Internet

- Infrastructure that provides services to network apps:
  - Web, email, messaging, games, e-commerce, social nets, maps, healthcare...
  - >1,500,000 apps in Google Play, most of which require network connections

- Provides programming interface to apps
  - Socket interface
  - Hooks that allows apps “connect” to each other
  - Provides service options: reliability, security, etc.
What is a Protocol?

a human protocol:

Hi

Hi

Got the time?

2:00

Thanks

a computer network protocol:

TCP connection request

TCP connection response

Get http://www.tulane.edu

<file>

acknowledgement

[Kurose and Ross]
TCP Protocol Handshakes

Host A

SYN(seq=x)

ACK(seq=x), SYN(seq=y)

ACK(seq=y)

DATA(seq=x+1)

ACK

FIN

ACK

FIN

ACK

Host B

[Y. Richard Yang]
What is a Network Protocol?

- A network protocol defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other events.

- Protocol standardization
  - Most widely used protocols are defined in standards
  - Internet standards are developed by Internet Engineering Task Force (IETF) in the form of Request for Comments (RFCs)
  - Ethernet and wireless WiFi standards: IEEE 802 LAN/MAN Standards Committee

- Wireshark packet sniffer: a useful tool to learn protocols
Outline

- Administrative trivia’s
- What Is the Internet?
- Access Networks
A closer look at network structure

- **Network Core**
  - Interconnected routers

- **Network Edge**
  - **hosts**: clients and servers
    - clients: desktops, smartphones, smart devices
    - servers: service/content providers, often in data centers
  - **access networks**: connect hosts to the core
    - DSL, Cable, Ethernet, Wireless, Fiber to the home (FTTH), Satellite

[Kurose and Ross]
Access network: digital subscriber line (DSL)

- Use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet,
  - voice over DSL phone line goes to telephone net

- ADSL: asymmetric downstream and upstream rates

[Image]

[Kurose and Ross]
Access network: cable network

- Homes *share access network* to cable headend
  - *multiple access protocol* for upstream transmission (frequency/time)

[Image of network diagram]

*data, TV transmitted at different frequencies over shared cable distribution network*

*[Kurose and Ross]*
Access network: home network

- home network to/from headend or central office
- cable or DSL modem
- router, firewall, NAT
- wired Ethernet (1 Gbps)
- wireless access point (54 Mbps)
- wireless devices
- often combined in single box

[Kurose and Ross]
Link Characteristics

- **Wired**
  - DSL: a few Mbps up to 5km
  - Cable: 10 Mbps over 1km
  - Ethernet: 100 Mbps up to 110m

- **Wireless**
  - WiFi: tens of Mbps up to hundred meters
  - Cellular: 10 Mbps over a few km

- **Optical**: 10Gbps over 80km

[Walrand and Parekh]
Summary

What is the Internet?
- A Nuts-and-Bolts View: hosts, communication links, packet switches
- A Service View: socket interface

What is a Network Protocol?
- Defines message format, order, and actions towards events
- Internet protocol suite: TCP/IP protocols

A closer look at network structure
- End systems = hosts (clients, servers)
- Access networks: DSL, Cable, WiFi, Ethernet, etc.
- Network core: Interconnected routers
Today’s Plan

- The Network Core
  - Packet Switching vs. Circuit Switching
  - Interconnection of ISPs
The Network Core

- mesh of interconnected routers

- packet-switching: hosts break application-layer messages into *packets*
  - A packet: header + payload (a set of bits)
  - forward packets from one router to the next, across links on path from source to destination
  - each packet transmitted at full link capacity

[Kurose and Ross]
Packet-switching: store-and-forward

- **store and forward**: entire packet must arrive at router before it can be transmitted on next link
- takes $L/R$ seconds to transmit (push out) $L$-bit packet into link at $R$ bps
  - Ex: $R = 7.5$ Mbps, $L = 1.5$ Mbits, one-hop transmission delay = 0.2 sec
- End-to-end delay = $2L/R$ (assuming zero propagation delay)
Packet-switching: store-and-forward

- How long it takes for the destination to receive all the three packets?
- $K$ packets? $N$ links?
- more on delay shortly ...

$L$ bits per packet

source $\quad R$ bps $\quad$ router $\quad R$ bps $\quad$ destination

\[
\frac{L}{R} \quad 2\frac{L}{R} \quad 3\frac{L}{R} \quad 4\frac{L}{R}
\]
Queueing delay and packet loss

- Each output link has a **queue** (buffer) of **finite** space
- An arriving packet will queue when link is **busy**
- **Packet loss** will occur when the output queue is **full**

[Kurose and Ross]
Two key network-core functions

**routing**: determines source-destination route taken by packets

- **routing algorithms**

**forwarding**: move packets from router’s input to appropriate router output

routing algorithm

<table>
<thead>
<tr>
<th>local forwarding table</th>
<th>header value</th>
<th>output link</th>
</tr>
</thead>
<tbody>
<tr>
<td>header value</td>
<td>output link</td>
<td></td>
</tr>
<tr>
<td>0100</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>0101</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>0111</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

destination address in arriving packet’s header

[Kurose and Ross]
Addressing

- Every host attached to the Internet has a unique IP address
  - IP – Internet Protocol
  - In IPv4, each IP address is a 32-bit string
  - conventionally written in the form $a.b.c.d$
    - $a, b, c, d$ are the decimal value of the four bytes
    - E.g., 169.229.60.32 corresponds to 10101001.11100101.00111100.00100000
Alternative core: circuit switching

- commonly used in traditional telephone networks
- resources reserved for “call” between source & dest:
  - resources: transmission rate, buffer, etc.
- In diagram, each link has four circuits

[Kurose and Ross]
Alternative core: circuit switching

- commonly used in traditional telephone networks
- resources reserved for “call” between source & dest:
  - resources: transmission rate, buffer, etc.
- In diagram, each link has four circuits
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - guaranteed performance
  - circuit segment idle if not used by call

[Kurose and Ross]
Multiplexing in Circuit-Switched Networks

FDM (frequency-division multiplexing)

Example:
4 users

TDM (time-division multiplexing)

[Kurose and Ross]
Packet Switching vs. Circuit Switching

- **Example**
  - 1Mb/s link
  - each user:
    - 100 kb/s when “active”
    - active 10% of time

- **How many users can be supported?**
  - **circuit switching**: 10 users
  - **packet switching**: with 35 users, probability that > 10 users active at same time is less than 0.0004
    - Assume that users become active independently
    - \( \Pr(\text{user } i \text{ is active}) = 0.1 \)
    - \( \Pr(k \text{ users are active}) = \binom{35}{k}0.1^k(1 - 0.1)^{35-k} \)
# Packet Switching vs. Circuit Switching

<table>
<thead>
<tr>
<th></th>
<th>Circuit Switching</th>
<th>Packet Switching</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource allocation</td>
<td>reserved</td>
<td>on demand</td>
</tr>
<tr>
<td>Routing</td>
<td>fixed routing</td>
<td>flexible routing</td>
</tr>
<tr>
<td>Resource sharing</td>
<td>FDM/TDM</td>
<td>statistical multiplexing</td>
</tr>
<tr>
<td>Performance guarantee</td>
<td>yes</td>
<td>no (&quot;best effort&quot; only)</td>
</tr>
</tbody>
</table>

- Robust to attacks
- Better for bursty traffic
Packet Switching vs. Circuit Switching

- The designers of the Internet opt for *simplicity*:
  - "*best effort*" service: the network should try to deliver the packets as well as possible (but no guarantee)
  - *end-to-end principle*: "tasks should not be performed by routers if they can be performed by the end devices"
    - e.g., reliable transfer, congestion control are implemented by end systems
  - *stateless routers*: router considers one packet at a time, no connection information -> robustness & scalability
Today’s Plan

- The Network Core
  - Packet Switching vs. Circuit Switching
  - Interconnection of ISPs

- A closer look at delay, loss, and throughput

- Layered architecture
Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - residential, company and university ISPs

- Access ISPs in turn must be interconnected.
  - so that any two hosts can send packets to each other

- Resulting network of networks is very complex
  - evolution was driven by economics and national policies
Internet structure: network of networks

Option: connect each access ISP to every other access ISP?

connecting each access ISP to each other directly doesn’t scale: $O(N^2)$ connections.

[Kurose and Ross]
Internet structure: network of networks

Option: connect each access ISP to one global transit ISP?

Customer and provider ISPs have economic agreement.

[Kurose and Ross]
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors ….

[Kurose and Ross]
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors .... which must be interconnected

[Note: Kurose and Ross]
Internet structure: network of networks

- at center: small # of well-connected large networks
  - “tier-1” commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g., Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

[Kurose and Ross]
Today’s Plan

- The Network Core
  - Packet Switching vs. Circuit Switching
  - Interconnection of ISPs
- A closer look at delay, loss, and throughput
- Layered architecture
Physical Media

- **bit**: propagates between transmitter/receiver 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
Link rate

- A link has a **transmission rate**: speed at which bits are transmitted
  - uplink rate: from the user’s device to the Internet
  - downlink rate: from the Internet to the user’s device
  - Ex: a 1Mbps link can send a 1MByte file in 8 sec (assume no gaps between packets and no propagation delay)

- **Bandwidth** of a link: the width of the range of frequencies
  - Ex: a telephone line can transmit signals over a range of frequencies from 300Hz to 1MHz (= 10^6 Hz), its bandwidth is about 1MHz (1MHz-300Hz ≈ 1MHz)
Link rate

- **Shannon Capacity**: maximum reliable link rate
- E.g., Shannon capacity for a bandlimited Gaussian channel

\[ C = W \log_2 \left(1 + \frac{P}{N_0 W}\right) \text{ bit per second} \]

- \( W \): bandwidth
- \( P \): power of the signal at the receiver (decreases with the length of the link)
- \( N_0 \): power density of the noise at the receiver
Four sources of packet delay

\( d_{\text{proc}}: \) nodal processing
- check bit errors
- determine output link
- typically < msec

\( d_{\text{queue}}: \) queueing delay
- time waiting at output link for transmission
- depends on congestion level of router

[Kurose and Ross]
Four sources of packet delay

\[ d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}} \]

\( d_{\text{trans}} \): transmission delay:
- \( L \): packet length (bits)
- \( R \): link bandwidth (bps)
- \( d_{\text{trans}} = L/R \)

\( d_{\text{prop}} \): propagation delay:
- \( d \): length of physical link
- \( s \): propagation speed (~2 \times 10^8 \text{ m/sec})
- \( d_{\text{prop}} = d/s \)

[Kurose and Ross]
Caravan analogy

- cars “propagate” at 100 km/hr
- toll booth takes 12 sec to service one car (bit transmission time)
- car ~ bit; caravan ~ packet
- Q: How long until 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
- A: 62 minutes

[Kurose and Ross]
Caravan analogy (cont.)

- 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?
  - **A:** Yes! after 7 min, first car arrives at second booth; three cars still at first booth
Queueing delay and packet loss

- Each output link has a queue (buffer) of finite space
- An arriving packet will queue when link is busy
- Packet loss will occur when the output queue is full
  - lost packet may be retransmitted by previous node, by source end system, or not at all
Queueing delay (revisited)

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

- $La/R \sim 0$: avg. queueing delay small
- $La/R \rightarrow 1$: avg. queueing delay large
- $La/R > 1$: more “work” arriving than can be serviced, average delay infinite!

\[ \text{traffic intensity } = \frac{La}{R} \]

[Kurose and Ross]
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

[Kurose and Ross]
traceroute www.osu.edu

1 010-stanley-d2001-8024.tulane.net (129.81.217.1) 2.943 ms 2.598 ms 5.689 ms
2 011-pyramid-8208.tulane.net (172.24.0.37) 2.475 ms
   172.21.0.137 (172.21.0.137) 2.715 ms
   011-pyramid-8208.tulane.net (172.24.0.37) 2.515 ms
3 poyd-1462-srx5800-1.tulane.net (172.21.0.158) 2.843 ms
   gn-7050.tulane.net (172.24.1.150) 2.933 ms
   poyd-1462-srx5800-1.tulane.net (172.21.0.158) 2.971 ms
4 bu-960.tulane.net (129.81.255.105) 2.729 ms
   bu-960.tulane.net (129.81.255.97) 2.701 ms
   bu-960.tulane.net (129.81.255.105) 2.498 ms
5 72.20.176.89 (72.20.176.89) 2.898 ms 2.923 ms 3.232 ms
6 border2.te13-1.huntbros-14.ext1a.dal.pnap.net (72.5.252.77) 29.610 ms 29.933 ms 29.782 ms
7 core2.po1-20g-bbnet1.dal006.pnap.net (216.52.191.10) 30.495 ms 30.364 ms 30.362 ms
8 bbr2.9a9.inapvox-26.dal006.pnap.net (64.95.158.242) 29.976 ms 30.013 ms 29.793 ms
9 bbr1-xe-1-2-1.inapbb-chg-dal-1.chg.pnap.net (64.95.158.157) 57.029 ms 53.058 ms 53.164 ms
10 eq-exchange.tr01-chcgil01.transitrail.net (206.223.119.116) 48.596 ms 48.998 ms 48.694 ms
11 ae-1.80.rtsw.chic.net.internet2.edu (64.57.20.150) 48.962 ms 50.221 ms 49.001 ms
12 * * *
13 clmbn-r5-et-5-0-0s100.core.oar.net (199.218.39.241) 51.810 ms 52.146 ms 52.501 ms
14 clmbs-r5-et-3-0-0s100.core.oar.net (199.218.20.33) 50.273 ms 50.432 ms 51.661 ms
15 franklin-r0-vl334.cpe.oar.net (199.18.169.10) 51.722 ms 52.362 ms 52.853 ms
16 socc4-forg6-4.net.ohio-state.edu (164.107.1.129) 52.217 ms 52.533 ms 50.957 ms

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

Kurose and Ross
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

[Kurose and Ross]
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

[Kurose and Ross]
Today’s Plan

- The Network Core
  - Packet Switching vs. Circuit Switching
  - Interconnection of ISPs
- A closer look at delay, loss, and throughput
- Layered architecture
Protocol layers

*Networks are complex, with many “pieces”:*

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

*Various services provided:*

- Move bits over various media: multiplex access, error control
- Packet switching: forwarding & routing
- Reliable transfer: retransmission, flow control, congestion
- Quality of service: delay, throughput, security

*Question:* is there any hope of *organizing* structure of network?

.... or at least our discussion of networks?
Layering of airline functionality

<table>
<thead>
<tr>
<th>Layer</th>
<th>Action 1</th>
<th>Action 2</th>
<th>Action 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ticket (purchase)</td>
<td>ticket (complain)</td>
<td>ticket</td>
<td></td>
</tr>
<tr>
<td>baggage (check)</td>
<td>baggage (claim)</td>
<td>baggage</td>
<td></td>
</tr>
<tr>
<td>gates (load)</td>
<td>gates (unload)</td>
<td>gate</td>
<td></td>
</tr>
<tr>
<td>runway (takeoff)</td>
<td>runway (land)</td>
<td>takeoff/landing</td>
<td></td>
</tr>
<tr>
<td>airplane routing</td>
<td>airplane routing</td>
<td>airplane routing</td>
<td></td>
</tr>
<tr>
<td>departure airport</td>
<td>intermediate air-traffic control centers</td>
<td>arrival airport</td>
<td></td>
</tr>
</tbody>
</table>

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

[Kurose and Ross]
Internet protocol stack

[ Walrand and Parekh ]
Why layering?

- simplify design
  - explicit structure allows identification of 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

- layering considered harmful?
  - Loss of efficiency: “cross-layer” approaches
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, 802.11 (WiFi), PPP

- **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?

[Reference: Kurose and Ross]
Encapsulation

message  \[ M \]
segment  \[ H_t \]
datagram  \[ H_n \]
frame  \[ H_l \]

source

application
transport
network
link
physical

destination

application
transport
network
link
physical

switch

router

[Kurose and Ross]