EE 122: Introduction To Communication Networks

Fall 2010 (MW 4-5:30 in 101 Barker)

Scott Shenker

TAs: Sameer Agarwal, Igor Ganichev, Prayag Narula

http://inst.eecs.berkeley.edu/~ee122/

Materials with thanks to Jennifer Rexford, Ion Stoica, Vern Paxson and other colleagues at Princeton and UC Berkeley

Today will cover two topics

- Overview of course
  - Topics
  - People
  - Policies
  - Core focus

  Break

- Three basic questions
  - Why is networking fascinating?
  - Why are networking courses so terrible?
  - Why is networking so hard?

EE122 Comes in Two Flavors

- Spring offering: taught by EE faculty
  - More emphasis on diverse link technologies, wireless, communication theory (and a simulation project)
    - No systems programming
- Fall offering: taught by CS faculty
  - More emphasis on Internet architecture, applications, and real-world practice (and a programming project)
    - (Almost) no mathematics, no simulation
- Make sure this class is the right one for you!

Course Overview

What Will You Learn in This Course?

- **Insight:** key concepts in networking
  - *What are the different ways you can route a packet?*
  - *What is congestion control?*
- **Knowledge:** how the Internet works
  - *What does an IP packet look like?*
  - *How can a single typo bring down a third of the Internet?*
- **Skills:** network programming
  - *Socket programming*
  - *Designing and implementing protocols*

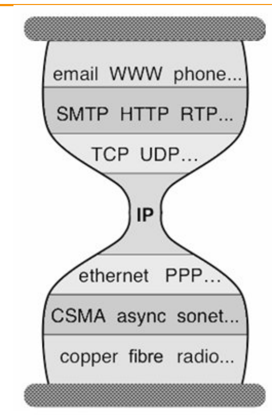
This class focuses on the Internet

- The core of the Internet "architecture":
  - IP, DNS, BGP
- Other technologies crucial to the Internet
  - Lower-level technologies: Ethernet, wireless…
  - Higher-level technologies: TCP, HTTP, applications.…
  - Component technologies: switches, routers, firewalls,…
- If a networking technology isn't a core piece of the Internet, we won't spend much time on it
  - E.g., sensornets

Will consider different perspectives

- Different geographic scales:
  - LAN vs WAN vs Interdomain
- Different conceptual approaches:
  - Architecture vs Protocol vs Algorithm
- Different aspects of functionality:
  - Layers

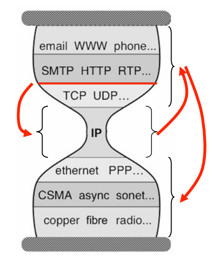
The Internet: an hourglass *with layers*



Structure of the Course (1st Half)

- Start at the top
  - Protocols: how to structure communication
  - Sockets: how applications view the Internet
- Then study the "narrow waist" of IP
  - IP best-effort packet-delivery service
  - IP addressing and packet forwarding
- And how to build on top of the narrow waist
  - Transport protocols (TCP, UDP)
  - Domain Name System (DNS)
  - Applications (Web, email, file transfer)
- Looking underneath IP
  - Link technologies (Ethernet, bridges, switches)

Hourglass Representation



Structure of the Course (2nd Half)

- How to get the traffic from here to there …
  - Glue (ARP, DHCP, ICMP)
  - Routing (intradomain, interdomain)
- … in a way that's both efficient and stable
  - How much data to keep in flight (the *window*)
  - Without clogging the network (*congestion*)
  - With some assurance (*quality of service*) … or not
- How to control network traffic …
  - Enforcing policy
  - Defending against attacks
- … and scale it to potentially huge structures
  - P2P and DHTs

Instructor: Scott Shenker

- Trained as a physicist (phase transitions, chaos)
- Research: physics, economics, operating systems, security, distributed systems, datacenter design
  - Diversity reflects my learning and teaching style
- For last 20+ years, *main* focus has been networking and Internet architecture
  - Particularly *clean-slate* designs
- Office hours W 5:45-6:45 in 449 Soda Hall
  - And by appointment (arrange by email)
  - On campus M, W, Th
  - Live in RAD Lab (no office, no phone)

Problems with my teaching style…

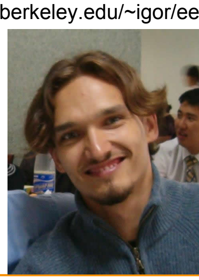
- I don't think visually
  - Ask me to draw pictures, if they would help
- When you look bored, I speed up
  - If you are bored, feel free to sleep (*at your peril*)
  - If you are lost, ask me a question!
- Weak on logistics
  - Will figure out as we go along
  - Will depend on my TAs!

TA: Sameer Agarwal

- Office hours Friday 3-4 in ??? Soda
  - And by appointment
- Section W 11-12 in 247 Cory
  - sameerag@cs.berkeley.edu
  - http://www.cs.berkeley.edu/~sameerag/ee122/fa10/

TA: Igor Ganichev

- Office hours Monday 3pm-4pm in ??? Soda
  - And by appointment
- Section T 10-11 in 237 Cory
  - igor@cs.berkeley.edu
  - http://www.eecs.berkeley.edu/~igor/ee122/index.html



TA: Prayag Narula

- Office hours Th 4-5 in ??? Soda
  - And by appointment
- Section M 11-12 in 247 Cory
  - prayag@ischool.berkeley.edu
  - http://people.ischool.berkeley.edu/~prayag/eecs122/index.html

Mystery TA…..

Don't be a passive listener!

- Ask questions!
  - Help me understand where I'm not being clear
  - Keep me from going too fast
- When I ask a question, I don't care if you answer it, but please ***think about the question!***
  - My questions let you think rather than just listen
  - And, some of these questions will show up on exams!
- The best way to understand networking is to *first* try to solve the design issues yourself
  - Then the current solution will make a lot more sense

Fourth Lecture

- We will design the Internet in 90 minutes
- We will walk through the task of sending bits from one host to another
- This will bring up a set of design decisions
  - What do addresses look like?
  - ….
- We will discuss possible alternatives
- Do you think we'll come up with something better than the current Internet?

Please ask for help!

- Even the best of you won't understand everything
  - That's my fault, but you need to ask for help.
- Come to office hours, request an appointment, communicate by e-mail
  - We are here to help, including general advice!
  - TAs first line for help with programming problems
- Give us suggestions/complaints/feedback as early as you can
- What's your background?
  - Fill out the survey (http://tinyurl.com/ee122survey)

Course Books

- Textbook
  - J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach,* 5th Edition, Addison Wesley, 2010.
    - We jump around a lot, used more as a reference than a narrative
    - 4th Edition ok, but make sure you translate the reading assignments
- Recommended and on reserve:
  - W. R. Stevens, B. Fenner, A. M. Rudoff, Unix Network Programming: The Sockets Networking API, Vol. 1, 3rd Ed., Addison-Wesley, 2004.
  - W. R. Stevens, TCP/IP Illustrated, Volume 1: The Protocols, Addison-Wesley, 1993.

Web Site and Mailing List

- Web site: *http://inst.eecs.berkeley.edu/~ee122/*
  - Assignments, lecture slides (but not *always* before lecture)
  - Note: if you are following the slides during lecture, please don't use them to answer questions I ask
- Mailing list: ee122@lists.berkeley.edu
  - Sign up from class home page
- Use bspace to hand in homework (details to be announced)

Class Workload

- Four homeworks spread over the semester
  - Strict due dates (no slip days!)
  - Deadlines are generally 3:50PM prior to lecture
- One large project divided into four stages:
  - Part 1 A/B and Part 2 A/B:
  - Distributed game: tiny World of Warcraft
  - Part 1: Client-server
  - Part 2: Distributed storage
  - C (or C++) required
  - Deadlines 11PM
- Exams
  - Midterm: Monday October 18 in class, 4-5:30PM
  - Final: Thursday Dec 16 location TBD, 8-11AM
  - Closed book, open crib sheet

Grading

Homeworks	20% (5% each)
Projects	40% (20+20)
Midterm exam	15%
Final exam	25%

- Course graded to mean of B
  - Relatively easy to get a B, harder to get an A or a C
  - ≈ 10% A, 15% A-, 15% B+, 20% B, 15% B-, 15% C+, 10% C
  - A+ reserved for superstars (1 or 2 per class)
  - Mean can shift up for an excellent class
    - For which the TAs have significant input

Assumptions

- You can program
  - Knowledge of C or C++
  - Ask a TA if you aren't sure of your programming
- You are comfortable thinking abstractly
  - And know basic probability
- Background material will ***not*** be covered in lecture. TAs will spend very little time reviewing material not specific to networking

No Cheating

- Cheating: not doing the assignment by yourself.
- Fine to *talk* with other students about assignments
  - But only general concepts, not specifics
- No copying, no Google, etc.
  - If you're unsure, then ask.
- Will use automated similarity detection
- *Don't be an idiot….*

5 Minute Break

Questions Before We Proceed?

Three Questions

Why is networking fascinating?

- The Internet has had a tremendous **impact**
- The Internet changed the networking **paradigm**
- The design of the Internet presents interesting intellectual **challenges**
- Many of these intellectual challenges remain **unsolved**

Impact

- Internet changed the way we gather information
  - Web, search engines
- Internet changed the way we relate to each other
  - Email, facebook, twitter
- Which would you choose?
  - Computers without the Internet (standalone PCs)
  - Internet without computers (or really old ones)

New Networking Paradigm

- Separation of application from network
- Statistical multiplexing
- Ad hoc deployment
- Autonomous policies

Intellectual Challenges

- Connecting two computers is easy
  - So why is designing the Internet hard?
- Internet must cope with unprecedented scale, diversity and dynamic range
  - More about this later in lecture….

Unsolved challenges

- Security
  - Security of infrastructure
  - Security of users
- Availability
  - Internet is very resilient
  - But availability is not sufficient for critical infrastructures
- Evolution
  - It is too hard to change the Internet architecture

Why do networking courses suck?

- Haven't changed the basic Internet architecture
  - Even IPv6 is very similar to IP
- You can't test an Internet architecture in your lab, or even a testbed
- So we only understand what we currently have
- **We are teaching history, not principles**
  - You will learn "first tries" not "fundamental answers"
  - As if we taught MS-DOS in an operating system course

Quote from John Day (Internet pioneer)

There is a tendency in our field to believe that everything we currently use is a paragon of engineering, rather than a snapshot of our understanding at the time. We build great myths of spin about how what we have done is the only way to do it to the point that our universities now teach the flaws to students (and professors and textbook authors) who don't know better.

My Goal

- Focus when possible on "fundamental questions"
  - And covering recent and future designs last 2 lectures
- You will have to learn the current design
  - But I will point out where it falls short
- For instance, you will learn what three things the Internet got the "most wrong"….
- You will end up with a mixture of the "big picture" and "current design details"

Why is Networking Hard?

- There are many challenges that make designing the Internet harder than just passing bits on a wire
- *Which of these apply to the phone network?*

Scale

Two Billion Internet Users

WORLD INTERNET USAGE AND POPULATION STATISTICS

World Regions	Population (2010 Est.)	Internet Users Dec. 31, 2000	Internet Users Latest Data	Penetration (% Population)	Growth 2000-2010	Users % of Table
Africa	1,013,779,050	4,514,400	**110,931,700**	10.9 %	2,357.3 %	5.6 %
Asia	3,834,792,852	114,304,000	**825,094,396**	21.5 %	621.8 %	42.0 %
Europe	813,319,511	105,096,093	**475,069,448**	58.4 %	352.0 %	24.2 %
Middle East	212,336,924	3,284,800	**63,240,946**	29.8 %	1,825.3 %	3.2 %
North America	344,124,450	108,096,800	**266,224,500**	77.4 %	146.3 %	13.5 %
Latin America/Caribbean	592,556,972	18,068,919	**204,689,836**	34.5 %	1,032.8 %	10.4 %
Oceania / Australia	34,700,201	7,620,480	**21,263,990**	61.3 %	179.0 %	1.1 %
WORLD TOTAL	6,845,609,960	360,985,492	**1,966,514,816**	**28.7 %**	444.8 %	100.0 %

Dynamic Range

- Round-trip times (latency) vary 10 μsec's to sec's
  - 5 orders of magnitude
- Data rates (bandwidth) vary from kbps to 10 Gbps
  - 7 orders of magnitude
- Queuing delays in the network vary from 0 to sec's
- Packet loss varies from 0 to 90+%
- …..

Diversity of end systems

- End system (host) capabilities vary from *cell phones* to *supercomputer clusters*

Diversity of application requirements

- Size of transfers
- Bidirectionality (or not)
- Latency sensitive (or not)
- Tolerance of jitter (or not)
- Tolerance of packet drop (or not)
- Need for reliability (or not)
- Multipoint (or not)
- …..

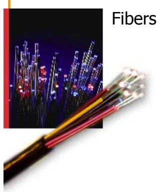
Ad hoc deployment

- Can't assume carefully managed deployment
  - Network must work without planning

Networks contain many components

Links

Fibers



Coaxial Cable

Interfaces

Ethernet card

Wireless card

Switches/routers

Large router

Telephone switch

They can all fail….

- Question: suppose a communication involves 50 components which work correctly (independently) 99% of the time. What's the likelihood the communication fails at a given point of time?
  - Answer: success requires that they all function, so failure probability = 1 - 0.9950 = 39.5%.
- Must design the system to expect failure
  - Why is the Internet like a 12-step program?

Greed

- There are greedy people out there who want to:
  - Steal your data
  - Use your computer for attacks
- There is a thriving underground economy for compromised computers and financial information

controlled.

71. ANCHETA would develop a worm which would cause infected computers, unbeknownst to the users of the infected computers, to:

a. report to the IRC channel he controlled;

b. scan for other computers vulnerable to similar infection; and

c. succumb to future unauthorized accesses, including for use as proxies for spamming.

his worm caused 1,000 to 10,000 new bots to join his botnet over the course of only three days.

73. ANCHETA would then advertise the sale of bots for the purpose of launching DDOS attacks or using the bots as proxies to send spam.

74. ANCHETA would sell up to 10,000 bots or proxies at a time.

75. ANCHETA would discuss with purchasers the nature and extent of the DDOS or proxy spamming they were interested in

79. ANCHETA would accept payments through Paypal.

103. In or about August 2004, ANCHETA updated his advertisement to increase the price of bots and proxies, to limit the purchase of bots to 2,000 "due to massive orders," and to warn,

adware on those computers without notice to or consent from the users of those computers, and by means of such conduct, obtained the following approximate monies from the following advertising service companies:

COUNT	APPROXIMATE DATES	APPROXIMATE NUMBER OF PROTECTED COMPUTERS ACCESSED WITHOUT AUTHORIZATION	APPROXIMATE PAYMENT
SEVEN	November 1, 2004 through November 19, 2004	26,975	$4,044.26 from Gammacash
EIGHT	November 16, 2004 through December 7, 2004	8,744	$1,306.52 from LOUDcash
NINE	January 15, 2005	[illegible]	[illegible]

Malice

- There are malicious people out there who want to:
  - Bring your system down and/or steal data
- When attacker is a nation-state, attacks are far harder to stop
  - Many defensive techniques involve stopping attacks that have been seen before
  - But nation-states can use *new* attack vectors

Speed of Light

- Question: how long does it take light to travel from Berkeley to New York?
- Answer:
  - Distance Berkeley → New York: 4,125 km (great circle)
  - Traveling 300,000 km/s: 13.75 msec

Networking Latencies

- Question: how long does it take an Internet "packet" to travel from Berkeley to New York?
- Answer:
  - For sure ≥ 13.75 msec
  - Depends on:
    - The *route* the packet takes (could be circuitous!)
    - The propagation speed of the *links* the packet traverses
      - E.g., in optical fiber light propagates at about 2/3 ***C***
    - The transmission rate (*bandwidth*) of the links (bits/sec)
      - and thus the size of the packet
    - Number of *hops* traversed (*store-and-forward* delay)
    - The "competition" for bandwidth the packet encounters (*congestion*). It may have to sit & wait in router queues.
  - In practice this boils down to:
    - ≥ 40 msec

Implications for Networking

- Question: how many cycles does your PC execute before it can possibly get a reply to a message it sent to a New York web server?
- Answer:
  - Round trip takes ≥ 80 msec
  - PC runs at (say) 3 GHz
  - 3,000,000,000 cycles/sec*0.08 sec = 240,000,000 cycles

= An Eon

- Communication **feedback** is always *dated*
- Communication fundamentally asynchronous

Even a Problem for LANs

- Question: what about between machines directly connected (via a *local area network* or LAN)?
- Answer:

```
% ping www.icir.org
PING www.icir.org (192.150.187.11): 56 data bytes
64 bytes from 192.150.187.11: icmp_seq=0 ttl=64 time=0.214 ms
64 bytes from 192.150.187.11: icmp_seq=1 ttl=64 time=0.226 ms
64 bytes from 192.150.187.11: icmp_seq=2 ttl=64 time=0.209 ms
64 bytes from 192.150.187.11: icmp_seq=3 ttl=64 time=0.212 ms
64 bytes from 192.150.187.11: icmp_seq=4 ttl=64 time=0.214 ms
```

- 200 μsec = 600,000 cycles
  - Still a loooong time …
  - … and asynchronous

Summary

- The Internet is a large complicated system that must meet a variety of challenges
- Not akin to e.g. programming languages
  - Which have well-developed theories to draw upon
- Much more akin to operating systems
  - Abstractions
  - Tradeoffs
  - Design principles / "taste"

Next Lecture

- Read through 1.1-1.3 of the Kurose/Ross book
- Take the survey (http://tinyurl.com/ee122survey)
- Subscribe to the mailing list
- Dust off your C/C++ programming skills if need be