CompSci514/ECE558: Computer Networks

Lecture 1: Course Introduction
Xiaowei Yang

xwy@cs.duke.edu
http://www.cs.duke.edu/~xwy
Roadmap

• Course introduction: what and why
• Audience
• Topics
• Workload
• Grading policy
• Staff
• Other questions?
What is networking?
A Plethora of Protocol Acronyms
A Heap of Header Formats?
TCP/IP Header Formats in Lego
A Big Bunch of Boxes

Router
Switch
Firewall
NAT
Load balancer
DHCP server
DNS server
Bridge
Repeater
Gateway
Intrusion Detection System
Scrubber
Route Reflector
Label Switched Router
WAN accelerator
Deep Packet Inspection
Base station
Packet sniffer
Packet shaper
Proxy
An Application Domain?
A place to apply theory?

- Algorithms and data structures
- Control theory
- Queuing theory
- Optimization theory
- Game theory and mechanism design
- Formal methods
- Information theory
- Cryptography
- Programming languages
- Graph theory
A place to build systems?

- Distributed systems
- Operating systems
- Computer architecture
- Software engineering
- …
What are the top ten classic problems in networking? I would like to solve one of them and submit a paper to SIGCOMM.” After hearing that we don't have such a list: "Then how do you consider networking a discipline?"

“So, these networking research people today aren't doing theory, and yet they aren't the people who brought us the Internet. What exactly are they doing?”

“Networking papers are strange. They have a lot of text.”

Is networking a problem domain or a scholarly discipline?
“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.” -- John Day (Internet pioneer)
My two cents …
Networking = “Plumbing”

Networking is the “plumbing” of computing
Almost all areas of computing are network-based.
  Distributed computing
  Big Data
  Cloud Computing
  Internet of Things
  Smart Cities

Networking is the backbone of computing.

Source: Raj Jain Keynote speech at SIGCOMM 2017
## Smart Everything

<table>
<thead>
<tr>
<th>Smart Watch</th>
<th>Smart TV</th>
<th>Smart Car</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Smart Health</th>
<th>Smart Home</th>
<th>Smart Kegs</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Smart Space</th>
<th>Smart Industries</th>
<th>Smart Cities</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
</tr>
</tbody>
</table>

**Source:** Raj Jain Keynote speech at SIGCOMM 2017
What’s Smart?

Old: Smart = Can think  Computation
        = Can Recall  Storage

Now: Smart = Can find quickly, Can Delegate
      Communicate = Networking

Smart Grid, Smart Meters, Smart Cars, Smart homes, Smart Cities, Smart Factories, Smart Smoke Detectors, …
Networking redefines CS

• All disciplines in the pre-networking CS field aim to solve problems for computing, storage, and the combination of two

• Networking adds a third dimension to problem space: computing, communication, and storage

• Old disciplines may apply

• New disciplines emerge
Why study networking?

Relevant

– Can measure/build things
– Can impact the real world

Widely-read papers

– Many of the most cited papers in CS are in networking
– Congestion control, distributed hash tables, resource reservation, self-similar traffic, multimedia protocols,…
– Three of top-ten CS authors (Shenker, Jacobson, Floyd)
Why studying networking?

- Young, relatively immature, and fast changing field
  - Many unsolved problems

- Require many skill sets
  - People from all fields of CS can find interesting networking problems to solve
    - Machine learning, Algorithms and data structures, Control theory, Queuing theory, Optimization theory, Game theory and mechanism design, Formal methods, Information theory, Cryptography, Programming languages, Graph theory, Systems, and Architecture
Why studying networking?

- Lots of platforms for building your ideas
  - Testbeds: Emulab, PlanetLab, Orbit, GENI
  - Programmability: Click, NetFPGA, Mininet
  - Routing software: Quagga, XORP, and Bird
  - Measurements: RouteViews, traceroute, Internet2
Networking is Fueling All Sectors of Economy

Networking companies are among the most valued companies: Apple, AT&T, Samsung, Verizon, Microsoft, China Mobile, Alphabet, Comcast, NTT, IBM, Intel, Cisco, Amazon, Facebook, …

All tech companies that are hiring currently are networking companies

Note: Apple became highly valued only after it switched from computing to communications (iPhone)

Networking = Economic Indicator

Source: Raj Jain Keynote speech at SIGCOMM 2017
Networks foster innovations

• Google, Facebook, Internet of Things, online games, e-commerce, cloud computing

• Fun examples: test of time paper awards


     Ethane ushered in the age of Software-Defined Networking (SDN) and a new generation of research that inspired both academia and industry to design network control planes that we can reason about.

Meraki was founded by two MIT PhD students, Sanjit Biswas and John Bicket, along with Hans Robertson. The company was based in part on the MIT Roofnet project, an experimental 802.11b/g mesh network developed by the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology.

Meraki was funded by Google and Sequoia Capital. The organization started in Mountain View, California in 2006, and before relocating to San Francisco. Meraki employed people who worked on the MIT Roofnet project.[2][3][4]

In 2007, Meraki selected San Francisco for their community-based Free the Net campaign.[why?] They started putting gateway devices in the Lower Haight neighborhood to provide Internet access and giving away repeaters. In the first year of the project, the growth of the network was primarily in the Mission District. By October 2007, they estimated 20,000 distinct users connected and about 5 terabytes of data transferred in this network. In July 2008, Meraki said 100,000 people in San Francisco used its "Free the Net" service. Since then, Meraki discontinued this public service, though many access points remain active, but with no connection to the Internet.

On November 18, 2012, Cisco Systems announced it would acquire Meraki for an estimated $1.2 billion.[1]
• You could be the next!
Architectural questions tend to dominate CS networking research
Decomposition of Function

Definition and placement of function
- What to do, and where to do it

The “division of labor”
- Between the host, network, and management systems
- Across multiple concurrent protocols and mechanisms
Course overview: who should be taking this class

• Interested in computer networks
• Has undergraduate level networking knowledge
  – Taken 356 or equivalent
List of topics

• The original Internet design
• Congestion control
• Routing
• Software defined networking
• Datacenter networks
• Network Function virtualization
• Programmable switches
• Fault diagnosis
• Remote direct memory access
• Measurement
• ...

Approach

• Materials
  – Read research papers
  – Come to lecture notes, available online

• Prerequisites:
  – An undergraduate-level networking course
  – Basic system programming experience
Course overview: your work

• Readings
  – Read papers before class (ideally)

• Projects (more later)
  – An assigned programming project to reproduce the results from a research paper
  – A self-chosen project that reproduces the results from a research paper
    • Talk to me if you want to do original research
  – Code, report, and presentation

• Two in-class midterms
  – Oct 4
  – Nov 20
Course overview: collaboration policy

• Encouraged to work with each other

• Team work: projects
  – 1~3 persons per team; recommended size is 2
  – Turn in one copy of code and report
Course overview: late and grading policy

• **Late**
  – The deadline for any assignment can be extended with a 10% penalty per day.
  – No deadline can be extended by more than two days. Assignments will NOT be accepted 48 hours after the due date.
  – The project presentations must be given on the day they are scheduled.
  – If you are ill: Contact the instructor and get a medical note.

• **Grading Policy**
  – Exams 50%
  – Project 50%
Course staff

• Instructor
  – Xiaowei Yang (xwy@cs.duke.edu)
  – Office hours: TuTh 2:50-3:50pm

• TAs
  – Zhenyu Zhou (zzy@cs.duke.edu)
    • Office hours: Wed 4-6pm @ LSRC D305
  – Shengbao Zheng (szheng@cs.duke.edu)
    • Office hours: Mon 4-6pm @ LSRC D330
The Design Philosophy of the DARPA Internet Protocols

David D. Clark
Massachusetts Institute of Technology
Laboratory for Computer Science
Cambridge, MA, 02139

(Originally published in Proc. SIGCOMM '88, Computer Communication Review Vol. 18, No. 4,
August 1988, pp. 106–114)

Abstract

The Internet protocol suite, TCP/IP, was first proposed fifteen years ago. It was developed by the Defense Advanced Research Projects Agency (DARPA), and has been used widely in military and commercial systems. While there have been papers and specifications that describe how the protocols work, it is sometimes difficult to deduce from these why the protocol is as it is. For example, the Internet protocol is based on a connectionless or datagram mode of service. The motivation for this has been greatly misunderstood. This paper attempts to capture some of the early reasoning which shaped the Internet protocols.

1. Introduction

For the last 15 years, the Advanced Research Projects Agency of the U.S. Department of Defense has been developing a suite of protocols for networking. This suite includes the Internet Protocol (IP), the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP), as well as many other protocols. One of the key goals of this development has been to create a protocol architecture that is flexible enough to accommodate a wide range of applications, while at the same time providing a reliable and efficient means of communication. In the early days of this work, there was some question as to whether it would be possible to design a protocol architecture into the IP and TCP layers. This seems basic to the design, but was also not a part of the original proposal. These changes in the Internet design arose through the repeated pattern of implementation and testing that occurred before the standards were set.

The Internet architecture is still evolving. Sometimes a new extension challenges one of the design principles, but in any case an understanding of the history of the design provides a necessary context for current design extensions. The connectionless configuration of ISO protocols has also been colored by the history of the Internet suite, so an understanding of the Internet design philosophy may be helpful to those working with ISO.

This paper catalogs one view of the original objectives of the Internet architecture, and discusses the relation between these goals and the important factors in the design of protocols.
David D. Clark (MIT)

• Chief Protocol Architect for the Internet from 1981.
• Continues to be a network visionary today.
• My PhD advisor 😊
• At the time of writing (1987)…
  – (Almost) no commercial Internet
  – Number of hosts reaches 10,000
  – NSFNET backbone 1 year old; 1.5Mb/s
  – 1 yr after Cisco’s 1st product, IETF started
How to Read*

You May Think You Already Know How To Read, But…

* Paper listed on class website. Some slides borrowed from Prof. Rexford's lecture.
You Spend a Lot of Time Reading

• Reading for grad classes
• Reviewing conference submissions
• Giving colleagues feedback
• Keeping up with your field
• Staying broadly educated
• Transitioning into a new area
• Learning how to write better papers 😊

It is worthwhile to learn to read effectively
Keshav’s Three-Pass Approach: Step 1

• A ten-minute scan to get the general idea
  – Title, abstract, and introduction
  – Section and subsection titles
  – Conclusion
  – Bibliography

• What to learn: the five C’s
  – Category: What type of paper is it?
  – Context: What body of work does it relate to?
  – Correctness: Do the assumptions seem valid?
  – Contributions: What are the main research contributions?
  – Clarity: Is the paper well-written?

• Decide whether to read further…
Keshav’s Three-Pass Approach: Step 2

• A more careful, one-hour reading
  – Read with greater care, but ignore details like proofs
  – Figures, diagrams, and illustrations
  – Mark relevant references for later reading

• Grasp the content of the paper
  – Be able to summarize the main idea
  – Identify whether you can (or should) fully understand

• Decide whether to
  – Abandon reading in greater depth
  – Read background material before proceeding further
  – Persevere and continue for a third pass
Keshav’s Three-Pass Approach: Step 3

• Several-hour virtual re-implementation of the work
  – Making the same assumptions, recreate the work
  – Identify the paper’s innovations and its failings
  – Identify and challenge every assumption
  – Think how you would present the ideas yourself
  – Jot down ideas for future work

• When should you read this carefully?
  – Reviewing for a conference or journal
  – Giving colleagues feedback on a paper
  – Understanding a paper closely related to your research
  – Deeply understanding a classic paper in the field
Other Tips for Reading Papers

• Read at the right level for what you need
  – “Work smarter, not harder”

• Read at the right time of day
  – When you are fresh, not sleepy

• Read in the right place
  – Where you are not distracted, and have enough time

• Read actively
  – With a purpose (what is your goal?)
  – With a pen or computer to take notes

• Read critically
  – Think, question, challenge, critique, …
Doing a literature survey

• Reading tens of papers in an unfamiliar field

1. Search keyword in Google Scholar or CiteSeer or ask your advisor
   – If you find a survey paper, you are done

2. Find shared citations and repeated author names
   – Go to key authors’ websites to find recent papers

3. Go to top conferences to find related work
Today

• Course Overview

• Last 15-20 minutes:
  – Talk to each other to find potential group partners