Computer Networks Basics

Robert H. Sloan

A few of these slides are taken from Introduction to Computing with Media Computation slides by Mark Guzdial, Georgia Institute of Technology, 2003–2004.

Networks: Two or more computers communicating

- Networks are formed when distinct computers communicate via some mechanism.
  - Rarely does the communication take the place of 0/1 voltages over a wire.
    - Too hard to make work over distances
  - More common is the use of frequencies (maybe in the sound range, but maybe not).
  - For example, a *modem* (modulator-demodulator) takes your computer's 0s and 1s and translates them into sound frequencies that can pass over the sound wire and be decoded on the other side.

Networks, networks everywhere

- If you’re driving a newer car, you probably have a network in there.
  - There are lots of computers in your car (controlling air flow, gas flow; making the air bag work) and they communicate.
- You can have a network in your own home, or even on an airplane.
  - Can use radio signals for communication (wireless)
  - Or can string a cable between two computers.

Networks have layers

- Networks have several layers to them.
  - At the bottom level is the physical substrate.
    - What are the signals being passed on?
  - Levels higher determine how data is encoded.
    - Do we use sound frequencies to represent 0s and 1s, or radio waves?
    - Do we send a bit at a time? A byte at a time? Or in *packets* larger than that?
  - Levels even higher determine the *protocol* of communication.
    - How do I *address* a particular computer I want to talk to? Or many computers?
    - How do I tell a computer that I want to talk to it? That I'm starting to send it data? What it's supposed to do with it? When we're done?

Internet: A collection of networks

- The Internet is a network of networks.
- If you put a device in your home so that your computers can talk to one another, you have a network.
  - A wireless base station, or an Ethernet *router*, perhaps.
  - You can probably reach printers on your network, or copy files between computers.
- If you now connect your network (through an *Internet Service Provider (ISP))* to the global Internet, your network becomes yet another part of the whole Internet.

Internet is based on agreements on encodings

- The Internet is built on a set of agreements about:
  - How computers will be addressed
    - A set of four numbers (each one byte now, soon to grow) separated by periods, e.g., 10.1.0.5.
    - A way of associating *domain names* with these numbers, like www.cnn.com (which really is a name that resolves to a set of four numbers), using *domain name servers.*
  - How computers will communicate
    - That data will be put into packets with various pieces in them.
    - That computers will format their data and talk to one another using *TCP/IP*
  - How packets are routed around the network to find their destination.

The Internet is not new

- The Internet agreements date back 40 years.
- It was originally set up for military applications.
  - One of the features of the Internet is that packets find their destination even if part of the Internet is destroyed, damaged, or subject to censorship.
- The Internet originally had only a handful of computers (*nodes*) on it, but it has grown dramatically in recent years.

Internet ≠ Web! (nor email)

- 1st four nodes of what would become the Internet went live October–December 1969; I started using it regularly in 1985 (still under 1,000 computers on Internet then!)
- World-wide Web became publically available in August 1991; I started using it in late 1994.
- Today, web and email are probably the two most popular applications or services on the Internet
  - In terms of numbers of people. File sharing might win in terms of amount of traffic.

Uh, Professor, so. . . .

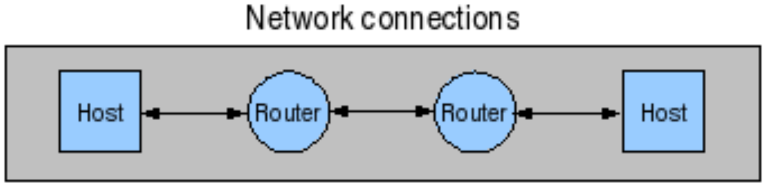
- What's the difference between the Web and the Internet anyway?
- Internet is the (mostly hardware) infrastructure.
- Crude analogy: Internet is like Cable company's wires, etc., web is a popular station.
- Mildly better: Internet is Cable company's wires, etc.
  - Web is set of TV stations—one service they provide
  - But just as cable may also provide pay-per-view movies, on-demand special events, some "radio stations," maybe Netflix streaming, Internet also has lots of other services.

About email

- Most providers today are willing to provide your email either
  - As a web page, that you will view in a web browser such as Firefox or Internet explorer. (At technology level, using the HTTP or just possibly HTTPS protocol.)
  - To an email reading program, such as Outlook, Outlook Express, Thunderbird, or Apple Mail. (Using either the POP or IMAP protocol to deliver to you, and the SMTP protocol for mail you send.)
- More than reasonable one way to do things, and many consumers like doing things via Web.

Some basic terminology

- **Network** is vast collection of **nodes** (**computers**) connected by bidirectional **links** (think wires, though could be fiber-optic, or earth-satellite link)
- Your computer/your home network is one node, connected to a single link to your ISP
- Crudely end-user nodes (you, cnn.com) are at edge with 1 connection; nodes in middle are **routers** (small electronic box = very dumb special-purpose computer)
- Packets flow from one end node, from router to router, to another end node.



Circuit switching

- **Circuit Switching Network** = dedicated **circuit** (or **channel**) established between two end points before the two principles can communicate.
  - E.g., early telphone exchanges, with the operator with the plug board at the town central exchange completing the connection.
  - Land line phone system in general
- *This is **not** how the Internet operates!*

Packet switching

- Traffic is split into chunks, called **packets** that are routed over shared network
- Each packet has header with at least the destination address, and may be routed independently
  - I.e., take its own distinct path along the network's nodes

Security: networked systems are different

- Network traffic is not subject to physical security, unlike web server or a desktop
  - Attackers can potentially see, modify, remove your traffic
- Multiple organizations by definition
  - Hence always issues of organizational trust

- And Internet, meaning the protocols or rules that govern its workings, was developed into nearly its current form by early to mid-1980s, when most of today's security concerns did not exist.

End-to-end principle

- Internet very unlike earlier communications networks: **intelligence is at the edges**
  - Smart computers are at edges; computers in middle of network, called **routers** are dumb, cheap special-purpose computers
- **End-to-end principle** says whenever there is a choice about where to handle some aspect of the communication, put it at the endpoint, not in the middle.
  - Put forward long ago by Internet techies as an engineering rule-of-thumb (because you probably need to do/check at endpoint anyway).

End-to-end principle as policy

- Edge computers more likely to know what user wants
- More edge computers than routers; more computing power
- Innovation typically easier at edge—Just need to get the change on computer providing the new thing and the computers of those who want it, not on every router
- Harder for network to discriminate among applications if middle treats all packets equally.

“Horizontal” view of internet

- Starting with your little home network on its wireless base station, traveling over one small connection to, e.g., your phone or cable company, and eventually onto very high-capacity transcontinental links owned by, e.g., Verizon
- As recently as early 1990s, federal government was still paying for the heart of the U.S. connections, and there were classifications that were very well defined.
- Today, tend to classify ISPs on whether they pay anybody to carry traffic for them and whether any ISP pays them to carry traffic

(Vertical) Layered View of Network

- Conceptually, networks are designed in layers. “Abstraction”
- In doing analysis and design at one level, you need pay attention only to current level, and a fixed small number of features of layer immediately below.
- 7, 5, and 4 layer models of networks all popular.
- 7 layer model is called Open Systems Interconnection (OSI), and was product of standards orgs and telecomms in early days of Internet, and has never really died, but isn’t really how people think either.

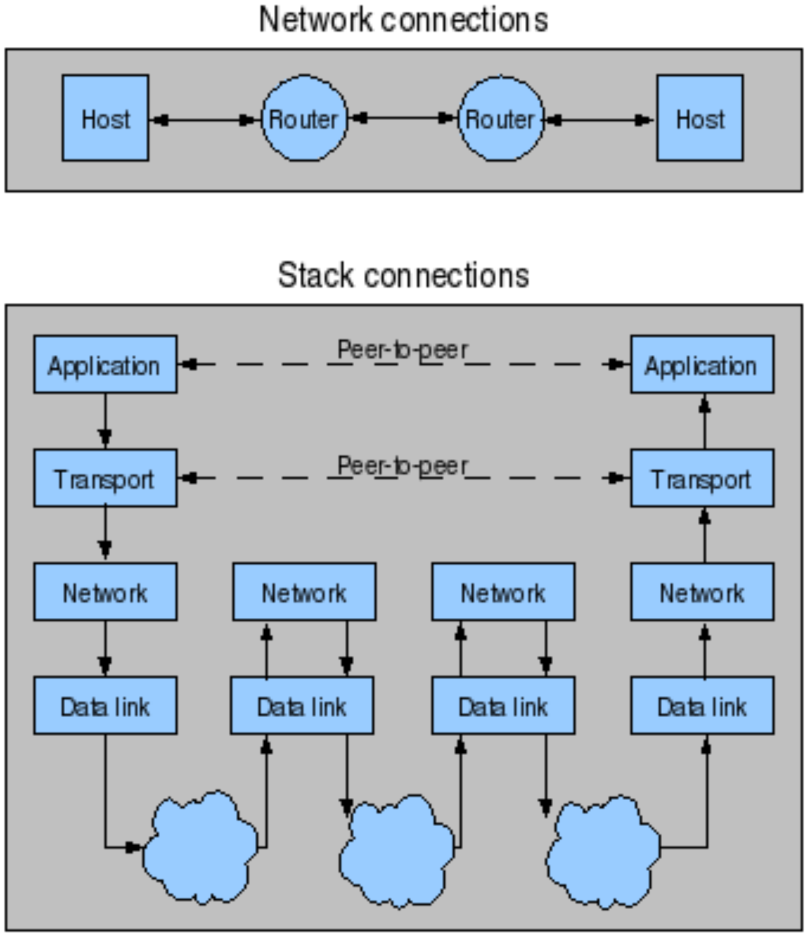
4-layer model of networking

- Bottom to top: Link, Internet, Transport, Application
- "Bottom" layer, link layer (split into two in 5-layer models) concerns electrical and computer engineers, and not really of interest to us.
- Incidentally, all of how the Internet works is specified in documents charmingly called "Request for Comments (RFCs)", that are publically available.
- The 4-layer Internet model is specified (in 10 million undergraduate textbooks and) RFC 1122 from 1989

4-layer model, top to bottom

- **Application** file sharing, email, web, stuff we actually want to get done. Lives only at endpoint computers.
- **Transport** Connections between process on one computer and process on second computer. Lives only at endpoint computers
- **Internet** Internet Protocol is run by every computer connected to Internet. Best-effort packet-switched delivery of packets
- **Link** Variety of hardware, province of electrical and computer engineers

Layers: Diagram



What's a protocol?

- We have two entities communicated in an automated way.
- Protocol is set of formal rules that specifies, as a function of what has been said so far and what an entity wants to say:
  - State of each entity:
    - Who is listening/receiving and who is talking/sending
    - Messages that can currently be sent
  - How state of entity changes as message is recieved

Internet Layer: Internet Protocol (IP)

- The narrow waist of the Internet's **hourglass architecture:** all devices on Internet must run this protocol
- Packet-switched, packet oriented protocol. Here packets referred to as **datagrams**—a packet that may or may not be delivered, with no notice of delivery/failure
- IP is best-effort, connectionless communication
  - US Mail analogy: drop it in mailbox, hope it gets there

IP addresses

- Each datagram has its own individual source and destination *IP address (aka IP number)*
  - There is no such thing as a connection!
- IP address: www.cnn.com = 157.166.224.25
  - Unique one for each computer on Internet

IP: Routing, etc.

- At IP level, each node finds best link to send datagram on to get it towards its final destination
  - I.e., network self-configures to find "best" route; routes constantly changing
- Any given router may get too much traffic to handle, in which case it simply throws away the datagrams it cannot handle.
- Also, can split datagrams (each is labeled with total original size and offset) because different networks on the Internet have different size limits

About those IP numbers

- Provide **unique** global addressing of computers on Internet to ensure that any two computers communicating can identify each other.
- Current version of IP, IPv4, provides "only" 4 billion IP addresses (4 byte addresses). We will run out.
- Transition very slowly starting to IPv6, which provides enough
- Would have run out by now, but since 2000, NAT: most homes have one (dynamic) IP address for the home, not one per device.

Internet layer security issues

- IP is early 1970s protocol, predates security concerns
- Datagrams have source address, but no verification
- Easy for somebody mildly savvy to forge
  - So don't know where IP datagrams really come from

Transport layer

- Most popular protocol at this layer is Transmission Control Protocol (TCP)
- #2 is UDP
- Protocols exist only at edge computers (*hosts*); not at routers
- Implemented by using IP
- At this level we have concept of
  - Connection
  - And connection is process-to-process (e.g., web browser on my office machine to web server on www.findlaw.com); IP is merely host-to-host

TCP

- TCP adds source and destination **ports:** Number that extends host name way extension extends phone number
- **Jargon: Socket** = host IP number + port
- TCP establishes a socket-to-socket connection
- TCP *uses* IP to send the packets
- TCP adds reliability: part of protocol is receiver acknowledging; sender resends if no ACK
- Also simple protection against random transmission errors (checksums), some congestion control

User Datagram Protocol: UPD

- Also built on top of IP
- No acknowledgement of receipt.
- Therefore generally less appropriate, but has two advantages:
  - Allows for one-many broadcast (TCP only one-to-one)
  - Less overhead, so if you want datagrams to arrive fast even if some are lost, a good choice

Transport layer security issues

- TCP was also designed in 1970s (and UPD in 1970s and very early 1980s) without any security
- Any host could request connection to any other host
- In 1970s, internetworking was over dedicated line between very expensive machines all in locked rooms

Application layer (top layer)

- Where we have all the protocols for the applications we actually want to run
- HTTP (web)
- IMAP, POP, SMTP (email)
- FTP, SSH, Telnet
- TLS/SSL, for encrypting traffic, about which a little more when we come to network security in a few weeks
- DNS, Domain Name service: translation between IP numbers and host names
- MANY more

Small example: HTTP protocol

- Web browsers (clients) & servers, communicate via HTTP
- Web **server** is waiting for **client** request in listening state
- Click of hyperlink in client-side browser starts opening of TCP connection between browser and web server
- Browser will use this connection to send HTTP message: “GET” along with URL of requested resource
- Server responds with a reply code, and, if request successful, the requested web page content.

Domain Name Service

- Every edge computer must run DNS to be able to use Internet
- User is going to type www.cnn.com, not 157.166.224.25, and soAndSo@gmail.com, not soAndSo@74.125.95.83
- So we have to have both directions of translation available
- This is what DNS does
- More later on this with network attacks, but the unsurprising punch line: it was designed a long time ago, and has been a rich source of security problems.

Your IP number

- Packets you send have it as from address
- Most home Internet service (AT&T DSL, Comcast, etc.) provides IP number that may change.
- In practice, stays the same for days to many weeks.
- Web site, e.g., Google, can know request is coming from same place
- Also, your ISP keeps records of who had what ISP what day, which can be subpoenaed.
- IP numbers assigned in blocks; usually can deduce geographic region; sometimes fairly precise.

Summary thoughts on security

- Internet provides packet-switched connections between any two edge computers
- Connections can originate anywhere in the world
- Traffic can potentially be seen, modified, or removed by attackers
- IP datagram source and destination addresses are world readable (and cannot really be encrypted—Why??)
- IP protocol and Transport Layer protocols are not at all security minded

Coda: Who can read your email?

- Realistically, who can read your email?
  - Snoopers close enough to get signals from your home wireless network?
  - Your ISP?
  - Other ISP?
  - Adversary tapping line somewhere (where?!) with resources of:
    - US NSA?
    - Fortune 500 company?
    - 500-employee company?

Transition: Web sites know you

- Google has a record of all the search queries you have made, perhaps forever, though they (attempt to?) make them anonymous after 18 months
- Useful to me:
  - So, e.g., Google Maps knows locations I like to map
  - Perhaps helps improve search results (Is "spell" witchcraft or spelling bees for this user)
  - Aggregate data surely helps Google improve its search results

HOW do web sites know you?

1. You log in (maybe have set it to be automatic)
   - I automatically log in to amazon, Google, NY Times. And, e.g., Google Maps knows locations I am likely to want to map
2. They record your IP number
   - Typical home IP number is assigned with DHCP, where D = Dynamic, but typically unchanging for weeks to months
3. Cookies

Cookies

- Small piece of text that site you visit leaves in your computer, specifically your browser's files.
  - Identifies you and your state, so web site can remember your information, and remember you.
  - Has expiration date.
  - Prior to late 2007 Google's used to be 2038; under pressure changed to 2 years from date it's set.
  - Though of my 50–100 Google cookies, found 1 still 2038 (scholar), four 2020.
  - Can set IE, Safari not to keep cookies at all (nuisance, impractical); can set Firefox not to accept cookies from certain sites