15213 Lecture 21: Network Programming Introduction

Introduction

This activity will motivate the structure of an interconnected network (internet). It will then introduce you to some common command-line tools, which you'll use to explore the global TCP/IP Internet's structure and protocols.

Learning Objectives

- Describe the structure of a LAN (Local-Area Network), and compare with that of an internet.
- Recognize the mapping between human-readable domain names and IP addresses via DNS.
- Compare and contrast domain names and IP addresses.
- Identify the properties of a connection (point-to-point, full-duplex, reliable).
- Explain the role of ports in a connection and the mapping between ports and services.
- Explain the role of the HTTP protocol in the functioning of a Web browser.
- Outline the steps of a client-server network transaction.

1 Model 1: Network Structure

Conventional postal delivery systems (aka snail mail) are designed to carry written messages and packages from place to place. Mail carriers collect letters from several houses and drop them off at local post offices. Before the postal system was unified nationwide, post offices delivered local mail and transferred mail directly to other post offices.

1. A Local Area Network (LAN) is composed of a series of hosts, connected by switches, which are connected by routers. Write the postal entity whose role most closely corresponds to the given computer network component in a LAN (there are multiple acceptable answers). A word bank is provided.

   Letter Mailbox House Mail carrier Postal relay box Local post office

Host	House
Hub	Mail carrier
Switch	Local post office

Today, post offices are linked by regional mail processing plants. Local post offices send their collected mail to mail processing plants, which then send the mail to one or more additional mail processing plants. The mail is then directed to the correct local post office and then given to a mail carrier for delivery.

2. An internet is composed of a series of LANs connected by one or more routers. Write the postal entity whose role most closely corresponds to the given computer network component in an internet (once again, there are multiple acceptable answers). A word bank is provided.

Letter Mailbox House Mail carrier Postal relay box
Local post office Mail processing plant

Host — House
Switch — Mail carrier or local post office
Router — Local post office or mail processing plant

We will now use the command line tool `traceroute` to see this structure in action. Log into a shark machine. First, try running `traceroute` on a domain in the same network.

```
shark> traceroute shark.ics.cs.cmu.edu
```

3. How many routers were encountered in the route to another shark machine? Why? No routers were encountered in the route to another shark machine, because the other host is in the same LAN as the current shark machine.

Now, try running `traceroute cs.cmu.edu`. You will see a printout of the intermediate routers between your computer (in this case, the shark machine you're logged in to) and the destination server.

4. How many intermediate routers are there now? What do you observe about the names of all the intermediate routers? Why might this be?
There are now 4 intermediate routers. The names of the intermediate routers all have CMU in them. This is because there are many networks within CMU, and the `cs.cmu.edu` network is not the same as the `shark.ics.cs.cmu.edu` network

2 Model 2: Network Protocol

Each host has a unique identifying *IP address*. IP addresses are mapped to a set of human-friendly *domain names*, which are maintained in a worldwide database called *DNS*, standing for Domain Name System. We will now explore DNS using a few common command line utilities.

The most basic way to find a domain name's IP address is through `nslookup`.

```
shark> nslookup cmu.edu
```

5. What is the IP address of `andrew.cmu.edu`? What about `cmu.edu`? What do you notice about these IP addresses?
The IP address of `andrew.cmu.edu` is 128.2.42.9, and the IP address of `cmu.edu` is 128.2.42.10. They are one apart. They also share the highest-order 24 bits with `cmu.edu`.

6. Here, `andrew.cmu.edu` is a *subdomain* of `cmu.edu`. Try finding the IP address of a few more subdomains of `cmu.edu`, such as `cs.cmu.edu`, `csd.cs.cmu.edu`, and `ece.cmu.edu`. What do the IP addresses of these subdomains have in common? Why might this be?
`cs.cmu.edu` has IP address 128.2.42.95, `csd.cs.cmu.edu` has IP address 128.2.42.98, and `ece.cmu.edu` has IP address 128.2.131.95. The IP addresses of the subdomains we've seen so far have the highest-order 16 bits in common, and the highest-order 24 bits in the case of `cs.cmu.edu` and its subdomain `csd.cs.cmu.edu`.

The IP addresses of subdomains often share the highest-order 16 or 24 bits with each other and the parent domain. This is because both IP addresses and domain names are **hierarchical**.

7. Does the most significant (*least* fine-grained) piece of information come first or last in a domain name? What about in an IP address?
   The most significant piece of information comes last in domain names; for example, in `csd.cs.cmu.edu`, `csd` is the least significant piece of information and `edu` is the most significant. In IP addresses, this order is reversed. The most significant piece of information comes first (128, in all of the IP addresses we have seen so far) and the least significant comes last.

8. *(Advanced)* You can use `host` to do a reverse DNS lookup, where you find the domain name given an IP address. Use `hostname` to get the domain name of the machine you're currently logged in to and `hostname -i` to get its IP address. Try typing

   ```
   shark> hostname -i
   ```

   then entering the result into `host [IP_ADDRESS]`. What do you observe?
   The correct shark domain name is returned from the IP address given by `hostname -i`.

9. *(Advanced)* Unfortunately, reverse DNS lookup is rarely so straightforward. Try using `host` to look up the domain names of the IP addresses you found of `cmu.edu` and its subdomains. You can also look up domain names of IP addresses neighboring those. What is different about the domain names returned by a reverse DNS lookup?
   The domain names returned by a reverse DNS lookup often are less descriptive and less recognizable. For example, instead of `cs.cmu.edu`, we see `SCS-WEB-LB.ANDREW.CMU.EDU`.

3 Model 3: Client-Server

We will now explore the `ping` tool, which sends messages (called packets) to a server under a given domain name and waits for a response. Type

```
shark> ping -c 5 cs.cmu.edu
```

This is a very basic example of a client-server transaction. The client (the shark machine) is sending a request to a server somewhere, which processes the request and sends a response.

10. `ping` prints the time taken for a response to be received. Try `ping`-ing the CS programs of CMU (`cs.cmu.edu`), MIT (`csail.mit.edu`), and the National University of Singapore (`comp.nus.edu.sg`). How do the ping response times compare?
    `cs.cmu.edu` has a very fast response time ($<$ 1 millisecond) because it resides in a neighboring network to the shark machines. Domestic domains are a bit slower, with around 10-100 milliseconds, and overseas domains can have response times as high as 300-500 milliseconds.

4 Model 4: Ports and Services

We have seen that IP addresses uniquely identify host machines.

11. Imagine that this were the most specific level of addressing. What limitations would this impose?
    The server would have no way of knowing (without receiving additional information from the client) which process or service an incoming connection request should be connected to. For example, a web server would not know whether to serve HTTP to an incoming request from a Web browser or ssh to a web admin's ssh request.

To address (no pun intended) this shortcoming, connection requests contain an additional port number appended to the IP address, which allows the client and server to route their requests to the appropriate service or process.

You can view a listing of services with their associated ports and protocols in the `/etc/services` file. Type

```
shark> head -n 100 /etc/services
```

12. Do you recognize any of the service names? What port number does `ssh` use? What about `http`? What are their associated protocols?
    You might recognize `ssh`, `http`, and `hostname`, and maybe even `ftp`, `lmtp`, and `kerberos`. `ssh` uses port 22 while `http` uses port 80. They both use the TCP and UDP protocols, and `http` uses the SCTP protocol in addition to those.

5 Model 5: Connection Properties

In this model, you will reinforce the properties of a network connection. You may be familiar with SMS messaging, also known as text messaging. Like a network connection, SMS messaging is **point-to-point**; that is, it connects a pair of recipients, rather than broadcasting to multiple recipients.

SMS messaging is also not necessarily **reliable**, which means that the sender receives no indication that messages are received, if they are received at all. Nor is SMS messaging particularly **ordered**; there is no guarantee that successive messages will be received in the order they were sent. However, network connections are reliable, meaning messages received are acknowledged by the recipient. Network connections are also ordered, meaning that messages will be received in the order they are sent.

Consider the following text message exchange, where Alice and Bob are discussing a time to meet. Alice sees:

```
Alice                                     Bob
Can we meet at 7:00 pm?
If not, I'm also available at 8:00 pm.
                                          Sure, that sounds good.
```

However, when Alice arrives at 8:00 pm, she finds that Bob had arrived at 7:00 pm and was waiting for her.

13. What went wrong? What does Bob's messaging history look like?
    Bob sent his message after Alice's first message and before her second, but due to the unreliable nature of SMS messaging, the cellular messaging service may have delivered Alice's second message before Bob's response arrived. Bob sees Alice's first message, then Bob's message, then Alice's second message. The order of Alice's messages could also have been reversed, or her second message could have been dropped.

Now consider the next text message exchange. Alice sees

```
Alice                                     Bob
I'm free at 7:00 pm.
We can meet at 8:00 pm?
                                          Let's do the first one.
```

This time, Alice arrives at 7:00 pm, but Bob shows up an hour later.

14. What went wrong here? What does Bob's messaging history look like?
    Here, due to the unordered nature of SMS messaging, Alice's messages were received out of order by Bob. He sees Alice's first two messages in reverse order, then his own message. Bob could have seen Alice's first message and not the second, but this is less likely.
15. *(Advanced)* You can use the `netstat` utility to see a printout of the current network connections, but it prints out a lot of information. Run (on a shark machine)

    ```
    shark> netstat --tcp
    ```

    to only see the connections using the TCP protocol. Can you pick out the `ssh` connection with your local host name? Run `netstat --tcp` (you may have to run `netstat -p tcp` instead) on your local machine. Can you pick out the `ssh` connection? What port is that process using on your local machine (the client), and what type of port is this?
    The specific port numbers may vary. On your local machine, this is an ephemeral port.

6 Model 6: HTTP Protocol

You will now use the `telnet` tool to explore sending HTTP requests to a Web server and receiving a response. Establish a connection to `cs.cmu.edu` on port 80 (the port associated with the HTTP service) by typing

```
shark> telnet cs.cmu.edu 80
```

Send an HTTP GET request to `cs.cmu.edu` by typing `GET /~213/index.html`.

16. What does the server response contain? What is this document?
    The server responds with a HTML document of the course homepage of 15-213.

This is an example of another client-server interaction, where the client is sending (via `telnet`) an HTTP GET request to the Web server of `cs.cmu.edu`, which processes the request and sends a response with the requested webpage.

17. How might you access the office hours page? What about the staff page?
    You would access the office hours page with the HTTP request `GET /~213/officehours.html` and the staff page with the request `GET /~213/staff.html`.

18. *(Advanced)* On the staff page, you'll find the paths to pictures of the course staff. Try using `telnet` to send an HTTP request for one of those images. What happened? Can you think of a reason why?
    Upon sending an HTTP request for an image, `telnet` printed a bunch of garbage characters and froze. This is because `telnet` is not particularly equipped for printing binary data such as images.