CS 458 — Module 4

1 Intro to Networks

- To create a network, you need 3 things:
  1. Devices able to receive and send signals
  2. A way to connect devices to each other
  3. Rules for communicating, or a protocol
- Some examples of protocols:
  - Token ring — a person can only talk if they have the token
  - CSMA/CD — all listen to the wire, if they hear no signal they try to transmit. If there is a collision, then they all stop and resend.
- Well what are some problems?
  - The Internet's design connects many computer networks together. It also assumes that participants are honest and will cooperate — they will not look at messages that don't belong to them, they will not delete your messages, etc. Everyone should mutually work together... right?
  - There's also no routing logic in the addressing scheme — given some IP address, who knows where it comes from? For example, a phone number has an area/country code. An IPV4 address like `136.192.63.0` could come from anywhere!
  - Nor can you control the path your message follows!
  - Your message can be broken up with each part following a different route.
  - There is no real hard stop limit to the number of nodes (at least everywhere).
  - It's really hard to conceptualize.
  - Nobody is in charge (both good and bad).

2 Daemons, Servers, Ports

- A server is a computer on a network to do tasks for other computers (clients).
- A daemon is like a servant that can only do one task within a server.
- We can think of a server like a huge apartment building, and each apartment can have one servant (daemon).
- For example, the mail sending daemon (SMTP) is 25.
- Some apartments (ports) can be empty. Many ports are actually empty!
- One could hide a service in a port it's not supposed to be in.
- For example, an HTTP daemon is in port 80. This is implied by default (ie: `https://www.uwaterloo.ca` implies `https://www.uwaterloo.ca:80`).
- But one could put a web service at, for example, port 8080.
- A "loose-lipped" system may reply to an attacker and advertise what services they are running *and* what at what port.