Week 3 Notes

Before we start, please complete the first part of the lab.

Review

Last week we hope to have gotten you guys somewhat comfortable with the UNIX environment, and you now have a basic feel for what it's like.
So, by now you should have:

- An OCF account. We will be using it today.
- Have completed Week 2 Lab, the solutions will be up soon.
- Know what basic UNIX commands mean, and be able to use them.
- Know how to add text and be able to edit a file using **vi**.

What is Internet?

It is a system of computer networks that transmit data by means of packets using one of the exisiting Internet protocols.

Today we will learn the basics of how Internet operates.

Packets

- All of the data being sent over the Internet is broken down into packets
- A packet usually contains a **header** which contains the necessary information to get the packet to the destination, a **data** which contains the information being sent, and a **trailer** which contains some sort of error-checking, to make sure that no errors occur during transmission.

Protocols

- Standard rules which data follows to effectively communicate
- Analogies could be drawn to Morse Code, or grammer.

OSI Model

- This is the model of protocols.

- 7th Layer - Applications Layer: **FTP** - File Transfer Protocol, **SMTP** - Simple Mail Transfer Protocol, **HTTP** - HyperText Transfer Protocol, **SSH** Secure Shell Procotol, **IPv4** - Internet Protocol version 4.
- 6th Layer - Presentation Layer: Makes App Layer not worrying about conversions
- 5th Layer - Session Layer: Establishes TCI/IP connections
- 4th Layer - Transport Layer: Provides for transfer of data between users. **TCP** - Transmission Control Protocol ensures complete data transfer
- 3rd Layer - Network Layer: Works with packets
- 2nd Layer - Data Link: Corrects errors that may occur in physical layer
- 1st Layer - Physical: Standards for wiring, voltages.

A few other major players:

DNS stands for Domain Name System.

- A system that keeps information about hostnames and domain names
- Translates domain names *e.g. www.google.com* into an ip address which allows to connect to the necessary server.

Why do we need DNS?

Imagine having to memorize ip addresses *e.g. 66.102.7.104* for everything!.
Let's take a look at the address of our class: http://www.ocf.berkeley.edu/sysadmin-class/

- **top-level domain**: edu Other example include *com, org, net, uk, ru*
- **subdomain**: berkeley.edu Other example *stanford, ucla, nyu*
- **hostname**: ocf Other examples *eecs, cs, ls, math, socrates*

So how does the Internet find the IP address I need?

- Each domain or subdomain has at least one authoritative DNS server which publishes information about the domain and name servers.

So, lets go through how www.ocf.berkeley.edu/sysadmin-class is found.

- Your browser asks asks a root server, say 198.41.0.4 (Verisign - company which operates 2 of the 13 root servers, and has control of domain names in the US) about where it can find www.ocf.berkeley.edu
- The root server will then reply with something like, I do not know the IP Address but I know that the following server knows it
- this continues until we get to one of OCF's root servers: 128.32.136.3

TCP/IP

- Is one of the first protocols defined that implements the Internet.
- Utlizes a handshake method

Three way handshake

1. You send a packet called SYN to establish communication
2. Yhe destination then sends you SYN/ACK packet
3. You send the ACK packet back acknowledging connection

DHCP

- stands for Dynamic Host Configuartion Protocol
- allocates ip addresses for the computer on the network
- three ways: manually - based on mac address, automatically - takes any free ip address, dynamically - for a certain period of time

NAT

- Stands for Network Address Translation
- Utilized by routers and firewalls
- Translates your privately assigned ip address into a public one