

Elijah Cordova
1425119
Lab 1 : CE 150

PRELAB:

1. \$ groups

2. The environmental variable \$ contains your PATH, which looks for executables in your system. For example, using echo \$SHELL yields the path of the current shell being used (bash, zsh, dash, etc.), or using \$(ls a) in a shell script allows you to use the contents of entry (ls a) in another command.

3. If you are running a process foregrounded in the shell, you can press [Ctrl + Z] to suspend the process and put into background.

Alternatively, you can use \$ kill [PID] to suspend a program, then a subsequent \$ bg command to place a recently suspended process in the background.

4.\$ uname -rn

or

\$ uname -r --nodename

flag -r will show your kernel version + release

flag -n, or --nodename will print the name of the network node

5. . refers to the current directory.

E.g. \$ cd /foo/

\$ ls .

will list all of the files in foo.

.. refers to the parent directory.

E.g. \$ cd /foo/bar/

\$ cd ..

will take me to directory foo.

~ refers to home directory of current user.

E.g. Logged in as user elijah running foo linux.

\$ cd ~

takes me to /home/elijah/

/ refers to root. This is the top level directory of the system.

6. pid is short for process ID. A unique process ID is assigned to every process when it is created. There are many ways to identify process IDs :

\$ htop or \$top is an easy way to see all sleeping, running, stopped or zombie tasks with their appropriate pid

\$ ps will show you all currently running processes with their pid on your computer

7. \$ cut -d: -f1,7 /etc/passwd

flag -d : -f1,7 uses ':' as a delimiter and selects column 1 and 7 to be printed.

as a result, cut prints out the name of the user, and the default shell they use separated by a :.

8. su allows you to drop into a new shell of any given user with their permissions, e.g.

users : elijah , root , foo

logged into elijah

\$ su foo

will log you into foo with a new shell with all the permissions of foo.

So, using \$su root switches user to root with a shell of the permissions of 'root'

sudo allows you, given you are given the correct permissions in /etc/sudoers, to run a process as root from another user. specifying sudo with flag -i allows you to drop into a new shell similar to ' su root '.By using \$ visudo , you can set the permissions for sudo with various options by editing /etc/sudoers, like:

- the ability to run sudo with or without a password for certain users
- the ability to run sudo with or without a password for certain users that are members of a specific group
- the ability to run sudo on certain programs only.

9. In *nix based systems. you can use a 'cron' software to schedule specific tasks to be scheduled into certain time intervals. There are many implementations of 'cron' softwares, like 'acron' or 'fcron'.

Cron software lets you schedule things yearly, monthly, weekly, daily, hourly, by the minute, etc. For example, I can set up a script that backs up a system, ./backup.sh, to run every week on monday at 1:00 and I would have weekly backups given that the computer is on at that time.

10. (see script)

LAB:

2. Each host is pinging all of the other hosts that it can reach : In this case, all of the hosts are connected to one switch, so it pings

all other hosts except for itself.

3. 'iperf' runs at 30 Gbits/second.

4a. There are 5 requests, and as a result there are 5 of _packet_in displayed on wireshark.

b. The source is '10.0.0.2' and the destination is '10.0.0.4' , which are the IP addresses assigned to the hosts.

using the filter does not yield any results using the filter 'of'.

c. There are 57 entries, meaning that there are ~14 entries per ping. The entries are either of 'request' or reply'.