Linux System Administration and IP Services
TERNET 2012

Linux Commands

Notes

* Commands preceded with "$" imply that you should execute the command as a general user - not as root.
* Commands preceded with "#" imply that you should be working as root with "sudo"
* Commands with more specific command lines (e.g. "RTR-GW>" or "mysql>") imply that you are executing commands on remote equipment, or within another program.

1. Log in as the sysadm user using ssh

    username: sysadm
    password: <given in class>

2. Become the root user

At the command prompt type the following command:

```
$ sudo -s
```

Enter your own password when prompted

Now that you are root the command prompt will change. We indicate this using the “#” symbol.

You are now the super user - be careful!

Ok, exit the root account:

```
# exit
$
```

3. Look at the network configuration of your host

```
$ cat /etc/network/interfaces
```

Notice that configuration of your host is done using DHCP.
"cat" is for "concatenate" and is one way to view what is in a file.

4. List files:

Use ls to list files:

```
$ cd                            [go to your home directory]
$ ls
```

Do you see anything? Try this instead:

```
$ ls -lah
```

What's inside one of these files?

```
$ cat .profile
$ less .profile
$ clear
```

Press “q” to get out of the less display.

If you don't understand what cat, clear or less do, then type:

```
$ man cat
$ man clear
$ man less
```

5. Working with the command prompt:

You can recall previous commands by using the up-arrow and down-arrow keys. Give this a try now.

Alternately, try typing this command:

```
$ history
```

If you wish to execute one of the commands in the list you saw type:

```
 $ !nn
```

Where “nn” is the number of the command in the history list. This is useful if you want to run a past command that was long and/or complicated.

Command completion:

With the bash shell you can auto-complete commands using the tab key. This means, if you type part of a command, once you have a unique string if you press the TAB key the command will complete. If you press the TAB key twice you'll see all your available options. Your instructor will demonstrate this, but give it a try by doing:

```
$ hist<TAB>
$ del<TAB><TAB>
$ rm <TAB><TAB>         [Include the space after the “rm”]
```

6. Working with pipes:

We saw an example of using pipes when we sorted the contents of our /sbin directory during the presentation. What if you wanted to have this information available in a file and sorted?

```
$ cd
$ ls /sbin | sort > sbin.txt
```

Now view the contents of what is in sbin.txt to verify that this worked.

```
$ less sbin.txt
```

Press the "q" key to quit viewing the contents.

7. Finding text strings:

Use the command grep to print lines matching a pattern in a data stream (such as a file). For example, view the entry for the sysadm account in the system passwd file:

```
$ grep sysadm /etc/passwd
```

You should see something like:

```
sysadm:x:1000:1000:System Administrator,,,:/home/sysadm:/bin/bash
```

The previous items above are:

```
userid:passwd:uid:gid:Name,extrastuff,,:HomeDir:LoginShell
```

grep is often used with a pipe to FILTER the output of commands. For instance:

```
$ history | grep ls
```

Will display your previous use of the ls command from exercise 2.

8. Editing the command line revisited:

It is particularly useful to realize that you can edit a command just as you would a line of text in a file. For instance, you can:

- Use your back-arrow (left) and forward-arrow (right) keys to change text in a command.
- Use the Home and End keys to go to the start and the end of a command:
  + ctrl-e = start
  + ctral-a = end
- NOTE: you do not need to go to the end of a command before pressing <ENTER> to execute the command.
- You can use the history command with grep to find a previous command. You can copy and paste this command, then edit it to make adjustments. For long commands this can save considerable time.
- To terminate a command without executing it press ctrl-c

- Alternatively you can use the reverse-search feature of bash:
 1.) Press <CTRL>-R
 2.) type the term you are searching for.
 3.) Press <CTRL>-R again to cycle through all occurrences of the term in your history.
 4.) Press the right or left-arrow, HOME or END key to start editing the command.

Let's give some of these editing rules a try:

```
$ ls -lah /usr/lib/ | grep postfix
```

Then, let's look for postfix

<CTRL>-R, type “postfix”, then press left arrow. Edit the previous command (which you should now have) and change “/usr/lib/” to “/usr/sbin/”. Use the left+right arrow key to move, and backspace to erase. You should now have:

```
$ ls -lah /usr/sbin/ | grep postfix
```

With your cursor just past the “/” in “/sbin/”, press <ENTER> to execute the command.