Class 3

Please practice the following commands and understand what they do.

**cat /etc/redhat-release –** the cat command is used to display text in a file, in this case the file is /etc/redhat-release – this shows the version of the operating system (tested on CentOS and RedHat):

**# cat /etc/redhat-release**



**passwd** (changes the password for the current user). You will be prompted to enter the password twice (you can’t see what you type). You could also do passwd user (where user is the username of the account that you want to change the password to).

**the “UP” arrow** – push the up arrow to re-type the previous command. This is very useful and avid Linux users use this all the time. You will save a lot of time by pushing the UP arrow to navigate through previous commands you already typed.

**history** (allows you to view list of commands that you previously used. You will see a numbered list of commands. By using !# (replace # by the line number) you can quickly execute that command.

**pwd** (stands for present working directory, this is used to check what directory you are currently under.

**cd** (navigate through directories, cd /home will get you in home which is a directory that is under / Remember that / is the very top directory of the hierarchy.

**cd ..** - gets you up one directory. If you are under /home and you type cd .. you get to /

**mkdir** (make directory) – example: mkdir docs (this creates a docs folder) or mkdir docs docs1 docs2 docs3 (this creates multiple folders at once)

**rmdir** (remove directory) – example rmdir docs (removes that folder).

**rm** (remove directory or file) - example rm filename (removes that file).

Extra notes for the rm command: you can use \* (the asterix symbol) to remove all files in a directory. Be very careful with this (don’t do rm \* in the / folder, that means you are erasing everything). The asterix symbol is used in conjunction with many other commands, not just rm. It is a filler for anything that could potentially be there. For example if I do ls \*.zip it will list all files with the .zip extension.

Example use of the asterix symbol:

[root@samir testfolder]# rm \*

rm: remove regular empty file `test'? y

rm: remove regular empty file `test2'? y

rm: remove regular empty file `test3'? y

The –rf flag makes it force to delete the files without having to type “y” or yes every time. The –r is recursive meaning it would erase subfolders and subfiles. So rm –rf testdirectory deletes everything in that folder and the subfolder. \*Remember this recursive trick because it’s useful when you want to delete more than just a given folder and you want to delete all subfolders and files at once.

**touch** – this creates an empty file

**man** - stands for manual, for example use man command to get a manual, push the ‘q’ key to get out. This is very useful when you want to know what a command does. Example: man rm. You may need to do “yum install man” in order to get this to work because the minimal install doesn’t have it. If yum is not working that’s ok, that is because your VM doesn’t have Internet access. We will get into network adapters later on.

**ls** (list directory contents)

-a flag lists all files that are normally hidden

-h presents the files in “human readable” format meaning with file sizes that can be read easily

-l shows long format – shows more information

**cp** - copy a file or directory – cp file1 file2 copies file2 and makes another file in the same directory named file2) – example: cp –r docs1 docs2 (copies docs1 folder into docs2).

**mv** - move directory, example mv directory1 directory2 will move directory 1 to directory 2.

**clear** - clears the content on the current screen.

**exit** - leaves the current SSH session.

**less** (less file1.txt writes the contents of a file onto the screen a page at a time – press the [space-bar] if you want to see another page and type [q] if you want to quit reading. As you can see, less is used in preference to cat for long files.

**head** (head file1.txt – this will show the first ten lines of a file to the scree, to head more you could use head -500 to head 500 lines for example)

**tail** (tail file1.txt – this will show the last ten lines of a file to the screen, to tail more you could use tail -500 to tail 500 lines for example)

**searching** a file for ‘file’

type: less file.txt

then still in less, type a forward slash [/] followed by the word to search (/example will search for the word example)

wc (word count)

wc –w file.txt counts words

wc –l file.txt counts # of lines

sort (this sorts a list alphabetically)

The **>** symbol is used to take the output into something else. For example I could do:

tail textfile > othertext file

That command will tail textfile and then save the result to another text file called othertext

sort < biglist will sort the list alphabetically but we need to output this content so we’d use:

sort < biglist > slist

**Pipes** or the **|** symbol can be used in Unix to combine different commands (outputs) and you can create some very powerful commands that way. Simple example: who (shows who was last, so if we mix that and wc to count the words & -l for the number of lines we get: who | wc –l)

**wildcard** (**\***) ls b\* for example would list everything that starts with a b (since we used colors, it shows black, brown) you can also use ls \*reen which listed green

If you are stuck editing something or tailing something, use CTRL-C or q to get out (one of the two)