CIS 90 Linux Lab Exercise

Lab 2: Using Commands
Fall 2018

Lab 2: Using Commands

The purpose of this lab is to explore command usage with the shell and miscellaneous UNIX commands.

Preparation
Everything you need to do this lab can be found in the Lesson 2 materials on the CIS 90 Calendar: http://simms-teach.com/cis90calendar.php. Review carefully all Lesson 2 slides, even those that may not have been covered in class.

Check the forum at: http://opus-ii.cis.cabrillo.edu/forum/ for any tips and updates related to this lab. The forum is also a good place to ask questions if you get stuck or help others.

Procedure

This lab must be done on Opus-II to get credit

Please log into the Opus-II server at opus-ii.cis.cabrillo.edu via port 2220. You will need to use the following commands in this lab.

apropos	cal	exit	info	type
banner	clear	finger	man	uname
bash	date	history	passwd	whatis
bc	echo	id	ps	who

Answer each of the questions below and be sure to issue all commands asked for. **It's not required that you turn these answers in.** Instead your command history will be scanned to verify you issued ALL the commands asked for below as well as any commands necessary to correctly answer the questions. Only your command history along with the three answers asked for by the submit script will be graded.

The Shell

1. There are a number of different command line shells that are available such as the Bourne shell (**sh**), the Bourne Again Shell (**bash**), the Korn shell (**ksh**), the C shell (**csh**), and many others.

    What shell are you currently using on Opus?

    _____

    What command did you use to determine this?

    _____

    (Hint: We did this in Lab 1)

2. Unix commands are located in multiple directories in the Unix file tree such as */bin*, */usr/bin, /sbin, /usr/sbin,* etc. The **type** command will search the directories on your path to locate commands. Use the name of the command you want to search for as an argument on the **type** command. For example, to locate the **ps** command use:

    ```
    /home/cis90/simben $ type ps
    ps is /bin/ps
    ```

    The first occurrence of the **ps** command was found in the */bin* directory. Now it's your turn. Use the **type** command and locate each of the following commands:

Command	Directory where command was located
man	_____
uname	_____
tryme	_____
echo	_____
type	_____
bogus	_____
bash	_____
scavenge	_____

    Issue this command:
    ```
    type ps tty ifconfig
    ```
    Does the **type** command accept multiple arguments? _____ (yes or no)

3. The **echo** command can be used to show the values of shell variables. For example, the PATH variable defines the directories and search order of your path. Benji can look at his PATH variable by using it as an argument on the echo command:

```
/home/cis90/simben $ echo $PATH
/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/cis90/sim
ben/../bin:/home/cis90/simben/bin:.
```

Note the $ in front of PATH. This instructs the shell to treat PATH as a variable instead of a text string. You can see that Benji has seven directories on his path each separated by a ":" character. The first directory is */usr/local/bin* and the last is the . (a single dot) directory.

Your turn now. Show the values of each of the following shell variables in your login session:

Variable	Value
PATH	_____
HOME	_____
TERM	_____
LOGNAME	_____
PWD	_____
PS1	_____
SHELL	_____

Issue this command:
```
echo $TERM $HOME $LOGNAME
```
Does the **echo** command accept multiple arguments? _____ (yes or no)

Issue the following **echo** commands:
```
echo $LOGNAME
echo LOGNAME
```
How does the shell interpret the **$** metacharacter?
_____

Issue this command:
```
echo $BOGUS
```
What happens on an **echo** command when a variable is used that does not exist?

_____

Issue this command:
```
echo I am $LOGNAME and I like the $SHELL shell
```
Are you getting the hang of using echo to show variable values now? _____ (yes or no)

4. Use the following to display your terminal type and your terminal device:
   ```
   echo $TERM
   tty
   ```
   Is your terminal type ($TERM) the same as your terminal device (**tty** output)?
   _____ (yes or no)

   Set the TERM variable to "dumb", show the new value, and execute the **clear** command.
   ```
   TERM="dumb"
   echo $TERM
   clear
   ```
   What happens? _____

   Change the terminal type to "vt100" or "ansi".
   ```
   TERM="ansi"
   echo $TERM
   clear
   ```
   What happens? _____

   Set the TERM environment variable back to what it was when you logged in.
   ```
   TERM="xterm"
   ```

5. What happens when you enter the following commands?
   ```
   DATE          _____
   Date          _____
   date          _____
   ```
   Why? _____

6. What results do you get from the command: **who -g**

   _____
   What program outputs the "invalid option" error message? _____

7. Enter each command below (exactly) and observe the results.
   ```
   echo one   two          threefour
   ```
   (be sure to include the extra spaces in the command above)
   Number of arguments in the command above is: _____

   ```
   echo "My TERM type is" $TERM
   ```
   Number of arguments in the command above is: _____

   ```
   echo one.two.three
   ```
   Number of arguments in the command above is: _____

8. Issue the following command:

```
echo red 'white
and blue'
```

What does the shell do when the closing quote mark is missing?

_____

Issue the following command (press Enter immediately after the backslash):

```
echo red white \
and blue
```

What does the shell do when you escape the invisible newline character at the end of the command line?

_____

9. Use the shell metacharacter "**;**" to write out a one line command that will clear the screen, print out the date and the current month's calendar:

_____

Commands

10. If you have not already done so, use the **passwd** command to change your password. Name three things you should never do with your password:

1._____

2._____

3._____

11. What **uname** command including options would output ONLY the *network node hostname,* the *kernel release* number and the *operating system*?

_____

Your command should produce the following output exactly:

```
opus-ii.cis.cabrillo.edu 3.10.0-862.9.1.el7.x86_64 GNU/Linux
```

Hint: Use the **man uname** command, scroll up/down and use **q** to quit.

12. What is the difference in output between the following two commands?

```
banner I am fine
banner "I am fine"
```

_____

13. simben90 is another user account on Opus. Use the **finger** command to find out what simben90's plan is. (Hint: Use simben90 as an argument on the **finger** command.)

    simben90's plan: _____

14. What is the UID (User ID) number for your own account?
    _____ (Hint: we did this in Lab 1)

Using online documentation

15. Issue a **man bc** command to read the bc manual page. Scroll up and down then use **q** to quit.

16. What is the **whatis** command? _____
    Use the **whatis** command with the argument: **bc**
    How does this compare to using the **man** command with **-f** option?
    man -f bc

    _____

17. Is **tryme** a UNIX command? _____ (yes or no)
    Hint: Use the documentation commands you know to find out.

18. What **who** command including options will output a count of the number of users logged on?

    _____

    Hint: Read the man (manual) page for the **who** command and experiment.

19. Run the command: **man -k boot**
    Use the manual pages to find out what the **-k** option does.
    What command is **man -k** equivalent to? _____
    Run the equivalent command and verify.

20. Run the command: **info bash**
    See if you can explore the hot links (marked with a *). Use the up and down arrows to select a link. Use Enter key to follow a link. Use **L** to go back to last page. Use **q** to quit.

21. Now use your web browser (outside of Opus) and google "linux bc command". If you find any interesting sites you can post them on the forum.

22. Here's a challenging task: Use the **man** command to discover how you can use the **bc** command to obtain the square root of 361. The **bc** command is an example of an interactive command, because you must enter the numbers to calculate from the keyboard while the program is running.

Submit your work

Now that you have finished this lab on Opus, you may submit your work using the following two commands:

```
history -a
submit
```

When the **submit** command asks you which assignment to submit, respond with **2** followed by the *Enter* key. Then answer the three questions.

You must submit your work to get credit. You can submit as many times as you wish up to the deadline. Only your last submittal will be graded. You can use the **verify** command to view what you submitted for grading.

Grading Rubric

- 27 points - For entering all the commands on Opus asked for in each step of the lab. The instructor will scan the commands in your history file and take off a point for any missing commands.
- 3 points - For correct answers to the three questions asked by the submit script (1 point each)
- 3 points - Extra credit (see below)

Remember, **late work is not accepted**. If you can't finish the lab before the deadline then submit what you have completed before the deadline for partial credit.

Extra Credit (1 point)

If you filled out the answers using this form then email the completed form to **risimms@cabrillo.edu** for 1 point extra credit. Be sure and cc: yourself to make sure you don't send me a blank form.

Even More Extra Credit (2 points)

For a small taste of what you would learn in CIS 191 let's add a new user to your Arya VM. Once added we will see how the new account is represented in */etc/passwd* and */etc/shadow*.

1. Log into your Arya VM as the cis90 user. Make sure it's your VM and not someone else's.

2. Install the latest updates:
   ```
   sudo apt-get update
   sudo apt-get upgrade
   ```

3. Add a new user account for yourself. You may make whatever username you wish. The example below shows how Benji would make the same username he uses on Opus:
   ```
   sudo useradd -G sudo -c "Benji Simms" -m -s /bin/bash simben90
   ```

4. Set a STRONG password for your new user. The example below is how Benji would do it:
   ```
   sudo passwd simben90
   ```

5. Exit back to Opus then log in again to your Arya using your new username and password.

6. View the *etc/passwd* file and find the new line added at the end. See if you can identify your UID, Group ID, home directory and shell in this line.
   ```
   cat /etc/passwd
   ```

7. Use the **id** command and confirm your UID and Group ID for your new account match the line in *etc/passwd*.

8. View the *etc/shadow* and find the new line added for your account. See if you can identify the encrypted version of your password.
   ```
   sudo cat /etc/shadow
   ```

Run the **submit** command on Opus to record information from the new lines in *etc/passwd* and *etc/shadow*.

Extra credit is due when the lab is due. **Remember, late work is not accepted.**