Dan Herve
CSCD 240
Steiner

Lab 2

1. Clearly explain why programs should be place in /bin and /sbin.

Programs should be placed in /bin and /sbin because they are then accessible to all users.

2. Capture the hostname of your linux computer

```
dherve@cscd-ceb202u04:~$ hostname
cscd-ceb202u04
```

1.

3. Capture a long list of all the files in your home directory including hidden files.

```
dherve@cscd-ceb202u04:~$ ls -al

total 476

drwx------ 29 dherve IT-GenericLinuxGroup   4096 Jun 30 12:57 .

drwxr-xr-x  3 root   root                      0 Jun 30 13:06 ..

drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 26 13:25 .adobe

-rw-------  1 dherve IT-GenericLinuxGroup   2128 Jun 26 14:40 .bash_history

-rw-r--r--  1 dherve IT-GenericLinuxGroup    220 Jan  8 16:30 .bash_logout

-rw-r--r--  1 dherve IT-GenericLinuxGroup   3486 Jan  8 16:30 .bashrc

drwx------ 10 dherve IT-GenericLinuxGroup   4096 Jun 30 12:55 .cache

drwx------ 12 dherve IT-GenericLinuxGroup   4096 Jun 30 12:57 .config

drwxr-xr-x  3 dherve IT-GenericLinuxGroup   4096 Jun 26 13:11 cscd240u14

drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .dbus

drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Desktop
```

```
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 26 14:40 Documents
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 26 12:07 Downloads
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 26 11:54 .fontconfig
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 30 12:55 .gconf
drwx------  5 dherve IT-GenericLinuxGroup   4096 Jun 26 12:19 .gnome2
drwx------  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .gnome2_private
-rw-r--r--  1 dherve IT-GenericLinuxGroup    182 Jun 30 12:55 .gtk-bookmarks
drwx------  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .gvfs
-rw-------  1 dherve IT-GenericLinuxGroup   1050 Jun 30 12:55 .ICEauthority
drwxr-xr-x  3 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .local
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 26 13:25 .macromedia
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .mission-control
drwx------  4 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .mozilla
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Music
lrwxrwxrwx  1 dherve IT-GenericLinuxGroup     14 Jun 23 19:21 netstorage ->
/mnt/ns-dherve
-rw-------  1 dherve IT-GenericLinuxGroup   4096 Jun 30 12:57
.nfs00000000033760060000001a
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .nv
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Pictures
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 26 11:55 .pki
-rw-r--r--  1 dherve IT-GenericLinuxGroup    675 Jan  8 16:30 .profile
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Public
```

```
drwx------  2 dherve IT-GenericLinuxGroup   4096 Jun 30 12:55 .pulse
-rw-------  1 dherve IT-GenericLinuxGroup    256 Jun 23 19:21 .pulse-cookie
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Templates
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 26 12:19 .thumbnails
drwx------  3 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 .uim.d
drwxr-xr-x  2 dherve IT-GenericLinuxGroup   4096 Jun 23 19:21 Videos
-rw-------  1 dherve IT-GenericLinuxGroup     59 Jun 30 12:55 .Xauthority
-rw-------  1 dherve IT-GenericLinuxGroup 330576 Jun 30 13:06 .xsession-errors
```

4. Capture all the following including creating a subdirectory from your cscd240 directory named lab2.

```
dherve@cscd-ceb202u04:~$ mkdir -v ~/cscd240u14/lab2/
mkdir: created directory `/home/EASTERN/dherve/cscd240u14/lab2/'
```

a. Complete a ls – l and capture the permissions for lab2

```
drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 30 13:29 lab2
```

b. Complete a ls –l lab2

```
dherve@cscd-ceb202u04:~/cscd240u14$ ls -l lab2
total 0
```

c. Explain the differences between a and b for question 4, meaning the output differences.

“a.” provides the permission for directories in ~/cscd240u14/ while “b.” generates of list of files and directories in ~/cscd240u14/lab2/

d. Is there a way to capture the permissions of lab2 without listing all the other information?

Explain your answer and execute the command if it exists.

There is no way to list only the permissions

e. Change the permissions on lab2 to remove executable from group and others using the letters ugo.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ chmod -v go-x .
mode of `.' changed from 0755 (rwxr-xr-x) to 0744 (rwxr--r--)
```

f. Change the permission on lab2 to add write permissions to the group using the octal values.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ chmod -v 744 .
mode of `.' changed from 0755 (rwxr-xr-x) to 0744 (rwxr—r--)
```

5. Find and describe a way to do a key-word search of the Linux man pages.

Ctrl+shft+f brings up a search window that allows the user to type in a keyword.

6. Explain the --help option for a program.

Shows command line options

7. Capture how to ask the ls command for help. (NOTE this is not man ls)

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ ls --help
```

8. Capture how to display the introduction section to any chapter in the Linux man pages.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ man -a intro
```

9. In class we discussed the use of the accent (back tick) for when it comes to executing the date
command. Can you use the accent on the pwd command? How would you use the accent on the
pwd command? Capture the usage of the accent on the pwd command.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ echo "I am visiting `pwd`"
I am visiting /home/EASTERN/dherve/cscd240u14/lab2
```

10. Capture the output of the stat command on the chmod executable. (Where does chmod live?)
Explain the information being displayed.

```
dherve@cscd-ceb208u10:/bin$ stat chmod
 File: `chmod'
Size: 55872       Blocks: 120       IO Block: 4096   regular file
Device: 806h/2054d    Inode: 425583     Links: 1
Access: (0755/-rwxr-xr-x)  Uid: (    0/    root)   Gid: (    0/    root)
Access: 2014-06-30 14:02:17.000000000 -0700
```

```
Modify: 2013-01-26 13:07:42.000000000 -0800
Change: 2013-12-04 09:32:44.000000000 -0800
Birth: -
```

chmod lives in the /bin/ folder.

File: – Absolute path name of the file.

- **Size**: – File size in bytes.
- **Blocks**: – Total number of blocks used by this file.
- **IO Block**: – IO block size for this file.
- **regular file** – Indicates the file type. This indicates that this is a regular file.
- **Device**: – Device number in hex and device number in decimal
- **Inode**: – Inode number is a unique number for each file which is used for the internal maintenance by the file system.
- **Links**: – Number of links to the file
- **Access**: Access specifier displayed in both octal and character format. Let us see explanation about both the format.
- **Uid**: – File owner's user id and user name are displayed.
- **Gid**: – File owner's group id and group name are displayed.
- **Access**: – Last access time of the file.
- **Modify**: – Last modification time of the file.
- **Change**: – Last change time of the inode data of that file.

Source: http://www.thegeekstuff.com/2009/07/unix-stat-command-how-to-identify-file-attributes/

11. Try and delete chmod. Did it delete why or why not?

```
dherve@cscd-ceb208u10:/bin$ rm chmod
rm: remove write-protected regular file `chmod'? y
rm: cannot remove `chmod': Permission denied
```

12. Capture the command to create test11, test12, test13, test33, stu11, stu12, stu22.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ touch test11, test12, test13, test33 stu11, stu12,           stu22
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ ls
stu11,  stu12,  stu22  test11,  test12,  test13,  test33
```

13. Using meta characters and a single ls command list all files named test.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ ls test*
test11,  test12,  test13,  test33
```

14. Using meta characters and a single ls command list only the files with the number 2 or 22 in them.

```
dherve@cscd-ceb208u10:~/cscd240u14/lab2$ ls *2*
stu12,  stu22  test12,
```

15. Using meta characters and a single ls command list only the files with a single 2 not 22 in them.

```
dherve@cslinux:~/cscd240u14/lab2$ ls *[?,0-1]2*
stu12,  test12,
```

16. In your home directory, capture the ls -l command piped to less, and the output from less.

```
        total 36
        drwxr-xr-x 4 dherve IT-GenericLinuxGroup 4096 Jun 30 13:29 cscd240u14
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Desktop
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 30 16:01 Documents
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 26 12:07 Downloads
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Music
        lrwxrwxrwx 1 dherve IT-GenericLinuxGroup   14 Jun 23 19:21 netstorage
-> /mnt/ns-dherve
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Pictures
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Public
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Templates
        drwxr-xr-x 2 dherve IT-GenericLinuxGroup 4096 Jun 23 19:21 Videos
```

17. Issue the which command on ls. Was and where was the command found?

```
dherve@cscd-ceb208u10:~$ which ls
/bin/ls
```

18. Issue the which command on set. Was the command found? If it was not found why not? How would you modify this.

It was not found. It is a bash command

19. Execute help set and explain the purpose of the set command in particular the –o option.

Allows the user to change options within the shell.

20. Issue a ls –al redirected (single >) to lsout.txt. What was the output and why?

lsout.txt was created and populated with a list of files and directories in the current directory. The redirect command moved the output from the terminal to a new file, lsout.txt.

21. Issue a pwd redirected (double >>) to lsout.txt. What was the output and why?

The present working directory was added to the end of the file lsout.txt. No output was given in the terminal. If only a singe redirect was used, it would have overwritten the ls data previously written.

22. Compare and contrast >> and >.

> redirects output to the specified location and overwrites any existing instance with the output. >> redirects output to the specified location and appends any existing output.

23. There are many other environment variables available to the user. Capture the printenv
command. Describe in detail SHELL and PATH

```
dherve@cslinux:/$ printenv
TERM=xterm
SHELL=/bin/bash
SSH_CLIENT=146.187.9.124 5638 22
SSH_TTY=/dev/pts/2
USER=dherve
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd
=40;33;01:cd=40;33;01:or=40;31;01:su=37;41:sg=30;43:ca=30;41:tw=30;42:
ow=34;42:st=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arj=01;31:*.taz=01;
31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.zip=01;31:*.z=01;31
:*.Z=01;31:*.dz=01;31:*.gz=01;31:*.lz=01;31:*.xz=01;31:*.bz2=01;31:*.bz=
01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=
01;31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.ace=01;31:*.zoo=
01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.jpg=01;35:*.jpeg=01;35:*.gif=0
1;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;35:*.tga=01;35:*.xb
m=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:*.svg=01;35:*.s
vgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.mpeg=01;
35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35:*.mp4=01;35:*.m
4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:
*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.
flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm
=01;35:*.emf=01;35:*.axv=01;35:*.anx=01;35:*.ogv=01;35:*.ogx=01;35:*.aa
c=00;36:*.au=00;36:*.flac=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.
mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.axa=00;36:
*.oga=00;36:*.spx=00;36:*.xspf=00;36:
```

```
MAIL=/var/mail/dherve
PATH=/usr/local/jdk1.8.0_05/bin:/usr/local/jdk1.8.0_05/jre/bin:/usr/local/sbin:/
usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games
PWD=/
LANG=en_US.UTF-8
KRB5CCNAME=FILE:/tmp/krb5cc_900724753_I20311
SHLVL=1
HOME=/home/EASTERN/dherve
LOGNAME=dherve
SSH_CONNECTION=146.187.9.124 5638 146.187.134.29 22
LESSOPEN=| /usr/bin/lesspipe %s
LESSCLOSE=/usr/bin/lesspipe %s %s
_=/usr/bin/printenv
OLDPWD=/home/EASTERN/dherve
```

24. Capture the command to print the current shell using echo.

```
        dherve@cslinux:/$ echo $SHELL
        /bin/bash
```

25. The current shell resides in /bin – capture a listing of the other shells that reside in /bin by using the *

```
        dherve@cslinux:/bin$ ls *sh
        bash  bsd-csh  csh  dash  ksh  rbash  rzsh  sh  static-sh  zsh
```

26. Capture the command to switch to a different shell

```
        dherve@cslinux:/bin$ csh
        %
```

a. Capture the command echo $SHELL.

```
        % echo $SHELL
        /bin/bash
```

b. What shell are you using? Why is the shell different than you expected?

I'm using csh. It's different in that it doesn't list my current location nor allow me to traverse my directories.

c. Capture the command to leave the different shell

```
        % exit
        dherve@cslinux:/bin$
```

27. Capture the output of the env command with no arguments.

```
dherve@cslinux:/bin$ env
TERM=xterm
SHELL=/bin/bash
SSH_CLIENT=146.187.9.124 5638 22
OLDPWD=/
SSH_TTY=/dev/pts/2
USER=dherve
LS_COLORS=rs=0:di=01
MAIL=/var/mail/dherve
PATH=/usr/local/jdk1.8.0_05/bin:/usr/local/jdk1.8.0_05/jre/bin:/usr/local/sbin:/
usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games
PWD=/bin
LANG=en_US.UTF-8
KRB5CCNAME=FILE:/tmp/krb5cc_900724753_I20311
SHLVL=1
HOME=/home/EASTERN/dherve
LOGNAME=dherve
SSH_CONNECTION=146.187.9.124 5638 146.187.134.29 22
LESSOPEN=| /usr/bin/lesspipe %s
LESSCLOSE=/usr/bin/lesspipe %s %s
_=/usr/bin/env
```

28. Compare and contrast env and printenv.

env has way less LS_COLORS and env lists OLDPWD=/ higher on the list

29. Show a shell command that will add the current directory (.) to the PATH (without removing any existing variables from the current value of PATH.)

```
    dherve@cslinux:/bin$ PATH=$PATH:.
    dherve@cslinux:/bin$ echo $PATH
    /usr/local/jdk1.8.0_05/bin:/usr/local/jdk1.8.0_05/jre/bin:/usr/local/sbin:/u
    sr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:.
```

30. Describe what you would have to do to make a change to the Shell permanent.

I would have to append the bash_profile file with the new path.

31. Capture the command help source and explain its meaning.

```
    dherve@cslinux:~/cscd240u14/lab1$ help source
    source: source filename [arguments]
    Execute commands from a file in the current shell.
```

Read and execute commands from FILENAME in the current shell. The entries in $PATH are used to find the directory containing FILENAME. If any ARGUMENTS are supplied, they become the positional parameters when FILENAME is executed.

Exit Status:
Returns the status of the last command executed in FILENAME; fails if FILENAME cannot be read.
source plus a filename will run any scripts in the file within the current shell. It will return the last command executed in the given file.

32. Explain how to make a Shell change permanent for all sessions including your current session.

(i.e. how do I reload my current session without closing and reopening)

You must go into the shell file (e.g. bash_profile) and make the changes there with a text editor (e.g. nano).

33. Explain double quotes, single quotes and the accent

double quotes: creates a weak string that still allows commands to be expanded (with $ or `)

single quotes: creates a strong string that doesn’t expand commands.

accent: putting accents around a command will expand it, even within double quotes.

34. Capture the output from the echo “Current time and date is `date`” command.

```
dherve@cslinux:~/cscd240u14/lab1$ echo "Current time and date is
`date`"
Current time and date is Tue Jul  1 20:58:50 PDT 2014
```

35. Issue the date command and capture its output. Now, capture the output from the echo 'Current time and date is `date`' command. Note that the ` character is an accent NOT an apostrophe ' . Explain why the output is different in particular to the single and double quotes. Also explain what the ` character does.

```
dherve@cslinux:~/cscd240u14/lab1$ date
Tue Jul  1 21:00:01 PDT 2014

dherve@cslinux:~/cscd240u14/lab1$ echo 'Current time and date is
`date`'
```

Current time and date is `date`

The double quotes allows commands surrounded with ` to be expanded while a single quote turns it into a string regardless.

36. Using man ln explain the difference between a hard link and a soft link.

With Hard links, the target of the link must exist. With soft links, the target need not exist, in which case the link is created but does not point to anything

37. Create a symbolic link called myLab that links to the lab2 directory. Capture the output

```
dherve@cslinux:~/cscd240u14/lab1$ ln --symbolic ~/cscd240u14/lab2/ mylab
dherve@cslinux:~/cscd240u14/lab1$ ls
files.zip  my.copy.bashrc  nothing   test   test21
file.tgz   mylab           something  test1  test3
```

38. Change to myLab and capture the output.

```
dherve@cslinux:~/cscd240u14/lab1$ cd mylab
dherve@cslinux:~/cscd240u14/lab1/mylab$ ls
lsout.txt  stu11,  stu12,  stu22  test11,  test12,  test13,  test33
```

39. Use "help" to get information on how to use the alias command.

a. What information is provided from “help”?

It tells you how to check existing aliases and how to create a new alias

b. When should you use “help” compared to when you should use “man”?

help is used for builtin commands while man is used for commands that are not built in

40. List two commands that can be found in both help and man.

exit and echo are listed in both

41. Create an alias named lsa that is ls –a. Capture the output and show it worked.

```
dherve@cslinux:~/cscd240u14/lab1/mylab$ alias lsa='ls -a'
dherve@cslinux:~/cscd240u14/lab1/mylab$ alias
```

```
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo
error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -alF'
alias ls='ls --color=auto'
alias lsa='ls -a'
```

42. Create a file named fa and fb – add some text to each file. Issue the command cat fa fb >| fa.

Explain what happened. What is the correct execution to append the file fb to the end of fa using the cat command?

```
cat: fa: input file is output file
```

```
dherve@cslinux:~/cscd240u14/lab1/mylab$ cat fb >> fa
```

43. Explain the difference between these two commands

ls chap[0-9] : will list any files/directories named chapn where n is {0-9}

ls chap{0..9} : will try to list 10 files/directories with names chap0 - chap9

44. Consider the two Bash initialization files: .bashrc and .bash_profile. Notice on cslinux

.bash_profile does not exist. What initialization commands should be kept in .bashrc and .bash_profile? Is a .bashrc or .bash_profile needed?

> .bash_profile contains default variables for the user when he logs on through the console while .bashrc contains the variables used when a new terminal window is opened. .bash_profile is for variables that apply to your whole session, such as programs that you want to start when you log in and environment variable definitions. .bashrc should contain variables that applies only to bash itself, such as alias and function definitions, shell options, and prompt settings. Neither one is necessary, as the system will default to .profile if the .bash files are not found.

45. Issue the command ls /bin. In class we discussed the history command, !!, and !some number.

> Explain !$ and where it might be useful.

!$ will take the last command of the history, print it, and try to run it. It could be useful if placed in quotation marks for variable substitution.