Lab 5: **The Linux Labs – Standard Linux Commands**

CSI4103 – Web Application Software Design

Faculty of Engineering – University of Ottawa

Objective:

Understand and practice how to use linux command line instructions. This is useful when working on server backends through a terminal window (e.g. using telnet or an SSH connection). This part of the lab reviews basic linux commands that are needed when navigating and using the file system from a linux command line

Instructions:

Linux is case sensitive; so that typing **CD**, **Cd**, **cD** or **cd** are all different. Generally, use all lower case for the standard commands. A command command is specified first and usually followed by arguments and/or options. Each command has its own rules about *syntax*, with each argument or option separated by a space.

Most of the Linux commands will follow the following standard structure:

```
command –options –other_options argument(arguments…)
```

Example: `ls -la --color=never /etc/`
where:

`ls` is the command
`-la` are two options
`--color=never` is also an option
and `/etc` is an argument

Linux files are organized within a hierarchical directory structure. We will work with two object types that exist within a directory structure: files and directories. Each file and each directory is accessed via a path, which can be specified as an absolute path or as a relative path. When working with files, Linux needs to know where files are located, exactly, within the file system. The path is the route to that exact location.

To explore the directory structure, the following commands will be used:

`cd` – change to a directory
`pwd` – print the current or working directory (all relative paths are defined from here)
`ls` – list directory contents
`mkdir` – create a directory
`rmdir` – remove a directory

cd and pwd

The cd (short for ***c**hange **d**irectory*) command allows you to navigate through the directory structure. The syntax of the cd command is:

```
cd directory
```

Simply typing **cd** will take you to your "home" directory. Typing in `cd directoryname` followed by the [**enter] key** will change your current or working directory to "`directoryname`".

pwd is short for **p**rint **w**orking **d**irectory

1. Type `man pwd`.
2. Read the `man` pages for `pwd`. What options does the `pwd` command take?

   _____

   (Type **q** to quit from `man`)
3. `cd` or `cd ~` – brings you to your home directory
4. pwd
5. `cd /` - brings you to the root directory
6. pwd
7. `cd /etc`
8. pwd
9. `cd ..` or `cd ../` – brings you up one level (starting with the current working directory)
10. pwd
11. `cd /home/user` – use your own user name, instead of 'user'
12. pwd
13. cd / -- go back to the root directory
14. `cd home/user` – use your own user name
15. pwd

    Q: *What is the difference between numbers 12 and 15?*

    _____

    _____
16. `cd /var/log; pwd`
17. `cd ../tmp`
18. `cd /` -- go back to the root directory
19. `cd bin -` This command uses a relative path. What command would you used if you wanted to use an absolute path?

    _____

ls **and** mkdir

One of the most basic commands in Linux is the ls or list directory command. The syntax is:

`ls [–options] [directory or file specification]`

There are several options available to allow you to display the contents of a directory in a format that suits your needs. Use the `man` or `info` pages to learn more. The most common options used are `–a` and `–l`. `–a` will display all files, including the hidden ones and `-l` displays extra file attributes such as dates and file permissions (and whether objects are files or directories).

`ls –al` [ENTER]

`ls –la` *`file_specification`* [ENTER]

The `mkdir` (short for make directory) command allows you to create a directory. The syntax is:

`mkdir` *`directory`*

1. Enter the following commands:

   `ls /bin/ls`

   `ls /home/user`

   `ls –a /home/user`

   `ls –la /home/user`

   `ls /ho` press the [TAB] key – the bash shell will fill in the rest of the name with tabbed auto-completion (can be useful with long names that are unique).

2. Press the [TAB] key a few times. What happens? _____

3. Press the up arrow twice. Note that the previous commands are recalled. The down arrow will "move" in the opposite direction.

   a. These commands are stored in a history file in your home directory. Can you find that file and list all of its contents?

   b. Note that you can re-execute a command (e.g. `!3` will execute the 3rd command in your history file and `!cat` will execute the previous command that started with `cat`)

4. In many situations, a directory listing might be longer than what fits on a single page. A few options are available, but try this one now:

   `ls –l /usr/bin | less`

This uses the "piping" capacity of Linux. Piping is the ability to direct or "pipe" the output of one command to the input of another command. In this case, we are piping the output of ***ls*** *to the input of* ***less.*** *The pipe symbol is the vertical bar and it is usually located above the* **** *key. Use the* ***q*** *key to quit from less. (For more fun, use the* ***/*** *key to search!)*

5. `cd`

   What is the purpose of the `cd` command without any arguments?

   _____

   `mkdir dir1 dir2` – this will create two directories

6. `ls`

   What is the output of this command?

   _____

7. `cd dir1`

   What is the purpose of this command?

   _____

   `ls`

   What is the output of this command?

   _____

   `mkdir subdir` – this will create another directory

8. `ls`

   What is the output of this command?

   _____

   `cd` – change back to your home directory

9. `mkdir parent/child`

   Record the error message:

   _____

   Explain why this command did not execute correctly.

   _____

Use the man or info pages. Record the correct command required to create parent/child.

_____

`rmdir` command

The `rmdir` or remove directory command allows you to remove a directory, but **only** if the directory is empty. The syntax for the `rmdir` command is:

`rmdir` *`directory_list`*

1. `cd`
2. `mkdir test`
3. `ls –l`

   What is the output of the command?

   _____

4. `rmdir test`
5. `ls –l`

   What is the output of the command?

   _____

   `cd dir1`

6. `rmdir dir2`

   Record the error message

   _____

   Explain why the command was not successful

   _____

   `rmdir ../dir2`

7. `cd ../dir2`

   Record the error message

   _____

   Explain why the command was not successful

   _____

   `cd`

8. `rmdir dir1`

   Record the error message

   _____

   Explain why the command was not successful

   _____

   `rmdir dir1/subdir`

9. `rmdir dir1`
10. `ls`

    Record the output:

    _____

    `rmdir parent/child parent`

    Does this command produce an error message?

    _____

Review exercise 1

Enter the commands below in your home directory in the order shown. Make sure that you are logged on as a regular user. After you have entered the commands, answer questions 1 – 6. Use the scrollbar of the terminal to review the input /output.

```
a. mkdir ~/lab
b. cd lab
c. mkdir ./tom
d. mkdir dick harry
e. rmdir ~/lab
f. rmdir harry
g. mkdir ~/course
h. cd ..
i. cd tom
j. cd lab/harry
k. rmdir ~/course
```

Answer these questions, based **only** on the above eleven commands, executed in sequence:

1. How many directories have you successfully deleted? _____
    a. List them by name:

    _____
2. How many directories in total have you created? _____
    a. List them using the absolute path:

    _____

    _____
3. How many directories are in the directory `lab`? _____
    a. List them using a path relative to the user's home directory:

    _____

    _____
4. How many error messages have you encountered? _____
    a. Record the error message along with the command letter (a-k):

    _____

    _____

    _____
5. Record the final working directory using the absolute path_____
6. Which command can you use to verify answer 5?_____

Review exercise 2

1. Change to your home directory.
    a. Record your command _____
2. Change to the `/etc` directory using a relative path.
    a. Record your command _____
3. Change to the `/etc` directory using an absolute path.
    a. Record your command _____
4. Display the contents of the `/etc` directory
    a. Record your command _____
5. Change to your home directory using an absolute path.
    a. Record your command _____
6. Display the contents of the `/etc` directory using a relative path
    a. Record your command _____
7. Display the contents of the `/etc` directory using an absolute path
    a. Record your command _____
8. Change to the `/var/log` directory using an absolute path.
    a. Record your command _____
9. Display the contents of the `/etc` directory using a relative path
    a. Record your command _____
10. Display the contents of the `/etc` directory using an absolute path
    a. Record your command _____