Wayne State University
CSC 4421: Computer Operating Systems: Lab
Winter 2017
Warmup: Intro to Linux/C
Instructor: David Warnke

Task 0

Start Ubuntu Linux, log in your account, and get familiar with the environment of this operating system.

Create a folder somewhere on your machine for the lab. In Linux you can use the GUI file browser, or use the following instructions.

- mkdir CSC4421
- mkdir CSC4421/Lab1
- cd CSC4421/Lab1

Task 1*

Open the terminal of Linux.

Navigate to the directory from Task 0. “cd CSC4421/Lab1”, or whatever your folder name is.

Try out the following Linux commands. For each command explain what it does (in one or two sentences). Put your answer in a file called Answer2task1.txt (Create in Windows or Ubuntu) (Don’t google it, you will get a chance to change the answer later).

Type the following lines and press enter after each one.

- who
- whoami
- uname
- ls
- mkdir Folder1
- cd Folder1
- ls
- cat /etc/os-release
- cp /etc/os-release ./
- ls
- cat os-release
- mv os-release newname.txt
- ls
- rm newname.txt
- ls

- cd ../
- ls
- rmdir Folder1 *(or)* rm –r Folder1
- ls
- ls -l

Task 2

To enable administrator rights, use sudo.
To install you can use apt-get. Type “sudo apt-get update; sudo apt-get install aptitude”
Now you can use aptitude instead of apt-get.
- note: “apt-get update”: updates the list pf available packages. Usually, it is recommended to update the list of packages, before installing a new one.

Choose an editor. Try vim, emacs, pico, nano, gedit, or anything you like.
Create a new text file with the editor of your choice named testfile.txt and add several lines of text to it.

Type the following commands
- cat testfile.txt
- less testfile.txt
- echo something
- echo something > redirect.txt
- history
- history >> redirect.txt
- ls –a ~/
- less ~/.bash_history

Try pressing up and down to see your previous commands.
Try typing “cat re” and then press tab.
Press Ctrl+r and start typing “echo so” and you will see the history command that you typed.

To copy and paste to and from the clipboard in the terminal use Ctrl+Shift+c and Ctrl+Shift+v.
- note: you have to add the shift key

Task 3

The man command is an extremely useful way to learn more about Linux commands (and also system calls). Type the command …

man man

… to learn a little more about the man command. It is probably more than you wanted to know. Keep hitting the space bar to get the next screen or the Enter to see next line. Look for the following sections as you go:

NAME
SYNOPSIS
DESCRIPTION

Task 4 *(modifies Task 1)

Were there any commands you had trouble with in Task 1? Use the man command to look up some of those commands. And then you can modify your answer in Answer2task1.txt.

Task 5 *

Now take a look at the man page for ls. Find the answers to the following questions and write them up.

1. What is the default sort order for ls?
2. What does “ls -a” do?
3. Is ”q” a legal option for ls?
4. Is ”R” a legal option for ls?
5. In what directory can “ls” usually be found? (The command “ls” is a file, which executes a program to do the ls command.)?

Put your answers in a file called Answer2task5.txt.

Task 6 *

Input the program below, using one of the Linux editors. Name your program cwd.cpp or cwd.c. You can copy/paste to/from the Linux terminal using ctrl+shift+c/v.

cwd.c

```
#include <limits.h>
#include <stdio.h>
#include <unistd.h>

#ifndef PATHMAX
#define PATHMAX 255
#endif

int main ( void) {
      char mycwd [PATHMAX];
      if ( getcwd (mycwd, PATHMAX) == NULL){
            perror ("Failed to get current working directory");
            return 1;
            }
      printf ("Current working directory : %s \n", mycwd);
      return 0;
}
```

Compile the program inputted in Task 6 using. “gcc cwd.c” or “g++ cwd.cpp”. If you don’t have g++ use “sudo apt-get install g++”.

This will create an executable called a.out. You can name the executable something else with -o as we do in task 7.

Run it to make sure it works. To run an executable file you need to write: “./[executable]” in this case it is “./a.out”.

Task 7 *

Create a makefile.

Input the following into a file named makefile. *Note: it must be called makefile.*

makefile

```
cwd: cwd.c
      gcc cwd.c -o cwd.out
```

Now type “make” followed by ls. You will see a new executable called cwd.out. Type “./cwd.out” to test it.

Task 8

Learn permissions and chmod (you can use the man page for chmod).

Task 9

To zip…
zip [archive_name].zip [file/directory_to_compress] (use -r for a directory)
tar -cf [archive_name].tar [file/directory_to_compress]

Notice that your current directory has to be outside the folder in order to run the command correctly.

To unzip…

unzip [zipname].zip
tar –xf [tarname].tar