

Introduction to COMP206

Lab A

The labs, for this course, are designed to be completed on your own at home. These labs are not graded. You do not hand in these labs. You can choose to do the labs on your own, or with the help of a TA during tutorial hours. You can see the tutorial hours on MyCourses (<https://mycourses2.mcgill.ca/d2l/le/news/462129/300257/view>). **Please note that, in order to take part in the tutorials, you will need to sign up for a tutorial time slot.** Please see the announcement on MyCourses.

This lab is about getting used to the School of Computer Science servers and the Linux command line.

Some labs will have a question zero. This question will not be covered by the TA during the tutorial. It represents extra content meant for you to do on your own.

At the end of this lab is a Mentoring section. The TA will use this during their tutorial section.

QUESTION ZERO: Optional problem

1. Let us learn a little about the Linux Manifesto.

Google “Linux Documentation Project Manifesto”. Summarize the manifesto in two to four sentences.

2. Popularity of development environments


There are two common environments that run software: the server and the personal device. Developers need to know where their software will be deployed in order to properly construct their algorithm so that it is sensitive to the deployed environment. Servers have different requirements compared to personal computers. In this course we will develop software for both personal computers and servers.

Look at the following two links and determine the popularity of Windows vs Apple vs Linux for (A) use as a personal computer and (B) use as a server.

The two links are: https://www.w3schools.com/browsers/browsers_os.asp and https://en.wikipedia.org/wiki/Usage_share_of_operating_systems

QUESTION ONE: Getting and using your CS user account

All assignments will be graded on the **mimi.cs.mcgill.ca** server. This means that the assignments you submit must be able to run on this machine. It is important that you establish the following procedure when submitting assignments:

1. Use the mimi.cs.mcgill.ca server for all assignments. Only this server will be accepted for assignments. The TAs will be using this server for grading. You must **ssh** into this machine to create your assignments; ssh is available for free on OS X (mac), Linux, and the latest versions of Windows. If you do not have ssh on your Windows computer, you can also download **putty**.
 - a. **HOWEVER**, you cannot log into this server using your first.last@mail.mcgill.ca account. Logging in with those credentials will redirect you to a McGill server where your files will be deleted as soon as you close your session, even if you logged in by typing
`ssh mimi.cs.mcgill.ca !!`
 - b. You must first create a computer science user account by doing the following:
 - i. Connect to McGill using the McGill VPN (instructions here: https://mcgill.service-now.com/itportal?id=kb_article&sysparm_article=KB0010687).
 - ii. Open your browser and type: <https://newuser.cs.mcgill.ca> and follow the instructions.
 - iii. If you wish to reset your password: <https://newpassword.cs.mcgill.ca>
 - iv. Disconnect the McGill VPN. **You only need the VPN to create the CS account. You do not need it to access mimi.cs.mcgill.ca for labs and assignments.**
2. Now try to login.
 - a. **OS X (mac)**: using the launchpad search for and launch "Terminal." In the terminal (command line), type: `ssh your_cs_username@mimi.cs.mcgill.ca`, then press enter. You will be prompted to enter your password. Type it in, then press enter. Note that nothing will happen when you type your password (no asterisks will appear); that is perfectly normal. If all goes well, you will see a welcome message and a prompt from the server.
 - b. **Windows (ssh)**: Open the start menu, then type "command". It should show you a list of search results with the top result: "command prompt." Press enter to launch it. (Alternatively, press and hold the windows key on your keyboard and press R, a small dialog box will open, type "cmd" and press enter.) Once the command prompt is open, type: `ssh your_cs_username@mimi.cs.mcgill.ca`, then press enter. You will be prompted to enter your password. Type it in, then press enter. Note that nothing will happen when you type your password (no asterisks will appear); that is perfectly normal. If all goes well, you will see a welcome message and a prompt from the server.
 - c. **Windows (putty)**: If your windows computer does not come with ssh, download the **putty** app. You can easily find it by Googling it. In the GUI, under hostname, enter "mimi.cs.mcgill.ca". Click "open". You will be prompted for your username and password. Note that nothing will happen when you type your password (no asterisks will appear); that is perfectly normal. If all goes well, you will see a welcome message and a prompt from the server.
3. Now, let us logout
Type `exit` to end your connection with the server. Type `exit` again to close the Terminal app (you may also need to click the  on the top left), the Windows command prompt, or the Linux terminal emulator.

QUESTION TWO: Backing up files

Try this experiment:

1. Login to **mimi.cs.mcgill.ca**
2. In your home directory `mkdir` a directory called **LabA**
3. Then `cd` into that directory and `mkdir` a subdirectory called **backup**
4. Using `cp` copy a random file from your home directory into LabA renaming it to **letter1.txt**. Select any file from your home directory to copy into the LabA subdirectory. Do not worry that you are changing the file's name within the LabA subdirectory. It will not affect the original file.
5. Using `cp` copy another random file into LabA renaming it to **letter2.txt**.
6. Using `touch` create a third file called **letter3.txt** in LabA.
7. Look at your files by typing `ls` and `cat`.
8. Using TAR archive your three text files into the directory backup. You will need to write something like this: `tar -cvf backup/myletters.tar *.txt`. This command takes all your text files (*.txt) and merges them into an archive file called myletters.tar within the subdirectory backup.
9. Using `ls`, look at the directory backup to see if your myletters.tar file exists. If it does, then you did it correctly.
10. Now, `cd` into the archive directory and extract the files. Check the class notes on how to do this. The files you archived should be extracted within the same directory (the **backup** directory). Once extracted they should appear within the directory.
11. If something unexpected occurred while archiving / extracting, discuss this with your classmates or the TA.

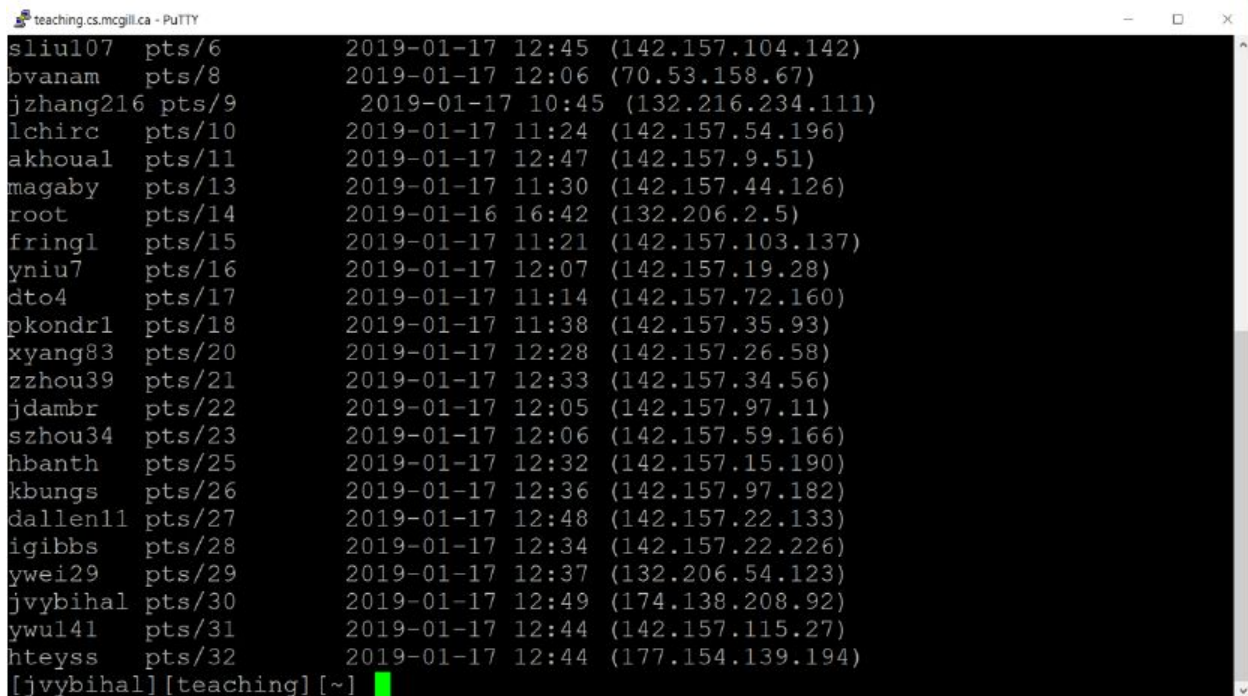
QUESTION THREE: Uploading and downloading files from a server

1. In order to transfer files to and from mimi, we are going to need a program that handles FTP (File Transfer Protocol). We recommend either:
 - a. **FileZilla**: available on OS X (mac), Windows, and Linux.
 - b. **WinSCP**: only available on Windows.
2. Once installed, launch your app and connect to mimi.
 - a. **FileZilla**: enter <sftp://mimi.cs.mcgill.ca> in the "Host" box, your username and password in their respective boxes, and leave the "Port" textbox empty. Click "Quickconnect" to connect to the server. If this fails, try entering 22 in the "Port" textbox. If you are asked to add a host key to a cache, accept.
 - b. **WinSCP**: enter mimi.cs.mcgill.ca under "Host name" and your username under "User name." Click "Save". Give it a name, click OK, and finally "Login." If you are asked to add a host key to a cache, accept. Enter your password when prompted.
3. You should now have a side-by-side view of your local folders (the folder on your machine) and the remote folders (the folders on the server, mimi). To upload a file, select a file from the left side and either drag-and-drop it over to the right side, or right-click and select upload.
4. Connect to mimi using ssh like you did in question two. Use the `ls` command to confirm that the file has been uploaded. Exit ssh and close the terminal.
5. Download the tar archive that you created in question two from mimi to your local machine. You can drag the file from the right and drop it on the left, or use the right-click menu and choose download.
6. Close your FTP program.

QUESTION FOUR: Interacting with other users

There are many ways to interact with users on a server. Let us look at one interesting way:

1. Login to mimi.cs.mcgill.ca
2. Find out the users currently logged in by typing the command: `who`
3. Use the `grep` command to find someone in the list of active users. You will need to type something like this: `who | grep 'abc'`
Where 'abc' is the string or substring of the username you want to find.
4. Notice that the command `who` displays something like this:




```
teaching.cs.mcgill.ca - PuTTY
sliul07 pts/6      2019-01-17 12:45 (142.157.104.142)
bvanam  pts/8      2019-01-17 12:06 (70.53.158.67)
jzhang216 pts/9      2019-01-17 10:45 (132.216.234.111)
lchirc  pts/10     2019-01-17 11:24 (142.157.54.196)
akhoual pts/11     2019-01-17 12:47 (142.157.9.51)
magaby  pts/13     2019-01-17 11:30 (142.157.44.126)
root    pts/14     2019-01-16 16:42 (132.206.2.5)
fringl  pts/15     2019-01-17 11:21 (142.157.103.137)
yniu7   pts/16     2019-01-17 12:07 (142.157.19.28)
dto4    pts/17     2019-01-17 11:14 (142.157.72.160)
pkondr1 pts/18     2019-01-17 11:38 (142.157.35.93)
xyang83 pts/20     2019-01-17 12:28 (142.157.26.58)
zzhou39 pts/21     2019-01-17 12:33 (142.157.34.56)
jdambr  pts/22     2019-01-17 12:05 (142.157.97.11)
szhou34 pts/23     2019-01-17 12:06 (142.157.59.166)
hbanth  pts/25     2019-01-17 12:32 (142.157.15.190)
kbungs  pts/26     2019-01-17 12:36 (142.157.97.182)
dallen11 pts/27     2019-01-17 12:48 (142.157.22.133)
igibbs  pts/28     2019-01-17 12:34 (142.157.22.226)
ywei29  pts/29     2019-01-17 12:37 (132.206.54.123)
jvybihal pts/30     2019-01-17 12:49 (174.138.208.92)
ywul41  pts/31     2019-01-17 12:44 (142.157.115.27)
hteyss  pts/32     2019-01-17 12:44 (177.154.139.194)
[jvybihal][teaching][~]
```

A fun activity is to communicate with other students who are currently logged in. One way to do that is to use the `write` command. You can `man write` to find more information. The basic syntax is the following:

```
write USERNAME
```

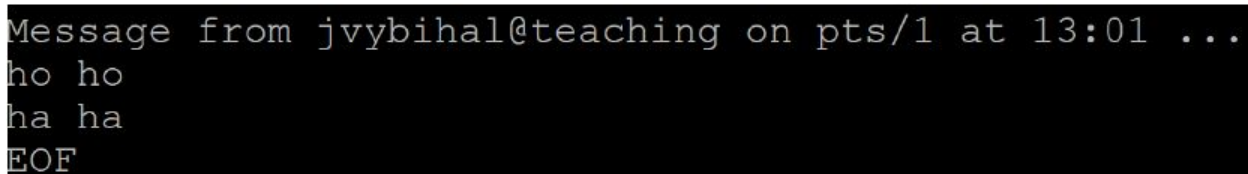
Where USERNAME is the user name that is displayed in the first column of the `who` output. Then you can type sentences to that user, press enter, and type more. All that will be displayed on your friend's screen. To end writing to your friend, press `control-c`. Be careful, this also works for random users, not only your friends.

For exemple, I logged in twice and spoke to myself:

A screenshot of a terminal window titled "teaching.cs.mcgill.ca - PuTTY". The terminal shows a user logging in twice and speaking to themselves. The first login shows the prompt "[jvybihal][teaching][~]" followed by the command "write jvybihal". The user then types "ho ho" and "ha ha". The second login shows the prompt "^C[jvybihal][teaching][~]" followed by a green cursor.

```
[jvybihal][teaching][~] write jvybihal
ho ho
ha ha
^C[jvybihal][teaching][~]
```

On the other account, I saw:

A screenshot of a terminal window showing a message from jvybihal@teaching on pts/1 at 13:01. The message content is "ho ho", "ha ha", and "EOF".

```
Message from jvybihal@teaching on pts/1 at 13:01 ...
ho ho
ha ha
EOF
```

To block people from doing this, use the command: `mesg n`.

You can let people write to you again with this command: `mesg y`.

The meaning of the command is obvious: Messaging NO, Messaging YES. Have fun annoying people.

You have completed the lab.

MENTORING SECTION

The TA will review the following things during their tutorial:

- Review
 - Begin this tutorial by making sure everyone has an account and can log into `mimi.cs.mcgill.ca`
 - Give a demonstration of how to transfer files from mimi to a laptop and back again
 - Review how to use `grep` from the command line
- Lab
 - Go step by step through the command line portion of the lab with the students