



Guide for Parallel Computing Labs

CSCI-UA.0480-051 Special Topics: Parallel Computing

Maya Balaji
Akash Gupta

Step 1: Access to CIMS

- Ensure you have a CIMS account as this will be required to compile and run the labs
- You can test this by running the following command in your local terminal/git bash:
 - `ssh <your_CIMS_ID>@access1.cims.nyu.edu`
- It should work as follows:

```
$ ssh mb8053@access1.cims.nyu.edu
(mb8053@access1.cims.nyu.edu) Password:
(mb8053@access1.cims.nyu.edu) Duo two-factor login for mb8053

Enter a passcode or select one of the following options:

1. Duo Push to XXX-XXX-1462
2. Phone call to XXX-XXX-1462
3. SMS passcodes to XXX-XXX-1462

Passcode or option (1-3): 1
Success. Logging you in...
Last login: Tue Jul 26 21:22:24 2022 from 76.116.67.112
#####
#
# CIMS Access Server
#
# Please do not run CPU-intensive jobs on this server. For
# information regarding appropriate systems on which to run such
# processes, see:
#
# http://cims.nyu.edu/u/computeservers
#
# If you have any questions, please send mail to:
# helpdesk@cims.nyu.edu
#
#####
You are using 0% of your 4.0G quota for /home/mb8053.
You are using 0% of your .4G quota for /web/mb8053.
[mb8053@access1 ~]$ ls -ltr
```

Step 2: SSH into crunchy machines

- Once you are connected through access1 as described in step 1, you can then ssh into one of the crunchy machines as follows:
 - `ssh <your_CIMS_ID>@crunchy1.cims.nyu.edu`
- It should work as follows:

```
[mb8053@access1 ~]$ ssh mb8053@crunchy1.cims.nyu.edu
The authenticity of host 'crunchy1.cims.nyu.edu (128.122.49.145)' can't be established.
ECDSA key fingerprint is SHA256:368f15Lxi+Xc0byQ4Fv4P01FUB1uPJWHJo23Pj2tB8s.
ECDSA key fingerprint is MD5:94:5e:c7:09:8a:9f:02:36:73:12:d3:43:cf:56:c7:eb.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'crunchy1.cims.nyu.edu,128.122.49.145' (ECDSA) to the list of known hosts.
Password:
Attempting to create directory /home/mb8053/per15
You are using 0% of your 4.0G quota for /home/mb8053.
You are using 0% of your .4G quota for /web/mb8053.
[mb8053@crunchy1 ~]$ |
```

(Information regarding other crunchy machines is linked at the end in the 'Useful Resources' slide)

Step 3: Coding in your local machine

- To write the code for your labs, use an IDE such as:
 - Visual Studio Code
 - CLion by JetBrains
 - Code::Blocks
- Bear in mind that just because the code (and tests) run successfully on your local machine that does not imply that it will work on the CIMS servers
- Make sure to test your code using CIMS before submitting your work
- Important things to maintain in your final submission:
 - Source code files (in C)
 - README text file specifying how to compile and run your code
- Once your source code is ready, add all the required files to a single zip file

Step 4: SFTP to CIMS

- Now, from your local machine, open a new window for your terminal/bash environment from the location where your zip file is saved to transfer your file to the specified location in your CIMS home directory:
 - sftp <your_CIMS_ID>@access1.cims.nyu.edu
 - cd <lab_directory> /* you can create one using mkdir */
 - put <filename>.zip
 - ls
 - exit

```
$ sftp mb8053@access1.cims.nyu.edu
(mb8053@access1.cims.nyu.edu) Password:
(mb8053@access1.cims.nyu.edu) Duo two-factor login for mb8053

Enter a passcode or select one of the following options:

1. Duo Push to XXX-XXX-1462
2. Phone call to XXX-XXX-1462
3. SMS passcodes to XXX-XXX-1462

Passcode or option (1-3): 1
Connected to access1.cims.nyu.edu.
sftp> cd sampleLab1/
sftp> put Lab1.zip
Uploading Lab1.zip to /home/mb8053/sampleLab1/Lab1.zip
Lab1.zip
sftp> ls
Lab1.zip
sftp> exit
```

100% 3576 151.3KB/s 00:00

Step 5: Extract source code on CIMS

- Now that the source code zip file has been transferred to the corresponding location on CIMS, let us extract the contents:
 - `ssh <your_CIMS_ID>@access1.cims.nyu.edu`
 - `cd <lab_directory>`
 - `unzip <filename>.zip`
 - `ls -ltr`

```
[mb8053@access1 ~]$ cd sampleLab1/
[mb8053@access1 sampleLab1]$ ls
Lab1.zip
[mb8053@access1 sampleLab1]$ unzip Lab1.zip
Archive: Lab1.zip
  inflating: Linker.cpp
  inflating: Makefile
  inflating: README.md
[mb8053@access1 sampleLab1]$ ls -ltr
total 11
-rw-----. 1 mb8053 mb8053grp   346 Oct  2  2021 README.md
-rw-----. 1 mb8053 mb8053grp 16568 Oct  2  2021 Linker.cpp
-rw-----. 1 mb8053 mb8053grp   102 Oct  2  2021 Makefile
-rw-----. 1 mb8053 mb8053grp   3576 Sep 12 22:47 Lab1.zip
[mb8053@access1 sampleLab1]$
```

Step 6: Compile and Run

- Once your code is transferred to the desired location, run the following commands to compile and run it:
 - `gcc <filename>.c -o <filename>`
 - `./<filename> <arguments> /* run the executable file with arguments if required */`
- Here is an example with a basic “Hello, world” program:

```
[mb8053@access1 lab1]$ cat helloworld.c
#include<stdio.h>
int main() {
    printf("Hello, world!\n");
    return 0;
}
[mb8053@access1 lab1]$ gcc helloworld.c -o helloworld
[mb8053@access1 lab1]$ ls -ltr
total 2
-rw-----. 1 mb8053 mb8053grp 346 Oct  2  2021 README.md
-rw-----. 1 mb8053 mb8053grp  72 Sep 12 22:36 helloworld.c
-rwx-----. 1 mb8053 mb8053grp 8360 Sep 12 22:41 helloworld
[mb8053@access1 lab1]$ ./helloworld
Hello, world!
[mb8053@access1 lab1]$ |
```

Notes

- Note the required output/error messages and adhere to the format carefully to avoid tests failing
- Pass all required arguments and add code checks to catch missing/invalid arguments
- Check the compiler version and if it needs to be changed to run your code ensure that it is mentioned in the README
- If you need to load a different compiler version, unload the existing one first otherwise they stack
 - Ex: module unload 4.8.5
 - module load gcc-9.2
- Understand the lab requirements well and reach out for help with the question or setup sooner than later

Useful Resources

- <http://www.cprogramming.com/tutorial/c-tutorial.html>
- <https://users.cs.cf.ac.uk/Dave.Marshall/C/node13.html>
- https://www.gnu.org/software/libc/manual/html_node/Example-of-Getopt.html
- <https://ubuntu.com/tutorials/command-line-for-beginners#1-overview>
- <https://www.geeksforgeeks.org/how-to-transfer-files-using-sftp/>
- https://cims.nyu.edu/webapps/content/systems/userservices/netaccess/secure_verification