An Introduction to Working in the Linux Command Line

January 23rd, 2014

Note: Do not attempt to copy/paste out of this document. Commands will likely not work!

Some software you will need (Note: these are examples of free software, not endorsements):

In order to	From Mac or Linux	From Windows
Connect to submit (ssh)	Terminal	PuTTY
Move files to/from submit	FileZilla, CyberDuck (Mac)	FileZilla, CyberDuck
Edit text files	TextWrangler (Mac)	Notepad++

Getting around in Linux:

- File paths (directories or folders): /, /home/magitz/, /scratch/lfs/magitz/
- `pwd, cd, ls` (Where am I, change directory, list directory)
- `cp, mv, rm` (copy, move, delete)
- `more, less, head, tail, cat` (examine files)
- `nano, vim` (text editors in Linux)

Making things easier:

- Tab completion- type part of a path and hit tab-key, shell will auto-complete for you
- `history`: redo something that you did before without retyping (use ↑)
- `man`: getting help, also –h or --help flag (e.g.: `man ls`)

Learning by doing:

1. Connect to HPC Center: `ssh <user>@hipergator.rc.ufl.edu`
    a. Type your password and hit return (no characters display while you type). Print Smar
2. Where are you when you login? `pwd`
3. What files are there? `ls`
4. Make a directory: `mkdir test_script`
5. Now what's there? `ls –l`
    a. Linux commands usually have flags to change how they work
    b. `man`, `–h` or `--help` often give you help
6. Change into test_script directory: `cd test_script` or `cd tes<tab>`
7. Copy a sample file here (.): `cp /scratch/lfs/bio/training/2014-01-23/simple.pbs .`
8. Check that the copy worked: `ls`
9. Delete that file: `rm simple.pbs`
    a. That file is now ***GONE***! Not in your recycle bin or trashcan, but gone! *
    *We *may* be able to retrieve *some* files, so if it's an important file, let us know **ASAP**.
10. Luckily we were working with a copy, let's copy it again: up-arrow, up-arrow, return

11. Look at the file: `less simple.pbs` (type q to exit when done)
12. For now, ignore lines that start with a # or `#PBS`, what does this script do?
    a. Can we run this script here? Why not?
    b. Connect to an interactive development/test node: `ssh dev[1 or 2]`
    c. Notice that when you connect to an interactive node, you are in your home directory—same thing happens when your jobs start on compute nodes (see 15a below).
    d. Get back where you want to be: `cd test_script`
    e. Now we can run our script: `./simple.pbs`
    f. Did it do what you expected?
    g. Logout of test node: `logout`

13. Use the scheduler to run this job:
    a. Can't run from home, so…
    b. Change to scratch file space: `cd /scratch/lfs/$USER`
    c. Submit the job: `qsub ~/test_script/simple.pbs` (~ means your home)
    d. The scheduler pays attention to the #PBS lines to schedule and manage your job
14. Look at the result file: `less test.out`
15. How do the results differ?
    a. **Note** this script, and any that you write, *should* have: `cd $PBS_O_WORKDIR`
        i. Notice that the pwd command shows your job was in your home directory, ***not*** where you launched the job from.
    b. Notice all of new environment variables like PBS_O_WORKDIR, PBS_NP, TMPDIR, etc. These can all be used in your scripts.

16. Time to get some data!
    a. Find a file on your computer, or a web site, with some text. Copy and paste that into a new file in your text editor (E.g. TextWrangler, NotePad++). We're just looking for some text to process. It doesn't matter what it is!
    b. Save the file as: **`some_text.txt`** (Check that line breaks are Unix)
    c. Using your text editor or FileZilla, upload this file to your space in /scratch/lfs/
        i. SFTP (port 22) to host: `submit.hpc.ufl.edu`
        ii. Make a directory in /scratch/lfs/$USER/ called: word_cloud

17. Go back to your ssh client and navigate to /scratch/lfs/$USER/word_cloud
18. Use `more, head` and `tail` to look at `some_text.txt`
19. Copy the example cloud.pbs script to your directory:
    a. `cp /scratch/lfs/bio/training/2014-01-23/cloud.pbs .`
20. Edit this script to have your e-mail: `nano cloud.pbs`
21. Submit the cloud script: `qsub cloud.pbs`
22. Check your e-mail
23. Use FileZilla to download your results.
24. Open in a web browser