Pete Supercomputer New User's Manual

High Performance Computing Center
Oklahoma State University

October 4, 2021

Contents

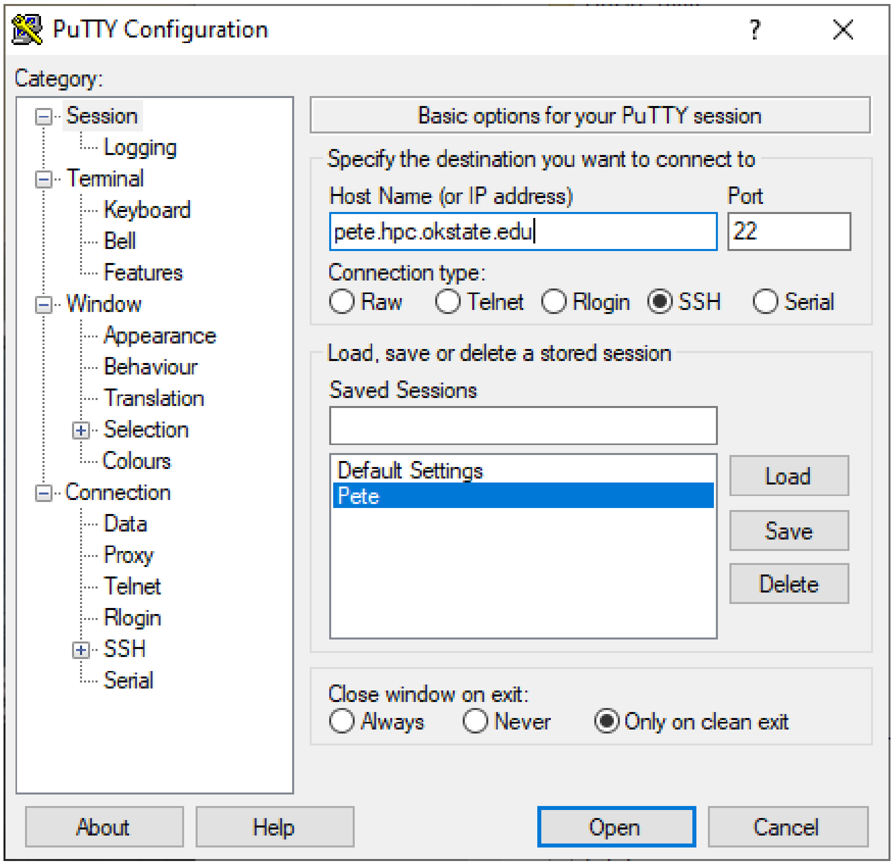
1 Logging In

1.1 Windows OS

To access Pete while using the Windows operating systems (OS) you must first download and install the free and open-source software PuTTY. PuTTY is a terminal emulator that supports network protocols such as Secured Shell (SSH), which is what is used on Pete.

1. First download and save the **'putty.exe'** to your desktop from **here** then run it.

2. Double click on the PuTTY icon to run it, and a window will open similar to the one below.



Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): pete.hpc.okstate.edu

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: Default Settings, Pete

Close window on exit: ○ Always ○ Never ● Only on clean exit

3. In the box labeled *Host Name*, enter the name of the system you want to connect to: **'pete.hpc.okstate.edu'**.

4. Be sure the *Connection type* is **'SSH'** and the *Port number* is **'22'**.

5. In the Saved Sessions box, you can give a name to these login settings, *i.e. Pete*, and click save.

    (a) The next time you open Putty, you can double-click this saved session and it will open a terminal window that is ready for you to login to Pete.

6. After saving your settings and clicking 'Open', you'll be prompted for your Pete username. Hit 'Enter' after you've typed your username.

    (a) A terminal will open, and you may be asked whether or not to trust the host identification.

    (b) If this is your first time logging in using this system click 'yes'.

7. Next you'll be prompted to enter your password followed by 'Enter'.
   (a) **NOTE:** Nothing will appear on the screen as you type your password.
8. If you entered your password correctly, you should get a command-line prompt.
   (a) If it says something like 'permission denied' or 'login invalid', you might have entered your password incorrectly. Remember passwords are case sensitive.
9. To terminate your session on Pete after logging in, type **'exit'** or **'logout'** on the command-line prompt.

NOTE: Using a mouse in the PuTTY terminal:

1. Highlighting text with the mouse = copy.
2. Right clicking with the mouse = paste.

1.2 Mac OS

Mac OS will have a preinstalled terminal application.

1. Open your Finder (hard-drive) window.
2. Double click on the Applications folder → Utilities folder → Terminal icon
3. When the terminal window opens, type: **'ssh username@pete.hpc.okstate.edu'**
   (a) Where 'username' is your login name.
   (b) For Example: If your username is 'osuuser', enter: **'ssh osuuser@pete.hpc.okstate.edu'**
4. Next you'll be prompted to enter your password followed by 'Enter'
   (a) **NOTE:** see 1.1.7.
5. If you entered your password correctly, you should get a command-line prompt.
   (a) If it says something like 'permission denied' or 'login invalid', you might have entered your password incorrectly. Remember passwords are case sensitive.
6. To terminate your session on Pete after logging in, type 'exit' or 'logout' on the command-line prompt.

NOTE: Using a mouse in terminal:

1. In the Mac terminal, highlight (with mouse) then press **'Command'+'C'** = copy.
2. Then use **'Command'+'V'** to paste.

1.3 Linux OS

Linux OS will have some type of terminal application preinstalled.

1. First open a terminal window.
   (a) If you are using Ubuntu press and hold **'Ctrl'+'Alt'+'T'** to bring up a terminal window.
   (b) If you are using CentOS press and hold **'Ctrl'+'Alt'+'F1'** to bring up a terminal window.
2. In the terminal type: **'ssh username@pete.hpc.okstate.edu'**
   (a) Where 'username' is your login name.
   (b) For Example: If your username is 'osuuser', enter: **'ssh osuuser@pete.hpc.okstate.edu'**

3. Next you'll be prompted to enter your password followed by 'Enter'
   (a) **NOTE:** see 1.1.7.
4. If you entered your password correctly, you should get a command-line prompt.
   (a) If it says something like 'permission denied' or 'login invalid', you might have entered your password incorrectly. Remember passwords are case sensitive.
5. To terminate your session on Pete after logging in, type 'exit' or 'logout' on the command-line prompt.

NOTE: Using a mouse in terminal:

1. In Linux's xterm, highlighting (with mouse) = copy.
   (a) Middle clicking = paste.
2. In the Linux GNOME terminal, **'Ctrl'+'Shift'+'C'** = copy.
   (a) **'Ctrl'+'Shift'+'V'** = paste.

2 Changing Your Password

1. Once you're logged in, on the command-line prompt type: **'passwd'**
   (a) **NOTE:** it is 'passwd' not 'password'
2. You will be prompted to enter your old password and then your new password twice.
   (a) **REMEMBER:** Your passwords will not appear as you type
3. Please use a **STRONG** password, which must be:
   (a) Minimum of 8 characters in length
   (b) At least one upper-case character
   (c) At least one lower-case character
   (d) At least one number
   (e) Cannot be based on your username
   (f) Cannot be based on a dictionary word
   (g) Should not be similar to a previously used password
   (h) Should not be known by anyone other than the account holder
4. To log out type: **'exit'** or **'logout'**

3 Useful Linux Commands

A command-line interface (CLI) works by taking commands you type and giving those commands to the operating system to perform.

A few decades ago, computer systems were dominated by Unix code using CLI. In the 1990s, Linux, a Unix-like operating system, was developed, and it drives most of the CLI code today. Below are some common Linux commands that you might use frequently on Pete:

- **'pwd'** – (print working directory) displays your current location on system
- **'ls'** – lists content of directory
- **'cd'** – (change directory) changes to a different directory

- **‘cp’** – copies a file to another name or location
- **‘mv’** – moves or renames a file
- **‘rm’** – removes/deletes files permanently
- **‘rm -r’** – removes/deletes directories and their contents permanently
- **‘grep’** - searches for specific words or strings
- **‘nano -w’** – nano is a text editor
  - The -w tells nano to not put unintended line breaks into your file
- **‘man’** - help/usage for all system commands
- **‘exit’** - terminate an SSH session
- **‘alias’** - replace one string with another while executing a command
- **‘cat’** - view the information stored within a file

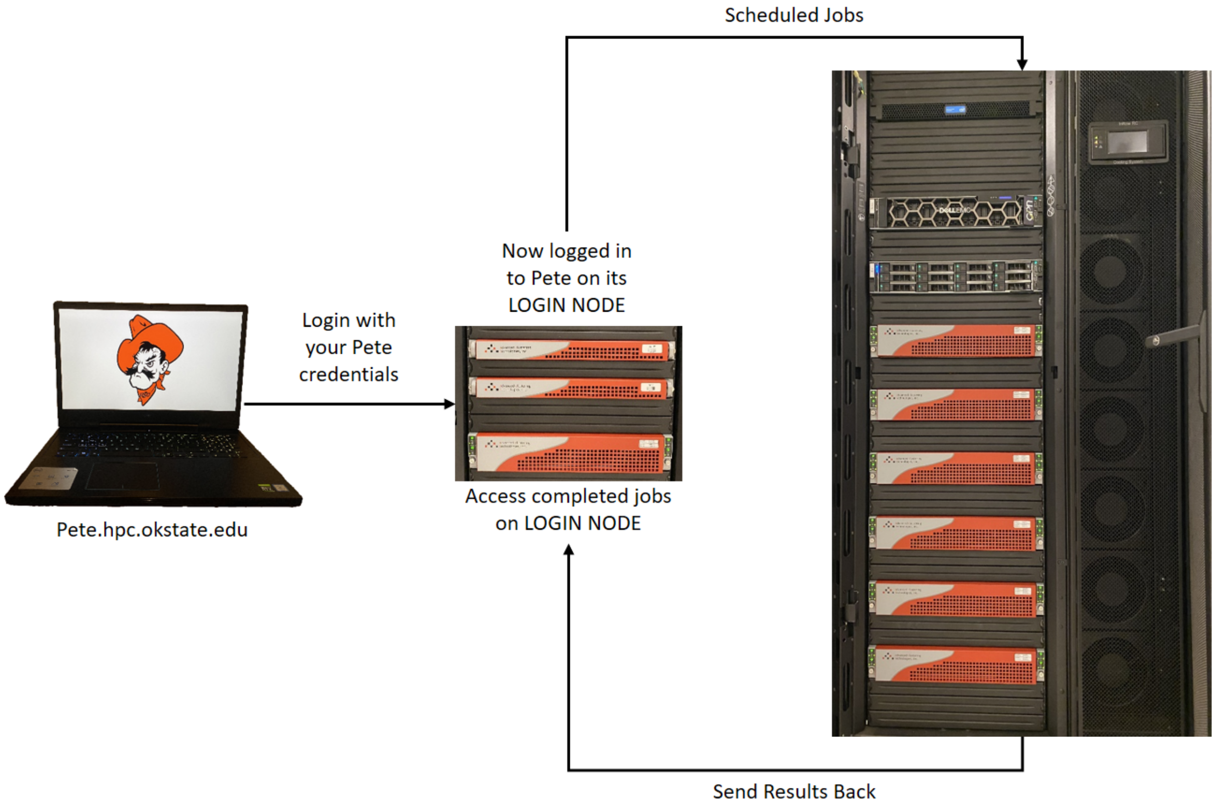
Also, some important keyboard shortcuts:

- **‘TAB’** - autocomplete commands and filenames
- The up arrow will cycle through your previously typed commands
- **‘CTRL’+‘D’** exits a terminal, same as typing ‘exit’
- **‘CTRL’+‘C’** breaks/cancels an ongoing operation, i.e. like the escape key

The OSU HPCC website has a page of links to more Linux tutorials at **HPCC-OSU**
Some of them are reference guides, but here’s a good beginner tutorial for Linux: **Linux-Tutorial**

4 Batch Scheduler on Pete

Remember, when you log in to Pete, you will be logged into one of the ‘login nodes’. The login nodes on Pete my be used to edit files and transfer data. Any intensive processes or scientific applications must be submitted through the batch scheduling system, also known as the **‘scheduler’**. The job scheduler takes some information from you, finds the best compute node(s) to use, and runs your job there. You can even have the scheduler email you when your jobs are finished!


Scheduled Jobs
Now logged in to Pete on its LOGIN NODE
Login with your Pete credentials
Pete.hpc.okstate.edu
Access completed jobs on LOGIN NODE
Send Results Back

This is the most efficient way to use the cluster.

1. Login.
2. Set up jobs with submission scripts.
3. Submit them.
4. Log out and wait for your job to complete.

5 Submitting an Example Job

5.1 Copying Files

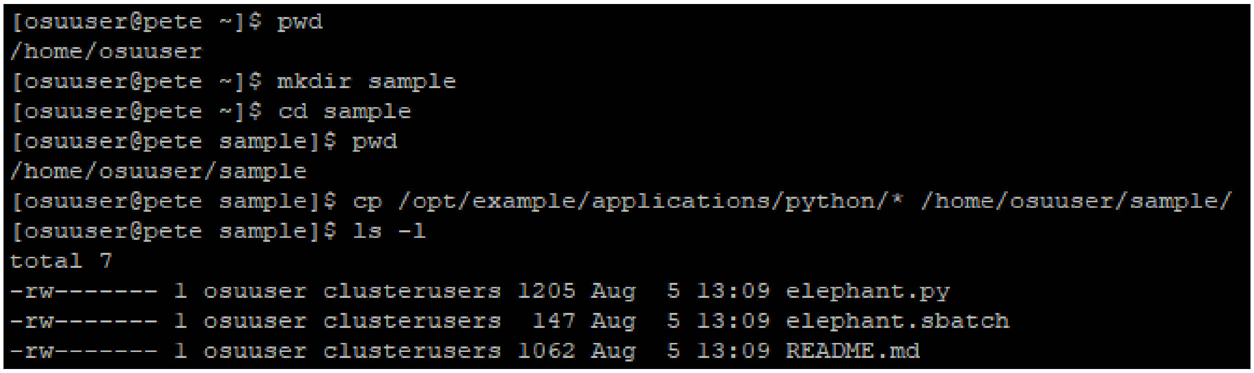
The first thing that will need to be done is creating a directory for the python and submission scripts. After logging in to Pete:

1. Type the command **'pwd'** and press **'Enter'** to see your current directory.
2. Use the command **'mkdir sample'** to create a new directory named "sample" to house all of the needed scripts.
3. Use the command **'cd sample'** to enter the new "sample" directory .
4. Again use the command **'pwd'** to see your new working directory.
5. Type **'cp /opt/example/applications/python/* <path to destination>'** and press **'Enter'** to copy the files to your new directory.

(a) The **’*’** will copy all files in the directory **‘/opt/example/applications/python/’** to your destination.

(b) Your **‘<path to destination>’** should be the same as the result of using the **‘pwd’** command in step 4.

6. U se the command **‘ls -l’** to make sure all files were copied over.



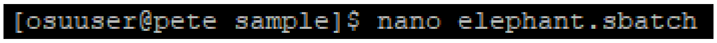
```
[osuuser@pete ~]$ pwd
/home/osuuser
[osuuser@pete ~]$ mkdir sample
[osuuser@pete ~]$ cd sample
[osuuser@pete sample]$ pwd
/home/osuuser/sample
[osuuser@pete sample]$ cp /opt/example/applications/python/* /home/osuuser/sample/
[osuuser@pete sample]$ ls -l
total 7
-rw------- 1 osuuser clusterusers 1205 Aug  5 13:09 elephant.py
-rw------- 1 osuuser clusterusers  147 Aug  5 13:09 elephant.sbatch
-rw------- 1 osuuser clusterusers 1062 Aug  5 13:09 README.md
```

5.2 Editing Files with Nano

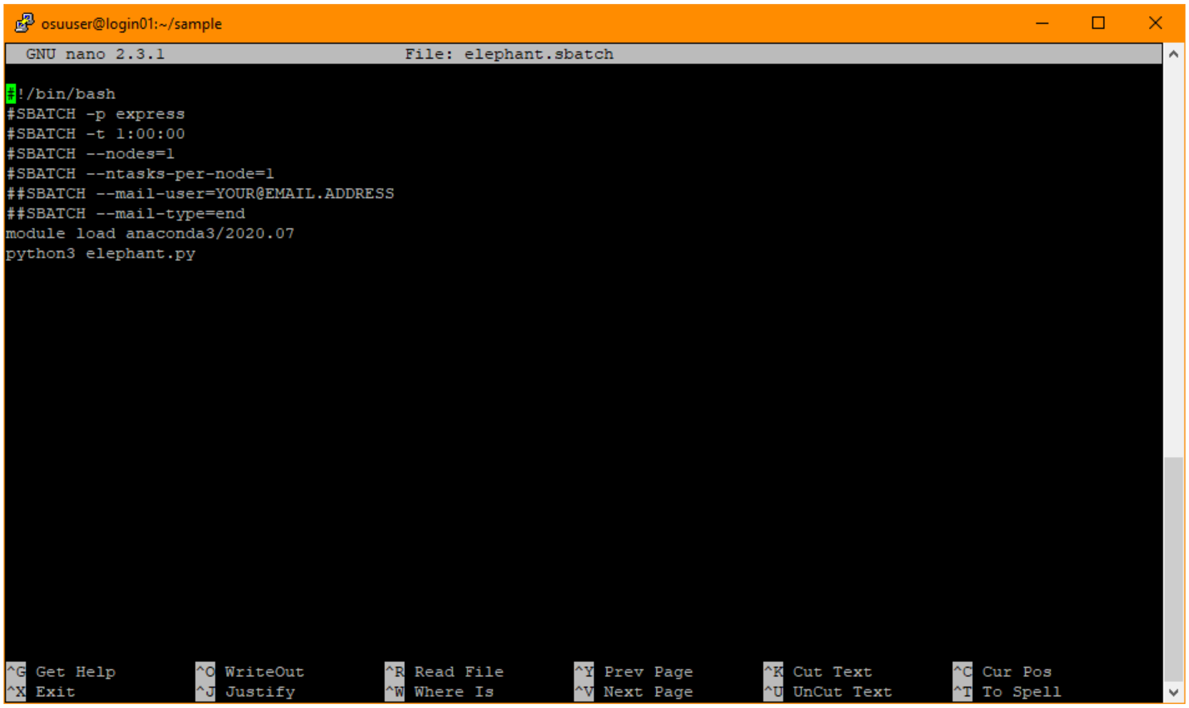
A text editor such as “nano” is a useful tool when working in the terminal environment. In this example we will use “nano” to alter the ‘elephant.sbatch’ to send an email when your calculation has finished. To do this you need to:

1. Open the text file that you want to change in nano with the command **‘nano <name of the file>’**

   (a) In this example **‘<name of the file>’** is **‘elephant.sbatch’**



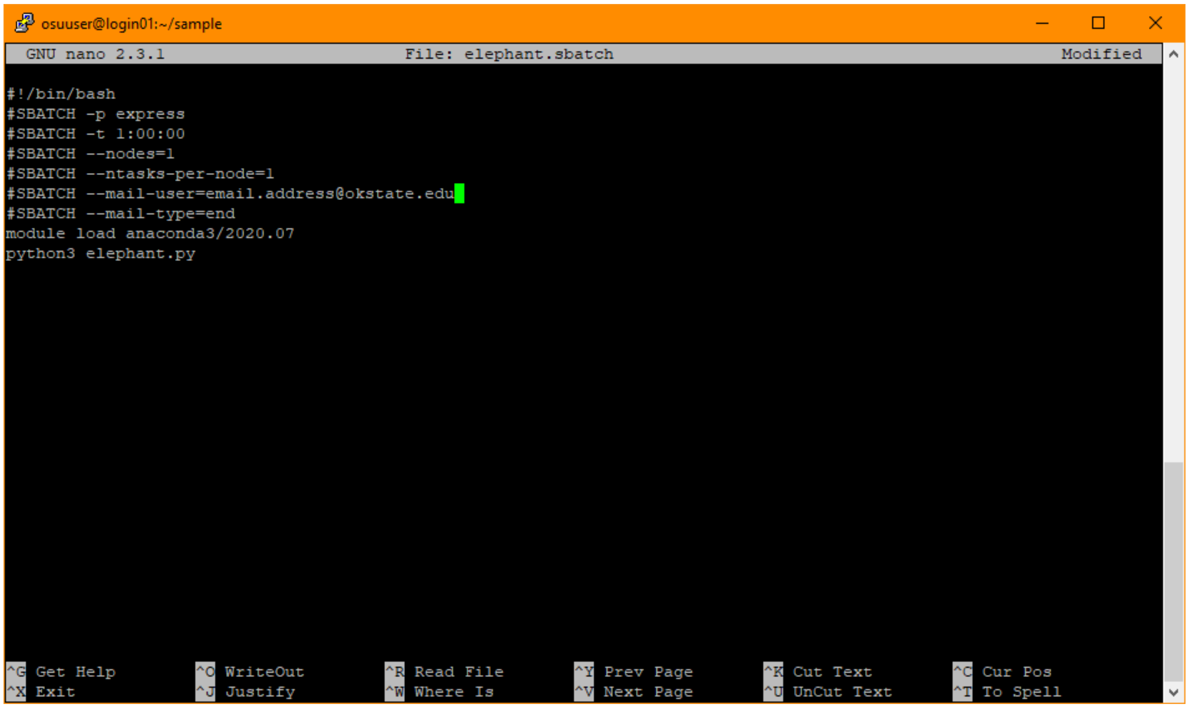
```
[osuuser@pete sample]$ nano elephant.sbatch
```



```
  GNU nano 2.3.1                    File: elephant.sbatch

#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
##SBATCH --mail-user=YOUR@EMAIL.ADDRESS
##SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
```

2. **Important:** First you need to remove the extra # signs from the **#SBATCH –mail-user=YOUR@EMAIL.ADDRESS** and **#SBATCH –mail-type=end** lines (see the image below)

3. You will also need to replace "**YOUR@EMAIL.ADDRESS**" with the email address you would like Pete to send a message to once your job has finished.



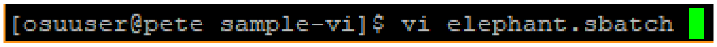
```
  GNU nano 2.3.1                    File: elephant.sbatch                    Modified

#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mail-user=email.address@okstate.edu
#SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
```

4. Once changed you will then press **‘ctrl’+‘X’** to exit and save the changes.
   (a) Nano will first ask you if you want to save your changes at the bottom of the screen
      i. To accept changes press **‘Y’**
      ii. Then press **‘Enter’** to save them to the same file.

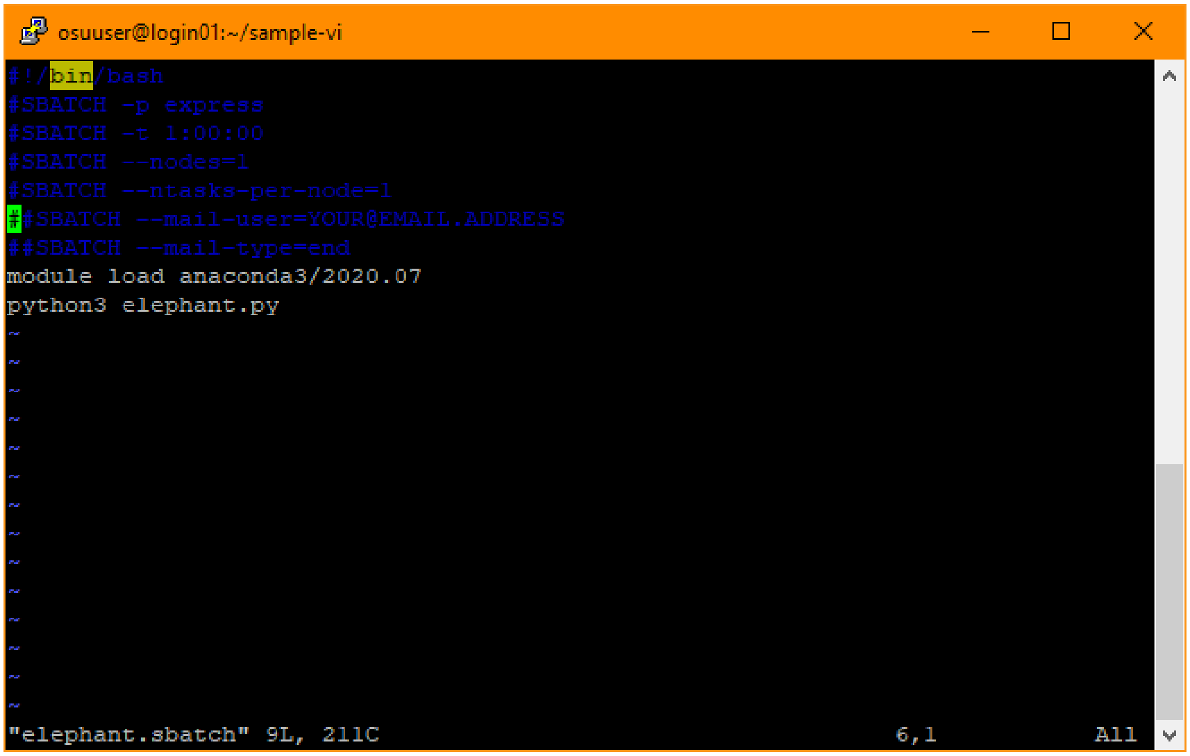
5.3 Editing Files with VI

Another useful text editor and viewer is “VI”. This text editor is slightly more complex than “nano”, so the following example will show how to make the same alterations to the ‘elephant.sbatch’ submission file. To make these changes you need to:

1. Open the text file that you want to change in vi with the command **‘vi <name of the file>’**
   (a) In this example **‘<name of the file>’** is **‘elephant.sbatch’**

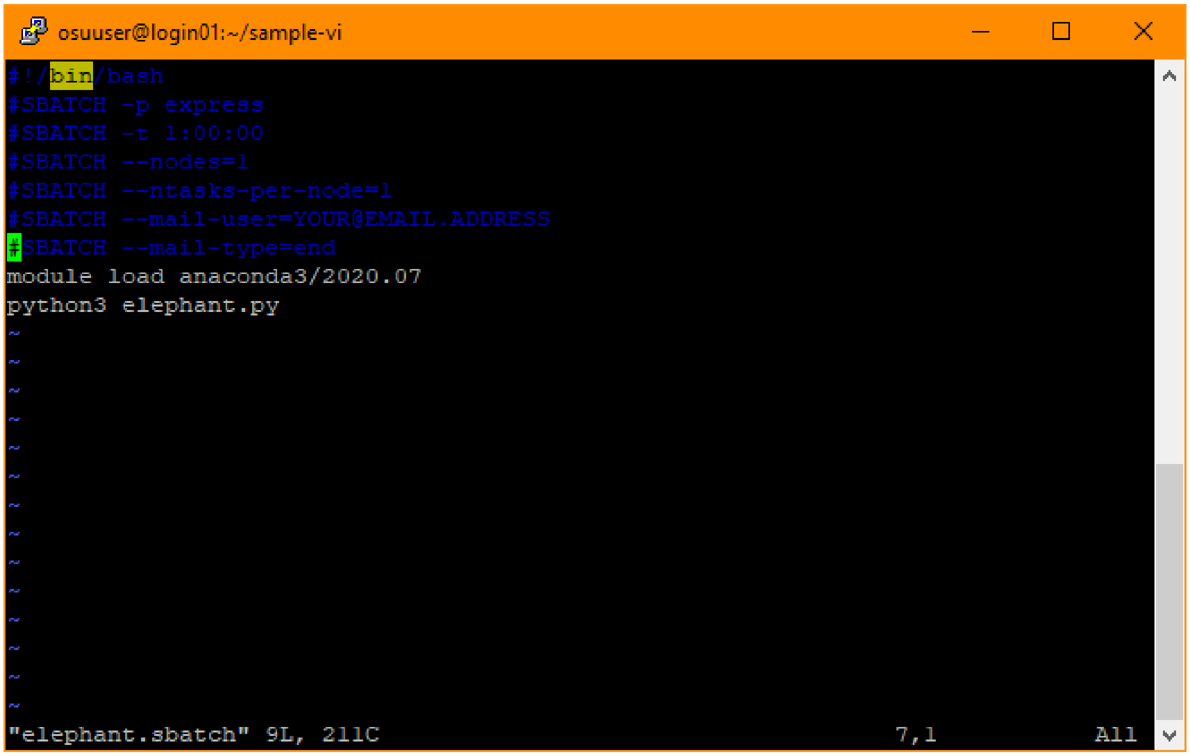


```
[osuuser@pete sample-vi]$ vi elephant.sbatch
```



```
#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
##SBATCH --mail-user=YOUR@EMAIL.ADDRESS
##SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"elephant.sbatch" 9L, 211C                                   6,1           All
```

2. **Important:** First you need to remove the extra # signs from the
   #SBATCH –mail-user=YOUR@EMAIL.ADDRESS and
   #SBATCH –mail-type=end lines
   (a) To do this you will use the arrow keys to navigate to these # then press the ‘del’ button on both.



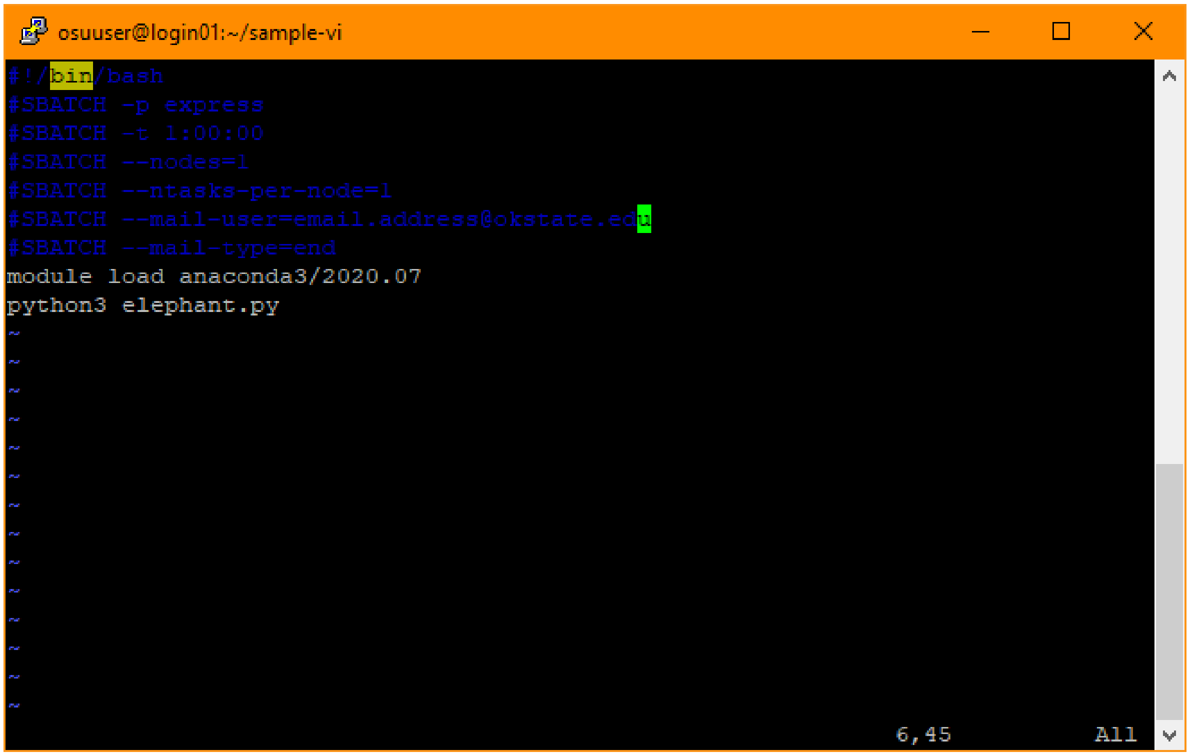
```
#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mail-user=YOUR@EMAIL.ADDRESS
#SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"elephant.sbatch" 9L, 211C                                   7,1           All
```

3. You will also need to replace "**YOUR@EMAIL.ADDRESS**" with the email address you would like Pete to send a message to once your job has finished.

   (a) First delete the "**YOUR@EMAIL.ADDRESS**"

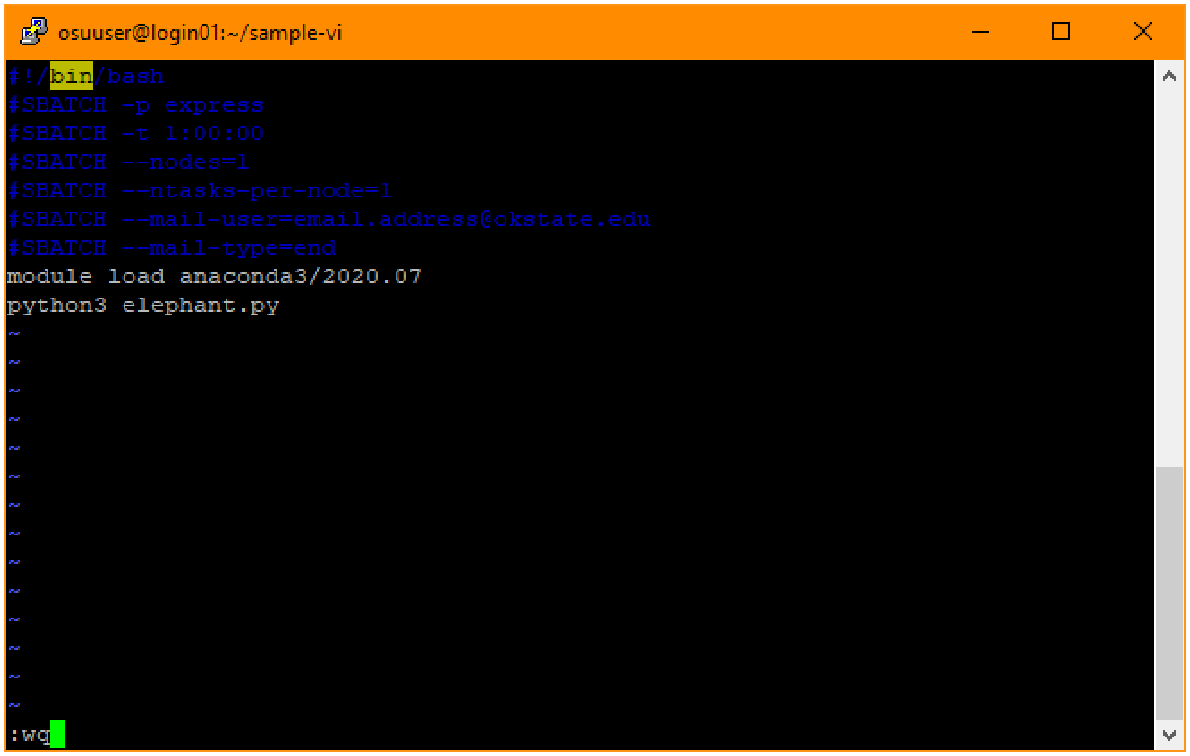
   (b) Then press the **'i'** button to insert your email address

   (c) Once you've added your email address, click **'esc'** to exit the text input



```
#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mail-user=email.address@okstate.edu
#SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
~
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                              6,45          All
```

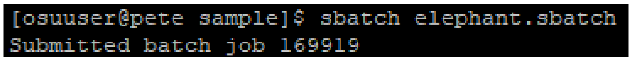
4. Once changed you will then press **'shift'**+**':'**, your text cursor will then be moved to the bottom of the window, type **'wq'**, and press **'enter'**



```
#!/bin/bash
#SBATCH -p express
#SBATCH -t 1:00:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mail-user=email.address@okstate.edu
#SBATCH --mail-type=end
module load anaconda3/2020.07
python3 elephant.py
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

5.4 Submitting Jobs

Once you have saved the changes to the submission script **‘elephant.sbatch’** you can submit it to the queue by using the command **‘sbatch elephant.sbatch’**.

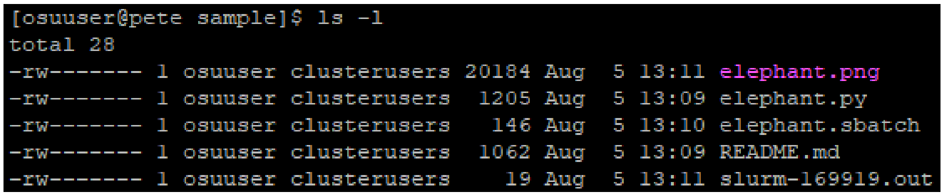


```
[osuuser@pete sample]$ sbatch elephant.sbatch
Submitted batch job 169919
```

Once submitted, Pete will tell you job number and which queue it was submitted to.

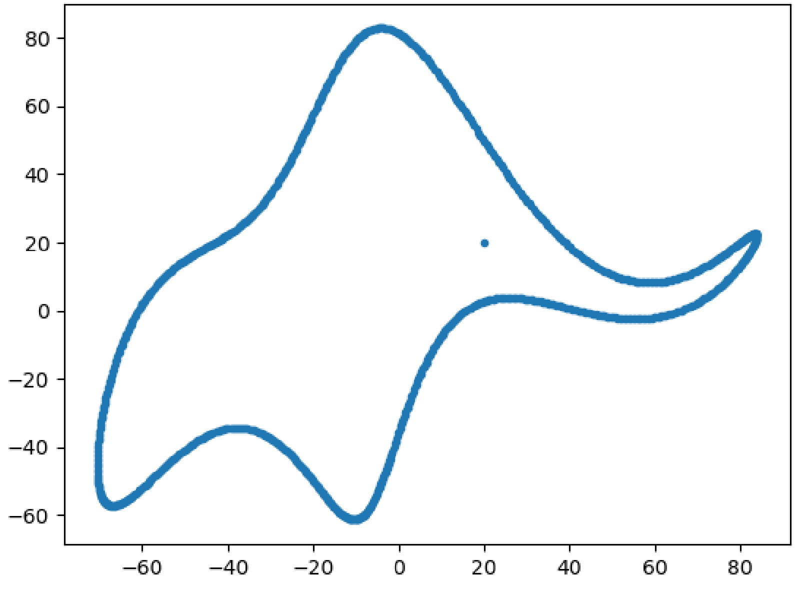
When the job has finished you will find two new files in your working directory:

1. The first is ‘elephant.png’
2. The second is ‘slurm-######.out’



```
[osuuser@pete sample]$ ls -l
total 28
-rw------- 1 osuuser clusterusers 20184 Aug  5 13:11 elephant.png
-rw------- 1 osuuser clusterusers  1205 Aug  5 13:09 elephant.py
-rw------- 1 osuuser clusterusers   146 Aug  5 13:10 elephant.sbatch
-rw------- 1 osuuser clusterusers  1062 Aug  5 13:09 README.md
-rw------- 1 osuuser clusterusers    19 Aug  5 13:11 slurm-169919.out
```

The ‘elephant.png’ contains the plot generated by the python script ‘elephant.py’ which can be seen below. This file can be downloaded to your personal computer using one of the file transfer methods described in section 5.



The ‘slurm-######.out’ file (where ###### is the automatically assigned job number) contains information from slurm about the job that you ran. If there were errors with your job they will be written to this file.

5.5 Cleaning Up

Since user storage space of your **'/home/<username>'** is limited, it is always a good idea to delete files and directories that are no longer needed. You can remove individual files with the command **'rm <filename>'**, and entire directories with the command **'rm -r <directory name>'**.

Note: Once deleted, there is *no way* to recover that file or directory, it is gone *forever*.

6 File Transfers

Many people prefer to use a graphical user interface (GUI) applications to transfer files, such as "WinSCP", "Cyberduck", "Filezilla", and "Globus". In this section we will present step-by-step instructions for using these different programs, including creating an account on "Globus". As a user you are not required to use all of these different programs. The best method for transferring files and directories between your computer and Pete is dependent on which method or program you prefer to use.

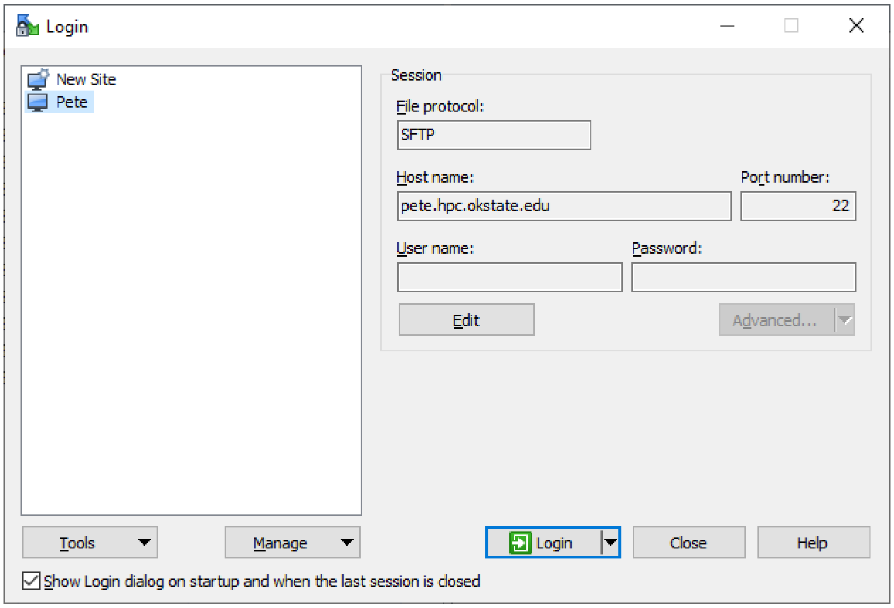
6.1 WinSCP (Windows)

Download the WinSCP installer from **WinSCP-Download**.

1. Once downloaded run the WinSCP-#.#.#-Setup.exe.

    (a) The installer should place the WinSCP.exe icon on your desktop.

2. To run WinSCP, double-click the WinSCP icon, and a window like the one below will open:



Login

New Site

Pete

Session

File protocol: SFTP

Host name: pete.hpc.okstate.edu

Port number: 22

User name:

Password:

☑ Show Login dialog on startup and when the last session is closed

3. In the 'Host name' box enter the hostname **'pete.hpc.okstate.edu'**, and Port number **'22'**.

4. In the 'User name' box enter your username, and in the 'Password' box enter your password.

5. If you would like to save the above information for future use:

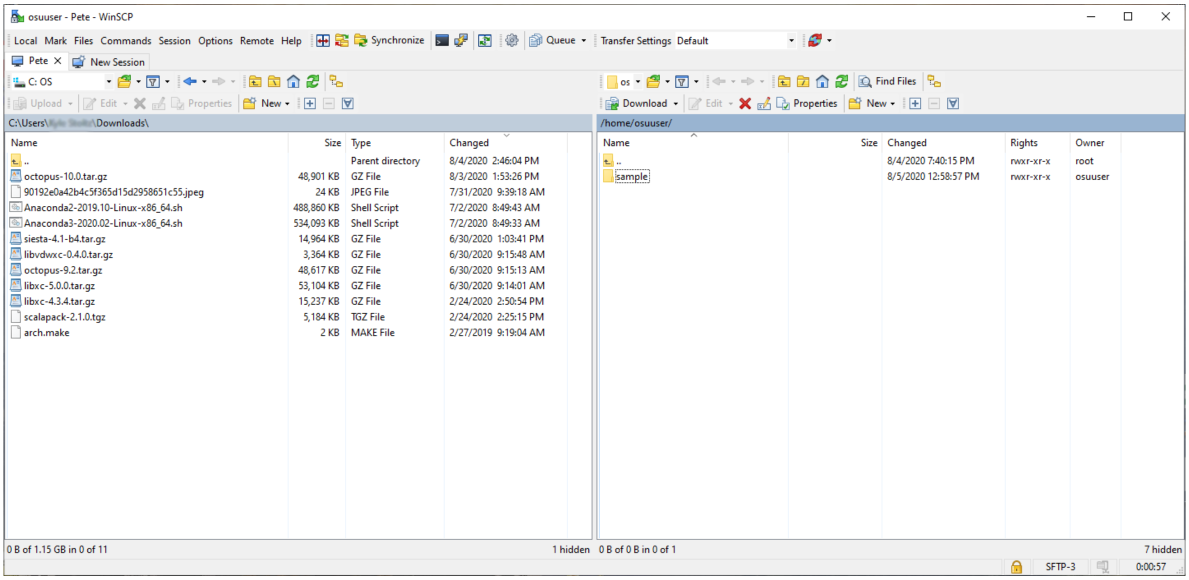
    (a) Click the 'Save' button underneath the 'User name' box.

    (b) Double-click on the session you just saved to login, or highlight it and click 'Login' – this is all you have to do in the future to connect to Pete using WinSCP.

6. If you choose not to save the session information, press enter and it will begin connecting.

Using WinSCP:

1. Once you are connected, you should be greeted with a window similar to the one below.



C:\Users\[redacted]\Downloads\

Name	Size	Type	Changed
..		Parent directory	8/4/2020 2:46:04 PM
octopus-10.0.tar.gz	48,901 KB	GZ File	8/3/2020 1:53:26 PM
90192e0a42b4c5f365d15d2958651c55.jpeg	24 KB	JPEG File	7/31/2020 9:39:18 AM
Anaconda2-2019.10-Linux-x86_64.sh	488,860 KB	Shell Script	7/2/2020 8:49:43 AM
Anaconda3-2020.02-Linux-x86_64.sh	534,093 KB	Shell Script	7/2/2020 8:49:33 AM
siesta-4.1-b4.tar.gz	14,964 KB	GZ File	6/30/2020 1:03:41 PM
libvdwxc-0.4.0.tar.gz	3,364 KB	GZ File	6/30/2020 9:15:48 AM
octopus-9.2.tar.gz	48,617 KB	GZ File	6/30/2020 9:15:13 AM
libxc-5.0.0.tar.gz	53,104 KB	GZ File	6/30/2020 9:14:01 AM
libxc-4.3.4.tar.gz	15,237 KB	GZ File	2/24/2020 2:50:54 PM
scalapack-2.1.0.tgz	5,184 KB	TGZ File	2/24/2020 2:25:15 PM
arch.make	2 KB	MAKE File	2/27/2019 9:19:04 AM

0 B of 1.15 GB in 0 of 11 1 hidden

/home/osuuser/

Name	Size	Changed	Rights	Owner
..		8/4/2020 7:40:15 PM	rwxr-xr-x	root
sample		8/5/2020 12:58:57 PM	rwxr-xr-x	osuuser

0 B of 0 B in 0 of 1 7 hidden

SFTP-3 0:00:57

2. The left side of the screen is your local working directory (i.e. your current working directory on your computer), and the right side of the screen is the current working directory on Pete.

3. ‘drag and drop’ to copy files from Pete to your local computer, or from your computer to Pete.

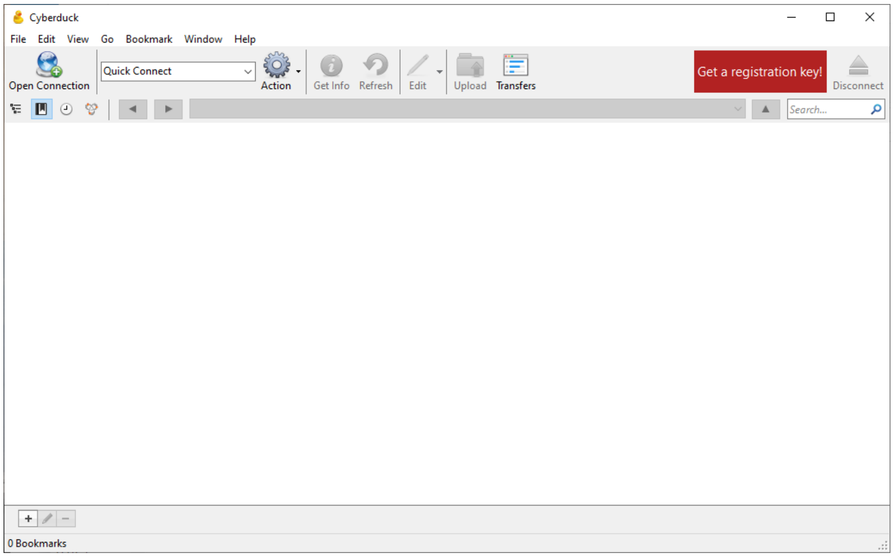
   (a) You may be asked to confirm that you want to copy files.

6.2 Cyberduck (Windows and Mac OS)

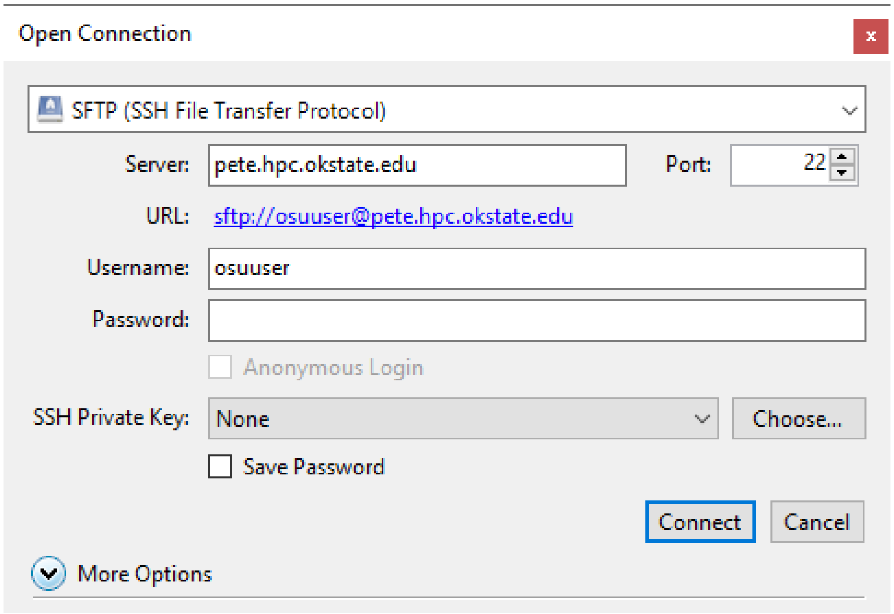
First you’ll need to download Cyberduck from here: **Cyberduck-Download**

1. Click the ‘Download’ icons below the duck – the ‘Start Download’ icon to the right of the duck is an ad.

2. After downloading and opening CyberDuck, you should have a window that looks like this:



3. Click on the ‘Open Connection’ icon at the top left. In the popup window (ee image), do the following:
   (a) Choose the SFTP (SSH File Transfer Protocol) option in the drop down menu at the top
   (b) For ‘Server’: ‘pete.hpc.okstate.edu’
   (c) Change ‘Port’ to ‘22’
   (d) Enter your Pete username and password
   (e) Click ‘Connect’


Open Connection

SFTP (SSH File Transfer Protocol)

Server: pete.hpc.okstate.edu Port: 22

URL: sftp://osuuser@pete.hpc.okstate.edu

Username: osuuser

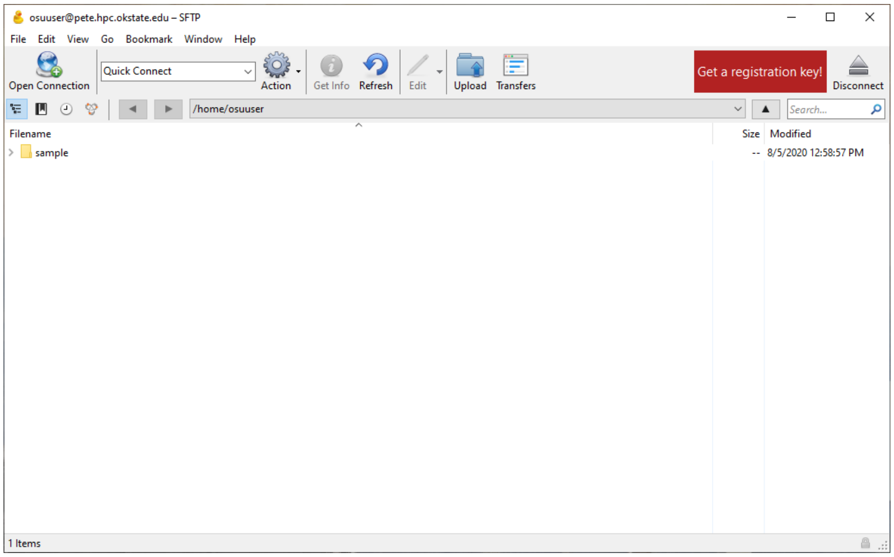
Password:

☐ Anonymous Login

SSH Private Key: None

☐ Save Password

4. After connecting, your CyberDuck window should now look like the window below – listing the contents of your home directory on Pete. You can ‘drag and drop’ files on Pete from this window onto your Desktop and vice versa:



/home/osuuser

Filename	Size	Modified
sample	--	8/5/2020 12:58:57 PM

1 Items

6.3 Command-line File Transfers (Linux, Mac OS, and Windows 10)

To copy a file from Pete to your local machine:

1. Open a terminal on your local machine, navigate to the directory you want the file to go, and type the following on the command line:
   (a) 'scp yourusername@pete.hpc.okstate.edu: directory-where-your-file-is-stored/yourfilename'.
   (b) For example: 'scp osuuser@pete.hpc.okstate.edu:/scratch/osuuser/data/examplefile.txt'.
2. You will be prompted to enter your Pete account password.
3. Once your file is copied from Pete to your local machine, in the terminal, you'll see the filename and 100%.
4. You can now find the file in the folder you sent it to on your local machine.

To copy a file from your local machine to Pete:

1. Open a terminal on your local machine and navigate to the directory where your file is located.
2. Once there, type the following on the command line:
   (a) 'scp localfilename yourusername@pete.hpc.okstate.edu:/directory/'
   (b) For example: 'scp examplefile.txt osuuser@pete.hpc.okstate.edu:/scratch/osuuser/data/'
3. You will be prompted to enter your Pete account password.
4. When the file transfer completes you will see some connection information, and if the transfer was successful the progress should display 100%.
5. Login to Pete and navigate to the directory that you uploaded the file to and type **'ls'** to ensure the transfer was successful.

6.4 Globus (Large Data Transfers)

Globus is an online service to facilitate fault tolerant large data transfers, Pete is setup as an 'endpoint' with name 'okstate#pete'. To use Globus go to Globus.org and follow these steps:

1. Click 'Login' at the top of the screen.
2. Select 'Oklahoma State University System' from the drop-down menu.
   (a) This will take you to the familiar Oklahoma State University login screen.
3. Login with your OKEY credentials.
4. One the next screen you will need to select what you account is used for and read through the "Terms of Service" and "Privacy Policy".
5. Once you have completed your account creation,or login authentication, you can connect to Pete by clicking on the empty 'Collection' bar.
   (a) You will enter in the blank area 'okstate#pete'.
6. You will then need to login in with your Pete username and password.

To connect Globus to your personal computer do the following:

1. Click on the empty 'Collection' bar.
2. Select 'More Options'.
3. Then click 'Install Globus Connect Personal'.
   (a) The Windows installer link will be visible.
   (b) For the Mac and Linux installer links click "Show me other supported operating systems"
   (c) **NOTE:** There are limits to the number of simultaneous transfers from a personal Globus.

For more information visit Globus-Get-Started.

7 File Systems

Each user has access to two basic directories: home and scratch.

- When you first log in, your current working directory is '/home/<username>', where <username> is your Pete username.
  - '/home/<username>' has a quota of 1GB by default. Your source code and executables should be stored here.
    - This filesystem is **NOT BACKED UP**, but it does have snapshots. To check your current usage, type 'du –sh' from your home directory.
  - '/scratch/<username>' is for temporary storage of large files and large collections of files. This filesystem is **NOT BACKED UP**.

NOTE: The first / in a file system path represents the 'root' – or the first or top-most directory in your file system. When you log into Pete, you are automatically in your home folder which stems from the root. Your current working directory would look like '/home/<username>', with the first / representing the root. All proceeding / in a file path serve as dividers between any subsequent directories and files.

8 Different Queues

There are six different queues available on Pete:

1. express (maximum walltime=1:00:00)
   (a) For short jobs, and debugging codes
   (b) Up to 32 cores and 96 GB of RAM
2. batch (maximum walltime=120:00:00)
   (a) For normal jobs
   (b) Up to 32 cores and 96 GB of RAM
3. long (maximum walltime=504:00:00)
   (a) For jobs requiring extended time
   (b) Up to 32 cores and 96 GB of RAM
4. bigmem (maximum walltime=168:00:00)
   (a) For large memory jobs
   (b) Up to 32 cores and 768 GB of RAM
5. supermem (maximum walltime=168:00:00)
   (a) For extra large memory jobs
   (b) Up to 32 cores and 1.5 TB of RAM
6. bullet (maximum walltime=120:00:00)
   (a) For jobs done in tandem or solely on GPUs
   (b) Up to 2 GPUs, 40 cores, and 192 GB of RAM

9 Pete Specs

- 164 compute nodes, each with dual Intel "Skylake" 6130 CPUs and 96 GB RAM
  - These nodes are used in the "express", "batch", and "long" queues
- 12 large memory nodes, each with dual Intel "Skylake" 6130 CPUs and 768GB RAM
  - These nodes belong to the "bigmem" queue
- 1 super memory node with dual Intel "Skylake" 6130 CPUs and 1.5TB RAM
  - These nodes belong to the "supermem" queue
- 10 GPU nodes, each containing dual NVIDIA Quadro RTX6000 GPUs for a total of 20 GPUs
  - These nodes belong to the "bullet" queue
- 100 Gb/s Intel Omni-path low-latency interconnect
- 360 TB high-speed parallel file system
- Over 1 PB of storage space for special projects