Linux Tutorial Cheat Sheet

by jkeelan via cheatography.com/55689/cs/14807/

Cluster commands

qstat	print list of all your jobs
qsub <*script*>	submit <*script*> as a job
qlogin	login to an interactive session on the cluster
qdel <*job ids*>	delete the jobs (<*job ids*> can be a pattern)

Browsing files

cd <*destination*>	change directory
ls *[pattern]*	list all files or match pattern
pwd	print current directory

You can return to your home directory by using **cd ~**

Modifying files

rm *[pattern]*	remove files (-r for recursive)
mv *[sources] [destination]*	move / rename file(s) or folder(s)
cp *[sources] [destination]*	copy file(s) (-r to create desination)

Modifying directories

mv *[sources] [destination]*	rename / move directory
mkdir <*directory*>	create a directory
rmdir <*directory*>	remove a directory
rm -rf <*directory*>	remove directory and all subdirectories

Finding files

find . -file -name "*.txt"	Find all .txt files in the current directory and below and print
locate *[pattern]*	match files with pattern anywhere in the full path and print

Can combine with **| grep**. locate may require **sudo updatedb** from time to time, and won't work on cluster without some modification.

Viewing files

head *[filenames]*	print first 10 lines of file
tail *[filenames]*	print last 10 lines of file
cat *[filenames]*	concatenate files and print

Task management

ps	See all of your active processes
top	Constantly updating list of ordered (by resources) processes
time <*command*>	print time taken to complete after command finishes running
kill <*pid*>	terminate process with id <*pid*>

Name Expansions

{a..z} or **{1..100}**	expands to the series e.g a b c d ...
*	expands to match anything, any number of times
?	Match anything once
$((2 + 2))	Arithmetic expansion (evaluates to 2)
$(<*command*>**)**	expands to the result of the command
~	absolute path to home directory

ls *.txt - *list all .txt files*

cp *{0..9} - *list files which end in a number between 0 and 9*

Processing stdout

awk -F "," `'{print $<column number>}'`	print only column n of files
sort *(-n)*	sort alphabetically (alphanumerically)
uniq *(-c)*	print only one instance of repeated lines (with count of lines)
grep *(-i) [pattern]*	print lines which contain *pattern* (ignore case)
wc *-l*	print number of lines
sed /<*pattern*>/<*replacement*>/g	replace all instances matching <*pattern*> with <*replacement*>

To use on a collection of files, all commands would be prefixed by: **cat** *[files]* **|**

Remote Managment

ssh <*username*>@<*host*>	login to multi-user machine
scp <*username*>@<*host*>:*[remote source]* <*local*>	Cope file(s) from <*host*> to <*local*> destination.
rsync -t <*username*>@<*host*>:*[remote source]* <*local*>	only copy updated files from <*host*> to <*local*>

For the multi-user linux machine, <*host*> should be stem-ssu-linux

Useful

chmod +x <*file*>	give executable priveleges to <*file*>
seq <*start*> <*step*> <*stop*>	print sequence of numbers from <*start*> to <*stop*> in increments of <*step*>
man <*command*>	open the manual page for man
more <*file*>	print output in navigateable pages
fdisk -l	list all the connected drives and partitions
mount <*partition*> <*directory*>	directory will now lead to the partiton (useful for usb storage)

stdout can be piped into **more** to make long outputs readable.