Linux and Command-Line Interface

- Understand basic jargon (shell, path, file permissions, command line)
- Learn common Linux commands
- Tips on making command-line easier
- File permissions
- Text editors
- SSH to connect to cluster

Eric J. Walter
Director of Research Computing
February 4th, 2025

Getting Help

RC/HPC website

USING HPC

- Prerequisites
- Logging in to HPC systems
- Environment modules
- Files & Filesystems
- Running HPC jobs with SLURM
- Compiling & MPI software
- Tutorials
- Software
- Tunneling HTTP for Jupyter

Departments & Offices / ... / Using / Prerequisites

HPC Prerequisites

William & Mary's HPC clusters run on a mixture of Red Hat Enterprise Linux and its derivative CentOS, so you will need basic Unix/Linux knowledge to use the university's HPC systems. If you are unfamiliar with the Unix/Linux command-line, please avail yourself of one or more of the following resources:

- Unix—the Bare Minimum
- UNIX / Linux Tutorial for Beginners
- Writing tcsh shell scripts
- The Linux command line for beginners (Ubuntu focused)

W&M users also have access to many relevant technical e-books through Swem Library, including Unix Power Tools (also available in print), Learning the Unix Operating System (also available in print), Using csh & tcsh, Linux Pocket Guide: Essential Commands, and Unix in a Nutshell.

Text editors

As part of your command-line proficiency, you will want to be familiar with some kind of "plain text" editor. Every W&M HPC login server has at least `vim`, `nano`, and `emacs`, of which `nano` is the easiest for a beginner (but ultimately least powerful). Alternatively, some users prefer to do their editing on their desktop or laptop computers (with the text editor or IDE of their choice), and then use a file transfer utility such as FileZilla, PuTTY, WinSCP, Fetch, rsync, or sftp to copy files to and from the clusters.

HPC webpage: https://www.wm.edu/it/rc

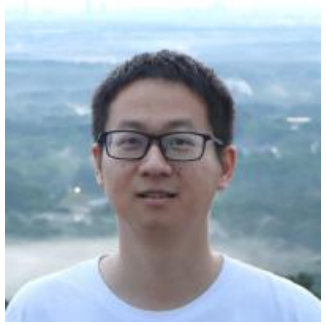
HPC ticket system mail: ***hpc-help@wm.edu***

Research Computing Staff

Eric J. Walter
Director

Matt Kennedy
Network, Filesystem and Data Specialist

Yang Li
Research Computing Assistant



Xiao Liang
Assistant Director

Daanish Fiaz
Systems Engineer

Chris Carpenter
Infrastructure Services Architect (VIMS)

Jay Kanukurthy
Applications Analyst

Malcolm Slaughter
Research Computing Specialist

We are located on 1st floor of ISC3.

The Linux Command Line Interface (CLI)

Windows, MacOS have some command-line features - "*dos prompt*", "*run*", "*powershell*" for Windows or "*terminal*" for MacOS (BSD shell)

Most of you are probably familiar with Graphical User Interfaces (GUI).

Linux especially for HPC is largely command-line. Some GUI interfaces, **but command-line interface (CLI) can be and often is superior.**

- ***command Line*** – Text based interface that lets you interact with computer
- ***shell*** – program that presents the command-line interface
- ***terminal*** – program that runs a shell
- ***prompt*** – beginning of command line which prompts you for action

Shells come in different flavors (tcsh, bash, zsh, ksh, fish, etc.)
On W&M/VIMS clusters, **tcsh** is default shell, users can request **bash**
Everything today will be **shell agnostic**

What shell am I running?:

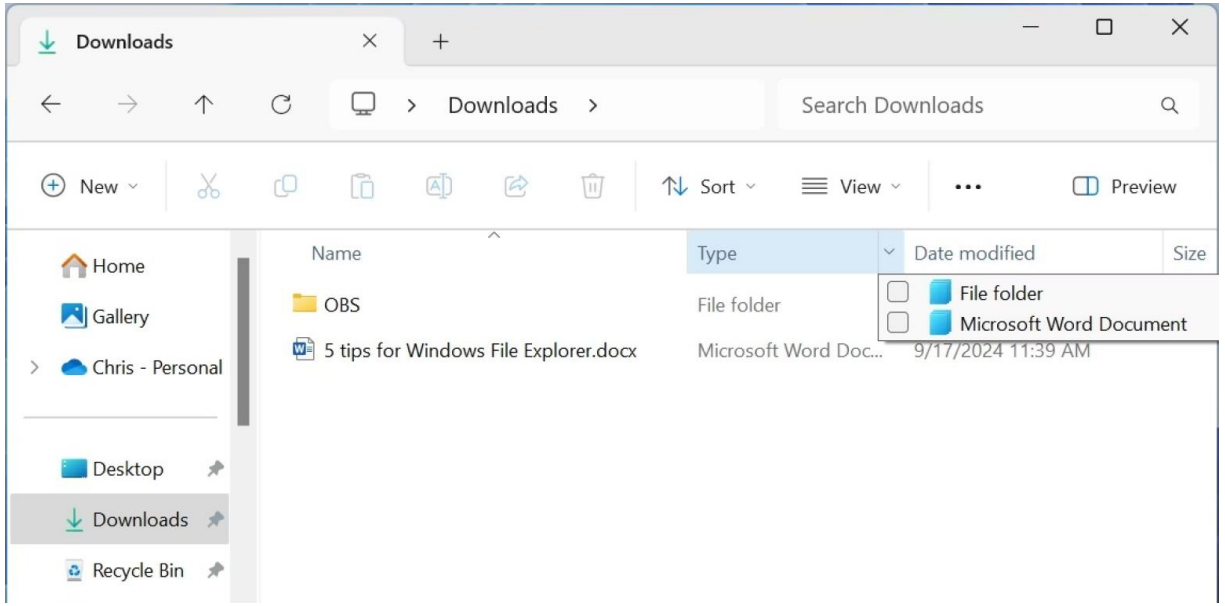
```
1 [bora]

2 [bora] echo $0
-tcsh
```

prompt (line number increments with each command)

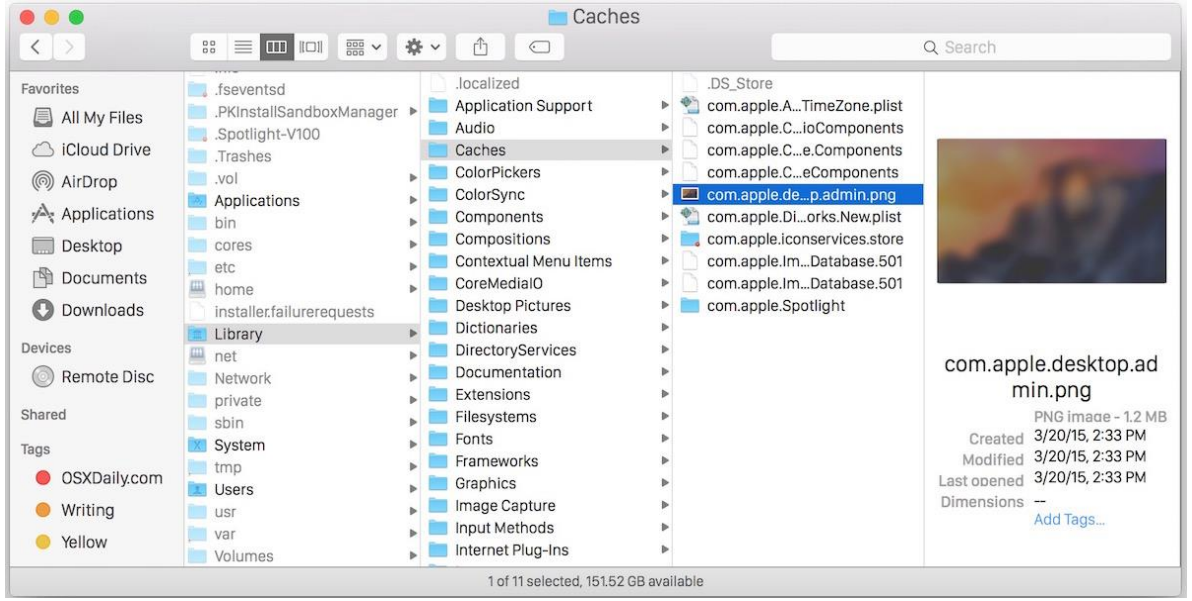
Other Linux Tutorials - https://www.wm.edu/offices/it/services/researchcomputing/using/prereqs/index.php

Files/folders/paths

Windows



Mac



pwd *– print working directory*
aka current ***path***

```
24 [bora] pwd
/sciclone/home/ewalter
```

<--- current path

```
sciclone      "up"
home
ewalter       "down"
```

Common Linux CLI Commands

mkdir *– make directory*
cd *– change directory*
rmdir *– remove directory*
rm *– remove file*
mv *– move / rename file*
cp *– copy file*
ls *– list contents of current directory*
cat *– dump file to screen*
history *– CLI history (!)*
clear *– clear the screen*
***** *– wildcard symbol*

Special chars for files and directories:

. *– current directory*
.. *– up one directory*
~ *– home directory*
- *– previous directory*

cd *– by itself means cd to home*

```
31 [bora] pwd                          print working directory
/sciclone/home/ewalter/LinuxPres
32 [bora] mkdir newdir1                make directory "newdir1"
33 [bora] mkdir newdir2                make directory "newdir2"
34 [bora] ls                           list contents of current dir
newdir1  newdir2
35 [bora] cd newdir1                   go into "newdir1"
36 [bora] ls                           list current content (it is empty)
37 [bora] cd ..                        go up one directory
38 [bora] ls                           list current directory
newdir1  newdir2
39 [bora] ls > dirlist                 ... send stdout of ls to file "dirlist"
40 [bora] cat dirlist                  dump "dirlist" to screen
dirlist
newdir1
newdir2
41 [bora] rmdir newdir2                remove newdir2
42 [bora] mv newdir1 newdir            rename newdir1 to "newdir"
43 [bora] history                      list command history
    31 20:09 pwd
    32 20:09 mkdir newdir1
    33 20:09 mkdir newdir2
    35 20:09 cd newdir1
    37 20:09 cd ..
    38 20:09 ls
    39 20:10 ls > dirlist
    40 20:10 cat dirlist
    41 20:10 rmdir newdir2
    42 20:10 mv newdir1 newdir
    43 20:10 history
44 [bora] !38
ls                                     repeat command #38
dirlist  newdir                        ! - reports what command was
```

```
45 [bora] cp dirlist newdir            copy file "dirlist" into newdir
46 [bora] ls                           list contents (nothing changed)
dirlist  newdir
47 [bora] ls newdir                    list contents of "newdir"
dirlist
48 [bora] cp dirlist dirlist2          make copy of dirlist named dirlist2
49 [bora] ls                           list
dirlist  dirlist2  newdir
51 [bora] cp dir* newdir               cp all files starting with dir to "newdir"
52 [bora] cd newdir                    go into newdir
53 [bora] ls                           list
dirlist  dirlist2
54 [bora] cd ..                        go up one directory
55 [bora] cd ~                         cd to home directory
56 [bora] pwd                          print current path
/sciclone/home/ewalter
57 [bora] cd ~ewalter                  cd to home directory (another way)
58 [bora] pwd                          print current path
/sciclone/home/ewalter
59[bora] cd ~ewalter/LinuxPres          go to LinuxPres (in my home directory)
60 [bora] pwd                          print current path
/sciclone/home/ewalter/LinuxPres
61 [bora] cd                           go to home directory (yet another way)
62 [bora] pwd                          print current path
/sciclone/home/ewalter
63 [bora] cd -                         go to previous directory
64 [bora] pwd                          print current path
/sciclone/home/ewalter/LinuxPres
65 [bora] cd ~                         go home
66 [bora] cd /sciclone/home/ewalter/LinuxPres    absolute path
67 [bora] cd ~                         go home
68 [bora] cd LinuxPres                 relative path
```

Paths, ENV variables, symbolic links

PATH – environmental variable ; list of directories which are searched for executable

```
22 [vortex] echo $PATH
/sciclone/home10/ewalter/bin:/usr/local/intel-2018/compilers_and_libraries_2018/linux/mpi/intel64/bin:/usr/local/inte
l-2018/compilers_and_libraries_2018.5.274/linux/bin/intel64:/usr/local/torque-6.1.1.1/bin:/usr/local/torque-6.1.1.1/s
bin:/usr/local/maui-r156-GRes/bin:/usr/local/Modules/3.2.10/bin:/usr/local/torque-6.1.1.1/bin:/usr/local/torque-6.1.1
.1/sbin:/usr/lib64/qt-3.3/bin:/usr/lib64/ccache:/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/sbin:/opt/ibu
tils/bin:/opt/puppetlabs/bin:/usr/local/bin:/usr/local/sbin:/usr/sbin:/usr/local/intel-2018/parallel_studio_xe_2018.4
.057/bin
```

Current directory (.) is usually not in path – (must type ./<executable>)

Symbolic link – allows an alias for another file/directory

```
23 [vortex] ls -l

lrwxrwxrwx 1 ewalter hpcf    24 Apr  9  2012 data10 -> /sciclone/data10/ewalter
lrwxrwxrwx 1 ewalter hpcf    18 Apr  9  2012 lscr -> /local/scr/ewalter
.
```

ln -s <PATH> <TARGET>

If you are missing symlinks in your home folder: ***restorewmlinks***

Switches / hidden files

Many Linux commands have options called switches

ls -l	list with "long" option
ls -ltr **or** ***ls -l -t -r***	"long" and sort by reverse modification time
rm -r testdir	remove "recursively" **CAREFUL!**

hidden files — files/folder that being with '.'

ls -l vs. ***ls –la*** — *"dot" files*

long vs. short switches

ls -a vs. ***ls --all***

Presence of trailing slash indicates directory

Note: directories can be referred to as "**<directory>**" or "**<directory>/**". The presence of the trailing slash (/) can change some commands behavior. e.g.

Consider you want to mv the file "newfile" into the directory "newdir" **but** you forget to make the directory "newdir" …

```
1 [bora] mv newfile newdir
```
renames newfile to newdir

```
2 [bora] mv newfile newdir/
mv: failed to access 'newdir/': Not a directory
```
complains

```
90 [bora] pwd                                                          print working directory
/sciclone/home/ewalter/LinuxPres
91 [bora] ls -l                                                        list with "-l" switch (long)
total 8
-rw-------. 1 ewalter hpcf 24 Sep 10 20:50 dirlist
-rw-------. 1 ewalter hpcf 24 Sep 10 20:11 dirlist2
drwx------. 2 ewalter hpcf 10 Sep 10 20:30 newdir
92 [bora] ls -ltr                                                      list with long and reverse by
total 8                                                                modification time
-rw-------. 1 ewalter hpcf 24 Sep 10 20:11 dirlist2
drwx------. 2 ewalter hpcf 10 Sep 10 20:30 newdir
-rw-------. 1 ewalter hpcf 24 Sep 10 20:50 dirlist
93 [bora] mkdir .hidden                                                make a "hidden" directory
94 [bora] ls -l                                                        list –l
total 8
-rw-------. 1 ewalter hpcf 24 Sep 10 20:50 dirlist
-rw-------. 1 ewalter hpcf 24 Sep 10 20:11 dirlist2
drwx------. 2 ewalter hpcf 10 Sep 10 20:30 newdir
95 [bora] ls -la                                                       list with –l and –a (for all)
total 56
drwx------.   4 ewalter hpcf    86 Sep 10 20:51 .
drwxr-xr--. 173 ewalter hpcf 86016 Sep 10 20:08 ..
-rw-------.   1 ewalter hpcf    24 Sep 10 20:50 dirlist
-rw-------.   1 ewalter hpcf    24 Sep 10 20:11 dirlist2
drwx------.   2 ewalter hpcf    10 Sep 10 20:51 .hidden                the hidden directory
drwx------.   2 ewalter hpcf    10 Sep 10 20:30 newdir
96 [bora] ls -a                                                        Short list all files with –a
.  ..  dirlist  dirlist2  .hidden  newdir
97 [bora] ls --all                                                     Short list with –all
.  ..  dirlist  dirlist2  .hidden  newdir
98 [bora]
99 [bora] rmdir newdir                                                 Try removing non-empty dir
rmdir: failed to remove ‘newdir’: Directory not empty
100 [bora] rm –r newdir                                                remove –r  recursively
101 [bora] ls
dirlist dirlist2
```

du, df, find

df – get filesystem usage (***df –h***)
du – get current directory disk usage

find – find a file

df and **du** are important in HPC since you share file-systems with other users.

du lets you identify large files/folders
df tells you how full that filesystem is

find . -name <file name>
'.' refers to where to search
-name means match the filename
<file name> is the file you are searching for

find . -type d -name <directory name>
Will search for directory

```
28 [bora] df -h
Filesystem                                  Size  Used Avail Use% Mounted on
.
.
.
192.168.56.208@o2ib,192.168.56.209@o2ib:/pscr  273T  102T  172T  38% /sciclone/pscr
po00-ib:/sciclone/scr10                        219T   38T  181T  18% /sciclone/scr10
cm00:/sciclone/home                             16T   12T  3.9T  76% /sciclone/home
lu00-ib:/sciclone/data10                       360T  334T   27T  93% /sciclone/data10
or00-i8:/sciclone/scr20                         73T   25T   48T  35% /sciclone/scr20
.
.
.
28 [bora] pwd
/sciclone/home/ewalter/examples
29 [bora] du -hs .
22G    total
30 [bora] du -hs *
4.0K CODE_OF_CONDUCT.md
4.0K CODEOWNERS
4.0K CONTRIBUTING.md
436K cpp
.
.
.
31 [bora] cd ..
32 [bora] find . -name CODEOWNERS
examples/CODEOWNERS
```

df with –h switch ("human readable")

pwd

-hs "human-readable" + "summarize" examples is 22 GB

-hs * usage of each file and folder in this directory

cd up one level
find file named "CODEOWNERS" in or below this directory

Tab completion / up arrow / man

Some really helpful commands…

tab completion

ls file<tab>

This will complete the word "file" with whatever files/folders in your current directory match this name.

up arrow

Hitting the up-arrow key will cycle back through your current history

man pages exist for almost all Linux commands

man ls

```
25 [bora] ls
data1 data2 runscript
26 [bora] ls d<TAB>   ->   ls data
data1 data2
26 [bora] ls data2
data2
29 [bora] ls ru<TAB>  ->   ls runscript
runscript


30 [bora] <up arrow>   ->  ls runscript
30 [bora] <up arrow>   ->  ls data2
30 [bora] <up arrow>   ->  ls

31 [bora] man ls

LS(1)                        User Commands                                LS (1)
NAME
       ls - list directory contents
SYNOPSIS
       ls [OPTION]... [FILE]…
DESCRIPTION
       List  information  about the FILEs (the current directory by default).  Sort entries
alphabetically if none of -cftuvSUX nor --sort is specified.

       Mandatory arguments to long options are mandatory for short options too.

       -a, --all
              do not ignore entries starting with .

       -A, --almost-all
              do not list implied . and ..

       --author
              with -l, print the author of each file
.
.
.
```

Grep / tar / gzip, bzip2

grep – globally search for a regular expression and print matching lines

"|" (pipe) - pipe output of one command into another

tar and gzip/bzip2 files:

tar – tape archive

gzip – GNU zip

bzip2 – Another zip program

tar – combine files into single file collection -- **does not compress**

Often combined with ***gzip*** or ***bzip2*** to compress a tarfile (tarball)

gzip – compresses a file -> file.gz

bzip2 – also compresses a file > file.bz2

There other types/suffixes (.xz)

See: https://www.geeksforgeeks.org/tar-command-linux-examples/

For more in depth discussion

```
37 [bora] grep pytorch README.md
![Run Examples](https://github.com/pytorch/examples/workflows/Run%20Examples/badge.svg)
https://pytorch.org/examples/ `pytorch/examples` is a repository showcasing examples of using
.
.
.
38 [bora] grep pytorch README.md |grep !
![Run Examples](https://github.com/pytorch/examples/workflows/Run%20Examples/badge.svg)
39 [bora]


115 [bora] pwd
/sciclone/home/ewalter/LinuxPres
116 [bora] ls
FILES
117 [bora] ls FILES
file1 file2 file3 otherfiles
118 [bora] tar czvf FILES.tgz FILES/
FILES/
FILES/otherfiles/
FILES/otherfiles/file4
FILES/otherfiles/file5
FILES/otherfiles/file6
FILES/file1
FILES/file2
FILES/file3
119 [bora] ls
FILES FILES.tgz
120 [bora] tar tzvf FILES.tgz
drwx------ ewalter/hpcf 0 2023-09-11 17:50 FILES/
drwx------ ewalter/hpcf 0 2023-09-11 17:49 FILES/otherfiles/
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/otherfiles/file4
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/otherfiles/file5
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/otherfiles/file6
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/file1
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/file2
-rw------- ewalter/hpcf 0 2023-09-11 17:49 FILES/file3
```

File/Folder Permissions

```
11 [bora] pwd
/sciclone/home/ewalter
12 [bora] ls -l
total 12
-rw-------. 1 ewalter hpcf 6 Jun  7 18:40 file1
-rw-------. 1 ewalter hpcf 6 Jun  7 18:41 file2
-rw-------. 1 ewalter hpcf 6 Jun  7 18:41 file3
drwx------. 2 ewalter hpcf 6 Jun  7 18:41 testdir
```

directory? user group other — user — group — file/folder

u – user (ewalter)
g – group (hpcf)
o – other (everyone else)

r - read permission
w – write permission
x – execute / traverse permission

File/folder permissions important for sharing files with others within the HPC clusters.

e.g. Want to allow everyone to copy *file1* out of my home directory:

```
77 [bora] ls -l file1
-rw-------. 1 ewalter hpcf 6 Jun  7 18:40 file1
78 [bora] ls -ld /sciclone/home/ewalter
drwx------. 173 ewalter hpcf 86016 Sep 12 12:17 /sciclone/home/ewalter
79 [bora] chmod go+rx /sciclone/home/ewalter
80 [bora] ls -ld /sciclone/home/ewalter
drwxr-xr-x. 173 ewalter hpcf 86016 Sep 12 12:17 /sciclone/home/ewalter
81 [bora] chmod go+r file1
82 [bora] ls -l file1
-r--r--r--. 1 ewalter hpcf 6 Jun  7 18:40 file1
```

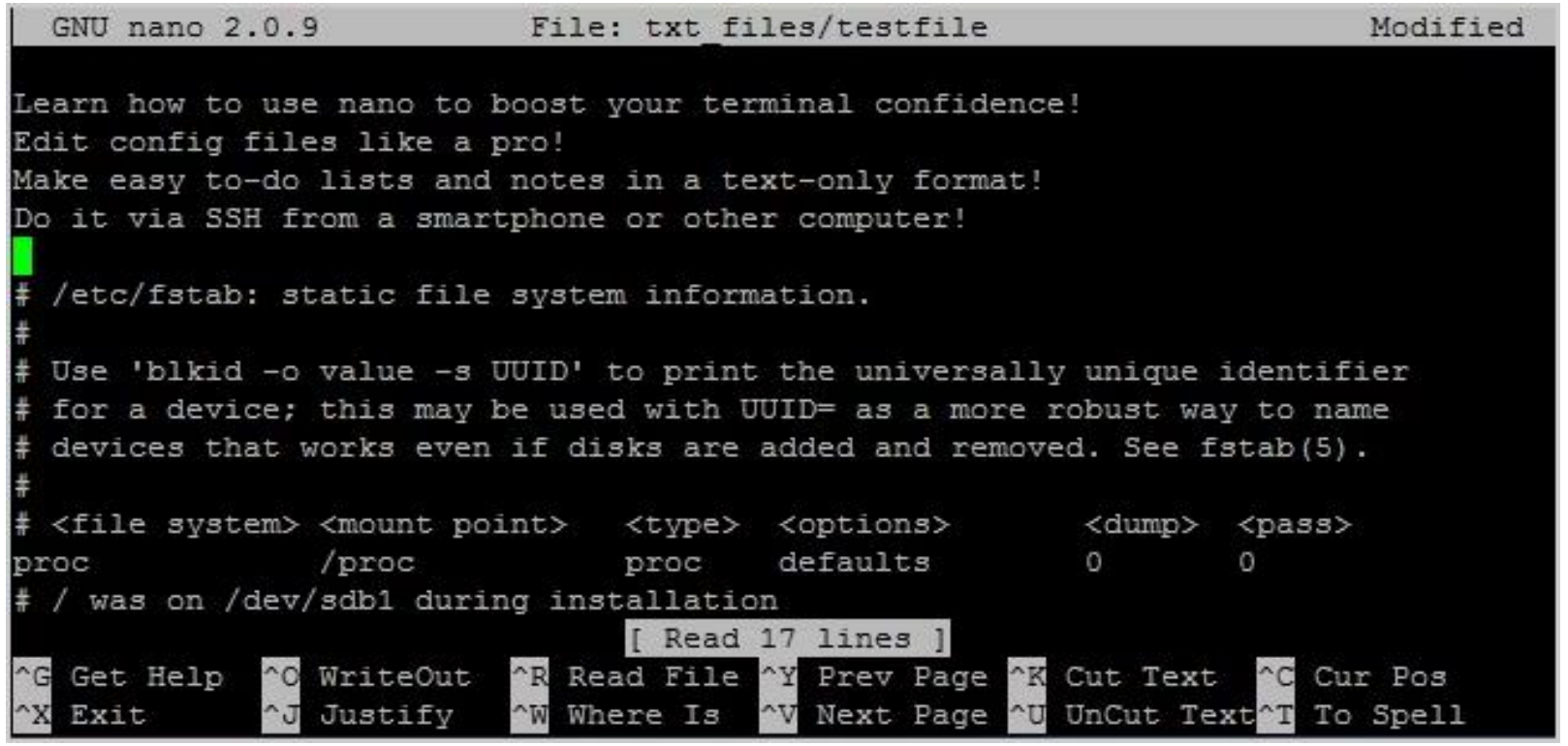
In depth discussion of Linux file permissions:
https://www.redhat.com/en/blog/linux-file-permissions-explained

Permissions discussed more in next presentation (tomorrow)

Text Editors

To create / edit files, need to use a **text editor**. All HPC systems have: ***vi/vim***, ***emacs***, and ***nano***

Nano is most useful for beginners https://www.howtogeek.com/42980/the-beginners-guide-to-nano-the-linux-command-line-text-editor/



How to launch

```
1 [bora] nano

or

1 [bora] nano <filename>
```

List of commands at bottom of screen: i. e. **cntrl-X = exit, cntrl-O = write file**

Also, can set up *visual studio code* to connect to files on HPC via ssh.

SSH to Cluster

ssh – is the standard app for connecting to a remote computer with Linux.
Standard software on Linux. Mac also supports ssh via ***terminal***.
Windows users can use ***powershell*** to do this or use ssh client program (***putty***).

For on W&M/VIMS campus or within W&M/VIMS VPN
ssh to bora - ***ssh ejwalt@bora.sciclone.wm.edu***

From off-campus – need to jump through bastion host
ssh to bora through bastion host - ***ssh -J ejwalt@bastion.wm.edu ejwalt@bora.sciclone.wm.edu***

Can configure your local ssh to jump through bastion host automatically:

My off-campus ~/.ssh/config file:

```
Host bora.sciclone.wm.edu
  HostName bora.sciclone.wm.edu
  ProxyJump ejwalt@bastion.wm.edu
  User ewalter
```

Is my username the same on my local machine?

If it is different use: ssh <username>@<host>.<domain>

Do I need graphics?

If yes, must log in with –Y

See website for more in depth help:
https://www.wm.edu/offices/it/services/researchcomputing/using/connecting/ - connecting/logging in to HPC
https://www.wm.edu/offices/it/services/researchcomputing/using/xfers/ - transferring files to/from HPC