

Introduction to UNIX/Linux

Biochemistry Boot Camp 2021

Session #3

Nicholas Fitzkee

nfitzkee@chemistry.misstate.edu

1

Some terms

- Operating system (OS)

Command-line interface (CLI)

```
File Edit View Terminal Help
top - 10:18:50 up 3:07, 3 users, load average: 0.79, 0.70, 0.73
tasks: 341 total, 1 running, 140 sleeping, 0 stopped, 0 zombie
runtim: 23.2min, 2.7Mv, 8.0Mst, 29.1Mio, 8.9Mm, 0.7Mio, 8.9Mio, 0.4Mst
Mem:  901400K total, 341356K used, 26224K free,  78976K buffers
Swap: 102800K total,  0K used, 102800K free,  53760K cached

  PID USER      PR  NI  VIRT  RES  SHR  %CPU  %MEM     time+  COMMAND
  480 root        0  0 43672   28  196  0.0  0.0  0:00.55 top
 3356 user@1    20  0 36900  116 8892  1.4  0.3  0:00.38 gnome-screensho
13330 user@1   20  0 88464  228 8824  2.0  2.4  0:56.74 compli
13280 user@1   20  0 42104  120 150  1.3  1.4  0:00.32 gnome-applet
13318 user@1   20  0 47140  120 8340  1.6  1.3  0:00.32 gnome-terminal
   0 root        0  0  0  0  0  0.0  0.0  0:00.00 events/9
  678 messageb 20  0 3140 3500  76 1.0  0.2  0:01.29 dbus-daemon
  789 root        0  0  0  0  0  0.0  0.0  3:29.58 php9
13220 user@1   20  0 88912  828 720  1.0  0.3  0:04.07 gnome-pattling-
1629 user@1   20  0 20360  812 794  1.0  0.3  0:02.95 gts-window-deco
1140 user@1   20  0 2100  80 836  0.0  0.0  1:26.54 coffee.Bin
13356 user@1   20  0 2344 1196  560  0.4  0.1  0:00.08 top
   1 root        0  0 2804 1040 1184  0.0  0.2  0:00.57 init
   1 root        0  0  0  0  0  0.0  0.0  0:00.00 systemd
   1 root        0  0  0  0  0  0.0  0.0  0:00.00 migration/0
   1 root        0  0  0  0  0  0.0  0.0  0:00.00 kstfting/0
   1 root        0  0  0  0  0  0.0  0.0  0:00.00 watchdog
```

Graphical user interface (GUI)



2

Why UNIX?

- **Stability:** Systems can run for months or more
- **Multitasking:** Easy to running many programs at once (used to be very unique)
- **Flexibility:** Graphical environment is optional, can be pared down to bare minimum, optimizing performance

3

Why UNIX?

- **Science focus:** Lots of scientific software runs exclusively on UNIX
- **Data Storage:** Unix handles lots of files well
- **Historical reasons:** mainframes, early software written on UNIX

4

Where did Linux come from?

- Linus Torvalds created it
 - with assistance from programmers around the world
 - first posted on Internet in 1991
- Linux 1.0 in 1994; 2.2 in 1999
- Today used on 7-10 million computers
 - with 1000's of programmers working to enhance it

5

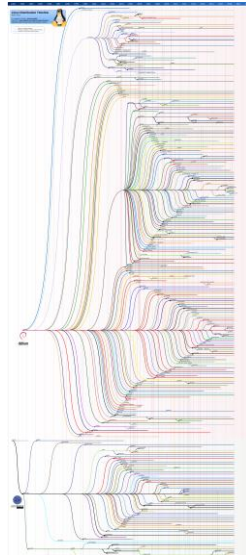
Flavors of Linux

- “Linux” only refers to the kernel: the core program that runs the operating system
- Many programs contribute to the OS experience:
 - Window environment (GUI)
 - Command line interface (CLI)
 - Even simple utilities like the list of printers
- Because Linux is open-source, there lots of combinations of programs that work around the same kernel
 - These are called **distributions** or *distros*

6

Linux Distributions: History

- Distributions made installing software easier
 - Lots of software needed for a working system
 - Oldest and still active: “Slackware”
- Distributions frequently “forked” as new features added or removed
- Some distros also died out



“Linux Distrubtion,” Wikipedia (Debian and Slackware Distros)

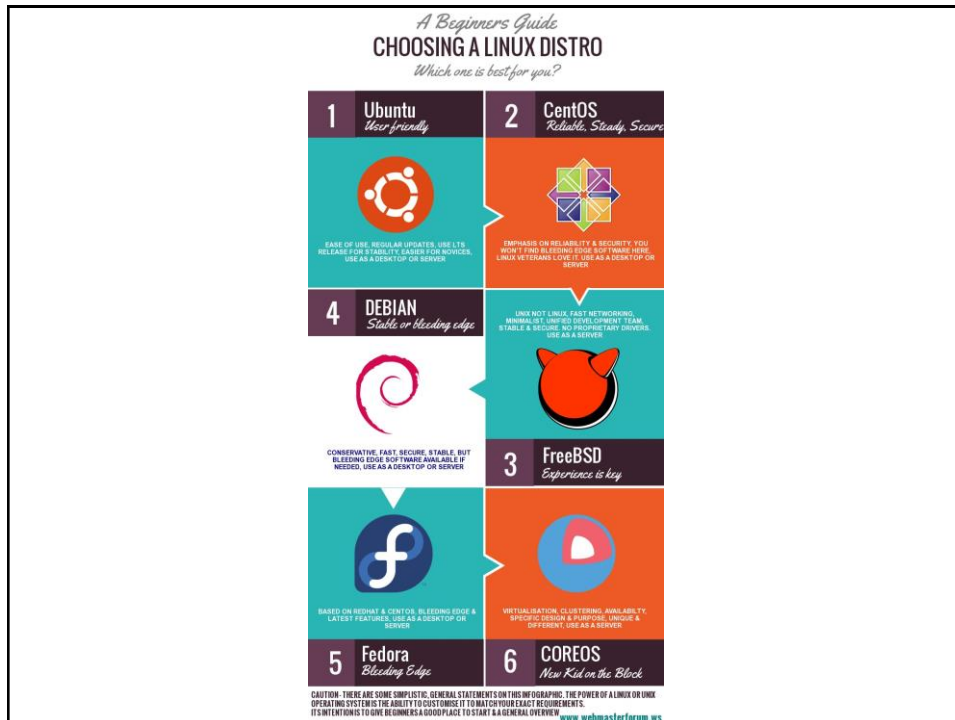
7

Common Linux Distributions

- **CentOS:** Very stable at the expense of some newer features & hardware support
- **Ubuntu:** User friendly, a good mix of “cutting edge” updates and stability
- **Debian:** Cutting edge, but stable software is available for installation too



8

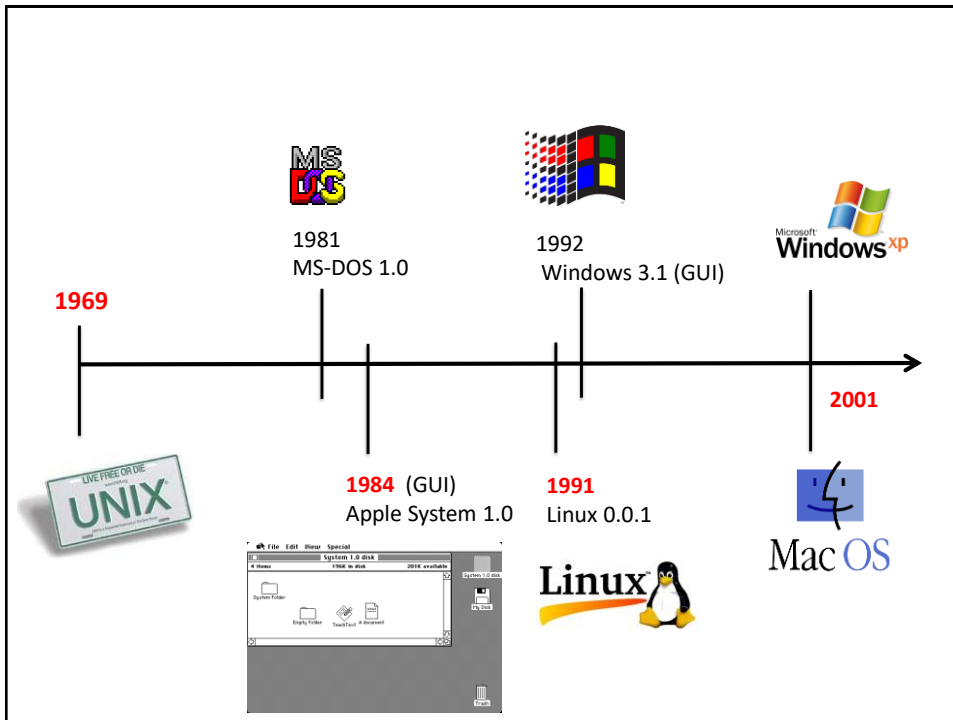


9

Linux vs. “Linux-like” Environment

- An OS may provide a Linux-like (POSIX) environment without actually running the linux kernel
- **MacOS X:** XNU kernel, built around the “mach” kernel; readily supports Linux-like shell scripting and windowing environment (true POSIX)
 - X-Windows supported via Xquartz (<https://www.xquartz.org/>)
- **Windows 10:** Windows kernel, but supports a Linux subsystem to provide Linux support (added in 2016)
 - This session may change in years to come!
<https://arstechnica.com/gadgets/2021/04/graphical-linux-apps-are-coming-to-windows-subsystem-for-linux/>

10



11

Practical Application

This is all very interesting, but...

How do you get to a Linux-like Command Line
and what can you do when you get there?

12

Mac Users

- Basic command line interface can be found in the “Applications/Utilities” folder – run the program “Terminal”
- We will use XQuartz as well, which can be found at www.xquartz.org
 - Download and install this program while PC owners struggle through tutorial!
- File Transfer Client: **Fugu** or **Cyberduck**
<https://sourceforge.net/projects/fugussh/>
<https://cyberduck.io/>

13

PC Users

- Getting to Linux is a bit more difficult
- Follow the instructions in the “Running X11 on Windows” handout
- File Transfer Client: **WinSCP**
<https://winscp.net/eng/download.php>

14

Connect to a Linux Server via SSH

- Mac Users
 - Make sure XQuartz is running
 - Open a new XTerm window and run SSH
 - **Example:** `ssh -Y jsmith@host.msstate.edu`
 - Replace `jsmith` with your username and `host` with the value given in class!
- PC Users
 - Follow directions in X11 Handout using the SSH Secure Shell Program (Quick Connect)

15

Getting Files To/From the Server

- **PC/Mac:** Open up WinSCP or Fugu/Cyberduck
- **Old School:** Open up another Xterm, then:
`sftp <username>@host.msstate.edu`
 - Need to use the `cd` command to navigate to the right folder, then get to copy the file
 - It will save the files to wherever you started the `sftp` program

16

Demo: Common Linux Tasks

- Navigating the file system: Where am I? What's here?
- Copying/Manipulating Files and Directories
- Running programs in the background
 - Ampersand (“&”) trick
 - Useful commands: Ctrl-Z, bg, fg
- Running software via X11
 - Text file editing (xemacs)

17

Try It Yourself: Linux Tutorial

- Very helpful, and covers basic to advanced topics:

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

- Your Linux account will be active for **30 days**, after which it will be disabled
 - Contact Dr. Fitzkee if you need more time

18

File Commands	System Info
ls - directory listing	date - show the current date and time
ls -la - formatted listing with hidden files	cal - show this month's calendar
cd dir - change directory to <i>dir</i>	uptime - show current uptime
cd - - change to home directory (e.g. <code>/home/nf1tzk00</code>)	w - display who is online
pwd - show current directory	whoami - who you are logged in as
mkdir dir - create a directory <i>dir</i>	finger user - display information about user
rm file - delete <i>file</i>	uname -a - show kernel information
rm -r dir - delete directory <i>dir</i>	cat /proc/cpuinfo - cpu information
rm -f file - force remove <i>file</i>	cat /proc/meminfo - memory information
rm -rf dir - force remove directory <i>dir</i> * (see warning below!)	man command - show the manual for <i>command</i>
cp file1 file2 - copy <i>file1</i> to <i>file2</i>	df - show disk usage
cp -r dir1 dir2 - copy <i>dir1</i> to <i>dir2</i> ; create <i>dir2</i> if it doesn't exist	du - show directory space usage
mv file1 file2 - rename or move <i>file1</i> to <i>file2</i>	free - show memory and swap usage
if <i>file2</i> is an existing directory, moves <i>file1</i> into directory <i>file2</i>	whereis app - show possible locations of <i>app</i>
ln -s file link - create symbolic link <i>link</i> to <i>file</i>	which app - show which <i>app</i> will be run by default
touch file - create or update <i>file</i>	man - Command Reference
cat > file - places standard input into <i>file</i>	tar of file.tar file - create a tar named <i>file.tar</i> containing <i>file</i>
more file - output the contents of <i>file</i> (alternatively: less file)	tar xf file.tar - extract the files from <i>file.tar</i>
head file - output the first 10 lines of <i>file</i>	tar czf file.tar.gz file - create a tar with Gzip compression
tail file - output the last 10 lines of <i>file</i>	tar xzf file.tar.gz - extract a tar using Gzip
tail -f file - output the contents of <i>file</i> as it grows, starting with the last 10 lines	zip -r file.zip file - create a Windows-compatible zip archive
Process Management	unzip file.zip - extract zip archive
ps - display your currently active processes	gzip file - compresses <i>file</i> and renames it to <i>file.gz</i>
top - display all running processes	gzip -d file.gz - decompresses <i>file.gz</i> back to <i>file</i>
kill pid - kill process id <i>pid</i>	Network
killall proc - kill all processes named <i>proc</i> *	ping host - ping <i>host</i> and output results
bg - lists stopped or background jobs; resume a stopped job in the background	whois domain - get whois information for <i>domain</i>
fg - brings the most recent job to foreground	dig domain - get DNS information for <i>domain</i>
fg n - brings job <i>n</i> to the foreground	dig -x host - reverse lookup <i>host</i>
File Permissions	wget file - download <i>file</i>
chmod octal file - change the permissions of <i>file</i> to <i>octal</i> , which can be found separately for user, group, and world by adding:	wget -o file - continue a stopped download
<ul style="list-style-type: none"> 4 - read (r) 2 - write (w) 1 - execute (x) 	Lab Utilities
Examples:	snmpviewers: snmpcr or nrdbrw or mwj or analysis
chmod 777 - read, write, execute for all	pipe2sniff file, file2 file, user - convert NMAP pipe spectrum to UCSSF format
chmod 755 - rwx for owner, rx for group and world	/home/databases/pdb/ - location of all PDB structures
For more options, see man chmod .	PDB Viewers: ppsmi or molcol or rsmol or wad
SSH	Text editors: xmooze or emooze or godit or vi
ssh user@host - connect to <i>host</i> as <i>user</i>	All can be invoked with a file, e.g. xmooze file
ssh -p port user@host - connect to <i>host</i> on port <i>port</i> as <i>user</i>	/usr/bin - run an executable file script in the current directory
rftp user@host - connect to <i>host</i> as <i>user</i> for file transfer	Shortcuts
rftp - graphical file transfer client	Ctrl+C - halts the current command
Searching	Ctrl+Z - stops the current command, resume with fg in the foreground or bg in the background
grep pattern files - search for <i>pattern</i> in <i>files</i>	Ctrl+B - log out of current session, similar to exit
grep -r pattern dir - search recursively for <i>pattern</i> in <i>dir</i>	Ctrl+W - erases one word in the current line
command grep pattern - search for <i>pattern</i> in the output of <i>command</i>	Ctrl+U - erases the whole line
locate file - find all instances of <i>file</i>	Ctrl+R - type to bring up a recent command
find -name "pattern" - search for the file named <i>pattern</i> in the current directory (or below); <i>pattern</i> can contain wildcards (e.g. <code>*)</code>)	!! - repeats the last command
	exit - log out of current session
	* use with extreme caution.

19

File Commands
ls - directory listing
ls -la - formatted listing with hidden files
cd dir - change directory to <i>dir</i>
cd - - change to home directory (e.g. <code>/home/nf1tzk00</code>)
pwd - show current directory
mkdir dir - create a directory <i>dir</i>
rm file - delete <i>file</i>
rm -r dir - delete directory <i>dir</i>
rm -f file - force remove <i>file</i>
rm -rf dir - force remove directory <i>dir</i> * (see warning below!)
cp file1 file2 - copy <i>file1</i> to <i>file2</i>
cp -r dir1 dir2 - copy <i>dir1</i> to <i>dir2</i> ; create <i>dir2</i> if it doesn't exist
mv file1 file2 - rename or move <i>file1</i> to <i>file2</i>
if <i>file2</i> is an existing directory, moves <i>file1</i> into directory <i>file2</i>
ln -s file link - create symbolic link <i>link</i> to <i>file</i>
touch file - create or update <i>file</i>
cat > file - places standard input into <i>file</i>
more file - output the contents of <i>file</i> (alternatively: less file)
head file - output the first 10 lines of <i>file</i>
tail file - output the last 10 lines of <i>file</i>
tail -f file - output the contents of <i>file</i> as it grows, starting with the last 10 lines

20

Process Management	System Info
ps – display your currently active processes	date – show the current date and time
top – display all running processes	cal – show this month's calendar
kill pid – kill process id <i>pid</i>	uptime – show current uptime
killall proc – kill all processes named <i>proc</i> *	w – display who is online
bg – lists stopped or background jobs; resume a stopped job in the background	whoami – who you are logged in as
fg – brings the most recent job to foreground	finger user – display information about <i>user</i>
fg n – brings job <i>n</i> to the foreground	uname -a – show kernel information
File Permissions	cat /proc/cpuinfo – cpu information
chmod octal file – change the permissions of <i>file</i> to <i>octal</i> , which can be found separately for user, group, and world by adding:	cat /proc/meminfo – memory information
<ul style="list-style-type: none"> □ 4 – read (r) □ 2 – write (w) □ 1 – execute (x) 	man command – show the manual for <i>command</i>
Examples:	df – show disk usage
chmod 777 – read, write, execute for all	du – show directory space usage
chmod 755 – rwx for owner, rx for group and world	free – show memory and swap usage
For more options, see man chmod	whereis app – show possible locations of <i>app</i>
SSH	which app – show which <i>app</i> will be run by default
ssh user@host – connect to <i>host</i> as <i>user</i>	Compression
ssh -p port user@host – connect to <i>host</i> on port <i>port</i> as <i>user</i>	tar cf file.tar files – create a tar named <i>file.tar</i> containing <i>files</i>
sftp user@host – connect to <i>host</i> as <i>user</i> for file transfer	tar xf file.tar – extract the files from <i>file.tar</i>
ftpt – graphical file transfer client	tar ozf file.tar.gz files – create a tar with Gzip compression
Searching	tar xzf file.tar.gz – extract a tar using Gzip
grep pattern files – search for <i>pattern</i> in <i>files</i>	zip -r file.zip files – create a Windows-compatible zip archive
grep -r pattern dir – search recursively for <i>pattern</i> in <i>dir</i>	unzip file.zip – extract zip archive
command grep pattern – search for <i>pattern</i> in the output of <i>command</i>	gzip file – compresses <i>file</i> and renames it to <i>file.gz</i>
locate file – find all instances of <i>file</i>	gzip -d file.gz – decompresses <i>file.gz</i> back to <i>file</i>
find . -name "pattern" – search for the file named <i>pattern</i> in the current directory (or below); <i>pattern</i> can contain wildcards (e.g. <i>*.*</i>)	

21

Network
ping host – ping <i>host</i> and output results
whois domain – get whois information for <i>domain</i>
dig domain – get DNS information for <i>domain</i>
dig -x host – reverse lookup <i>host</i>
wget file – download <i>file</i>
wget -c file – continue a stopped download
Lab Utilities
NMR Viewers: sparky or nmrDraw or nvj or analysis
pipe2ucsf file.ft2 file.ucsf – convert NMRPipe spectrum to UCSF format
/home/databases/pdb/ – location of all PDB structures
PDB Viewers: pymol or molmol or rasmol or vmd
Text editors: xemacs or emacs or gedit or vi
All can be invoked with a file, e.g. xemacs file
./script – run an executable file <i>script</i> in the current directory
Shortcuts
Ctrl+C – halts the current command
Ctrl+Z – stops the current command, resume with fg in the foreground or bg in the background
Ctrl+D – log out of current session, similar to exit
Ctrl+W – erases one word in the current line
Ctrl+U – erases the whole line
Ctrl+R – type to bring up a recent command
!! – repeats the last command
exit – log out of current session
* use with extreme caution.

22



When there is a problem:

Linux = be root

Windows = reboot