Introduction to UNIX/Linux

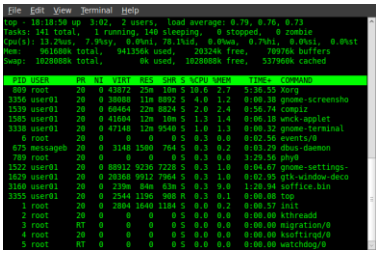
Biochemistry Boot Camp 2021
Session #3
Nicholas Fitzkee
nfitzkee@chemistry.misstate.edu

1

Some terms

- Operating system (OS)

Command-line interface (CLI)



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

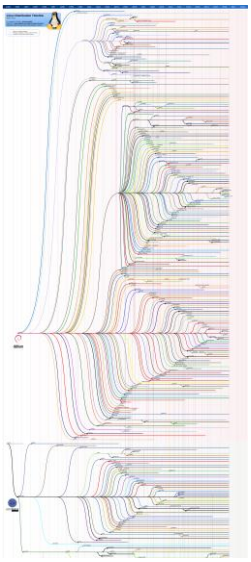
Flavors of Linux

- "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

CentOS

- **Ubuntu:** User friendly, a good mix of "cutting edge" updates and stability

ubuntu

- **Debian:** Cutting edge, but stable software is available for installation too

debian

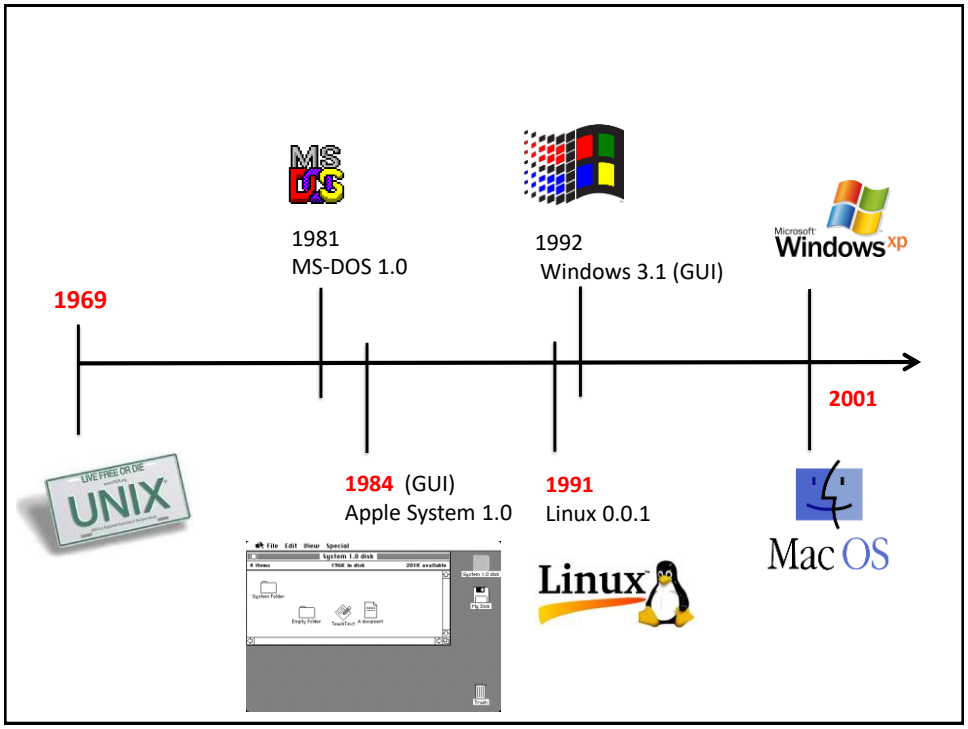
8

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

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

Process Management

ps – display your currently active processes
top – display all running processes
kill pid – kill process id *pid*
killall proc – kill all processes named *proc* *
bg – lists stopped or background jobs; resume a stopped job in the background
fg – brings the most recent job to foreground
fg n – brings job *n* to the foreground

File Permissions

chmod octal file – change the permissions of *file* to *octal*, which can be found separately for user, group, and world by adding:

- 4 – read (r)
- 2 – write (w)
- 1 – execute (x)

Examples:
chmod 777 – read, write, execute for all
chmod 755 – rwx for owner, rx for group and world
For more options, see **man chmod**.

SSH

ssh user@host – connect to *host* as *user*
ssh -p port user@host – connect to *host* on port *port* as *user*
sftp user@host - connect to *host* as *user* for file transfer
gftp – graphical file transfer client

Searching

grep pattern files – search for *pattern* in *files*
grep -r pattern dir – search recursively for *pattern* in *dir*
command | grep pattern – search for *pattern* in the output of *command*
locate file – find all instances of *file*
find . -name "pattern" – search for the file named *pattern* in the current directory (or below); *pattern* can contain wildcards (e.g. "*")

System Info

date – show the current date and time
cal – show this month's calendar
uptime – show current uptime
w – display who is online
whoami – who you are logged in as
finger user – display information about *user*
uname -a – show kernel information
cat /proc/cpuinfo – cpu information
cat /proc/meminfo – memory information
man command – show the manual for *command*
df – show disk usage
du – show directory space usage
free – show memory and swap usage
whereis app – show possible locations of *app*
which app – show which *app* will be run by default

Compression

tar cf file.tar files – create a tar named *file.tar* containing *files*
tar xf file.tar – extract the files from *file.tar*
tar czf file.tar.gz files – create a tar with Gzip compression
tar xzf file.tar.gz – extract a tar using Gzip
zip -r file.zip files – create a Windows-compatible zip archive
unzip file.zip – extract zip archive
gzip file – compresses *file* and renames it to *file.gz*
gzip -d file.gz – decompresses *file.gz* back to *file*

Network

ping host – ping *host* and output results
whois domain – get whois information for *domain*
dig domain – get DNS information for *domain*
dig -x host – reverse lookup *host*
wget file – download *file*
wget -c file – continue a stopped download

Lab Utilities

NMR Viewers: **sparky** or **marDraw** or **nmrj** or **analysis**
pipe2ucsf file.ft2 file.ucsf – convert NMRPipe spectrum to UCSF format
/home/database/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 *script* 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.

Adapted from FOSSWire.com, http://fosswire.com/post/2007/08/unixlinux-command-cheat-sheet/

19

File Commands

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

20

Process Management

ps – display your currently active processes
top – display all running processes
kill pid – kill process id *pid*
killall proc – kill all processes named *proc* *
bg – lists stopped or background jobs; resume a stopped job in the background
fg – brings the most recent job to foreground
fg n – brings job *n* to the foreground

File Permissions

chmod octal file – change the permissions of *file* to *octal*, which can be found separately for user, group, and world by adding:

- 4 – read (r)
- 2 – write (w)
- 1 – execute (x)

Examples:
chmod 777 – read, write, execute for all
chmod 755 – rwx for owner, rx for group and world
For more options, see **man chmod**.

SSH

ssh user@host – connect to *host* as *user*
ssh -p port user@host – connect to *host* on port *port* as *user*
sftp user@host – connect to *host* as *user* for file transfer
gftp – graphical file transfer client

Searching

grep pattern files – search for *pattern* in *files*
grep -r pattern dir – search recursively for *pattern* in *dir*
command | grep pattern – search for *pattern* in the output of *command*
locate file – find all instances of *file*
find . -name "pattern" – search for the file named *pattern* in the current directory (or below); *pattern* can contain wildcards (e.g. "*")

System Info

date – show the current date and time
cal – show this month's calendar
uptime – show current uptime
w – display who is online
whoami – who you are logged in as
finger user – display information about *user*
uname -a – show kernel information
cat /proc/cpuinfo – cpu information
cat /proc/meminfo – memory information
man command – show the manual for *command*
df – show disk usage
du – show directory space usage
free – show memory and swap usage
whereis app – show possible locations of *app*
which app – show which *app* will be run by default

Compression

tar cf file.tar files – create a tar named *file.tar* containing *files*
tar xf file.tar – extract the files from *file.tar*
tar czf file.tar.gz files – create a tar with Gzip compression
tar xzf file.tar.gz – extract a tar using Gzip
zip -r file.zip files – create a Windows-compatible zip archive
unzip file.zip – extract zip archive
gzip file – compresses *file* and renames it to *file.gz*
gzip -d file.gz – decompresses *file.gz* back to *file*

21

Network

ping host – ping *host* and output results
whois domain – get whois information for *domain*
dig domain – get DNS information for *domain*
dig -x host – reverse lookup *host*
wget file – download *file*
wget -c file – continue a stopped download

Lab Utilties

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 *script* 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

23