Introduction to Linux

Dr. Sreenu Vattipally
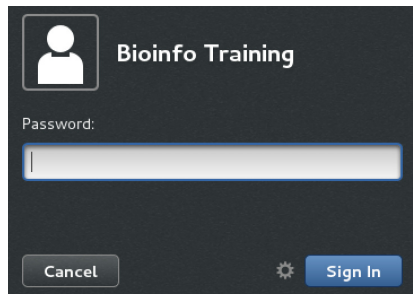
MRC-University of Glasgow Centre for Virus Research

History of Linux

- Unix is an operating system
- Developed by AT&T employees at Bell Labs (1969-1971)
- MULTICS: Multiplexed Information and Computing Service
- UNICS: Uniplexed Information and Computing Service
- Later renamed to Unix
- Rewritten in the C programming language in 1972
    - C programming language was developed for the Unix OS
    - C rendered Unix portable

What is Linux?

- ▶ Unix available outside AT&T (1975)
- ▶ Because of AT&T core business, they were not allowed to sell software
- ▶ Universities, government agencies and corporate companies got Unix and source code
- ▶ Free license and SOURCE CODE to run on MainFrames
- ▶ First release 1975: System 5
- ▶ Linux is Unix like kernel developed by Linus Torvalds
- ▶ Various OS distributions are release from then on
    - ▶ Open source BSD Linux
    - ▶ Closed source: Solaris(Sun/Oracle), AIX(IBM), HP/UX (Hewlett-Packard), Irix (Silicon Graphics)
    - ▶ Mixed source: Mac OSX (Apple)
- ▶ These are all Unix like systems
- ▶ Mobile devices are unix
    - ▶ IOS, Android, Kindle etc...

Logging in

- ▶ User:manager
- ▶ Password:manager

We will be using Ubuntu Linux in this course.

Terminal and Shell

Terminal: Terminal is program that lets you interact with the shell.

Shell: Program runs in a terminal, acts as an interface between the user and the system.

Common Shells

- ▶ sh: Thompson shell (1971)
- ▶ sh: Bourne shell (1977) (replaced previous shell)
- ▶ csh: C shell (1979)
- ▶ tcsh: Tabbed C shell (1979)
- ▶ ksh: Korn shell (1982)
- ▶ bash: Bourne-Again shell (1987)
- ▶ zsh: Z shell (1990)

Shells can be nested.

ANATOMY OF A COMMAND

```
command option(s) arguments
```

- ▶ Command always is a single word (or words stitched with "_" or "-")
- ▶ Options have to follow command
- ▶ Options can start with a single hyphen and a character or a double hyphen and a word
- ▶ Single character options can be combined
- ▶ Some times options need a value (cut -f 1)
- ▶ Argument can be one or two inputs
- ▶ You can write more than one command separating with a semicolon;

Help!

Manual pages: man

- ▶ Most of the commands have manual pages
- ▶ Gives summary of a command
- ▶ Gives all available options
- ▶ Gives examples
- ▶ Gives developer information

Information: info

- ▶ More detailed information than man
- ▶ Available in newer versions

First Commands

- ▶ ls: list directory contents
- ▶ pwd: present working directory
- ▶ cd: change directory
- ▶ mkdir: make directory
- ▶ rmdir: remove directory
- ▶ touch: timestamp change or create an empty file
- ▶ rm: remove file(s)/directory(ies)
- ▶ cp: cp file(s)/directory(ies)
- ▶ mv: move or rename file/directory
- ▶ ln: create a link to file/directory

FILE COMMANDS

- ▶ cat: concatenates and prints files
- ▶ pg: shows file contents pagewise
- ▶ more: shows file content
- ▶ less: shows file content
- ▶ head: shows file's first few lines
- ▶ tail: shows file's last few lines

FILE PERMISSIONS

- Each file has it's own permissions
- `drwx rwx rwx`: user-group-other (ugo) permissions
- default permissions can be changed

Change permissions

```
chmod u=rwx,g=r,o=r file
chmod ug+w file
chmod go-r file
chmod ugo+r file
chmod a+r file
```

Permission values
r=4,w=2,x=1
```
chmod 755 file
chmod 600 file
```

Default permission are set with umask command

CHANGE OWNER AND GROUP

- ▶ Owner and group can be changed with `chown`
- ▶ Only group can be changed with `chgrp`
- ▶ Recursive changes can be made with "-R" option

```
chown user:group file
chown :group file
chgrp group file
```

Only root can change "user"

EDITING FILES

We need editors to create/modify/delete files. Here are some of the pre-installed editors.

- ed
- emacs
- vi
- pico
- nano
- gedit
- nedit, etc...

Text editor:nano

- ▶ Ctr+G: Help
- ▶ Ctr+O: Write out
- ▶ Ctr+R: Read file
- ▶ Ctr+Y: Goto previous page
- ▶ Ctr+V: Goto next page
- ▶ Ctr+K: Cut text
- ▶ Ctr+K: Uncut text
- ▶ Ctr+X: Exit

Compressing and archiving

Linux has many commands to compress and archive files/directories
Text files are best suited for compression

- ► compress/uncompress
- ► zip/unzip
- ► gzip/gunzip
- ► bzip2/bunzip2

ToDo
Compare runtime and percentage of compression of the above commands

Process control

Programs can be run in foreground or background

- `Ctr+c`
- `Ctr+z`
- `&`
- `bg`
- `fg`
- `jobs`
- `ps`
- `top`
- `nohup`
- `kill`
- `killall`

Do not send an interactive program to background

I/O control: Output

- By default stdout is sent to terminal/screen
- Can be redirected to a file ($>$)
- Can be appended to a file ($»$)

Should not use same file as input and output (it will remove the content or go into infinite loop)

I/O control: Input

- By default stdin is taken from keyboard
- Can be taken from a file ($<$)

I/O CONTROL: |(PIPES)

▶ Pipe makes output of one command as input to another command

▶ One can use as many pipes as needed

▶ Pipes can be used to make command strings

▶ Pipe can only take input from a command, not from a file

Text processing

- wc
- cut
- sort
- grep
- uniq
- sed
- tr
- rev

Some terminal shortcuts

- ▶ Up/Down arrows: Previous commands
- ▶ !!: Reruns previous command
- ▶ Tab: Auto complete
- ▶ Tab+Tab: All available options
- ▶ Ctr+a: Move cursor to start of line
- ▶ Ctr+e: Move cursor to end of line
- ▶ Ctr+ : Alternates between terminals
- ▶ Ctr+l: Clear screen ((or Command+k on Mac)
- ▶ Ctr+c: Terminates the running program
- ▶ Ctr+z: Suspends the running program
- ▶ Ctr+w: Removes a previous word
- ▶ Ctr+d: Logout
- ▶ Ctr+d(in a command): Removes a character
- ▶ Ctr+u: Removes till the beginning