UNIVERSITY OF CAMBRIDGE

MRC | Cognition and Brain Sciences Unit

Introduction to Linux at the CBSU
Part 1
Basics and the CLI

Jeff Berry

MRC Cognition and Brain Sciences Unit, Computing group

Motivation – Why do you need this?

- **Shared CBSU environment**
  - Central storage: images, scripts, documents, etc.
    - Backup (automatic)

- **Computing Cluster**
  - High-Performance Computing: analysis
    - Speed (only if you know the “tricks”)

Ways and Means

- **Graphical Desktop Environment via VNC (or X2GO)**
  - Similar to other desktop OS GUIs (Windows, Mac)
  - Menu-driven
  - Can be slow, especially when operating on many files
  - Available on HPC login nodes, but not on compute nodes

- **Command Line Interface (CLI)**
  - Via ‘terminal’
  - Similar to DOS command prompt
  - Available natively in Mac OS X (Mac sits on top of a BSD Unix)
  - Fast, once you learn how it works
  - Used for scripts on compute nodes

Getting Started

Connect to the Cluster via SSH

- **PuTTY (for Windows)**
  - Search for PuTTY
  - Host Name = ‘login’ ; Port = 22 ; Connection type=ssh
    - may also specify specific login node, eg login15
  - Opens a window: ‘login as:’
    - Use trainXXlinux
  - enter password
  - Congratulations! If you see this (or something like it):

```
[trainXXlinux@loginYY ~]$
```

You’re logged on!

Getting Started

Connect to the Cluster via SSH

- **SSH (from Mac or another Linux machine)**
  - From terminal window (on a Mac search for it or find it in 'Utilities')
  - *ssh login* - defaults to existing account.
    - Use *ssh USER@login* for a different account
    - may also specify specific login node, eg *ssh login15* or *ssh fred@login15*
  - should ask for password but not 'login as:'
    - But! It is possible to set up automatic authentication. (The training accounts have this on most nodes)
  - Congratulations (again)! If you see this (or something like it):

```
[trainXXlinux@loginYY ~]$
```

You're logged on!

Getting Started

Connected, Now What?

- **Some Linux Conventions and Things to Know**
  - linux is case sensitive – Home, HOME, and home are different
  - Directory is a hierarchical tree starting at root (= /)
    - /
      - /home
        - /home/trainXXlinux
      - /var
        - /var/log
    - . is the current directory
    - .. is the parent directory
    - ~ is the logged in user's home directory

```
[train01linux@login12 ~]$
```

username machine name ↑ Your home directory!

Getting Started

Connected, Now What?

- **Useful commands to try**
  - *pwd* – print working directory (where am I?)
    - note the full path starting with /
  - *cd* – change directory. Give it an argument after a space
    - *cd ..* (change to parent directory) then try pwd again
    - *cd ~* (change to home directory) then pwd
    - *cd /imaging/trainXXlinux*
      - NB – this is a 'fully qualified path' rather than a 'relative path'
  - *ls* – list the contents of your current directory

    `LinuxClass`

    There is a single entry

Getting Started

Connected, Now What?

- **More useful stuff**
  - most commands take options/flags
    - -X - a single dash and a single letter for short form
    - --Xpect – a double dash and a word for long form
  - *ls -F*
    - note the trailing "/" in the output which indicates a directory
  - *cd LinuxClass*
    - this is a relative path, changing into a directory relative to the current directory
    - tab completion - type *cd Lin* and then hit the tab key, and it should autocomplete

Getting Started

Connected, Now What?

- **Things you can do and learn with ls**
  - *ls -al*
    - multiple flags only need a single leading -
    - -a = all files, -l = long listing
    - the command ‘man ls’ will show the manual page for ls. This works for (almost) any command

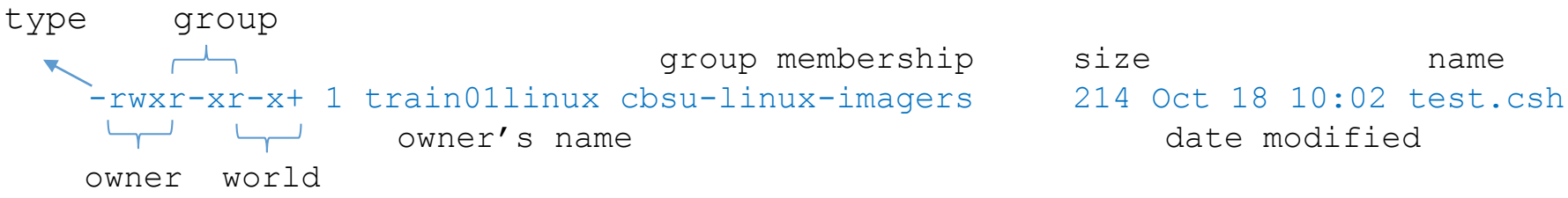
```
drwxr-sr-x+ 2 train01linux cbsu-linux-imagers      12 Oct 18 10:02 .
drwxr-s---+ 3 train01linux cbsu-linux-imagers       3 Oct 18 10:02 ..
-rwx------+ 1 train01linux cbsu-linux-imagers  385158 Oct 18 10:02 dicom.dcm
```

Notice the two special directories (. and ..) are listed, thanks to -a. As for the rest …

Getting Started

Linux Filenames, Users, Groups, and Permissions

- Every user has a username and one or more group memberships
- Files can have arbitrary names – extensions are not required
- Files have an owner and a single group membership
- Files (and directories) have three sets of permissions
  - owner, group, other/world
- Files (and directories) have three types of permissions
  - read, write, execute



Getting VNC Connected

Starting a VNC Server

- *vncserver* (*man vncserver*)
  - if it is the first time, it will ask you to set a password.
    - Do not use your usual password!
  - Output includes "..... on display loginXX:Y"
    - :Y is the display number

Connecting to a VNC Server

- Find TurboVNC on your computer and start it
- For server, use the output from the previous command, eg loginXX:Y
- Use the password you setup when running vncserver
- If a desktop window appears, then you've got your vncserver running!

More detail and screenshots at:

http://intranet.mrc-cbu.cam.ac.uk/computing/cluster-access/

Getting VNC Connected

Why use a VNC server?

- Support for graphics
- Multiple windows
- Menus make some things easier

Menus and Tools

- Accessories:
  - gedit
  - Take Screenshot
- System Tools
  - File Browser – **nautilus**
  - System Monitor
  - Terminal – gives a CLI

X2GO (coming soon)

X2GO is a VNC replacement

- Has similar functionality
- Does not require a server!
- Will be available on the new cluster, not the current one.

Shells, Environments, and More

- **Shell**
  - Encapsulated ("in a shell") environment
    - environment variables – *env* command
      - 'pipe' the output of one command to the input of another
        - *env | more*
      - redirect the output to a file
        - *env > YourFileHere*
        - or append it with >>
      - look at output with an editor or
        - *cat FileName* (and if it is more than one page ...)
    - *echo $OS*
    - *echo $PATH*
    - *echo $SHELL*

Shells, Environments, and More

Command editor

- up and down arrows
- tab completion
- command line is editable
- !<part command> - execute most recent matching command from history
  - *ls -alt; cd .. ; !ls*
- Break: Ctrl+C
  - terminate running program and return to prompt

Shells, Environments, and More

- **Variables**
  - environment variables propagate – regular variables don't
    - *set MYVAR="LocalVar"*
    - *echo $MYVAR*
    - *tcsh*
    - *echo $MYVAR*
    - *exit*
    - *unset $MYVAR*
    - *setenv MYVAR "EnvVar"*
    - *echo $MYVAR*
    - *tcsh*
    - *echo $MYVAR*

Shells, Environments, and More

Aliases:

- short-cuts to command/script/program
- *alias* (with no arguments to show existing aliases)
- *alias <alias> <aliased commend>*
  - *alias cdi 'cd /imaging/trainXXlinux'*
  - *cd ~; ls ; cdi; ls*
- *unalias <alias>*
  - *unalias cdi; cdi*

Unix Primer – Make your life easier!

- **File browser - nautilus**
  - Create Bookmarks
  - Once a browser, always a browser

- **Symbolic links**
  - References to folders/files – not a copy
  - Create: *ln -s <original> <link>*
  - Delete: *rm <link>, unlink <link>*

Unix Primer – Useful Commands

Help/Info	*man, -h, --help*	help MANual
	man -k	find similar or related commands
	pwd	Present(print) Working Directory
	echo	display text or variable
Listing	*ls*	LiSt directory contents
	cat	conCATenate file to display
	top	show processes
	id -G -n <username>	list group memberships
Searching	*locate*	find files
	which	which executable will a command use
Navigate	*ssh*	connect to a linux machine
	cd	Change Directory

Unix Primer – Useful Commands

Manipulate	*mkdir*	MaKe Directory
	cp (-r)	CoPy (-r for recursively)
	mv	MoVe/rename
	rm (-r)	ReMove/delete irreversibly (and -r)
	chown (-R)	CHange OWNer (-R for recursively)
	chmod (-R)	CHange MODe/permissions (and -R)
	vncserver	create/kill VNC session
	dos2unix, unix2dos	convert text file (EOL)
	wget <URL>	download
	tar	Tape ARchive – -c create tar archive file -x extract from tar file -z compress('zip') or uncompress

Unix Primer – Useful Commands (toolkit)

grep <filename>	Print lines with pattern fit
grep -c pttrn <filename>	Print number of lines with pattern fit
awk '{print $2}' <filename>	Print second entry only (of each line)
sed 's/ptrn1/ptrn2/' <filename>	Print content with ptrn1 replaced with ptrn2 (first instance only)
sed 's/ptrn1/ptrn2/g' <filename>	Print content with ptrn1 replaced with ptrn2 (globally)
printf "format" inpt	Formatted print (~ MATLAB)
echo "statement" | bc	Calculator
seq n	Sequence of integer up to n (useful for loops)

Further Information

- **Linux:**
  - Basics:
    - http://imaging.mrc-cbu.cam.ac.uk/methods/unixsurvivalguide
    - http://www.ee.surrey.ac.uk/Teaching/Unix
  - Scripting:
    - http://tldp.org/LDP (both Beginner and Advanced)