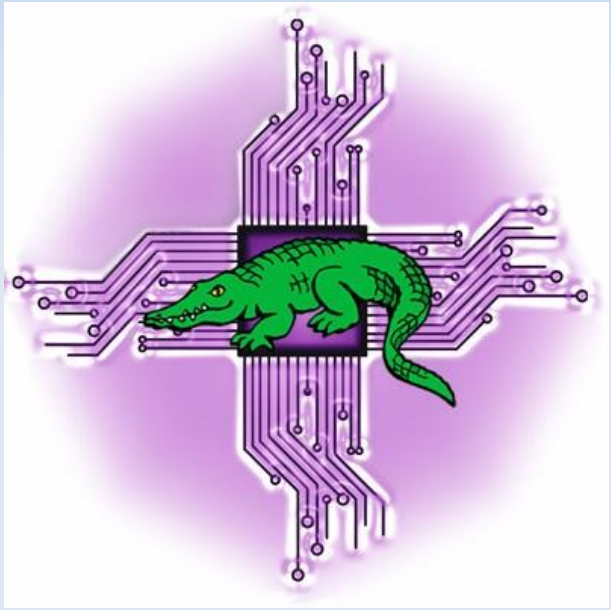
More Linux on Spartan

COMP90024 Cluster and Cloud Computing



University of Melbourne, 21st March 2023

lev.lafayette@unimelb.edu.au

Capability and Performance

Part One of this workshop introduced the Linux CLI on an HPC system.

Part Two will extend the command knowledge and introduce scripting.

Scripting allows for the automation of tasks.

When combined with wildcards, looping logic, conditional statements, etc relatively short and simple commands can be constructed that would otherwise be very time-consuming, robotic, and demanding involvement from the user.

Archiving and Compression

UNIX-like systems have several archiving and compression options often referred to as “tarballs”. The tar command (tape archive) is the archiver; compression systems include gzip, bzip2, xz and more.

To determine the type of a file: file class.tar.gz
To review the contents of a tarball: tar tf class.tar.gz

To recover from a tarbomb: tar tf tarbomb.tar | xargs rm -rf
To extract a tarball: tar xvf class.tar.gz

To create a tarball: tar cvfz name.tar.gz directory/
To create a bzip2 tarball: tar cvfj class.tar.bz2 class/
To create a xz tarball: tar cvfJ class.tar.xz class/

Advanced Redirections

Redirections can be modified by placing a file descriptor (fd) next immediately before the redirector. These fd numbers are 0 (standard input, keyboard), 1 (standard output, display), 2 (standard error, display).

Standard error can also be redirected to the same destination that standard output is directed to using 2>&1; it merges stderr (2) into stdout (1).

ls -d seismic 2> error.txt

ls -d seismic 2>&1 error.txt

The tee command applies command1 and command2 to file
who -u | tee whofile.txt | grep username

File Attributes and Types

The ls -l command illustrates file ownership (user, group), file type, permissions, and date when last modified.

The first character is type; a "-" for a regular file, a "d" for a directory, and "l" for a symbolic link. Devices are expressed like files: "b" for block devices, "c" for character devices.

Permissions are expressed in a block of three for "user, group, others" and for permission type (read r, write w, execute x).

An "s" in the execute field indicated setuid. There is "t", "save text attribute", or more commonly known as "sticky bit" in the execute field allows a user to delete or modify only those files in the directory that they own or have write permission for.

Change Mode

The change permissions of a file use the chmod command. To chmod a file you have to own it. The command is : `chmod [option] [symbolic | octal] file`. For options, the most common is `-R` which changes files and directories recursively. In symbolic notation user reference is either u (user, owner), g (group), o (others), a (all). Operation is either + (add), - (remove), = (only equals), permissions are r (read), w (write), x (execute).

In octal notation a three or four digit base-8 value is the sum of the component bits. The value "r" adds 4 in octal notation (binary 100), "w" adds 2 in octal notation (binary 010) and "x" adds 1 in octal notation (binary 001). For special modes the first octal digit is either set to 4 (setuid), 2 (setgid), or 1 (sticky).

Links and File Content

The ln command creates a link, associating one file with another. There are two basic types; a hard link (the default) and a symbolic link.

A hard link is a specific location of physical data, whereas a symbolic link is an abstract location. Hard links cannot link directories and nor can they cross system boundaries; symbolic links can do both of these.

With a hard link: File1 -> Data1 and File2 -> Data1 . With a symbolic link: File2 -> File1 -> Data1

The general syntax for links is: ln [option] source destination. e.g., ls -s file1 file2

File Manipulation

The “cut” command copies a section from each line of a file, based on the arguments parsed to it, whereas “paste” merges lines of files together. See also “join”

```
cut -d',' -f3 shakes.csv > latitude.txt
cut -d',' -f4 shakes.csv > longitude.txt
paste -d " " latitude.txt longitude.txt > shakeslist.txt
```

The “sort” command will organise a text file into an order specified by options. The general syntax is sort [option] [input file] -o [filename]. Options include -b (ignore beginning spaces), -d (use dictionary order, ignore punctuation), -m (merge two input files into one sorted output, and -r (sort in reverse order).

Shell Scripts

Shell scripts combine Linux commands with logical operations.

The most basic form of scripting simply follows commands in sequence (e.g., /usr/local/common/AdvLinux/backup1.sh). Variables and command substitutions add to a script (e.g., /usr/local/common/AdvLinux/backup2.sh). Variables use ($) to refer to its value. ${var}bar, invoke the variable, append "bar". Command substitution comes in the form of $ (command).

Quotes disable and encapsulate some content; single quotes interprets everything literally. e.g.,
echo "There are $(ls | wc -l) files in $(pwd)"
echo 'There are $(ls | wc -l) files in $(pwd)'

Scripts with Loops

Scripting allows for loops (for/do, while/do, util/do). The for loop executes stated commands for each value in the list. The while loop allows for repetitive execution of a list of commands, as long as the command controlling the while loop executes successfully. The until loop executes until the test condition executes successfully.

```
for item in ./*.jpeg ; do convert "$item" "${item%.*}.png" ; done
while read line; do sleep 5; ./setquota.sh $line; done < quotalist.txt
x=5; until [ $x -le 0 ]; do echo "Until-do count down $x"; x=$(( $x - 1 )); done
```

Scripts with Conditional Branches

A set of branched conditions can be expressed through an if/then/fi structure. A single test with an alternative set of commands is expressed if/then/else/fi. Finally, a switch-like structure can be constructed through a series of elif statements in a if/then/elif/elif/.../else/fi

Conditionals can also be interrupted and resumed using the “break” and “continue” statements. The break command terminates the loop (breaks out of it), while continue causes a jump to the next iteration (repetition) of the loop, skipping all the remaining commands in that particular loop cycle.

A variant on the conditional is the “case” statement. The first match executes the listed commands.

Shell Scripting and HPC Jobs

All shell commands and all shell scripting operations can be included in Slurm job submission.

See: /usr/local/comm/AdvLinux/NAMD/drugdock.slurm for an example which makes use of simple shell commands (cp, mv, rm), globbing (*), variables (date2), shell substitution ($(date +F%-H%.M%), redirects (2>), loops (for loop in {1..5}) etc.

A heredoc is a file or input literal, a section of source code that is treated as a separate file. Heredocs can also be used however to create Slurm scripts. (e.g, /usr/local/common/AdvLinux/heres/herescript.sh).


THANKS FOR WATCHING
& LISTENING PATIENTLY
makeameme.org