The day-to-day Unix Commands

Tian Guo
CICS Umass Amherst

Before you start

- Fork and clone the unix git repo
  - From CSWomenUMass
- Pre workshop survey
  - You can find the google doc link inside the CSWomenUMass/unix issues

Overview

- Objective: Get familiar with the Unix commands that
  - allow you to navigate directory, manipulate file, manage permission and locate files
- Prerequisite: A unix-like system
  - Refer to bootcamp to get a VM running

The beginning

- Shell: the most basic program run inside the terminal
- Terminal: runs a shell
- How it works:
  - Launch your favorite terminal

  - Type a command (case sensitive), hit Enter!
  - Shell processes the command and returns output
- Exercises:
  - Fire up your terminal
  - Exit from your terminal

ls

- Function: List directories and files
- Usage: ls *directory* ← Default to current directory

  ↑

  Command

- Exercises*:*
  - Check what is inside your current directory
  - Check what is inside directory */bin*

ls

- Function: List directories and files
- Usage: ls *directory* ← Default to current directory

  ↑

  Command

- Exercises*:*
  - Check what is inside your current directory: ls
  - Check what is inside directory */bin:* ls /bin

man

- Function: Look up information about commands
- Usage: *man command*
- How to navigate the man page?
  - Flip pages: Spacebar(next page) and b key(previous page)
  - Search: /string(forward) v.s. ?string(backward)
  - Repeat previous search: hit n/N(reverse)
  - Exit: hit q key
- Exercises:
  - Look up the usage of *ls*
  - *Check out the flags -l -h -a usages*

Anatomy of a command

- Executable: program name you want to run
  - ls #list files in the current directory
- Flags: options for the command
  - ls -alh #show everything inside (-a) with a human readable way (-h) in long format (-l)
- Arguments: additional parameters for the command
  - ls /bin # list the contents of the directory /bin

cd

- Function: Move around directories
- Usage: cd *directory*
  - There is no place like ~ , / is just root
  - .. contains . while . is the current directory
  - - is just the previous working directory
- Exercises:
  - Go to /bin and list the files inside
  - Go back to your previous directory
  - Move up a directory
  - Go back to your home directory

cd

- Function: Move around directories
- Usage: cd *directory*
  - There is no place like ~ , / is just root
  - .. contains . while . is the current directory
  - - is just the previous working directory
- Exercises:
  - Go to /bin and list the files inside: cd /bin; ls
  - Go back to your previous directory: cd -
  - Move up a directory: cd ..
  - Go back to your home directory: cd ~

pwd

- Function: Display both path and name of the directory
- usage: pwd
  - The return is called an absolute path (always start with /)
  - Relative path: ~/Desktop
- Exercises:
  - Find out what your home directory is

pwd

- Function: Display both path and name of the directory
- usage: pwd
  - The return is called an absolute path (always start with /)
  - Relative path: ~/Desktop

- Exercises:
  - Find out what your home directory is: cd ~; pwd

mkdir

- Function: make a new directory
- Usage: mkdir *directory*
  - Works with both absolute and relative path
- Exercises:
  - Create a new directory named *myverylongdirectoryname*
  - Create a new directory named *myverylongdirectorynamechild* inside *myverylongdirectoryname*
  - *List the long format information about the new directories, How do you know they are directories but not files?*

mkdir

- Exercises:
  - Create a new directory named *myverylongdirectoryname*

    mkdir myverylongdirectoryname
  - Create a new directory named *myverylongdirectorynamechild* inside *myverylongdirectoryname:*

    cd myverylongdirectoryname;

    mkdir myverylongdirectorynamechild
  - *List the long format information about the new directories, How do you know they are directories but not files?*

    ls -l myverylongdirectoryname; and look for **d**rwxrwxr-x

Some command line tips

- Use <tab> to auto complete
  - Super useful for long name, such as *myverylongdirectoryname*
  - <tab> once or <tab> twice
- Use <ctrl+c> to cancel what you have typed
  - Or quit some programs

- Exercises
  - Go to *myverylongdirectorynamechild* directory using <*tab*>
  - Try <*ctrl*+*c*>

History

- Function: Shows a list of commands you have run
- Usage: history
- Use <ctrl+r> to search a previous command
  - Specify a pattern to search
  - Find the most recent command first
  - Keep pressing <ctrl+r> to iterate all the matched commands
- Exercises
  - Check how many commands you have typed so far
  - Search for *pwd* and execute it again

touch

- Function: Creates an empty file
- Usage: touch *empty_file_name*
  - Naming conventions: use underscore for filename; use hyphen for directory name
  - Example*: my-very-long-directory-name*

- *Exercises:*
  - Create two empty file named *first_empty_file_name, second_empty_file_name* inside *my-very-long-directory-name*
  - List the long format information about this two files

touch

- *Exercises:*
  - Create two empty file named *first_empty_file_name, second_empty_file_name* inside *my-very-long-directory-name*

    cd my-very-long-directory-name

    touch first_empty_file_name second_empty_file_name
  - List the long format information about this two files

    ls -l *_*empty_file_name*

cp

- Function: Copy directories and files
- Usage:
  - Copy a directory: cp -r old-directory new-directory
  - Copy a file: cp old-file new-file
- Exercises:
  - Make a backup directory of *my-very-long-directory-name-child* so that they have the same parent directory
  - Make a backup file of *first_empty_file_name*

cp

- Function: Copy directories and files
- Usage:
  - Copy a directory: cp -r old-directory new-directory
  - Copy a file: cp old-file new-file
- Exercises:
  - Make a backup directory of *my-very-long-directory-name-child* so that they have the same parent directory:

    cp -r *my-very-long-directory-name-child my-very-long-directory-name-child-bk*
  - Make a backup file of *first_empty_file_name*

    cp first_empty_file_name first_empty_file_name_bk

mv

- Function: Moving directories and files from one location to another
- Usage: mv *old new*
- *Exercises:*
  - Rename the file *first_empty_file_name* to *new_empty_file_name*
  - Move the file *new_empty_file_name* inside directory *my-very-long-directory-name-child*
  - *Rename my-very-long-directory-name-child to my-very-long-directory-name-baby*

mv

- *Exercises:*
  - Rename the file *first_empty_file_name* to *new_empty_file_name*

    mv first_empty_file_name new_empty_file_name
  - Move the file *new_empty_file_name* inside directory *my-very-long-directory-name-child*

    mv new_empty_file_name my-very-long-directory-name-child/
  - *Rename my-very-long-directory-name-child to my-very-long-directory-name-baby*

    mv my-very-long-directory-name-child my-very-long-directory-name-baby

rm

- Function: Remove directories and files
- Usage:
  - Remove a file: rm -i file_name
  - Remove multiple files: rm -i
  - Remove a directory: rm -ir directory-name
- One dangerous command: rm -rf /
- Exercises:
  - Organize your working directory: removing files or directories that are not needed

rmdir

- Function: Remove an empty directory
- Usage: rmdir empty-directory

- Exercises:
  - Find an empty directory and use rmdir to remove it

rmdir

- Function: Remove an empty directory
- Usage: rmdir empty-directory

- Exercises:
  - Find an empty directory and use rmdir to remove it

    use ls to check if a directory is empty first

    rm empty-directory

find

- Function: A much more efficient way to find files or directories than simply using ls
  - Look through all accessible subdirectories
- Usage : find starting-directory criteria actions
- Example: find . -type d -empty -print
- Exercises:
  - Find empty files in the current directory
  - Find all files and directories with names starting with string *my*
  - Find everything with names contain string *my*
  - Find all empty directories and delete them

find

- Exercises:
  - Find empty files in the current directory

    find . -type f -empty -print
  - Find all files and directories with names starting with string *my*

    find . -name “my” -print*
  - Find everything with names contain string *my*

    *find . -name “*my*” -print*
  - Find all empty directories and delete them

    find . -type d -empty -ok rm {} \;

locate

- Function: locating Unix system files
- Usage: locate filename
  - Returns a list of all system files that contain “filename” in them
- Exercises:
  - Locates all files that contain *passwd*
  - How do you get the same list using find?
  - Which way is faster?

locate

- Function: locating Unix system files
- Usage: locate filename
  - Returns a list of all system files that contain “filename” in them
- Exercises:
  - Locates all files that contain *passwd*

    locate passwd
  - How do you get the same list using find?

    sudo find / -name “*passwd*” -print
  - Which way is faster?

    locate

Ownership and Permissions

- Three levels of file ownership
  - User(u), group(g), other(o), or all(a)
- For each level, there are three permission categories
  - Read (r), write (w), execute (e)
  - Use 0/1 to indicate the permission
- Example
  - -rw-rw-r-- 1 tian tian 0 Nov 3 13:35 empty_file
  - Binary equivalent: 110110100
  - Numeric equivalent : 664

chmod

- Function: change who has what access
- Usage for file
  - set permissions: chmod u=rwx,g=rw,o=r new_file
  - add permissions: chmod o+w new_file
  - remove permissions: chmod go-w new_file
- Usage for directory: make changes recursively
  - chmod -R go-rwx *
- Exercises:
  - What is numeric equivalent of rwxr-xr-x permission?
  - Create a file new_file with rwxr-xr-x in the numeric way

chmod

- Exercises:
  - What is numeric equivalent of rwxr-xr-x permission?

    755
  - Create a file new_file with rwxr-xr-x in the numeric way

    chmod 755 new_file

chgrp

- Function: change the file/directory group association
- Usage:
  - chgrp newgroup file
  - chgrp -R newgroup directory
- Exercises:
  - Change one file to root group
  - Change a directory to root group

chgrp

- Exercises:
  - Change one file to root group

    sudo chgrp root empty_file

    Output: -rw-rw-r-- 1 tian root 0 Nov 3 13:35 empty_file

  - Change a directory to root group

    sudo chgrp -R root myverylongdirectoryname

    Output: drwxrwxr-x 3 tian root 4096 Nov 3 14:53 myverylongdirectoryname

chown

- Function: change the file/directory ownership
- Usage:
  - chown newowner file
  - chown -R newowner directory
- Exercises:
  - Change the owner of a file to root
  - Change the owner of a directory to root

chown

- Exercises:
  - Change the owner of a file to root

    sudo chown root empty_file

    Output: -rw-rw-r-- 1 root root 0 Nov 3 13:35 empty_file

  - Change the owner of a directory to root

    sudo chgown -R root myverylongdirectoryname

    Output: drwxrwxr-x 3 root root 4096 Nov 3 14:53 myverylongdirectoryname

cat

- Function: display the full contents of a file
- Usage: cat filename

```
> cat days.txt
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
```

head and tail

- Functions: Print out the top or bottom of a file
- Usages:
  - head -n X filename to print out the top X lines
  - tail -n X filename to print out the bottom X lines
- Exercises:
  - Display the first 20 lines of texts from alice.txt
  - Display the last 20 lines of texts from alice.txt

head and tail

- Functions: Print out the top or bottom of a file
- Usages:
  - head -n X filename to print out the top X lines
  - tail -n X filename to print out the bottom X lines
- Exercises:
  - Display the first 20 lines of texts from alice.txt

    head -n 20 alice.txt
  - Display the last 20 lines of texts from alice.txt

    tail -n 20 alice.txt

Clear

- Function: Clear the screen
- Usage: clear
- Exercises:
  - Clear your terminal
  - Check what commands you have learnt so far

Clear

- Function: Clear the screen
- Usage: clear
- Exercises:
  - Clear your terminal: clear
  - Check what commands you have learnt so far: history

less

- Function: Look around
- Usage: less really_long_file
  - The navigation is the same as man page

- Exercises:
  - Look around the alice.txt file
  - Search for the word “good” from the beginning
  - Search for the word “good” from the end

less

- Function: Look around
- Usage: less really_long_file
  - The navigation is the same as man page

- Exercises:
  - Look around the alice.txt file: less alice.txt
  - Search for the word “good” from the beginning: /good
  - Search for the word “good” from the end: ?good

The end

- If there is only one thing you have learnt today
  - I hope it is about how to use man page!