Final Notes

Table of Contents

Lecture 2

Raspberry Pi:

- The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins, allowing you to control electronic components for physical computing and explore the Internet of Things (IoT).
- It is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse.
- Capable of doing everything you'd expect a desktop computer to do.
- 4 versions of Raspberry Pi available.
- Raspberry Pi 400 is the latest version that comes with a dedicated keyboard. picOfRaspPi

Virtualization:

- Using multiple operating systems at the same time in a same computer.
- General types of virtualizations: **Server-based** and **Client-based**.
- Reduces costs by decreasing the buying costs of physical hardware.

Lecture 3

Important Commands cheatSheet cheatSheet cheatSheet cheatSheet **CD:** Current directory

- Cd + destination
- Single dot (.): the current working directory, double dot (..): goes one step back;
- Cd../home; cd../.. (goes two steps back)
- Cd ~ (easiest way)

cheatSheet

DE: Desktop Environment

<u>The Bash Shell</u>

- Most Linux distribution use Bash Shell.

- The GNU bash shell is a program that provides interactive access to the Linux system.
- It runs as a regular program and is normally started whenever a user logs in into a terminal.

Bash Shortcut

- Copy and paste: Ctrl+Shift+c and Ctrl+shift+v
- !! – run last command
- !iptables – run the most recent command that starts that include iptables.

Shell Prompt

- launch a terminal: ubuntu@nova:-$

DPKG- A software package installation and management tool. It is used by the Debian Linux operating system, and other Linux systems that rely on the extensive Debian package repository.

APT- Advanced package tool

Ubuntu PPAs (Personal Package Achieves)

> **LINUX Dictionary Structures** -File system: Linux organizes its files in what is called a Hierarchical director system (tree-like pattern or folders) -Filesystem Hierarchy Standard (FHS) specifies requirement and guidelines -Relative path: Specifies the immediate path or the child path. ex: **usr/math** Types of path names: -Absolute path: States the full pathname starting from the root (/). Always starts from the root **/home/usr/math**

Lecture 4

The Touch Command

- The touch command for creating files. cheatSheet

INODE (index files)

- Access: (read, write, execute permissions), ownership, file type, file size, group, number of links, etc. Each inode is identified by an integer number.
- A data structure that contains all the information about a file except the file name and its content.
- Every file in the file system has an inode + created file location: ls -i ~/filename.

Hard Links:

- To create a hard link: ln file ~/Downloads/FileHL
- This link needs to be the same file system to create links between them.

Soft Links:

- Connects to the hard link.
- To create a symbolic link: ln -s fileName fileNameSL
- these links can lead to files that are stored in different locations.

Important Commands cheatSheet cheatSheet cheatSheet cheatSheet

Man command cheatSheet cheatSheet

Lecture 5

Wildcards

- Wildcard represents letters and characters used to specify a filename for searches.
- They are called meta character wildcards.

The * wildcard:

- Searches for any matches whether string or number. Ex: **ls .dpkg* will find all the files.

The '?' wildcard meta character matches precisely one character. To match all files that have a vowel after letter f:

- ls f[aeiou]* To match all files that do not have a vowel after letter f:
- ls f[!aeiou]* To match all files that have a range of letters after f:
- ls fla-z]* To match all files whose name has at least one number:
- ls *[0-9]* To match all the files whose name does not have a number in their file name:
- ls *[!0-9].* To match all files whose name begins with a letter from a-p or start with letters s or c:
- ls [a-psc]* To match all files whose name begins with any of these two sets of characters: letters from af or p-z:
- ls [a-fp-z]* To match all files whose name begins with any 3 combination of numbers and the current user's username:
- ls [0-9] [0-9] [0-9] $USER

The [] character

- The brackets wildcard match a single character in a range.

Handling Text File cheatSheet

grep

- The Grep command is used to match a string pattern from a file.
- Example: **grep + option + Pattern to match + file**

cat

- The Cat command is used for displaying the content of a file. Example: **cat todo.md**

tac

- Tac displays the files from tail to head (reverse order)

more

- Ex: **more + file to view**
- Display first 10 lines of a file: head /etc/passwd
- Display first 5 lines of a file: head -5 /etc/passwd

- To Display last 5 lines: tail -5 /etc/passwd

cut

- Allows you to extract files from a specific field
- Display the last 5 users: tail -5 /etc/passwd | cut -d ':' -f 1

sort

- Sort -o filename.txt oldname.txt (to sort the file and save with a new name)
- Sort with numeric data: sort -n filename.txt
- Check if a file is sorted: sort -c filename
- Sort in reverse order: sort -r filename
- Sort by column number: sort -k 2 fileName
- Remove duplicate user: sort -u filename

1. Match all lines that start with uppercase letters **grep "^[[:upper:]]" /etc/passwd**
2. Match all lines that end with a digit **grep "[[:digit:]]$ data.csv**
3. Match only lines containing IPv4 addresses **grep -E '[[:digit:]]{1,3}.[[:digit:]]{1,3}.[[:digit:]]{1,3}.[[:digit:]]{1,3}' ipaddresses**
4. Match one word or the other. **grep -E 'hello|hi' file.txt** Grep can search for pattern sequences using {n}
5. Search all lines that contain a character repeated 3 times **grep -E "A{3}" file.txt**
6. Search all lines that contain a phone number of the format 973-111-2222 **grep "[[:digit:]]{3}[-][[:digit:]]{3}[-][[:digit:]]{4}" file**

The period character (.) is used to represent any single character. For example, search for all lines that contain any word ending in "able" and has 3 characters before "able". grep "...able" logbackup.log

> man ls | grep "comma separated" output: -m fill width with a comma separated list of entries

Lecture 6

- To create an archive: tar -cf example.tar file1 file2 file3
- -f is always required
- *-v display the details. Not required
- To list the files in archive: tar tf filename.tar
- To add new file to the archive: tar rf filename.tar newfile.txt
- To update an existed file after edit: tar uf filename.tar newfile.txt
- To delete: tar --delete -f filename.tar filename.txt
- To move files to a new directory: tar -xf filename.tar -C newDir/
- To display all files including number in name for image and videos: tar cf allfiles.tar *.txt *[0-9].*{jpg,png,svg} Video.mp4
- To generate a text file and save the output: lorem > filename.txt cheatSheet cheatSheet

Symbolic mode

- u = user/owner, g = group, o = other
- r = read, w = write, x = execute

- To read, write, and execute: chmod u=rwx,g=rw,o=r
- Example: chmod u=rwe,g=rw,u=r filename.txt (user= rwe, group= rw, other = r)

Numeric mode

- Read = 4
- Write = 2
- Execute = 1
- RWe = 7, rw = 6, rx = 5
- Example: chmod 765 filename.txt (user= rwe, group= rw, other = r)

Gzip

- Gzip filename.txt
- Compress multiple files: gzip file1.txt file2.txt file3.txt
- Compress and keep the original: gzip -k file.txt
- Decompress: gzip -d filename.txt
- Decompress a file from another directory to the present working directory: gzip -dkc < ../otherDirectoryName/filename.txt.gz > presentFile.txt

File Permission

- To execute a file: chmod u+x filename.sh
- To run the script: ./scriptname.sh

Ar

- To create an archive: ar r archive.a *
- List of the files: ar t archive.a
- To add new file: ar r archive.a meme.txt
- To delete a file: ar d archive.a meme.txt

> Extras **sudo apt install zip** To zip one or more files, specify the files you want to add to the archive separated by space, as shown below:**zip archivename.zip filename1 filename2 filename3** Create a zip archive of a directory including the content of subdirectories. The -r option allows you to traverse the whole directory structure recursively: **zip -r archivename.zip directory_name** Add multiple files and directories in the same archive: **zip -r archivename.zip directory_name1 directory_name2 file1 file1**

Lecture 7

Vim :w to save file **:w** fruit.txt to save file as new.txt **:wq** to save file and quit **:wqa!** to save file and close all
 cheatSheet

Insert mode: used for writing text

- Normal mode: used for manipulating text
- Command mode: used for entering vim commands
- Visual mode: used for navigation and manipulation of text selections
- Select mode: similar to visual mode

- Ex-mode: Similar to the command-line mode but optimized for batch processing.

> When you start vim, you are in normal mode From normal mode press i to enter insert mode. The word -INSERT- will appear on the bottom left corner of the terminal indicating that you are in insert mode. To switch back to normal mode press esc. In the lack of the esc key press ctrl + c

Create user:

- /etc/login.defs
- /etc/default/useradd
- /etc/skel/
- /etc/passwd
- /etc/shadow
- /etc/group
- /etc/login.defs file: grep -ve ^$ /etc/login.defs | grep -v ^#

~ To see students user info: grep student /etc/passwd ~ To view info about user's acc n passwd: getent passwd student, sudo getent shadow student ~ To update passwd for current user: sudo passwd ~ To update psswd for another user: sudo passwd + username ~ To lock and unlock account: sudo passwd -l or pass -u ~ Create a home directory for user: sudo usermod -md /home/nova ~ Loging with new user : su username ~ Logit: exit ~ Change the default shell: sudo usermod -s /bin/bash username

Managing Group: • Cat /etc/passwd | grep "Nova" • Cat /etc/passwd | ^ "Nova" • Sudo groups Nova

Lecture 8

- To display a line of text use the echo command: Ex: **echo "This is a message"**
- echo "This is another message" "and another message"
- echo -n "this is , again another message"

> Some environment variables are the same for all users logged in to a machine, such as the HOST environment variable that specifies the computer name. The env command allows you to see all environment variables You can use the echo command to see the value of an environment variable.

Looping: is used to perform a set of commands repeatedly.

-d [Checks whether the file exists.] -F [Checks whether the file is a directory.] -r [Checks whether the file is a regular file.] -X [Checks whether the user has read permission for the file.] -O [Checks whether the file contains data.] -e [Checks whether the user has write permission for the file.] -G [Checks whether the user belongs to the group owner of the file.]

file1 -nt file2 [Checks whether file1 is newer than file2.] file1 -ot 11le2 [Checks whether 11le1 is older than f1le2.]

Review

Command: (mkdir, I/O redirection, grep, head/tail, relative path, absolute path, chmod, ip address, vim

#mkdir: make directories. mkdir food. (current directory) Create mkdir in a different directory (/usr/share): mkdir /usr/share/food. Absolute path is the (/usr/share) mkdir -p class/{math,eng,bio} (create a directory structure) #brace expansion!

Input and Output redirection:

> ip add > ipadd.txt (save something to a txt file) Redirecting a data and overwrite (ls Downloads > ipadd.txt Executes and append to the data of the file (ls Downloads >> ipadd.txt Redirects ls doc 2> error.log. ls doc ~/ & >> all.txt

Tar command:

- Tar + options + file.tar + files to archive
- Tar cf to create
- tar cf allmymusic.tar ~/music ~/Downloads /newmusic
- xf for the archived extract

Grep:

- grep works line by line in a file.
- grep + option + “string to look for” + file to work with
- grep “08075” address.cvs
- Modify the behavior grep -v “08075” address.cvs (the txt file will show every other results than 08075)
- grep -i “jack” address.cvs (allows any case sensitive word)
- Piping into grep (ls ~/ | grep -i “Downloads”)

head/tail:

- Head- displays the first 10 lines of a text file. (read -3 all.txt)
- Tail - displays the last 10 lines of a text file. (append the 3rd line of)

Absolute Path:

- /usr/share/theme (complete+ specific path)

Relative Path:

- share/theme

Chmod:

- Chmod u+x script.sh

Vim:

- Do the vim lab

mkdir -p spring21/{ma108,com101,cis295}/{notes,hws,extra}

- Download wget (the url of the file)
- Extract the archive: tar xf