Lecture 1

Why Learn Linux? Linux is everywhere

Linux is the core of IOT

- (Microsoft uses linux for cloud services)

What is linux?

- Linux is a kernel (A Kernel is a core of an OS)

Linux has variety of Distros (Versions)

Independent Distros

- Slackware

- Arch Linux

s

Well known

Debian and Redhat (Kali Linux)

Linux is FREE!

License of freedom (GNU)

Lecture 2

09/08/21

The concept of cloud computing consists of physical computers that store data as backups in case a device connected breaks down or files corrupt.

Github is the google document for programmers

Github is front end

Git tracks changes to local files

Markdown

Mark down is markup language that’s let you write plain text documents with a few lightweight formatting options

Created by John Gruber in 2004

Markdown is content focus - the user does not need to worry about tags like in HTML.

Markdown is future proof since it works with plain text instead of proprietary document formats.

Multiple implementations of markdown exists, these are examples

• CommonMark

• Github Flavored - This is what we are using

• MultiMarkdown

What you can do with Markdown?

Add formatting to plain text

Convert plain text document to HTML PDF DOCX and PPX

Used by other technologies like:

Github a Gitlab

HUGO and Jekyll static site generators

Notes Taking apps, social media pages and communication platforms like slack

Write industry standard documentation

Headings are used to guide readers through a document. In general, heading indicate what the following paragraph or section is about.

To format any text

Github and Git

Git - Git is an open-source, version control tool created in 2005 by developers

Linux kernel

> Libraries

> Apps

>GNU Software

>Drivers

>Compiler

/root beginning

Nothing before it goes after

Lecture 3

In Linux you will hear directory many times. Directory is also known as "folder"

Navigating the FS in the CLI

The Linux Directory Structure

- Think of the file system as a tree where every branch represents a directory (folder)
- You are always working insde a particular directory and you can move forward to a subdirectory
- The directory where you are at the moment is called the current working directory or present working directory
- In a filesystem, every file has a pathname which indicates the location of the file in the filesystem (like an address).

The file structure always starts at the root '/'. It follows the sub-folders to create a pathname until it reaches the file in search of.

#ALWAYS USE ABSOLUTE PATH The path might be long

Relative path

A few commands *** The pwd command-** used for displaying the current working directory

- The cd command - used for changing directory What it does? The working directory is represent by a Single dot or 2 dots.

commands cd takes you home cd ~ takes you home as well (~ = home$ User) cd $HOME

~ short hand for home directory Home directory /home/student | $HOME/student | Home directory

- The ls command - used for displaying all the files inside a given directory. Whne no directory is specified, ls displays the files in the current working directory.

  ## Types of pathnames:

*** Absolute path - States the full pathname starting from root (/). Always starts from the root**

- Example: /home/juniornunez920/Downloads/egypt.mp3

*** Relative Path - Specifies the pathname starting from the current directory. Always starts with a subdirectory.**

- Example: Downloads/egypt.mp3

Sources https://github.com/ra559/cis106

Lecture 4

Hard Links

- Hard Links are files that point to data on the hard drive

- When you create a file by default its linked to the data stored in the hard drive and its assigned an inode number.

- To create a Hard link: " ln file ~Downloads/fileHL "

Soft Links

- Symbolic links (soft links) are a special type of file that point to other files instead of data in the hard drive

- Soft links do not share the same inode number as a hard link does

- If you modify a soft link the target file is modified as well

- The advantage of soft links is that they can point to files stored on different partitions

- To create a symbolic link: " ln -s file fileSL "

    ## Linux blackhole

NEVER RUN

"sudo rm-r /home/#USER 2>/dev/NULL"

Note to self:

- Never delete hard links

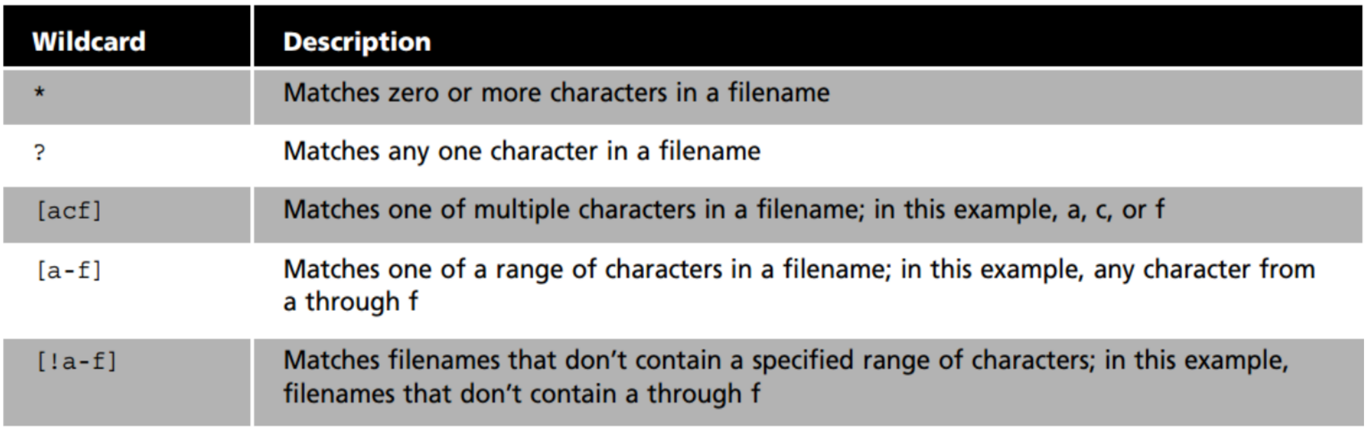
Getting Help

WILL PLACE HARD LINKS IN MIDTERM

- man ls
- man passwd
- man 5 passwd
- man -k passwords

Using Wildcards / File Globbing

The main wildcared is a star or asterik (*) character *Searches for anything that macthes any number or characters. For example, ls *.txt will match all files that end in .txt regardless of the size of the file



Wildcard	Description
*	Matches zero or more characters in a filename
?	Matches any one character in a filename
[acf]	Matches one of multiple characters in a filename; in this example, a, c, or f
[a-f]	Matches one of a range of characters in a filename; in this example, any character from a through f
[!a-f]	Matches filenames that don't contain a specified range of characters; in this example, filenames that don't contain a through f

b -1 ls -i w/file

- Expects to use wildcard to move file TO FIND .png in the end ' mv ~/Downloads/*.png /var/www/html/html/Assets/image/ '

To find anything in the middle and everything ' ls B?nana* '

Continuation on 10/27/21

vacation |-Italy/Nature/Night |-DR/Nature/Day |-Dubai/Nature/Day |-Tokyo/Nature/Night

juniornunez920@juniornunez920-VirtualBox:~$ 'cd' juniornunez920@juniornunez920-VirtualBox:~$ mkdir vac21/{italy, dr, dubai, tokyo}/nature juniornunez920@juniornunez920-VirtualBox:~$ tree vac21/

Handling Text Files

- cat todo.md

more -d /var/log/syslog

motr -10 var/log/syslog

head + option +

johnthereaper

log files on linux - find online.

Cut commmand

cut + option + file

cut -f1 hostnames.txt

To Display all addresses with IP

'ip add | grep inet'

'&>' Redirects everyhting

'2>' Redirects error

'>' Standard

man ls | grep "comma"

man ls | grep "comma separated"

What is VIM

- Vim is not included in Ubuntu
- To install vim
    - Sudo apt install vim

  Vim has insert modes -i

• if vi is installed in all Linux distros, Why am i learning VIM? vim has more features vim is easier to learn vim is also very light weight most vim commands are backwards compatible with vim

How to start and quit Vim?

• To start vim type vim. The text editor will start in normal mode. • To quit vim press esc and type :ga! g a -D prefix for entering command line mode -5 short for quit short for all buffers -. force quit all now • To set line numbers: set number

Vim modes:

• 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 +

Insert text:

You can create a file and open vim at the same time by typing vim and a file name. Example: vim notes. txt In insert mode, you can use: The arrow keys to move around, Enter Key to continue in the next line, Backspace for deleting

Install vim

sudo apt install vim -Y Start vim 2 vim Enter insert mode Press letter i Enter normal mode 2 Press esc key Quit vim Type :g! Start vim vim Enter insert mode Press letter i Type 3 sentences I like linux. Pizza is great. I go to school. Exit vim without saving Press esc key Type :g!

Saving and quitting vim

Terminsl Helo To save a text file you need to enter normal mode using : and the use the w key File Edit View an a tilel Terminal Tabr I will save the file w new. txt will save the file as new.txt + wa will save the file and quit + wqa will save the file and close all files open in the buffer

#Managing US

Linux File permissions | File Ownership

A file can be owned only one user

Commands

- The chmod command

- Symbolic Notation

Managing user accounts involves adding, modifying and deleting user accounts and account's information To add user accounts we use the useradd or adduser command.

In Ubuntu, the adduser program is recommended over useradd due to useradd being a low-level utility. To modify user's information we use the ugermod program. To delete a users we use the userdel program.The following files are involved in the user creation process:

/etc/login.defs

/etc/default/useradd

/etc/skel/

/etc/passwd

/etc/shadow

/etc/group

How to add user?

adduser followed by username

To delete

userdel -r username

The /etc/login.defs file

It contains directives for use in various shadow password suite commands.Shadow password suite is an umbrella term for commands dealing with account credentials, such as the useradd, userdel, and passwd. The directives (entries or configuration variables) in this configuration file control an array of settings from password length all the way to whether or not a home directory is created when a user is created. The file is typically filled with comments and commented-¡ut directives (which make the directives inactive).Here is an example of how the /etc/login.defs file looks like:

https://robertalberto.com/cis106/login.defs-file-example.txt

Here is how the /etc/login.defs file looks like in your system without any comments:

grep -ve ^$ /etc/login.defs | grep -V A#

PASS_MIN_LENGTH Minimum number of characters required in password.

PASS_WARN_AGE Number of days a warning is issued to the user prior to a

password expiration.

CREATE_HOME Default is no. If set to yes, a user account home directory is

created.

ENCRYPT_METHOD The method used to hash account passwords.

This file stores the system default configuration for creating new users with the, useradd utility.

To view the default parameters in the /etc/default/useradd file use either of these

commands:

useradd

-D

cat /etc/default/useradd

If the /etc/login.defs has the directive CREATE_HOME not set or set to no, then the

users created with the useradd utility will not get a home directory unless the -m

option is given.

You can view all the directives as well as what each directive mean in by

examining the content of the /etc/default/useradd file.

cat /etc/default/useradd grep -ve ^$

/etc/default/useradd | grep -VAt

HOME Base directory for user account directories.

INACTIVE Number of days after a password has expired and has not been changed until the account will be deactivated.

SHELL User account default shell program. In Ubuntu, this variable is set to /bin/sh which means that after using the useradd command, unless bash is specified, sh will be the default login shell for any user. The /etc/skel/directory The /etc/skel directory stores files that are copied to each user's homedirectory

Lecture 8

12/01/21

Shell Scripting (Intro)

Before you start shell scripting it is important that you understand the following concepts:

Source code vs machine code:

Source code is the human readable code written in a programming language like Python. machine code consists of binary 1s and Os and is the language a computer's CPU understands.

Compiler: a program used for converting a complete source code into machine code.Interpreter: a program that converts source code into machine code line by line. Compiled program vs Interpreted program:a compiled program is a binary file produced by a compiler. an interpreted program is a program that requires an interpreter to interpret and run each Shell scripts are examples of interpreted programs, and the interpreter used is the BASH shell. Creating a shell script is as simple as creating a file and then assigning execute permission for it. After creating a shell script and assigning execute permission, you must enter the absolute or relative path to where it's stored to run it.

To Dipslay a line of text use the echo command: Example

echo "This is a message"

echo "This is another message" "and another message"

echo -n "this is , again another message"

Working with variables

Variable: placeholder for data.

Environment variable; is a placeholder for data that can change; typically, it gets its value automatically from the OS startup or the shell being used. Each user has environment variables with different values to define his or her working environment. The HOME environment variable stores the absolute pathname to a user's home directory, so it varies for each user. 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. 1 Example:

- echo SHOME
- echo $HOST

Practice: Type vim script2.sh, enable line numbers, and enter insert mode.

Type:

*** #!/bin/bash computer="office-computer" echo "What is your name:" read name echo "Hello, $name you are in $computer" Save your file and close vim Make the file executable with: chmod u+ script2.sh Run the script with: ./script2.sh**

Lecture 8

12/01/21

Shell Scripting (Continuation)

Looping

Looing is used to perform a set of commands repeatdly. In the menu script, the user is given a list of options.

Options:

while loop -

until loop - repeats commands between do and done as long as the testted condition is fals (exit satus code is greater than 0) in other words, until condition is met with the equivalent of true.

for loop - repeats commands between do and done a specified number of times. Each time the script carries out the commands in the loop, a new is value is given to variable.

https://kapeli.com/cheat_sheets/Bash_Test_Operators.docset/Contents/Resources/Documents/index (For reference on operators)