**#### Module 1 : Essentials Commands in Linux**

**#### Module 2 : User and Group Managements and Permissions**

**====================================================================**

**# Module 1 : Essentials Commands**

0. What is Operating System

1. Installing Linux

2. Using Essential Tools

3. File Management Tools

4. Working with Text Files

5. Advance Text File Processing

6. Connecting to a Server(AWS class)

7. Working with the Bash Shell(AWS Class)

**#### Module 2 : User and Group Managements and Permissions**

**==============================**

**==============================**

1. Add Users

2. Add Group

3. Add users to group

4. Change user password

5. Understand user and user password files

6. Understand user permission

**==============================**

**# INSTALLING LINUX**

**=============================**

**================================**

**# 1.1 Understanding the Linux Operating System**

Let's talk about Linux. Before we go deep about linux, let's take a step back and talk about computer as a whole.

Computer is made up of multiple parts.

\* The Hardware

\* The Software

The Hardware comprise of the

- The CPU -

\* Control Unit

\* Arithmetic Logic Unit

\* The Cache

\* The Buses

\* Registers

\* Storage Units

- RAM

- Hard Drive

- Solid-State Drive(SSD) \*

- Network Interface card

- ETC

The Software comprise

- The Operating System(OS)

- Device Drivers

- Utilities

- Application or User Downloaded Softwares such as Microsoft WOrd,Spreadsheet, Graphics applications

- Etc

The part of the computer that is available to the end-user is the Applications, or users downloaded software such as word, spread sheet, google chrome. The Application needs the physical component of the computer to perform whatever task you ask them to do, But they can not communicate directly with this physical component, so then need interpreter or thing that speaks both the language of the user's application and the computer hardware, and that is where the OS comes in.

The OS manages all of the software and hardware on the computer. Most of the time, there are multiple or different computer programs running at the same time and they all need to access your computer CPU, Memory and storage. The OS coordinates all of this to make sure each program gets what it needs.

Type of OS

============

There are multiple OS out there.The Window, Mac OS and Linux are the most popular.

For this course, we are going to be looking at the Linux Operating System.

**# 1.2 Understanding the Linux Distributions**

There are multiple type(distribution) of Linux OS.

\* The Debian - Ubuntu and Linux Mint, Kali-Linux(Pentesting)

\* Red Hat Enterprise ( RHEL) - CentOS, Fedora

\* SUSE - Opensuse

**# 1.3 Getting Red Hat Linux Family**

**# A. Download Red Hat Enterprise Linux**

Step 1: Go to developers.redhat.com

Step 2: Click on Login to login into your existing account or Register for a Red Hat Account

Step 3: After login, go to Products link, Click on Red Hat Enterprise Linux and download the software

**# B. Download and Installing CentOS**

Step 1: Go to Vagrantup.com

Step 2: Download vagrant Amd64 for windows if your are using window system otherwise download for your system OS

Step 3: Go to your project working directory, and run vagrant init

Step 4: open the Vagrantfile that was downloading in that working directory, under config.os change "base" to centos/7 and save the file

Step 5: Run vagrant up

Step 6: Run Vagrant ssh

Step 7: At this point you are now inside the centos

**# 1.4 Using Windows Subsystem for Linux (Ubuntu)**

This is linux integrated into window Operating system. To use Linux on window, follow the step below to install it.

Step 1: Enable HyperV on your window system. Search for Hyper-V from the search bar and click on the "turn on Windows features off or on" Then check Hyper-V or Virtual Machine Platform(window 11).

Step 2: Restart your machine

Step 3: Go to microsoft store and download ubuntu 20.04 LTS

**========================================================================================**

**# 2. Using Essential Tools**

2.1 Logging in to Linux

2.2 Using su and sudo

2.3 Starting on the command Line

2.4 Using the Essentials Linux Command Line Tools

2.5 Getting Help with man

**# 2.1 Log in to Linux**

\*\* To log in, you need a user account

\*\* Most Linux distributions create an administrative/root account while installing

\*\* You can login from a console, a graphical user interface like window or remotely using ssh

\*\* If you login via the GUI, you have limited functionality. So the best way to login on linux is via the command line.

\*\* If you log in via the GUI, start the terminal and you still have full access to the shell or the full functionalities.

\*\* Do not waste your time, learning the GUI!

**# 2.2 Using su and sudo for administrative privilege**

Administrative users or the root users have unrestricted access to the system. So avoid using the root user for your day to day task.

\*\* To have administrative privilege, you have to a member of Administrative group which is WHEEL(RedHat) and sudo(UBUNTU)

\*\* You can change from normal user to root user using su -

\*\* sudo allows authorized users to run tasks with escalated privileges e.g sudo command ==> "sudo ls /root"

\*\* To allow sudo usage, users must be a member of the sudo group.

==> Wheel on Red Hat

==> sudo on Ubuntu

**# 2.3 Starting on the command Line**

Before we start working deeply in linux, let keep the following in mind;

a. Linux is case sensitive e.g LS is invalid, ls is valid

b. Commands are often used with options

\* Long options start with --; e.g ls --help valid, ls -help is invalid

\* Short options start with - and can be combined: ls -lrt

c. User --help on any command to learn how to use the command

**# 2.4 Using the Essential Linux Command Line Tools**

\*\* whoami - who the login user is

\*\* pwd - current working directory

\*\* mkdir - Make directory

\*\* cd - change directory

\*\* touch - create file

\*\* echo

\*\* ls - list folder or file

\*\* cat ==> Print out/Read the content of a file

\*\* ip a ===> here your ip address is the 2 one for centos or last one for ubuntu,

\*\* passwd == > Allow you to change your password

\*\* up arrow, down arrow, right arrow and left arrow

**# 2.5 Getting Help with man**

\*\* man is the Systems Programmers Manual. This is the manual of Linux Operating System.

\*\* It is the main (but not the only) source of getting information about linux command usage

\*\* Use man to get information about commands, configuration files and more

**=========================================================**

**# 3. Essential File Management Tools**

3.1 Essential File Management

3.2 Understanding the Linux File System Hierarchy

3.3 Listing Files with ls

3.4 Using Wildcards

3.5 Copying files with cp

3.6 Working with Directories

3.7 Using Absolute and Relative Paths

3.8 Moving Files with mv

3.9 Removing Files with rm

**# 3.1 Essential File Management**

\*\* On linux everything is file.

\*\* Configuration is stored in text files.

\*\* Devices are addressed by using device files

\*\*So in order to define where you file is, there is linux file hierarchy standard.

\*\* Unlike windows, Directories are highly standardized on linux

\*\* To read more about File System visit "man hier" or https://refspecs.linuxfoundation.org/FHS\_3.0/fhs/index.html

**# 3.2 Understanding the Linux File System Hierarchy**

\*\* Regular users only have write-access to two directories only:

\* /home

\* /tmp

Other commonly relevant directories

\* /

\* /bin

\* /sbin

\* /etc => this is one of the most important directories and it contain text based configuration files e.g /etc/passwd - These is the user database where all information about the users are stored.

\* /dev => these is where you find device that you attached to the machine such as sda volume, sr0 (cd rom)

\* /mnt what you used to connect devices to the machine

\* /var use to store logs

\* /sys ==> managing hardware

**# 3.3 Listing Files with ls**

\* User pwd to find the name of the current directory

\* ls is used to list files and directories

\* ls -a will show hidden files also

\* ls -l provides long listing (many details)

\* Use ls -ld /directory to see properties and not the content of a directory. This is essential if you want to set permission on a directory

\* ls -lrt shows a time-sorted list of files

**# 3.4 Using Wildcards**

Most command that are use for file management support wildcards

\*\* ls a\* e.g cd /etc; ls b\*

Many commands also support working with groups and ranges

\*\* touch file{1..100} e. g touch file{1..4} ==> file1 file2 file3 file4

**# 3.5 Copying files with cp**

\* cp /dir/somefile /somedir to copy a file

\* cp -a to include hidden file in the copy

\* cp -R will copy recursively: including subdirectories

**# 3.6 Working with Directories**

\* cp copy directory

\* mkdir ==> create a new directory

\* mkdir -p to make parent directory if not exist then create a child directory

**# 3.7 Using Absolute and Relative Paths**

\* An Absolute path contains the full name from the root directory to a file

e.g /var/log/messages

\* A relative path is related to the current directory and contains the rest that is needed to get to a file e.g etc/ssh/sshd

\* In relative paths, .. can be used for "one directory up"

**# 3.8 Moving Files with mv**

\* mv to move a data from one location to the other location

**# 3.9 Removing Files and directories with rm and rmdir**

\* rm to rm a file

\* rmdir to remove an empty directory

\* rm -i will prompt for confirmation on all files

\* rm -f will force and don't ask anything

\* rm -rf removes an entire directory tree without asking any confirmation

\* rm -rf /--no-preserve-root will remove absolutely everything

**======================================================================================================**

**# 4. Advanced File Management Tools**

4.1 finding Files with find

4.2 Archiving Files with tar

4.3 Managing File Compression

4.4 Mounting File Systems

**# 4.1 Finding Files with find**

Find is a powerful and thorough utility that helps you to find any file no matter where it is stored.

**## Examples**

\* find / -name "hosts" [\*\*\* Note : when you run this, you will see a lot of permission denied error, just append 2> /dev/null to send all the error messages to /dev/null]

\* find / -name "hosts\*"

\* find / -user linda

\* find / -size +2G

\* find / -user vagrant -exec cp {} /root/linda \; ==> find linda and cp {} to /root/linda then escape(\; mean escape)

\* find / -perm /4000 ==> find users with full permission

**# 4.2 Archiving Files with tar**

tar is the Tape ARchiver and basically use is to compress, extract or list

\* Use to compress, extract or list directories

**# 4.3 Mounting File System**

\* lsblk -- lists block devices

\* mount -- lists all current mounts(administrative mounts included)

\* umount -- To remove the mounted or attached device. Do not remove without using umount because you risk loosing all your data.

\* df -h (disk free human readable format)-- Presents mounts devices, including available disk space

5. Working with Text Files

\* 5.1 Working with text editor

\* 5.2 Working with less and more

\* 5.3 Using cat, head and tail to see file start and end

\* 5.4 Working with grep

**# 5.1 Working with text editor**

\* Working in Linux mostly happens from the command line, so you need a command line text editor

\* nano is easy to use and available on many linux installations by default

\* vim is vi improved, a very powerful editor with advanced programming features

\* VIM

\* On some systems you'll only find vi, so you need to learn vi

\* it is always available, even on very old UNIX systems

\* It is powerful and programmable

\* After starting vim, it opens in command mode. You can't type in this mode.

\* use i, insert or o to get into insert mode

\* use ESC to get back to command mode

\*\* :wq! write, quite and don't complain to save and quit the vi

\*\* u to undo the last changes

\*\* dd deletes the current line

use vimtutor to learn more

**# 5.2 Working with less and more**

\* cat dump all the file content on the screen

\* more show content page by page but can not go up or down with arrow keys

\* less show content page by page and can go up or down with arrow keys

**# 5.3 Using cat, head and tail to see file start and end**

\* Tails print the last 10 lines of the contents in the file

\* prints the first 10 lines of the contents in the file

\* You can modified the numbers of line to show for both the tail or the head by specifying the numbers of line of content to show with -ve

**# 5.4 Working with grep**

\* grep is one of the most important command in the linux. You can use it with pipe to pass the output of a command as input of the grep

\* it is you to find text

\* grep vagrant\* /home

**# 6. Connecting to a Server**

\* 6.1 Understanding the root user

\* 6.2 Creating a simple sudo configuration

\* 6.3 Using ssh to connect to a Remote server

\* 6.4 Working with linux from window

**# 6.1 Understanding the root user, sudo and su**

\* The root user exits in Linux kernel space. It has all the privilege and it can do everything on the system.

\* Root user needs no permission to perform any task. It has kernel level access.

\* Avoid using root user for your day to day task

\* If you need root user access, use sudo

\* Instead of log in as a root user, use sudo or alternatively, use su to open root shell after logging in as an ordinary user.

\*\* SU

\* su is switch user, it allows you to open a shell as a specific user

\* To open a root shell as ordinary user use su -, you need a password of the target user to run this.

\* on ubuntu su will not work so you have to use sudo

\*sudo -i

**# 6.2 Creating a simple sudo configuration**

\* To modified the sudo configuration,

\*\* As root go to "vi /etc/sudoers"

\*\* As ordinary user, use "sudo vi /etc/sudoers"

\* on Ubuntu, the sudoers group is called sudo

\* on CentOS, the sudoers group is called wheel

**# 6.3 Using ssh to connect to a Remote server**

\* ssh is Secure Shell

\* Is allows you to connect to a remote server through shell client

\* In order to use secure shell, you need to installed and enabled it.

For Ubuntu==> sudo apt install openssh-server

For RedHat(CentOs) ==> sudo dnf install openssh-server; sudo systemctl enable --now sshd

\* You also need secure shell client such as MobaXterm or Putty or Vscode, to access the ssh server( /etc/ssh/openssh-server)

\* Use scp to securely copy file from your local system to remote computer ==> scp /etc/hosts 172.17.163.42:/tmp/

**#### Module 2 : User and Group Managements and Permissions**

\* There are three type of user, root user, system user and normal users

\* Whenever you create a user, a uid(user id), gid(group id) and groups( group that the user belongs) are all created automatically

\* Root user have user id 0 and a sign of #

\* Normal user start with 1001 and a sign of $

\* root@localhost ~ # ==>

root ==> username

@ ==> sign

localhost ==> machine name

~ ==> current working directory

**# ==> root user's sign , $ ==> normal user's sign**

\* To get the id of a user, run "id <username>"

\* To add a user, run useradd <username>

\* User profile is create in /home . On Ubuntu, /home/profile is not create for new user. You have to run "useradd -m username" to add a user with home profile created.

\* Every Linux user is a member of at least one group

\* Most linux distribution uses "private groups" meaning that while creating the user, a group with the name of the user is created and the user is the only member of that group.

\* A User can be member of multiple groups such as admin, sudo group etc

\* Users database is located in /etc/passwd

\* To get more detail about a user, /etc/passwd

\* When a user create a profile on linux, that user becomes the file owner.

**### USERNAME DEFINED**

*\*\**\*\* kevick:x:1001:1001:empty:/home/kevick:/bin/sh

1. Kevick ==> username

2. x ==> user's password

3. 1001 ==> user's id

4. 1001 ==> user's group id

5. empty==> ::==> GECOS field or Space for comment

6. /home/kevick ==> user's home directory

7: /bin/sh ==> user's shell

**#### PASSWORD DEFINED**

\* To view/modified user related password, go to /etc/shadow

user password information ==> user-name:!:19217:0:99999:7:::

1. user-name ==> username

2. ! or !! ==> password not set

\* To set password, sign in as root user, the run "passwd user-name" then type in the user password

\* To modified the password parameter, run "vi /etc/shadow"

username:$6$y5iC0Zkr51FU4iOc$Ew3W.TcLGOFprEH.yh.94xw/vgeIr6gPWxJPusZECjFWR5foY8UQAXhlX6Jg9jvytjcU3CACekx69IU4Sir040:19217:0:99999:7:::

1. username

2. encrypted password

3. Date of last password changed, express as number of days since 1/1/1970

4. Minimum password age, zero if password was just created

5. Maximum password age. That is the number of days before a password expired.

6. Password warning period - Number of days before a password going to expired

7. Password inactive period

8. Account expiration date - date that user will not be able to login

9. Reserved field for future use.

**## GROUP DEFINE**

\* To verify the name or Id of a group run "id username"

\* To create a group, run "groupadd <groupname>"

\* To add a member to a group run " usermod" -aG <groupname> <username>

**#### Module 3: Operating Running Systems**