Operating System: An OS is system software that manages computer hardware and software resources, and provides common services for computer programs.

Types of OS :-

1. Desktop OS – Microsoft windows, macOS, linux such as ubuntu
2. Server OD – windows server, linux distributions like centOS, RHEL
3. Mobile OS – Android, iOS, Windows mobile
4. Embedded OS – Routers, smart TVs, automobiles, home appliances
5. Real-time OS(RTOS) – medical equipment, car ECUs, aerospace, defence, network firewalls, home security systems

Linux is an open source OS that manages computer hardware and software resources.

Developed by linus Torvalds

Linux Distributions

1. Ubuntu
2. Fedora
3. Debian
4. RHEL
5. CentOS
6. Arch Linux
7. openSUSE
8. linux mint
9. gentoo linux
10. slackware
11. alpine linux
12. kali linux

Install Oracle Virtualbox to use linux systems - <https://www.virtualbox.org/>

Filesystem – used by an OS to manage files. The system controls how data is saved or retrieved.

/boot – contains file that is used by the boot loader

/root – root user home directory

/dev – system devices(eg. Disk,cdrom,speakers etc)

/etc – configuration files

/bin - /usr/bin – everyday user commands

/sbin - /usr/sbin – system/filesystem commands

/opt – optional add on applications(not part of OS apps)

/proc – running processes(only exist in memory)

/lib - /usr/lib – C programming lib files needed by commands and apps

/tmp - directory for temporary files

/hom – directory for user

/var – system logs

/run – system daemons that start very early to store temporary runtime files like PID files

/mnt – to mount external filesystem

/media – for cdrom mounts

cd – change directory

pwd – print working directory

ls – list

whoami

ls -l

cd ..

su - -> become root user

ls -lrt

**Type of files**

drwx-xr-x – directory

lrwxrwxrwx – link

-rw-w-r-- - file

- ->regular file

D -> directory

l –> link

c –> special file or device file

p –> named pipe

b –> block device

change password – passwd

FILE SYSTEM PATHS

Absolute Path – begins with /– cd /var/log

Relative Path – does not begin with /, identifies a location relative to your current position – cd /var , cd log

Creating files

touch filename -can create multiple files ( touch filename1 filename2)

cp filename filename1

vi

creating directories

mkdir foldername foldername1

copy directory – cp -r sourcefolder destinationfolder

find a file - find . -name “filename” or find / -name “filename”

to find the location of filename - locate filename

To update local database – updatedb and then give locate command so that it shows the location. Only when the database is updated the locate command works.

Wildcards

Permissions

User group others

-rwxr-xr-x – read write execute permissions

- -> filetype

useradd

groupadd

chown -R ansible:devops /opt

chmod o-r /opt - > remove read permission for others

chmod g+r /opt -> add read permission for group

u – user, g – group, o – others

4- read, 2-write, 1 -execute

chmod 777 filename – full permissions

------------------------------------------------------------------------------------------------------------------------Bash Scripting

Bash scripting is used to automate repetitive tasks and simplify system administration.

#! – Shebang/hashbang is used to specify the interpreter that should be used to execute the script.

/bin/bash – tells the system to use the bash shell located at /bin/bash to interpret the script

#!/bin/bash

echo “The uptime of the system is” //prints The uptime of the system is

uptime //run commands

to execute the script - ./script.sh

# - comments

#!/bin/bash

sudo yum install wget unzp httpd -y

sudo systemctl start httpd

sudo systemctl enable httpd

mkdir -p /tmp/webfiles

cd /tmp/webfiles

wget <https://www/tooplate.com/zip-templates/2098_health.zip>

unzip 2098\_health.zip

sudo cp -r 2098\_health/\* /var/www/html

systemctl restart httpd

rm -rf /tmp/webfiles

SKILL=”DevOps”

echo $SKILL – DevOps

#!/bin/bash

TEMPDIR=/data/sample/example

#Creating temp directory

mkdir -p $TEMPDIR

cd $TEMPDIR

when we give $Y without giving the variable a value, we can give the values in arguments

example –

#!/bin/bash

echo “Value of 0 is”

$0

echo “Value of 1 is”

$1

Execute the script – ./script.sh Linux

Output – Value of 0 is

./script.sh

Value of 1 is

Linux

To print exit status of last command – echo $?

0 – true

Non zero - false

Use double quotes to print $ value in output

VIRUS=”COVID19”

echo “Due to $VIRUS”

or

VIRUS=”COVID19"

echo ‘Due to \$VIRUS’

prints -> Due to COVID19

.bashrc – environment variables(root user)

/etc/profile – globally for all users

read – to enter in console

#!/bin/bash

echo “Enter your skills:”

read SKILL

echo “Your $SKILL skill is in high demand”

MM HH DOM mm DOW COMMAND

\* \* \* \* \* /opt/scripts/11\_monit.sh $>> /var/log/monit\_httpd.log

LOOPS

Decision making – if

#!/bin/bash

read -p “Enter a number: “ NUM

echo

if [ $NUM -gt 100 ]

then

echo “num is greater than 100”

sleep 3

date

fi

elif

#!/bin/bash

value=$(ip addr show | grep -v LOOPBACK | grep -ic mtu)

if [ $value -eq 1 ]

then

echo “1 active network found”

elif [ $value -gt 1 ]

then

echo “value greater than 1”

else

echo “No active interface found”

fi

fi – to mark an end for if statement

For loop

#!/bin/bash

for VAR1 in java .net python ruby php

do

echo “Looping”

echo “Value of VAR1 is $VAR1”

date

done

#!/bin/bash

MYUSERS-“alpha beta gamma”

for usr in $MYUSERS

do

echo “Adding users $MYUSERS”

useradd $usr

id $usr

done

for i in {0..10..2}

do

echo “$i”

done

for (( i=0; i<=5; i++ ))

do

echo "Element $i"

done

infinite loop – for (( ; ; ))

While loops

#!/bin/bash

counter=0

while [ $counter -lt 5 ]

do

echo “Looping”

echo “Value of counter is $counter”

counter=$(( $counter+1 ))

done

Infinite loop – while true