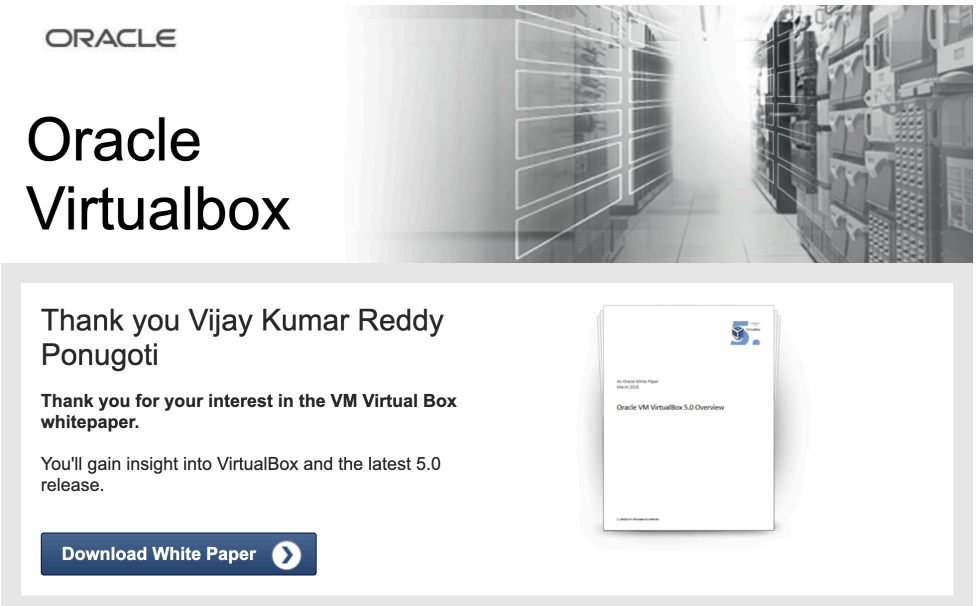
CS 553
CLOUD COMPUTING

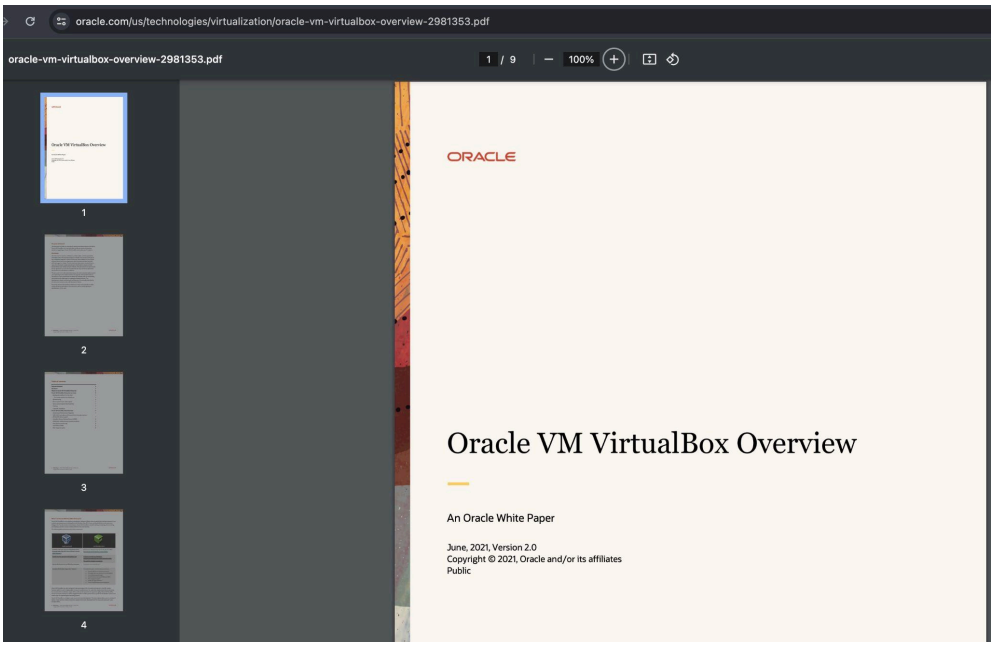
HOMEWORK-1

Vijay Kumar Reddy Ponugoti
A20537909
01/20/2024

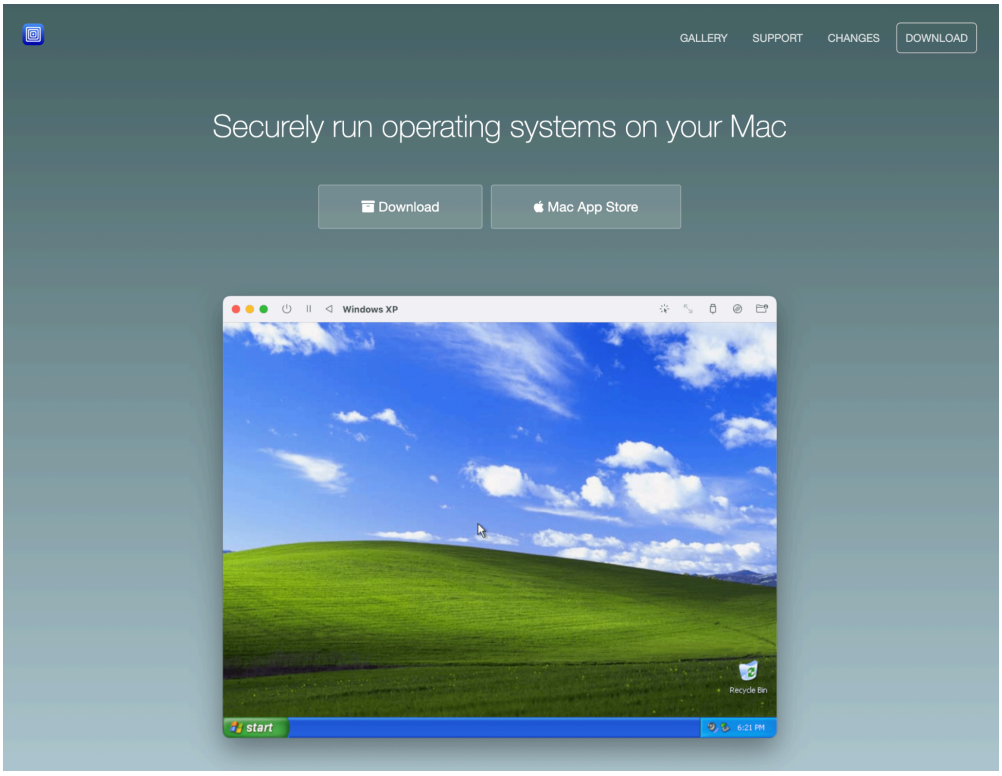
1. Question 1

a. Read & Download Oracle VirtualBox White Paper

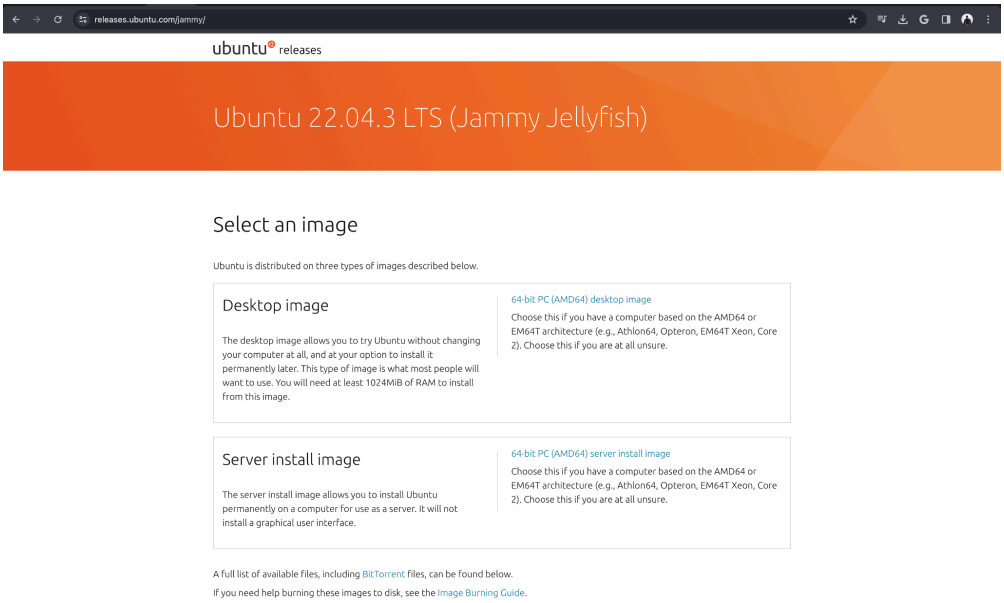




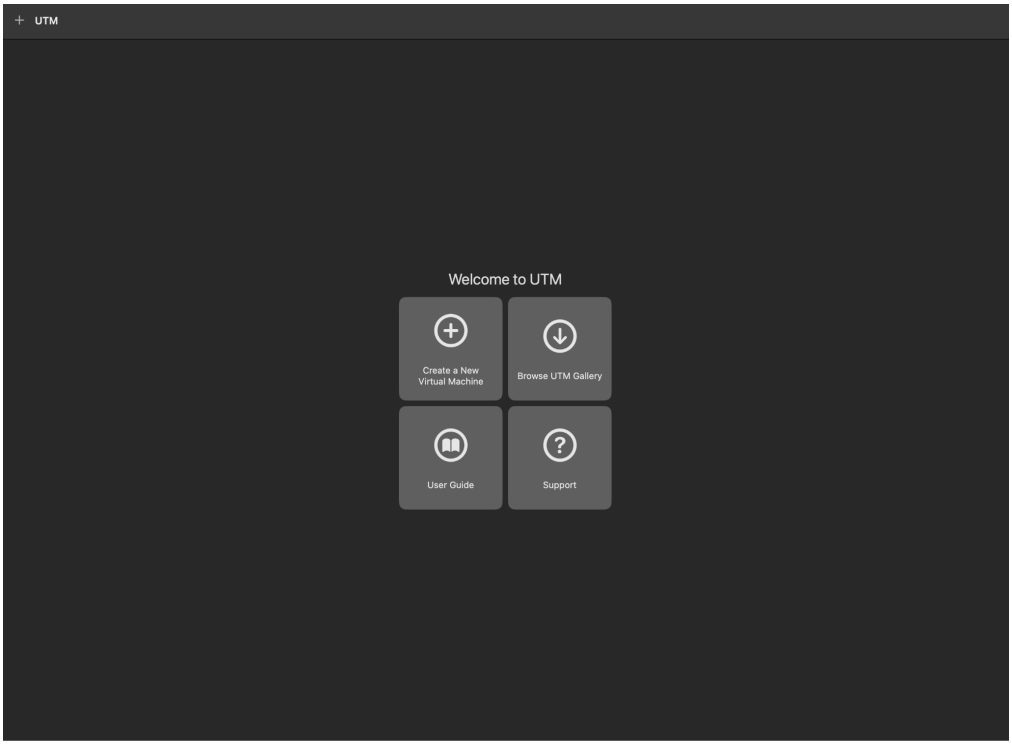
b. Download VirtualBox 7.0



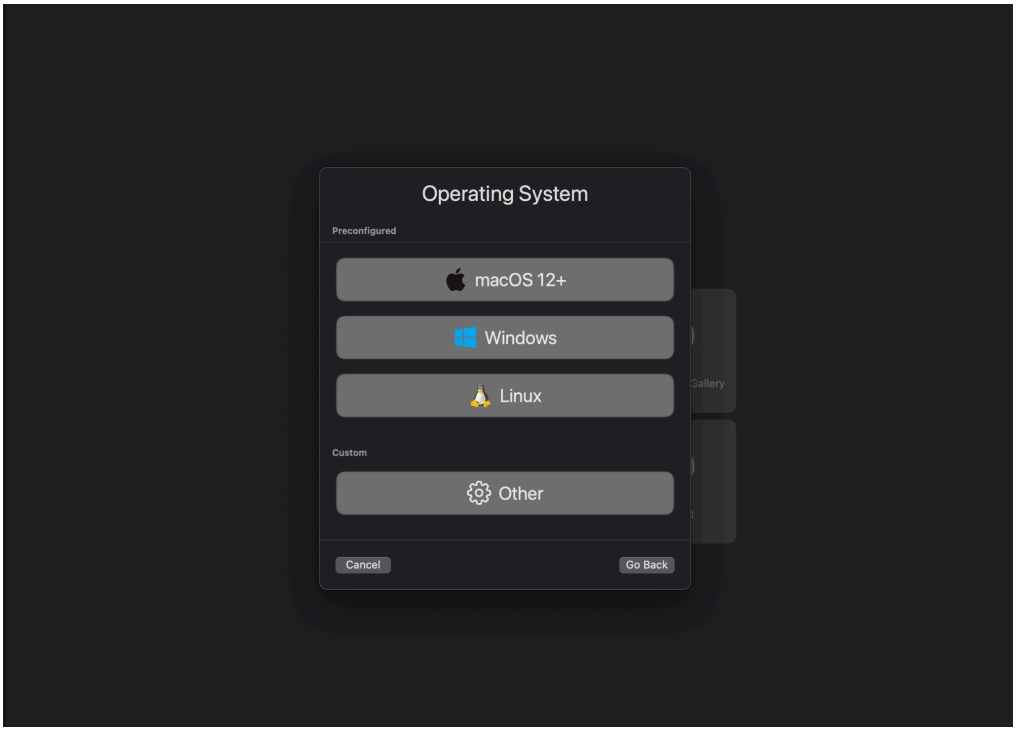
c. Install VirtualBox 7.0 using UTM

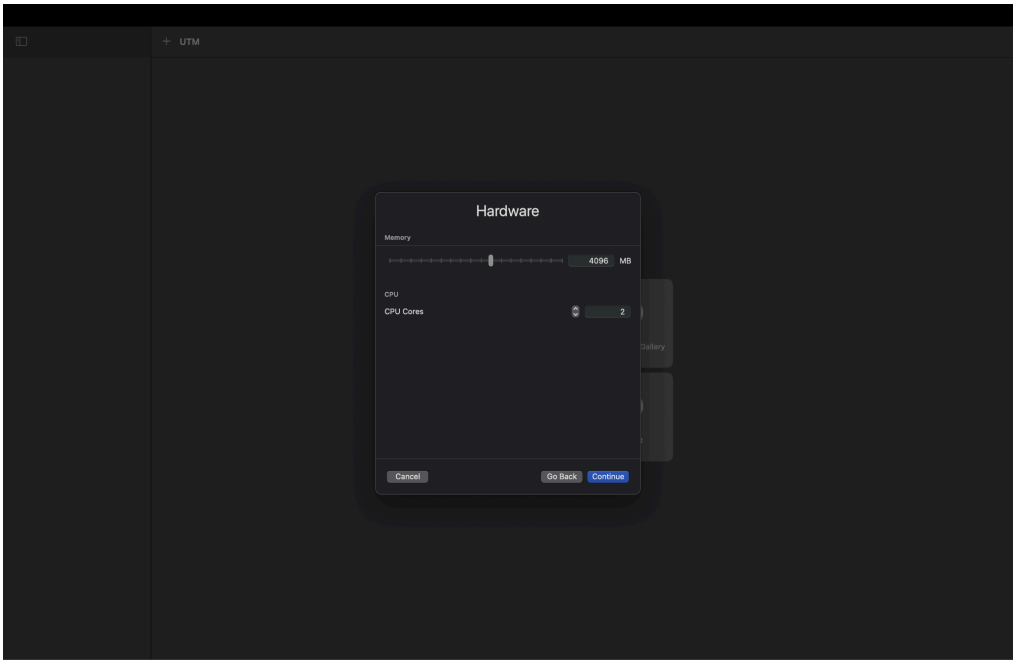


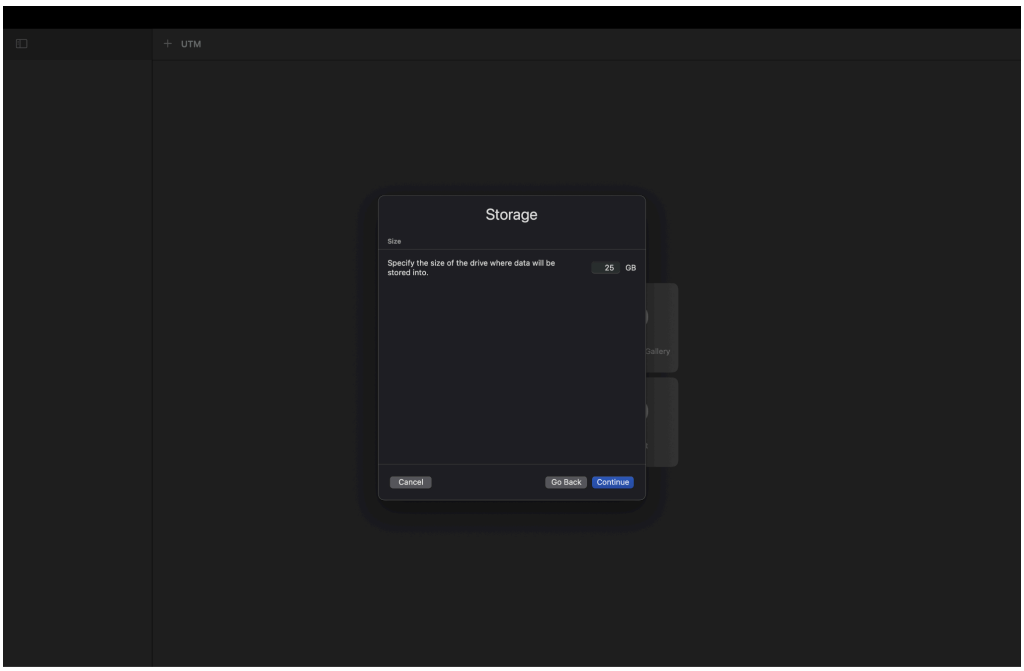
d. Download Ubuntu 22.04 Linux ISO image



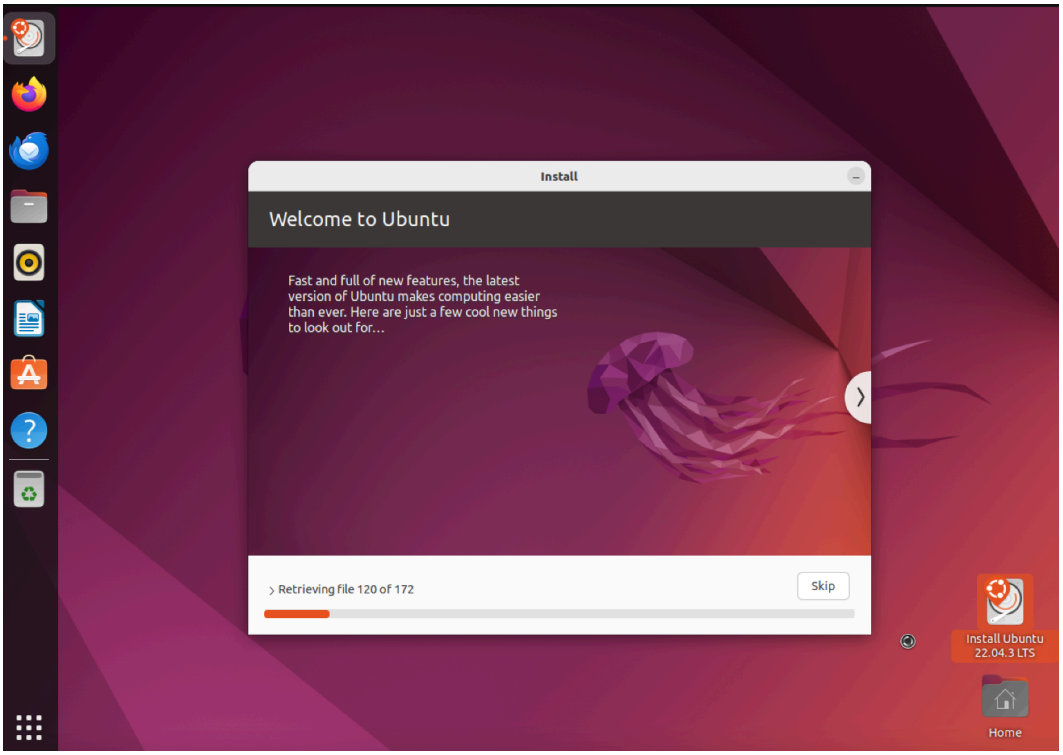
e. Create Virtual Machine (VM), to support Linux, Ubuntu, 64-bit, 4GB RAM, Virtual Disk 25GB, VDI image, dynamically allocated, 2-core, and a network interface with a Bridged Adapter in the machine



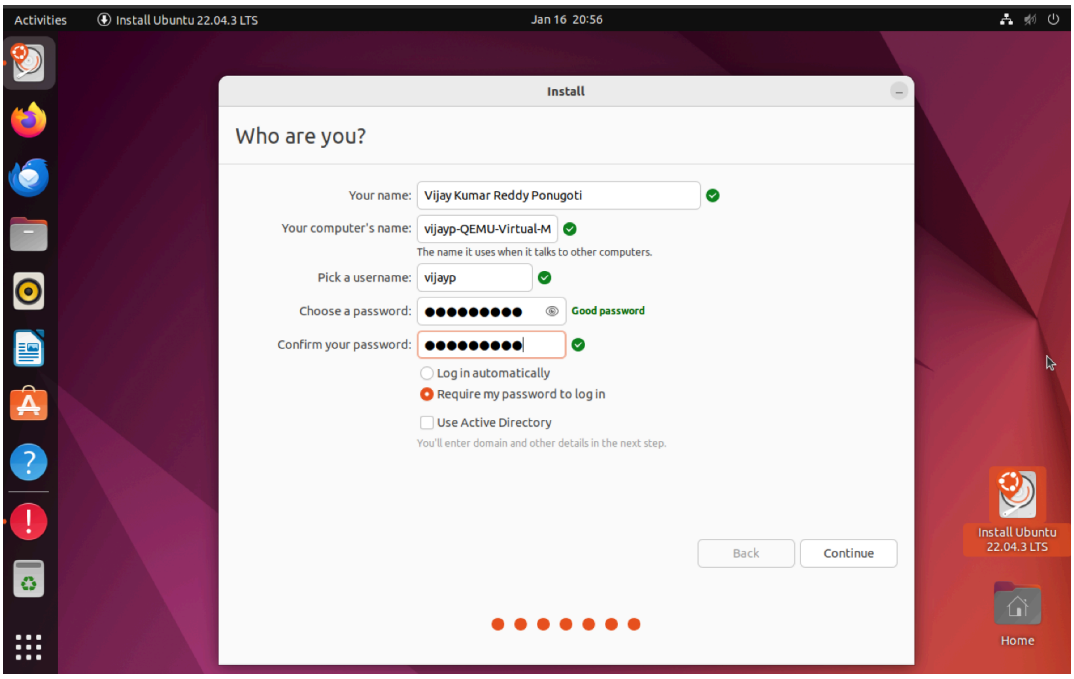

UTM
Hardware
Memory
4096
MB
CPU
CPU Cores
2
Cancel
Go Back
Continue


UTM
Storage
Size
Specify the size of the drive where data will be stored into.
25
GB
Cancel
Go Back
Continue

f. Install Linux from the ISO image



g. Create a user id and password



h. Turn on Firewall and block all ports

```
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status
[sudo] password for vijayp:
Status: inactive
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw enable
Firewall is active and enabled on system startup
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status
Status: active
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw disable
Firewall stopped and disabled on system startup
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status
Status: inactive
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw default deny outgoing
Default outgoing policy changed to 'deny'
(be sure to update your rules accordingly)
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status verbose
Status: inactive
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw enable
Firewall is active and enabled on system startup
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), deny (outgoing), disabled (routed)
New profiles: skip
```

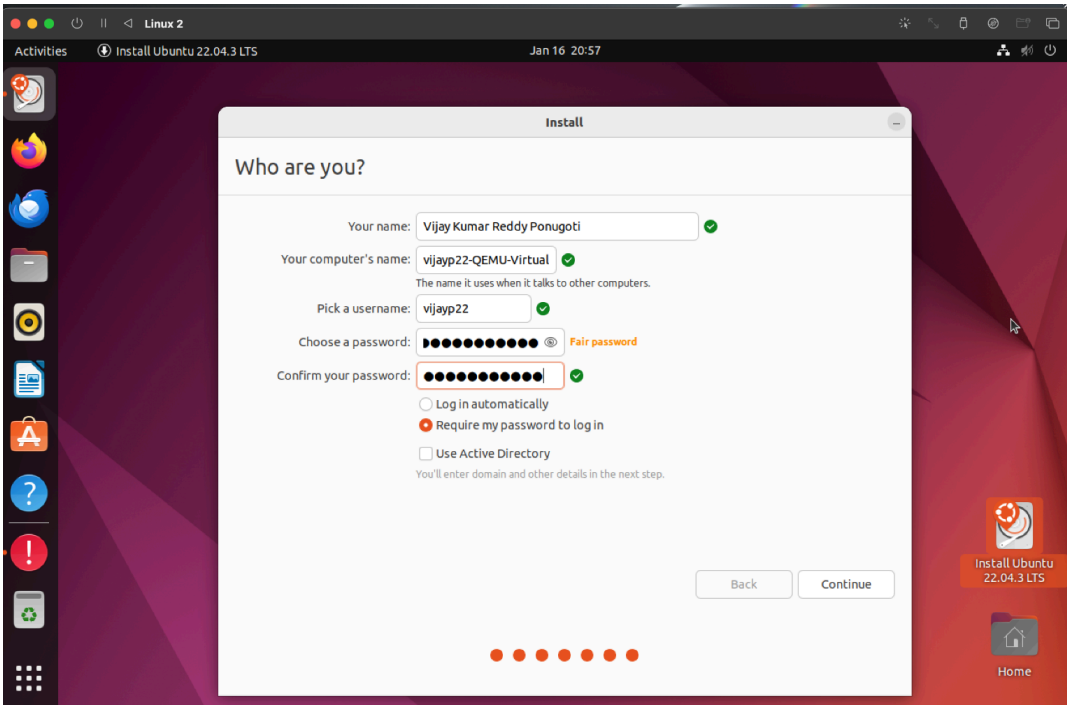
i. Enable SSH access to your new Linux installation; open SSH port in firewall

```
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo service ssh start
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw allow 22/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), deny (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW IN    Anywhere
22/tcp (v6)                ALLOW IN    Anywhere (v6)
```

```
vijayp@vijayp-QEMU-Virtual-Machine:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:M7DrI8bVdmPnhctWphCA8iTUmgcsXr4k0UeiYixqF0U.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
vijayp@localhost's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-generic aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro
```

```
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo apt update
Hit:1 http://us.ports.ubuntu.com/ubuntu-ports jammy InRelease
Hit:2 http://us.ports.ubuntu.com/ubuntu-ports jammy-backports InRelease
Get:3 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates InRelease [119 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease [110 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports jammy-proposed InRelease [270 kB]
Fetched 499 kB in 2s (223 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
vijayp@vijayp-QEMU-Virtual-Machine:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
```

j. Repeat steps 5 through 9, and create another VM with the same specifications as the first one



```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo service ssh start
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo ufw allow 22/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), deny (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW IN    Anywhere
22/tcp (v6)                ALLOW IN    Anywhere (v6)
```

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:SanR09gWW3lt5V3/W4v97V3OaSY60OY2NJ0PHy6cynI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
vijayp22@localhost's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-generic aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

k. Create private/public keys and install them properly in both of your new VMs

VM-1:

```
vijayp@vijayp-QEMU-Virtual-Machine: ~
vijayp@vijayp-QEMU-Virtual-Machine:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/vijayp/.ssh/id_rsa):
/home/vijayp/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/vijayp/.ssh/id_rsa
Your public key has been saved in /home/vijayp/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Y2tLwk92t2Qwtz1Mxp6PRpfc+/lYBn3q7MENp0oL/Q4 vijayp@vijayp-QEMU-Virtual-Machine
The key's randomart image is:
+---[RSA 3072]----+
|                 |
|                 |
|                 |
|             . . |
|        S o . B *|
|     . . o = O.@+|
|      o * o E.@.=|
|       B o * O.X.|
|        o   ++O.*|
+----[SHA256]-----+
vijayp@vijayp-QEMU-Virtual-Machine:~$
```

VM-2:

```
vijayp22@vijayp22-QEMU-Virtual-Machine: ~
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/vijayp22/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/vijayp22/.ssh/id_rsa
Your public key has been saved in /home/vijayp22/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:hb8sRDw96GZLDj8zXZ/+rysJoYzFRsAmWijLhsdV0xE vijayp22@vijayp22-QEMU-Virtua
l-Machine
The key's randomart image is:
+---[RSA 3072]----+
|    . o+oEo      |
|. . + ooo+       |
|o+ + o o* +      |
|oo+    o++..     |
|..    .=S....    |
|      .Oo+.o . . |
|        O +. .o  |
|          =  o.  |
|             .+++|
+----[SHA256]-----+
```

I. Test that you can connect remotely to your VMs with your keys, from one VM to the other VM

VM-1:

```
vijayp@vijayp-QEMU-Virtual-Machine: ~
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ssh vijayp@192.168.64.3
vijayp@192.168.64.3's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-generic aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Jan 16 22:02:50 2024 from 127.0.0.1
vijayp@vijayp-QEMU-Virtual-Machine:~$
```

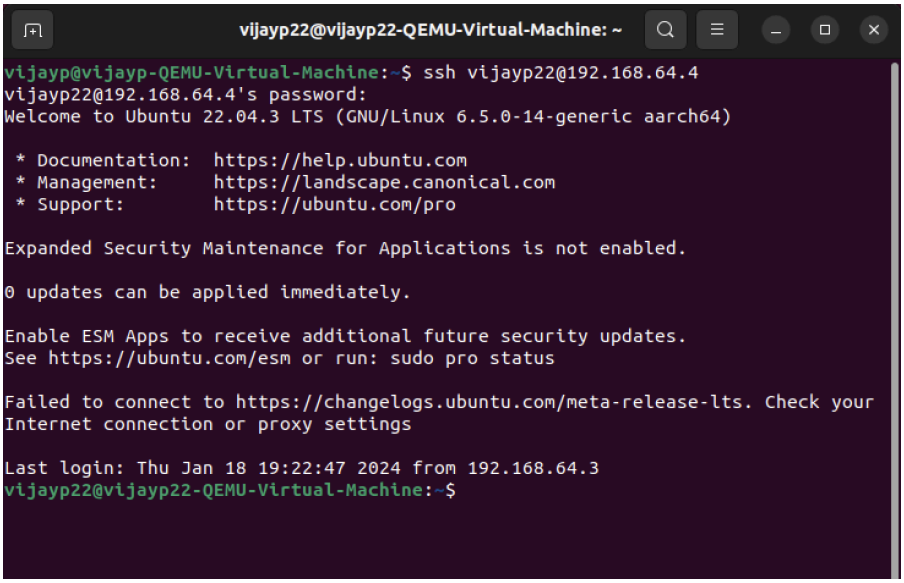
VM-2:

```
vijayp22@vijayp22-QEMU-Virtual-Machine: ~
vijayp@vijayp-QEMU-Virtual-Machine:~$ ssh vijayp22@192.168.64.4
vijayp22@192.168.64.4's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-generic aarch64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Last login: Thu Jan 18 19:22:47 2024 from 192.168.64.3
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

2. Question 2

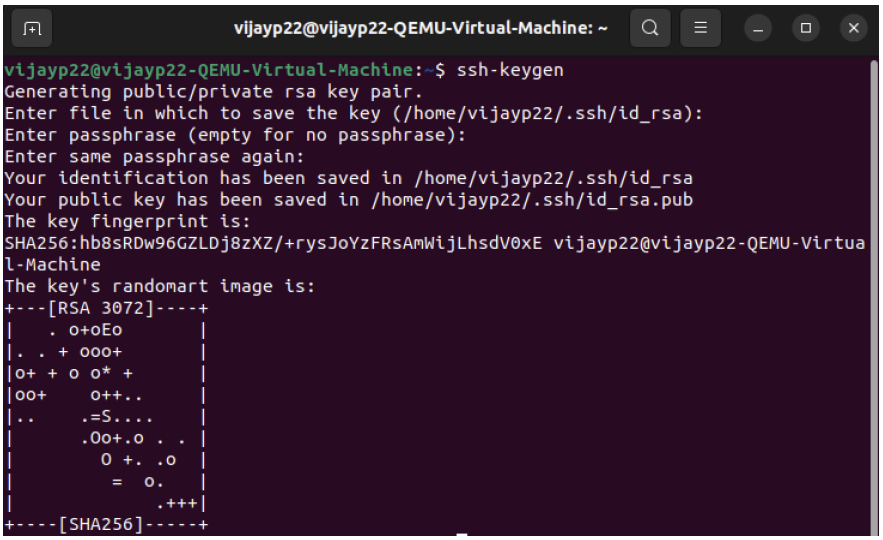
a. ssh

A strong Linux program called SSH (Secure Shell) enables you to create a safe, encrypted network connection with another computer. Similar to a virtual tunnel, it shields your data from prying eyes while allowing you to remotely operate the other computer. Consider it as a safe link connecting your computer to the one that is located far away.



b. ssh-keygen

The tool called ssh-keygen is used to generate and maintain SSH keys, which are essential for secure remote verification and interaction. The main thing it does is generate pairs of public and private keys, which are necessary to make safe SSH connections.



c. scp

With the use of the SSH protocol and the Linux command-line utility SCP (Secure Copy), you can safely move data among computers connected to a network. It is frequently used for file sharing between servers, backups, and remote file management.

```
vijayp22@vijayp22-QEMU-Virtual-Machine: ~
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ scp file1.txt vijayp@192.168.64.3:/hom
e/vijayp/newfile1.txt
vijayp@192.168.64.3's password:
file1.txt
                                                                    100%   16     2
6.5KB/s   00:00
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

d. history

You can save time and effort by using the Linux command history, which maintains a log of all commands typed into the terminal. It resembles a breadcrumb trail of the actions you took at the terminal.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ history
    1  sudo ufw default deny outgoing
    2  sudo ufw default deny incoming
    3  sudo ufw status verbose
    4  sudo service ssh start
    5  clear
```

e. sudo

Sudo, an acronym for "substitute user do," enables authorized users to execute commands with the access rights of another user, usually the root user (the most powerful user on a Linux system). Because of this, security and accountability are increased because you may carry out administrative operations without logging in as root.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo ufw enable
Firewall is active and enabled on system startup
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

f. ip

The ip command is a robust tool in Linux that gives you the ability to control and set up network interfaces, routing, and tunnels. Compared to the more dated ifconfig command, it provides a more thorough and contemporary method.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ip route show
default via 192.168.64.1 dev enp0s1 proto dhcp metric 100
169.254.0.0/16 dev enp0s1 scope link metric 1000
192.168.64.0/24 dev enp0s1 proto kernel scope link src 192.168.64.4 metric 100
```

g. touch

There are two main purposes for it:

-A new, empty file with the given name is created instantly when you type 'touch' filename. For files you'll fill up later, this is useful.

-'touch' alters the timestamps of files that already exist without changing the content. Atime and Mtime, the access and modification times, are usually updated to reflect the current time.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ touch file2.txt
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls
Desktop    file1.txt  Pictures  Templates     v2.txt  wget-log
Documents  file2.txt  Public    textfile.txt  v3.txt  wget-log.1
Downloads  Music      snap      v1.txt        Videos
```

h. ls

The "ls" command is a standard tool for browsing and inspecting directories' contents. It lists the names of files and directories in the present working directory by default, but it is also flexible. It is required to navigate and manage files easily in the Linux shell.

```
vijayp@vijayp-QEMU-Virtual-Machine:~$ ls -l ~/
total 36
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Desktop
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Documents
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Downloads
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Music
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Pictures
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Public
drwx------ 3 vijayp vijayp 4096 Jan 16 20:33 snap
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Templates
drwxr-xr-x 2 vijayp vijayp 4096 Jan 16 20:33 Videos
vijayp@vijayp-QEMU-Virtual-Machine:~$ ls -l ~/.ssh
total 16
-rw------- 1 vijayp vijayp 3401 Jan 16 22:13 id_rsa
-rw-r--r-- 1 vijayp vijayp  760 Jan 16 22:13 id_rsa.pub
-rw------- 1 vijayp vijayp  978 Jan 16 22:02 known_hosts
-rw-r--r-- 1 vijayp vijayp  142 Jan 16 22:02 known_hosts.old
```

i. mkdir

In Linux, the mkdir command is used to create new directories, much as folders in graphical interfaces. It's simple to use: just type mkdir followed by the desired directory name. For example, mkdir Documents will create a directory called "Documents" at your current location.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ mkdir newdirectory
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls
Desktop    file1.txt  newdirectory  snap          v1.txt  Videos    wget-log.1
Documents  file2.txt  Pictures      Templates     v2.txt  vijay
Downloads  Music      Public        textfile.txt  v3.txt  wget-log
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

j. cd

The "cd" command functions similarly to a GPS in navigating among directories. It stands for "change directory" and allows you to navigate the file system in the same way you would browse different files on your desktop. To use it, type "cd" followed by the path of the directory you wish to access.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cd vijay
vijayp22@vijayp22-QEMU-Virtual-Machine:~/vijay$
```

k. dd

dd, often known as "disk destroyer" or "data duplicator," is a strong Linux tool for copying and converting raw data from one source to another.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo dd if=file1.txt of=/dev/sda of=backup
.img bs=4M
0+1 records in
0+1 records out
16 bytes copied, 0.00021762 s, 73.5 kB/s
```

l. fdisk

fdisk is a sophisticated command-line tool in Linux that manages disk partitions. It's similar to a specialized toolkit for organizing and separating your hard drive into several portions, each capable of housing multiple operating systems or data.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo fdisk -l
Disk /dev/loop0: 69.07 MiB, 72429568 bytes, 141464 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop1: 4 KiB, 4096 bytes, 8 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop2: 230.82 MiB, 242032640 bytes, 472720 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

m. apt

apt, which stands for Advanced Packaging utility, is a powerful command-line utility that simplifies software administration in Debian-based Linux distributions such as Ubuntu. It allows you to easily install, upgrade, delete, and manage software packages (including their dependencies) from a large number of repositories.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo apt update
Hit:1 http://us.ports.ubuntu.com/ubuntu-ports jammy InRelease
Get:2 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates InRelease [119 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease [110 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports jammy-proposed InRelease [270 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports jammy-security/main arm64 Packages [908 kB]
Get:6 http://us.ports.ubuntu.com/ubuntu-ports jammy-backports InRelease [109 kB]
Get:7 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 Packages [1,118 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports jammy-security/main Translation-en [201 kB]
Get:9 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates/main Translation-en [262 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports jammy-security/restricted arm64 Packages [851 kB]
Get:11 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates/universe arm64 Packages [963 kB]
Get:12 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates/universe Translation-en [231 kB]
Get:13 http://ports.ubuntu.com/ubuntu-ports jammy-security/restricted Translation-en [204 kB]
Get:14 http://ports.ubuntu.com/ubuntu-ports jammy-security/universe arm64 Packages [766 kB]
Get:15 http://us.ports.ubuntu.com/ubuntu-ports jammy-updates/multiverse arm64 Packages [24.0 kB]
Get:16 http://us.ports.ubuntu.com/ubuntu-ports jammy-backports/main arm64 Packages [64.4 kB]
Get:17 http://us.ports.ubuntu.com/ubuntu-ports jammy-backports/universe arm64 Packages [26.2 kB]
Get:18 http://ports.ubuntu.com/ubuntu-ports jammy-security/universe Translation-en [158 kB]
Get:19 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/main arm64 Packages [139 kB]
Get:20 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/main Translation-en [40.1 kB]
Get:21 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/restricted arm64 Packages [149 kB]
Get:22 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/restricted Translation-en [35.6 kB]
Fetched 6,750 kB in 1min 36s (70.3 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

n. vi

vi is a sophisticated and versatile text editor that is widely used in Linux and Unix environments. It functions in two different modes:

-The command mode is the default mode, in which you can explore, edit, and change text with single or multiple keystrokes. Working on the file is similar to using a keyboard-driven control panel.

-The insert mode is used to actually enter text into the file. To return to command mode, use the Esc key.

```
This is a new file.
It is created by vi editor
```

o. time

The time command in Linux serves as a useful stopwatch, monitoring the running time of a particular command while giving data on its resource utilization.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ time (ls -l && echo "Hello World")
total 68
-rw-r--r-- 1 root     root       16 Jan 18 20:42 backup.img
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Desktop
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Documents
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Downloads
-rw-rw-r-- 1 vijayp22 vijayp22   16 Jan 18 19:40 file1.txt
-rw-rw-r-- 1 vijayp22 vijayp22    0 Jan 18 20:20 file2.txt
-rw-rw-r-- 1 vijayp22 vijayp22   47 Jan 18 20:52 file3.txt
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Music
drwxrwxr-x 2 vijayp22 vijayp22 4096 Jan 18 20:33 newdirectory
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Pictures
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Public
drwx------ 3 vijayp22 vijayp22 4096 Jan 16 21:10 snap
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Templates
-rw-rw-r-- 1 vijayp22 vijayp22   52 Jan 17 18:27 textfile.txt
-rw-rw-r-- 1 vijayp22 vijayp22    0 Jan 18 19:55 v1.txt
-rw-rw-r-- 1 vijayp22 vijayp22    0 Jan 18 19:55 v2.txt
-rw-rw-r-- 1 vijayp22 vijayp22    0 Jan 18 19:55 v3.txt
drwxr-xr-x 2 vijayp22 vijayp22 4096 Jan 16 21:10 Videos
drwxrwxr-x 2 vijayp22 vijayp22 4096 Jan 18 20:33 vijay
-rw-rw-r-- 1 vijayp22 vijayp22  165 Jan 17 18:35 wget-log
-rw-rw-r-- 1 vijayp22 vijayp22  219 Jan 17 18:38 wget-log.1
Hello World

real    0m0.004s
user    0m0.001s
sys     0m0.003s
```

p. tar

tar, which is short for Tape Archive, is a flexible command for packing and compressing numerous files and directories into one archive file, usually with the.tar extension. It functions as a virtual container for your data, making them easier to handle and store.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ tar cvf compress_txt.tar *.txt
file1.txt
file2.txt
file3.txt
textfile.txt
v1.txt
v2.txt
v3.txt
```

q. rm

rm is a strong Linux command that means "remove." It is used to permanently remove files and directories from the file system. Similar to a digital shredder, this tool is vital for clearing up space and deleting undesirable objects.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls
backup.img        Desktop    Downloads  file2.txt  Music         Pictures  snap       textfile.txt  v2.txt  Videos  wget-log
compress_txt.tar  Documents  file1.txt  file3.txt  newdirectory  Public    Templates  v1.txt        v3.txt  vijay   wget-log.1
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ rm backup.img
rm: remove write-protected regular file 'backup.img'? yes
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls
compress_txt.tar  Documents  file1.txt  file3.txt  newdirectory  Public  Templates     v1.txt  v3.txt  vijay     wget-log.1
Desktop           Downloads  file2.txt  Music      Pictures      snap    textfile.txt  v2.txt  Videos  wget-log
```

r. cat

The cat command, which stands for "concatenate," is a flexible tool for displaying, combining, and creating files in the Linux terminal.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat file1.txt
This is a file.
```

s. bash

Bash, which is short for Bourne-Again SHell, is the most widely used command-line parser in Linux and other Unix-based operating systems. It serves as a link between you and the OS's core, accepting and executing your commands to complete various tasks.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat>bashfile1.sh
#!/bin/bash
for i in {1..5}
do
 echo $i
done
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ bash bashfile1.sh
1
2
3
4
5
```

t. more

more allows you to display the contents of a text file one screenful at a time, which is useful for exploring huge files without overwhelming your terminal.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ more bashfile1.sh
#!/bin/bash
for i in {1..5}
do
 echo $i
done
```

u. watch

It functions as a relentless assistant, repeatedly running a command for you while showing the results on the screen. This makes it perfect for tracking changes or events over time. Consider it a built-in time-lapse cam for your terminals.

```
Every 2.0s: ls -l                                                    vijayp22-QEMU-Virtual-Machine: Thu Jan 18 21:11:02 2024

total 84
-rw-rw-r-- 1 vijayp22 vijayp22    45 Jan 18 21:09 bashfile1.sh
-rw-rw-r-- 1 vijayp22 vijayp22    44 Jan 18 21:08 bashfile.sh
-rw-rw-r-- 1 vijayp22 vijayp22 10240 Jan 18 21:04 compress_txt.tar
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Desktop
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Documents
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Downloads
-rw-rw-r-- 1 vijayp22 vijayp22    16 Jan 18 19:40 file1.txt
-rw-rw-r-- 1 vijayp22 vijayp22     0 Jan 18 20:20 file2.txt
-rw-rw-r-- 1 vijayp22 vijayp22    47 Jan 18 20:52 file3.txt
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Music
drwxrwxr-x 2 vijayp22 vijayp22  4096 Jan 18 20:33 newdirectory
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Pictures
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Public
drwx------ 3 vijayp22 vijayp22  4096 Jan 16 21:10 snap
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Templates
-rw-rw-r-- 1 vijayp22 vijayp22    52 Jan 17 18:27 textfile.txt
-rw-rw-r-- 1 vijayp22 vijayp22     0 Jan 18 19:55 v1.txt
-rw-rw-r-- 1 vijayp22 vijayp22     0 Jan 18 19:55 v2.txt
-rw-rw-r-- 1 vijayp22 vijayp22     0 Jan 18 19:55 v3.txt
drwxr-xr-x 2 vijayp22 vijayp22  4096 Jan 16 21:10 Videos
drwxrwxr-x 2 vijayp22 vijayp22  4096 Jan 18 20:33 vijay
-rw-rw-r-- 1 vijayp22 vijayp22   165 Jan 17 18:35 wget-log
-rw-rw-r-- 1 vijayp22 vijayp22   219 Jan 17 18:38 wget-log.1
```

v. ps

The "ps" command displays information about the active processes on your system. It refers to "process status" and provides an overview of what's going on under the hood.

```
vijayp22     995  0.0  0.9 589896 37888 ?        Sl   19:23   0:00  \_ /usr/libe
vijayp22    1004  0.0  0.7 854232 28544 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1005  0.0  0.1 323616  7168 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1010  0.0  0.1 245188  5888 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1023  0.0  0.3 349640 13440 ?        Sl   19:23   0:00  \_ /usr/libe
vijayp22    1027  0.0  0.1 245064  5888 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1036  0.0  0.1 156716  5120 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1037  0.0  0.6 683796 26752 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1043  0.0  0.1 246380  6144 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1071  0.0  0.1 162948  6656 ?        Sl   19:23   0:00  \_ /usr/libe
vijayp22    1075  0.0  0.6 2608040 25604 ?       Sl   19:23   0:00  \_ /usr/bin/
vijayp22    1084  0.0  0.0   2304  1280 ?        Ss   19:23   0:00  \_ sh -c /us
vijayp22    1112  0.0  0.2 471376 11060 ?        Sl   19:23   0:04  |   \_ /usr/
vijayp22    1182  0.0  0.1 246012  6784 ?        Sl   19:23   0:00  |       \_ /
vijayp22    1183  0.0  0.6 358984 27628 ?        Sl   19:23   0:01  |       \_ /
vijayp22    1206  0.0  0.1 172068  6656 ?        Sl   19:23   0:01  |       \_ /
vijayp22    1085  0.0  0.1 319104  6144 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1086  0.0  0.6 400660 25092 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1087  0.0  0.3 387140 14592 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1089  0.0  0.1 320856  7168 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1091  0.0  0.5 351740 20968 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1092  0.0  0.6 655024 25448 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1098  0.0  0.6 388484 24340 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1099  0.0  0.2 259304  9984 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1100  0.0  0.1 466756  6144 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1101  0.0  0.1 244896  5504 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1103  0.0  0.2 475260  9472 ?        Ssl  19:23   0:00  \_ /usr/libe
vijayp22    1104  0.0  0.1 468792  7552 ?        Ssl  19:23   0:00  \_ /usr/libe
```

w. top

It functions as a real-time dashboard for your system's functionality. It provides an immediate overview of ongoing processes, CPU and memory utilization, and other system statistics.

```
top - 21:12:29 up  1:48,  1 user,  load average: 0.40, 0.22, 0.13
Tasks: 180 total,   2 running, 178 sleeping,   0 stopped,   0 zombie
%Cpu(s):  2.2 us,  0.7 sy,  0.0 ni, 97.0 id,  0.2 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   3905.3 total,   1985.1 free,    795.7 used,   1124.4 buff/cache
MiB Swap:   2630.0 total,   2630.0 free,      0.0 used.   2880.1 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
    900 vijayp22  20   0 4622936 426684 139056 S   4.0  10.7   3:07.05 gnome-shell
    340 systemd+  20   0   15000   6272   5504 S   0.3   0.2   0:13.01 systemd-oomd
   4899 vijayp22  20   0  578532  55400  42196 S   0.3   1.4   0:00.71 gnome-terminal-
   5086 vijayp22  20   0   21332   3456   2816 R   0.3   0.1   0:00.02 top
      1 root      20   0  166768  10724   7396 S   0.0   0.3   0:00.80 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kthreadd
      3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_gp
      4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_par_gp
      5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 slub_flushwq
      6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 netns
      8 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
     11 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
     12 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_kthread
     13 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
     14 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
     15 root      20   0       0      0      0 R   0.0   0.0   0:00.21 ksoftirqd/0
     16 root      20   0       0      0      0 I   0.0   0.0   0:01.55 rcu_preempt
     17 root      rt   0       0      0      0 S   0.0   0.0   0:00.07 migration/0
     18 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/0
     19 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
     20 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
     21 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/1
     22 root      rt   0       0      0      0 S   0.0   0.0   0:00.07 migration/1
     23 root      20   0       0      0      0 S   0.0   0.0   0:00.07 ksoftirqd/1
     25 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/1:0H-events_highpri
     26 root      20   0       0      0      0 S   0.0   0.0   0:00.00 kdevtmpfs
     27 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 inet_frag_wq
```

x. htop

htop is a sophisticated command-line utility that displays the processes running on your Linux system in real time and interactively. It's essentially a visual improvement to the traditional "top" command, with a user-friendly interface, expanded functions, and improved customisation.

```
  0[|||                                                          2.0%]   Tasks: 113, 248 thr; 1 running
  1[|||||                                                        3.9%]   Load average: 0.47 0.27 0.15
Mem[||||||
```

y. gcc

GCC (GNU Compiler Collection) is a sophisticated and adaptable command-line tool for Linux that compiles and links source code composed of a variety of programming languages, particularly C, C++, and Fortran. It effectively converts human-readable code to machine-executable commands that a computer can interpret and execute.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat>cprgm.c
#include<stdio.h>
int main()
{
  printf("This is a C program.");
  return 0;
}
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ gcc cprgm.c
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ /.a.out
bash: /.a.out: No such file or directory
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ./a.out
This is a C program.vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

z. tail

The tail command in Linux enables you to quickly review the final segment of a file, which usually is the last 10 lines by default. It's like looking at the end of a file without opening the whole thing.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ tail -n 3 cprgm.c
  printf("This is a C program.");
  return 0;
}
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

aa. grep

Grep, which stands for "global regular expression print," is a sophisticated command-line utility that searches for specific text patterns in files. It's comparable to a very efficient "find" function for text-related stuff.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ grep -c "}"  cprgm.c
1
```

bb. kill

The kill command sends a signal to a process, often telling it to terminate. It's analogous to sending a message to a particular software program instructing it to halt its operations and shut down.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ps
    PID TTY          TIME CMD
   5707 pts/0    00:00:00 bash
   5786 pts/0    00:00:00 ps
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ kill 5707
```

cc. killall

It is a command that terminates numerous processes at once on the basis of their names. This differs from kill, which kills processes based on the process IDs (PIDs).

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ killall -l
HUP INT QUIT ILL TRAP ABRT BUS FPE KILL USR1 SEGV USR2 PIPE ALRM TERM STKFLT
CHLD CONT STOP TSTP TTIN TTOU URG XCPU XFSZ VTALRM PROF WINCH POLL PWR SYS
```

dd. du

du refers to "disk usage," and it is a command that allows you to evaluate and report on the amount of disk space is being used by files and directories. It's an excellent tool for discovering huge files or directories that may be taking up too much space, controlling disk utilization efficiently, and evaluating general storage consumption patterns.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ du vijay
4       vijay
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

ee. df

df, short for "disk free," acts as a useful Linux tool that displays a brief snapshot of the amount of disk space accessible through your PC's mounted file systems.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           391M  1.7M  389M   1% /run
/dev/vda2        24G   11G   13G  45% /
tmpfs           2.0G     0  2.0G   0% /dev/shm
tmpfs           5.0M  4.0K  5.0M   1% /run/lock
efivarfs        256K   25K  232K  10% /sys/firmware/efi/efivars
/dev/vda1       512M  5.3M  507M   2% /boot/efi
tmpfs           391M   92K  391M   1% /run/user/1000
```

ff. screen

Screen is a terminal switcher which enables you to open numerous virtual terminal sessions in the same terminal window. This allows you to run many processes, effortlessly move between them.

```
GNU Screen version 4.09.00 (GNU) 30-Jan-22

Copyright (c) 2018-2020 Alexander Naumov, Amadeusz Slawinski
Copyright (c) 2015-2017 Juergen Weigert, Alexander Naumov, Amadeusz Slawinski
Copyright (c) 2010-2014 Juergen Weigert, Sadrul Habib Chowdhury
Copyright (c) 2008-2009 Juergen Weigert, Michael Schroeder, Micah Cowan, Sadrul Habib Chowdhury
Copyright (c) 1993-2007 Juergen Weigert, Michael Schroeder
Copyright (c) 1987 Oliver Laumann

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either
version 3, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program (see the file COPYING); if not, see https://www.gnu.org/licenses/, or contact Free
Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02111-1301  USA.

Send bugreports, fixes, enhancements, t-shirts, money, beer & pizza to screen-devel@gnu.org


Capabilities:
+copy +remote-detach +power-detach +multi-attach +multi-user +font +color-256 +utf8 +rxvt +builtin-telnet
```

gg. vim

vim is a sophisticated and flexible text editor that is extensively used on Linux and other Unix-based platforms. It is renowned for its efficiency and adaptability.

```
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"newfile3.txt" [New]                                    0,0-1         All
```

hh. chmod

chmod is a strong Linux tool that lets you manage who can read, write, and run files and directories. It signifies "change mode," which refers to how it alters the rights associated with these things.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls -l v1.txt
-rw-rw-r-- 1 vijayp22 vijayp22 0 Jan 18 19:55 v1.txt
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ chmod u=rwx v1.txt
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls -l v1.txt
-rwxrw-r-- 1 vijayp22 vijayp22 0 Jan 18 19:55 v1.txt
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

ii. chown

The chown command modifies the permissions of files or directories by assigning them to a certain user or group. It is necessary for controlling file access and authorization management inside the system.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ chown root v1.txt
chown: changing ownership of 'v1.txt': Operation not permitted
```

jj. useradd

Generates new accounts for users on a Linux machine.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo adduser aryan
Adding user `aryan' ...
Adding new group `aryan' (1002) ...
Adding new user `aryan' (1002) with group `aryan' ...
Creating home directory `/home/aryan' ...
Copying files from `/etc/skel' ...
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
Sorry, passwords do not match.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for aryan
Enter the new value, or press ENTER for the default
        Full Name []: Aryan
        Room Number []: Saha
        Work Phone []: 1234
        Home Phone []: 4567
        Other []:
Is the information correct? [Y/n] y
```

kk. mv

mv, which refers to "move," is a flexible command for relocating and renaming files and directories in a Linux filesystem. It works by successfully transferring the material to the new location and then deleting the original, resulting in efficient file management.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ mv -i v2.txt /home/vijayp22/vijay
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ ls -l vijay
total 0
-rw-rw-r-- 1 vijayp22 vijayp22 0 Jan 18 19:55 v2.txt
```

ll. man

The "man" command acts as a built-in guide to the huge array of commands and tools. It is short for "manual" and enables you to easily obtain detailed records, or "man pages," for practically any command or operation you encounter.

```
TOUCH(1)                        User Commands                        TOUCH(1)

NAME
       touch - change file timestamps

SYNOPSIS
       touch [OPTION]... FILE...

DESCRIPTION
       Update  the  access  and modification times of each FILE to the current
       time.

       A FILE argument that does not exist is created empty, unless -c  or  -h
       is supplied.

       A  FILE  argument  string of - is handled specially and causes touch to
       change the times of the file associated with standard output.

       Mandatory arguments to long options are  mandatory  for  short  options
       too.

       -a     change only the access time
```

mm. locate

It's an extremely fast technique to locate files and directories by name. Instead of searching the entire filesystem, it refers to a pre-built database that constantly gets updated with file info. This makes it much quicker than the ‘find’ command, particularly on larger systems.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ locate v2.txt
/home/vijayp22/vijay/v2.txt
```

nn. find

It locates files and directories in a file system by specific criteria.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ find -mtime 2
./Downloads
./Videos
./Public
./.config/user-dirs.dirs
./.config/update-notifier
./.config/user-dirs.locale
./.config/goa-1.0
./.config/pulse
./.config/pulse/8e6cf6c280cf40a78ad4eb9be0d77931-stream-volumes.tdb
./.config/pulse/8e6cf6c280cf40a78ad4eb9be0d77931-device-volumes.tdb
./.config/pulse/cookie
./.config/pulse/8e6cf6c280cf40a78ad4eb9be0d77931-card-database.tdb
./.config/evolution
./.config/evolution/sources
./.config/evolution/sources/system-proxy.source
```

oo. sed

sed, which stands for "stream editor," is a robust command-line program that lets you change text files without viewing them in a standard text editor. It reads text line by line, executes provided instructions on each line, and prints the modified result.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sed -n '1,3p' cprgm.c
#include<stdio.h>
int main()
{
```

pp. awk

awk is a strong text processing tool in Linux that specializes in pattern-based searching, collecting data, and generating reports from text files.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ awk '/{/' cprgm.c
{
```

qq. diff

Compares two files line by line to find variations in content.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ diff file1.txt file2.txt
1d0
< This is a file.
```

rr. sort

Arranges lines of text in a specific sequence, from either alphabetically or numerically.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sort cprgm.c
{
}
#include<stdio.h>
int main()
  printf("This is a C program.");
  return 0;
```

ss. export

It's an embedded command that lets you make environment variables available to child processes. Environment variables contain information such as paths, settings, and customs that programs might require.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ export MY_VARIABLE="VIJAY"
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ env
```

```
MY_VARIABLE=VIJAY
QT_IM_MODULE=ibus
XDG_RUNTIME_DIR=/run/user/1000
XDG_DATA_DIRS=/usr/share/ubuntu:/usr/local/share/:/usr/share/:/var/lib/snapd/desktop
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin
GDMSESSION=ubuntu
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
_=/usr/bin/env
```

tt. pwd

pwd means "print working directory." It's a basic but useful command that informs you exactly where you are in the Linux file system. As you enter pwd in the terminal, it displays the full path to your current working directory, beginning with the root of that directory. This route allows you to navigate, arrange files, and execute instructions in the appropriate area.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ pwd
/home/vijayp22
```

uu. crontab

crontab is a program for scheduling jobs to execute automatically at certain periods or intervals on your Linux machine. It operates by changing a file called crontab, that includes an array of commands and their corresponding schedules.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ crontab -e
no crontab for vijayp22 - using an empty one

Select an editor.  To change later, run 'select-editor'.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.basic
  3. /usr/bin/vim.tiny
  4. /bin/ed

Choose 1-4 [1]:
```

vv. mount

It's similar to plugging in a USB device, however for file systems. It connects a distinct file system to a blank directory in your existing file system tree. This integration allows you to access files and directories on a different file system just as if they're part of your primary system.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=1943504k,nr_inodes=485876,mode=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=399900k,mode=755,inode64)
/dev/vda2 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,inode64)
```

ww. passwd

Typically used to update user passwords. Provides additional tools for managing password-related chores.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo passwd aryan
[sudo] password for vijayp22:
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
Sorry, passwords do not match.
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
Sorry, passwords do not match.
New password:
Retype new password:
passwd: password updated successfully
```

xx. uname

It's a command-line utility that shows important data about the OS and hardware that it's running upon.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ uname
Linux
```

yy. whereis

It instantly finds the binary, source code, and guide page files associated with a specific command.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man5/passwd.5.gz /usr/sha
re/man/man1/passwd.1.gz /usr/share/man/man1/passwd.1ssl.gz
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

zz. whatis

Quickly returns a brief description of a command, tool, or file. It functions as a convenient source of information when you require a basic idea regarding what a command accomplishes before digging into the entire manual page.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ whatis gcc
gcc (1)              - GNU project C and C++ compiler
```

```
#include<stdio.h>
int main()
{
  printf("This is a C program.");
  return 0;
}
cprgm.c (END)
```

aaa. less

It's a useful tool for quickly viewing enormous text files, a single page at a time. It's like browsing through a book rather than attempting to read it in one sitting.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ less cprgm.c
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

bbb. su

The "su" command switches you temporarily to another user account, usually the root account, that possesses full administrative privileges.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo su aryan
aryan@vijayp22-QEMU-Virtual-Machine:/home/vijayp22$
```

ccc. ping

ping, or Packet Internet Groper, is a fundamental network diagnostic tool that lets you to test the connection between the machine you're using and another host or device on a network.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo ping google.com
[sudo] password for vijayp22:
PING google.com (172.217.4.78) 56(84) bytes of data.
64 bytes from ord37s18-in-f14.1e100.net (172.217.4.78): icmp_seq=1 ttl=57 time=13.8 ms
64 bytes from lga15s47-in-f78.1e100.net (172.217.4.78): icmp_seq=2 ttl=57 time=14.1 ms
64 bytes from ord37s18-in-f14.1e100.net (172.217.4.78): icmp_seq=3 ttl=57 time=15.0 ms
64 bytes from ord37s18-in-f14.1e100.net (172.217.4.78): icmp_seq=4 ttl=57 time=14.9 ms
```

ddd. traceroute

traceroute functions as a GPS for network traffic, displaying the precise path your data travels to get to its destination.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ sudo traceroute wikipedia.com
traceroute to wikipedia.com (208.80.154.232), 30 hops max, 60 byte packets
 1  _gateway (192.168.64.1)  0.325 ms  0.294 ms  0.254 ms
 2  puma7-atom.ht.home (192.168.0.1)  4.996 ms  4.951 ms  5.196 ms
 3  10.20.0.1 (10.20.0.1)  17.771 ms  26.278 ms  26.235 ms
 4  216.80.78.91 (216.80.78.91)  26.214 ms  26.197 ms  26.592 ms
 5  hge0-0-0-0.core1.chgo.il.rcn.net (207.172.18.100)  26.583 ms 207.172.17.5 (207.172.17.5)  26.574 ms 207.172.17.17
 6  * hge0-0-0-11.border2.eqnx.il.rcn.net (207.172.19.123)  19.815 ms *
```

eee. date

It displayed the current date and time in the system. We can also change the date and time with this command.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ date -d "2022-01-20" +%j
020
```

fff. wget

wget is a sophisticated command-line application in Linux that allows you to retrieve files from the web without interaction.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ wget http://world.std.com/world/about.html
--2024-01-18 22:17:24--  http://world.std.com/world/about.html
Resolving world.std.com (world.std.com)... 192.74.137.5
Connecting to world.std.com (world.std.com)|192.74.137.5|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://std.com/world/about.html [following]
--2024-01-18 22:17:25--  https://std.com/world/about.html
Resolving std.com (std.com)... 192.74.137.5
Connecting to std.com (std.com)|192.74.137.5|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://theworld.com/world/about.html [following]
--2024-01-18 22:17:25--  https://theworld.com/world/about.html
Resolving theworld.com (theworld.com)... 192.74.137.5
Connecting to theworld.com (theworld.com)|192.74.137.5|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: ‘about.html’
```

ggg. wc

wc, which stands for "word count," is a useful command that counts the total number of lines, words, and characters in data.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ wc cprgm.c
 6 12 79 cprgm.c
```

hhh. clear

The clear command removes all the current text and results from your terminal window, returning only a blank screen and a new prompt to type new commands.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ clear
```

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

3. Question 3

a.

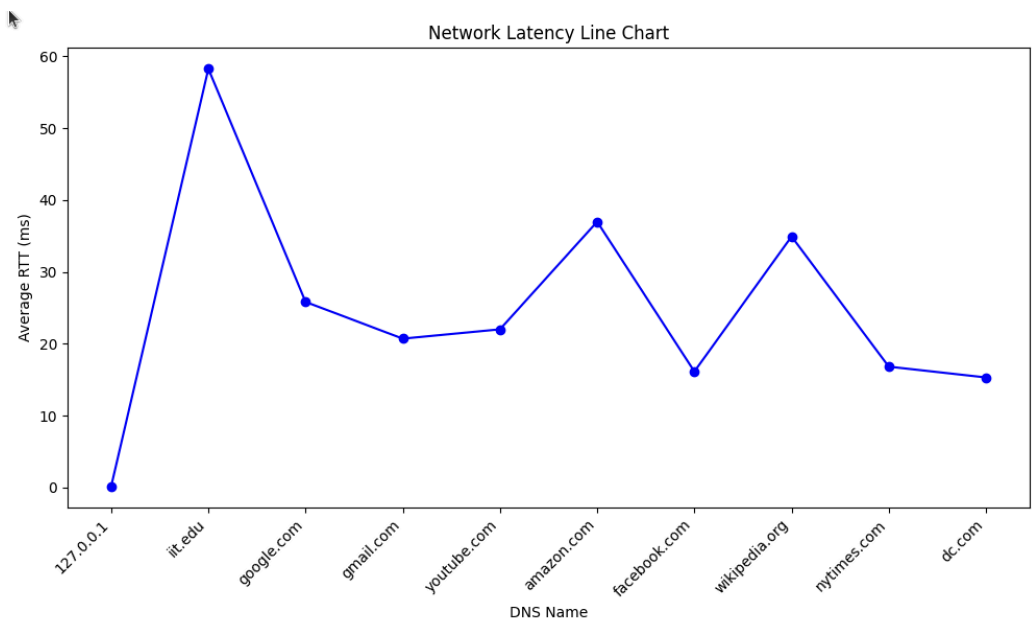
```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat disk-benchmark-background.sh
#!/bin/bash
dd if=/dev/zero of=output1.txt bs=1M count=2000 conv=fdatasync > disk-benchmark-log.txt 2>&1 &
echo "The dd command is running in the background."
sleep 10
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ time bash disk-benchmark-background.sh
The dd command is running in the background.

real    0m10.014s
user    0m0.012s
sys     0m1.543s
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat disk-benchmark-log.txt
2000+0 records in
2000+0 records out
2097152000 bytes (2.1 GB, 2.0 GiB) copied, 1.52378 s, 1.4 GB/s
vijayp22@vijayp22-QEMU-Virtual-Machine:~$
```

b.

```
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat>network-test-machinelist.txt
127.0.0.1
iit.edu
google.com
gmail.com
youtube.com
amazon.com
facebook.com
wikipedia.org
nytimes.com
dc.com
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat network-test.sh
#!/bin/bash
input_file="network-test-machinelist.txt"
output_file="network-test-latency.txt"
while IFS= read -r dns_name; do
    if [ -n "$dns_name" ]; then
        rtt_avg=$(ping -c 3 "$dns_name" | tail -1 | awk -F '/' '{print $5}')
        echo "$dns_name $rtt_avg" >> "$output_file"
    fi
done < "$input_file"
echo "The results are added to network-test-latency.txt."
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ bash network-test.sh
The results are added to network-test-latency.txt.
vijayp22@vijayp22-QEMU-Virtual-Machine:~$ cat network-test-latency.txt
127.0.0.1 0.176
iit.edu 58.268
google.com 25.835
gmail.com 20.722
youtube.com 22.023
amazon.com 36.995
facebook.com 16.148
wikipedia.org 34.939
nytimes.com 16.828
dc.com 15.327
```

c.

```
import matplotlib.pyplot as plt
dns_names = []
avg_rtt_values = []
with open("network-test-latency.txt", 'r') as file:
  for line in file:
            dns_name, avg_rtt = line.split()
            dns_names.append(dns_name)
            avg_rtt_values.append(float(avg_rtt))
plt.figure(figsize=(10, 6))
plt.plot(dns_names, avg_rtt_values, marker='o', linestyle='-', color='b')
plt.xlabel('DNS Name')
plt.ylabel('Average RTT (ms)')
plt.title('Network Latency Line Chart')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



4. Question 4

a.

- <u>Impact of Changing Processors:</u>
  - Enhanced Processing Power: More processors allow the VM to perform more tasks at once and execute them more quickly.
  - Possibility for Overhead: Every virtual processor provides a little overhead to the hypervisor, which may have an influence on performance if assigned in excess.

- <u>Scenario for Minimum Processors:</u>
  - Basic operations: For simple operations such as mail, browsing the internet, or software testing in isolation, one processor can typically suffice, freeing up resources for additional virtual machines.
  - Background Processes: Virtual machines (VMs) that conduct non-critical background processes can also work with few processors.

- Scenario for Maximum processors:
  - CPU-intensive applications: VMs that run demanding applications like editing videos, 3D modeling, or scientific simulations frequently benefit from more processors.
  - High-Traffic Servers: Server virtual machines that handle a large number of concurrent connections may require more processor resources to work optimally.

- <u>Setting to maximum can be problematic:</u>
  - Overcommitment: Assigning the maximum number of processors to many VMs might result in overcommitment, which occurs when the real hardware lacks the capacity to meet all virtual processor requests, thereby causing performance loss across VMs.
  - Scheduler Overhead: Having too many virtual processors might raise scheduling overhead for the hypervisor, reducing performance.
  - Licensing Issues: Certain software licensing models charge according to the amount of virtual processors, leaving maximal allotment more expensive.

b.

Paravirtualization Overview:

- A technique for improving virtual machine efficiency by making the guest operating system aware that it is executing in a virtual environment.
- The guest operating system can then employ specialized drivers and interfaces developed for effective virtualized hardware operations.

Options for VirtualBox:

- None: disables paravirtualization, which may restrict performance but ensures compatibility with older guest OSes.
- Legacy: Uses earlier paravirtualization interfaces to provide compatibility with older virtual machines.
- Minimal: Basic paravirtualization, advertising the virtualized environment, and reporting CPU attributes to the guest operating system.
- Hyper-V: Provides a Microsoft Hyper-V hypervisor interface designed primarily for Windows guests.
- KVM: Displays a Linux KVM hypervisor interface intended for Linux guests.

Top choice for Ubuntu Linux:

KVM is often the best option for Ubuntu Linux guests.

- KVM is optimized for Linux kernels, providing greater compatibility and potential performance gains.
- Shared Kernel: KVM can make use of the host's Linux kernel to reduce overhead and boost performance.
- Virtio Drivers: Ubuntu offers virtio drivers that are highly optimized paravirtualized drivers for disks, networks, and other devices that improve performance.

c.

- Integrated Drive Electronics (IDE):
    - The oldest and slowest interface.
    - The transfer speed is limited to 133 MB/s.
    - Scenario:
        - Older operating systems: Windows 95/98/Me and Linux variants without SATA or NVMe capability.
        - Testing Legacy Software: Requires IDE compatibility.

- Serial ATA (SATA):
    - Successor to IDE; significantly faster and more common.
    - SATA III supports transfer speeds of up to 600 MB/s.
    - Scenario:
        - Most general-purpose virtual machines: Provides a nice blend of performance and compatibility.
        - Storing non-critical data: Where speed is not a major requirement.

- Non-volatile Memory Express (NVMe):
    - The newest and quickest interface, built particularly for SSDs.
    - Transfer speeds can reach 3500 MB/s or more.
    - Scenario:
        - Performance-critical virtual machines: Supporting demanding workloads such as database servers, high-traffic internet servers, and video editing workstations.
        - I/O-intensive applications: Requires fast access to data and low latency.

Feature	IDE	SATA	NVMe
Interface	Parallel	Serial	PCIe
Max Speed	133 MB/s	600 MB/s	3500+ MB/s
Compatibility	Older sys.	Most sys.	Modern sys.
Use Cases	Legacy VMs	General VMs	High-Performance VMs

d.

- Network Address Translation (NAT):
    - The VM shares the host's IP address but has a distinct private IP address.
    - The virtual machine can connect to the internet, but external network devices cannot.

    - Use for Scenario:
        - Private internet access. Web browsing, software upgrades, and testing in isolation.
        - Protecting virtual machines from direct internet threats.

- Bridged Adapter:
    - The virtual machine functions as a distinct device on the physical network.
    - Its personal IP address allows it to communicate directly with other network devices.
    - Use for Scenario:
        - Run network services. Web servers, file servers, and game servers are all accessible to various other devices.
        - Interacting with other network devices. Sharing files and printers.

- Internal Network:
    - VMs are able to interact with one another and with the host, however not with external networks.
    - Makes a standalone virtual network.
    - Use for Scenario:
        - Testing network applications in a controlled environment free of external interference.
        - Establishing a private network: To allow numerous virtual machines to communicate safely.
- Host-only Network:
    - VMs can only communicate with the host and not with one another or external networks.
    - Use for Scenario:
        - Securely transferring data between hosts and virtual machines without network overhead.
        - Testing network applications: These programs just need to communicate with the host system.

e.

- Speed:
    - USB 1.1 is the slowest version, with a maximum transfer speed of 12 Mbps. Ideal for basic devices such as keyboards and mouse.

    - USB 2.0 is a widely used technology with rates of up to 480 Mbps. Suitable for most USB devices, including devices such as printers, scanners, and webcams.
    - USB 3.0 (SuperSpeed USB) is significantly faster than 2.0 and has two variants:
        - USB 3.0 supports transfer speeds of up to 5 Gbps, making it perfect for external hard drives, SSDs, even high-res webcams.
        - USB 3.1 (SuperSpeed+ USB) supports speeds of up to 10 Gbps, making it ideal for demanding activities such as modifying movies and high-bandwidth network connections.

- Compatibility:
    - All of the versions are compatible with prior versions, which means that a USB 3.0 controller is capable of handling devices designed for 1.1 and 2.0. However, the speed is going to be limited to the poorest version of the chain.

- Power Consumption:
    - USB 1.1 and 2.0 devices often use less energy than USB 3.0 devices.

- Cost:
    - USB 1.1 is the lowest option, although USB 3.0 controllers typically are more expensive.

- Answering the doubt of Selecting the Right Controller:
  The ideal option relies on your virtual machine's requirements and what kind of USB devices you're going to be using:
    - Minimal USB requirements (simple tasks and legacy software): Use USB 1.1 since it is low-cost and compatible.
    - General-purpose virtual machine with several USB devices: USB 2.0 provides a decent blend of speed and compatibility with most accessories.
    - High-performance Virtual Machine with intensive USB devices: Consider USB 3.0 or 3.1 for better data transfer rates.

References

1. Textbook- The Linux Command Line by William Shotts: https://sourceforge.net/projects/linuxcommand/
2. https://www.geeksforgeeks.org/linux-commands/
3. https://www.git-tower.com/blog/git-cheat-sheet/
4. https://stackoverflow.com/questions/948008/linux-command-to-list-all-available-commands-and-aliases
5. https://www.datacamp.com/tutorial/how-to-write-bash-script-tutorial
6. https://www.youtube.com/watch?v=sWbUDq4S6Y8