

Assignment 1

Questions:

1. Read Oracle VirtualBox White Paper

I have gone through the paper and registered my email id and got below mail. Below information is given in that paper

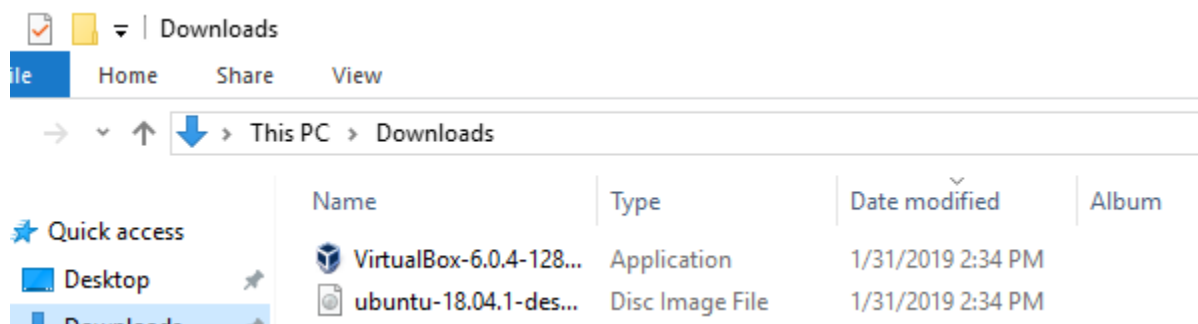
- Information about Oracle VM virtual box.
- Its two components which are VM virtual box and virtual box extension pack: Virtual box is free and extension version requires paid license as it contains few more features.
- Uses of virtual box which includes development and debugging of the various applications which requires different environment, Testing the new changes, Software upgrades, to perform demos and parallel execution of demo applications.
- Sharing information between VM's or from your system is secure as it uses encryption.
- For training purpose also, we can use virtual box.
- Also features about extension pack are also given few are virtual RDP, It allows hosts to use web-cam through their PC's, Encryption of whole disk data is possible.

VirtualBox 5.0 Whitepaper



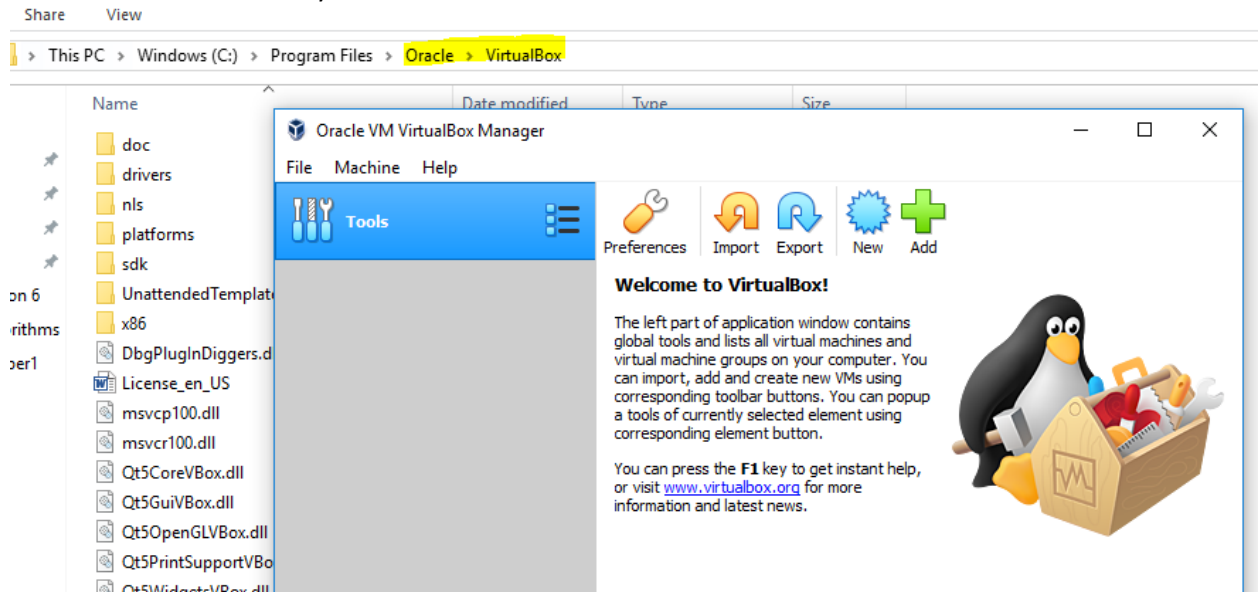
2. Download Oracle VirtualBox 6.0.4

Downloaded virtual box 6.0.4 below is the screenshot for that



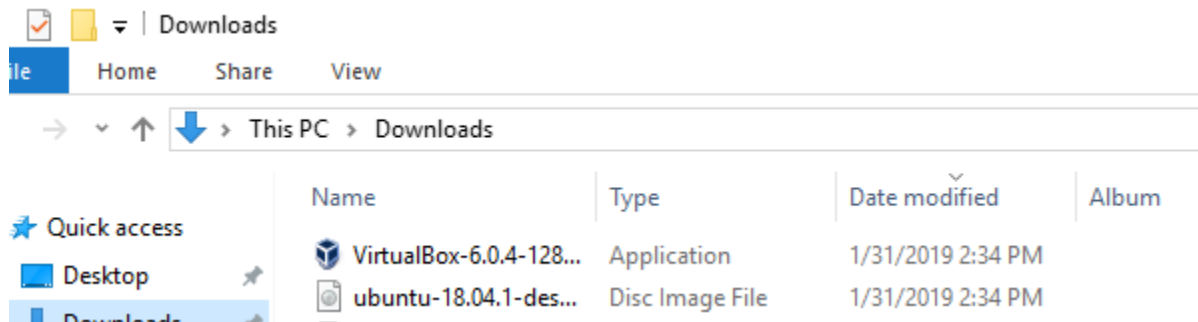
3. Install VirtualBox

Installed Virtual box on my machine below is the screenshot for the same

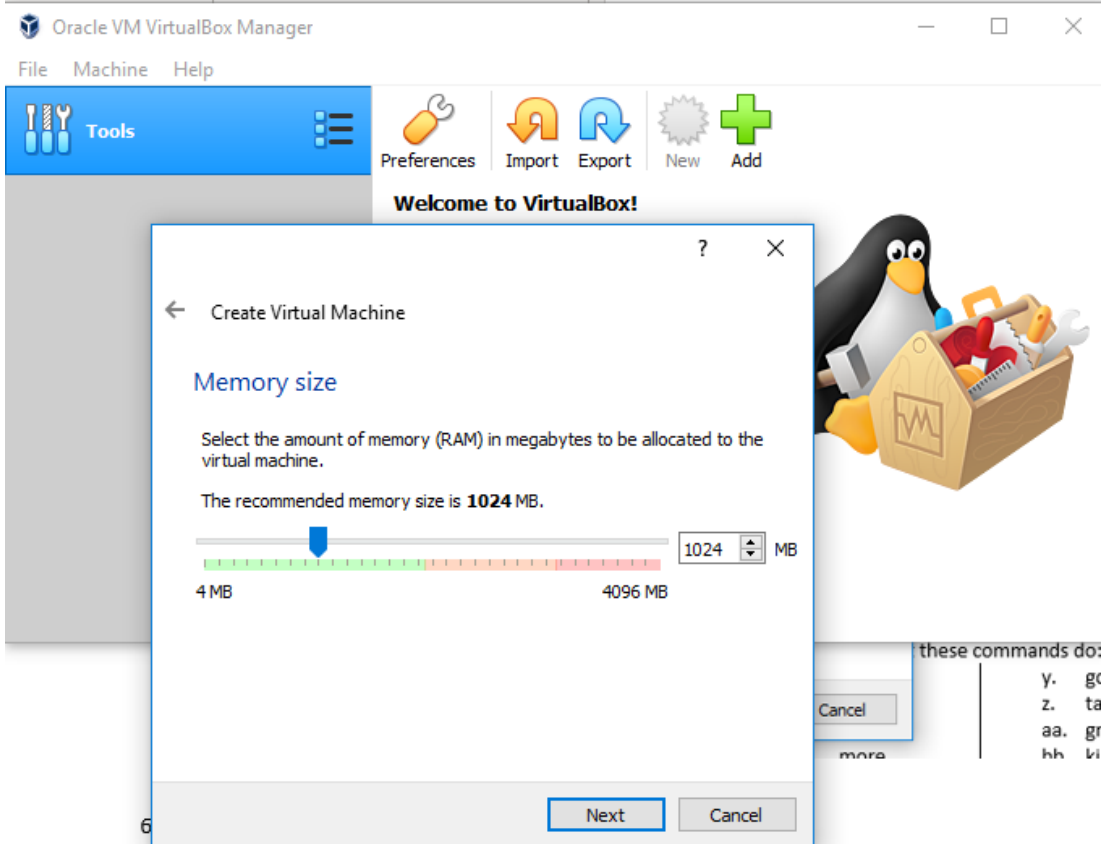
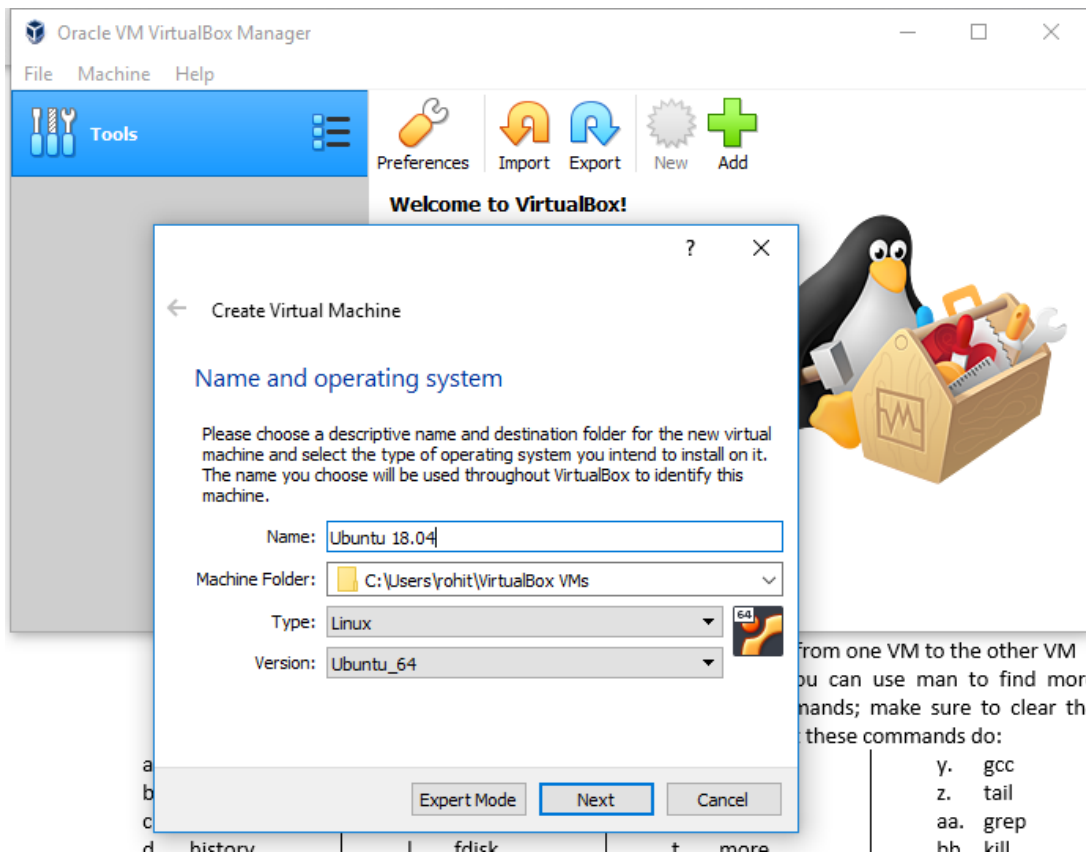


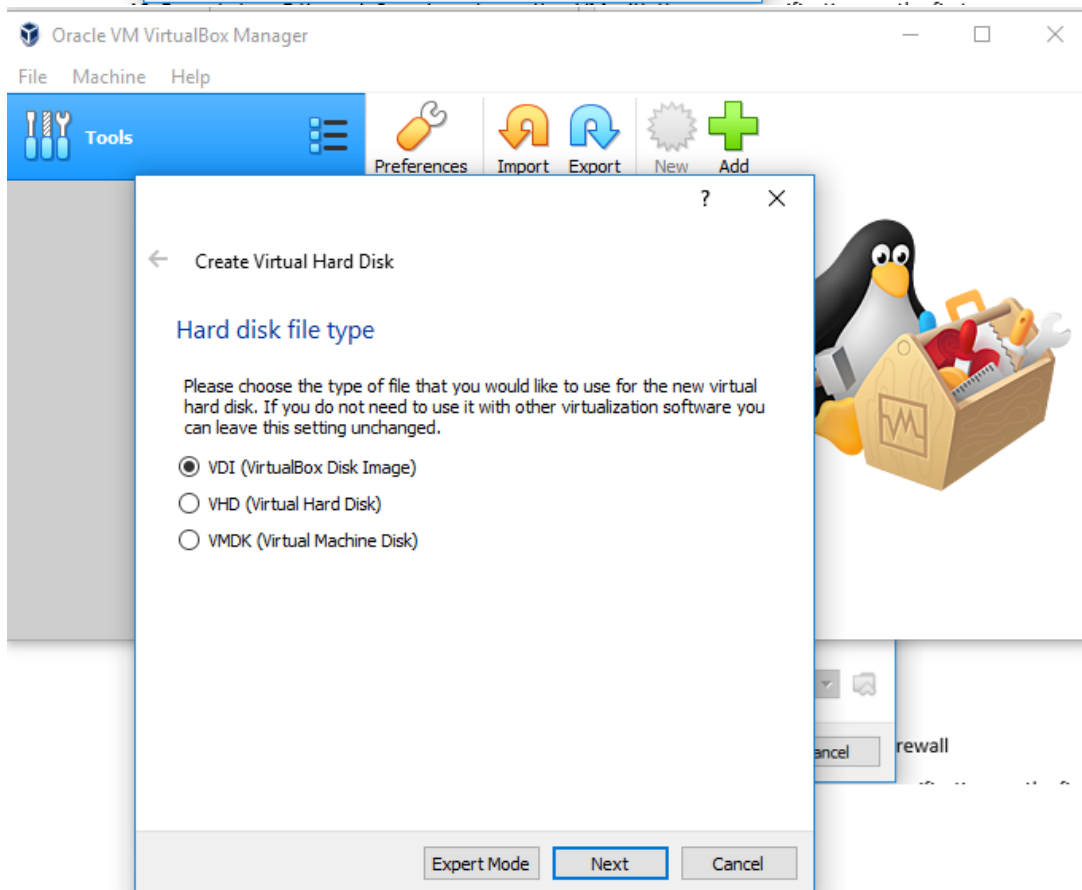
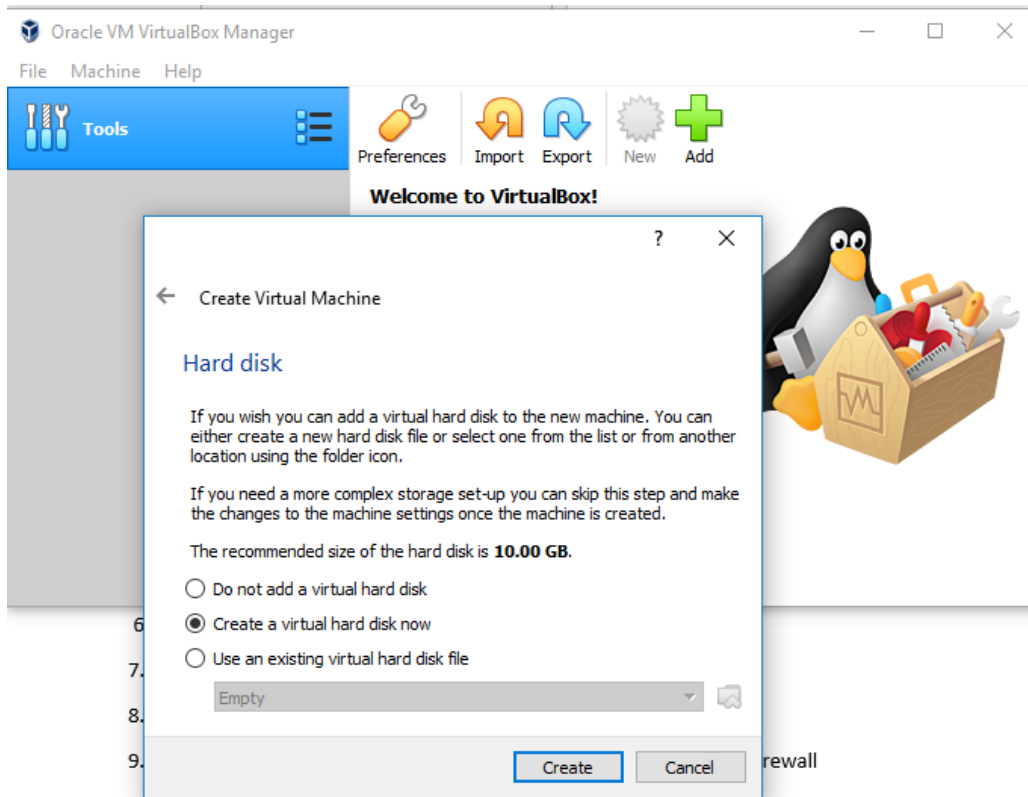
4. Download Ubuntu 18.04.1 Linux ISO image

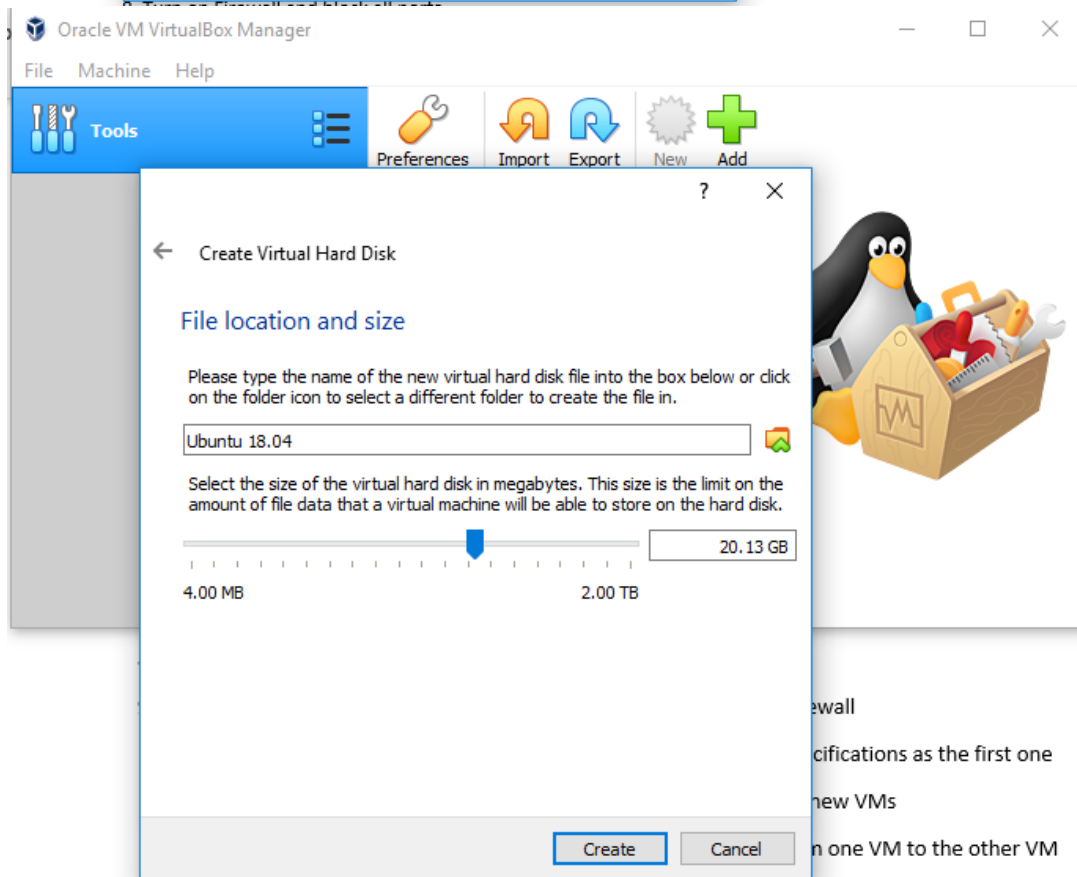
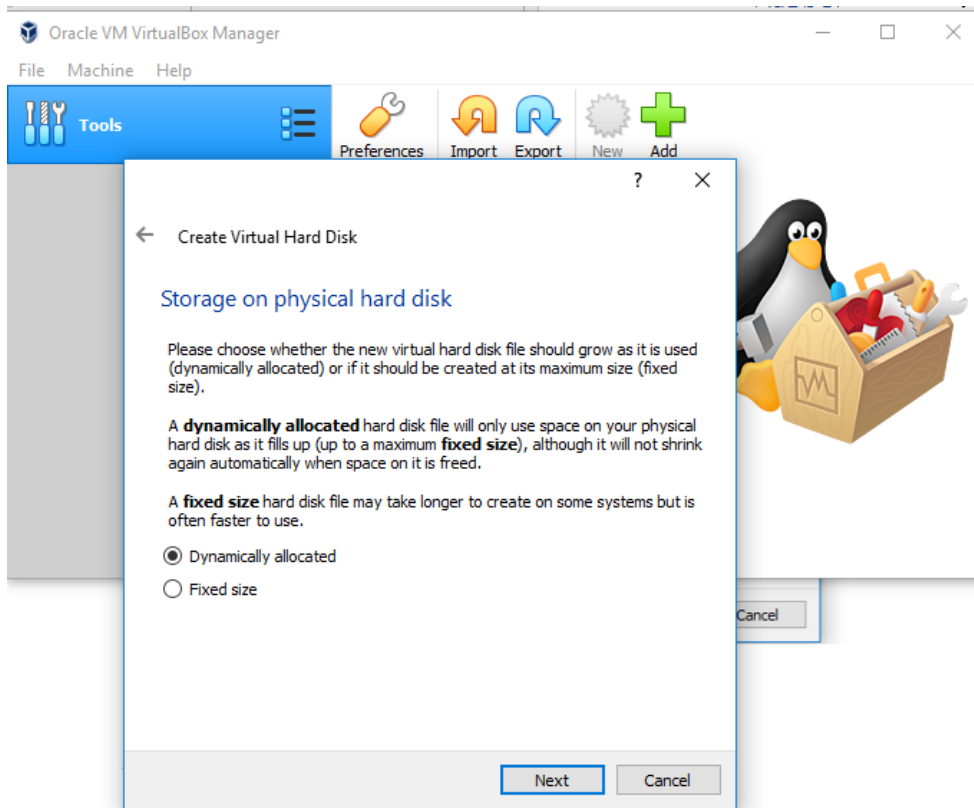
Downloaded Ubuntu 18.04.1 below is the screenshot for the same

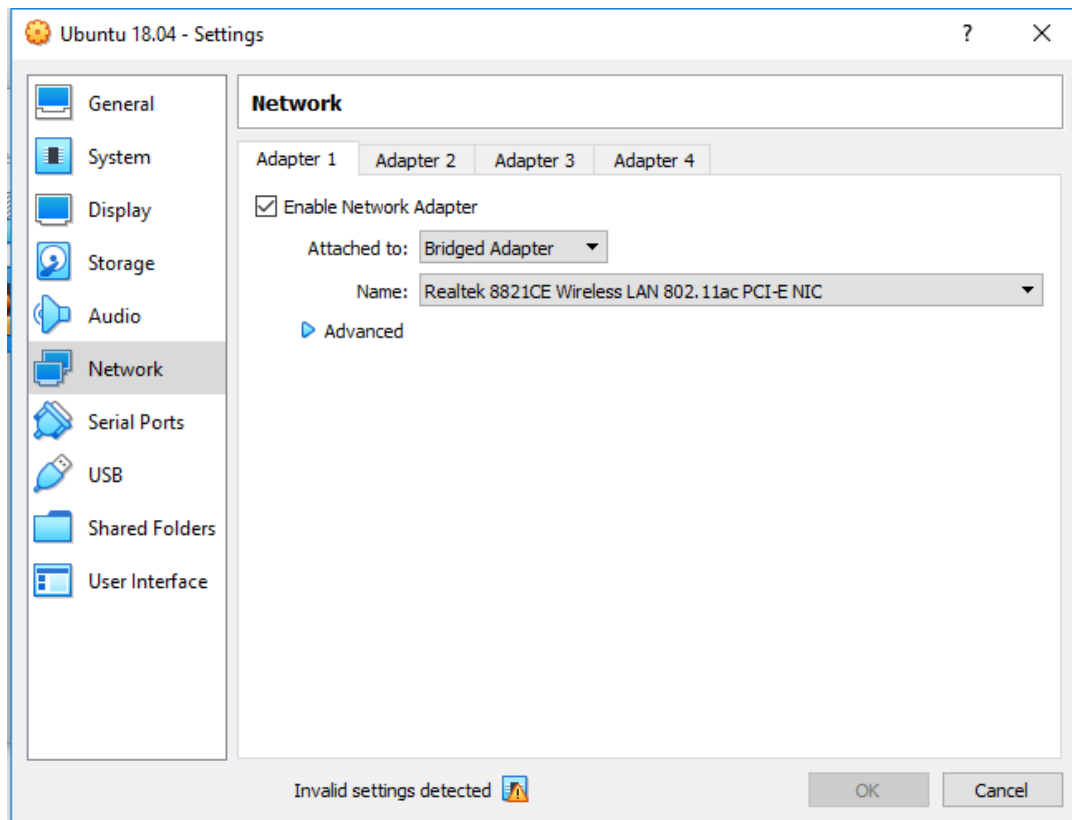
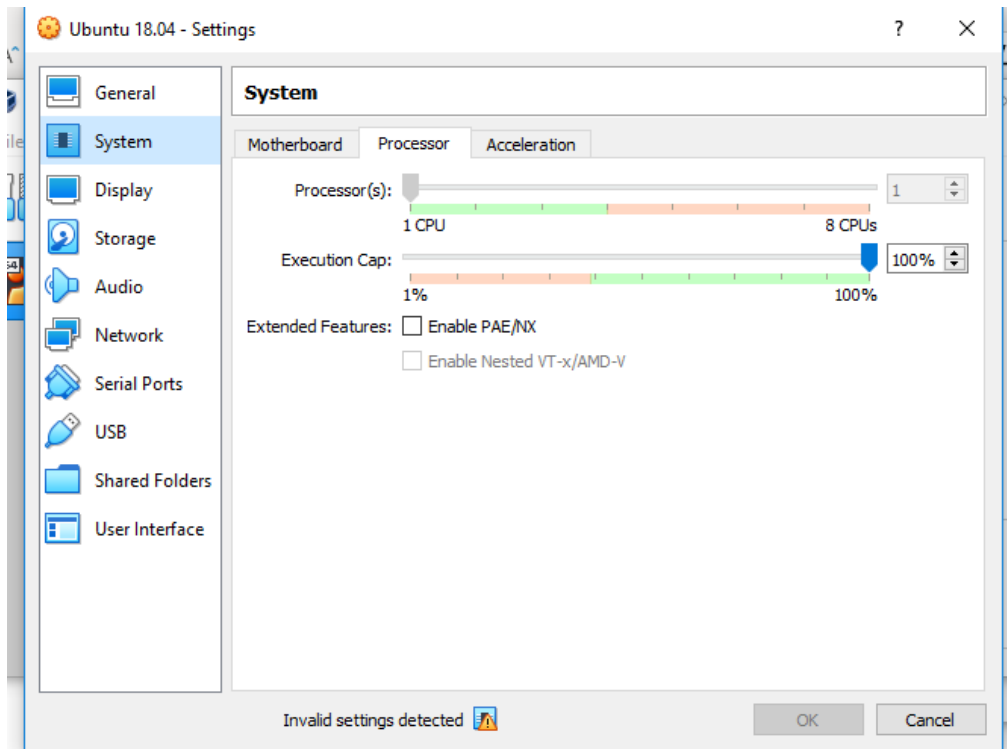


5. Create Virtual Machine (VM), to support Linux, Ubuntu, 64-bit, 1GB RAM, Virtual Disk 20GB, VDI image, dynamically allocated, 1-core, and a network interface (1GbE or WiFi) with Bridged Adapter

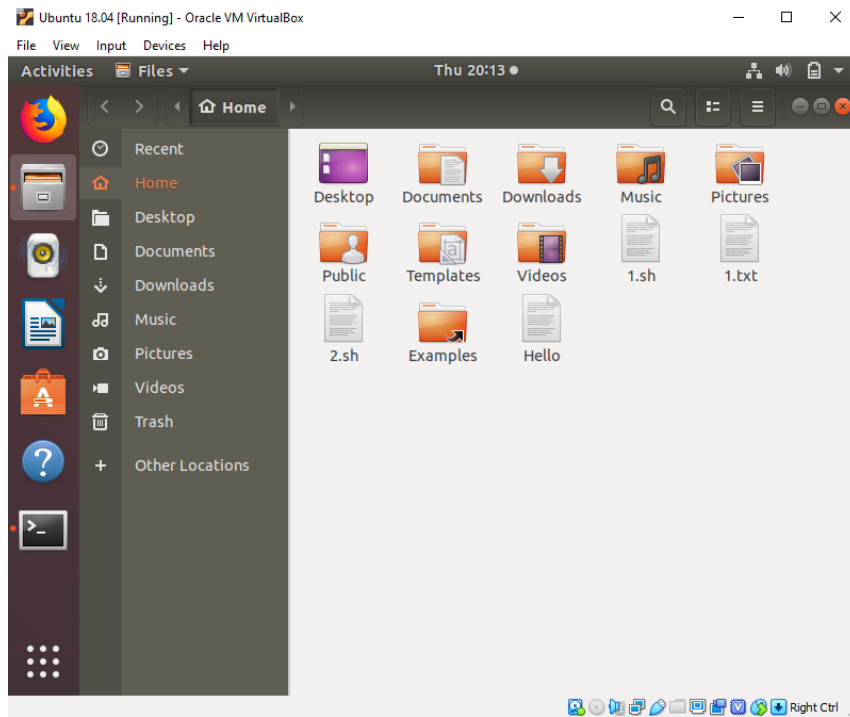




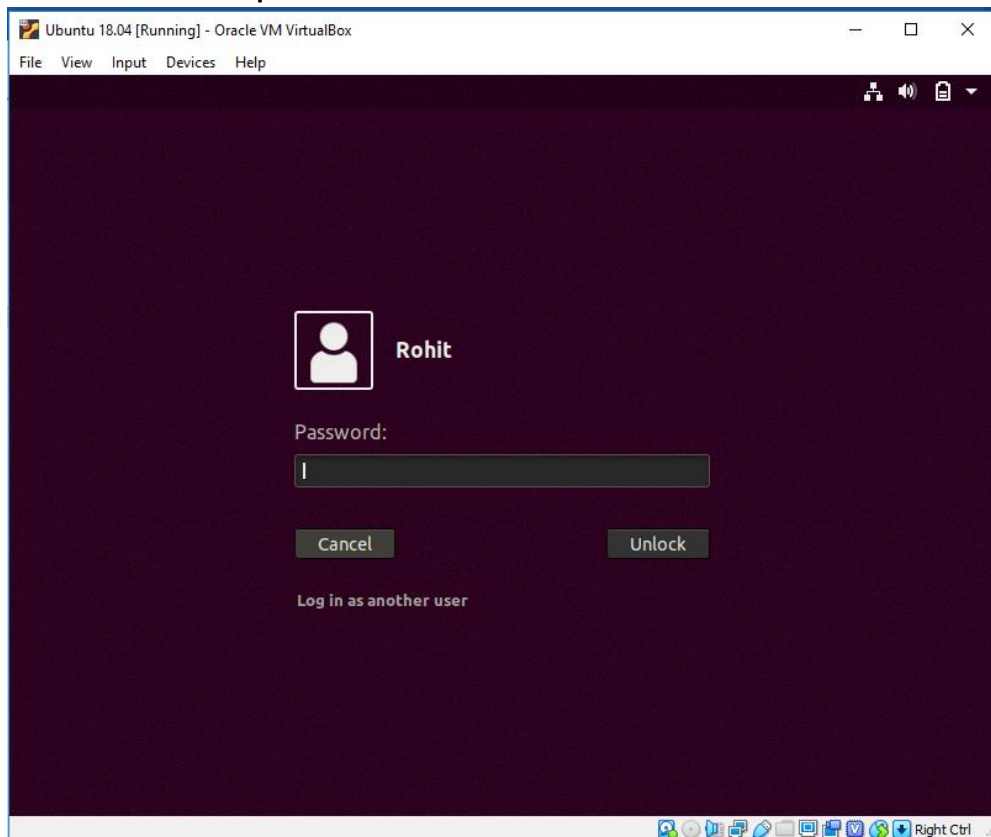




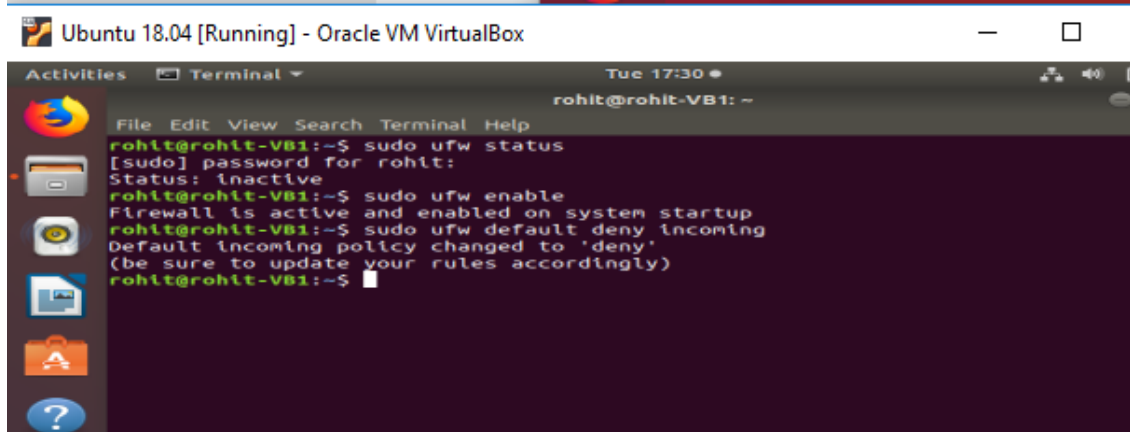
6. Install Linux from the ISO image



7. Create a user id and password

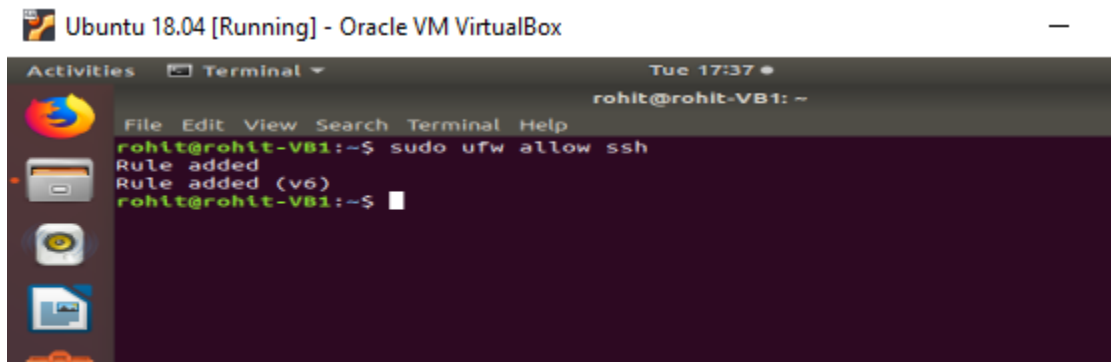


8. Turn on Firewall and block all ports



```
rohit@rohit-VB1:~$ sudo ufw status
[sudo] password for rohit:
Status: inactive
rohit@rohit-VB1:~$ sudo ufw enable
Firewall is active and enabled on system startup
rohit@rohit-VB1:~$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
rohit@rohit-VB1:~$
```

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

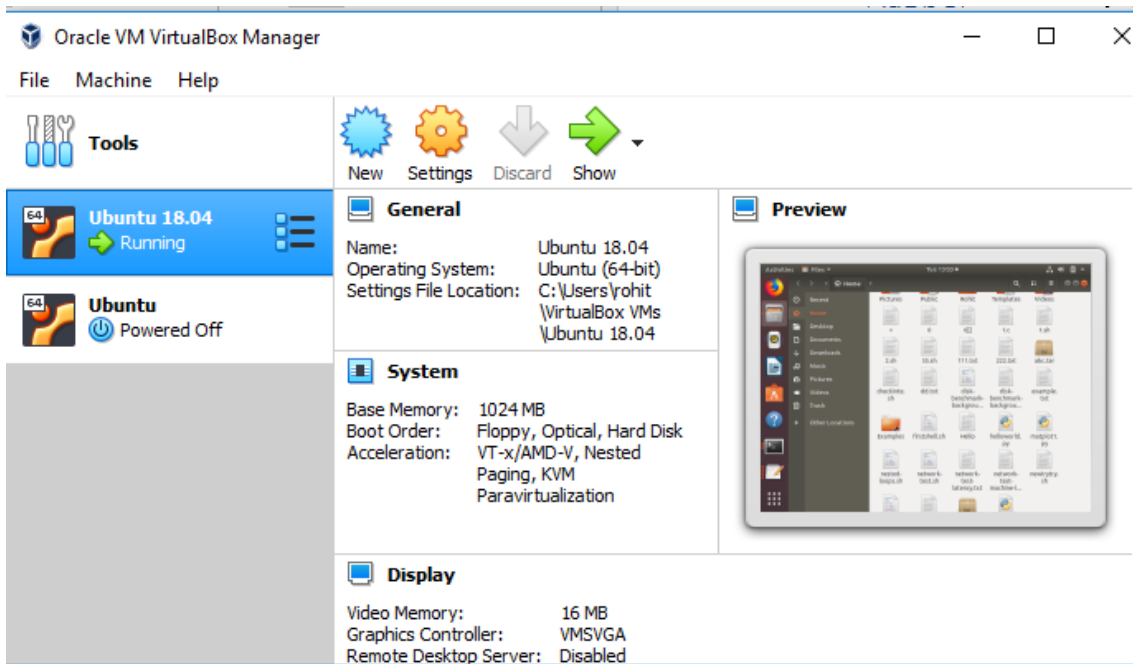


```
rohit@rohit-VB1:~$ sudo ufw allow ssh
Rule added
Rule added (v6)
rohit@rohit-VB1:~$
```

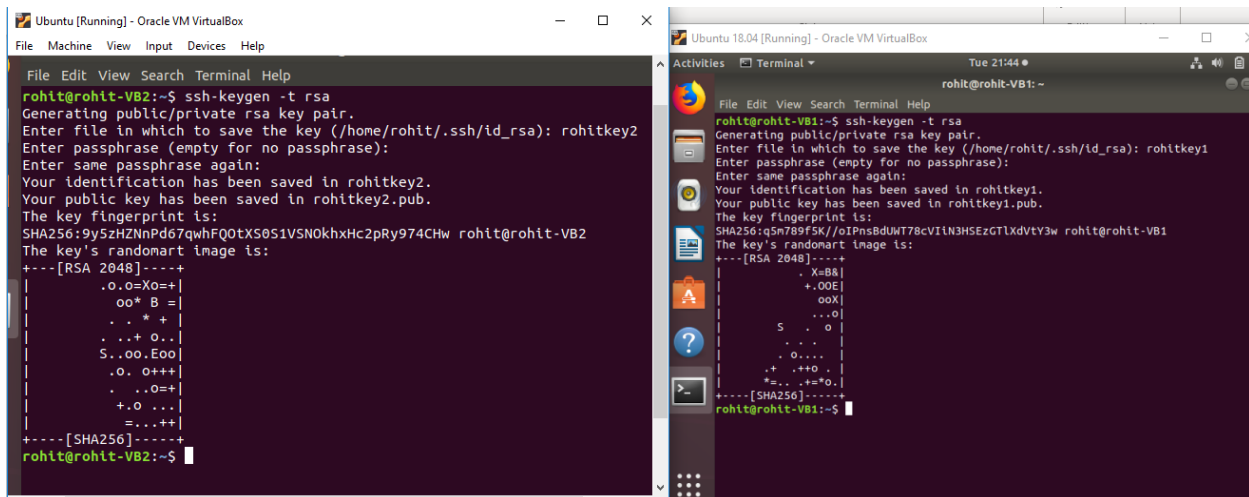


10. Repeat steps 5 through 9, and create another VM with the same specifications as the first one

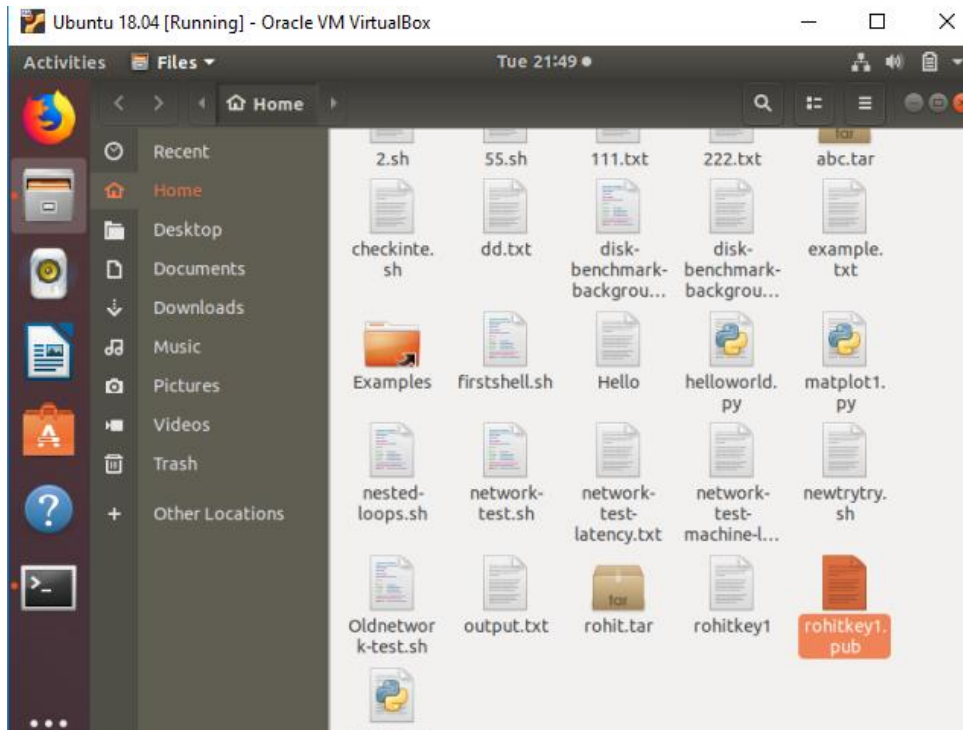
Created another VM with the same details



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

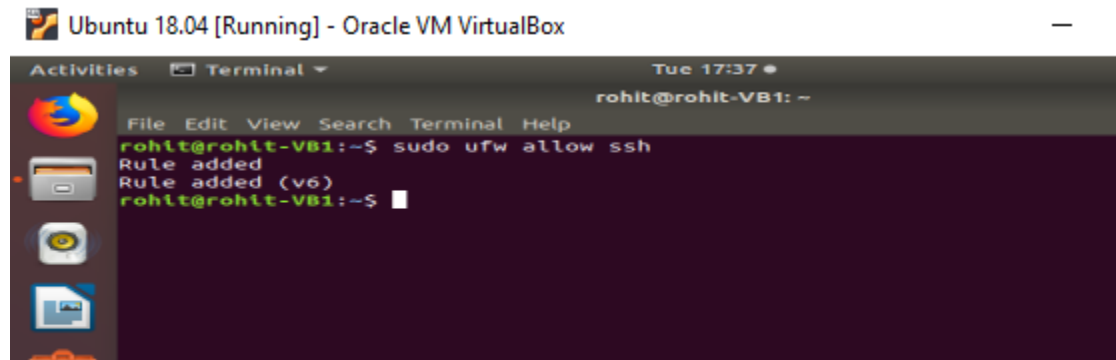


11. Test that you can connect remotely to your VMs with your keys, from one VM to the other VM

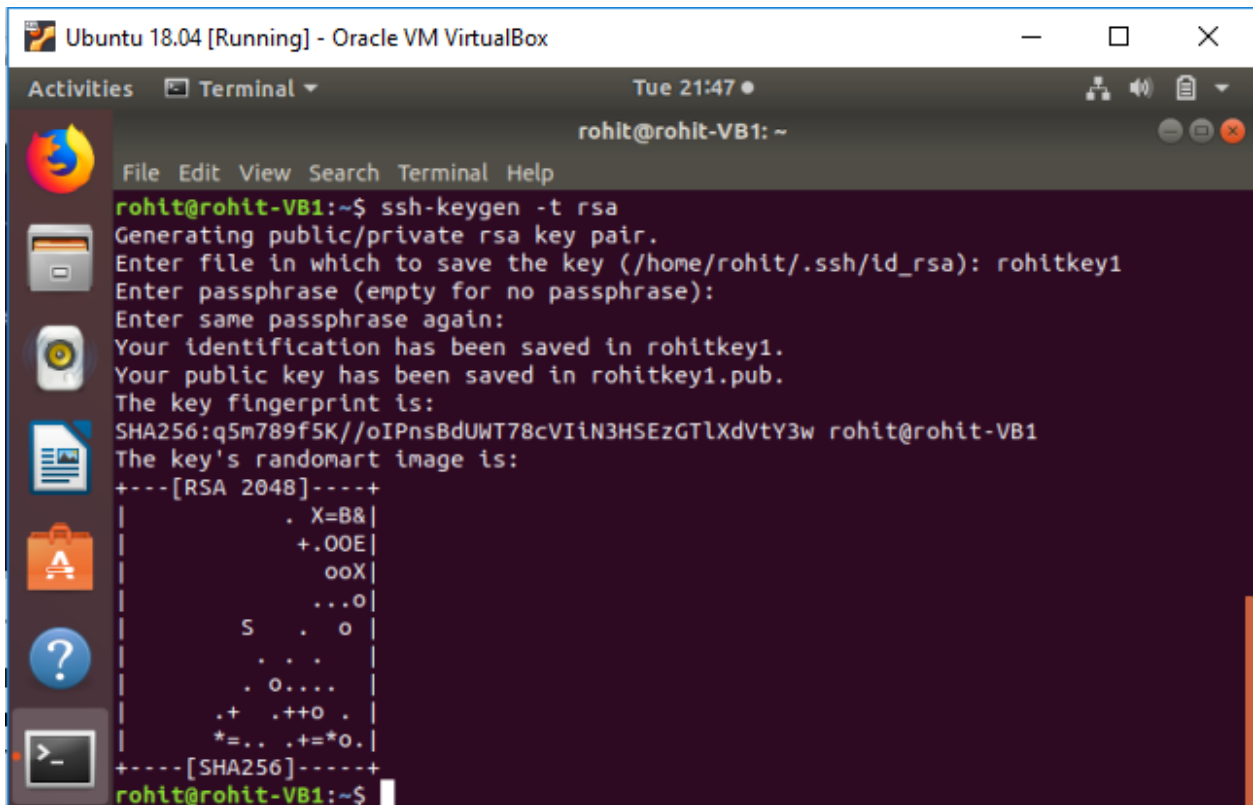


13. Show an example of using the following commands (hint: you can use man to find more information about each one); take screen shots of your commands; make sure to clear the screen between each command; explain in your own words what these commands do:

- a. **Ssh:** Given command is used to establish a secure network between two machines.



- b. **Ssh-keygen:** It is used for key generation in public key authentication protocol



The screenshot shows a terminal window titled "Ubuntu 18.04 [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ ssh-keygen -t rsa  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/rohit/.ssh/id_rsa): rohitkey1  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in rohitkey1.  
Your public key has been saved in rohitkey1.pub.  
The key fingerprint is:  
SHA256:q5m789f5K//oIPnsBdUWT78cVIiN3HSEzGTLXdVtY3w rohit@rohit-VB1  
The key's randomart image is:  
+---[RSA 2048]---+  
| . X=B&|  
| +.00E|  
| ooX|  
| ...O|  
| S . O|  
| . . .|  
| . 0....|  
| .+ .++O .|  
| *=.. .+=*O.|  
+-----[SHA256]-----+  
rohit@rohit-VB1:~$
```

- c. **Scp:** It is used to copy files securely from one host to another
- d. **History:** It provides list of commands which we used in past

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ history
 1  ls
 2  touch 1.txt
 3  cat 1.txt
 4  cat 1.txt >> "Hello"
 5  touch 1.txt
 6  rm 1.txt
 7  ls
 8  touch 1.txt
 9  1.txt >> "Hello"
10  cat 1.txt>>"Hello"
11  cat 1.txt
12  touch 1.sh
13  man file
14  touch 2.sh
15  clear
16  man ssh
17  man scp
18  history
19  clear
20  history
rohit@rohit-VB1:~$
```

e. **Sudo**

Sudo gives higher privileges while running the commands, after running it, it will ask for the password

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ sudo apt-install
[sudo] password for rohit:
```

- f. **Ip**: It is used to get information about ip utility, Ip -v can be used to get version of ip utility.

```
rohit@rohit-VB1: ~/Rohit
File Edit View Search Terminal Help
rohit@rohit-VB1:~/Rohit$ ip -V
ip utility, iproute2-ss180129
rohit@rohit-VB1:~/Rohit$
```

- g. **Touch:** Touch is used for creating files through terminal

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop Pictures Videos
1.txt Documents Hello Public
2.sh Downloads Music Templates
rohit@rohit-VB1:~$ touch NewFile.txt
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop NewFile.txt Templates
1.txt Documents Hello Pictures Videos
2.sh Downloads Music Public
rohit@rohit-VB1:~$
```

- h. **Ls:** It is used to print all content present in given directory.

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop NewFile.txt Templates
1.txt Documents Hello Pictures Videos
2.sh Downloads Music Public
rohit@rohit-VB1:~$
```

- i. **Mkdir :** It is used to create directory.

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop NewFile.txt Templates
1.txt Documents Hello Pictures Videos
2.sh Downloads Music Public
rohit@rohit-VB1:~$ mkdir Rohit
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop NewFile.txt Rohit
1.txt Documents Hello Pictures Templates
2.sh Downloads Music Public Videos
rohit@rohit-VB1:~$
```

- j. **Cd** : It is used to change directory.

```
rohit@rohit-VB1: ~/Rohit
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls
1.sh Desktop examples.desktop NewFile.txt Rohit
1.txt Documents Hello Pictures Templates
2.sh Downloads Music Public Videos
rohit@rohit-VB1:~$ cd Rohit
rohit@rohit-VB1:~/Rohit$
```

- k. **Dd**: It is used to copy or convert the files, Even it is used for backup, restore of taking image of hard drive.

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ cat cp2.txt
rohit@rohit-VB1:~$ dd if=cp1.txt of=cp2.txt
0+1 records in
0+1 records out
24 bytes copied, 0.0143146 s, 1.7 kB/s
rohit@rohit-VB1:~$ cat cp2.txt
Copying to CP2 from CP1
rohit@rohit-VB1:~$
```

- l. **Fdisk**: fdisk is used for disk partitioning or getting details about disks in your system


```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ sudo fdisk -l  
Disk /dev/loop0: 2.3 MiB, 2355200 bytes, 4600 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: 86.9 MiB, 91099136 bytes, 177928 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 is nothing but advanced package tool which is used for installation and uninstallation of software's

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ sudo apt install  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
0 upgraded, 0 newly installed, 0 to remove and 263 not upgr  
aded.  
  
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ sudo fdisk -l  
Disk /dev/loop0: 2.3 MiB, 2355200 bytes, 4600 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: 86.9 MiB, 91099136 bytes, 177928 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
```

- n. **Vi** :

It is used to insert update text or files in console. If file is not present then it will create that file and vi I user can write into the file.


```
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls | grep '.tar'
abc.tar
rohit.tar
rohit@rohit-VB1:~$
```

- q. **Rm:** Command rm is used to remove or delete the files

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ touch tryrm.txt
rohit@rohit-VB1:~$ ls
'='
0
0]]
1.sh
1.txt
2.sh
55.sh
checkinte.sh
Desktop
Documents
Downloads
examples.desktop
example.txt
firstshell.sh
Hello
helloworld.py
matplot1.py
Music
nested-loops.sh
network-test-latency.txt
network-test-machine-list.txt
network-test.sh
NewFile.txt
Oldnetwork-test.sh
output.txt
Pictures
Public
Rohit
Templates
tryrm.txt
Videos
WRNGmatplot2.py
rohit@rohit-VB1:~$ rm tryrm.txt
rohit@rohit-VB1:~$ ls | grep 'try'
rohit@rohit-VB1:~$
```

- r. **Cat:** cat is used to print content of the file in output window

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ cat example.txt
abc 500
dhf 634
wqs 721
gls 396
pqr 400
wnf 550
qmd 300
aws 200
acz 250
rohit@rohit-VB1:~$
```

- s. **Bash :** Command bash is used to run shell scripts

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ bash firstshell.sh
Please enters values of i, j & k seperated by space 2 3
0
0
0
```

- t. **More:** We can say it is manual scroll, Without more system prints every output on your screen and user has to scroll the screen to see the outputs

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
=  
0  
0]]  
111.txt  
1.c  
1.sh  
222.txt  
2.sh  
55.sh  
abc.tar  
checkinte.sh  
cp1.txt  
cp2.txt  
dd.txt  
Desktop  
disk-benchmark-background-log.txt  
disk-benchmark-background.sh  
Documents  
Downloads  
examples.desktop  
example.txt  
firstshell.sh  
Hello  
helloworld.py  
matplotlib.py  
Music  
nested-loops.sh  
--More--
```

- u. **Watch:** Given command will display the output of given command after given interval and it will run into a loop

```
File Edit View Search Terminal Help  
Every 2.0s: ls rohit-VB1: Mon Feb 4 18:36:  
=  
0  
0]]  
1.sh  
2.sh  
55.sh  
abc.tar  
checkinte.sh  
Desktop  
Documents  
Downloads  
examples.desktop  
example.txt  
firstshell.sh  
Hello  
helloworld.py  
matplotlib.py
```

- v. **Ps:** It is used to get information about currently running processes and their process ids

```

rohith@rohith-VB1: ~/Rohit
File Edit View Search Terminal Help
rohith@rohith-VB1:~/Rohit$ ps
  PID TTY          TIME CMD
 3285 pts/0        00:00:00 bash
 3513 pts/0        00:00:00 ps
rohith@rohith-VB1:~/Rohit$

```

w. Top :

It is used to display information about CPU processes

```

rohith@rohith-VB1: ~
File Edit View Search Terminal Help
top - 18:39:53 up 4:40, 1 user, load average: 0.01, 0.04, 0.07
Tasks: 201 total, 1 running, 168 sleeping, 2 stopped, 0 zombie
%Cpu(s): 6.0 us, 2.0 sy, 0.0 ni, 92.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1009112 total, 72100 free, 662392 used, 274620 buff/cache
KiB Swap: 976320 total, 656064 free, 320256 used. 197844 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 1529 rohit     20   0 2983020 177432 27572 S   4.3  17.6   8:55.88 gnome-s
 1385 rohit     20   0 361720   35012 11432 S   2.0   3.5   1:41.73 Xorg
 3932 rohit     20   0 799616   35592 25852 S   1.0   3.5   0:04.08 gnome-t
 4153 rohit     20   0   51188   4036   3352 R   1.0   0.4   0:00.11 top
    1 root      20   0 225320    5024  3208 S   0.0   0.5   0:03.23 systemd
    2 root      20   0      0      0      0 S   0.0   0.0   0:00.00 kthread
    4 root      0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker
    6 root      0 -20      0      0      0 I   0.0   0.0   0:00.00 mm_perc
    7 root      20   0      0      0      0 S   0.0   0.0   0:00.54 ksoftir
    8 root      20   0      0      0      0 I   0.0   0.0   0:02.40 rcu_sch
    9 root      20   0      0      0      0 I   0.0   0.0   0:00.00 rcu_bh

```

x. Htop :

It's an user friendly process viewer, Which shows process details in interactive way

```

rohith@rohith-VB1: ~
File Edit View Search Terminal Help

CPU[|||||5.2%] Tasks: 136, 294 thr: 1 running
Mem[|||||705M/985M] Load average: 0.84 0.36 0.15
Swp[|||||271M/953M] Uptime: 01:40:50

  PID USER      PRI  NI    VIRT    RES    SHR S  CPU%  MEM%     TIME+  Command
 2549 rohit     20   0 40600   4432   3628 R   2.6   0.4   0:00.25 htop
 1148 rohit     20   0 2896M   168M 24244 S   1.3  17.1   1:11.53 /usr/bin/gnome-
 1018 rohit     20   0   352M   31220 10940 S   0.7   3.1   0:09.70 /usr/lib/xorg/X
 1679 rohit     20   0   708M  17892   8884 S   0.7   1.8   0:02.30 /usr/lib/gnome-
 1024 rohit     20   0   352M   31220 10940 S   0.0   3.1   0:01.40 /usr/lib/xorg/X
   643 messagebu 20   0   51628   3268   1476 S   0.0   0.3   0:03.71 /usr/bin/dbus-d
   811 gdm        20   0 2843M   41212 12336 S   0.0   4.1   0:01.21 /usr/bin/gnome-
    1 root      20   0   220M   3488   1740 S   0.0   0.3   0:02.21 /sbin/init spla
  225 root      19  -1  95176   5488  4956 S   0.0   0.5   0:00.97 /lib/systemd/sy
  244 root      20   0 47460   1012    984 S   0.0   0.1   0:00.53 /lib/systemd/sy
  311 systemd-r 20   0  70728   1924   1680 S   0.0   0.2   0:01.00 /lib/systemd/sy

```

y. Gcc

It is used to execute c programs

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ sudo gcc 1.c  
[sudo] password for rohit:  
/usr/lib/gcc/x86_64-linux-gnu/7/../../../../x86_64-linux-gnu  
_start':  
(.text+0x20): undefined reference to `main'  
collect2: error: ld returned 1 exit status  
rohit@rohit-VB1:~$
```

z. Tail

Tail is used to get last rows or bottom rows from file

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ man vi  
rohit@rohit-VB1:~$ cat example.txt  
abc 500  
dhf 634  
wqs 721  
gls 396  
pqr 400  
wnf 550  
qmd 300  
aws 200  
acz 250  
rohit@rohit-VB1:~$
```

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ cat example.txt | tail -2  
aws 200  
acz 250  
rohit@rohit-VB1:~$
```

aa. Grep

Grep is used for pattern matching


```

rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ls
'='                                matplotlib.py
0                                  Music
0]]                                nested-loops.sh
1.sh                               network-test-latency.txt
1.txt                             network-test-machine-list.txt
2.sh                               network-test.sh
55.sh                             NewFile.txt
abc.tar                           NewTets
checkinte.sh                      Oldnetwork-test.sh
Desktop                           output.txt
Documents                         Pictures
Downloads                         Public
examples.desktop                 Rohit
example.txt                      Templates
firstshell.sh                    Videos
Hello                            WRNGmatplotlib2.py
helloworld.py
rohit@rohit-VB1:~$ ls | grep 'loops'
nested-loops.sh
rohit@rohit-VB1:~$

```

bb. Kill : It is used to get information about the signals and also to kill the specific signals.

```

rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ kill -l
1) SIGHUP          2) SIGINT          3) SIGQUIT         4) SIGILL          5)
6) SIGABRT         7) SIGBUS         8) SIGFPE          9) SIGKILL          10)
11) SIGSEGV        12) SIGUSR2       13) SIGPIPE        14) SIGALRM          15)
16) SIGSTKFLT      17) SIGCHLD       18) SIGCONT        19) SIGSTOP          20)
21) SIGTTIN        22) SIGTTOU       23) SIGURG         24) SIGXCPU          25)
26) SIGVTALRM      27) SIGPROF       28) SIGWINCH        29) SIGIO            30)
31) SIGSYS         34) SIGRTMIN       35) SIGRTMIN+1      36) SIGRTMIN+2      37)
38) SIGRTMIN+4     39) SIGRTMIN+5    40) SIGRTMIN+6      41) SIGRTMIN+7      42)
43) SIGRTMIN+9     44) SIGRTMIN+10   45) SIGRTMIN+11     46) SIGRTMIN+12     47)
48) SIGRTMIN+14    49) SIGRTMIN+15   50) SIGRTMAX-14     51) SIGRTMAX-13     52)
53) SIGRTMAX-11    54) SIGRTMAX-10   55) SIGRTMAX-9      56) SIGRTMAX-8      57)
58) SIGRTMAX-6     59) SIGRTMAX-5    60) SIGRTMAX-4      61) SIGRTMAX-3      62)
63) SIGRTMAX-1     64) SIGRTMAX
rohit@rohit-VB1:~$

```

cc. Killall : It is used to kill the processes by name, even it is possible to kill all processes using this command, Or user can kill processes which follows specific regular expressions

```

rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ killall -l
HUP INT QUIT ILL TRAP ABRT IOT BUS FPE KILL USR1 SEGV USR2 PIPE ALRM T
STKFLT CHLD CONT STOP TSTP TTIN TTOU URG XCPU XFSZ VTALRM PROF WINCH I
UNUSED
rohit@rohit-VB1:~$

```

dd. Du : It shows the file space usage

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

rohit@rohit-VB1:~$ du
4      ./Videos
84     ./local/lib/python2.7/site-packages/numpy/tests
28     ./local/lib/python2.7/site-packages/numpy/lib/tests/data
1416   ./local/lib/python2.7/site-packages/numpy/lib/tests
3128   ./local/lib/python2.7/site-packages/numpy/lib
160    ./local/lib/python2.7/site-packages/numpy/random/tests
3324   ./local/lib/python2.7/site-packages/numpy/random
52     ./local/lib/python2.7/site-packages/numpy/fft/tests
356    ./local/lib/python2.7/site-packages/numpy/fft
```

ee. Df: It is same as that of DU, which is used to get disk space usage.

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

rohit@rohit-VB1:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            474312         0   474312   0% /dev
tmpfs           100912    1552    99360   2% /run
/dev/sda1       20643288 7552144 12019480 39% /
tmpfs           504556         0   504556   0% /dev/shm
tmpfs           5120         4     5116   1% /run/lock
tmpfs           504556         0   504556   0% /sys/fs/cgroup
/dev/loop0      2304        2304         0 100% /snap/gnome-calculator/260
/dev/loop3      93184     93184         0 100% /snap/core/6350
/dev/loop4      3840        3840         0 100% /snap/gnome-system-monitor/
/dev/loop5     144128    144128         0 100% /snap/gnome-3-26-1604/74
/dev/loop6     14848     14848         0 100% /snap/gnome-logs/37
/dev/loop7     14976     14976         0 100% /snap/gnome-logs/45
/dev/loop9     13312     13312         0 100% /snap/gnome-characters/139
/dev/loop13    35456     35456         0 100% /snap/gnome-characters/139
```

ff. Screen: Multiply given terminal for multiple processes

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

GNU Screen version 4.06.02 (GNU) 23-Oct-17

Copyright (c) 2015-2017 Juergen Weigert, Alexander Naumov, Amade
Copyright (c) 2010-2014 Juergen Weigert, Sadrul Habib Chowdhury
Copyright (c) 2008-2009 Juergen Weigert, Michael Schroeder, Mica
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 mo
the terms of the GNU General Public License as published by the
Foundation; either version 3, or (at your option) any later vers
This program is distributed in the hope that it will be useful
```

gg. Vim :

Used to print or edit content of file using console

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
abc 500
dhf 634
wqs 721
gls 396
pqr 400
wnf 550
qmd 300
aws 200
acz 250
~
~
```

hh. **Chmod** : Chmod is used to give permissions to file, permissions like read write and execute access. There are three users to which we assign permissions and users are like user, group, other. 777 means give access to all the users and all the accesses.

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ./newtrytry.sh
bash: ./newtrytry.sh: Permission denied
rohit@rohit-VB1:~$ chmod 777 newtrytry.sh
rohit@rohit-VB1:~$ ./newtrytry.sh
rohit@rohit-VB1:~$
```

ii. **Chown** : It is used to change file owner or group

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ chown root /bin
chown: changing ownership of '/bin': Operation not permitted
rohit@rohit-VB1:~$
```

jj. **Useradd** : Used to add new user to existing system

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ sudo useradd aayush
[sudo] password for rohit:
rohit@rohit-VB1:~$
```

kk. **Mv** : It is used to move the file from one folder to another


```
rohit@rohit-VB1: ~/Public
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ sudo mv NewFile.txt /Public
rohit@rohit-VB1:~$ cd Public
rohit@rohit-VB1:~/Public$ ls
NewFile.txt
rohit@rohit-VB1:~/Public$
```

- II. **Man** : It is used to get user manual information for given command, See below output for man ls command

```
rohit@rohit-VB1: ~/Public
File Edit View Search Terminal Help
LS(1) User Commands
NAME
    ls - list directory contents
SYNOPSIS
    ls [OPTION]... [FILE]...
DESCRIPTION
    List information about the FILES (the current directory
    Sort entries alphabetically if none of -cftuvSUX nor --sort
    fied.
```

- mm. **Locate** : It is used to find the files which matches given pattern

```
rohit@rohit-VB1: ~/Public
File Edit View Search Terminal Help
locate: no pattern to search for specified
rohit@rohit-VB1:~/Public$ locate -A '.txt'
/boot/grub/gfxblacklist.txt
/etc/X11/rgb.txt
/etc/brltty/Input/ba/all.txt
/etc/brltty/Input/bd/all.txt
/etc/brltty/Input/bl/18.txt
/etc/brltty/Input/bl/40_m20_m40.txt
/etc/brltty/Input/ec/all.txt
/etc/brltty/Input/ec/spanish.txt
/etc/brltty/Input/en/all.txt
```

- nn. **Find**: Used to search a file in given folder

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ find example.txt
example.txt
rohit@rohit-VB1:~$
```

- oo. **Sed**: it is used to edit or transform the text in console

- pp. **Awk**: Used for pattern matching


```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ echo "97" | awk '/regex|^[0-9]+$/{print "I"}'  
I  
rohit@rohit-VB1:~$
```

qq. Diff: It is used to compare files line by line

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ diff 111.txt 222.txt  
1c1  
< hi Its me rohit  
---  
> Hi Its me Prasad  
rohit@rohit-VB1:~$
```

rr. Sort

Sort is used to sort records according to ascending or descending order.

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ echo "H e l l o" | tr ' ' '\n' | sort  
e  
H  
l  
l  
o  
rohit@rohit-VB1:~$
```

ss. Export: Used to export the value of variables available to child processes or other resources

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ echo "Hi!" | export  
declare -x CLUTTER_IM_MODULE="xim"  
declare -x COLORTERM="truecolor"  
declare -x DBUS_SESSION_BUS_ADDRESS="unix:path=/run/user/1000/bus"  
declare -x DESKTOP_SESSION="ubuntu"  
declare -x DISPLAY=":0"  
declare -x GDMSESSION="ubuntu"  
declare -x GJS_DEBUG_OUTPUT="stderr"  
declare -x GJS_DEBUG_TOPICS="JS ERROR;JS LOG"  
declare -x GNOME_DESKTOP_SESSION_ID="this-is-depre"
```

tt. Pwd

Used to print working directory

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ pwd  
/home/rohit  
rohit@rohit-VB1:~$
```

uu. Crontab:

Used to maintain crontab program files for individual users

vv. Mount: All files in Ubuntu are maintained in a tree, Mount is used to attach filesystem found on other system to tree

ww. Passwd

Given command is used to change the password

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ passwd  
Changing password for rohit.  
(current) UNIX password: █
```

xx. Uname : Is is used to get username

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ uname  
Linux  
rohit@rohit-VB1:~$
```

yy. Whereis : It is used to get location of executable source code for given command

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ whereis ls  
ls: /bin/ls /usr/share/man/man1/ls.1.gz  
rohit@rohit-VB1:~$
```

zz. Whatis : It is used to get single line summary of linux commands

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ whatis head  
head (1) - output the first part of files  
HEAD (1p) - Simple command line user agent  
rohit@rohit-VB1:~$
```

aaa. Less

Used to print less outputs on console screen

```
rohit@rohit-VB1: ~/Rohit
File Edit View Search Terminal Help
rohit@rohit-VB1:~/Rohit$ ls | less
```

```
rohit@rohit-VB1: ~/Rohit
File Edit View Search Terminal Help
1.cpp
5.c
(END)
```

bbb. Su : It is used to change the user for given session

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ su rohit
Password:
```

ccc. Ping

Ping command is used to get response time from a server of DNS which is also called as latency

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help
rohit@rohit-VB1:~$ ping -c 3 www.google.com
PING www.google.com (172.217.15.100) 56(84) bytes of data.
64 bytes from iad30s21-in-f4.1e100.net (172.217.15.100): icmp_seq=1 ttl=64 time=773 ms
64 bytes from iad30s21-in-f4.1e100.net (172.217.15.100): icmp_seq=2 ttl=64 time=867 ms
64 bytes from iad30s21-in-f4.1e100.net (172.217.15.100): icmp_seq=3 ttl=64 time=506 ms

--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 4994ms
rtt min/avg/max/mdev = 506.168/715.795/867.864/153.168 ms
rohit@rohit-VB1:~$
```

ddd. **Traceroute:** It is used to see how our packet is travelling in internet it given all the server names from which our packet is going

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ traceroute www.google.com  
traceroute to www.google.com (172.217.15.68), 30 hops max, 60 byte packets  
1 _gateway (192.168.0.1) 6.186 ms 6.923 ms 6.820 ms  
2 bdl1.mcm-cbr1.chi-mcm.il.cable.rcn.net (10.20.0.1) 54.536 ms 54.340 ms  
4.173 ms  
3 216.80.78.91 (216.80.78.91) 54.102 ms 54.042 ms 53.732 ms  
4 207.172.19.158 (207.172.19.158) 53.576 ms 53.527 ms 207.172.19.166 (207.  
72.19.166) 53.472 ms  
5 207.172.19.163 (207.172.19.163) 53.175 ms 53.123 ms 207.172.19.171 (207.  
72.19.171) 52.440 ms  
6 207.172.9.38 (207.172.9.38) 52.256 ms 72.14.197.93 (72.14.197.93) 1619.5  
9 ms 207.172.9.38 (207.172.9.38) 1619.239 ms  
7 108.170.243.175 (108.170.243.175) 1619.116 ms^Z
```

eee. **Date:** It prints date stamp along with zone

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ date  
Mon Feb 4 19:35:52 CST 2019  
rohit@rohit-VB1:~$
```

fff. **Time :** Gives time required to complete given request

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$ time ls  
'='  
0  
0]]  
111.txt  
1.c  
1.sh  
222.txt  
2.sh  
55.sh  
abc.tar  
checkinte.sh  
Desktop  
Documents  
Downloads  
examples.desktop  
example.txt  
firstshell.sh  
Hello  
helloworld.py  
matplotlib1.py  
Music  
nested-loops.sh  
network-test-latency.txt  
network-test-machine-list.txt  
network-test.sh  
NewTets  
newtrytry.sh  
Oldnetwork-test.sh  
output.txt  
Pictures  
Public  
Rohit  
rohit.tar  
Templates  
Videos  
WRNGmatplotlib2.py  
real 0m0.005s  
user 0m0.003s  
sys 0m0.000s  
rohit@rohit-VB1:~$
```

ggg. **Wget :** It's a non-interactive network downloader


```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

rohit@rohit-VB1:~$ wget -V
GNU Wget 1.19.4 built on linux-gnu.

-cares +digest -gpgme +https +ipv6 +iri +large-file -
nk +nls
+ntlm +opie +psl +ssl/openssl

Wgetrc:
    /etc/wgetrc (system)
Locale:
    /usr/share/locale
```

hhh. W : It is used to get information about users and their instances

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

rohit@rohit-VB1:~$ w
19:46:14 up 5:46, 1 user, load average: 0.01, 0.04, 0.
1
USER      TTY      FROM          LOGIN@      IDLE        JCPU
CPU WHAT
rohit     :0       :0            10:10      ?xdm?      2:18
.00s /usr/li
rohit@rohit-VB1:~$
```

iii. Clear

Command clear is used to clear the content of console

```
rohit@rohit-VB1: ~
File Edit View Search Terminal Help

rohit@rohit-VB1:~$ cat example.txt
abc 500
dhf 634
wqs 721
gls 396
pqr 400
wnf 550
qmd 300
aws 200
acz 250
rohit@rohit-VB1:~$
```

Clear cleared all the results

```
rohit@rohit-VB1: ~  
File Edit View Search Terminal Help  
rohit@rohit-VB1:~$
```

jjj. Exit

Exit is used to exit the console.

```
rohit@rohit-VB1:~$ exit
```

14. Write bash scripts to do the following:

a. Write a script called “nested-loops.sh” that takes in 3 arguments (I, J, and K), each with numerical values, which then will implement 3 nested for loops that iterate from 0 to I, J, and K. You should test the input arguments that they are integers, and that they are a positive number; if they are not, the script should exit with the appropriate error message. If there are more, or less arguments than the required 3 arguments, the script should again fail with the appropriate message.

b. Write a script called “disk-benchmark-background.sh” that uses the dd command to run a benchmark against the local disk in the background, that captures all the output (both standard out and error output) to a file “disk-benchmark-background-log.txt”. Use the “time” command to show how long the benchmark took to complete. The benchmark should run for at least 10 seconds, and it should complete even if the ssh (or bash) session is terminated.

c. Write a script called “network-test.sh” that takes input a file “network-test-machinelist.txt” with a list of DNS names (e.g. google.com, iit.edu, anl.gov), each name on a separate line, and runs the ping utility collecting 3 samples from each DNS name, and writing the RTT (round trip time) average latency into a file “network-test-latency.txt” where each line will have the DNS name and average RTT separated by a space. Make sure it works with at least 10 DNS names, but it should work for an unspecified number of DNS names.

d. Write a script using Ploticus or Matplotlib (pick one) to generate a graph of the “network-test-latency.txt” data. The graph should automatically adjust to the number of entries, and the scale of the data.

15. Answer the following questions:

a. In the system configuration of the VM, explain how changing the number of processors changes the behavior of your VM. Explain a scenario where you want to set this to the minimum, and a scenario where you want to set it to the maximum. Why is setting it to the maximum potentially a bad idea?

To some extent it will improve the performance of your system keeping your RAM constant but to improve it to further level we must increase the RAM. If you consider the case where you are allocating more than one processor to system but virtualization techniques are not present in the BIOS then it won't help in maximizing the performance. Allocating more processors to VM can cause spin lock situation in which host is blocked by the multiprocessor guest system. Sharing more CPU's with guest can cause problems like degradation in the responsiveness of host, glitches. If user wants his/her host to perform at peak then it is good to allocate minimum processors to guest system.

b. In the system configuration of the VM, under the Acceleration Tab, explain the difference between the paravirtualization options: None, Legacy, Minimal, Hyper-V, and KVM. Explain which one would be best to use with Ubuntu Linux, and why.

Given options provides the partial virtualization options which we can provide to guest operating system

1. None: No paravirtualization
2. Legacy: It is for older virtual box applications.
3. Minimal: This is for mac OS guest users, It provides TSC and APIC frequency to guest operating systems.
4. Hyper-V: This is for windows guest machines, commonly identified by windows 7 or newer windows systems. It supports features like Para virtualized clocks, Guest crash reporting and relaxed timer clocks
5. KVM: This is for good for Linux systems, It supports Para virtualized clocks and SMP sniplocks.

KVM good for Linux reasons:

- Smaller and faster
- Can be used with other guests
- Can save machines state on hard drive and close.

c. In storage devices when configuring the VM, there are multiple types of storage controllers: explain the difference between the IDE, SATA, and NVMe controller. Give an example for each type of storage controller of a scenario where you may want to use this type of controller.

Points	IDE	SATA	NVMe
Pin	40	9	Around 70
Connections	Two devices	Allows only one connection	Two
Speed	133 mebibytes/second	1.5Gbits/second	3.5GB/sec
Developed by	Western Digital Electronics in association	Serial ATA Working Group	NVM express
Cost	Better value for money	Least expensive	Higher cost

When we need to process critical applications with heavy databases at that time we have to use NVMe, For high capacity, low availability and sequential needs we need SATA else for regular processing IDE is required.

d. In the network configuration of the VM, there are multiple types of network adapters: explain the difference between NAT, Bridged Adapter, Internal Network, and Host-only Network. Give an example for each type of network of a scenario where you may want to use this type of network.

Connectivity	NAT	Bridged Adapter	Internal Network	Host Only
VM & Host connectivity	No	Yes	No	Yes
Between two VM's	No	Yes	Yes	Yes
VM to internet	Yes	Yes	No	No
Internet to VM	No (Can be done by port forwarding)	Yes	No	No
Network activities	Mask all network activities	Replicates another node in current network	Can directly communicate to outside network	Network operations with host OS

e. For the USB configuration of the VM, explain the difference between USB 1.1, 2.0, and 3.0 controllers.

Attributes	USB 1.1	USB 2.0	USB 3.0
Bandwidth	12 Mbps	480 Mbps	4.8 Gbps
Ideal For	Keyboard, Mouse, Printers	Mass storage devices, Video adapters, Data transfer cables	Large mass storage devices, Video adapters
Power required for configured devices	500mA	500mA	900-1000mA
Power required for non-configured devices	100mA	100mA	150mA
Speed	Average speed	High speed	Super high speed
Backward compatible	NA	USB 1.1	USB 2.0/USB 1.1