GOVERNMENT OF KARNATAKA

DEPARTMENT OF COLLEGIATE & TECHNICAL EDUCATION

OPERATING SYSTEM AND ADMINISTRATION

LAB RECORD

Academic Year	:	2023 – 2024
Course Code	:	20CS42P
Semester	:	IV Semester
Branch	:	Computer Science & Engineering

Submitted By:

Student Name :

Register Number :

Department of Computer Science and Engineering

GOVERNMENT POLYTECHNIC

SIDDAPUR (U.K) – 581355

GOVERNMENT OF KARNATAKA

DEPARTMENT OF COLLEGIATE & TECHNICAL EDUCATION

GOVERNMENT POLYTECHNIC
SIDDAPUR (U.K) - 581355

Certificate

This is to Certify that it is bonafied record of Practical work done by Mr./Ms ____bearing the Register No. _____________ of IV semester, Computer Science & Engineering Branch in the OPERATING SYSTEMA ND ADMINISTRATION – 20CS42P *Laboratory during the academic year 2023 – 2024.*

Course Coordinator **Head of the Department**

<u>Examiners:</u> ***1.*** __________________________________

2. __________________________________

Index

Name: **Register Number:**

Branch: Computer Science & Engineering **Class: IV Semester**

Institution: Government Polytechnic, Siddapur

Week No	Practical Exercise	Marks obtained	Initials

Average Marks: $\frac{\quad}{10}$

Signature of the student **Signature of the Course Coordinator**

Week 1 Practice

Exp #1: Types of OS installation, boot methods, file system

1. Types of OS installation

(a) Clean installation (or) no create backup
(b) Dual boot installation
(c) Upgrading installation
(d) OS migration (backup created)

Clean installation: here we format the driver completely and then install the OS. No backup is created. New copy of OS will be installed.

Dual boot installation: installing two OS in the same computer is called as dual-boot. It allows user to choose and works with one OS at a time.

Upgrading installation: installing the OS on the computer without removing the older version first & without saving any data. It preserves previously installed software's, configuration files etc.

(OR)

Installs newer OS files over the old ones and preserves all your files.

OS migration: moving from one OS to another such as migrating from windows to Linux.

2. Boot methods

Available boot methods are:

(a) CD or DVD

(b) HARD DRIVE

(c) USB FLASH DRIVE

(d) SSDs

Booting is the process of starting the computer. When CPU is switched on it has nothinginside the memory. So, during booting OS is loaded into main memory.

Types of Booting: -

(1) Cold booting: also called as hard-boot. Computer is started from its initial stateby pressing the power button.

(2) Warm booting: also called as soft-boot. It refers to when we restart thecomputer. Here, the computer does not start from the initial state.

3. File system and formatting

FILE SYSTEM

- A file system is the method used to organize data on disk.
- It controls the allocation of disk space to files and associates each file with filename, directory, permissions and other information.

The Linux file system contains the following sections:

- The root directory (/)
- A specific data storage format (EXT3, EXT4, BTRFS, XFS and so on)
- A partition or logical volume having a particular file system.

Types of Linux File System

Linux offers many file systems such as **Ext, Ext2, Ext3, Ext4, JFS,ReiserFS, XFS, btrfs,** and **swap**.

(a) Ext, Ext2, Ext3 and Ext4 file system

- Ext stands for **Extended File System**.
- It was primarily developed for **MINIX OS**. The Ext file system is an olderversion, and is no longer used due to some limitations.
- **Ext2** is the first Linux file system that allows managing two terabytes of data.
- Ext3 is developed through Ext2; it is an upgraded version of Ext2 and contains backward compatibility.
- The major drawback of Ext3 is that it does not support servers because this filesystem does not support file recovery.
- **Ext4** file system is the faster file system among all the Ext file systems.

- It is a very compatible option for the SSD (solid-state drive) disks, and it is thedefault file system in Linux distribution.

(b) JFS File System

- JFS stands for **Journaled File System**, and it is developed by **IBM for AIXUnix**.
- It is an alternative to the Ext file system.
- It can also be used in place of Ext4, where stability is needed with few resources.

(c) ReiserFS File System

- ReiserFS is an alternative to the Ext3 file system.
- It has improved performance and advanced features.

(d) XFS File System

- XFS file system was considered as high-speed JFS, which is developed forparallel I/O processing.

(e) Btrfs File System

- Btrfs stands for the **B tree file system**.
- It is used for fault tolerance, repair system, administration, extensive storage configuration, and more.

(f) Swap File System

- The swap file system is used for memory paging in Linux operating systemduring the system hibernation.
- A system that never goes in hibernate state is required to have swap spaceequal to its RAM size.

Week 2 Practice

Exp #1: install and configure virtual machine

Virtual box installation

Follow below steps to install and configure.

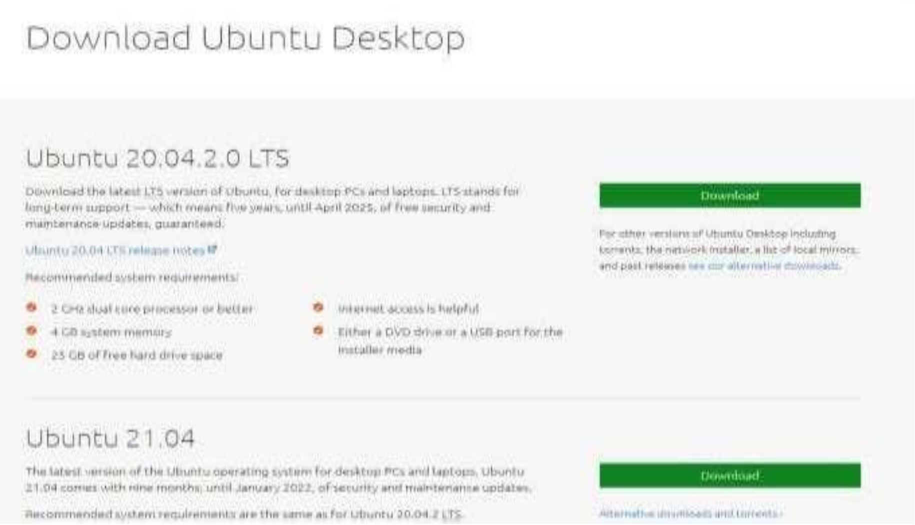
1. Open the VirtualBox website. Go to https://www.virtualbox.org/ in the computer's Internetbrowser. This is the website from which you can download the VirtualBox setup file.

2. Click Windows hosts. You will see this link below the "VirtualBox 6.1.34 platform packages"heading. The VirtualBox EXE file will begin downloading onto your computer.
3. Open the VirtualBox EXE file. Go to the location to which the EXE file downloaded and double-click the file. Doing so will open the VirtualBox installation window.
4. Navigate through the installation prompts.
5. Do the following:

- Click Next on the first three pages.
- Click Yes when prompted.
- Click Install
- Click Yes when prompted**.**

6. Click Install when prompted. Doing so will allow VirtualBox to begin installing on yourcomputer.
7. Click Finish when prompted. It's in the lower-right side of the window. Doing so will close the installation window and open VirtualBox. Now that you have installed and opened VirtualBox, youcan create a virtual machine in order to run any operating system on your PC.

Setting up Virtual machine (Linux on Virtual box)
Download the ISO file for Ubuntu



Download Ubuntu Desktop

Ubuntu 20.04.2.0 LTS

Download the latest LTS version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years, until April 2025, of free security and maintenance updates, guaranteed.

Ubuntu 20.04 LTS release notes

Recommended system requirements:

- 2 GHz dual core processor or better
- 4 GB system memory
- 25 GB of free hard drive space
- Internet access is helpful
- Either a DVD drive or a USB port for the installer media

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors, and past releases see our alternative downloads.

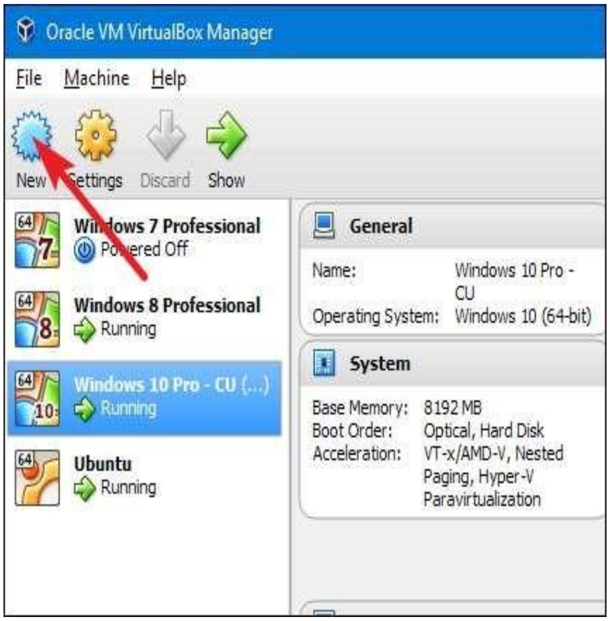
Ubuntu 21.04

The latest version of the Ubuntu operating system for desktop PCs and laptops, Ubuntu 21.04 comes with nine months, until January 2022, of security and maintenance updates.

Recommended system requirements are the same as for Ubuntu 20.04.2 LTS.

Setting up Virtual machine (Linux on Virtual box)

Open up your VM app and click the **"New"** button to create a new virtual machine
Type the name of the OS in the "Name" box, the app will most likely automatically select the type and version for the OS. When you are done, click "Next."



Windows 7 Professional
Powered Off

Windows 8 Professional
Running

Windows 10 Pro - CU (...)
Running

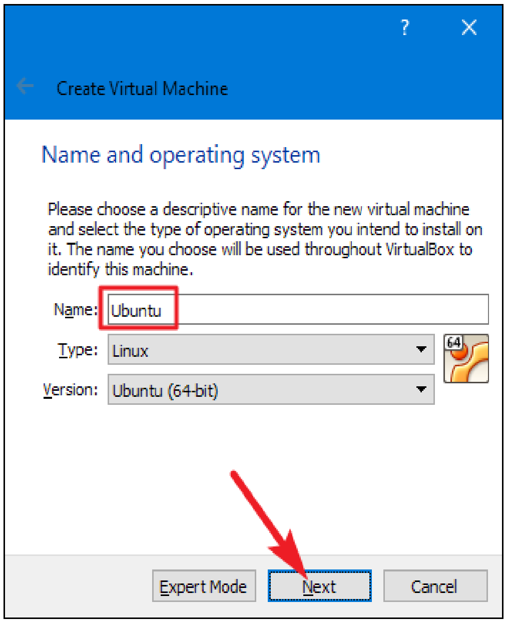
Ubuntu
Running

General

Name: Windows 10 Pro - CU
Operating System: Windows 10 (64-bit)

System

Base Memory: 8192 MB
Boot Order: Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, Hyper-V Paravirtualization

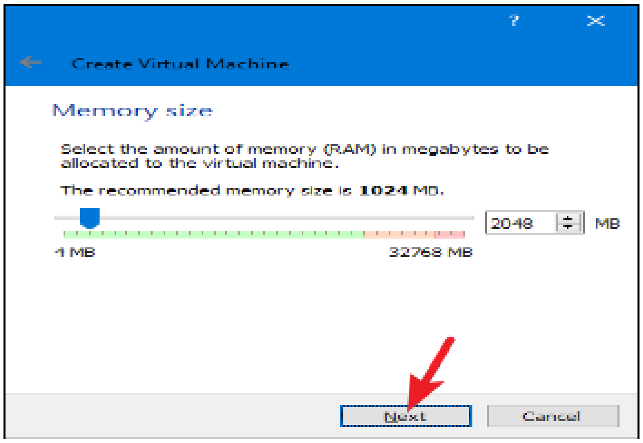


Name and operating system

Please choose a descriptive name for the new virtual machine and select the type of operating system you intend to install on it. The name you choose will be used throughout VirtualBox to identify this machine.

Name: Ubuntu
Type: Linux
Version: Ubuntu (64-bit)

You will be asked how much memory to allocate to the VM. If you want something other than the default, select it here. Otherwise, just click "Next."
The wizard will also create the virtual hard disk



Create Virtual Machine

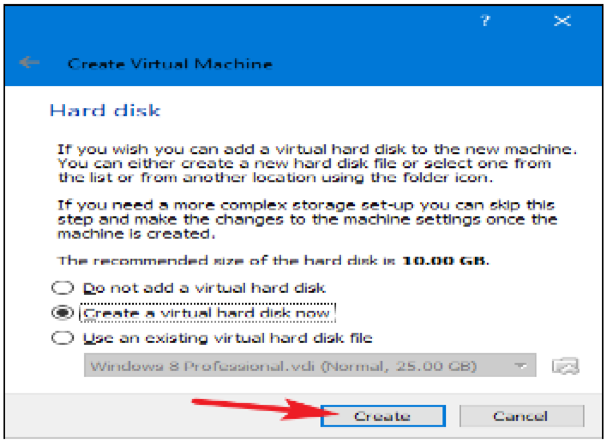
Memory size

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024** MB.

2048 MB

4 MB 32768 MB



Create Virtual Machine

Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **10.00 GB**.

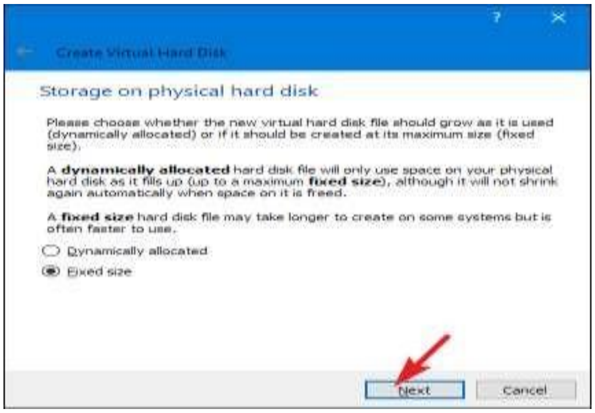
○ Do not add a virtual hard disk

● Create a virtual hard disk now

○ Use an existing virtual hard disk file

Windows 8 Professional.vdi (Normal, 25.00 GB)

You will also be asked whether to create a dynamically allocated or fixed size disk.
You'll then be able to set the size of the virtual disk. You're free to go with the default setting or change the size to suit your needs. Once you click "Create," the virtual hard disk is created.



Create Virtual Hard Disk

Storage on physical hard disk

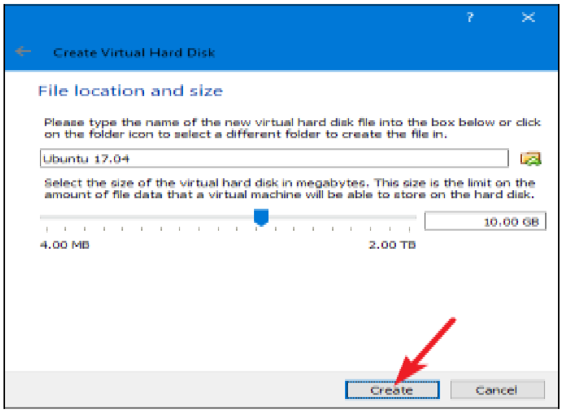
Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

○ Dynamically allocated

● Fixed size



Create Virtual Hard Disk

File location and size

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file in.

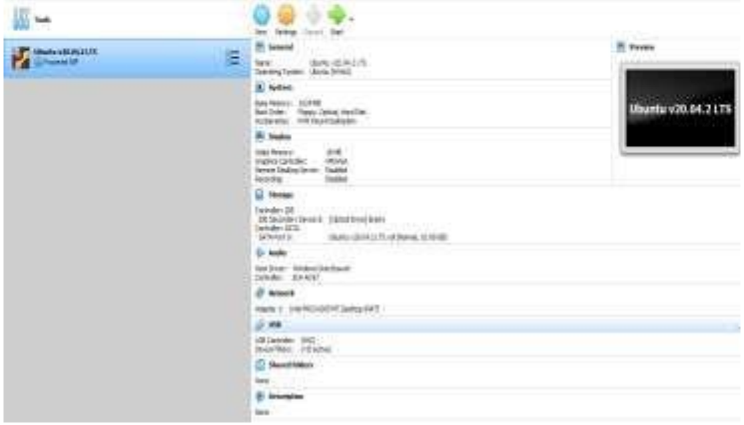
Ubuntu 17.04

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

10.00 GB

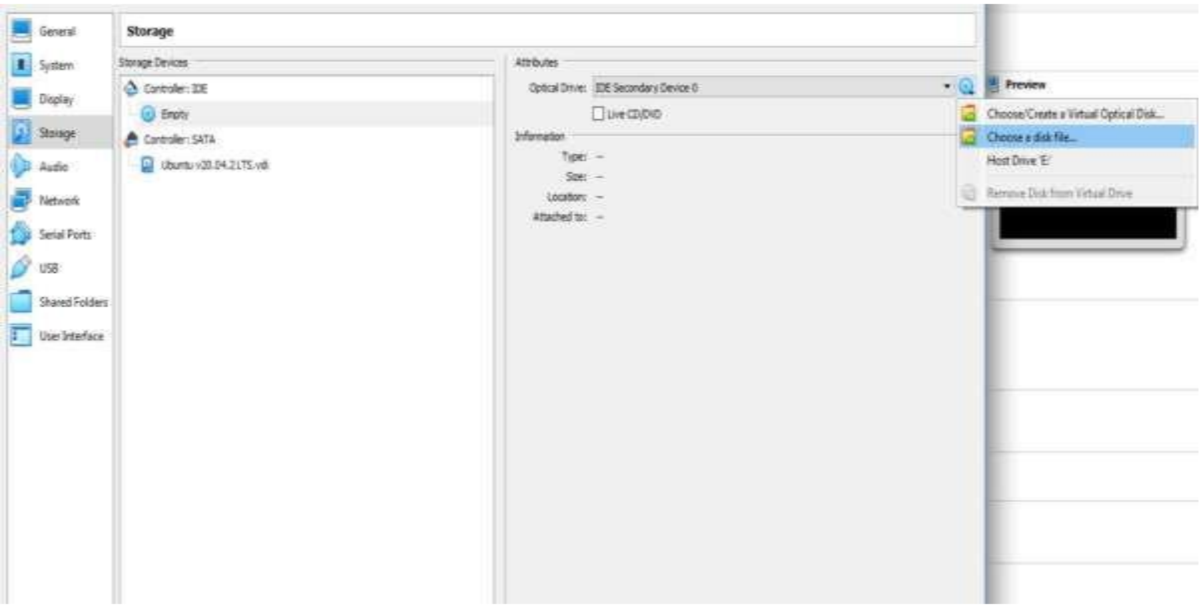
4.00 MB 2.00 TB

This completes the first part of the configuration process. Our Virtual machine has been set up Now make sure that you have hardware virtualization enabled in our Bios settings. Now will see Ubuntu listed on Virtual box's left sidebar. Select it and click on settings.



Look for the **storage** on the left sidebar of the **settings** dialog box. Click on the tiny disc icon in the **attribute** section. Select **Choose a disk file**, navigate to the ISO file, and select **OK**

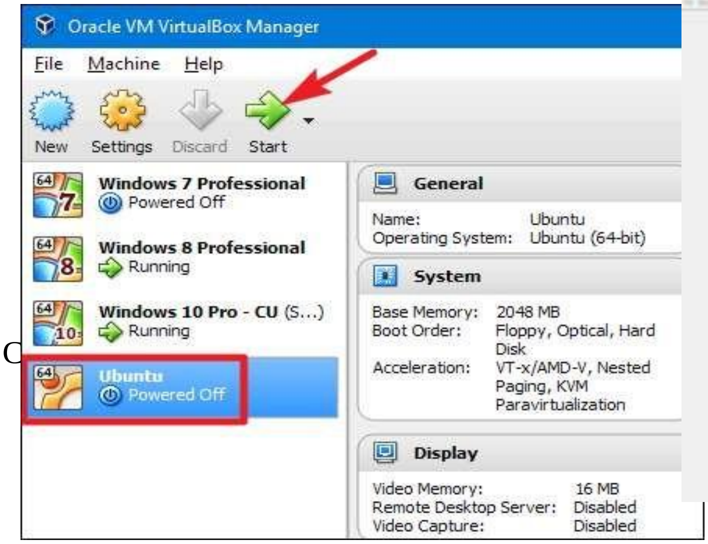
You can run your new VM by selecting it and hitting "Start."



Optical Drive: IDE Secondary Device 0
☐ Live CD/DVD
Information
Type: –
Size: –
Location: –
Attached to: –

Start by clicking the Start button on the VirtualBox home screen. Select the Ubuntu ISO file as the start-up disk in the dialog box that pops up and select Start. This may take a few minutes

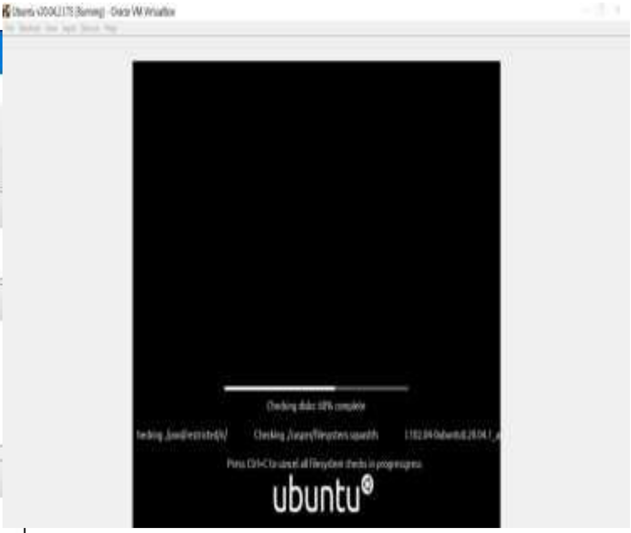
C



Name: Ubuntu
Operating System: Ubuntu (64-bit)

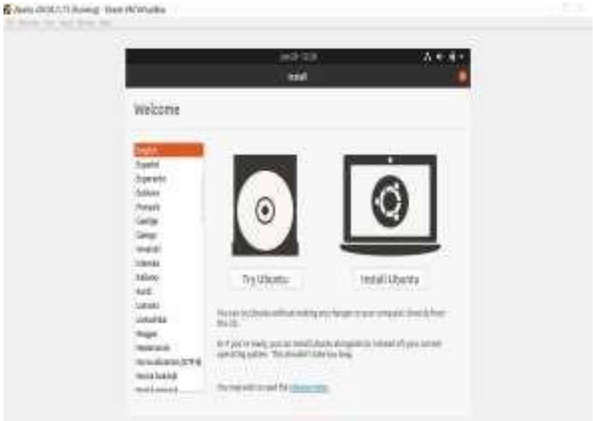
Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, KVM Paravirtualization

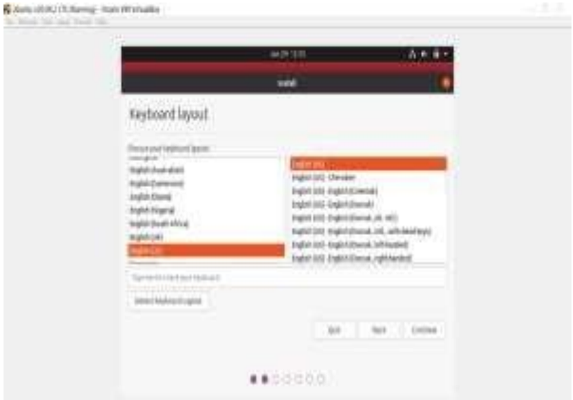
Video Memory: 16 MB
Remote Desktop Server: Disabled
Video Capture: Disabled



Checking disks: 18% complete

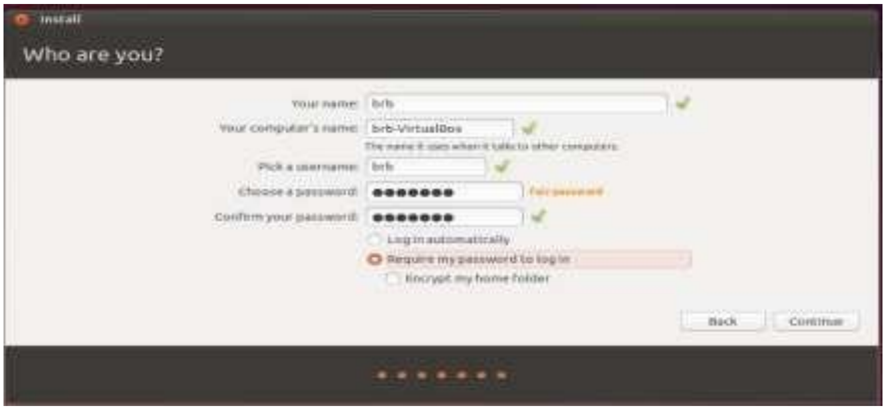
Press Ctrl+C to cancel all filesystem checks in progress





- ✓ Keep following the installation wizard’s prompts.

- ✓ You may be asked if you want to **Erase disk and install Ubuntu**. This is normal. Just select **Install Now** and move forward.
- ✓ Choose your Region and select Continue.
- ✓ You will be asked to give your details such as your name, computer's name, username and password.



Install

Who are you?

Your name: brb ✓

Your computer's name: brb-VirtualBox ✓

The name it uses when it talks to other computers.

Pick a username: brb ✓

Choose a password: ●●●●●●● Fair password

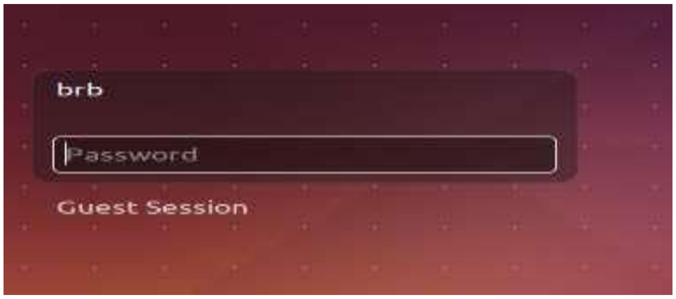
Confirm your password: ●●●●●●● ✓

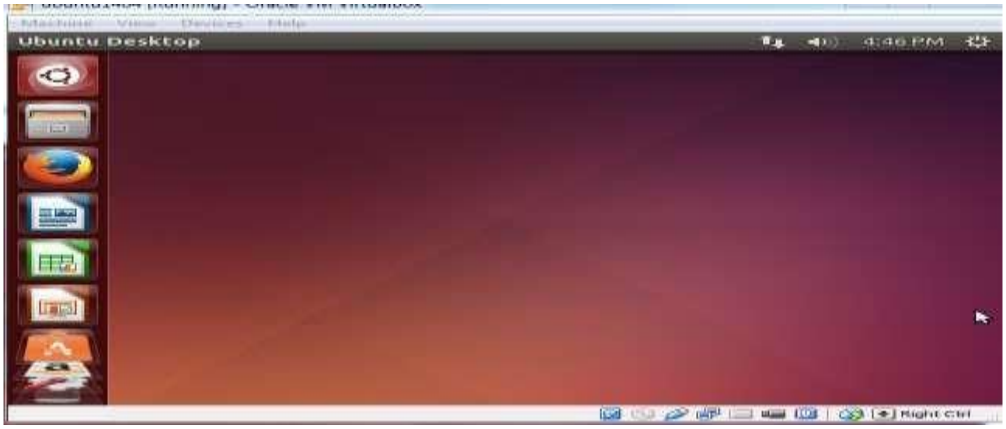
○ Log in automatically

● Require my password to log in

□ Encrypt my home folder

- ✓ The installation wizard will continue the installation process by itself after collecting the details. This could take a few minutes.
- ✓ When the installation completes, your virtual machine will automatically reboot. Run the Virtual machine again, and it should boot into Ubuntu.





2. TERMINAL EMULATOR

- A Terminal emulator is a computer program that reproduces a video terminal within some other display structure (i.e., remote machine desktop will be appeared in local machine as a terminal).
  - In other words, the Terminal emulator has the ability to make a dumb machine appear like a client computer networked to the server.
- The terminal emulator allows an end-user to access the console as well as its applications such as text user interface and command-line interface.

Installation of terminal emulator

Example 1: xterm

- The **xterm** terminal application is a standard terminal emulator for the X Window System. It allows users to run programs which require a command-line interface.

Installation:

- **sudo apt-get update**
- **sudo apt-get install xterm**

Example 2: mate-terminal

- MATE Terminal also has the ability to use multiple terminals in a single window (tabs) and supports management of different configurations (profiles).
- MATE Terminal is a fork of GNOME-Terminal.

Installation:

- **sudo apt-get update -y**
- **sudo apt-get install -y mate-terminal**

Week 3 Practice

Exp #1: Create and delete directories and files, File copy commands, Commands for viewing File, File comparison, Altering file permission and Text Processing Commands.

File creating command

gedit :

syntax: gedit filename

this command creates a file with filename specified. If the file has already created, then it will open the file.

mkdir (Make Directory):creating Directory

- mkdir will create a new directory (folder)

Syntax: **mkdir directoryname(folder name)**

Example 1: $**mkdir polytechnic** // will create directory with name polytechnic

Example 2: Create some additional subdirectories within **polytechnic**. Listnew directories after the command completes.

$mkdir polytechnic/mechanical polytechnic/computer polytechnic/civil

$ls polytechnic

ls (list):

The ls command will list the contents of your current directory. Entering ls alonewill show a list of the current directory contents.

Syntax:

```
ls [option];
```

The Four commonly used options are **-a**,and **-l**

- **ls -a:** List files all begin with a "**dot**", which indicates they are "**hidden**" files.
  - **ls** with **-l** to obtain a "long" listing of files. An explanation of the informationit provides appears below. **ls -l**

```
-rwxr-xr-x    1 jsmith    staff   43     Mar 23 18:14 prog1
-rw-r--r--    1 jsmith    staff   10030  Mar 22 20:41 sample.f
drwxr-sr-x    2 jsmith    staff   512    Mar 23 18:07 subdir1
drwxr-sr-x    2 jsmith    staff   512    Mar 23 18:06 subdir2
drwxr-sr-x    2 jsmith    staff   512    Mar 23 18:06 subdir3
      1       2    3        4       5       6             7
```

1 = access modes/permissions

2 = number of links

3 = owner

4 = group

5 = size (in bytes)

6 = date/time of last modification

7 = name of file

pwd (Present Working Directory)

- Will give the present working directory path informatoin

```
$pwd
/Home/Ubuntu
```

cd (Change Directory)

- cd is used to Change Directory
- Change to home direcotory
  - **cd** // Change to default home directory
  - **cd ~** // Change to default home directory
- Change to a subdirectory within user home directory
  - **cd polytechnic/computer**
  - pwd
- Go up one level back to the current directory's parent directory
  - **cd ..**
  - pwd
- Change to the root (top-most) directory
  - **cd /**
  - pwd
- Change to another directory
  - **cd ~/polytechnic**
  - pwd
- Change to another one of subdirectories
  - **cd ~/polytechnic/computer**
  - pwd

rmdir (Remove Directory):

- Will remove (delete) the directory.Make sure that folder is empty before issuingrmdir command.
- Make sure current directory is not part of the directory being deleted.

(assume current directory is /home/polytechnic) (assume /polytechnic/computer folder exists)

```
$rmdir civil
```

```
$rmdir /polytechnic/computer
```

rm File (Remove File)

Will remove (delete) the files

- Syntax:

```
$rm FileName
```

```
$rm path/Filename
```

- Example:

rm a.txt // Assume a.txt is present in current directory

rm polytechnic/civil/a.txt // Assume polytechnic/civil folder has file a.txt rm -i * // will delete all the files but interactively, asking user aboutconfirmation of deleting the files

rm polytechnic/civil/*// will delete all files in polytechnic/civil folder

cp command

The **cp** program copies files and directories. In its simplest form, it copies asingle file:

```
$ cp file1 file2
```

It can also be used to copy multiple files (and/or directories) to a different directory:

```
$ cp file... directory
```

Commands for viewing File

- **Syntax**: **cat filename**

cmp(File comparison)

- The *file comparison command* helps us to *compare* the *files* and find the similarities and *differences* between these *files*.
- The different file comparison commands used in Linux are: cmp, comm, anddiff.

1. **Cmp:** This command is used to compare two files byte by byte.

✓ It displays the byte and line number where first difference is found.

✓ If no difference is found no output will be displayed.

Syntax :cmp <file1><file2>

Example: $ cmp file1 file2 file1 file2 differ: byte 1, line 1

Contents of file1	**Contents of file2**
It"s fun playing	It"s fun Playing
Cricket	Cricket

2. **comm**: This command compares two sorted files line by line . This command

displays 3 column output, the first column contains unique lines of file1, the second column contains the unique lines of file2 and the third column contains lines common to both file1 and file2

Syntax: comm file1 file2 //The contents of input files should be sorted alphabetically.E.g.: Create 2 files with name file1 and file2

Contents of file1:
apple,
mango,
orange

Contents of file2:
grape,
mango, musk
melon

```
Example: comm file1 file2apple
                    grape
                                      mango
                    muskm
orange
                    elon
```

Column 1 gives the names which are present in first file but not in second file **Column 2** gives the names which are present in second file but not present in first file**Column 3** gives the names which are present in both the Files

3. **diff:** This command is used to find difference between two files. Diff displayswhich lines in one file have to be changed to make the file identical to other.

Syntax: diff file1 file2

1
c
1
“
”

3c3

E.g.: **Contents of file**1:

apple,
mang
o,
orang
e

Contents of file2

grape,
mango,
musk melon

Example: diff file1 file2

Understanding Output

c – change d – delete a –
append1c1
<apple
…
grape>

1c1: 1st line of 1st file to be changed to match 1st line of 2nd file“”
means 2nd file

3c3:likewise can be understood.

< orange

musk melon >

Altering file permission (chmod):

Linux chmod command is used to change the access permissions of files and directories. It stands for **change mode**..

In the Linux file system, each file is associated with a particular owner and have permission access for different users. The user classes may be:

- User(u)
- group member(g)

- Others (o)

The file permissions in Linux are the following three types:

- read (r)
- write (w)
  - execute (x)

Let's see how to change the file permission using the chmod command.

Syntax:

$chmod <options> <permissions> <file name>

Example:

1)chmod u+x file1.txt 2)chmod u-x file1.txt 2)Chmod 743 demo.txt;

Digits	**Permissions**	**Decimal Equivalent**
000	No permission	0
001	Execute permission	1
010	Write permission	2
011	Write and execute permissions	3
100	Read permission	4
101	Read and execute permissions	5
110	Read and write permissions	6
111	Read, write, and execute permissions	7

Filters / Text Processing Commands

Filters are commands which accept data from standard input manipulate it and writethe results to standard Output.

1. **head**: This command is used to displays the top of the file. When used withoutany option, it will display first 10 lines of the file.

Syntax : $ head filename // shows first 10 linesSyntax : $ head –n filename

Create a file sample1.txt using text editor and type at least 25 lines

Example: $ head -n 20 sample1.txt or $ head 20 sample1.txt Output: /* will display first 20 lines*/

2. **tail:** command displays the end of the file. By default it will display last 10 lines ofthe file.

Syntax: tail filename Example: tail sample1.txt

/*display last 10 lines*/

(Note: Create a file sample.txt using text editor and type at least 15 lines)

tail -n followed by a number will display that many number of lines from lastrespectively.

$tail -n 15 sample1.txt

/* will display last 15 lines */

3. **wc (statistic of file)** : This command is used to count lines, words and characters, Depending on the option used.

Syntax : wc [options] [file name]

Options : -l: Number of lines

-w: Number of words

-c: Number of characters

Example: $ wc sample1.txt

Output: 65 2776 17333 sample1.txt

Which means sample1.txt file has 65 lines, 2776 words, and 17333 Characters.

Week 4 Practice

Exp #1: linux commands related to process creation and management

Process creation and Management

- A process is simply an instance of a running program. A process is said to be born when the program starts execution and remains alive as long as the program is active.
- After execution is complete, the process is said to die. It is the kernel who isresponsible for the management of processes (not the shell).

Two important attributes of a process are:

- **The Process-ID (PID):** Each process is uniquely identified by a unique integer calledthe **Process-id(PID)** that is allotted by the kernel when the process is born.
- **The Parent PID (PPID):** The PIS of the parent is also available as a processattribute.

MECHANISM OF PROCESS CREATION:

The three distinct stages in process creation are:

- **fork():** A process in UNIX is created with the fork system call, which created a copy of the process that invokes it. The process image is practically identical to that of the calling process.
- **exec():** Forked child needs to overwrite its own image with the code and date of the new program. This mechanism is called exec, and the child process is said to exec a new program.
- **wait():** The parent then executes the wait system call to wait for the child process tocomplete.
- **ps: Process status**

ps displays the processes owned by the user running the command **$ps**

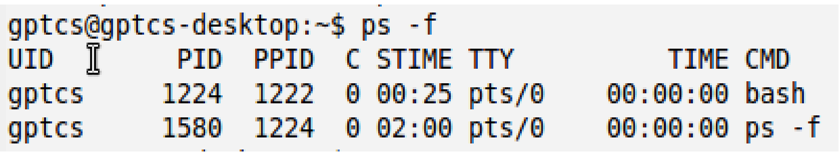
Each line in the output shows

- *PID*
- *Controlling Terminal (TTY):* The terminal with which process is associated.
- *The Cumulative Processor Time (TIME):* The time that has been consumed since theprocess has been started.
- *The Process Name (CMD)*

ps options

1. **–f** - full listing to get detailed listing which also shows the parent of every process
2. **–u** - to know the activities of any user
3. **–a** - all option displays or lists the processes of all users but do not displays the system processes.

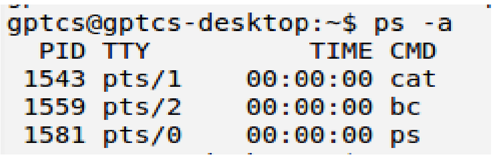
$ ps –f



```
gptcs@gptcs-desktop:~$ ps -f
UID         PID  PPID  C STIME TTY          TIME CMD
gptcs      1224  1222  0 00:25 pts/0    00:00:00 bash
gptcs      1580  1224  0 02:00 pts/0    00:00:00 ps -f
```

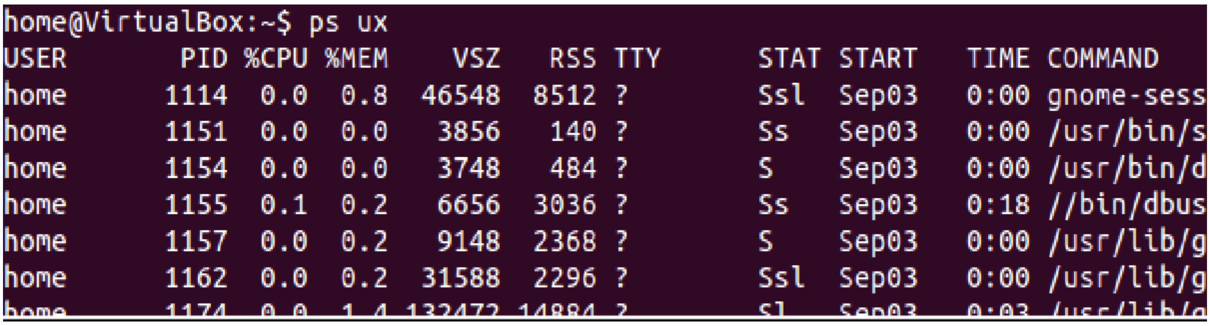
Displaying All User Process (-a): The –a option lists process of all processes of all users but doesn’t display the system process.

$ps -a



```
gptcs@gptcs-desktop:~$ ps -a
  PID TTY          TIME CMD
 1543 pts/1    00:00:00 cat
 1559 pts/2    00:00:00 bc
 1581 pts/0    00:00:00 ps
```

$ps ux

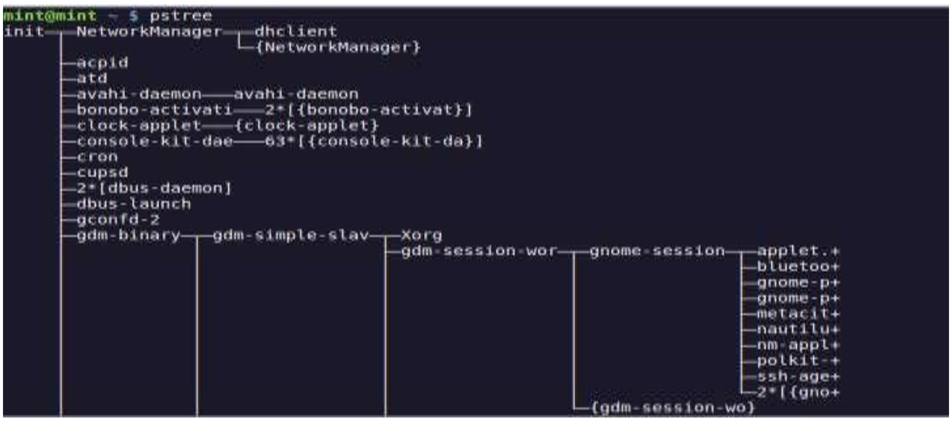


```
home@VirtualBox:~$ ps ux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
home      1114  0.0  0.8  46548  8512 ?        Ssl  Sep03   0:00 gnome-sess
home      1151  0.0  0.0   3856   140 ?        Ss   Sep03   0:00 /usr/bin/s
home      1154  0.0  0.0   3748   484 ?        S    Sep03   0:00 /usr/bin/d
home      1155  0.1  0.2   6656  3036 ?        Ss   Sep03   0:18 //bin/dbus
home      1157  0.0  0.2   9148  2368 ?        S    Sep03   0:00 /usr/lib/g
home      1162  0.0  0.2  31588  2296 ?        Ssl  Sep03   0:00 /usr/lib/g
home      1174  0.0  1.4 132472 14884 ?        Sl   Sep03   0:03 /usr/lib/g
```

pstree

This command helps visualize the processes by displaying a tree diagram of the processes showing the relationship between them.

$pstree

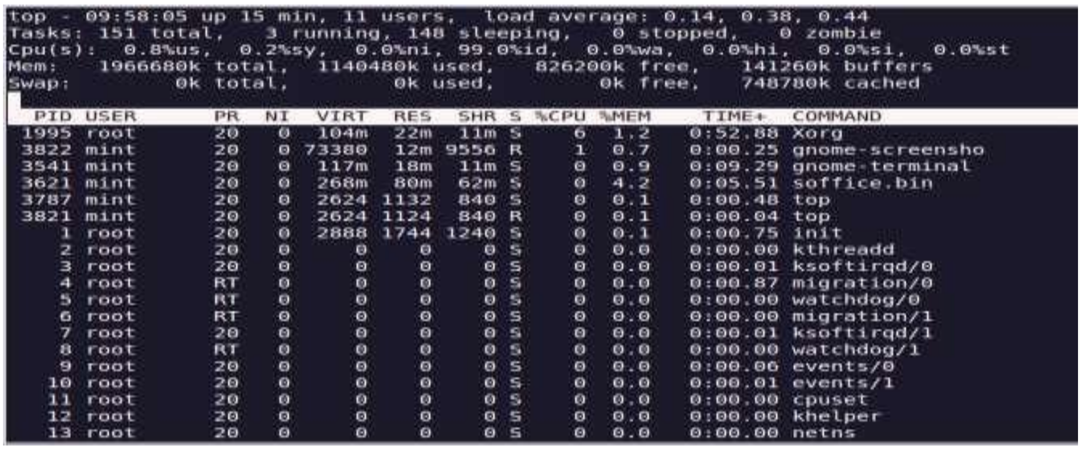


```
mint@mint ~ $ pstree
init─┬─NetworkManager─┬─dhclient
     │                └─{NetworkManager}
     ├─acpid
     ├─atd
     ├─avahi-daemon───avahi-daemon
     ├─bonobo-activati───2*[{bonobo-activat}]
     ├─clock-applet───{clock-applet}
     ├─console-kit-dae───63*[{console-kit-da}]
     ├─cron
     ├─cupsd
     ├─2*[dbus-daemon]
     ├─dbus-launch
     ├─gconfd-2
     ├─gdm-binary─┬─gdm-simple-slav─┬─Xorg
     │            │                 ├─gdm-session-wor─┬─gnome-session─┬─applet.+
     │            │                 │                 │               ├─bluetoo+
     │            │                 │                 │               ├─gnome-p+
     │            │                 │                 │               ├─gnome-p+
     │            │                 │                 │               ├─metacit+
     │            │                 │                 │               ├─nautilu+
     │            │                 │                 │               ├─nm-appl+
     │            │                 │                 │               ├─polkit-+
     │            │                 │                 │               ├─ssh-age+
     │            │                 │                 │               └─2*[{gno+
     │            │                 │                 └─{gdm-session-wo}
```

top

This utility tells the user about all the running processes on the Linux machine.

$top



```
top - 09:58:05 up 15 min, 11 users,  load average: 0.14, 0.38, 0.44
Tasks: 151 total,   3 running, 148 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.8%us,  0.2%sy,  0.0%ni, 99.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   1966680k total,  1140480k used,   826200k free,   141260k buffers
Swap:        0k total,        0k used,        0k free,   748780k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 1995 root      20   0  104m  22m  11m S    6  1.2   0:52.88 Xorg
 3822 mint      20   0 73380  12m 9556 R    1  0.7   0:00.25 gnome-screensho
 3541 mint      20   0  117m  18m  11m S    0  0.9   0:09.29 gnome-terminal
 3621 mint      20   0  268m  80m  62m S    0  4.2   0:05.51 soffice.bin
 3787 mint      20   0  2624 1132  840 S    0  0.1   0:00.48 top
 3821 mint      20   0  2624 1124  840 R    0  0.1   0:00.04 top
    1 root      20   0  2888 1744 1240 S    0  0.1   0:00.75 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   0:00.01 ksoftirqd/0
    4 root      RT   0     0    0    0 S    0  0.0   0:00.87 migration/0
    5 root      RT   0     0    0    0 S    0  0.0   0:00.00 watchdog/0
    6 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/1
    7 root      20   0     0    0    0 S    0  0.0   0:00.01 ksoftirqd/1
    8 root      RT   0     0    0    0 S    0  0.0   0:00.00 watchdog/1
    9 root      20   0     0    0    0 S    0  0.0   0:00.06 events/0
   10 root      20   0     0    0    0 S    0  0.0   0:00.01 events/1
   11 root      20   0     0    0    0 S    0  0.0   0:00.00 cpuset
   12 root      20   0     0    0    0 S    0  0.0   0:00.00 khelper
   13 root      20   0     0    0    0 S    0  0.0   0:00.00 netns
```

Press ‘q’ on the keyboard to move out of the process display

Fi e l d	Description	Example 1	Example 2
PID	The process ID of each task	1525	961
User	The username of task owner	Home	Root
PR	Priority Can be 20(highest) or -20(lowest)	20	20
NI	The nice value of a task	0	0
VIRT	Virtual memory used (kb)	1775	75972
RES	Physical memory used (kb)	100	51
SHR	Shared memory used (kb)	28	7952

S	Status‘D’ = uninterruptible sleep‘R’ = running‘S’ = sleeping	S	R
%CPU	% of CPU time	1.7	1.0
%MEM	Physical memory used	10	5.1
TIME+	Total CPU time	5:05.34	2:23.42
Field	Description	Example 1	Example 2
Command	Command name	Photoshop.exe	Xorg

Week 5 Practice

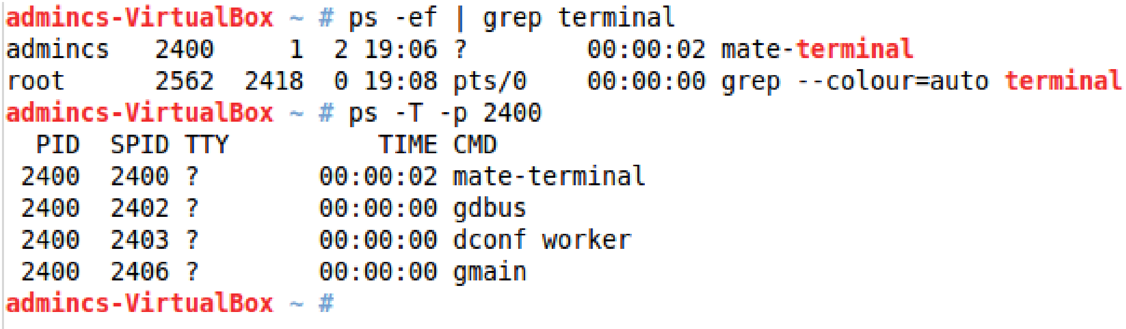
Exp #1 commands to exhibit thread concepts.

ps -T -p <pid>

Example:

List the threads for the **mate-terminal** process:

- First find the all process containing word **terminal**



```
admincs-VirtualBox ~ # ps -ef | grep terminal
admincs    2400     1  2 19:06 ?        00:00:02 mate-terminal
root       2562  2418  0 19:08 pts/0    00:00:00 grep --colour=auto terminal
admincs-VirtualBox ~ # ps -T -p 2400
  PID  SPID TTY          TIME CMD
 2400  2400 ?        00:00:02 mate-terminal
 2400  2402 ?        00:00:00 gdbus
 2400  2403 ?        00:00:00 dconf worker
 2400  2406 ?        00:00:00 gmain
admincs-VirtualBox ~ #
```

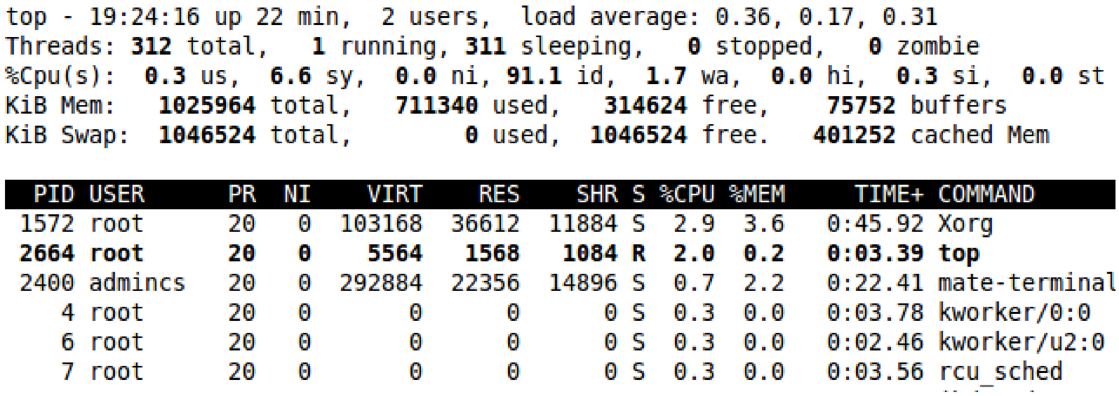
- The "**SPID**" column represents **thread IDs**, and "CMD" column shows threadnames.

Commands to exhibit thread concepts

Using the top

- The top command can show a real-time view of individual threads.
- To enable thread views in the top output, invoke top with "-H" option. This will list allLinux threads.
- User can also toggle on or off thread view mode while top is running, by pressing 'H'key.

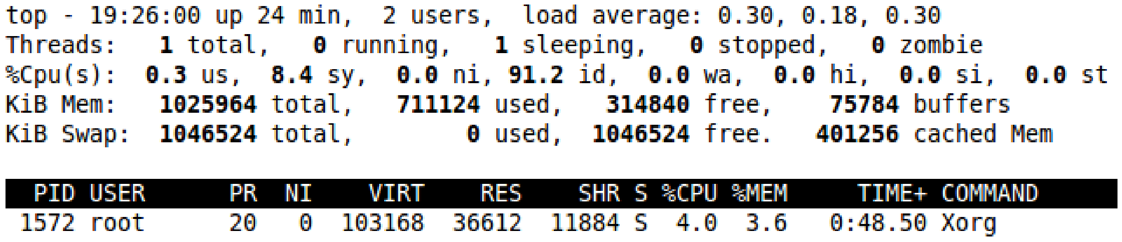
Example 1: top



```
top - 19:24:16 up 22 min,  2 users,  load average: 0.36, 0.17, 0.31
Threads: 312 total,   1 running, 311 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  6.6 sy,  0.0 ni, 91.1 id,  1.7 wa,  0.0 hi,  0.3 si,  0.0 st
KiB Mem:   1025964 total,   711340 used,   314624 free,    75752 buffers
KiB Swap:  1046524 total,        0 used,  1046524 free.   401252 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1572 root      20   0  103168  36612  11884 S  2.9  3.6   0:45.92 Xorg
 2664 root      20   0    5564   1568   1084 R  2.0  0.2   0:03.39 top
 2400 admincs   20   0  292884  22356  14896 S  0.7  2.2   0:22.41 mate-terminal
    4 root      20   0       0      0      0 S  0.3  0.0   0:03.78 kworker/0:0
    6 root      20   0       0      0      0 S  0.3  0.0   0:02.46 kworker/u2:0
    7 root      20   0       0      0      0 S  0.3  0.0   0:03.56 rcu_sched
```

- To restrict the top output to a particular process and check all threads running inside the process:

Example 2: top -H -p 1572



```
top - 19:26:00 up 24 min,  2 users,  load average: 0.30, 0.18, 0.30
Threads:   1 total,   0 running,   1 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.3 us,  8.4 sy,  0.0 ni, 91.2 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem:   1025964 total,   711124 used,   314840 free,    75784 buffers
KiB Swap:  1046524 total,        0 used,  1046524 free.   401256 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 1572 root      20   0  103168  36612  11884 S  4.0  3.6   0:48.50 Xorg
```

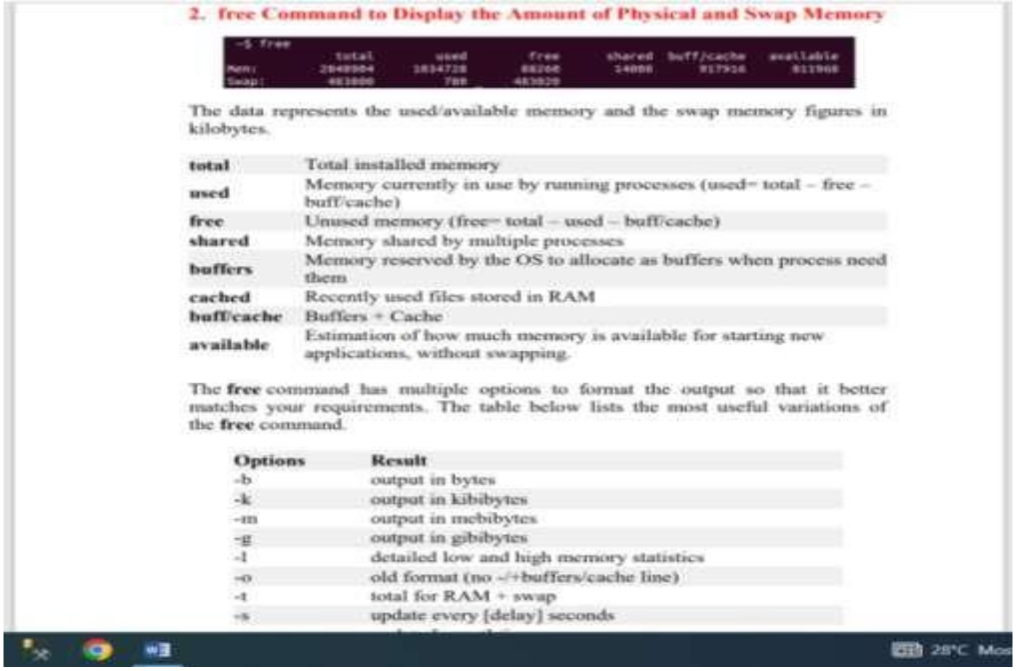
Week 6 Practice

Exp #1: commands to view memory consumption.

. cat Command to Show Linux Memory Information Entering cat /proc/meminfo in your terminal opens the /proc/meminfo file.

• This is a virtual file that reports the amount of available and used memory. • It contains real-time information about the system's memory usage as well as the buffers and shared memory used by the kernel.

• $ cat /proc/meminfo



2. free Command to Display the Amount of Physical and Swap Memory

```
~$ free
              total        used        free      shared  buff/cache   available
Mem:        2040984     1034728       88268       14088      917916      811968
Swap:        483800         780      483020
```

The data represents the used/available memory and the swap memory figures in kilobytes.

total	Total installed memory
used	Memory currently in use by running processes (used= total – free – buff/cache)
free	Unused memory (free= total – used – buff/cache)
shared	Memory shared by multiple processes
buffers	Memory reserved by the OS to allocate as buffers when process need them
cached	Recently used files stored in RAM
buff/cache	Buffers + Cache
available	Estimation of how much memory is available for starting new applications, without swapping.

The **free** command has multiple options to format the output so that it better matches your requirements. The table below lists the most useful variations of the **free** command.

Options	**Result**
-b	output in bytes
-k	output in kibibytes
-m	output in mebibytes
-g	output in gibibytes
-l	detailed low and high memory statistics
-o	old format (no -/+buffers/cache line)
-t	total for RAM + swap
-s	update every [delay] seconds

Week 7 Practice

Exp #1: write shell scripts to illustrate decision making and iteration statements.

```
# Find biggest of Three

numbersecho Enter Three

Numbers

read a b c


if (( $a>=$b && $a>=$c
            ))thenbig=$a
elif(( $b>=$a && $b>=$c
            ))thenbig=$b
elif(( $c>=$a && $c>=$b
            ))thenbig=$c
fi
echo Biggest among $a $b $c is $big



#  Display Numbers  in Matrix

Format # -e in echo is for Enabling

Back Slash# -n in echo is for

ignoring newline


read -p "Enter Rows and Columns: " r c


for((i=1; i<=r; i++))do
            for((j=1; j<= c;
             j++))doecho -n
             "$i "
         d
     one
     ech
     o
done
```

Week 7 Practice

Exp #2: write shell scripts to perform file operations.

Read the contents of a file and display the same on Screen

```
read -p "Enter file name : " filenamewhile read
line
do
        echo $linedone <
$filename
```

Copy the contents of a one file to another file (Demo cp command)

```
while read linedo
        echo $line >> outputdone <
$filename
echo ====== Contents of $filename =========
cat output
```

Week 9 Practice

Exp #1: test and manage network using

Ifconfig, ss, wget, ping, traceroute, netstat, nslookup, host, hostname, whois

1. ifconfig

Linux ifconfig stands for interface configurator. It is one of the most basic commands used in network inspection.

ifconfig is used to initialize an interface, configure it with an IP address, and enable or disable it. It is also used to display the route and the network interface.

Syntax:

```
ifconfig -a
ifconfig -s
ifconfig lo
```

- -a display all interfaces available even if they are down.
- -s display short list instead of details.
- lo- get details of local network

2. traceroute

Linux traceroute is one of the most useful commands in networking. It is used to troubleshoot the network. It detects the delay and determines the pathway to your target. It basically helps in the following ways:

1. It provides the names and identifies every device on the path.
2. It follows the route to the destination
3. It determines where the network latency comes from and reports it.

```
traceroute <destination>
```

If you don't have the traceroute service installed in your system, you can install it using the following command:

```
sudo apt-get install inetutils-traceroute
```

Example:

Command:

```
$ traceroute google.com
```

The output provides the following information:

1. The specified hostname
2. Size of the packets
3. The maximum number of hops required.
4. The IP address.

3. ping

Linux ping is one of the most used network troubleshooting commands. It basically checks for the network connectivity between two nodes.

ping stands for Packet INternet Groper.

The ping command sends the ICMP echo request to check the network connectivity.

It keeps executing until it is interrupted.

Use Ctrl+C Key to interrupt the execution.

Syntax:

```
ping <destination>
```

Example:

Command:

```
$ ping google.com
```

The ping shows a successful connection to google.com

4. netstat

Linux netstat command refers to the network statistics.

It provides statistical figures about different interfaces which include open sockets, routing tables, and connection information.

Syntax:

```
netstat -s
netstat -a
```

Output:

- -s it display statistics ports
- -a all listening and non-listening ports

5. ss

Linux ss command is the replacement for netstat command. It is regarded as a much faster and more informative command than netstat.

The faster response of ss is possible as it fetches all the information from within the kernel userspace.

Syntax:

```
ss
```

This command gives information about all TCP, UDP, and UNIX socket connections.

You can use -t, -u, -x in the command respectively to show TCP/UDP or UNIX sockets. You can combine each of these with "a" to show the connected and listening sockets.

Syntax:

```
ss -ta

ss -ua

ss -xa
```

6. nslookup

Linux nslookup is also a command used for DNS related queries. It is the older version of dig.

Syntax:

```
 nslookup <domainName>
```

Example:

```
nslookup google.com
```

Output:

As we see in the output above, it displays the record information relating to

```
google.com
```

7. host

Linux host command displays the domain name for a given IP address and IP address for a given hostname. It is also used to fetch DNS lookup for DNS related query.

Example:

```
host google.com

host 149.77.21.18
```

8. .hostname

Linux hostname is the simple command used to view and set the hostname of a system.

Syntax:

```
hostname
```

Output:

- To set the hostname

Use the syntax below to set the hostname.

Syntax:

```
 sudo hostname <newName>
```

9. wget

Linux curl and wget commands are used in downloading files from the internet through CLI. The curl command has to be used with the option "O" to fetch the file, while the wget command is used directly.

Syntax:

```
 wget <fileLink>
```

Example:

```
wget google.com/doodles/new-years-day-2012
```

10. whois

Linux whois command is used to fetch all the information related to a website. You can get all the information about a website including the registration and the owner information.

Syntax:

```
    whois <websiteName>
```

Example:

```
whois google.com
```

Week 10 Practice

Exp #1: work on user accounts, passwd.

Work on user accounts:

- A user is an entity, in a Linux operating system, that can manipulate files and perform several other operations. Each user is assigned an ID that is unique for each user in the operating system.
- After installation of the Linux operating system, the ID 0 is assigned to the root user and the IDs 1 to 999 (both inclusive) are assigned to the system users and hence the ID for local user begins from 1000 onwards.
- Now we will discuss the important commands to manage users in Linux.

useradd:

- useradd is a command in Linux that is used to add user accounts to your system.
- Only root or users with sudo privileges can use the useradd command to createnew user accounts.
- When we run the 'useradd' command in the Linux terminal, it performs the following major things:
  - ✓ It edits /etc/passwd, /etc/shadow, /etc/group and /etc/gshadow files for thenewly created user accounts.
  - ✓ Creates and populates a home directory for the new user.
  - ✓ Sets permissions and ownerships to the home directory.

Syntax:

The syntax of the useradd command is:

```
# useradd [options] [user name]
```

Important options used are:

- ✓ -m : Adds a home directory (this is a default on some distributions)
- ✓ -s : Sets the user's preferred shell if it's different from /bin/bash
- ✓ -u : Specifies a particular user ID (UID)
- ✓ -g: We can create users with specific group IDs as well with the -g option.

Example:
command

sudo su

```
[root@tecmint ~]# useradd -u 1002 navin
[root@tecmint ~]# cat /etc/passwd | grep navin
navin:x:1002:1002::/home/navin:/bin/bash
```

passwd:

- The passwd command changes passwords for user accounts.
- A normal user may only change the password for their own account, while the superuser may change the password for any account.
- passwd also changes the account or associated password validity period.

Syntax:

The syntax of the useradd command is:

```
        # passwd [options] [username]
```

- Important options used along with passwd:

✓ **-d, –delete:** This option deletes the user password and makes the accountpassword-less.

✓ **-e, –expire:** This option immediately expires the account password and forcesthe user to change password on their next login.

Example:

```
[root@tecmint ~]# passwd tecmint
Changing password for user tecmint.New
UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
```

<u>userdel:</u>

➢ userdel command in Linux system is used to delete a user account and related files.

➢ This command basically modifies the system account files, deleting all the entries which refer to the username LOGIN.

Syntax:

```
          # userdel [options] username
```

Important Options used along with userdel:

✓ **-f:** This option forces the removal of the specified user account. It doesn’t matter that the user is still logged in. It also forces the *userdel* to remove the user’s home directory and mail spool, even if another user is using the same home directory or even if the mail spool is not owned by the specified user.

✓ **-r:** Files in the user's home directory will be removed with the home directoryitself and the user's mail spool. Files located in other file systems have to be searched for and deleted manually.

Example:

```
gptcs@gptcs-VirtualBox:~$ cat /etc/passwd | tail -1
test_user:x:1001:1001::/home/test_user:/bin/sh
gptcs@gptcs-VirtualBox:~$ sudo userdel -f test_user
gptcs@gptcs-VirtualBox:~$ cat /etc/passwd | tail -1
openldap:x:128:136:OpenLDAP Server:/var/lib/ldap:/bin/false
```

Week 11 Practice

Exp #1: system monitoring and maintenance commands

System/Log monitoring Commands

top

- The top command has been around a long time and is very useful for viewing details of running processes and quickly identifying issues such as memory hogs. Its default view is shown below.

```
top - 11:56:28 up 1 day, 13:37,  1 user,  load average: 0.09, 0.04, 0.03
Tasks: 292 total,  3 running, 225 sleeping,  0 stopped,   0 zombie
%Cpu(s): 0.1 us,  0.2 sy,  0.0 ni, 99.7 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 16387132 total, 10854648 free, 1859036 used,  3673448 buff/cache
KiB Swap:        0 total,        0 free,        0 used. 14176540 avail Mem
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+
COMMAND
17270 alan      20   0 3930764 247288  98992 R   0.7  1.5   5:58.22 gnome-shell
20496 alan      20   0  816144  45416  29844 S   0.5  0.3   0:22.16 gnome-terminal-
21110 alan      20   0   41940   3988   3188 R   0.1  0.0   0:00.17 top
1 root      20   0  225564   9416   6768 S   0.0  0.1   0:10.72 systemd
2 root     20 0         0    0          0 S 0.0  0.0    0:00.01 kthreadd
4 root      0 -20       0    0          0 I 0.0  0.0    0:00.00 kworker/0:0H
6 root      0 -20       0    0          0 I 0.0  0.0    0:00.00 mm_percpu_wq
7 root     20 0         0    0          0 S 0.0  0.0    0:00.08 ksoftirqd/0
```

- The update interval can be changed by typing the letter **s** followed by the number of seconds the user prefers for updates.
- To make it easier to monitor required processes, user can call top and pass the PID(s) using the **-p** option.

df command:

- The **df** command (short for disk free), is used to display information related to file systems about total space and available space.
- **Syntax :** df [OPTION]... [FILE]...
- If no file name is given, it displays the space available on all currently mounted file systems.

Example 1: **df**

Portion of output:

```
Filesystem           1K-blocks        Used Available Use% Mounted onudev
                     3996816          0 3996816 0% /dev
tmpfs                804624      10020      794604 2% /run
/dev/sda9            68117056 18036160 46597712  28% /
```

Example 2: specify particular file, then it will show mount information of thatparticular file.

For example: **df /home/admincs/test/test.cpp**

Output:

```
Filesystem           1K-blocks        Used Available Use% Mounted on
/dev/sda10            78873504 67528220 7315640  91% /home
```

Free command:

- **free** command is used to view memory consumption
- **free -m**

```
total                          used        free      shared     buffers      cached
Mem:                 7976      6459        1517          0        865        2248
-/+ buffers/cache:              3344       4631
Swap:                1951          0       1951
```

- The m option displays all data in MBs.
- The total os 7976 MB is the total amount of RAM installed on the system, thatis 8GB.
- The used column shows the amount of RAM that has been used by linux, in this case around 6.4 GB. The second line tells that 4.6 GB is free. This is the free

  memory in first line added with the buffers and cached amount of memory (1517 + 865 + 2248 around 4631 MB= 4.6GB).
- Linux has the habit of caching lots of things for faster performance, so that memory can be freed and used if needed.
- The last line is the swap memory, which in this case is lying entirely free

System maintenance commands

Shutdown

- "**Shutdown**" refers to the process of stopping and shutting down a computer or server.
- This involves cutting the power to the main components of the system using a controlled process.
- Applications are closed, active processes and protocols are saved to the hard drive, device drivers are removed, and user settings are saved in the process.
- Linux operating systems can easily be stopped, shut down, and restarted using the shutdown command and its various options.
- **Standard command for shutting down Linux**
  - **shutdown –h**
    - Linux will shut down in under a minute. The "-h" option explicitly stands for the shutting down or powering off of a system.
  - **shutdown**
    - User can usually produce the same results by just entering the shutdown command on its own.
- **Standard command for restarting Linux**
  - **shutdown -r**
    - Linux will be restarted in under a minute.
    - The "-r" option stands for reboot or restart.
- **Command for shutting down Linux immediately**
  - **shutdown -h 0** **// time Specification 0**
  - **shutdown now**
    - Another common command for shutting down Linux immediately:
- **Command for restarting Linux immediately**
  - **shutdown -r 0** **// time Specification 0**
  - **shutdown -r now**

- **Command for Shutting Down/Restart Linux after 20 minutes**
  - shutdown -h 20
  - shutdown +20
  - shutdown -r 20
  - shutdown –r +20
  - shutdown -h 17:30 // Shutting down at 5.30pm
  - shutdown -r 17:30 // Restarting at 5.30pm

reboot :

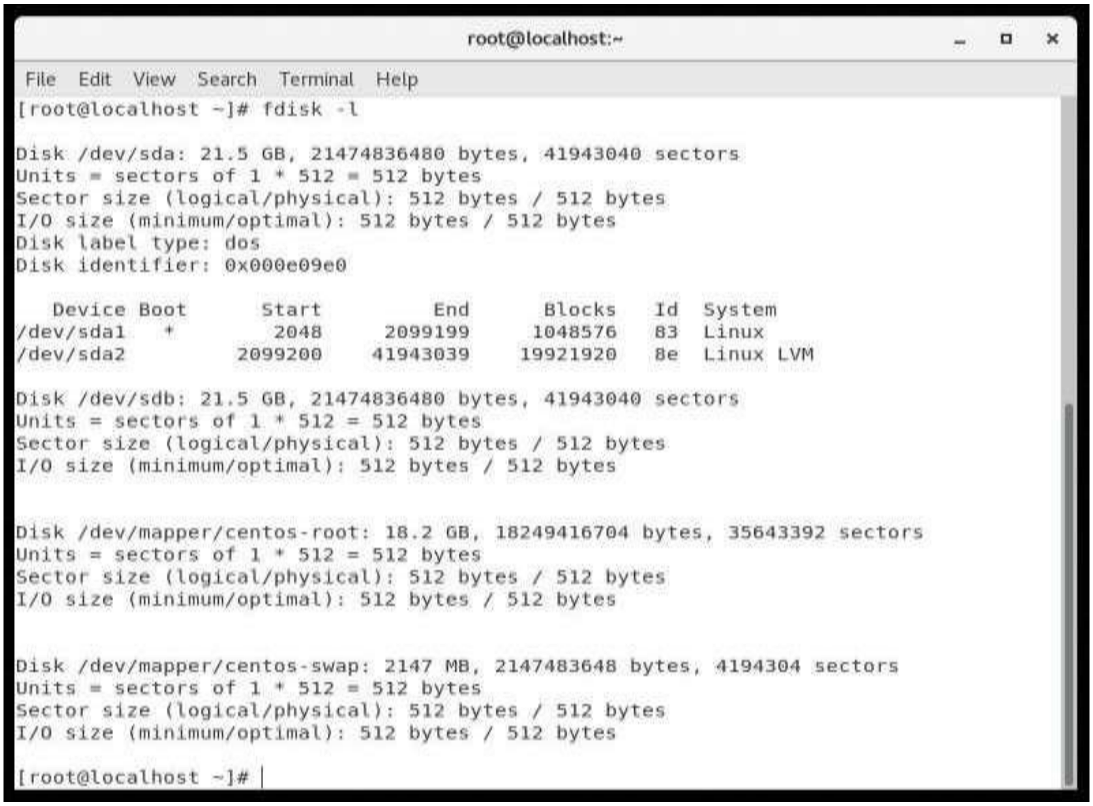
- Booting is **starting a computer's operating system**, so rebooting is to start itfor a second or third time.
- Rebooting is usually necessary after a computer crashes, meaning it stops working because of a malfunction.
- **sudo reboot**
- **sudo systemctl reboot**
- **sudo shutdown -r**

Week 13 Practice

Exp #1: basic commands for storage partitions.

fdisk

- fdisk is used to check the partitions on a disk.
- The fdisk command can display the partitions and details like file system type.
- However it does not report the size of each partitions.
    - **Sudo fdisk -l**

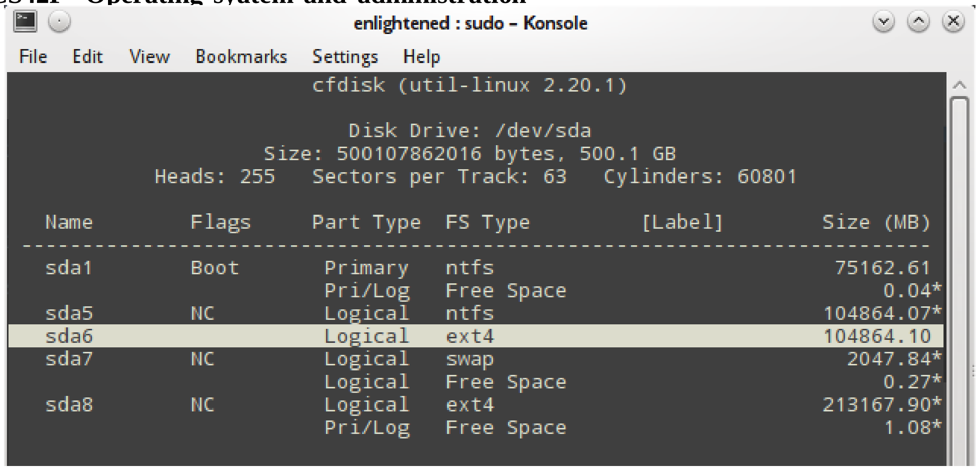


```
[root@localhost ~]# fdisk -l

Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 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 label type: dos
Disk identifier: 0x000e09e0

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048     2099199     1048576   83  Linux
/dev/sda2         2099200    41943039    19921920   8e  Linux LVM

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 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/mapper/centos-root: 18.2 GB, 18249416704 bytes, 35643392 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/mapper/centos-swap: 2147 MB, 2147483648 bytes, 4194304 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

[root@localhost ~]#
```

- Each device is reported separately with details about size, seconds, id and individual partitions.

cfdisk

- cfdisk is a linux partition editor with an interactive user interface based on **ncurses**.
- It can be used to list out the existing partitions as well as create or modify them.
- An example of how to use cfdisk to list the partitions.
    - sudo cfdisk


enlightened : sudo – Konsole
File Edit View Bookmarks Settings Help

```
                        cfdisk (util-linux 2.20.1)

                           Disk Drive: /dev/sda
                     Size: 500107862016 bytes, 500.1 GB
            Heads: 255   Sectors per Track: 63   Cylinders: 60801

   Name        Flags     Part Type  FS Type          [Label]        Size (MB)
 ------------------------------------------------------------------------------
   sda1        Boot       Primary   ntfs                             75162.61
                          Pri/Log   Free Space                           0.04*
   sda5        NC         Logical   ntfs                           104864.07*
   sda6                   Logical   ext4                           104864.10
   sda7        NC         Logical   swap                             2047.84*
                          Logical   Free Space                           0.27*
   sda8        NC         Logical   ext4                           213167.90*
                          Pri/Log   Free Space                           1.08*
```

- cfdisk works with one partition at a time.
  - So if user need to see the details of a particular disk, then pass the device name tcfdisk.
  - **Sudo cfdisk /dev/sdb**

parted

- partedutility is to list out partitions and modify them if needed.
  - **sudo parted -l**

lsblk

- To view the available hard disk in our system, use the command
  - lsblk
  - cat /proc/partitions

blkid

- Prints the block device (partitions and storage media) attributes like uuid and file system type. Does not report the space on the partitions.
  - **Sudo blkid**

Week 13 Practice

Exp #2: install and configure LVM

INSTALL AND CONFIGURE LVM

- Logical Volume Manager (LVM) is used on Linux to manage hard drivesand other storage devices.
- To install LVM on Ubuntu, Debian, and Linux Mint, use the following command,

  sudo apt install lvm2 **or** sudo dnf install lvm2

- LVM has three terms, Physical Volume PV, Volume Group VG, Logical Volume LV.
  1. PV – It's a raw hard drive that it initialized to work with LVM, such as /dev/sdb, /dev/sdc, /dev/sdb1 etc.
  2. VG – Many PV is combined into one VG. You can createmany VGs and each of them has a unique name.
  3. LV – You can create many LVs from a VG. You can extend, reduce the LV size on the fly.The LV also has unique names.

     You can format the LV into ext4, zfs, btrfs etc filesystems, mount it and use it as you do other ordinarypartitions.

Steps to Install LVM on Ubuntu

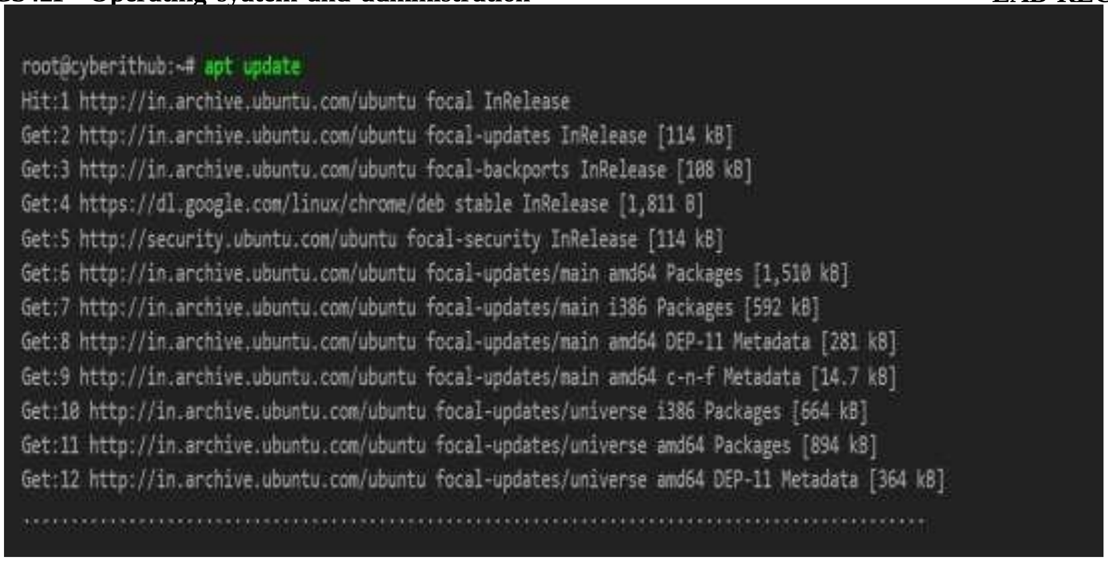
Step 1: Prerequisites

1. Running Ubuntu 20.04 LTS Server.
2. sudo or root access to run privileged command.
3. apt or apt-get utility available in the System.

Step 2: Update the Server

a. Before installing lvm2 package, first sync your package cache with the all enabled source repository using **apt update** command.

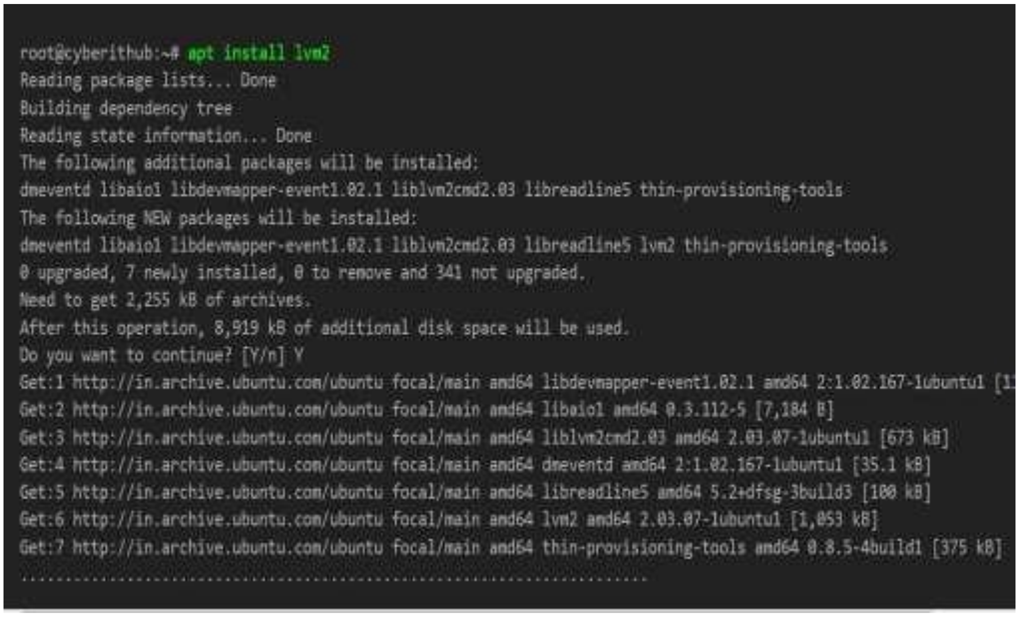
b. If any package needs to be upgraded, then upgrade it by using **apt upgrade** command.



```
root@cyberithub:~# apt update
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:4 https://dl.google.com/linux/chrome/deb stable InRelease [1,811 B]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1,510 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [592 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [281 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [14.7 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/universe i386 Packages [664 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [894 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [364 kB]
..............................................................................................
```

Step 3: Install LVM Utilities

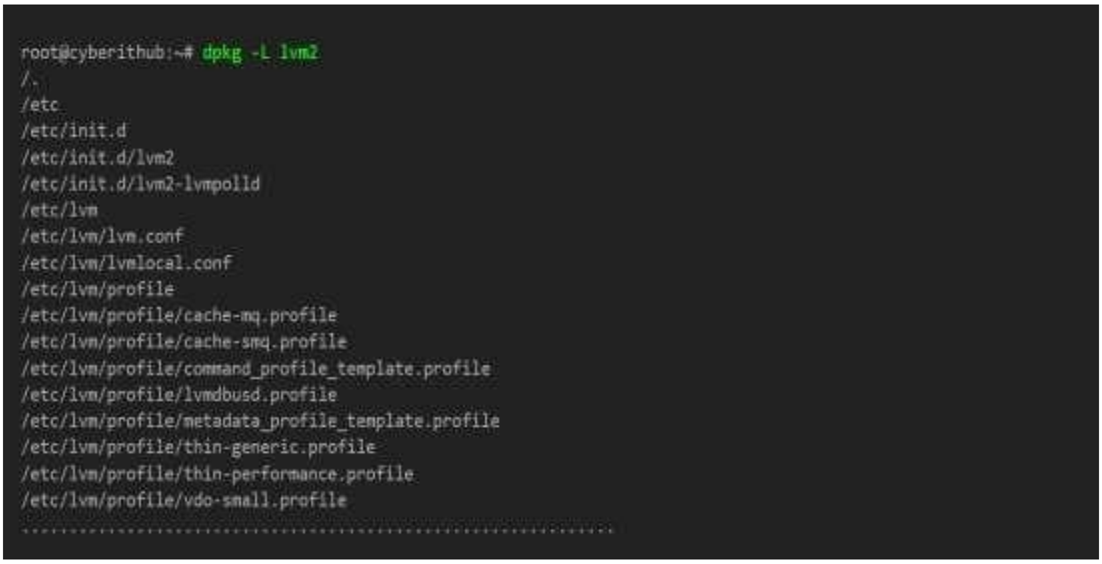
Next step is to install LVM utilities using **apt install lvm2** command as shown below.This will install lvm2 package along with all its required dependencies



```
root@cyberithub:~# apt install lvm2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
dmeventd libaio1 libdevmapper-event1.02.1 liblvm2cmd2.03 libreadline5 thin-provisioning-tools
The following NEW packages will be installed:
dmeventd libaio1 libdevmapper-event1.02.1 liblvm2cmd2.03 libreadline5 lvm2 thin-provisioning-tools
0 upgraded, 7 newly installed, 0 to remove and 341 not upgraded.
Need to get 2,255 kB of archives.
After this operation, 8,919 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://in.archive.ubuntu.com/ubuntu focal/main amd64 libdevmapper-event1.02.1 amd64 2:1.02.167-1ubuntu1 [1
Get:2 http://in.archive.ubuntu.com/ubuntu focal/main amd64 libaio1 amd64 0.3.112-5 [7,184 B]
Get:3 http://in.archive.ubuntu.com/ubuntu focal/main amd64 liblvm2cmd2.03 amd64 2.03.07-1ubuntu1 [673 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu focal/main amd64 dmeventd amd64 2:1.02.167-1ubuntu1 [35.1 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu focal/main amd64 libreadline5 amd64 5.2+dfsg-3build3 [100 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal/main amd64 lvm2 amd64 2.03.07-1ubuntu1 [1,053 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu focal/main amd64 thin-provisioning-tools amd64 0.8.5-4build1 [375 kB]
.........................................................................
```

Step 4: Verify Package Installation

You can verify the installation path and list of utilities available through lvm2package using **dpkg -L lvm2 command** as shown below



```
root@cyberithub:~# dpkg -L lvm2
/.
/etc
/etc/init.d
/etc/init.d/lvm2
/etc/init.d/lvm2-lvmpolld
/etc/lvm
/etc/lvm/lvm.conf
/etc/lvm/lvmlocal.conf
/etc/lvm/profile
/etc/lvm/profile/cache-mq.profile
/etc/lvm/profile/cache-smq.profile
/etc/lvm/profile/command_profile_template.profile
/etc/lvm/profile/lvmdbusd.profile
/etc/lvm/profile/metadata_profile_template.profile
/etc/lvm/profile/thin-generic.profile
/etc/lvm/profile/thin-performance.profile
/etc/lvm/profile/vdo-small.profile
..............................................................
```

Week 13 Practice
Exp #3: Working with LVM.

- Scan the system for block devices that LVM can see and manage.

```
Sudo lvmdiskscan
```

Output:

```
  /dev/ram0  [      64.00 MiB]
  /dev/ram1  [      64.00 MiB]
  . . .
  /dev/ram15 [      64.00 MiB]
/dev/sdb   [     100.00 GiB]                           // physical Device
1 disks
  17 partitions
  0 LVM physical volume whole disks
  0 LVM physical volumes
```

- The partitions are mostly **/dev/ram*** partitions that are used the system as a Ram disk for performance enhancements. The disks in this example is**/dev/sdb**, which has 100G of space.

- Now mark the **physical device**as **physical volumes** within LVM using the `pvcreate` command:

```
Sudo pvcreate /dev/sdb
```

S