

NAME: Kadiyala Sri Pavan

UNIVERSITY: Aditya University

EMAIL: pavancc1478@gmail.com

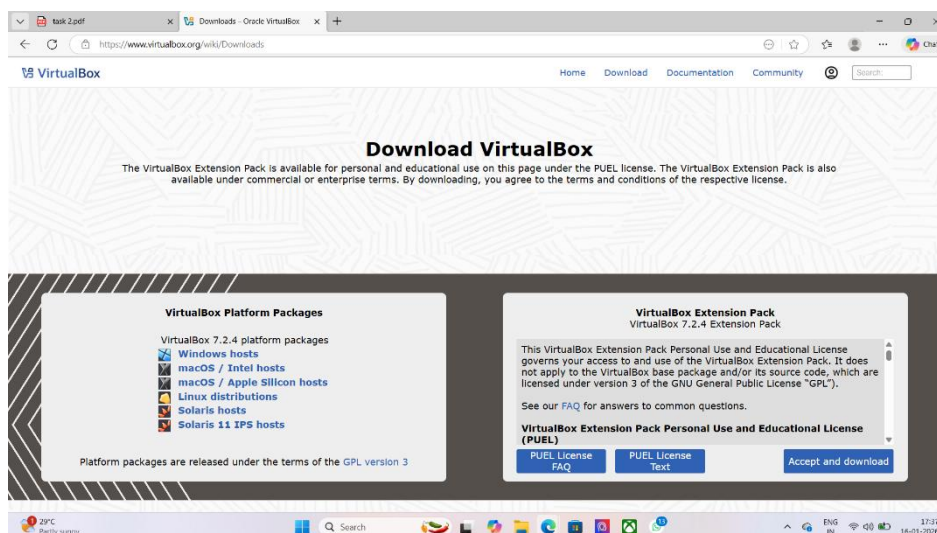
Task-2

Operating System Security Fundamentals (Linux/Windows)

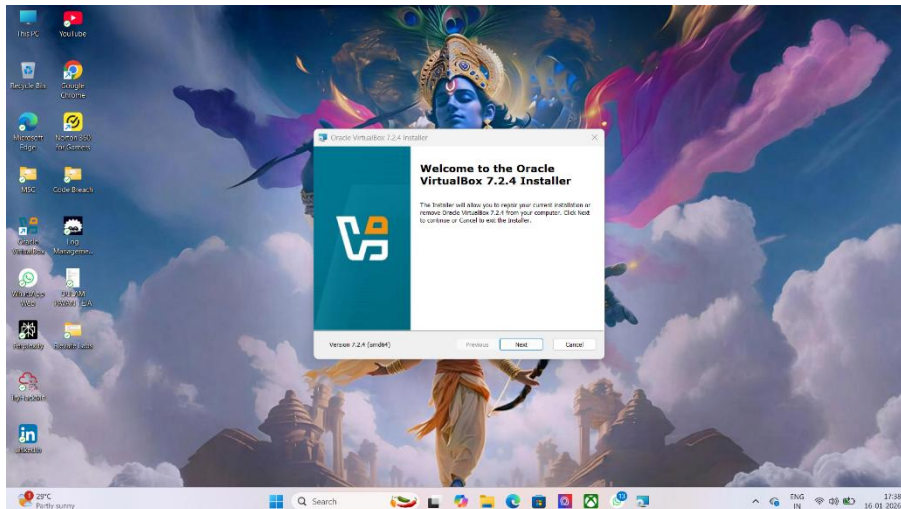
Step-1 Downloading VM & installing Kali Linux

- Go to the official website Oracle Virtual Machine.

Downloads – Oracle VirtualBox



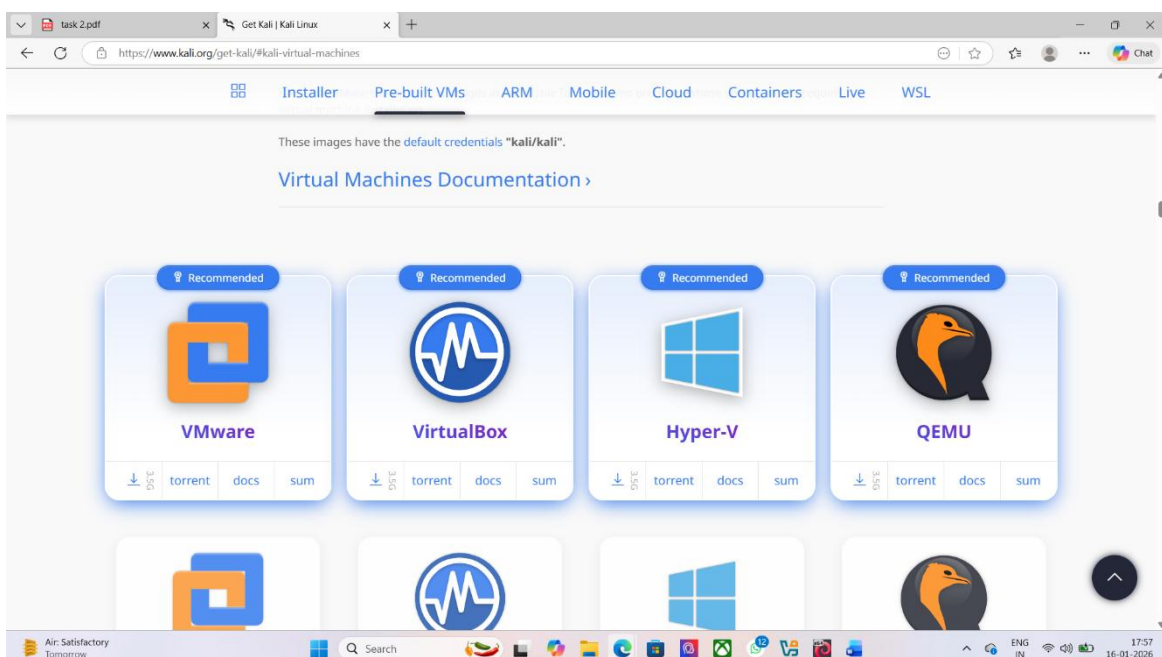
- Click on -> Window Host and it will Download start.
- After download we have to install VM in windows.



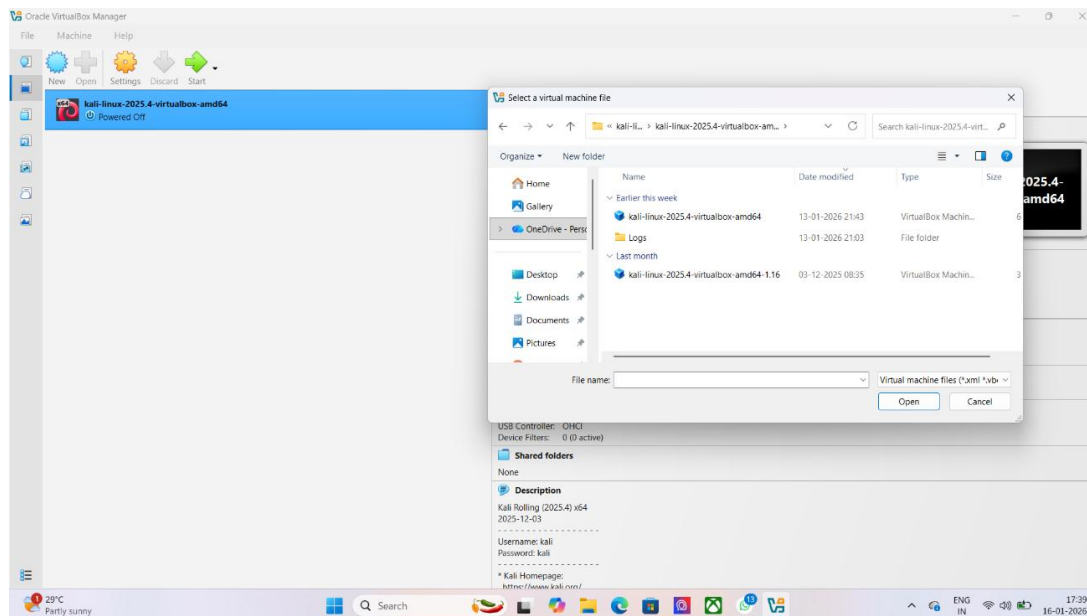
- After completing the VM install we have to download Kali Linux from official website.

[Get Kali | Kali Linux](https://www.kali.org/get-kali/)

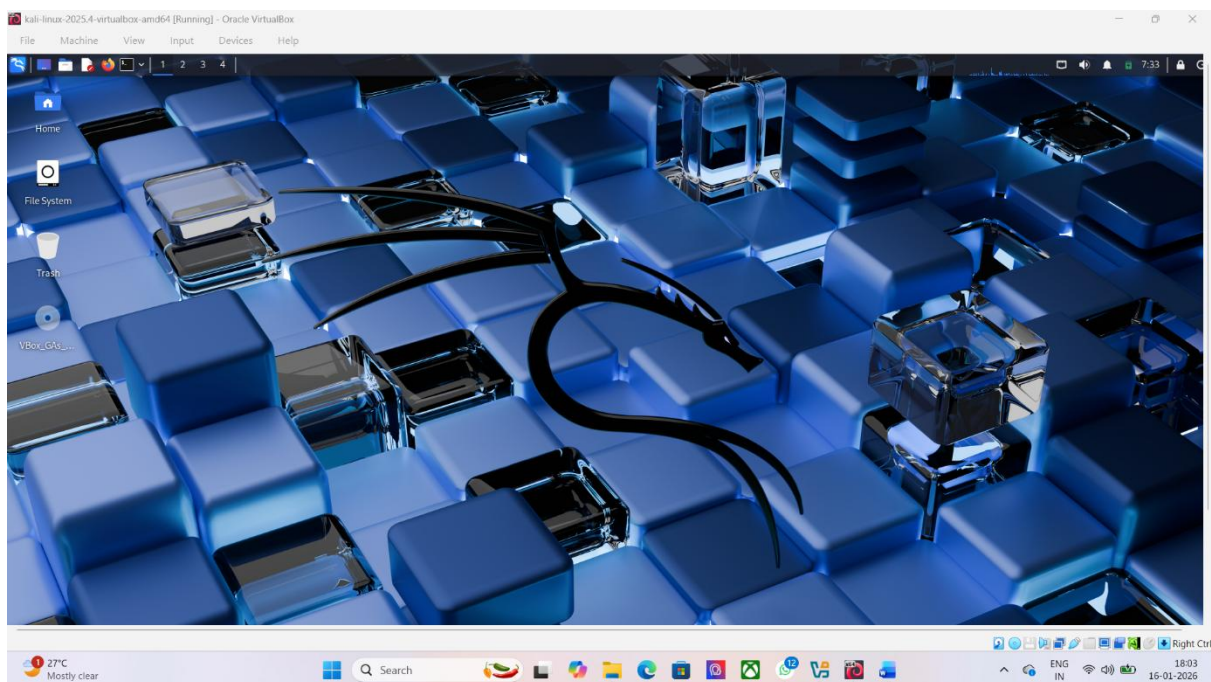
- Then we click on -> Pre-built VMs in top bar.
- Select the Virtual Box and download it.



- After completing downloading the kali Linux.
- open VM and click on -> Open and select the Kali Linux image file

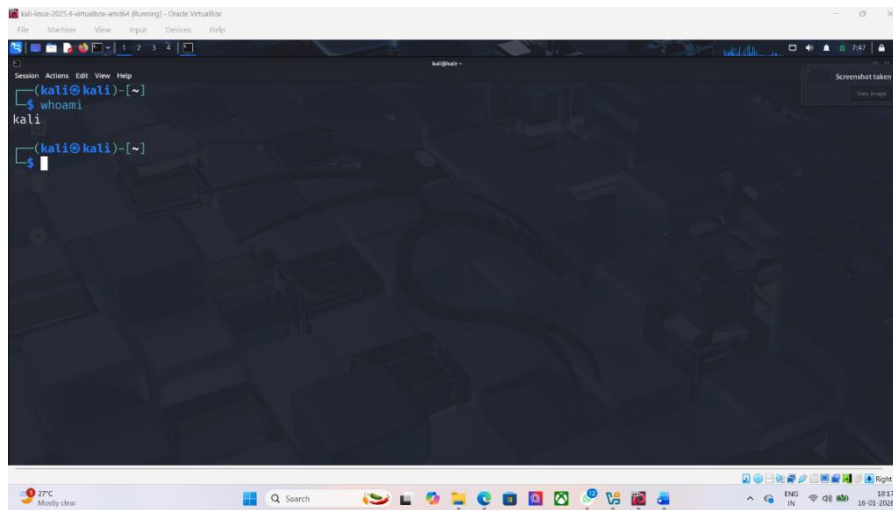


- Then we RUN the Kali click on -> Start.
- We can see the interface.

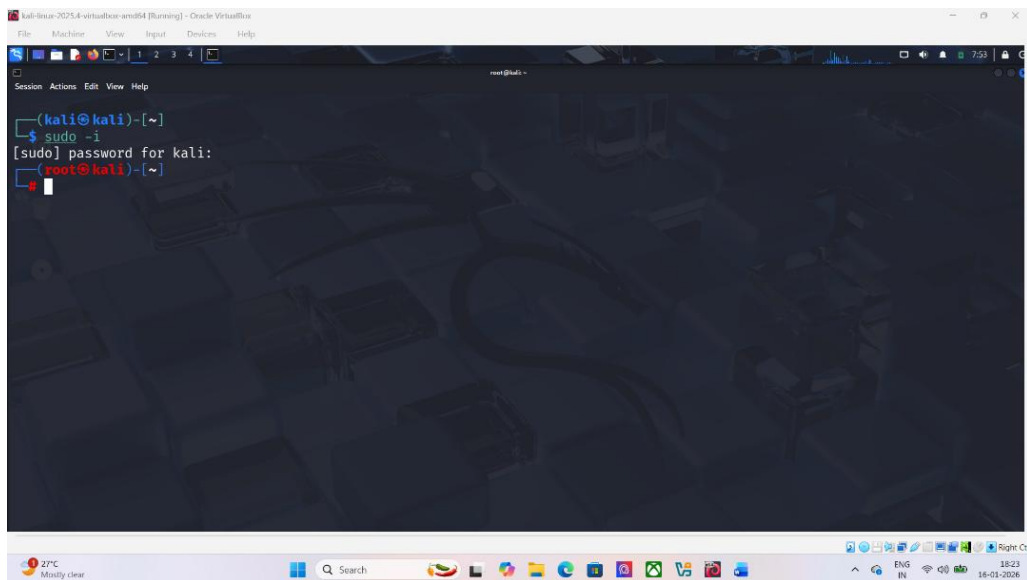


Step-2: User Accounts & Access Control.

- Open the CMD and check the current user using the Command -> whoami

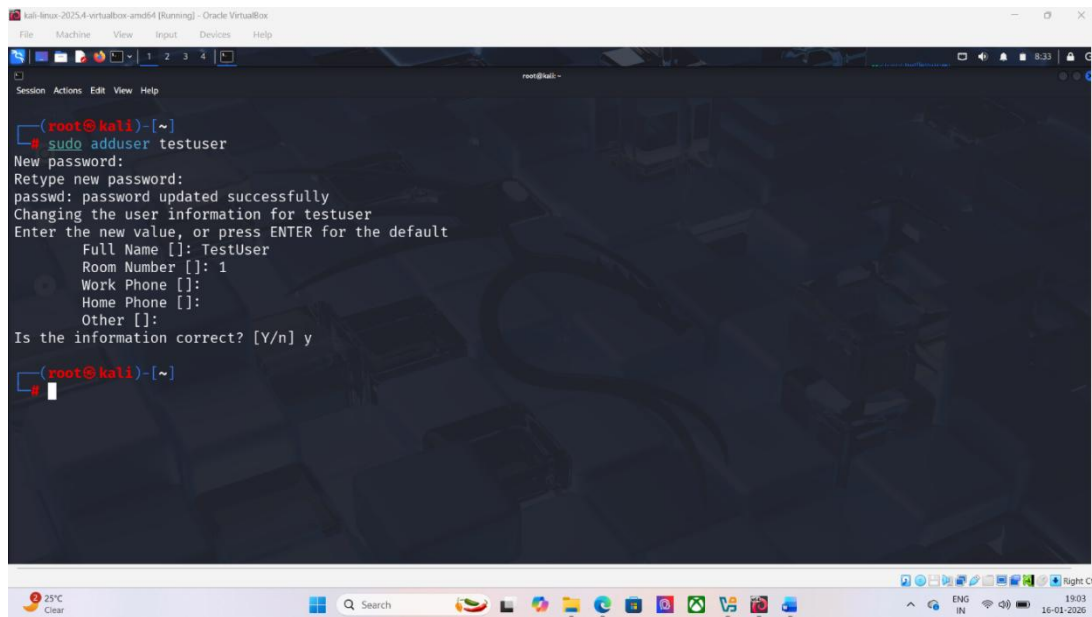


- Check the sudo privileges using the Command -> sudo -l
- Here Sudo is NOT use directly.
- Sudo is used for admin tasks.



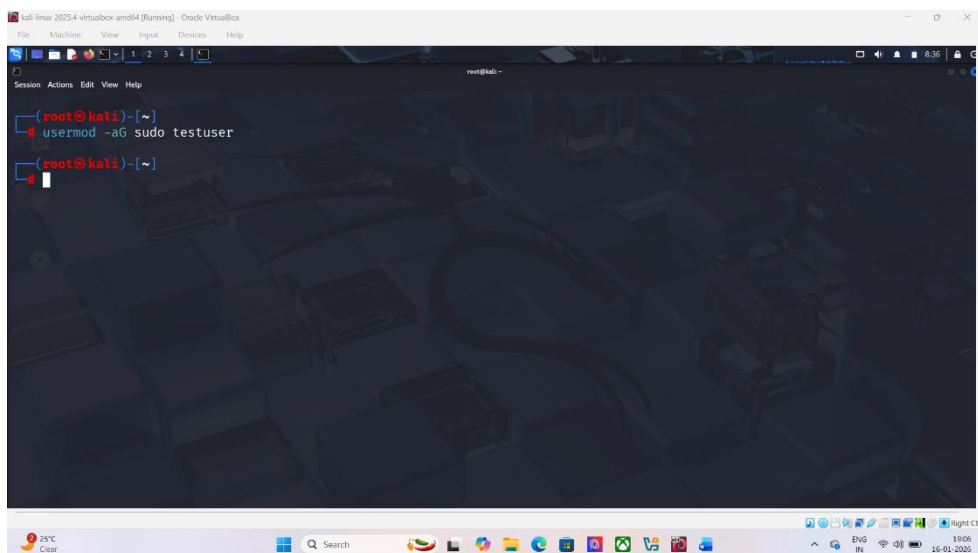
Step-3 Create a Standard User.

- Create a New User by using Command -> sudo adduser testuser



```
(root@kali)-[~]
# sudo adduser testuser
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for testuser
Enter the new value, or press ENTER for the default
Full Name []: TestUser
Room Number []: 1
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] y
(root@kali)-[~]
#
```

- Give the sudo access to testuser.



```
(root@kali)-[~]
# usermod -sG sudo testuser
(root@kali)-[~]
#
```

Step-4: File Permissions (Create File, Change Permission & Change ownership).

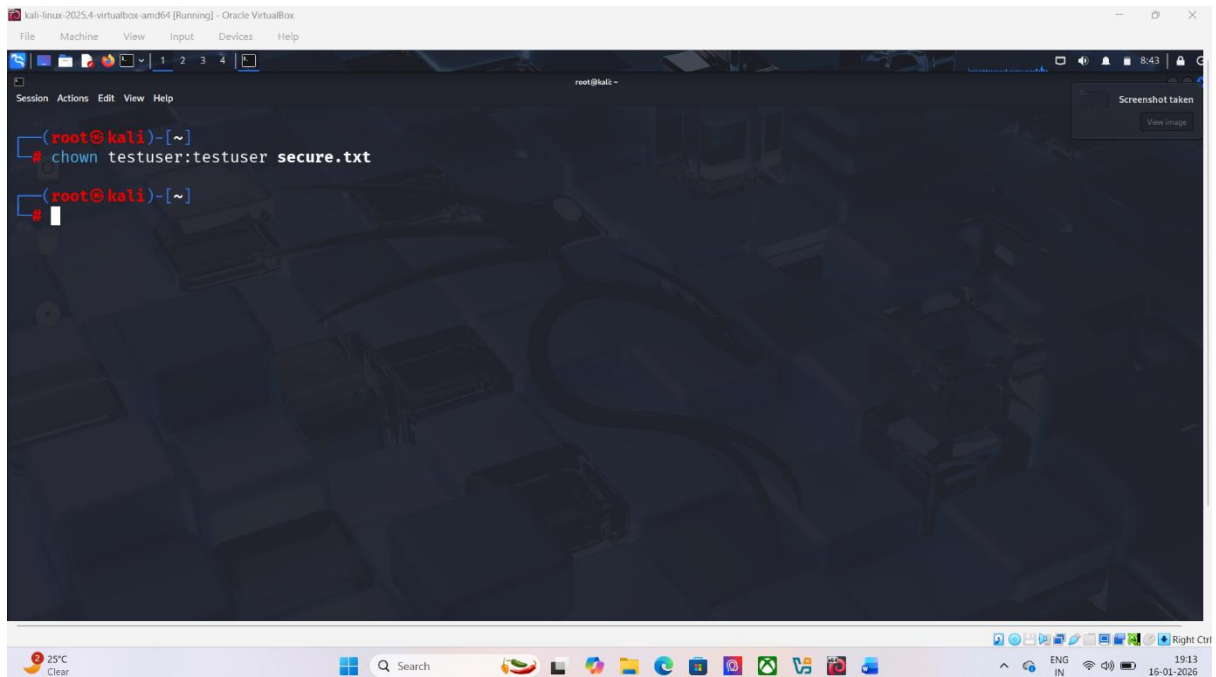
- Create a File using the Command -> touch secure.txt, ls -l secure.txt.


```
(kali@kali)-[~]
$ sudo -i
[sudo] password for kali:
(root@kali)-[~]
# touch secure.txt
(root@kali)-[~]
# ls -l secure.txt
-rw-r--r-- 1 root root 0 Jan 16 08:01 secure.txt
#
```

- Now Change the Permission of the file bu using the commend -> chmod 600 secure.txt, ls -l secure.txt

```
(root@kali)-[~]
# chmod 600 secure.txt
(root@kali)-[~]
# ls -l secure.txt
-rw----- 1 root root 0 Jan 16 08:01 secure.txt
#
```

- Now, Change the Ownership of the secure.txt File using the command -> `sudo chown testuser:testuser secure.txt`



The screenshot shows a terminal window titled 'kali-linux-2025.4-virtualbox-amd64 [Running] - Oracle VM VirtualBox'. The terminal prompt is '(root@kali)~'. The command 'chown testuser:testuser secure.txt' has been entered and executed. The output shows the prompt again, indicating the command was successful. A 'Screenshot taken' notification is visible in the top right corner of the terminal window. The Windows taskbar is visible at the bottom of the screen.

Step-5 Administrator vs Standard User:

Administrator (Root User):

- Administrator in Linux is called root
- Has full control over the operating system
- Can:
 - Install / remove software
 - Change system & security settings
 - Modify system files (/etc, /usr)
 - Create, delete, and manage users
 - Enable/disable services and firewall
- Commands run without restriction

Security Risk:

- If malware or attacker gets root access → complete system compromise
- That's why direct root usage is not recommended for daily work.

Standard User (Normal User) :

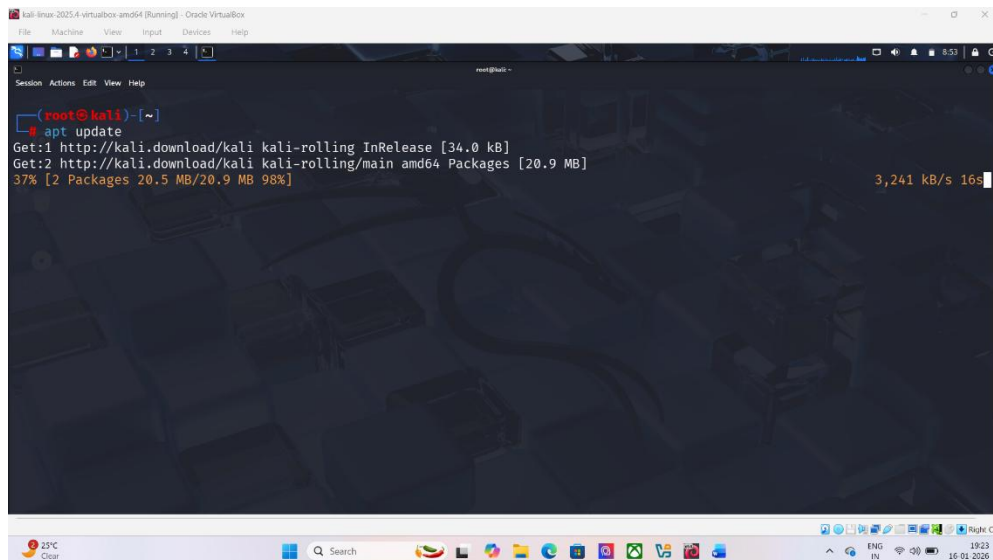
- Limited privileges
- Used for daily activities
- Cannot:
 - Change system files
 - Install software
 - Modify security settings
- Needs sudo password to perform admin tasks

Security Benefit:

- Reduces damage if account is compromised
- Follows Principle of Least Privilege.

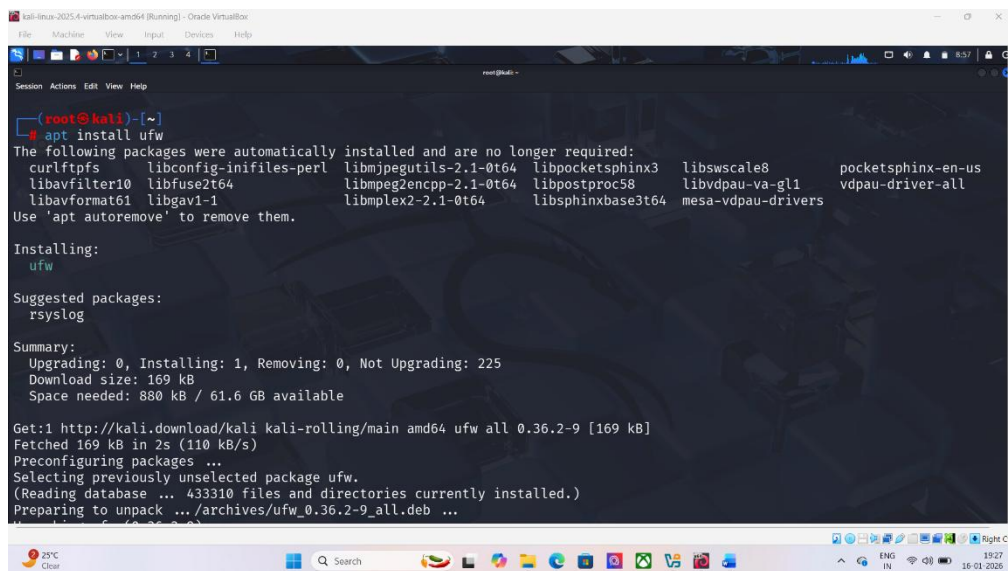
Step-6 Firewall Configuration (Kali UFW).

- Before installing the UFW, We have to update Kali using the command -> apt update



```
(root@kali)~# apt update
Get:1 http://kali.download/kali kali-rolling InRelease [34.0 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.9 MB]
37% [2 Packages 20.5 MB/20.9 MB 98%] 3,241 kB/s 16s
```

- Now, install the UFW using command -> apt install ufw



```
(root@kali)~# apt install ufw
The following packages were automatically installed and are no longer required:
curlftpfs libconfig-inifiles-perl libjpegutils-2.1-0t64 libpocketsphinx3 libswscale8 pocketsphinx-en-us
libavfilter10 libfuse2t64 libmpeg2encpp-2.1-0t64 libpostproc58 libvdpau-va-gl1 libvdpau-driver-all
libavformat61 libgav1-1 libmpeg2-2.1-0t64 libsphinxbase3t64 mesa-va-pau-drivers
Use 'apt autoremove' to remove them.

Installing:
ufw

Suggested packages:
rsyslog

Summary:
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 225
Download size: 169 kB
Space needed: 880 kB / 61.6 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 ufw all 0.36.2-9 [169 kB]
Fetched 169 kB in 2s (110 kB/s)
Preconfiguring packages ...
Selecting previously unselected package ufw.
(Reading database ... 433310 files and directories currently installed.)
Preparing to unpack .../archives/ufw_0.36.2-9_all.deb ...
```

- Then, we Enable the Firewall using the command -> ufw enable.



```
(root@kali)~# ufw enable
Firewall is active and enabled on system startup

(root@kali)~#
```

- We can check the status of the firewall is active or not using the command -> ufw status

```
(root@kali)-[~]
# ufw status
Status: active

(root@kali)-[~]
#
```

Step-7 Identify running processes and services.

- We can observe which processes are running in the kali by using the command -> ps aux, top

```
(root@kali)-[~]
# ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.7	25200	15448	?	Ss	07:10	0:01	/sbin/init splash
root	2	0.0	0.0	0	0	?	S	07:10	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	07:10	0:00	[pool_workqueue_release]
root	4	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-rcu_gp]
root	5	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-sync_wq]
root	6	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-kvfree_rcu_reclaim]
root	7	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-slub_flushwq]
root	8	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-netns]
root	10	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/0:0H-events_highpri]
root	12	0.0	0.0	0	0	?	I	07:10	0:00	[kworker/u8:0-ipv6_addrconf]
root	13	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-mm_percpu_wq]
root	14	0.0	0.0	0	0	?	S	07:10	0:01	[ksoftirqd/0]
root	15	0.1	0.0	0	0	?	I	07:10	0:12	[rcu_preempt]
root	16	0.0	0.0	0	0	?	S	07:10	0:00	[rcu_exp_par_gp_kthread_worker/0]
root	17	0.0	0.0	0	0	?	S	07:10	0:00	[rcu_exp_gp_kthread_worker]
root	18	0.0	0.0	0	0	?	S	07:10	0:00	[migration/0]
root	19	0.0	0.0	0	0	?	S	07:10	0:00	[idle_inject/0]
root	20	0.0	0.0	0	0	?	S	07:10	0:00	[cpuhp/0]
root	21	0.0	0.0	0	0	?	S	07:10	0:00	[cpuhp/1]
root	22	0.0	0.0	0	0	?	S	07:10	0:00	[idle_inject/1]
root	23	0.0	0.0	0	0	?	S	07:10	0:01	[migration/1]
root	24	0.1	0.0	0	0	?	S	07:10	0:10	[ksoftirqd/1]
root	28	0.0	0.0	0	0	?	I	07:10	0:00	[kworker/u10:0-events_unbound]
root	31	0.0	0.0	0	0	?	S	07:10	0:00	[kdevtmpfs]
root	32	0.0	0.0	0	0	?	I<	07:10	0:00	[kworker/R-inet_frag_wq]
root	33	0.0	0.0	0	0	?	I	07:10	0:00	[rcu_tasks_kthread]

“ps aux” is used to display detailed information about all running processes in the system.

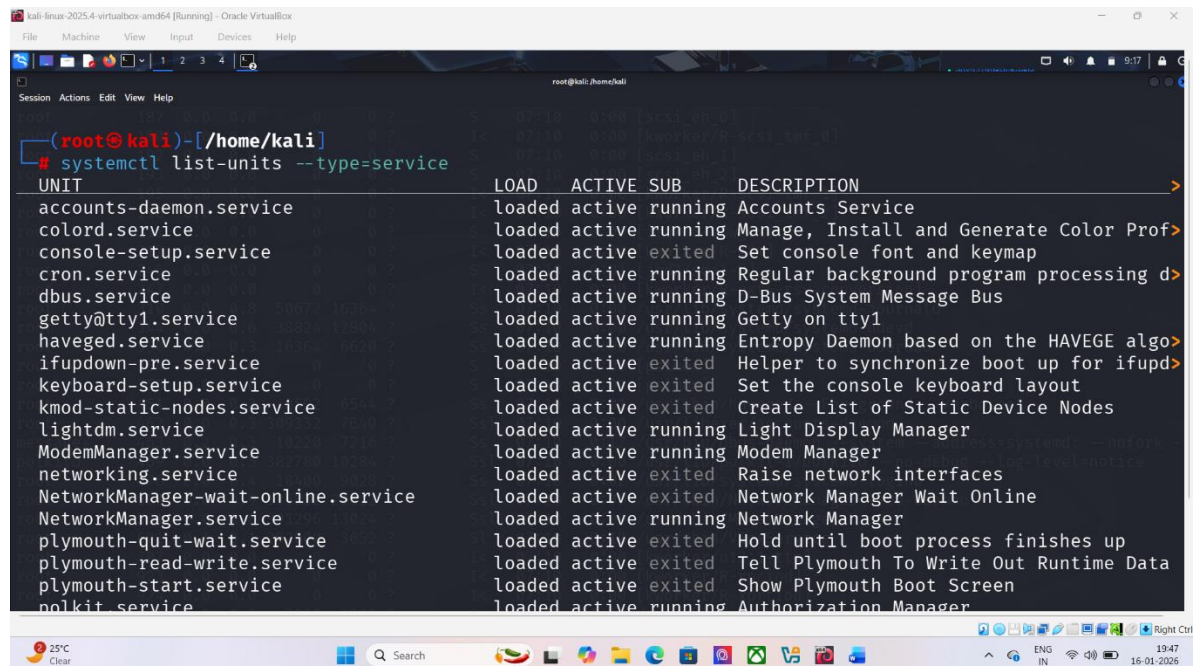
```
(root@kali)-[~]
# top
```

top - 09:09:02 up 1:58, 1 user, load average: 0.24, 0.19, 0.21
Tasks: 170 total, 1 running, 168 sleeping, 0 stopped, 1 zombie
%Cpu(s): 1.6 us, 1.6 sy, 0.0 ni, 96.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1972.6 total, 377.6 free, 876.3 used, 914.5 buff/cache
MiB Swap: 953.7 total, 953.7 free, 0.0 used, 1096.3 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1507	root	20	0	522832	199456	92136	S	2.0	9.9	4:26.99	Xorg
1894	kali	20	0	296848	49156	21920	S	0.7	2.4	1:48.94	wrapper-2.0
19413	kali	20	0	660128	72052	56012	S	0.7	3.6	0:10.12	qterminal
60549	root	20	0	10696	5892	3760	R	0.7	0.3	0:00.09	top
15	root	20	0	0	0	0	I	0.3	0.0	0:12.73	rcu_preempt
1851	kali	20	0	567656	138724	96400	S	0.3	6.9	1:25.97	xfwm4
1896	kali	20	0	273056	29972	22844	S	0.3	1.5	1:06.65	wrapper-2.0
58708	root	20	0	0	0	0	I	0.3	0.0	0:00.66	kworker/1:1-mm_percpu_wq
1	root	20	0	25200	15448	11316	S	0.0	0.8	0:01.68	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-sync_wq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-kvfree_rcu_reclaim
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-slub_flushwq
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-netns
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-events_highpri
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	kworker/u8:0-ipv6_addrconf
13	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-mm_percpu_wq

Here “top” commend is used to monitor running processes in real time.

- Now, we can observe the which “services” are running in the kali by using the commend -> systemctl list-units – type=service.

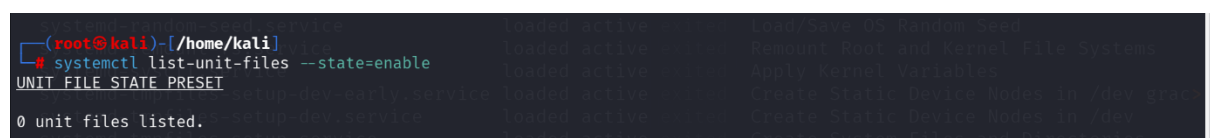


```
(root@kali)-[/home/kali]
# systemctl list-units --type=service
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
accounts-daemon.service	loaded	active	running	Accounts Service
colord.service	loaded	active	running	Manage, Install and Generate Color Prof
console-setup.service	loaded	active	exited	Set console font and keymap
cron.service	loaded	active	running	Regular background program processing d
dbus.service	loaded	active	running	D-Bus System Message Bus
getty@tty1.service	loaded	active	running	Getty on tty1
haveged.service	loaded	active	running	Entropy Daemon based on the HAVEGE algo
ifupdown-pre.service	loaded	active	exited	Helper to synchronize boot up for ifupd
keyboard-setup.service	loaded	active	exited	Set the console keyboard layout
kmmod-static-nodes.service	loaded	active	exited	Create List of Static Device Nodes
lightdm.service	loaded	active	running	Light Display Manager
ModemManager.service	loaded	active	running	Modem Manager
networking.service	loaded	active	exited	Raise network interfaces
NetworkManager-wait-online.service	loaded	active	exited	Network Manager Wait Online
NetworkManager.service	loaded	active	running	Network Manager
plymouth-quit-wait.service	loaded	active	exited	Hold until boot process finishes up
plymouth-read-write.service	loaded	active	exited	Tell Plymouth To Write Out Runtime Data
plymouth-start.service	loaded	active	exited	Show Plymouth Boot Screen
polkit.service	loaded	active	running	Authorization Manager

Step-8 Disable Unnecessary Services reduce attack surface.

- First we have to check which services are enabled by using the commend -> systemctl list-unit-files – state=enabled .



```
(root@kali)-[/home/kali]
# systemctl list-unit-files --state=enable
```

UNIT	FILE	STATE	PRESET
0 unit files listed.			

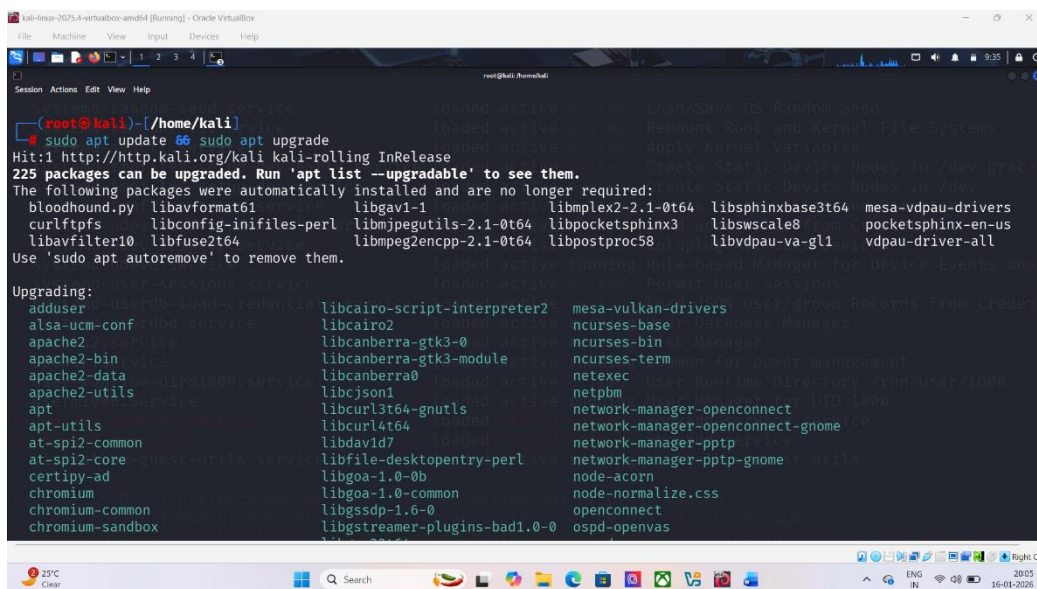
Here 0 services are running

Step-9 OS Hardening practices.

- OS Hardening means securing an operating system by reducing Vulnerabilities and attack surface through multiple security measures.
 - Installing security patches
 - Fixing known Vulnerabilities
 - Updating kernel and software

By using the command -> `sudo apt update && sudo apt upgrade`.

We can update and secure the system.



```
kali-linux-2025.4-vmtoolsd-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@kali: ~# sudo apt update && sudo apt upgrade
Hit:1 http://http.kali.org/kali kali-rolling InRelease
225 packages can be upgraded. Run 'apt list --upgradable' to see them.
The following packages were automatically installed and are no longer required:
bloodhound.py libavformat61 libgav1-1 libmplex2-2.1-0t64 libsphinxbase3t64 mesa-vaapi-drivers
curlftps libconfig-inifiles-perl libmjpegutils-2.1-0t64 libpocketsphinx3 libswscale8 pocketsphinx-en-us
libavfilter10 libfuse2t64 libmpeg2encpp-2.1-0t64 libpostproc58 libvdpau-va-gl1 vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
Upgrading:
adduser libcairo-script-interpret2 mesa-vulkan-drivers
alsa-ucm-conf libcairo2 ncurses-base
apache2 libcanberra-gtk3-0 ncurses-bin
apache2-bin libcanberra-gtk3-module ncurses-term
apache2-data libcanberra0 netexec
apache2-utils libcjsn1 netpbm
apt libcurl3t64-gnutls network-manager-openconnect
apt-utils libcurl4t64 network-manager-openconnect-gnome
at-spi2-common libdavid7 network-manager-pptp
at-spi2-core libfile-desktopentry-perl network-manager-pptp-gnome
certipy-ad libgoa-1.0-0b node-acorn
chromium libgoa-1.0-common node-normalize.css
chromium-common libgssdp-1.6-0 openconnect
chromium-sandbox libgststreamer-plugins-bad1.0-0 osdp-openvas
```

Conclusion:

In this task, operating system security was successfully implemented using Kali Linux. User account management and privilege control were performed using sudo instead of direct root access. File permissions and ownership were configured using chmod and chown to restrict unauthorized access. The UFW firewall was enabled to control network traffic, and running processes and services were monitored using system commands. Unnecessary services were disabled to reduce the system attack surface. Through this task, a practical understanding of OS-level security and hardening techniques in a Linux environment was achieved.