

Fatima Jinnah Women University



Cloud Computing

Lab 4

Submitted By :

Inshal Nasir

Section :

A

Roll No:

38

Submitted To:

Sir Shoaib

LAB TITLE: Virtualization & Linux Fundamentals

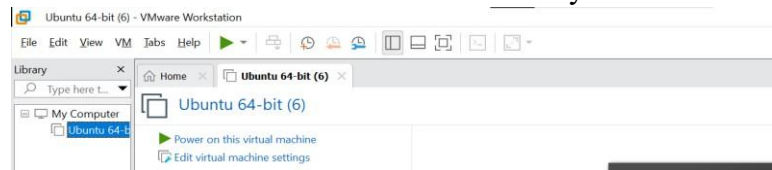
TASK:

Task 1: Verify VM resources in VMware

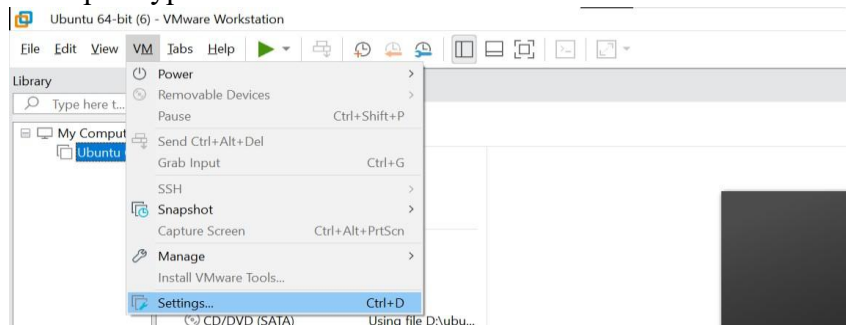
Confirm the VM resources that were allocated in Lab 1.

Steps

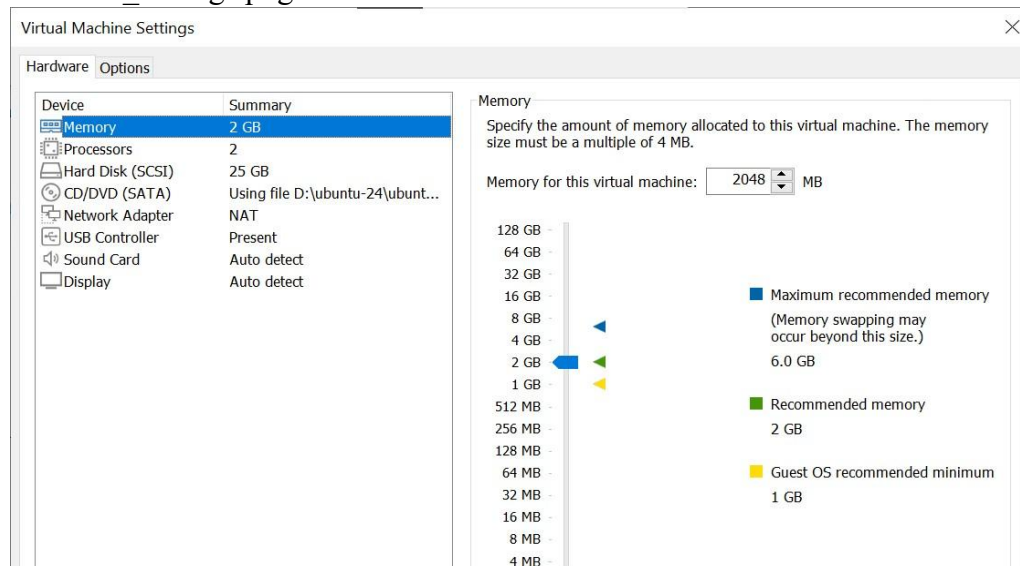
1. Open VMware Workstation and locate the Ubuntu Server VM you used in Lab 1.



2. Inspect VM settings and note the following (no commands required for GUI): VM name, RAM, CPU, disk, and network adapter type.



3. Take a screenshot of the VM settings window showing RAM, CPU, disk and networking. Save screenshot as: vm_settings.png

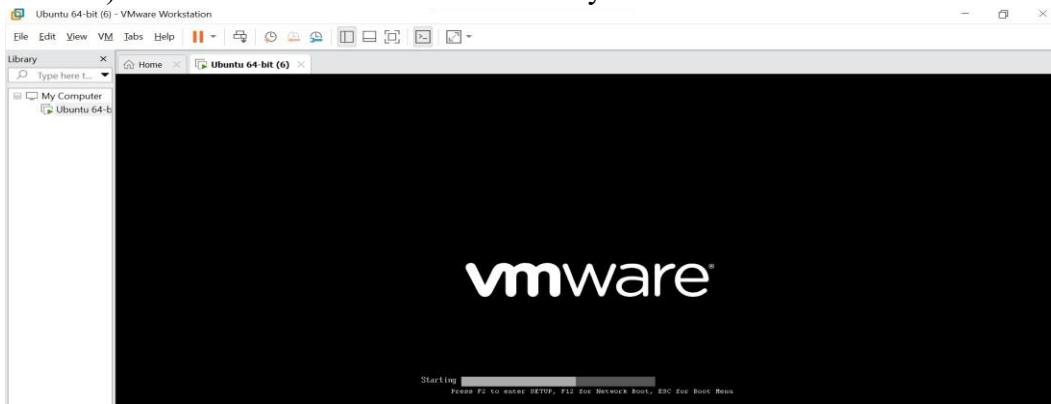


Task 2: Start VM and log in (use your preferred host terminal method only)

Use a single preferred host-terminal method to connect to the VM. Do not switch between methods during the task.

Steps

1. Start (or resume) the VM in VMware Workstation on your host.



2. From your host, open your preferred terminal (for example: Windows Command Prompt, PowerShell, macOS Terminal, or Linux Terminal) and connect to the VM using SSH. Example: `ssh student@<vm-ip-address>`

1. Find the IP address of your Ubuntu Server using “ip addr”

```
ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether 00:0c:29:6a:f6:87 brd ff:ff:ff:ff:ff:ff
altname enp2s1
inet 192.168.161.129/24 metric 100 brd 192.168.161.255 scope global dynamic ens33
    valid_lft 1082sec preferred_lft 1082sec
inet6 fe80::20c:29ff:fe6a:f687/64 scope link
    valid_lft forever preferred_lft forever
```

2. Connect via SSH from Windows

```
PS C:\Users\HP> ping 192.168.161.129

Pinging 192.168.161.129 with 32 bytes of data:
Reply from 192.168.161.129: bytes=32 time=1ms TTL=64
Reply from 192.168.161.129: bytes=32 time<1ms TTL=64
Reply from 192.168.161.129: bytes=32 time<1ms TTL=64
Reply from 192.168.161.129: bytes=32 time=3ms TTL=64

Ping statistics for 192.168.161.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 1ms
```

3. Accept the fingerprint (first time only) Type yes when prompted.

```
PS C:\Users\HP> ssh hajra@192.168.161.129
The authenticity of host '192.168.161.129 (192.168.161.129)' can't be established.
ED25519 key fingerprint is SHA256:M5unwK7+jxUhr6KM3bbMPRkghlGd+Kq5uspTQZ5Up3w.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

4. Enter your password

Use the same password you set up during the Ubuntu Server installation.

```
Warning: Permanently added '192.168.161.129' (ED25519) to the list of known hosts.
hajra@192.168.161.129's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-84-generic x86_64)

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

System information as of Sat 27 Sep 10:28:45 UTC 2025

System load:  1.46           Processes:    276
Usage of /:   70.2% of 11.21GB Users logged in: 1
Memory usage: 57%           IPv4 address for ens33: 192.168.161.129
Swap usage:   9%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

hajra@ubuntu-lab: $
```

- After logging in, run both commands and capture them together in a single screenshot:
whoami
pwd

```
hajra@ubuntu-lab:~$ whoami
hajra
hajra@ubuntu-lab:~$ pwd
/home/hajra
hajra@ubuntu-lab:~$
```

Task 3: Filesystem exploration — root tree and dotfiles

Steps (run inside VM terminal)

- List root directory contents:

```
ls -la /
```

```
hajra@ubuntu-lab:~$ ls -la
total 44
drwxr-xr-x 4 hajra hajra 4096 Sep 26 22:08 .
drwxr-xr-x 3 root  root 4096 Sep 26 21:38 ..
-rw-r--r-- 1 hajra hajra 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 hajra hajra 3771 Mar 31 2024 .bashrc
drwx----- 2 hajra hajra 4096 Sep 26 21:40 .cache
-rw----- 1 hajra hajra 20 Sep 26 22:08 .lessshst
-rw-r--r-- 1 hajra hajra 807 Mar 31 2024 .profile
drwx----- 2 hajra hajra 4096 Sep 26 22:00 .ssh
-rw-r--r-- 1 hajra hajra 0 Sep 26 21:48 .sudo_as_admin_successful
-rw-r--r-- 1 hajra hajra 9839 Sep 26 22:06 'systemctl status ssh'
hajra@ubuntu-lab:~$
```

- Inspect these directories (run each command and screenshot the output):

```
ls -la /bin
```

```
hajra@ubuntu-lab:~$ ls -la /bin
lrwxrwxrwx 1 root root 7 Apr 22 2024 /bin -> usr/bin
hajra@ubuntu-lab:~$
```

```
ls -la /sbin
```

```
hajra@ubuntu-lab:~$ ls -la /sbin
lrwxrwxrwx 1 root root 8 Apr 22 2024 /sbin -> usr/sbin
hajra@ubuntu-lab:~$
```

```
ls -la /usr
```

```
hajra@ubuntu-lab:~$ ls -la /usr
total 96
drwxr-xr-x 12 root root 4096 Aug 5 16:54 .
drwxr-xr-x 23 root root 4096 Sep 26 21:19 ..
drwxr-xr-x 2 root root 36864 Oct 21 17:19 bin
drwxr-xr-x 2 root root 4096 Apr 22 2024 games
drwxr-xr-x 35 root root 4096 Oct 14 08:07 include
drwxr-xr-x 83 root root 4096 Oct 21 17:20 lib
drwxr-xr-x 2 root root 4096 Sep 26 21:18 lib64
drwxr-xr-x 13 root root 4096 Oct 14 08:06 libexec
drwxr-xr-x 10 root root 4096 Aug 5 16:54 local
drwxr-xr-x 2 root root 20480 Oct 21 17:18 sbin
drwxr-xr-x 128 root root 4096 Oct 14 08:07 share
drwxr-xr-x 7 root root 4096 Oct 21 17:20 src
hajra@ubuntu-lab:~$
```

```
ls -la /opt
```

```
hajra@ubuntu-lab:~$ ls -la /opt
total 16
drwxr-xr-x 4 root root 4096 Sep 26 21:51 .
drwxr-xr-x 23 root root 4096 Sep 26 21:19 ..
drwxr-xr-x 2 root root 4096 Sep 26 21:51 cni
drwxr-xr-x 4 root root 4096 Sep 26 21:51 containerd
hajra@ubuntu-lab:~$
```

```
ls -la /etc
```

```

drwxr-xr-x 2 root root 4096 Aug 5 17:14 services
-rw-r--r-- 1 root root 12813 Mar 27 2021 services
drwxr-xr-x 2 root root 4096 Aug 5 17:02 shadow
-rw-r--r-- 1 root shadow 385 Sep 26 21:19 shadow
-rw-r--r-- 1 root shadow 367 Sep 26 21:18 shadow
-rw-r--r-- 1 root root 149 Aug 5 17:14 shells
drwxr-xr-x 2 root root 4096 Aug 5 16:55 sudo
drwxr-xr-x 6 root root 4096 Aug 5 17:14 sudo
drwxr-xr-x 4 root root 4096 Sep 26 21:19 sudo
drwxr-xr-x 4 root root 4096 Oct 21 17:18 subgid
-rw-r--r-- 1 root root 19 Sep 26 21:18 subgid
drwxr-xr-x 2 root root 4096 Aug 5 16:54 subuid
-rw-r--r-- 1 root root 19 Sep 26 21:18 subuid
-rw-r--r-- 1 root root 4096 Aug 5 16:54 subuid
-rw-r--r-- 1 root root 4343 Jun 25 12:42 sudo.conf
-rw-r--r-- 1 root root 1800 Jan 29 2024 sudoers
drwxr-xr-x 2 root root 4096 Aug 5 17:02 sudo
drwxr-xr-x 2 root root 4096 Aug 5 17:14 sudo_logsrvd.conf
-rw-r--r-- 1 root root 2289 Mar 24 2024 sysctl.conf
drwxr-xr-x 2 root root 4096 Sep 26 21:51 systemd
drwxr-xr-x 2 root root 4096 Aug 5 17:14 systemd
drwxr-xr-x 6 root root 4096 Aug 5 16:49 systemd
drwxr-xr-x 2 root root 4096 Aug 5 17:00 systemd
drwxr-xr-x 2 root root 4096 Sep 26 21:19 systemd
-rw-r--r-- 1 root root 8 Aug 5 17:02 timezone
drwxr-xr-x 2 root root 4096 Aug 5 17:14 tzdata
drwxr-xr-x 2 root root 4096 Aug 5 17:14 tzdata
-rw-r--r-- 1 root root 1269 Jan 27 2023 ucf.conf
drwxr-xr-x 4 root root 4096 Aug 5 17:02 ucf
drwxr-xr-x 2 root root 4096 Sep 26 21:19 udev
drwxr-xr-x 3 root root 4096 Aug 5 17:14 udev
-rw-r--r-- 1 root root 208 Aug 5 16:54 .updated
drwxr-xr-x 3 root root 4096 Aug 5 17:02 udev
drwxr-xr-x 2 root root 4096 Sep 26 21:51 udev
drwxr-xr-x 2 root root 4096 Aug 5 17:14 udev
drwxr-xr-x 2 root root 4096 Sep 26 21:19 udev
-rw-r--r-- 1 root root 1523 Aug 5 17:14 usb_modeswitch.conf
drwxr-xr-x 2 root root 4096 Aug 5 17:14 usb
-rw-r--r-- 1 root root 16 Aug 5 17:02 vconsole.conf -> default/keyboard
drwxr-xr-x 2 root root 4096 Sep 26 21:24 vga
drwxr-xr-x 4 root root 4096 Oct 21 17:15 vga
drwxr-xr-x 2 root root 23 Feb 26 2024 vga -> /etc/alternatives/vtgrb
-rw-r--r-- 1 root root 4942 Aug 5 17:14 wgetrc
drwxr-xr-x 4 root root 4096 Aug 5 17:02 x
-rw-r--r-- 1 root 681 Mar 8 2024 xattr.conf
drwxr-xr-x 4 root root 4096 Aug 5 17:02 x
drwxr-xr-x 2 root root 4096 Aug 5 17:02 x
-rw-r--r-- 1 root 460 Aug 5 17:14 zsh_command_not_found
hajra@ubuntu-lab:~$

```

1. click inside or press Ctrl+G.

ls -la /dev

```

chrp-pts 1 root dialout 4, 91 Oct 22 20:00 tty627
chrp-pts 1 root dialout 4, 92 Oct 22 20:00 tty628
chrp-pts 1 root dialout 4, 93 Oct 22 20:00 tty629
chrp-pts 1 root dialout 4, 67 Oct 22 20:00 tty63
chrp-pts 1 root dialout 4, 94 Oct 22 20:00 tty630
chrp-pts 1 root dialout 4, 95 Oct 22 20:00 tty631
chrp-pts 1 root dialout 4, 68 Oct 22 20:00 tty64
chrp-pts 1 root dialout 4, 69 Oct 22 20:00 tty65
chrp-pts 1 root dialout 4, 70 Oct 22 20:00 tty66
chrp-pts 1 root dialout 4, 71 Oct 22 20:00 tty67
chrp-pts 1 root dialout 4, 72 Oct 22 20:00 tty68
chrp-pts 1 root dialout 4, 73 Oct 22 20:00 tty69
drwxr-xr-x 2 root root 4096 Oct 22 20:00 udev
chrp-pts 1 root kvm 10, 124 Oct 22 20:00 udev
chrp-pts 1 root root 10, 239 Oct 22 20:00 udev
chrp-pts 1 root root 10, 223 Oct 22 20:00 uinput
chrp-pts 1 root root 1, 9 Oct 22 20:00 urandom
chrp-pts 1 root root 10, 156 Oct 22 20:00 userfaultfd
chrp-pts 1 root root 10, 240 Oct 22 20:00 userio
chrp-pts 1 root tty 7, 0 Oct 22 20:00 vc5
chrp-pts 1 root tty 7, 1 Oct 22 20:00 vc51
chrp-pts 1 root tty 7, 2 Oct 22 20:00 vc52
chrp-pts 1 root tty 7, 3 Oct 22 20:00 vc53
chrp-pts 1 root tty 7, 4 Oct 22 20:00 vc54
chrp-pts 1 root tty 7, 5 Oct 22 20:00 vc55
chrp-pts 1 root tty 7, 6 Oct 22 20:00 vc56
chrp-pts 1 root tty 7, 128 Oct 22 20:00 vc5a
chrp-pts 1 root tty 7, 129 Oct 22 20:00 vc5a1
chrp-pts 1 root tty 7, 130 Oct 22 20:00 vc5a5
chrp-pts 1 root tty 7, 131 Oct 22 20:00 vc5a3
chrp-pts 1 root tty 7, 132 Oct 22 20:00 vc5a4
chrp-pts 1 root tty 7, 133 Oct 22 20:00 vc5a5
chrp-pts 1 root tty 7, 134 Oct 22 20:00 vc5a6
chrp-pts 1 root tty 7, 64 Oct 22 20:00 vc5u
chrp-pts 1 root tty 7, 65 Oct 22 20:00 vc5u1
chrp-pts 1 root tty 7, 66 Oct 22 20:00 vc5u2
chrp-pts 1 root tty 7, 67 Oct 22 20:00 vc5u3
chrp-pts 1 root tty 7, 68 Oct 22 20:00 vc5u4
chrp-pts 1 root tty 7, 69 Oct 22 20:00 vc5u5
chrp-pts 1 root tty 7, 70 Oct 22 20:00 vc5u6
drwxr-xr-x 2 root root 4096 Oct 22 20:00 vga
chrp-pts 1 root root 10, 127 Oct 22 20:00 vga_arbiter
chrp-pts 1 root root 10, 137 Oct 22 20:00 vchi
chrp-pts 1 root kvm 10, 238 Oct 22 20:00 vhost-net
chrp-pts 1 root kvm 10, 241 Oct 22 20:00 vhost-vsock
chrp-pts 1 root root 10, 122 Oct 22 20:00 vmi
chrp-pts 1 root root 10, 121 Oct 22 20:00 vsock
chrp-pts 1 root root 1, 5 Oct 22 20:00 zero
chrp-pts 1 root root 10, 249 Oct 22 20:00 zfs
hajra@ubuntu-lab:~$

```

1. click inside or press Ctrl+G.

ls -la /var

```

hajra@ubuntu-lab:~$ ls -la /var
total 56
drwxr-xr-x 13 root root 4096 Sep 26 21:19 .
drwxr-xr-x 23 root root 4096 Sep 26 21:19 ..
drwxr-xr-x 2 root root 4096 Oct 22 00:00 base-files
drwxr-xr-x 16 root root 4096 Sep 27 08:39 base
drwxrwxrwt 2 root root 4096 Aug 5 17:02 crash
drwxr-xr-x 46 root root 4096 Sep 26 08:39 data
drwxrwxr-x 2 root staff 4096 Apr 22 2024 local
lrwxrwxrwx 1 root root 9 Aug 5 16:54 lock -> /run/lock
drwxrwxr-x 12 root syslog 4096 Oct 22 20:00 log
drwxrwxr-x 2 root mail 4096 Aug 5 16:54 mail
drwxr-xr-x 2 root root 4096 Aug 5 16:54 net
lrwxrwxrwx 1 root root 4 Aug 5 16:54 run -> /run
drwxr-xr-x 20 root root 4096 Sep 26 22:00 snap
drwxr-xr-x 4 root root 4096 Aug 5 17:14 tmp
drwxrwxrwt 7 root root 4096 Oct 22 20:01 tmp
-rw-r--r-- 1 root root 208 Aug 5 16:54 .updated
hajra@ubuntu-lab:~$

```

ls -la /tmp

```

hajra@ubuntu-lab:~$ ls -la /tmp
total 52
drwxrwxrwt 13 root root 4096 Oct 22 20:11 .
drwxr-xr-x 23 root root 4096 Sep 26 21:19 ..
drwxrwxrwt 2 root root 4096 Oct 22 20:00 .font-unix
drwxrwxrwt 2 root root 4096 Oct 22 20:00 .ICE-unix
drwxr----- 6 root root 4096 Oct 22 20:00 snap-private-409628d4bd7640d3a02ef310ac1abe5-ModemManager.service-FDyMC2
drwxr----- 3 root root 4096 Oct 22 20:00 snap-private-409628d4bd7640d3a02ef310ac1abe5-pollit.service-knp11
drwxr----- 3 root root 4096 Oct 22 20:00 snap-private-409628d4bd7640d3a02ef310ac1abe5-systemd-logind.service-vtrwJf
drwxr----- 3 root root 4096 Oct 22 20:00 snap-private-409628d4bd7640d3a02ef310ac1abe5-systemd-resolved.service-g1ewrk
drwxr----- 3 root root 4096 Oct 22 20:00 snap-private-409628d4bd7640d3a02ef310ac1abe5-systemd-timesyncd.service-jld0e
drwxrwxrwt 2 root root 4096 Oct 22 20:00 .X11-unix
drwxrwxrwt 2 root root 4096 Oct 22 20:00 .XIM-unix
hajra@ubuntu-lab:~$

```


3. List your home directory and show hidden (dot) files:

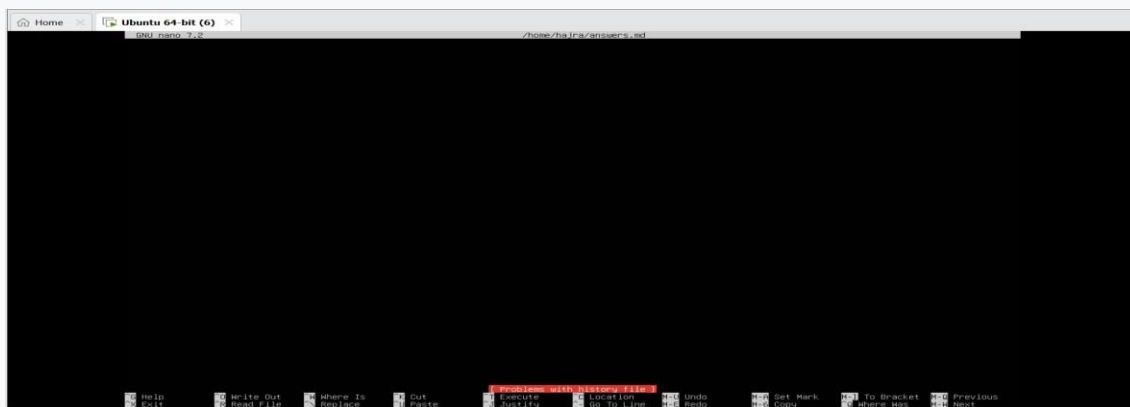
```
ls -la ~
```

```
hajra@ubuntu-lab:~$ ls -la $HOME
total 48
drwxr-x--- 5 hajra hajra 4096 Oct 23 04:41 .
drwxr-xr-x 3 root root 4096 Sep 26 21:38 ..
-rw-r--r-- 1 hajra hajra 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 hajra hajra 3771 Mar 31 2024 .bashrc
drwx----- 2 hajra hajra 4096 Sep 26 21:40 .cache
-rw-r--r-- 1 hajra hajra 20 Sep 26 22:08 .lesshst
-rw-r--r-- 1 hajra hajra 807 Mar 31 2024 .profile
drwx----- 2 hajra hajra 4096 Sep 26 22:00 .ssh
-rw-r--r-- 1 hajra hajra 0 Sep 26 21:48 .sudo_as_admin_successful
-rw-r--r-- 1 hajra hajra 9839 Sep 26 22:06 'systemctl status ssh'
```

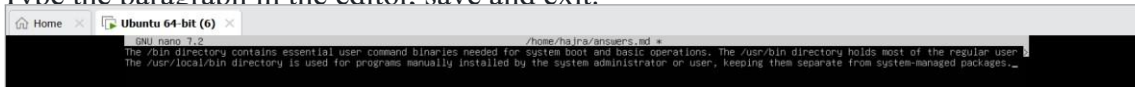
4. Write a short paragraph (3–5 sentences) that explains the difference between /bin, /usr/bin and /usr/local/bin.

Open your editor:

```
nano ~/answers.md
```



- Type the paragraph in the editor, save and exit.



- After saving, open the editor display (or show the file) and capture a screenshot of the paragraph.

```
hajra@ubuntu-lab:~$ cat $HOME/answers.md
The /bin directory contains essential user command binaries needed for system boot and basic operations. The /usr/bin directory holds most of the regular user c
ommands and applications that are not required for system startup.
The /usr/local/bin directory is used for programs manually installed by the system administrator or user, keeping them separate from system-managed packages.
hajra@ubuntu-lab:~$
```

Task 4: Essential CLI Tasks — Navigation and File Operations

Steps Performed

1. Create a Workspace and Navigate

```
mkdir -p ~/lab4/workspace/python_project
```

```
hajra@ubuntu-lab:~$ mkdir -p $HOME/lab4/workspace/python_project
hajra@ubuntu-lab:~$
```

Created a new workspace directory for the Python project.

```
cd ~/lab4/workspace/python_project
```

```
hajra@ubuntu-lab:~$ cd $HOME/lab4/workspace/python_project
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Navigated into the newly created directory.

Pwd

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ pwd
/home/hajra/lab4/workspace/python_project
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Verified the current working directory path.

2. Create Files Using an Editor nano

README.md

```
GNU nano 7.2 README.md *
lab 4 README_
```

Added the text “**Lab 4 README**” and saved the file.

nano main.py

```
GNU nano 7.2 main.py *
print('hello lab4')
```

Added the Python code: `print("hello lab4")` and saved the file.

nano .env

```
GNU nano 7.2 .env *
ENV = lab4_
```

Added the line **ENV=lab4** and saved the file.

3. List Files

ls -la

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ ls -la
total 20
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:14 .
drwxrwxr-x 3 hajra hajra 4096 Oct 23 13:11 ..
-rw-rw-r-- 1 hajra hajra 11 Oct 23 14:14 .env
-rw-rw-r-- 1 hajra hajra 21 Oct 23 14:11 main.py
-rw-rw-r-- 1 hajra hajra 13 Oct 23 14:07 README.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Displayed all files, including hidden ones, in the current directory.

4. Copy, Move, and Remove Files cp README.md

README.copy.md

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cp README.md README.md.copy.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Created a copy of the README file.

mv README.copy.md README.dev.md

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ mv README.md.copy.md README.dev.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Renamed (moved) the copied file.

rm README.dev.md

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ rm README.dev.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Deleted the renamed file.

5. Work with Directories

mkdir -p ~/lab4/workspace/java_app

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ mkdir -p $HOME/lab4/workspace/java_app
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Created another directory for a Java app.

cp -r ~/lab4/workspace/python_project ~/lab4/workspace/java_app_copy

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cp -r $HOME/lab4/workspace/python_project $HOME/lab4/workspace/java_app_copy
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Copied the entire Python project directory recursively.

ls -la ~/lab4/workspace

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ ls -la $HOME/lab4/workspace
total 16
drwxrwxr-x 4 hajra hajra 4096 Oct 23 14:34 .
drwxrwxr-x 3 hajra hajra 4096 Oct 23 13:11 ..
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:34 java_app_copy
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:27 python_project
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Verified the copied directories.

6. Use Command History and Tab Completion

History

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ history
1 ls -la $HOME
2 nano $HOME/answers.md
3 cat $HOME/answers.md
4 mkdir -p $HOME/lab4/workspace/python_project
5 cd $HOME/lab4/workspace/python_project
6 pwd
7 nano README.md
8 nano main.py
9 nano .env
10 ls -la
11 cp README.md README.md.copy.md
12 mv README.copy.md README.dev.md
13 mv README.md.copy.md README.dev.md
14 rm README.dev.md
15 mkdir -p $HOME/lab4/workspace/java_app
16 cp -r $HOME/lab4/workspace/python_project $HOME/lab4/workspace/java_app_copy
17 ls -la $HOME/lab4/workspace
18 history
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Displayed a list of previously executed commands.

- Demonstrated tab completion by typing part of a file or directory name and pressing **Tab** to autocomplete it.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cat main.py
print ('hello lab4')
```

Task 5: System info, resources & processes

Collect system information and observe processes. Use screenshots only.

Steps (inside VM terminal)

1. Kernel and OS:

uname -a

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ uname -a
Linux ubuntu-lab 6.8.0-85-generic #85-Ubuntu SMP PREEMPT_DYNAMIC Thu Sep 18 15:26:59 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

2. CPU (ensure model name visible):

cat /proc/cpuinfo


```
Home x Ubuntu 64-bit (6) x
core_id : 0
cpu_cores : 1
apicid : 0
initial_apicid : 0
fpu : yes
fpu_exception : yes
cpuid_level : 22
wp : yes
flags : fpu vme de pse tsc mtr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ss syscall nx pdpe1gb rdtscp lm constant_tsc
arch_perfmon nopl xtopology tsc_reliable nonstop_tsc cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer
r_aes xsave avx f16c rdrand hypervisor lahf_lm adm 3dnowprefetch pti ssbd ibrs lbrs stibp fsgsbase tsc_adjust bmi1 avx2 smep bmi2 invpcid rdseed adx smap clflush
host_xsaveopt xsavec xgetbv1 xsave_early ad_clear_flush_lid arch_capabilities
bugs : cpu_mitdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swaps itlb_multihit rrbds mmio_stale_data retbleed gds bhi
bogomips : 3791.99
clflush_size : 64
cache_alignment : 64
address_sizes : 45 bits physical, 48 bits virtual
power management:

processor : 1
vendor_id : GenuineIntel
cpu family : 6
model : 142
model name : Intel(R) Core(TM) i5-8350U CPU @ 1.70GHz
stepping : 10
microcode : 0xffffffff
cpu mhz : 1895.977
cache size : 6144 KB
physical_id : 2
siblings : 1
core_id : 0
cpu_cores : 1
apicid : 2
initial_apicid : 2
fpu : yes
fpu_exception : yes
cpuid_level : 22
wp : yes
flags : fpu vme de pse tsc mtr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ss syscall nx pdpe1gb rdtscp lm constant_tsc
arch_perfmon nopl xtopology tsc_reliable nonstop_tsc cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer
r_aes xsave avx f16c rdrand hypervisor lahf_lm adm 3dnowprefetch pti ssbd ibrs lbrs stibp fsgsbase tsc_adjust bmi1 avx2 smep bmi2 invpcid rdseed adx smap clflush
host_xsaveopt xsavec xgetbv1 xsave_early ad_clear_flush_lid arch_capabilities
bugs : cpu_mitdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swaps itlb_multihit rrbds mmio_stale_data retbleed gds bhi
bogomips : 3791.99
clflush_size : 64
cache_alignment : 64
address_sizes : 45 bits physical, 48 bits virtual
power management:

hajra@ubuntu-lab:~/lab4/workspace/python_projects$
```

3. Memory: free -h

```
hajra@ubuntu-lab:~/lab4/workspace/python_projects$ free -h
             total        used        free      shared  buff/cache   available
Mem:          1.9Gi          1.2Gi          132Mi          12Mi          734Mi          667Mi
Swap:          2.0Gi          223Mi           1.8Gi
hajra@ubuntu-lab:~/lab4/workspace/python_projects$
```

4. Disk:

df -h

```
hajra@ubuntu-lab:~/lab4/workspace/python_projects$ free -h
             total        used        free      shared  buff/cache   available
Mem:          1.9Gi          1.2Gi          132Mi          12Mi          734Mi          667Mi
Swap:          2.0Gi          223Mi           1.8Gi
hajra@ubuntu-lab:~/lab4/workspace/python_projects$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            192M  1.0M  191M   1% /run
/dev/mapper/ubun--vg-ubuntu--lv 12G  11G  343M  97% /
tmpfs            960M   0  960M   0% /dev/shm
tmpfs            5.0M   0   5.0M   0% /run/lock
/dev/sda2        2.0G  192M   1.6G  11% /boot
tmpfs            192M  20K  192M   1% /run/user/1000
shm              64M   0   64M   0% /var/snap/microk8s/common/run/containerd/io.containerd.grpc.v1.cri/sandboxes/ce047894ba4dc68148504b884d
d8e81bfeef7c1524cd31f3f71c0866c562fc57/shm 64M   0   64M   0% /var/snap/microk8s/common/run/containerd/io.containerd.grpc.v1.cri/sandboxes/907005c3fa2edabd7e07c5bb55
f734ead13372804c9de10e7c3258af5f4114e6/shm 64M   0   64M   0% /var/snap/microk8s/common/run/containerd/io.containerd.grpc.v1.cri/sandboxes/1fa57bb780d67f58e774952f1e
e1858625b771458a25591d41300447ba4e2367/shm
hajra@ubuntu-lab:~/lab4/workspace/python_projects$
```

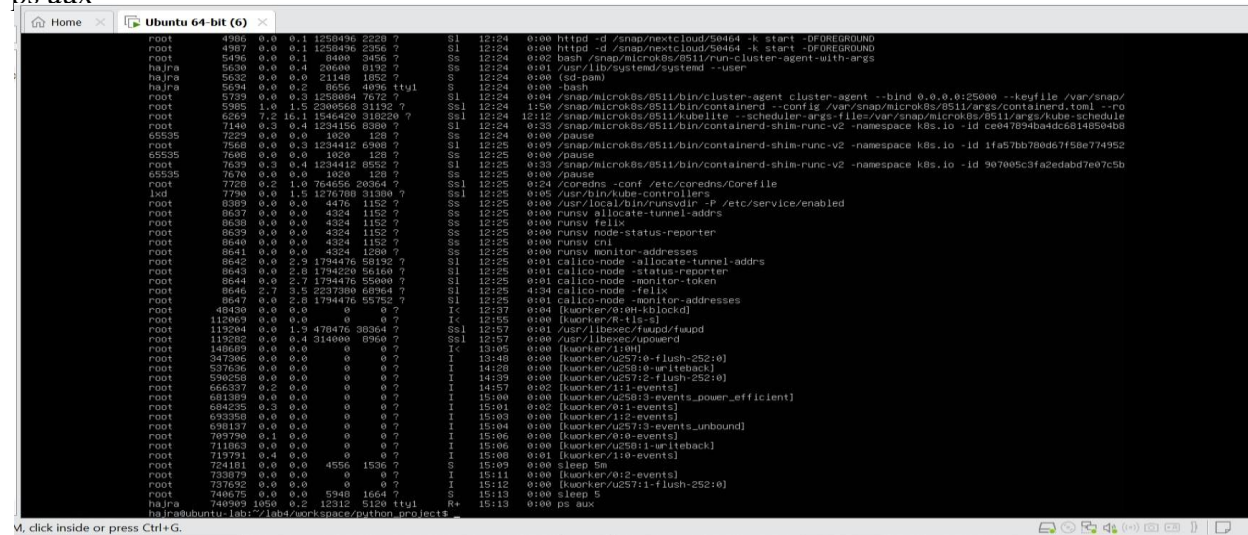
5. View OS release information:

cat /etc/os-release

```
hajra@ubuntu-lab:~/lab4/workspace/python_projects$ cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04.3 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.3 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
hajra@ubuntu-lab:~/lab4/workspace/python_projects$
```

6. Processes (show top lines of ps output):

ps aux



Task 6: Users and account verification (no sudo group change)

Steps (inside VM terminal)

1. Create a new user named lab4user:

sudo adduser lab4user

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ sudo adduser lab4user
[sudo] password for hajra:
info: Adding user `lab4user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `lab4user' (1002) ...
info: Adding new user `lab4user' (1002) with group `lab4user (1002)' ...
info: Creating home directory `/home/lab4user' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for lab4user
Enter the new value, or press ENTER for the default
  Full Name []: hajra
  Room Number []: twenty two
  Work Phone []: 03two7-5007883
  Home Phone []: 03two7-5007883
  Other []: nil
Is the information correct? [Y/n] Y
info: Adding new user `lab4user' to supplemental / extra groups `users' ...
info: Adding user `lab4user' to group `users' ...
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

2. Verify the user entry:

getent passwd lab4user

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ getent passwd lab4user
lab4user:x:1002:1002:hajra,twenty two,03two7-5007883,03two7-5007883,nil:/home/lab4user:/bin/bash
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

3. Switch to the new user to verify login:

su - lab4user

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ su - lab4user
Password:
lab4user@ubuntu-lab:~$
```

4. From the new user you may attempt a sudo command to show that sudo is not available for this account (expected failure), e.g.:

sudo whoami

```
lab4user@ubuntu-lab:~$ sudo whoami
[sudo] password for lab4user:
lab4user is not in the sudoers file.
lab4user@ubuntu-lab:~$ _
```

5. Return to the original user:

[Exit](#)

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```

6. (Optional) Remove the test user when finished:

```
sudo deluser --remove-home lab4user
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ sudo deluser --remove-home lab4user
[sudo] password for hajra:
info: Looking for files to backup/remove ...
info: Removing files ...
info: Removing crontab ...
info: Removing user 'lab4user' ...
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```

Bonus Task 7: Create a small demo script using an editor and run it Steps (inside VM)

1. Open an editor to create the script:

```
nano ~/lab4/workspace/run-demo.sh
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ nano $HOME/lab4/workspace/python_project/run-demo.sh
```

- Type the following lines into the editor (manually or paste), save and exit:

```
#!/bin/bash
echo "Lab 4 demo: current user is $(whoami)"
echo "Current time: $(date)" uptime

free -h
```

```
GNU nano 7.2 /home/hajra/lab4/workspace/python_project/run-demo.sh *
#!/bin/bash
echo "Lab 4 demo: current user is $(whoami)"
echo "Current time: $(date)"
uptime
free -h_
```

2. Make the script executable:

```
chmod +x ~/lab4/workspace/run-demo.sh
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ chmod +x ~/lab4/workspace/python_project/run-demo.sh
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```

3. Run the script as your regular user:

```
~/lab4/workspace/run-demo.sh
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ ~/lab4/workspace/python_project/run-demo.sh
Lab 4 demo: current user is $(whoami)
Current time: $(date)
19:02:29 up 6:39, 1 user, load average: 0.78, 0.74, 0.76
Mem:      1.9Gi total, 1.2Gi used, 111Mi free, 740Ki shared, 730Mi buff/cache, 652Mi available
Swap:    2.0Gi total, 225Mi used, 1.8Gi free
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

4. Optionally run it with sudo:

```
sudo ~/lab4/workspace/run-demo.sh
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ sudo ~/lab4/workspace/python_project/run-demo.sh
[sudo] password for hajra:
Lab 4 demo: current user is $(whoami)
Current time: $(date)
19:03:47 up 6:41, 1 user, load average: 0.57, 0.66, 0.73
Mem:          total        used        free      shared  buff/cache   available
Swap:         1.9Gi        1.2Gi        128Mi       824Ki       731Mi        671Mi
Swap:         2.0Gi        225Mi        1.8Gi
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Exam Evaluation Questions

1. Remote Access Verification (Cyber Login Check)

Scenario:

You are part of a SOC (Security Operations Center) investigating unauthorized access to a Linux server hosted on VMware. Prove you can securely connect and verify your identity.

Steps:

1. Connect to the Ubuntu VM remotely from your host terminal.

```
Warning: Permanently added '192.168.161.129' (ED25519) to the list of known hosts.
hajra@192.168.161.129's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-84-generic x86_64)

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

System information as of Sat 27 Sep 10:28:45 UTC 2025

System load:  1.46           Processes:           276
Usage of /:   70.2% of 11.21GB Users logged in:     1
Memory usage: 57%           IPv4 address for ens33: 192.168.161.129
Swap usage:   9%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

hajra@ubuntu-lab: $
```

2. Verify your current user and home directory path.

```
hajra@ubuntu-lab: $ whoami
hajra
hajra@ubuntu-lab: $ pwd
/home/hajra
hajra@ubuntu-lab: $
```

3. Confirm you are connected to the correct host machine.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ hostname
ubuntu-lab
```

2. Filesystem Inspection for Forensic Evidence

Scenario:

The incident response team suspects malicious files in system directories. You must explore the filesystem to locate and document the system's structure.

Steps:

1. Display the contents of the root directory.

```

hajra@ubuntu-lab:~$ ls -la
total 44
drwxr-x--- 4 hajra hajra 4096 Sep 26 22:08 .
drwxr-xr-x 3 root  root 4096 Sep 26 21:38 ..
-rw-r--r-- 1 hajra hajra 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 hajra hajra 3771 Mar 31 2024 .bashrc
drwx----- 2 hajra hajra 4096 Sep 26 21:40 .cache
-rw----- 1 hajra hajra 20 Sep 26 22:08 .lessshst
-rw-r--r-- 1 hajra hajra 807 Mar 31 2024 .profile
drwx----- 2 hajra hajra 4096 Sep 26 22:00 .ssh
-rw-r--r-- 1 hajra hajra 0 Sep 26 21:48 .sudo_as_admin_successful
-rw-r--r-- 1 hajra hajra 9839 Sep 26 22:06 'systemctl status ssh'
hajra@ubuntu-lab:~$ _

```

2. Display the OS version and release information.

```

hajra@ubuntu-lab:~/lab4/workspace/python_project$ uname -a
Linux ubuntu-lab 6.8.0-85-generic #85-Ubuntu SMP PREEMPT_DYNAMIC Thu Sep 18 15:26:59 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux
hajra@ubuntu-lab:~/lab4/workspace/python_project$

```

3. Explore and record directory listings for /bin, /sbin, /usr, /opt, /etc, /dev, /var, and /tmp.

4. Display all hidden files in your home directory.

```

hajra@ubuntu-lab:~$ ls -la $HOME
total 48
drwxr-x--- 5 hajra hajra 4096 Oct 23 04:41 .
drwxr-xr-x 3 root  root 4096 Sep 26 21:38 ..
-rw-r--r-- 1 hajra hajra 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 hajra hajra 3771 Mar 31 2024 .bashrc
drwx----- 2 hajra hajra 4096 Sep 26 21:40 .cache
drwxrwxr-x 2 hajra hajra 4096 Oct 23 04:41 .kexec
-rw----- 1 hajra hajra 20 Sep 26 22:08 .lessshst
-rw-r--r-- 1 hajra hajra 807 Mar 31 2024 .profile
drwx----- 2 hajra hajra 4096 Sep 26 22:00 .ssh
-rw-r--r-- 1 hajra hajra 0 Sep 26 21:48 .sudo_as_admin_successful
-rw-r--r-- 1 hajra hajra 9839 Sep 26 22:06 'systemctl status ssh'
hajra@ubuntu-lab:~$

```

5. Create a markdown file summarizing your findings on key binary directories.

1. Go to your **home directory** (or your lab workspace folder):

```
cd ~ or cd ~/lab4/workspace
```

```
hajra@ubuntu-lab:~$ cd ~/lab4/workspace
```

2. Create and open a new Markdown file using **nano editor**:

```
nano report.md
```

```
hajra@ubuntu-lab:~/lab4/workspace$ nano report.md_
```

3. Inside nano, type your short summary.

```

GNU nano 7.2 report.md *
<!--System Directory Summary-->
- **/bin** - contains essential user command binaries (like ls, cp, mv)
- **/sbin** - Contain system binaries used mainly by the root user for system admi
- **/usr** - Contains user-installed programs, libraries, and documentation.
- **/opt** - Used for optional or third-party software.
- **/etc** - Contains configuration files for the system and installed services.
- **/dev** - Contains device files representing hardware devices.
- **/var** - Contains variable data such as logs, caches, and spool files.
- **/tmp** - Temporary files created by running processes.

```

4. When done, **save and exit nano**.

5. Verify your file exists and display its contents:

```
cat report.md
```



```
hajra@ubuntu-lab:~/lab4/workspace$ cat report.md
<!--System Directory Summery-->
- **/bin** - contains essential user command binaries (like ls, cp, mv)
- **/sbin** - Contain system binaries used mainly by the root user for system admi
- **/usr** - Contains user-installed programs, libraries, and documentation.
- **/opt** - Used for optional or third-party software.
- **/etc** - Contains configuration files for the system and installed services.
- **/dev** - Contains device files representing hardware devices.
- **/var** - Contains variable data such as logs, caches, and spool files.
- **/tmp** - Temporary files created by running processes.
hajra@ubuntu-lab:~/lab4/workspace$ _
```

3. Evidence Handling & File Operations Scenario:

You are creating a sandbox environment to safely analyze and handle suspicious files collected from a compromised system.

Steps:

1. Create a structured folder hierarchy under your home directory for analysis.


```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ pwd
/home/hajra/lab4/workspace/python_project
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```

2. Create three text files, including one hidden file, in your workspace.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ ls -la
total 20
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:14 .
drwxrwxr-x 3 hajra hajra 4096 Oct 23 13:11 ..
-rw-rw-r-- 1 hajra hajra 11 Oct 23 14:14 .env
-rw-rw-r-- 1 hajra hajra 21 Oct 23 14:11 main.py
-rw-rw-r-- 1 hajra hajra 13 Oct 23 14:07 README.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

3. Create a hackun conv of one file. rename it. and then delete it after verification.

```
cp README.md README.copy.md
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cp README.md README.md.copy.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Created a copy of the README file.

```
mv README.copy.md README.dev.md
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ mv README.md.copy.md README.dev.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Renamed (moved) the copied file.

```
rm README.dev.md
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ rm README.dev.md
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Deleted the renamed file.

4. Copy the entire workspace as an evidence backup folder.

```
mkdir -p ~/lab4/workspace/java_app
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ mkdir -p $HOME/lab4/workspace/java_app
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Created another directory for a Java app.

```
cp -r ~/lab4/workspace/python_project ~/lab4/workspace/java_app_copy
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cp -r $HOME/lab4/workspace/python_project $HOME/lab4/workspace/java_app_copy
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Copied the entire Python project directory recursively.

```
ls -la ~/lab4/workspace
```

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ ls -la $HOME/lab4/workspace
total 16
drwxrwxr-x 4 hajra hajra 4096 Oct 23 14:34 .
drwxrwxr-x 3 hajra hajra 4096 Oct 23 13:11 ..
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:34 java_app_copy
drwxrwxr-x 2 hajra hajra 4096 Oct 23 14:27 python_project
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

Verified the copied directories.

5. Display your command history to document all actions performed.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ history
1  ls -la $HOME
2  nano $HOME/answers.md
3  cat $HOME/answers.md
4  mkdir -p $HOME/lab4/workspace/python_project
5  cd $HOME/lab4/workspace/python_project
6  pwd
7  nano README.md
8  nano main.py
9  nano .env
10 ls -la
11 cp README.md README.md.copy.md
12 mv README.copy.md README.dev.md
13 mv README.md.copy.md README.dev.md
14 rm README.dev.md
15 mkdir -p $HOME/lab4/workspace/java_app
16 cp -r $HOME/lab4/workspace/python_project $HOME/lab4/workspace/java_app_copy
17 ls -la $HOME/lab4/workspace
18 history
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

6. Demonstrate Linux auto-completion by typing a partial command or filename.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ cat main.py
print('hello lab4')
```

4. System Profiling and Process Monitoring

Scenario:

You are investigating a potential malware infection that is consuming excessive resources on the Linux VM.

Steps:

1. Display the system's OS and kernel version for the investigation report.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ uname -a
Linux ubuntu-lab 6.8.0-85-generic #85-Ubuntu SMP PREEMPT_DYNAMIC Thu Sep 18 15:26:59 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux
hajra@ubuntu-lab:~/lab4/workspace/python_project$
```

2. Display CPU, memory, and disk usage information.

```

hajra@ubuntu-lab:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:            1.9Gi       938Mi       322Mi       9.9Mi       830Mi       981Mi
Swap:           2.0Gi        30Mi       2.0Gi
hajra@ubuntu-lab:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs            192M  1.5M  191M   1% /run
/dev/mapper/ubun--vg-ubuntu--lv 12G   11G  335M  97% /
tmpfs            960M   0    960M   0% /dev/shm
tmpfs            5.0M   0    5.0M   0% /run/lock
/dev/sda2        2.0G  192M  1.6G  11% /boot
tmpfs            192M  20K  192M   1% /run/user/1000
hajra@ubuntu-lab:~$ _

```

3. Display all active running processes to identify suspicious activity.

```

Home  Ubuntu 64-bit (6)
root  4986  0.0  0.1 1258496 2228 ?    S1 12:24 0:00 httpd -d /snap/nginxcloud/50464 -k start -DFOREGROUND
root  4987  0.0  0.1 1258496 2356 ?    S1 12:24 0:00 httpd -d /snap/nginxcloud/50464 -k start -DFOREGROUND
root  5436  0.0  0.1 8400 3456 ?      Ss 12:24 0:02 bash /snap/microk8s/8511/run-cluster-agent-with-args
hajra  5630  0.0  0.4 20600 8132 ?    Ss 12:24 0:01 /usr/lib/systemd/systemd --user
hajra  5632  0.0  0.0 21148 1852 ?      S  12:24 0:00 (sd-pam)
hajra  5634  0.0  0.2 6656 4096 tty1    S  12:24 0:00 -bash
root  5739  0.0  0.3 1258004 7672 ?    S1 12:24 0:04 /snap/microk8s/8511/bin/cluster-agent cluster-agent --bind 0.0.0.0:25000 --keyfile /var/snap/
root  5905  1.0  1.5 2300568 31192 ?   Ssl 12:24 1:50 /snap/microk8s/8511/bin/containerd --config /var/snap/microk8s/8511/args/containerd.toml --ro
root  6269  7.2 16.1 1546429 218220 ?  Ssl 12:24 12:12 /snap/microk8s/8511/ kubelet --scheduler-args=/var/snap/microk8s/8511/args/kube-schedule
root  7140  0.3  0.4 1234156 6380 ?    S1 12:24 0:33 /snap/microk8s/8511/bin/containerd-shim-runc-v2 -namespace k8s.io -id ce047094b4dc68140504b8
65535  7229  0.0  0.0 1020 128 ?      Ss 12:24 0:00 /pause
root  7608  0.0  0.3 1234412 6300 ?    S1 12:25 0:00 /snap/microk8s/8511/bin/containerd-shim-runc-v2 -namespace k8s.io -id 1fa57bb700d67f50e774952
65535  7608  0.0  0.0 1020 128 ?      Ss 12:25 0:00 /pause
root  7639  0.3  0.4 1234412 6300 ?    S1 12:25 0:33 /snap/microk8s/8511/bin/containerd-shim-runc-v2 -namespace k8s.io -id 907005c3fa2edab70e7c5b
65535  7670  0.0  0.0 1020 128 ?      Ss 12:25 0:00 /pause
root  7720  0.2  1.0 764656 20364 ?   Ssl 12:25 0:24 /coredns -conf /etc/coredns/Corefile
lxd  7780  0.0  1.5 1276708 31580 ?   Ssl 12:25 0:05 /usr/bin/lxd-containers
root  8389  0.0  0.0 4476 1152 ?      Ss 12:25 0:00 /usr/local/bin/runsvdir -P /etc/service/enabled
root  8638  0.0  0.0 4324 1152 ?      Ss 12:25 0:00 runsv allocate-tunnel-addr
root  8639  0.0  0.0 4324 1152 ?      Ss 12:25 0:00 runsv felix
root  8639  0.0  0.0 4324 1152 ?      Ss 12:25 0:00 runsv node-status-reporter
root  8640  0.0  0.0 4324 1152 ?      Ss 12:25 0:00 runsv cal
root  8641  0.0  0.0 4324 1280 ?      Ss 12:25 0:00 runsv monitor-addresses
root  8642  0.0  0.0 1794476 58192 ?   S1 12:25 0:01 calico-node -allocate-tunnel-addr
root  8643  0.0  2.9 1794220 56160 ?   S1 12:25 0:01 calico-node -status-reporter
root  8644  0.0  2.7 1794476 55000 ?   S1 12:25 0:01 calico-node -monitor-token
root  8645  2.7  3.5 2327200 68964 ?   S1 12:25 0:34 calico-node -felix
root  8647  0.0  2.0 1794476 55752 ?   S1 12:25 0:01 calico-node -monitor-addresses
root  8648  0.0  0.0 0 0 ?          Ic 12:27 0:04 [worker/0:0:0:blockd]
root  112469 0.0  0.0 0 0 ?          Ic 12:25 0:00 [worker/0:1:1:~]
root  119204 0.0  1.9 478476 30364 ?   Ssl 12:27 0:01 /usr/libexec/fuupd/fuupd
root  119202 0.0  0.4 214000 8360 ?    Ssl 12:27 0:00 /usr/libexec/upower
root  149689 0.0  0.0 0 0 ?          Ic 13:05 0:00 [worker/1:0:0]
root  347006 0.0  0.0 0 0 ?          I  13:40 0:00 [worker/u257:0-flush-252:0]
root  537656 0.0  0.0 0 0 ?          I  14:20 0:00 [worker/u258:0-uriteback]
root  590258 0.0  0.0 0 0 ?          I  14:39 0:00 [worker/u257:2-flush-252:0]
root  606507 0.2  0.0 0 0 ?          I  14:57 0:02 [worker/1:1:events]
root  601309 0.0  0.0 0 0 ?          I  15:00 0:00 [worker/u258:3-events.power.efficient]
root  604235 0.3  0.0 0 0 ?          I  15:01 0:02 [worker/0:1:events]
root  605358 0.0  0.0 0 0 ?          I  15:03 0:00 [worker/1:2:events]
root  636137 0.0  0.0 0 0 ?          I  15:04 0:00 [worker/u257:3-events.unbound]
root  709790 0.1  0.0 0 0 ?          I  15:06 0:00 [worker/0:0:events]
root  711063 0.0  0.0 0 0 ?          I  15:06 0:00 [worker/u258:1-uriteback]
root  719701 0.4  0.0 0 0 ?          I  15:08 0:01 [worker/1:0:events]
root  724101 0.0  0.0 4556 1536 ?    S  15:09 0:00 sleep 5m
root  730879 0.0  0.0 0 0 ?          I  15:11 0:00 [worker/0:2:events]
root  737632 0.0  0.0 0 0 ?          I  15:12 0:00 [worker/u257:1-flush-252:0]
root  740675 0.0  0.0 5948 1664 ?   S  15:13 0:00 sleep 5
hajra  74000 1050 0.2 12312 5120 tty1    R  15:13 0:00 ps aux
hajra@ubuntu-lab:~/lab4/workspace/python_project$

```

5. User Account Audit & Privilege Escalation

Simulation Scenario:

You are performing a user activity audit on a compromised Linux server. The SOC suspects a newly created account (lab4user) may have been used for unauthorized access. Your task is to simulate the account creation, perform privilege tests, and analyze authentication logs for forensic evidence.

Steps:

1. Create a new test user named lab4user.

```

hajra@ubuntu-lab:~/lab4/workspace/python_project$ sudo adduser lab4user
[sudo] password for hajra:
info: Adding user `lab4user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `lab4user' (1002) ...
info: Adding new user `lab4user' (1002) with group `lab4user (1002)' ...
info: Creating home directory `/home/lab4user' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for lab4user
Enter the new value, or press ENTER for the default
  Full Name []: hajra
  Room Number []: twenty two
  Work Phone []: 03two7-5007883
  Home Phone []: 03two7-5007883
  Other []: nil
Is the information correct? [Y/n] Y
info: Adding new user `lab4user' to supplemental / extra groups `users' ...
info: Adding user `lab4user' to group `users' ...
hajra@ubuntu-lab:~/lab4/workspace/python_project$

```

2. Verify that the new user record exists in the system's user database.

```

hajra@ubuntu-lab:~/lab4/workspace/python_project$ cat /etc/passwd | grep lab4user
lab4user:x:1002:1002:hajra:twenty two,03two7-5007883,03two7-5007883,nil:/home/lab4user:/bin/bash
hajra@ubuntu-lab:~/lab4/workspace/python_project$

```

3. Log in as lab4user and confirm successful login.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ su - lab4user
Password:
lab4user@ubuntu-lab:~$ _
```

4. Attempt to run an administrative command as lab4user (expect permission denied).

```
lab4user@ubuntu-lab:~$ sudo whoami
[sudo] password for lab4user:
lab4user is not in the sudoers file.
lab4user@ubuntu-lab:~$ _
```

5. Switch back to your main analyst account.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```

6. (Optional) Remove the lab4user account after the audit and verify deletion.

```
hajra@ubuntu-lab:~/lab4/workspace/python_project$ sudo deluser --remove-home lab4user
[sudo] password for hajra:
info: Looking for files to backup/remove ...
info: Removing files ...
info: Removing crontab ...
info: Removing user 'lab4user' ...
hajra@ubuntu-lab:~/lab4/workspace/python_project$ _
```
