**1a-1 Virtualisation & Linux Setup**

**1. Install VirtualBox or another virtualization tool.   
A screenshot of a computer

AI-generated content may be incorrect.  
  
2. Download an Ubuntu ISO (Lubuntu or regular Ubuntu).   
**

**3. Configure a new VM in VirtualBox with at least 2048 MB of memory. A screenshot of a computer

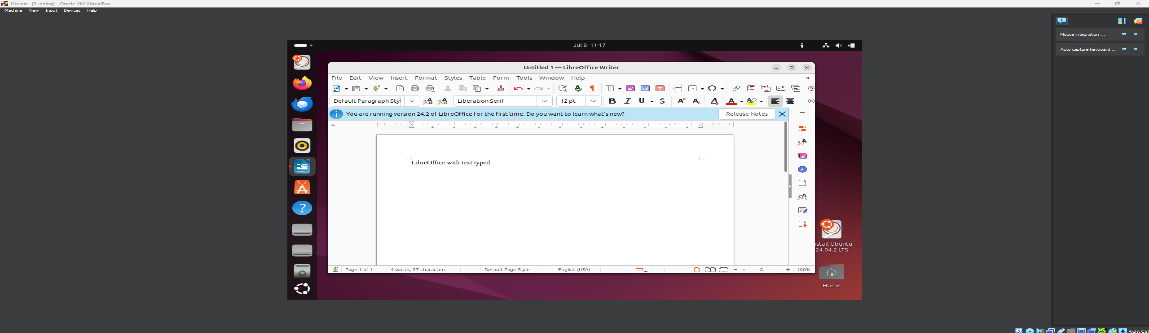
AI-generated content may be incorrect.  
  
  
4. Configure network settings to allow bridged or NAT internet access.   
A computer screen shot of a computer

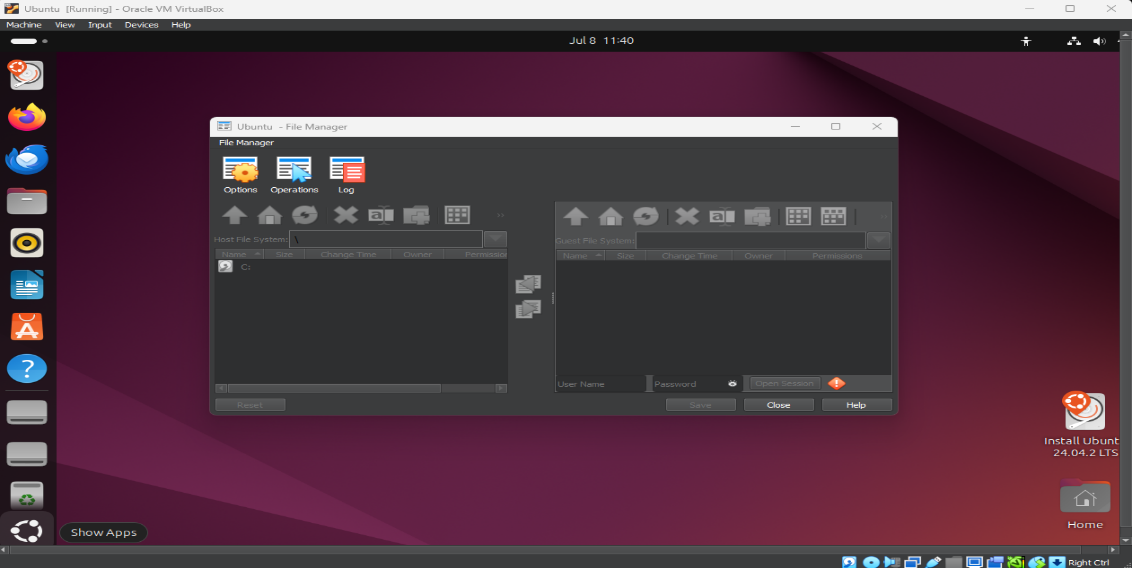
AI-generated content may be incorrect.**

**1a-2 Ubuntu Desktop and Command Line Familiarisation  
  
  
Part 1: Ubuntu Desktop GUI Familiarizations**

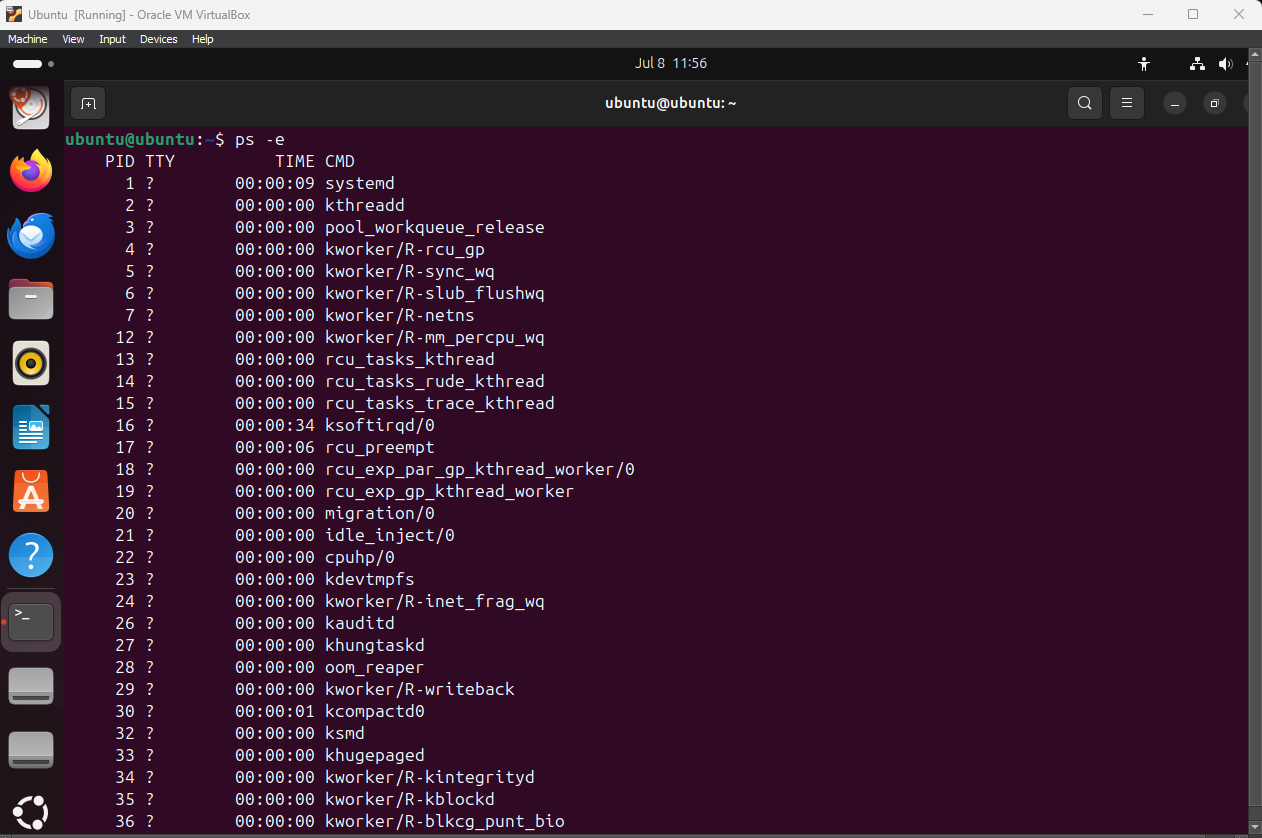
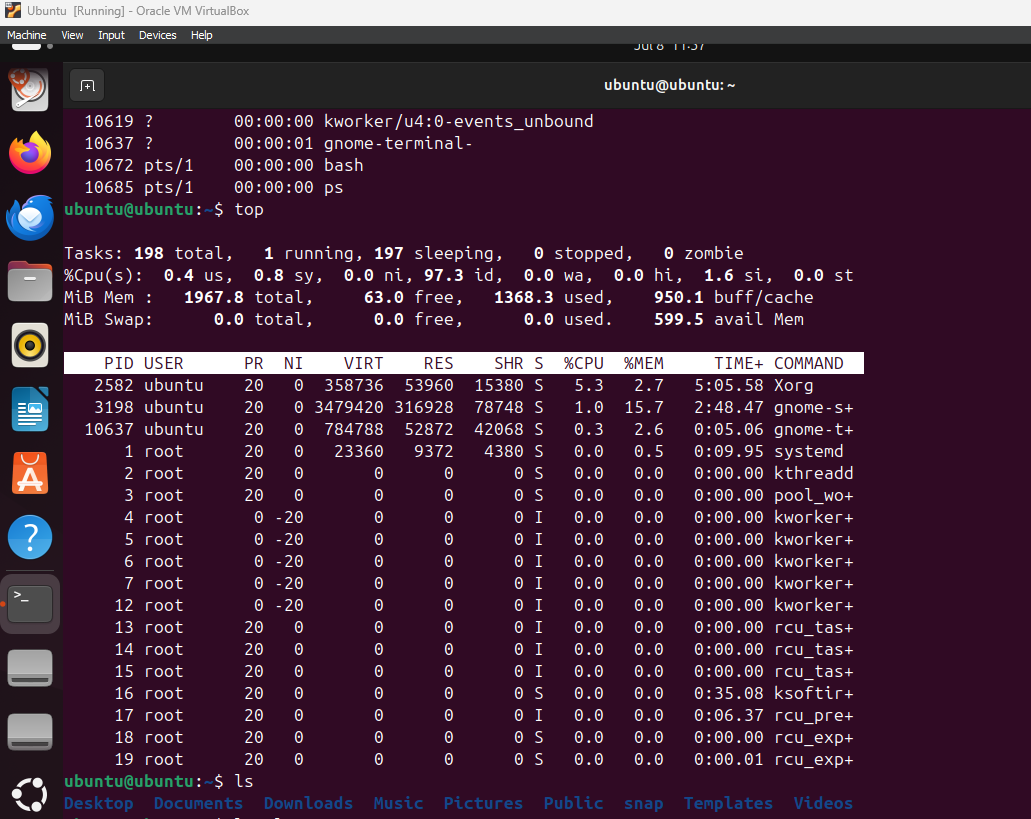
· - Check the Internet using Firefox.  
A computer screen shot of a computer

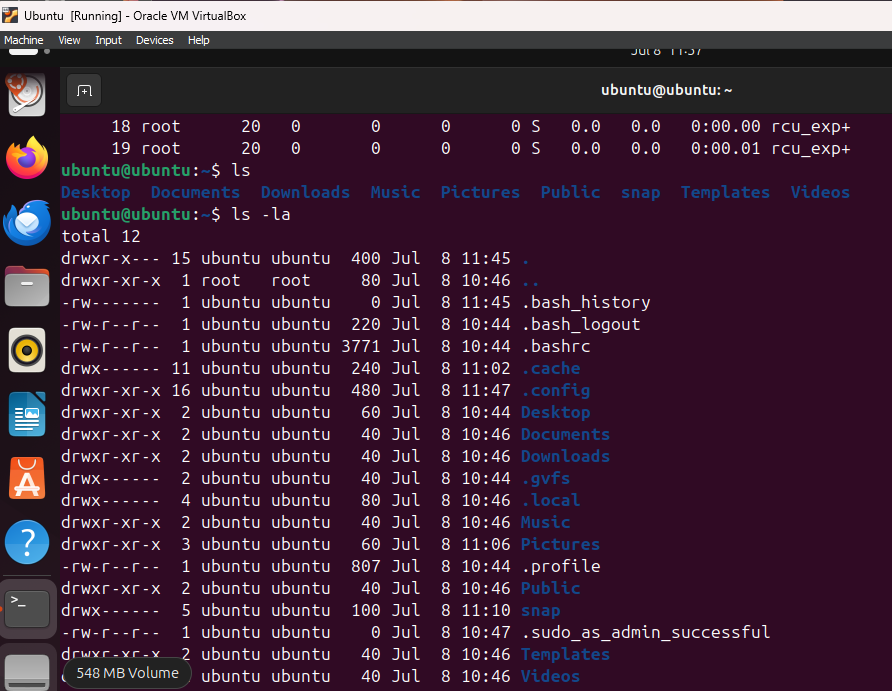
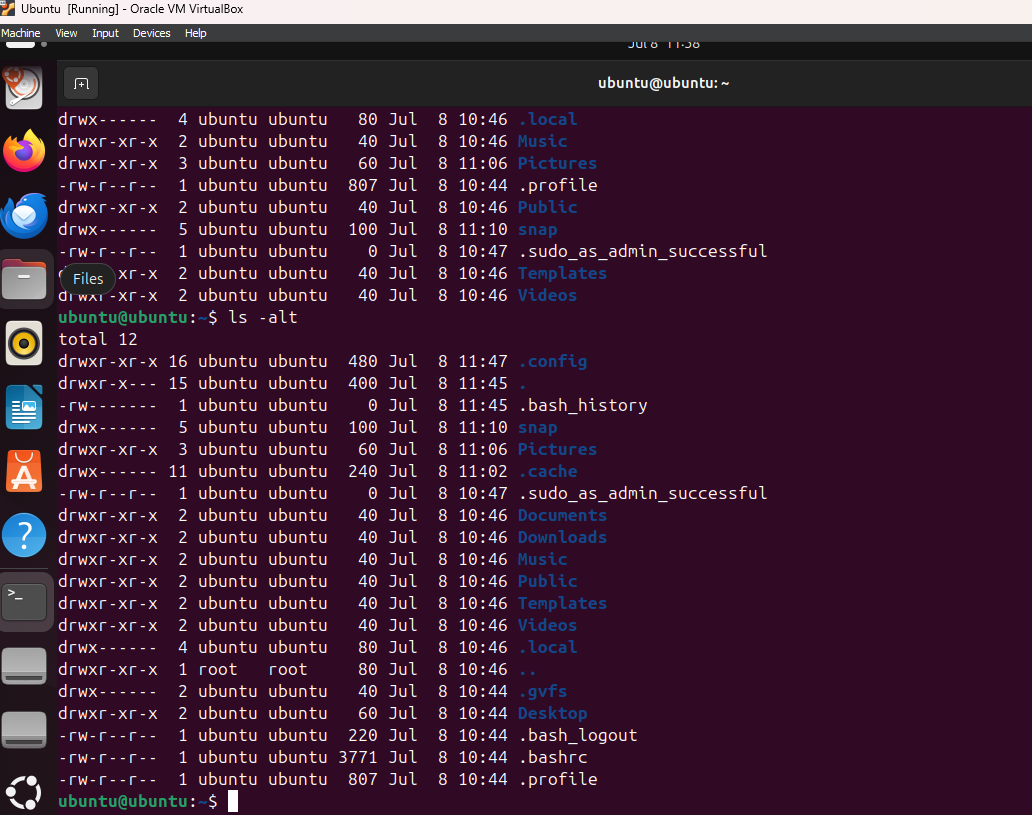
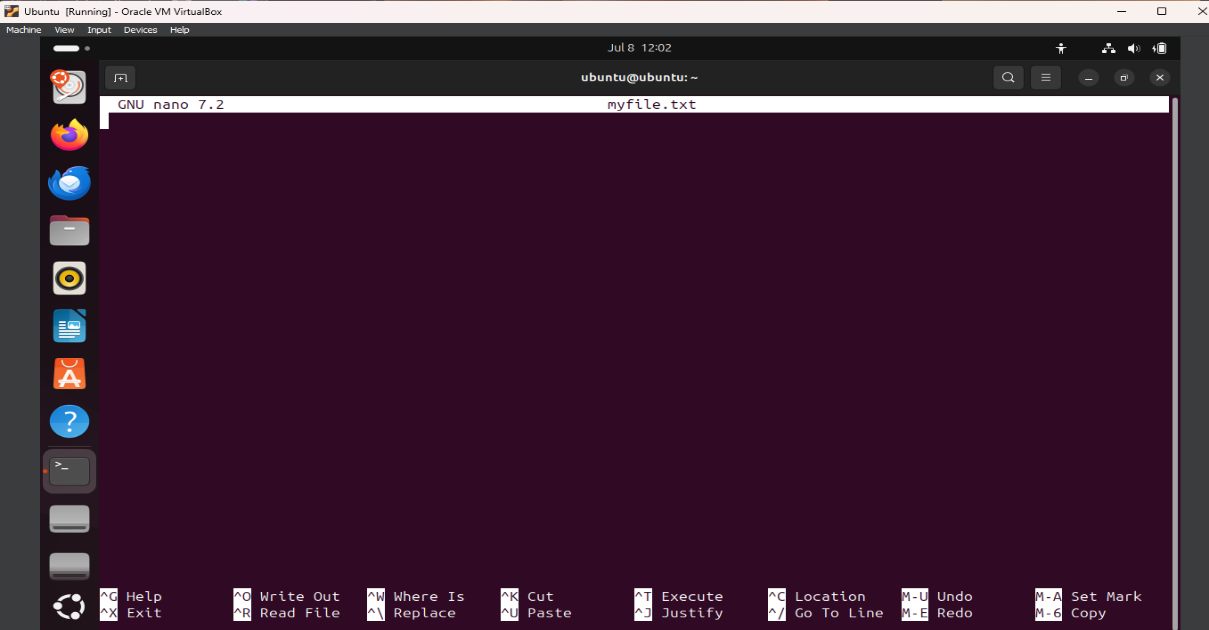
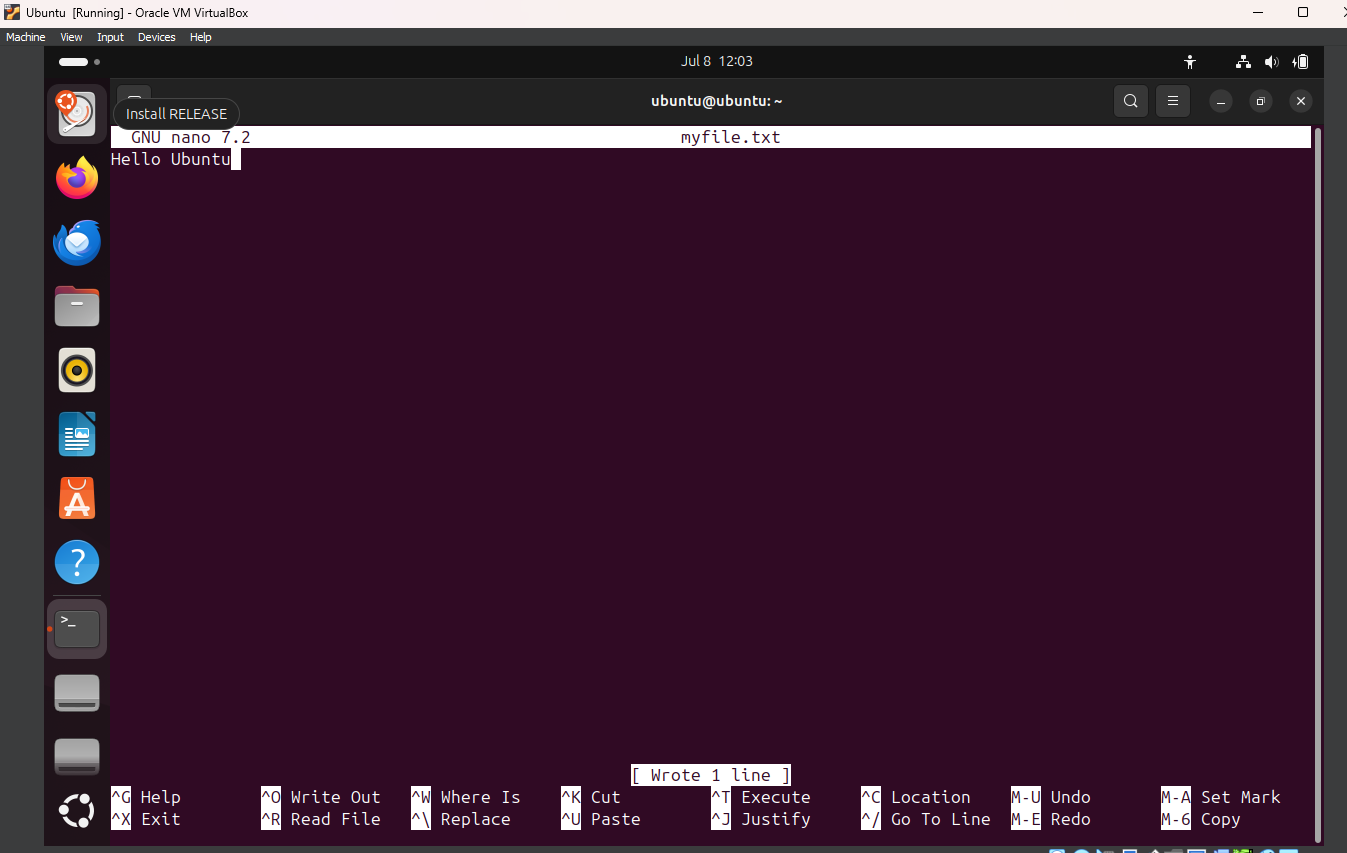
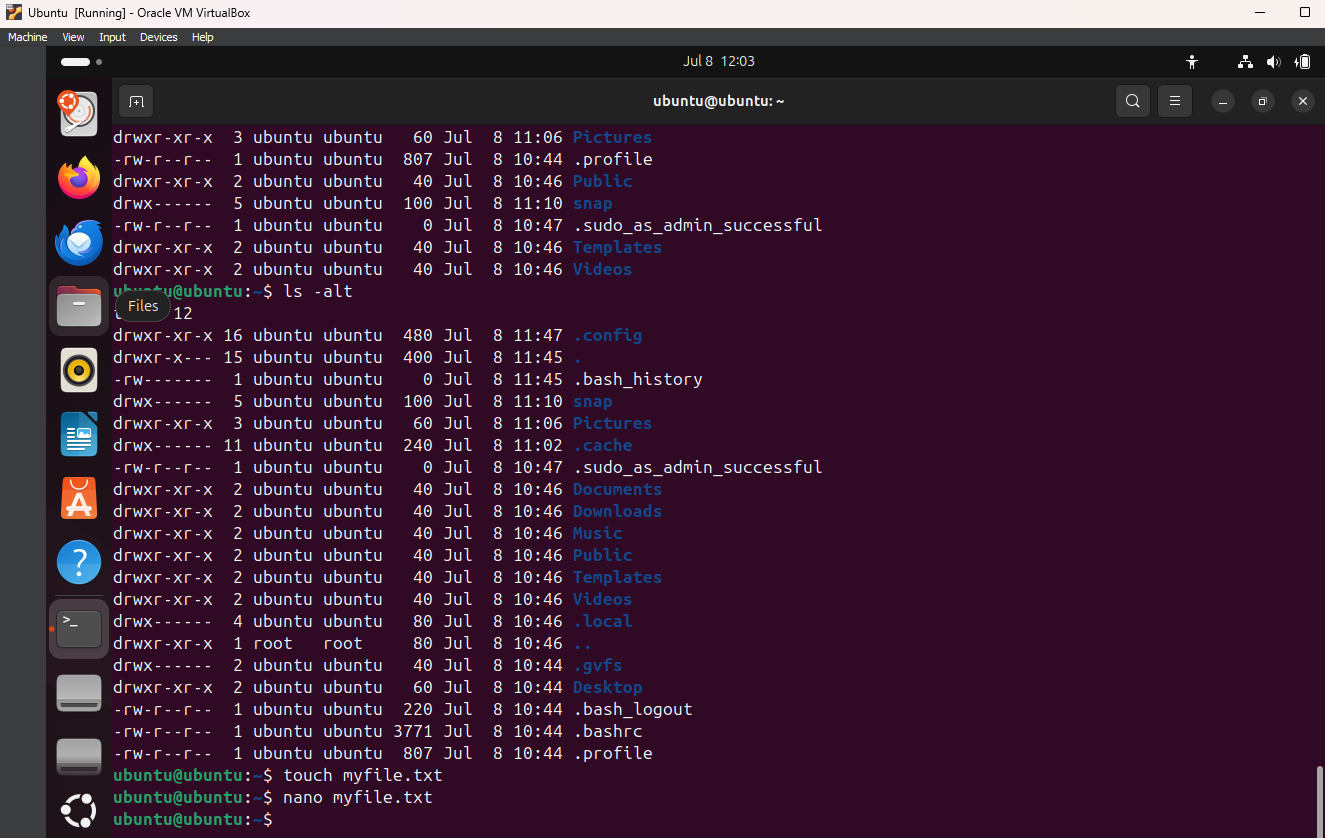
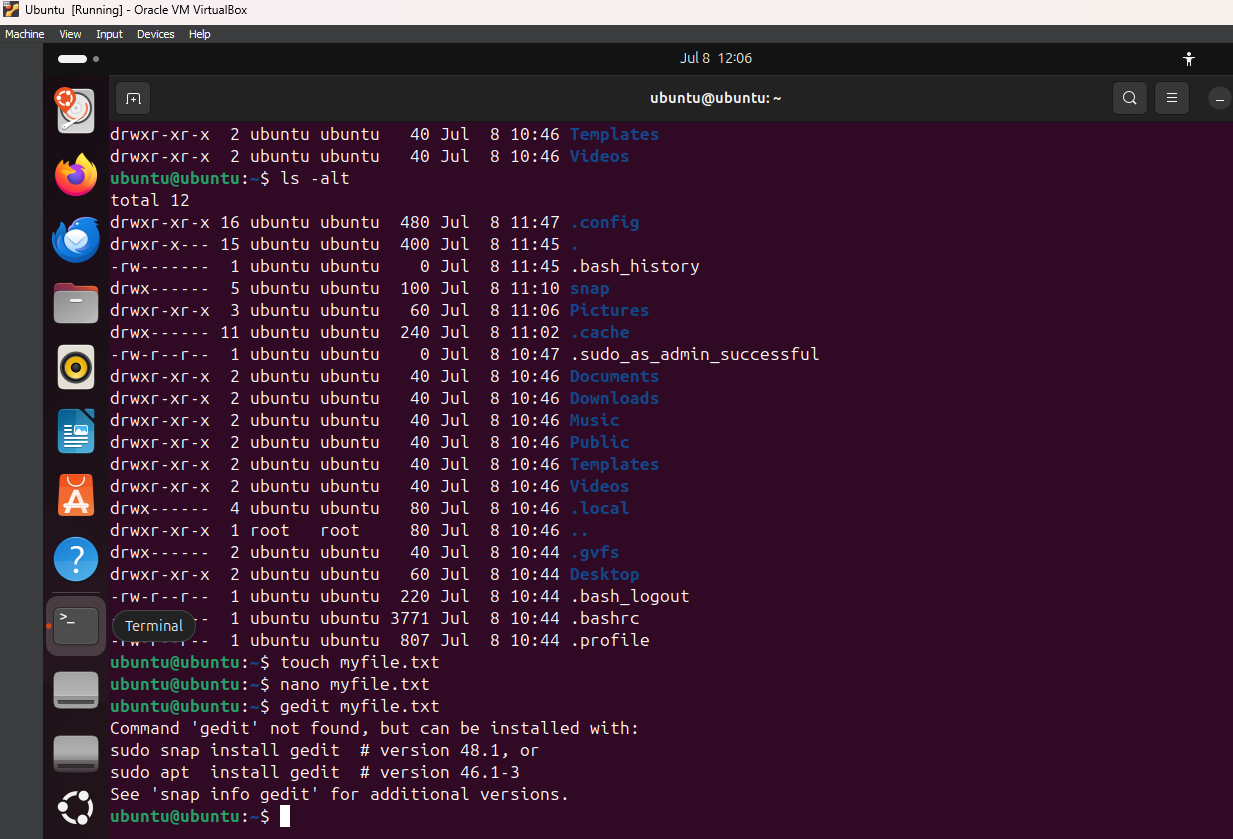
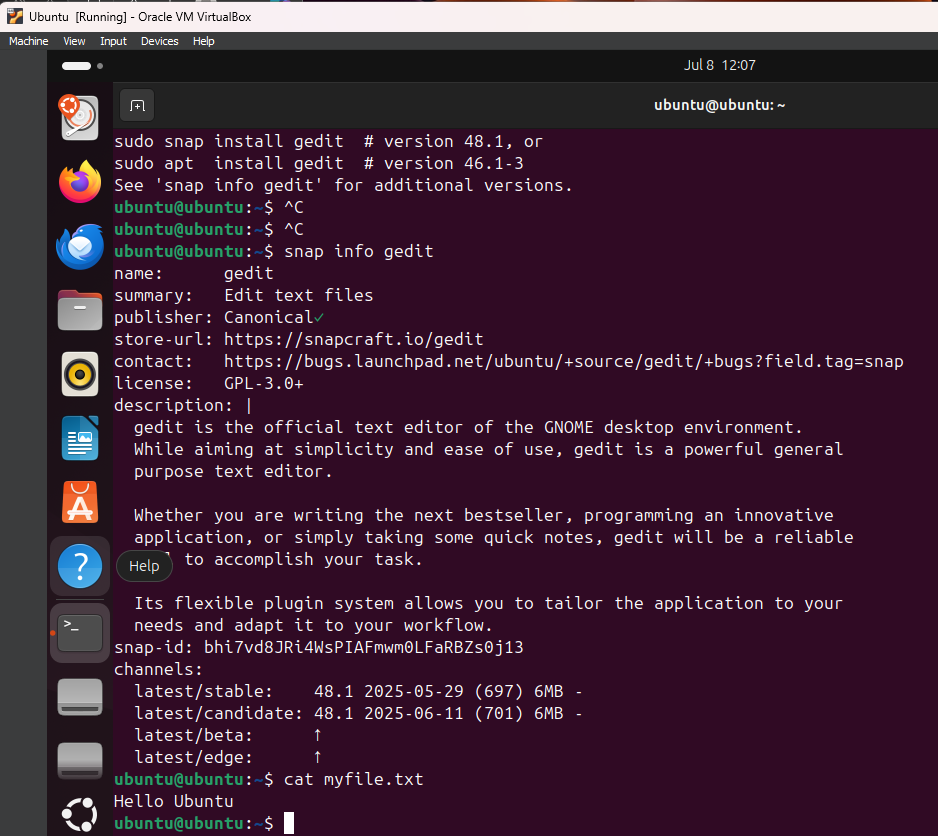
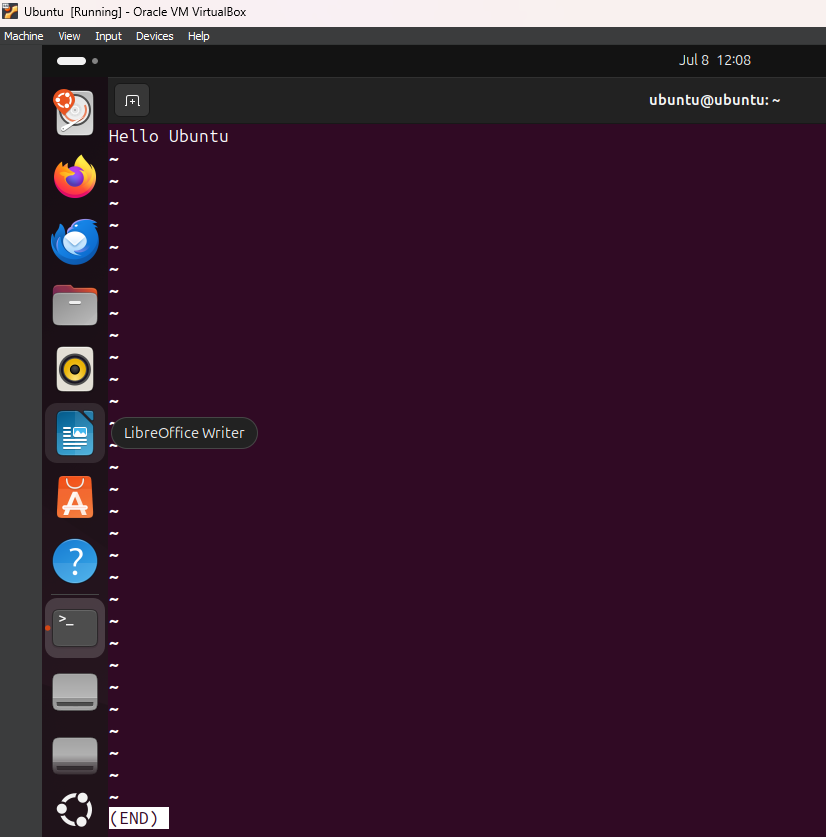
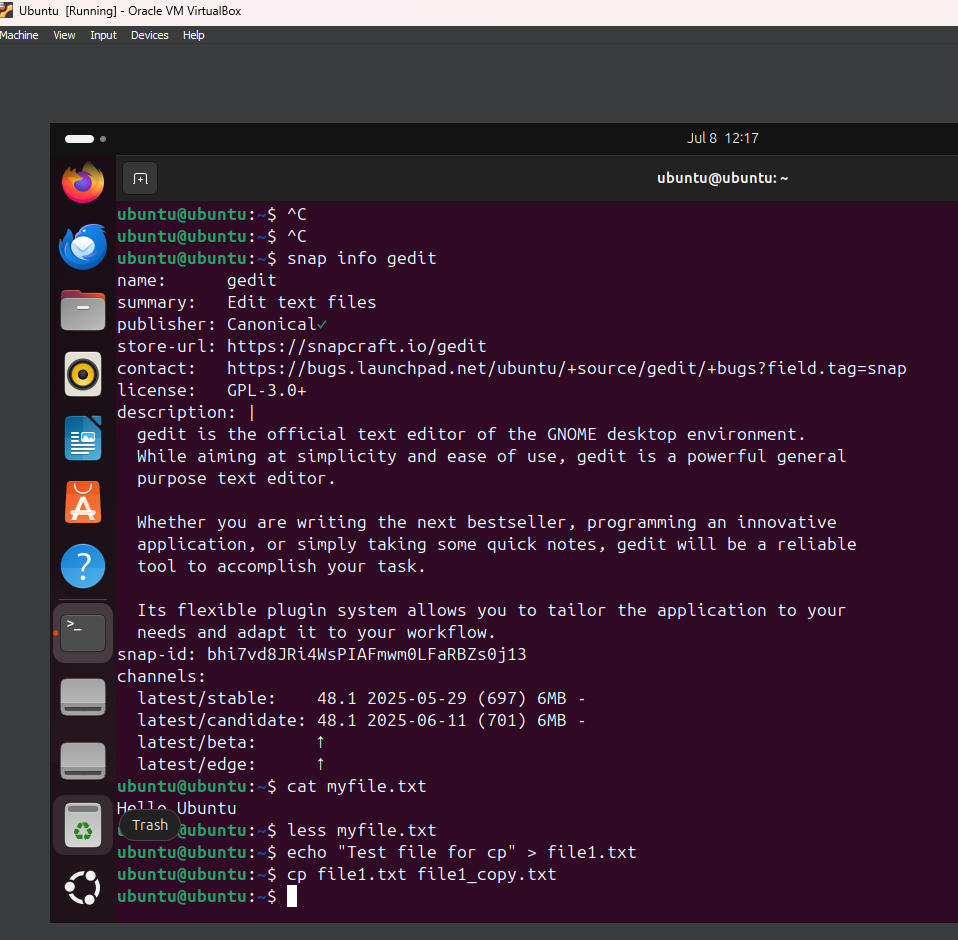
AI-generated content may be incorrect.

· - Open LibreOffice and type a document.  


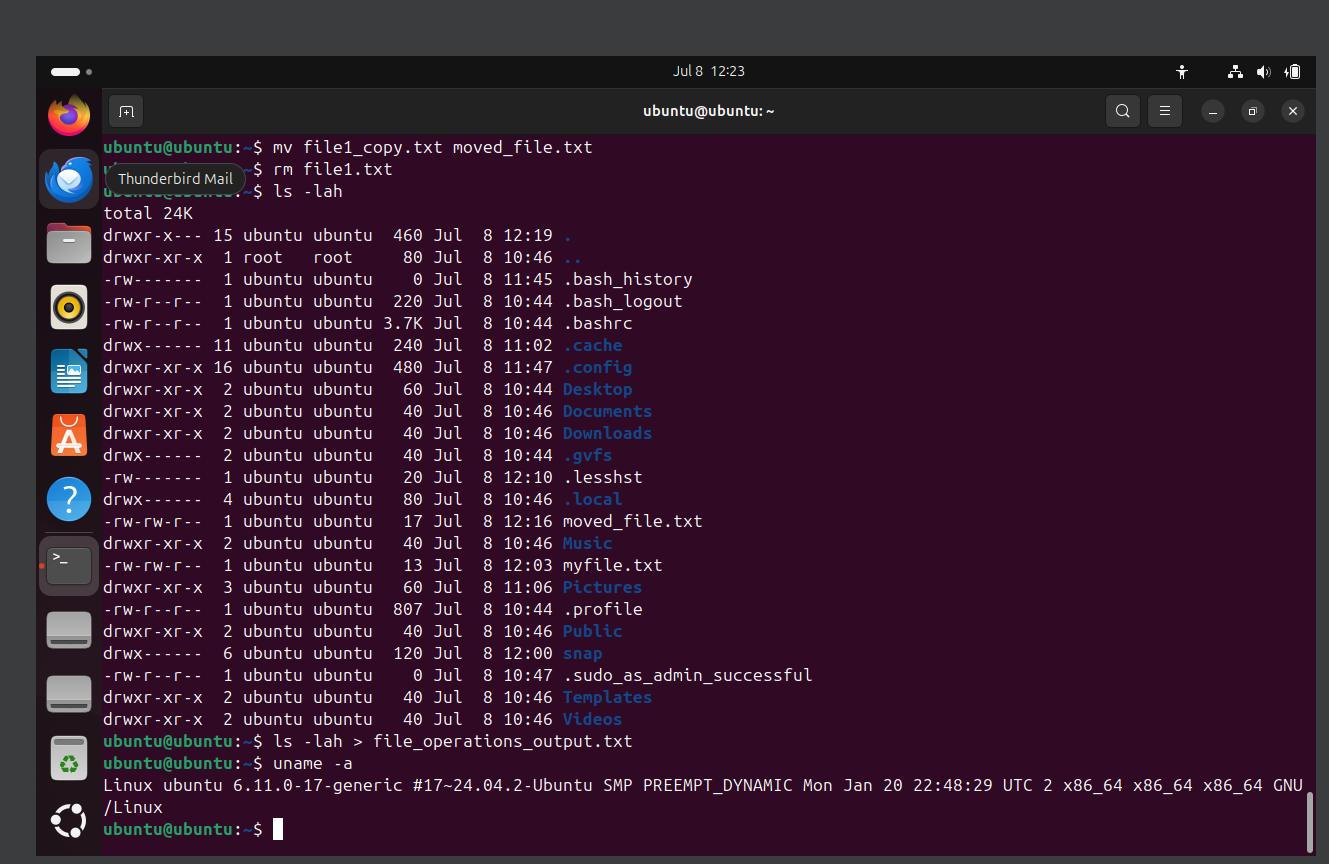
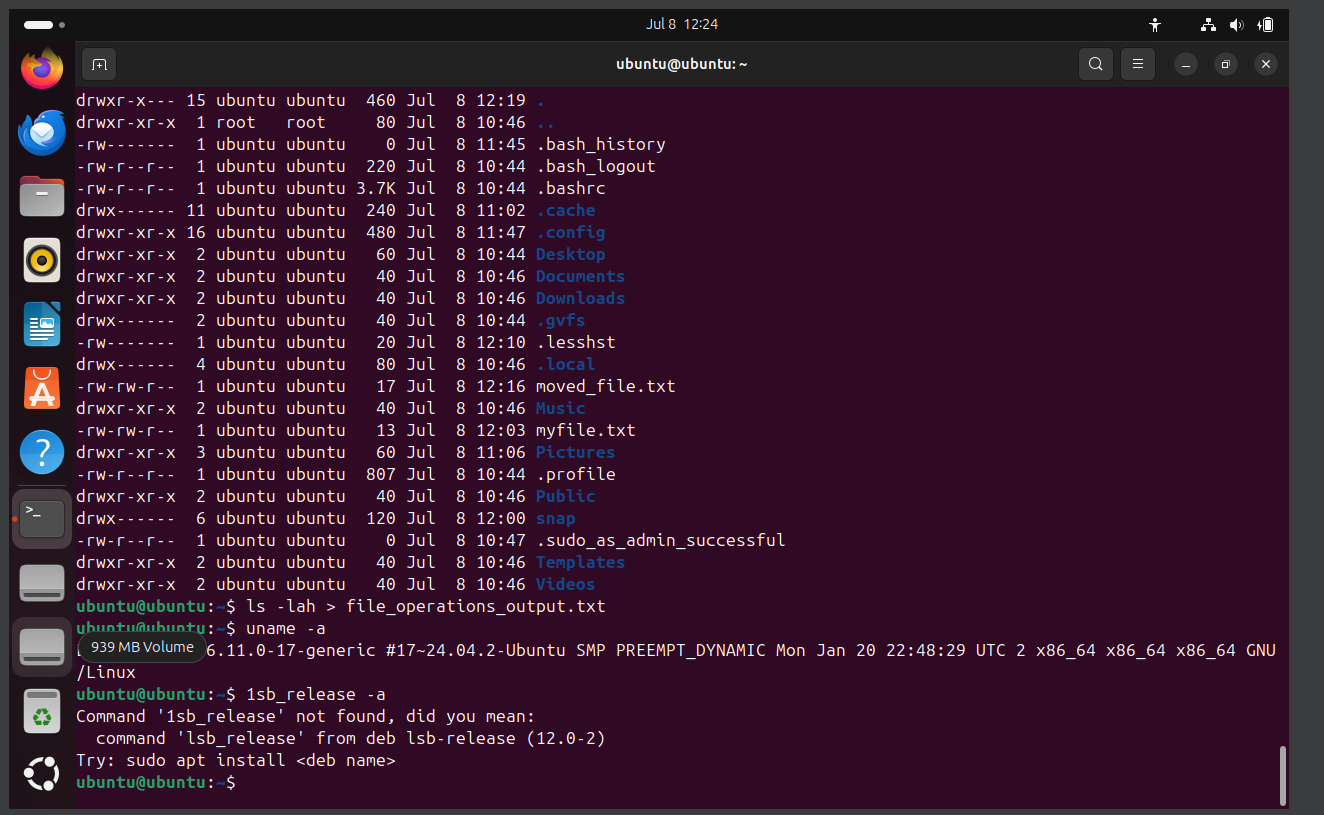
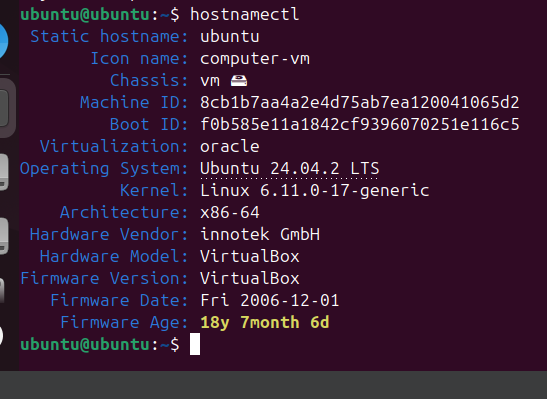
· - Navigate directories using File Manager.  


**Part 2: CLI Basics and File Operations**

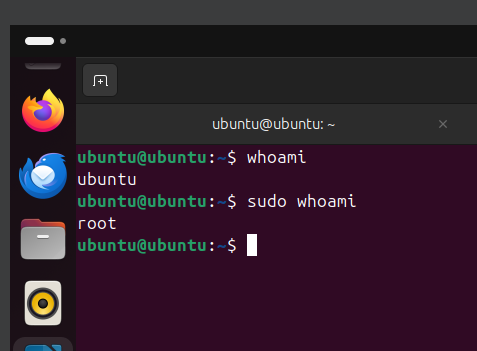
· - Use `ps -e`, `top`, and `1` to monitor processes.  
  
  


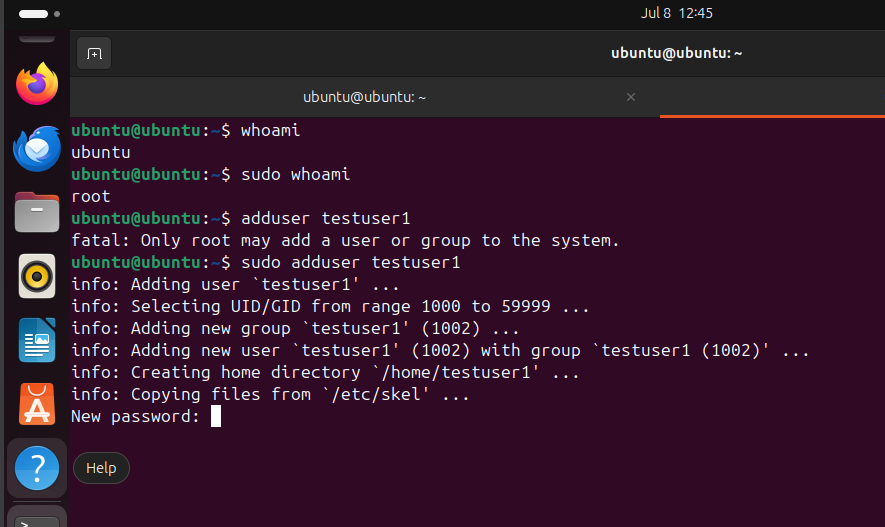
· - Use `ls`, `ls -la`, `ls -alt`, `ls -lah` to explore files.  
  
  
  
  
  
  
  
  
  
  
· - Create/edit/view files using `touch`, `gedit`, `nano`, `cat`, `less`.  
  
  
  
  
  
  


· - Copy and move files with `cp`, `mv`; delete with `rm`.  


· - View system info using `uname -a`, `lsb\_release -a`, `hostnamectl`.  
  
  
  
  


**Part 3: Super User and Permissions**

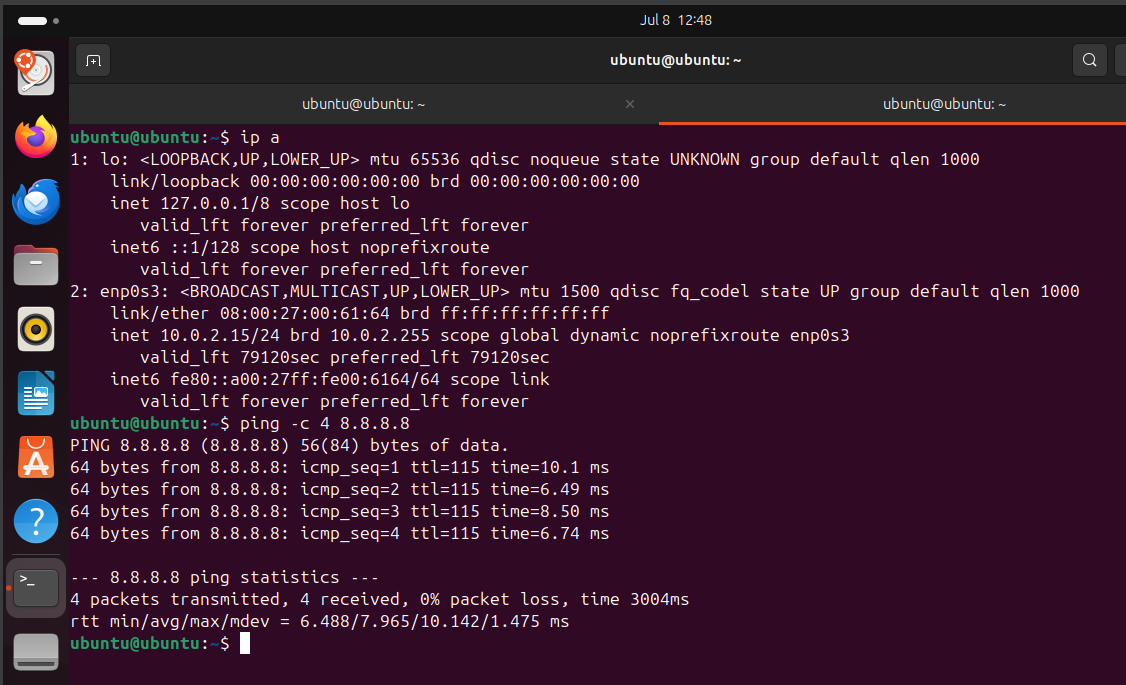
· - Use `whoami`, `sudo whoami` to demonstrate privilege escalation.  


· - Attempt `adduser` as regular user, then with `sudo`.  


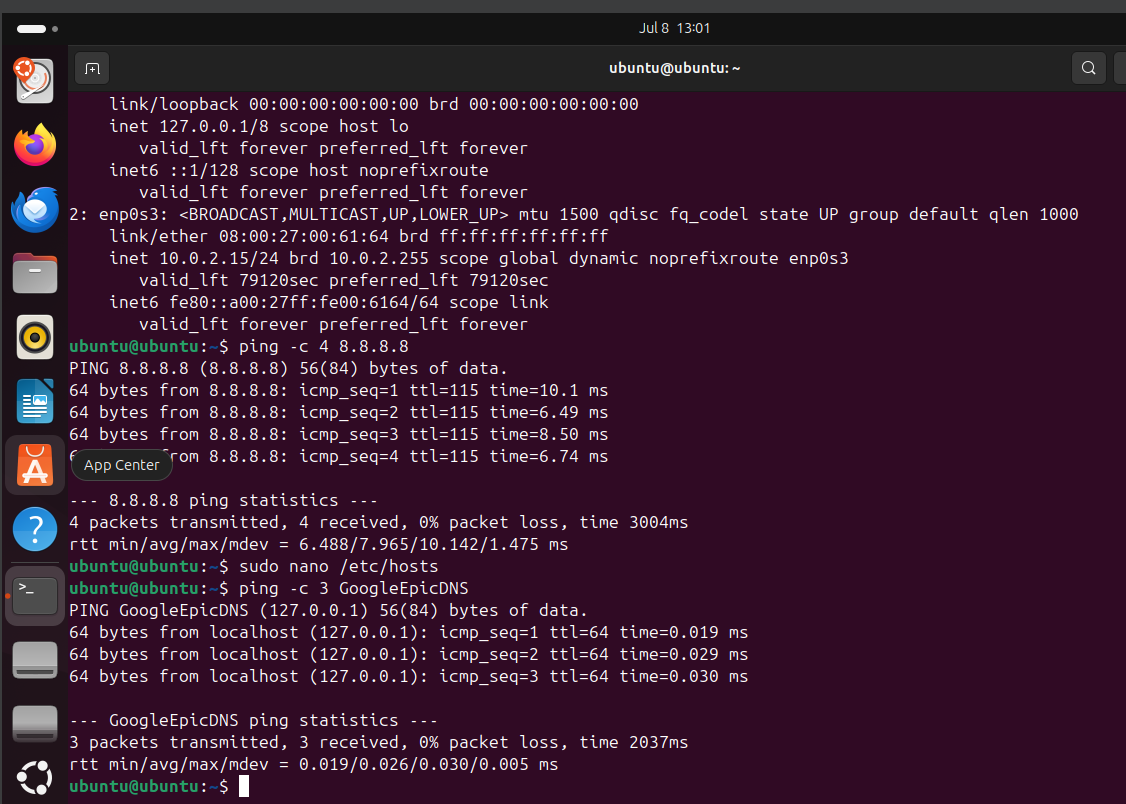
**Part 4: Network Configuration and DNS**

· - Use `ip a` to identify private IP.A screenshot of a computer

AI-generated content may be incorrect.

· - Ping local devices or 8.8.8.8 and discuss DNS.  
  
A screenshot of a computer

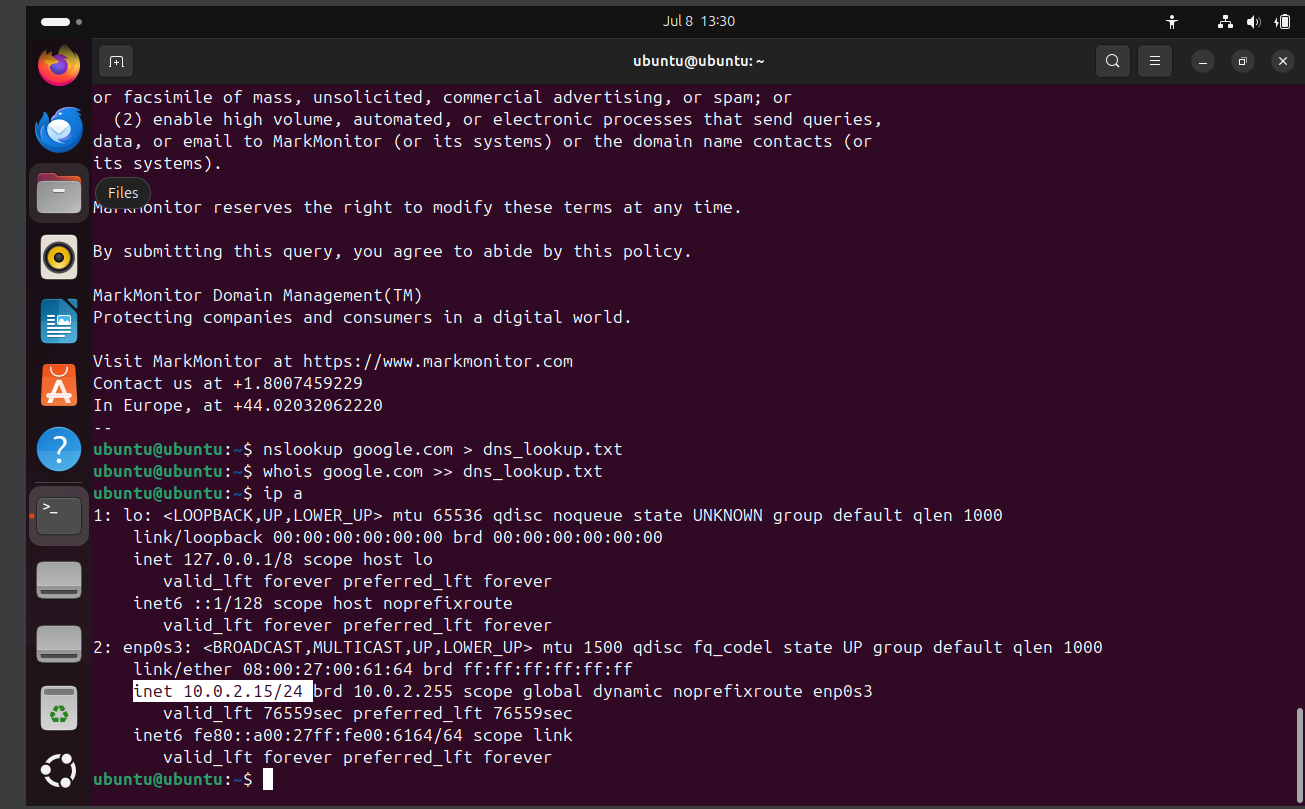
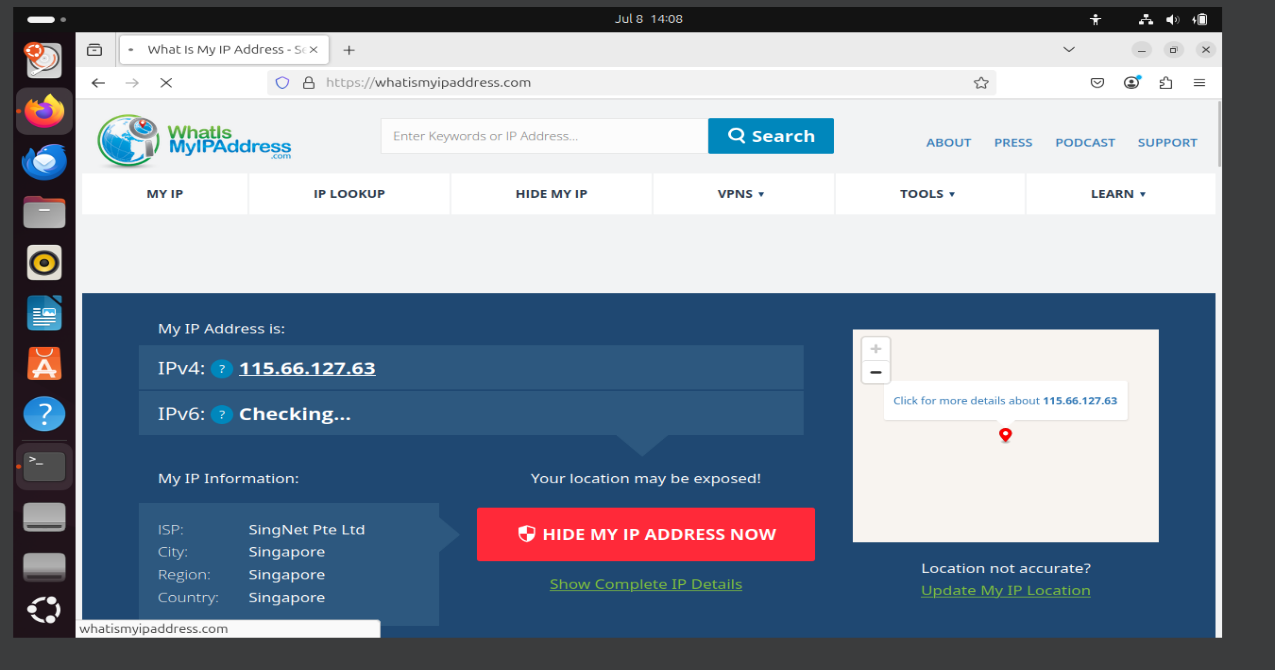
AI-generated content may be incorrect.

· - Edit `/etc/hosts` to create custom aliases; test with `ping`.  


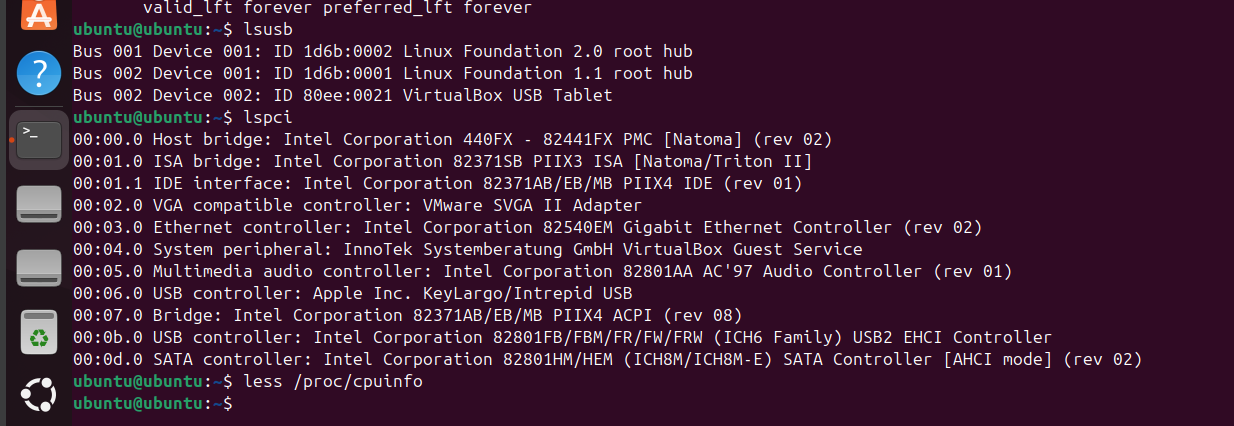
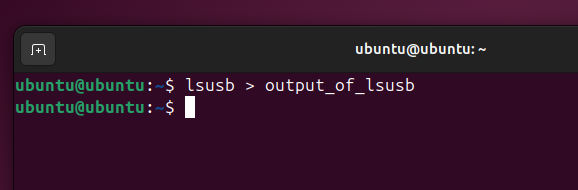
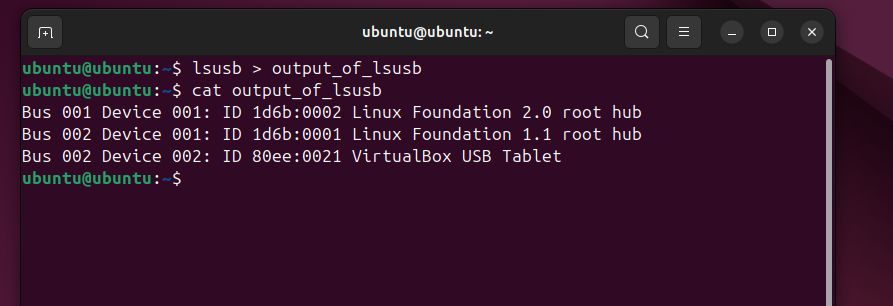
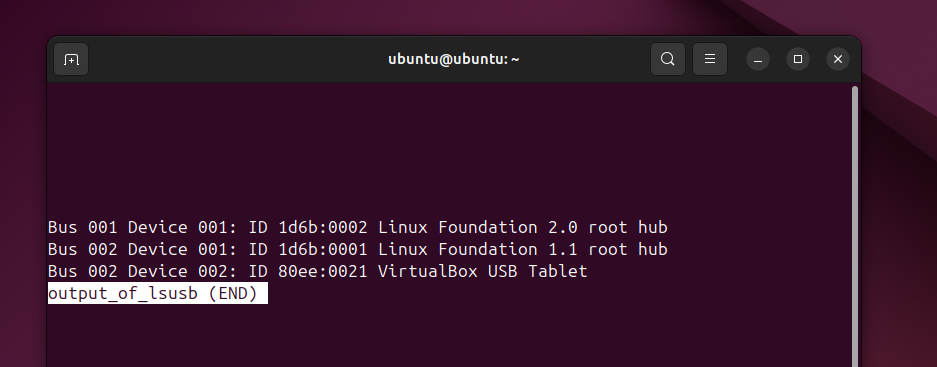
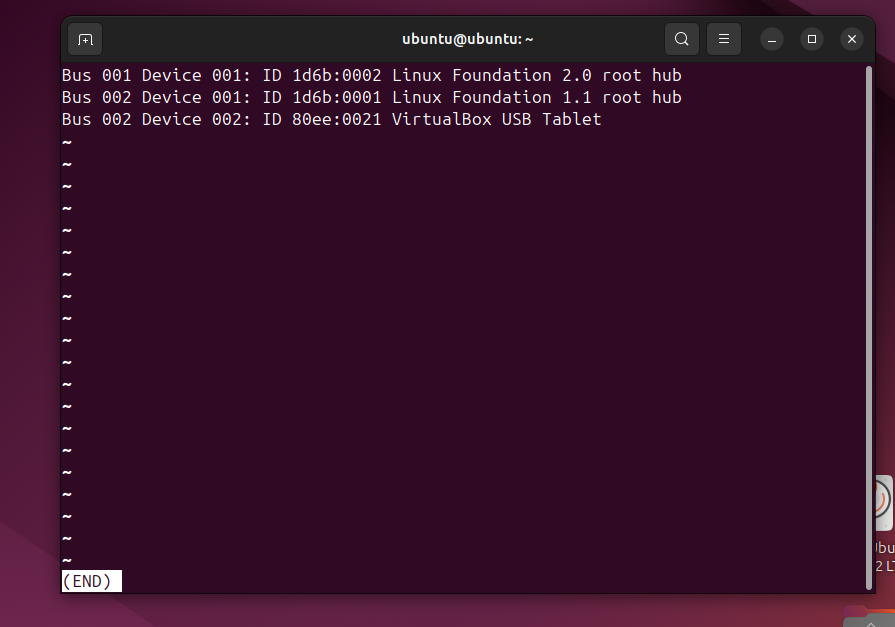
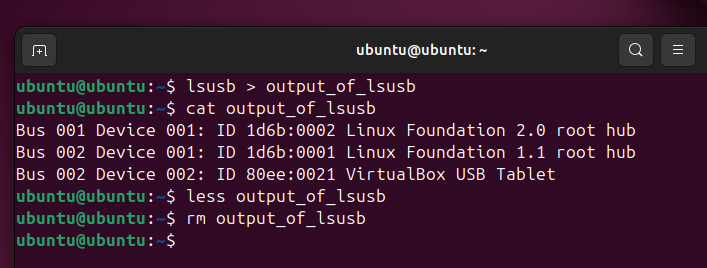
· - Use `nslookup` and install/use `whois` to investigate domains.A screenshot of a computer

AI-generated content may be incorrect.  
  
A screenshot of a computer

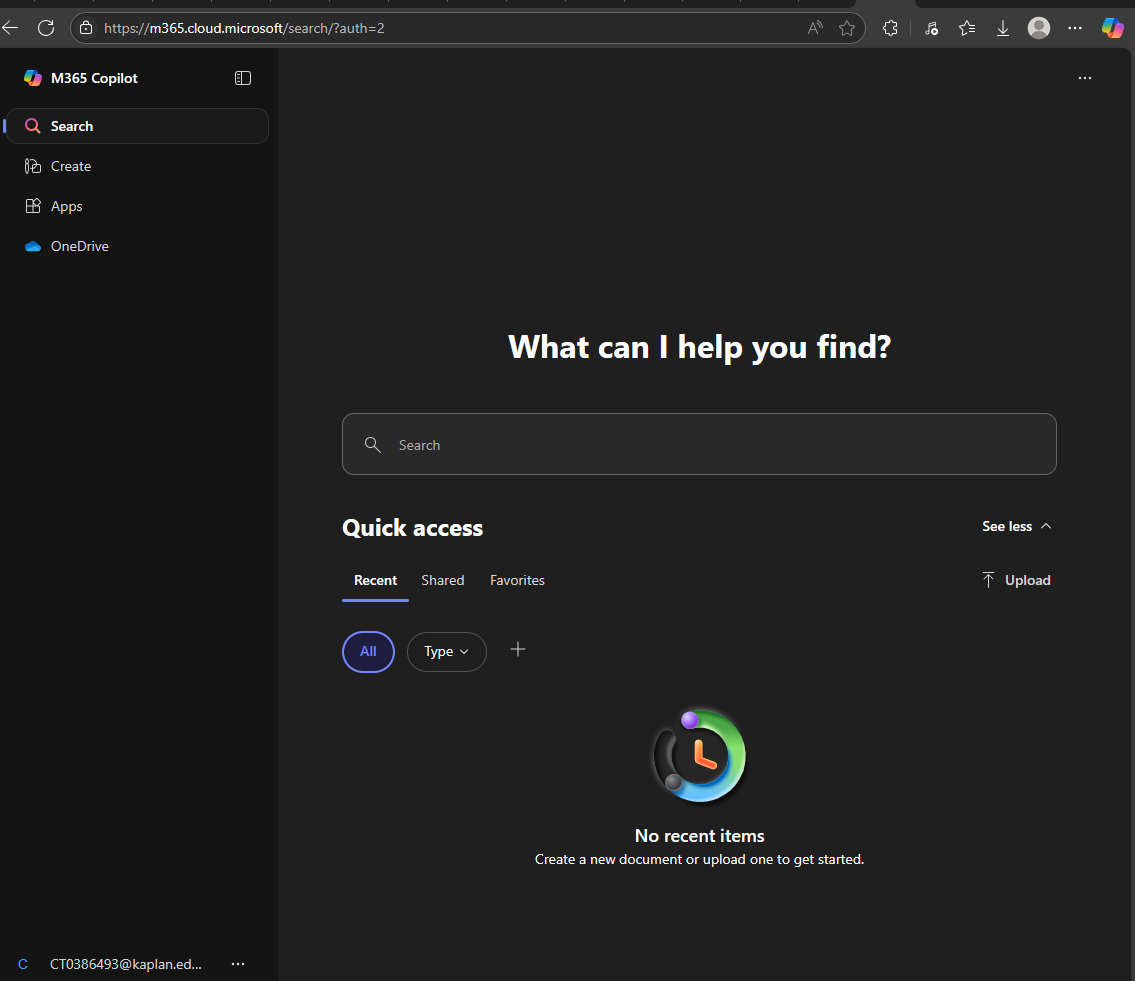
AI-generated content may be incorrect.

· -Compare public/private IP via https://whatismyipaddress.com/.  
  
  


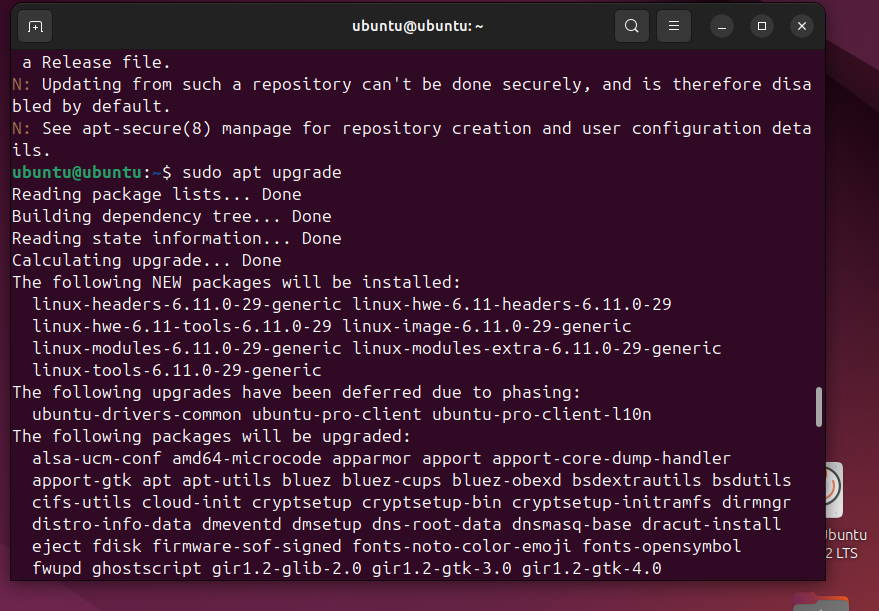
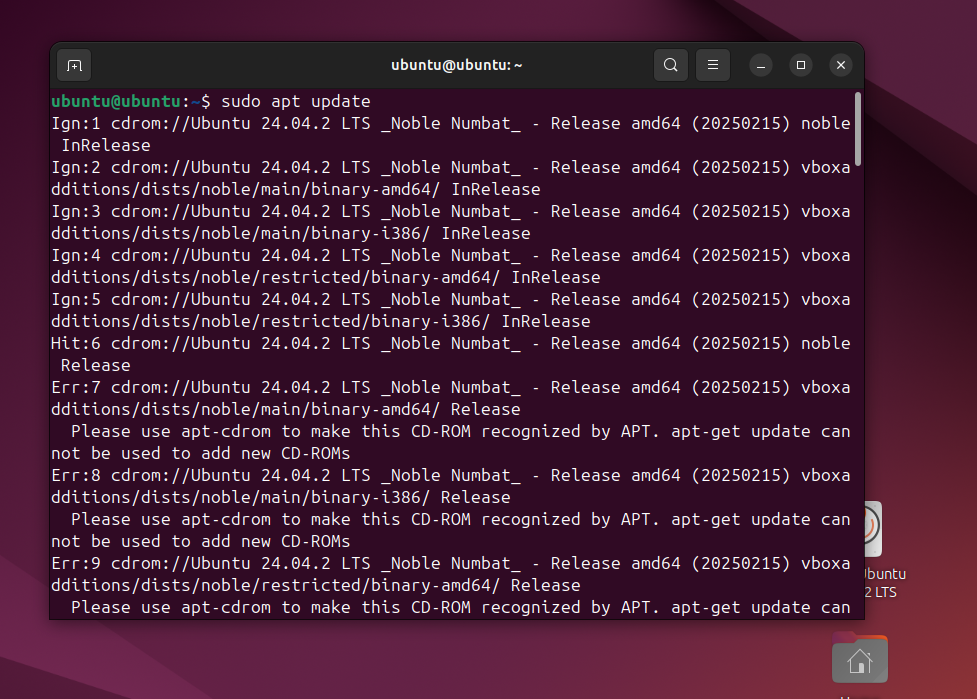
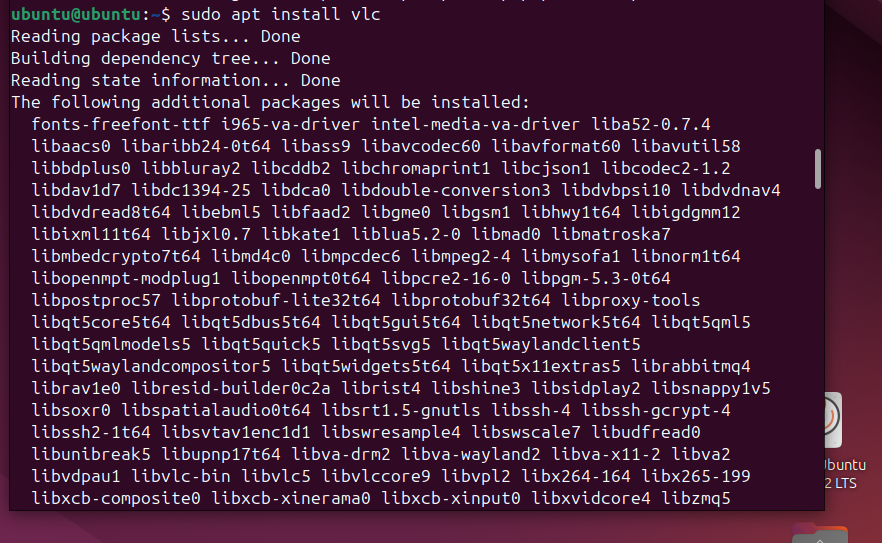
**Part 5: System and Hardware Info**

· - Use `lsusb`, `lspci`, `less /proc/cpuinfo` to inspect hardware.  
  
  
· - Compare CLI outputs with 'About This Computer' in settings.  
  
· - Redirect output using `>`, e.g., `lsusb > output\_of\_lsusb`, then view with `less` and `cat`.  
  
  
  
  


**Part 6: Software Installation Methods**

· - Install a browser (Chrome/Opera) from a .deb file.  


· - Install apps via Ubuntu Software Centre.

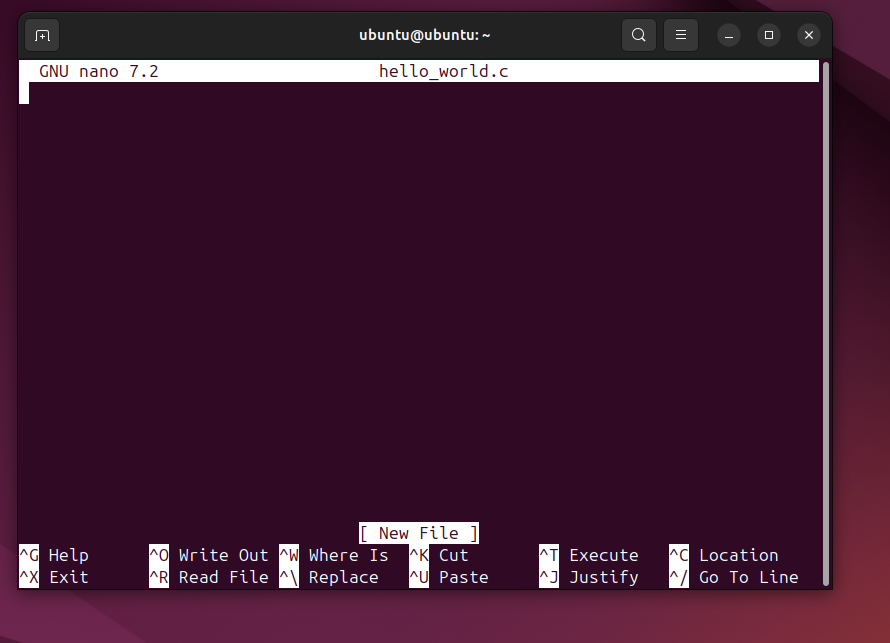
· - Use `sudo apt update`, `sudo apt upgrade`, `sudo apt install vlc`.  
  


· - Search packages with `sudo apt search [keyword]`.

· - Explore `/etc/apt/sources.list` to discuss trusted repositories.

**Part 7: Compiling from Source (Optional)**

· - Install build tools: `sudo apt install build-essential`.

· - Write a basic `hello\_world.c` and compile with `gcc`.  
  


· - Run the executable with `./hello\_world\_executable`.A screenshot of a computer program

AI-generated content may be incorrect.

· - Adjust permissions with `chmod 777` if needed.A screenshot of a computer

AI-generated content may be incorrect.

**1b-1 Linux Services, SSH, Firewalls & Compression**

**✅ 1. Install Apache Web Server**

bash

CopyEdit

sudo apt update

sudo apt install apache2

Visit in browser:  
[**http://127.0.0.1**](http://127.0.0.1)

Screenshot: Apache default page

**✅ 2. Modify index.html**

bash

CopyEdit

sudo nano /var/www/html/index.html

Replace content with something like:

html

CopyEdit

<h1>Hello from [Your Name]!</h1>

Save and access in browser:

* On your machine: <http://127.0.0.1>
* On partner’s machine: http://[your IP]

**✅ 3. Identify and Share IP**

Check your IP:

bash

CopyEdit

ip a

Note:

* **Loopback**: 127.0.0.1
* **Local IP**: e.g., 192.168.1.10

Exchange IPs with your partner and try accessing their Apache page via browser.

**✅ 4. Nmap Port Scan**

Install Nmap if needed:

bash

CopyEdit

sudo apt install nmap

Scan before Apache removal:

bash

CopyEdit

nmap [partner IP]

Then stop Apache:

bash

CopyEdit

sudo systemctl stop apache2

Scan again to show port 80 is now closed.

**✅ 5. Firewall (UFW) Configuration**

Check status:

bash

CopyEdit

sudo ufw status verbose

Enable UFW:

bash

CopyEdit

sudo ufw enable

Allow port 80:

bash

CopyEdit

sudo ufw allow 80/tcp

Confirm:

bash

CopyEdit

sudo ufw status

Partner can scan again with nmap [your IP] to verify port 80 is open.

**✅ 6. Enable and Test SSH**

Install and start SSH:

bash

CopyEdit

sudo apt install openssh-server

sudo systemctl enable ssh

sudo systemctl start ssh

Partner connects:

bash

CopyEdit

ssh yourusername@[your IP]

Try both:

bash

CopyEdit

ssh [IP] # default user

ssh username@[IP] # explicitly declared user

**✅ 7. Create a New User**

bash

CopyEdit

sudo adduser newstudent

Then verify:

bash

CopyEdit

grep newstudent /etc/passwd

**✅ 8. Compression & Decompression**

**Create a file and compress it:**

bash

CopyEdit

echo "sample" > test.txt

tar cf archive.tar test.txt

bzip2 archive.tar # becomes archive.tar.bz2

bunzip2 archive.tar.bz2 # unzips back

tar -xvf archive.tar # extract file

List files with size:

bash

CopyEdit

ls -la

**✅ 9. SCP File Transfer**

**Copy a file:**

bash

CopyEdit

scp test.txt user@partnerIP:~

**Bonus: Copy a directory:**

bash

CopyEdit

scp -r folder/ user@partnerIP:~

**✅ 10. Modify /etc/hosts**

bash

CopyEdit

sudo nano /etc/hosts

Add:

CopyEdit

8.8.8.8 GoogleEpicDNS

Then:

bash

CopyEdit

ping -c 3 GoogleEpicDNS

**✅ 11. DNS Lookup and Whois**

bash

CopyEdit

nslookup google.com

sudo apt install whois

whois google.com

**✅ 12. Public vs Private IP**

bash

CopyEdit

ip a

Visit <https://whatismyipaddress.com> in browser.

Write 1–2 lines:

My local IP is private and used inside LAN, while my public IP is assigned by my ISP for internet access.

**✅ 13. Extract Hardware Info**

bash

CopyEdit

lsusb

lspci

less /proc/cpuinfo

To get CPU core count:

bash

CopyEdit

grep -c ^processor /proc/cpuinfo

**✅ 14. Redirect Output**

bash

CopyEdit

lsusb > output\_of\_lsusb

cat output\_of\_lsusb

less output\_of\_lsusb

ls -la output\_of\_lsusb

rm output\_of\_lsusb

**✅ 15. SSH and Create a File on Partner’s Desktop**

bash

CopyEdit

ssh user@partnerIP

touch /home/user/Desktop/Hi\_[partnername]

exit

**✅ 16. Launch gedit Over SSH**

bash

CopyEdit

ssh user@partnerIP

gedit

Expected:

Error or nothing happens — GUI apps over SSH require X11 forwarding. Not available by default without extra setup.

**✅ 17. SCP Multiple Files**

bash

CopyEdit

scp file1.txt file2.txt file3.txt user@partnerIP:~/Downloads

If successful, you’ll see transfer logs with 100%.

**✅ 18. Bonus: Gutenberg Books**

1. Download:

bash

CopyEdit

wget http://www.gutenberg.org/files/1342/1342-0.txt -O pride.txt

(Do this for 10 books.)

1. Compress:

bash

CopyEdit

tar cf books.tar \*.txt

bzip2 books.tar

1. Send to partner:

bash

CopyEdit

scp books.tar.bz2 user@partnerIP:~

✅ 20. Reflection Paragraph – Linux Networking, SSH & File Transfer

Working on Linux networking and file transfer tasks gave me a solid understanding of how systems communicate and interact securely. Setting up the Apache server and seeing a personalized web page load from another machine was incredibly satisfying—it made the network feel real and alive. Enabling SSH and accessing another machine remotely made me realize how powerful and flexible Linux is for system administration. Learning tools like scp for file transfers and nmap for port scanning helped me understand both functionality and security aspects of networking.

Compression using tar and bzip2 was also interesting—it showed how important it is to manage storage and bandwidth, especially when transferring files between machines. One challenge was setting up GUI tools like gedit over SSH, which introduced me to the concept of X11 forwarding and its limitations. Overall, this hands-on experience deepened my confidence in managing servers, networking, and secure communication between systems. I now feel better prepared to work in real-world environments where remote access, web hosting, and file handling are daily tasks.