Sri Lanka Institute of Information Technology

Student ID – IT23265592

Name – R.M.P.U . Rajapaksha

Date of submission: 6th October 2024

IE2012 – Systems and Network Programming

B.Sc. (Hons) in Information Technology

specializing in Cyber Security.

Table of Contents

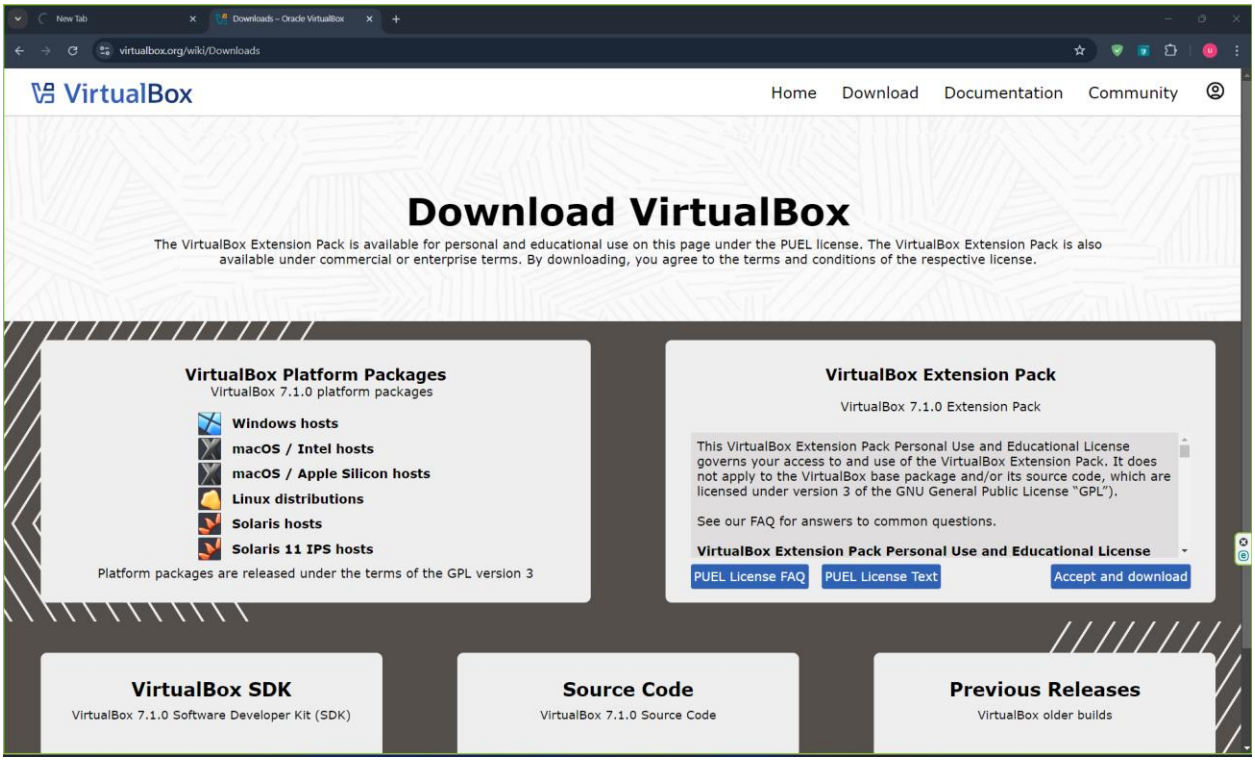
Basic of Linux Environment

Virtual Machine Setup

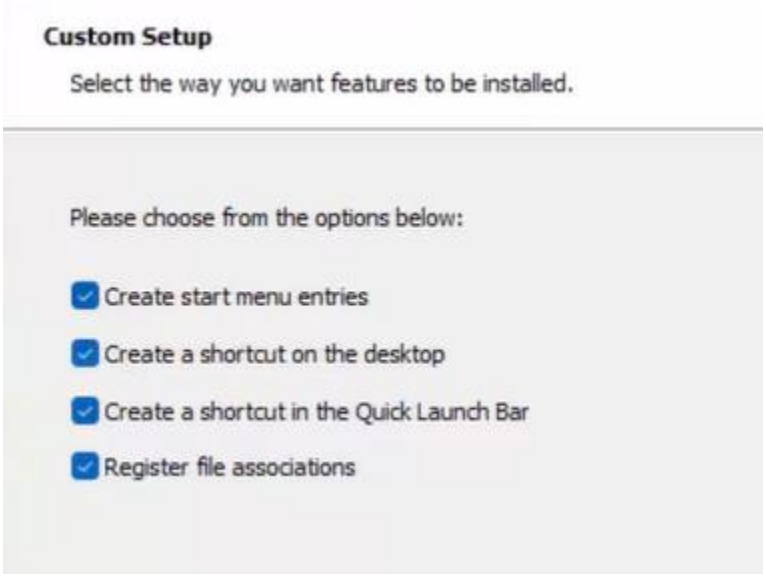
You have to download Virtual Box from the official website for your Windows, Mac or Linux operating system to run virtual machine.

Use following link to download:

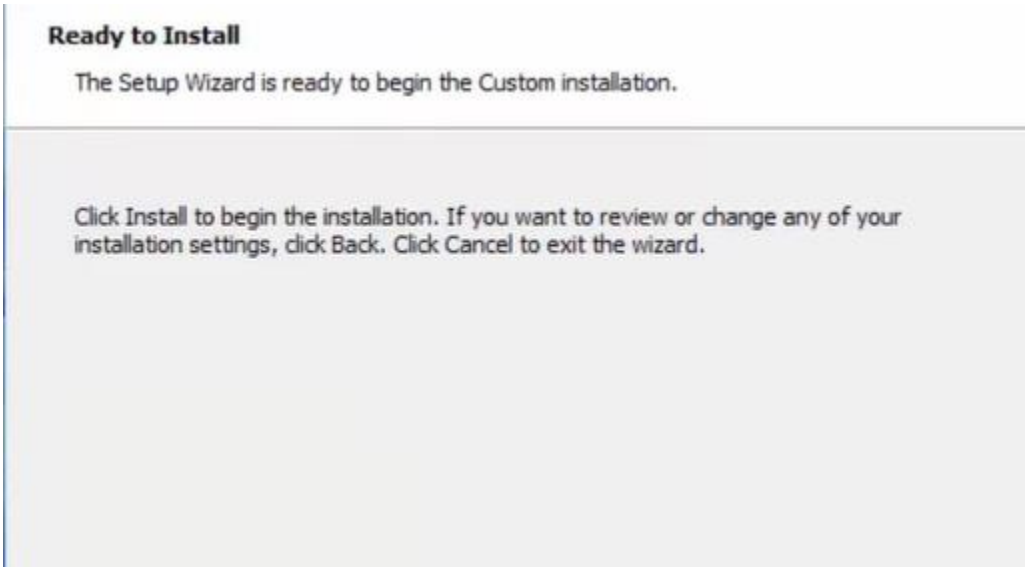
https://www.virtualbox.org/wiki/Downloads



- When successfully downloaded the virtual box, then you must install it by run and the administrator privilege supplement pop-up has been allowed.
- You have to choose method and the location that the virtual box has to be installed.
- Then you have to choose from option following:
    - Create start menu entries
    - Create a shortcut on the desktop
    - Create a shortcut in the Quick Launch Bar
    - Register file associations
- Choose all from these check box



- Then select “NEXT” button and then click on the “Install” button

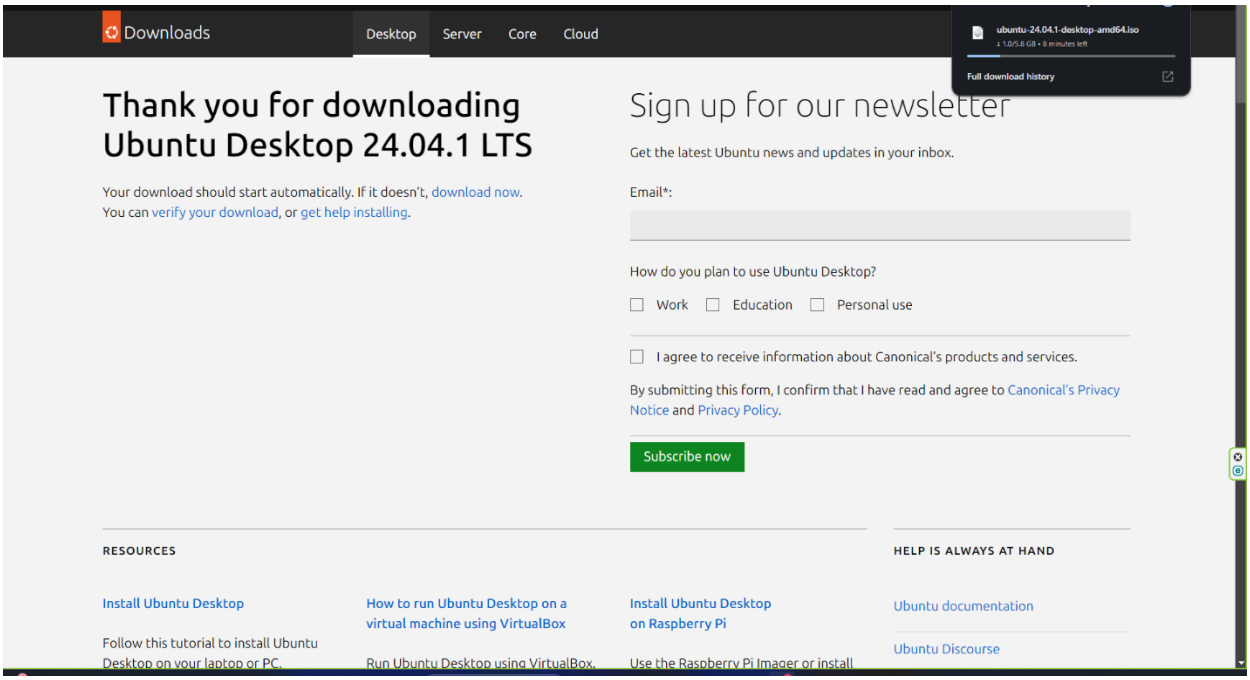


- After the successfully installed virtual box, you can see the icon of virtual box in Desktop.

- When click on the virtual box you can add virtual machine to your virtual box.
- Before add a virtual box , you have to download it.

- You can download Ubuntu virtual machine using following link

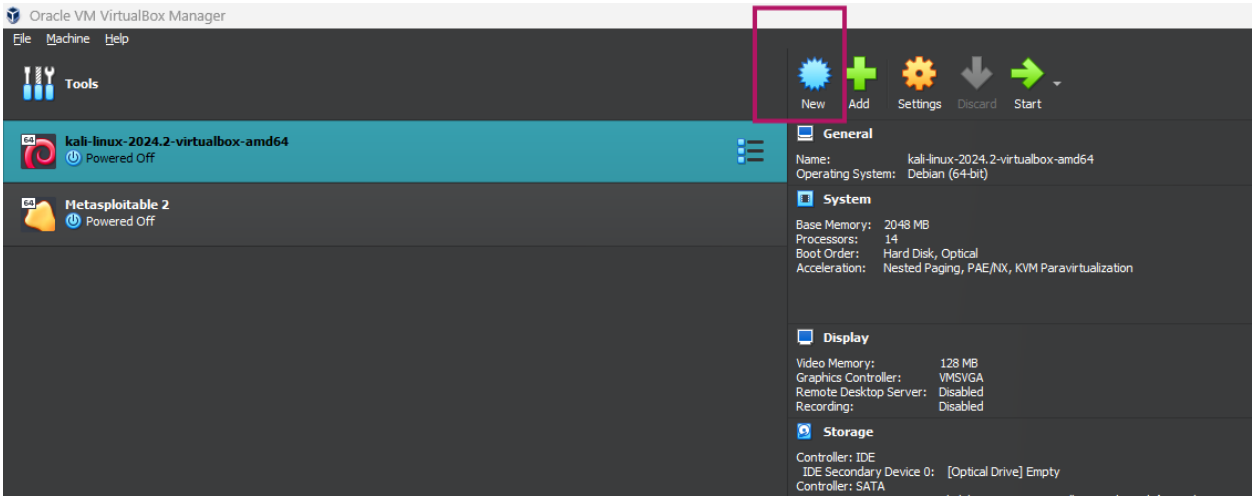
  https://ubuntu.com/download/desktop



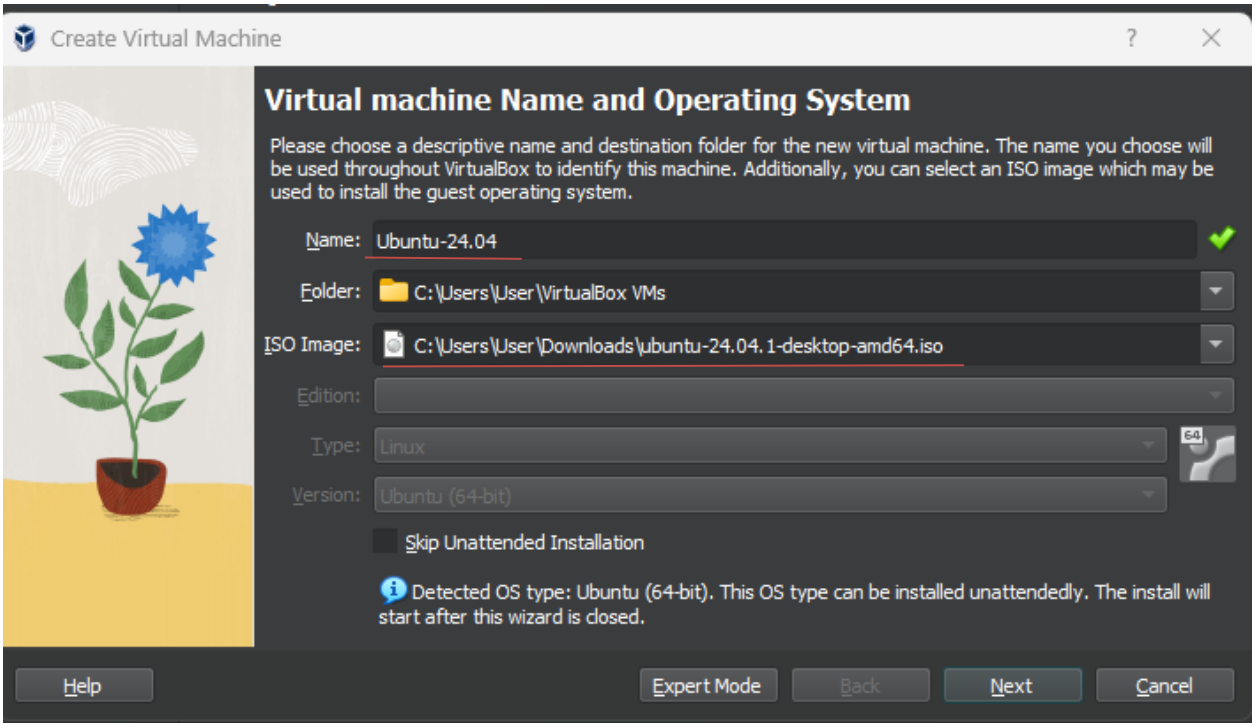
- After the successfully download Ubuntu, then you have to install it.

Steps for Ubuntu Installations:

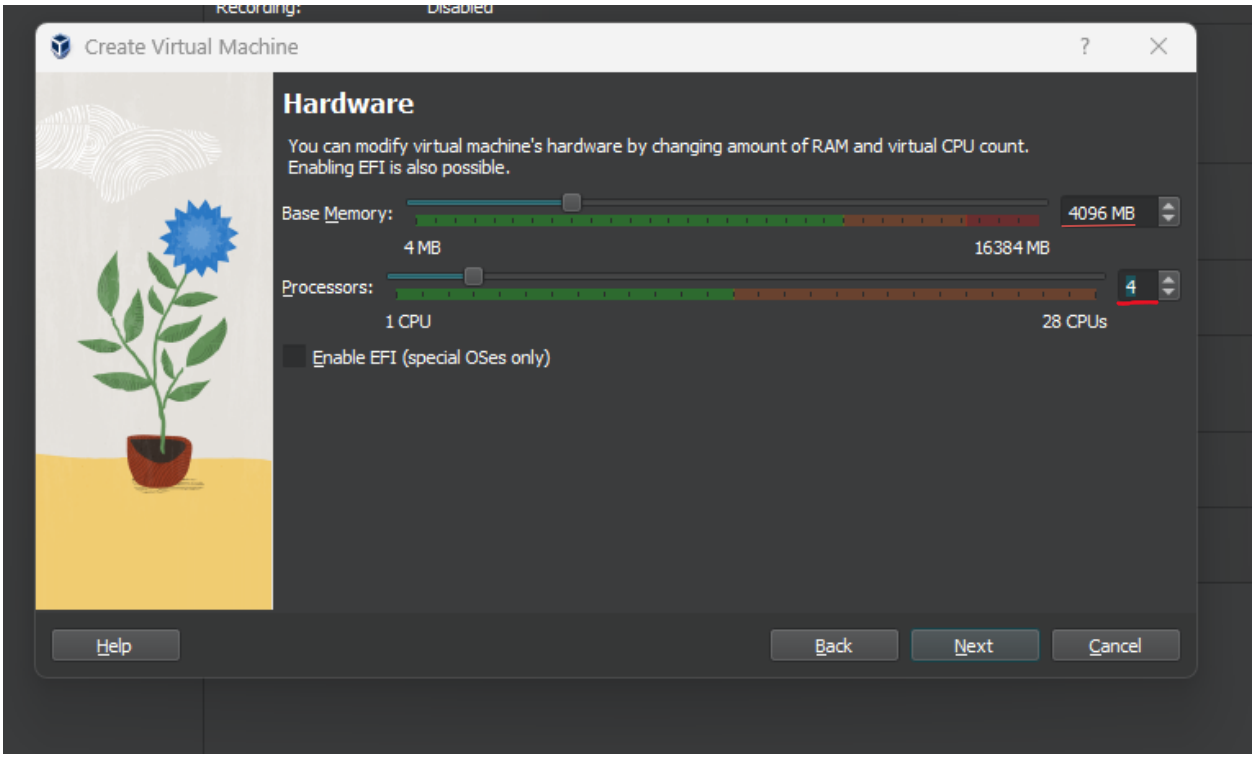
- Open VirtualBox and Click on “New” button to create a new virtual machine.



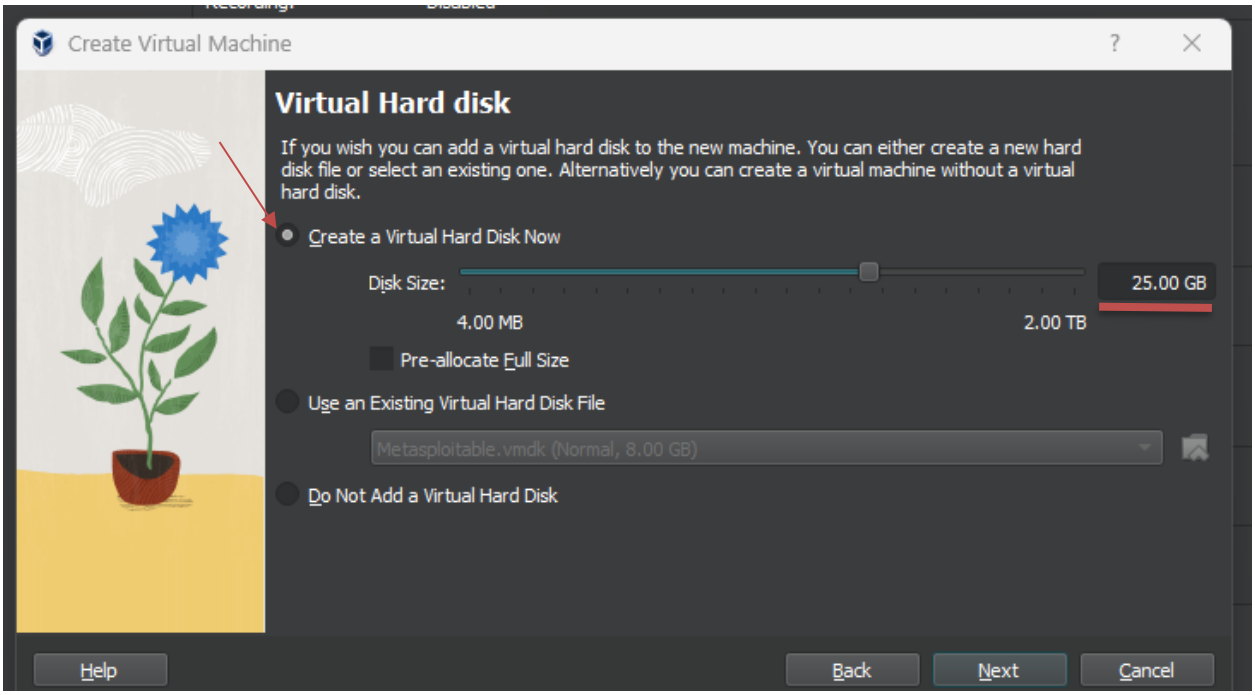
- Type the Virtual Machine name as “Ubuntu-24.04”.
- Select the ISO image of Ubuntu browsing your downloaded file of Ubuntu.
- When you selects ISO file, then automatically fill other fields also by themselves.
- To install itself without our intervention , you can keep it as it without tick on the check box which is, “Skip Unintended Installation”.
- Then click on the “NEXT” button.



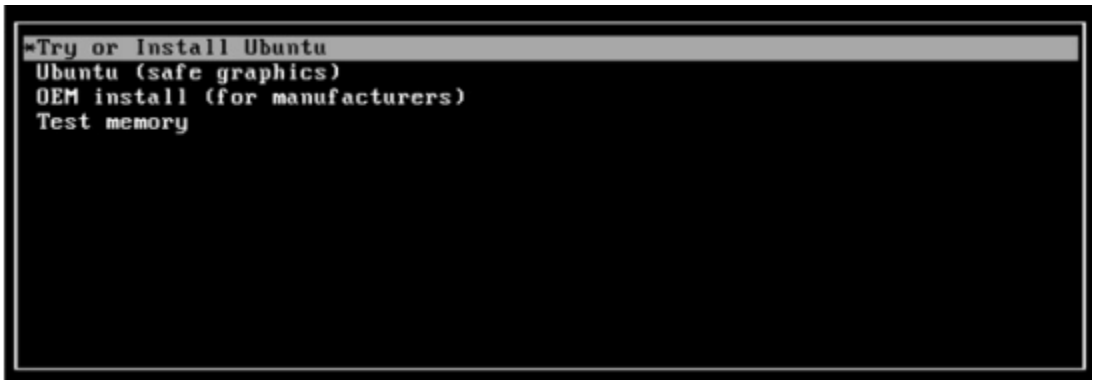
- Then you can see the “Unattended Guest OS install Setup” that is asking to setup our host information like username and password and all.
- If you wish to install guest additions after OS, then put a tick on “Guest Additions”. These enhancements allow you to copy and paste data between the host and virtual machines, adjust the virtual OS's resolution to match the host OS's window size, and all.
- Not only that we have to use the virtual box software to set up our Linux environment manually.
- Furthermore, we will be responsible for installing any guest additional features after the OS is installed. You can go with the manual installation method. In case that we did not check that box, the following window will open and ask us to supply the resources for our virtual machine in order for it to function. Allocating memory comes first, then allocating CPU cores. In this part, we have to be careful to only allocate what we truly need. Since providing the virtual machine with too many resources will cause the host machine to slow down



After click on the “Next” you can create a new hard disk file by adjusting the Disk size as you want.



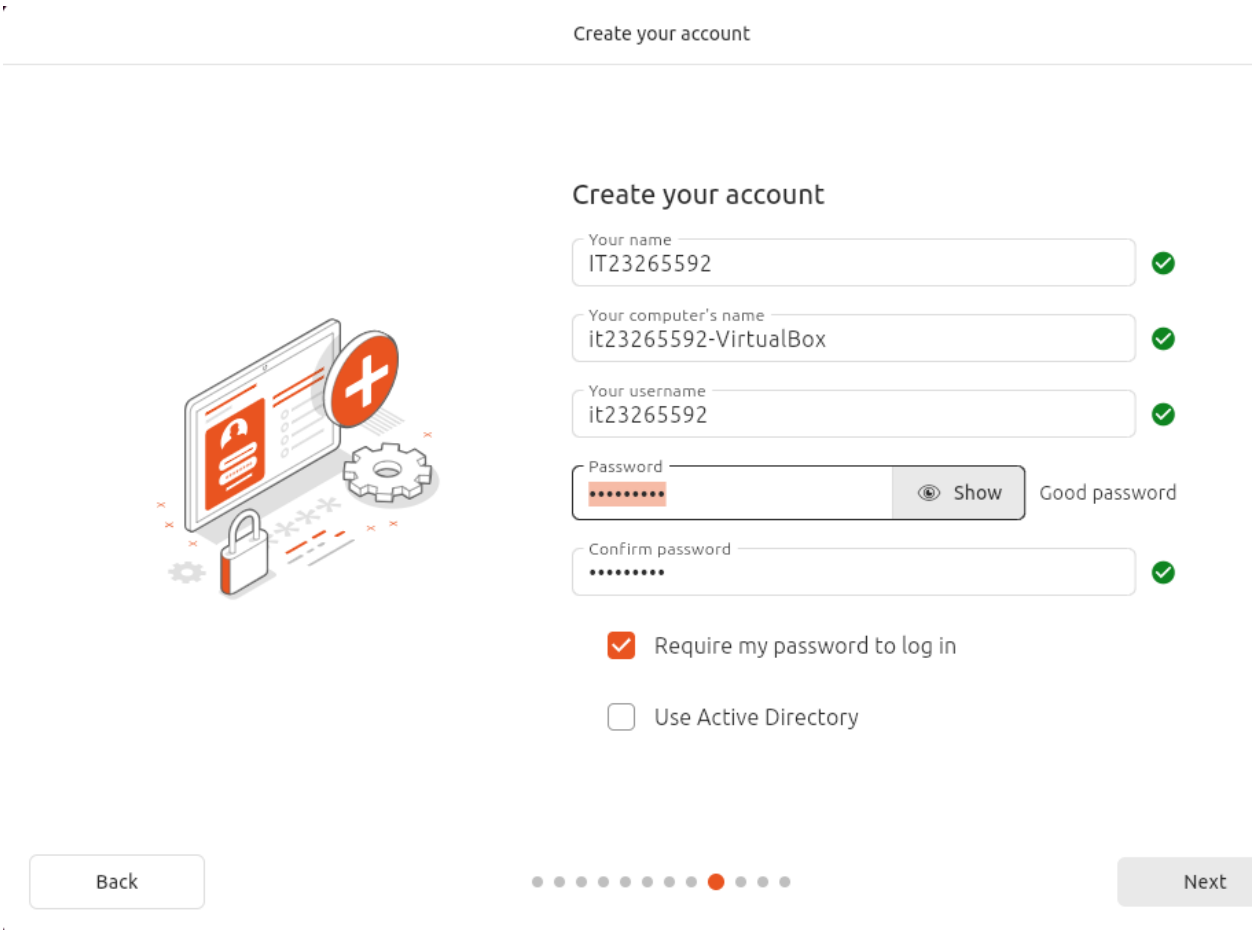
- After the click on the “Next” , you can see the summery of the given settings by you.
- After the double checking them , you can click on “Finish” .
- Then you can see the following window and choose “try or Install Ubuntu”.



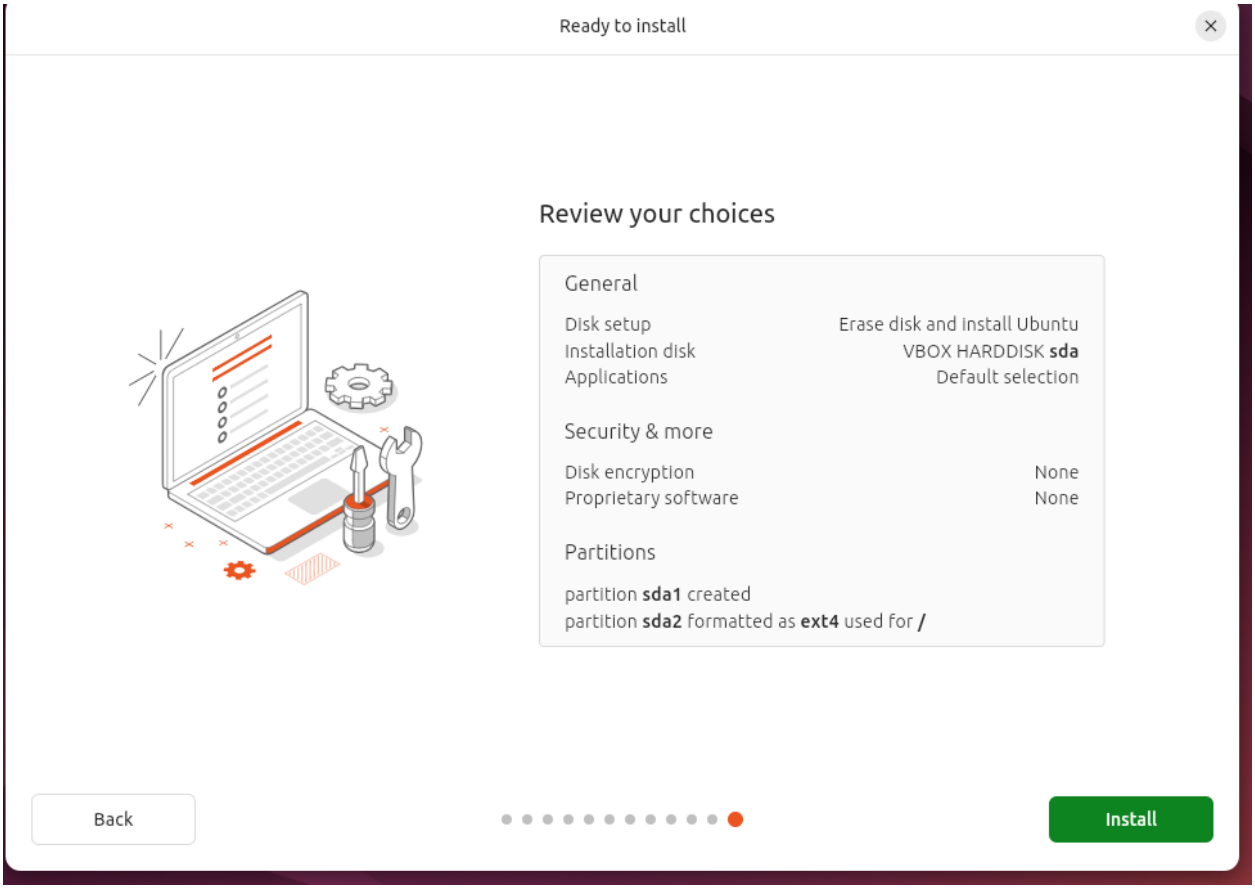
Then you are asked for some questions which are related to installation process as following :

- Choose your language as English
- Select keyboard layout as “English (US)”
- Select “install Ubuntu” as “you want to do with Ubuntu”
- Then select “Interactive installation” to get guide step by step through the installation.
- Then select “Default selection” for “apps would you like to install start with”.
- By “Default selection” is provids just the essentials, web browser and basic utilities.
- Don’t tick any following checkbox if you don’t need these proprietary software.
- Then select “Erase disk and install Ubuntu” as method to install Ubuntu.

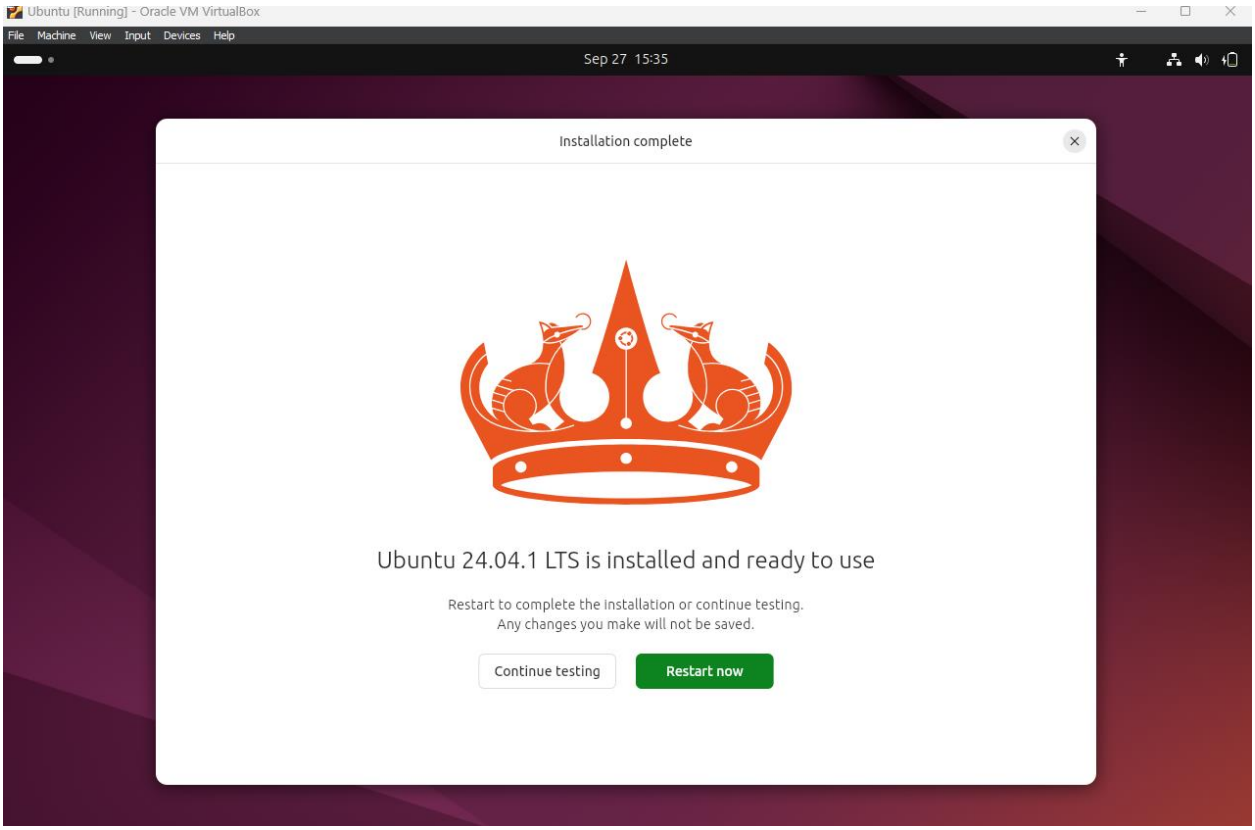
- Then you can create you account by giving your information like name and setting passwords as you wish.
- Don't forget to tick on " Require my password to log in" to provide security to your Linux environment.



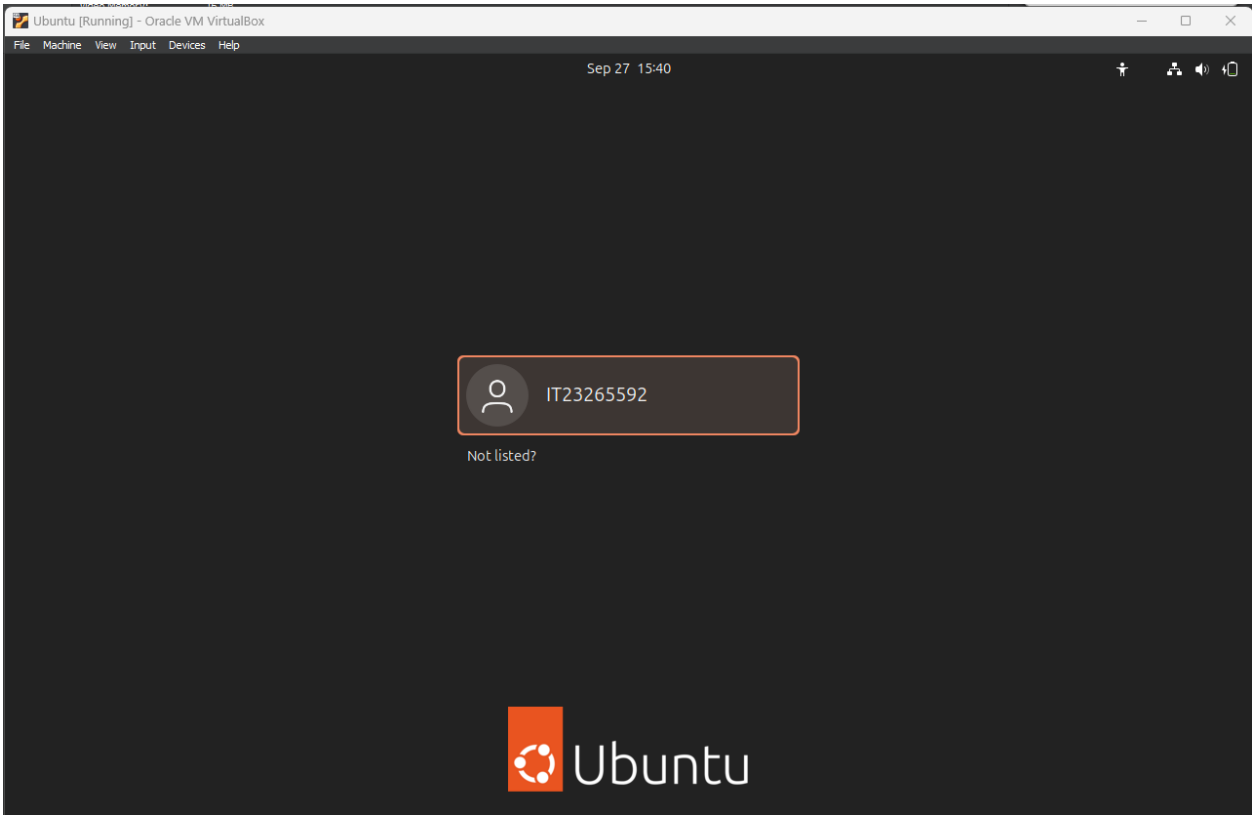
- Then you choose your location as “Colombo” and your time zone as “Asia/Colombo”.
- Then you can see the summery of your choices as following .
- After that you can click on “Install” to install Ubuntu



- After the installation you have to restart the VM



- Then you can see your created account and log in with is by entering your password



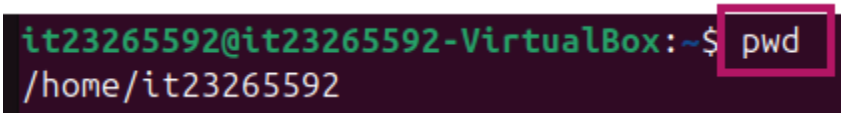
Command Line Introduction

- You have to log in to your virtual machine by using your credentials.
- Then open the terminal and try commands that you know.

Basic navigation commands

1. **pwd** : (Print Working Directory)

   Display the current working directory (folder) that you are in now.



2. **ls** : (List)

   List files and directories in your current directory

2.1. **ls -l**

Provide detailed list which includes permissions, owner, size and all.

```
it23265592@it23265592-VirtualBox:~$ ls -l
total 36
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Desktop
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Documents
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Downloads
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Music
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Pictures
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Public
drwx------ 4 it23265592 it23265592 4096 Sep 27 18:58 snap
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Templates
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Videos
```

2.2. **ls -a**

Show hidden files which are starting with dot (.)

```
it23265592@it23265592-VirtualBox:~$ ls -a
.               .bash_logout   .config     Downloads  Music     Public  Templates
..              .bashrc        Desktop     .gnupg     Pictures  snap    Videos
.bash_history   .cache         Documents   .local     .profile  .ssh
```

2.3. **ls -al**

list all (including hidden files) files and directories with size, permission, date.

```
it23265592@it23265592-VirtualBox:~/Desktop$ ls -al
total 20
drwxr-xr-x  5 it23265592 it23265592 4096 Sep 29 10:08 .
drwxr-x--- 16 it23265592 it23265592 4096 Sep 27 19:41 ..
drwxrwxr-x  2 it23265592 it23265592 4096 Sep 27 23:10 DMSS
-rw-rw-r--  1 it23265592 it23265592    0 Sep 29 10:08 lab1.c
drwxrwxr-x  2 it23265592 it23265592 4096 Sep 27 23:25 SNP
drwxrwxr-x  4 it23265592 it23265592 4096 Sep 27 23:06 SOS
```

2.4. **ls -r**

list all content even from the subdirectories

```
it23265592@it23265592-VirtualBox:~/Desktop$ ls -r
SOS  SNP  lab1.c  DMSS
```

2.5. **ls -al -lh**

list all files and directories with **readable size format**, permission, date

```
it23265592@it23265592-VirtualBox:~/Desktop$ ls -al -lh
total 20K
drwxr-xr-x  5 it23265592 it23265592 4.0K Sep 29 10:08 .
drwxr-x--- 16 it23265592 it23265592 4.0K Sep 27 19:41 ..
drwxrwxr-x  2 it23265592 it23265592 4.0K Sep 27 23:10 DMSS
-rw-rw-r--  1 it23265592 it23265592    0 Sep 29 10:08 lab1.c
drwxrwxr-x  2 it23265592 it23265592 4.0K Sep 27 23:25 SNP
drwxrwxr-x  4 it23265592 it23265592 4.0K Sep 27 23:06 SOS
```

3. **cd** : Change Directory

3.1. You can go to directory that you want by giving "cd <directory name or location>"

```
it23265592@it23265592-VirtualBox:~$ cd Desktop
it23265592@it23265592-VirtualBox:~/Desktop$
```

Then you can see , you are in the "/Desktop" directory.

3.2. **cd ..** : To move up one directory

```
it23265592@it23265592-VirtualBox:~$ cd Desktop
it23265592@it23265592-VirtualBox:~/Desktop$ cd ..
it23265592@it23265592-VirtualBox:~$
```

3.3. **cd ~** : go to root directory.

```
it23265592@it23265592-VirtualBox:~/Desktop$ cd ~
it23265592@it23265592-VirtualBox:~$
```

```
it23265592@it23265592-VirtualBox:~$ cd Desktop/SNP
it23265592@it23265592-VirtualBox:~/Desktop/SNP$
```

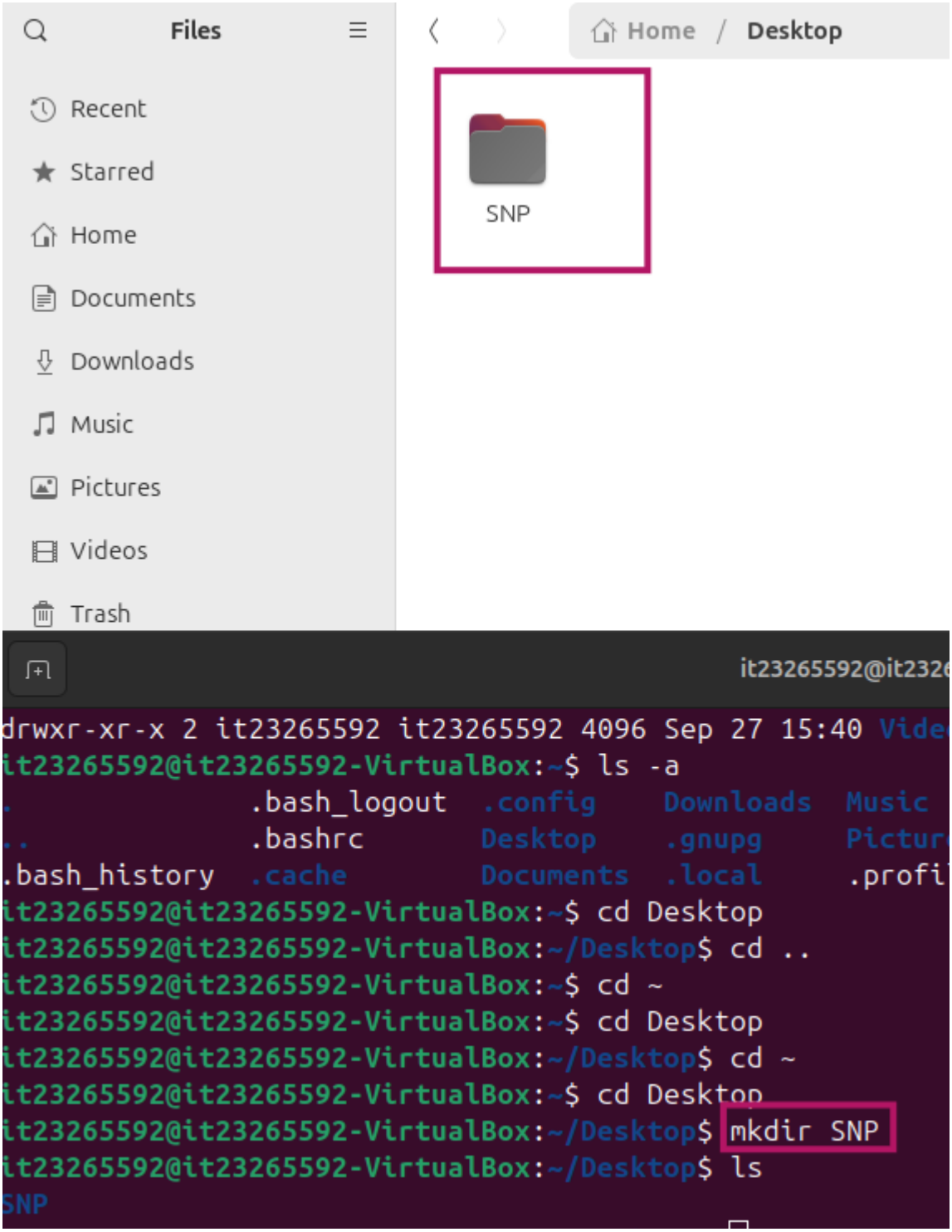
3.4. **cd -** : switch to previous directory

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cd -
/home/it23265592/Desktop
```

4. **mkdir** : Make Directory

   mkdir <direcotry name>

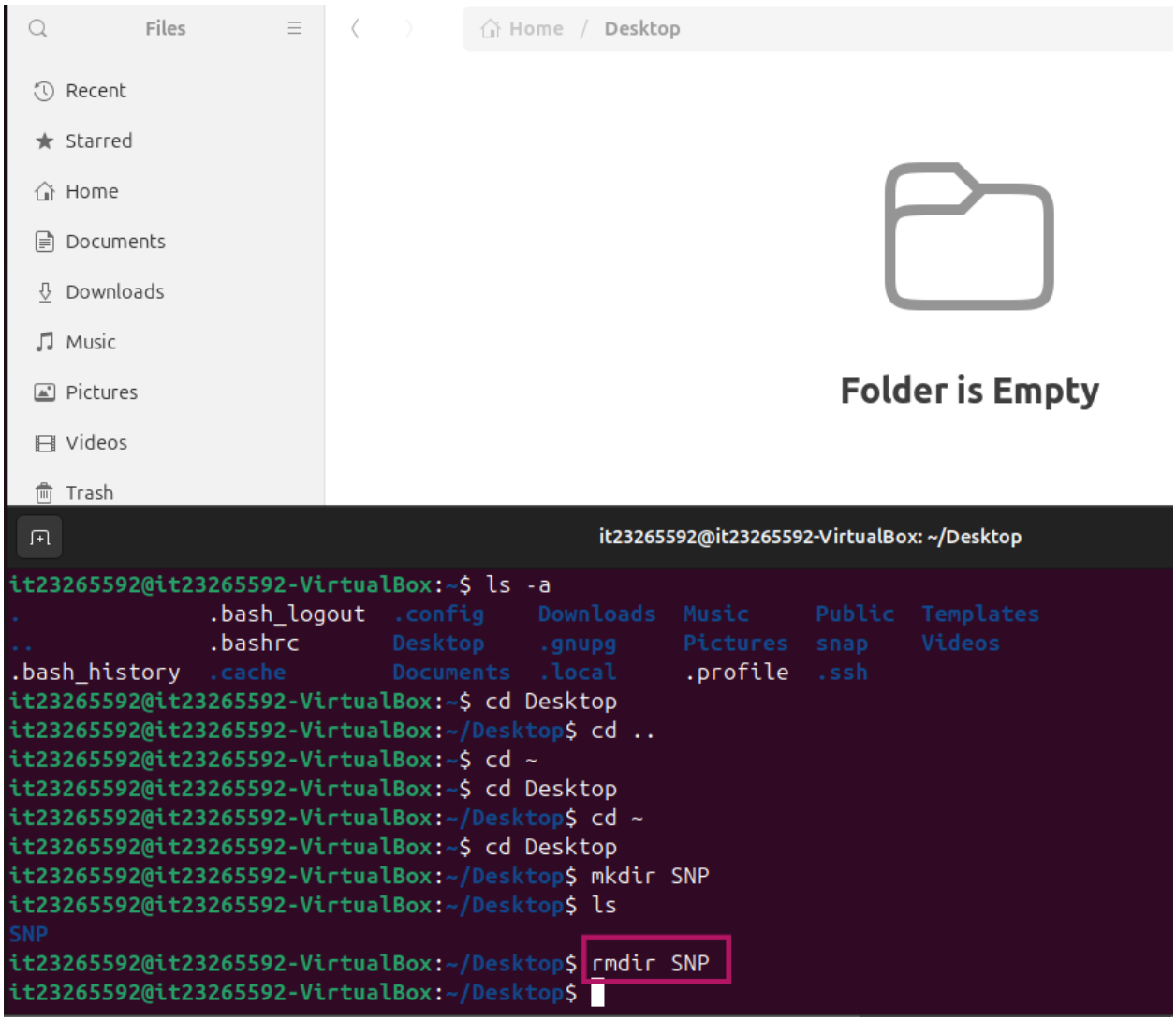
   Create a new directory in your current directory


Files
Recent
Starred
Home
Documents
Downloads
Music
Pictures
Videos
Trash
Home / Desktop
SNP
it23265592@it232
drwxr-xr-x 2 it23265592 it23265592 4096 Sep 27 15:40 Vide
it23265592@it23265592-VirtualBox:~$ ls -a
. .bash_logout .config Downloads Music
.. .bashrc Desktop .gnupg Pictur
.bash_history .cache Documents .local .profi
it23265592@it23265592-VirtualBox:~$ cd Desktop
it23265592@it23265592-VirtualBox:~/Desktop$ cd ..
it23265592@it23265592-VirtualBox:~$ cd ~
it23265592@it23265592-VirtualBox:~$ cd Desktop
it23265592@it23265592-VirtualBox:~/Desktop$ cd ~
it23265592@it23265592-VirtualBox:~$ cd Desktop
it23265592@it23265592-VirtualBox:~/Desktop$ mkdir SNP
it23265592@it23265592-VirtualBox:~/Desktop$ ls
SNP

5. **rmdir** : Remove directory

   rmdir <directory name>

Delete directory that you mentioned.



6. **touch** : create new empty file

   touch <filename>

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ touch lab1.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab1.c
```

7. **rm** : remove file that you mentioned

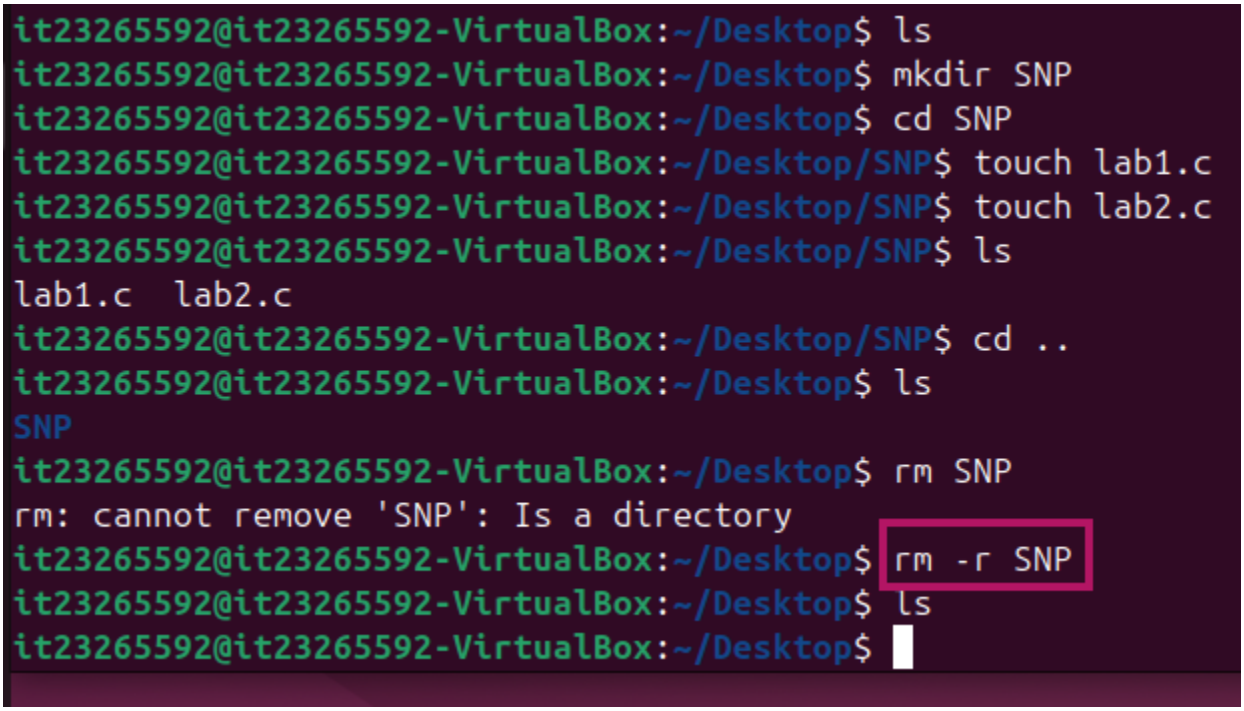
   rm <filename>

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab1.c  lab2.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ rm lab2.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab1.c
```

8. **rm -r** : Remove Recursive

   Removes a directory and its contents recursively.

   Rm -r <directoryName>



9. **cp** : Copy

copy files or directories

cp <old_filename> <new_filename>

```
it23265592@it23265592-VirtualBox:~/Desktop$ cd SNP
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ touch lab1.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ touch lab2.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cp lab1.c lab3.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab1.c  lab2.c  lab3.c
```

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab1.c  lab2.c  lab3.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cp lab1.c lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cp lab1.c lab
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cp lab1 lab6.c
cp: cannot stat 'lab1': No such file or directory
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab  lab1.c  lab2.c  lab3.c  lab4.py
```

9.1. **cp -r** : copy directory to another directory

cp -r <old_directory_name> <new_directory_name>

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
lab  lab1.c  lab2.c  lab3.c  lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cd ..
it23265592@it23265592-VirtualBox:~/Desktop$ cp -r SNP SOS
it23265592@it23265592-VirtualBox:~/Desktop$ ls
SNP  SOS
it23265592@it23265592-VirtualBox:~/Desktop$ cd SOS
it23265592@it23265592-VirtualBox:~/Desktop/SOS$ ls
lab  lab1.c  lab2.c  lab3.c  lab4.py
```

10. **mv** : Move or Rename

    mv <oldfile> <destination>

    Move or rename files or directories.

```
it23265592@it23265592-VirtualBox:~/Desktop/SOS$ ls
lab  lab1.c  lab2.c  lab3.c  lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SOS$ mv lab1.c lab8.c
it23265592@it23265592-VirtualBox:~/Desktop/SOS$ ls
lab  lab2.c  lab3.c  lab4.py  lab8.c
it23265592@it23265592-VirtualBox:~/Desktop/SOS$
```

11. **nano** : Nano editor

    nano <filename>

    if there is a file you can edit it by using nano edit or if there is a not such file, then it creates a

    new file using given name by you

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ nano lab2.c
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls
DMSS  lab  lab2.c  lab3.c  lab4.py
```

Firefox Web Browser

it23265592@it23265592-VirtualBox: ~/Desktop/SNP

```
 GNU nano 7.2                                lab2.c *
#include<stdio.h>
int main()
{
        printf("Hello  world ! ");
        return 0;
}
```

12. **cat** : Concatenate

cat <filename>

Displays the content of the file that you have mentioned.

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ cat lab2.c
#include<stdio.h>
int main()
{
        printf("Hello  world ! ");
        return 0;
}
it23265592@it23265592-VirtualBox:~/Desktop/SNP$
```

12.1. if you want , you can give location with “cat ” command

```
it23265592@it23265592-VirtualBox:~$ cat Desktop/SNP/lab2.c
#include<stdio.h>
int main()
{
        printf("Hello  world ! ");
        return 0;
}
```

13. **file** : determine the file type
file <filename>

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ file lab3.c
lab3.c: empty
```

13.1. You must mention the file extension also when check the file type

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ file lab2
lab2: cannot open `lab2' (No such file or directory)
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ file lab2.c
lab2.c: C source, ASCII text
```

13.2. if you want to check all files in the directory, you can type as following :
file ./*

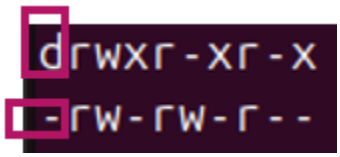
```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ file ./*
./DMSS:    empty
./lab:     empty
./lab2.c:  C source, ASCII text
./lab3.c:  empty
./lab4.py: empty
```

14. **chmod** : Change Mode

chmod <changes> <filename>

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ chmod u+x DMSS
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rwxrw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$
```

- There are some information related to the file and directories as following :



  - ✓ **d** : directories
  - ✓ **-** : files

- There are 3 access types :
  - ✓ User : **u**
  - ✓ Group : g
  - ✓ Other : **o**

- Permissions types and their meaning:
  - ✓ **r** : read
  - ✓ **x** : execute
  - ✓ **w** : write

- You can maximize permissions (+) and you can minimize permissions (-) too.

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rwxrw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ chmod u+x lab
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ chmod u-x DMSS
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rwxrw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
```

- You can use values to change file permissions

change file permissions	read	write	execute	value
Cannot read , write, execute	0	0	0	0
Execute only	0	0	1	1
Write only	0	1	0	2
Write + execute	0	1	1	3
Read only	1	0	0	4
Read + execute	1	0	1	5
Read + write	1	1	0	6
Read + write + execute	1	1	1	7

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rwxrw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ chmod +750 lab4.py
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ ls -al
total 12
drwxrwxr-x 2 it23265592 it23265592 4096 Sep 27 23:25 .
drwxr-xr-x 5 it23265592 it23265592 4096 Sep 29 10:08 ..
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 DMSS
-rwxrw-r-- 1 it23265592 it23265592    0 Sep 27 22:50 lab
-rw-rw-r-- 1 it23265592 it23265592   72 Sep 27 23:25 lab2.c
-rw-rw-r-- 1 it23265592 it23265592    0 Sep 27 22:49 lab3.c
-rwxrwxr-- 1 it23265592 it23265592    0 Sep 27 22:50 lab4.py
```

- The meaning of above example is lab4.py file can :
  - ✓ User can read + write + execute
  - ✓ Group can read + write + execute because it had write permission since it was created
  - ✓ Other can read only

15. **find** : search for files or directory in the given path or requirements
    find <path>

```
it23265592@it23265592-VirtualBox:~$ find Desktop/SNP
Desktop/SNP
Desktop/SNP/lab3.c
Desktop/SNP/lab4.py
Desktop/SNP/lab2.c
Desktop/SNP/lab
Desktop/SNP/DMSS
```

15.1. find by giving user , group and other details

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ find -user it23265592
.
./lab3.c
./lab4.py
./lab2.c
./lab
./DMSS
```

- This shows by giving user as a parameter within find command

Note :

- Use " find " command to get the file that have these properties

✓ find : to search for files and directories within a directory hierarchy

✓ -user : to give user as a parameter

✓ -group : to give group as a parameter

✓ -size : size of the file

✓ c : gives size for bytes

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ find -user it23265592 -size 72c
./lab2.c
```

- This example shows a file which is "lab2.c" with the user of it23265592 and the size of 72bits

16. **whoami** : Get the active username

```
it23265592@it23265592-VirtualBox:~$ whoami
it23265592
```

17. **who** : give information about currently logged in users

```
it23265592@it23265592-VirtualBox:~$ who
it23265592 seat0        2024-09-29 09:59 (login screen)
it23265592 tty2         2024-09-29 09:59 (tty2)
```

18. **passwd** : you can change password of current logged user account

    1. you have to give your current password
    2. next you have to enter new password
    3. then you have to re enter your next password
    4. then it will be changed

```
it23265592@it23265592-VirtualBox:~$ passwd
Changing password for it23265592.
Current password:
New password:
Retype new password:
passwd: password updated successfully
```

19. **id** : give details about user and group details with chosen name to user .

    if user hasn't specific name it will be prints details of current logged user

```
it23265592@it23265592-VirtualBox:~$ id
uid=1000(it23265592) gid=1000(it23265592) groups=1000(it23265592),4(adm),24(cdro
m),27(sudo),30(dip),46(plugdev),100(users),114(lpadmin)
```

20. **free** : give information about memory
    such as :
    - Memory usage details of main memory and swap memory.
    - Give information about total memory , used memory, shared memory, cache memory , free memory , buffer memory and available memory and the base will be kibibytes that is 1024 bytes in number.

```
it23265592@it23265592-VirtualBox:~$ free
               total        used        free      shared  buff/cache   available
Mem:         4010092     1001460     2452704       33188      809996     3008632
Swap:        4009980           0     4009980
```

21. **grep** : search word in content (grep testword testfile)

```
it23265592@it23265592-VirtualBox:~/Desktop/SNP$ grep "printf" lab2.c
        printf("Hello  world ! ");
```

22. **uname** : get basic information about OS

```
it23265592@it23265592-VirtualBox:~$ uname
Linux
```

DHCP installation steps

1. you have to check whether you have updated system and you have to update up to date

   Command **: sudo apt update** : update the system.

```
it23265592@it23265592-VirtualBox:~$ sudo apt update
[sudo] password for it23265592:
Hit:1 http://lk.archive.ubuntu.com/ubuntu noble InRelease
```

You have to give the password for your account to continue this process

2. Then install DHCP server
   In the terminal : **sudo apt install isc-dhcp-server**
   Then give the password of your account to continue your process

```
it23265592@it23265592-VirtualBox:~$ sudo apt install isc-dhcp-server
[sudo] password for it23265592:
Reading package lists... Done
```

- After the completion of installing , show the mistake and fix them.

3. You can get information about it by using "**dhcpd**"

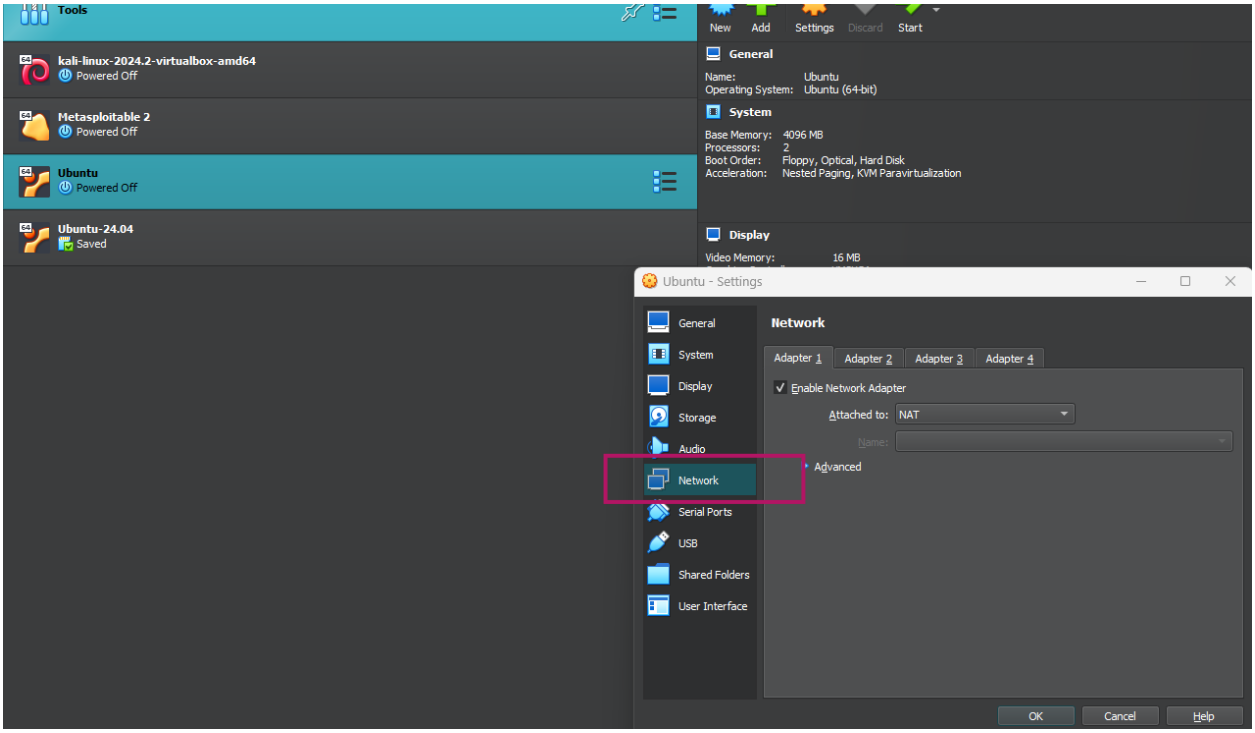
```
it23265592@it23265592-VirtualBox:~$ dhcpd
Internet Systems Consortium DHCP Server 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
unable to create icmp socket: Operation not permitted
Config file: /etc/dhcp/dhcpd.conf
Database file: /var/lib/dhcp/dhcpd.leases
PID file: /var/run/dhcpd.pid
Can't open /var/lib/dhcp/dhcpd.leases for append.

If you think you have received this message due to a bug rather
than a configuration issue please read the section on submitting
bugs on either our web page at www.isc.org or in the README file
before submitting a bug.  These pages explain the proper
process and the information we find helpful for debugging.

exiting.
```

- First we have to make sure that the file in **"/var/lib/dhcp/dhcp.leases"** file is available because it is the file which is causing the mistake .
- If it is there : change permission to give permission it needs.
- If not is there : create new one and give permission it needs.

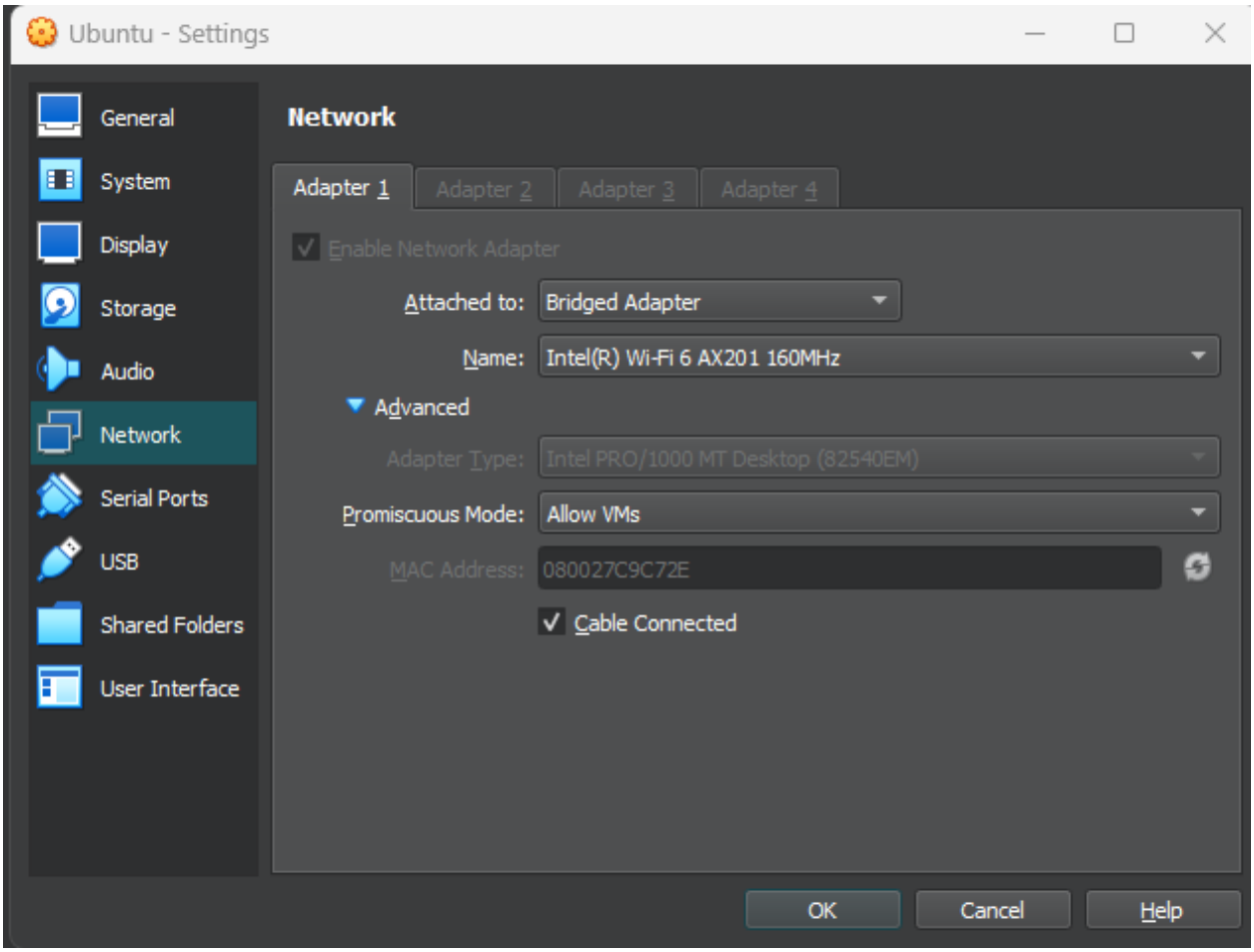
Before do the changes we have to do settings to our internet network then the DHCP server and client can talk to each other.

By default this private network can only talk with VMs.

1. Shutdown the Ubuntu virtual machine.
2. Go to VM network in the settings.



3. Then do the changes as following :



Then your VM can act like a separate device on your network and get its own IP from your wi-fi router and communicate with other devices on your network.

4. After that give “ok” and open Ubuntu VM again.

5. Then configure DHCP server

5.1. Install the net tools by giving "**sudo apt install net-tools**"

```
it23265592@it23265592-VirtualBox:~$ sudo apt install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 204 kB of archives.
After this operation, 811 kB of additional disk space will be used.
Get:1 http://lk.archive.ubuntu.com/ubuntu noble/main amd64 net-tools amd64 2.10-0.1ubuntu
Fetched 204 kB in 2s (99.6 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 148097 files and directories currently installed.)
Preparing to unpack .../net-tools_2.10-0.1ubuntu4_amd64.deb ...
Unpacking net-tools (2.10-0.1ubuntu4) ...
Setting up net-tools (2.10-0.1ubuntu4) ...
Processing triggers for man-db (2.12.0-4build2) ...
```

5.2. Then use "ifconfig" to get information about network .

```
it23265592@it23265592-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.15  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 2402:d000:812c:8e1:e1d3:f57e:2575:3e02  prefixlen 64  scopeid 0x0<global>
        inet6 2402:d000:812c:8e1:a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:c9:c7:2e  txqueuelen 1000  (Ethernet)
        RX packets 30994  bytes 37200920 (37.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 11352  bytes 1511677 (1.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 681  bytes 92748 (92.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 681  bytes 92748 (92.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

5.3. Open the DHCP configuration file with nano editor

```
it23265592@it23265592-VirtualBox:~$ sudo nano /etc/dhcp/dhcp.conf
[sudo] password for it23265592:
```

By giving your password for it23265592 , you can continue this process.

5.4. Then modify as following lines

```
it23265592@it23265592-VirtualBox: ~
  GNU nano 7.2                                  /etc/dhcp/dhcp.conf *
# DHCP Server Configuration

# Default lease time in seconds
default-lease-time 900;

# Maximum lease time in seconds
max-lease-time 3600;

# Specify that this server is authoritative
authoritative;

# Subnet configuration
subnet 192.168.1.15 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.200;      # Range of IP addresses to be assigned
    option routers 192.168.1.254;             # Default gateway
    option subnet-mask 255.255.255.0;      # Subnet mask
    option domain-name-servers 8.8.8.8, 8.8.4.4;  # DNS servers
    option domain-name "it23265592.com";    # Domain name
    option broadcast-address 192.168.1.255; # Broadcast address (optional)
}
```

```
default-lease-time 900;
max-lease-time 3600;
authoritative;

subnet 192.168.1.15 netmask 255.255.255.0 {
   range 192.168.1.100 192.168.1.200;       # IP address range
   option routers 192.168.1.254;            # Gateway/router IP
   option subnet-mask 255.255.255.0;        # Subnet mask
   option domain-name-servers 8.8.8.8, 8.8.4.4; # DNS servers
   option domain-name "it23265592.com";     # Domain name
}
```

- This configures the DHCP server to assign IP addresses in the range of 192.168.1.100 to 192.168.1.200

Verify interfaceesv4 settings by checking “/etc/default/isc-dhcp-server” file and make sure you have correctly mentioned your network interface.

In this case it is “enp0s3”.

This helps to DHCP server can know which network interface to use.

View it using "**sudo nano /etc/default/isc-dhcp-server** "

```
it23265592@it23265592-VirtualBox:~$ sudo nano /etc/default/isc-dhcp-server
```

```
it23265592@it23265592-VirtualBox: ~
  GNU nano 7.2                                /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
#       Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="enp0s3"
INTERFACESv6=""
```

Then bring the interface up manually using "**sudo ip link set enp0s3 up**" command.

```
it23265592@it23265592-VirtualBox:~$ sudo ip link set enp0s3 up
[sudo] password for it23265592:
```

Then verify after whether is interface up by giving "**ip addr show enp0s3**"

```
it23265592@it23265592-VirtualBox:~$ ip addr show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:c9:c7:2e brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.15/24 brd 192.168.1.255 scope global dynamic noprefixroute en
p0s3
       valid_lft 257393sec preferred_lft 257393sec
    inet6 2402:d000:812c:8e1:e1d3:f57e:2575:3e02/64 scope global temporary dynam
ic
       valid_lft 258917sec preferred_lft 84520sec
    inet6 2402:d000:812c:8e1:a00:27ff:fec9:c72e/64 scope global dynamic mngtmpad
dr
       valid_lft 258917sec preferred_lft 172517sec
    inet6 fe80::a00:27ff:fec9:c72e/64 scope link
       valid_lft forever preferred_lft forever
```

Then you can see that the state is "up".

5.5. Then restart DHCP server to apply these modifications.
Give command as : "**sudo systemctl restart isc-dhcp-server**"

```
it23265592@it23265592-VirtualBox:~$ sudo systemctl restart isc-dhcp-server
```

5.6. Check the status to see whether the service started correctly.
Use command : "**sudo systemctl status isc-dhcp-server**"

```
it23265592@it23265592-VirtualBox:~$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP IPv4 server
     Loaded: loaded (/usr/lib/systemd/system/isc-dhcp-server.service; enabled; preset: enabled)
     Active: active (running) since Fri 2024-10-04 22:00:01 +0530; 11s ago
       Docs: man:dhcpd(8)
   Main PID: 3154 (dhcpd)
      Tasks: 1 (limit: 4615)
     Memory: 3.7M (peak: 4.1M)
        CPU: 12ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─3154 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dhcp/dhcpd.conf enp0s3

Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Database file: /var/lib/dhcp/dhcpd.leases
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: PID file: /run/dhcp-server/dhcpd.pid
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Wrote 5 leases to leases file.
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Listening on LPF/enp0s3/08:00:27:c9:c7:2e/192.168.1.0/24
Oct 04 22:00:01 it23265592-VirtualBox sh[3154]: Listening on LPF/enp0s3/08:00:27:c9:c7:2e/192.168.1.0/24
Oct 04 22:00:01 it23265592-VirtualBox sh[3154]: Sending on   LPF/enp0s3/08:00:27:c9:c7:2e/192.168.1.0/24
Oct 04 22:00:01 it23265592-VirtualBox sh[3154]: Sending on   Socket/fallback/fallback-net
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Sending on   LPF/enp0s3/08:00:27:c9:c7:2e/192.168.1.0/24
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Sending on   Socket/fallback/fallback-net
Oct 04 22:00:01 it23265592-VirtualBox dhcpd[3154]: Server starting service.
it23265592@it23265592-VirtualBox:~$
```

Then you can see the current state of operations .

```
it23265592@it23265592-VirtualBox:~$ dhcpd
Internet Systems Consortium DHCP Server 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
unable to create icmp socket: Operation not permitted
Config file: /etc/dhcp/dhcpd.conf
Database file: /var/lib/dhcp/dhcpd.leases
PID file: /var/run/dhcpd.pid
Can't open /var/lib/dhcp/dhcpd.leases for append.

If you think you have received this message due to a bug rather
than a configuration issue please read the section on submitting
bugs on either our web page at www.isc.org or in the README file
before submitting a bug.  These pages explain the proper
process and the information we find helpful for debugging.

exiting.
it23265592@it23265592-VirtualBox:~$
```

Small Network Environment

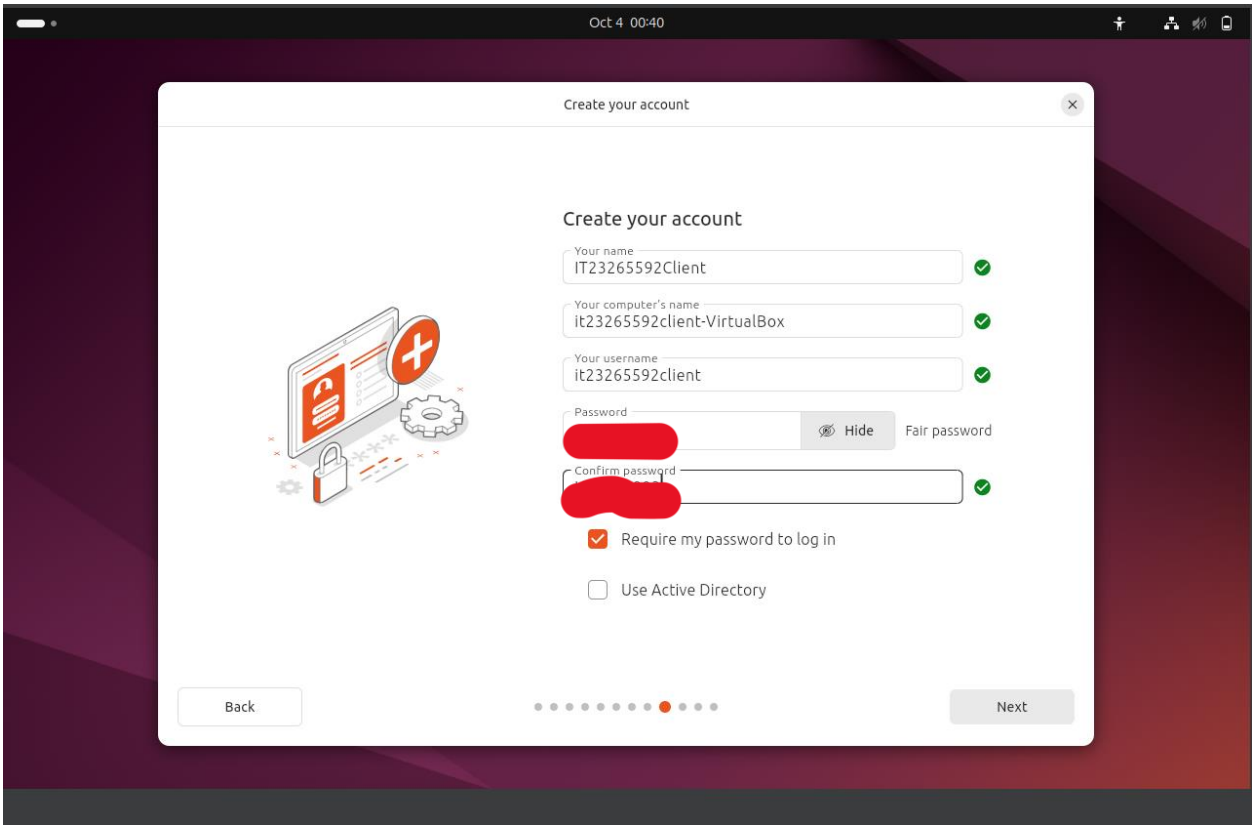
Simulate a small network environment with your virtual machine and configure it to obtain an IP address automatically from the DHCP server.

- After the configuration of server machine, you have to set the client machine.

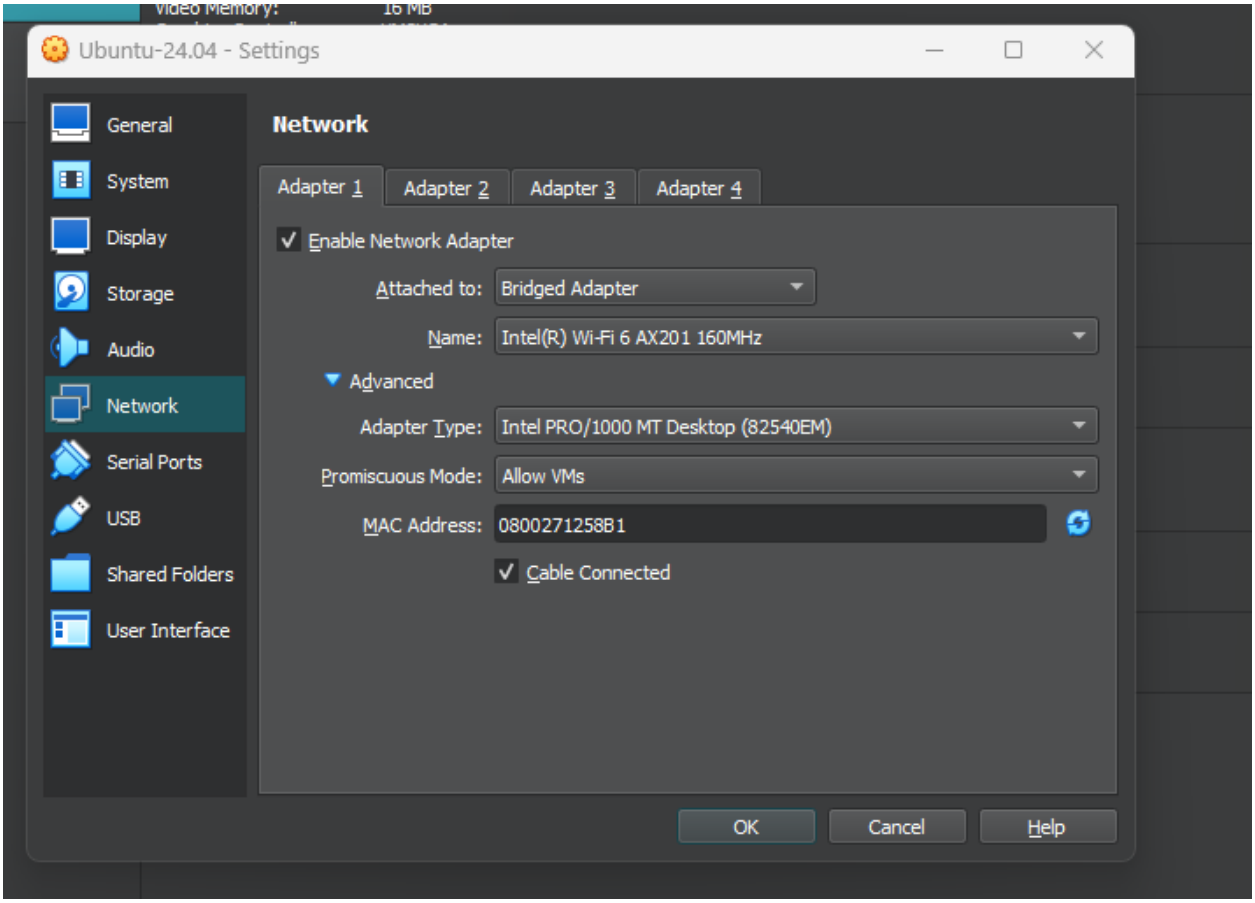
1. First you have to install another VM as client machine

- I installed another UBUNTU VM to set the client.

Username : IT23265592Client



Then change network tab settings under your settings in Virtual Box as “bridge adapter”



1. Check your network details
   Command : **ifconfig**

```
it23265592client@it23265592client-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.104  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 2402:d000:812c:66e0:4572:d26e:f54c:7dbe  prefixlen 64  scopeid 0x0<gl
        inet6 2402:d000:812c:66e0:a00:27ff:fe12:58b1  prefixlen 64  scopeid 0x0<glc
        inet6 fe80::a00:27ff:fe12:58b1  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:12:58:b1  txqueuelen 1000  (Ethernet)
        RX packets 40  bytes 6599 (6.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 112  bytes 16446 (16.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 141  bytes 13921 (13.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 141  bytes 13921 (13.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

2. Install DHCP client
   Command :
   sudo apt update
   sudo apt install isc-dhcp-client

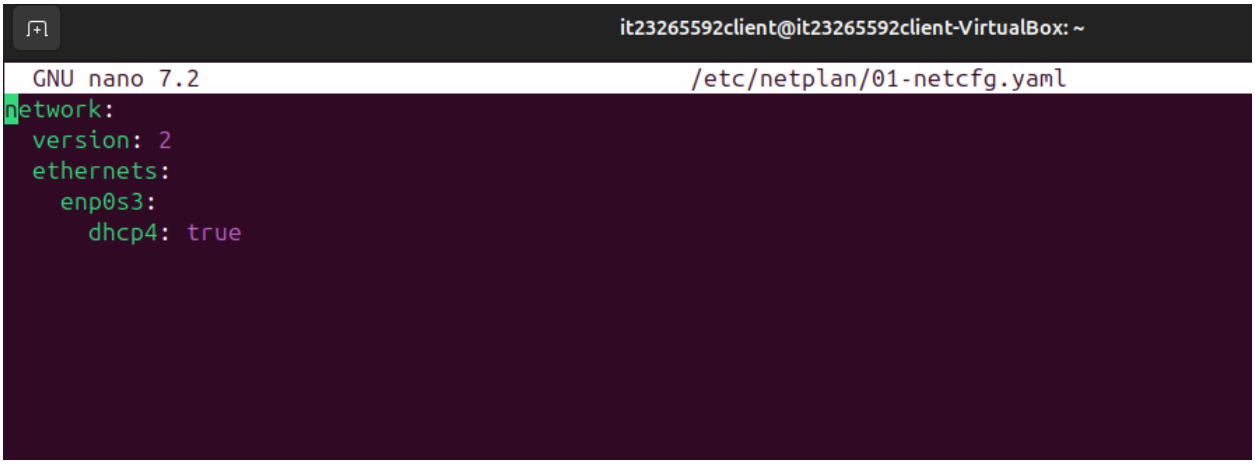
```
it23265592client@it23265592client-VirtualBox:~$ sudo apt update
sudo apt install isc-dhcp-client
[sudo] password for it23265592client:
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://lk.archive.ubuntu.com/ubuntu noble InRelease
Get:3 http://lk.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:4 http://lk.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:5 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [537 kB
Get:6 http://lk.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [38
Fetched 1,047 kB in 4s (299 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
32 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  isc-dhcp-common
Suggested packages:
  avahi-autoipd isc-dhcp-client-ddns
The following NEW packages will be installed:
  isc-dhcp-client isc-dhcp-common
0 upgraded, 2 newly installed, 0 to remove and 32 not upgraded.
Need to get 375 kB of archives.
After this operation, 1,011 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

3. Configure Network Interface to use DHCP
   Edit the netplan configuration file.
   Command : **sudo nano /etc/netplan/01-netcfg-.yaml**

```
it23265592client@it23265592client-VirtualBox:~$ sudo nano /etc/netplan/01-netcfg.yaml
```

4. Include following configuratiom

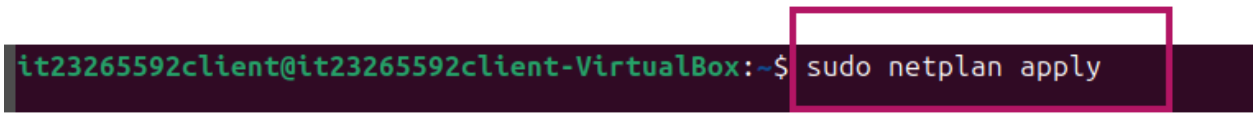
network:
version:2
ethernets:
enp0s3:
dhcp: true



5. Apply the network configuration

- Run the following command to apply the new settings

Command : **sudo netplay apply**



6. Check the assigned IP address
    - Verify that the client has received an IP address from the DHCP server.
      Command : **ip addr show enp0s3**

```
it23265592client@it23265592client-VirtualBox:~$ ip addr show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group def
    link/ether 08:00:27:12:58:b1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.104/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 860sec preferred_lft 860sec
    inet6 2402:d000:812c:66e0:6595:8bc7:5dd8:3230/64 scope global temporary dynamic
       valid_lft 259190sec preferred_lft 85798sec
    inet6 2402:d000:812c:66e0:a00:27ff:fe12:58b1/64 scope global dynamic mngtmpaddr
       valid_lft 259190sec preferred_lft 172790sec
    inet6 fe80::a00:27ff:fe12:58b1/64 scope link
       valid_lft forever preferred_lft forever
```

The IP address 192.168.1.104 has been assigned to the enp0s3 interface, which falls within the DHCP range you configured (192.168.1.100 to 192.168.1.200). The IP is marked as **dynamic**, indicating it was assigned by the DHCP server.

NTP installation

1. Update your package list
   Command : **sudo apt update**

   By giving your password of account you can continue this process.

```
it23265592@it23265592-VirtualBox:~$ sudo apt update
[sudo] password for it23265592:
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://lk.archive.ubuntu.com/ubuntu noble InRelease
Get:3 http://lk.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:4 http://lk.archive.ubuntu.com/ubuntu noble-backports InRelease
Get:5 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [535
 kB]
Get:6 http://lk.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages
[377 kB]
Fetched 1,038 kB in 4s (257 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
7 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

2. Then install NTP package

Command : **sudo apt install ntp**

```
it23265592@it23265592-VirtualBox:~$ sudo apt install ntp
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ntpsec python3-ntp
Suggested packages:
  certbot ntpsec-doc ntpsec-ntpviz
The following packages will be REMOVED:
  systemd-timesyncd
The following NEW packages will be installed:
  ntp ntpsec python3-ntp
0 upgraded, 3 newly installed, 1 to remove and 7 not upgraded.
Need to get 450 kB of archives.
After this operation, 1,102 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lk.archive.ubuntu.com/ubuntu noble/universe amd64 python3-ntp amd64
 1.2.2+dfsg1-4build2 [91.2 kB]
Get:2 http://lk.archive.ubuntu.com/ubuntu noble/universe amd64 ntpsec amd64 1.2.
2+dfsg1-4build2 [343 kB]
Get:3 http://lk.archive.ubuntu.com/ubuntu noble/universe amd64 ntp all 1:4.2.8p1
5+dfsg-2~1.2.2+dfsg1-4build2 [15.7 kB]
Fetched 450 kB in 3s (156 kB/s)
```

3. Then you have to configure NTP using configure file in **"etc/ntp.conf"**

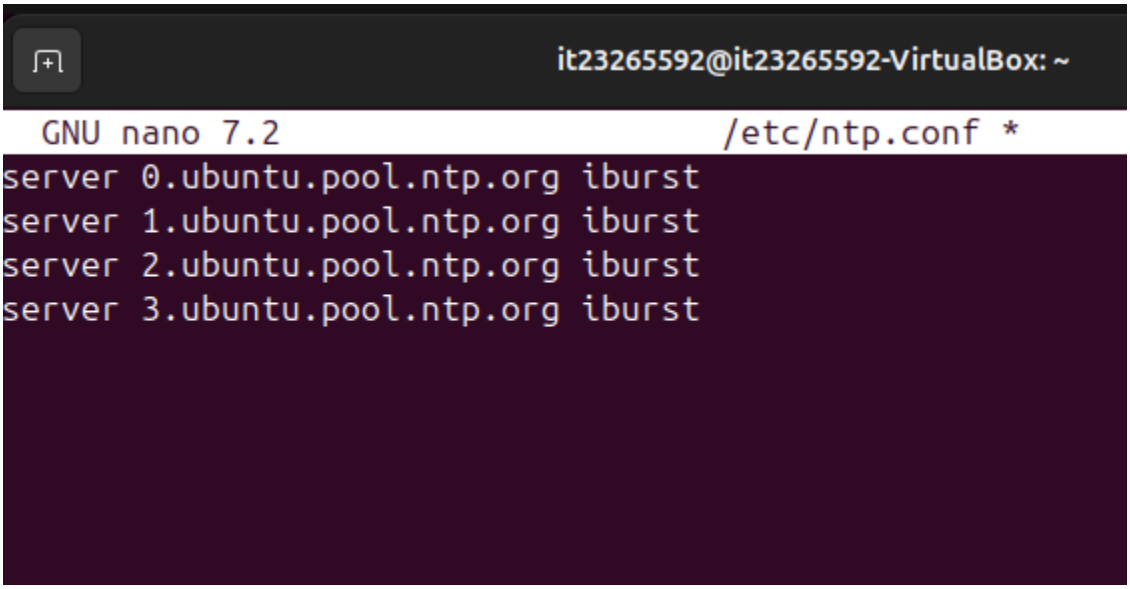
Open it by using "nano" editor.

Command : **sudo nano /etc/ntp.conf**

```
it23265592@it23265592-VirtualBox:~$ sudo nano /etc/ntp.conf
```

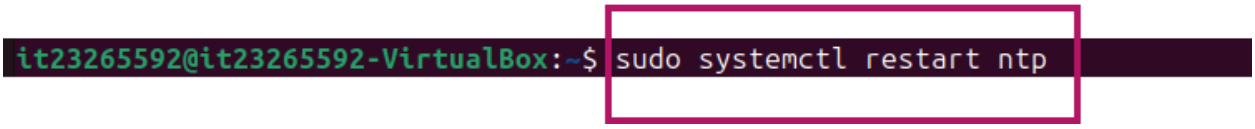
4. Then you can user default Ubuntu NTP servers

   This will show how to add servers :



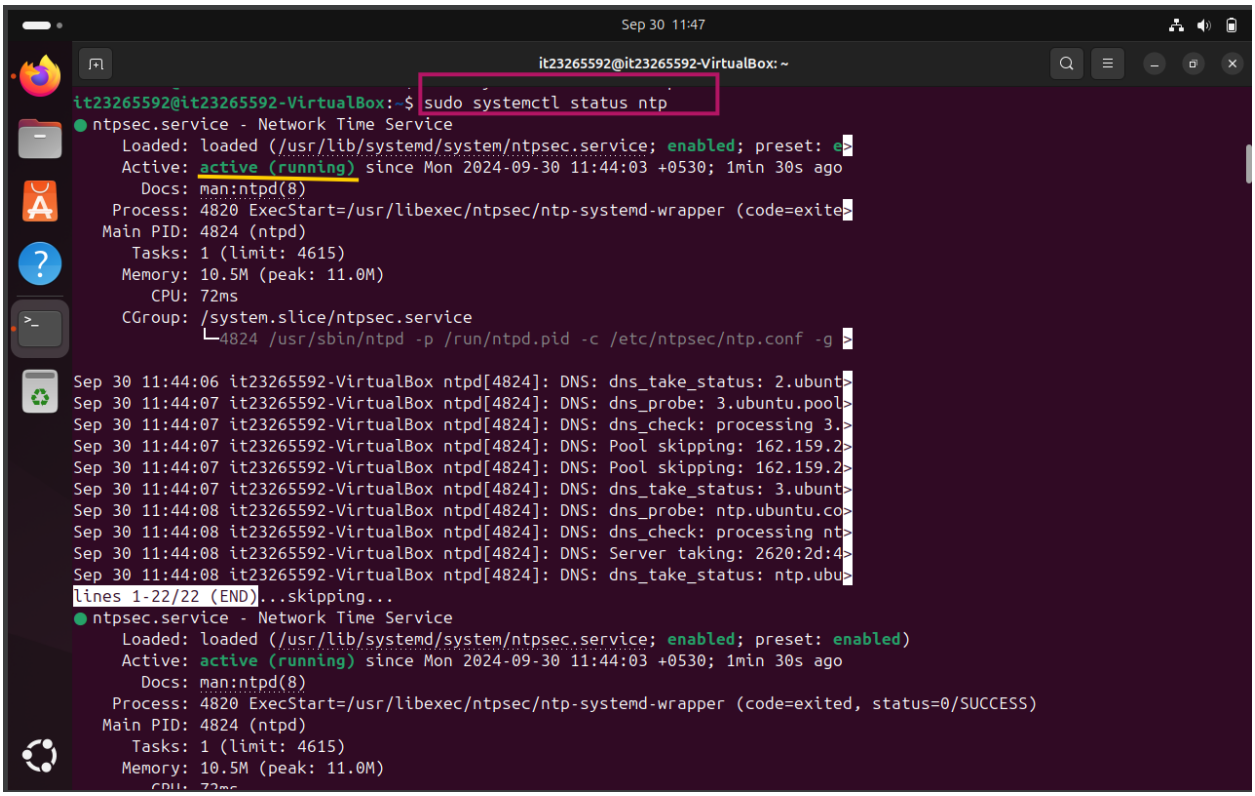
5. Then restart Ubuntu NTP service to add your modifications

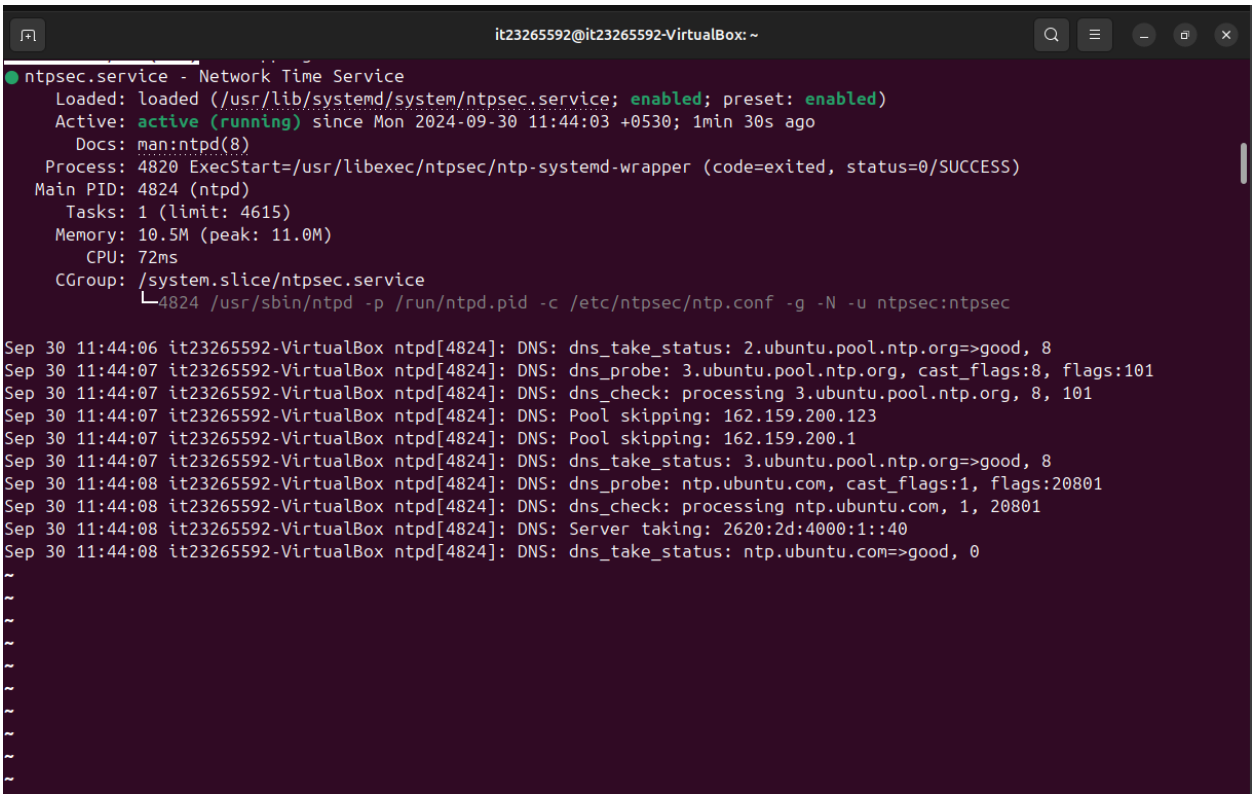
Command : **sudo systemctl restart ntp**



6. Then view the state whether it is running or not.

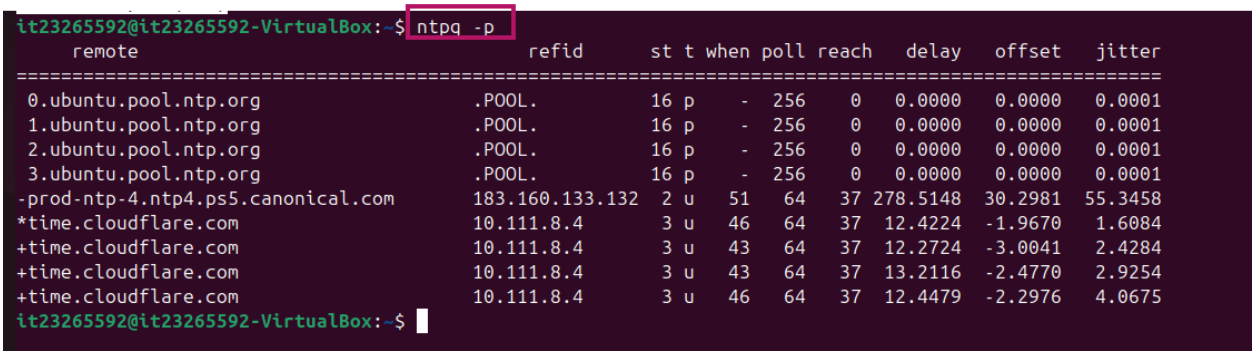
Command : **sudo systemctl status ntp**





7. Then verify the synchronization

Command : **ntpq -p**



This verify your server is synchronized with NTP servers .

This shows that the list of peers and their synchronization status.

DNS server installation

1. Update your system

   Command : **sudo apt update**

   You can continue this processs by giving your ubuntu account password

```
it23265592@it23265592-VirtualBox:~$ sudo apt upgrade
[sudo] password for it23265592:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following upgrades have been deferred due to phasing:
  python3-distupgrade ubuntu-release-upgrader-core
  ubuntu-release-upgrader-gtk
The following packages will be upgraded:
  gir1.2-mutter-14 libmutter-14-0 mutter-common mutter-common-bin
4 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
Need to get 1,625 kB of archives.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 mutter-common all 46.2-1ubuntu0.24.04.2 [49.1 kB]
Get:2 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 gir1.2-mutter-14 amd64 46.2-1ubuntu0.24.04.2 [131 k
B]
Get:3 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 libmutter-14-0 amd64 46.2-1ubuntu0.24.04.2 [1,392 k
B]
Get:4 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 mutter-common-bin amd64 46.2-1ubuntu0.24.04.2 [52.1
 kB]
Fetched 1,625 kB in 4s (442 kB/s)
(Reading database ... 148205 files and directories currently installed.)
Preparing to unpack .../mutter-common_46.2-1ubuntu0.24.04.2_all.deb ...
Unpacking mutter-common (46.2-1ubuntu0.24.04.2) over (46.2-1ubuntu0.24.04
```

2. Install the BIND9 package along with utilities
   Command : **sudo apt install bind9 bind9utils bind9-doc**

```
it23265592@it23265592-VirtualBox:~$ sudo apt install bind9 bind9utils bind9-doc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bind9-utils
Suggested packages:
  bind-doc
The following NEW packages will be installed:
  bind9 bind9-doc bind9-utils bind9utils
0 upgraded, 4 newly installed, 0 to remove and 3 not upgraded.
Need to get 3,666 kB of archives.
After this operation, 8,936 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 bind9-utils amd64 1:9.18.28-0ubuntu0.24.04.1 [159 kB
]
Get:2 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 bind9 amd64 1:9.18.28-0ubuntu0.24.04.1 [254 kB]
Get:3 http://lk.archive.ubuntu.com/ubuntu noble-updates/main amd64 bind9-doc all 1:9.18.28-0ubuntu0.24.04.1 [3,249 kB]
Get:4 http://lk.archive.ubuntu.com/ubuntu noble-updates/universe amd64 bind9utils all 1:9.18.28-0ubuntu0.24.04.1 [3,682
 B]
Fetched 3,666 kB in 4s (869 kB/s)
Selecting previously unselected package bind9-utils.
(Reading database ... 148205 files and directories cur
rently installed.)
Preparing to unpack .../bind9-utils_1%3a9.18.28-0ubunt
u0.24.04.1_amd64.deb ...
Unpacking bind9-utils (1:9.18.28-0ubuntu0.24.04.1) ...
Selecting previously unselected package bind9.
Preparing to unpack .../bind9_1%3a9.18.28-0ubuntu0.24.
04.1_amd64.deb ...
Unpacking bind9 (1:9.18.28-0ubuntu0.24.04.1) ...
Selecting previously unselected package bind9-doc.
```

3. Then check your ip address
   Command : **ifconfig**

```
it23265592@it23265592-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.15  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 2402:d000:812c:2900:55b1:4b42:c407:1db4  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x20<link>
        inet6 2402:d000:812c:2900:a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x0<global>
        ether 08:00:27:c9:c7:2e  txqueuelen 1000  (Ethernet)
        RX packets 21344  bytes 24011041 (24.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10626  bytes 1524696 (1.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 1114  bytes 167409 (167.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1114  bytes 167409 (167.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

4. Then change the directory to "/etc/bind"
   Command : **cd /etc/bind**

```
it23265592@it23265592-VirtualBox:~$ cd /etc/bind
it23265592@it23265592-VirtualBox:/etc/bind$
```

5. Then check what are the file in there.
   Command : **ls**

```
it23265592@it23265592-VirtualBox:/etc/bind$ ls
bind.keys   db.127   db.empty   named.conf                named.conf.local    rndc.key
db.0        db.255   db.local   named.conf.default-zones  named.conf.options  zones.rfc1918
```

6. Then go to “name.conf.options”

   Command : **sudo nano name.conf.options**

```
it23265592@it23265592-VirtualBox:/etc/bind$ sudo nano named.conf.options
```

7. Then edit as following :

```
it23265592@it23265592-VirtualBox: /etc/bind
 GNU nano 7.2                          named.conf.options
//define LAN network
acl MYLAN{
        192.168.1.0/24;
};
options {
        //default directory
        directory "/var/cache/bind";
        //allow queries form localhost and LAN network
        allow-query{
                localhost;
                MYLAN;
};
        //use google DNS as a forwarder
    forwarders {
        8.8.8.8;  // Google DNS
        8.8.4.4;  // Google DNS
    };

        //Allow recursive queries
        recursion yes;

};
```

```
//define LAN network
        acl MYLAN{
                192.168.1.0/24;
        };

        options {
                //default directory
                directory "/var/cache/bind";
                //allow queries form localhost and LAN network
                allow-query{
                        localhost;
                        MYLAN;
        };
                //use google DNS as a forwarder
                forwarders {
                        8.8.8.8; //google DNS
                        8.8.4.4; //google DNS

                //Allow recursive queries
                recursion yes;

        };
```

8. Then check for any syntax errors of your configuration file:

   Command : **sudo named-checkconf named.conf.options**

```
it23265592@it23265592-VirtualBox:/etc/bind$ sudo named-checkconf named.conf.options
```

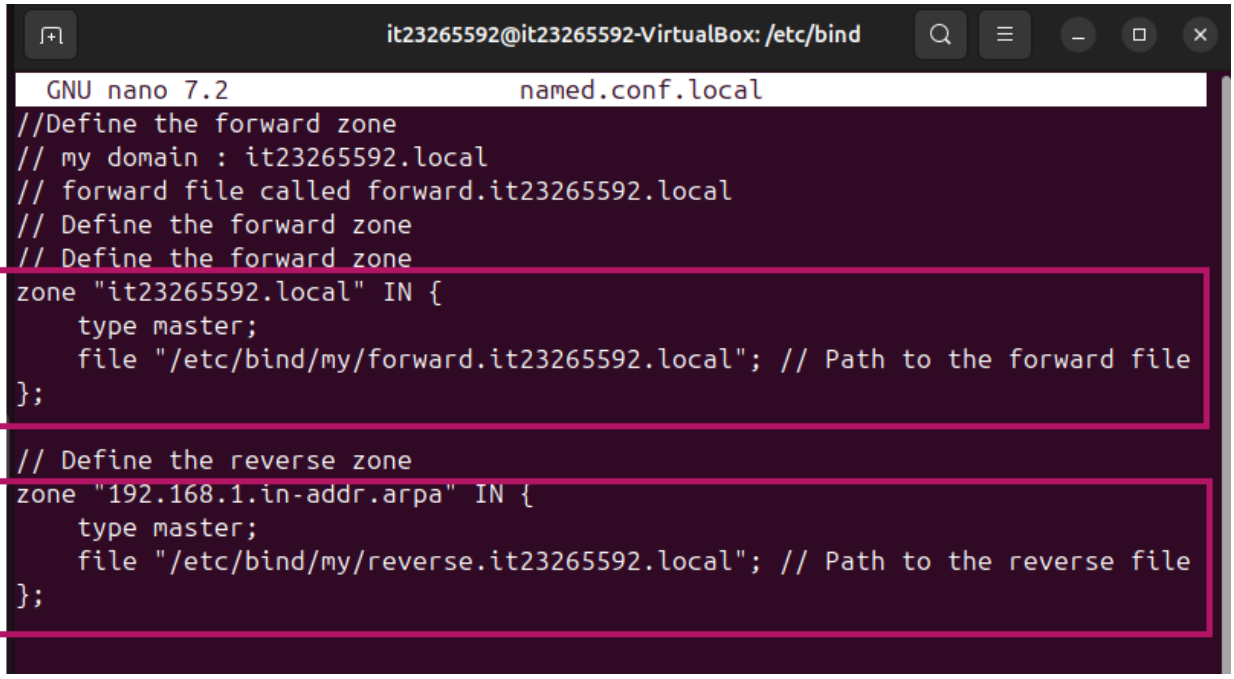
   - If it will not give any error message, your configuration file is syntax error free.

9. Then go to "named.conf.local" file and open it by nano editor

   Command : **sudo nano named.conf.local**

```
it23265592@it23265592-VirtualBox:/etc/bind$ sudo nano named.conf.local
```

   - Then do your configuration according to your network (my : IP address : 192.168.1.15)



10. Then check whether your configuration file is syntax error free;

Command : **sudo named-checkconf named.conf.local**

- If it is correct , it will not give any message.

```
it23265592@it23265592-VirtualBox:/etc/bind$ sudo named-checkconf named.conf.local
it23265592@it23265592-VirtualBox:/etc/bind$
```

11. Then create directory named “my” as your configuration file

Command : **sudo mkdir my**

```
it23265592@it23265592-VirtualBox:/etc/bind$ sudo mkdir my
it23265592@it23265592-VirtualBox:/etc/bind$
```

12. Then go inside of that directory which is “my”

Command : **cd my**

```
it23265592@it23265592-VirtualBox:/etc/bind$ cd my
it23265592@it23265592-VirtualBox:/etc/bind/my$
```

See whether is any file there

Command : **ls**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ ls
it23265592@it23265592-VirtualBox:/etc/bind/my$
```

- You can see , there are no any file in there.

13. Then create your forward and reverse file as following :

13.1. First change your forward.it23265592.local file

Command : **sudo nano forward.it2326559.local**

13.2. Then change it as following :

```
$TTL 604800
; SOA record with MNAME and RNAME updated
@ IN SOA ns.it23265592.local. admin.it23265592.local. (
   3         ; Serial (increment this number after each change)
   604800    ; Refresh
   86400     ; Retry
   2419200   ; Expire
   604800 )  ; Negative Cache TTL

; Name server record
@ IN NS ns.it23265592.local.

; A record for name server (DNS server IP)
ns IN A 192.168.1.15

; A record for main domain IP
www IN A 192.168.1.21
; Mail handler (MX record) for the domain it23265592.local
it23265592.local. IN MX 10 mail.it23265592.local.
mail IN A 192.168.1.15
; A record for subdomains or clients
client1 IN A 192.168.1.111
client2 IN A 192.168.1.112
```

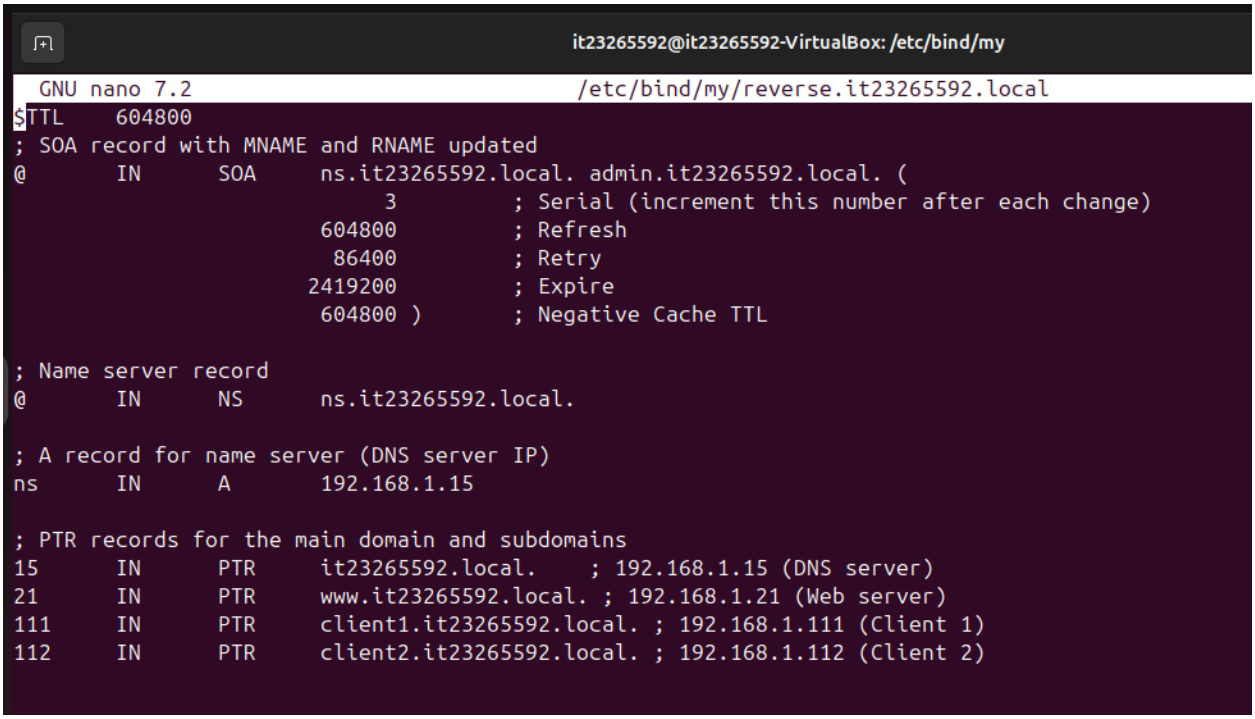
```
it23265592@it23265592-VirtualBox: /etc/bind/my
  GNU nano 7.2                                    forward.it23265592.local *
$TTL    604800
;SOA record with MNAME and RNAME updated
@       IN      SOA     ns.it23265592.local. admin.it23265592.local. (
                              3         ; Serial(increment this number after each change)
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
; Name server record
@       IN      NS      ns.it23265592.local.
; A record for name server(DNS server IP)
ns      IN      A       192.168.1.15

; A record for main domain IP
www IN A 192.168.1.21

;Mail handler (MX record) for domain it23265592.local
it23265592.local. IN MX 10 mail.it23265592.local
mail IN A 192.168.1.15

;A record for subdomain or clients
client1 IN A 192.168.1.111
client2 IN A 192.168.1.112
```

13.3. Then set your reverse file which is “reverse.it23265592.local”

Command : **sudo nano reverse.it2326592.local**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo nano reverse.it23265592.local
```

13.4. Then edit your file as following :



```
$TTL 604800
; SOA record with MNAME and RNAME updated
@ IN SOA ns.it23265592.local. admin.it23265592.local. (
    3            ; Serial (increment this number after each change)
    604800     ; Refresh
    86400      ; Retry
    2419200    ; Expire
    604800 )   ; Negative Cache TTL

; Name server record
@ IN NS ns.it23265592.local.
; A record for name server(DNS server IP)
ns IN A 192.168.1.15

; PTR records for reverse mapping for the main domain and subdomains
15  IN PTR it23265592.local.          ; 192.168.1.15 (DNS server)
21  IN PTR www.it23265592.local.      ; 192.168.1.21 (Web server)
111 IN PTR client1.it23265592.local.  ; 192.168.1.111 (Client 1)
112 IN PTR client2.it23265592.local.  ; 192.168.1.112 (Client 2)
```

14. Then check any syntax error in your forward file

Command : **sudo named-checkzone it23265592.local forward.it23265592.local**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo named-checkzone it23265592.local forward.it23265592.local
zone it23265592.local/IN: it23265592.local/MX 'mail.it23265592.local.it23265592.local' has no address records (A or AAAA
)
zone it23265592.local/IN: loaded serial 3
OK
```

15. Then check any syntax error in your reverse file

Command : **sudo named-checkzone it23265592.local reverse.it23265592.local**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo nano /etc/bind/my/reverse.it23265592.local
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo named-checkzone it23265592.local reverse.it23265592.local
zone it23265592.local/IN: loaded serial 3
OK
```

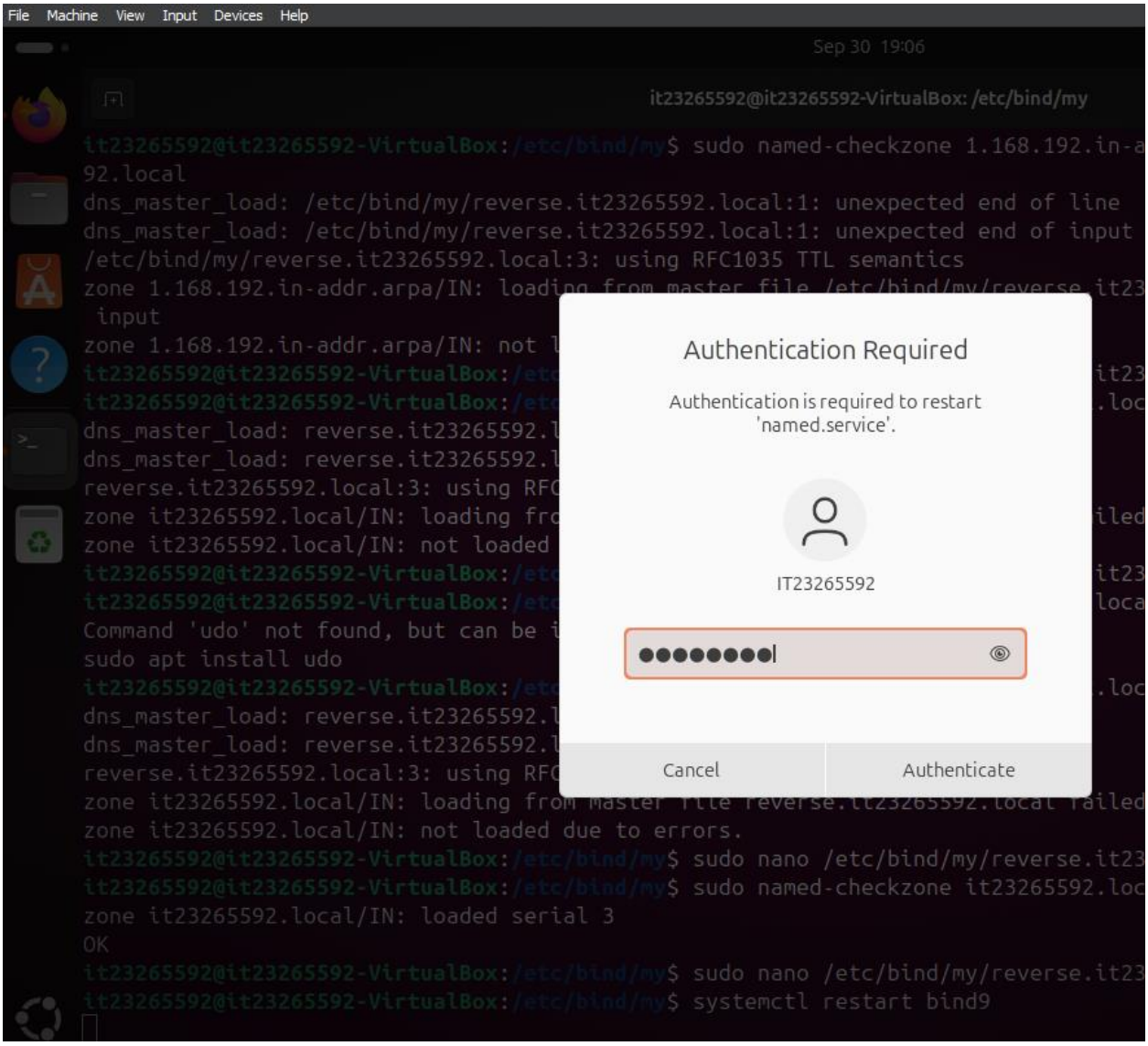
16. Then you can identify that there is no any syntax error in these both files which are “reverse.it23265592.local” and “forward.it23265592.local”

17. Then restart your DNS service

Command : **systemctl restart bind9**

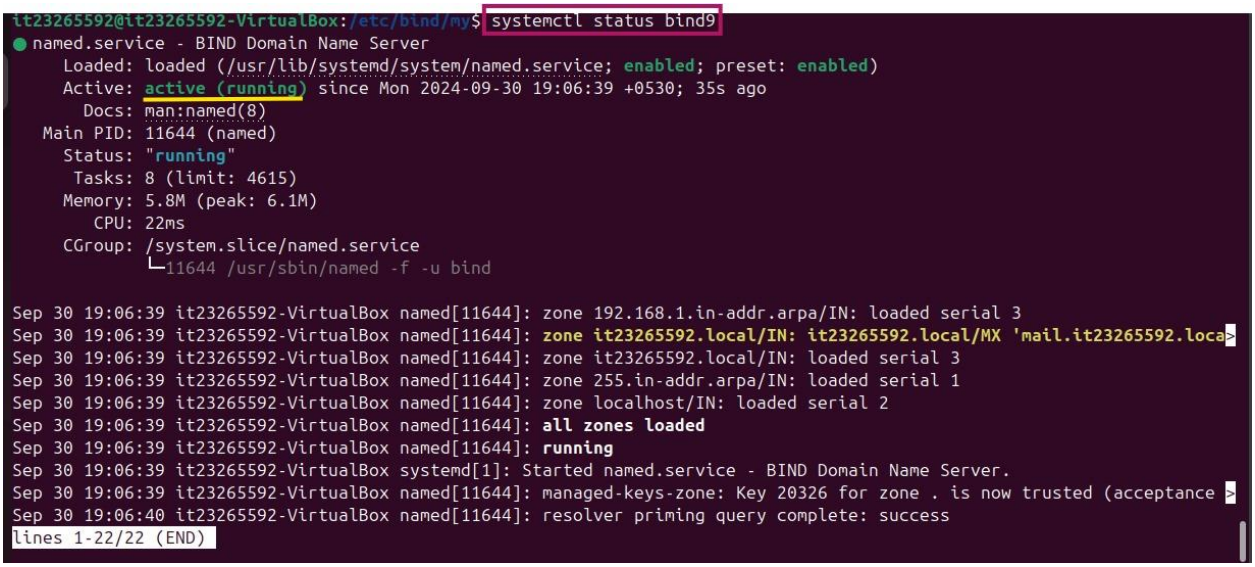
```
it23265592@it23265592-VirtualBox:/etc/bind/my$ systemctl restart bind9
```

- Then give the password for continue this


File Machine View Input Devices Help
Sep 30 19:06
it23265592@it23265592-VirtualBox: /etc/bind/my
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo named-checkzone 1.168.192.in-a
92.local
dns_master_load: /etc/bind/my/reverse.it23265592.local:1: unexpected end of line
dns_master_load: /etc/bind/my/reverse.it23265592.local:1: unexpected end of input
/etc/bind/my/reverse.it23265592.local:3: using RFC1035 TTL semantics
zone 1.168.192.in-addr.arpa/IN: loading from master file /etc/bind/my/reverse.it23
input
zone 1.168.192.in-addr.arpa/IN: not l
dns_master_load: reverse.it23265592.l
dns_master_load: reverse.it23265592.l
reverse.it23265592.local:3: using RFC
zone it23265592.local/IN: loading fro
zone it23265592.local/IN: not loaded
Command 'udo' not found, but can be i
sudo apt install udo
dns_master_load: reverse.it23265592.l
dns_master_load: reverse.it23265592.l
reverse.it23265592.local:3: using RFC
zone it23265592.local/IN: loading from master file reverse.it23265592.local failed
zone it23265592.local/IN: not loaded due to errors.
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo nano /etc/bind/my/reverse.it23
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo named-checkzone it23265592.loc
zone it23265592.local/IN: loaded serial 3
OK
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo nano /etc/bind/my/reverse.it23
it23265592@it23265592-VirtualBox:/etc/bind/my$ systemctl restart bind9
Authentication Required
Authentication is required to restart 'named.service'.
IT23265592
Cancel
Authenticate

18. Then check the status whether is BIND DNS running or not.

Command : **systemctl status bind9**



19. Then you have to configure the bind9 service to be allowed through the firewall.

19.1 See whether is bind9 service allow throgh the firewall

Command : **sudo ufw staus**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo ufw status
Status: inactive
```

19.2. Then enable the firewall

Command : **sudo ufw enable**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo ufw enable
Firewall is active and enabled on system startup
```

19.3. Then allow the default pors for BIND (usually TCP/UDP 53) through the firewall

Command : **sudo ufw alloww bind9**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo ufw allow bind9
Skipping adding existing rule
Skipping adding existing rule (v6)
```

20. See whether is bind9 service allow throgh the firewall
    Command : **sudo ufw staus**

```
it23265592@it23265592-VirtualBox:/etc/bind/my$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
67/udp                     ALLOW       Anywhere
Bind9                      ALLOW       Anywhere
67/udp (v6)                ALLOW       Anywhere (v6)
Bind9 (v6)                 ALLOW       Anywhere (v6)
```

Shell Scripting and Security

There are some basic components of the shell scripting :

1. Shebang : **#!/bin/bash**
   At the top of the script indicates that the script should be run with the Bash shell.
2. Comments : Use **#** to add comments
3. Variables : Define variables to store values
4. Control flow : Use **if, for, while for** control flow
5. Functions : define reusable code blocks

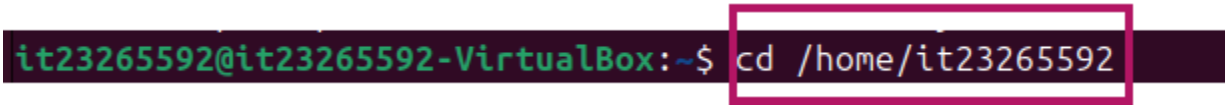
Task 1

Write a script to automate a report that captures key system details every day. This script can be scheduled to run using cron jobs. Get System Information such as Date, Uptime, Free memory and Disk Usage. Create a report file at the location with the file name as mentioned below. Destination directory : /home/user/system_reports

- I didn't use "sudo" because these commands like date, uptime,free, df are executed by the regular users.

1. First change the directory as mentioned above as " /home/it23265592/"

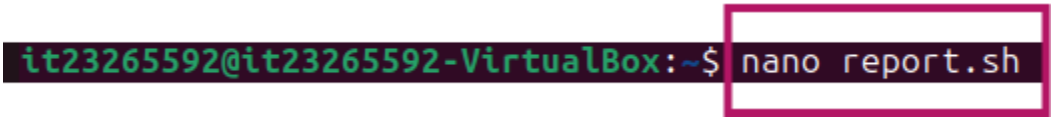
Command : **cd /home/it23265592**



2. Then create directory which is "system_reports"

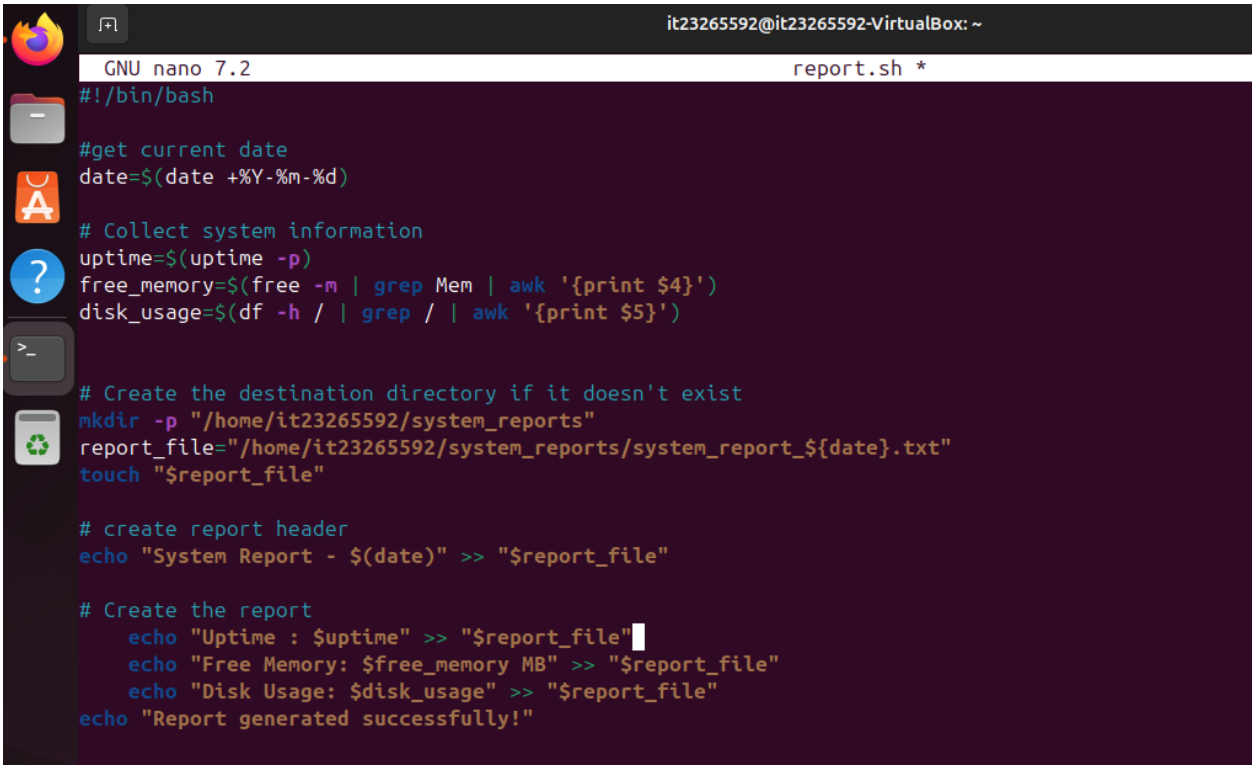
Then create the system_report script file

command : **nano report.sh**



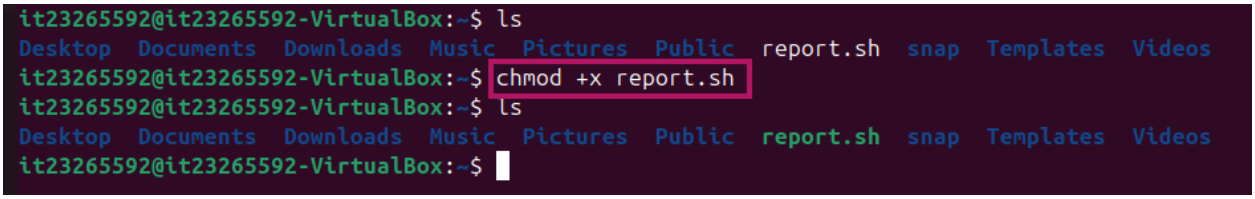
3. Then write your shell scripting :

```
#!/bin/bash
# Get current date
date=$(date +%Y-%m-%d)
# Get system information
uptime=$(uptime -p)
free_mem=$(free -m | grep Mem: | awk '{print $4}')
disk_usage=$(df -h / | grep / | awk '{print $5}')
# Create report file path
mkdir -p "/home/user/system_reports"
report_file="/home/user/system_reports/system_report_${date}.txt"
touch "$report_file"
# Create report header
echo "System Report - $(date)" >> "$report_file"
# Append system information
echo "Uptime: $uptime" >> "$report_file"
echo "Free Memory: $free_mem MB" >> "$report_file"
echo "Disk Usage: $disk_usage" >> "$report_file"
echo "Report generated successfully!"
```



4. Make the script executable

Command : **chmod +x report.sh**



5. Test the script

Run it manually to see whether is it worked or not ?

Command : **./report.sh**

```
it23265592@it23265592-VirtualBox:~$ ./report.sh
Report generated successfully!
```

6. Then you can see that there is a file directory which is located in "/home/it23265592/system_reports"

Command : **ls -al**

```
it23265592@it23265592-VirtualBox:~$ ls -al
total 100
drwxr-x--- 17 it23265592 it23265592 4096 Oct  1 09:57 .
drwxr-xr-x  3 root       root       4096 Sep 27 15:40 ..
-rw-------  1 it23265592 it23265592 4751 Sep 30 23:15 .bash_history
-rw-r--r--  1 it23265592 it23265592  220 Mar 31  2024 .bash_logout
-rw-r--r--  1 it23265592 it23265592 3771 Mar 31  2024 .bashrc
drwx------ 10 it23265592 it23265592 4096 Sep 27 15:47 .cache
drwx------ 13 it23265592 it23265592 4096 Sep 29 13:33 .config
drwxr-xr-x  5 it23265592 it23265592 4096 Oct  1 09:33 Desktop
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Documents
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Downloads
drwx------  2 it23265592 it23265592 4096 Sep 27 20:10 .gnupg
-rw-------  1 it23265592 it23265592   20 Sep 30 19:16 .lesshst
drwx------  4 it23265592 it23265592 4096 Sep 27 15:40 .local
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Music
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Pictures
-rw-r--r--  1 it23265592 it23265592  807 Mar 31  2024 .profile
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Public
-rw-rw-r--  1 it23265592 it23265592   23 Oct  1 09:57 report_file
-rwxrwxr-x  1 it23265592 it23265592  689 Oct  1 09:56 report.sh
drwx------  5 it23265592 it23265592 4096 Sep 29 19:43 snap
drwx------  2 it23265592 it23265592 4096 Sep 27 15:40 .ssh
-rw-r--r--  1 it23265592 it23265592    0 Sep 29 12:38 .sudo_as_admin_successful
drwxrwxr-x  2 it23265592 it23265592 4096 Oct  1 09:57 system_reports
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Templates
drwxr-xr-x  2 it23265592 it23265592 4096 Sep 27 15:40 Videos
```

7. When you go to inside you can see that there is a generated report in there

   Go inside to system_reports : cd system_reports

See files in there : **ls -al**

```
it23265592@it23265592-VirtualBox:~$ cd system reports
it23265592@it23265592-VirtualBox:~/system_reports$ ls
system_report_2024-10-01.txt
```

8. Then open the crontab

Command : **crontab -e**

```
it23265592@it23265592-VirtualBox:~$ crontab -e
```

9. Add your crone job :

Command : **09*** /home/it23265592/system_reports/report.sh**

```
it23265592@it23265592-VirtualBox: ~
 GNU nano 7.2                        /tmp/crontab.PfC5go/crontab *
0 9 * * * /home/it23265592/system_reports/report.sh
```

This crone job run in following specific time :

0: 0th minute

9: 9 Am

*: everyday of the month

*: every month

*: every day of the week

To verify whether your crone was added check it

Command : **crontab -l**

```
it23265592@it23265592-VirtualBox:~$ crontab -l
0 9 * * * /home/it23265592/system_reports/report.sh
```

Task 2

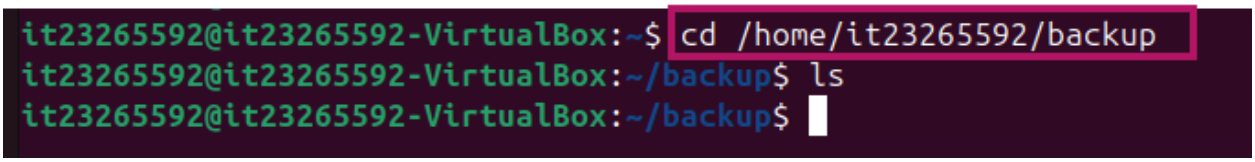
Write a script to automate the backup of a critical directory (/home/user/documents) containing important files. This script can be scheduled to run periodically. Make sure to name the backup file with the date.

Source directory : /home/user/documents

Destination directory : /home/user/backup/document

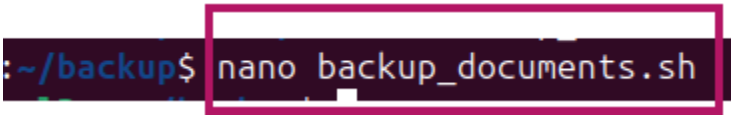
1. Create directory "/home/it23265592/backup"

Command : **cd /home/it23265592/backup**

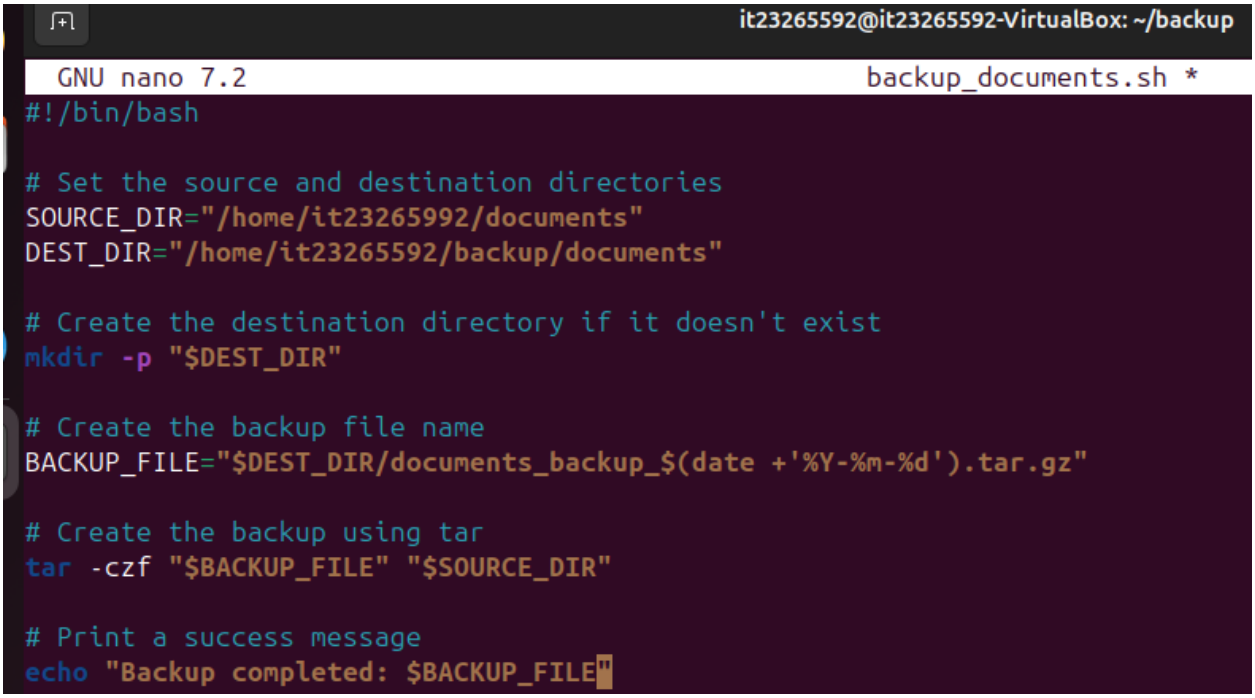


2. Open nano edit to start script

Command : **nano backup_documents**



3. Create your script




```
#!/bin/bash

# set the source and destination directories

SOURCE_DIR="/home/it23265592/documents"

BACKUP_DIR="/home/it23265592/backup/documents"

# Create backup directory if it doesn't exist

mkdir -p "$DEST_DIR"

# create the backup file name

BACKUP_FILE="$DEST_DIR/documents_backup_$(date  +'%Y-%m-%d').tar.gz"

# create the backup using tar

tar -czf "$BACKUP_FILE"  "$SOURCE_DIR"

echo "Backup completed: $BACKUP_FILE"
```

4. The make it as executable file

Command : **chmod +x backup_documents.sh**

```
it23265592@it23265592-VirtualBox:~$ cd backup
it23265592@it23265592-VirtualBox:~/backup$ ls
backup_documents.sh
it23265592@it23265592-VirtualBox:~/backup$ chmod +x backup_documents.sh
it23265592@it23265592-VirtualBox:~/backup$ ls
backup_documents.sh
it23265592@it23265592-VirtualBox:~/backup$
```

5. Then add crontab

5.1. Go inside of documents directory

Command : **cd documents**

```
it23265592@it23265592-VirtualBox:~/backup$ ls
backup_documents.sh  documents
it23265592@it23265592-VirtualBox:~/backup$ cd documents
it23265592@it23265592-VirtualBox:~/backup/documents$
```

5.2. Command : **crontab -e**

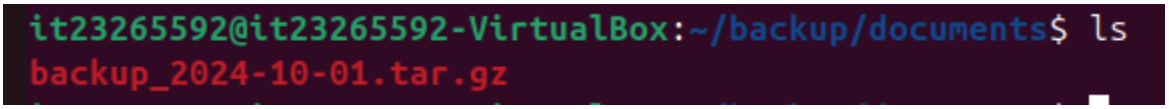
```
it23265592@it23265592-VirtualBox:~/backup$ crontab -e
```

5.3. Add crone job

Command : **0 8 * * * /home/it23265592/backup/backup_documents.sh**

```
0 8 * * * /home/it23265592/backup/backup_documents.sh
```

5.4. Then you can see some .tar.gz file there



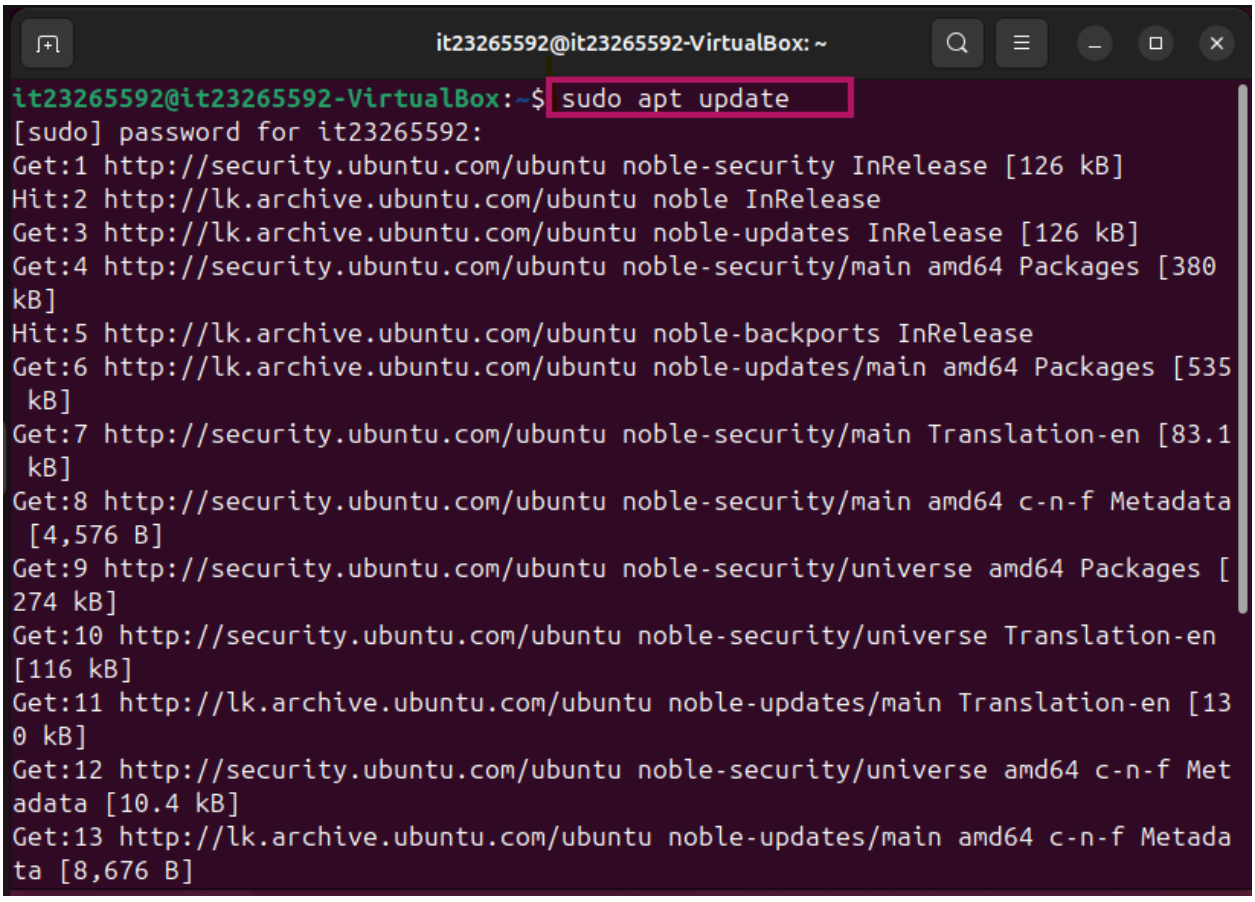
Configuration steps for SSH server, iptables and ACLs.

SSH server configuration

1. Update system

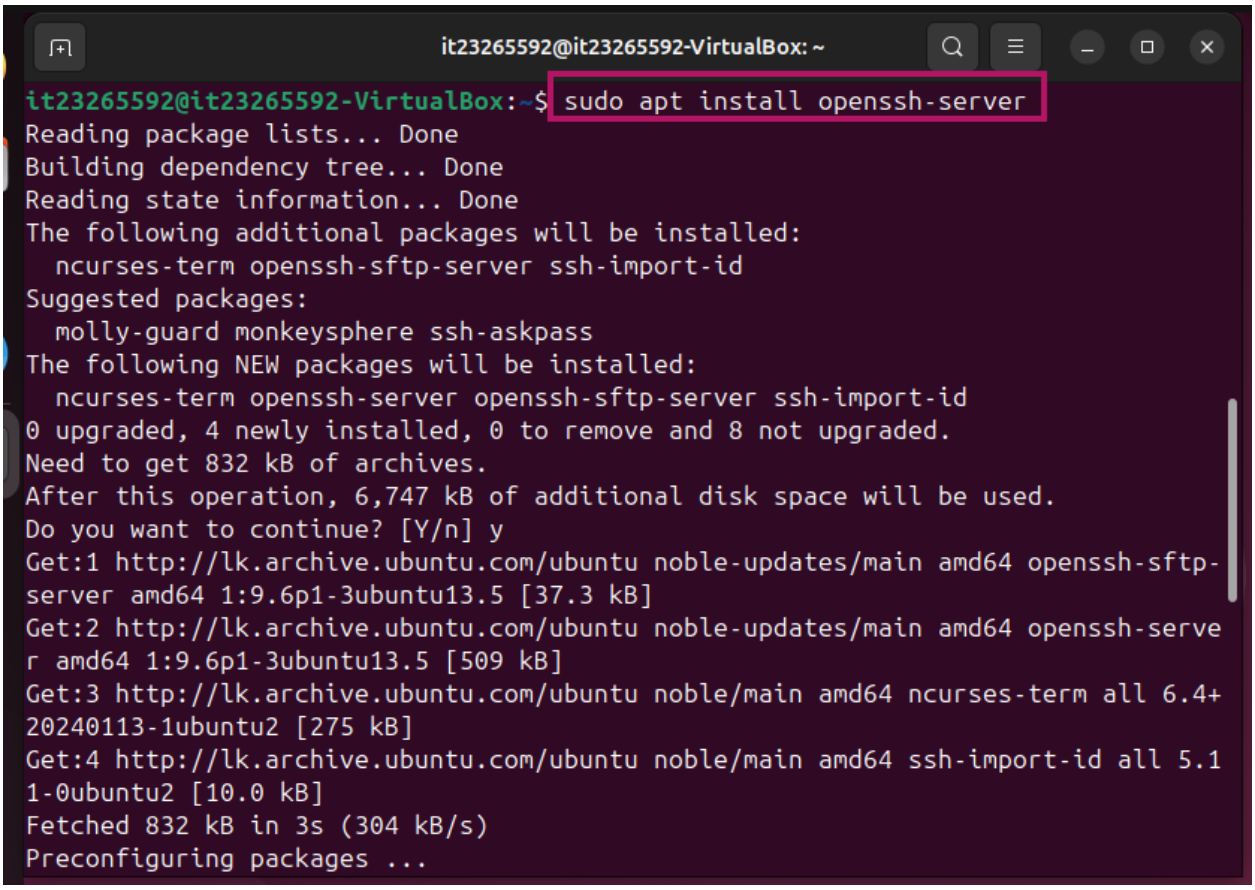
Command : **sudo apt update**

Give your password to continue this process



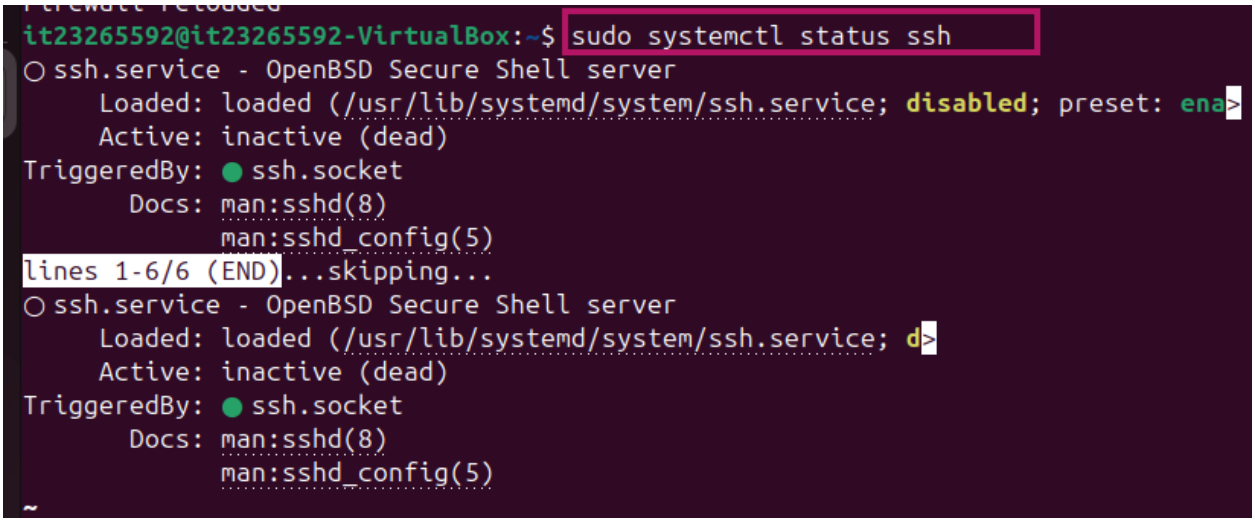
2. Install openssh server

Command : **sudo apt install openssh-server**



3. Check the status of the ssh

Command : **sudo systemctl status ssh**

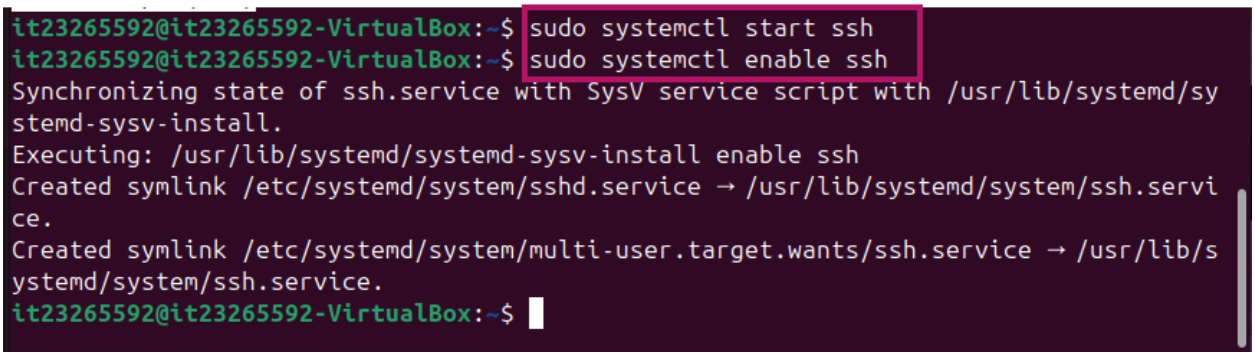


4. Enable open ssh

4.1. Command : **sudo systemctl start ssh**

After that

4.2. Command : **sudo systemctl enable ssh**



5. Then restart your service to apply modifications

command :**sudo systemctl restart ssh**

6. Then check whether is ssh service running or not

Command : **sudo sytemctl status ssh**

7. Port 22 (TCP) is used for SSH protocol by default.

- Then allow ssh through the firewall with which can listen for incoming requests.

- Allow ssh through the firewall

Command : **sudo ufw allow ssh**

```
it23265592@it23265592-VirtualBox:~$ sudo ufw allow ssh
Rule added
Rule added (v6)
```

Command : **sudo ufw enable**

8. Thnen check the status

Command : **sudo ufw status**

```
it23265592@it23265592-VirtualBox:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
67/udp                     ALLOW       Anywhere
Bind9                      ALLOW       Anywhere
22/tcp                     ALLOW       Anywhere
67/udp (v6)                ALLOW       Anywhere (v6)
Bind9 (v6)                 ALLOW       Anywhere (v6)
22/tcp (v6)                ALLOW       Anywhere (v6)
```

Then we have to locate the IP address of the server computer to connect with client.

9. Then use command : **ifconfig**

```
it23265592@it23265592-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.15  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 2402:d000:812c:40e2:a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::a00:27ff:fec9:c72e  prefixlen 64  scopeid 0x20<link>
        inet6 2402:d000:812c:40e2:e08c:84fe:ed74:3284  prefixlen 64  scopeid 0x0<global>
        ether 08:00:27:c9:c7:2e  txqueuelen 1000  (Ethernet)
        RX packets 3680  bytes 2038093 (2.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2711  bytes 437674 (437.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 556  bytes 87770 (87.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 556  bytes 87770 (87.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

it23265592@it23265592-VirtualBox:~$
```

10. Then I use another Ubuntu VM as client computer to be top up and then its terminal used to travers into our original machine.

Prerequisites:

You have access to both the client machine and the Ubuntu VM (virtual machine).

Both systems are connected to the same network or have accessible IP addresses.

The openssh-server is installed and running on the VM.

11. install SSH
command : sudo apt-get install openssh-client

```
it23265592client@it23265592client-VirtualBox:~$ sudo apt-get install openssh-client
[sudo] password for it23265592client:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-client is already the newest version (1:9.6p1-3ubuntu13.5).
openssh-client set to manually installed.

0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
```

12. SSH from Client to Ubuntu VM
    - Then you have to know your username and IP address.
      - ✓ Username : it23265592
      - ✓ IP address : 192.168.1.15
    - Then connect through the ssh

    Command : **ssh <username@ipAddress>**

    In my case : it23265592@192.168.1.15

```
it23265592client@it23265592client-VirtualBox:~$ ssh it23265592@192.168.1.15
The authenticity of host '192.168.1.15 (192.168.1.15)' can't be established.
ED25519 key fingerprint is SHA256:rYIWPe723W5GP/x0I+HkDPHpbolJDlXmRNebY1Fm0ec
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.1.15' (ED25519) to the list of known host
it23265592@192.168.1.15's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-45-generic x86_64)

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

Expanded Security Maintenance for Applications is not enabled.

6 updates can be applied immediately.
5 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

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

Last login: Tue Oct  1 16:44:18 2024 from 192.168.1.14
```

12.1. Accept the SSH Key (First Time Only)
Type 'yes' and press Enter.

12.2. Enter the Password
After entering the password correctly, you'll have SSH access to the Ubuntu VM.

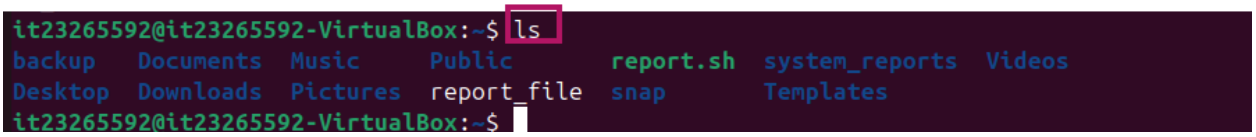
13. Commands You Can Run After SSH
Once logged in via SSH, you can run any terminal commands on the Ubuntu VM as if you were directly using the VM.

13.1. To check system info
Command : uname -a

```
it23265592@it23265592-VirtualBox:~$ uname -a
Linux it23265592-VirtualBox 6.8.0-45-generic #45-Ubuntu SMP PREEMPT_DYNAMIC Fri Aug 30 12
86_64 GNU/Linux
```

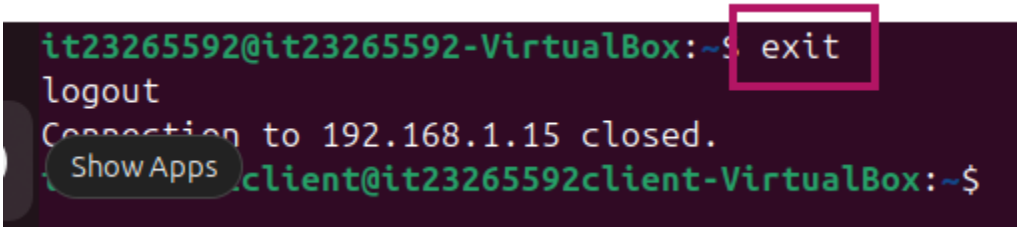
13.2. To list the files in the home directory:

Command : ls



14. Exit from SSH

Command : exit



Iptables

- A firewall, or iptables, is a program that runs on all Linux operating systems. The greatest feature is that this comes with many distributions under the name "ufw," which is typically the firewall's default. Initially, we will use install to disable ufw and then run iptables.

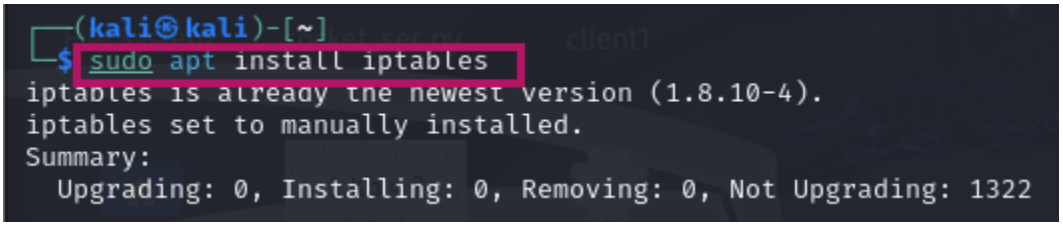
1. Disable ufw

Command : **sudo ufw disable**

- By giving your password you can continue the process

2. Then install the iptables

Command : **sudo apt install iptables**



Note : **General Command Syntax for iptables**

- ✓ -A : Append rule
- ✓ -P : Set default policy
- ✓ -i : Specify the input interface
- ✓ -s : Source IP
- ✓ --dport : Specify the destination port

3. Then you can manage your iptables as you wish .

- You can delete and clear and also add new rules to your iptables firewall.
- Then change your user mode as “super user”

Command : **sudo su**

- There are some flag you have to know
  - ✓ -F : flushes out all rules
  - ✓ -X : flushing out all the existing chains
  - ✓ -L : see the active firewall rules now

- iptables -F
- iptables -X
- iptables -L

Rules written in the firewall(or in iptable)

For Securing a Web Server

- iptables -P INPUT **DROP**
  - *all incoming* network traffic will be *blocked* if not explicitly allowed.
- iptables -A **INPUT** -p tcp --dport 80 -j ACCEPT
  - allows incoming port 80 (HTTP) requests to the web server, ensuring the server can respond to web traffic.
- iptables -A INPUT -p tcp --dport 443 -j ACCEPT

  allows incoming (443) HTTPS requests to the web server, enabling encrypted connections.

```
┌──(root㉿kali)-[/home/kali]
└─# iptables -P INPUT DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -A INPUT -p tcp --dport 80 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# iptables -A INPUT -p tcp --dport 443 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Remote Administration Access

Allow ssh port(22) only from specific IP address (your machine IP address)

Command : **sudo iptables -A INPUT -p tcp -s 192.168.1.15 --dport 22 -j ACCEPT**

```
┌──(root㉿kali)-[/home/kali]
└─# sudo iptables -A INPUT -p tcp -s 192.168.1.15 --dport 22 -j ACCEPT
```

Block ssh traffic from all other IP address

Command : **sudo iptables -A INPUT -p tcp --dport 22 -j DROP**

```
┌──(root㉿kali)-[/home/kali]
└─# sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

Full picture

```
┌──(root㉿kali)-[/home/kali]
└─# sudo iptables -A INPUT -p tcp -s 192.168.1.15 --dport 22 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# sudo iptables -A INPUT -p tcp --dport 22 -j DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  192.168.1.15         anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Allow Specific Applications

If the app uses the standard HTTPS port (443), this rule is already covered. However, if it uses a different port, you can modify it as follows:

Command : **sudo iptables -A INPUT -p tcp --dport <application-port> -j ACCEPT**

But you can try it as following

```
┌──(root㉿kali)-[/home/kali]
└─# iptables -P INPUT DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -P OUTPUT DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -P FORWARD DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -A INPUT -p tcp --dport 443 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  192.168.1.15         anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
```

Command :

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT
```

Allow Pings (ICMP Echo Request)

Command :

iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT

iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT

iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT

iptables -A INPUT -p udp --dport 53 -j ACCEPT

iptables -A OUTPUT -p udp --dport 53 -j ACCEPT

```
┌──(root㉿kali)-[/home/kali]
└─# systemctl status ufw
● ufw.service - Uncomplicated firewall
     Loaded: loaded (/usr/lib/systemd/system/ufw.service; enabled; preset: enabled)
     Active: active (exited) since Sun 2024-10-06 12:21:20 EDT; 16min ago
       Docs: man:ufw(8)
    Process: 532 ExecStart=/usr/lib/ufw/ufw-init start quiet (code=exited, status=0/SUCCESS)
   Main PID: 532 (code=exited, status=0/SUCCESS)
        CPU: 683ms

Oct 06 12:21:06 kali systemd[1]: Starting ufw.service - Uncomplicated firewall...
Oct 06 12:21:20 kali systemd[1]: Finished ufw.service - Uncomplicated firewall.

┌──(root㉿kali)-[/home/kali]
└─# ufw allow out 53
Skipping adding existing rule
Skipping adding existing rule (v6)

┌──(root㉿kali)-[/home/kali]
└─# ufw status
Status: inactive

┌──(root㉿kali)-[/home/kali]
└─# ufw enable
Firewall is active and enabled on system startup

┌──(root㉿kali)-[/home/kali]
└─# ufw status
Status: active

To                         Action      From
--                         ------      ----
443                        ALLOW       Anywhere
80/tcp                     ALLOW       Anywhere
443 (v6)                   ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)

53                         ALLOW OUT   Anywhere
53 (v6)                    ALLOW OUT   Anywhere (v6)


┌──(root㉿kali)-[/home/kali]
└─# ufw allow https
Skipping adding existing rule
Skipping adding existing rule (v6)

┌──(root㉿kali)-[/home/kali]
└─# ufw status
Status: active

To                         Action      From
--                         ------      ----
443                        ALLOW       Anywhere
80/tcp                     ALLOW       Anywhere
443 (v6)                   ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)

53                         ALLOW OUT   Anywhere
53 (v6)                    ALLOW OUT   Anywhere (v6)
```

systemctl enable ufw

ufw allow out 53

ufw status

ufw enable

ufw status

ufw allow https

ufw status

ping facebook.com

```
┌──(root㉿kali)-[/home/kali]
└─# ping facebook.com
PING facebook.com (157.240.235.35) 56(84) bytes of data.
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=1 ttl=50 time=44.3 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=2 ttl=50 time=47.1 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=3 ttl=50 time=47.0 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=4 ttl=50 time=47.8 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=5 ttl=50 time=45.0 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=6 ttl=50 time=49.6 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=7 ttl=50 time=45.4 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=8 ttl=50 time=52.0 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=9 ttl=50 time=46.4 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=10 ttl=50 time=47.9 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=11 ttl=50 time=46.3 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=12 ttl=50 time=55.1 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=13 ttl=50 time=46.5 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=14 ttl=50 time=59.7 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=15 ttl=50 time=57.2 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=16 ttl=50 time=45.2 ms
64 bytes from edge-star-mini-shv-04-sin6.facebook.com (157.240.235.35): icmp_seq=17 ttl=50 time=45.6 ms
^C
── facebook.com ping statistics ──
17 packets transmitted, 17 received, 0% packet loss, time 16035ms
rtt min/avg/max/mdev = 44.250/48.709/59.688/4.437 ms
```

Printer Server Access

```
┌──(root㉿kali)-[/home/kali]
└─# iptables -A INPUT -p tcp -s 192.168.1.15 --dport 9100 -j ACCEPT

┌──(root㉿kali)-[/home/kali]
└─# iptables -A INPUT -p tcp --dport 9100 -j DROP

┌──(root㉿kali)-[/home/kali]
└─# iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     tcp  --  192.168.1.15         anywhere             tcp dpt:ssh
DROP       tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
ACCEPT     udp  --  anywhere             anywhere             udp dpt:domain
```

Command :

iptables -A INPUT -p tcp -s 192.168.1.15 --dport 9100 -j ACCEPT

iptables -A INPUT -p tcp --dport 9100-j DROP

Implementing 5 best practices in a Linux based environment

1. **Update system**

- This includes update packages , operating system and services up to date.
- This ensures that the system has the latest security patches , performance improvements and bug fixes.
- It means they are set up properly with the correct settings (like IP address , gateways and DNS)
- It also includes drivers or services like DHCP, IPTABLES to avoid any vulnerabilities.
- After changing network interface settings, testing is critical
- So you can use commands like “ping ” to verify that the connection is stable and the correct IP addresses and routes are applied.

To update
Command : sudo apt update

2. Apply Strong Password Policies

*** Currently Needed Password:**
In order to modify a password, users must enter their existing password.
By limiting illegal access, authentication makes sure that only authorized users are able to update their passwords.

*** Changing Your Password:**
To develop strong passwords, enforce password complexity restrictions (such as minimum length and inclusion of special characters and numerals).
Password complexity improves defenses against guessing and brute-force attacks.

*** Replying with the Novel Password:**
The best practice is to make users confirm the new password in order to reduce mistakes and errors.
By lowering the possibility of entering a password incorrectly, error prevention helps to prevent user shutdowns.

*** Password Modification Verification:**

After a successful password change, the system hides the new password from view by not displaying it on the screen.

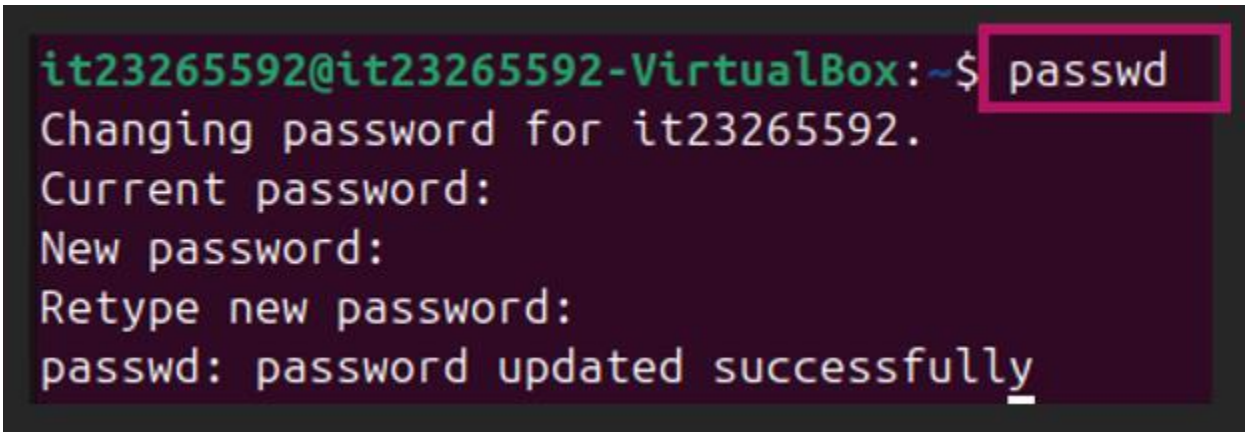
To improve security, a lot of Linux systems have password expiration rules in place that mandate regular password changes.

Strong Password Policies:
Passwords should contain a combination of capital and lowercase letters, numerals, and special characters, and they should be sufficiently long—at least 8 to 12 characters.

Stop users from selecting common or weak passwords like "admin" or "password123".

- ✓ Strengthen user authentication to reduce the risk of brute-force attacks.

- Update the system
  Command : **sudo apt update**


it23265592@it23265592-VirtualBox:~$ passwd
Changing password for it23265592.
Current password:
New password:
Retype new password:
passwd: password updated successfully

3. **Enable Firewall (UFW - Uncomplicated Firewall)**
   - ✓ Control and restrict incoming and outgoing network traffic to minimize potential vulnerabilities.

Implementing steps :

- Install UFW (it you not install yet)
  Command : **sudo apt install ufw**

```
it23265592@it23265592-VirtualBox:~$ sudo apt install ufw
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ufw is already the newest version (0.36.2-6).
0 upgraded, 0 newly installed, 0 to remove and 8 not upgraded.
```

- Then enable ufw
  Command : sudo ufw enable

```
it23265592@it23265592-VirtualBox:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

- Then add rules :
  Commands :
  sudo ufw allow ssh
  sudo ufw allow https
  sudo ufw allow http

```
it23265592@it23265592-VirtualBox:~$ sudo ufw allow ssh
Skipping adding existing rule
Skipping adding existing rule (v6)
it23265592@it23265592-VirtualBox:~$ sudo ufw allow https
Skipping adding existing rule
Skipping adding existing rule (v6)
it23265592@it23265592-VirtualBox:~$ sudo ufw allow http
Rule added
Rule added (v6)
```

- Then check the status :

Command : **sudo ufw status verbose**

```
it23265592@it23265592-VirtualBox:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
67/udp                     ALLOW IN    Anywhere
53 (Bind9)                 ALLOW IN    Anywhere
22/tcp                     ALLOW IN    Anywhere
443                        ALLOW IN    Anywhere
80/tcp                     ALLOW IN    Anywhere
67/udp (v6)                ALLOW IN    Anywhere (v6)
53 (Bind9 (v6))            ALLOW IN    Anywhere (v6)
22/tcp (v6)                ALLOW IN    Anywhere (v6)
443 (v6)                   ALLOW IN    Anywhere (v6)
80/tcp (v6)                ALLOW IN    Anywhere (v6)
```

4. Hardening

- Hardening is the process of making a system less vulnerable to threats in order to increase security. This includes putting in place a variety of security measures to guard against intrusions by reducing the attack surface, or the number of ways in which a system can be penetrated.

- IP Forwarding and Hardening:

  One popular hardening technique is to disable IP forwarding, especially for systems that aren't supposed to be routers.

- Stopping Intentional Routing:

  A system that is not intended to route traffic may unintentionally forward packets between networks if IP forwarding is allowed, potentially exposing private internal traffic to outside networks. This risk is decreased by turning off IP forwarding.

- Reducing Attack Surface:

  Attackers are unable to use the system to maliciously route or interrupt network traffic while IP forwarding is stopped (e.g., man-in-the-middle attacks, unauthorized network bridging).
  This avoids situations in which an infected system might be exploited to redirect attacks to other areas of the network.

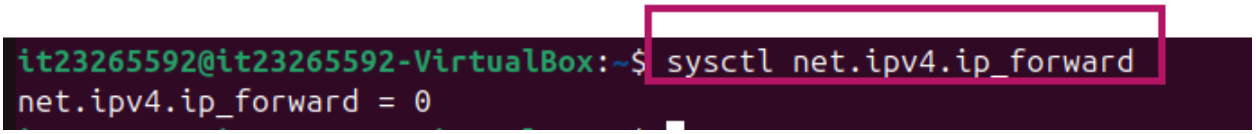
Other Common Hardening Practices:

- Disable Unnecessary Services
- Apply the Principle of Least Privilege
- Keep Software Up to Date
- Firewall Configuration
- Secure SSH

- Disable Unused Network Protocols

✓ Prevent the system from being used as a router, which could inadvertently expose internal network traffic.

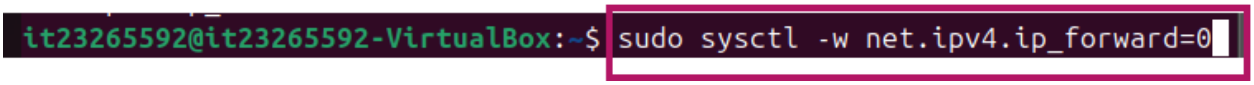
- Check the current status of IP forwarding
  Command : **sysctl net.ipv4.ip_forward**



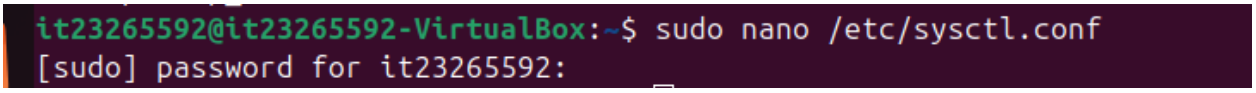
In my case secure .

But if your machine it may be not secure
Then you can implement it.

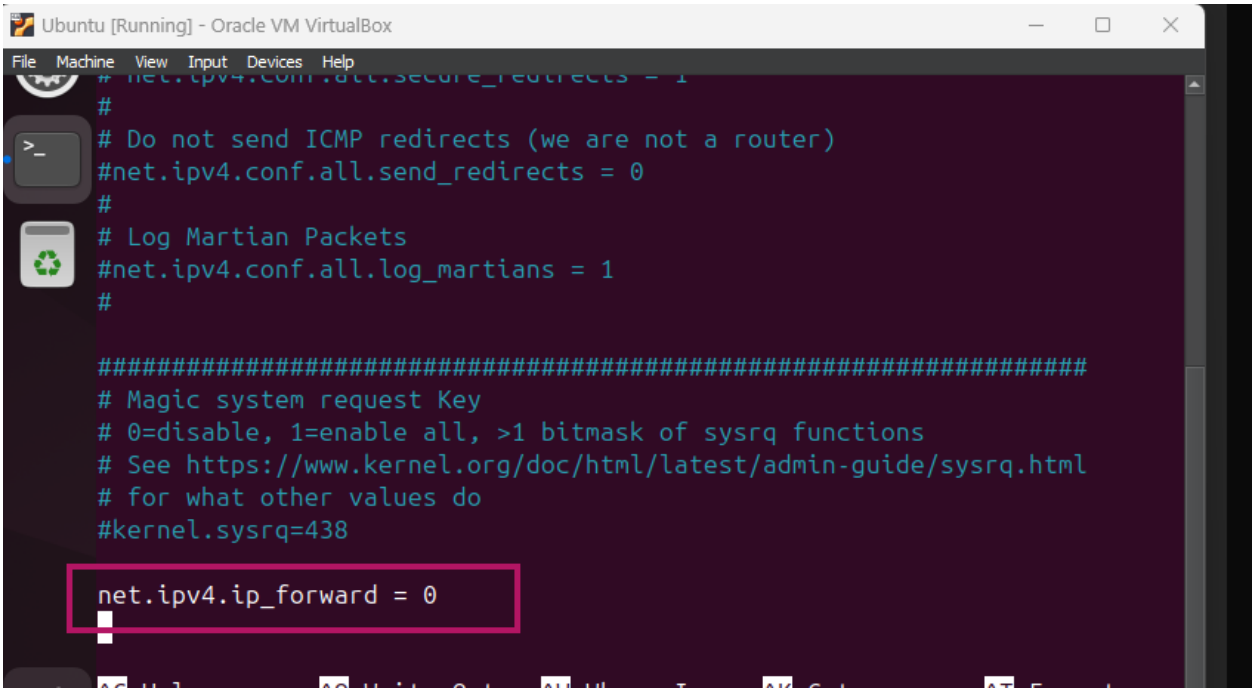
- Disable IP forwarding
  command : **sudo sysctl -w net.ipv4.ip_forward=0**



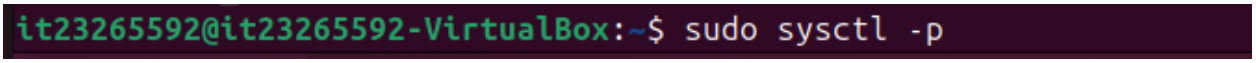
- To make it persistent , edit the "/etc/sysctl.conf" file
  Command : **sudo nano /etc/sysctl.conf**



- Then edit it as following :
  command : **net.ipv4.ip_forward = 0**



If you newly apply this you can apply the change with the following command
Command : sudo sysctl -p



Hardening is important because it provides :

- Prevent unauthorized access to exploit vulnerabilities
- Limit impact of a breach which is reduced the attacker's ability to mobe laterally within the network or escalate their privileges.

5. **Indentation is considered a best practice in a Linux OS**

- *Improved Readability*

Scripts, configuration files, and code are easier to read when properly indented, allowing developers and administrators to quickly grasp the logic and flow.

Users can find specific parts more quickly thanks to well-indented code, which also cuts down on the time it takes to understand complex structures.

- *Error Prevention*

In programming languages such as Python, where code blocks are defined by indentation, improper indentation can result in logical or grammatical mistakes that might cause problems with network setups or automation.

Improper indentation in structured configuration files (such as YAML or JSON) can result in misconfigurations, which can compromise security or cause network problems.

- *Structured Organization*

By illustrating the relationships between various script and configuration elements, indentation aids in the visual grouping of related commands and settings.

Correct indentation distinguishes parent-child links in complex setups (such as firewall rules and network settings), which facilitates understanding of the overall structure.

- *Facilitates Collaboration*

By guaranteeing that everyone can read and edit each other's work without confusion, maintaining standardized indentation encourages collaboration in settings where numerous team members are involved.

During collaborative development is much easier with indented code since it is easy to compare

- *Maintenance and Debugging*

Updating and maintaining code that is well structured and indented is easier. In network systems where configurations may change often, this is especially crucial.

Administrators can detect and resolve problems more quickly when there is obvious indentation, which lowers downtime and boosts system reliability.

In a Linux OS network context, using appropriate indentation as a best practice not only improves readability and helps eliminate errors, but it also fosters greater teamwork, upkeep, and debugging. These elements support a safer and more effective network administration procedure.