Sri Lanka Institute of Information Technology

B.Sc. (Hons) in Information Technology Specialized in Cyber Security

IE2012 – Systems and Network Programming

Individual Assignment

IT23623972-T.M.G.H Thennakoon

Contents

- **What is the Linux Environment?**

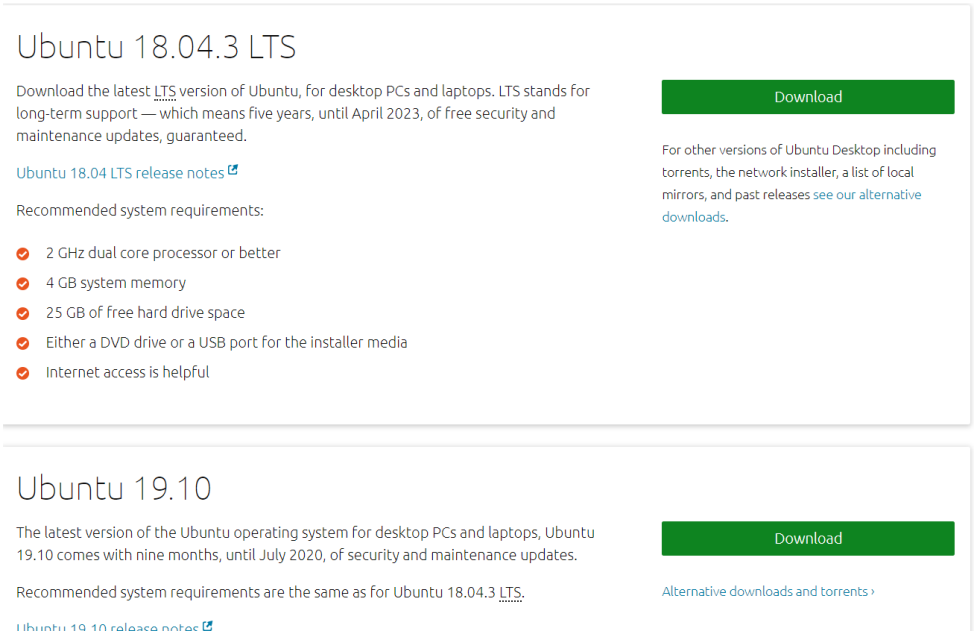
Linux is a family of open-source Unix-like operating systems based on the Linux kernel, which was first released by Linus Torvalds in 1991. What started as a hobby project has become a powerful and versatile OS that powers everything from smartphones (Android is built on the Linux kernel!) and servers to embedded systems and supercomputers.

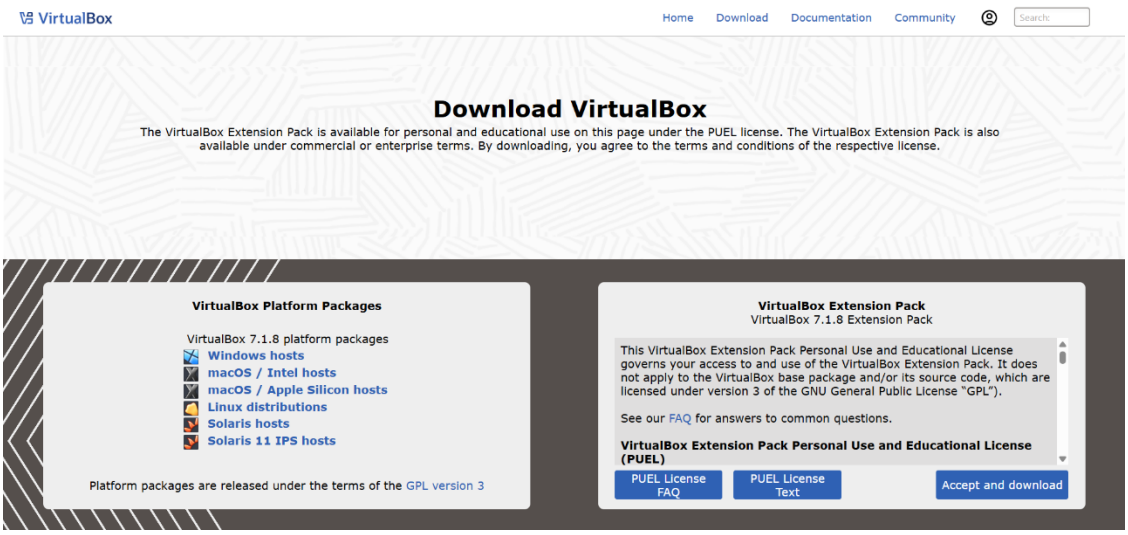
There are some Linux-based operating systems:

- Ubuntu (in this case, we are using this OS to learn the Linux environment)
- Linux-mint
- Kali-Linux
- Android

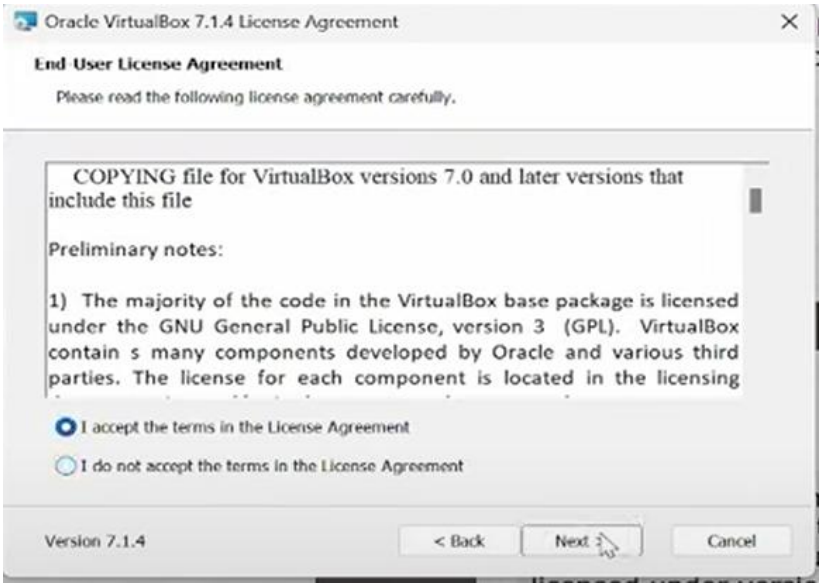
- **Virtual Machine installation**

1. Before beginning the installation process, we need to [download the ISO for Ubuntu](#) and [Oracle VirtualBox](#).

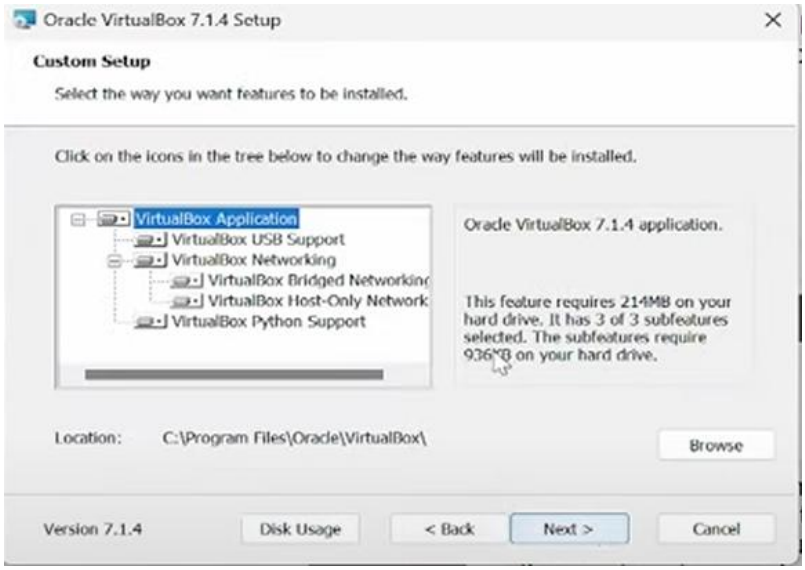




2. Next, install Oracle VirtualBox by double-clicking on the downloaded setup exe file and click next, accept the License Agreement, again click next again.



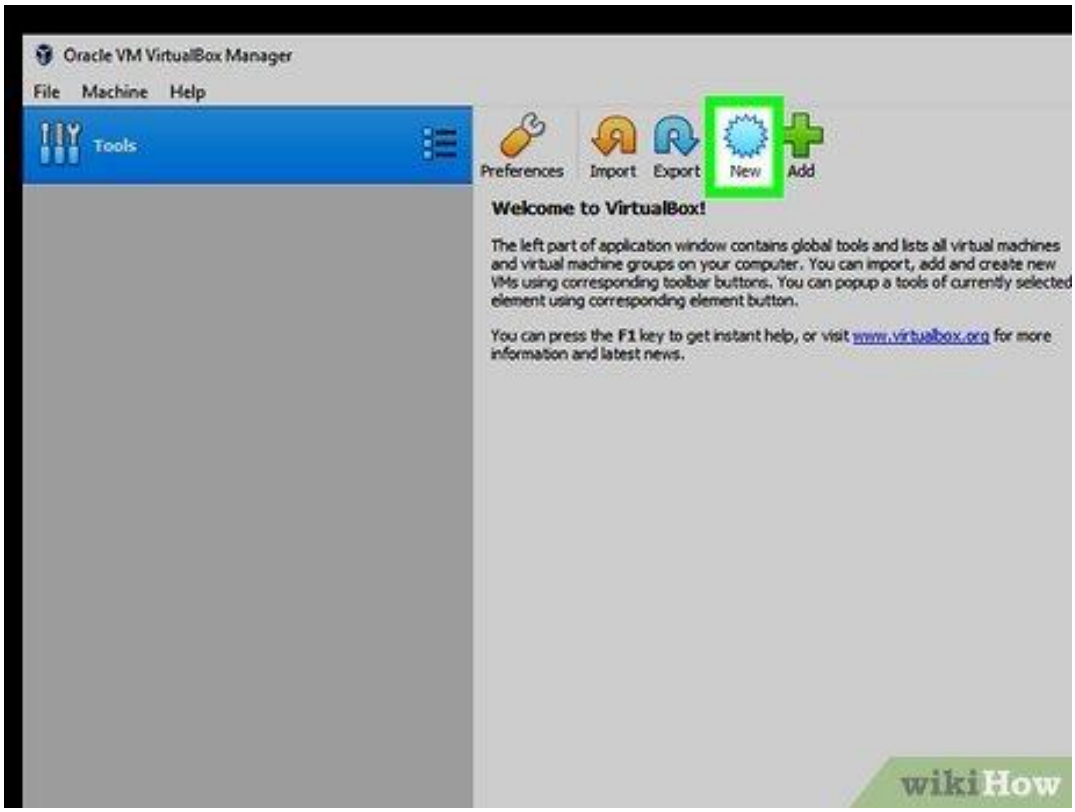
3. Click on next (do not change anything)



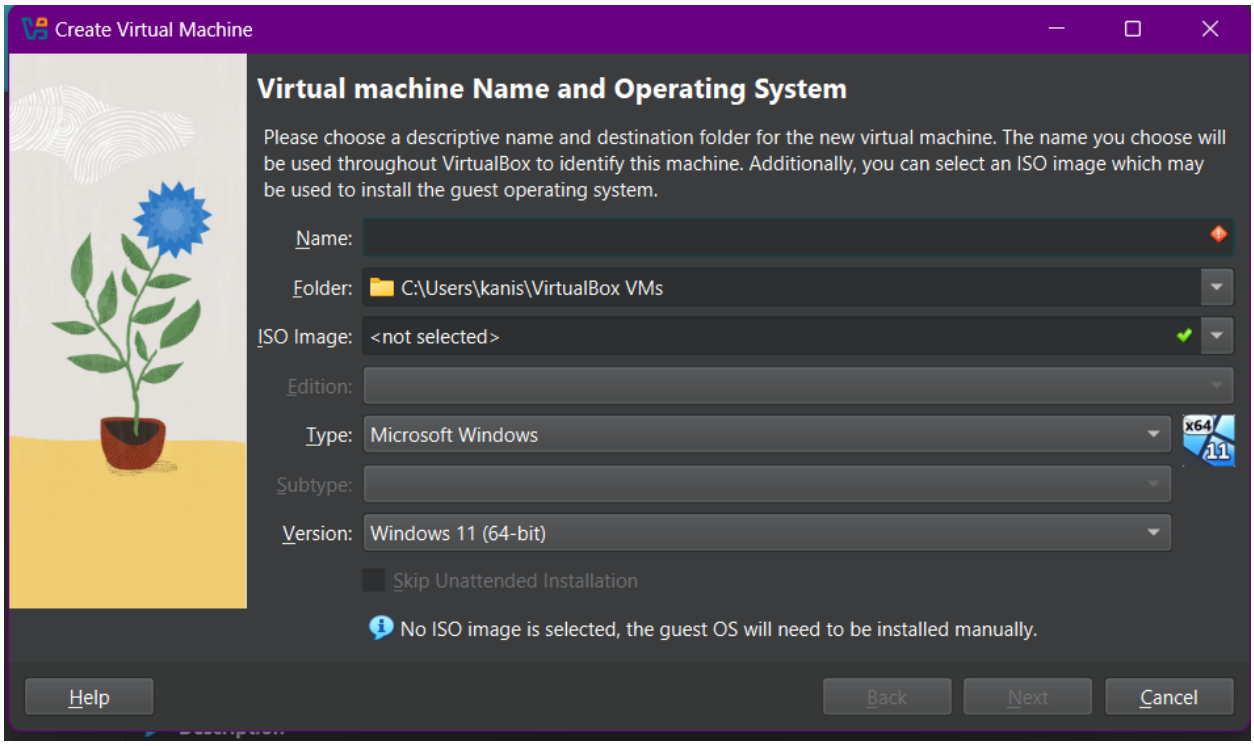
4. Click yes. Then it will begin the installation.

Waiting for complete installation

5. Open Oracle VirtualBox and click on New.



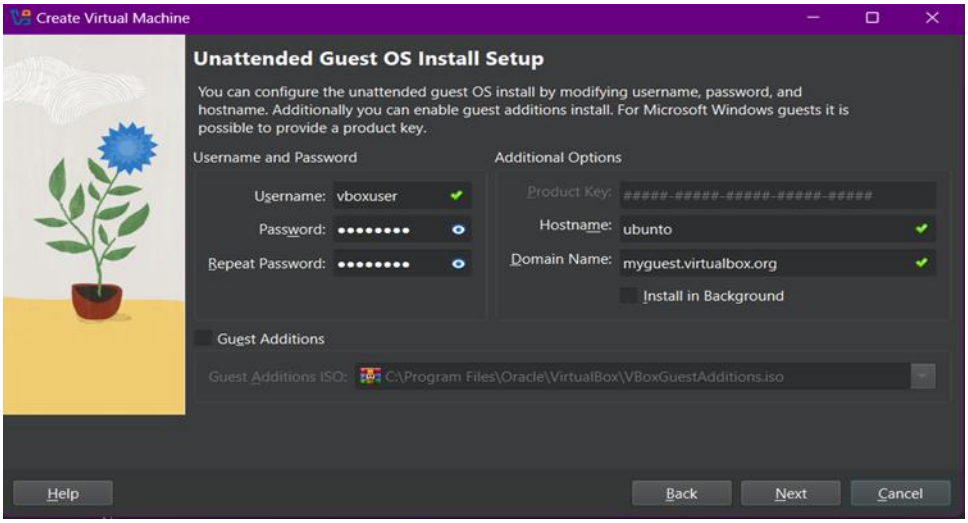
6. Then provide the details in the pop-up menu that opens.



- Name: Give a name for your operating system.
- Folder: Location where you want to install the OS.
- ISO Image: Browse the location of the ISO file that was downloaded (Ubuntu ISO file).

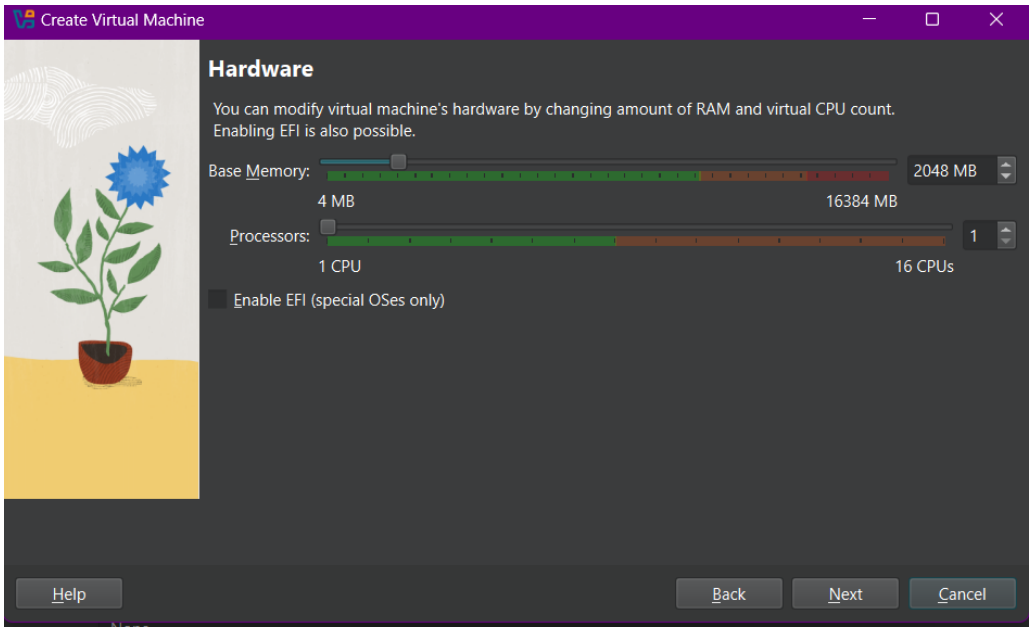
Then click the “Next” button.

7. Add username and password.



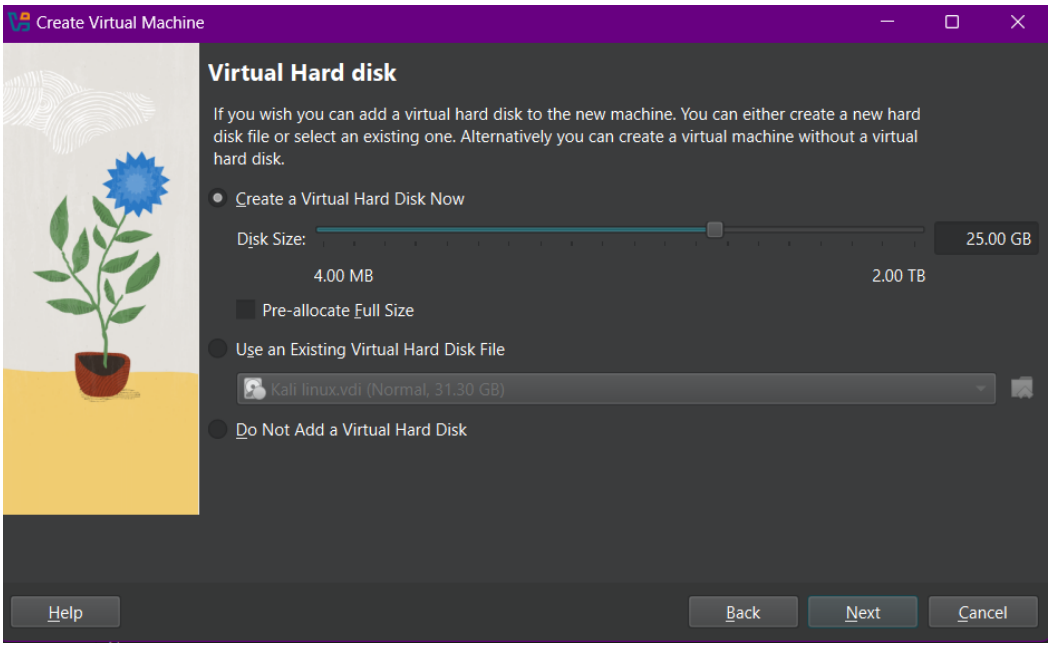
Then click the “Next” button. (Don’t change Additional Options)

8. Choose hardware performance.

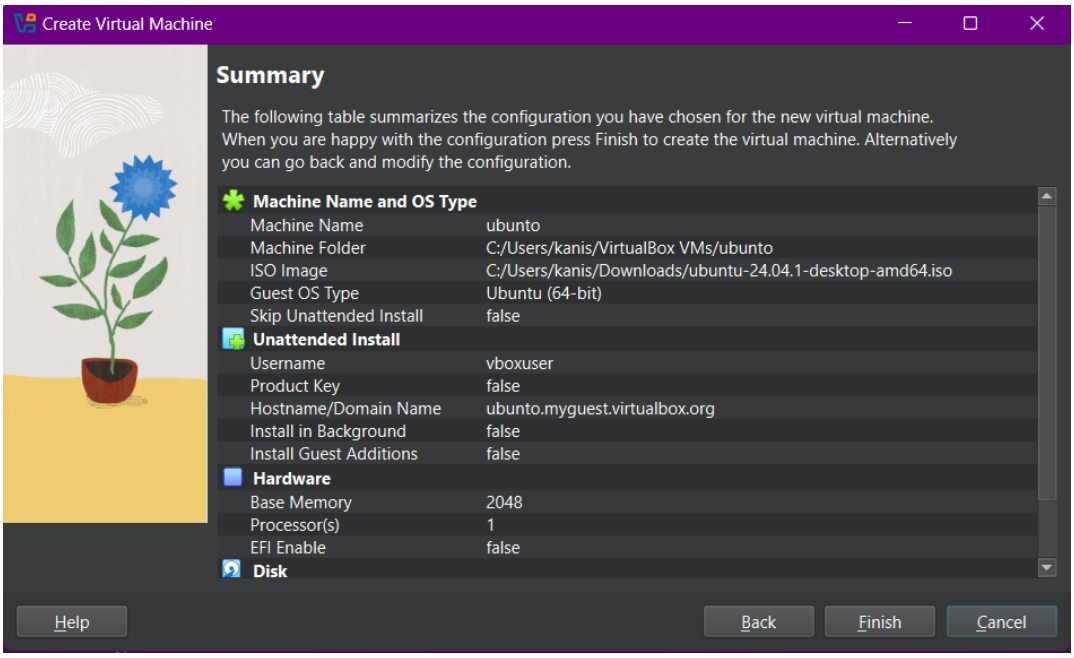


Choose hardware performance as you prefer (recommended to select performance between the green area)and click on next.

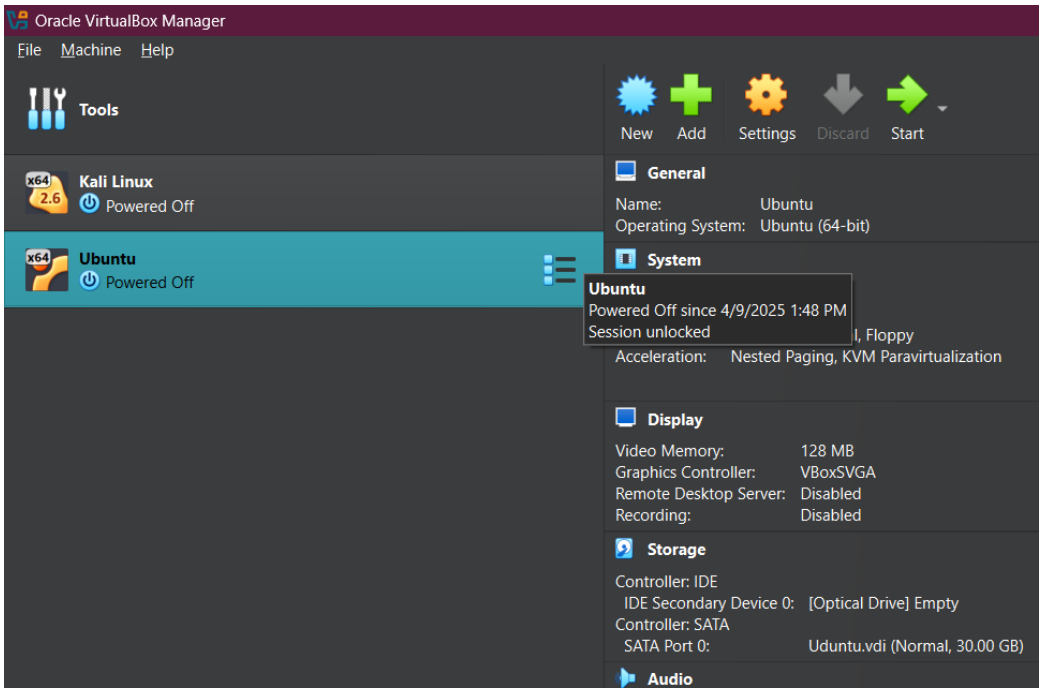
9. Give disk space size for the virtual machine.



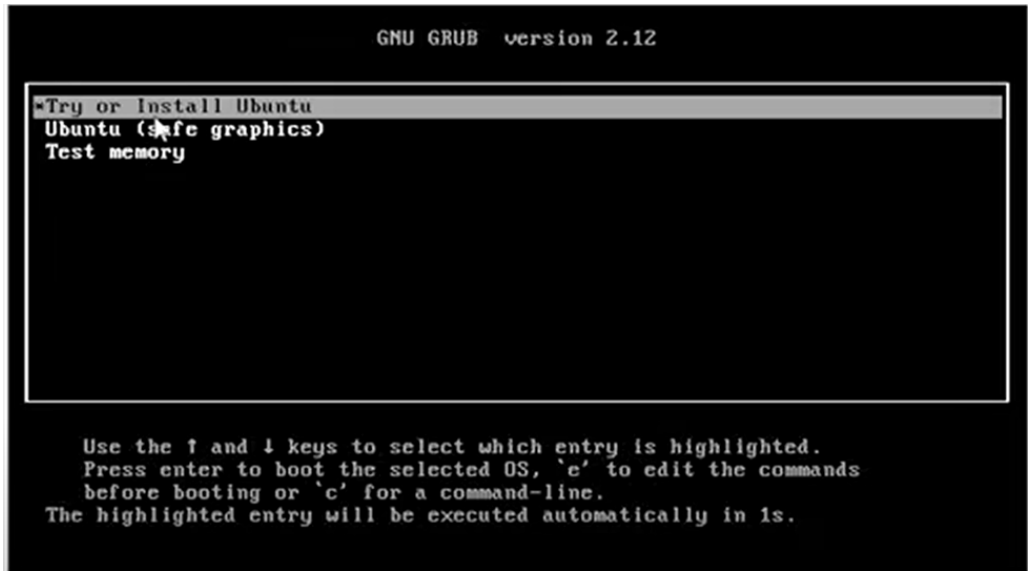
10. Then click on next and click finish button.



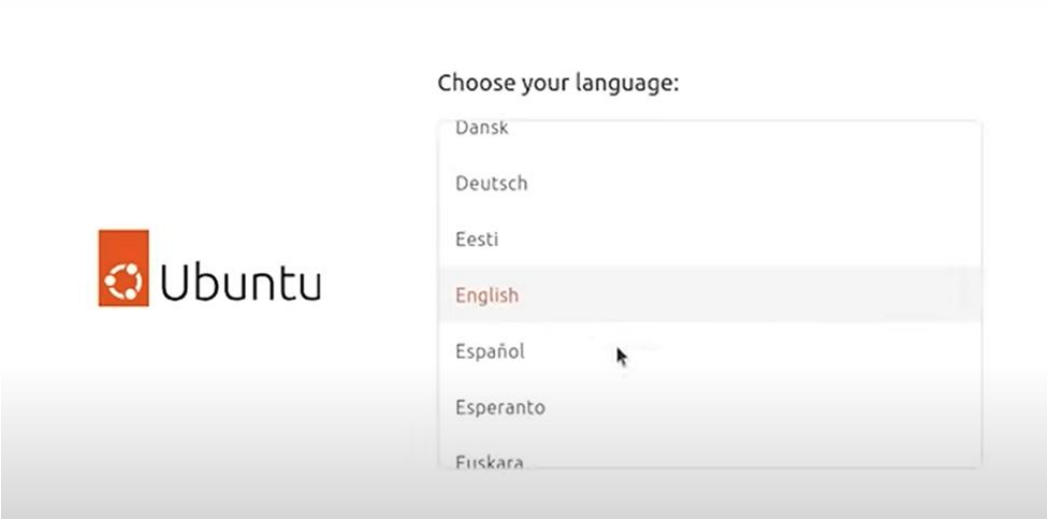
11. The Created Virtual machine will automatically open. (If not open the VM).



12. Click “Try or Install Ubuntu”.



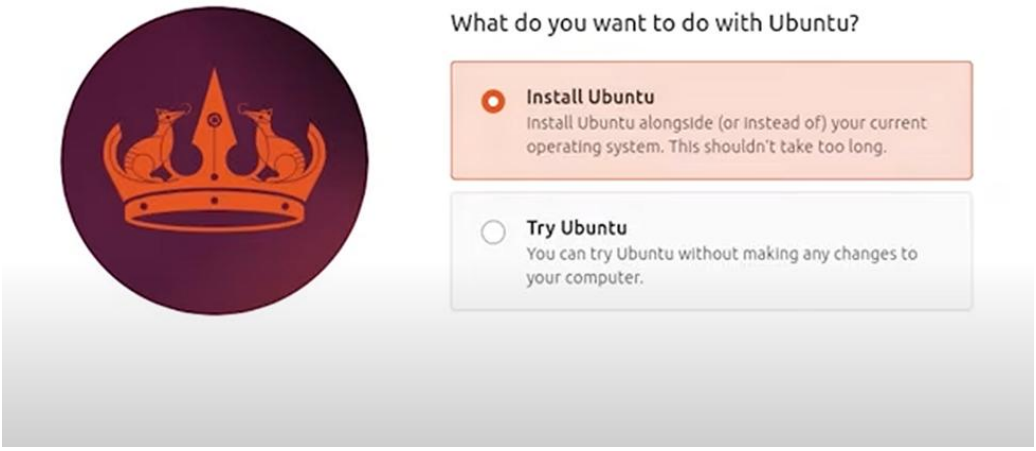
13. Choose a language.



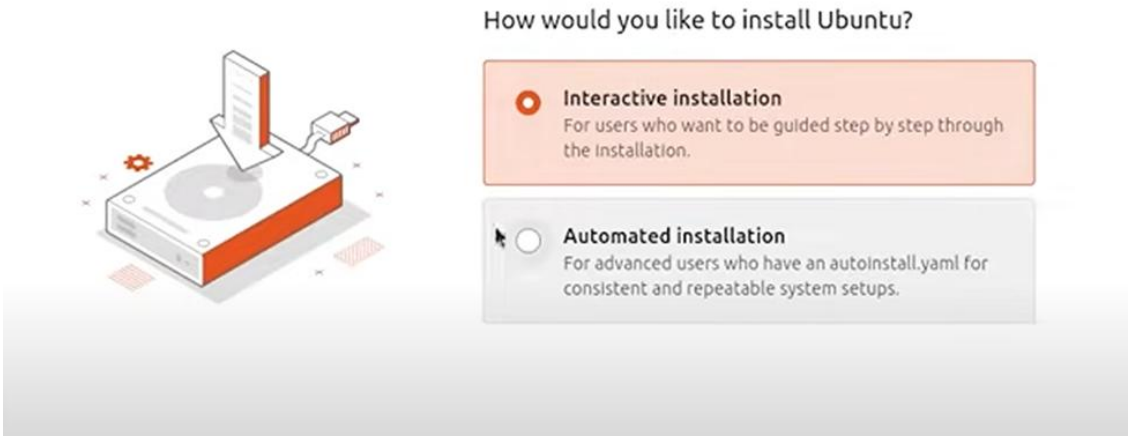
Give the necessary options that you prefer and set up Ubuntu.

14. Choose “Use wired connection”.

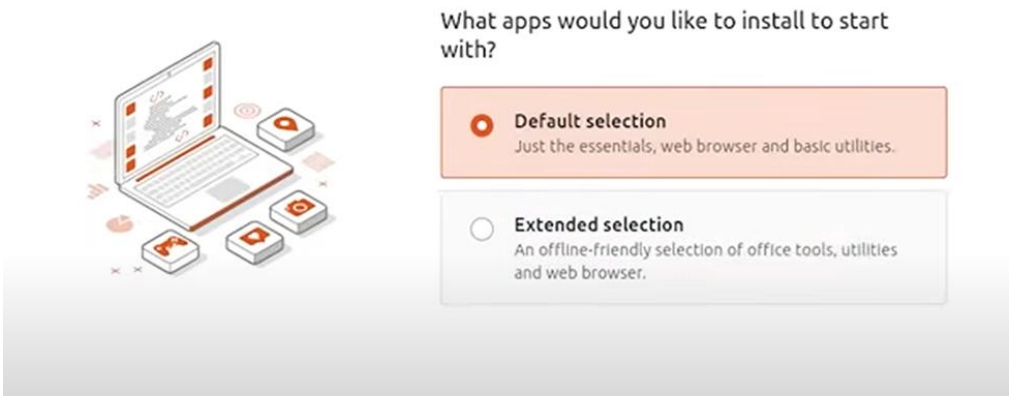
15. Select “Install Ubuntu”.



16. Select “Interactive installation”.

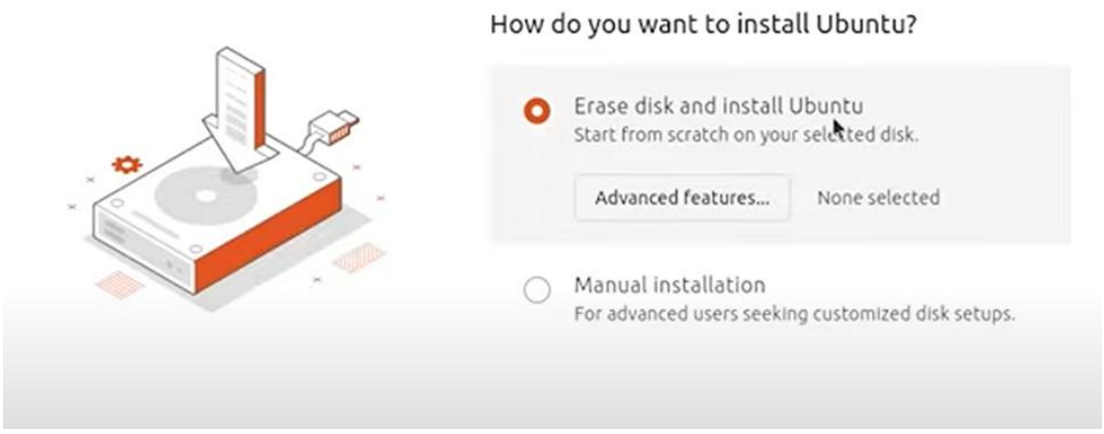


17. Select "Default Selection”.



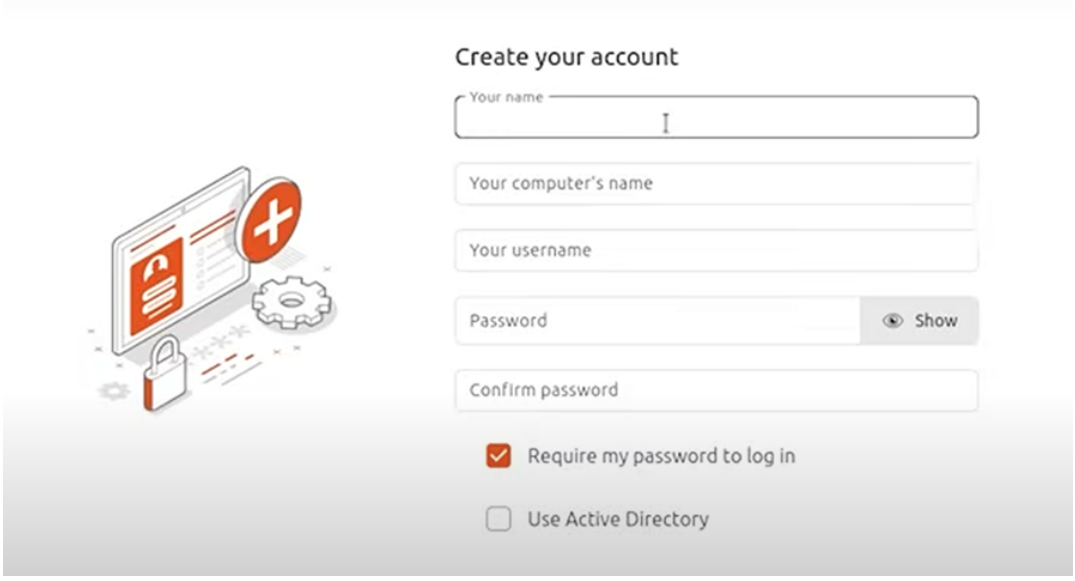
18. Select “Install third-party software for graphics and Wi-Fi hardware”.

19. Select “Erase disk and install Ubuntu”.

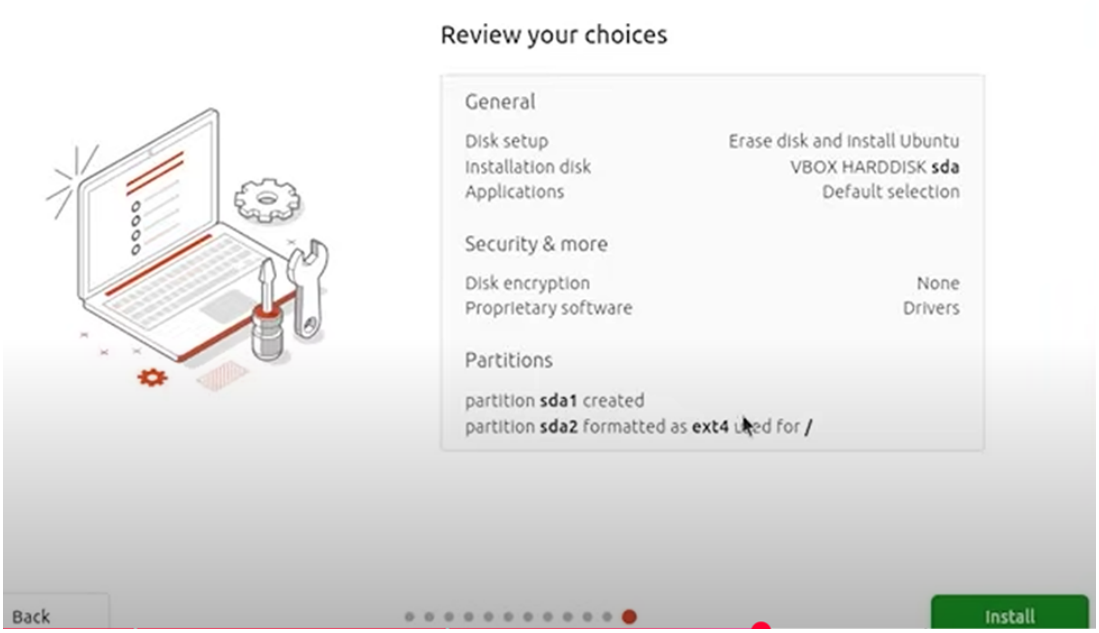


Erase disk won’t perform any deletion.

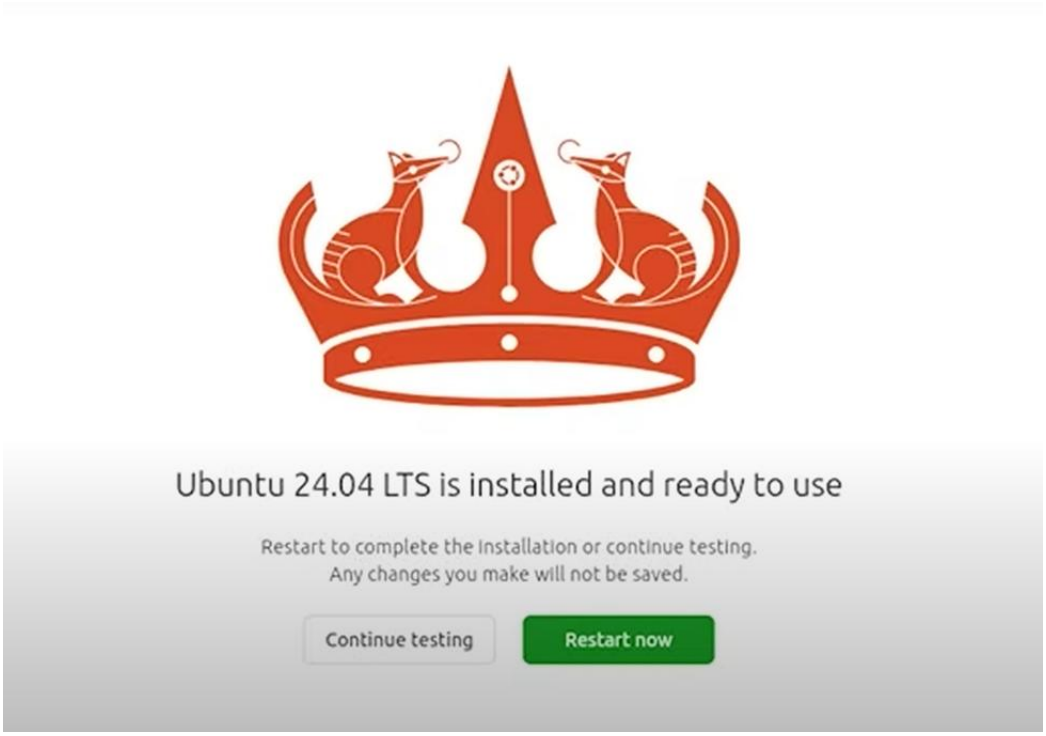
20. Add details.



21. Click Install.



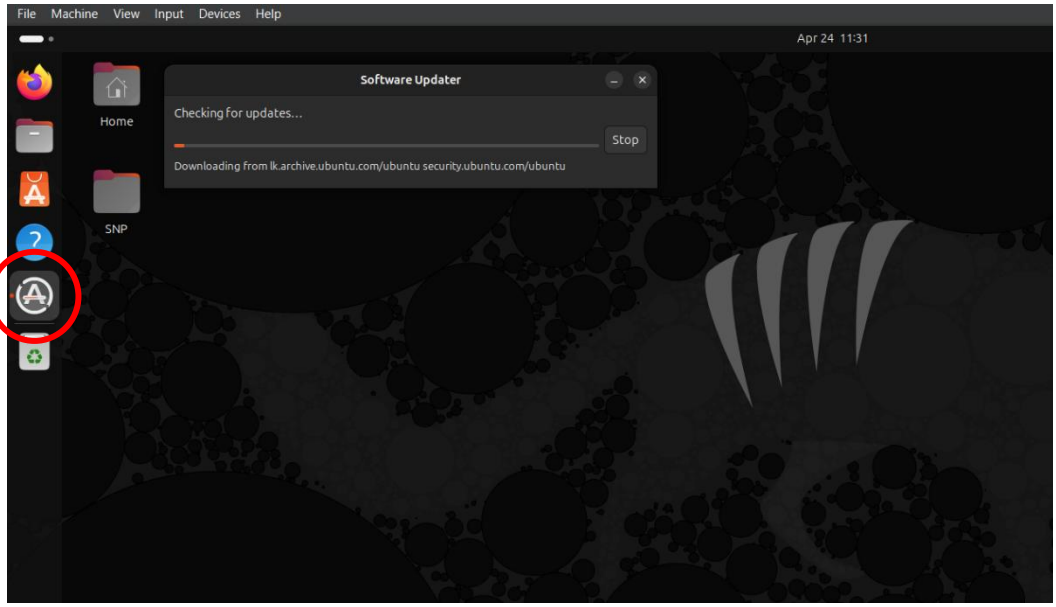
22. Enjoy the new virtual machine after clicking “Restart now”.



23. Update the system.

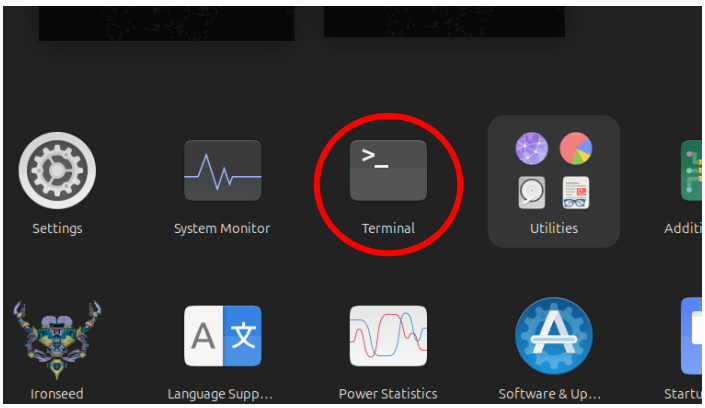
a. Method 1

- To update the system, find the Software updater application and click on it.



b. Method 2

- Open your terminal and run this code.
  - `sudo apt update && sudo apt upgrade -y`



• Basic Linux commands

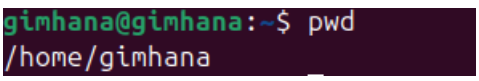
- **Navigation commands:**

  - pwd
  - cd
  - cd .. - Moves up one directory level (to the parent directory)
  - cd - - Switches to the previous directory.
  - ls
  - ls -l - Lists files/directories with detailed information (permissions, size, date, etc.).
  - ls -a - Lists all files, including hidden files (those starting with a dot).
  - mkdir
  - rmdir

- **File Manipulation Commands:**

  - touch
  - cp
  - cp -r - Copies a directory and its contents recursively.
  - mv
  - rm
  - rm -r - Deletes a directory and its contents recursively.
  - cat
  - nano
  - less

- **`pwd` (print working directory):** Displays the full path of the directory you are currently in.



- **`cd` (change directory):** Allows you to navigate to a different directory in the file system.

```
gimhana@gimhana:~$ cd Documents
gimhana@gimhana:~/Documents$
```

- **`ls` (list):** Shows the files and directories within your current directory.

```
gimhana@gimhana:~$ ls
02.c      Documents  ex2    lab      lab3.c     lab4ex2.c  lab5ex2.c  labex2  new1     newlab.c  pg1.c     snap          Student.text  usrinfo.sh
03.c      Downloads  ex3    lab07    lab4ex1.c  lab5       lab5ex3.c  labex3  new1.c   oddeven   Pictures  student1.txt  Sublist3r     Videos
Desktop   ex1        ex5.c  lab07.c  lab4ex2    lab5.c     labex1     Music   new1.sh  pg1       Public    student2      Templates
```

- **`mkdir` (make directory):** Creates a new directory with the specified name.

```
gimhana@gimhana:~/Documents$ mkdir Example
```

- **`rmdir` (remove directory):** Deletes an empty directory.

```
gimhana@gimhana:~/Documents$ rmdir Example
```

- **`rm` (remove):** This command deletes files and directories. Use with caution, especially with the—r (recursive) and—`f` (force) options.

```
gimhana@gimhana:~/Documents$ rm example.txt
```

- **`cp` (copy):** Copies files and directories from a source to a destination. Use `-r` for recursive directory copying.

```
gimhana@gimhana:~/Documents$ cp file1.txt file2.txt
```

- **`mv` (move):** Moves files and directories, or renames them.

```
gimhana@gimhana:~/Documents$ mv old.txt new.txt
gimhana@gimhana:~/Documents$  mv file.txt Downloads/
```

- **`touch`:** Creates an empty file or updates the timestamp of an existing file.

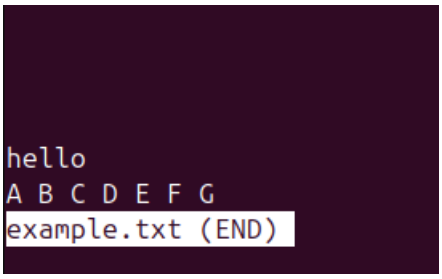
```
gimhana@gimhana:~/Documents$ touch example.txt
```

- **`cat` (concatenate):** Displays the content of one or more files on the terminal.

```
gimhana@gimhana:~/Documents$ cat example.txt
hello
A B C D E F G
```

- **`less`:** Displays file content page by page, allowing you to navigate through large files.

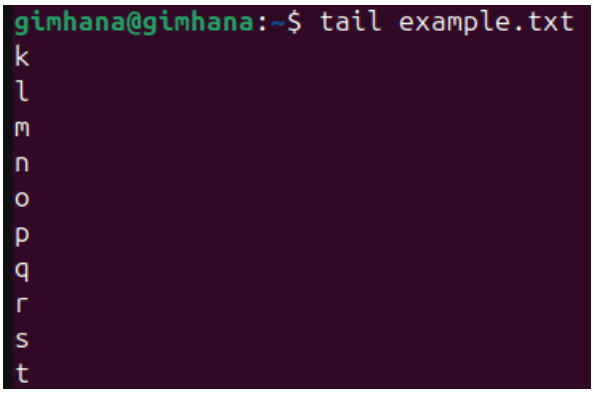
```
gimhana@gimhana:~/Documents$ less example.txt
```



- **`head`:** Shows the first few lines (default is 10) of a file.

```
gimhana@gimhana:~$ head example.txt
a
b
c
d
e
f
g
h
i
g
```

- **`tail`:** Shows the last few lines (default is 10) of a file. Useful for monitoring logs.



- **`man` (manual):** Displays the manual page for a given command, providing detailed information about its usage and options. For example, `man ls`.

- **`clear`:** Clears the terminal screen.

- # DHCP

What is DHCP?

The Dynamic Host Configuration Protocol (DHCP) is a network management protocol used on Internet Protocol (IP) networks to automatically assign IP addresses and other communication parameters to devices connected to the network. Think of it as an automated system that hands out network addresses so you don't have to configure each device manually.

How to install DHCP?

1. Frist update your os
    - Open terminal and run this code
        - `sudo apt update && sudo apt upgrade -y`

```
gimhana@gimhana:~$ sudo apt update && sudo apt upgrade -y
```

2. Install the DHCP Server Package. Run this command on your terminal
        - `sudo apt install isc-dhcp-server`

```
gimhana@gimhana:~$ sudo apt install isc-dhcp-server
```

3. After this step open new terminal and check your network interface
    - Run this command to check network interface
        - `ip a`

```
gimhana@gimhana:~$ ip a
```

    - Find your Ethernet card interface

```
gimhana@gimhana:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:0a:0c:0a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85726sec preferred_lft 85726sec
    inet6 fd00::6d5a:ab78:d9ba:4e42/64 scope global temporary dynamic
       valid_lft 86236sec preferred_lft 14236sec
    inet6 fd00::a00:27ff:fe0a:c0a/64 scope global dynamic mngtmpaddr
       valid_lft 86236sec preferred_lft 14236sec
    inet6 fe80::a00:27ff:fe0a:c0a/64 scope link
       valid_lft forever preferred_lft forever
```

4. Configure the DHCP Server:
    - Edit the file /`etc/default/isc-dhcp-server` to define which network interface the DHCP server should listen on. Replace eth0 (or enp0s8, etc.) with your actual interface name.
    - Run this code : `sudo nano /etc/default/isc-dhcp-server`

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

    - Edit “IN TERFACESv4=” writing your Ethernet card interface

```
#       Separate mul
INTERFACESv4=""
INTERFACESv6=""
```

```
 GNU nano 7.2
# 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=""
```

- Click CTRL+O , Enter to save it and click CTRL+X to exit.

5. Edit the DHCP Configuration File .
   - Edit `/etc/dhcp/dhcpd.conf` to define the IP address range and other DHCP options
   - Run this code : `sudo nano /etc/dhcp/dhcpd.conf`

```
gimhana@gimhana:~$ sudo nano /etc/dhcp/dhcpd.conf
```

   - Add a configuration similar to this, adjusting values for your network .

```
  GNU nano 7.2                                                    /etc/dhc
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# Attention: If /etc/ltsp/dhcpd.conf exists, that will be used as
# configuration file instead of this file.
#

# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 600;
max-lease-time 7200;

# The ddns-updates-style parameter controls whether or not the server will
# attempt to do a DNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn't
# have support for DDNS.)
ddns-update-style none;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
#authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
#log-facility local7;

# No service will be given on this subnet, but declaring it helps the
# DHCP server to understand the network topology.
```

- Frist give a Domain name

```
option domain-name "example.org";  →  option domain-name "gimhana.local";
```

- Change domain name server

```
option domain-name-servers ns1.example.org, ns2.example.org;
```

↓

```
option domain-name-servers 1.1.1.1, 8.8.8.8;
```

- Uncomment this removing “#”.

```
#authoritative;  →  authoritative;
```

- Uncomment and add your IP details.

```
# A slightly different configuration for an internal subnet.
#subnet 10.5.5.0 netmask 255.255.255.224 {
#  range 10.5.5.26 10.5.5.30;
#  option domain-name-servers ns1.internal.example.org;
#  option domain-name "internal.example.org";
#  option subnet-mask 255.255.255.224;
#  option routers 10.5.5.1;
#  option broadcast-address 10.5.5.31;
#  default-lease-time 600;
#  max-lease-time 7200;
#}
```

↓

```
# A slightly different configuration for an internal subnet.
subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.50 192.168.1.100;
    option domain-name-servers 8.8.8.8;
    option domain-name "gimhana.local";
    option subnet-mask 255.255.255.0;
    option routers 192.168.1.1;
    option broadcast-address 192.168.1.255;
    default-lease-time 600;
    max-lease-time 7200;
}
```

- Click CTRL+O , Enter to save it and click CTRL+X to exit.

6. Start and Enable the DHCP Server
   - Restart DHCP Server
     - `sudo systemctl restart isc-dhcp-server`

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

   - Start the DHCP service and enable it to run at boot, run this code
     - `sudo systemctl start isc-dhcp-server`
     - `sudo systemctl enable isc-dhcp-server`

   - Check status your DHCP Server to verify that the DHCP server is running
     - `sudo systemctl status isc-dhcp-server`

```
gimhana@gimhana:~$ 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 2025-04-25 01:01:21 +0530; 16s ago
       Docs: man:dhcpd(8)
   Main PID: 4539 (dhcpd)
      Tasks: 1 (limit: 10534)
     Memory: 3.7M (peak: 4.0M)
        CPU: 15ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─4539 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dhcp/dhcpd.conf enp0s3

Apr 25 01:01:21 gimhana sh[4539]: Wrote 0 leases to leases file.
Apr 25 01:01:21 gimhana dhcpd[4539]: PID file: /run/dhcp-server/dhcpd.pid
Apr 25 01:01:21 gimhana dhcpd[4539]: Wrote 0 leases to leases file.
Apr 25 01:01:21 gimhana dhcpd[4539]: Listening on LPF/enp0s3/08:00:27:0a:0c:0a/10.0.2.0/24
Apr 25 01:01:21 gimhana sh[4539]: Listening on LPF/enp0s3/08:00:27:0a:0c:0a/10.0.2.0/24
Apr 25 01:01:21 gimhana sh[4539]: Sending on   LPF/enp0s3/08:00:27:0a:0c:0a/10.0.2.0/24
Apr 25 01:01:21 gimhana sh[4539]: Sending on   Socket/fallback/fallback-net
Apr 25 01:01:21 gimhana dhcpd[4539]: Sending on   LPF/enp0s3/08:00:27:0a:0c:0a/10.0.2.0/24
Apr 25 01:01:21 gimhana dhcpd[4539]: Sending on   Socket/fallback/fallback-net
Apr 25 01:01:21 gimhana dhcpd[4539]: Server starting service.
```

7. Let's try DHCP server.
   - Create Client machine.
     - Create new virtual machine or clone our virtual machine.
   - Open terminal your client machice.
     - Install the Internet Systems Consortium (ISC) DHCP client

```
sudo apt install isc-dhcp-client
```

     - Renewing DHCP lease

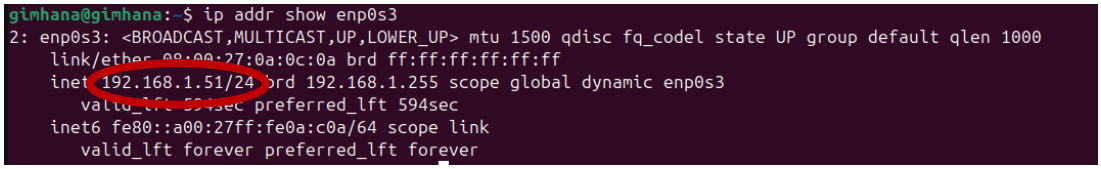
```
nmcli connection down enp0s3 && nmcli connection up enp0s3
```

- Force Your Client PC to Obtain a New IP Address Again

```
sudo dhclient -r enp0s3
sudo dhclient -v enp0s3
```

- Check the Client's IP Address

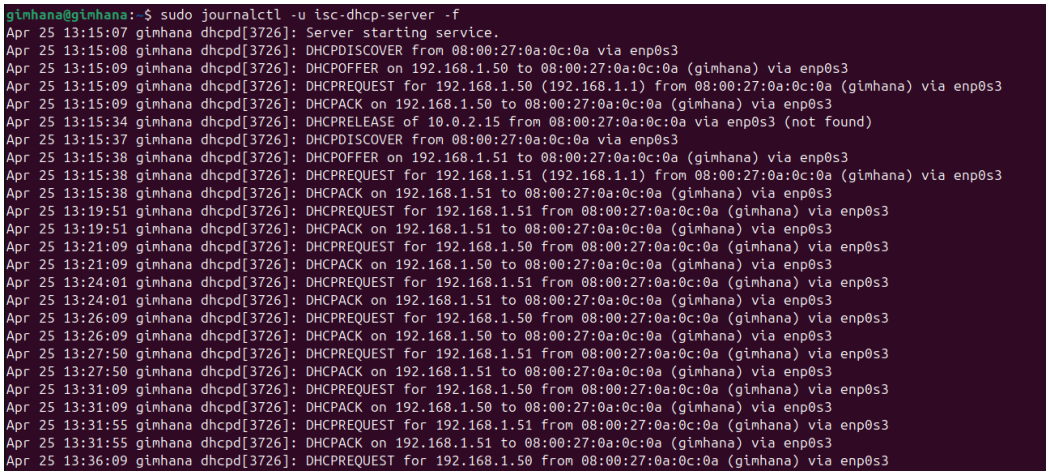
```
ip addr show enp0s3
```



you should see if it receives an IP address from the range you configured on your Ubuntu DHCP server .

- Monitor Your Ubuntu DHCP Server Logs
  - On your Ubuntu server, run this code

```
sudo journalctl -u isc-dhcp-server -f
```



Watch this output while the client PC is trying to get an IP address. You should see the client's DHCPDISCOVER request coming in and the Ubuntu server's DHCPOFFER, DHCPREQUEST, and DHCPACK responses.

- # DNS

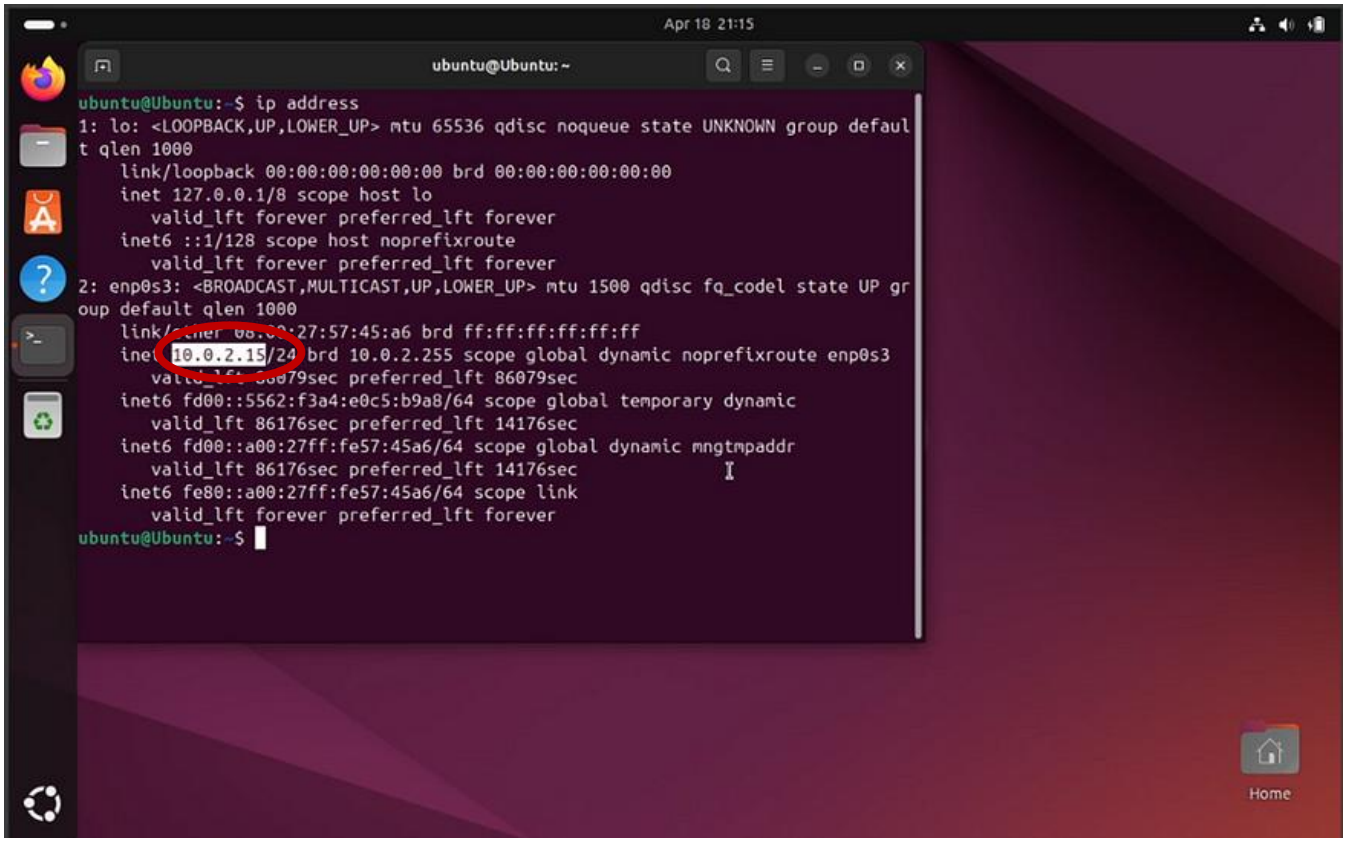
- ## What is DNS ?

DNS, the "internet's phonebook," translates human-friendly domain names (like google.com) into numerical IP addresses (like 172.217.160.142) that computers use to locate websites. This crucial system enables seamless web browsing without memorizing complex numbers, making the internet user-friendly.

- ## DNS installation

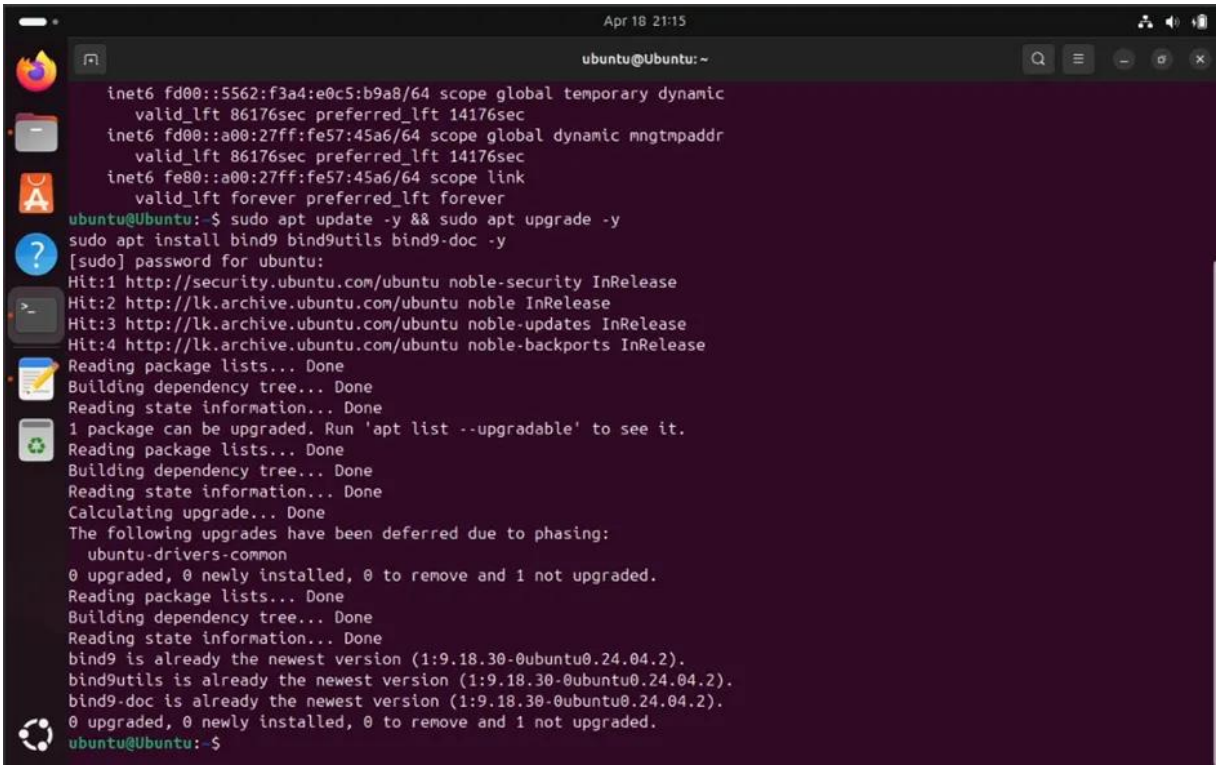
Setting up your own DNS server with BIND on Ubuntu.

1. First Check the IP address. Using this command : `ip address`



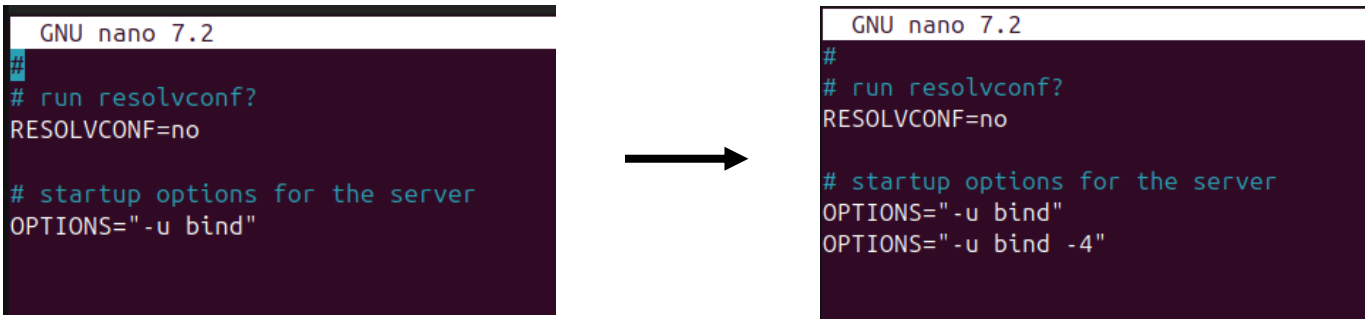
2. Secondly update and install BIND9 Using this command.

- `sudo apt update -y && sudo apt upgrade -y`
- `sudo apt install bind9 bind9utils bind9-doc -y`

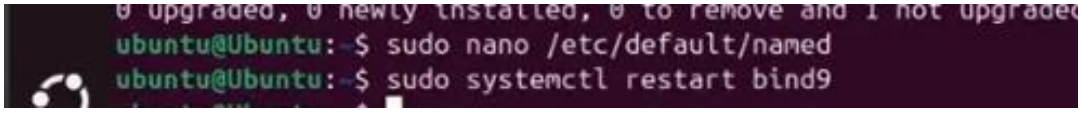


3. Now configure BIND to use IPv4. Open the file `/etc/default/named` using the Nano editor. Now modify the line in the file.

- `sudo nano /etc/default/named`



4. Now restart the BIND using this command : `sudo systemctl restart bind9` and Restart OS.



5. Now configure BIND options using this command :
    - `sudo nano /etc/bind/named.conf.options`

Now replace the option block with using these commands.

- `directory "/var/cache/bind";`
    - This sets the working directory for BIND.
    - It's where zone files, cache, and runtime data are stored.
- `recursion yes;`
    - Enables recursive DNS queries.
    - This allows BIND to resolve queries by asking other DNS servers on behalf of clients.
- `allow-query {localhost; 10.0.2.0/24;};`
    - Limits which IPs are allowed to make DNS queries.
    - Only localhost (127.0.0.1) and devices on the subnet 10.0.2.0/24 can query this DNS server.
- `listen-on {10.0.2.15; 127.0.0.1; };`
    - Specifies which IPv4 addresses BIND listens on for DNS requests.
    - In this case, it listens on the local loopback and a specific IP 10.0.2.15.
- `listen-on-v6 {none; };`
    - Disables IPv6 listening. The server won't respond to IPv6 DNS queries.
- `dnssec-validation auto;`
    - Enables DNSSEC (DNS Security Extensions) validation.
    - auto lets BIND manage the trusted key automatically, improving DNS trust and integrity.
    - Now set up DNS Zones

```
 GNU nano 7.2                                    /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";
    recursion yes;
    allow-query { localhost; 10.0.2.0/24; };
    listen-on { 10.0.2.15; 127.0.0.1; };
    listen-on-v6 { none; };
    dnssec-validation auto;
};
```

6. Assume my domain is example.com and edit my local zones file using this command
    - `sudo nano /etc/bind/named.conf.local`

Now edit the forward and reverse zones.

```
zone "example.com" {
type master;
file "/etc/bind/db.example.com";
};
zone "2.0.10.in-addr.arpa" {
type master;
file "/etc/bind/db.10.0.2";
};
```

```
 GNU nano 7.2                                    /etc/bind/named.conf.l
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "example.com" {
    type master;
    file "/etc/bind/db.example.com";
};

zone "2.0.10.in-addr.arpa" {
    type master;
    file "/etc/bind/db.10.0.2";
};
```

7. Now create the forward zone files.

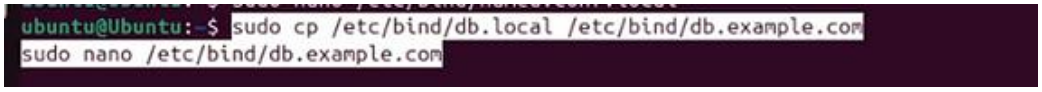
Firstly coping the exiting db.local file as a starting point to a new zone file for db.example.local .

The coping command is :

```
sudo cp /etc/bind/db.local/etc/bind/db.example.com
```

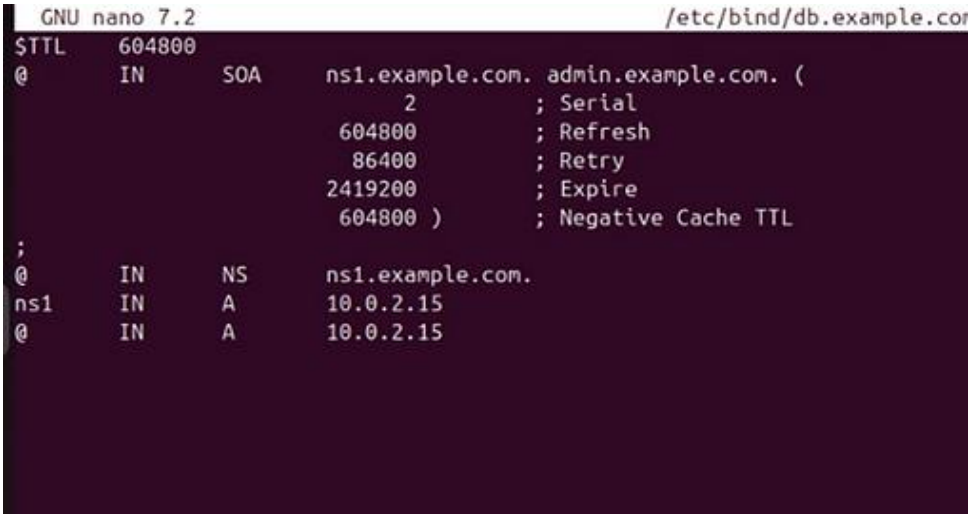
Now edit the new copy file using this command

```
sudo nano /etc/bind/db.example.com
```



Now edit the opened zone file using this command :

```
$TTL    604800

@       IN      SOA     ns1.example.com. admin.example.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.example.com.
ns1     IN      A       10.0.2.15
@       IN      A       10.0.2.15
```



8. Now create the reverse zone files.

Firstly coping the exiting db.127 file as a starting point to a new zone file for db.10.0.2 .

- The coping command is:
  `sudo cp /etc/bind/db.127 /etc/bind/db.10.0.2`
- Now edit the new copy file using this command:
  `sudo nano /etc/bind/db.10.0.2`

Now, edit the opened zone file using this command

```
$TTL    86400
@       IN      SOA     ns1.example.com. admin.example.com. (
                        2025042501  ; Serial
                        3600        ; Refresh
                        900         ; Retry
                        604800      ; Expire
                        86400       ; Minimum TTL
                        )
@       IN      NS      ns1.example.com.
15      IN      PTR     ns1.example.com.  ; Maps 10.0.2.15 to ns1.example.com
```

```
  GNU nano 7.2
$TTL    86400
@       IN      SOA     ns1.example.com. admin.example.com. (
                        2025042501  ; Serial
                        3600        ; Refresh
                        900         ; Retry
                        604800      ; Expire
                        86400       ; Minimum TTL
                        )
@       IN      NS      ns1.example.com.
15      IN      PTR     ns1.example.com.  ; Maps 10.0.2.15 to ns1.example.com
```

9. Now check configuration and restart. The commands are,

```
sudo named-checkconf
sudo named-checkzone example.com /etc/bind/db.example.com
sudo named-checkzone 2.0.10.in-addr.arpa /etc/bind/db.10.0.2
sudo systemctl restart bind9
```

```
ubuntu@Ubuntu:~$ sudo named-checkconf
sudo named-checkzone example.com /etc/bind/db.example.com
sudo named-checkzone 2.0.10.in-addr.arpa /etc/bind/db.10.0.2
sudo systemctl restart bind9
zone example.com/IN: loaded serial 2
OK
zone 2.0.10.in-addr.arpa/IN: loaded serial 1
OK
ubuntu@Ubuntu:~$
```

10. Now test the DNS server using this commands

```
nslookup example.com 10.0.2.15
nslookup 10.0.2.15 10.0.2.15
```

```
ubuntu@Ubuntu:~$ nslookup example.com 10.0.2.15
nslookup 10.0.2.15 10.0.2.15
Server:         10.0.2.15
Address:        10.0.2.15#53

Name:   example.com
Address: 10.0.2.15

15.2.0.10.in-addr.arpa  name = ns1.example.com.

ubuntu@Ubuntu:~$
```

- **NTP**

Network Time Protocol (NTP) is a networking protocol used to synchronize the clocks of computers over packet-switched, variable-latency networks, such as the internet. Developed in the 1980s, NTP ensures participating systems are synchronized to within milliseconds of Coordinated Universal Time (UTC), using a hierarchical system of time sources and robust algorithms to mitigate network delays and maintain accuracy.

In this case we are using the Chrony package. Chrony is another Network Time Protocol (NTP) client and server, similar to ntp. It's often preferred for its faster synchronization and better performance in unstable network conditions.

1. Firstly, update the system and install chrony using this commands:
    - `sudo apt update`
    - `sudo apt install chrony`

```
gimhana@gimhana:~$ sudo apt install chrony
```

2. Now check the chrony status using this command
    - `sudo systemctl status chrony`

```
gimhana@gimhana:~$ sudo systemctl status chrony
● chrony.service - chrony, an NTP client/server
     Loaded: loaded (/usr/lib/systemd/system/chrony.service; enabled; preset: enabled)
     Active: active (running) since Sat 2025-04-26 09:45:26 +0530; 8s ago
       Docs: man:chronyd(8)
             man:chronyc(1)
             man:chrony.conf(5)
    Process: 22397 ExecStart=/usr/lib/systemd/scripts/chronyd-starter.sh $DAEMON_OPTS (code=exited, status=0/SUCCESS)
   Main PID: 22406 (chronyd)
      Tasks: 2 (limit: 10534)
     Memory: 1.4M (peak: 2.4M)
        CPU: 90ms
     CGroup: /system.slice/chrony.service
             ├─22406 /usr/sbin/chronyd -F 1
             └─22407 /usr/sbin/chronyd -F 1

Apr 26 09:45:26 gimhana systemd[1]: Starting chrony.service - chrony, an NTP client/server...
Apr 26 09:45:26 gimhana chronyd[22406]: chronyd version 4.5 starting (+CMDMON +NTP +REFCLOCK +RTC +PRIVDROP +SCFILTER +SIGND +ASYNCDNS +NTS +SECHASH +IPV6 -DEBUG)
Apr 26 09:45:26 gimhana chronyd[22406]: Loaded 0 symmetric keys
Apr 26 09:45:26 gimhana chronyd[22406]: Initial frequency -500.000 ppm
Apr 26 09:45:26 gimhana chronyd[22406]: Using right/UTC timezone to obtain leap second data
Apr 26 09:45:26 gimhana chronyd[22406]: Loaded seccomp filter (level 1)
Apr 26 09:45:26 gimhana systemd[1]: Started chrony.service - chrony, an NTP client/server.
Apr 26 09:45:34 gimhana chronyd[22406]: Selected source 185.125.190.56 (ntp.ubuntu.com)
Apr 26 09:45:34 gimhana chronyd[22406]: System clock TAI offset set to 37 seconds
Apr 26 09:45:35 gimhana chronyd[22406]: Selected source 222.165.180.134 (0.ubuntu.pool.ntp.org)
```

3. Now edit the chrony configuration.

Open this path `/etc/chrony/chrony.conf` using nano editor. The command is

- `sudo nano /etc/chrony/chrony.conf`

```
# Welcome to the chrony configuration file. See chrony.conf(5) for more
# information about usable directives.

# Include configuration files found in /etc/chrony/conf.d.
confdir /etc/chrony/conf.d

# This will use (up to):
# - 4 sources from ntp.ubuntu.com which some are ipv6 enabled
# - 2 sources from 2.ubuntu.pool.ntp.org which is ipv6 enabled as well
# - 1 source from [01].ubuntu.pool.ntp.org each (ipv4 only atm)
# This means by default, up to 6 dual-stack and up to 2 additional IPv4-only
# sources will be used.
# At the same time it retains some protection against one of the entries being
# down (compare to just using one of the lines). See (LP: #1754358) for the
# discussion.
#
# About using servers from the NTP Pool Project in general see (LP: #104525).
# Approved by Ubuntu Technical Board on 2011-02-08.
# See http://www.pool.ntp.org/join.html for more information.
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2

# Use time sources from DHCP.
sourcedir /run/chrony-dhcp

# Use NTP sources found in /etc/chrony/sources.d.
sourcedir /etc/chrony/sources.d

# This directive specify the location of the file containing ID/key pairs for
# NTP authentication.
keyfile /etc/chrony/chrony.keys
```

Now add the following line.

- `server pool.ntp.org iburst`

```
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2
```

```
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2
server pool.ntp.org iburst
```

then save and close nano editor.

4. Now restart the chrony using this command
    - `sudo systemctl restart chrony`

```
gimhana@gimhana:~$ sudo systemctl restart chrony
[sudo] password for gimhana:
```

5. Now check the synchronization status. Using these commands:
    - `chronyc tracking`
    - `chronyc sources`

```
gimhana@gimhana:~$ chronyc tracking
Reference ID    : DEA5B486 (ntp.sltidc.lk)
Stratum         : 3
Ref time (UTC)  : Sat Apr 26 05:28:16 2025
System time     : 0.001107899 seconds slow of NTP time
Last offset     : -0.000242567 seconds
RMS offset      : 0.047167484 seconds
Frequency       : 7.071 ppm fast
Residual freq   : +0.097 ppm
Skew            : 14.563 ppm
Root delay      : 0.078511447 seconds
Root dispersion : 0.004977691 seconds
Update interval : 64.6 seconds
Leap status     : Normal
gimhana@gimhana:~$ chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^- prod-ntp-3.ntp1.ps5.cano>     2   7   377    31    -12ms[  -12ms] +/-   98ms
^- prod-ntp-5.ntp4.ps5.cano>     2   7   377    30    +15ms[  +15ms] +/-  121ms
^+ prod-ntp-4.ntp1.ps5.cano>     2   7   377    32  +6699us[+7129us] +/-  109ms
^- alphyn.canonical.com          2   7   377    30  +7103us[+7534us] +/-  163ms
^+ time.cloudflare.com           3   7   377    96    +12ms[  +12ms] +/-  115ms
^* ntp.sltidc.lk                 2   6   377    30  -6814us[-6383us] +/-   41ms
^? time.cloudflare.com           0   8     0     -     +0ns[   +0ns] +/-    0ns
^? time.cloudflare.com           0   8     0     -     +0ns[   +0ns] +/-    0ns
^- time.cloudflare.com           3   7   377     8    +26ms[  +26ms] +/-  129ms
gimhana@gimhana:~$
```

6. Synchronizing with a Local NTP Server

Edit the chrony configuration file using this command

- `sudo nano /etc/chrony/chrony.conf`

Now replace the line. (`server 10.0.2.1 iburst`)

```
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2
server pool.ntp.org iburst
```

```
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2
server pool.ntp.org iburst
server 10.0.2.1 iburst
```

7. Allowing other machines to sync my ubuntu machine.

Edit the chrony configuration file using this command:

- `sudo nano /etc/chrony/chrony.conf`

Now add this line to allow my subnet. `(allow 10.0.2.0/24)`

```
pool ntp.ubuntu.com        iburst maxsources 4
pool 0.ubuntu.pool.ntp.org iburst maxsources 1
pool 1.ubuntu.pool.ntp.org iburst maxsources 1
pool 2.ubuntu.pool.ntp.org iburst maxsources 2
server pool.ntp.org iburst
server 10.0.2.1 iburst
allow 10.0.2.0/24
```

8. Now restart the chrony.
    - `sudo systemctl restart chrony`
9. Checking the network connectivity.

Test the NTP server reachability using this command. (`ping pool.ntp.org`)

```
gimhana@gimhana:~$ ping pool.ntp.org
PING pool.ntp.org (162.159.200.123) 56(84) bytes of data.
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=1 ttl=255 time=133 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=2 ttl=255 time=104 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=3 ttl=255 time=129 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=4 ttl=255 time=107 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=5 ttl=255 time=103 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=6 ttl=255 time=106 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=7 ttl=255 time=101 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=8 ttl=255 time=240 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=9 ttl=255 time=125 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=10 ttl=255 time=124 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=11 ttl=255 time=148 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=12 ttl=255 time=109 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=13 ttl=255 time=103 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=14 ttl=255 time=115 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=15 ttl=255 time=104 ms
64 bytes from time.cloudflare.com (162.159.200.123): icmp_seq=17 ttl=255 time=154 ms
```

- # Shell Scripting

What is Shell Scripting ?

Shell scripting is the process of writing a series of commands for a Unix/Linux shell (like Bash, Zsh, or Sh) to automate tasks. These scripts are plain text files that contain command-line instructions executed in sequence.

Uses of Shell Scripting:

- Automating system administration tasks (e.g., backups, updates).
- Running batch jobs.
- Setting up environments.
- Processing text files or logs.
- Interacting with other programs or scripts.

Basic shell scripting syntax

1. Shebang Line :

```
#!/bin/bash
```

- Always the **first line** of a script.
- Tells the OS to use the Bash shell.

2. Comment :

```
# This is a comment
```

- Lines starting with `#` are ignored by the shell.

3. Variables :

```
name= "Alice"

echo "Hello, $name"
```

- No space around `=`.
- Use `$name` to access the variable.

4. User Inputs

```
read name

echo "You entered: $name"
```

- `read` gets input from the user.

5. Printing Output

```
echo "Hello, World"
```

- `echo` is used to display text on the screen.

6. Common operators :
    a. `-eq`: equal
    b. `-ne`: not equal
    c. `-gt`: greater than
    d. `-lt`: less than
    e. `-ge`: greater or equal
    f. `-le`: less or equal

7. IF,ELSE,ELSSIF

```
if [ condition ]; then

    # code if condition is true

elif [ another_condition ]; then

    # code if second condition is true

else

    # code if none of the above conditions are true

Fi
```

- Used to make decisions based on **conditions**.

8. For Loop

```
for i in 1 2 3
do
    echo "Item: $i"
done
#This prints each number in the list: 1, 2, 3.
```

- Used to repeat a set of commands for each item in a list.

9. While loop

```
count=1
while [ $count -le 5 ]
do
    echo "Count: $count"
    count=$((count + 1))
done
#Runs while $count is less than or equal to 5.
```

- Repeats as long as a condition is true.

Scripts to automate log cleanup

```
#!/bin/bash


# sets variables to files location

LOG_DIR= "/var/log/custom_logs"

BACKUP_FILE= "/var/log/logs_backup.tar.gz"

TEMP_DELETE_LIST= "/tmp/deleted_logs.txt"

TEMP_ARCHIVE_LIST= "/tmp/archived_logs.txt"


# Ensure temp files are clean

> "$TEMP_DELETE_LIST"

> "$TEMP_ARCHIVE_LIST"


echo "Starting log cleanup and archiving process..."


#  Find and delete .log files older than 7 days

echo "Searching for .log files older than 7 days in $LOG_DIR..."

find "$LOG_DIR" -name "*.log" -type f -mtime +7 -print -delete >
"$TEMP_DELETE_LIST"


# remaining .log files

echo "Archiving remaining .log files..."

find "$LOG_DIR" -name "*.log" -type f > "$TEMP_ARCHIVE_LIST"

tar -czf "$BACKUP_FILE" -T "$TEMP_ARCHIVE_LIST"


# Print summary
```

```
echo "-----------------------------------"

echo "Log Cleanup Summary:"

echo ""

echo "Deleted files:"

cat "$TEMP_DELETE_LIST"

echo ""

echo "Archived files:"

cat "$TEMP_ARCHIVE_LIST"

echo ""

echo "Backup archive created at: $BACKUP_FILE"

echo "-----------------------------------"


# Clean up temp files

rm -f "$TEMP_DELETE_LIST" "$TEMP_ARCHIVE_LIST"
```

Cron Job: Run Every Sunday at 12:00 AM

1. Open the crontab editor

   Code : `crontab -e`

   ```
   gimhana@gimhana:~$ crontab -e
   ```

2. Set date, time and path :

   Code : `0 0 * * 0 /home/gimhana/log_cleanup.sh`

   ```
   0 0 * * 0 /home/gimhana/log_cleanup.sh
   ```

• SSH Server

SSH (Secure Shell) is a network protocol that enables secure remote login and command execution on another computer over a network. It encrypts the connection to protect sensitive data like passwords and commands from eavesdropping. Commonly used by system administrators, SSH allows secure file transfers and remote server management. It replaces older, insecure protocols like Telnet, offering confidentiality, integrity, and authentication during communication between client and server.

Install & Configure SSH Server

1. Install the OpenSSH Server

   Code : `sudo apt install openssh-server`

```
gimhana@gimhana:~$ sudo apt install openssh-server
[sudo] password for gimhana:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 64 not upgraded.
Need to get 832 kB of archives.
After this operation, 6,743 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

2. Enable and Start the SSH Service

   Code : `sudo systemctl enable ssh`

   `sudo systemctl start ssh`

3. Check SSH Service Status

   Code : `sudo systemctl status ssh`

Ensure the run your server correctly.

4. Test SSH server

   Code : `ssh localhost`

```
gimhana@gimhana:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:C0qWVEudk35K+LhRfg8Uuosk4Hb6gkXlKz/DHTjX+kw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
gimhana@localhost's password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-24-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.

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

10 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

5. Allow SSH Through the Firewall

Code : `sudo ufw allow ssh`

`sudo ufw enable`

```
gimhana@gimhana:~$ sudo ufw allow ssh
[sudo] password for gimhana:
Rules updated
Rules updated (v6)
gimhana@gimhana:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
```

6. Get your IP

Code : `ip a`

```
gimhana@gimhana:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:0a:0c:0a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85619sec preferred_lft 85619sec
    inet6 fd00::cd40:4afd:e349:acde/64 scope global temporary dynamic
       valid_lft 86335sec preferred_lft 14335sec
    inet6 fd00::a00:27ff:fe0a:c0a/64 scope global dynamic mngtmpaddr noprefixroute
       valid_lft 86335sec preferred_lft 14335sec
    inet6 fe80::a00:27ff:fe0a:c0a/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

7. Now go to your client machine and repeat the first 3 steps

8. Connect from the client machine using an SSH Client

   Code : `ssh gimhana@10.0.2.15`

Replace `gimhana` with your Linux username

Replace `10.0.2.15` with your Linux machine's IP

```
gimhana@gimhana:~$ ssh gimhana@10.0.2.15
The authenticity of host '10.0.2.15 (10.0.2.15)' can't be established.
ED25519 key fingerprint is SHA256:C0qWVEudk35K+LhRfg8Uuosk4Hb6gkXlKz/DHTjX+kw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.15' (ED25519) to the list of known hosts.
gimhana@10.0.2.15's password:
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-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.

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

10 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

Do you want to disconnect SSH server use this code : `EXIT` and hit enter.

- # iptables and ACLs

`iptables` is a command-line utility used to set up, maintain, and inspect the *tables of IP packet filter rules* in the Linux kernel. It's essentially a **firewall tool** that controls **incoming and outgoing network traffic**.

ACLs (Access Control Lists) provide a more **granular permission system** for files and directories in Linux than the standard user/group/other model. With ACLs, you can set **different permissions for multiple users or groups** on a single file or directory.

iptables configuration and set rules

1. Frist Install iptables

   Code : `sudo apt install iptables`

2. Block Social Media Websites

First, find IPs using `nslookup`

Code : `nslookup facebook.com`

`nslookup instagram.com`

`nslookup twitter.com`

```
gimhana@gimhana:~$ nslookup facebook.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   facebook.com
Address: 163.70.144.35
Name:   facebook.com
Address: 2a03:2880:f188:181:face:b00c:0:25de

gimhana@gimhana:~$ nslookup instagram.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   instagram.com
Address: 163.70.144.174
Name:   instagram.com
Address: 2a03:2880:f288:1e2:face:b00c:0:4420

gimhana@gimhana:~$ nslookup twitter.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   twitter.com
Address: 162.159.140.229
```

Then, add `iptables` rules:

Code: `sudo iptables -A OUTPUT -p tcp -d 163.70.144.35 -j REJECT`

`sudo iptables -A OUTPUT -p tcp -d 163.70.144.174 -j REJECT`

`sudo iptables -A OUTPUT -p tcp -d 162.159.140.229 -j REJECT`

```
gimhana@gimhana:~$ sudo iptables -A OUTPUT -p tcp -d 163.70.144.35 -j REJECT
[sudo] password for gimhana:
gimhana@gimhana:~$ sudo iptables -A OUTPUT -p tcp -d 163.70.144.174 -j REJECT
gimhana@gimhana:~$ sudo iptables -A OUTPUT -p tcp -d 162.159.140.229 -j REJECT
```

to check those rules were added :

Code : `sudo iptables -L OUTPUT -v -n`

```
gimhana@gimhana:~$ sudo iptables -L OUTPUT -v -n
Chain OUTPUT (policy ACCEPT 5 packets, 200 bytes)
 pkts bytes target     prot opt in     out     source               destination
 2639  177K ufw-before-logging-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
 2639  177K ufw-before-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  480 37457 ufw-after-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  480 37457 ufw-after-logging-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  480 37457 ufw-reject-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  480 37457 ufw-track-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 REJECT     6    --  *      *       0.0.0.0/0            163.70.144.35        reject-with icmp-port-unreachable
    0     0 REJECT     6    --  *      *       0.0.0.0/0            163.70.144.174       reject-with icmp-port-unreachable
    0     0 REJECT     6    --  *      *       0.0.0.0/0            162.159.140.229      reject-with icmp-port-unreachable
```

3. Allow Only Secure Web Browsing

Block HTTP (port 80), allow HTTPS (port 443):

Code : Block HTTP

`sudo iptables -A OUTPUT -p tcp --dport 80 -j REJECT`

Allow HTTPS

`sudo iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT`

```
gimhana@gimhana:~$ sudo iptables -A OUTPUT -p tcp --dport 80 -j REJECT
gimhana@gimhana:~$ sudo iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT
```

Save iptables rules;

Code : `sudo apt install iptables-persistent`

`sudo netfilter-persistent save`

```
gimhana@gimhana:~$ sudo apt install iptables-persistent
```

```
gimhana@gimhana:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
```

Remove a Rule (if needed)

Frist, find the rule number :

Code : `sudo iptables -L OUTPUT --line-numbers`

```
gimhana@gimhana:~$ sudo iptables -L OUTPUT --line-numbers
Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    ufw-before-logging-output  all  --  anywhere             anywhere
2    ufw-before-output  all  --  anywhere             anywhere
3    ufw-after-output  all  --  anywhere             anywhere
4    ufw-after-logging-output  all  --  anywhere             anywhere
5    ufw-reject-output  all  --  anywhere             anywhere
6    ufw-track-output  all  --  anywhere             anywhere
7    REJECT     tcp  --  anywhere             edge-star-mini-shv-02-bom2.facebook.com  reject-with icmp-port-unreachable
8    REJECT     tcp  --  anywhere             instagram-p42-shv-02-bom2.fbcdn.net  reject-with icmp-port-unreachable
9    REJECT     tcp  --  anywhere             162.159.140.229      reject-with icmp-port-unreachable
10   REJECT     tcp  --  anywhere             anywhere             tcp dpt:http reject-with icmp-port-unreachable
11   ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
```

Delete it:

Code : `sudo iptables -D OUTPUT <Re-place your rule number>`

Access Control Lists (ACLs) Install

1. Install ACL

Code : `sudo apt install acl`

```
gimhana@gimhana:~$ sudo apt install acl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
acl is already the newest version (2.3.2-1build1.1).
acl set to manually installed.
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 61 not upgraded.
```

2. Verify ACL support on a mounted filesystem

Code : `mount | grep ' / '`

```
gimhana@gimhana:~$ mount | grep ' / '
/dev/sda2 on / type ext4 (rw,relatime)
```

- # Web Server

A web server is a system (hardware and software) that stores website files and delivers them to users' browsers upon request. The software listens for these requests via HTTP and sends back the corresponding files, enabling users to view websites. Popular examples include Apache, Nginx, LiteSpeed and etc. In this case we are using Apache.

1. Install the web server

Code : `sudo apt install apache2`

```
gimhana@gimhana:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom ufw
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64
0 upgraded, 8 newly installed, 0 to remove and 61 not upgraded.
Need to get 1,900 kB of archives.
After this operation, 7,455 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

2. Create a simple web page

Create a newindex.html file in the web server's default directory. For Apache, the default directory is usually `/var/www/html/`.

Code : `sudo nano /var/www/html/newindex.html`

```
gimhana@gimhana:~$ sudo nano /var/www/html/newindex.html
```

Add some basic HTML content to the file and save it.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Chek web server</title>
</head>
<body>
    <header>
        <h1>Hello World!</h1>
        <p>This webpage for check web sever.</p>
    </header>

    <main>
        <section>
            <h2>Welcome</h2>
            <p>Welcome to my website! This is a simple example of an HTML page.</p>
             <p>Here is a list of my favorite fruits:</p>
            <ul>
              <li>Mango</li>
              <li>Banana</li>
              <li>Apple</li>
            </ul>
        </section>

        <section>
            <h2>About Me</h2>
            <p>I am a web developer learning HTML.</p>
            <p>You can contact me at: <a href="mailto:example@email.com">example@email.com</a></p>
        </section>
    </main>
```

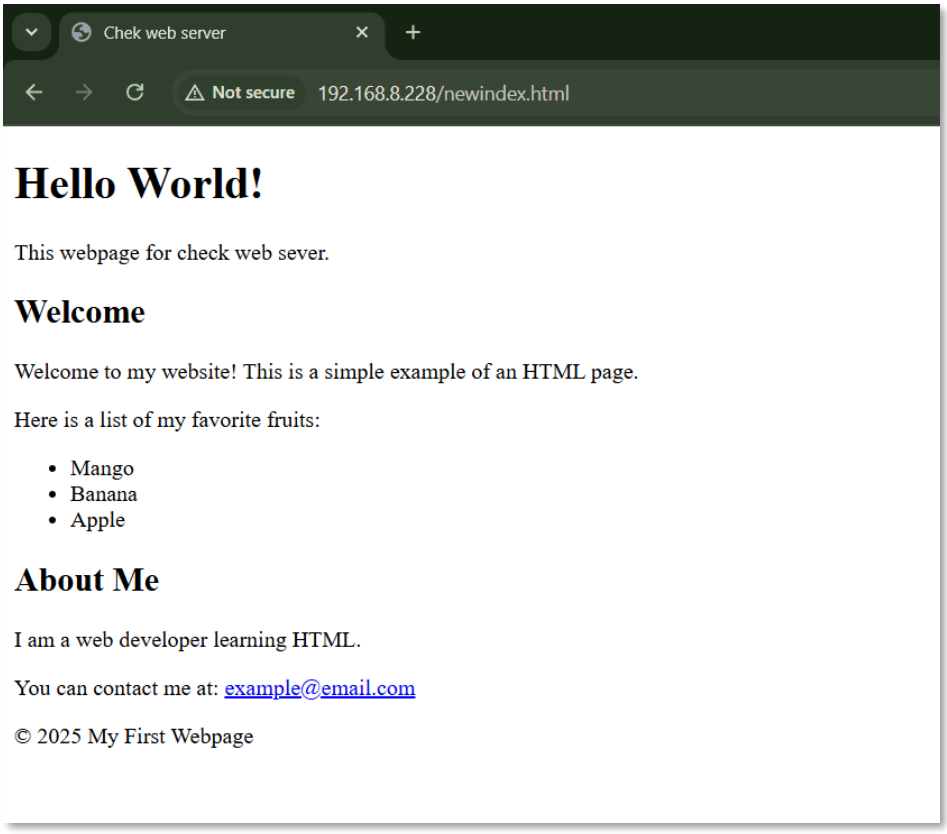
3. Now get your IP address.

Code : `ifconfig`

```
gimhana@gimhana:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.8.228  netmask 255.255.255.0  broadcast 192.168.8.255
        inet6 fe80::a00:27ff:fe0a:c0a  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:0a:0c:0a  txqueuelen 1000  (Ethernet)
        RX packets 328  bytes 83647 (83.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 362  bytes 59823 (59.8 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 373  bytes 35229 (35.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 373  bytes 35229 (35.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

4. Next go to your windows chrome or other browser and enter the IP address and webpage file name of your virtual machine.

   `http://192.168.8.228/newindex.html`



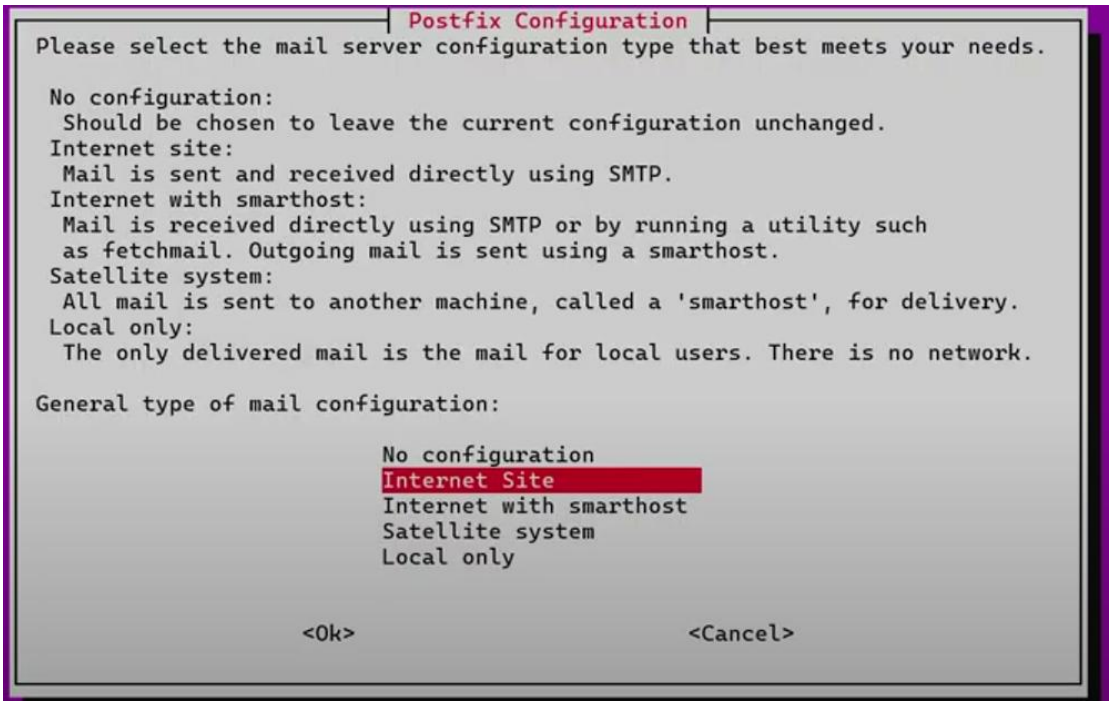
- # Email Server

An **Email Server** is a digital post office, crucial for sending, receiving, and storing electronic mail. Utilizing protocols like SMTP, POP3, and IMAP, it routes messages across networks, facilitated by DNS lookups for recipient servers. Whether managed internally or via cloud services, these servers ensure reliable communication in our digital world.

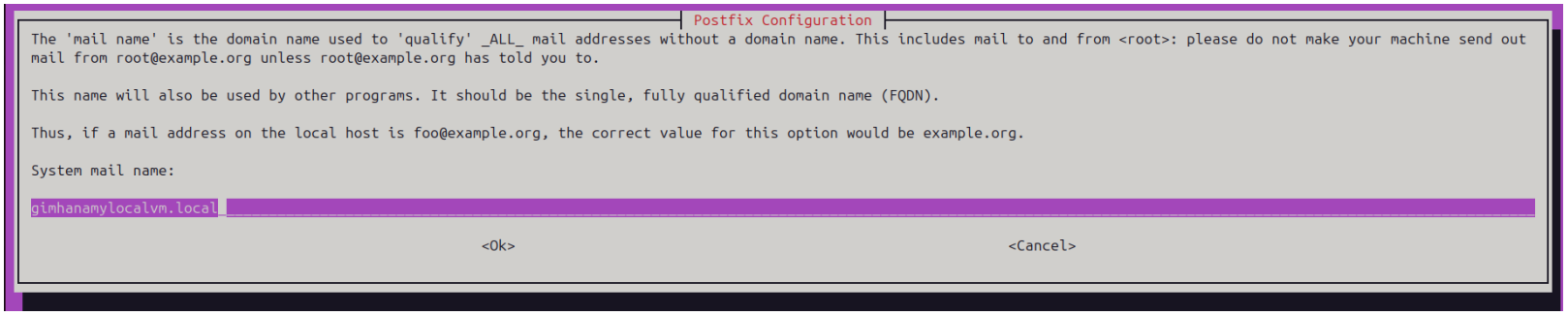
1. Frist install Postfix

   Code : `sudo apt install postfix -y`

Select "Internet Site"



Give a email



2. Next install the `mailutils` package

Code : `sudo apt-get install mailutils -y`

```
gimhana@gimhana:~$ sudo apt-get install mailutils -y
```

Check the statues of postfix :

Code : `sudo systemctl status postfix`

```
gimhana@gimhana:~$ sudo systemctl status postfix
● postfix.service - Postfix Mail Transport Agent
     Loaded: loaded (/usr/lib/systemd/system/postfix.service; enabled; preset: enabled)
     Active: active (exited) since Fri 2025-05-02 21:50:23 +0530; 14min ago
       Docs: man:postfix(1)
    Process: 4181 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 4181 (code=exited, status=0/SUCCESS)
        CPU: 1ms

May 02 21:50:23 gimhana systemd[1]: Starting postfix.service - Postfix Mail Transport Agent...
May 02 21:50:23 gimhana systemd[1]: Finished postfix.service - Postfix Mail Transport Agent.
```

3. Configure Postfix

Code : `sudo nano /etc/postfix/main.cf`

```
  GNU nano 7.2
# See /usr/share/postfix/main.cf.dist for a commented, more complete version


# Debian specific:  Specifying a file name will cause the first
# line of that file to be used as the name.  The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 3.6 on
# fresh installs.
compatibility_level = 3.6



# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_tls_security_level=may

smtp_tls_CApath=/etc/ssl/certs
smtp_tls_security_level=may
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
```

Now edit the file. Add the mail you create before.

```
smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination
myhostname = gimhanamylocalvm.local
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = gimhanamylocalvm.local
mydestination = $myhostname, gimhanamylocalvm.local, gimhana, localhost.localdomain, localhost
relayhost =
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = all
```

Save and exit.

4. Restarts and check the postfix.

Code : `sudo systemctl restart postfix`

`sudo systemctl status postfix`

```
gimhana@gimhana:~$ sudo systemctl restart postfix
[sudo] password for gimhana:
gimhana@gimhana:~$ sudo systemctl status postfix
● postfix.service - Postfix Mail Transport Agent
     Loaded: loaded (/usr/lib/systemd/system/postfix.service; enabled; preset: enabled)
     Active: active (exited) since Fri 2025-05-02 22:35:46 +0530; 10s ago
       Docs: man:postfix(1)
    Process: 5575 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 5575 (code=exited, status=0/SUCCESS)
        CPU: 3ms

May 02 22:35:46 gimhana systemd[1]: Starting postfix.service - Postfix Mail Transport Agent...
May 02 22:35:46 gimhana systemd[1]: Finished postfix.service - Postfix Mail Transport Agent.
```

5. Testing Configuration

Code : `echo "This is a test email from my Postfix server!" | mail -s "Test Email" gimhana@ gimhanamylocalvm.local`

Replace this `"This is a test email from my Postfix server!"` with your massage.

Replace this `"Test Email"` with your massage subject.

Replace this `gimhana@ gimhanamylocalvm.local` with machince's user name and created host.

```
gimhana@gimhana:~$ echo "This is a test email from my Postfix server!" | mail -s "Test Email" gimhana@gimhanamylocalvm.local
```

6. Check and read mail

Code : `mail`

```
gimhana@gimhana:~$ mail
"/var/mail/gimhana": 1 message 1 new
>N   1 GIMHANA            Fri May  2 22:47  13/516   Test Email
```

Mail number

Now read the mail :

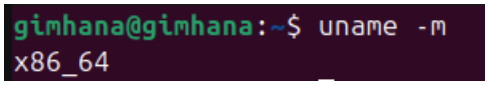
Type your mail number and hit enter.

```
gimhana@gimhana:~$ mail
"/var/mail/gimhana": 1 message 1 new
>N   1 GIMHANA            Fri May  2 22:47  13/516   Test Email
? 1
Return-Path: <gimhana@gimhana>
X-Original-To: gimhana@gimhanamylocalvm.local
Delivered-To: gimhana@gimhanamylocalvm.local
Received: by gimhanamylocalvm.local (Postfix, from userid 1000)
        id 5EB3241915; Fri,  2 May 2025 22:47:59 +0530 (+0530)
Subject: Test Email
To: <gimhana@gimhanamylocalvm.local>
User-Agent: mail (GNU Mailutils 3.17)
Date: Fri,  2 May 2025 22:47:59 +0530
Message-Id: <20250502171759.5EB3241915@gimhanamylocalvm.local>
From: GIMHANA <gimhana@gimhana>

This is a test email from my Postfix server!
```

- **Linux GDB**

GDB, the GNU Debugger, is a powerful command-line tool for debugging programs written in various languages on Linux. It allows you to inspect program execution, set breakpoints, examine variables, and analyze core dumps when crashes occur. GDB is essential for understanding program behavior and identifying the root causes of errors, aiding in efficient software development and debugging workflows.

1. Open linux terminal and run this code : `uname -m`

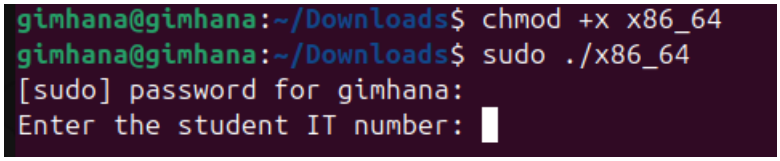


2. Give the executable execute permissions .

   Code : `chmod +x x86_64`

3. Run the executable with root privileges using

   Code : `sudo ./x86_64`

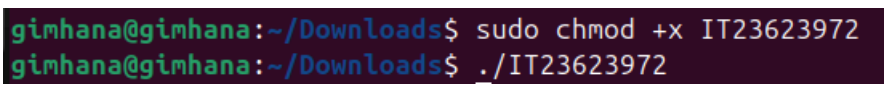


Give the IT number.

4. Run the Generated Executable

   Code : `sudo chmod +x IT23623972`

   `./IT23623972`



5. Open `data.txt` using a text editor `(cat,nano)`

Code : `cat data.txt`

`nano data.txt`

`strings data.txt`

```
gimhana@gimhana:~/Downloads$ cat data.txt
KZRgimhana@gimhana:~/Downloads$ cat data.txt
KZRgimhana@gimhana:~/Downloads$ strings data.txt
\LS\T^QARHIZQ
gimhana@gimhana:~/Downloads$ nano data.txt
gimhana@gimhana:~/Downloads$
```

```
  GNU nano 7.2
        \M^O\LS\T^QARHIZQ^_F^DO\^AT\Q^]_WM
^AKZR^\
```

Debug the New Executable Using GDB and Analyzing

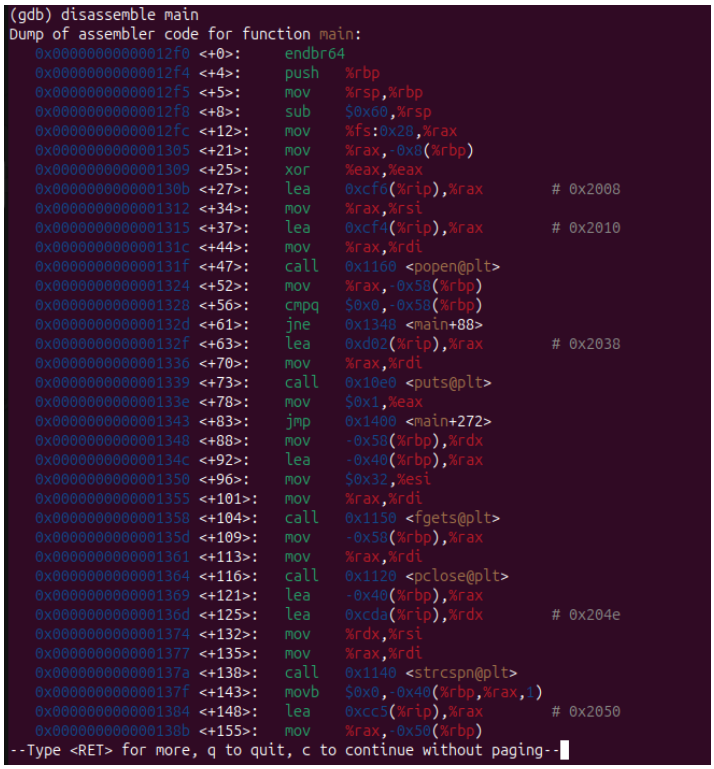
6. Start GDB

Code : `gdb IT23623972`

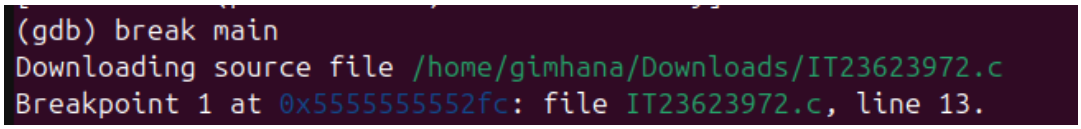
```
gimhana@gimhana:~/Downloads$ gdb IT23623972
GNU gdb (Ubuntu 15.0.50.20240403-0ubuntu1) 15.0.50.20240403-git
Copyright (C) 2024 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from IT23623972...
(gdb)
```

GDB commands and output

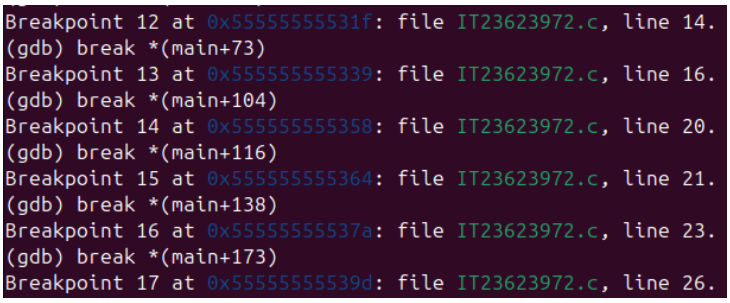
- Command : `disassemble main`

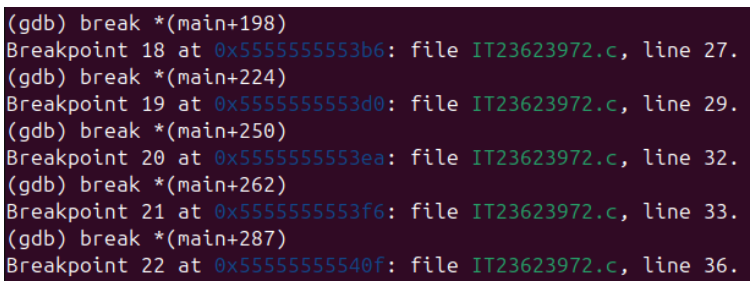


- Command : `break main`



- Put breakpoints where the important function calls(popen(),fgets(),strcspn(),..etc) :





- Run after add breakpoint :

Used the command "`x/64xb $rbp-0x50`" to check the memory dumb of the memory address "rbp-0x50". To check the message buffer content before reading.

```
(gdb) run
Starting program: /home/gimhana/Downloads/IT23623972
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, 0x000055555555531f in main () at IT23623972.c:14
warning: 14     IT23623972.c: No such file or directory
(gdb) x/64xb $rbp-0x50
0x7fffffffe320: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe328: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe330: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe338: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe340: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe348: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe350: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe358: 0xf0    0x5a    0xfe    0xf7    0xff    0x7f    0x00    0x00
```

Values changed at breakpoint 6

```
Breakpoint 6, 0x000055555555539d in main () at IT23623972.c:26
26      in IT23623972.c
(gdb) x/64xb $rbp-0x50
0x7fffffffe320: 0x50    0x60    0x55    0x55    0x55    0x55    0x00    0x00
0x7fffffffe328: 0x00    0x00    0x00    0x00    0x00    0x00    0x00    0x00
0x7fffffffe330: 0x62    0x39    0x34    0x64    0x39    0x35    0x38    0x39
0x7fffffffe338: 0x2d    0x35    0x34    0x38    0x39    0x2d    0x30    0x31
0x7fffffffe340: 0x34    0x66    0x2d    0x61    0x36    0x37    0x64    0x2d
0x7fffffffe348: 0x37    0x34    0x64    0x34    0x32    0x34    0x66    0x64
0x7fffffffe350: 0x32    0x31    0x37    0x65    0x00    0x00    0x00    0x00
0x7fffffffe358: 0xf0    0x5a    0xfe    0xf7    0xff    0x7f    0x00    0x00
```

- Command : `run`

```
(gdb) run
Starting program: /home/gimhana/Downloads/IT23623972
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, main () at IT23623972.c:13
warning: 13     IT23623972.c: No such file or directory
```

- Command : `next`

```
(gdb) next
14      in IT23623972.c
(gdb) next
[Detaching after vfork from child process 7557]
15      in IT23623972.c
```

- Command : `step`

```
(gdb) step
Download failed: Invalid argument.  Continuing without source file ./libio/./libio/pclose.c.
__new_pclose (fp=0x5555555592a0) at ./libio/pclose.c:37
```

- Command : `info registers`

```
(gdb) info registers
rax            0x5555555592a0      93824992252576
rbx            0x7fffffffded8      140737488346840
rcx            0x1                 1
rdx            0xfbad2488          4222428296
rsi            0x5555555594c1      93824992253121
rdi            0x5555555592a0      93824992252576
rbp            0x7fffffffddb0      0x7fffffffddb0
rsp            0x7fffffffdd48      0x7fffffffdd48
r8             0x5555555594e5      93824992253157
r9             0x0                 0
r10            0x1                 1
r11            0x246               582
r12            0x1                 1
r13            0x0                 0
r14            0x555555557d78      93824992247160
r15            0x7ffff7ffd000      140737354125312
rip            0x7ffff7c8f460      0x7ffff7c8f460 <__new_pclose>
eflags         0x10246             [ PF ZF IF RF ]
cs             0x33                51
ss             0x2b                43
ds             0x0                 0
es             0x0                 0
fs             0x0                 0
gs             0x0                 0
fs_base        0x7ffff7fa9740      140737353783104
gs_base        0x0                 0
```

- Command : `disas`

```
(gdb) disas
Dump of assembler code for function __new_pclose:
=> 0x00007ffff7c8f460 <+0>:     endbr64
   0x00007ffff7c8f464 <+4>:     jmp    0x7ffff7c85290 <_IO_new_fclose>
```

- Command : `continue`

```
(gdb) continue
Continuing.
[Inferior 1 (process 7549) exited normally]
```

Before run

```
gimhana@client:~/Downloads$ ls -lt
total 40
-rw-rw-r-- 1 gimhana gimhana    36 May  3 09:49 data.txt
-rwxrwxr-x 1 gimhana gimhana 20344 May  3 09:48 IT23623972
drwxrwxr-x 4 gimhana gimhana  4096 May  3 09:44 Executables
-rw-rw-r-- 1 gimhana gimhana  8514 May  3 09:44 Executables.zip
gimhana@client:~/Downloads$ stat data.txt
  File: data.txt
  Size: 36              Blocks: 8          IO Block: 4096   regular file
Device: 8,2     Inode: 1442501     Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/ gimhana)   Gid: ( 1000/ gimhana)
Access: 2025-05-03 09:49:04.626666936 +0530
Modify: 2025-05-03 09:49:04.626666936 +0530
Change: 2025-05-03 09:49:04.626666936 +0530
 Birth: 2025-05-03 09:49:04.626666936 +0530
gimhana@client:~/Downloads$
```

After run

```
gimhana@client:~/Downloads$ ls -lt
total 40
-rw-rw-r-- 1 gimhana gimhana    36 May  3 10:03 data.txt
-rwxrwxr-x 1 gimhana gimhana 20344 May  3 09:48 IT23623972
drwxrwxr-x 4 gimhana gimhana  4096 May  3 09:44 Executables
-rw-rw-r-- 1 gimhana gimhana  8514 May  3 09:44 Executables.zip
gimhana@client:~/Downloads$ stat data.txt
  File: data.txt
  Size: 36              Blocks: 8          IO Block: 4096   regular file
Device: 8,2     Inode: 1442501     Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/ gimhana)   Gid: ( 1000/ gimhana)
Access: 2025-05-03 09:49:04.626666936 +0530
Modify: 2025-05-03 10:03:40.352609065 +0530
Change: 2025-05-03 10:03:40.352609065 +0530
 Birth: 2025-05-03 09:49:04.626666936 +0530
gimhana@client:~/Downloads$
```

```
gimhana@client:~/Downloads$ strings data.txt
YSM^STYVK[H@    QNF]JYWT        U
gimhana@client:~/Downloads$ hexdump -C data.txt
00000000  59 01 18 59 53 4d 5e 53  54 59 56 4b 5b 48 40 09  |Y..YSM^STYVK[H@.|
00000010  51 4e 46 5d 4a 59 57 54  09 55 1d 09 04 4c 0f 52  |QNF]JYWT.U...L.R|
00000020  18 59 56 49                                       |.YVI|
00000024
```

```
gimhana@client:~/Downloads$ cat data.txt
YYSM^STYVK[H@   QNF]JYWT        U       LRYVIgimhana@client:~/Downloads$
```

Tools used for analysis.

- `cat`
- `ls -lt`
- `stat`
- `strings`
- `hexdump`