Sri Lanka Institute of Information Technology (SLIIT)

SNP Assignment

IT23586116 – LUTHFI H P

BSc Honors degree in IT (Sp. Cyber Security)

Systems and Network Programming- IE2012

Table of Contents

1. Basics of Linux Environments

a. Steps of setting up the VM and Kali Linux

Steps of installing the Virtual Machine

- Visit the oracle virtual box website - https://www.virtualbox.org
- Navigate to downloads tab.
- Click on "Accept and download.
- Double click on the exe virtual Box installer file.
- VirtualBox Setup Wizard will open, click Next and proceed with the default settings and install the software.
- Finally click finish to exit the successful installation

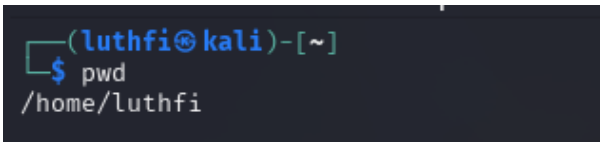
Steps of installing and setting up Kali Linux

- Visit Kali Linux official Website - https://www.kali.org/get-kali/
- Click on virtual machines and download the ISO Kali image for virtual box. (around 4.1 GB)
- To create a new virtual machine, open oracle virtual box and click "New".
- Enter a name for your machine.
- Select the type as Linux and the version as Debian 64 bit and move to the next section.
- Select the RAM space and move to the next section. (Recommended 4GB)
- Select the hard disk size, at least 25GB (Recommended 50GB)
- Select the downloaded ISO kali image and click finish.
- To configure the system settings, click on settings and navigate to system to processor and allocate 2 or more CPUs.
- For network settings, click network on settings and set the adapter to NAT.
- Now open the VM and click Start.
- Select the graphical install and press Enter.
- Select the language as English and press Enter.
- Configure the region and keyboard layout and press Enter.
- Customize the host name if needed and move to domain name and continue.
- Set up a new user with a username and a password.
- Select guided disk partitioning and choose the virtual disk as all in one partition and press confirm to finish the process.

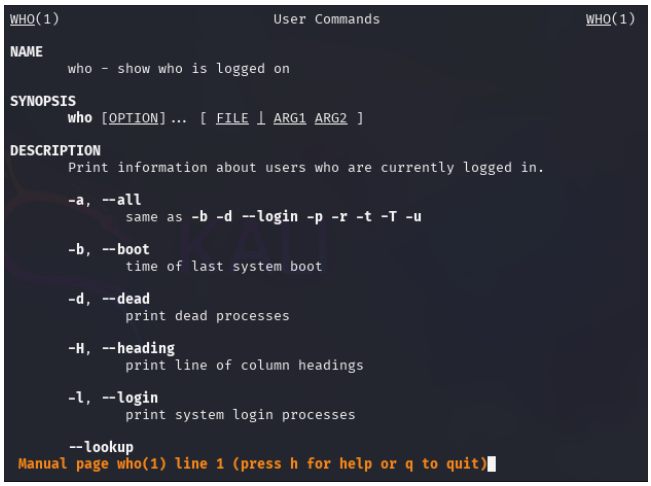
b. CLI commands demonstration

- whoami – To find the current user. Displays the username of the current user who has logged in.

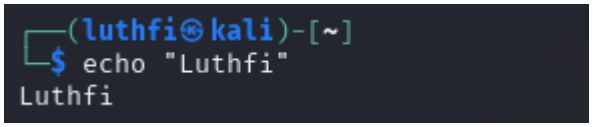
- pwd- Print the current working directory. Displays the full path the user is in.



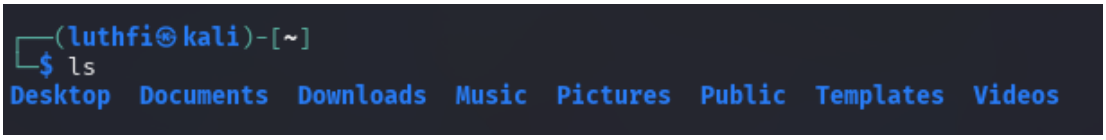
- man – Display the user manuals of commands. It describes how to use commands and their purpose.



- echo – Prints a string.

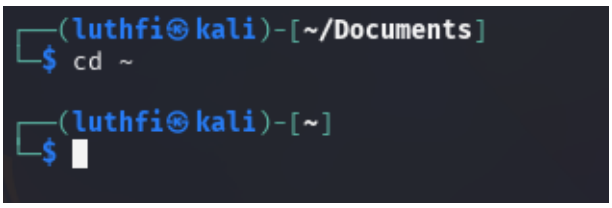


- ls – List the files and directories.



- cd – Used to change the directory.

- cd ~ - Move back to home directory



- cd .. – Move up one directory

- cat – Used to concatenate and display content.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ cat  snpA.txt
HI SNP
```

- cat /proc/version – Display information about Linux kernal

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ cat /proc/version
Linux version 6.12.13-amd64 (devel@kali.org) (x86_64-linux-gnu-gcc-14 (Debian
 14.2.0-16) 14.2.0, GNU ld (GNU Binutils for Debian) 2.44) #1 SMP PREEMPT_DYN
AMIC Kali 6.12.13-1kali1 (2025-02-11)
```

- mkdir – Create a directory.

```
┌──(luthfi㉿kali)-[~]
└─$ mkdir Luthfi

┌──(luthfi㉿kali)-[~]
└─$ ls
Desktop    Downloads  Music     Public     Videos
Documents  Luthfi     Pictures  Templates
```

- rmdir – Delete a directory.

```
┌──(luthfi㉿kali)-[~]
└─$ ls
Desktop    Downloads  Music     Public     Templates
Documents  Luthfi     Pictures  snp        Videos

┌──(luthfi㉿kali)-[~]
└─$ rmdir snp

┌──(luthfi㉿kali)-[~]
└─$ ls
Desktop    Downloads  Music     Public     Videos
Documents  Luthfi     Pictures  Templates
```

- cp – Copy file

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ cp snpA.txt newSNP.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ cat newSNP.txt
HI SNP
```

- mv – move or rename a file

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ mv snpA.txt moved.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ cat moved.txt
HI SNP
```

- df -h – Displays the disk space usage in human-readable format.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            1.9G     0  1.9G   0% /dev
tmpfs           393M  1.1M  392M   1% /run
/dev/sda1        79G   15G   60G  20% /
tmpfs           2.0G  4.0K  2.0G   1% /dev/shm
tmpfs           1.0M     0  1.0M   0% /run/credentials/systemd-journald.service
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           2.0G   28K  2.0G   1% /tmp
tmpfs           1.0M     0  1.0M   0% /run/credentials/getty@tty1.service
tmpfs           393M  116K  393M   1% /run/user/1000
tmpfs           393M  124K  393M   1% /run/user/1001
```

- grep – search word in a file. It highlights word in red.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ grep HI newSNP.txt
HI SNP
```

- touch – To create a empty file.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ touch TOUCHME.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ls
moved.txt  newSNP.txt  TOUCHME.txt
```

c. System Information and User Management commands

❖ uname -a – Displays the kernel information.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ uname -a
Linux kali 6.12.13-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.13-1kali1 (2025-02-11) x
86_64 GNU/Linux
```

❖ hostnamectl - Displays the hostname and OS or kernel details.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ hostnamectl
 Static hostname: kali
       Icon name: computer-vm
         Chassis: vm
      Machine ID: aeff9826fa3d4445943d8d2b88b5e03a
         Boot ID: 75f345ed3c2249c19d9059702d358669
  Virtualization: oracle
Operating System: Kali GNU/Linux Rolling
          Kernel: Linux 6.12.13-amd64
    Architecture: x86-64
 Hardware Vendor: innotek GmbH
  Hardware Model: VirtualBox
Firmware Version: VirtualBox
   Firmware Date: Fri 2006-12-01
    Firmware Age: 18y 4month 1w 4d
```

❖ ip a – Displays the IP address.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 08:00:27:04:42:0f brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 69703sec preferred_lft 69703sec
    inet6 fd00::fe8:a236:89a1:e94/64 scope global dynamic noprefixroute
       valid_lft 85405sec preferred_lft 13405sec
    inet6 fe80::17f3:6d9d:29c8:1d90/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

❖ top – View active process.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ top
top - 15:08:12 up  4:40,  4 users,  load average: 0.16, 0.07, 0.07
Tasks: 237 total,   1 running, 229 sleeping,   7 stopped,   0 zombie
%Cpu(s):  0.9 us,  0.9 sy,  0.0 ni, 98.2 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   3921.4 total,   1971.6 free,    992.5 used,   1226.0 buff/cache
MiB Swap:   1024.0 total,   1024.0 free,      0.0 used.   2928.9 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   3928 root      20   0  380768 102708  59104 S   2.3   2.6   2:43.11 Xorg
  18165 luthfi    20   0  724584  61704  47160 S   1.3   1.5   0:18.53 qterminal
     31 root      rt   0       0      0      0 S   0.3   0.0   0:01.63 migration/2
   4183 luthfi    20   0  215924   3076   2820 S   0.3   0.1   2:08.35 VBoxClient
   4414 luthfi    20   0  311292  64988  22992 S   0.3   1.6   0:56.08 wrapper-2.0
   4418 luthfi    20   0  273856  28844  21640 S   0.3   0.7   0:48.59 wrapper-2.0
      1 root      20   0   23620  14588  10452 S   0.0   0.4   0:03.89 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.16 kthreadd
      3 root      20   0       0      0      0 S   0.0   0.0   0:00.00 pool_workqueue_+
      4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-rcu_gp
      5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-sync_+
      6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-slub_+
      7 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-netns
     11 root      20   0       0      0      0 I   0.0   0.0   0:00.00 kworker/u16:0-i+
     12 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/R-mm_pe+
```

❖ ss -tulnp – Lists the open ports.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ss -tulnp
Netid State  Recv-Q Send-Q                    Local Address:Port    Peer Address:Port
Process
udp   UNCONN 0      0                             0.0.0.0:47307        0.0.0.0:*

udp   UNCONN 0      0                           10.0.2.15:3702         0.0.0.0:*
 users:(("python3",pid=5049,fd=9))
udp   UNCONN 0      0                       239.255.255.250:3702         0.0.0.0:*
 users:(("python3",pid=5049,fd=7))
udp   UNCONN 0      0                           10.0.2.15:3702         0.0.0.0:*

udp   UNCONN 0      0                       239.255.255.250:3702         0.0.0.0:*

udp   UNCONN 0      0                             0.0.0.0:36687        0.0.0.0:*
 users:(("python3",pid=5049,fd=8))
udp   UNCONN 0      0                                   *:45746              *:*

udp   UNCONN 0      0                                   *:33611              *:*
 users:(("python3",pid=5049,fd=11))
udp   UNCONN 0      0          [fe80::17f3:6d9d:29c8:1d90]%eth0:3702           [::]:*
 users:(("python3",pid=5049,fd=12))
udp   UNCONN 0      0                          [ff02::c]%eth0:3702           [::]:*
 users:(("python3",pid=5049,fd=10))
udp   UNCONN 0      0          [fe80::17f3:6d9d:29c8:1d90]%eth0:3702           [::]:*

udp   UNCONN 0      0                          [ff02::c]%eth0:3702           [::]:*
```

❖ w – shows the active users and processes.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ w
 15:10:04 up  4:41,  4 users,  load average: 0.05, 0.07, 0.07
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
luthfi   -        -                10:20    4:54m  0.00s  0.02s lightdm --session-child
luthfi   -        -                10:20    4:54m  0.00s  0.36s /usr/lib/systemd/system
kali     -        -                10:15    4:54m  0.00s  0.29s /usr/lib/systemd/system
```

❖ who – list of users Logged.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ sudo who
luthfi   pts/1        2025-04-12 15:11
```

❖ chmod – change file permissions.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ls -al newSNP.txt
-rw-rw-r-- 1 luthfi luthfi 7 Apr 12 14:01 newSNP.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ chmod 777 newSNP.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ls -al newSNP.txt
-rwxrwxrwx 1 luthfi luthfi 7 Apr 12 14:01 newSNP.txt
```

❖ chown – change file owner.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ls -l newSNP.txt
-rwxrwxrwx 1 luthfi luthfi 7 Apr 12 14:01 newSNP.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ sudo chown kali  newSNP.txt

┌──(luthfi㉿kali)-[~/Luthfi]
└─$ ls -l newSNP.txt
-rwxrwxrwx 1 kali luthfi 7 Apr 12 14:01 newSNP.txt
```

❖ sudo -i – start root shell.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ sudo -i
┌──(root㉿kali)-[~]
└─#
```

❖ sudo – To grant root permissions.

```
┌──(luthfi㉿kali)-[~/Luthfi]
└─$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-ABkNnS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-ABkNnS] [-g group] [-h host] [-p prompt] [-U user]
            [-u user] [command [arg ...]]
usage: sudo [-ABbEHkNnPS] [-r role] [-t type] [-C num] [-D directory]
            [-g group] [-h host] [-p prompt] [-R directory] [-T timeout]
            [-u user] [VAR=value] [-i | -s] [command [arg ...]]
usage: sudo -e [-ABkNnS] [-r role] [-t type] [-C num] [-D directory]
            [-g group] [-h host] [-p prompt] [-R directory] [-T timeout]
            [-u user] file ...
```

2. DHCP, DNS and NTP Services

a. Installation and Configuration of DHCP Server

DHCP is the Dynamic Host Configuration Protocol which is used to assign IP addresses Automatically in a network.

Benefits of having DHCP

- ◊ Automatically assigns IP address. (DHCP server)
- ◊ Managing the IP address leases. (DHCP lease)
- ◊ Preventing duplication of IP addresses. (Provide a Scope)
- ◊ Provide subnet masks and defaults gateways.

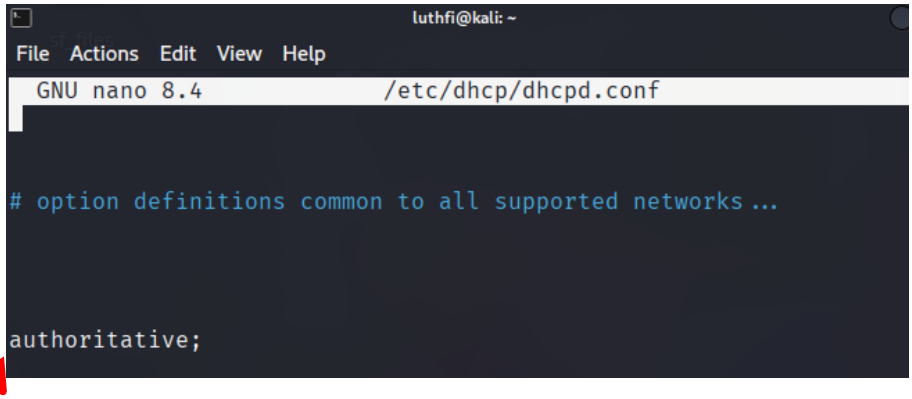
Step 1 -Installing ISC-DHCP Server (**sudo apt install isc-dhcp-server**)

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt install isc-dhcp-server
Installing:
  isc-dhcp-server

Suggested packages:
  isc-dhcp-server-ldap

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 947
  Download size: 0 B / 1,481 kB
  Space needed: 6,249 kB / 63.9 GB available

Preconfiguring packages ...
Selecting previously unselected package isc-dhcp-server.
(Reading database ... 408775 files and directories currently installed.)
Preparing to unpack .../isc-dhcp-server_4.4.3-P1-5+b1_amd64.deb ...
Unpacking isc-dhcp-server (4.4.3-P1-5+b1) ...
Setting up isc-dhcp-server (4.4.3-P1-5+b1) ...
Generating /etc/default/isc-dhcp-server...
update-rc.d: We have no instructions for the isc-dhcp-server init script.
update-rc.d: It looks like a network service, we disable it.
Processing triggers for kali-menu (2025.1.1) ...
Processing triggers for man-db (2.13.0-1) ...
```

Step 2 – Navigating to the configuration file

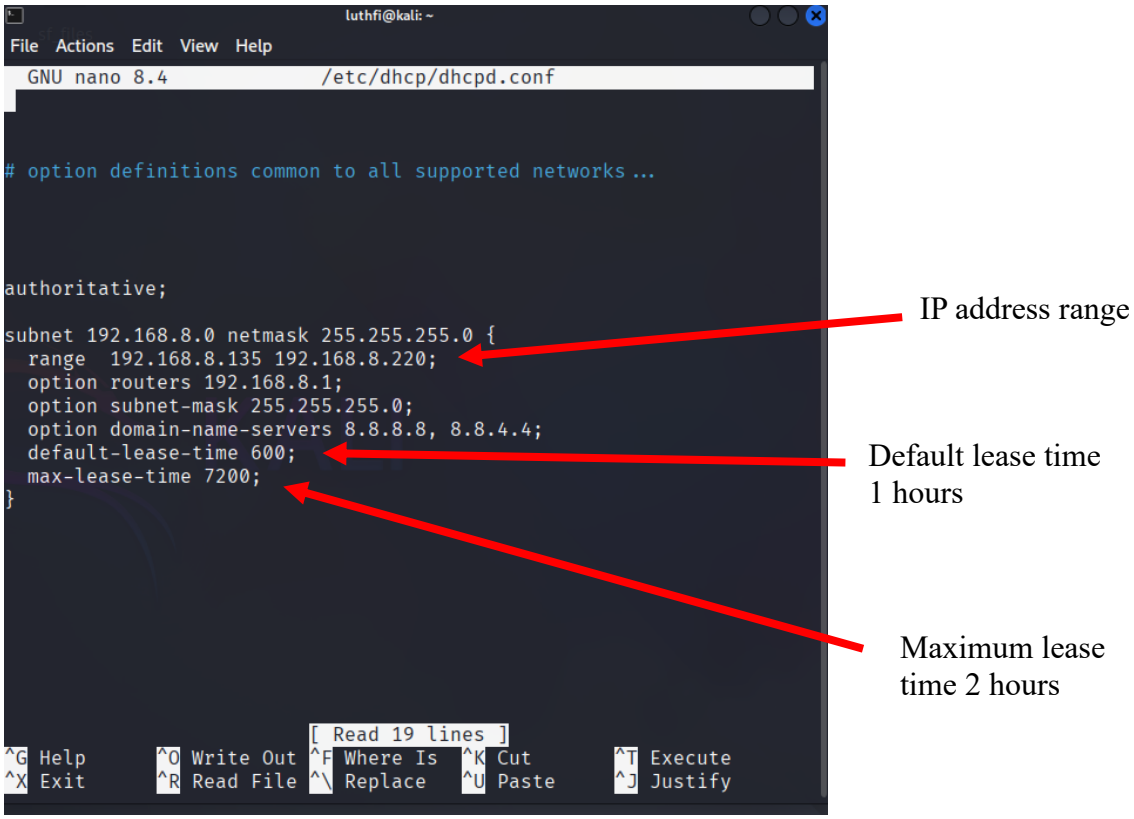
```
┌──(luthfi㉿kali)-[~]
└─$ cd /etc/dhcp

┌──(luthfi㉿kali)-[/etc/dhcp]
└─$ sudo nano dhcpd.conf
```

Step 3- activate the server in **dhcpd.conf** file



Activate the server by removing the hash (#) in the authoritative.

Step 4- configure the server settings in dhcpd.conf file



Step 5- change the ethernet port to “eth0”

```
┌──(luthfi㉿kali)-[~]
└─$ vi /etc/default/isc-dhcp-server
```

```
INTERFACESv4="eth0"
INTERFACESv6=""
```

- Enable the dhcp service **sudo systemctl enable isc-dhcp-server**
- **Systemctl** is used to manage the system services.

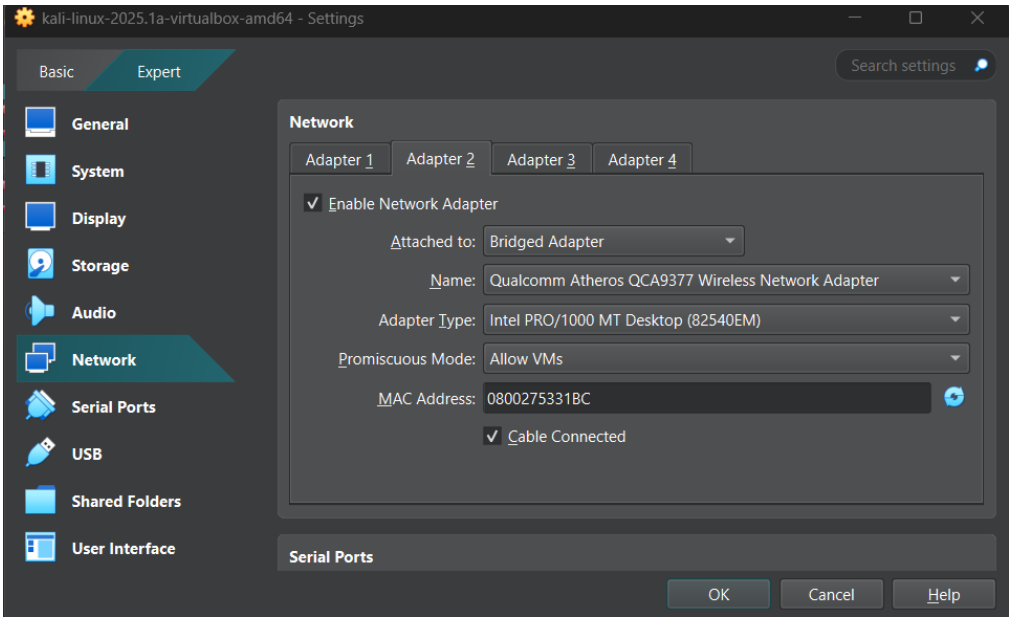
```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl enable isc-dhcp-server
Synchronizing state of isc-dhcp-server.service with SysV service script with /usr/lib
/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable isc-dhcp-server
```

- Start the dhcp server (**sudo systemctl start isc-dhcp-server**)

Step 6 – check restart and check the server status. (**sudo systemctl status isc-dhcp-server)**

```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl restart  isc-dhcp-server

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP Server
     Loaded: loaded (/etc/systemd/system/isc-dhcp-server.service; enabled; preset: d>
     Active: active (running) since Wed 2025-04-30 09:40:15 +0530; 4s ago
 Invocation: e8577f6e58be47e283ffdb8be6c6aa81
    Process: 7559 ExecStart=/usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf -pf /var/>
   Main PID: 7560 (dhcpd)
      Tasks: 1 (limit: 3951)
     Memory: 3.8M (peak: 4.1M)
        CPU: 16ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─7560 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf -pf /var/run/dhcp>

Apr 30 09:40:15 kali systemd[1]: Starting isc-dhcp-server.service - ISC DHCP Server.>
Apr 30 09:40:15 kali dhcpd[7560]: Wrote 1 leases to leases file.
Apr 30 09:40:15 kali dhcpd[7560]: Server starting service.
Apr 30 09:40:15 kali systemd[1]: Started isc-dhcp-server.service - ISC DHCP Server.
lines 1-16/16 (END)
[2]+  Stopped                 sudo systemctl status isc-dhcp-server
```

Step 7 – change the network settings in both DHCP client and DHCP server to **Bridged Adapter**.



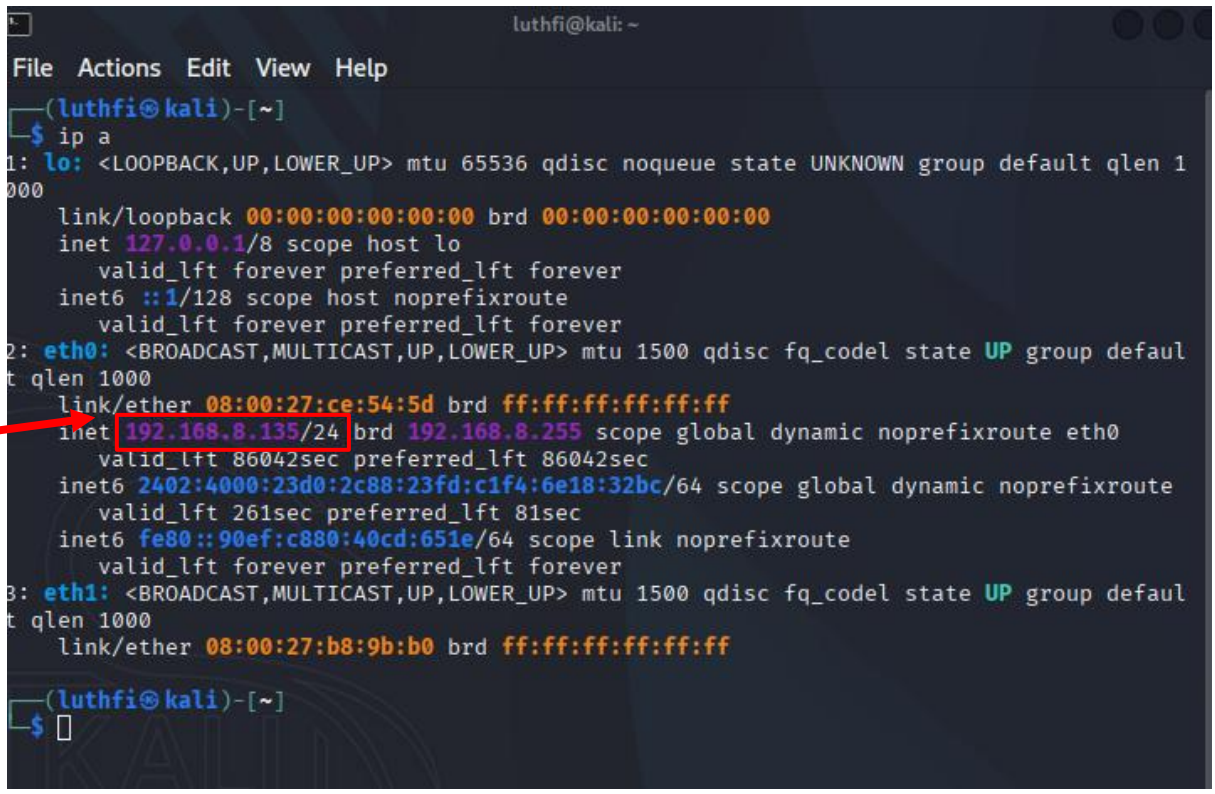
DHCP server Machine

Server machine IP address

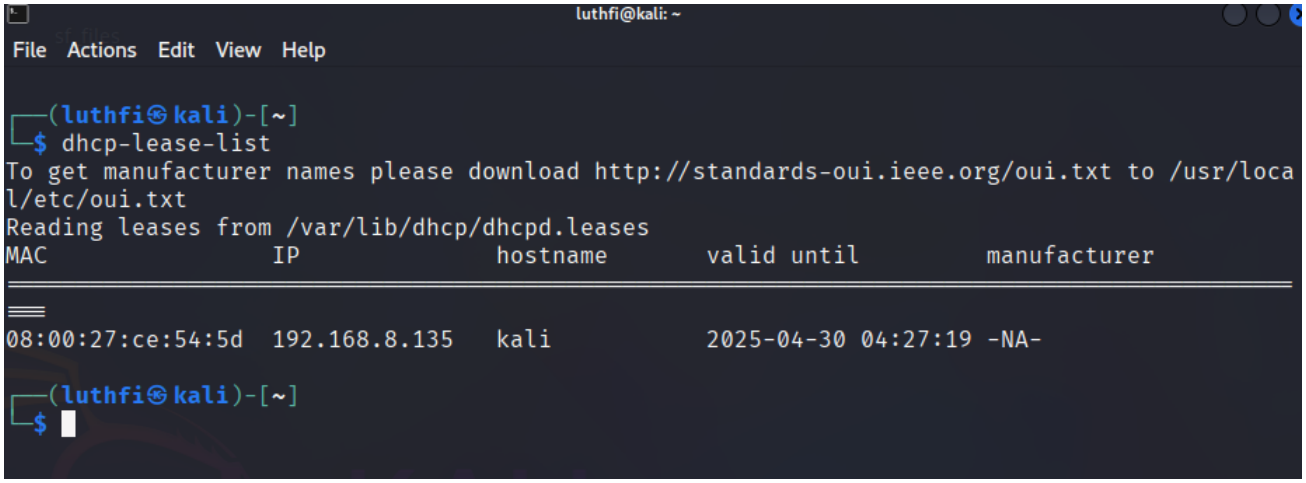
```
┌──(luthfi㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qle
n 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 08:00:27:53:31:bc brd ff:ff:ff:ff:ff:ff
    inet 192.168.8.129/24 brd 192.168.8.255 scope global dynamic noprefixroute eth0
       valid_lft 85356sec preferred_lft 85356sec
    inet6 2402:4000:23d0:2c88:b81e:66e0:9e82:c546/64 scope global dynamic noprefixrou
te
       valid_lft 251sec preferred_lft 71sec
    inet6 fe80::7ee2:7403:68db:cfc2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

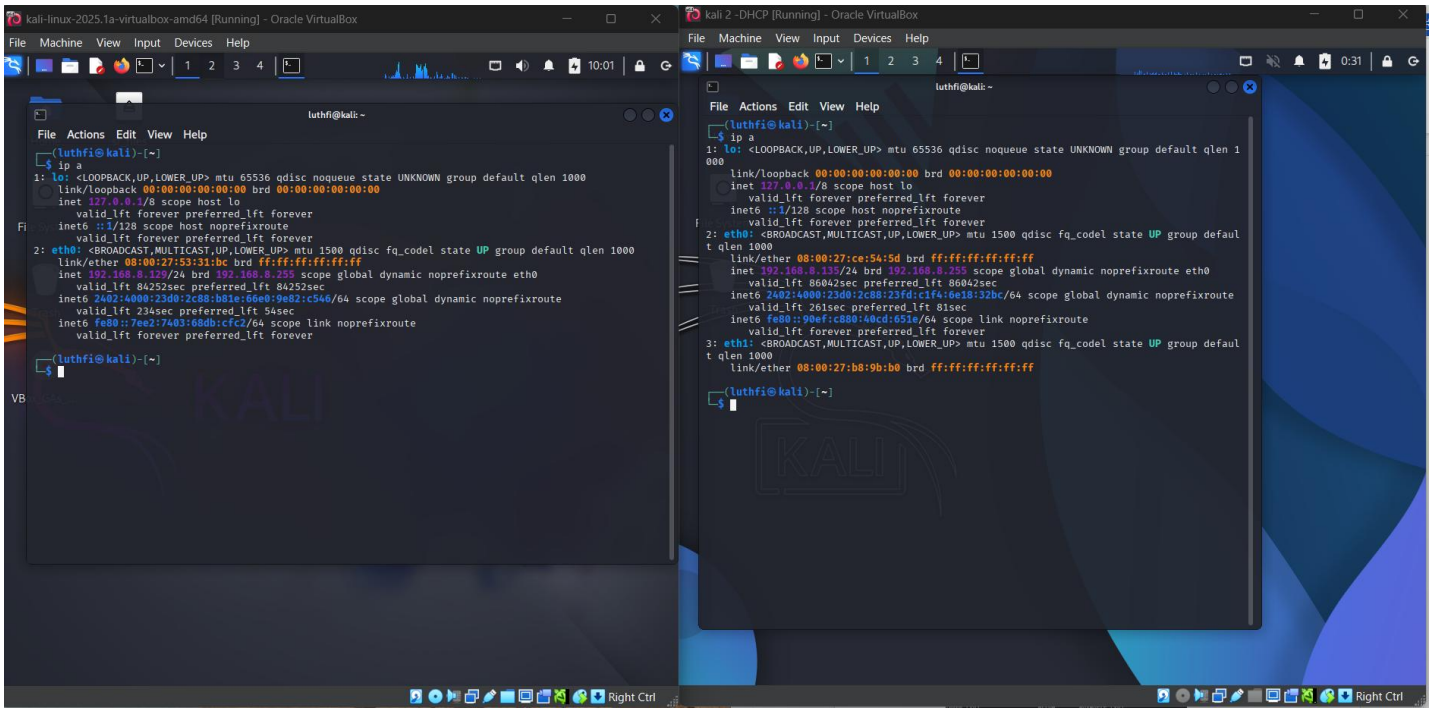
DHCP Client (kali Cloned)

Client machine IP address



Check the **dhcp-lease-list** to confirm the clients.





b. Installation and Configuration of DNS server

- DNS (Domain Name System) is used to map the hostnames to IP addresses. It helps uses to access websites through human-friendly names instead of numbers.

Step1- Installing BIND (Berkeley Internet Name Domain)

This code installs all packages related to bind.

(**sudo apt install bind9 bind9tils bind9-doc -y**)

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt install bind9 bind9utils bind9-doc -y
[sudo] password for luthfi:
Note, selecting 'bind9-utils' instead of 'bind9utils'
The following packages were automatically installed and are no longer required:
  icu-devtools   libglapi-mesa  libpoppler145          libpython3.12t64      ruby-zeitwerk
  libflac12t64   libicu-dev     libpython3.12-minimal  python3-setproctitle  strongswan
  libgeos3.13.0  liblbfgsb0     libpython3.12-stdlib   python3.12-tk
Use 'sudo apt autoremove' to remove them.

Installing:
  bind9  bind9-doc  bind9-utils

Suggested packages:
  bind-doc  resolvconf

Summary:
  Upgrading: 0, Installing: 3, Removing: 0, Not Upgrading: 7
  Download size: 3,991 kB
  Space needed: 9,284 kB / 63.4 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 bind9-utils amd64 1:9.20.7-1 [183 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 bind9 amd64 1:9.20.7-1 [250 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 bind9-doc all 1:9.20.7-1 [3,559 kB]
Fetched 3,991 kB in 7s (534 kB/s)
Selecting previously unselected package bind9-utils.
(Reading database ... 416714 files and directories currently installed.)
Preparing to unpack .../bind9-utils_1%3a9.20.7-1_amd64.deb ...
Unpacking bind9-utils (1:9.20.7-1) ...
Selecting previously unselected package bind9.
Preparing to unpack .../bind9_1%3a9.20.7-1_amd64.deb ...
Unpacking bind9 (1:9.20.7-1) ...
Selecting previously unselected package bind9-doc.
Preparing to unpack .../bind9-doc_1%3a9.20.7-1_all.deb ...
Unpacking bind9-doc (1:9.20.7-1) ...
Setting up bind9-doc (1:9.20.7-1) ...
Setting up bind9-utils (1:9.20.7-1) ...
Setting up bind9 (1:9.20.7-1) ...
wrote key file "/etc/bind/rndc.key"
update-rc.d: We have no instructions for the named init script.
update-rc.d: It looks like a network service, we disable it.
named-resolvconf.service is a disabled or a static unit, not starting it.
named.service is a disabled or a static unit, not starting it.
Processing triggers for ufw (0.36.2-9) ...
Processing triggers for man-db (2.13.0-1) ...
Processing triggers for kali-menu (2025.1.1) ...
```

Check the IP of the machine

```
┌──(luthfi㉿kali)-[~]
└─$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:53:31:bc brd ff:ff:ff:ff:ff:ff
    inet 192.168.8.129/24 brd 192.168.8.255 scope global dynamic noprefixroute eth0
       valid_lft 84252sec preferred_lft 84252sec
    inet6 2402:4000:23d0:2c88:b81e:66e0:9e82:c546/64 scope global dynamic noprefixroute
       valid_lft 234sec preferred_lft 54sec
    inet6 fe80::7ee2:7403:68db:cfc2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Step 2- Edit the /etc/bind/named.conf.options file.

```
 GNU nano 8.3                                 /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";

    // Forward DNS queries to an external DNS server (like Google)
    forwarders {
        8.8.8.8;
        8.8.4.4;
    };
```

Address of google DNS

Secondary Address of google DNS

Forwarders are used to when the BIND server cannot solve something it should send it to the DNS sever listed.

Check if BLIND server is active

```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl restart bind9
[sudo] password for luthfi:

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl status bind9
● named.service - BIND Domain Name Server
     Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: disabled)
     Active: active (running) since Wed 2025-04-30 10:17:42 +0530; 6s ago
 Invocation: deac051574ab419196890c1a46049bc7
       Docs: man:named(8)
   Main PID: 25898 (named)
     Status: "running"
      Tasks: 8 (limit: 3951)
     Memory: 27.7M (peak: 28.2M)
        CPU: 243ms
     CGroup: /system.slice/named.service
             └─25898 /usr/sbin/named -f -u bind

Apr 30 10:17:42 kali named[25898]: configuring command channel from '/etc/bind/rndc.key'
Apr 30 10:17:42 kali named[25898]: command channel listening on ::1#953
Apr 30 10:17:42 kali named[25898]: managed-keys-zone: loaded serial 35
Apr 30 10:17:42 kali named[25898]: zone 2.0.10.in-addr.arpa/IN: loaded serial 1
Apr 30 10:17:42 kali named[25898]: zone SNPAssignment.com/IN: loaded serial 2
Apr 30 10:17:42 kali named[25898]: all zones loaded
Apr 30 10:17:42 kali named[25898]: FIPS mode is disabled
Apr 30 10:17:42 kali named[25898]: running
Apr 30 10:17:42 kali named[25898]: managed-keys-zone: Key 20326 for zone . is now trusted (acceptance t>
Apr 30 10:17:42 kali named[25898]: managed-keys-zone: Key 38696 for zone . is now trusted (acceptance t>
lines 1-23/23 (END)
[1]+  Stopped                 sudo systemctl status bind9
```

Step 3 – Creating two zones. (one to forward and one to reverse the name)

Edit the /**etc/bind/db.SNP.com** file. (SNPAssignment is a domain created for demonstration) – Zone 1 (used to map the IP address to domain name)

```
luthfi@kali: ~
File  Actions  Edit  View  Help
  GNU nano 8.4                          /etc/bind/db.SNP.com *
$TTL    604800
@       IN      SOA     ns1.SNPAssignment.com. admin.SNPAssignment.com. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.SNPAssignment.com.
@       IN      A       192.168.8.129
ns1     IN      A       192.168.8.129
www     IN      A       192.168.8.129
```

Kali machine's IP

Step 4 – Create a file called **/etc/bind/db.10** and Edit it the details below. – Zone 2 (used to map the domain name to IP address)

```
  GNU nano 8.3                                    /etc/bind/db.10
$TTL    604800
@       IN      SOA     ns1.SNPAssignment.com. admin.SNPAssignment.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.SNPAssignment.com.
10      IN      PTR     ns1.SNPAssignment.com.
20      IN      PTR     www.SNPAssignment.com.
```

Step 5 – Linking the two zones created, in **/etc/bind/named.conf.local** file

```
File  Actions  Edit  View  Help
  GNU nano 8.3                                /etc/bind/named.conf.local
//
// Do any local configuration here
//

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


zone "2.0.10.in-addr.arpa" {
    type master;
    file "/etc/bind/db.10";
};
```

Step 6 – Check if the Bind server is running. (**sudo systemctl restart bind9 / sudo systemctl status bind9**)

```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl restart bind9
[sudo] password for luthfi:

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl status bind9
● named.service - BIND Domain Name Server
     Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: disabled)
     Active: active (running) since Wed 2025-04-30 10:46:01 +0530; 5s ago
 Invocation: a352a4cd41584c14a0b1bafd891095a7
       Docs: man:named(8)
   Main PID: 39654 (named)
     Status: "running"
      Tasks: 8 (limit: 3951)
     Memory: 29M (peak: 29.5M)
        CPU: 152ms
     CGroup: /system.slice/named.service
             └─39654 /usr/sbin/named -f -u bind

Apr 30 10:46:01 kali named[39654]: configuring command channel from '/etc/bind/rndc.key'
Apr 30 10:46:01 kali named[39654]: command channel listening on ::1#953
Apr 30 10:46:01 kali named[39654]: managed-keys-zone: loaded serial 36
Apr 30 10:46:01 kali named[39654]: zone 2.0.10.in-addr.arpa/IN: loaded serial 1
Apr 30 10:46:01 kali named[39654]: zone SNPAssignment.com/IN: loaded serial 2
Apr 30 10:46:01 kali named[39654]: all zones loaded
Apr 30 10:46:01 kali named[39654]: FIPS mode is disabled
Apr 30 10:46:01 kali named[39654]: running
Apr 30 10:46:01 kali named[39654]: managed-keys-zone: Key 20326 for zone . is now trusted (acceptance t
Apr 30 10:46:01 kali named[39654]: managed-keys-zone: Key 38696 for zone . is now trusted (acceptance t
lines 1-23/23 (END)
```

Step 7 – Enable the ufw (firewall) and all the Bind rule.

```
┌──(luthfi㉿kali)-[~]
└─$ sudo ufw enable
Firewall is active and enabled on system startup

┌──(luthfi㉿kali)-[~]
└─$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
67                         ALLOW       Anywhere
10000                      ALLOW       Anywhere
67 (v6)                    ALLOW       Anywhere (v6)
10000 (v6)                 ALLOW       Anywhere (v6)


┌──(luthfi㉿kali)-[~]
└─$ sudo ufw allow bind9
Rule added
Rule added (v6)

┌──(luthfi㉿kali)-[~]
└─$ sudo ufw reload
Firewall reloaded

┌──(luthfi㉿kali)-[~]
└─$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
67                         ALLOW       Anywhere
10000                      ALLOW       Anywhere
Bind9                      ALLOW       Anywhere
67 (v6)                    ALLOW       Anywhere (v6)
10000 (v6)                 ALLOW       Anywhere (v6)
Bind9 (v6)                 ALLOW       Anywhere (v6)
```

- Check if the zone name is correct.

```
┌──(luthfi㉿kali)-[~]
└─$ sudo named-checkzone SNPAssignment.com /etc/bind/db.SNP.com
zone SNPAssignment.com/IN: loaded serial 2
OK
```

Step 8 – Test the DNS server Created.

dig @192.168.8.129 SNPAssignment.com (dig - Domain Information Groper it would find the IP address associated to the domain name and returns the IP address)

```
┌──(luthfi㉿kali)-[~]
└─$ dig @192.168.8.129 SNPAssignment.com

; <<>> DiG 9.20.7-1-Debian <<>> @192.168.8.129 SNPAssignment.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 35289
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 674cfa7fd3f19f37010000006811b375b9ca9d959304c712 (good)
;; QUESTION SECTION:
;SNPAssignment.com.             IN      A

;; ANSWER SECTION:
SNPAssignment.com.      604800  IN      A       192.168.8.129

;; Query time: 32 msec
;; SERVER: 192.168.8.129#53(192.168.8.129) (UDP)
;; WHEN: Wed Apr 30 10:51:57 +0530 2025
;; MSG SIZE  rcvd: 90
```

c. Installation and Configuration of NTP Client

NTP (Network Time Protocol) is a protocol that is used to synchronize the system time of devices through the internet.

Importance of NTP

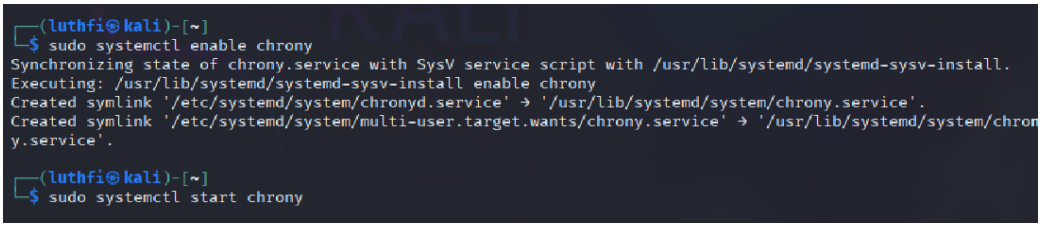
- Ensures if security protocols function properly.
- Maintains the accuracy of time.
- Keeps time stamps consistent.

Since NTP client doesn't work properly, I have used chrony (modern NTP client)

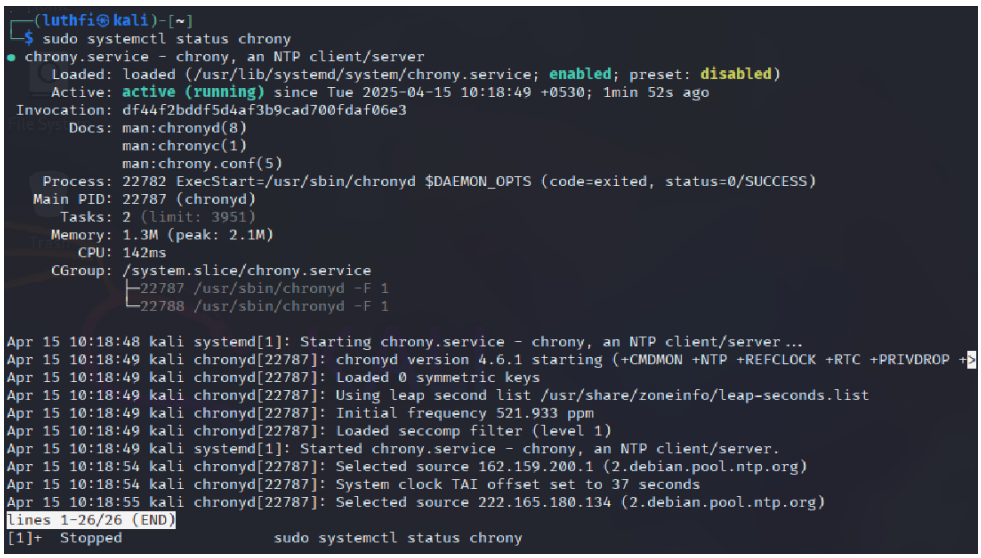
Step 1 – Installing a chrony

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt install chrony -y
Installing:
  chrony

Suggested packages:
  networkd-dispatcher

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 20
  Download size: 309 kB
  Space needed: 705 kB / 63.4 GB available

Get:1 http://mirror.freedif.org/kali kali-rolling/main amd64 chrony amd64 4.6.1-2 [309 kB]
Fetched 309 kB in 2s (165 kB/s)
Selecting previously unselected package chrony.
(Reading database ... 415832 files and directories currently installed.)
Preparing to unpack .../chrony_4.6.1-2_amd64.deb ...
Unpacking chrony (4.6.1-2) ...
Setting up chrony (4.6.1-2) ...
Creating config file /etc/chrony/chrony.conf with new version
Creating config file /etc/chrony/chrony.keys with new version
dpkg-statoverride: warning: --update given but /var/log/chrony does not exist
update-rc.d: We have no instructions for the chrony init script.
update-rc.d: It looks like a network service, we disable it.
chrony.service is a disabled or a static unit, not starting it.
Processing triggers for kali-menu (2025.1.1) ...
Processing triggers for man-db (2.13.0-1) ...
```

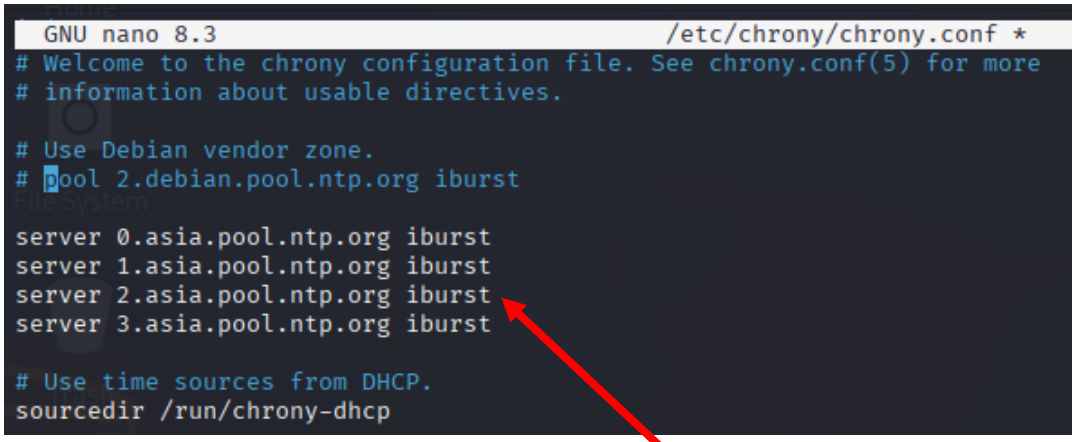
Step 2 – Enabling chrony and starting the server.



Step 3 – Check the status of chrony. (**sudo systemctl status chrony**)

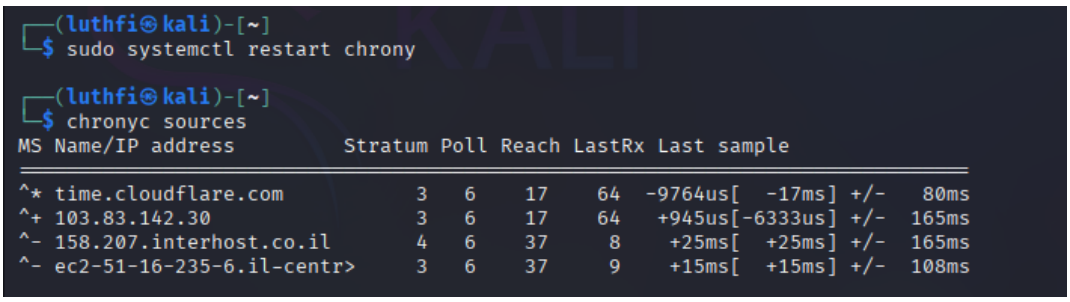


Step 4 – Configure the **/etc/chrony/chrony.conf** file.

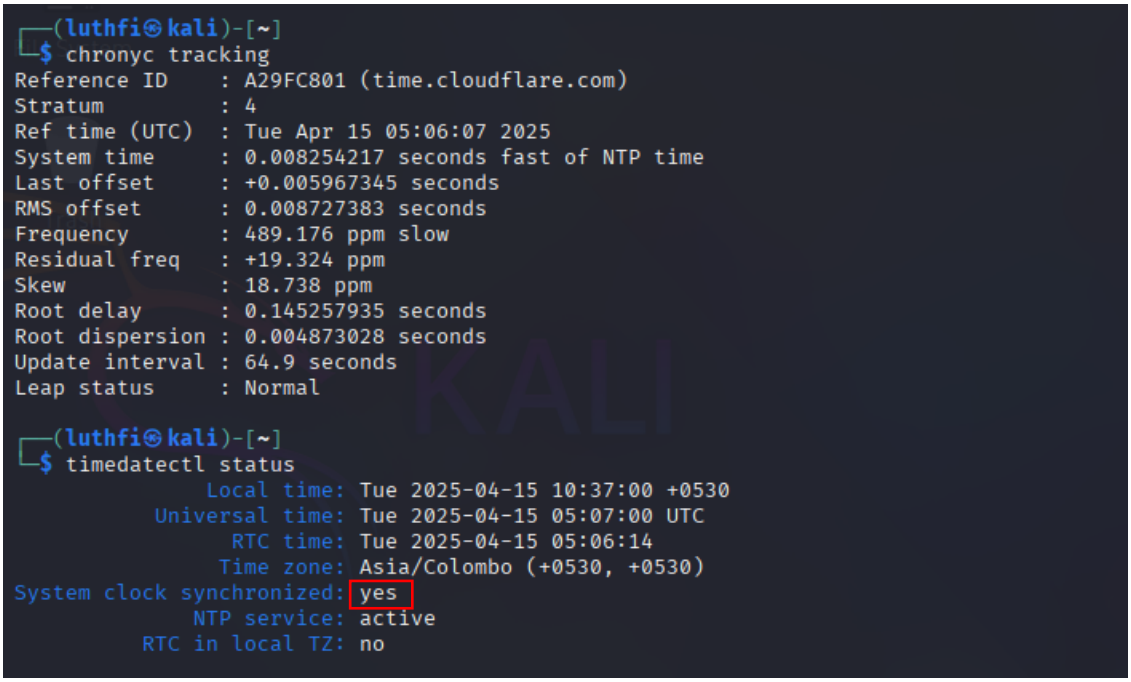


NTP servers closer to Sri Lanka

Step 5 – Restart and check if the server works properly. (**sudo systemctl restart chrony**)



Step 6 – Check the status to confirm if the clock has been synchronized. (**chronyc tracking / timedatectl status**)



d. Importance of Network Services in Network Administration

DHCP - Dynamic Host Configuration Protocol

Need – Automates the process of assigning IP address since, assigning manually is time consuming and errors can be caused.

Uses

- Reduces the possibility of errors happening.
- Simplifies the process of adding new devices to the network.
- Automatically assigns IP addresses, subnet masks and default gateways.
- Enable efficient IP addressing mechanism through reuse and having a lease time.

DNS - Domain Name System

Need – Computers cannot understand if names (ex- courseweb.sliit.lk) are used to communicate when accessing the web and IP addresses (ex- 192.168.17.20) cannot be used by humans because it is not human-friendly, so as a solution DNS is used to map the domain and IP address.

Uses

- Easy to access websites because of the friendly names.
- Distributes the traffic across multiple DNS servers improve performance and reliability.
- Filters content to block any harmful websites.
- DNS queries can be used to monitor any malicious activity.

NTP - Network Time Protocol

Need – Time is very critical for logging, file sharing and for security so time synchronization is crucial.

Uses

- Ensures accurate time for logs.
- Keeps the system clock synchronized in the network.
- Helps to resolve errors in a network aligning to event times.
- Ensures all automated tasks run at the correct time.

3. Security and other servers

a. Shell Scripting

Shell scripting is a set of commands which is executed in the shell, which is a command line interpreter.

```
#!/bin/bash
#Tells the system to use bash(shebang)

#Declared Variables
logs="/var/log/custom_logs"
backup_file="logs_backup.tar.gz"
backup_loc="/var/log"
system_infor_file="/var/log/system_info.log"
temp_deleted_files="/tmp/deleted_logs.txt"
temp_archived_files="/tmp/archived_logs.txt"


# Clear previous temporary files if they exsits
> "$temp_deleted_files"
> "$temp_archived_files"

# i) Delete .log files older than 7 days
echo "Deleting log files older than 7 days..."

#searches the .log files stored in logs. -mtime(modifed more than 7 days ago) and saves it to the temporary file
find "$logs" -name "*.log" -type f -mtime +7 -print -delete > "$temp_deleted_files"

# ii) Archive remaining .log files
echo "Archiving remaining log files..."

#search all .log files which is remaining and save them to $temp_archived_files.
find "$logs" -name "*.log" -type f > "$temp_archived_files"

# use tar to compress them into .tar.gz archive file.  -czf creates(c), compress(z), output file(f)
tar -czf "$backup_loc/$backup_file" -T "$temp_archived_files" # -T reads the list of files including txt files

# ── (System Info) Create a system details report ──
{
    echo "===== System Details Report ====="
    echo "Date: $(date)"
    echo "Uptime: $(uptime -p)" #How long the system has been running
    echo "Free Memory:"
    free -h #available memory
    echo "Disk Usage:"
    df -h  #disk usage
    echo "================================="
} | tee  "$system_infor_file" # all are saved and printed (tee)

# iii) Summary
echo ""
echo "===== Summary ====="
echo "Deleted Log Files:"
cat "$temp_deleted_files"
echo ""
echo "Archived Log Files:"
cat "$temp_archived_files"
echo ""
echo "System details saved to $system_infor_file"
echo "Backup archive created at $backup_loc/$backup_file"
echo "==================="

# Deletes the temporary files used for storing the deleted and archived file names.(-f refers to force)
 rm -f "$temp_deleted_files" "$temp_archived_files"
```

Log_cleanip.sh file

```
#!/bin/bash

#Tells the system to use bash(shebang)

#Declared Variables

logs="/var/log/custom_logs"

backup_file="logs_backup.tar.gz"

backup_loc="/var/log"

system_infor_file="/var/log/system_info.log"

temp_deleted_files="/tmp/deleted_logs.txt"

temp_archived_files="/tmp/archived_logs.txt"

# Clear previous temporary files if they exsits

> "$temp_deleted_files"

> "$temp_archived_files"


# i) Delete .log files older than 7 days

echo "Deleting log files older than 7 days..."

#searches the .log files stored in logs. -mtime(modifed more than 7 days ago) and saves it to the temporary file

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


# ii) Archive remaining .log files

echo "Archiving remaining log files..."

#search all .log files which is remaining and save them to $temp_archived_files.

find "$logs" -name "*.log" -type f > "$temp_archived_files"

# use tar to compress them into .tar.gz archive file.  -czf creates(c), compress(z), output file(f)

tar -czf "$backup_loc/$backup_file" -T "$temp_archived_files" # -T reads the list of files including txt files
```

```
# --- (System Info) Create a system details report ---
{
    echo "===== System Details Report ====="
    echo "Date: $(date)"
    echo "Uptime: $(uptime -p)" #How long the system has been running
    echo "Free Memory:"
    free -h #available memory
    echo "Disk Usage:"
    df -h  #disk usage
    echo "================================="
} | tee "$system_infor_file" # all are saved and printed(tee)


# iii) Summary
echo ""
echo "===== Summary ====="
echo "Deleted Log Files:"
cat "$temp_deleted_files"
echo ""
echo "Archived Log Files:"
cat "$temp_archived_files"
echo ""
echo "System details saved to $system_infor_file"
echo "Backup archive created at $backup_loc/$backup_file"
echo "==================="
# Deletes the temporary files used for storing the deleted and archived file names.(-f refers to force)
 rm -f "$temp_deleted_files" "$temp_archived_files"
```

To execute it we have to change the file permissions.

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ chmod +x log_cleanup.sh

┌──(luthfi㉿kali)-[~/SNP]
└─$ ls -l log_cleanup.sh
-rwxrwxr-x 1 luthfi luthfi 1964 Apr 15 16:26 log_cleanup.sh
```

Output of log_cleanup.sh file

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ sudo ./log_cleanup.sh
Deleting log files older than 7 days ...
Archiving remaining log files ...
tar: Removing leading `/' from member names
tar: Removing leading `/' from hard link targets
===== System Details Report =====
Date: Tue Apr 15 06:08:52 PM +0530 2025
Uptime: up 6 hours, 23 minutes
Free Memory:
               total        used        free      shared  buff/cache   available
Mem:           3.3Gi       1.2Gi       1.4Gi        24Mi       1.1Gi       2.2Gi
Swap:          1.0Gi          0B       1.0Gi
Disk Usage:
Filesystem      Size  Used Avail Use% Mounted on
udev            1.7G     0  1.7G   0% /dev
tmpfs           342M  992K  341M   1% /run
/dev/sda1        79G   16G   60G  21% /
tmpfs           1.7G  4.0K  1.7G   1% /dev/shm
tmpfs           1.0M     0  1.0M   0% /run/credentials/systemd-journald.service
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           1.7G   16K  1.7G   1% /tmp
tmpfs           1.0M     0  1.0M   0% /run/credentials/getty@tty1.service
tmpfs           342M  128K  342M   1% /run/user/1001
==================================

===== Summary =====
Deleted Log Files:

Archived Log Files:
/var/log/custom_logs/recent.log

System details saved to /var/log/system_info.log
Backup archive created at /var/log/logs_backup.tar.gz
====================
```

Scheduling the cron job (-e *edit,* -l *list)*

- Opens the crontab file to make changes like edit, delete and add.

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ crontab -e
no crontab for luthfi - using an empty one
Select an editor.  To change later, run select-editor again.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.basic
  3. /usr/bin/vim.tiny

Choose 1-3 [1]: 1
crontab: installing new crontab

┌──(luthfi㉿kali)-[~/SNP]
└─$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command

# (0 0) to run it at midnight
# (* * 0) to schedule it run at sunday - day 0

0 0 * * 0 sudo /home/luthfi/SNP/log_cleanup.sh
```

b. Configuration of SSH Server

An SSH server (secure Shell) is a program that is used to allow secure remote access to a computer or a device. It enables a device to connect over the network to manage file or execute a command remotely.

VM1 – Kali Linux (SSH Sever)

VM2 – Kali Linux Cloned (SSH Client)

Step 1 - Install and start the SSH server (VM1)

-sudo apt install openssh-server

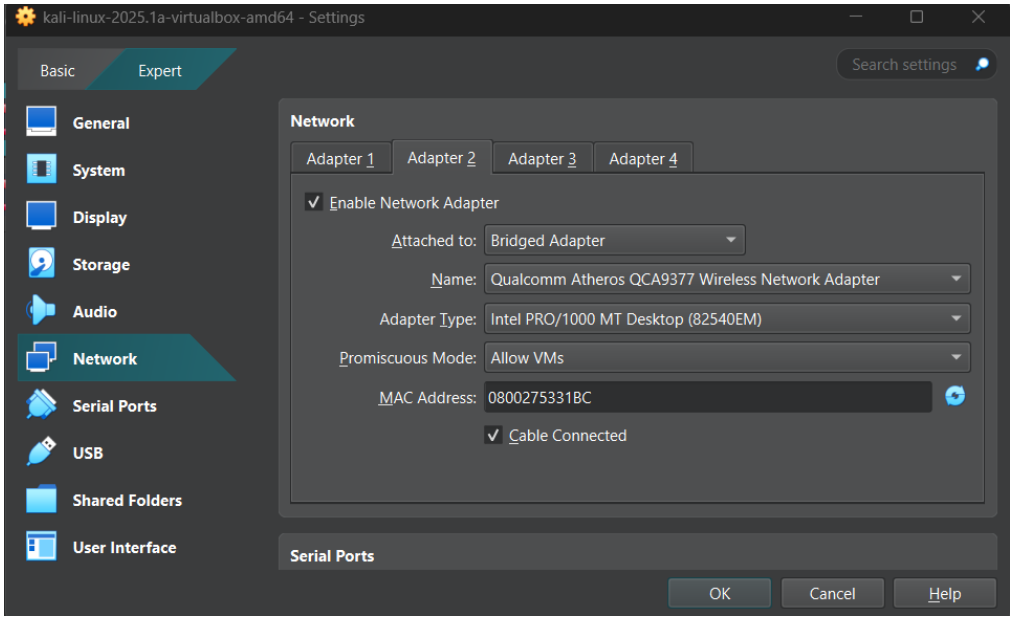
-sudo systemctl enable ssh

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt install openssh-server
openssh-server is already the newest version (1:9.9p2-1).
openssh-server set to manually installed.
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 25

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl start ssh

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
Created symlink '/etc/systemd/system/sshd.service' → '/usr/lib/systemd/system/ssh.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/ssh.service' → '/usr/lib/systemd/system/ssh.service'.
```

Step 1 - Check the status of the SSH server (VM1) - **sudo systemctl status ssh**

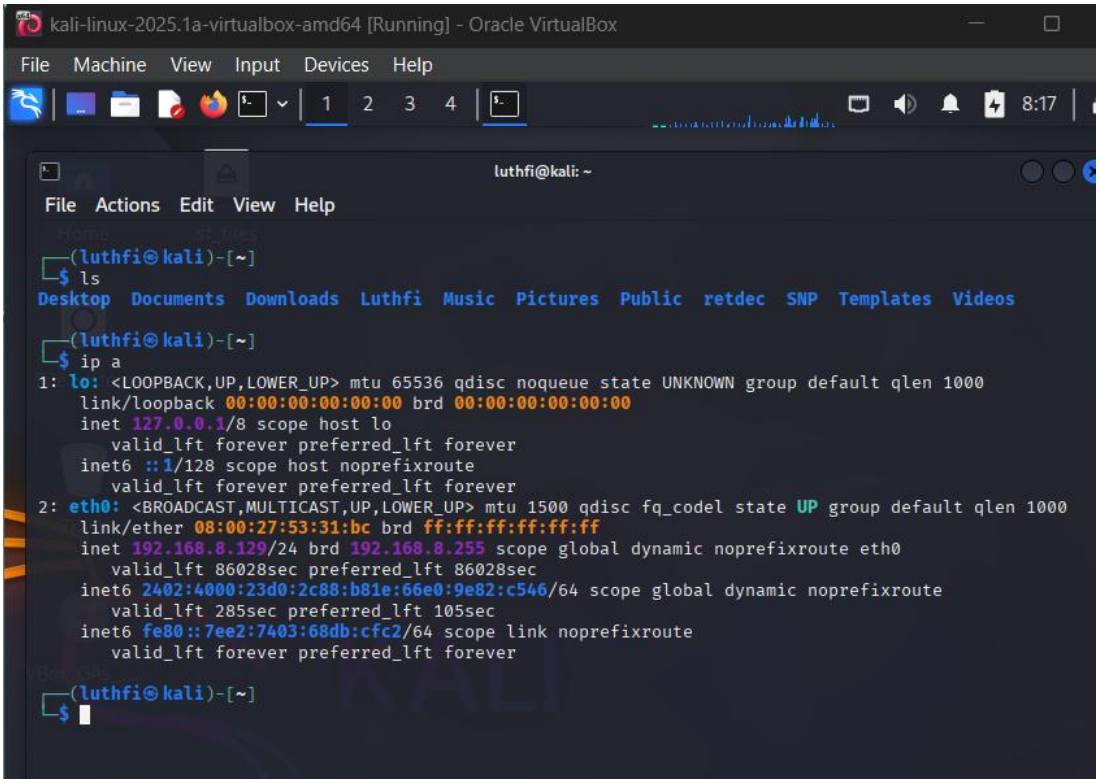
```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-04-15 18:33:14 +0530; 1min 28s ago
 Invocation: 373294318aa44c9989e1f9904135617f
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 193861 (sshd)
      Tasks: 1 (limit: 3951)
     Memory: 2M (peak: 2.4M)
        CPU: 124ms
     CGroup: /system.slice/ssh.service
             └─193861 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Apr 15 18:33:14 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server ...
Apr 15 18:33:14 kali sshd[193861]: Server listening on 0.0.0.0 port 22.
Apr 15 18:33:14 kali sshd[193861]: Server listening on :: port 22.
Apr 15 18:33:14 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
```

Step 3 – Change the both VM1 and VM2 to **Bridged Adapter** to connect to SSH server.

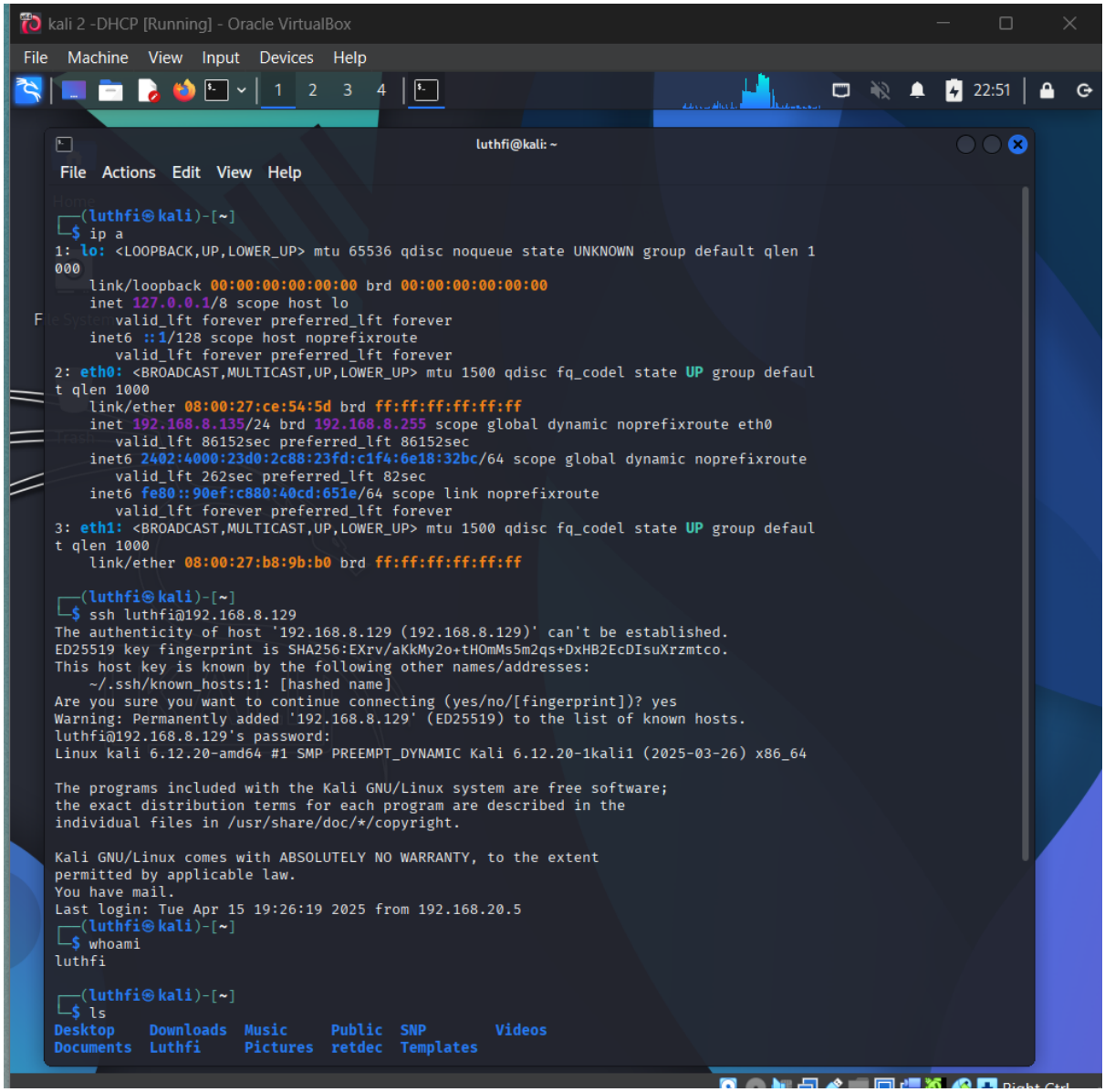


Step 4 - check the IP of VM1 and make the connection.

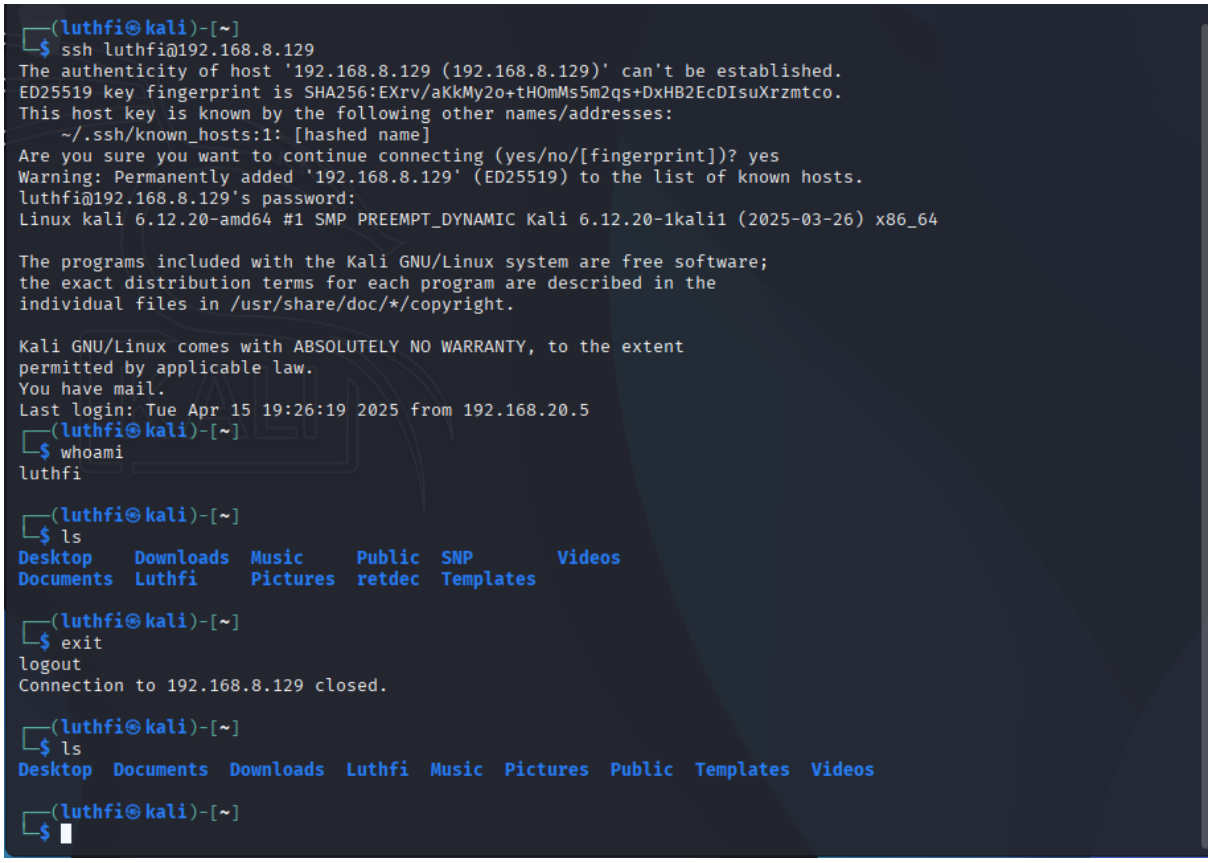
VM1 (server)



VM2 (Client)



To establish the connection you have to use **ssh username@ip** address of the other machine.



So here as a demonstration **ls** command is used when the connection is established, we could view the host machine's files.

c. iptables and ACLs

- iptables are used to configure the firewall rules and manage the traffic in the network.

Step 1

- Installing iptables. - **sudo apt-get install iptables iptables-persistent**

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt-get  install iptables iptables-persistent
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
iptables is already the newest version (1.8.11-2).
The following package was automatically installed and is no longer required:
  libfuse3-3
Use 'sudo apt autoremove' to remove it.
The following packages will be REMOVED:
  ufw
The following NEW packages will be installed:
  iptables-persistent netfilter-persistent
0 upgraded, 2 newly installed, 1 to remove and 6 not upgraded.
Need to get 18.5 kB of archives.
After this operation, 783 kB disk space will be freed.
Do you want to continue? [Y/n] y
Get:1 http://xsrv.moratelindo.io/kali kali-rolling/main amd64 netfilter-persistent all 1.0.23 [7,948
B]
Get:2 http://mirror.primelink.net.id/kali kali-rolling/main amd64 iptables-persistent all 1.0.23 [10.
5 kB]
Fetched 18.5 kB in 2s (10.9 kB/s)
Preconfiguring packages ...
(Reading database ... 446681 files and directories currently installed.)
Removing ufw (0.36.2-9) ...
Skip stopping firewall: ufw (not enabled)
Selecting previously unselected package netfilter-persistent.
(Reading database ... 446586 files and directories currently installed.)
Preparing to unpack .../netfilter-persistent_1.0.23_all.deb ...
Unpacking netfilter-persistent (1.0.23) ...
Selecting previously unselected package iptables-persistent.
Preparing to unpack .../iptables-persistent_1.0.23_all.deb ...
Unpacking iptables-persistent (1.0.23) ...
Setting up netfilter-persistent (1.0.23) ...
update-rc.d: We have no instructions for the netfilter-persistent init script.
update-rc.d: It looks like a non-network service, we enable it.
netfilter-persistent.service is a disabled or a static unit, not starting it.
Setting up iptables-persistent (1.0.23) ...
Processing triggers for man-db (2.13.0-1) ...
Processing triggers for kali-menu (2025.2.0) ...
```

Step 2

- Dropping all incoming connections and allowing all outgoing connections.

  -sudo iptables -P INPUT DROP (-p means the protocol)
  -sudo iptables -P OUTPUT ACCEPT

```
┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -P INPUT DROP

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -P OUTPUT ACCEPT
```

Step 3

Blocking HTTP traffic and allowing HTTPS traffic.

To allow HTTPS requests

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

-sudo iptables -A INPUT -p tcp --sport 443 -j ACCEPT

To block HTTP requests

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

-sudo iptables -A INPUT -p tcp --sport 80 -j REJECT

- -A OUTPUT - appends rule to the output .
- -p tcp - maps to TCP protocol
- --dport 443 - maps to the destination port 443
- -j ACCEPT – jumps to accept incoming packets.
- --sport 443 -maps the source port 443
- -j REJECT – jump to reject incoming packets.

```
┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A INPUT -p tcp --sport 443 -j ACCEPT

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -p tcp --dport 80 -j REJECT

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A INPUT -p tcp --sport 80 -j REJECT
```

Step 4

- Check the iptables to see if the rules are added - **sudo iptables -L**

```
┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp spt:https
REJECT     tcp  --  anywhere             anywhere             tcp spt:http reject-with icmp-port-unreachable

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
REJECT     tcp  --  anywhere             anywhere             tcp dpt:http reject-with icmp-port-unreachable
```

Step 5 – Blocking the Ips of the social medias

-m means string, --algo bm specifies the algorithm , -j jump to target.

sudo iptables -A OUTPUT -m string --string "facebook.com" --algo bm -j DROP

sudo iptables -A OUTPUT -m string --string "instagram.com" --algo bm -j DROP

sudo iptables -A OUTPUT -m string --string "twitter.com" --algo bm -j DROP

sudo iptables -A OUTPUT -m string --string "youtube.com" --algo bm -j DROP

sudo iptables -A OUTPUT -m string --string "threads.net" --algo bm -j DROP

```
┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -m string --string "facebook.com" --algo bm -j DROP

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -m string --string "instagram.com" --algo bm -j DROP

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -m string --string "twitter.com" --algo bm -j DROP

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -m string --string "youtube.com" --algo bm -j DROP

┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -A OUTPUT -m string --string "threads.net" --algo bm -j DROP
```

Step 6

- Check the iptables to see if the rules are added - **sudo iptables -L**

```
┌──(luthfi㉿kali)-[~]
└─$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp spt:https
REJECT     tcp  --  anywhere             anywhere             tcp spt:http reject-with icmp-port-unreachable
DROP       all  --  anywhere             anywhere             STRING match  "facebook.com" ALGO name bm

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  anywhere             edge-star-mini-shv-02-bom2.facebook.com  reject-with icmp-port-unrea
chable

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:https
REJECT     tcp  --  anywhere             anywhere             tcp dpt:http reject-with icmp-port-unreachable
DROP       all  --  anywhere             anywhere             STRING match  "facebook.com" ALGO name bm
DROP       all  --  anywhere             anywhere             STRING match  "instagram.com" ALGO name bm
DROP       all  --  anywhere             anywhere             STRING match  "twitter.com" ALGO name bm
DROP       all  --  anywhere             anywhere             STRING match  "youtube.com" ALGO name bm
DROP       all  --  anywhere             anywhere             STRING match  "threads.net" ALGO name bm
```

d. Installation and Configuration of a Web Server

A web server is a software that is used to deliver web pages over the internet.

Step 1- Installing Apache.

```
┌──(luthfi㉿kali)-[~]
└─$ sudo apt install apache2 -y
apache2 is already the newest version (2.4.63-1).
apache2 set to manually installed.
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 25
```

Step 2- Start the Apache server. (**sudo systemctl start apache2**)

```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl start apache2

┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
Created symlink '/etc/systemd/system/multi-user.target.wants/apache2.service' → '/usr/lib/systemd/system/ap
ache2.service'.
```

Step 3- Allow Apache from the firewall.

```
┌──(luthfi㉿kali)-[~]
└─$ sudo ufw allow 'Apache'
Rules updated
Rules updated (v6)

┌──(luthfi㉿kali)-[~]
└─$ sudo ufw enable
Firewall is active and enabled on system startup
```

Step 4- Creating a Web page.(Location- **/var/www/html/index.html**)

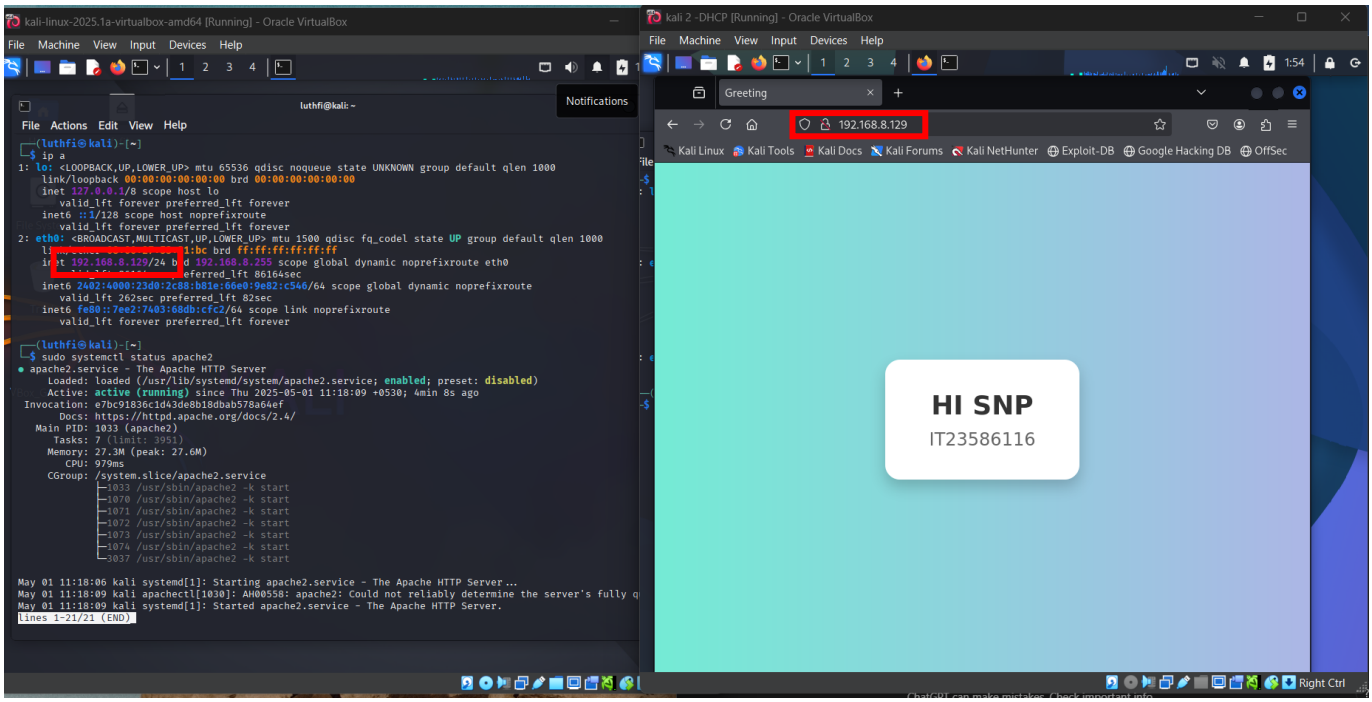
```
 GNU nano 8.3                                    /var/www/html/index.html
<!DOCTYPE html>
<html>
<head>
  <title>Greeting</title>
  <style>
    body {
      background: linear-gradient(to right, #74ebd5, #acb6e5);
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      margin: 0;
    }

    .card {
      background-color: white;
      padding: 40px 60px;
      border-radius: 20px;
      box-shadow: 0 10px 20px rgba(0, 0, 0, 0.2);
      text-align: center;
    }

    .title {
      font-size: 36px;
      font-weight: bold;
      color: #333;
    }

    .subtitle {
      font-size: 24px;
      color: #666;
      margin-top: 10px;
    }
  </style>
</head>
<body>
  <div class="card">
    <div class="title">HI SNP</div>
    <div class="subtitle">IT23586116</div>
  </div>
</body>
</html>
```

Step 5- Check the status of the server. (**sudo systemctl status apache2**)

```
┌──(luthfi㉿kali)-[~]
└─$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-04-15 22:50:38 +0530; 4min 43s ago
 Invocation: 720ef2032c7246f1a8f67cd51142eb70
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 87882 (apache2)
      Tasks: 6 (limit: 3951)
     Memory: 21.1M (peak: 21.6M)
        CPU: 469ms
     CGroup: /system.slice/apache2.service
             ├─87882 /usr/sbin/apache2 -k start
             ├─87884 /usr/sbin/apache2 -k start
             ├─87885 /usr/sbin/apache2 -k start
             ├─87886 /usr/sbin/apache2 -k start
             ├─87887 /usr/sbin/apache2 -k start
             └─87888 /usr/sbin/apache2 -k start

Apr 15 22:50:37 kali systemd[1]: Starting apache2.service - The Apache HTTP Server...
Apr 15 22:50:38 kali apachectl[87873]: AH00558: apache2: Could not reliably determine the server's fully q>
Apr 15 22:50:38 kali systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-20/20 (END)
```

Step 6 – Accessing the website from another machine (Kali Cloned)

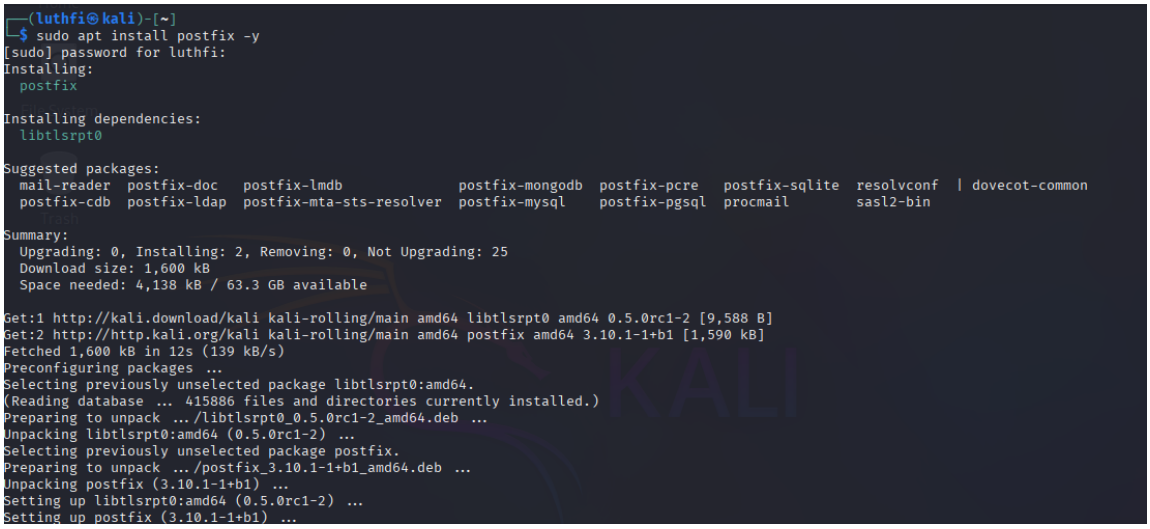
http://192.168.8.129

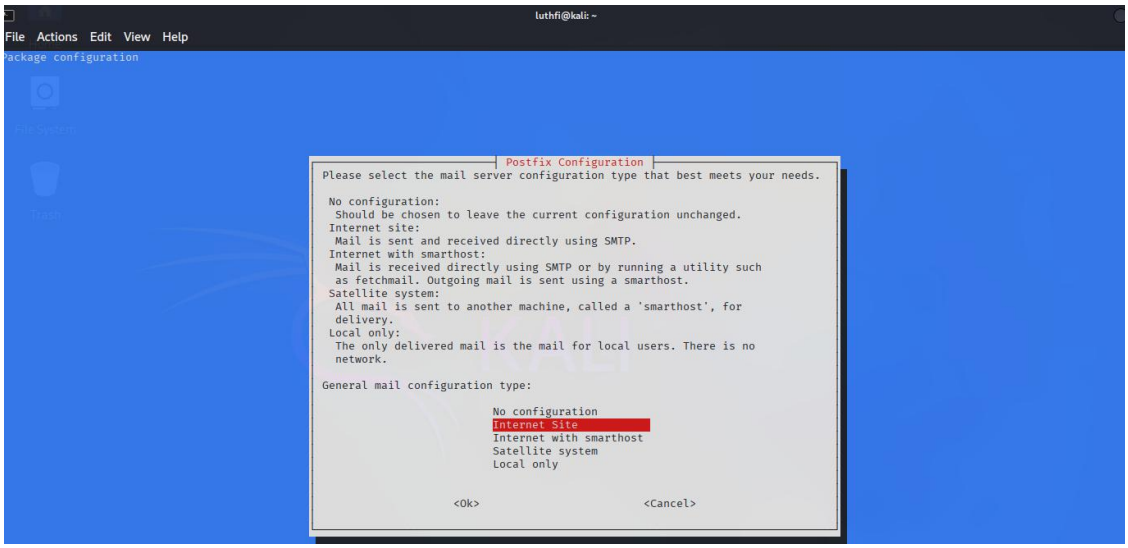


e. Installation and demonstration of an Email Server

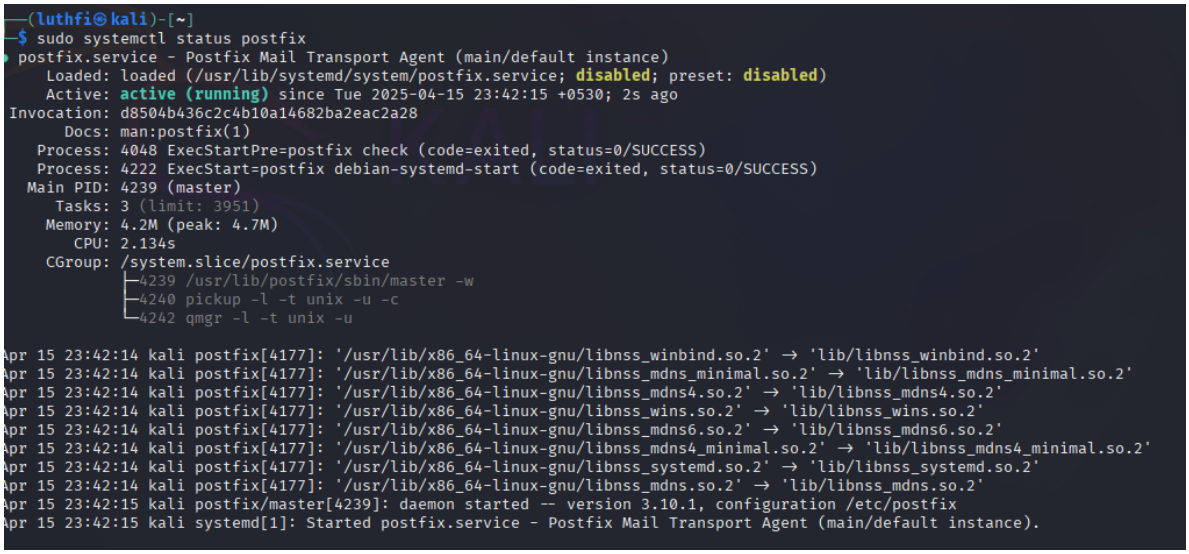
- An email server is a software that sends, receives and manages email messages.

Step 1 – Installing Postfix (**sudo apt install postfix -y**)

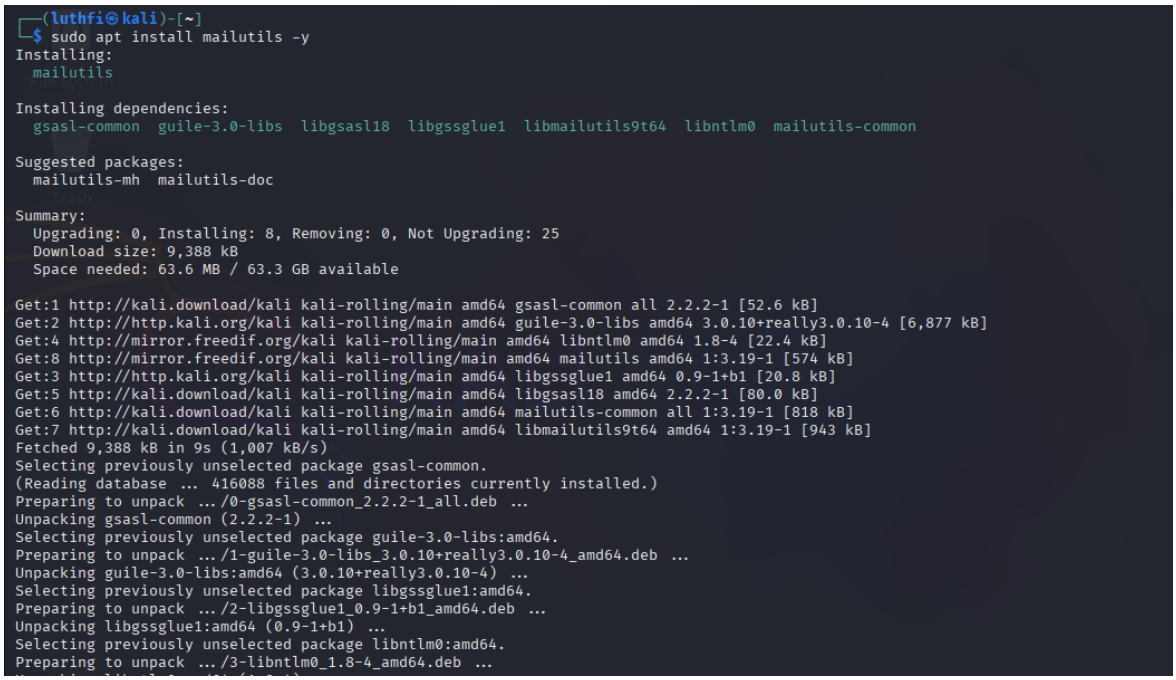




Step 2- Check the status of the server. (**sudo systemctl status postfix**)

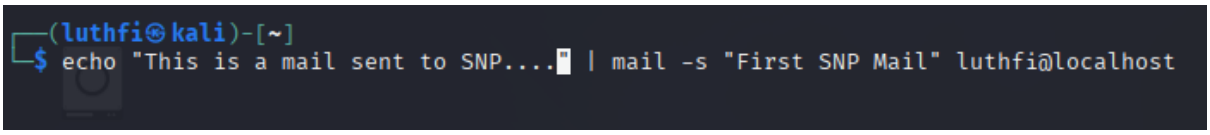


Step 3- To test sending mails install mailutils.

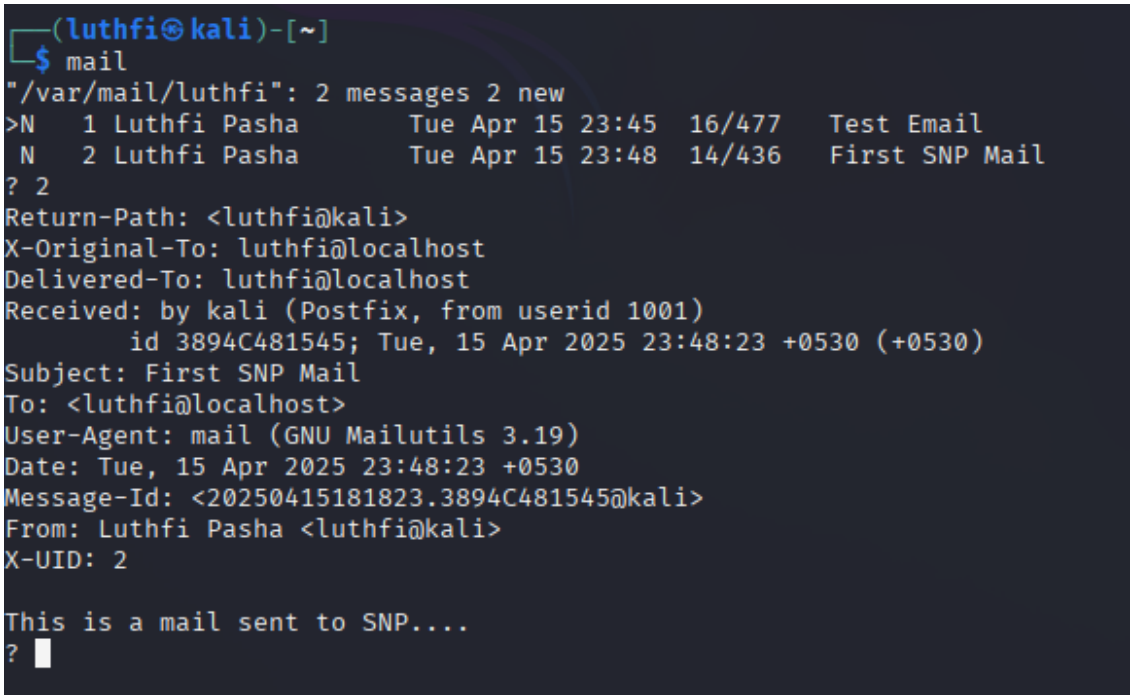


- Provides tools to manage mails.

Step 4- Sending a mail.



Step 5- Reading the mail.



4. Linux GDB

a. Execution Process

- Checking the suitable system.

```
┌──(luthfi㉿kali)-[~]
└─$ uname -m
x86_64
```

Step 1 – Execute the file.

```
┌──(luthfi㉿kali)-[~/Desktop]
└─$ ./x86_64
bash: ./x86_64: Permission denied

┌──(luthfi㉿kali)-[~/Desktop]
└─$ sudo ./x86_64
[sudo] password for luthfi:
sudo: ./x86_64: command not found

┌──(luthfi㉿kali)-[~/Desktop]
└─$ chmod +x x86_64

┌──(luthfi㉿kali)-[~/Desktop]
└─$ sudo ./x86_64
Enter the student IT number: IT23586116
```

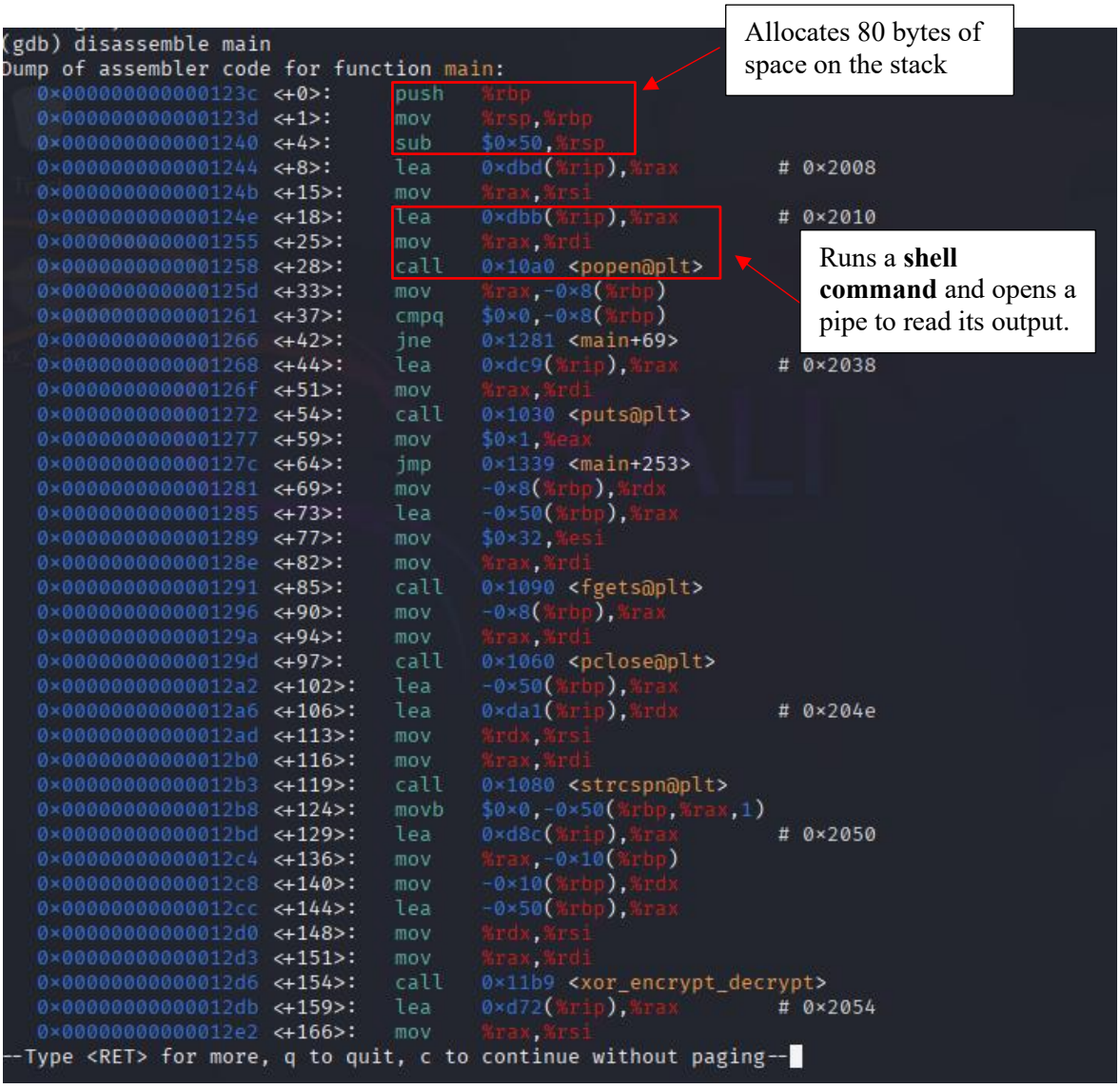
Step 2 – Running the newly created file.

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ ./IT23586116
[sudo] password for luthfi:

┌──(luthfi㉿kali)-[~/SNP 1]
└─$ ls -l
total 24
-rw-rw-r-- 1 luthfi luthfi    36 May  2 22:15 data.txt
-rwxr-xr-x 1 luthfi luthfi 20288 Apr 16 00:00 IT23586116

┌──(luthfi㉿kali)-[~/SNP 1]
└─$ cat data.txt
      R RVMZTYP@XHN_Q]N_WT    NZWJSV
```

b. Debugging Process

I. Step-by-step debugging using GDB.

- ./IT23586116 file is opened through the GDB debugger and has **disassemble main.**



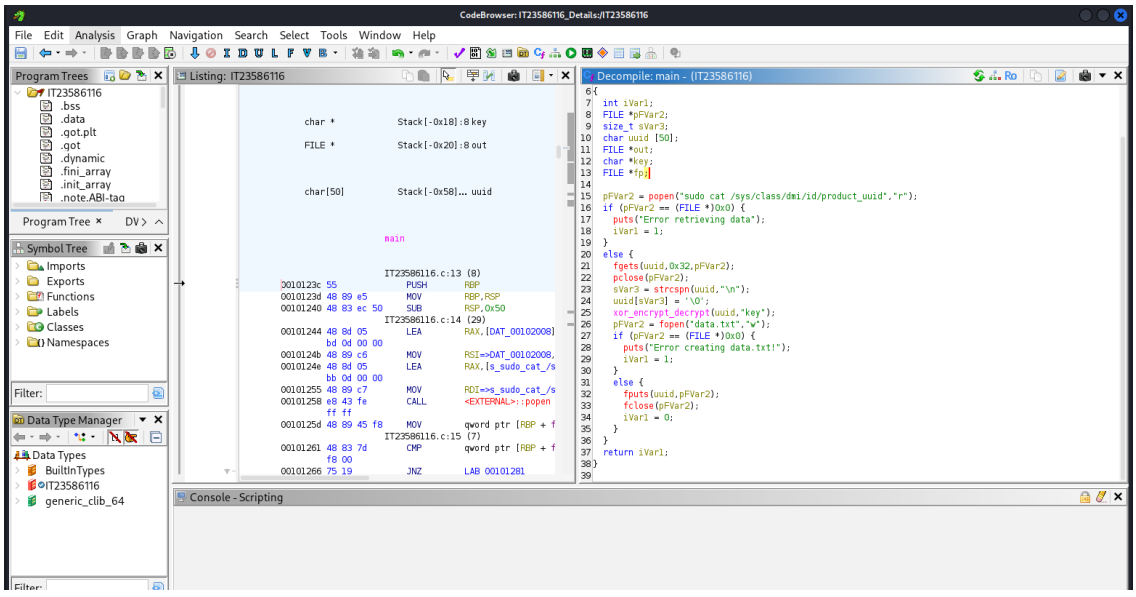

```
(gdb) disassemble main
Dump of assembler code for function main:
   0x000000000000123c <+0>:     push   %rbp
   0x000000000000123d <+1>:     mov    %rsp,%rbp
   0x0000000000001240 <+4>:     sub    $0x50,%rsp
   0x0000000000001244 <+8>:     lea    0xdbd(%rip),%rax        # 0x2008
   0x000000000000124b <+15>:    mov    %rax,%rsi
   0x000000000000124e <+18>:    lea    0xdbb(%rip),%rax        # 0x2010
   0x0000000000001255 <+25>:    mov    %rax,%rdi
   0x0000000000001258 <+28>:    call   0x10a0 <popen@plt>
   0x000000000000125d <+33>:    mov    %rax,-0x8(%rbp)
   0x0000000000001261 <+37>:    cmpq   $0x0,-0x8(%rbp)
   0x0000000000001266 <+42>:    jne    0x1281 <main+69>
   0x0000000000001268 <+44>:    lea    0xdc9(%rip),%rax        # 0x2038
   0x000000000000126f <+51>:    mov    %rax,%rdi
   0x0000000000001272 <+54>:    call   0x1030 <puts@plt>
   0x0000000000001277 <+59>:    mov    $0x1,%eax
   0x000000000000127c <+64>:    jmp    0x1339 <main+253>
   0x0000000000001281 <+69>:    mov    -0x8(%rbp),%rdx
   0x0000000000001285 <+73>:    lea    -0x50(%rbp),%rax
   0x0000000000001289 <+77>:    mov    $0x32,%esi
   0x000000000000128e <+82>:    mov    %rax,%rdi
   0x0000000000001291 <+85>:    call   0x1090 <fgets@plt>
   0x0000000000001296 <+90>:    mov    -0x8(%rbp),%rax
   0x000000000000129a <+94>:    mov    %rax,%rdi
   0x000000000000129d <+97>:    call   0x1060 <pclose@plt>
   0x00000000000012a2 <+102>:   lea    -0x50(%rbp),%rax
   0x00000000000012a6 <+106>:   lea    0xda1(%rip),%rdx        # 0x204e
   0x00000000000012ad <+113>:   mov    %rdx,%rsi
   0x00000000000012b0 <+116>:   mov    %rax,%rdi
   0x00000000000012b3 <+119>:   call   0x1080 <strcspn@plt>
   0x00000000000012b8 <+124>:   movb   $0x0,-0x50(%rbp,%rax,1)
   0x00000000000012bd <+129>:   lea    0xd8c(%rip),%rax        # 0x2050
   0x00000000000012c4 <+136>:   mov    %rax,-0x10(%rbp)
   0x00000000000012c8 <+140>:   mov    -0x10(%rbp),%rdx
   0x00000000000012cc <+144>:   lea    -0x50(%rbp),%rax
   0x00000000000012d0 <+148>:   mov    %rdx,%rsi
   0x00000000000012d3 <+151>:   mov    %rax,%rdi
   0x00000000000012d6 <+154>:   call   0x11b9 <xor_encrypt_decrypt>
   0x00000000000012db <+159>:   lea    0xd72(%rip),%rax        # 0x2054
   0x00000000000012e2 <+166>:   mov    %rax,%rsi
--Type <RET> for more, q to quit, c to continue without paging--
```

- An attempt to retrieving the source code from executable file.

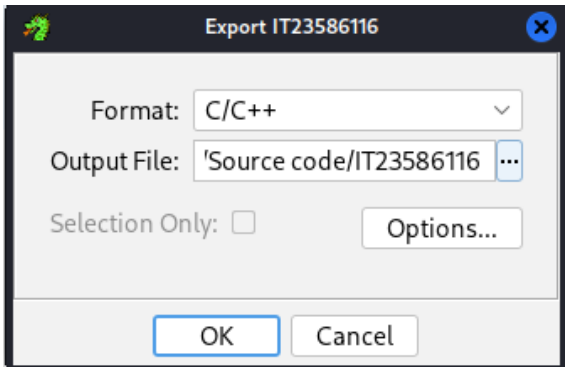
Step 1 – Installing Ghidra

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ sudo apt install ghidra  # Kali Linux
[sudo] password for luthfi:
Upgrading:
  openjdk-21-jre  openjdk-21-jre-headless

Installing:
  ghidra

Installing dependencies:
  ghidra-data  openjdk-21-jdk  openjdk-21-jdk-headless

Suggested packages:
  openjdk-21-demo  openjdk-21-source  visualvm

Summary:
  Upgrading: 2, Installing: 4, Removing: 0, Not Upgrading: 156
  Download size: 585 MB
  Space needed: 835 MB / 62.5 GB available

Continue? [Y/n] y
Get:1 http://http.kali.org/kali kali-rolling/main amd64 openjdk-21-jre amd64 21.0.7~8ea-1 [205 kB]
Get:3 http://http.kali.org/kali kali-rolling/main amd64 openjdk-21-jdk-headless amd64 21.0.7~8ea-1 [82.8 MB]
Get:5 http://http.kali.org/kali kali-rolling/main amd64 ghidra amd64 11.3.1+ds-0kali1 [379 MB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 openjdk-21-jre-headless amd64 21.0.7~8ea-1 [41.8 MB]
Get:4 http://http.kali.org/kali kali-rolling/main amd64 openjdk-21-jdk amd64 21.0.7~8ea-1 [3,541 kB]
Get:6 http://kali.download/kali kali-rolling/main amd64 ghidra-data all 10.5-0kali1 [78.1 MB]
```

Step 2 – open Ghidra And import the file



Step 3 – Export the file



Step 4 – check the decompiled file.

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ ls
data.txt  IT23586116  IT23586116_Details.gpr  IT23586116_Details.rep  log_cleanup.sh  Source_code

┌──(luthfi㉿kali)-[~/SNP]
└─$ cd Source_code

┌──(luthfi㉿kali)-[~/SNP/Source_code]
└─$ ls
IT23586116.c  IT23586116.h
```

Step 5 – sample code (code may not be 100% accurate)

```
int main(void)

{
  int iVar1;
  FILE *pFVar2;
  size_t sVar3;
  char uuid [50];
  FILE *out;
  char *key;
  FILE *fp;

  pFVar2 = popen("sudo cat /sys/class/dmi/id/product_uuid","r");
  if (pFVar2 == (FILE *)0x0) {
    puts("Error retrieving data");
    iVar1 = 1;
  }
  else {
    fgets(uuid,0x32,pFVar2);
    pclose(pFVar2);
    sVar3 = strcspn(uuid,"\n");
    uuid[sVar3] = '\0';
    xor_encrypt_decrypt(uuid,"key");
    pFVar2 = fopen("data.txt","w");
    if (pFVar2 == (FILE *)0x0) {
      puts("Error creating data.txt!");
      iVar1 = 1;
    }
    else {
      fputs(uuid,pFVar2);
      fclose(pFVar2);
      iVar1 = 0;
    }
  }
  return iVar1;
}


void _fini(void)

{
  return;
}


┌──(luthfi㉿kali)-[~/SNP]
└─$
```

II. Key findings from debugging.

- Setting break points

```
(gdb) break malloc
Function "malloc" not defined.
Make breakpoint pending on future shared library load? (y or [n]) y
Breakpoint 1 (malloc) pending.
(gdb) run
Starting program: /home/luthfi/SNP 1/IT23586116
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 1, 0×00007ffff7e52b50 in malloc () from /lib/x86_64-linux-gnu/libc.so.6
(gdb) backtrace
#0  0×00007ffff7e52b50 in malloc () from /lib/x86_64-linux-gnu/libc.so.6
#1  0×00007ffff7e3028a in popen () from /lib/x86_64-linux-gnu/libc.so.6
#2  0×000055555555525d in main () at IT23586116.c:14
(gdb) info registers rdi
rdi            0×100               256
(gdb) info registers rax
rax            0×555555556010      93824992239632
(gdb)
```

- Malloc is the memory allocation. And used to request a specific amount of memory from system's heap.
- Backtrace shows that malloc was called by popen() inside main() function.
- Rdi shows that malloc(256) was requested.
- So finally it is visible that 256 bytes of memory was allocated.

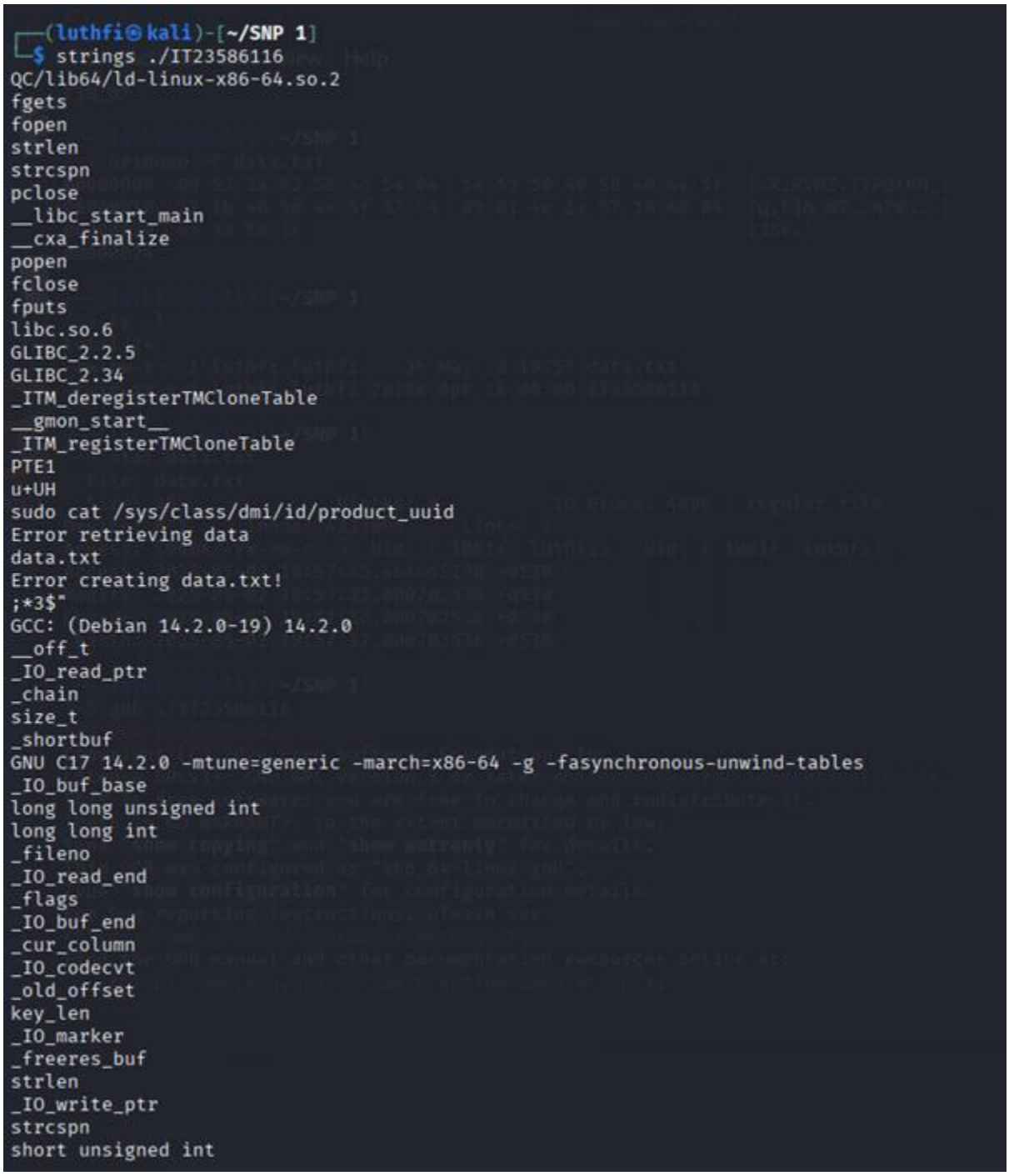
```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ stat ./IT23586116
  File: ./IT23586116
  Size: 20288           Blocks: 40         IO Block: 4096   regular file
Device: 8,1     Inode: 4333066     Links: 1
Access: (0755/-rwxr-xr-x)  Uid: ( 1001/  luthfi)   Gid: ( 1001/  luthfi)
Access: 2025-05-02 19:57:31.956702658 +0530
Modify: 2025-04-16 00:00:47.077671183 +0530
Change: 2025-05-02 19:53:50.153318694 +0530
 Birth: 2025-05-02 19:53:50.153318694 +0530
```

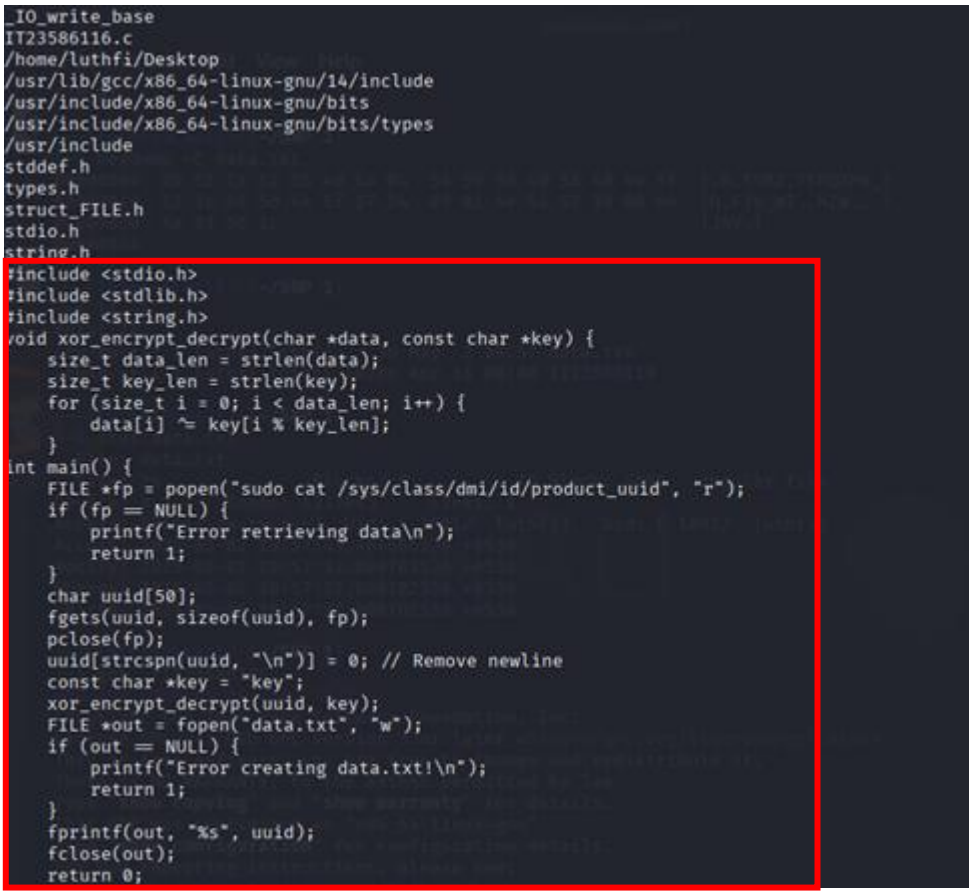
- Stats give detail information of a file.
- Metadata – file permissions (7 – rwx 7- r-x 5- r-w – owner,group,other)

- Strace is used to trace the system calls in the program.

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ strace ./IT23586116
execve("./IT23586116", ["./IT23586116"], 0x7ffe9eb3dcc0 /* 56 vars */) = 0
brk(NULL)                               = 0x55e04593a000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fbe8b3ff000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
fstat(3, {st_mode=S_IFREG|0644, st_size=101882, ...}) = 0
mmap(NULL, 101882, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fbe8b3e6000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0p\236\2\0\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 840, 64) = 840
fstat(3, {st_mode=S_IFREG|0755, st_size=2003408, ...}) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 840, 64) = 840
mmap(NULL, 2055800, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fbe8b1f0000
mmap(0x7fbe8b218000, 1462272, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7fbe8b218000
mmap(0x7fbe8b37d000, 352256, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18d000) = 0x7fbe8b37d000
mmap(0x7fbe8b3d3000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e2000) = 0x7fbe8b3d3000
mmap(0x7fbe8b3d9000, 52856, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7fbe8b3d9000
close(3)                                = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fbe8b1ed000
arch_prctl(ARCH_SET_FS, 0x7fbe8b1ed740) = 0
set_tid_address(0x7fbe8b1eda10)         = 70934
set_robust_list(0x7fbe8b1eda20, 24)     = 0
rseq(0x7fbe8b1ed680, 0x20, 0, 0x53053053) = 0
mprotect(0x7fbe8b3d3000, 16384, PROT_READ) = 0
mprotect(0x55e031adc000, 4096, PROT_READ) = 0
mprotect(0x7fbe8b43b000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7fbe8b3e6000, 101882)          = 0
getrandom("\x55\xc8\xee\x4e\x52\x20\xce\xd2", 8, GRND_NONBLOCK) = 8
brk(NULL)                               = 0x55e04593a000
brk(0x55e04595b000)                     = 0x55e04595b000
pipe2([3, 4], O_CLOEXEC)                = 0
prlimit64(0, RLIMIT_NOFILE, NULL, {rlim_cur=1024, rlim_max=512*1024}) = 0
prlimit64(0, RLIMIT_NOFILE, NULL, {rlim_cur=1024, rlim_max=512*1024}) = 0
mmap(NULL, 36864, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) = 0x7fbe8b3f6000
rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0
clone3({flags=CLONE_VM|CLONE_VFORK|CLONE_CLEAR_SIGHAND, exit_signal=SIGCHLD, stack=0x7fbe8b3f6000, stack_size=0x9000}, 88) =
 70935
munmap(0x7fbe8b3f6000, 36864)           = 0
rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0
close(4)                                = 0
fcntl(3, F_SETFD, 0)                    = 0
fstat(3, {st_mode=S_IFIFO|0600, st_size=0, ...}) = 0
read(3, "b7c9341a-2593-744b-8742-bd712aca"..., 4096) = 37
close(3)                                = 0
wait4(70935, [{WIFEXITED(s) && WEXITSTATUS(s) == 0}], 0, NULL) = 70935
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=70935, si_uid=1001, si_status=0, si_utime=0, si_stime=0} ---
openat(AT_FDCWD, "data.txt", O_WRONLY|O_CREAT|O_TRUNC, 0666) = 3
fstat(3, {st_mode=S_IFREG|0664, st_size=0, ...}) = 0
write(3, "\tR\32RVMZ\4TYP@XHN_Q\33F]N_WT\t\1NZW\30\10\4"..., 36) = 36
close(3)                                = 0
exit_group(0)                           = ?
+++ exited with 0 +++
```

- Strings is used to extract readable text from a file.





```
_IO_write_base
IT23586116.c
/home/luthfi/Desktop
/usr/lib/gcc/x86_64-linux-gnu/14/include
/usr/include/x86_64-linux-gnu/bits
/usr/include/x86_64-linux-gnu/bits/types
/usr/include
stddef.h
types.h
struct_FILE.h
stdio.h
string.h
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void xor_encrypt_decrypt(char *data, const char *key) {
    size_t data_len = strlen(data);
    size_t key_len = strlen(key);
    for (size_t i = 0; i < data_len; i++) {
        data[i] ^= key[i % key_len];
    }
int main() {
    FILE *fp = popen("sudo cat /sys/class/dmi/id/product_uuid", "r");
    if (fp == NULL) {
        printf("Error retrieving data\n");
        return 1;
    }
    char uuid[50];
    fgets(uuid, sizeof(uuid), fp);
    pclose(fp);
    uuid[strcspn(uuid, "\n")] = 0; // Remove newline
    const char *key = "key";
    xor_encrypt_decrypt(uuid, key);
    FILE *out = fopen("data.txt", "w");
    if (out == NULL) {
        printf("Error creating data.txt!\n");
        return 1;
    }
    fprintf(out, "%s", uuid);
    fclose(out);
    return 0;
```

- So here machine retrieves the UUID (Universal Unique Identifier - 128-bit identifier) using popen() function.
- XOR-encrypts the UUID with key
- Writes the encrypted string to data.txt.

c. File System Analysis

- Strings is used to extract readable text from a file.
- Hexdump -c used to dsplay the contents of a file in hexadecimal and ASCII format

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ strings data.txt
RVMZ
TYP@XHN_Q
F]N_WT

┌──(luthfi㉿kali)-[~/SNP 1]
└─$ hexdump -C data.txt
00000000  09 52 1a 52 56 4d 5a 04  54 59 50 40 58 48 4e 5f  |.R.RVMZ.TYP@XHN_|
00000010  51 1b 46 5d 4e 5f 57 54  09 01 4e 5a 57 18 08 04  |Q.F]N_WT..NZW...|
00000020  4a 53 56 1c                                       |JSV.|
00000024
```

- In the below code we can see the changes in the files and metadata creation before and after creation of data.txt file.

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ ls -l
total 20
-rwxr-xr-x 1 luthfi luthfi 20288 Apr 16 00:00 IT23586116

┌──(luthfi㉿kali)-[~/SNP 1]
└─$ ./IT23586116
[sudo] password for luthfi:

┌──(luthfi㉿kali)-[~/SNP 1]
└─$ ls -l
total 24
-rw-rw-r-- 1 luthfi luthfi    36 May  2 22:15 data.txt
-rwxr-xr-x 1 luthfi luthfi 20288 Apr 16 00:00 IT23586116
```

d. Analysis of "data.txt"

I. Explanation of its content.

- Content of data.txt

```
┌──(luthfi㉿kali)-[~/SNP]
└─$ cat data.txt
      R RVMZTYP@XHN_Q]N_WT    NZWJSV
```

- So in the below file we can see statics of the data.txt file (changes, modifications and metadata-permissions)

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ stat data.txt
  File: data.txt
  Size: 36              Blocks: 8          IO Block: 4096   regular file
Device: 8,1     Inode: 4333142     Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1001/  luthfi)   Gid: ( 1001/  luthfi)
Access: 2025-05-02 22:16:25.321457295 +0530
Modify: 2025-05-02 22:15:23.898997693 +0530
Change: 2025-05-02 22:15:23.898997693 +0530
 Birth: 2025-05-02 22:15:23.890997893 +0530
```

- Checking if the file is encoded or encrypted.

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ file data.txt
data.txt: data
```

- If it says ‘data’ is could be binary or encrypted.

```
┌──(luthfi㉿kali)-[~/SNP 1]
└─$ strings data.txt
RVMZ
TYP@XHN_Q
F]N_WT
```

- This suggests that it could be XOR encryption. (output is scrambled but not fully random)

II. Source of the contents of data.txt

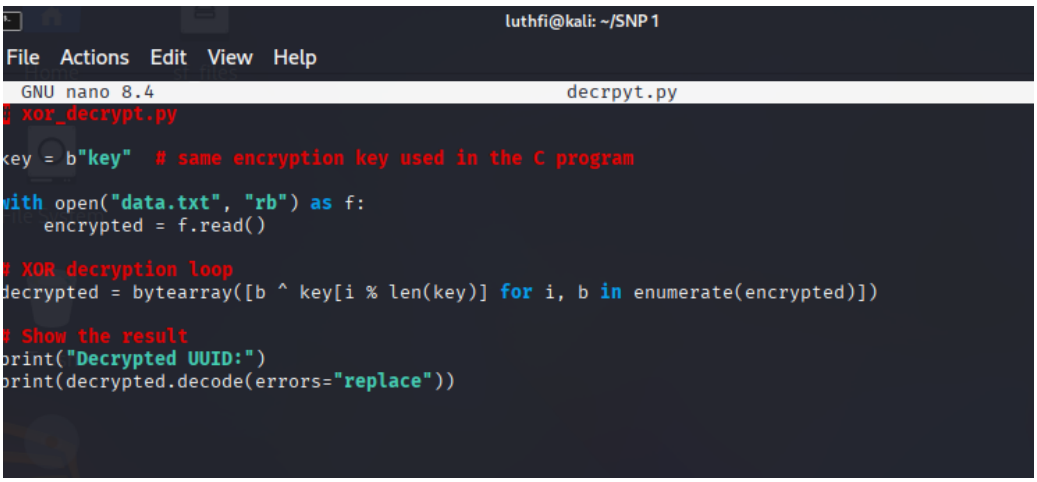
- Source of the content comes from the machine's UUID, it is retrieved from **/sys/class/dmi/id/product_uuid.**
- In the c program **FILE *fp = popen("sudo cat /sys/class/dmi/id/product_uuid", "r");** retrieved.
- It is stored in the data.txt file.

FILE *out = fopen("data.txt", "w");

fprintf(out, "%s", uuid);

III. Any decoded/processed information.

- The c program used the system UUID (Universally Unique Identifier) and the XOR encryption using the key. So this key is mixed and saved in data.txt.
- So to take the original UUID back we can decrypt it using XOR process with the same key.



- Here I have used a simple python code to decrypt the data.txt and map it the machine UUID.

- So here we can confirm now the machine's UUID matches the decrypted code.

IV. Tools used for analysis.

- Tools– gihidra was used to retrieve the source code from exe file. Cat, strings, file, stats were used to identify what was in data.txt and if it is encrypted or encoded.