SLIIT
Discover Your Future

BSc (Hons) in Information Technology
Specialized in Cyber Security

Sri Lanka Institute of Information Technology

Systems and Network Programming (SNP):

Linux Environments

MNM. RUHAIM

IT23256446

Contents

INTRODUCTION

This report delves into key aspects of system and network programming.in this report mainly focusing on foundational, practical and functional skills for configuring and securing modern network infrastructures. It exploring the basic of Linux (Unix) environment. Linux is backbone of many server and network operations.

Next this report is move on to the installation and configuration of **DHCP, DNS, NTP** servers, these services are fundamental of managing network name and time synchronized across multiple systems. Following this focus on the shell scripting and security, its highlights efficiency and strengthens system security. Security system is further explored through **SSH** configuration, a critical aspect of securing remote access and communications.

The iptables creation and configuration its main role in controlling network traffic protecting systems from unauthorized access. Last thing is the best practice for network interface configuration discussed to ensure the optimal performance and reliability of network communication.

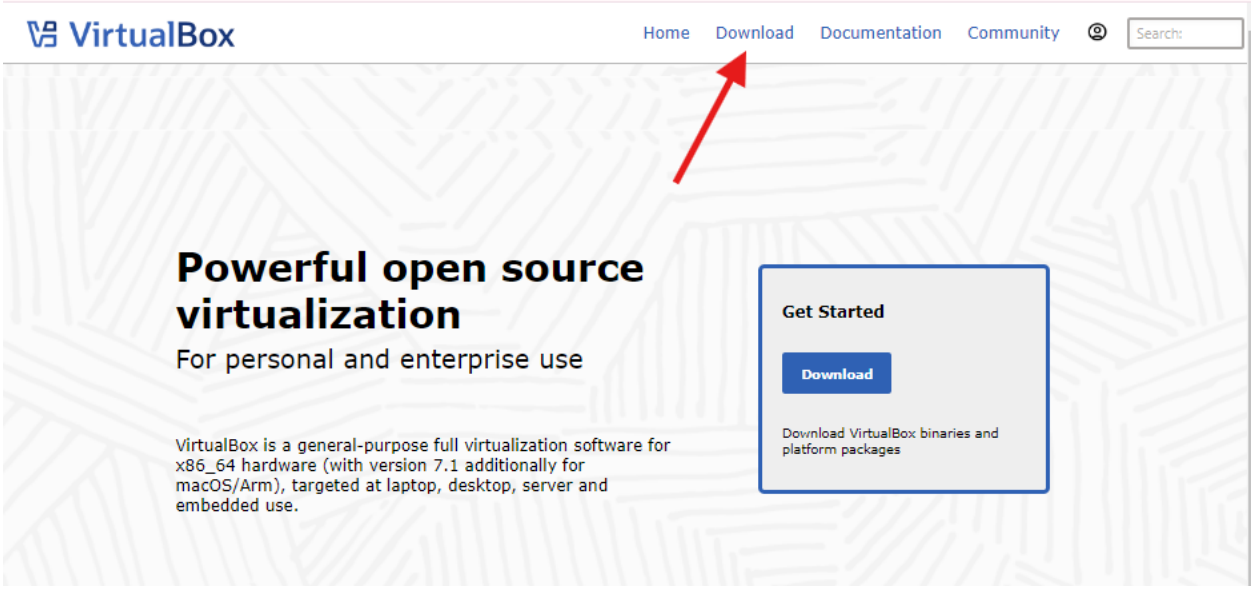
Basics of Linux Environments

Virtual machine installation setup

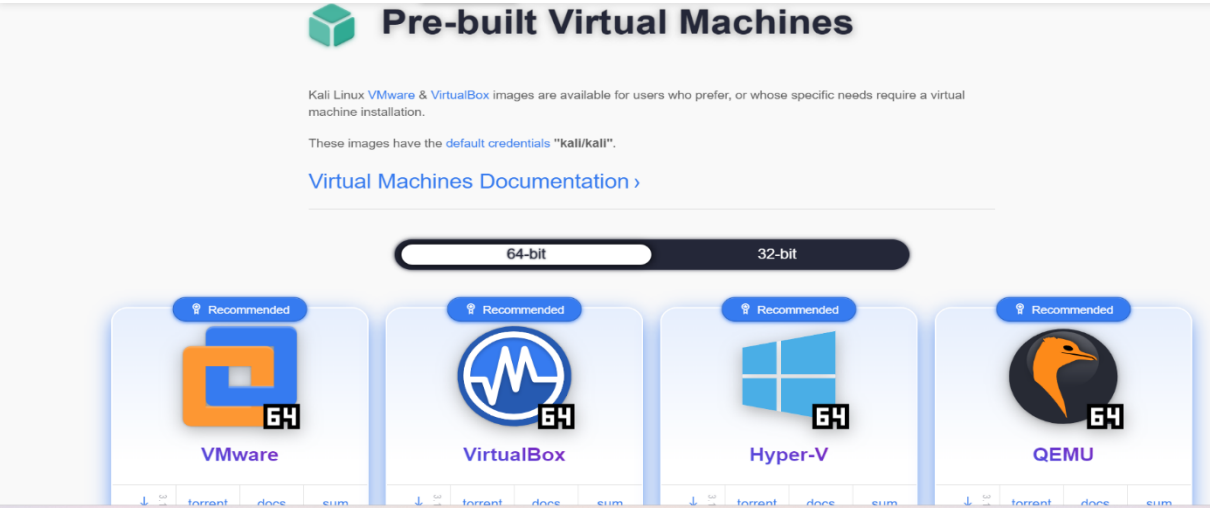
Step 01: Click the virtual machine installation link

Installation link : C:\Users\Mohemmed Ruhaim\OneDrive - Sri Lanka Institute of Information Technology\Documents\Virtual Machines Oracle VirtualBox

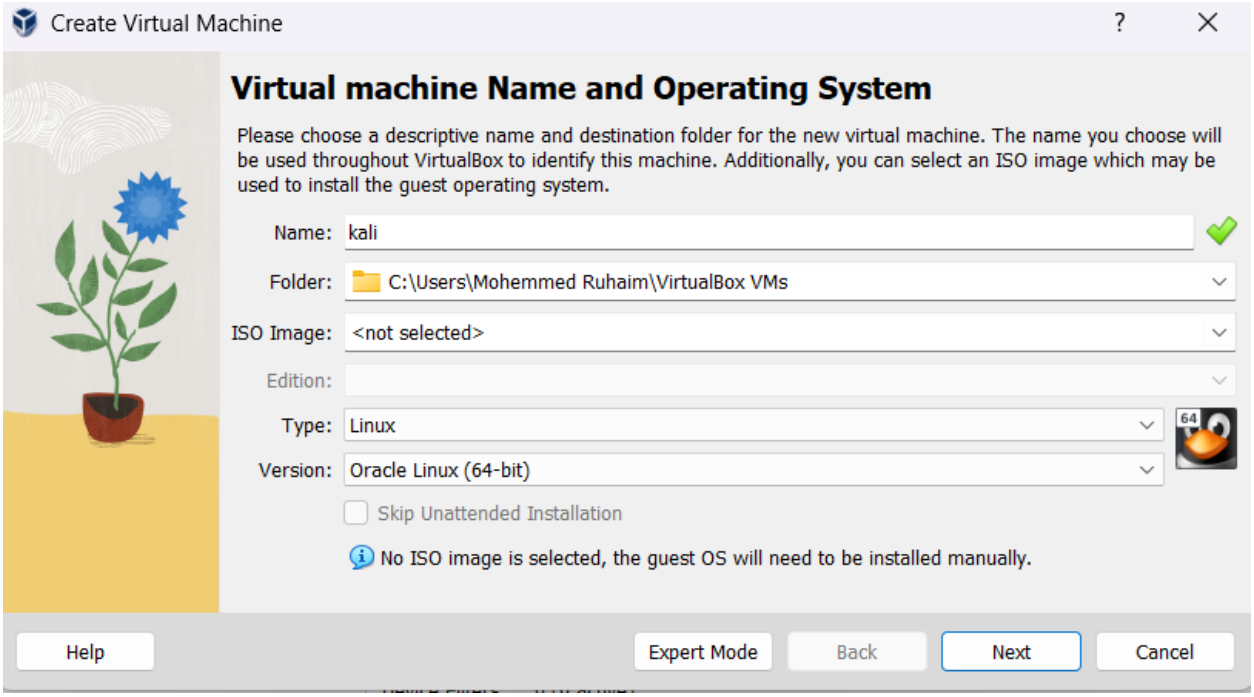
Step 02: Click the Download button.



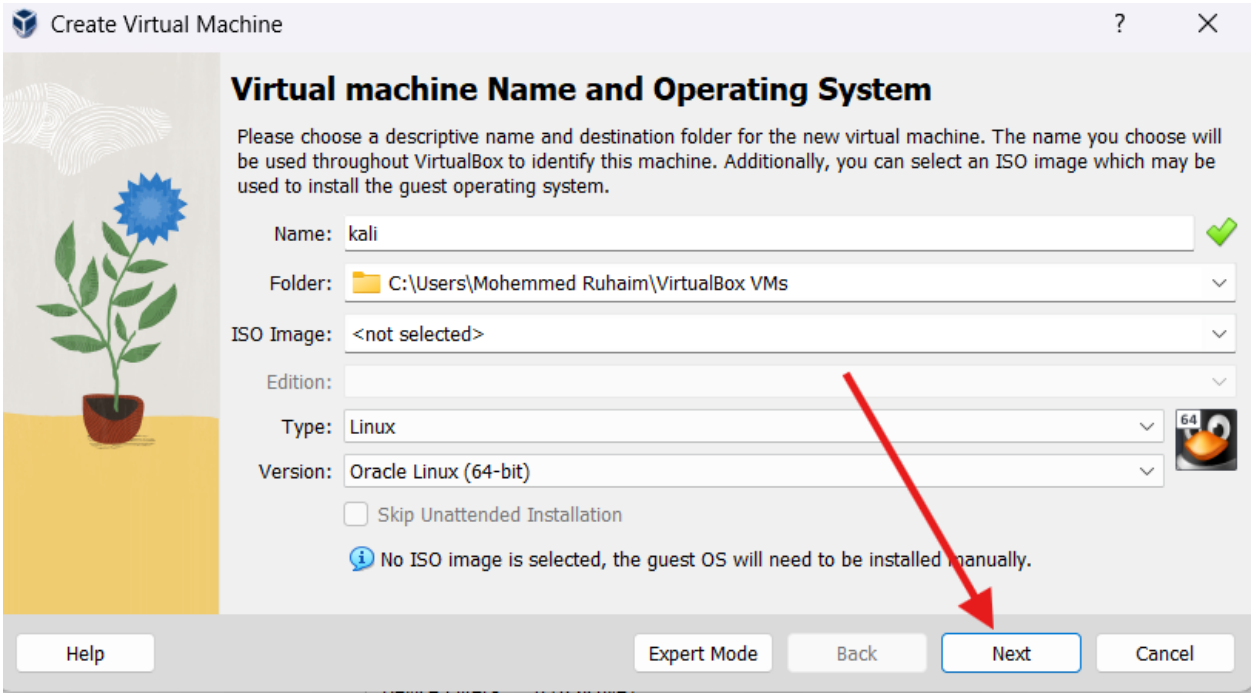
Step 03: click on the VirtualBox.



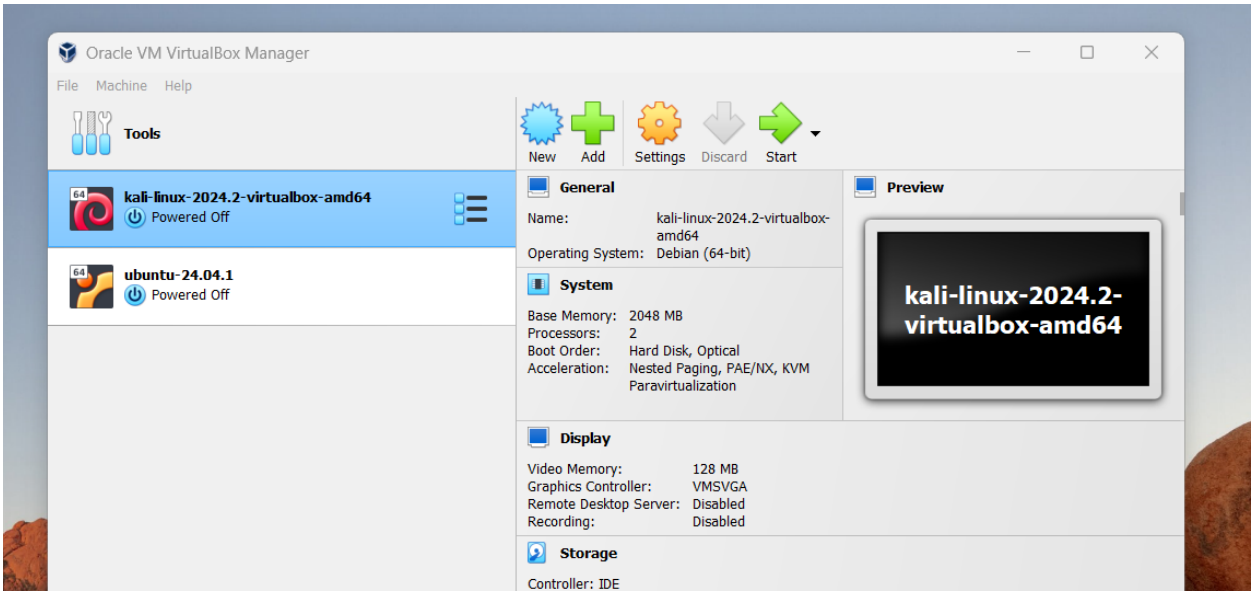
Step 04: Enter the Virtual machine name and version.



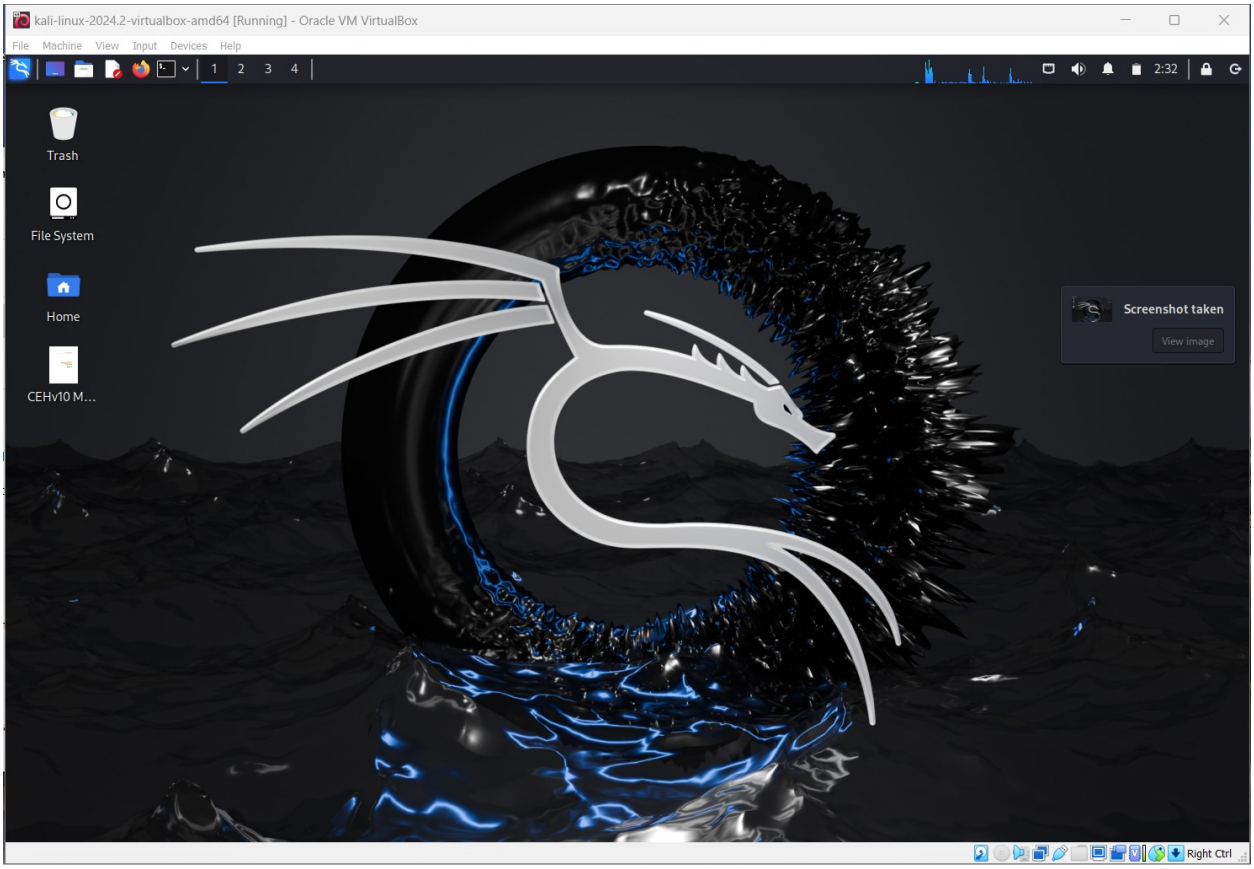
Step 05: Click on the next button.



step 05: click on kali linux-2024.2-virtualbox-amd64 and press the start button.



Step 06: this is a user interface of kali Linux.

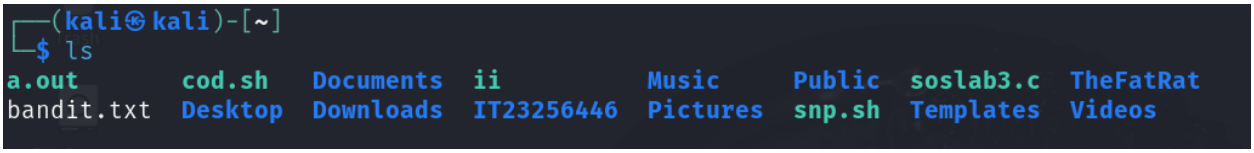


Note: *successfully installed kali Linux operating system.*

The basic commands on kali Linux

1. **pwd**: will stand on current path currently stay in place.

2. **ls**: it's list down the Directory and files.

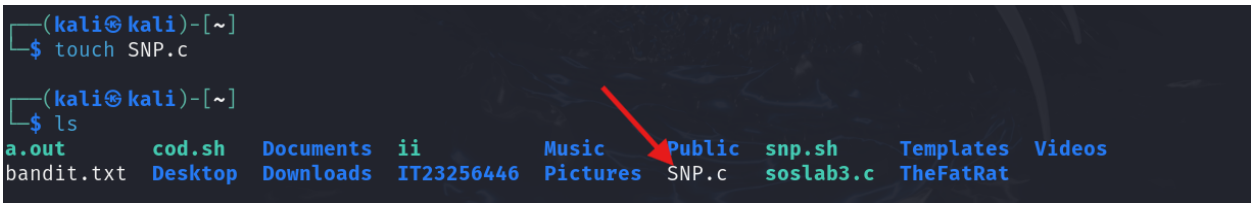


3. **ls -a**: list down the all hidden files.

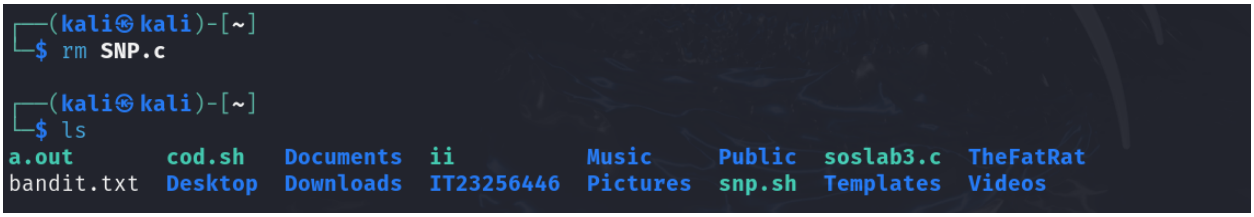
4. **mkdir_Directory name:** make a new directory or create a new directory

Note: *you should be see the previous image (2.ls)- there are no directory such as SNP and Now see the current image here it is* ***SNP*** *directory.*

5. **touch_filename:** make a new file or create a new file (ex:- txt, c, sh, etc)



6. **rm_filename**: remove file or delete file

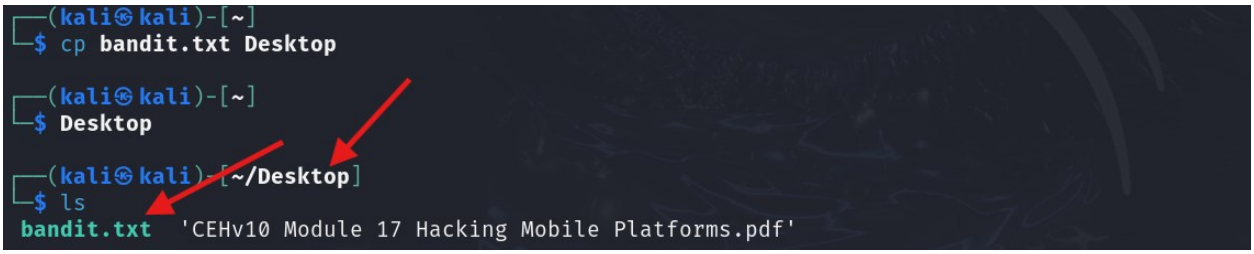


7. cat_filename: read the content or whatever include in the file

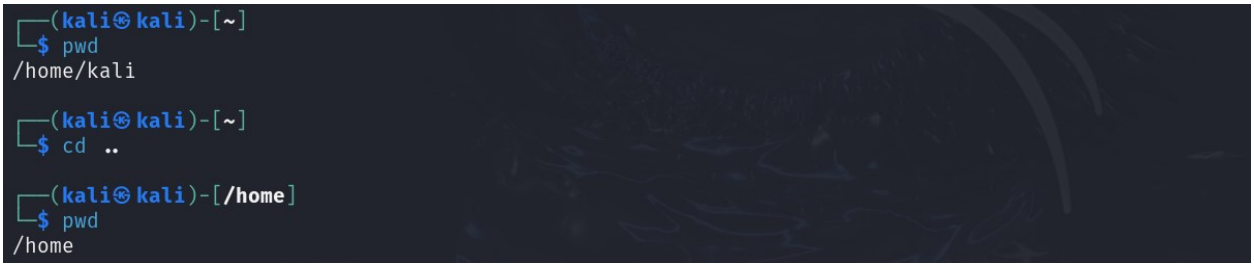
8. **nano_filename:** edit file, when edit is complete press control + x and press y and press the enter button (Save and exit).

9. **chmod_filename:** Change the file mode. (EX:- read – write | write – read | execute – read | execute – write)
   read- r
   write- w
   execute- x

10. **cp filename Directotyname:** copy file and past the another directory.



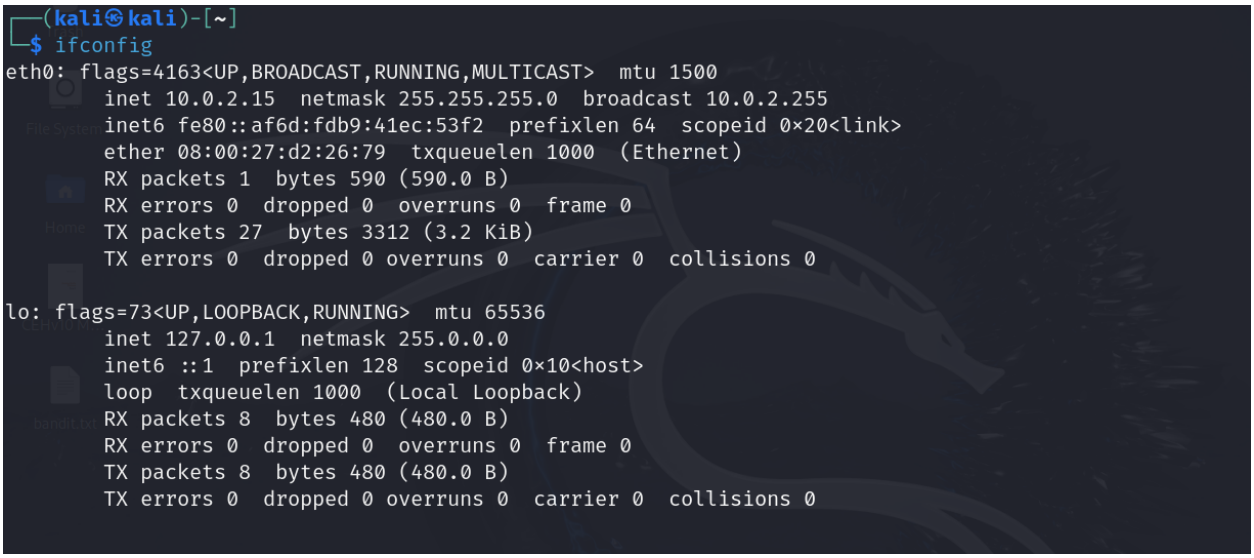
11. **cd ..:** this is revers to the parent directory to current directory .



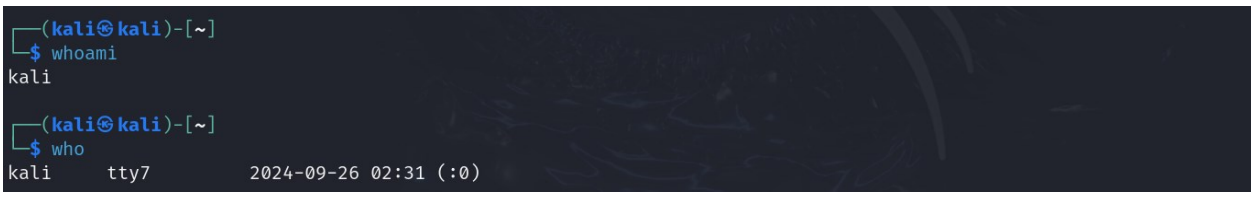
12. **mv**: Moving file [source] [destination] / rename file (old_name_new_name)

13. **sudo su**: going to effectively gaining root privileges without having to log out and log back in as the root user.

14. **Ifconfig**: ifconfig is considered deprecated in many Linux distributions in favor of the ip command. However, it is still widely used and available in many systems. Display network interface, configure network interface.



15. **whoami:** gives your user name.
16. **who:** gives your some details.



System introduction and user management on kali linux

1. **uname**: its gives your os name [Linux]

```
┌──(kali㉿kali)-[~]
└─$ uname
Linux
```

2. **uame** -a: gives your full details on your os system

```
┌──(kali㉿kali)-[~]
└─$ uname -a
Linux kali 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.15-2kali1 (2024-05-17) x86_64 GNU/Linux
```

3. **cat /proc/version**: its read your os system version

```
┌──(kali㉿kali)-[~]
└─$ cat /proc/version
Linux version 6.6.15-amd64 (devel@kali.org) (gcc-13 (Debian 13.2.0-24) 13.2.0
, GNU ld (GNU Binutils for Debian) 2.42) #1 SMP PREEMPT_DYNAMIC Kali 6.6.15-2
kali1 (2024-05-17)
```

4. **df -h**: When you run df -h, you will see a list of all mounted file systems along with the following information for each:
   Filesystem: The name of the file system or disk.
   Size: The total size of the file system.
   Used: The amount of space that is currently used.
   Available: The amount of space that is still available for use.
   Use%: The percentage of the file system that is currently used.
   Mounted on: The mount point or directory where the file system is accessed.

```
┌──(kali㉿kali)-[~]
└─$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            948M     0  948M   0% /dev
tmpfs           198M  992K  197M   1% /run
/dev/sda1        79G   17G   58G  23% /
tmpfs           989M     0  989M   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           198M  124K  198M   1% /run/user/1000
```

5. **free -m:** you will see your summery of the memory

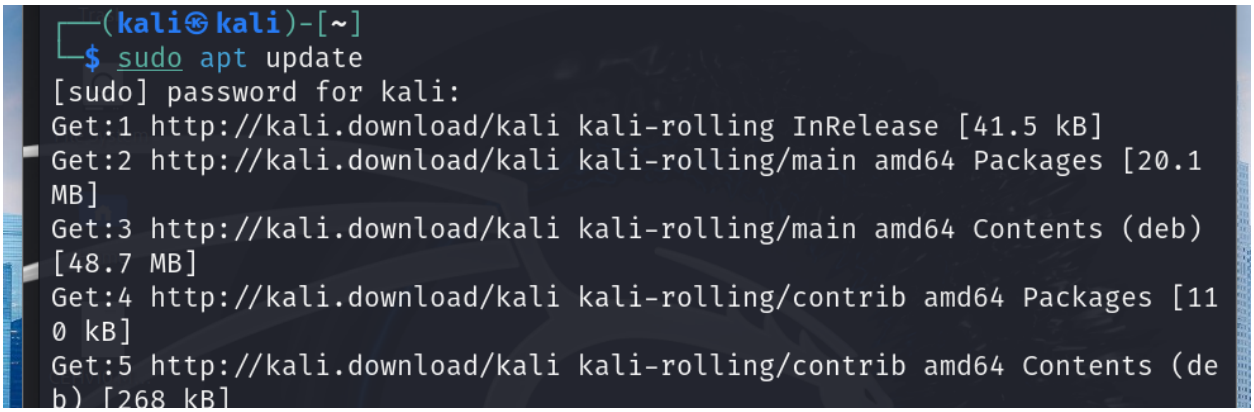
```
┌──(kali㉿kali)-[~]
└─$ free -m
               total        used        free      shared  buff/cache   available
Mem:            1976         788         851          20         497        1188
Swap:           1023           0        1023
```

DHCP, DNS and NTP Services

DHCP (Dynamic Host Configuration Protocol)

Step 01: you need to update your system. So we used this command for update the system.

- **sudo apt update**



step 02: install DHCP server. The installation command is

- **sudo apt install isc-dhcp-server**

step 03: you needs to edit dhcp configuration file for The command should be.

- **Sudo nano /etc/dhcp/dhcpd.conf**

This is the configuration and you can start with

subnet 10.0.2.0 netmask 255.255.255.0 {

range 10.0.2.20 10.0.2.50;

option domain-name-servers 8.8.8.8, 8.8.4.4;

option subnet-mask 255.255.255.0;

}

Step 04: then start or restart your dhcp server. The command should be.

- **sudo systemctl restart isc-dhcp-server.service.**

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

step 05: finally want to check the dhcp server is works or not, the command should be check status about the dhcp server,

- **sudo systemctl status isc-dhcp-server.service**

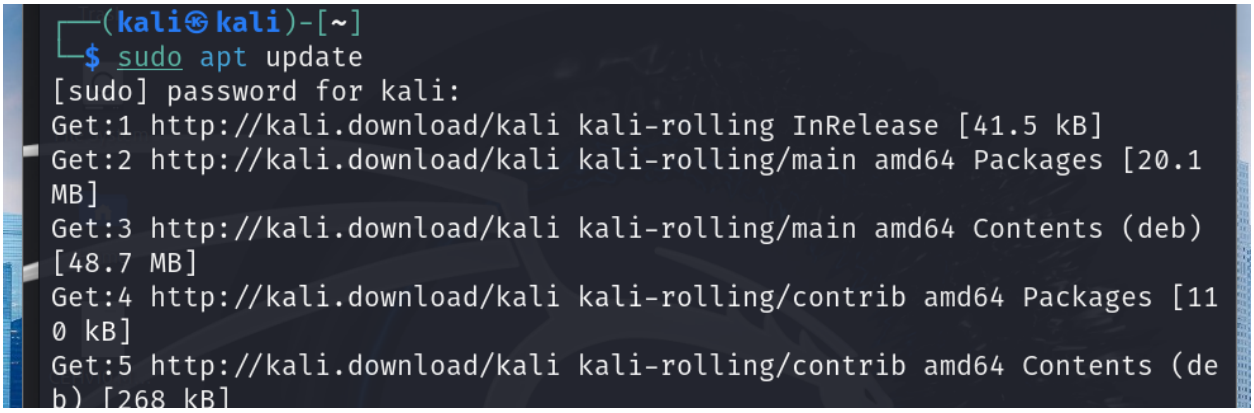
```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status isc-dhcp-server.service

● isc-dhcp-server.service - LSB: DHCP server
     Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
     Active: active (running) since Tue 2024-10-01 01:32:43 EDT; 33s ago
 Invocation: dd1058cb79084417a444190f30b1ac3e
       Docs: man:systemd-sysv-generator(8)
    Process: 7340 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=0/SUCCES
      Tasks: 1 (limit: 2221)
     Memory: 6.4M (peak: 8.2M)
        CPU: 44ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─7353 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf eth0

Oct 01 01:32:41 kali systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP server...
Oct 01 01:32:41 kali isc-dhcp-server[7340]: Launching IPv4 server only.
Oct 01 01:32:41 kali dhcpd[7353]: Wrote 0 leases to leases file.
Oct 01 01:32:41 kali dhcpd[7353]: Server starting service.
Oct 01 01:32:43 kali isc-dhcp-server[7340]: Starting ISC DHCPv4 server: dhcpd.
Oct 01 01:32:43 kali systemd[1]: Started isc-dhcp-server.service - LSB: DHCP server.
```

Note: Successfully Run DHCP server.

DNS (Domain Name System)

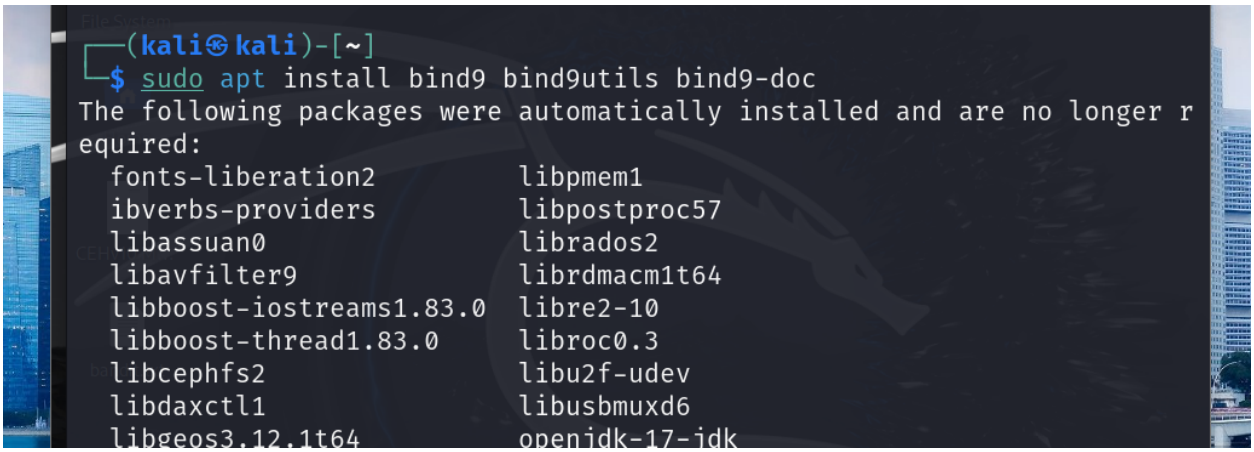
Step 01: you needs to update your operating system. The command should be update to the system

- **sudo apt update**



step 02: install DNS Server. The command should be installation DNS.

- **sudo apt install bind9 bind9utils bid9-doc**



Step 03: edit configuration file. The command should be

- **sudo nano /etc/bind/named.conf.local /etc/bind/db/itsme.com**
- **sudo nano /etc/bind/db/itsme.com**

```
┌──(kali㉿kali)-[~]
└─$ sudo cp /etc/bind/db.local /etc/bind/db.itsme.com

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/bind/db.itsme.com
```

- we needs to edit should be.

  You need to change the Ip address for your Ip address

```
File  Actions  Edit  View  Help
  GNU nano 8.1              /etc/bind/db.itsme.com
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns.itsme.com. root.itsme.com. (
                              1         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns.itsme.com.
ns      IN      A       10.0.2.15
```

Step 04: edit option file. To the command should be

- **sudo nano /etc/bind/named.conf.options**

```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/bind/named.conf

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/bind/named.conf.options

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/bind/named.conf.local
```

- we want to do some change on this option file.
  That is change the forwarders {8.8.8.8 or 8.8.4.4}

```
 GNU nano 8.1          /etc/bind/named.conf.options
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multip>
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses repl>
        // the all-0's placeholder.

        // forwarders {
        //      8.8.8.8;

        // };
```

Step 05: we needs to edit BIND server data file. That's cod should be

1. copy db.127 to past db.0
- **nano /etc/bind/db.0**

```
┌──(kali㉿kali)-[~]
└─$ sudo cp /etc/bind/db.127 /etc/bind/db.0

┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/bind/db.0
```

- Change the Ip address and puts your Ip address

```
  GNU nano 8.1                    /etc/bind/db.127
;
; BIND reverse data file for 10.0.2.15/24
;
$TTL    604800
@       IN      SOA     ns1.itsme.com. (
                        2024092901      ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
@       IN      NS      ns1.itsme.com.
1.0.0   IN      PTR     ns1.itsme.com.
```

Step 06: checkzone file is it correct or not. For check it code should be.

- **sudo named-checkzone file path**

```
┌──(kali㉿kali)-[~]
└─$ sudo named-checkzone itsme.com /etc/bind/db.itsme.com
zone itsme.com/IN: loaded serial 1
OK
```

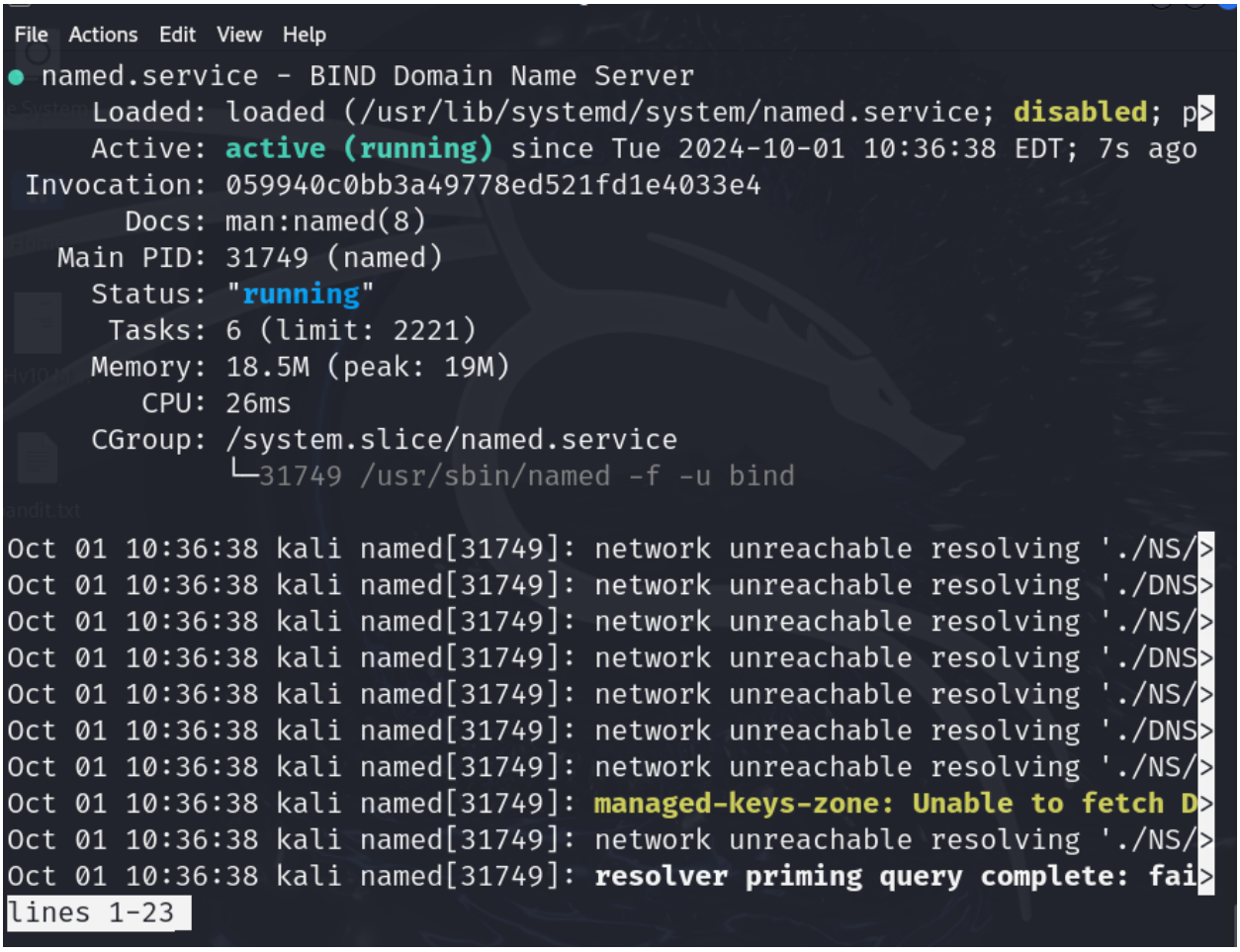
step 07: Restart or start the dns server.

- **sudo systemctl restart named.service**

```
┌──(kali㉿kali)-[~]
└─$ sudo named-checkconf

┌──(kali㉿kali)-[~]
└─$ sudo systemctl start named.service
```

step 08: checked the status for DNS is active or not.

- **Sudo systemctl status**



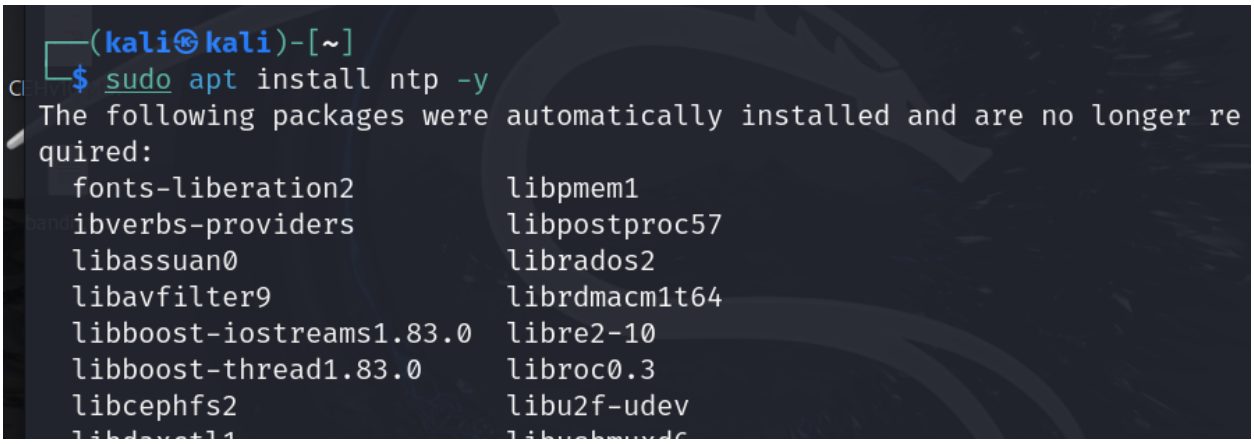
***Note**: successfully Run DNS server.*

NTP (Network Time Protocol)

Step 01: update your operating system.

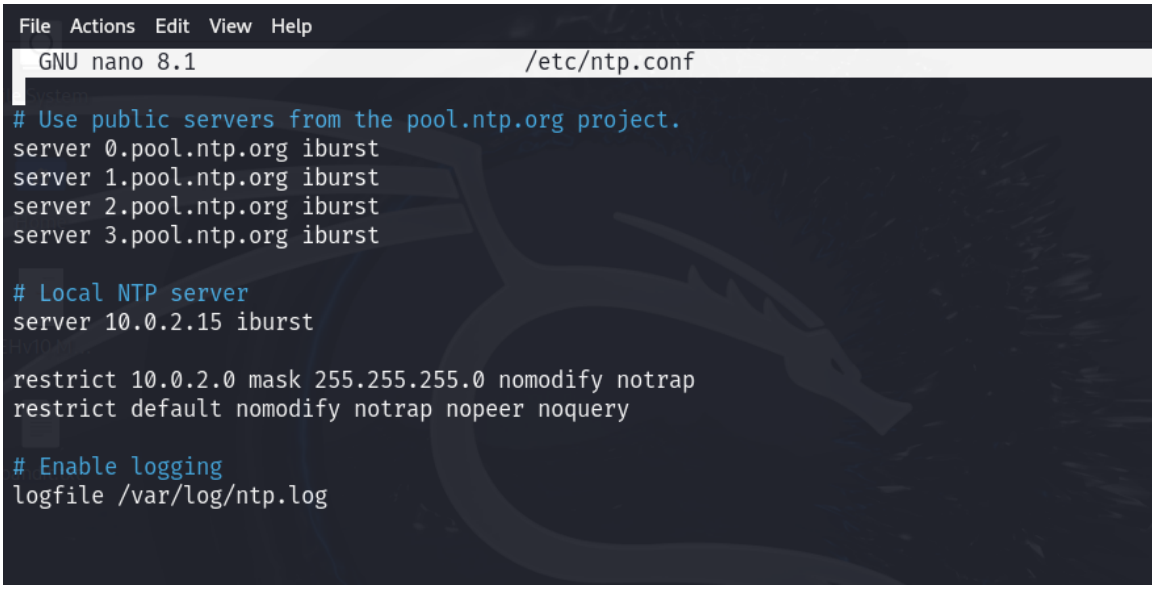
- Using this command – **sudo apt update**

Step 02.: install NTP server. Using **sudo apt install ntp -y**



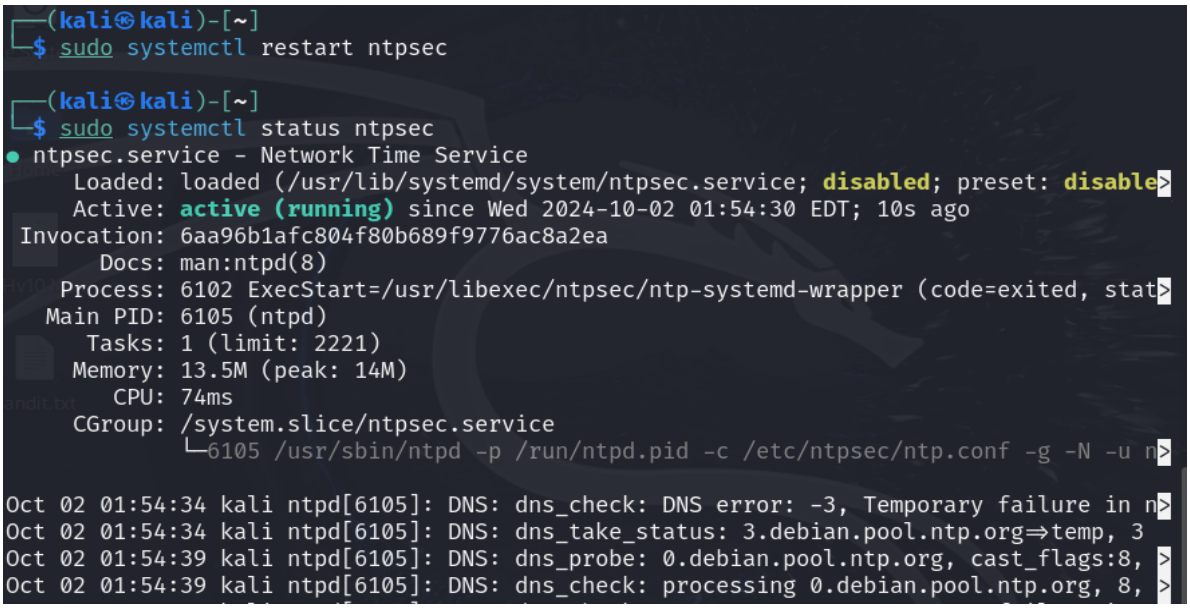
Step 03: edit your ntp configuration file.

- Changer your Ip address and edit your subnet and mask.



Step 04: restart and checked the status

- For the restart command should be – **sudo systemctl restart ntpsec or sudo systemctl restart ntpserver**



Shell Scripting and Security

Basic Shell scripting on Linux system

This script provides to automate a report that captures key system details every day.

Step 01: Create a Shell Scripting file. Using this command

- **touch filename.sh**

```
┌──(kali㉿kali)-[~]
└─$ cd system_reports

┌──(kali㉿kali)-[~/system_reports]
└─$ ls
daily_reports  daily_reports.sh
```

Step 02: edit file. Using

- **nano filename.sh**

```
  GNU nano 8.1                          daily_reports.sh
#!/bin/bash


# Create the destination directory if it doesn't exist
DEST_DIR="/home/kali/system_reports"
mkdir -p "$DEST_DIR"


# get the corrent date
        DATE=$(date +"%Y-%m-%d")

# get the system time
        UPTIME=$(uptime -p)

#get the free memory
        FREE_MEMORY=$(free -h | grep "Mem:" | awk '{print $4}')

#get disk usage
        DISK_USAGE=$(df -h | grep "^/dev/" | awk '{print $1, $3, $2, $5}')

REPORT_FILE="/home/kali/system_reports/system_reports_$DATE.txt"
```

```
# write the report
        {
                echo "system Report -$DATE"
                echo "_____________________"
                echo "uptime: $UPTIME"
                echo "Free Memory: $FREE_MEMORY"
                echo "Disk Usage:"
                echo "$DISK_USAGE"
} > "$REPORT_FILE"

echo "Report generated: $REPORT_FILE"
```

Step 03: change to execute mode**.** Using

- **chmod +x /home/kali/system_reports/daily_reports.sh**

```
┌──(kali㉿kali)-[~/system_reports]
└─$ chmod +x /home/kali/system_reports/daily_reports.sh
```

step 04: Run the script. Using

- **./ daily_reports.sh**

```
┌──(kali㉿kali)-[~/system_reports]
└─$ ./daily_reports.sh
Report generated: /home/kali/system_reports/system_reports_2024-10-02.txt
```

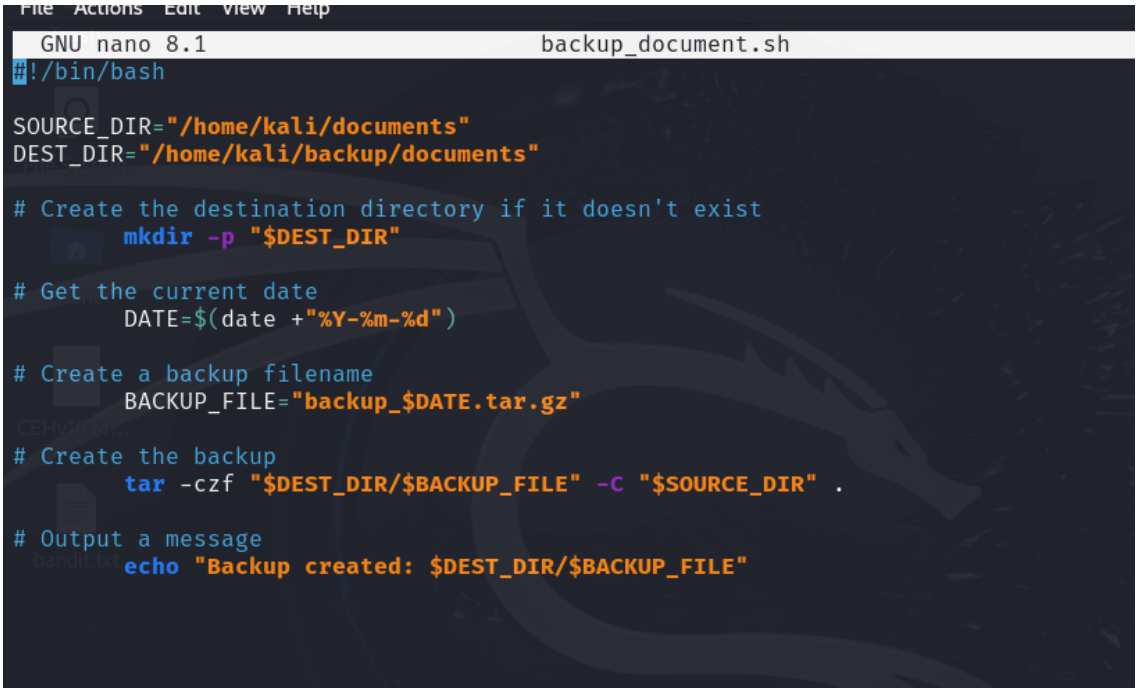
This script provide to automate the backup of a critical directory (/home/user/documents) containing important files.

Step 01: You need to create backup directory. Using

- **mkdir directory name**

step 02: Create the new file edit the file. Using,

- **To Create new file – touch filename.sh**
- **To edit file- nano filename.sh**

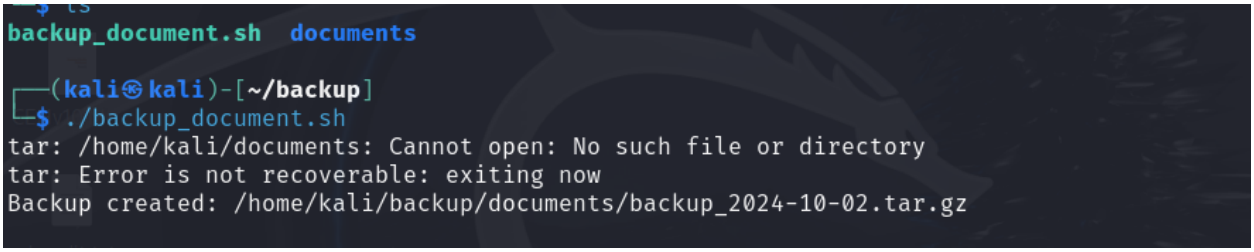


Step 03: change file mode to execute format using,

- **chmod +x backup_document.sh**

step 04: execute and run the script, using

- **./backup_document. sh**



SSH (Secure Shell)

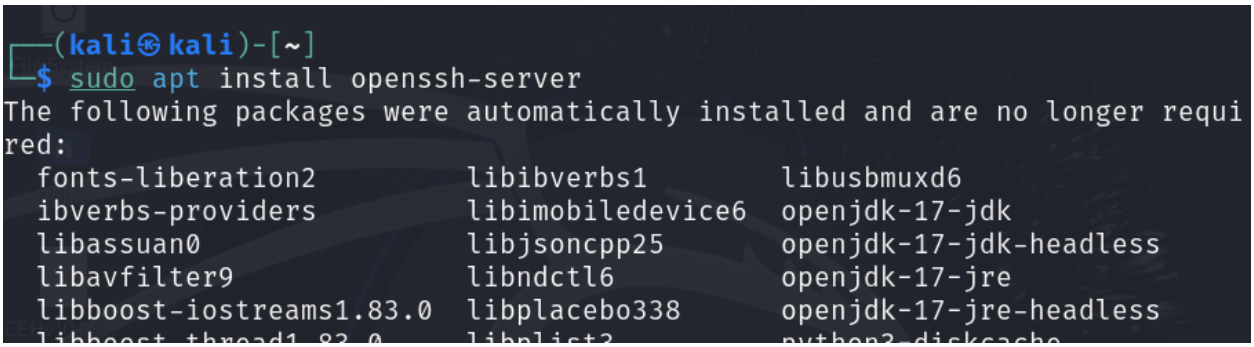
Connect to virtual machine remotely using an SSH client from another computer.

Step 01: update your system**,** using

- **sudo apt update**

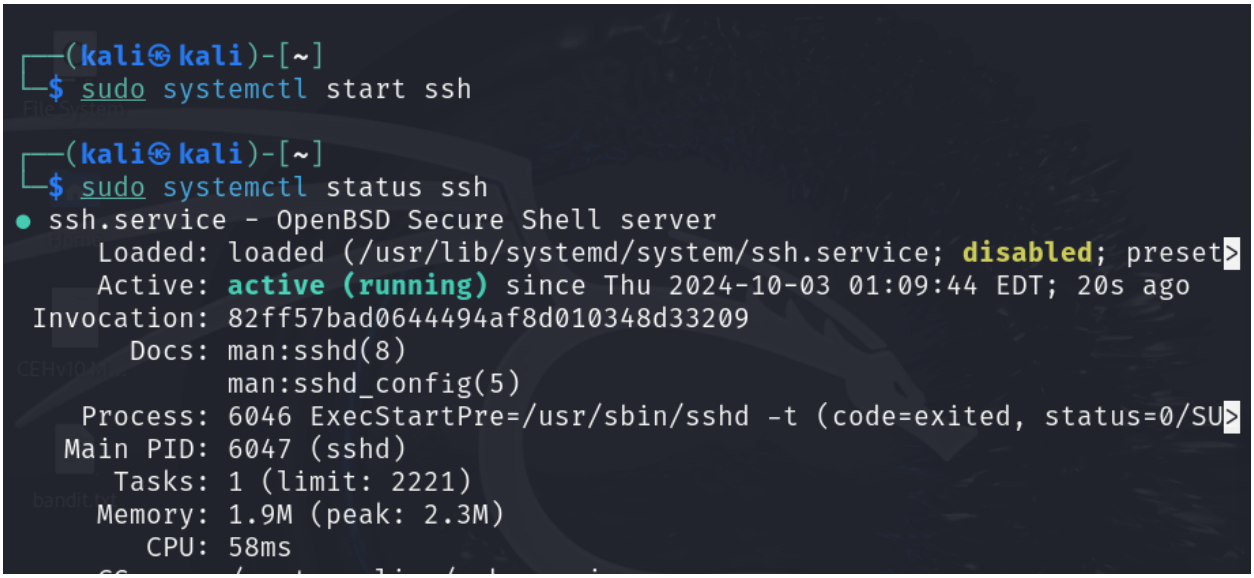
Step 02: install SSH Server, using

- **sudo apt install openssh-server**



Step 03: Start or restart SSH Server, using

- **sudo systemctl start ssh**



iptables and ACLs

Basic firewall rules using iptables to allow specific ports and services while blocking unwanted traffic.

Step 01: Set the Iptables rules. Using

- **sudo iptables -A INPUT -p tcp -- dport 22 -j ACCEPT**
- **sudo iptables -A INPUT -p tcp -- dport 443 -j ACCEPT**
- **sudo iptables -A INPUT -p icmp -- icmp-type echo-request -j ACCEPT**

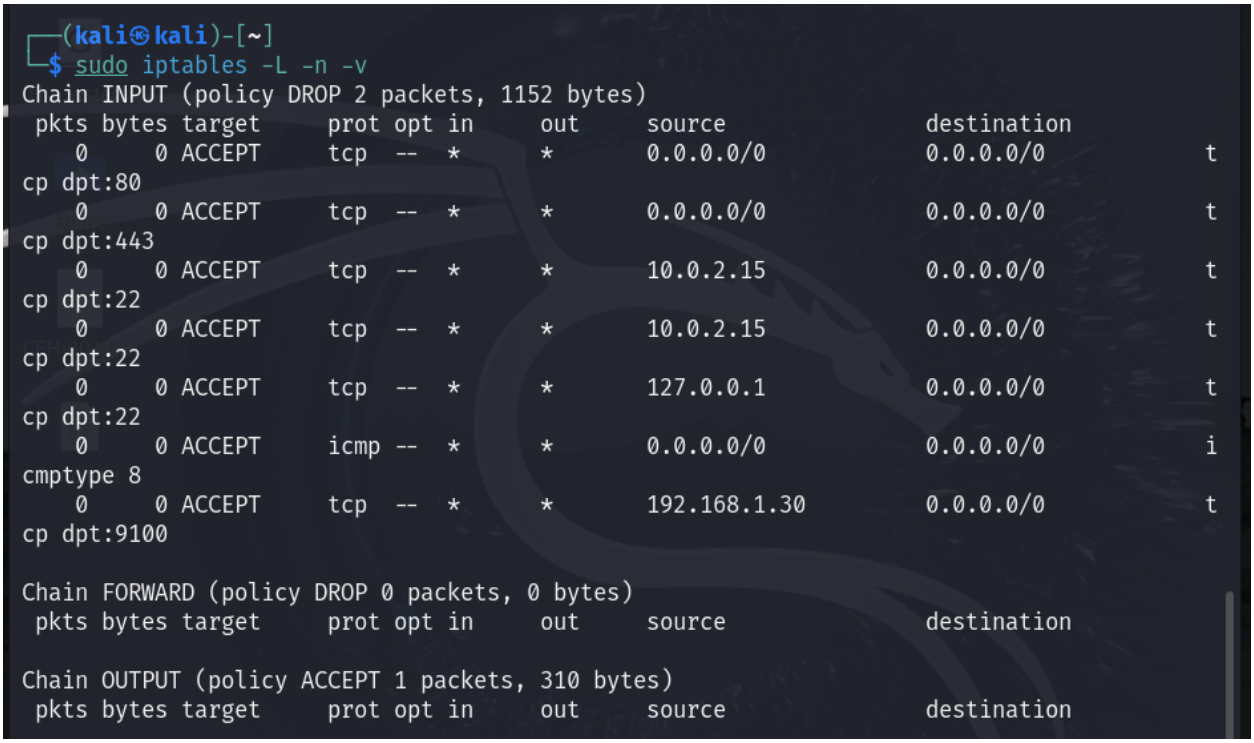
```
┌──(kali㉿kali)-[~]
└─$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT

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

┌──(kali㉿kali)-[~]
└─$ sudo iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
```

Step 02: Set the trusted ip address

```
┌──(kali㉿kali)-[~]
└─$ sudo iptables -A INPUT -p tcp -s 10.0.2.15 --dport 9100 -j ACCEPT
```

Step 03: check the available iptables using,

- **sudo iptables -L -n -v**



Best practices for security aspects of network interface configuration

There are some best practices in security aspects of network.

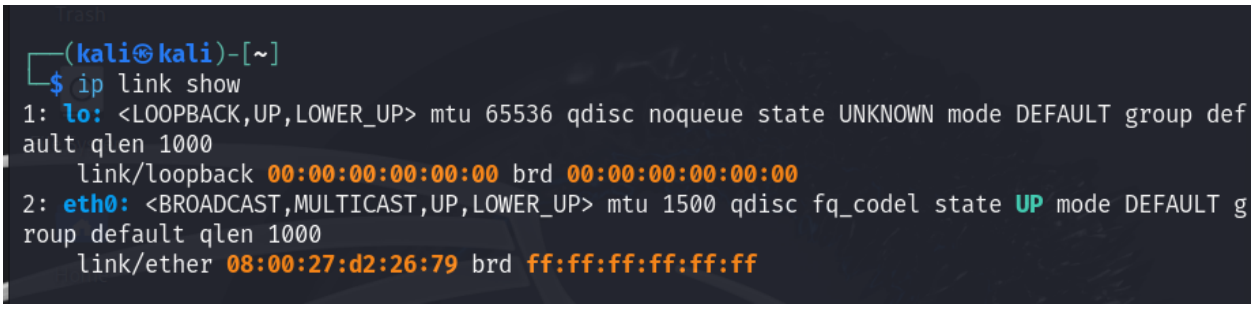
1. Disable unsend Network interfaces
2. Configure a firewall
3. Monitor network traffic
4. Disable IPv6 it not used
5. Enable logging and auditing

Disable and unsend Network interface

Disabling unused interfaces reduces the attack surface.

Step 01: List network interface, to using

- **Ip link show**



Step 02: identify unsend interfaces

Step 03: disable unsend interfaces, using

- **sudo ip link set <interface_name> down**

step 04: Restart Networking, using

- **sudo systemctl restart networking**

Configuration firewall

A firewall helps filter incoming and outgoing traffic, protecting the system from unauthorized access.

Step 01: Install UFW, using

- **sudo apt install ufw**

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install ufw
[sudo] password for kali:
The following packages were automatically installed and are no longer required:
  fonts-liberation2         libibverbs1         libusbmuxd6
  ibverbs-providers         libimobiledevice6   openjdk-17-jdk
  libassuan0                libjsoncpp25        openjdk-17-jdk-headless
  libavfilter9              libndctl6           openjdk-17-jre
  libboost-iostreams1.83.0  libplacebo338       openjdk-17-jre-headless
  libboost-thread1.83.0     libplist3           python3-diskcache
  libcephfs2                libpmem1            python3-mistune0
  libdaxctl1                libpostproc57       python3-pendulum
  libgeos3.12.1t64          librados2           python3-pytzdata
  libgfapi0                 librdmacm1t64       rwho
  libgfrpc0                 libre2-10           rwhod
  libgfxdr0                 libroc0.3
  libglusterfs0             libu2f-udev
Use 'sudo apt autoremove' to remove them.

Installing:
  ufw

Suggested packages:
  rsyslog
```

Step 02: Enable UFW, using

- **sudo ufw enable**

Step 03: Set default policies, using

- **sudo ufw default deny incoming**
- **sudo ufw default allow outgoing**

Step 04: Allowing specific Services, using

- **sudo ufw allow ssh**

Step 05: Check status, using

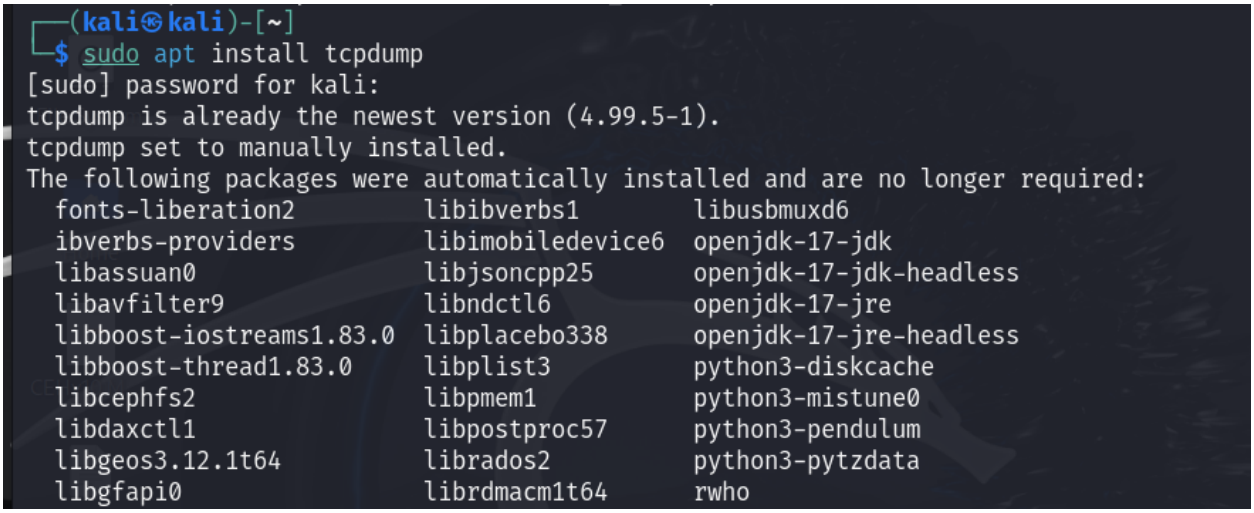
- **sudo ufw status verbose**

Monitor network traffic

Monitoring traffic helps in early detection of anomalies and potential breaches.

Step 01: Install tcpdump, using

- **sudo apt install tcpdump**



Step 02: Capture Network Traffic, using

- **Sudo tcpdump -i <interface_name>**

Step 03: Save Captured data, using

- **sudo tcpdump -i <interface_name> -w capture.pcap**

step 04: Install a Monitoring tool, using

- **sudo apt install iftop**

Step 05: Run iftop, using

- **sudo iftop -i <interface_name>**

Disable IPv6 if Not Used

If your network does not use IPv6, disabling it can reduce complexity and potential vulnerabilities.

Step 01: Check Current IPv6 Status, using

- **Ip a | grep inet6**

Step 02: Disable IPv6, using

- **sudo nano /etc/sysctl.conf**

```
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html
# for what other values do
#kernel.sysrq=438

net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

Step 03: Apply the Changes, using

- **sudo sysctl -p**

```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/sysctl.conf

┌──(kali㉿kali)-[~]
└─$ sudo sysctl -p
net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

Step 04: Verify IPv6 is Disabled, using

- **Ip a | grep inet6**

Enable Logging and Auditing

Logging network activity helps in detecting unusual patterns that may indicate a security breach.

Step 01: Install and Configure, using

- sudo apt install rsyslog

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install rsyslog
The following packages were automatically installed and are no longer required:
  fonts-liberation2         libibverbs1         libusbmuxd6
  ibverbs-providers         libimobiledevice6   openjdk-17-jdk
  libassuan0                libjsoncpp25        openjdk-17-jdk-headless
  libavfilter9              libndctl6           openjdk-17-jre
  libboost-iostreams1.83.0  libplacebo338       openjdk-17-jre-headless
  libboost-thread1.83.0     libplist3           python3-diskcache
  libcephfs2                libpmem1            python3-mistune0
  libdaxctl1                libpostproc57       python3-pendulum
  libgeos3.12.1t64          librados2           python3-pytzdata
  libgfapi0                 librdmacm1t64       rwho
  libgfrpc0                 libre2-10           rwhod
  libgfxdr0                 libroc0.3
  libglusterfs0             libu2f-udev
Use 'sudo apt autoremove' to remove them.

Installing:
  rsyslog

Installing dependencies:
  libestr0  libfastjson4  liblognorm5
```

Step 02: Edit the configuration file, using

- **sudo nano /etc/rsyslog.conf**

Step 03: Edit the logrotate configuration for /var/log/syslog, using

- **sudo nano /etc/logrotate.d/rsyslog**

Step 04: Restart Rsyslog, using

- **sudo systemctl restart rsyslog**