

IT23440418



SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

YEAR 2, SEMESTER 1

INDIVIDUAL ASSIGNMENT

IE2012 – SYSTEMS AND NETWORK PROGRAMMING

IT23440418

D.O.J. Silva

TERMS OF REFERENCE

This report is presented to meet the criteria of the Sri Lanka Institute of Information Technology's IE2012 – Systems and Network Programming.

Table of Content

1. Basics of the Linux Environment	4
1.1. Installing Kali Linux	4
1.2. Command Line Introduction	8
2. DHCP, DNS, and NTP Services	21
2.1. Understanding and Configuring DHCP	21
2.2. Understanding and Configuring DNS	28
2.3. Understanding and Configuring NTP	34
3. Shell Scripting and Security	39
3.1. Shell Scripting	39
3.2. SSH (Secure Shell) Configuration	45
3.3. Network Traffic Management Using iptables and ACLs	48
4. Best Practices for Network Interface Configuration Security	55
4.1. Ensuring OpenSSH Server Security	55
4.2. Securing the iptables Rules File: Ownership and Permission.....	57
4.3. Implementing Auditing Software for Security of the iptables Rule File	58
4.4. Enhancing Security by Disabling IP Forwarding on Linux....	60
4.5. Transitioning from Dynamic to Static IP Addresses	61

1. Basics of Linux Environment

1.1 Installing Kali Linux

Kali Linux can be installed as normal operating system or a virtual operating system, which can be run virtually on windows or any other operating system. To install it virtually the following prerequisites should be met:

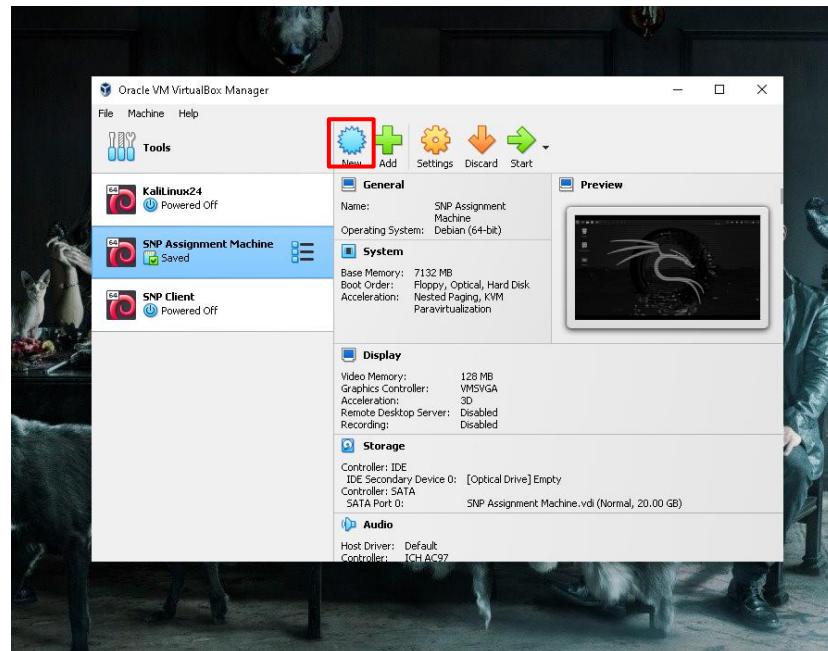
- A computer having at least 8GB of RAM, and a 4-6 Core CPU.
- A software to run virtual operating systems.
- Kali Linux operating system.

Oracle virtual box is a good choice for virtual operating system, because its free and open source, while offering cross-compatibility and many other features. Oracle virtual box can be downloaded from the official website which will be linked [here](#).

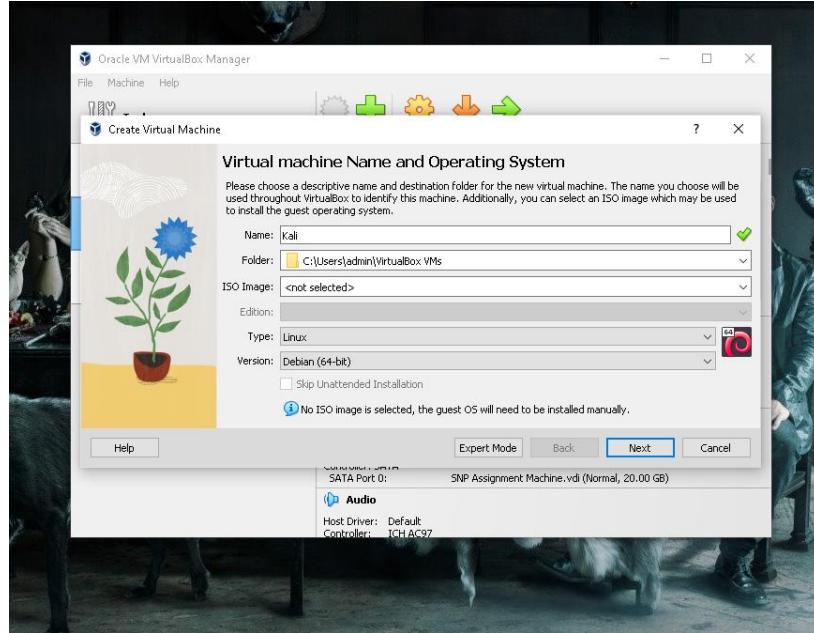
Kali Linux OS can be downloaded either as an ISO file for custom installation, or as a pre-installed version to be run on virtual box software. The difference lies between the installation, while the ISO file offers the user to customize the installation, the pre-installed version is ready to use. Both can be downloaded from official website, which will be linked [here](#).

After the above prerequisites are met, Kali Linux can be installed with the following steps.

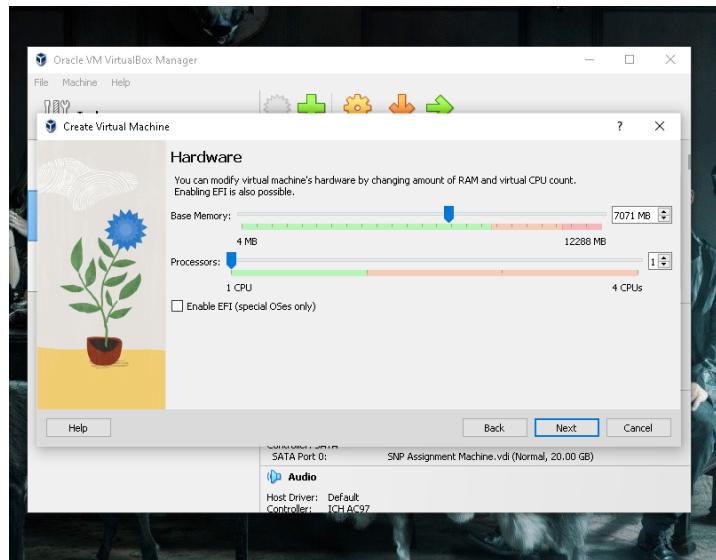
1. Open Oracle virtual box.
2. Select New option to create a new virtual machine on VirtualBox.



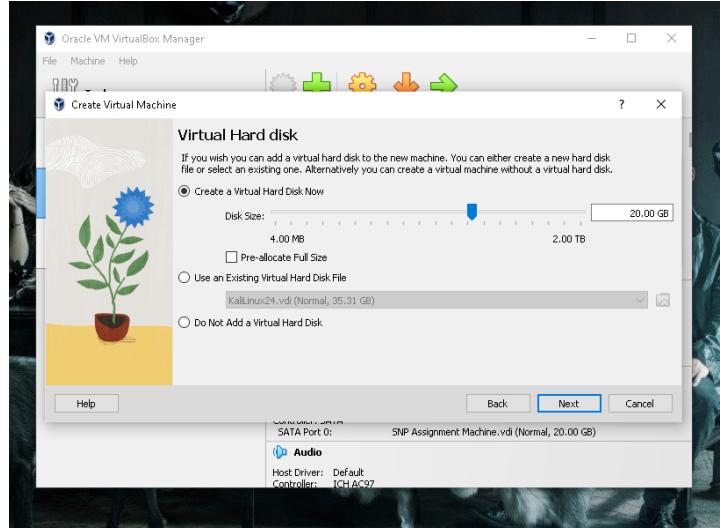
3. Name the machine and select location, and select type as Linux and Version debain 64 (as I use Kali Linux distribution). Then click next.



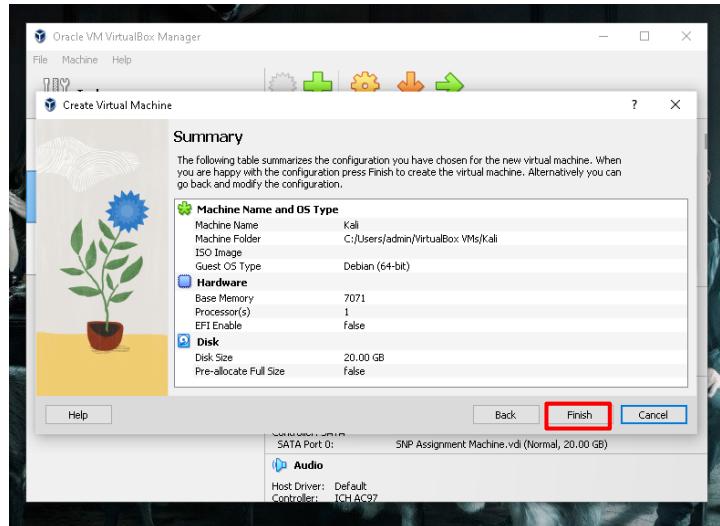
4. Allocate memory and CPU cores. It is good to select a number within green color range.



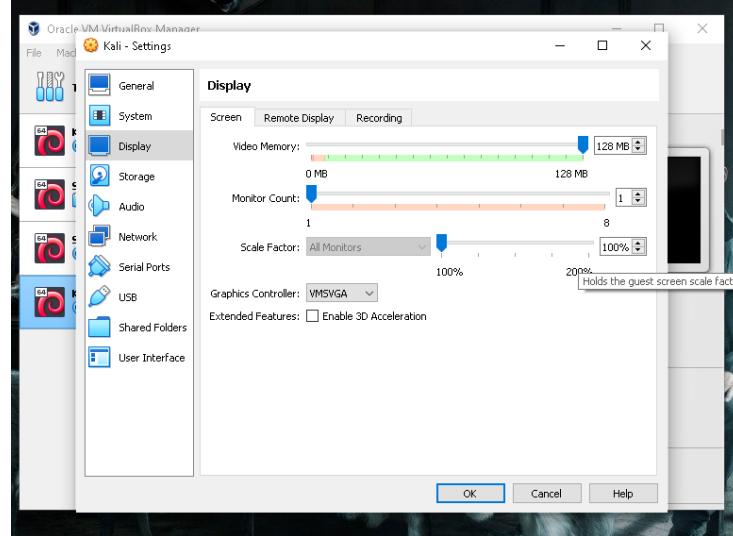
5. Then allocated disk size to virtual machine. It is good to allocate about 20GB.



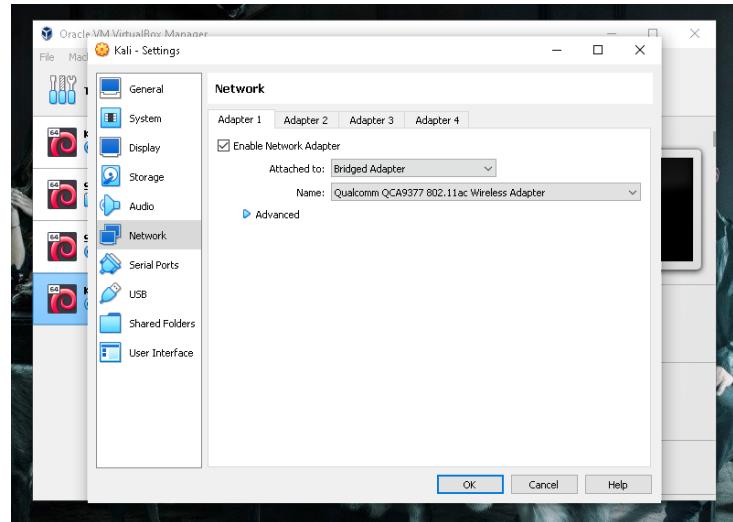
6. Then click finish.



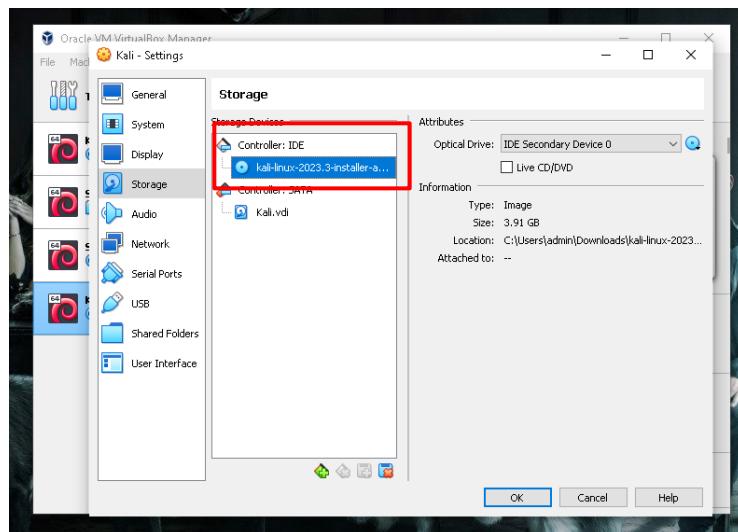
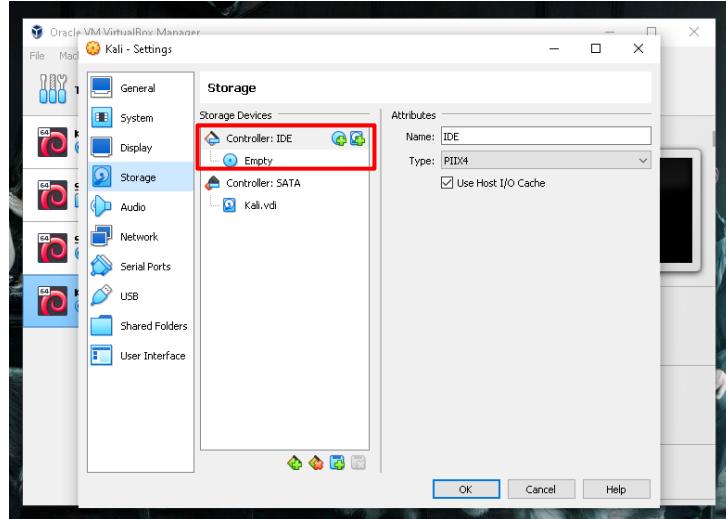
7. Then click settings and go to display option and allocate video memory.



8. Configure the network settings.



9. Mount an OS installation ISO file.



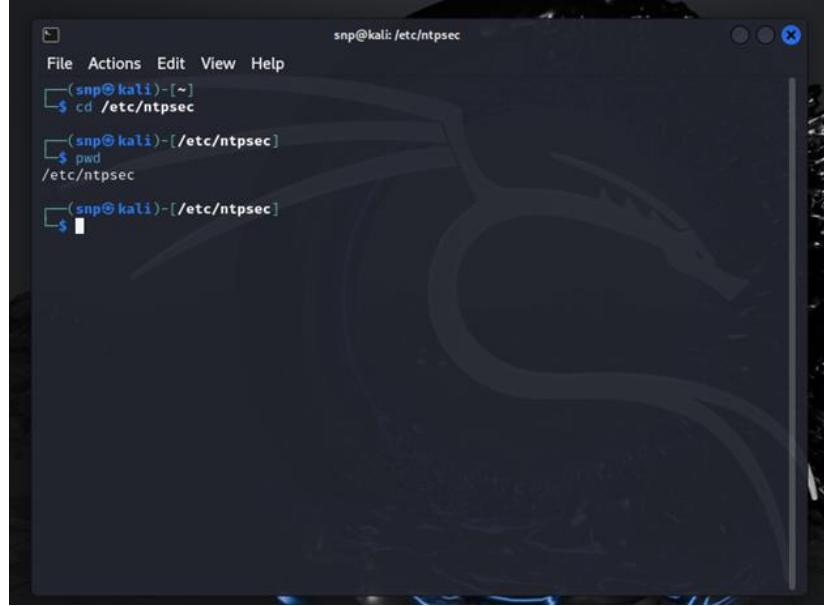
10. Start the VM and follow the OS installation steps.
11. Log in with the username and password given in OS installation steps.

1.2 Command Line Introduction

Navigation Commands

cd - The cd command in Linux is used to change the current directory in the terminal. It allows you to navigate between directories in the file system.

pwd - The pwd command stands for Print Working Directory. It displays the full path of the current directory you are in.

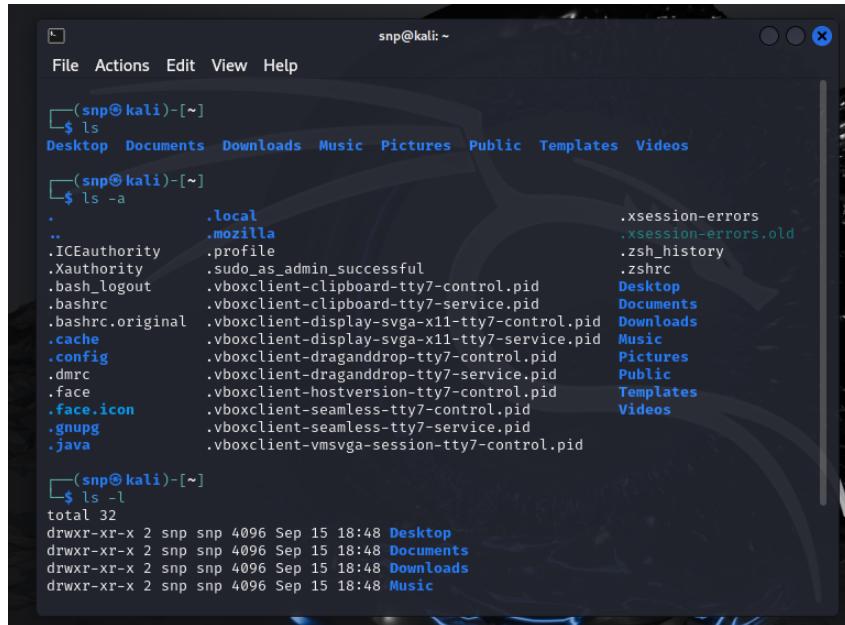


```
snp@kali: /etc/ntpsec
File Actions Edit View Help
(snp@kali)~]
$ cd /etc/ntpsec
(snp@kali) [/etc/ntpsec]
$ pwd
/etc/ntpsec
(snp@kali) [/etc/ntpsec]
$ [
```

ls - used to list files and directories in the current directory or a specified directory.

ls -a - lists all files and directories, including hidden ones, in the current directory. Hidden files and directories in Linux are those that start with a dot (.) .

ls -l - lists files and directories in a detailed format, showing additional information such as permissions, number of links, owner, group, file size, and last modification date.

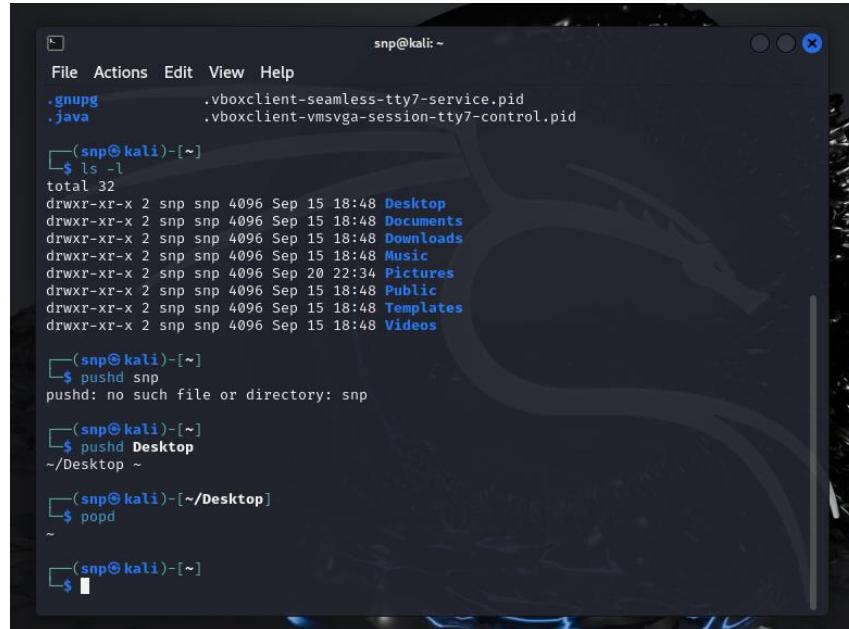


```
snp@kali: ~
File Actions Edit View Help
(snp@kali)~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
(snp@kali)~]
$ ls -a
. .local .profile .xsession-errors
.. .mozilla .sudo_as_admin_successful .xsession-errors.old
.Xauthority .profile .zsh_history
.Xauthority .sudo_as_admin_successful .zshrc
.bash_logout .vboxclient-clipboard-tty7-control.pid Desktop
.bashrc .vboxclient-clipboard-tty7-service.pid Documents
.bashrc.original .vboxclient-display-svga-x11-tty7-control.pid Downloads
.cache .vboxclient-display-svga-x11-tty7-service.pid Music
.config .vboxclient-draganddrop-tty7-control.pid Pictures
.dmrcc .vboxclient-draganddrop-tty7-service.pid Public
.face .vboxclient-hostversion-tty7-control.pid Templates
.face.icon .vboxclient-seamless-tty7-control.pid Videos
.gnupg .vboxclient-seamless-tty7-service.pid
.java .vboxclient-vmsvga-session-tty7-control.pid

(snp@kali)~]
$ ls -l
total 32
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Desktop
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Documents
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Downloads
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Music
```

pushd - used to change the current directory and save the previous directory on a stack.

popd - used to change the current directory back to the directory that was most recently saved on the directory stack by the pushd command.



```
snp@kali: ~
File Actions Edit View Help
.gnugp .vboxclient-seamless-tty7-service.pid
.java .vboxclient-vmsvga-session-tty7-control.pid
(snp@kali)-[~]
$ ls -l
total 32
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Desktop
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Documents
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Downloads
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Music
drwxr-xr-x 2 snp snp 4096 Sep 20 22:34 Pictures
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Public
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Templates
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Videos

(snp@kali)-[~]
$ pushd snp
pushd: no such file or directory: snp

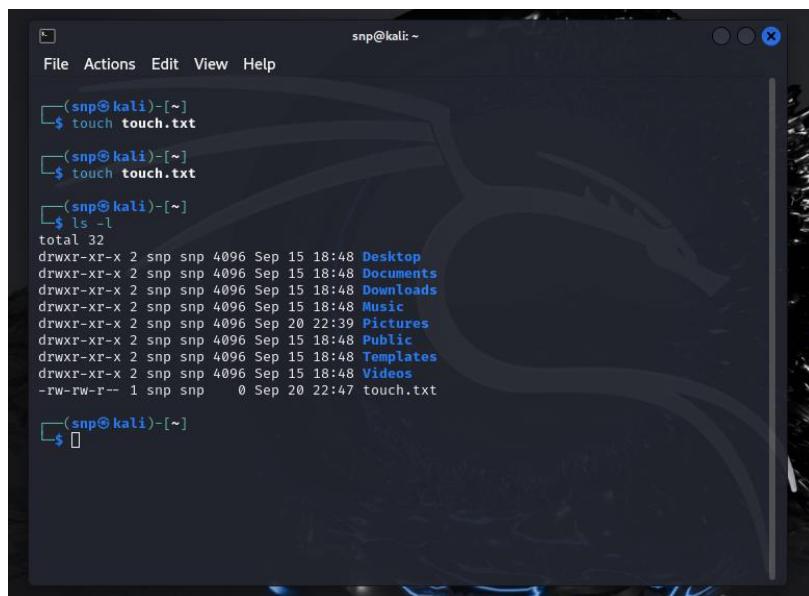
(snp@kali)-[~]
$ pushd Desktop
~/Desktop ~

(snp@kali)-[~/Desktop]
$ popd
~

(snp@kali)-[~]
$
```

File Manipulation Commands

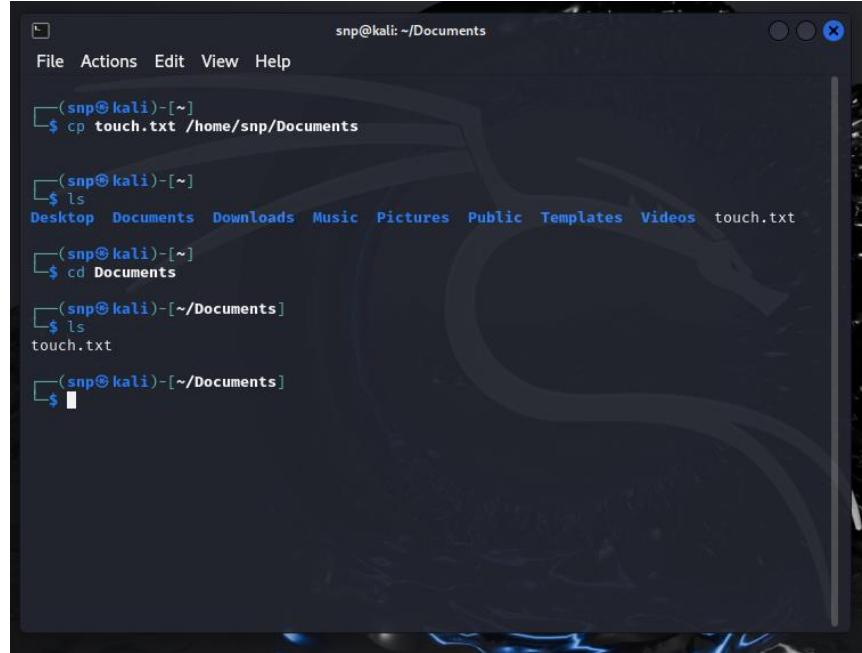
touch - used in Linux to create empty files or to update the timestamps of existing files. It's a simple way to create a new file without needing to open a text editor.



```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ touch touch.txt
(snp@kali)-[~]
$ touch touch.txt
(snp@kali)-[~]
$ ls -l
total 32
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Desktop
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Documents
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Downloads
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Music
drwxr-xr-x 2 snp snp 4096 Sep 20 22:39 Pictures
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Public
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Templates
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Videos
-rw-rw-r-- 1 snp snp 0 Sep 20 22:47 touch.txt

(snp@kali)-[~]
$
```

cp - used to copy files and directories in Linux. It allows you to create a duplicate of a file or directory at a specified location.



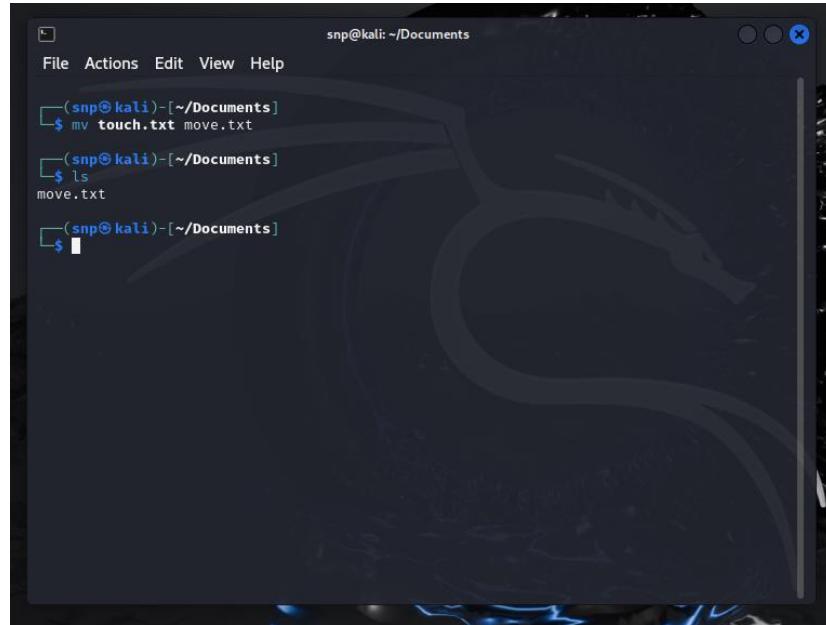
```
snp@kali: ~/Documents
File Actions Edit View Help
└─(snp@kali)~]
$ cp touch.txt /home/snp/Documents

└─(snp@kali)~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos touch.txt

└─(snp@kali)~]
$ cd Documents
└─(snp@kali)~/Documents]
$ ls
touch.txt

└─(snp@kali)~/Documents]
```

mv - used to move or rename files and directories in Linux. It allows you to change the location of a file or directory or give it a new name.

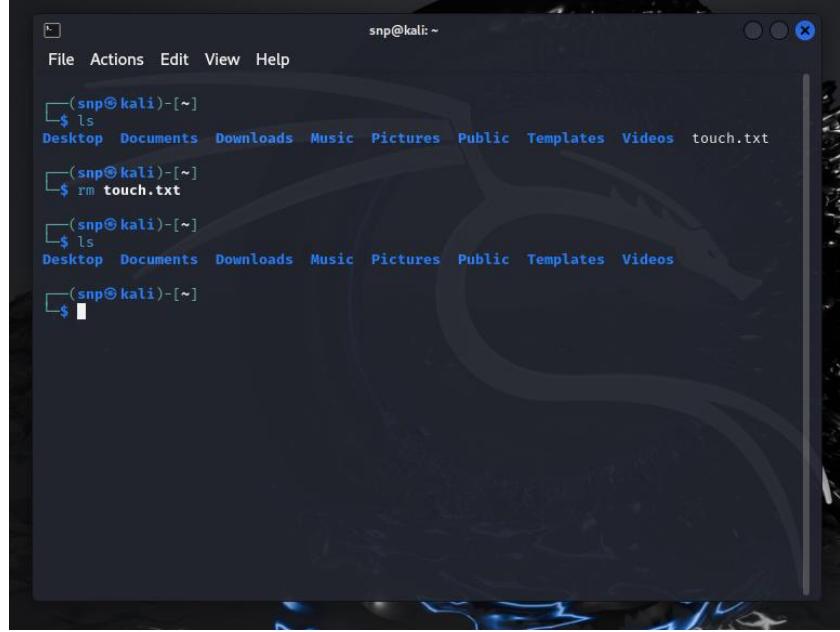


```
snp@kali: ~/Documents
File Actions Edit View Help
└─(snp@kali)~/Documents]
$ mv touch.txt move.txt

└─(snp@kali)~/Documents]
$ ls
move.txt

└─(snp@kali)~/Documents]
```

rm - used to remove files and directories in Linux. It permanently deletes the specified files or directories.



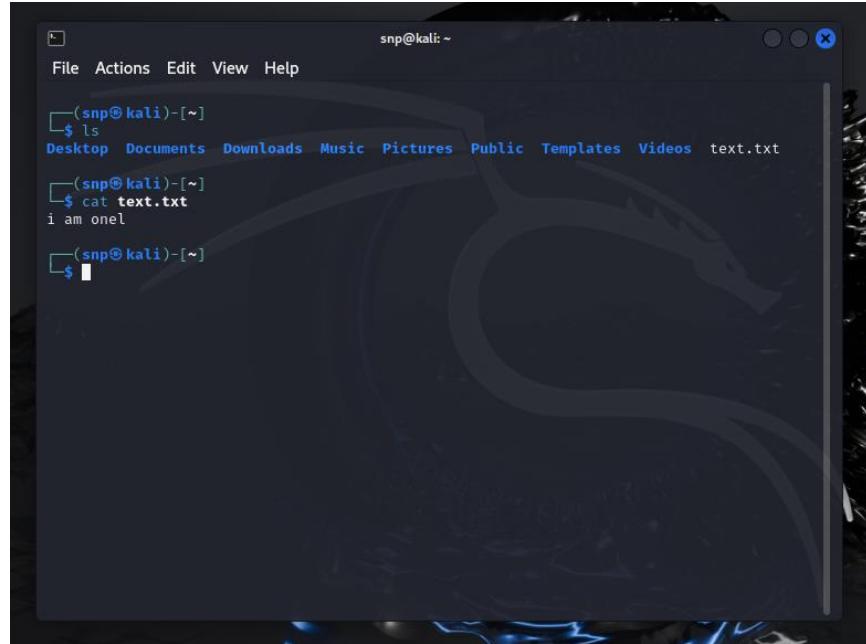
```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos touch.txt
└─(snp@kali)─[~]
$ rm touch.txt
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
└─(snp@kali)─[~]
$
```

A screenshot of a terminal window titled "snp@kali: ~". The window has a dark background with a faint dragon watermark. The terminal shows the following session:

```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos touch.txt
└─(snp@kali)─[~]
$ rm touch.txt
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
└─(snp@kali)─[~]
$
```

The user first lists the contents of the current directory, which includes a file named "touch.txt". Then, they run the command "rm touch.txt" to delete the file. After the deletion, the file is no longer listed when they run "ls" again.

cat - used to display the contents of files, combine multiple files, and create new files in Linux.



```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos text.txt
└─(snp@kali)─[~]
$ cat text.txt
i am one!
└─(snp@kali)─[~]
$
```

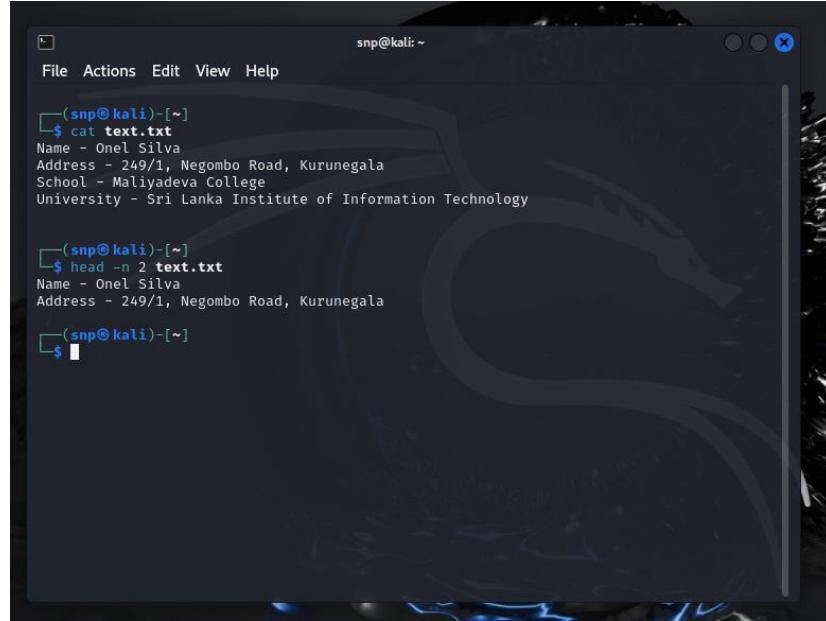
A screenshot of a terminal window titled "snp@kali: ~". The window has a dark background with a faint dragon watermark. The terminal shows the following session:

```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos text.txt
└─(snp@kali)─[~]
$ cat text.txt
i am one!
└─(snp@kali)─[~]
$
```

The user lists the files in the current directory, finding a file named "text.txt". They then run the command "cat text.txt" to display its contents, which are "i am one!".

basic Linux commands

head - used to display the first part of files. It shows the first 10 lines of a file by default, but you can specify how many lines or bytes to display.



The screenshot shows a terminal window titled "snp@kali: ~". The user has run the command `cat text.txt`, which displays a block of text about a person's details. Then, the user runs the command `head -n 2 text.txt`, which shows only the first two lines of the text file.

```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ cat text.txt
Name - Onel Silva
Address - 249/1, Negombo Road, Kurunegala
School - Maliyadeva College
University - Sri Lanka Institute of Information Technology

(snp@kali)-[~]
$ head -n 2 text.txt
Name - Onel Silva
Address - 249/1, Negombo Road, Kurunegala

(snp@kali)-[~]
$
```

tail - used to display the last part of files. It shows the last 10 lines of a file by default, but you can specify how many lines or bytes to display.



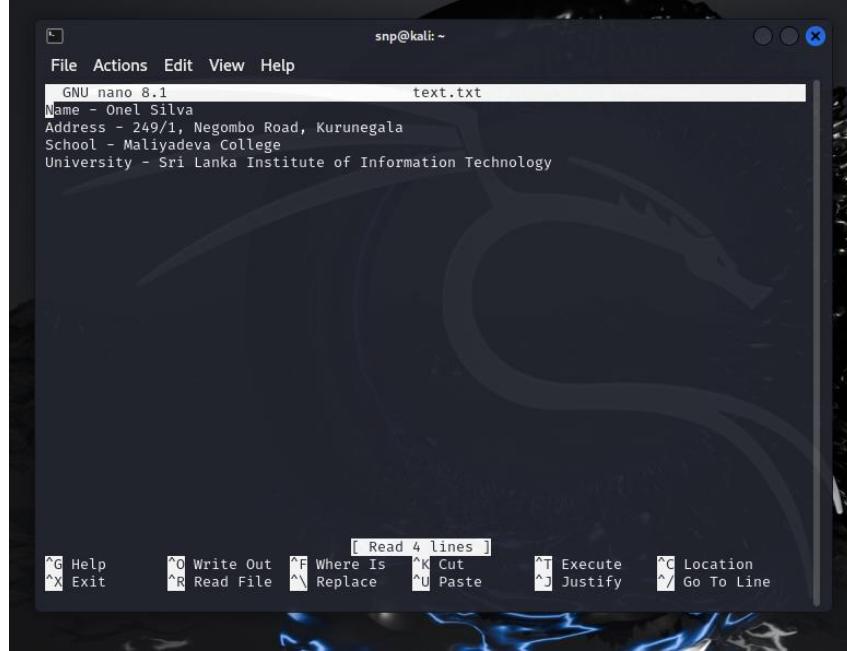
The screenshot shows a terminal window titled "snp@kali: ~". The user has run the command `cat text.txt`, which displays a block of text about a person's details. Then, the user runs the command `tail -n 2 text.txt`, which shows only the last two lines of the text file.

```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ cat text.txt
Name - Onel Silva
Address - 249/1, Negombo Road, Kurunegala
School - Maliyadeva College
University - Sri Lanka Institute of Information Technology

(snp@kali)-[~]
$ tail -n 2 text.txt
School - Maliyadeva College
University - Sri Lanka Institute of Information Technology

(snp@kali)-[~]
$
```

nano - command-line text editor and opens the editor.



chmod - Change the permission of a file or directory. It controls who can read, write, or execute a file by setting or modifying permissions for the file owner, group, and others.

A screenshot of a terminal window titled "snp@kali: ~". The user runs the command "ls -l" to list files, showing a file named "text.txt" with permissions "-rw-rw-r--". The user then runs the command "chmod 700 text.txt" to change the permissions to "rwx-----". After changing the permissions, the user runs "ls -l" again, which now shows the file with permissions "-rwx-----".

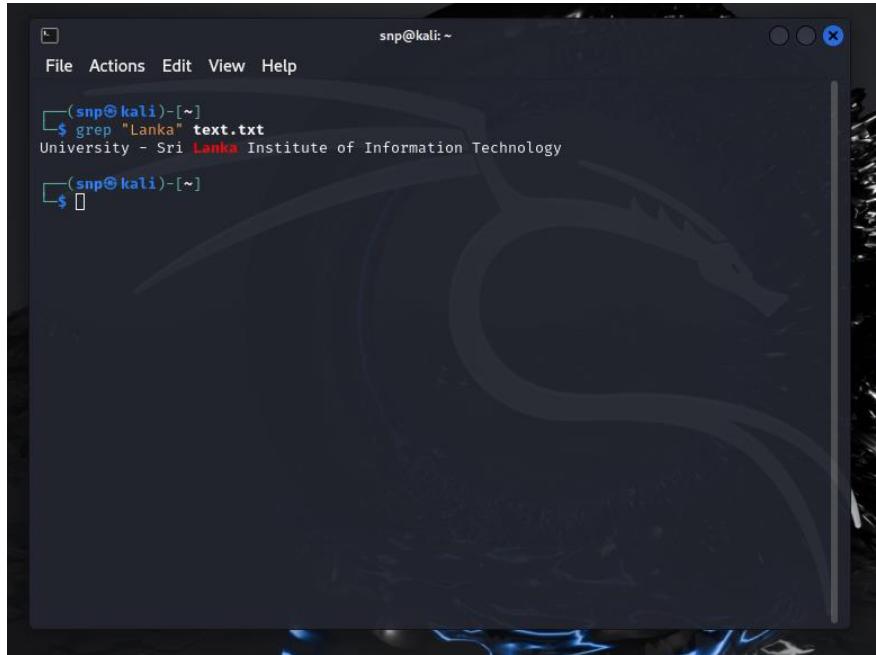
```
total 36
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Desktop
drwxr-xr-x 2 snp snp 4096 Sep 20 22:49 Documents
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Downloads
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Music
drwxr-xr-x 2 snp snp 4096 Sep 20 23:08 Pictures
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Public
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Templates
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Videos
-rw-rw-r-- 1 snp snp 147 Sep 20 23:04 text.txt

(snp@kali)-[~]
$ chmod 700 text.txt

(snp@kali)-[~]
$ ls -l
total 36
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Desktop
drwxr-xr-x 2 snp snp 4096 Sep 20 22:49 Documents
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Downloads
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Music
drwxr-xr-x 2 snp snp 4096 Sep 20 23:08 Pictures
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Public
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Templates
drwxr-xr-x 2 snp snp 4096 Sep 15 18:48 Videos
-rwx----- 1 snp snp 147 Sep 20 23:04 text.txt

(snp@kali)-[~]
$
```

grep - used to search for specific patterns or strings within files. It scans the input and outputs lines that match the pattern you specify.

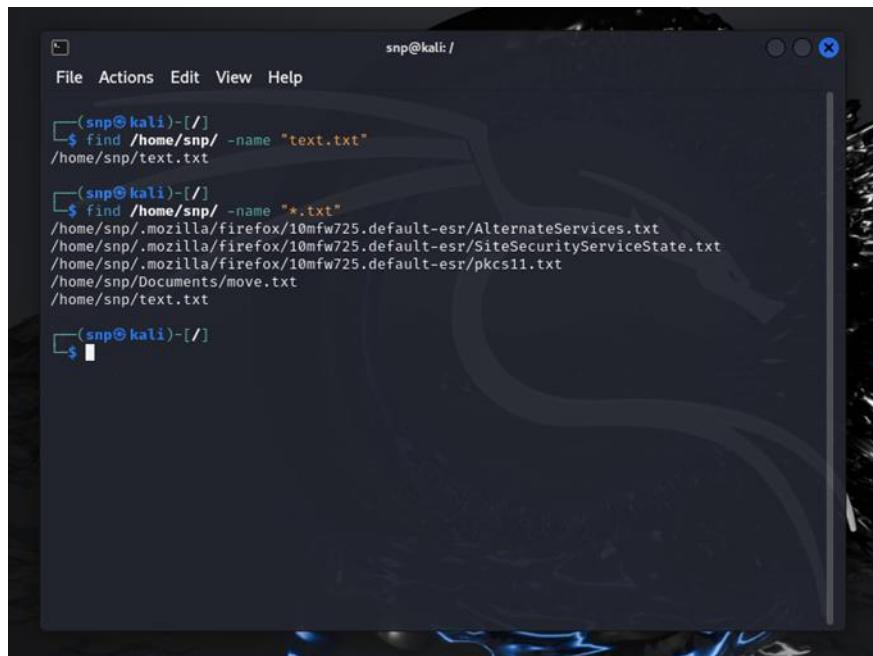


A screenshot of a terminal window titled "snp@kali: ~". The terminal shows the command \$ grep "Lanka" text.txt being run, which outputs the line "University - Sri Lanka Institute of Information Technology". The background of the terminal window features a dark, stylized dragon logo.

```
(snp@kali)-[~]
$ grep "Lanka" text.txt
University - Sri Lanka Institute of Information Technology
(snp@kali)-[~]
$
```

find - used to search for files and directories in a directory hierarchy.

find /path/to/search -name "*. extension" – used to search for files and directories using extensions.



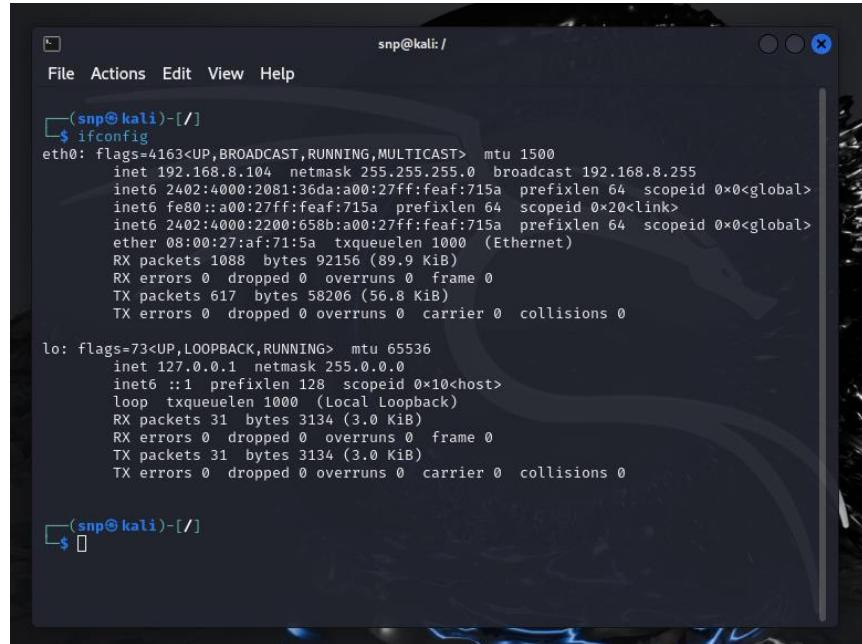
A screenshot of a terminal window titled "snp@kali: /". The terminal shows the command \$ find /home/snp/ -name "text.txt" being run, which outputs the file path /home/snp/text.txt. Below it, the command \$ find /home/snp/ -name "*.txt" is run, listing several text files from the user's home directory. The background of the terminal window features a dark, stylized dragon logo.

```
(snp@kali)-[/]
$ find /home/snp/ -name "text.txt"
/home/snp/text.txt

(snp@kali)-[/]
$ find /home/snp/ -name "*.txt"
/home/snp/.mozilla/firefox/10mfw725.default-esr/AlternateServices.txt
/home/snp/.mozilla/firefox/10mfw725.default-esr/SiteSecurityServiceState.txt
/home/snp/.mozilla/firefox/10mfw725.default-esr/pkcs11.txt
/home/snp/Documents/move.txt
/home/snp/text.txt

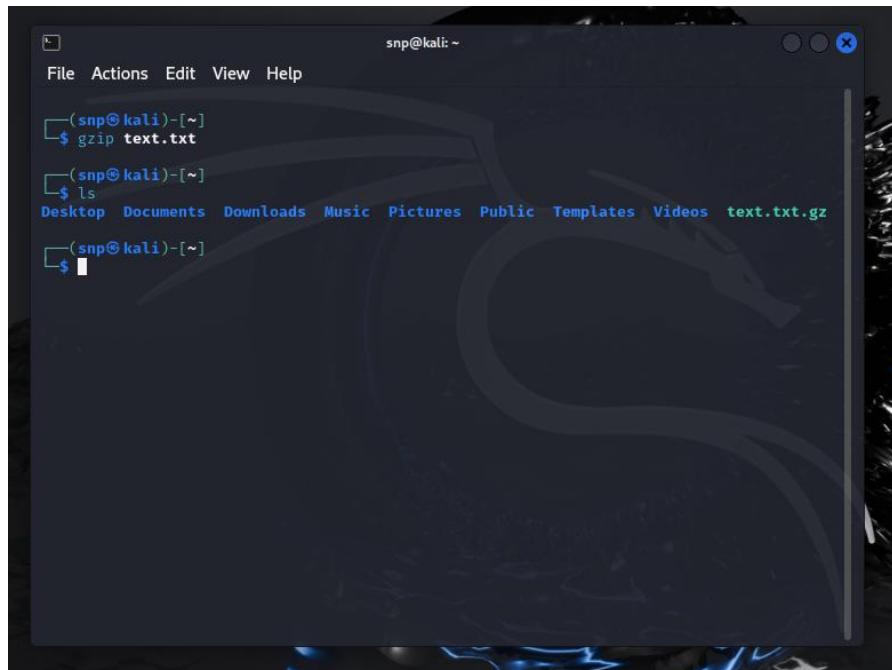
(snp@kali)-[/]
$
```

ifconfig - used to display network interface parameters. It provides information about a system's network interfaces, such as their IP addresses, MAC addresses, subnet masks, and more.



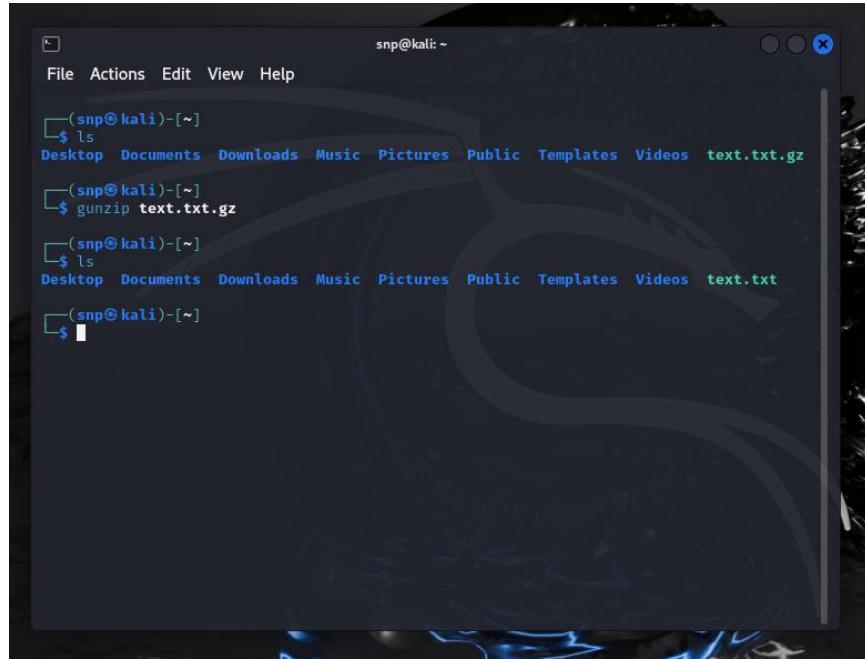
```
snp@kali: /  
File Actions Edit View Help  
└─(snp@kali)-[~]  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
        inet 192.168.8.104 netmask 255.255.255.0 broadcast 192.168.8.255  
        inet6 2402:4000:2081:36da:a00:27ff:feaf:715a prefixlen 64 scopeid 0x0<global>  
        inet6 fe80::a00:27ff:feaf:715a prefixlen 64 scopeid 0x20<link>  
        ether 08:00:27:af:71:5a txqueuelen 1000 (Ethernet)  
        RX packets 1088 bytes 92156 (89.9 KiB)  
        RX errors 0 dropped 0 overruns 0 frame 0  
        TX packets 617 bytes 58206 (56.8 KiB)  
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
        inet 127.0.0.1 netmask 255.0.0.0  
        inet6 ::1 prefixlen 128 scopeid 0x10<host>  
        loop txqueuelen 1000 (Local Loopback)  
        RX packets 31 bytes 3134 (3.0 KiB)  
        RX errors 0 dropped 0 overruns 0 frame 0  
        TX packets 31 bytes 3134 (3.0 KiB)  
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
└─(snp@kali)-[~]  
$
```

gzip - used to compress files, reducing their size using the GNU zip algorithm. One of the most common tools for file compression in Linux.



```
snp@kali: ~  
File Actions Edit View Help  
└─(snp@kali)-[~]  
$ gzip text.txt  
└─(snp@kali)-[~]  
$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos text.txt.gz  
└─(snp@kali)-[~]  
$
```

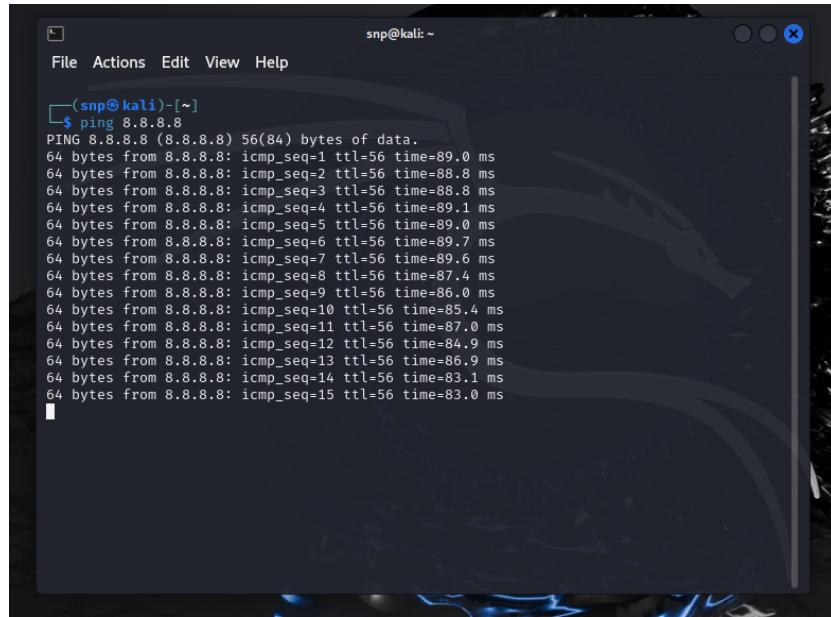
gunzip - used to decompress files that were compressed using the gzip. It restores the original file from its compressed gzip version.



```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos text.txt.gz
└─(snp@kali)─[~]
$ gunzip text.txt.gz
└─(snp@kali)─[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos text.txt
└─(snp@kali)─[~]
$
```

A screenshot of a terminal window titled "snp@kali: ~". The window shows a sequence of commands: listing files, running the gunzip command on a file named "text.txt.gz", and then listing files again to show that the compressed file has been removed and a new file "text.txt" has been created.

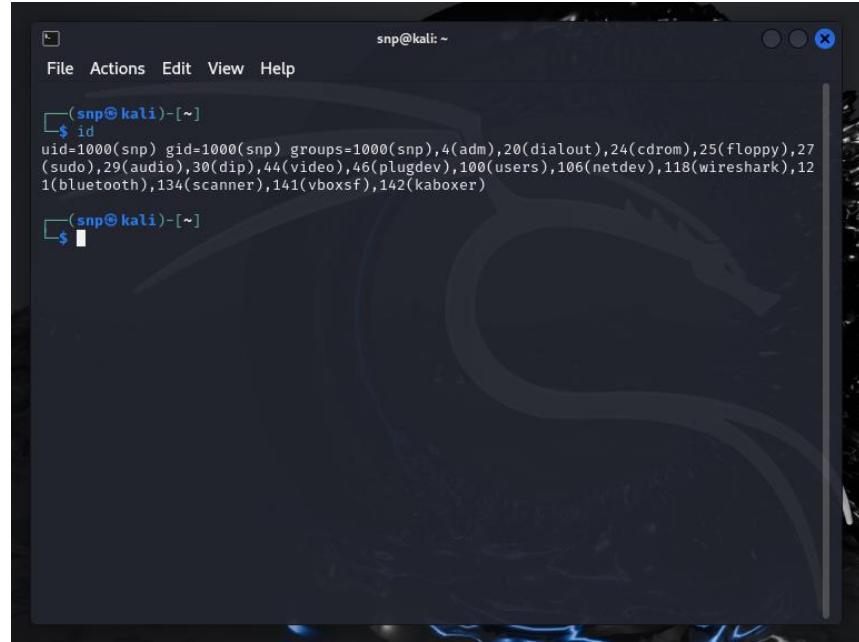
ping - Network diagnostic tool that tests the reachability of a host on a network. It works by sending ICMP Echo Request packets to the specified host and waits for an Echo Reply.



```
snp@kali: ~
File Actions Edit View Help
└─(snp@kali)─[~]
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=56 time=89.0 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=56 time=88.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=56 time=88.8 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=56 time=89.1 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=56 time=89.0 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=56 time=89.7 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=56 time=89.6 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=56 time=87.4 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=56 time=86.0 ms
64 bytes from 8.8.8.8: icmp_seq=10 ttl=56 time=85.4 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=56 time=87.0 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=56 time=84.9 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=56 time=86.9 ms
64 bytes from 8.8.8.8: icmp_seq=14 ttl=56 time=83.1 ms
64 bytes from 8.8.8.8: icmp_seq=15 ttl=56 time=83.0 ms
└─
```

A screenshot of a terminal window titled "snp@kali: ~". The window shows the "ping" command being run against the IP address 8.8.8.8. The output displays multiple ICMP Echo Request and Echo Reply messages, showing the round-trip time (time) for each packet.

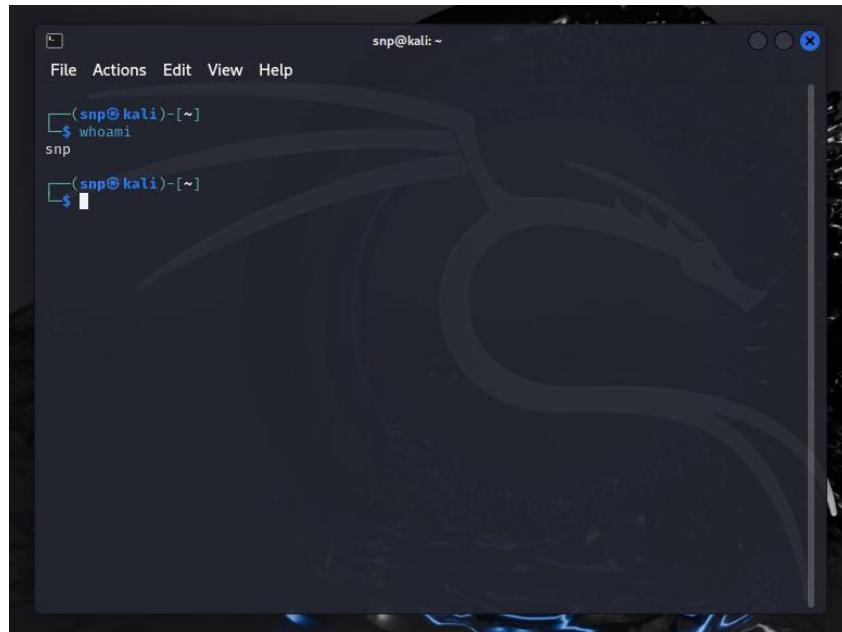
id - used to display user and group information for the current user or a specified user.



```
snp@kali: ~
File Actions Edit View Help
[(snp@kali)-[~]
$ id
uid=1000(snp) gid=1000(snp) groups=1000(snp),4(adm),20(dialout),24(cdrom),25(floppy),27
(sudo),29(audio),30(dip),44(video),46(plugdev),100(users),106(netdev),118(wireshark),12
1(bluetooth),134(scanner),141(vboxsf),142(kaboxer)
[(snp@kali)-[~]
$ ]
```

A terminal window titled "snp@kali: ~" showing the output of the "id" command. The command "id" is run at the prompt, and the output shows the user's UID (1000), GID (1000), and various group memberships. The window has a dark background with a faint Kali Linux logo watermark.

whoami - used to display the username of the currently logged in user.



```
snp@kali: ~
File Actions Edit View Help
[(snp@kali)-[~]
$ whoami
snp
[(snp@kali)-[~]
$ ]
```

A terminal window titled "snp@kali: ~" showing the output of the "whoami" command. The command "whoami" is run at the prompt, and the output displays the current user's name, "snp". The window has a dark background with a faint Kali Linux logo watermark.

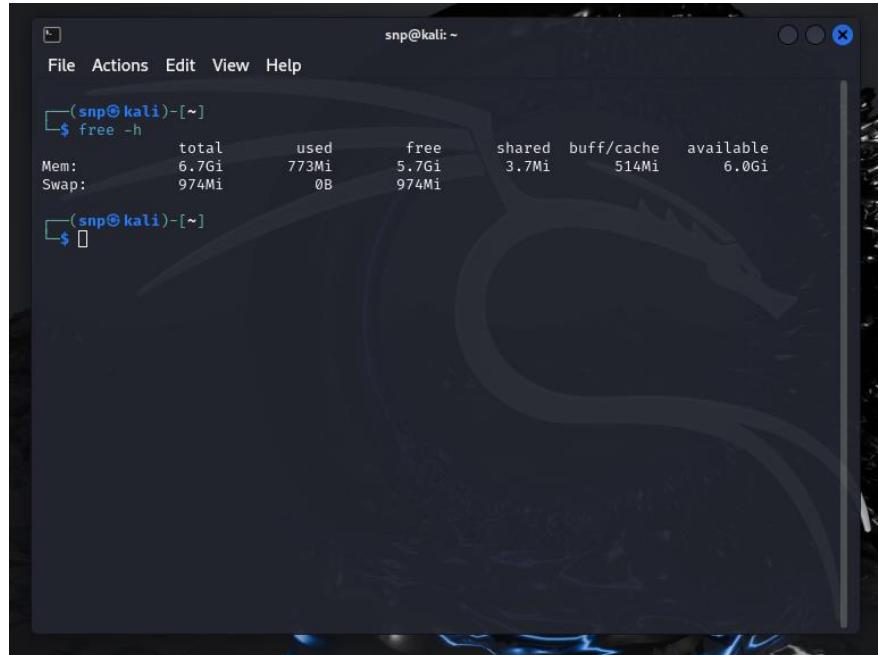
uptime - used to display how long the system has been running since its last reboot, along with information about system load averages.



The screenshot shows a terminal window titled "snp@kali: ~". The window contains the following text:

```
(snp@kali)-[~]
$ uptime
23:26:48 up 1:01, 2 users,  load average: 0.74, 0.27, 0.16
(snp@kali)-[~]
$
```

free -h - used to display information about the system's memory usage, amount of free and used memory. “-h” option stands for "human-readable" and formats the output in a more human readable way using units like MB or GB.

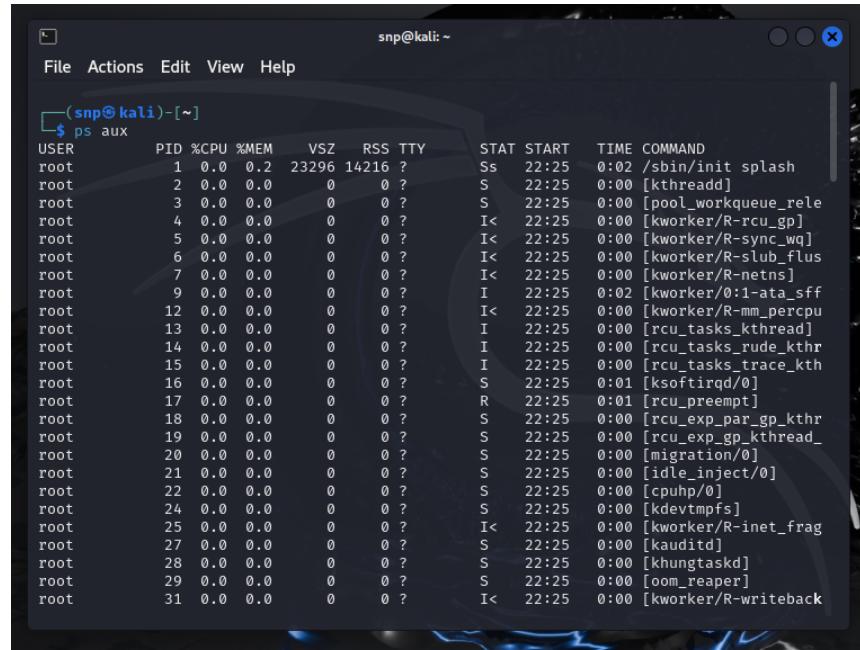


The screenshot shows a terminal window titled "snp@kali: ~". The window contains the following text:

```
(snp@kali)-[~]
$ free -h
              total        used        free      shared  buff/cache   available
Mem:       6.7Gi      773Mi      5.7Gi      3.7Mi      514Mi     6.0Gi
Swap:  974Mi          0B      974Mi

(snp@kali)-[~]
$
```

ps aux - used to display detailed information about currently running processes.



The screenshot shows a terminal window titled '(snp@kali)-[~]' with the command '\$ ps aux' entered. The output lists numerous processes running on the system, primarily managed by the kernel (root user). The columns displayed are USER, PID, %CPU, %MEM, VSZ, RSS, TTY, STAT, START, TIME, and COMMAND. The processes include system daemons like /sbin/init splash, kthreadd, pool_workqueue_rele, and various rcu_*_kthr and rcu_*_gp threads, along with other kernel-related tasks such as migration/0, idle_inject/0, and cpuhp/0.

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.2	23296	14216	?	Ss	22:25	0:02	/sbin/init splash
root	2	0.0	0.0	0	0	?	S	22:25	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	22:25	0:00	[pool_workqueue_rele]
root	4	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-rcu_gp]
root	5	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-sync_wa]
root	6	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-slub_flush]
root	7	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-netsns]
root	9	0.0	0.0	0	0	?	I	22:25	0:02	[kworker/0:1-ata_sff]
root	12	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-mm_percpu]
root	13	0.0	0.0	0	0	?	I	22:25	0:00	[rcu_tasks_kthread]
root	14	0.0	0.0	0	0	?	I	22:25	0:00	[rcu_tasks_rude_kthr]
root	15	0.0	0.0	0	0	?	I	22:25	0:00	[rcu_tasks_trace_kth]
root	16	0.0	0.0	0	0	?	S	22:25	0:01	[ksoftirqd/0]
root	17	0.0	0.0	0	0	?	R	22:25	0:01	[rcu_preempt]
root	18	0.0	0.0	0	0	?	S	22:25	0:00	[rcu_exp_par_gp_kthr]
root	19	0.0	0.0	0	0	?	S	22:25	0:00	[rcu_exp_gp_kthread]
root	20	0.0	0.0	0	0	?	S	22:25	0:00	[migration/0]
root	21	0.0	0.0	0	0	?	S	22:25	0:00	[idle_inject/0]
root	22	0.0	0.0	0	0	?	S	22:25	0:00	[cpuhp/0]
root	24	0.0	0.0	0	0	?	S	22:25	0:00	[kdevtmpfs]
root	25	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-inet_frag]
root	27	0.0	0.0	0	0	?	S	22:25	0:00	[kauditid]
root	28	0.0	0.0	0	0	?	S	22:25	0:00	[khungtaskd]
root	29	0.0	0.0	0	0	?	S	22:25	0:00	[oom_reaper]
root	31	0.0	0.0	0	0	?	I<	22:25	0:00	[kworker/R-writeback]

2. DHCP, DNS, NTP Services

2.1 Understanding and configuring DHCP

Role of DHCP in Network Configuration

DHCP is a network protocol that is used for the distribution of configuration parameters-such as IP addresses, subnet masks, default gateways, and so on-in an automated manner to clients on the network. The main purpose for which DHCP was designed is to connect devices to the network through automatic means, rather than having to enter the settings of the network manually[2]. Every computer on a network has an IP address that is unique to the computer.

There are 2 ways to assign an IP address to a computer connected to a network.

- Static IP
- Dynamic IP

Static IP

This mode of allocation requires user to manually configure the network settings for the devices.

This allocation mode consumes more time and is prone to errors since the configuration is done by the user.

Dynamic IP

This mode involves the client sending a request message to the server. After, the server shares the required network configuration and resources to the client and configures the device automatically. Then the client can access the internet and network services.

DHCP Dynamic Allocation Process

DHCP Discover

When the client, or device, connects onto the network, it does not yet have an IP address. So the client broadcasts a packet, which is called the DHCP Discover message, to discover which DHCP servers are available on that network. Within that message, it includes its MAC address and a request for an IP address [2].

DHCP Offer

Each DHCP server that receives the DHCP Discover message from the client replied with a DHCP Offer message. And this message includes an available IP address, subnet mask, lease time and other configurations. The offer message is sent to client's MAC address as a broadcast packet [2].

DHCP Request

In return, the client gets one or more DHCP offers from the servers. The client then selects one of the offers and responds with a DHCP Request, informing the server of which offer it has accepted. A DHCP request message contains the IP address that the client device is requesting [2].

DHCP Acknowledgment

The DHCP request is received by the DHCP server, which then sends a DHCP Acknowledgment message in return for acknowledging that the IP address has been assigned to the client. On reception of the DHCP Acknowledgment message, the client configures its network interface with the assigned IP address and other settings [2].

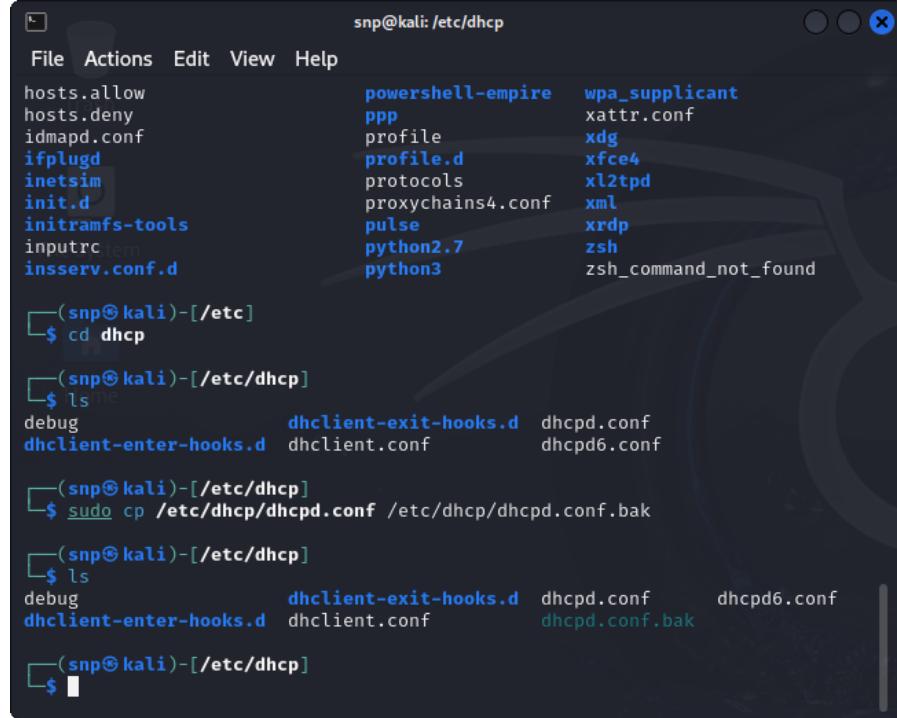
Installing and Configuring a DHCP Server

1. Install the DHCP server package.



A screenshot of a terminal window titled "File Actions Edit View Help". The window shows a dark background with a faint watermark of a person's face. The terminal prompt is "snp@kali: ~". Below the prompt, the command "\$ sudo apt-get install isc-dhcp-server" is visible, with the cursor at the end of the command line.

2. Create a backup of DHCP configuration file



```
snp@kali: /etc/dhcp
File Actions Edit View Help
hosts.allow           powershell-empire   wpa_supplicant
hosts.deny            ppp                  xattr.conf
idmapd.conf          profile               xdg
ifplugd              profile.d             xfce4
inetsim              protocols             xl2tpd
init.d               proxychains4.conf  xml
initramfs-tools       pulse                xrpd
inputrc              python2.7            zsh
inserv.conf.d        python3              zsh_command_not_found

└─(snp㉿kali)-[~/etc]
$ cd dhcp

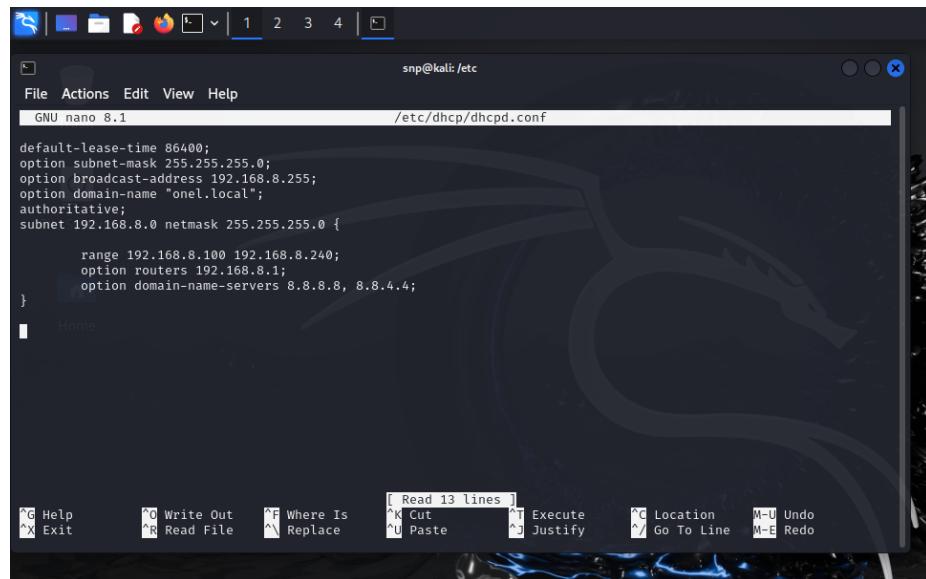
└─(snp㉿kali)-[~/etc/dhcp]
$ ls
debug                 dhclient-exit-hooks.d  dhcpd.conf
dhclient-enter-hooks.d dhclient.conf         dhcpd6.conf

└─(snp㉿kali)-[~/etc/dhcp]
$ sudo cp /etc/dhcp/dhcpd.conf /etc/dhcp/dhcpd.conf.bak

└─(snp㉿kali)-[~/etc/dhcp]
$ ls
debug                 dhclient-exit-hooks.d  dhcpd.conf      dhcpd6.conf
dhclient-enter-hooks.d dhclient.conf         dhcpd.conf.bak

└─(snp㉿kali)-[~/etc/dhcp]
$
```

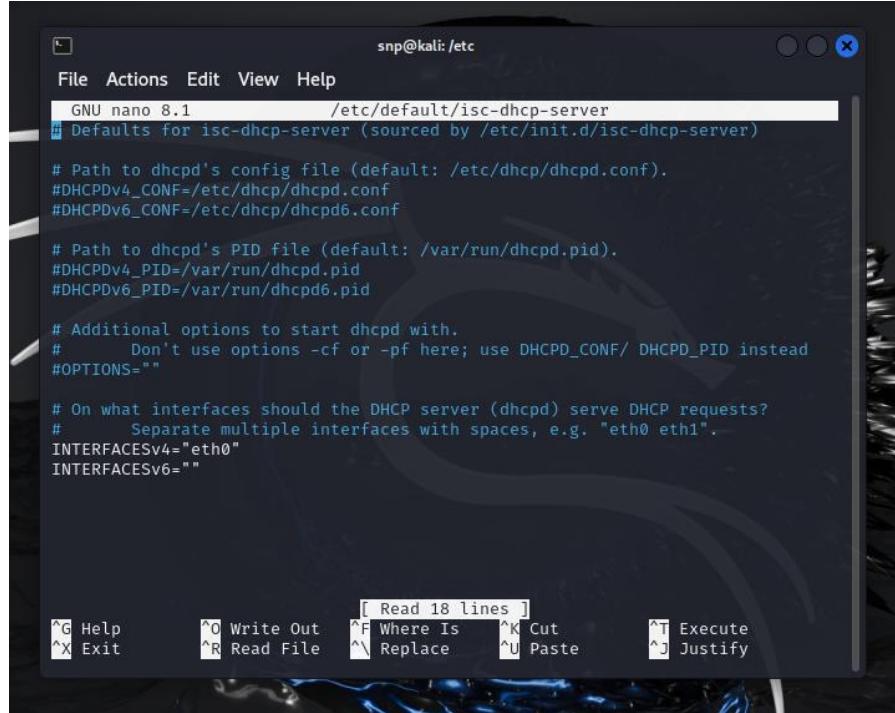
3. Open the DHCP configuration file and modify the configuration according to your requirement.



```
snp@kali: /etc
File Actions Edit View Help
GNU nano 8.1           /etc/dhcp/dhcpd.conf

default-lease-time 86400;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.8.255;
option domain-name "onel.local";
authoritative;
subnet 192.168.8.0 netmask 255.255.255.0 {
    range 192.168.8.100 192.168.8.240;
    option routers 192.168.8.1;
    option domain-name-servers 8.8.8.8, 8.8.4.4;
}
```

4. Assign network Interfaces for the DHCP server.



The screenshot shows a terminal window titled "snp@kali: /etc". The file being edited is "/etc/default/isc-dhcp-server". The content of the file is as follows:

```
GNU nano 8.1          /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpcd's config file (default: /etc/dhcp/dhcpcd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpcd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpcd6.conf

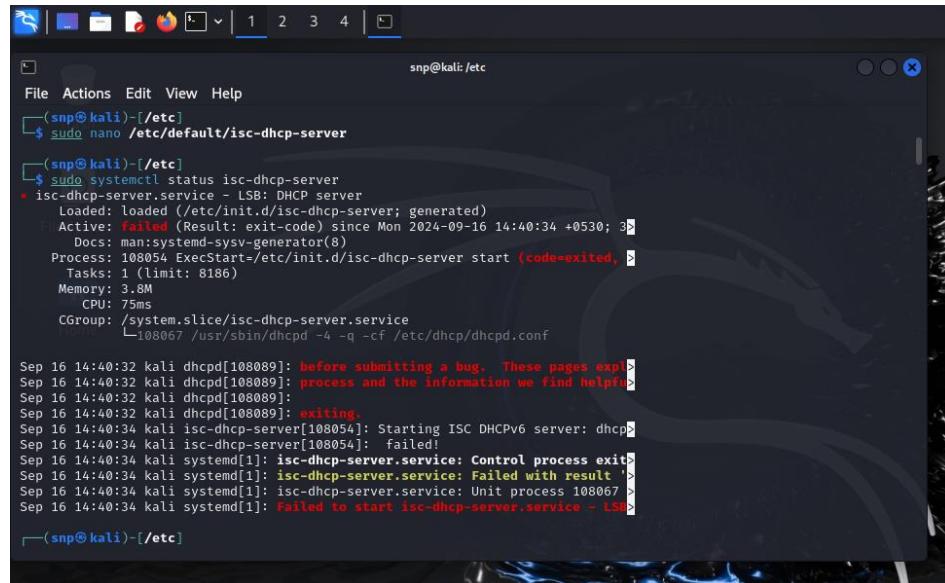
# Path to dhcpcd's PID file (default: /var/run/dhcpcd.pid).
#DHCPDv4_PID=/var/run/dhcpcd.pid
#DHCPDv6_PID=/var/run/dhcpcd6.pid

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

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

At the bottom of the terminal window, there are various keyboard shortcuts: ^G Help, ^O Write Out, ^F Where Is, ^K Cut, ^T Execute, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, and ^J Justify.

5. Check status of DHCP server.



The screenshot shows a terminal window titled "(snp@kali)-[/etc]". The command run is "sudo systemctl status isc-dhcp-server". The output is as follows:

```
[sn0@snp kali] ~
[sudo] password for sn0:
(snp@kali)-[/etc]
$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: failed (Result: exit-code) since Mon 2024-09-16 14:40:34 +0530; 3s
    Docs: man:systemd-sysv-generator(8)
  Process: 108054 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, ex
  Tasks: 1 (limit: 8186)
  Memory: 3.8M
  CPU: 75ms
  CGroup: /system.slice/isc-dhcp-server.service
           └─108067 /usr/sbin/dhcpcd -4 -q -cf /etc/dhcp/dhcpcd.conf

Sep 16 14:40:32 kali dhcpcd[108089]: before submitting a bug. These pages exp>
Sep 16 14:40:32 kali dhcpcd[108089]: process and the information we find helpfu>
Sep 16 14:40:32 kali dhcpcd[108089]:
Sep 16 14:40:32 kali dhcpcd[108089]: exiting.
Sep 16 14:40:32 kali dhcpcd[108089]: Starting ISC DHCPV6 server: dhcp>
Sep 16 14:40:32 kali isc-dhcp-server[108054]: failed!
Sep 16 14:40:32 kali systemd[1]: isc-dhcp-server.service: Control process exit>
Sep 16 14:40:34 kali systemd[1]: isc-dhcp-server.service: Failed with result '>>
Sep 16 14:40:34 kali systemd[1]: isc-dhcp-server.service: Unit process 108067 >
Sep 16 14:40:34 kali systemd[1]: Failed to start isc-dhcp-server.service - LSB>
```

6. Restart and Enable the DHCP Server.

```
snp@kali: /etc
Sep 16 14:40:34 kali systemd[1]: isc-dhcp-server.service: Unit process 108067
Sep 16 14:40:34 kali systemd[1]: Failed to start isc-dhcp-server.service - LSB
(snp@kali)-[/etc]
$ sudo systemctl start isc-dhcp-server
(snp@kali)-[/etc]
$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: active (running) since Mon 2024-09-16 14:41:25 +0530; 6s ago
    Docs: man:systemd-sysv-generator(8)
   Process: 108584 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, z
     Tasks: 1 (limit: 8186)
    Memory: 3.8M
      CPU: 48ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─108067 /usr/sbin/dhcpd -4 -q -cf /etc/dhcp/dhcpd.conf

Sep 16 14:41:23 kali systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP
Sep 16 14:41:23 kali isc-dhcp-server[108584]: Launching IPv4 server only.
Sep 16 14:41:23 kali isc-dhcp-server[108584]: Starting ISC DHCPV4 server: dhcp
Sep 16 14:41:25 kali isc-dhcp-server[108584]: .
Sep 16 14:41:25 kali systemd[1]: Started isc-dhcp-server.service - LSB: DHCP s
(snp@kali)-[/etc]
$ sudo nano /etc/dhcp/dhcpd.conf
```

7. Configure Firewall

```
snp@kali: /etc
Processing triggers for kali-menu (2024.3.1) ...
Processing triggers for man-db (2.11.2-3) ...
(snp@kali)-[/etc]
$ sudo ufw status
Status: inactive

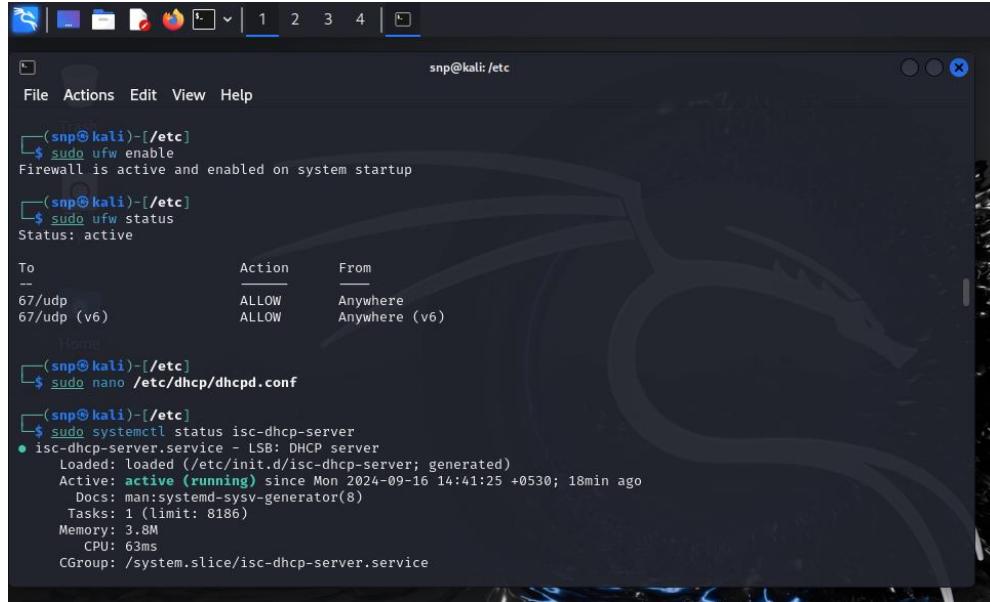
(snp@kali)-[/etc]
$ sudo ufw allow 67/udp
Rules updated
Rules updated (v6)

(snp@kali)-[/etc]
$ sudo ufw status
Status: inactive

(snp@kali)-[/etc]
$ sudo ufw enable
Firewall is active and enabled on system startup

(snp@kali)-[/etc]
$ sudo ufw status
Status: active

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



```

snp@kali: /etc
File Actions Edit View Help
(snp@kali)-[/etc]
└─$ sudo ufw enable
Firewall is active and enabled on system startup
(snp@kali)-[/etc]
└─$ sudo ufw status
Status: active

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

Home
(snp@kali)-[/etc]
└─$ sudo nano /etc/dhcpd.conf

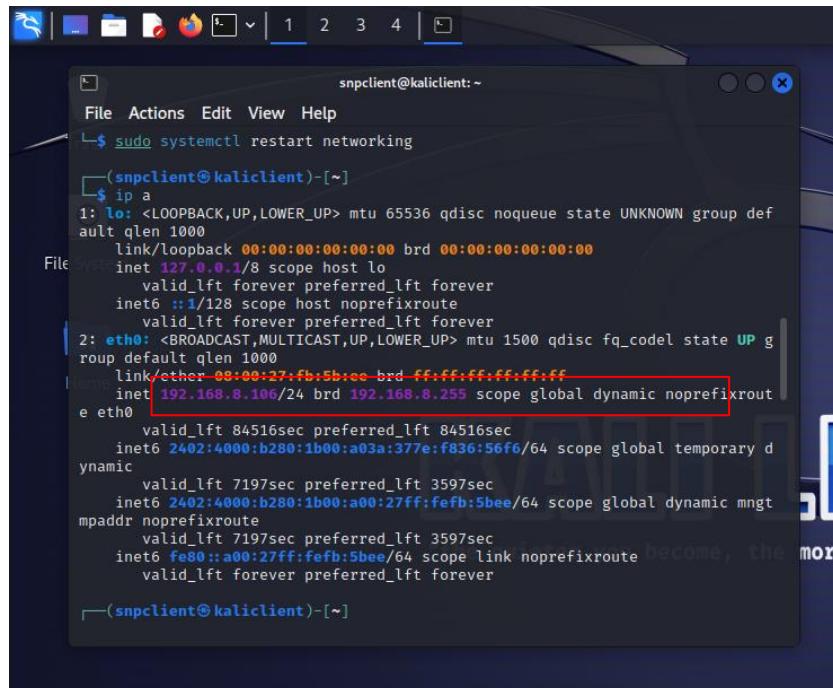
(snp@kali)-[/etc]
└─$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - LSB: DHCP server
   Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
   Active: active (running) since Mon 2024-09-16 14:41:25 +0530; 18min ago
     Docs: man:systemd-sysv-generator(8)
   Tasks: 1 (limit: 8186)
   Memory: 3.8M
     CPU: 63ms
   CGroup: /system.slice/isc-dhcp-server.service

```

Testing DHCP in virtual environment

A separate virtual machine is created in Oracle VirtualBox specifically for the client system. This VM was configured to use DHCP for network settings, ensuring it could request an IP address from the DHCP server running on another VM. Both the client and server VMs were connected to the same virtual network within VirtualBox to allow communication between them.

1. Log to the client machine and check network configuration.



```

snpcclient@kaliclient: ~
File Actions Edit View Help
└─$ sudo systemctl restart networking
(snpcclient@kaliclient)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 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 00:00:27:fb:5b:ee brd ff:ff:ff:ff:ff:ff
    inet 192.168.8.106/24 brd 192.168.8.255 scope global dynamic noprefixroute
        valid_lft 84516sec preferred_lft 84516sec
        inet6 2402:4000:b280:1b00:a03a:377e:f836:56f6/64 scope global temporary dynamic
            valid_lft 7197sec preferred_lft 3597sec
            inet6 2402:4000:b280:1b00:a00:27ff:feff:5bee/64 scope global dynamic mngtmpaddr noprefixroute
                valid_lft 7197sec preferred_lft 3597sec
                inet6 fe80::a00:27ff:feff:5bee/64 scope link noprefixroute
                    valid_lft forever preferred_lft forever
(snpcclient@kaliclient)-[~]

```

2. Confirm DHCP Server logs

```
snp@kali: /etc
File Actions Edit View Help
Sep 16 15:23:00 kali systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP server ...
Sep 16 15:23:00 kali isc-dhcp-server[129750]: Launching IPv4 server only.
Sep 16 15:23:00 kali dhcpcd[129763]: Wrote 1 leases to leases file.
Sep 16 15:23:00 kali dhcpcd[129763]: Server starting service.
Sep 16 15:23:02 kali isc-dhcp-server[129750]: Starting ISC DHCPv4 server: dhcpcd.
Sep 16 15:23:02 kali systemd[1]: Started isc-dhcp-server.service - LSB: DHCP server.

(snp@kali)-[/etc]
$ sudo journalctl -u isc-dhcp-server -f
Sep 16 15:23:00 kali systemd[1]: isc-dhcp-server.service: This usually indicates unclean termination of a previous run, or service implementation deficiencies.
Sep 16 15:23:00 kali systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP server ...
Sep 16 15:23:00 kali isc-dhcp-server[129750]: Launching IPv4 server only.
Sep 16 15:23:00 kali dhcpcd[129763]: Wrote 1 leases to leases file.
Sep 16 15:23:02 kali isc-dhcp-server[129750]: Starting ISC DHCPv4 server: dhcpcd.
Sep 16 15:23:02 kali systemd[1]: Started isc-dhcp-server.service - LSB: DHCP server.
Sep 16 15:25:49 kali dhcpcd[129763]: reuse_lease: lease age 1305 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.106
Sep 16 15:25:49 kali dhcpcd[129763]: DHCPREQUEST for 192.168.8.106 from 08:00:27:fb:5b:ee (kaliclient) via eth0
Sep 16 15:25:49 kali dhcpcd[129763]: DHCPACK on 192.168.8.106 to 08:00:27:fb:5b:ee (kaliclient) via eth0
^C

(snp@kali)-[/etc]
$ sudo nano /etc/dhcp/dhcpd.conf

(snp@kali)-[/etc]
$ sudo nano /etc/default/isc-dhcp-server
```

3. Confirm DHCP Lease

```
snp@kali: /etc
File Actions Edit View Help
Sep 16 15:23:00 kali systemd[1]: Starting isc-dhcp-server.service - LSB: DHCP server ...
Sep 16 15:23:00 kali isc-dhcp-server[129750]: Launching IPv4 server only.
Sep 16 15:23:00 kali dhcpcd[129763]: Wrote 1 leases to leases file.
Sep 16 15:23:00 kali dhcpcd[129763]: Server starting service.
Sep 16 15:23:02 kali isc-dhcp-server[129750]: Starting ISC DHCPv4 server: dhcpcd.
Sep 16 15:23:02 kali systemd[1]: Started isc-dhcp-server.service - LSB: DHCP server.
Sep 16 15:25:49 kali dhcpcd[129763]: reuse_lease: lease age 1305 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.106
Sep 16 15:25:49 kali dhcpcd[129763]: DHCPREQUEST for 192.168.8.106 from 08:00:27:fb:5b:ee (kaliclient) via eth0
Sep 16 15:25:49 kali dhcpcd[129763]: DHCPACK on 192.168.8.106 to 08:00:27:fb:5b:ee (kaliclient) via eth0
^C

(snp@kali)-[/etc]
$ sudo nano /etc/dhcp/dhcpd.conf

(snp@kali)-[/etc]
$ sudo nano /etc/default/isc-dhcp-server

(snp@kali)-[/etc]
$ dhcpcd -l
To get manufacturer names please download http://standards-oui.ieee.org/oui.txt to /usr/local/etc/oui.txt
Reading leases from /var/lib/dhcp/dhcpd.leases
MAC          IP           hostname      valid until      manufacturer
08:00:27:fb:5b:ee  192.168.8.106  kaliclient    2024-09-17 09:34:04 -NA-
```

2.2 Understanding and configuring DNS

Role of DNS

DNS translates human-readable, readable domain names into their respective IP addresses, which computers use to find each other on any network. At its very core, DNS means a way for ease of access by humans trying to access websites and services on the internet. Instead of having to remember long strings of numbers for each and every website they want to go to, users can simply type in a domain name into their web browsers and DNS will find the proper IP address that is associated with that domain name [1].

DNS Process

1. A user types a URL in their browser's address bar.
2. The browser checks its cache for IP address relevant to the domain. If not found, the request is sent to the DNS resolver.
3. The DNS resolver checks if it has the IP address cached. If IP address is cached, it is returned to the user. If not, the resolver forwards query to a Root DNS server.
4. The root dns server direct the resolver to the appropriate TLD server.
5. The resolver queries the TLD server, which knows the Authoritative Name Server for the domain given and return the address of the domain's authoritative server to the DNS.
6. Then DNS resolver contacts the Authoritative Name server for the domain given. And authoritative server responds with the IP address associated with the domain.
7. The DNS resolver caches the IP address for future requests and sends the Ip address back to the browser.
8. The browser uses the provided IP address to access the web server associated with the domain.
9. The web server responds by sending the requested content, and the browser displays the website to the user.

Configuring a DNS Server (BIND)

1. Install DNS server software package.

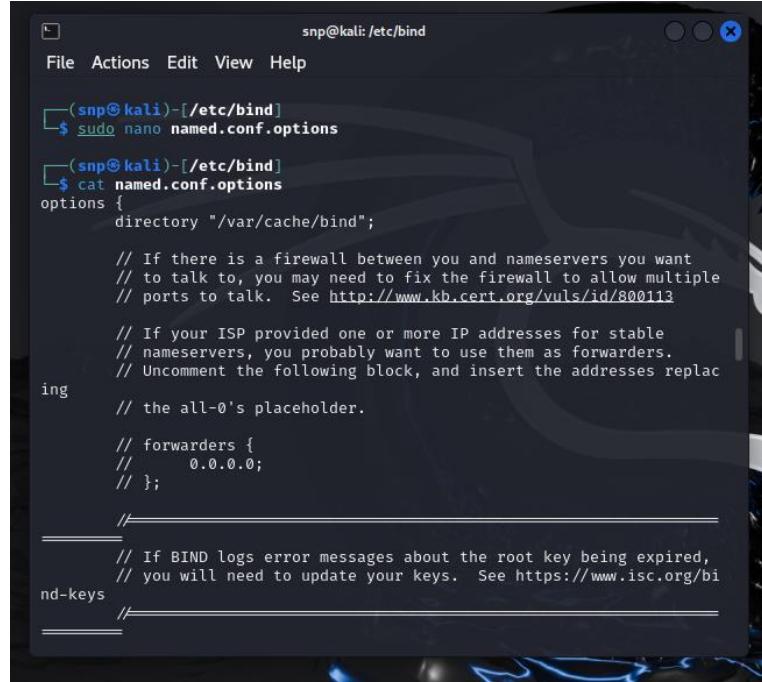
BIND (Berkeley Internet Name Domain) is the most common DNS server software.

```
snp@kali: ~
File Actions Edit View Help
└─(snp㉿kali)-[~]
$ sudo apt install bind9
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  cython3 debtags kali-debtags libabsl20220623 libaio1 libgphoto2-l10n
  liblsm-dev libpthread-stubs0-dev libtirpc-dev libucl1
  python3-backcall python3-debian python3-diskcache python3-future
  python3-jdcal python3-mistune0 python3-pendulum python3-picickleshare
  python3-pyminifier python3-pypdf2 python3-persistent python3-pytzdata
  python3-requests-toolbelt python3-rfc3986 python3-unicodecsv zenity-common
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bind9-dnsutils bind9-host bind9-libs bind9-utils libdb5.3t64
  libpam-modules libpam-modules-bin libssl3t64 liburcu8t64 libuv1-dev
  libuv1t64 openssh-client openssh-server openssh-sftp-server openssl
Suggested packages:
  bind-doc resolvconf libuv1-doc keychain libpam-ssh monkeysphere
  ssh-askpass molly-guard
The following packages will be REMOVED:
  libdb5.3 libssl3 liburcu8 libuv1
The following NEW packages will be installed:
  bind9 bind9-utils libdb5.3t64 libssl3t64 liburcu8t64 libuv1t64
The following packages will be upgraded:
  bind9-dnsutils bind9-host bind9-libs libpam-modules
  libpam-modules-bin libuv1-dev openssh-client openssh-server
  openssh-sftp-server openssl
```

2. Configure BIND

Configure named.conf.options – This file contains global server options, including forwarders.

- Forwarders - Specifies upstream DNS servers that BIND should forward queries to if it does not have the answer locally. These are usually our ISP's DNS servers or public DNS servers (Google DNS) [1].



```
snp@kali: /etc/bind
File Actions Edit View Help
└─(snp@kali)-[/etc/bind]
$ sudo nano named.conf.options
└─(snp@kali)-[/etc/bind]
$ cat named.conf.options
options {
    directory "/var/cache/bind";

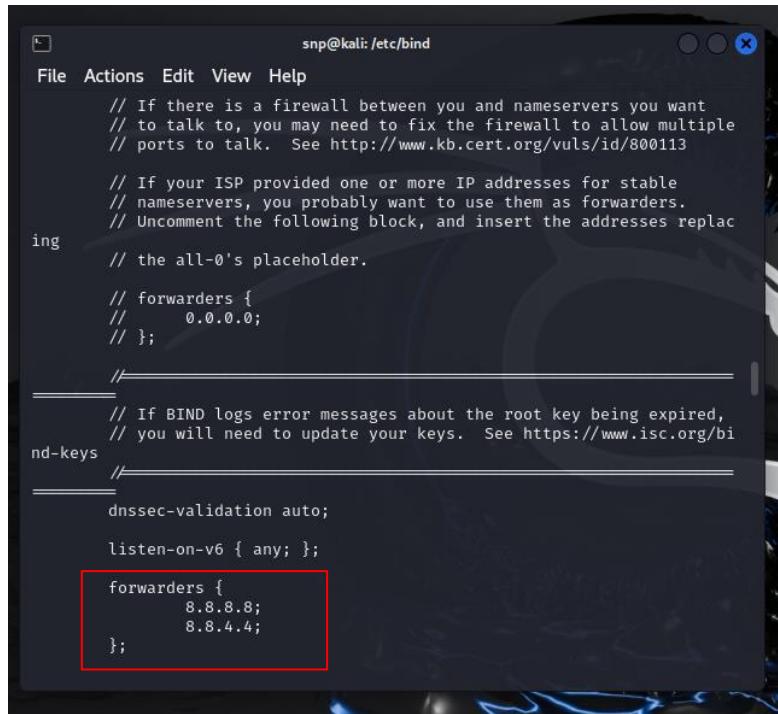
    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // 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 replacing
    // the all-0's placeholder.

    // forwarders {
    //     0.0.0.0;
    // };

    //

    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bi
nd-keys
//
```



```
// If there is a firewall between you and nameservers you want
// to talk to, you may need to fix the firewall to allow multiple
// 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 replacing
// the all-0's placeholder.

// forwarders {
//     0.0.0.0;
// };

//

// If BIND logs error messages about the root key being expired,
// you will need to update your keys. See https://www.isc.org/bi
nd-keys
//
```

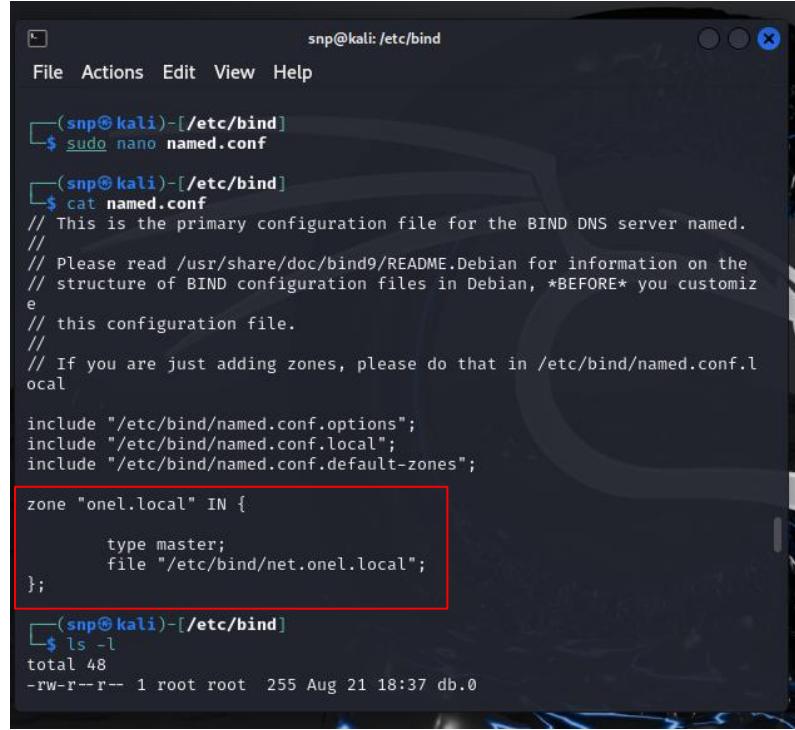
dnssec-validation auto;

listen-on-v6 { any; };

```
forwarders {
    8.8.8.8;
    8.8.4.4;
};
```

Configure named.conf

The named.conf file is the main configuration file for BIND (Berkeley Internet Name Domain), a popular DNS server software. This file defines the overall behavior of the DNS server, including the zones it manages, access control settings, logging, and various server options [1].



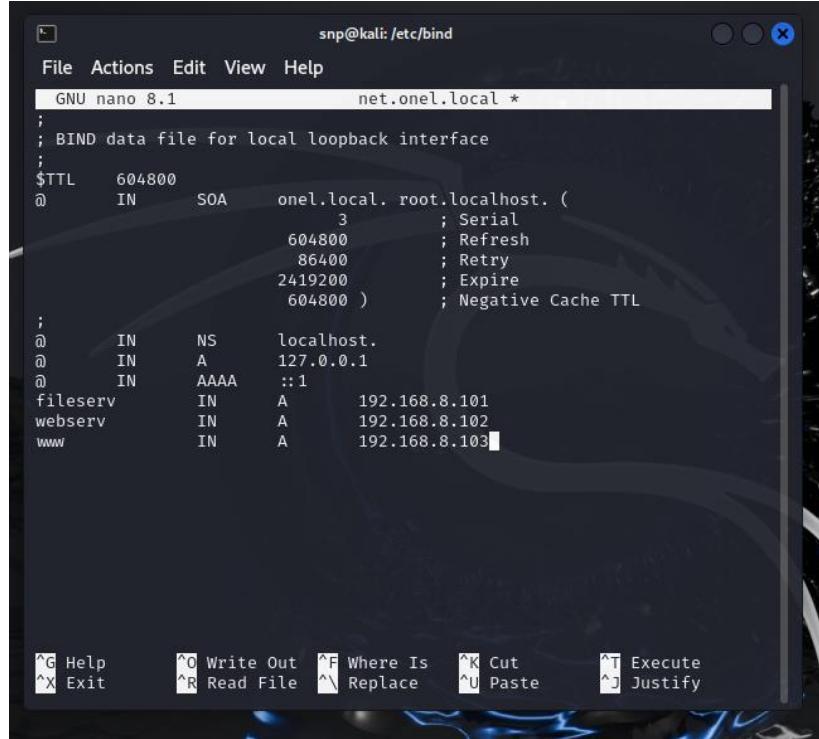
```
snp@kali: /etc/bind
File Actions Edit View Help
(snp@kali)-[/etc/bind]
$ sudo nano named.conf
(snp@kali)-[/etc/bind]
$ cat named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
zone "onel.local" IN {
    type master;
    file "/etc/bind/net.onel.local";
};

(snp@kali)-[/etc/bind]
$ ls -l
total 48
-rw-r--r-- 1 root root 255 Aug 21 18:37 db.0
```

Zone - The zone section is one of the most important parts of named.conf. This is where you define the DNS zones that the server is responsible for.

Create Zone File

Zone files are simple text files that follow a specific format. They contain **resource records** (RRs) that map domain names to various types of information (such as IP addresses, mail servers, etc.). Each zone file begins with **SOA (Start of Authority)**, which defines administrative information about the zone, followed by various resource records [1].

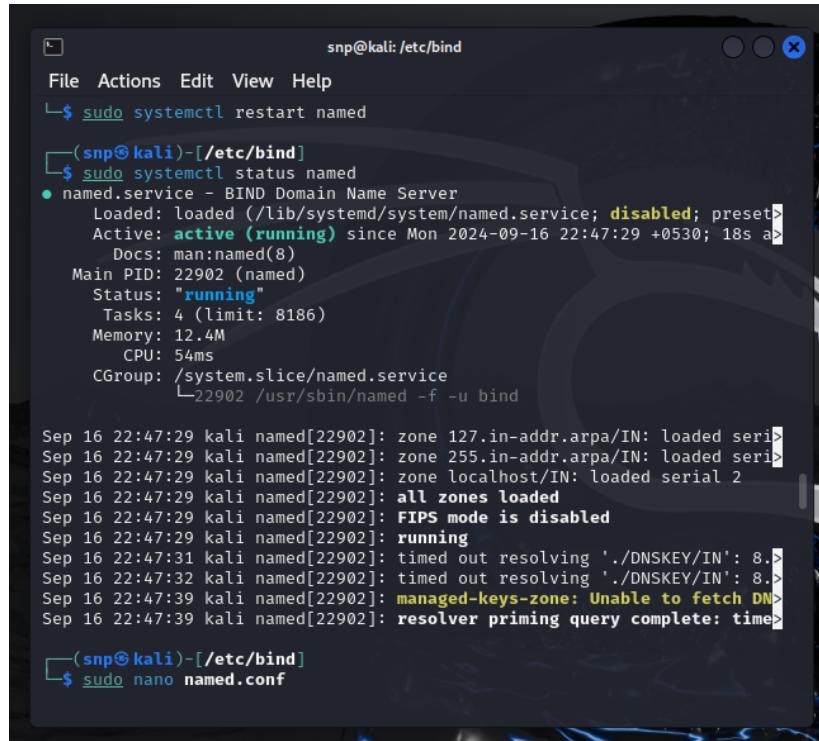


```
snp@kali: /etc/bind
GNU nano 8.1          net.onel.local *

;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     onel.local. root.localhost. (
                            3           ; Serial
                            604800      ; Refresh
                            86400       ; Retry
                           2419200     ; Expire
                           604800 )    ; Negative Cache TTL
;
@       IN      NS      localhost.
@       IN      A       127.0.0.1
@       IN      AAAA    ::1
fileserv   IN      A       192.168.8.101
webserv    IN      A       192.168.8.102
www        IN      A       192.168.8.103

^G Help      ^O Write Out  ^F Where Is  ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace   ^U Paste      ^J Justify
```

3. Enable and Restart BIND



```
snp@kali: /etc/bind
File Actions Edit View Help
└─$ sudo systemctl restart named

[sudo@kali] - [/etc/bind]
$ sudo systemctl status named
● named.service - BIND Domain Name Server
  Loaded: loaded (/lib/systemd/system/named.service; disabled; preset)
  Active: active (running) since Mon 2024-09-16 22:47:29 +0530; 18s ago
    Docs: man:named(8)
 Main PID: 22902 (named)
   Status: "running"
     Tasks: 4 (limit: 8186)
   Memory: 12.4M
     CPU: 54ms
   CGroup: /system.slice/named.service
           └─22902 /usr/sbin/named -f -u bind

Sep 16 22:47:29 kali named[22902]: zone 127.in-addr.arpa/IN: loaded serial 2
Sep 16 22:47:29 kali named[22902]: zone 255.in-addr.arpa/IN: loaded serial 2
Sep 16 22:47:29 kali named[22902]: zone localhost/IN: loaded serial 2
Sep 16 22:47:29 kali named[22902]: all zones loaded
Sep 16 22:47:29 kali named[22902]: FIPS mode is disabled
Sep 16 22:47:29 kali named[22902]: running
Sep 16 22:47:31 kali named[22902]: timed out resolving './DNSKEY/IN': 8.>
Sep 16 22:47:32 kali named[22902]: timed out resolving './DNSKEY/IN': 8.>
Sep 16 22:47:39 kali named[22902]: managed-keys-zone: Unable to fetch DN>
Sep 16 22:47:39 kali named[22902]: resolver priming query complete: time>

[sudo@kali] - [/etc/bind]
$ sudo nano named.conf
```

4. Test the DNS Server

Use dig or nslookup to test if the DNS server is responding correctly.

```
snp@kali:/etc/bind
File Actions Edit View Help
Sep 16 22:57:47 kali named[28031]: FIPS mode is disabled
Sep 16 22:57:47 kali systemd[1]: Started named.service - BIND Domain Name Server
Sep 16 22:57:47 kali named[28031]: running
Sep 16 22:57:47 kali named[28031]: zone onel.local/IN: sending notifies (to 192.168.8.104#53)
Sep 16 22:57:47 kali named[28031]: managed-keys-zone: Initializing automap zones
Sep 16 22:57:57 kali named[28031]: resolver priming query complete: time>
(snp@kali)-[/etc/bind]
$ nslookup
> server 192.168.8.104
Default server: 192.168.8.104
Address: 192.168.8.104#53
> fileserv.onel.local
Server: 192.168.8.104
Address: 192.168.8.104#53

Name: fileserv.onel.local
Address: 192.168.8.101
> webserv.onel.local
Server: 192.168.8.104
Address: 192.168.8.102

Name: webserv.onel.local
Address: 192.168.8.102
>

(snp@kali)-[/etc/bind]
$ dig @192.168.8.104 fileserv.onel.local
; <>> DiG 9.20.1-1-Debian <>> @192.168.8.104 fileserv.onel.local
```

```
snp@kali:/etc/bind
File Actions Edit View Help
$ dig @192.168.8.104 fileserv.onel.local
; <>> DiG 9.20.1-1-Debian <>> @192.168.8.104 fileserv.onel.local
; (1 server found)
; global options: +cmd
; Got answer:
; WARNING: .local is reserved for Multicast DNS
; You are currently testing what happens when an mDNS query is leaked to DNS
; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 55225
; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 67b67925e8247f450100000066e86b4cd5acbe49d5b8f602 (good)
; QUESTION SECTION:
;fileserv.onel.local. IN A
; ANSWER SECTION:
fileserv.onel.local. 604800 IN A 192.168.8.101
; Query time: 0 msec
; SERVER: 192.168.8.104#53(192.168.8.104) (UDP)
; WHEN: Mon Sep 16 23:00:52 +0530 2024
; MSG SIZE rcvd: 92

(snp@kali)-[/etc/bind]
$ sudo systemctl status named
● named.service - BIND Domain Name Server
```

2.3 Understanding and Configuring NTP

What is NTP

Network Time Protocol NTP is a protocol that synchronizes the time of computers and network devices over a network. By using a hierarchy of time sources, NTP makes it possible for all participating systems to maintain a consistent time for logging, transaction, and system operations applications

Configuring a NTP Server

1. Install the NTP Software

Install the NTP package using your distribution's package manager

```
snp@kali: /var/log
File Actions Edit View Help
drwx----- 3 inetsim      inetsim      4096 Sep 15 18:34 inetsim
drwxr-xr-x  3 root        root        4096 Sep 15 18:47 installer
drwxr-sr-x+ 3 root        systemd-journal 4096 Sep 15 18:47 journal
-rw-rw-r--  1 root        utmp        0 Sep 15 18:02 lastlog
drwxr-xr-x  2 root        root        4096 Sep 19 19:23 lightdm
-rw-r--r--  1 root        root        5148 Sep 19 19:23 macchanger.log
drwxr-xr-x  2 mosquitto   root        4096 Jul 21 2023 mosquitto
drwxr-xr-x  2 root        adm         4096 Sep 15 18:27 nginx
drwxr-xr-x  2 _gvm       _gvm        4096 Apr 21 2023 notus-scanner
drwxr-xr-x  2 root        root        4096 May 20 2023 openvpn
drwxrwxr-t  2 root        postgres    4096 Sep 15 18:33 postgresql
drwx----- 2 root        root        4096 Sep 15 18:02 private
drwxr-s---  2 redis       adm         4096 Sep 15 18:28 redis
drwxr-xr-x  3 root        root        4096 Sep 15 18:09 runit
drwxr-x---  2 root        adm         4096 Aug  4 2023 samba
drwx----- 2 speech-dispatcher root      4096 Aug  6 2023 speech-dispatcher
drwxr-xr-x  2 stunnel4   stunnel4    4096 Sep 15 18:31 stunnel4
drwxr-xr-x  2 root        root        4096 Dec  5 2022 sysstat
-rw-rw-r--  1 root        utmp      29184 Sep 19 19:23 wtmp

(snp@kali)-[~/var/log]
$ cd named
cd: no such file or directory: named

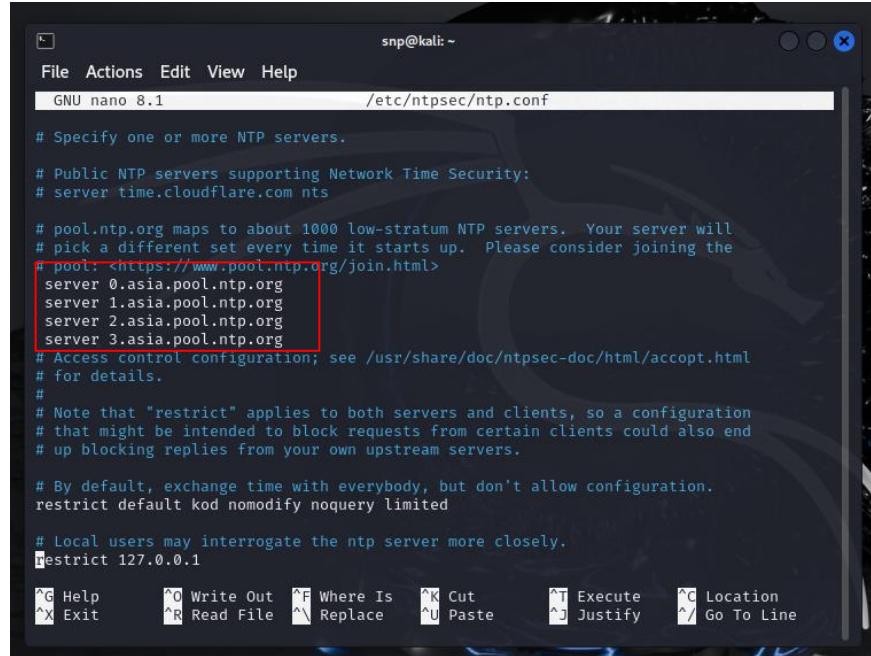
(snp@kali)-[~/var/log]
$ sudo nano /etc/bind/named.conf

(snp@kali)-[~/var/log]
$ sudo apt install ntp
```

2. Configure NTP Server

The primary configuration file for the NTP server is /etc/ntp.conf .

The default configuration usually contains public NTP servers. We can modify these by replacing the server entries with our preferred NTP servers.



```
snp@kali: ~
File Actions Edit View Help
GNU nano 8.1          /etc/ntpsec/ntp.conf

# Specify one or more NTP servers.

# Public NTP servers supporting Network Time Security:
# server time.cloudflare.com nts

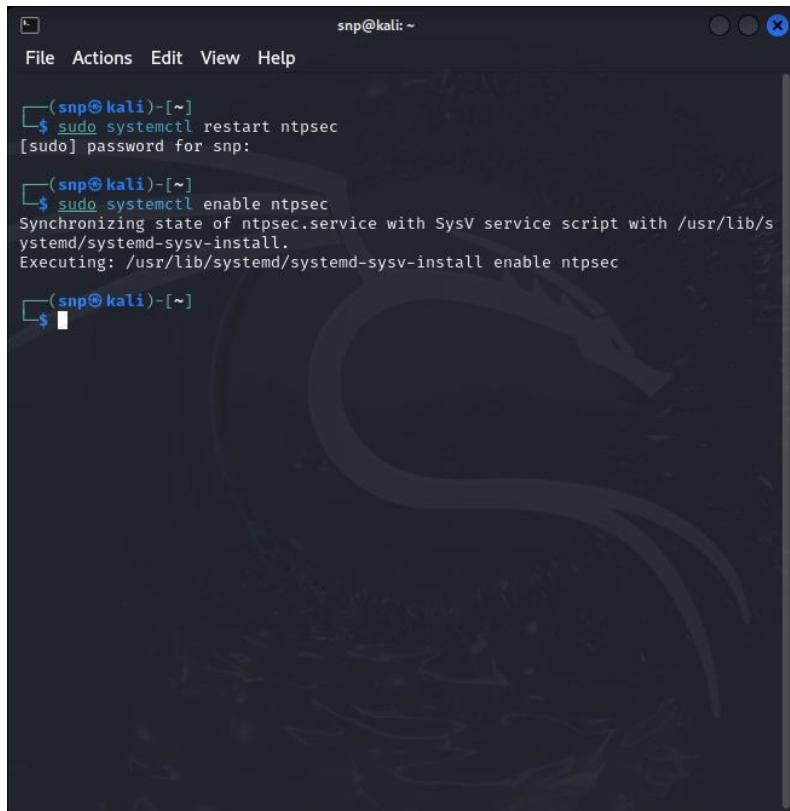
# pool.ntp.org maps to about 1000 low-stratum NTP servers. Your server will
# pick a different set every time it starts up. Please consider joining the
# pool: <https://www.pool.ntp.org/join.html>
server 0.asia.pool.ntp.org
server 1.asia.pool.ntp.org
server 2.asia.pool.ntp.org
server 3.asia.pool.ntp.org
# Access control configuration; see /usr/share/doc/ntpsec-doc/html/accept.html
# for details.
#
# Note that "restrict" applies to both servers and clients, so a configuration
# that might be intended to block requests from certain clients could also end
# up blocking replies from your own upstream servers.

# By default, exchange time with everybody, but don't allow configuration.
restrict default kod nomodify noquery limited

# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1

^G Help      ^O Write Out  ^F Where Is  ^K Cut        ^T Execute  ^C Location
^X Exit      ^R Read File  ^W Replace   ^U Paste     ^J Justify  ^/ Go To Line
```

3. Restart and Enable the NTP Service.



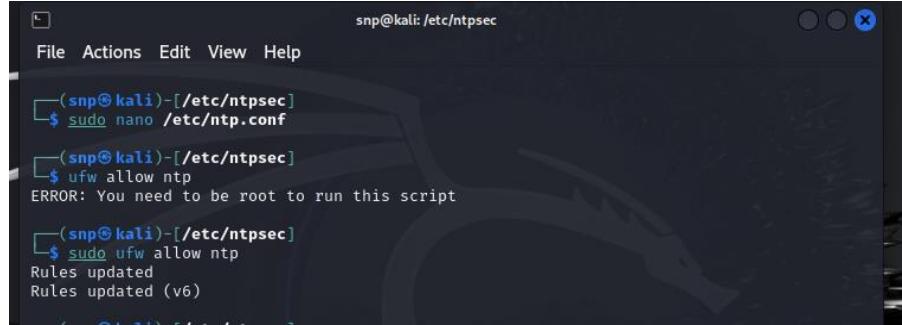
```
(snp@kali)-[~]
$ sudo systemctl restart ntpsec
[sudo] password for.snp:

(snp@kali)-[~]
$ sudo systemctl enable ntpsec
Synchronizing state of ntpsec.service with SysV service script with /usr/lib/
systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ntpsec

(snp@kali)-[~]
$
```

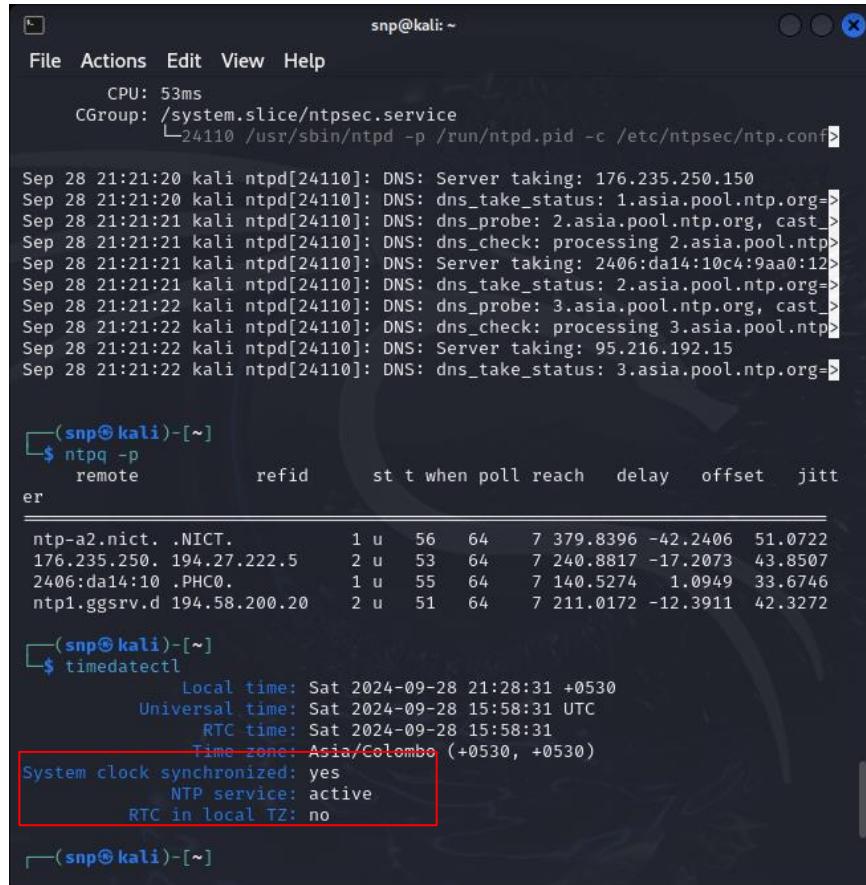
4. Open NTP port in firewall.

NTP uses port 123 (UDP) for communication. Make sure this port is open in firewall.



```
snp@kali: /etc/ntpsec
File Actions Edit View Help
(snp@kali)-[/etc/ntpsec]
$ sudo nano /etc/ntp.conf
(snp@kali)-[/etc/ntpsec]
$ ufw allow ntp
ERROR: You need to be root to run this script
(snp@kali)-[/etc/ntpsec]
$ sudo ufw allow ntp
Rules updated
Rules updated (v6)
```

5. Verify NTP Server Status



```
snp@kali: ~
File Actions Edit View Help
CPU: 53ms
CGroup: /system.slice/ntpsec.service
└─24110 /usr/sbin/ntpd -p /run/ntpd.pid -c /etc/ntpsec/ntp.conf

Sep 28 21:21:20 kali ntpd[24110]: DNS: Server taking: 176.235.250.150
Sep 28 21:21:20 kali ntpd[24110]: DNS: dns_take_status: 1.asia.pool.ntp.org=>
Sep 28 21:21:21 kali ntpd[24110]: DNS: dns_probe: 2.asia.pool.ntp.org, cast_>
Sep 28 21:21:21 kali ntpd[24110]: DNS: dns_check: processing 2.asia.pool.ntp>
Sep 28 21:21:21 kali ntpd[24110]: DNS: Server taking: 2406:da14:10c4:9aa0:12>
Sep 28 21:21:21 kali ntpd[24110]: DNS: dns_take_status: 2.asia.pool.ntp.org=>
Sep 28 21:21:22 kali ntpd[24110]: DNS: dns_probe: 3.asia.pool.ntp.org, cast_>
Sep 28 21:21:22 kali ntpd[24110]: DNS: dns_check: processing 3.asia.pool.ntp>
Sep 28 21:21:22 kali ntpd[24110]: DNS: Server taking: 95.216.192.15
Sep 28 21:21:22 kali ntpd[24110]: DNS: dns_take_status: 3.asia.pool.ntp.org=>

(snp@kali)-[~]
$ ntpq -p
      remote          refid      st t when poll reach   delay    offset  jitter
er
+ntp-a2.nict. .NICT.        1 u    56   64    7 379.8396 -42.2406  51.0722
176.235.250. 194.27.222.5  2 u    53   64    7 240.8817 -17.2073  43.8507
2406:da14:10 .PHC0.        1 u    55   64    7 140.5274  1.0949  33.6746
ntp1.ggsrv.d 194.58.200.20  2 u    51   64    7 211.0172 -12.3911  42.3272

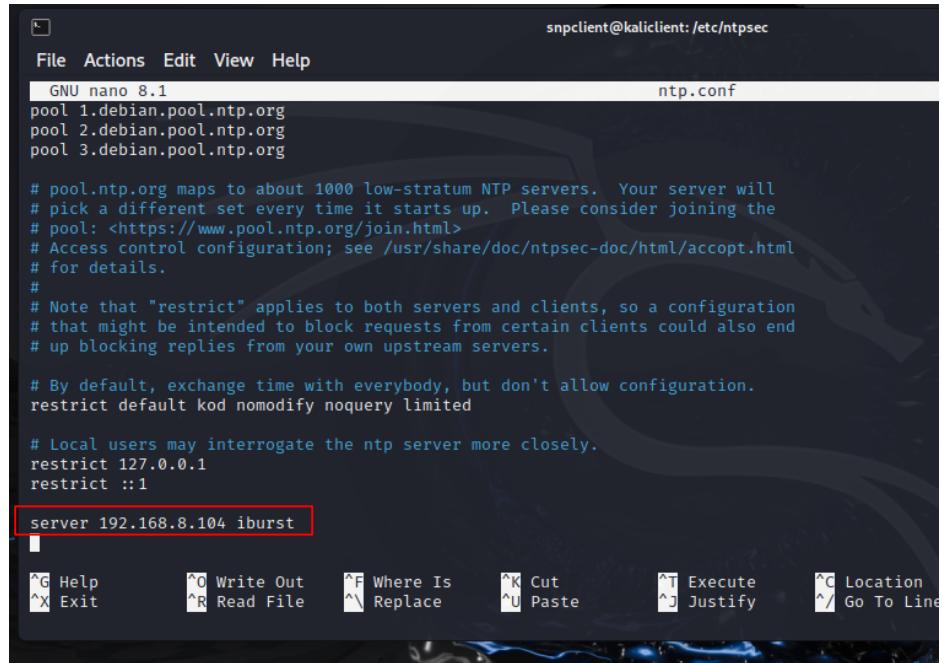
(snp@kali)-[~]
$ timedatectl
      Local time: Sat 2024-09-28 21:28:31 +0530
      Universal time: Sat 2024-09-28 15:58:31 UTC
            RTC time: Sat 2024-09-28 15:58:31
           Time zone: Asia/Colombo (+0530, +0530)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

Configure NTP Clients

A separate virtual machine is created in Oracle VirtualBox specifically for the client system. This VM was configured to use NTP Server running on another VM. Both the client and server VMs were connected to the same virtual network within VirtualBox to allow communication between them.

1. Configure /etc/ntpsec/ntp.conf

Edit and specify your NTP server running on host machine.



```
snpclient@kaliclient: /etc/ntpsec
File Actions Edit View Help
GNU nano 8.1          ntp.conf
pool 1.debian.pool.ntp.org
pool 2.debian.pool.ntp.org
pool 3.debian.pool.ntp.org

# pool.ntp.org maps to about 1000 low-stratum NTP servers. Your server will
# pick a different set every time it starts up. Please consider joining the
# pool: <https://www.pool.ntp.org/join.html>
# Access control configuration; see /usr/share/doc/ntpsec-doc/html/accept.html
# for details.
#
# Note that "restrict" applies to both servers and clients, so a configuration
# that might be intended to block requests from certain clients could also end
# up blocking replies from your own upstream servers.

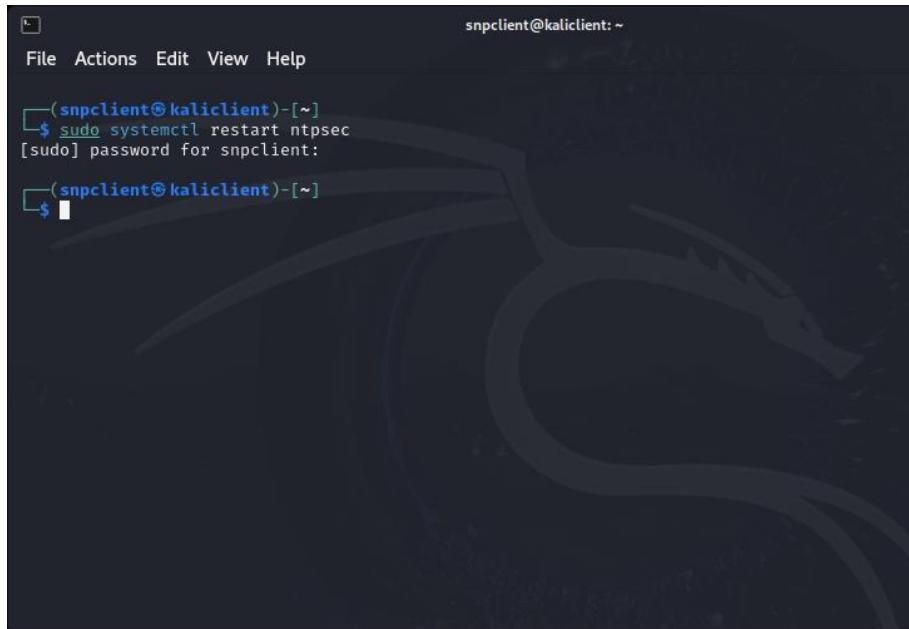
# By default, exchange time with everybody, but don't allow configuration.
restrict default kod nomodify noquery limited

# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1

server 192.168.8.104 iburst

^G Help      ^O Write Out   ^F Where Is   ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File    ^V Replace    ^U Paste     ^J Justify   ^/ Go To Line
```

2. Restart NTP on client machine



```
(snpclient@kaliclient)-[~]
$ sudo systemctl restart ntpsec
[sudo] password for snpclient:

(snpclient@kaliclient)-[~]
$
```

3. Verify Client Synchronization

IT23440418

The screenshot shows a terminal window titled 'snpclient@kaliclient:~'. It displays the output of two commands: 'ntpq -p' and 'timedatectl status'. The 'ntpq -p' command shows a list of NTP servers with their refid, stratum (st), type (t), when, poll, reach, delay, offset, and jitter values. The 'timedatectl status' command shows the system's time synchronization status, including local, universal, and RTC times, as well as the time zone and clock synchronization status.

```
snpclient@kaliclient:~ [sudo] password for snpclient:
[snpclient@kaliclient:~]
$ ntpq -p
      remote                  refid      st t when poll reach   delay    offset  jitter
+192.168.8.104          .POOL.        16 p    - 256    0  0.0000  0.0000  0.0001
0.debian.pool.ntp.org    .POOL.        16 p    - 256    0  0.0000  0.0000  0.0001
1.debian.pool.ntp.org    .POOL.        16 p    - 256    0  0.0000  0.0000  0.0001
2.debian.pool.ntp.org    .POOL.        16 p    - 256    0  0.0000  0.0000  0.0001
3.debian.pool.ntp.org    .POOL.        16 p    - 256    0  0.0000  0.0000  0.0001
+192.168.8.104          225.88.224.236 2 u     5 64    3  0.7711 -191.362  2.6253
time.cloudflare.com      10.4.8.56   3 u     9 64    3  56.5117 -219.088  8.7451
time.cloudflare.com      10.4.8.56   3 u    11 64    3  30.5418 -226.459  1.4184
time.cloudflare.com      10.4.8.56   3 u     8 64    3  30.7359 -226.721  2.6657
time.cloudflare.com      10.4.8.56   3 u     9 64    3  31.0084 -226.111  7.3081

[snpclient@kaliclient:~]
$ timedatectl status
           Local time: Sat 2024-09-28 21:44:43 +0530
         Universal time: Sat 2024-09-28 16:14:43 UTC
              RTC time: Sat 2024-09-28 16:14:43
            Time zone: Asia/Colombo (+0530, +0530)
System clock synchronized: yes
          NTP service: active
RTC in local TZ: no

[snpclient@kaliclient:~]
```

3 . Shell Scripting and Security

3.1 Shell Scripting

Automating a System Report

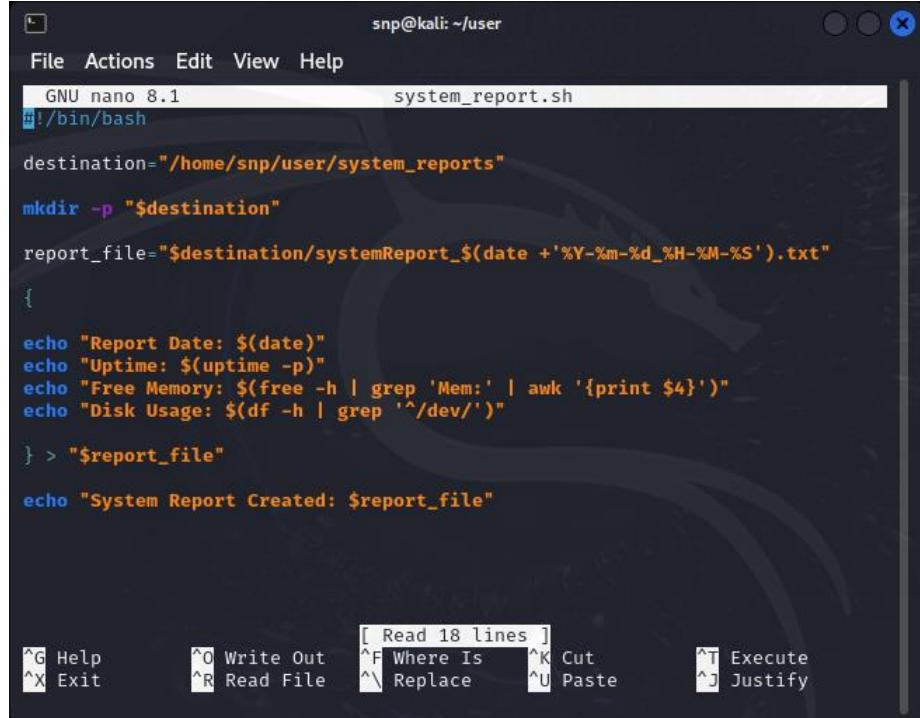
The script automates the process of generating a system report that includes:

- The current date and time.
- The system's uptime.
- The amount of free memory.
- The disk usage for mounted file systems.

It saves this information in a text file with a filename that includes the current date, ensuring that each report is unique and easily identifiable.

```
#!/bin/bash

destination="/home/snp/user/system_reports"
mkdir -p "$destination"
report_file="$destination/systemReport_$(date +'%Y-%m-%d_%H-%M-%S').txt"
{
echo "Report Date: $(date)"
echo "Uptime: $(uptime -p)"
echo "Free Memory: $(free -h | grep 'Mem:' | awk '{print $4}')"
echo "Disk Usage: $(df -h | grep '^/dev/')"
} > "$report_file"
echo "System Report Created: $report_file"
```



```

snp@kali: ~/user
File Actions Edit View Help
GNU nano 8.1           system_report.sh
#!/bin/bash

destination="/home/snp/user/system_reports"
mkdir -p "$destination"
report_file="$destination/systemReport_$(date +'%Y-%m-%d_%H-%M-%S').txt"
{
echo "Report Date: $(date)"
echo "Uptime: $(uptime -p)"
echo "Free Memory: $(free -h | grep 'Mem:' | awk '{print $4}')"
echo "Disk Usage: $(df -h | grep '^/dev/')"
} > "$report_file"
echo "System Report Created: $report_file"

[ Read 18 lines ]
^G Help      ^O Write Out   ^F Where Is   ^K Cut       ^T Execute
^X Exit      ^R Read File   ^\ Replace    ^U Paste     ^J Justify

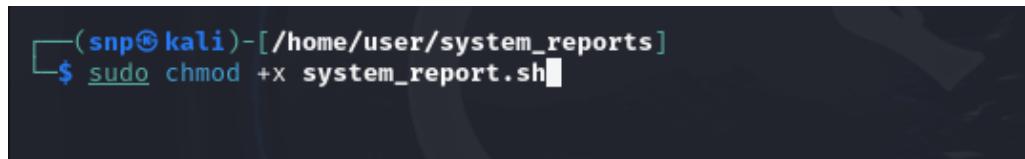
```

Automating Task with crontab

Crontab is a Unix-based utility that allows users to schedule tasks (known as "cron jobs") to run automatically at specified intervals. This is especially useful for automating repetitive tasks, such as running scripts, generating reports, or performing system maintenance.

1. First, we need to ensure it is executable by running.

```
sudo chmod +x system_report.sh
```



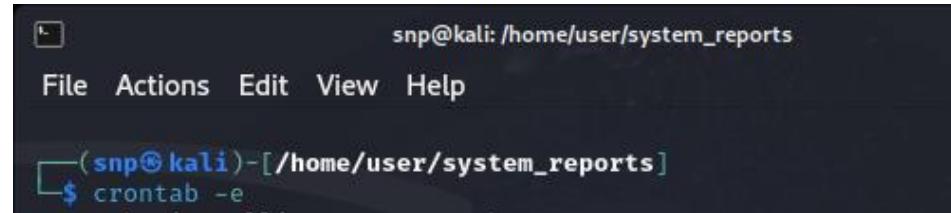
```

(snp㉿kali)-[~/user/system_reports]
$ sudo chmod +x system_report.sh

```

2. Open crontab file.

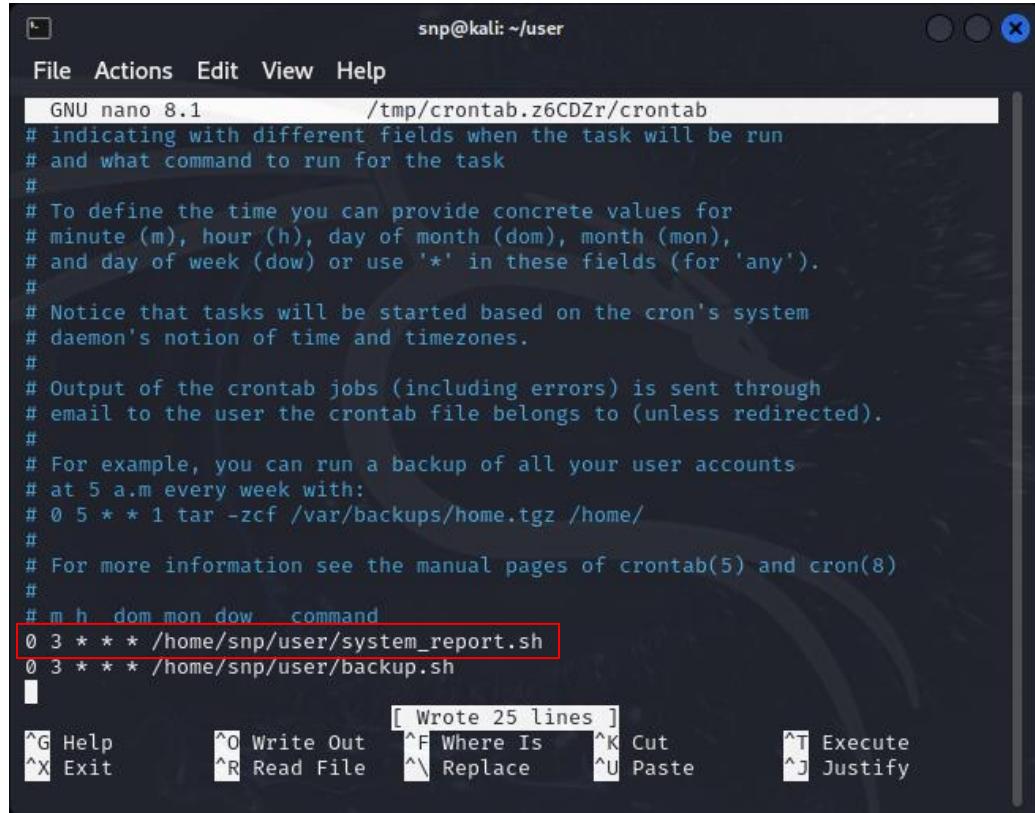
```
cron tab -e
```



```
snp@kali: /home/user/system_reports
File Actions Edit View Help
(snp㉿kali)-[~/home/user/system_reports]
$ crontab -e
```

3. Adding a cron job.

```
0 3 * * * /home/snp/user/system_reports/system_report.sh
```



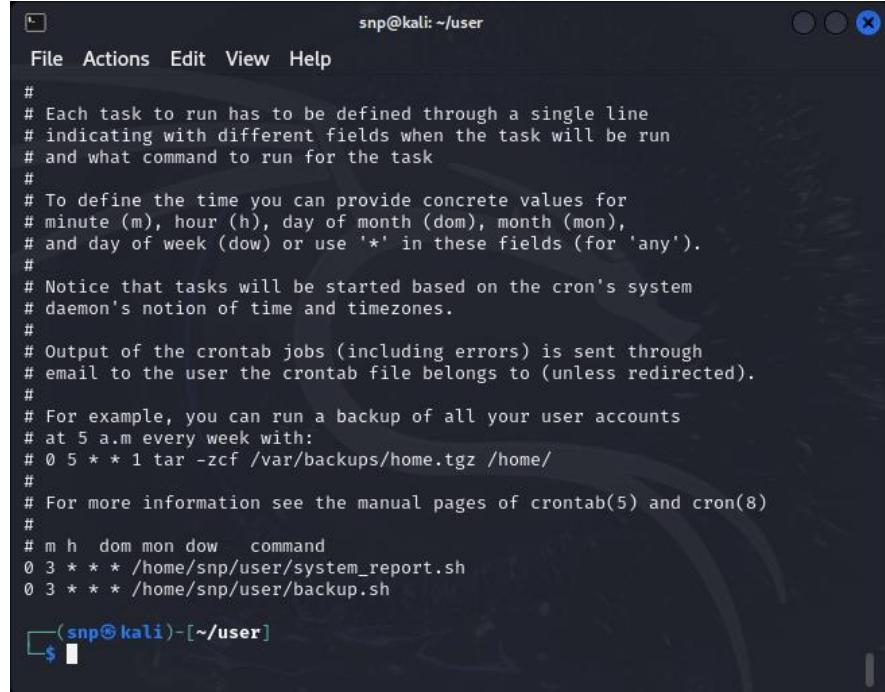
```
GNU nano 8.1      /tmp/crontab.z6CDZr/crontab
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' 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 3 * * * /home/snp/user/system_report.sh
0 3 * * * /home/snp/user/backup.sh
```

[Wrote 25 lines]

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify

4. Save and exit.

5. Check scheduled jobs.



```

snp@kali: ~/user
File Actions Edit View Help
#
# 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 3 * * * /home/snp/user/system_report.sh
0 3 * * * /home/snp/user/backup.sh

```

(snp@kali)-[~/user]
\$

Automating Backup of a critical Directory

This Bash script automates the process of backing up files from a specified source directory to a destination directory in a compressed format.

```

#!/bin/bash

source="/home/snp/user/documents"

destination="/home/snp/user/backup/documents"

mkdir -p "$destination"

backup="$destination/backupFile_$(date +'%Y-%m-%d_%H-%M-%S').tar.gz"

tar -czf "$backup" -C "$source" .

echo "Backup completed: $backup"

```

The screenshot shows a terminal window titled "File Actions Edit View Help" with the command "GNU nano 8.1" and the file name "backup.sh". The script content is as follows:

```
source="/home/snp/user/documents"
destination="/home/snp/user/backup/documents"

mkdir -p "$destination"
backup="$destination/backupFile_$(date +'%Y-%m-%d_%H-%M-%S').tar.gz"
tar -czf "$backup" -C "$source" .
echo "Backup completed: $backup"
```

At the bottom of the terminal, there is a status bar with keyboard shortcuts: [Read 14 lines], ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^F Where Is, ^\ Replace, ^K Cut, ^U Paste, ^T Execute, ^J Justify.

To automate the task, We'll setup cron job as previous task.

1. Make script executable.

```
sudo chmod +x backup.sh
```

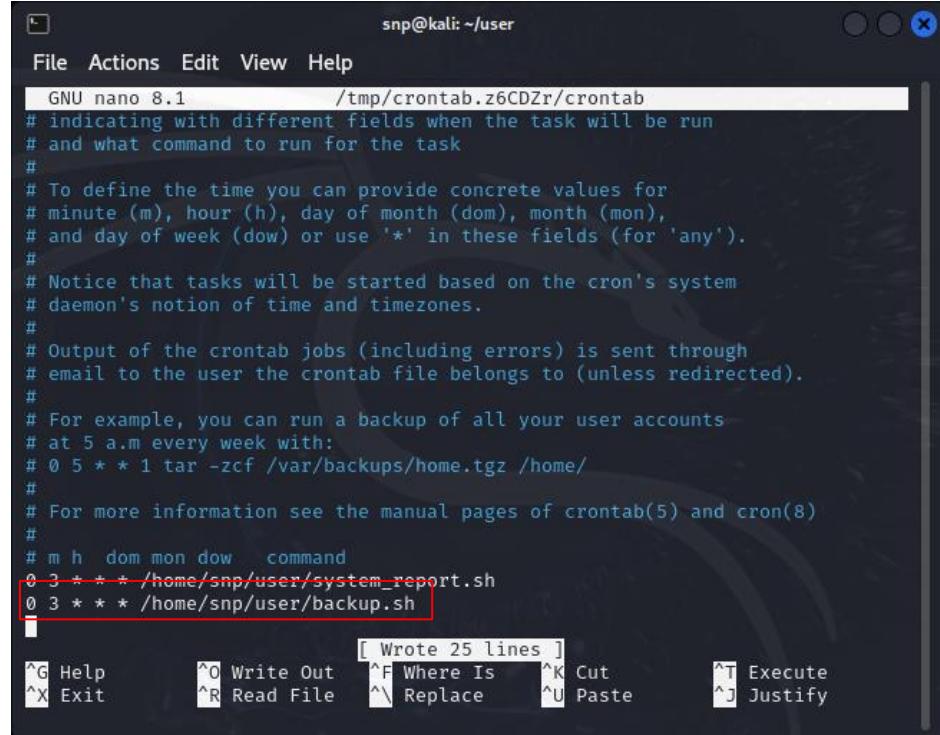
The screenshot shows a terminal window with the command "sudo chmod +x backup.sh" entered and its output displayed. The output shows the command was successful.

2. Open cron tab file

```
crontab -e
```

3. Adding a cron job.

```
0 3 * * * /home/snp/user/backup.sh
```



```
snp@kali: ~/user
File Actions Edit View Help
GNU nano 8.1      /tmp/crontab.z6CDZr/crontab
# 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 3 * * * /home/snp/user/system_report.sh
0 3 * * * /home/snp/user/backup.sh

[ Wrote 25 lines ]
^G Help      ^O Write Out    ^F Where Is    ^K Cut        ^T Execute
^X Exit      ^R Read File    ^\ Replace     ^U Paste      ^J Justify
```

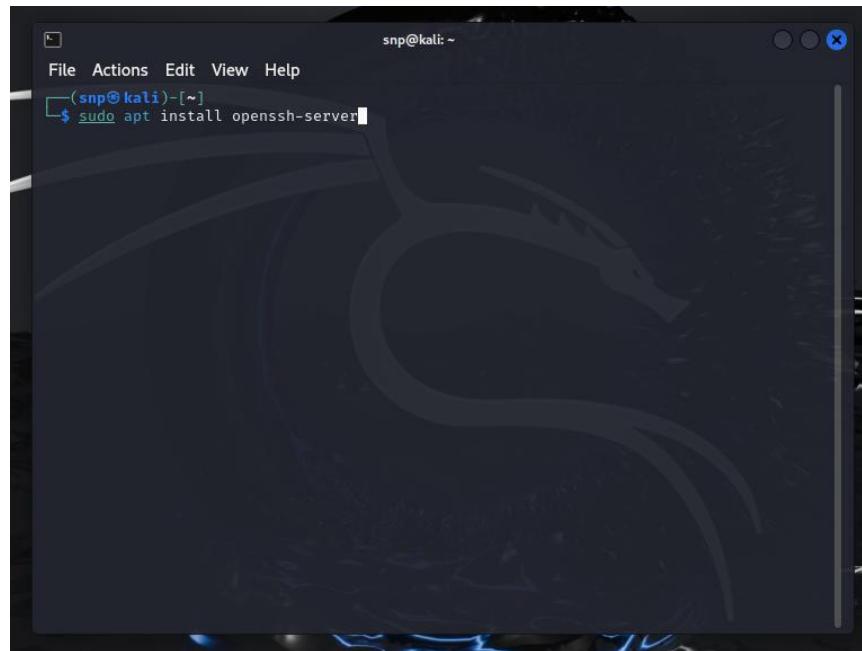
4. Save and exit.

3.2 SSH (Secure Shell) Configuration

Secure Shell (SSH) is a cryptographic network protocol used to securely access and manage network devices and servers over an unsecured network. It provides a secure channel for communication between a client and a server, allowing users to execute commands, transfer files, and manage remote systems safely.

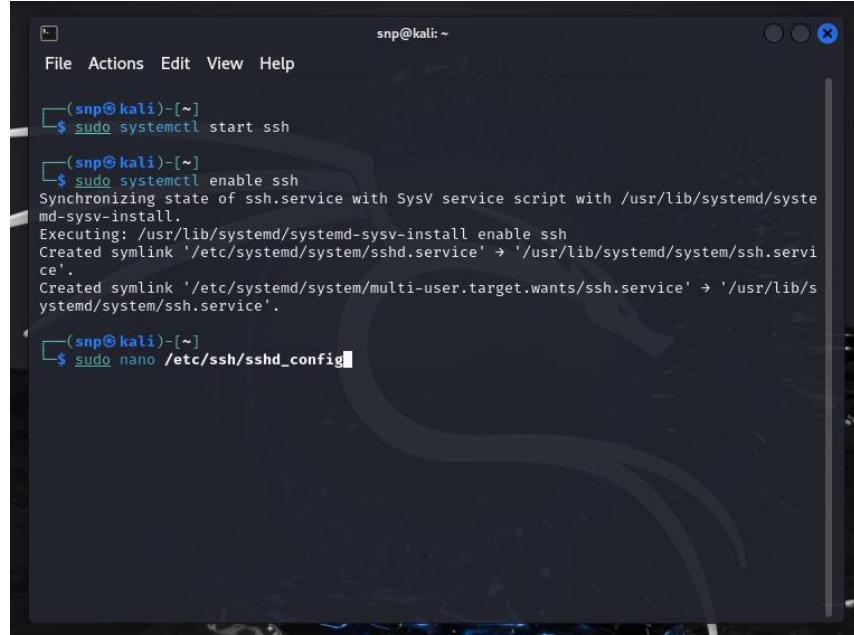
Configuring SSH Server for Secure Remote Access

1. Install SSH Server



A screenshot of a terminal window titled "snp@kali: ~". The window has a dark background with a dragon logo. The terminal menu bar includes "File", "Actions", "Edit", "View", and "Help". Below the menu, it shows the prompt "(snp@kali)-[~]". A cursor is visible at the end of the command line, where the user has typed "\$ sudo apt install openssh-server".

2. Start and Enable SSH.



A terminal window titled 'snp@kali: ~' showing the following commands:

```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ sudo systemctl start ssh
(snp@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/sshd.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/sshd.service' → '/usr/lib/systemd/system/sshd.service'.
(snp@kali)-[~]
$ sudo nano /etc/ssh/sshd_config
```

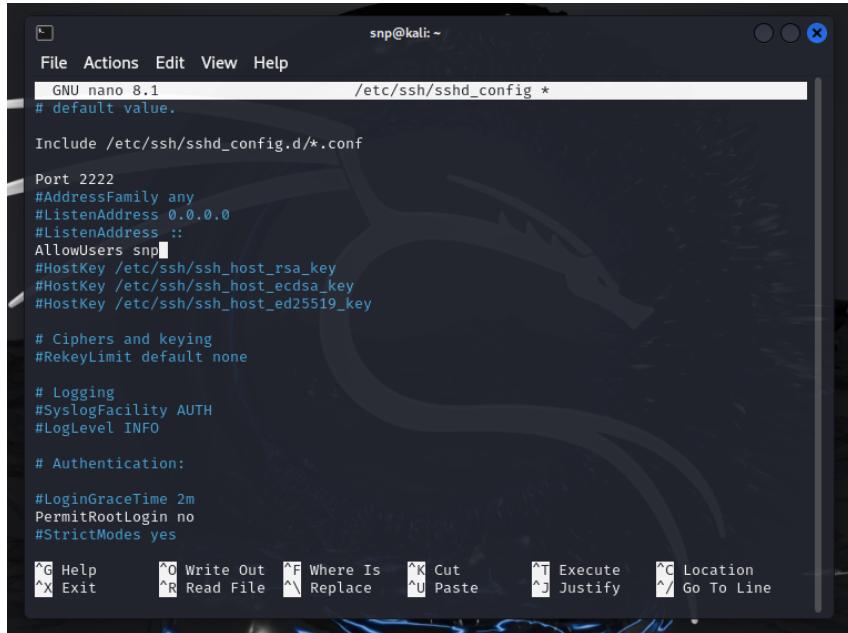
3. Configure SSH Settings

This file controls the behavior of the SSH server.

Port - specifies the port number.

AllowUsers - restricts which users or groups can log in via SSH.

PermitRootLogin – Controls whether root login is allowed.



A terminal window titled 'snp@kali: ~' showing the contents of the /etc/ssh/sshd_config file in the nano editor:

```
GNU nano 8.1          /etc/ssh/sshd_config *
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

AllowUsers snp
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

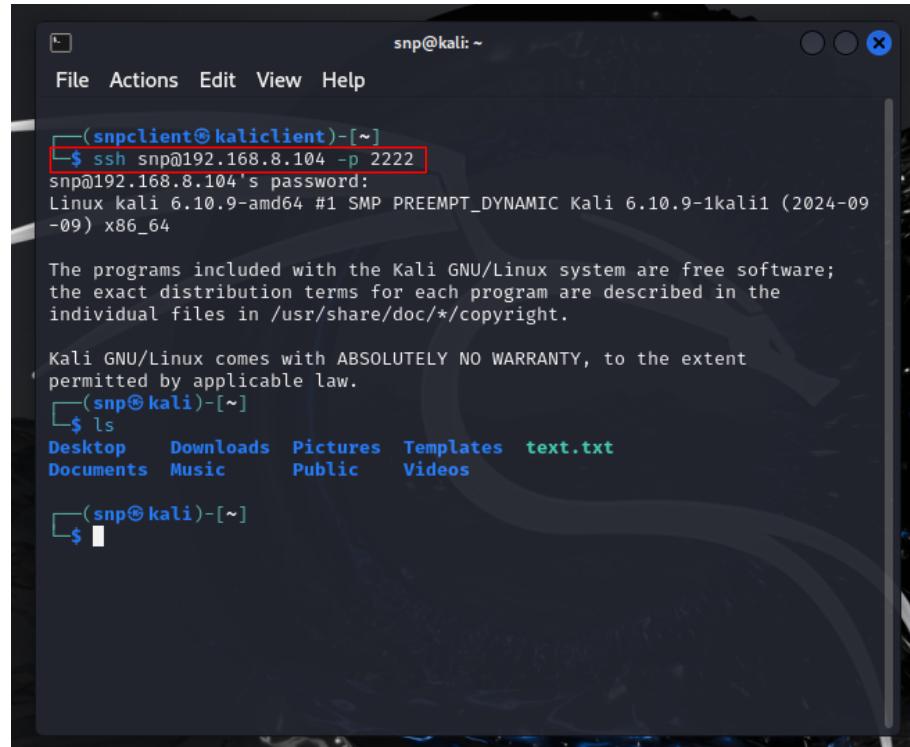
# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
```

4. Test SSH Connection

Attempt to connect to the remote machine using SSH.



The screenshot shows a terminal window titled "snp@snpclient: ~". The window has a dark background with light-colored text. At the top, there's a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu, the terminal prompt is "(snpclient@snpclient)~". A red box highlights the command "\$ ssh snp@192.168.8.104 -p 2222". The terminal then displays the password prompt "snp@192.168.8.104's password:", followed by the system information "Linux kali 6.10.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.10.9-1kali1 (2024-09-09) x86_64". It also includes the copyright notice "The programs included with the Kali GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*copyright." and the warranty notice "Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law." Below this, the command "\$ ls" is run, showing directory contents: Desktop, Downloads, Pictures, Templates, text.txt, Documents, Music, Public, and Videos. The final prompt is "(snp@snp)~\$".

3.3 Network Traffic Management Using iptables and ACLs

iptables is a command-line utility used to configure the Linux kernel's netfilter framework, which acts as a packet filtering mechanism. It is commonly used to set up, maintain, and inspect the rules of a firewall in Linux-based systems.

An **Access Control List (ACL)** is a set of rules that defines which users or system processes have access to particular resources and what operations they are allowed to perform. ACLs are commonly used in networking, file systems, and system security to regulate permissions.

Web Server Security

1. Set the default policy to drop all incoming traffic.

```
sudo iptables -P INPUT DROP
```

2. Allow incoming HTTP (port 80) traffic

```
sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

3. Allow incoming HTTPS (port 443) traffic

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

The screenshot shows a terminal window titled 'snp@kali: ~' with the following command history and output:

```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ sudo iptables -F
(snp@kali)-[~]
$ sudo iptables -P INPUT DROP
(snp@kali)-[~]
$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT    tcp  --  anywhere             anywhere            tcp dpt:http
ACCEPT    tcp  --  anywhere             anywhere            tcp dpt:https

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
(snp@kali)-[~]
$
```

4. Allow established and related traffic

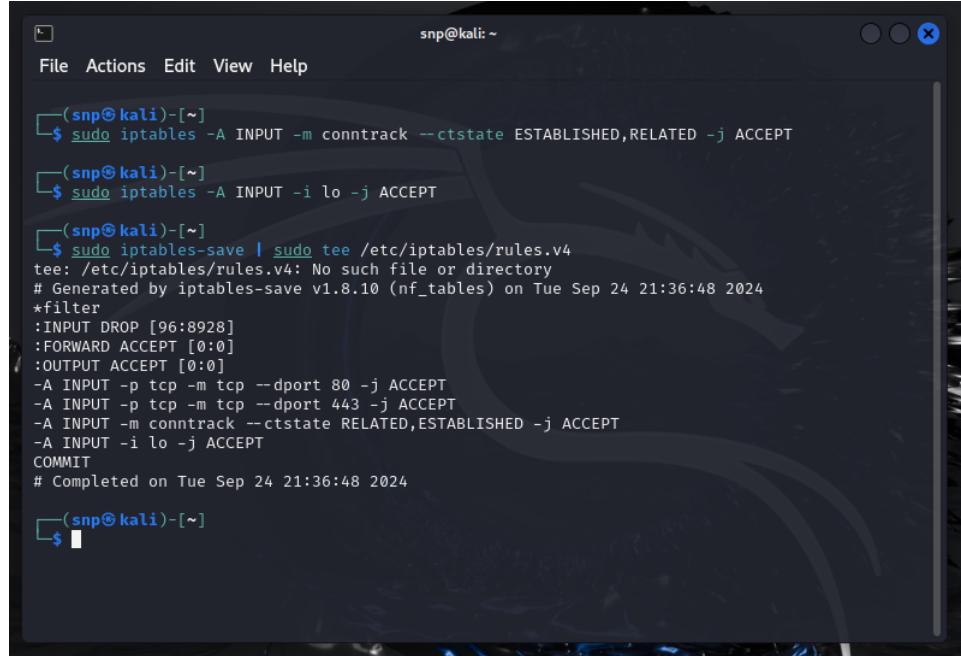
```
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

5. Allow incoming traffic on the loopback interface

```
sudo iptables -A INPUT -i lo -j ACCEPT
```

6. Save the rules.

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```



The screenshot shows a terminal window titled 'snp@kali: ~' with the following command history:

```
(snp@kali)-[~]
$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -A INPUT -i lo -j ACCEPT
(snp@kali)-[~]
$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
tee: /etc/iptables/rules.v4: No such file or directory
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Sep 24 21:36:48 2024
*filter
:INPUT DROP [96:8928]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
COMMIT
# Completed on Tue Sep 24 21:36:48 2024
(snp@kali)-[~]
$
```

Remote Administration Access

1. Allow incoming SSH (port 22) traffic from specific trusted IP addresses

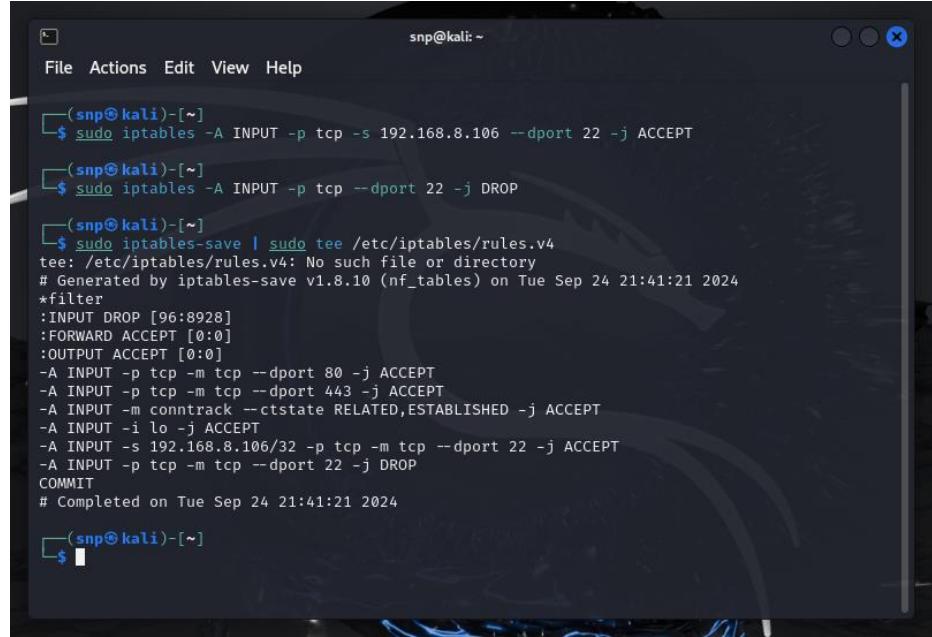
```
sudo iptables -A INPUT -p tcp -s 192.168.8.106 --dport 22 -j ACCEPT
```

2. Block all other SSH access attempts

```
sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

3. Save the rule

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```



The screenshot shows a terminal window titled "snp@kali: ~". The user has run several commands to manage network rules:

```
(snp@kali)-[~]
$ sudo iptables -A INPUT -p tcp -s 192.168.8.106 --dport 22 -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -A INPUT -p tcp --dport 22 -j DROP
(snp@kali)-[~]
$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
tee: /etc/iptables/rules.v4: No such file or directory
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Sep 24 21:41:21 2024
*filter
:INPUT DROP [96:8928]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -s 192.168.8.106/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j DROP
COMMIT
# Completed on Tue Sep 24 21:41:21 2024
(snp@kali)-[~]
$
```

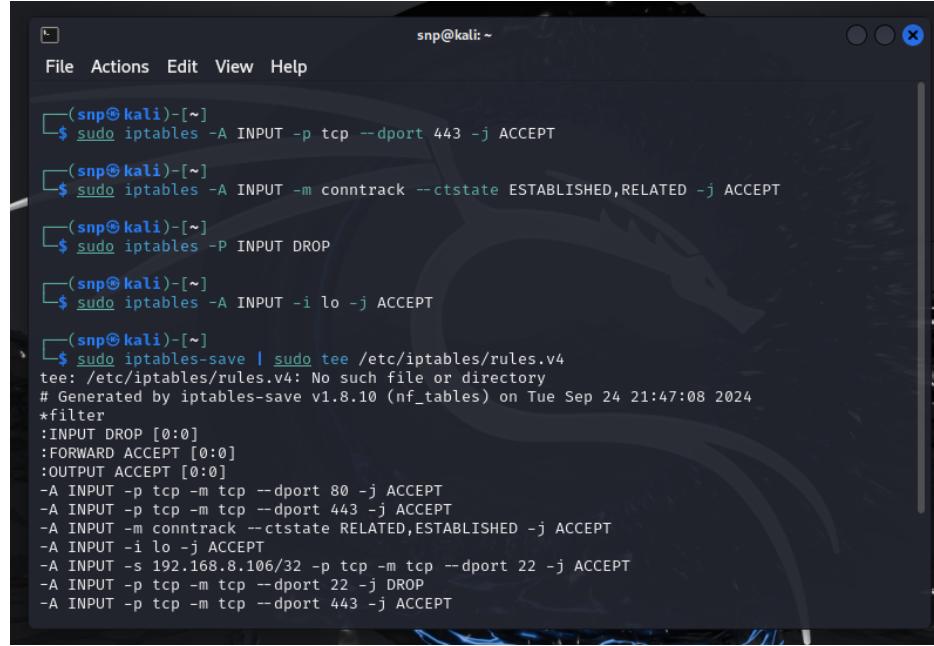
Application Specific Rules

1. Allow HTTPS traffic for video conferencing apps (port 443)

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

2. Save the rules

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```



The screenshot shows a terminal window titled 'snp@kali: ~' with the following command history:

```
(snp@kali)-[~]
$ sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
(snp@kali)-[~]
$ sudo iptables -P INPUT DROP
(snp@kali)-[~]
$ sudo iptables -A INPUT -i lo -j ACCEPT
(snp@kali)-[~]
$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
tee: /etc/iptables/rules.v4: No such file or directory
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Sep 24 21:47:08 2024
*filter
:INPUT DROP [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -s 192.168.8.106/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j DROP
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
```

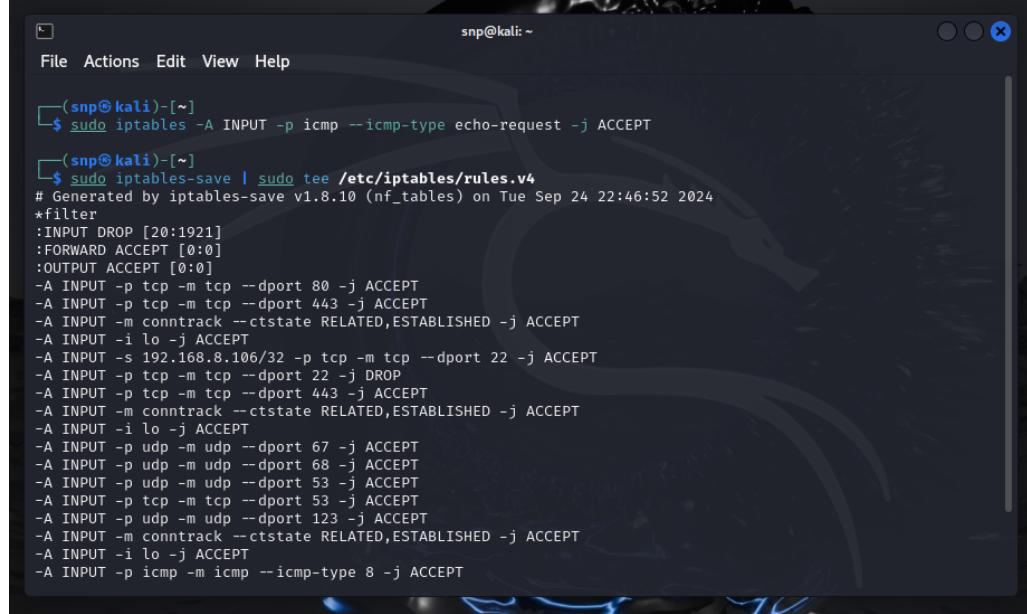
Allowing ICMP Echo Requests (Ping)

1. Allow ICMP Echo requests

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

2. Save the rule

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```



The screenshot shows a terminal window titled 'File Actions Edit View Help' with the command history and output of the following commands:

```
(snp@kali)-[~]
$ sudo iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
(snp@kali)-[~]
$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Sep 24 22:46:52 2024
*filter
:INPUT DROP [20:1921]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -s 192.168.8.106/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j DROP
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p udp -m udp --dport 67 -j ACCEPT
-A INPUT -p udp -m udp --dport 68 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 53 -j ACCEPT
-A INPUT -p udp -m udp --dport 123 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p icmp -m icmp --icmp-type 8 -j ACCEPT
```

Printer Server Security

1. Allow printer traffic from specific ip/ ip range

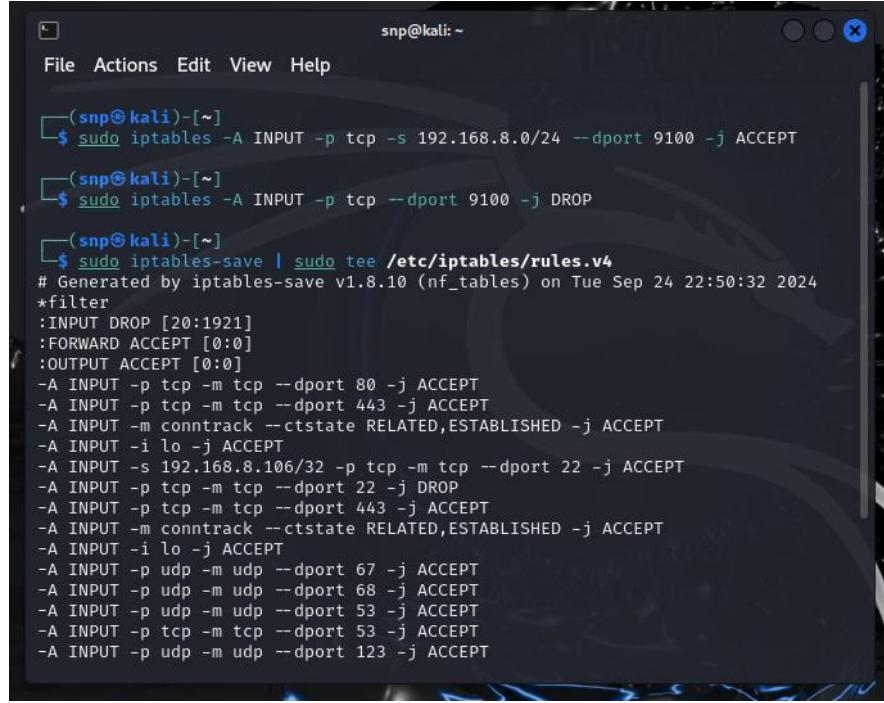
```
sudo iptables -A INPUT -p tcp -s 192.168.8.0/24 --dport 9100 -j ACCEPT
```

2. Drop all incoming TCP traffic directed to port 9100

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

3. Save the rules

```
sudo iptables-save | sudo tee /etc/iptables/rules.v4
```



The screenshot shows a terminal window titled "snp@kali: ~" with the following content:

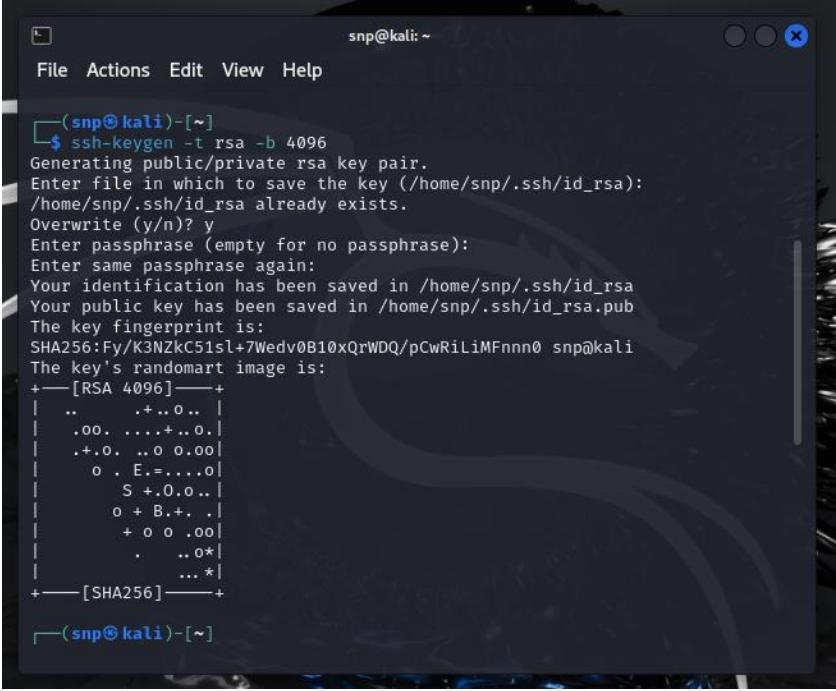
```
(snp@kali)~$ sudo iptables -A INPUT -p tcp -s 192.168.8.0/24 --dport 9100 -j ACCEPT
(snp@kali)~$ sudo iptables -A INPUT -p tcp --dport 9100 -j DROP
(snp@kali)~$ sudo iptables-save | sudo tee /etc/iptables/rules.v4
# Generated by iptables-save v1.8.10 (nf_tables) on Tue Sep 24 22:50:32 2024
*filter
:INPUT DROP [20:1921]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -s 192.168.8.106/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j DROP
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p udp -m udp --dport 67 -j ACCEPT
-A INPUT -p udp -m udp --dport 68 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 53 -j ACCEPT
-A INPUT -p udp -m udp --dport 123 -j ACCEPT
```

4. Best Practices for Network Interface Configuration Security

1. Ensure OpenSSH Server Security

Use SSH key pairs for authentication

One of the best practices includes securing it using an authentication mechanism through SSH keys. The key pairs enable stronger ways of protecting remote access, as opposed to traditional password-based authentication. This implies the generation of a public-private key pair, where the public key is placed on the server and the private key is kept securely on the client machine. This will prevent brute-force attacks on the password but at the same time ensures that only the correct and authorized user, with his or her private key, gains access.



```
snp@kali: ~
File Actions Edit View Help
└─(snp㉿kali)-[~]
$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/snp/.ssh/id_rsa):
/home/snp/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/snp/.ssh/id_rsa
Your public key has been saved in /home/snp/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Fy/K3NZkC51sl+7WedvOB10xQrWDQ/pCwRiLiMFnnn0 snp@kali
The key's randomart image is:
+---[RSA 4096]---+
| .. .+..o.. |
| .oo. ....+..o.|
| .+.o. ..o o.o.|
| o . E=....o.|
| S +.0.o..|
| o + B.+.. .|
| + o o .oo.|
| . ..o*|
| ...*|
+---[SHA256]---+
└─(snp㉿kali)-[~]
```

```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ ssh-copy-id snp@192.168.8.104
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/snp/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to
filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys
snp@192.168.8.104's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'snp@192.168.8.104'"
and check to make sure that only the key(s) you wanted were added.

(snp@kali)-[~]
$ ssh snp@192.168.8.104
Linux kali 6.10.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.10.9-1kali1 (2024-09-09) x86_64

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

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
```

Disable root login

By default, external root access is turned on for all Linux machines. It leaves an open SSH security vulnerability to brute-force attacks by hackers. By disabling the server SSH root login, unauthorized people cannot take control of the system. An active root account allows an attacker to gain or guess the password of the root with full administrative privileges.

```
snp@kali: ~
File Actions Edit View Help
GNU nano 8.1          /etc/ssh/sshd_config *
# Authentication:
#LoginGraceTime 2m
PermitRootLogin no
PasswordAuthentication no
PubkeyAuthentication yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile      .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication

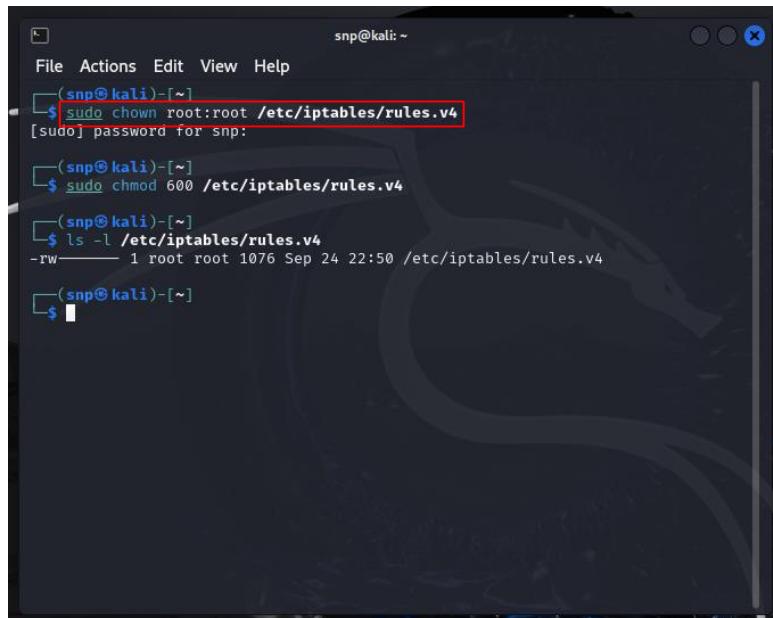
^G Help      ^O Write Out   ^F Where Is    ^K Cut        ^T Execute
^X Exit     ^R Read File   ^V Replace    ^U Paste      ^J Justify
```

2. Securing the iptables Rules File: Ownership and Permissions

This file should be editable and owned by a root user only, in order to secure the iptables rules file. It is of utmost importance that unauthorized editing, which may open up security vulnerabilities, be blocked along with unexpected disclosure and loss of control over network traffic.

1. Ownership

Ensuring that the iptables rules file is owned by the root user prevents non-privileged users from having the ability to alter firewall settings.

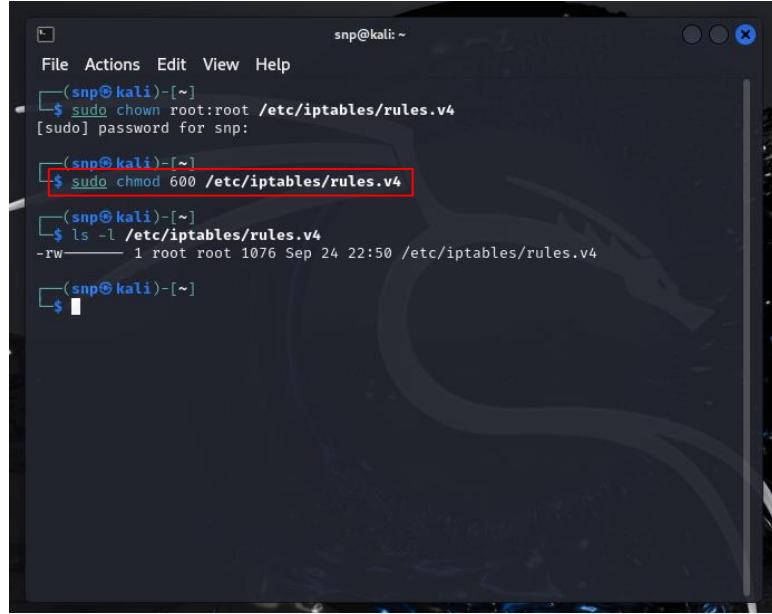


The screenshot shows a terminal window titled "snp@kali: ~". The user runs the command `sudo chown root:root /etc/iptables/rules.v4`, which prompts for a password. After entering the password, the user runs `sudo chmod 600 /etc/iptables/rules.v4`. Finally, the user runs `ls -l /etc/iptables/rules.v4` to verify the file's permissions, showing it is owned by root and has mode 600.

```
(snp㉿kali)-[~]
$ sudo chown root:root /etc/iptables/rules.v4
[sudo] password for snp:
(snp㉿kali)-[~]
$ sudo chmod 600 /etc/iptables/rules.v4
(snp㉿kali)-[~]
$ ls -l /etc/iptables/rules.v4
-rw——— 1 root root 1076 Sep 24 22:50 /etc/iptables/rules.v4
(snp㉿kali)-[~]
$
```

2. Permissions

The rules file should have permissions that allow only the root user to read and write to it while denying all other users access.



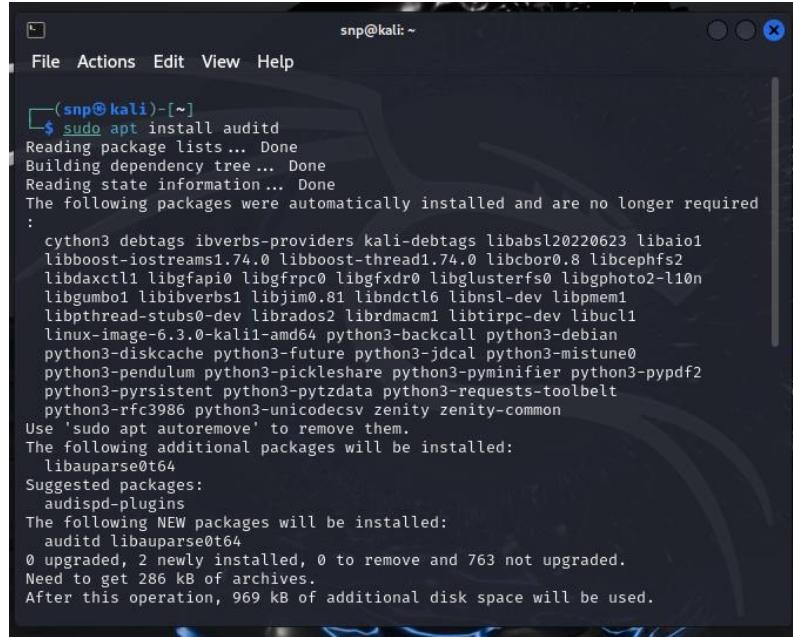
The screenshot shows a terminal window titled "snp@kali: ~". The user has run the command `sudo chown root:root /etc/iptables/rules.v4`, which prompts for a password. The user then runs `sudo chmod 600 /etc/iptables/rules.v4`. After this, they run `ls -l /etc/iptables/rules.v4`, which shows the file has permissions `-rw----- 1 root root 1076 Sep 24 22:50 /etc/iptables/rules.v4`. The terminal prompt ends with a dollar sign.

3. Implementing Auditing Software for Security of the iptables Rules File

Setting up auditing software is majorly critical in monitoring the integrity of your iptables rules file in your network configurations. Auditing allows you to track changes made to a rules file. This gives good visibility into who made those changes, at what time they were made, and what exactly was changed. Such proactive detection helps in unauthorized alterations and further fortifies the general security posture of your Linux system.

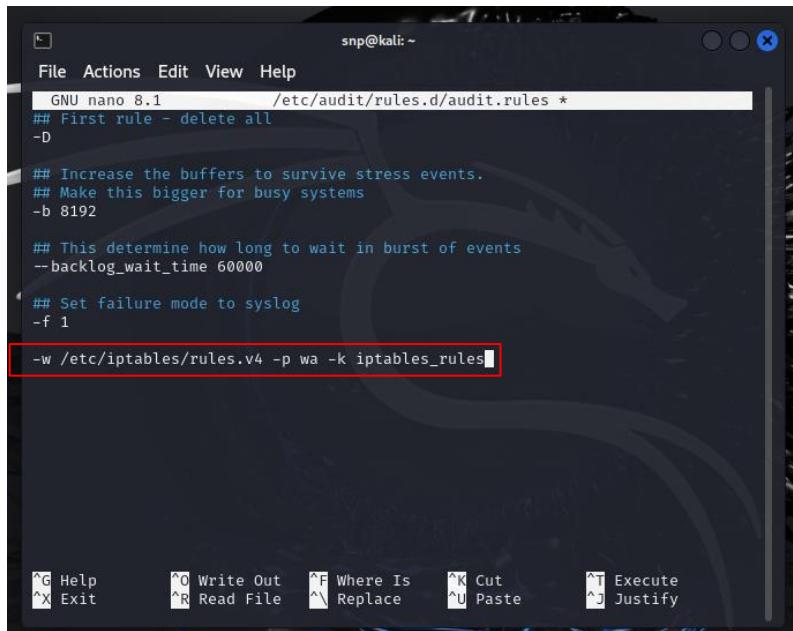
Setting up Auditing with auditd

1. Install auditd



```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ sudo apt install auditd
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required
:
  cython3 debtags ibverbs-providers kali-debtags libabsl20220623 libaio1
  libboost-iostreams1.74.0 libboost-thread1.74.0 libcbor0.8 libcephfs2
  libdaxctl libgapi0 libgRPC0 libgRPCxdr0 libglusterfs0 libgphoto2-l10n
  libgumbo1 libibverbs1 libjim0.81 libndctl6 libnsl-dev libpmem1
  libpthread-stubs0-dev librados2 librDMAcm1 libtirpc-dev libubl1
  linux-image-6.3.0-kali1-amd64 python3-backcall python3-debian
  python3-diskcache python3-future python3-jdcal python3-mistune0
  python3-pendulum python3-pickleshare python3-pyminifier python3-pypdf2
  python3-persistent python3-pytzdata python3-requests-toolbelt
  python3-rfc3986 python3-unicodecsv zenity zenity-common
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libauparse0t64
Suggested packages:
  audispd-plugins
The following NEW packages will be installed:
  auditd libauparse0t64
0 upgraded, 2 newly installed, 0 to remove and 763 not upgraded.
Need to get 286 kB of archives.
After this operation, 969 kB of additional disk space will be used.
```

2. Add a monitoring rule



```
File Actions Edit View Help
GNU nano 8.1      /etc/audit/rules.d/audit.rules *
## First rule - delete all
-D

## Increase the buffers to survive stress events.
## Make this bigger for busy systems
-b 8192

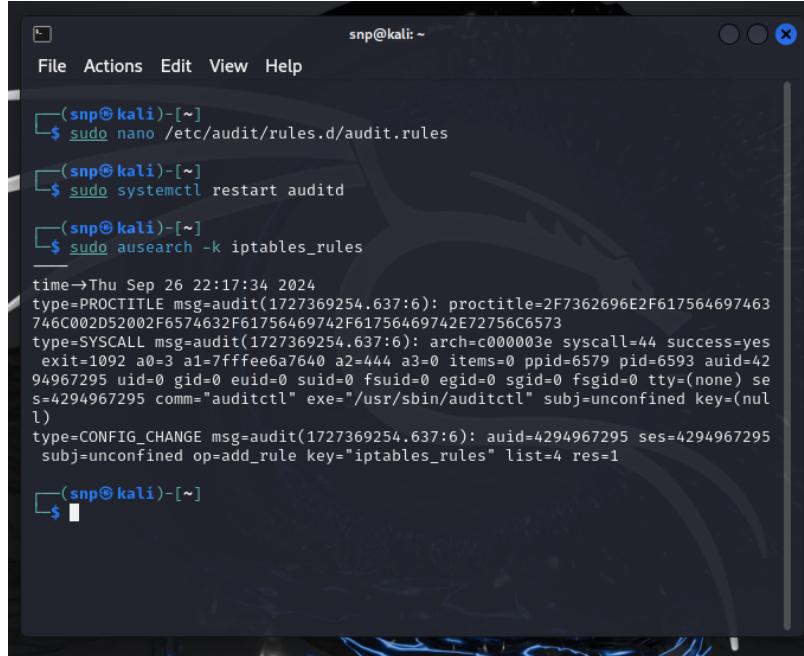
## This determine how long to wait in burst of events
--backlog_wait_time 60000

## Set failure mode to syslog
-f 1

-w /etc/iptables/rules.v4 -p wa -k iptables_rules■
```

3. Restart auditd

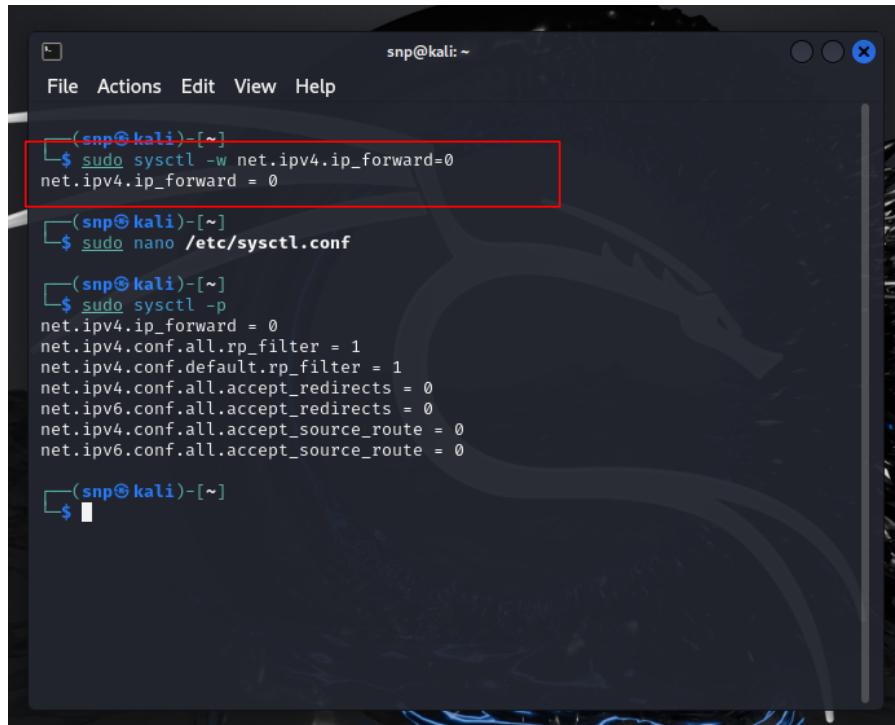
4. View audit logs



```
snp@kali: ~
File Actions Edit View Help
└─(snp㉿kali)-[~]
  └─$ sudo nano /etc/audit/rules.d/audit.rules
  └─(snp㉿kali)-[~]
  └─$ sudo systemctl restart auditd
  └─(snp㉿kali)-[~]
  └─$ sudo ausearch -k iptables_rules
time→Thu Sep 26 22:17:34 2024
type=PROCTITLE msg=audit(1727369254.637:6): proctitle=2F7362696E2F617564697463
746C002D52002F6574632F61756469742F61756469742E72756C6573
type=SYSCALL msg=audit(1727369254.637:6): arch=c000003e syscall=44 success=yes
exit=1092 a0=3 a1=7fffe6a7640 a2=444 a3=0 items=0 ppid=6579 pid=6593 auid=42
94967295 uid=0 gid=0 euid=0 suid=0 egid=0 sgid=0 fsgid=0 tty=(none) se
s=4294967295 comm="auditctl" exe="/usr/sbin/auditctl" subj=unconfined key=(nul
l)
type=CONFIG_CHANGE msg=audit(1727369254.637:6): auid=4294967295 ses=4294967295
subj=unconfined op=add_rule key="iptables_rules" list=4 res=1
└─(snp㉿kali)-[~]
  └─$
```

4. Enhancing Security by Disabling IP Forwarding on Linux

Disabling IP forwarding is a critical security measure for Linux systems that do not require routing capabilities. By setting net.ipv4.ip_forward=0, you effectively prevent the system from forwarding packets between different network interfaces.



```
snp@kali: ~
File Actions Edit View Help
└─(snp㉿kali)-[~]
  └─$ sudo sysctl -w net.ipv4.ip_forward=0
  net.ipv4.ip_forward = 0
  └─(snp㉿kali)-[~]
  └─$ sudo nano /etc/sysctl.conf
  └─(snp㉿kali)-[~]
  └─$ sudo sysctl -p
  net.ipv4.ip_forward = 0
  net.ipv4.conf.all.rp_filter = 1
  net.ipv4.conf.default.rp_filter = 1
  net.ipv4.conf.all.accept_redirects = 0
  net.ipv6.conf.all.accept_redirects = 0
  net.ipv4.conf.all.accept_source_route = 0
  net.ipv6.conf.all.accept_source_route = 0
  └─(snp㉿kali)-[~]
  └─$
```

Positive implications for security by disable IP forwarding

- Reduced attack surface
- Prevention unauthorized Access
- Improved firewall efficiency
- Simplified incident response and monitoring

5. Transitioning from Dynamic to Static IP Addresses

Dynamic IP addresses are in use for convenience and ease of management on any network. However, there are a few major security risks associated with them that might even lead to revealing systems and networks.

Dynamic IP addressing can make an attacker's potential for an already assigned IP address impersonate users and devices. This impersonation may facilitate unauthorized access to sensitive resources, especially in systems using only verification of IP addresses as a means of authentication, thus raising a serious risk to security.

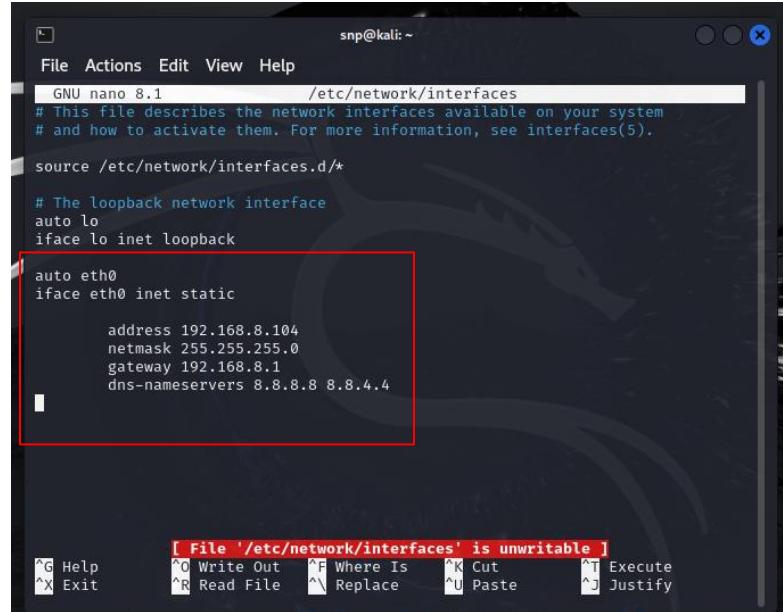
Setting a static IP address means it would assign a certain fixed IP address of a device on a network, so upon reboots or disconnections, it retains the same address. It is quite essential for devices like servers, printers, and networked storage devices when continuous access and identification over a network are needed.

Setting a static IP address

1. Select an IP address within the subnet of your network that is not already in use by other devices.

192.168.8.104 /24

2. Edit network configuration



```
snp@kali: ~
File Actions Edit View Help
GNU nano 8.1          /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

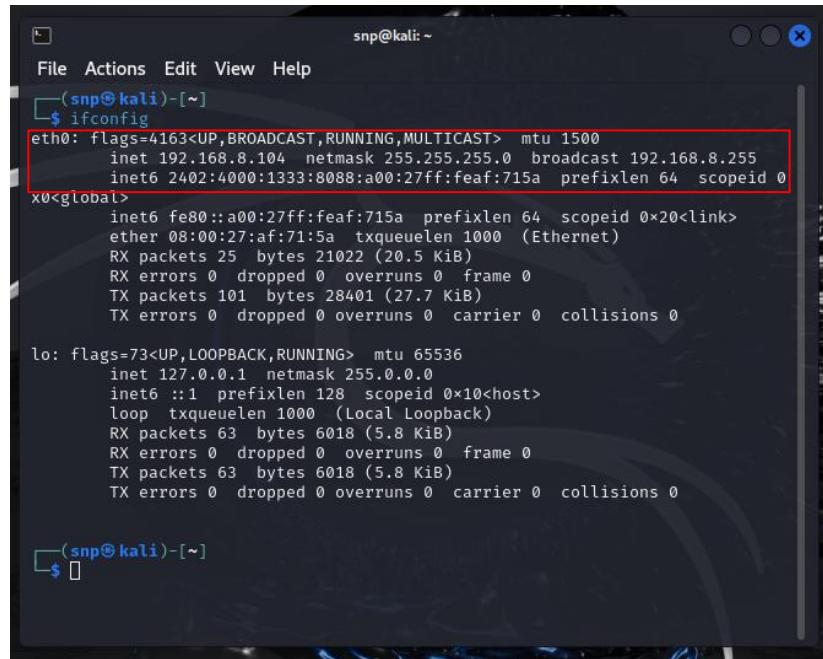
# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.8.104
    netmask 255.255.255.0
    gateway 192.168.8.1
    dns-nameservers 8.8.8.8 8.8.4.4

[ File '/etc/network/interfaces' is unwritable ]
```

The screenshot shows a terminal window titled "File Actions Edit View Help" with the command "GNU nano 8.1 /etc/network/interfaces". The file content is displayed, showing the configuration for the "eth0" interface. A red box highlights the "auto eth0" section, which includes the "inet static" configuration. The message "[File '/etc/network/interfaces' is unwritable]" is visible at the bottom of the screen.

3. Restart networking service
4. Verify configuration



```
snp@kali: ~
File Actions Edit View Help
(snp@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.8.104 netmask 255.255.255.0 broadcast 192.168.8.255
        inet6 2402:4000:1333:8088:a00:27ff:feaf:715a prefixlen 64 scopeid 0
            inet6 fe80::a00:27ff:feaf:715a prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:af:71:5a txqueuelen 1000 (Ethernet)
                    RX packets 25 bytes 21022 (20.5 KiB)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX packets 101 bytes 28401 (27.7 KiB)
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 63 bytes 6018 (5.8 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 63 bytes 6018 (5.8 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(snp@kali)-[~]
$
```

The screenshot shows a terminal window titled "File Actions Edit View Help" with the command "\$ ifconfig". The output displays network interface configurations. A red box highlights the "eth0" interface entry, which shows its IP address (192.168.8.104), subnet mask (255.255.255.0), broadcast address (192.168.8.255), and various statistics for received and transmitted packets.

References

- [1] 07 A. 2017 D. B. 500up 5 comments, “Build your own DNS server on Linux,” *Opensource.com*. <https://opensource.com/article/17/4/build-your-own-name-server>
- [2] A. Shibani, “Baeldung,” *Baeldung on Linux*, Jan. 24, 2024. <https://www.baeldung.com/linux/install-configure-dhcp-server>.