

Sri Lanka Institute of Information Technology



**Systems and Network Programming (SNP):
Linux Environments**

MNM. RUHAIM

IT23256446

Contents

INTRODUCTION.....	3
Basics of Linux Environments	4
Virtual machine installation setup.....	4
The basic commands on kali Linux.....	8
System introduction and user management on kali linux	13
DHCP, DNS and NTP Services	15
DHCP (Dynamic Host Configuration Protocol).....	15
DNS (Domain Name System)	18
NTP (Network Time Protocol)	24
Shell Scripting and Security.....	26
Basic Shell scripting on Linux system	26
SSH (Secure Shell)	30
iptables and ACLs.....	32
Best practices for security aspects of network interface configuration.....	34
Disable and unsend Network interface	34
Configuration firewall	35
Monitor network traffic.....	36
Disable IPv6 if Not Used.....	37
Enable Logging and Auditing	38

INTRODUCTION

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

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

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

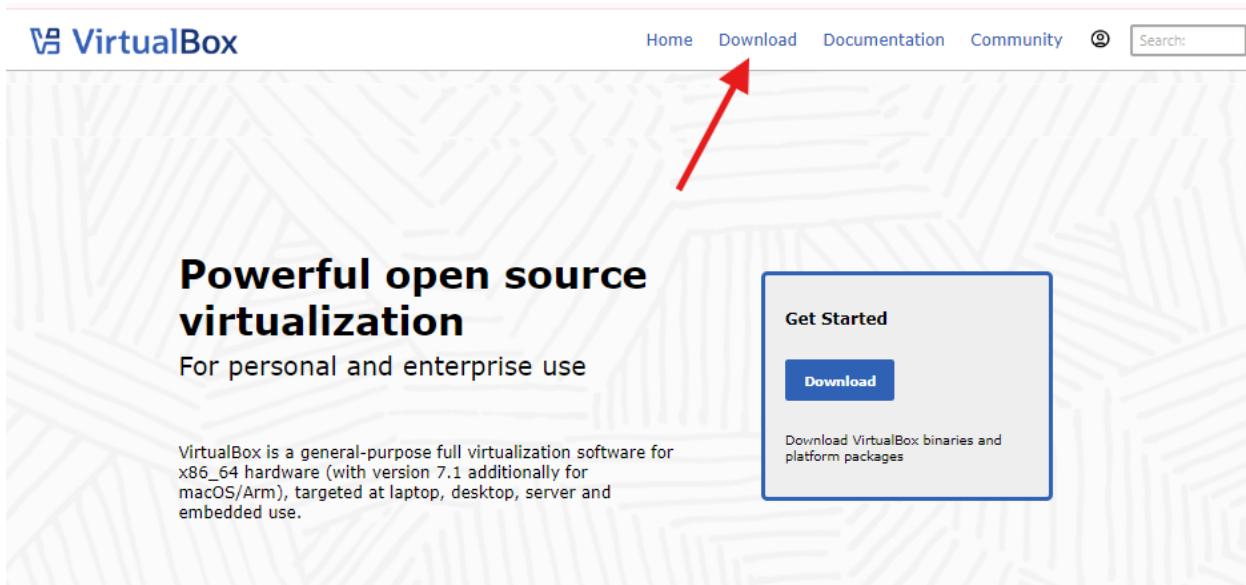
Basics of Linux Environments

Virtual machine installation setup

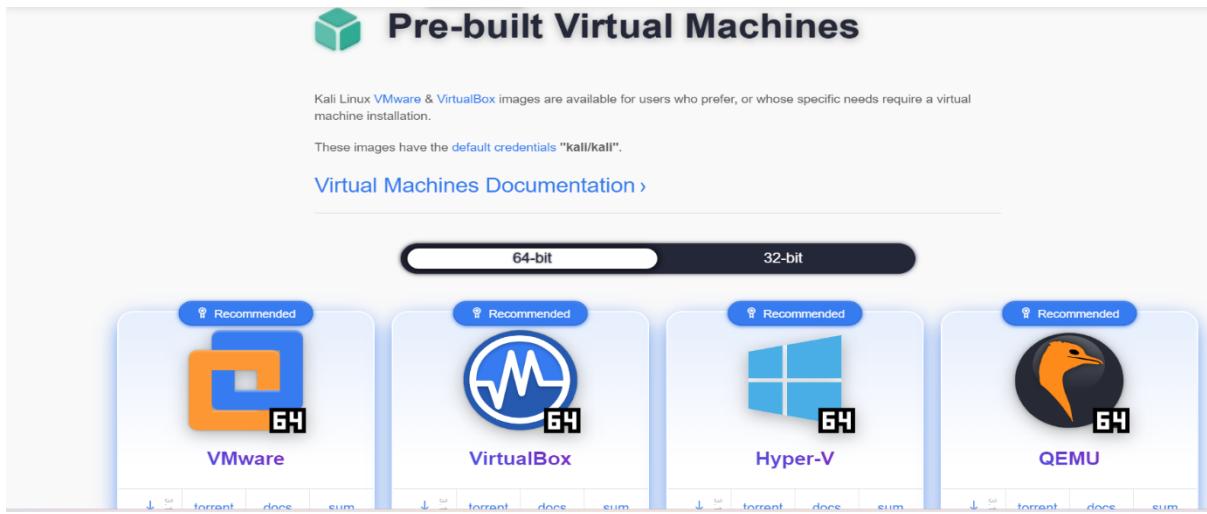
Step 01: Click the virtual machine installation link

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

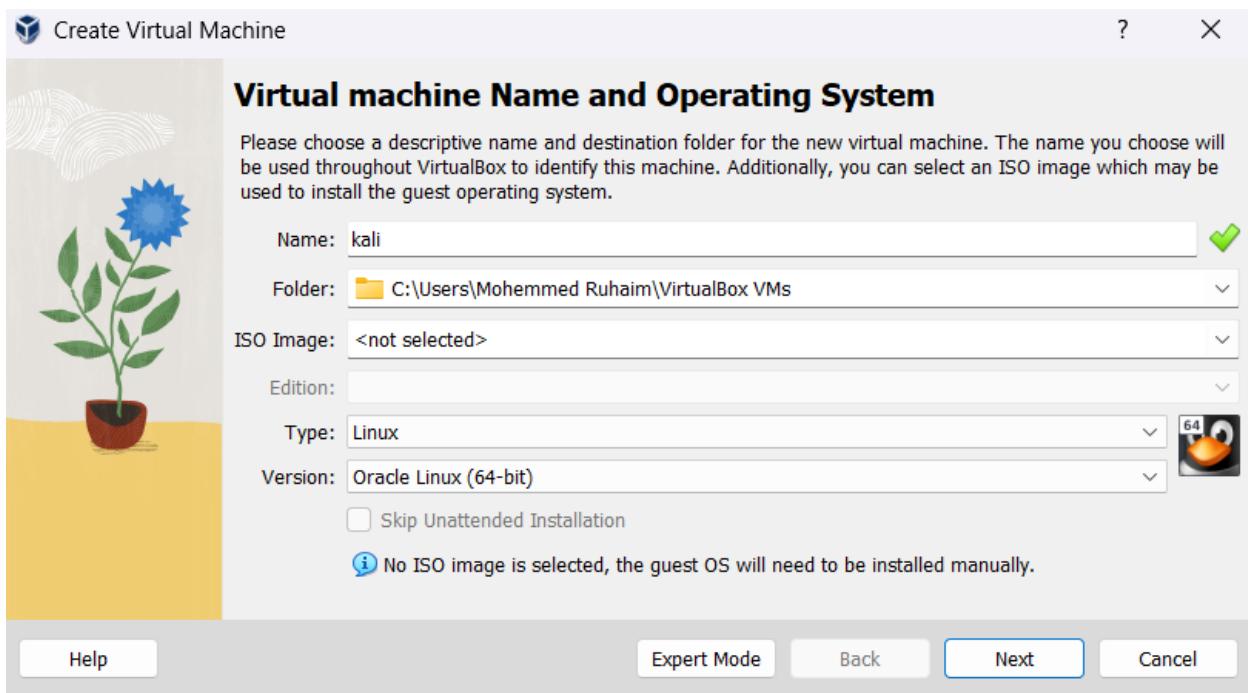
Step 02: Click the Download button.



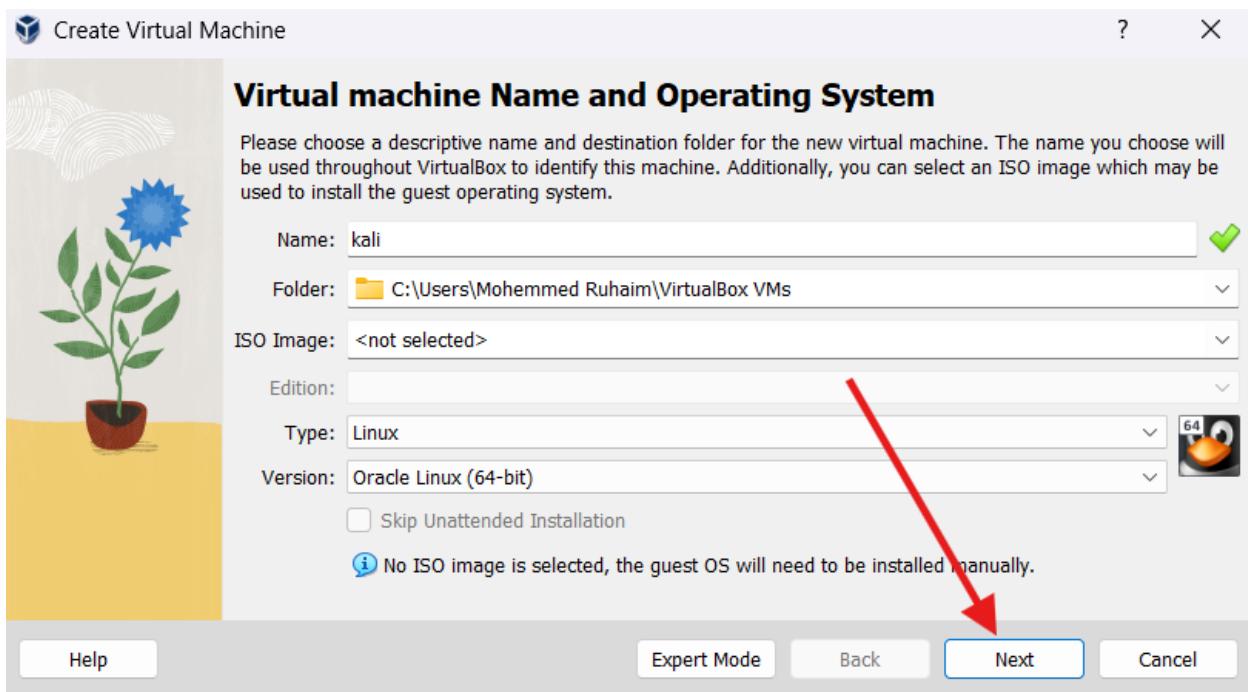
Step 03: click on the VirtualBox.



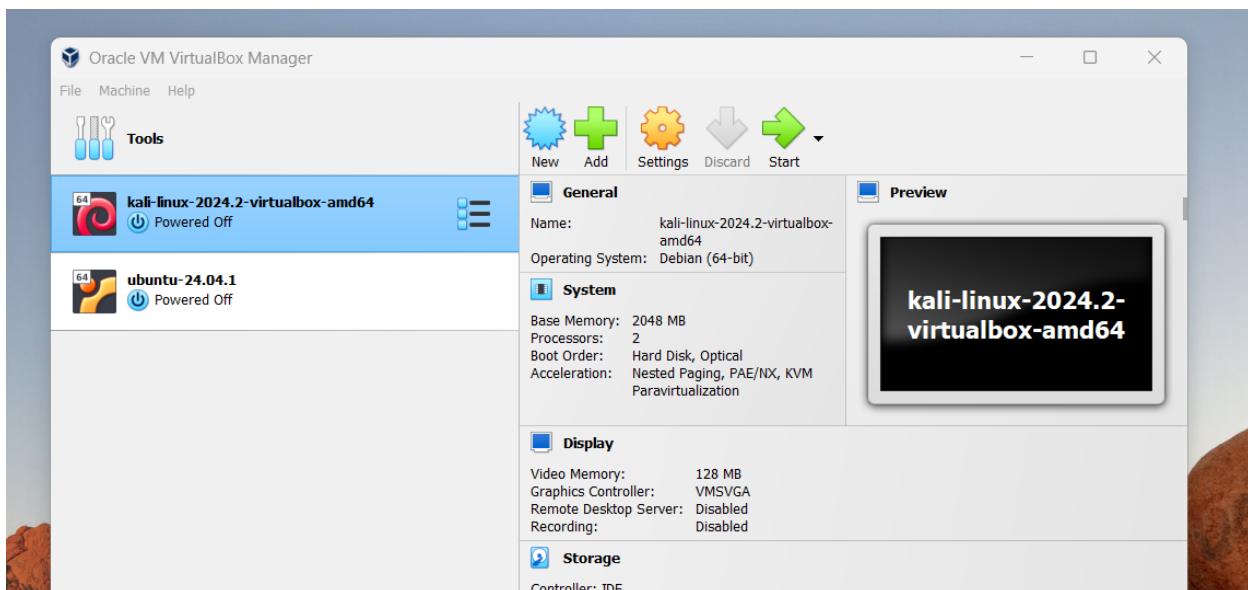
Step 04: Enter the Virtual machine name and version.



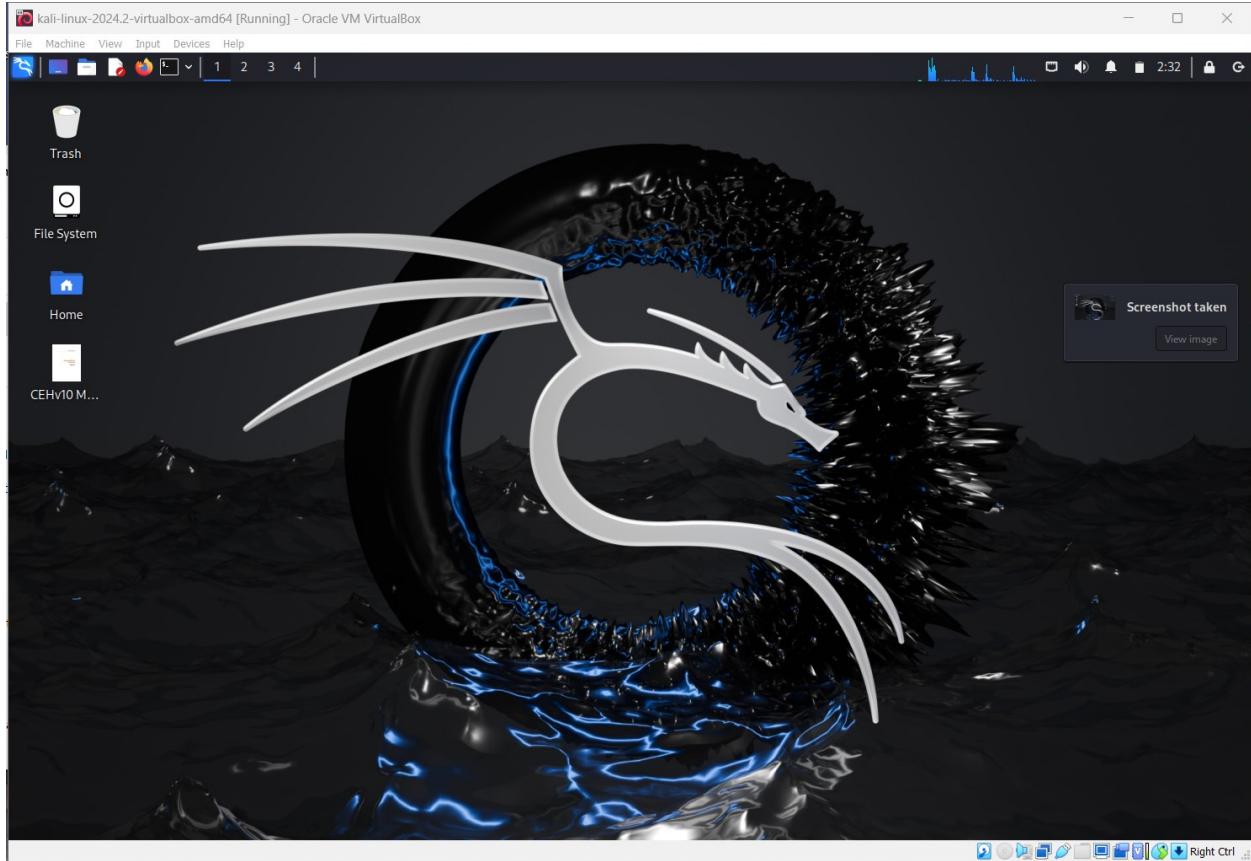
Step 05: Click on the next button.



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



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



Note: successfully installed kali Linux operating system.

The basic commands on kali Linux

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

```
(kali㉿kali)-[~]
└─$ pwd
/home/kali
```

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

```
(kali㉿kali)-[~]
└─$ ls
a.out      cod.sh    Documents  ii          Music      Public    soslab3.c  TheFatRat
bandit.txt  Desktop   Downloads  IT23256446  Pictures   snp.sh    Templates  Videos
```

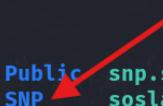
3. **ls -a**: list down the all hidden files.

```
(kali㉿kali)-[~]
└─$ ls -a
.
..              .ICEauthority
a.out          .ICEdaemon
bandit.txt     .ii
.bash_logout    .java
.bashrc         .local
.bashrc.original .mozilla
.BurpSuite      .msf4
.cache          .Music
.cache          .php_history
cod.sh          .Pictures
.config         .pki
 dbus           .profile
Desktop        .Public
.dmrc           .ruhaim.swp
Documents       .snp.sh
Downloads       .soslab3.c
.face           .ssh
.face.icon      .student.swo
.gnupg          .student.swp
.gvfs           .sudo_as_admin_successful
Templates
TheFatRat
.vboxclient-clipboard-tty7-control.pid
.vboxclient-clipboard-tty7-service.pid
.vboxclient-display-svga-x11-tty7-control.pid
.vboxclient-display-svga-x11-tty7-service.pid
.vboxclient-draganddrop-tty7-control.pid
.vboxclient-draganddrop-tty7-service.pid
.vboxclient-hostversion-tty7-control.pid
.vboxclient-seamless-tty7-control.pid
.vboxclient-seamless-tty7-service.pid
.vboxclient-vmsvga-session-tty7-control.pid
Videos
.viminfo
.Xauthority
.xsession-errors
.xsession-errors.old
.zsh_history
.zshrc
```

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

```
[kali㉿kali)-[~]
$ mkdir SNP

[kali㉿kali)-[~]
$ ls
a.out      cod.sh    Documents  ii          Music       Public     SNP        snp.sh    Templates  Videos
bandit.txt  Desktop   Downloads  IT23256446  Pictures    SNP        SNP       soslab3.c  TheFatRat

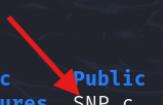

```

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

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

```
[kali㉿kali)-[~]
$ touch SNP.c

[kali㉿kali)-[~]
$ ls
a.out      cod.sh    Documents  ii          Music       Public     SNP.c      snp.sh    Templates  Videos
bandit.txt  Desktop   Downloads  IT23256446  Pictures    SNP       SNP.c    soslab3.c  TheFatRat


```

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

```
[kali㉿kali)-[~]
$ rm SNP.c

[kali㉿kali)-[~]
$ ls
a.out      cod.sh    Documents  ii          Music       Public     soslab3.c  TheFatRat
bandit.txt  Desktop   Downloads  IT23256446  Pictures    SNP       snp.sh    Templates  Videos


```

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

```
[kali㉿kali)-[~]
$ cat cod.sh
#!/bin/bash
echo "hello world"
echo "welcom Ruhaim"
cal
ls
date
echo "complete"
```

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

```
(kali㉿kali)-[~]
└─$ nano bandit.txt
```

```
File Actions Edit View Help
GNU nano 8.1                                bandit.txt
level01- ZjljtM6FvvyRnrb2rfNW0ZOTa6ip5If
level02- 263JGPfgU6LtdEvgfWU1XP5yac29mFx
level03- MNk8KNH3Usioo41PRUEoDFPqfxLPlSmx
level04- 2WmrDFRmJIq3IPxneAaMGhap0pFhF3NJcay
level05- 4oQYVPkxZ00EO05pTW81FB8j8lxXGUQw
level16- kSkuUpMQ7lBYyCM4GBPvCvT1BfWRy0Dx
```

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

read- r

write- w

execute- x

```
(kali㉿kali)-[~]
└─$ ls
a.out      cod.sh    Documents  ii          Music      Public    soslab3.c  TheFatRat
bandit.txt  Desktop   Downloads  IT23256446  Pictures   sns.sh    Templates Videos
(kali㉿kali)-[~]
└─$ chmod 755 bandit.txt
```

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

```
(kali㉿kali)-[~]
└─$ cp bandit.txt Desktop
```

```
(kali㉿kali)-[~]
└─$ Desktop
```

```
(kali㉿kali)-[~/Desktop]
└─$ ls
bandit.txt  'CEHv10 Module 17 Hacking Mobile Platforms.pdf'
```

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

```
└─(kali㉿kali)-[~]
└─$ pwd
/home/kali

└─(kali㉿kali)-[~]
└─$ cd ..
└─(kali㉿kali)-[/home]
└─$ pwd
/home
```

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

```
└─(kali㉿kali)-[~]
└─$ mv IT23256446 TEST

└─(kali㉿kali)-[~]
└─$ ls
a.out      cod.sh    Documents  ii      Pictures  sfp.sh     Templates  TheFatRat
bandit.txt  Desktop   Downloads  Music   Public    soslab3.c  TEST      Videos
```

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

```
└─(kali㉿kali)-[~]
└─$ sudo su
[sudo] password for kali:
└─(root㉿kali)-[/home/kali]
└─#
```

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

```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
          inet6 fe80::af6d:fdb9:41ec:53f2  prefixlen 64  scopeid 0x20<link>
            ether 08:00:27:d2:26:79  txqueuelen 1000  (Ethernet)
              RX packets 1  bytes 590 (590.0 B)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 27  bytes 3312 (3.2 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 8  bytes 480 (480.0 B)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 8  bytes 480 (480.0 B)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

15. **whoami:** gives your user name.

16. **who:** gives your some details.

```
(kali㉿kali)-[~]
$ whoami
kali

(kali㉿kali)-[~]
$ who
kali    tty7        2024-09-26 02:31 (:0)
```

System introduction and user management on kali linux

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

```
[kali㉿kali)-[~]
$ uname
Linux
```

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

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

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

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

4. **df -h:** When you run df -h, you will see a list of all mounted file systems along with the following information for each:

Filesystem: The name of the file system or disk.

Size: The total size of the file system.

Used: The amount of space that is currently used.

Available: The amount of space that is still available for use.

Use%: The percentage of the file system that is currently used.

Mounted on: The mount point or directory where the file system is accessed.

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

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

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

DHCP, DNS and NTP Services

DHCP (Dynamic Host Configuration Protocol)

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

- **sudo apt update**

```
(kali㉿kali)-[~]
$ sudo apt update
[sudo] password for kali:
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.1
MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb)
[48.7 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [11
0 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (de
b) [268 kB]
```

step 02: install DHCP server. The installation command is

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

```
(kali㉿kali)-[~]
$ sudo apt install isc-dhcp-server
[sudo] password for kali:
isc-dhcp-server is already the newest version (4.4.3-P1-5).
The following packages were automatically installed and are no longer required:
  fonts-liberation2      libibverbs1      libibusmuxd6
  ibverbs-providers     libimobiledevice6  openjdk-17-jdk
  libassuan0            libjsoncpp25    openjdk-17-jdk-headless
  libavfilter9          libndctl16     openjdk-17-jre
  libboost-iostreams1.83.0  libplacebo338  openjdk-17-jre-headless
  libboost-thread1.83.0   libplist3      python3-diskcache
  libcephfs2           libpmem1      python3-mistune0
```

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

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

This is the configuration and you can start with

```
subnet 10.0.2.0 netmask 255.255.255.0 {  
    range 10.0.2.20 10.0.2.50;  
    option domain-name-servers 8.8.8.8, 8.8.4.4;  
    option subnet-mask 255.255.255.0;  
}
```

```
GNU nano 8.1                               /etc/dhcp/dhcpd.conf  
#}  
  
#shared-network 224-29 {  
#    subnet 10.17.224.0 netmask 255.255.255.0 {  
#        option routers rtr-224.example.org;  
#    }  
#    subnet 10.0.29.0 netmask 255.255.255.0 {  
#        option routers rtr-29.example.org;  
#    }  
#    pool {  
#        allow members of "foo";  
#        range 10.17.224.10 10.17.224.250;  
#    }  
#    pool {  
#        deny members of "foo";  
#        range 10.0.29.10 10.0.29.230;  
#    }  
#}  
subnet 10.0.2.0 netmask 255.255.255.0 {  
    range 10.0.2.20 10.0.2.50; # IP range for clients  
    option routers 10.0.2.1;      # Default gateway  
    option domain-name-servers 8.8.8.8, 8.8.4.4; # DNS servers  
    option subnet-mask 255.255.255.0; # Subnet mask  
}
```

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

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

```
(kali㉿kali)-[~]
└─$ sudo systemctl restart isc-dhcp-server.service
[CEHv10-M1] 2024-10-01 01:32:43
```

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

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

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

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

Note: Successfully Run DHCP server.

DNS (Domain Name System)

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

- **sudo apt update**

```
(kali㉿kali)-[~]
$ sudo apt update
[sudo] password for kali:
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.1
MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb)
[48.7 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [11
0 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (de
b) [268 kB]
```

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

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

```
(kali㉿kali)-[~]
$ sudo apt install bind9 bind9utils bind9-doc
The following packages were automatically installed and are no longer r
equired:
  fonts-liberation2          libpmem1
  ibverbs-providers          libpostproc57
  libassuan0                 librados2
  libavfilter9                librdmacm1t64
  libboost-iostreams1.83.0   libre2-10
  libboost-thread1.83.0      libroco.3
  libcephfs2                  libu2f-udev
  libdaxctl1                  libusbmuxd6
  libgeos3.12.1t64            openjdk-17-jdk
```

Step 03: edit configuration file. The command should be

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

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

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

- we needs to edit should be.

You need to change the Ip address for your Ip address

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

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

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

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

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

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

- we want to do some change on this option file.

That is change the forwarders {8.8.8.8 or 8.8.4.4}

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

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

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

    // forwarders {
    //     8.8.8.8;

    // };
```

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

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

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

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

- Change the Ip address and puts your Ip address

```
GNU nano 8.1          /etc/bind/db.127
; BIND reverse data file for 10.0.2.15/24 ←
; Do any local configuration here
$TTL    604800
@      IN      SOA     ns1.itsme.com. (
// Consider adding the 2024092901 here; Serial
// organization        604800           ; Refresh
//include "/etc/bind/zones.rev1918";   ; Retry
//                                86400            ; Expire
//                                2419200         ; Negative Cache TTL
zone "itsme.com"{
;
@      IN      NS      ns1.itsme.com.
1.0.0  IN      PTR      ns1.itsme.com.

zone "2.0.10.in-addr-arpa" {
;
```

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

- **sudo named-checkzone file path**

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

step 07: Restart or start the dns server.

- **sudo systemctl restart named.service**

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

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

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

- **Sudo systemctl status**

```
File Actions Edit View Help
● named.service - BIND Domain Name Server
  Loaded: loaded (/usr/lib/systemd/system/named.service; disabled; p>
    Active: active (running) since Tue 2024-10-01 10:36:38 EDT; 7s ago
  Invocation: 059940c0bb3a49778ed521fd1e4033e4
    Docs: man:named(8)
   Main PID: 31749 (named)
     Status: "running"
       Tasks: 6 (limit: 2221)
      Memory: 18.5M (peak: 19M)
        CPU: 26ms
      CGroup: /system.slice/named.service
              └─31749 /usr/sbin/named -f -u bind

Oct 01 10:36:38 kali named[31749]: network unreachable resolving './NS/>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './DNS>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './NS/>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './DNS>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './NS/>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './DNS>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './NS/>
Oct 01 10:36:38 kali named[31749]: managed-keys-zone: Unable to fetch D>
Oct 01 10:36:38 kali named[31749]: network unreachable resolving './NS/>
Oct 01 10:36:38 kali named[31749]: resolver priming query complete: fai>
lines 1-23
```

Note: successfully Run DNS server.

NTP (Network Time Protocol)

Step 01: update your operating system.

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

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

```
(kali㉿kali)-[~]
$ sudo apt install ntp -y
The following packages were automatically installed and are no longer required:
  fonts-liberation2           libpmem1
  ibverbs-providers          libpostproc57
  libassuan0                  librados2
  libavfilter9                 libre2-10
  libboost-iostreams1.83.0    librdmacm1t64
  libboost-thread1.83.0       libroc0.3
  libcephfs2                  libu2f-udev
  libdav1d1                   libusbx-0.2.10
```

Step 03: edit your ntp configuration file.

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

```
File Actions Edit View Help
GNU nano 8.1                               /etc/ntp.conf

# Use public servers from the pool.ntp.org project.
server 0.pool.ntp.org iburst
server 1.pool.ntp.org iburst
server 2.pool.ntp.org iburst
server 3.pool.ntp.org iburst

# Local NTP server
server 10.0.2.15 iburst

restrict 10.0.2.0 mask 255.255.255.0 nomodify notrap
restrict default nomodify notrap nopeer noquery

# Enable logging
logfile /var/log/ntp.log
```

Step 04: restart and checked the status

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

```
(kali㉿kali)-[~]
$ sudo systemctl restart ntpsec

(kali㉿kali)-[~]
$ sudo systemctl status ntpsec
● ntpsec.service - Network Time Service
  Loaded: loaded (/usr/lib/systemd/system/ntpsec.service; disabled; preset: disable)
  Active: active (running) since Wed 2024-10-02 01:54:30 EDT; 10s ago
    Invocation: 6aa96b1afc804f80b689f9776ac8a2ea
      Docs: man:ntpd(8)
   Process: 6102 ExecStart=/usr/libexec/ntpsec/ntp-systemd-wrapper (code=exited, stat=>
   Main PID: 6105 (ntpd)
     Tasks: 1 (limit: 2221)
    Memory: 13.5M (peak: 14M)
      CPU: 74ms
     CGroup: /system.slice/ntpsec.service
             └─6105 /usr/sbin/ntpd -p /run/ntpd.pid -c /etc/ntpsec/ntp.conf -g -N -u ntp

Oct 02 01:54:34 kali ntpd[6105]: DNS: dns_check: DNS error: -3, Temporary failure in name resolution
Oct 02 01:54:34 kali ntpd[6105]: DNS: dns_take_status: 3.debian.pool.ntp.org⇒temp, 3
Oct 02 01:54:39 kali ntpd[6105]: DNS: dns_probe: 0.debian.pool.ntp.org, cast_flags:8, 8
Oct 02 01:54:39 kali ntpd[6105]: DNS: dns_check: processing 0.debian.pool.ntp.org, 8,
```

Shell Scripting and Security

Basic Shell scripting on Linux system

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

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

- touch filename.sh

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



Step 02: edit file. Using

- nano filename.sh

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

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

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

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

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

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

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

```
# write the report
# cat > bandit.txt << EOF
#   echo "system Report -$DATE"
#   echo "-----"
#   echo "uptime: $UPTIME"
#   echo "Free Memory: $FREE_MEMORY"
#   echo "Disk Usage:"
#   echo "$DISK_USAGE"
# } > "$REPORT_FILE"
#
# echo "Report generated: $REPORT_FILE"
```

Step 03: change to execute mode. Using

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

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

step 04: Run the script. Using

- `./daily_reports.sh`

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

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

Step 01: You need to create backup directory. Using

- mkdir directory name

The screenshot shows a terminal session on a Kali Linux system. The user has run the command `mkdir backup`, which has created a new directory named "backup" in their home directory. A red arrow points from the text "step 01" in the previous section to this terminal command.

```
(kali㉿kali)-[~]
$ mkdir backup

(kali㉿kali)-[~]
$ ls
a.out      cod.sh      Downloads  myscript.sh  snp.sh      Templates  Videos
backup    Desktop     ii          Pictures    soslab3.c  TEST
bandit.txt Documents  Music      Public      system_reports TheFatRat
```

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

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

The screenshot shows a terminal session where the user is editing a script named `backup_document.sh` using the `nano` text editor. The script contains shell commands to set up a source directory, destination directory, create the destination if it doesn't exist, get the current date, create a backup filename, create the backup using `tar`, and output a message. A red arrow points from the text "step 02" in the previous section to this terminal session.

```
File  Actions  Edit  View  Help
GNU nano 8.1                                backup_document.sh
#!/bin/bash

SOURCE_DIR="/home/kali/documents"
DEST_DIR="/home/kali/backup/documents"

# Create the destination directory if it doesn't exist
mkdir -p "$DEST_DIR"

# Get the current date
DATE=$(date +"%Y-%m-%d")

# Create a backup filename
BACKUP_FILE="backup_$DATE.tar.gz"

# Create the backup
tar -czf "$DEST_DIR/$BACKUP_FILE" -C "$SOURCE_DIR" .

# Output a message
echo "Backup created: $DEST_DIR/$BACKUP_FILE"
```

Step 03: change file mode to execute format using,

- `chmod +x backup_document.sh`

```
[kali㉿kali] ~/backup]$ chmod +x backup_document.sh
```

step 04: execute and run the script, using

- `./backup_document.sh`

```
[kali㉿kali] ~/backup]$ ./backup_document.sh documents  
tar: /home/kali/documents: Cannot open: No such file or directory  
tar: Error is not recoverable: exiting now  
Backup created: /home/kali/backup/documents/backup_2024-10-02.tar.gz
```

SSH (Secure Shell)

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

Step 01: update your system, using

- **sudo apt update**

Step 02: install SSH Server, using

- **sudo apt install openssh-server**

```
(kali㉿kali)-[~]
$ sudo apt install openssh-server
The following packages were automatically installed and are no longer required:
  fonts-liberation2      libibverbs1      libusbmuxd6
  ibverbs-providers     libimobiledevice6  openjdk-17-jdk
  libassuan0             libjsoncpp25    openjdk-17-jdk-headless
  libavfilter9            libndctl6       openjdk-17-jre
  libboost-iostreams1.83.0 libplacebo338  openjdk-17-jre-headless
  libboost_thread1.83.0   libplist3       python3-diskcache
```

Step 03: Start or restart SSH Server, using

- **sudo systemctl start ssh**

```
(kali㉿kali)-[~]
$ sudo systemctl start ssh

(kali㉿kali)-[~]
$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset)
  Active: active (running) since Thu 2024-10-03 01:09:44 EDT; 20s ago
    Invocation: 82ff57bad0644494af8d010348d33209
      Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 6046 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 6047 (sshd)
     Tasks: 1 (limit: 2221)
    Memory: 1.9M (peak: 2.3M)
       CPU: 58ms
          ▲ . . . . .
```

iptables and ACLs

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

Step 01: Set the Iptables rules. Using

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

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

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

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

Step 02: Set the trusted ip address

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



Step 03: check the available iptables using,

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

```
[(kali㉿kali)-[~]]$ sudo iptables -L -n -v
Chain INPUT (policy DROP 2 packets, 1152 bytes)
pkts bytes target    prot opt in     out    source          destination
  0     0 ACCEPT     tcp   --  *      *      0.0.0.0/0        0.0.0.0/0      t
cp dpt:80
  0     0 ACCEPT     tcp   --  *      *      0.0.0.0/0        0.0.0.0/0      t
cp dpt:443
  0     0 ACCEPT     tcp   --  *      *      10.0.2.15       0.0.0.0/0      t
cp dpt:22
  0     0 ACCEPT     tcp   --  *      *      10.0.2.15       0.0.0.0/0      t
cp dpt:22
  0     0 ACCEPT     tcp   --  *      *      127.0.0.1       0.0.0.0/0      t
cp dpt:22
  0     0 ACCEPT     icmp  --  *      *      0.0.0.0/0       0.0.0.0/0      i
cmptype 8
  0     0 ACCEPT     tcp   --  *      *      192.168.1.30    0.0.0.0/0      t
cp dpt:9100

Chain FORWARD (policy DROP 0 packets, 0 bytes)
pkts bytes target    prot opt in     out    source          destination

Chain OUTPUT (policy ACCEPT 1 packets, 310 bytes)
pkts bytes target    prot opt in     out    source          destination
```

Best practices for security aspects of network interface configuration

There are some best practices in security aspects of network.

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

Disable and unsend Network interface

Disabling unused interfaces reduces the attack surface.

Step 01: List network interface, to using

- **Ip link show**

```
(kali㉿kali)-[~]
$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode DEFAULT group default qlen 1000
    link/ether 08:00:27:d2:26:79 brd ff:ff:ff:ff:ff:ff
```

Step 02: identify unused interfaces

Step 03: disable unused interfaces, using

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

step 04: Restart Networking, using

- **sudo systemctl restart networking**

Configuration firewall

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

Step 01: Install UFW, using

- **sudo apt install ufw**

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

Use 'sudo apt autoremove' to remove them.

Installing:
  ufw

Suggested packages:
  rsyslog
```

Step 02: Enable UFW, using

- **sudo ufw enable**

Step 03: Set default policies, using

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

Step 04: Allowing specific Services, using

- **sudo ufw allow ssh**

Step 05: Check status, using

- **sudo ufw status verbose**

Monitor network traffic

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

Step 01: Install tcpdump, using

- `sudo apt install tcpdump`

```
└─(kali㉿kali)-[~]
$ sudo apt install tcpdump
[sudo] password for kali:
tcpdump is already the newest version (4.99.5-1).
tcpdump set to manually installed.
The following packages were automatically installed and are no longer required:
  fonts-liberation2      libibverbs1      libusbmuxd6
  ibverbs-providers     libimobiledevice6  openjdk-17-jdk
  libassuan0            libjsoncpp25    openjdk-17-jdk-headless
  libavfilter9          libndctl6       openjdk-17-jre
  libboost-iostreams1.83.0 libplacebo338   openjdk-17-jre-headless
  libboost-thread1.83.0  libplist3        python3-diskcache
  libcephfs2           libpmem1        python3-mistune0
  libdaxctl1           libpostproc57   python3-pendulum
  libgeos3.12.1t64      librados2      python3-pytzdata
  libgfapi0             librdmacm1t64   rwho
```

Step 02: Capture Network Traffic, using

- `Sudo tcpdump -i <interface_name>`

Step 03: Save Captured data, using

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

step 04: Install a Monitoring tool, using

- `sudo apt install iftop`

Step 05: Run iftop, using

- `sudo iftop -i <interface_name>`

Disable IPv6 if Not Used

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

Step 01: Check Current IPv6 Status, using

- `Ip a | grep inet6`

Step 02: Disable IPv6, using

- `sudo nano /etc/sysctl.conf`

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

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

Step 03: Apply the Changes, using

- `sudo sysctl -p`

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

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

(kali㉿kali)-[~]
```

Step 04: Verify IPv6 is Disabled, using

- `Ip a | grep inet6`

Enable Logging and Auditing

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

Step 01: Install and Configure, using

- `sudo apt install rsyslog`

```
(kali㉿kali)-[~]
$ sudo apt install rsyslog
The following packages were automatically installed and are no longer required:
fonts-liberation2      libibverbs1      libusbmuxd6
ibverbs-providers      libimobiledevice6 openjdk-17-jdk
libassuan0              libjsoncpp25    openjdk-17-jdk-headless
libavfilter9             libndctl6      openjdk-17-jre
libboost-iostreams1.83.0 libplacebo338 openjdk-17-jre-headless
libboost-thread1.83.0   libplist3      python3-diskcache
libcephfs2              libpmem1       python3-mistune0
libdaxctl1              libpostproc57 python3-pendulum
libgeos3.12.1t64        librados2      python3-pytzdata
libgfapi0                librdmacm1t64 rwho
libgfrpc0                libre2-10      rwhod
libgwdx0                 libroco.3     libu2f-udev
libglusterfs0            libu2f-udev
Use 'sudo apt autoremove' to remove them.

Installing:
  rsyslog

Installing dependencies:
  libestr0  libfastjson4  liblognorm5
```

Step 02: Edit the configuration file, using

- `sudo nano /etc/rsyslog.conf`

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

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

Step 04: Restart Rsyslog, using

- `sudo systemctl restart rsyslog`