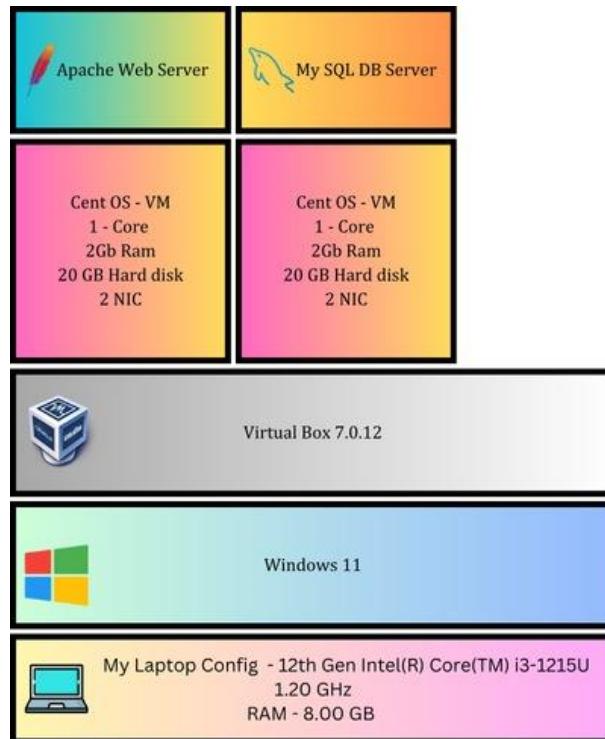


# VIRTUALIZATION PROJECT

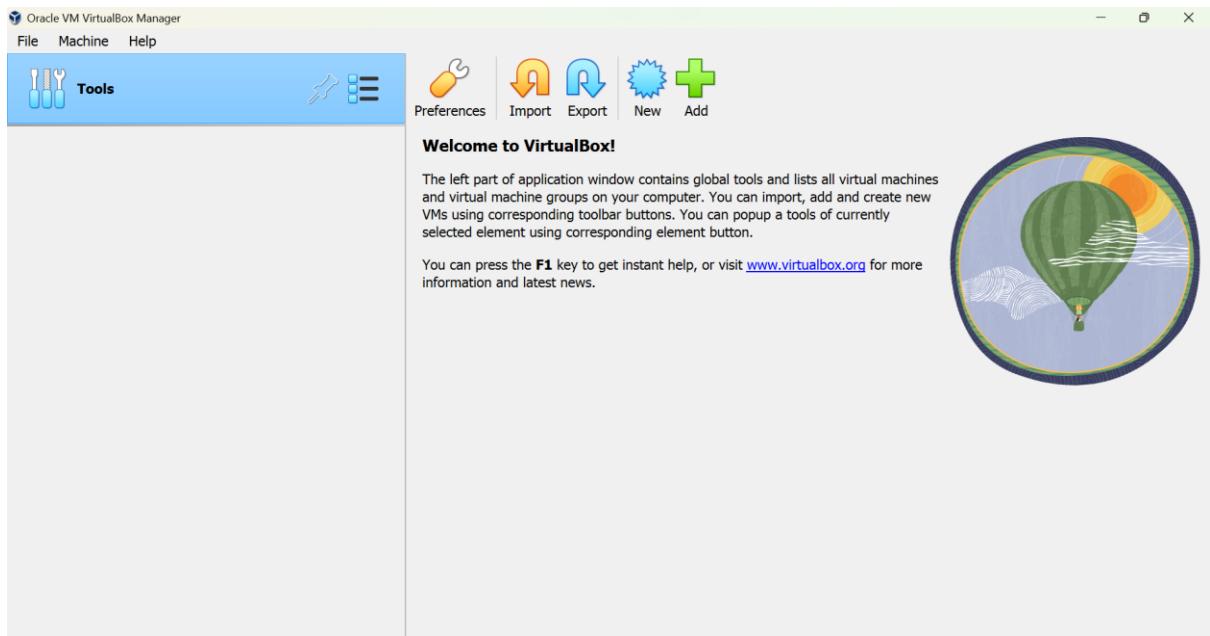
CREATED A DESIGN AS PER THE REQUIREMENT:



DOWNLOADED THE SOFTWARE AND OS:

WinSCP-6.3.5-Setup	13-09-2024 07:06	Application	11,340 KB
putty-64bit-0.81-installer	13-09-2024 07:06	Windows Installer Pa...	3,625 KB
CentOS-Stream-9-latest-x86_64-boot	13-09-2024 07:02	Disc Image File	9,80,992 KB
VirtualBox-7.0.12-159484-Win	13-09-2024 06:17	Application	1,08,298 KB

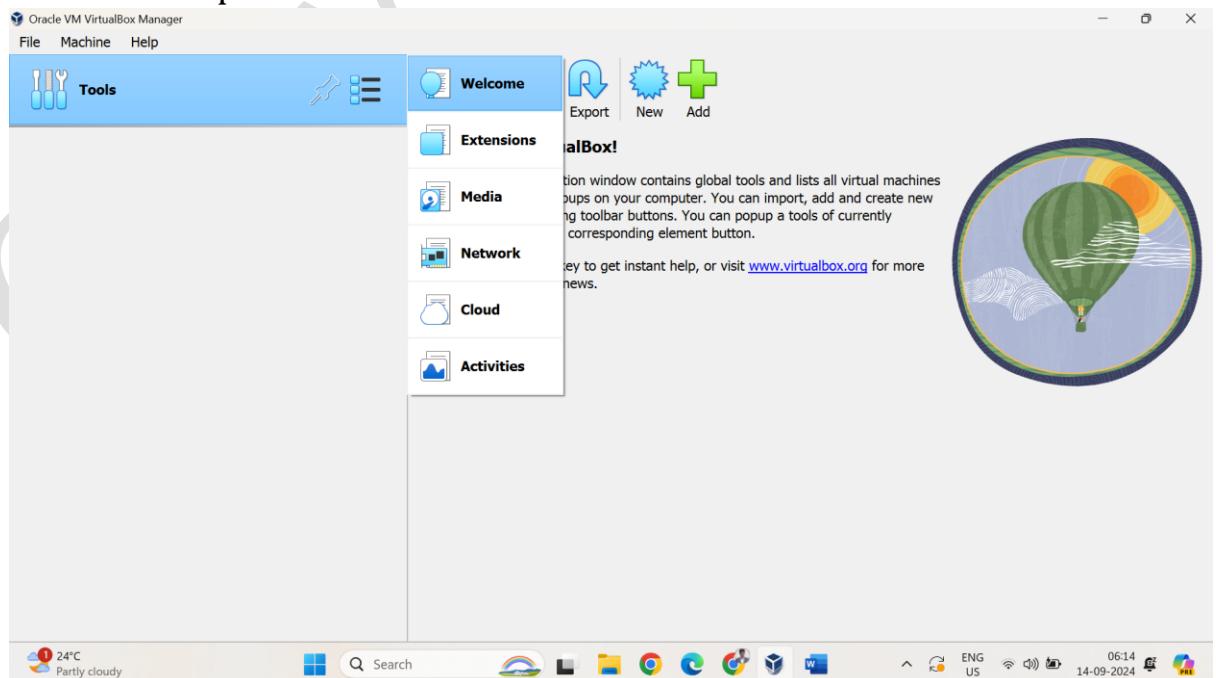
## INSTALLED THE VIRTUALBOX:

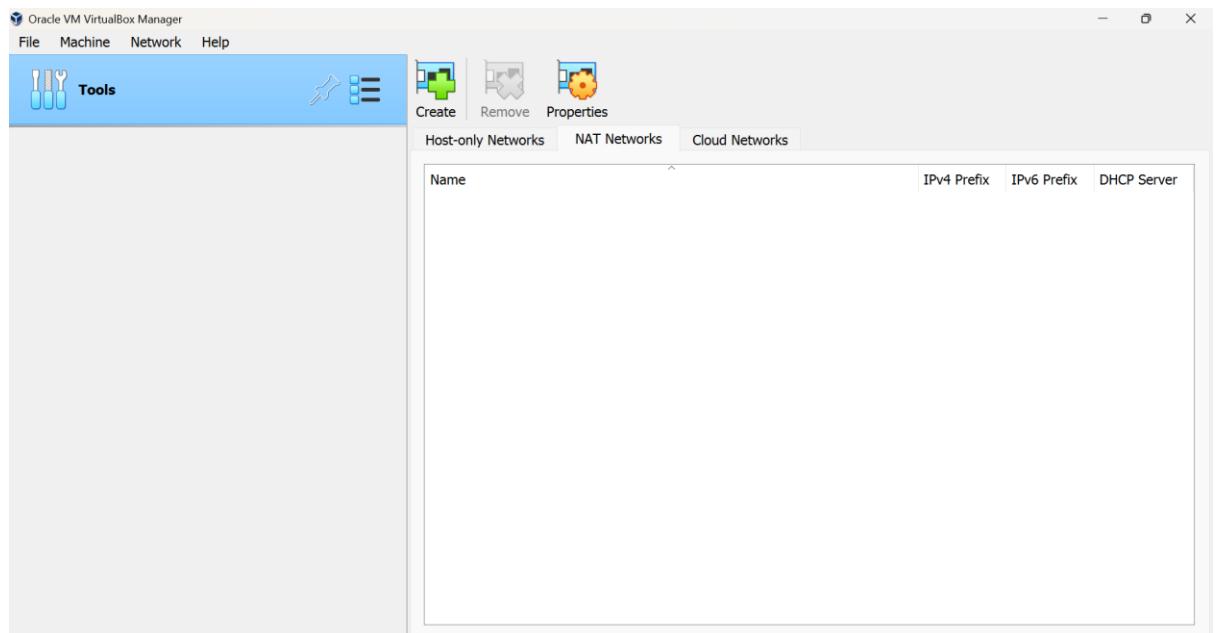


## SETUP THE NETWORK IN VIRTUALBOX:

1. HERE WE ARE GOING TO SETUP THE IP CONFIGURATION IN NAT NETWORK.

**Network Address Translation (NAT)** is a service that allows private IP networks to access the internet and cloud by translating private IP addresses to public IP addresses.





## CONFIGURE IP

This screenshot shows the configuration dialog for a Host-only Network named "NatNetwork". The table lists the network details: IPv4 Prefix is 10.0.2.0/24 and the DHCP server is Enabled. Below the table are two tabs: General Options and Port Forwarding. Under General Options, the Name is set to "DB Network", the IPv4 Prefix is 192.168.30.0/24, and the Enable DHCP checkbox is checked. The Enable IPv6 checkbox is unchecked. The IPv6 Prefix field is empty. The Advertise Default IPv6 Route checkbox is unchecked. At the bottom right are Apply and Reset buttons.

Name	IPv4 Prefix	IPv6 Prefix	DHCP Server
NatNetwork	10.0.2.0/24		Enabled

**General Options**

Name: DB Network

IPv4 Prefix: 192.168.30.0/24

Enable DHCP

Enable IPv6

IPv6 Prefix:

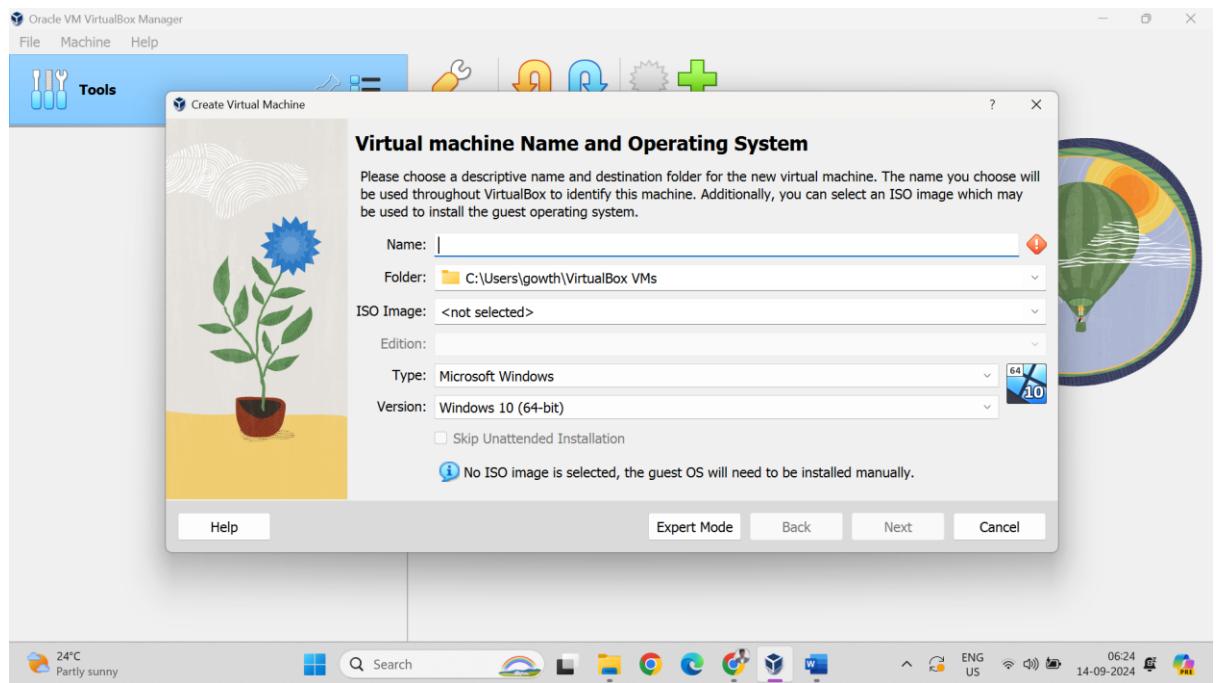
Advertise Default IPv6 Route

**Port Forwarding**

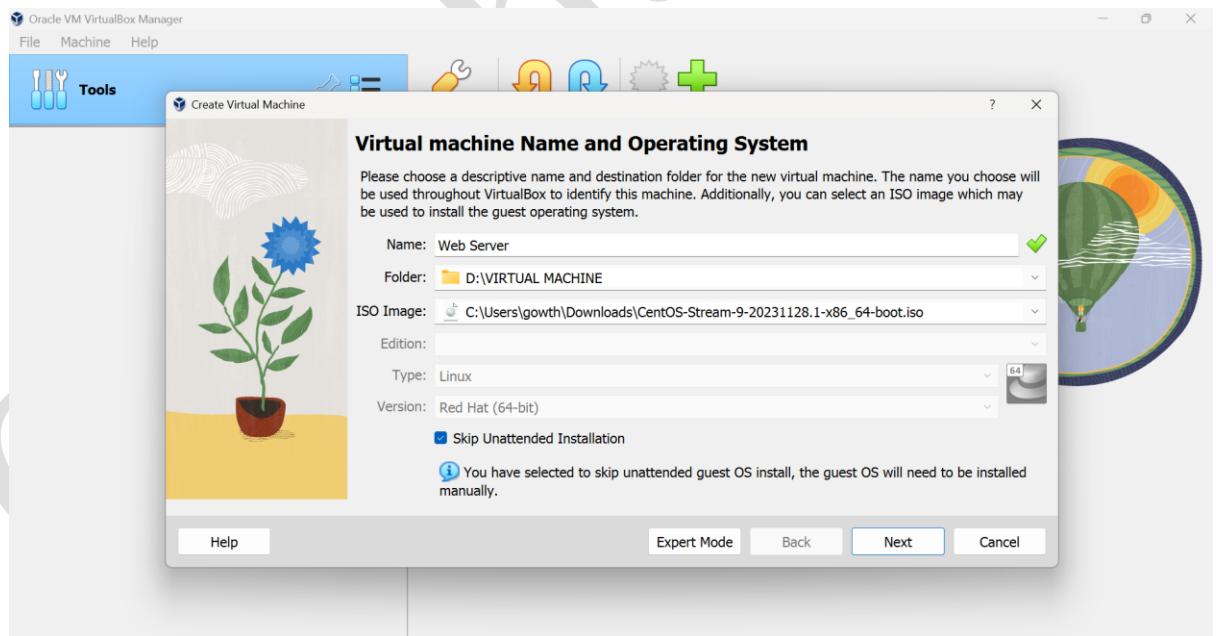
Apply      Reset

## CREATING VIRTUAL MACHINE:

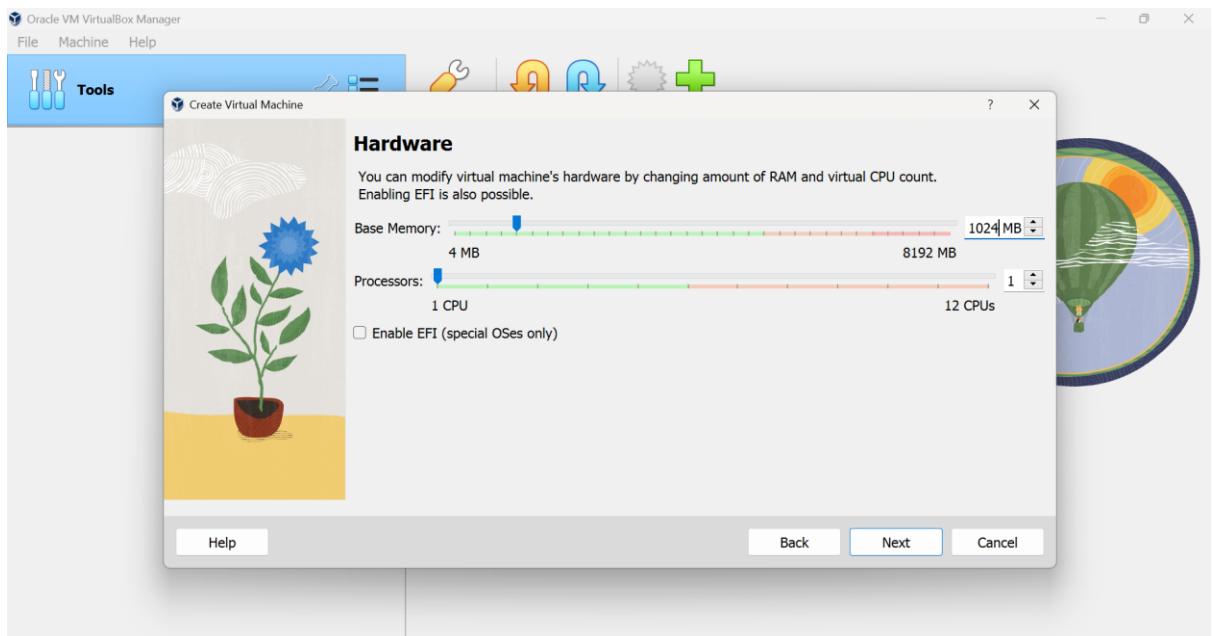
### 1. OPEN NEW OPTION IN THE HOME PAGE OF THE VIRTUAL BOX



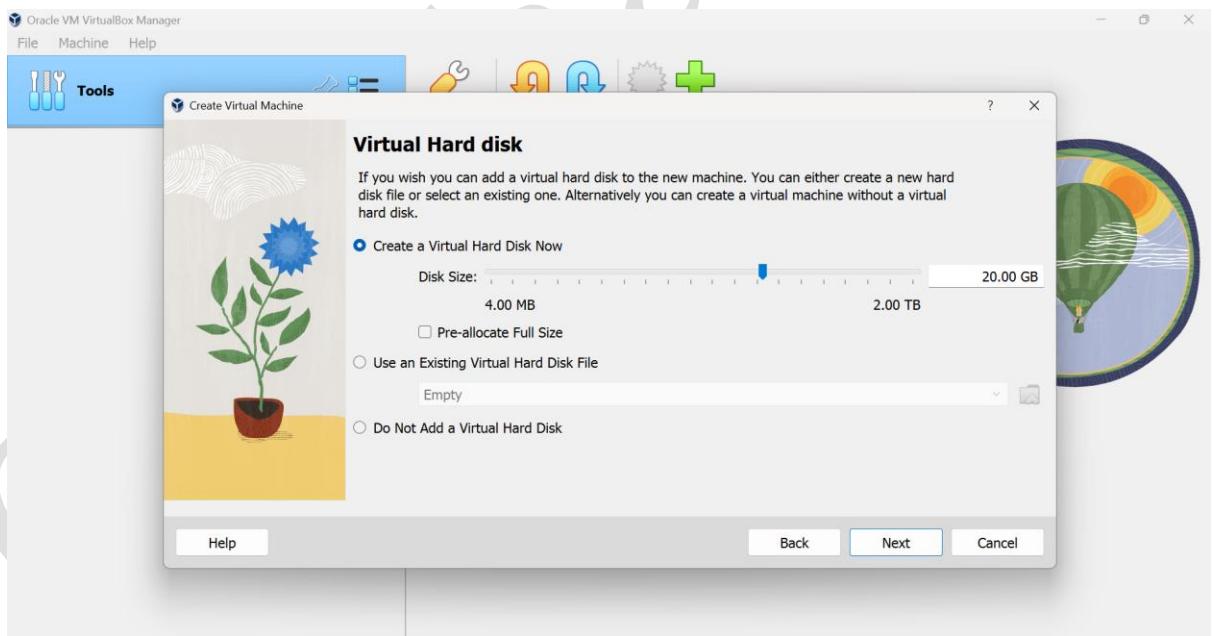
### 2. NAMING THE VM AND SETTING UP THE FILE LOCATION AS VM WILL BE STORED AS A FILE



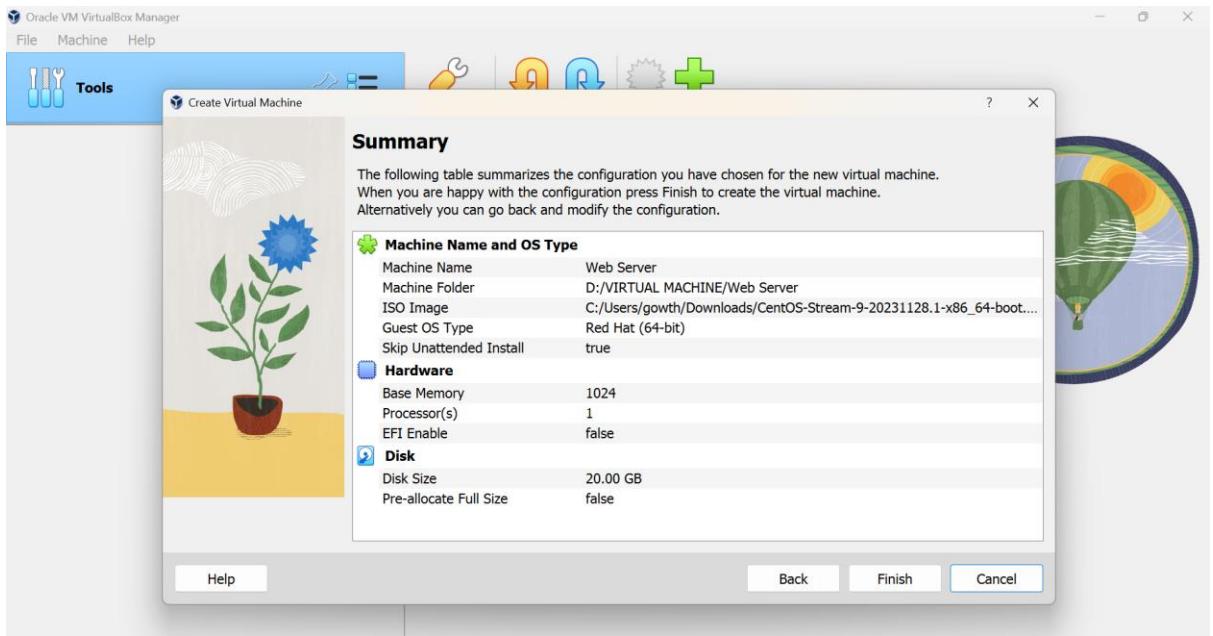
### 3. SETTING UP THE RAM AND THE PROCESSOR CORE AS MY SYSTEM IS 12 CORE I HAVE ALLOCATED THE 1 CORE FOR THE VM OF DB



### 4. GIVEN A DEFAULT OF 20GB STORAGE AND IT IS DYNAMIC NOT FIXED

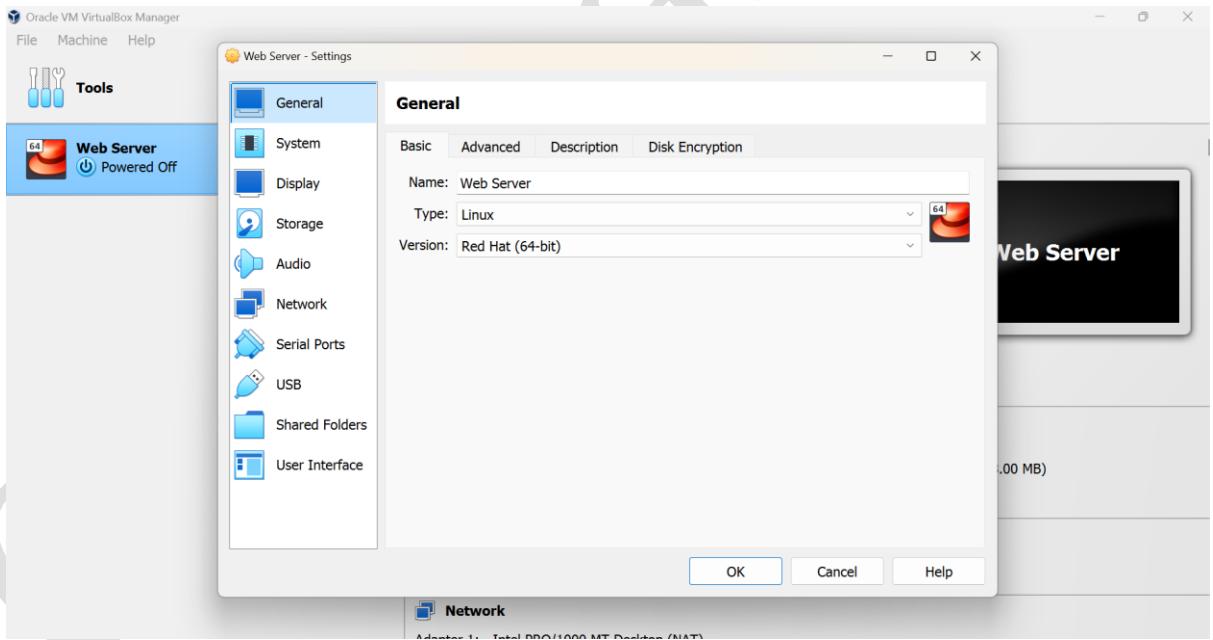


## 5. THIS IS SUMMARY OF MY FIRST VIRTUAL MACHINE

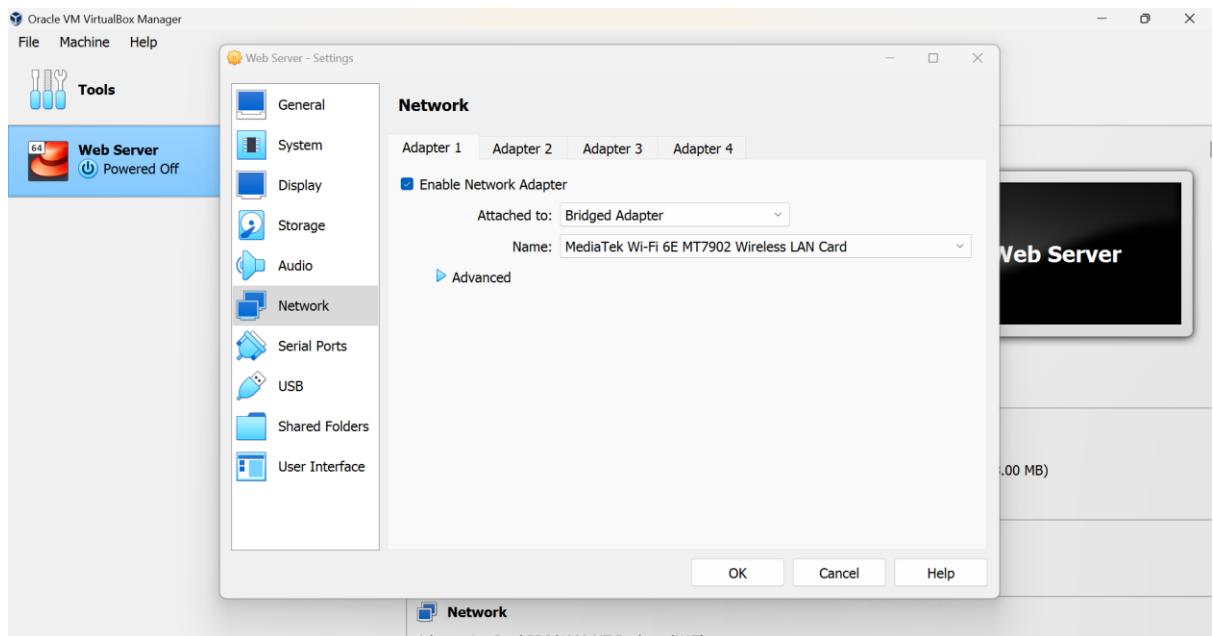


## ADD NETWORK TO OUR VIRTUAL MACHINE:

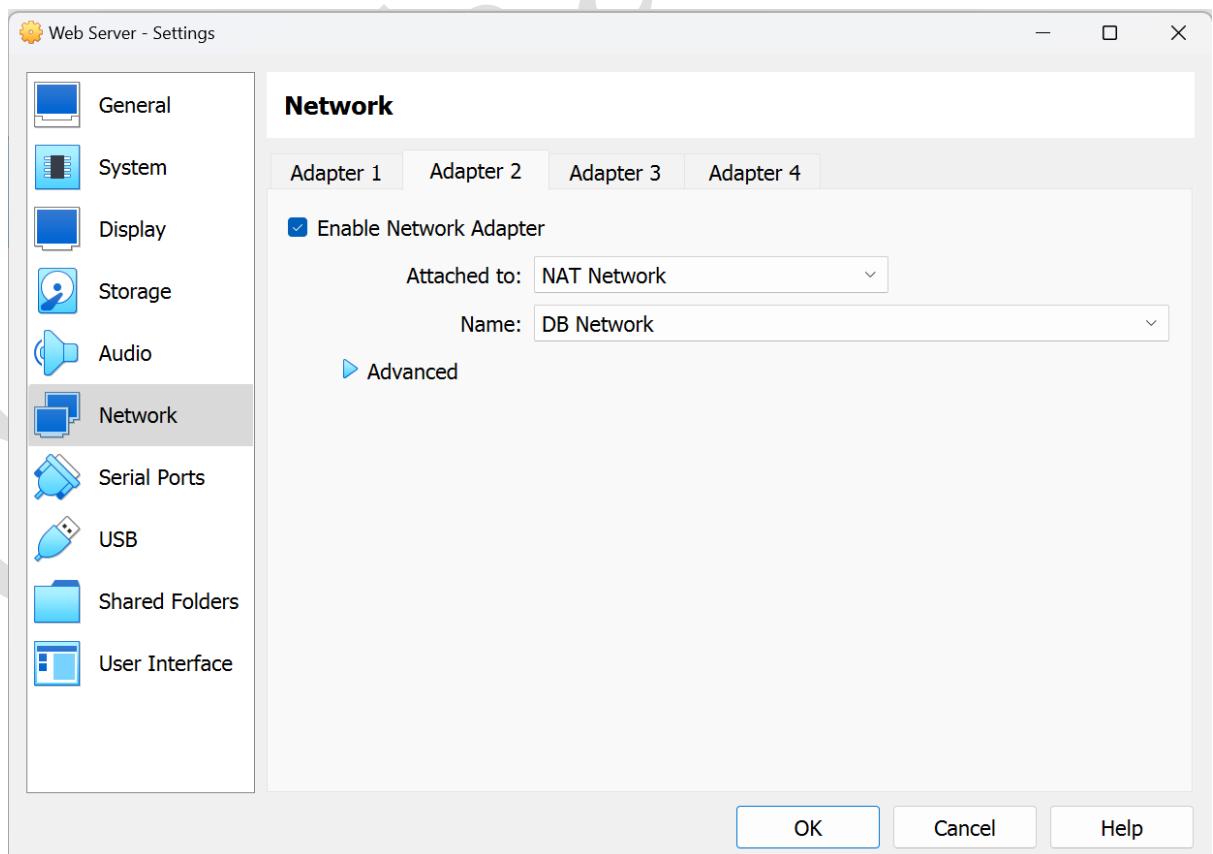
### 1. GIVE THE SETTING OPTION



2. I HAVE CREATED 2 NETWORK ADAPTER TO MY VIRTUAL MACHINE THE FIRST I WILL GIVE MY OWN NIC TO THE VM SO I



3. AND NEXT NETWORK I AM GOING TO SETUP THE NETWORK WHAT I HAVE CREATED THE NETWORK IS NAT NETWORK AND IT IS DB NETWORK WHICH WE HAVE CREATED ABOVE



## I HAVE CREATED MY VIRTUAL MACHINE

**General**

Name: Web Server  
Operating System: Red Hat (64-bit)

**System**

Base Memory: 1024 MB  
Boot Order: Floppy, Optical, Hard Disk  
Acceleration: Nested Paging, PAE/NX, KVM Paravirtualization

**Display**

Video Memory: 16 MB  
Graphics Controller: VMSVGA  
Remote Desktop Server: Disabled  
Recording: Disabled

**Storage**

Controller: IDE  
IDE Secondary Device 0: [Optical Drive] CentOS-Stream-9-20231128.1-x86\_64-boot.iso (958.00 MB)  
Controller: SATA  
SATA Port 0: Web Server.vdi (Normal, 20.00 GB)

**Audio**

Host Driver: Default  
Controller: ICH AC97

**Network**

Adapter 1: Intel PRO/1000 MT Desktop (Bridged Adapter, MediaTek Wi-Fi 6E MT7902 Wireless LAN Card)  
Adapter 2: Intel PRO/1000 MT Desktop (NAT Network, 'DB Network')

**USB**

USB Controller: OHCI, EHCI  
Device Filters: 0 (0 active)

**Shared folders**

None

**Description**

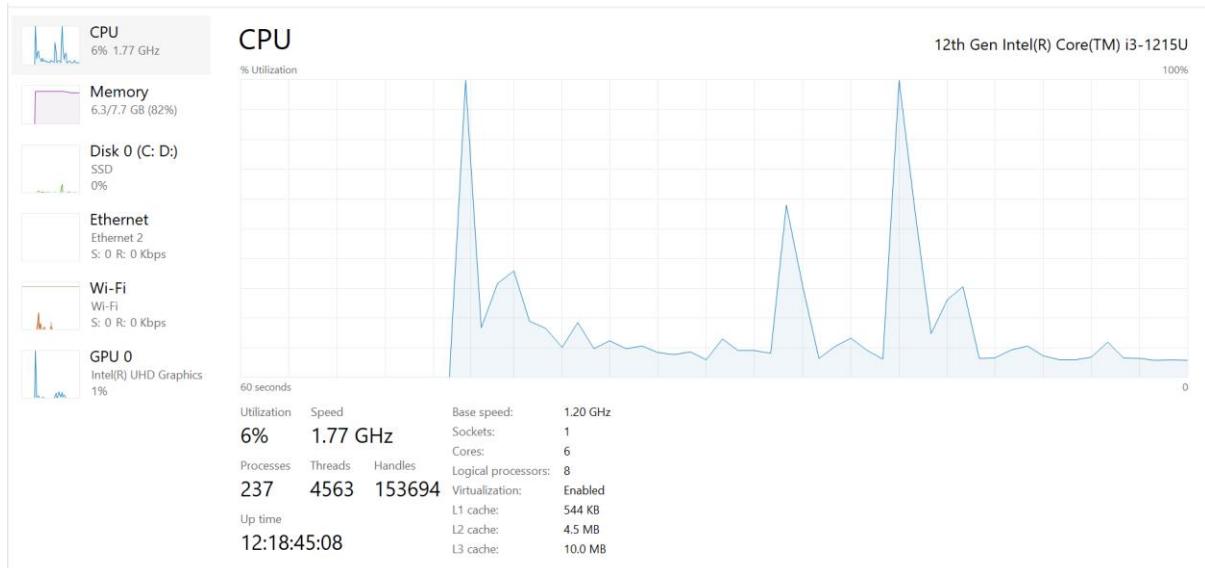
THIS IS MY VIRTUAL MACHINE USED FOR THE WEB SERVER

### WHY TWO NETWORK:

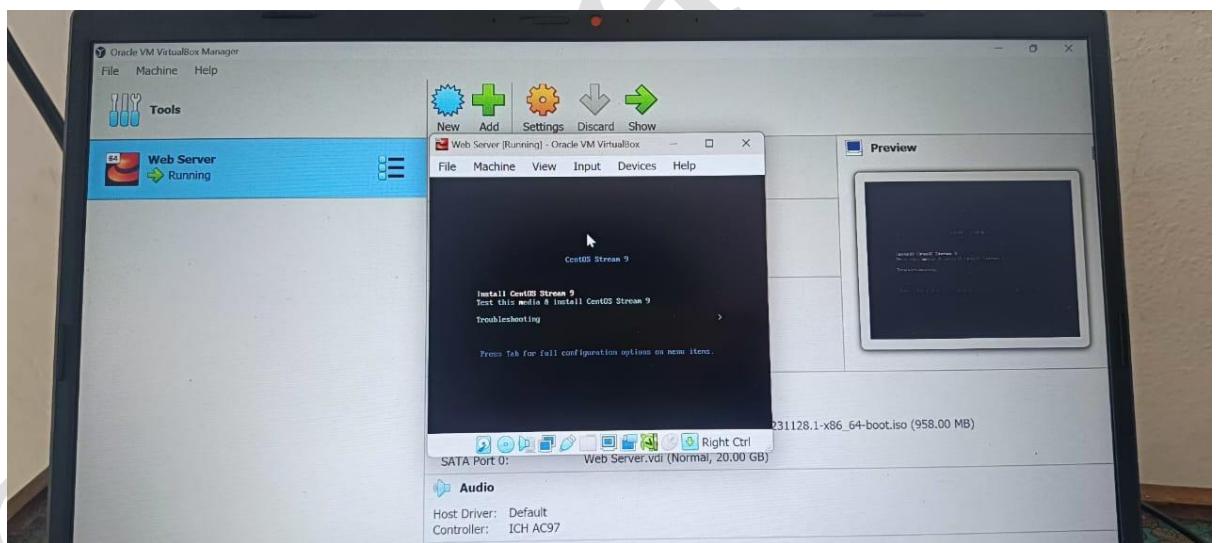
1. THE BRIDGED NETWORK IS USED BECAUSE WE NEED THE INTERNET ACCESS TO THE WEB SERVER VIRTUAL MACHINE SO WE USE THE ADAPTER 1 AS BRIDGED NETWORK SO THIS IS A PUBLIC NETWORK.
2. THE NAT NETWORK IS BECAUSE FOR SECURITY AS WE HAVE THE DATABASE THERE NO EXTERNAL PERSON SHOULD ACCES IT SO THEY ARE PRIVATE NETWORK.

NOW OUR VM IS READY FOR INSTALLATION WE ARE GOING TO INSTALL NOW

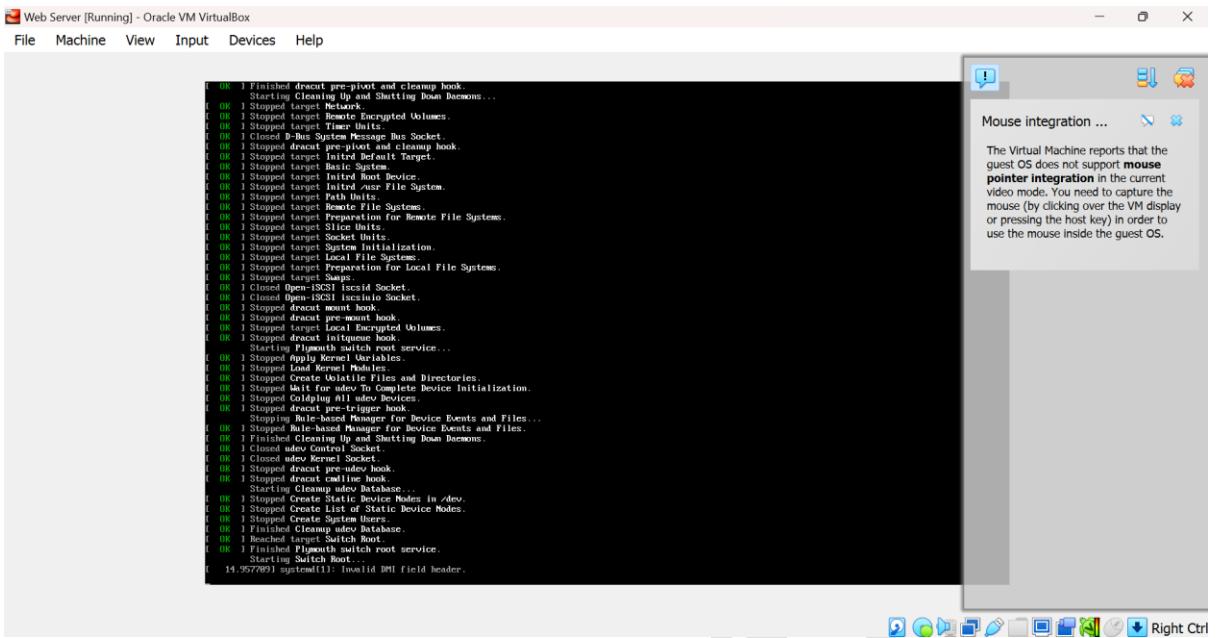
## 1. THIS IS MY CPU UTILIZATION BEFORE INSTALLING THE VM



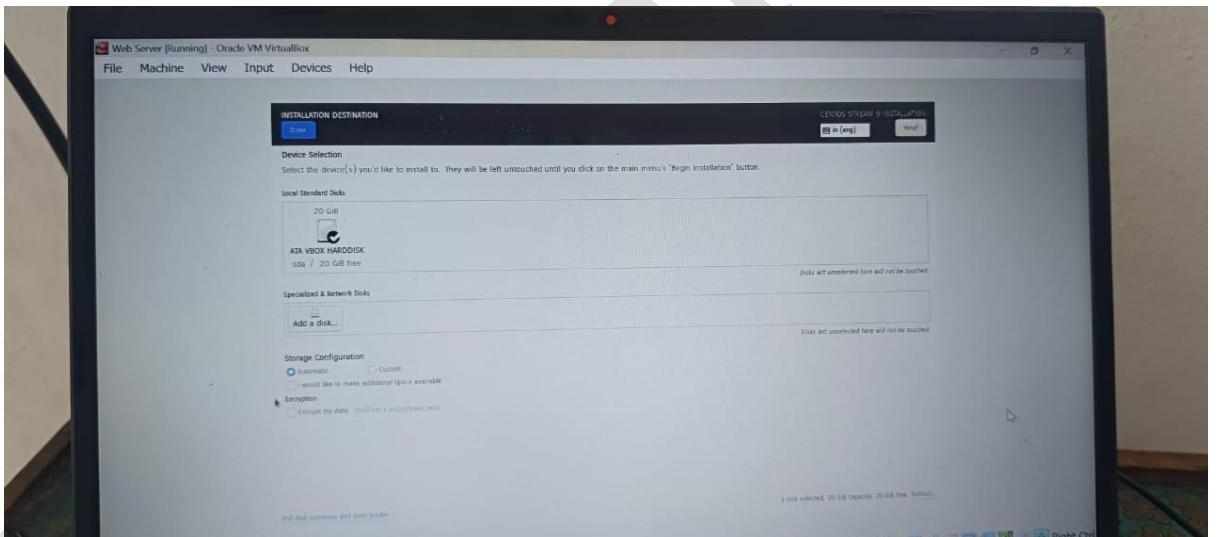
## 2. INSTALL THE OS FOR VM



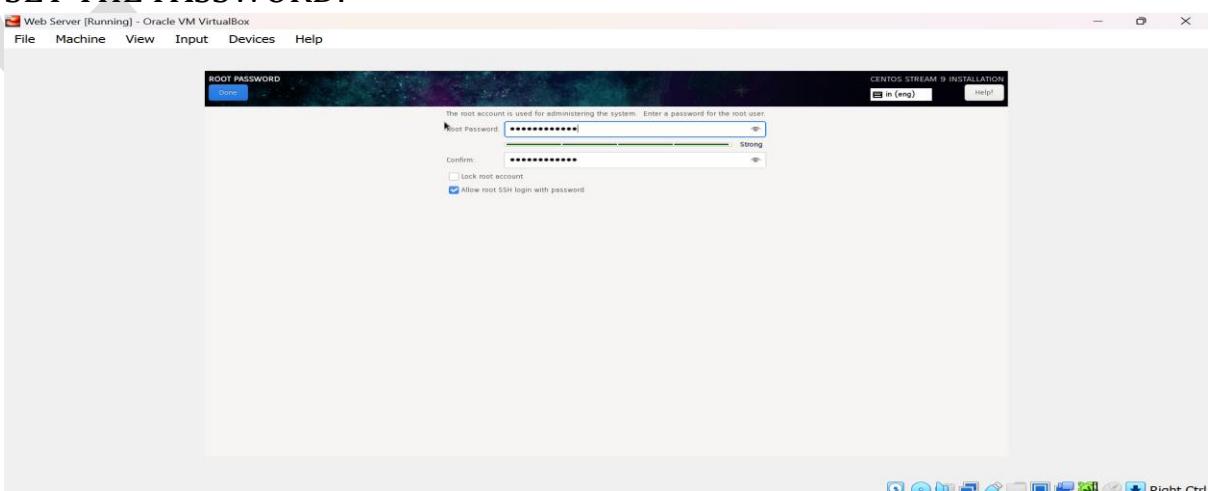
### 3. THE INSTALLATION IS IN PROCESS



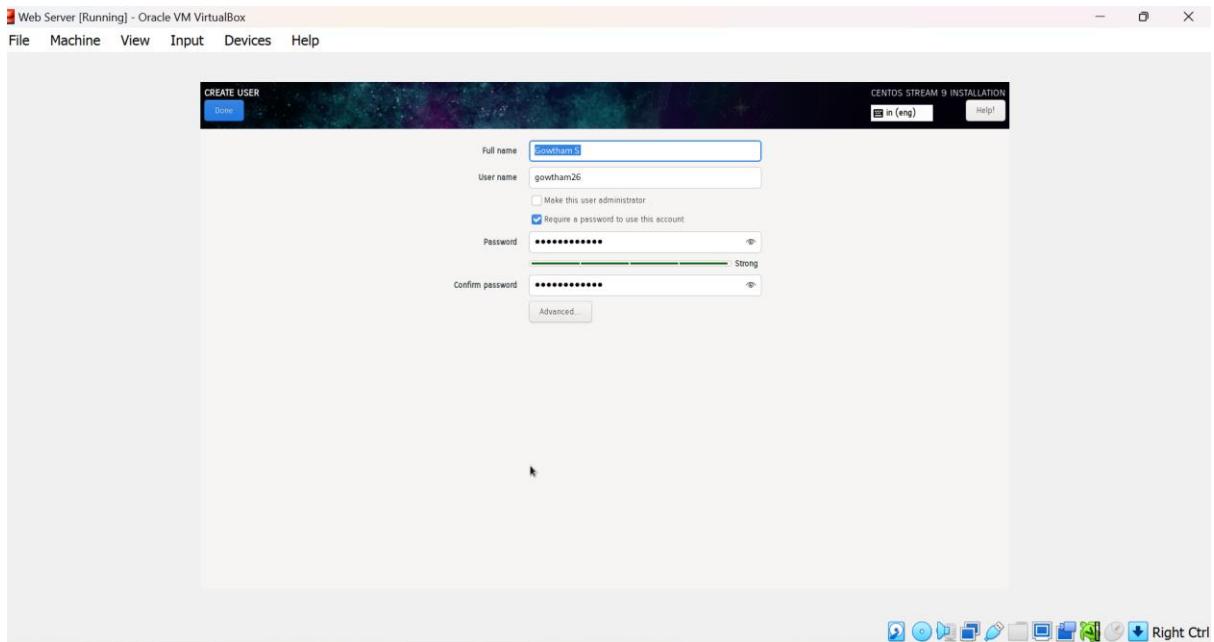
### 4. SETUP THE INSTALLATION DESTINATION:



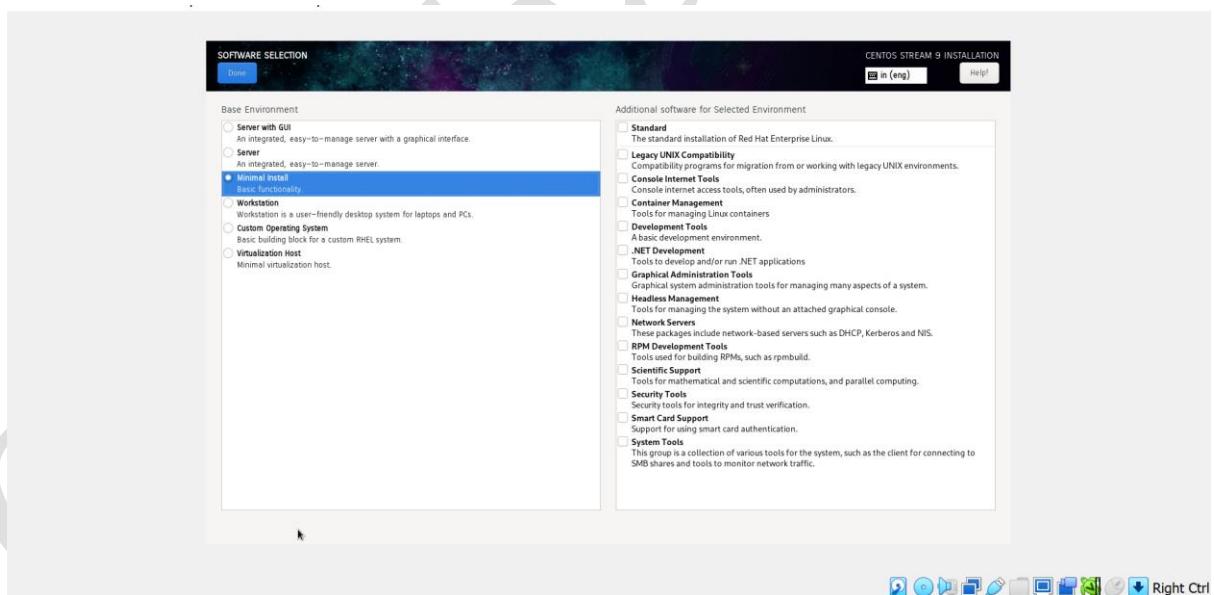
### 5. SET THE PASSWORD:



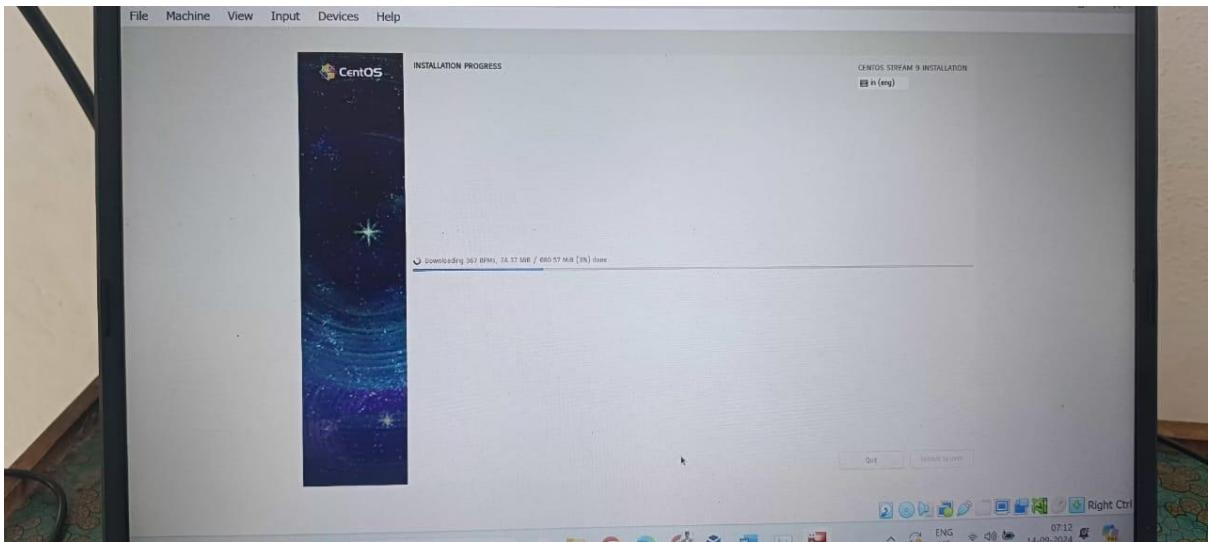
## 6. SET THE USER'S NAME:



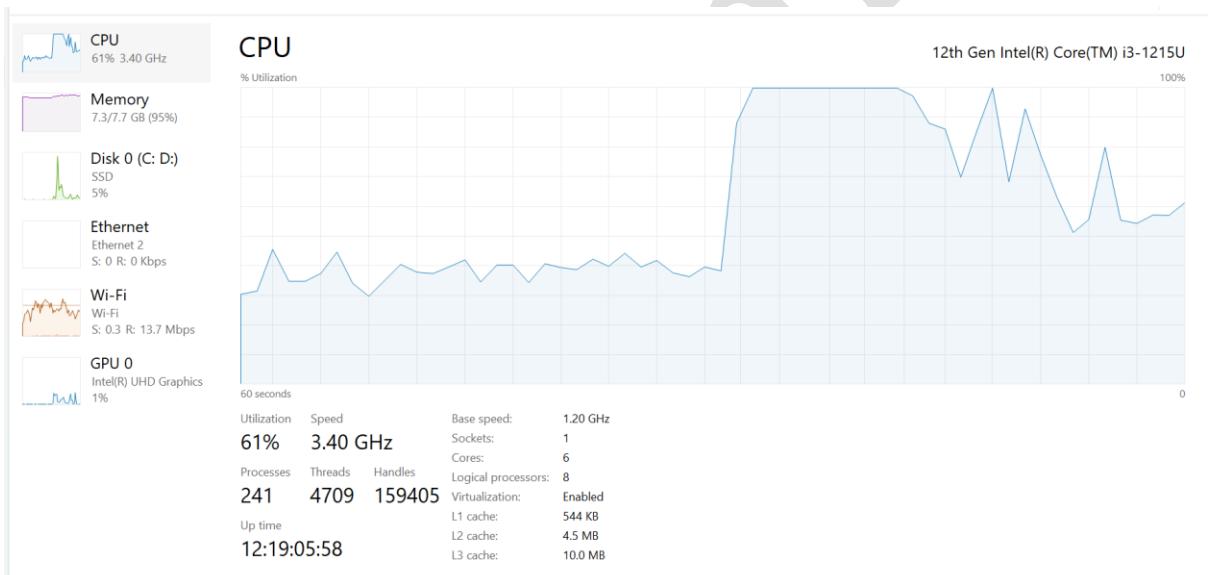
## 7. I HAVE GIVEN MINIMAL RATHER THAN GIVING THE GUI BECAUSE OF REMOVING UNWANTED SPACE ALLOCATION IN SOFTWARE INSTALLATION:



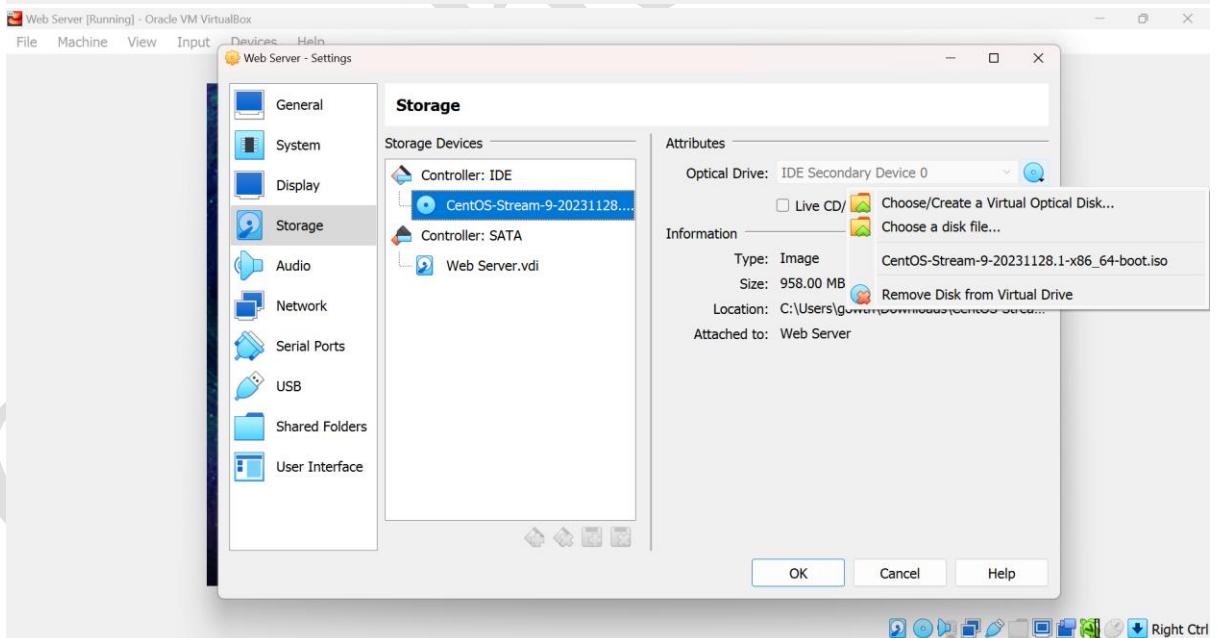
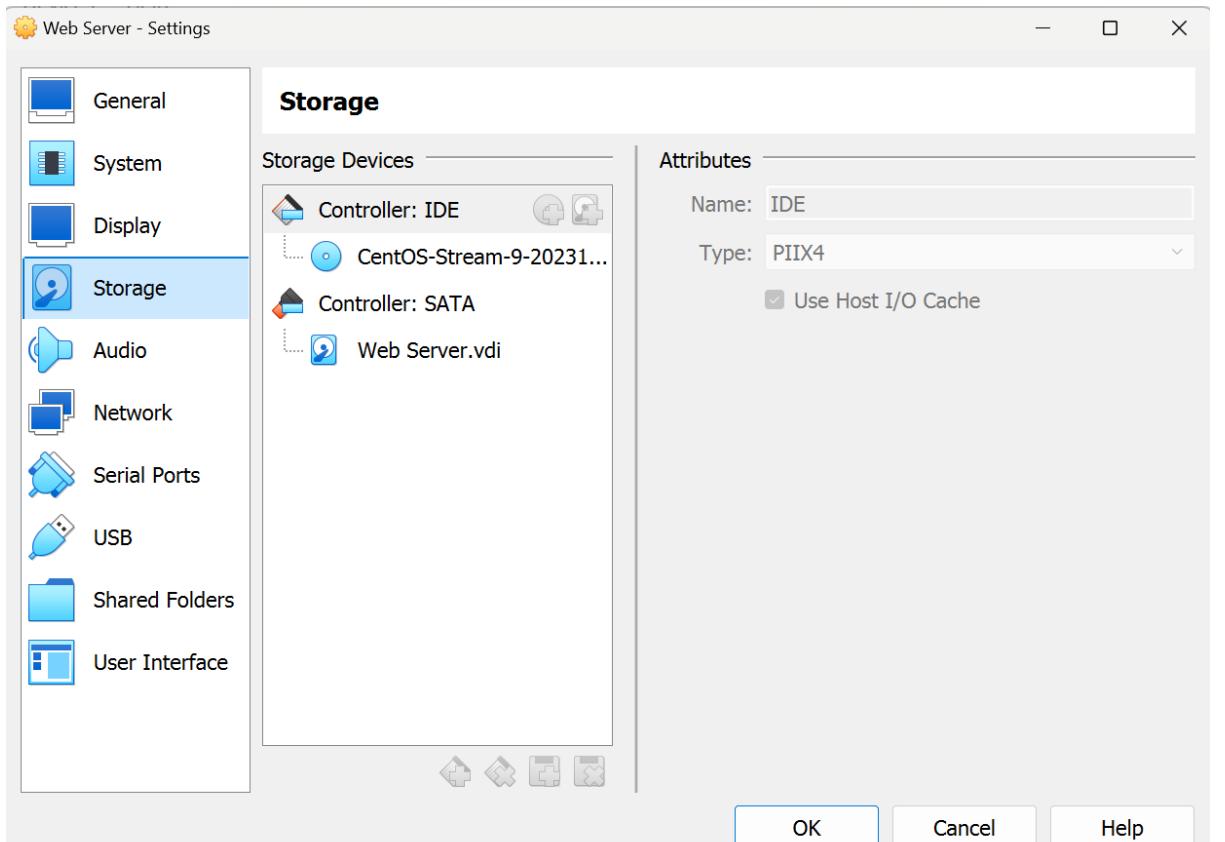
## 8. INSTALLATION IS IN PROGRESS



## 9. MY CPU UTILIZATION NOW:

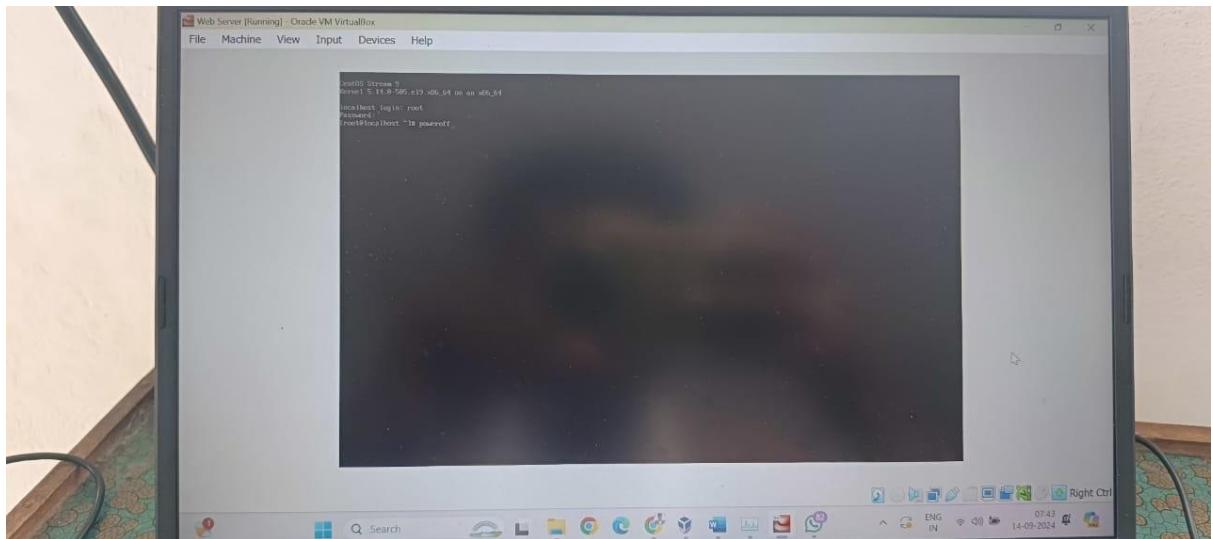


## 10. REMOVE THE ISO FILE FROM DISK

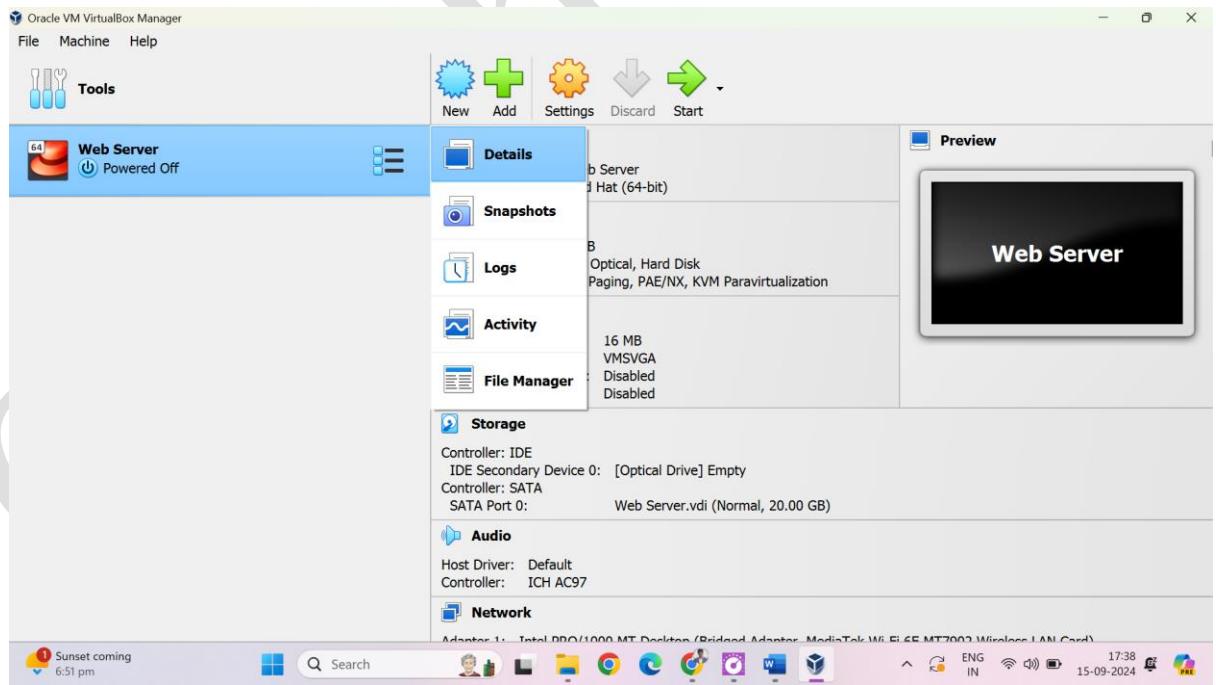


AFTER THIS RESET THE SYSTEM FROM MACHINE OPTION

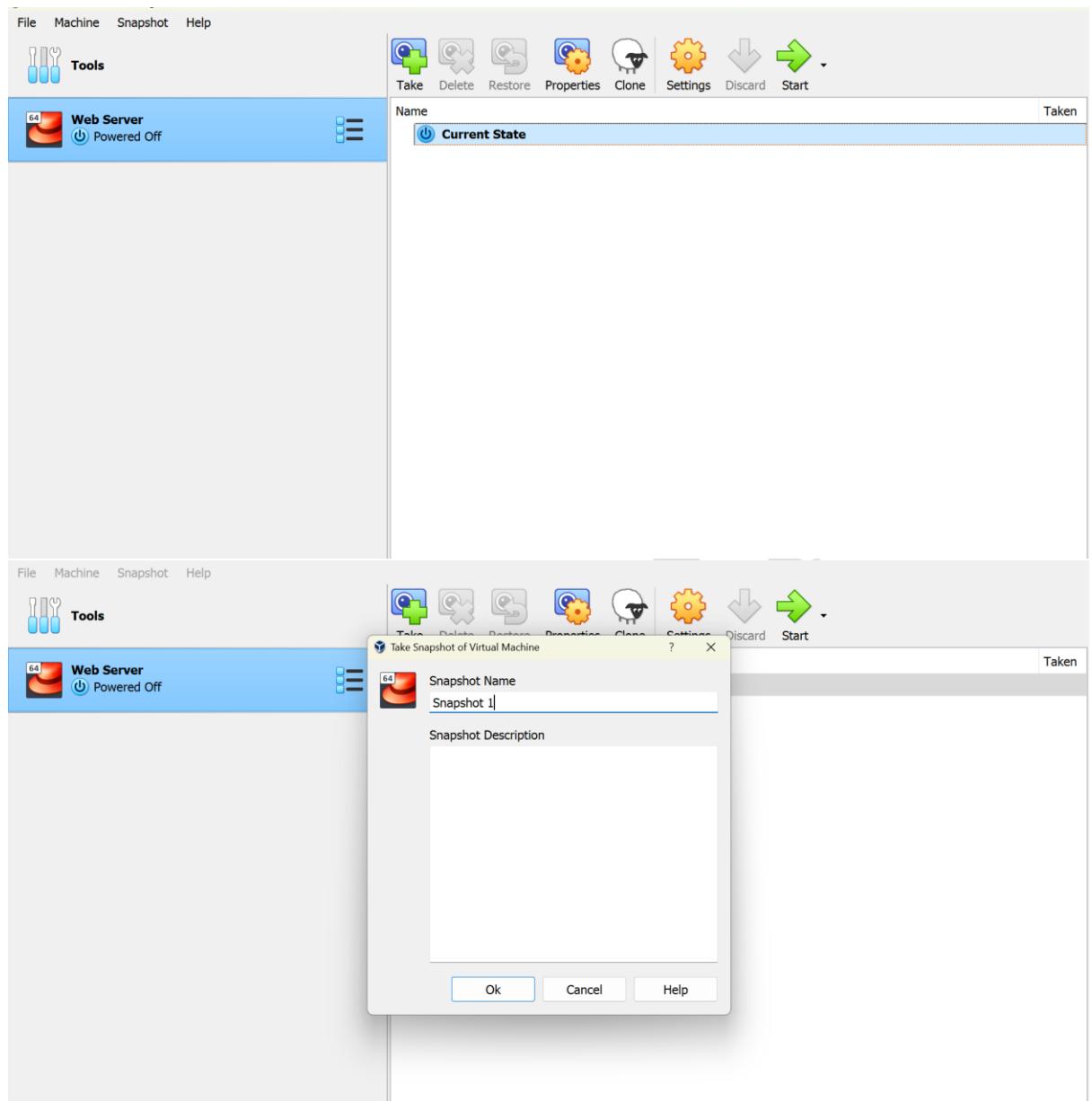
NOW OUR VM IS READY:



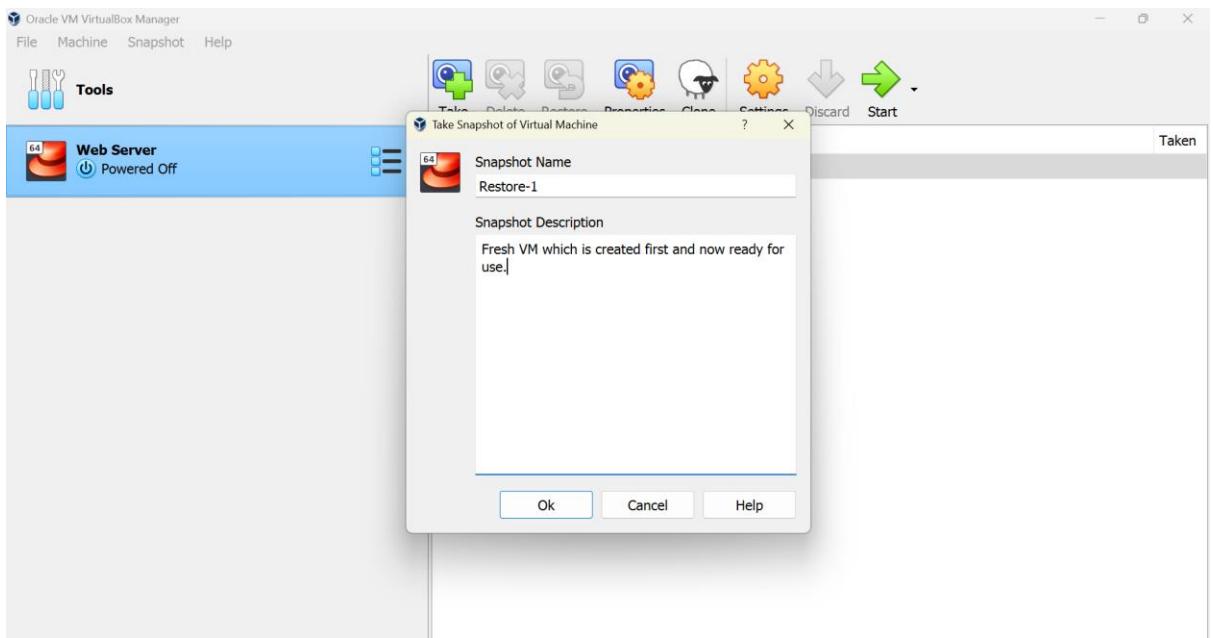
ADDING A SNAPSHOT / CLONE TO MY VIRTUAL MACHINE WHICH HELP TO RESTORE OUR MACHINE TO THE LAST STATE WHERE YOU WANT TO WHICH HELP WHEN WE MAKE ANY MISTAKE IN VM AND WE WANT TO GO TO THE OLD STATE CLICK THE SNAPSHOT



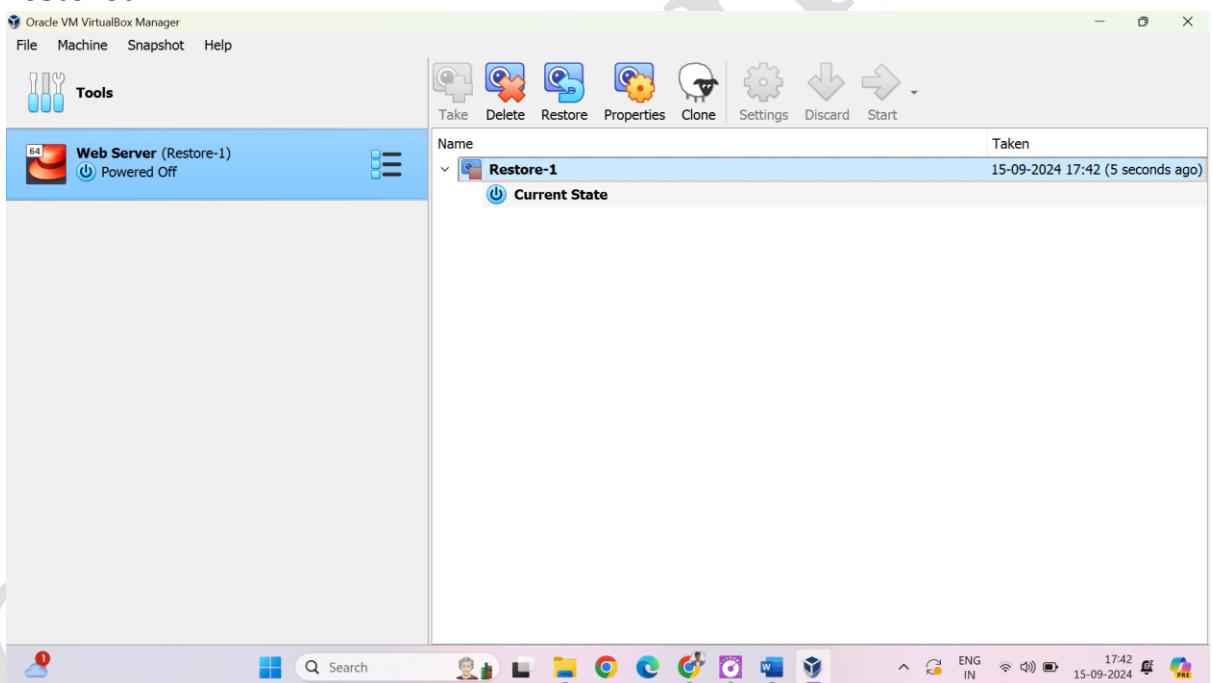
## CLICK THE CURRENT STATE AND GIVE THE TAKE OPTION



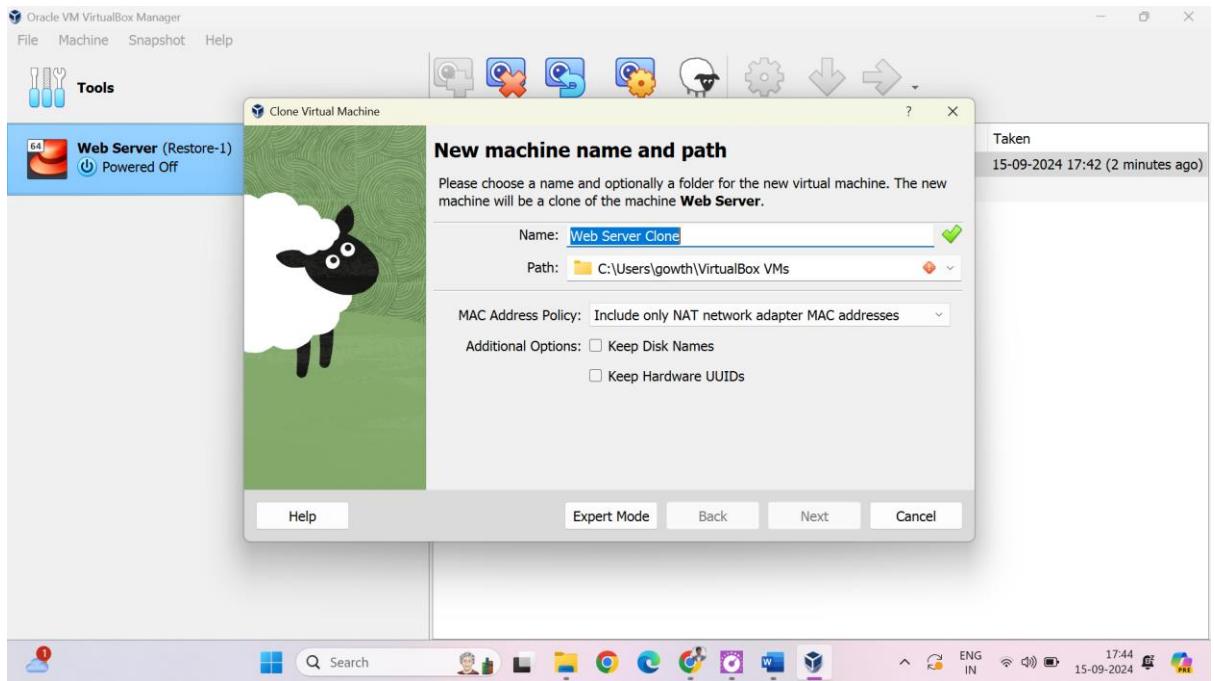
GO



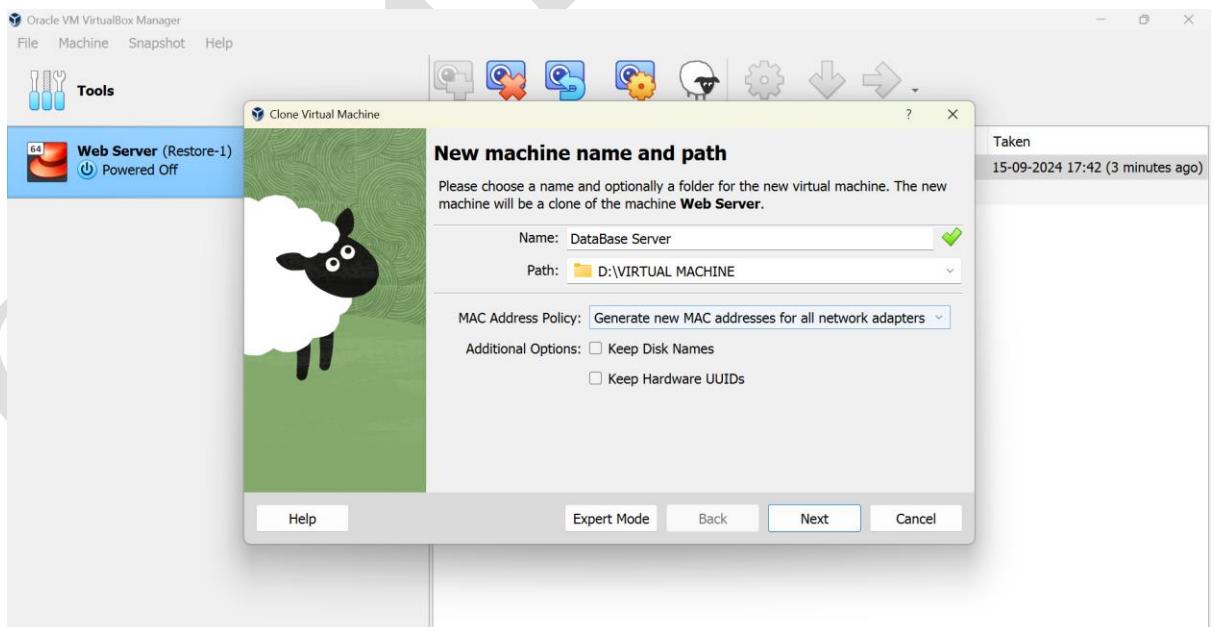
We can see the restore option above if we click that our VM will be restored



## CLONING OUR VIRTUAL MACHINE:

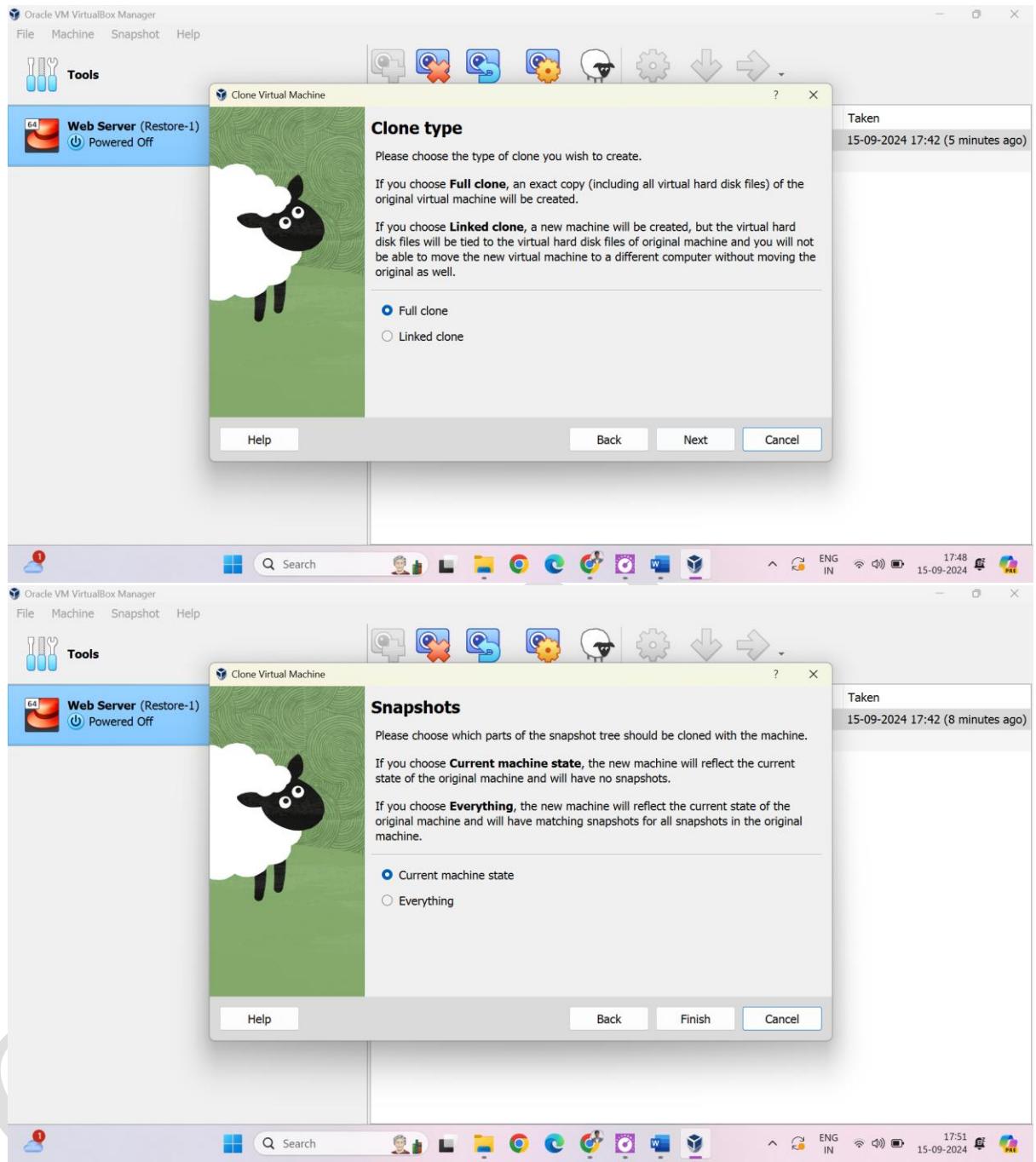


NOW WE CAN NAME THAT AND ALSO, WE CAN ASSIGN THE LOCATION FOR THE CLONE AND ALSO, WE CHANGE MAC ADDRESS THIS OPTION WHICH I HAVE GIVEN WILL CREATE THE NEW MAC ADDRESS FOR ALL THE VM.

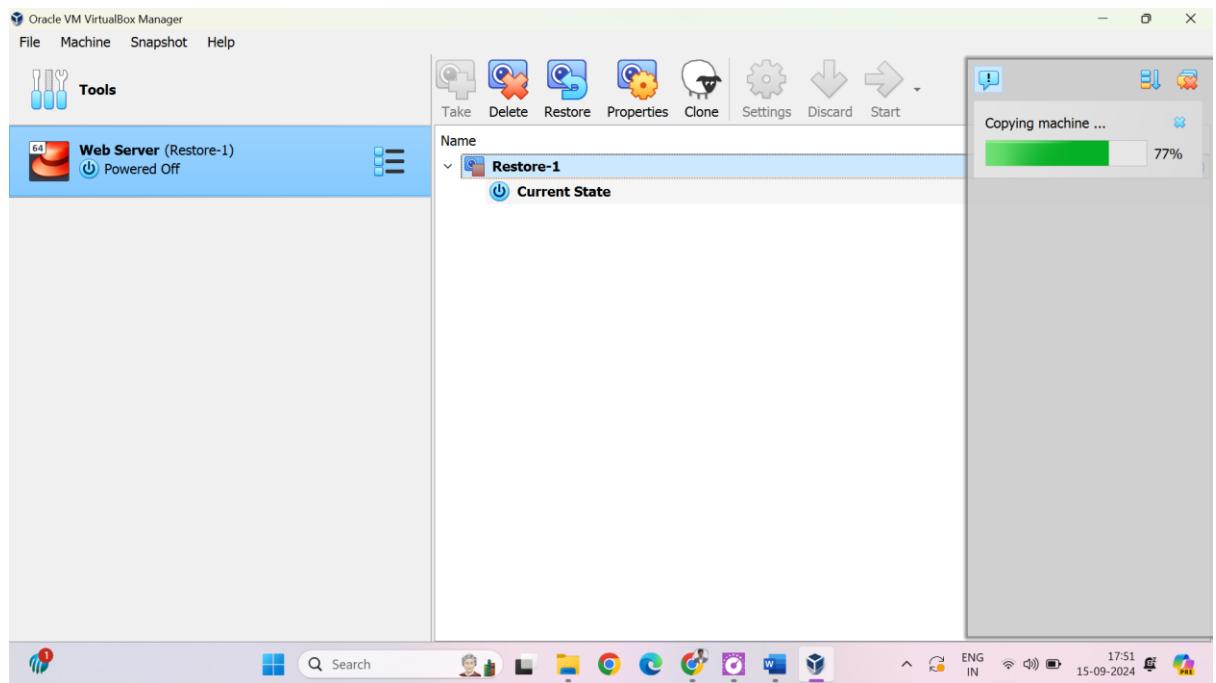


FOR MY CLONED VM I AM GOING TO ASSIGN A FULL CLONE OPTION WHICH WILL CREATE A SEPARATE FILE FOR THE VIRTUAL

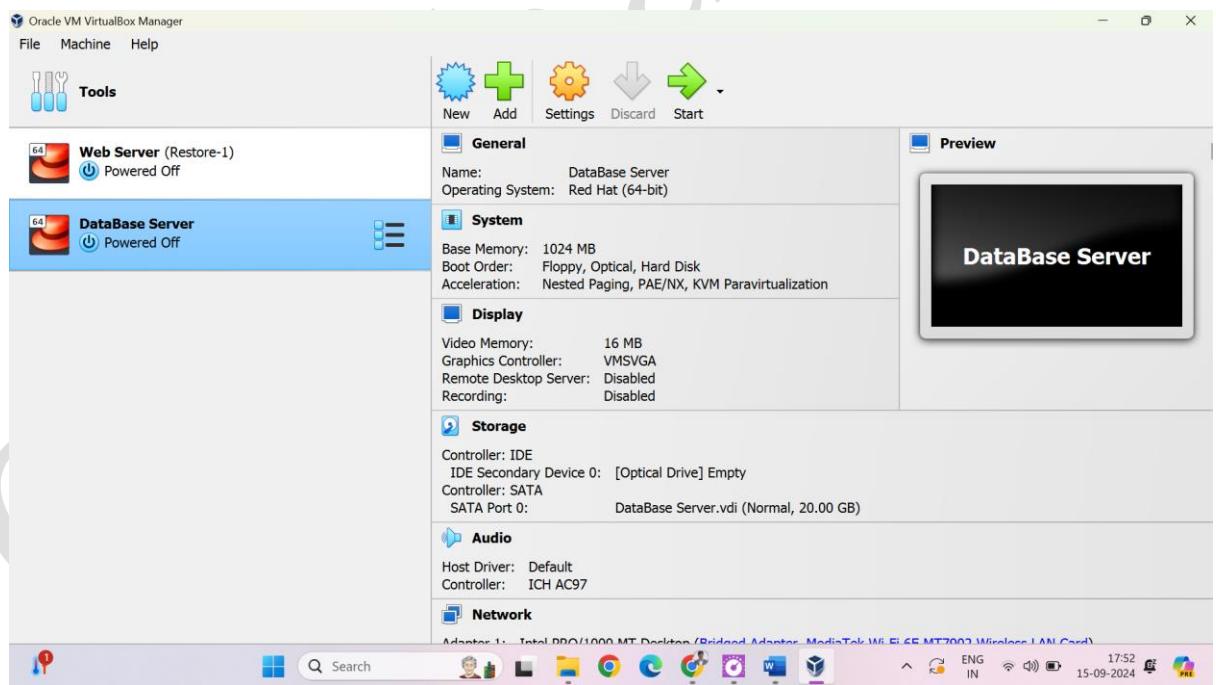
## MACHINE. IF WE GIVE LINKED CLONE THEN OUR FILES WILL BE LINKED FOR NEW VM FROM OLD VM



## MACHINE IS BEING COPIED



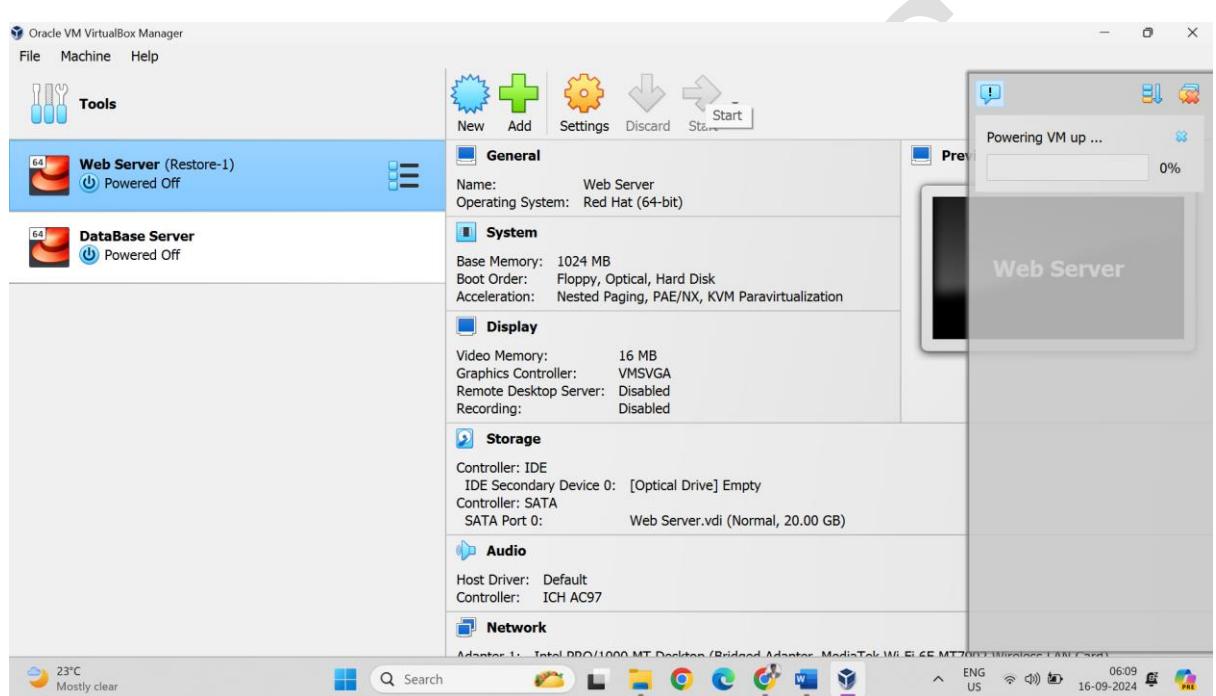
NEW CLONED MACHINE IS SUCCESSFULLY CREATED AS SAME AS THE OLD MACHINE



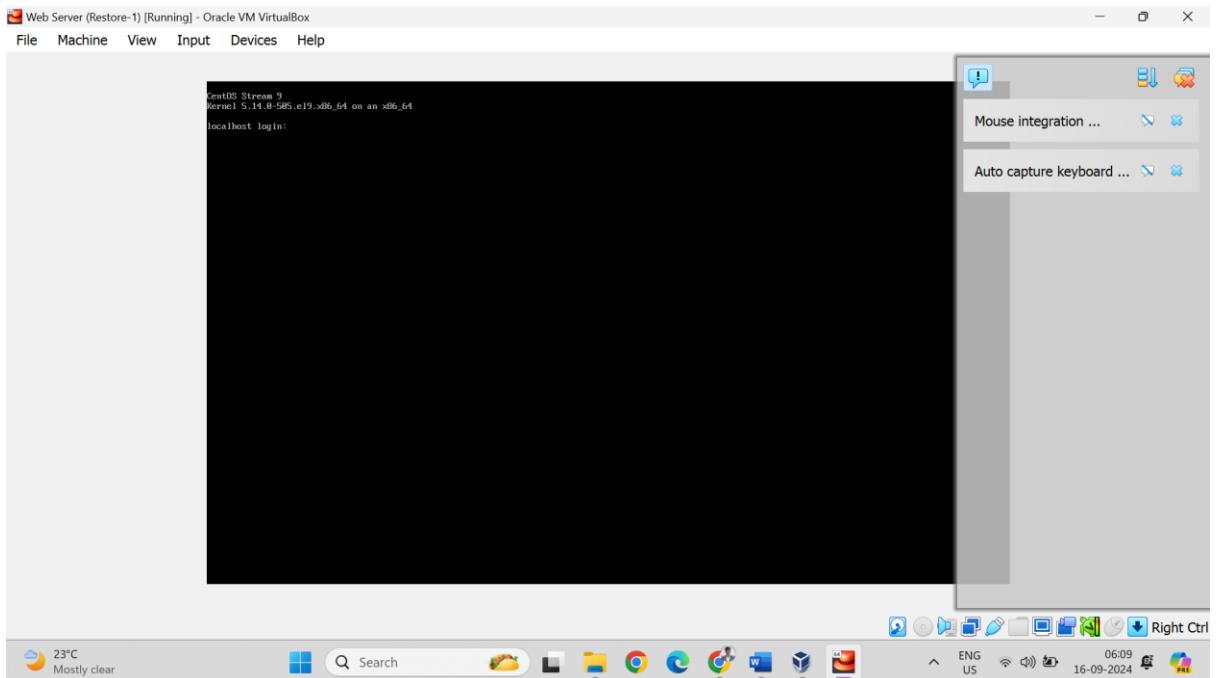
NOW WE ARE GOING TO CONFIGURE THE WEBSER VM WITH SSH AND FTP

WE WILL USE PUTTY SOFTWARE FOR CONFIGURING THE SSH  
WE WILL USE WinSCP SOFTWARE FOR CONFIGURING THE FTP

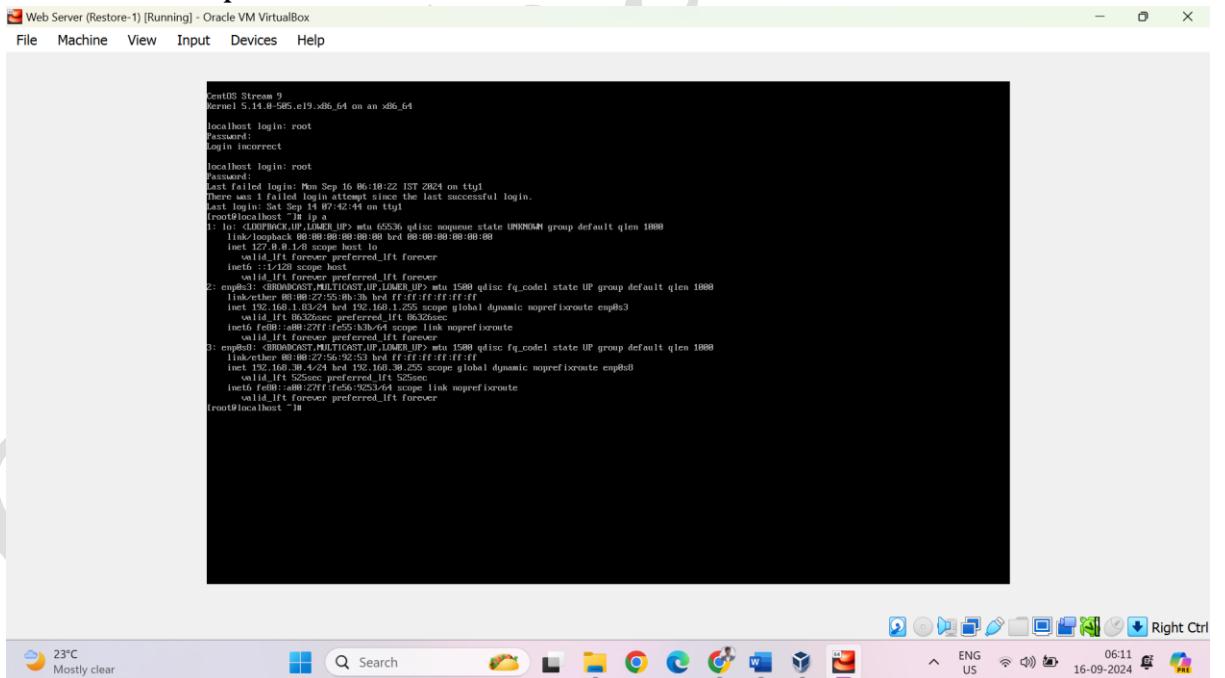
NOW WE CAN START OUR WEBSERVER VM TO CONFIGURE SSH AND FTP



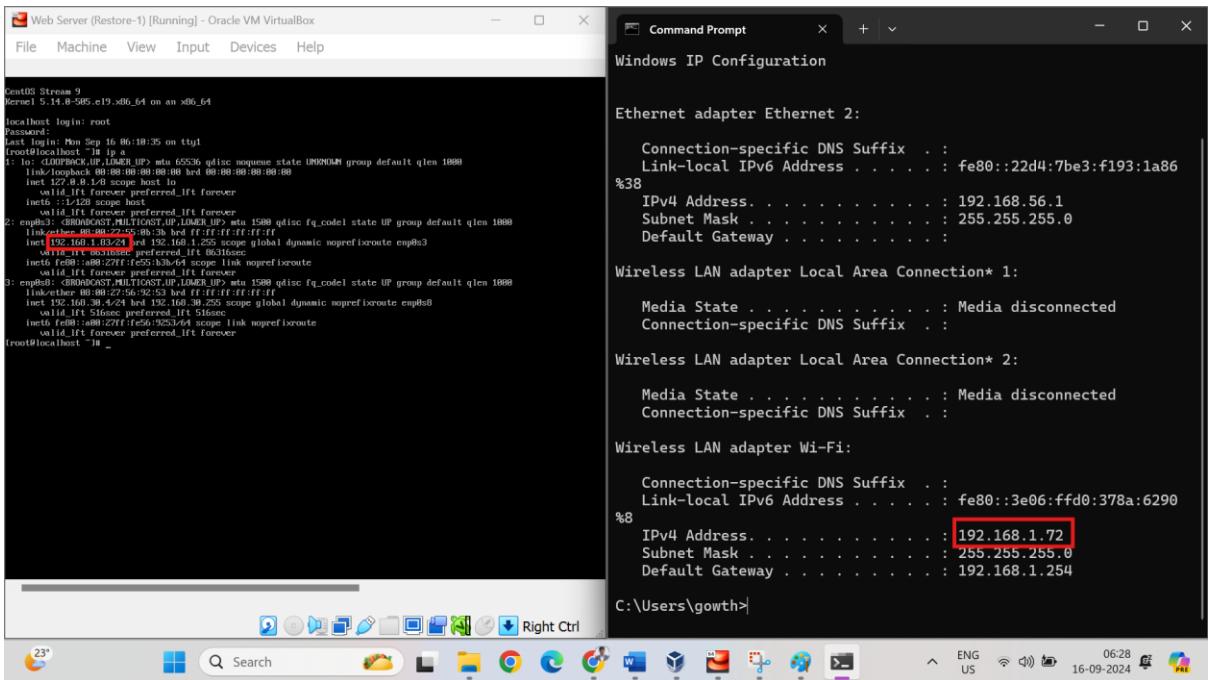
## READY TO USE



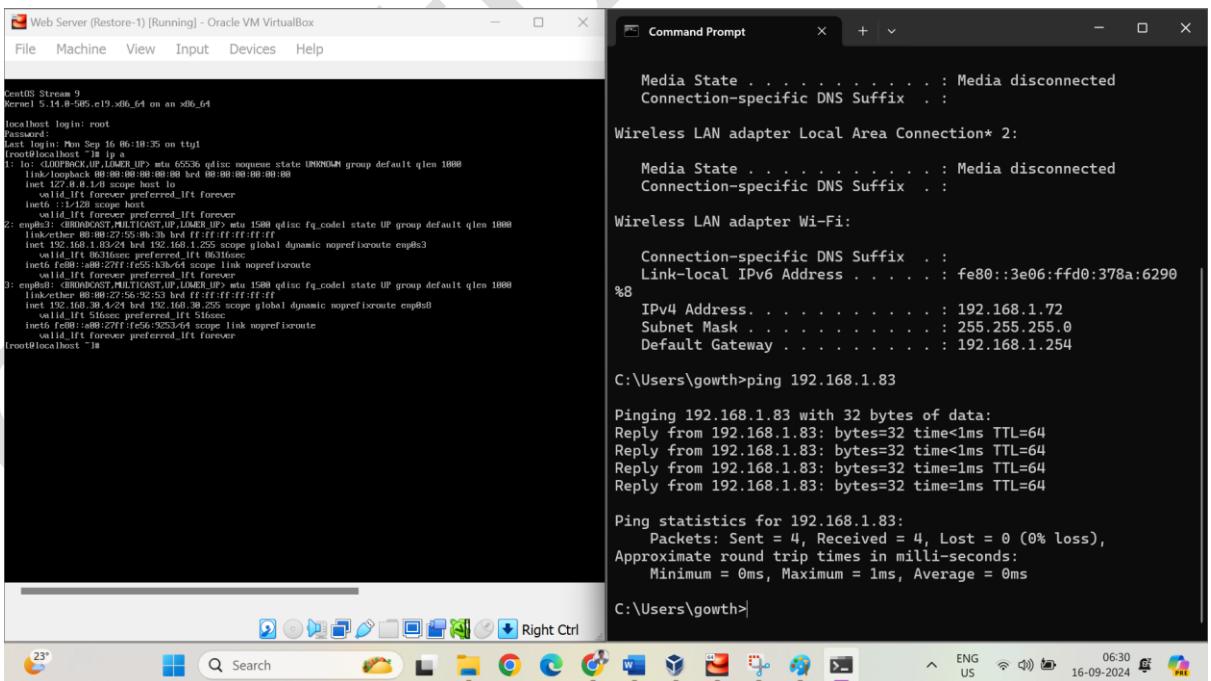
WE CAN CHECK IP ADDRESS  
COMMAND : ip a



THIS IS COMPARISION BETWEEN MY IP LAPTOP IP ADDRESS AND THE VM IP ADDRESS AS I HAVE CONNECTED MY WIFI TO THE VM THEN BOTH WILL BE IN SAME NETWORK.



WE CAN PING OUR VM WITH OUR COMMAND PROMPT



## NOW WE CAN PING THE LAPTOP FROM VM

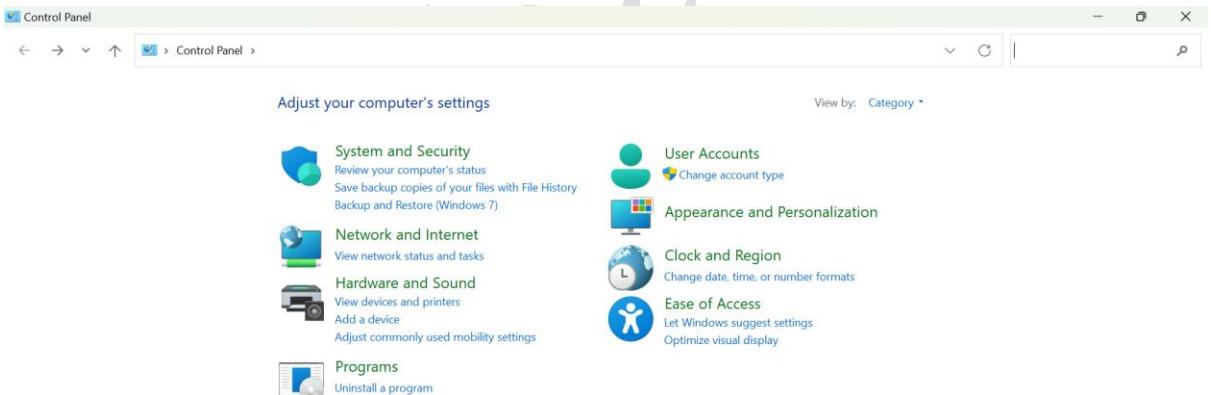
```
CentOS Stream 9
Kernel 5.14.0-585.119.x86_64 on an x86_64
localhost login: root
Password:
root@localhost ~% Last login: Mon Sep 16 06:10:35 on ttys0
root@localhost ~% ip a
1: ens3: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state UNKNOWN group default qlen 1000
    link/ether 00:0c:29:bb:6f:98 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.72 brd 192.168.1.255 scope global dynamic noprefixroute
        valid_lft forever preferred_lft forever
        link-layer-brd 00:0c:29:bb:6f:98
        brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:febb:6f98 brd ff:ff:ff:ff:ff:ff scope link noprefixroute
        valid_lft forever preferred_lft forever
        link-layer-brd 00:0c:29:bb:6f:98
        brd ff:ff:ff:ff:ff:ff
3: enp0s3: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:0c:29:bb:6f:98 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.72 brd 192.168.1.255 scope global dynamic noprefixroute
        valid_lft forever preferred_lft forever
        link-layer-brd 00:0c:29:bb:6f:98
        brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:febb:6f98 brd ff:ff:ff:ff:ff:ff scope link noprefixroute
        valid_lft forever preferred_lft forever
        link-layer-brd 00:0c:29:bb:6f:98
        brd ff:ff:ff:ff:ff:ff
root@localhost ~% ping 192.168.1.72
PING 192.168.1.72 (192.168.1.72) 56(84) bytes of data:
C:\Users\gowth>
```

Pinging 192.168.1.83 with 32 bytes of data:  
Reply from 192.168.1.83: bytes=32 time<1ms TTL=64  
Reply from 192.168.1.83: bytes=32 time<1ms TTL=64  
Reply from 192.168.1.83: bytes=32 time=1ms TTL=64  
Reply from 192.168.1.83: bytes=32 time=1ms TTL=64

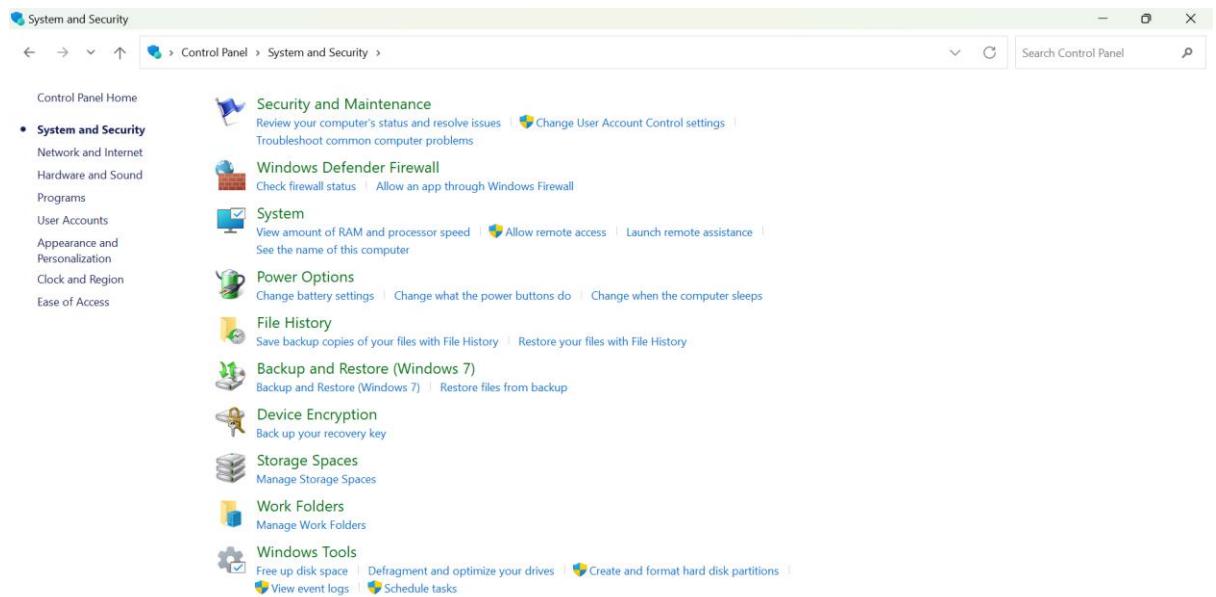
Ping statistics for 192.168.1.83:  
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\gowth>

IT IS NOT PINGING BECAUSE MY FIREWALL IS ON CO I WILL OFF THAT AND I WILL TRY IT  
OPEN CONTROL PANEL



## THEN WE CAN CLICK THE SYSTEM AND SECURITY OPTION



WE HAVE AN OPTION CALLED WINDOWS DEFENDER FIREWALL  
CLICK THAT



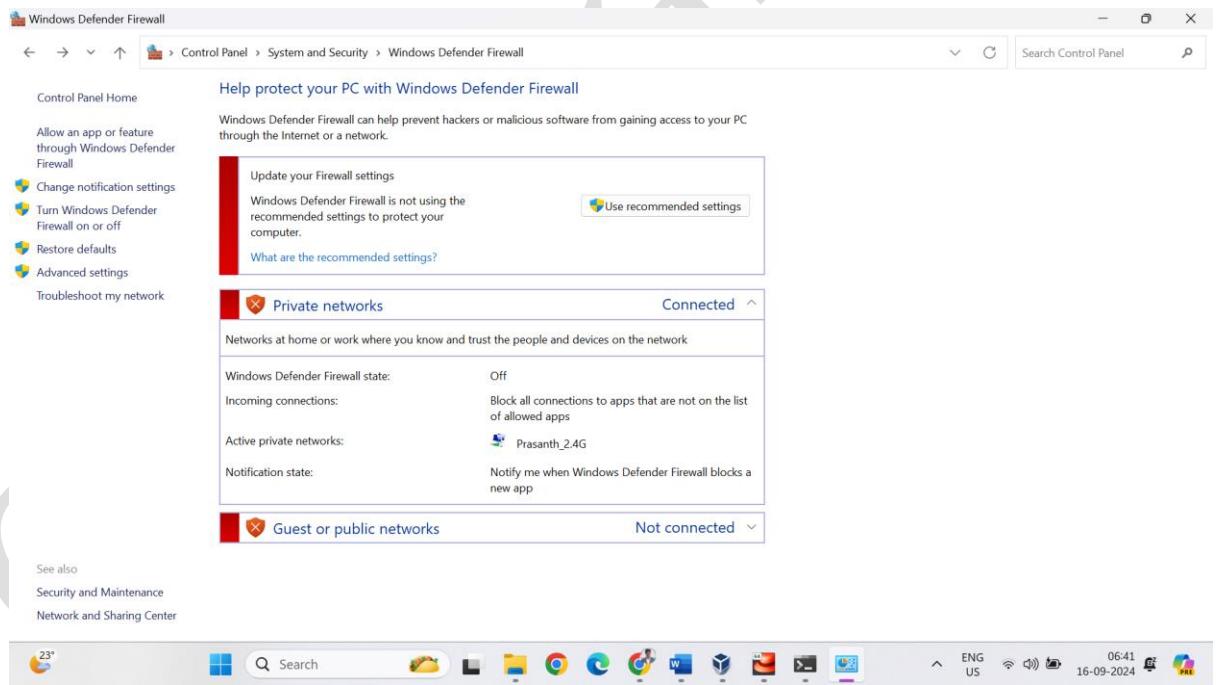
See also  
Security and Maintenance  
Network and Sharing Center



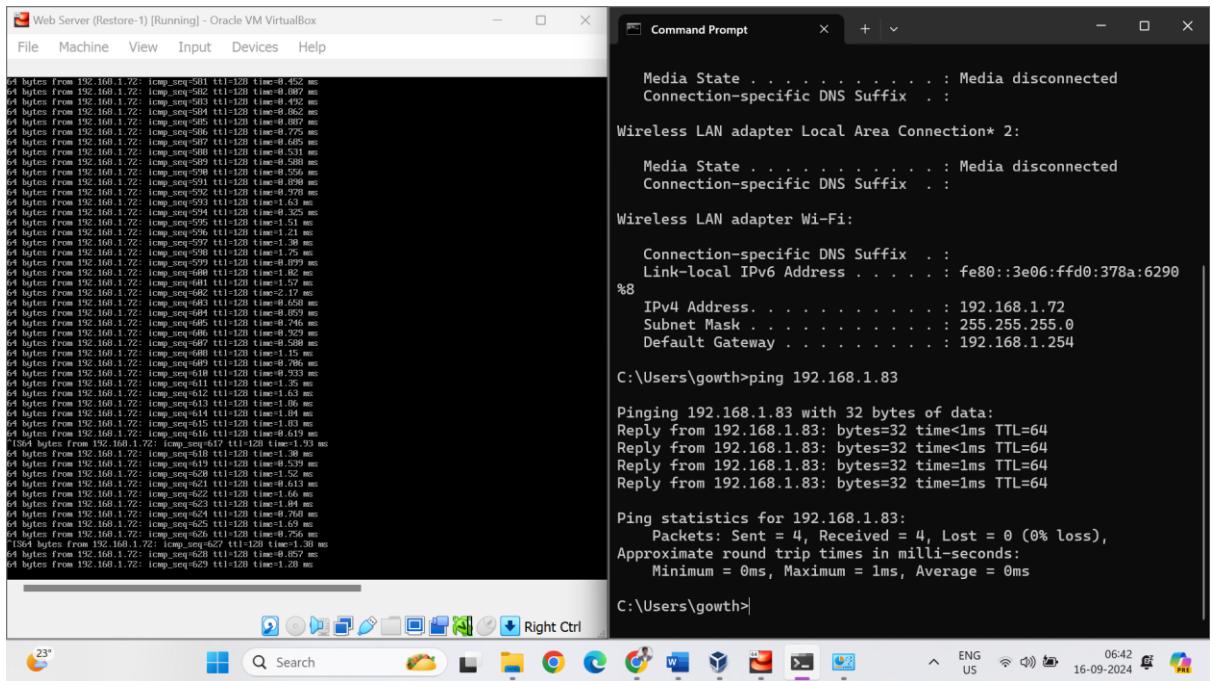
**THERE IS AN OPTION CALLED TURN WINDOWS DEFENDER FIREWALL ON OR OFF SO CLICK THAT**



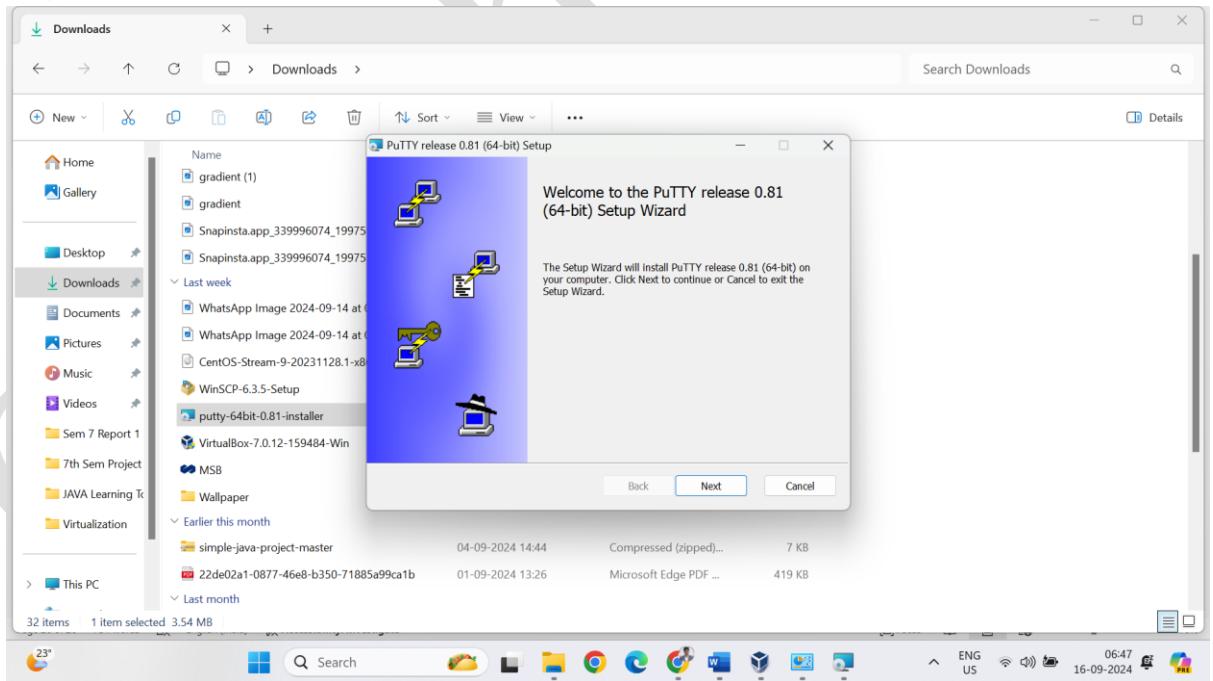
**NOW WE CAN TURN OFF BOTH PUBLIC AND PRIVATE NETWORK**



## NOW LET'S TRY TO PING IT IS PINGING NOW

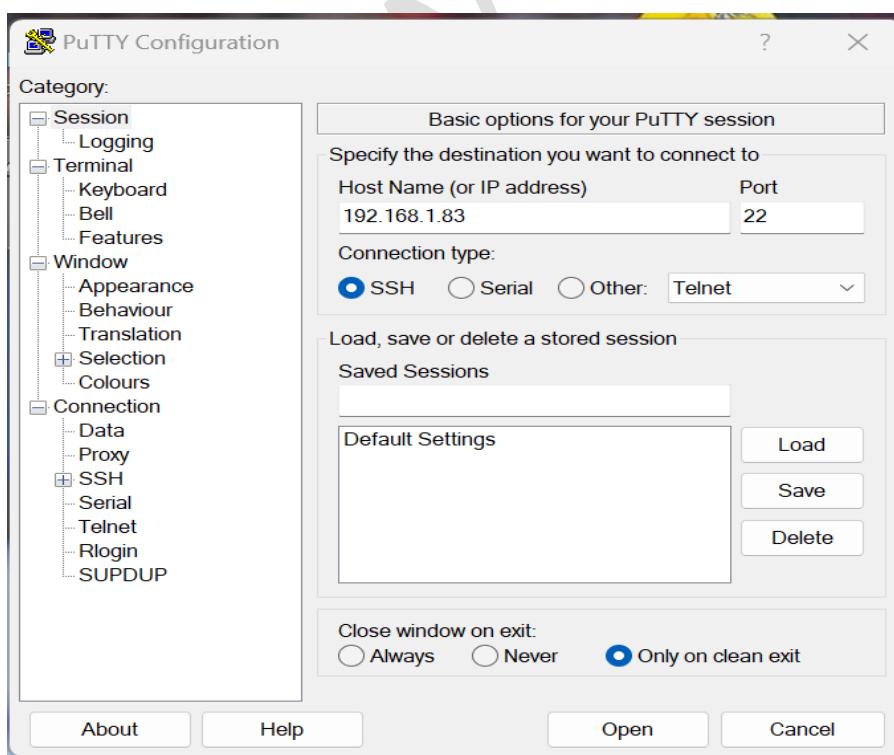
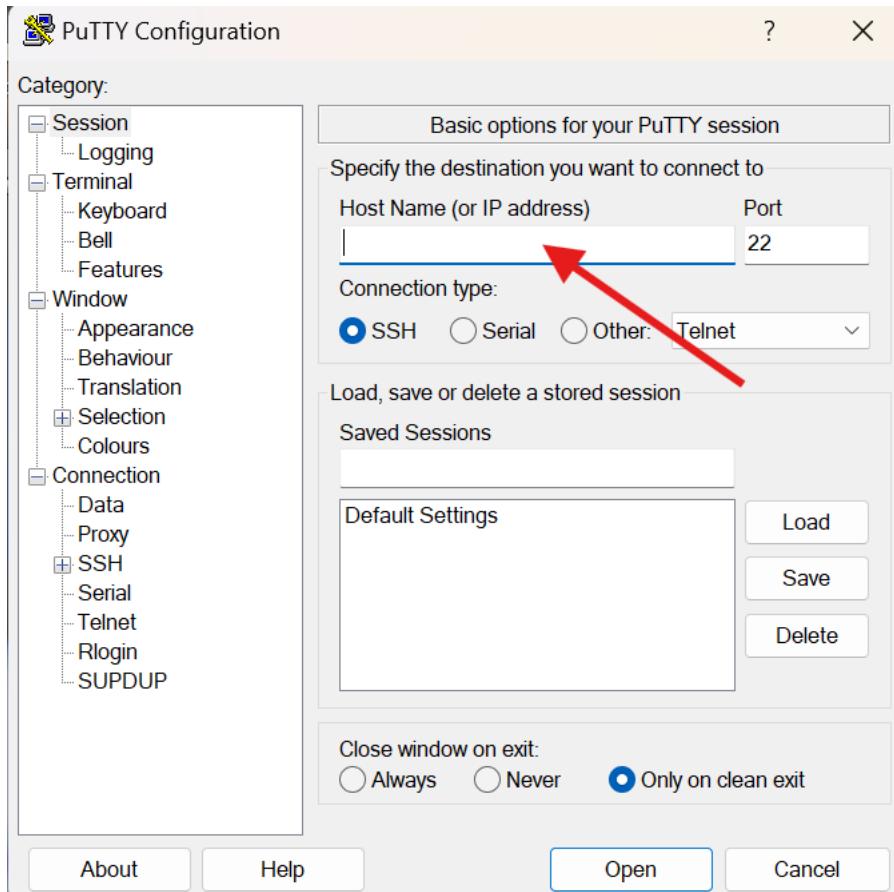


## NOW WE CAN CONNECT OUR VM IN REMOTE SO, INSTALL THE PUTTY SOFTWARE

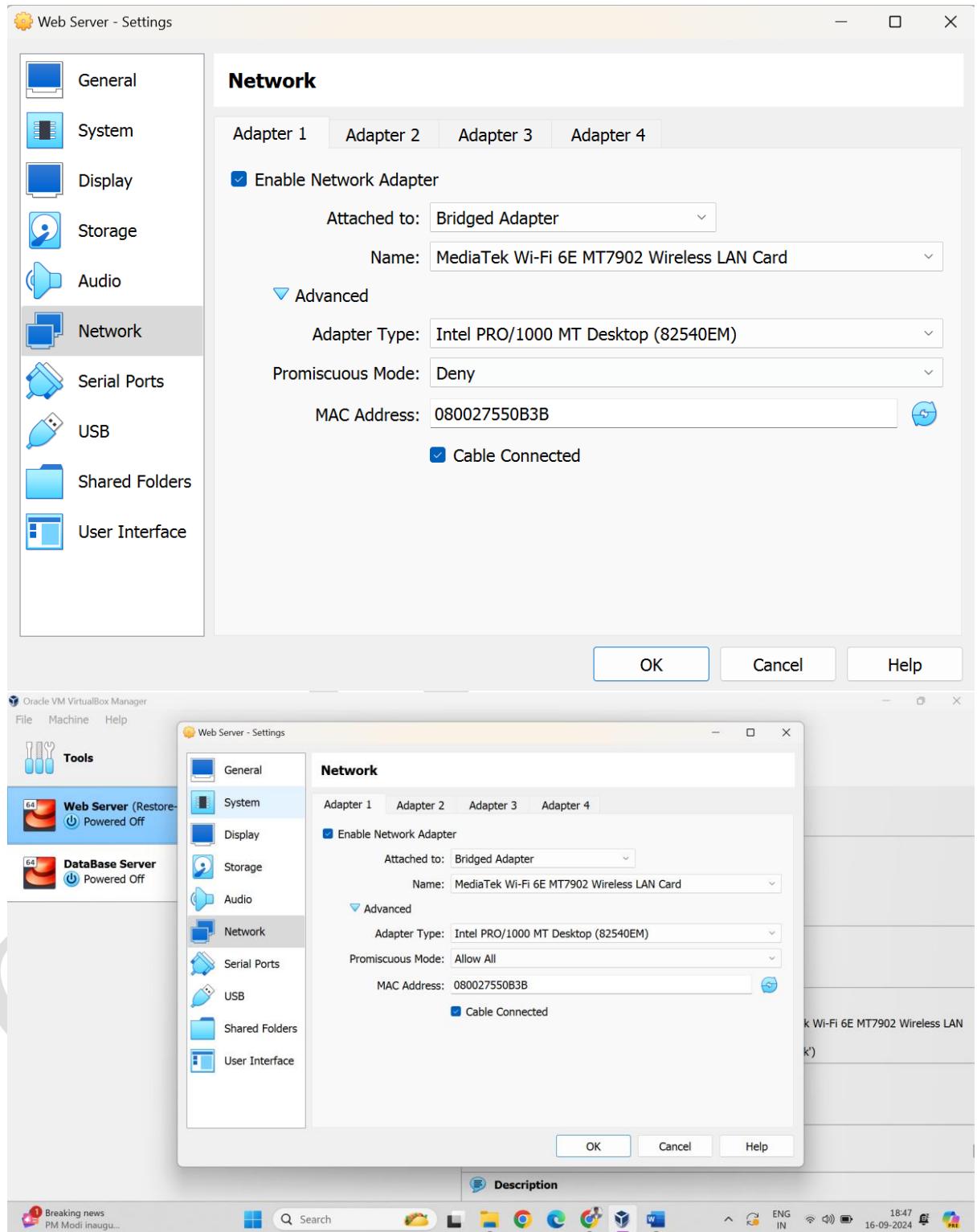


OUR PUTTY SOFTWARE HAS BEEN INSTALLED SO LET US START THE WORK

GIVE OUR VM IP ADDRESS IN THE ARROW



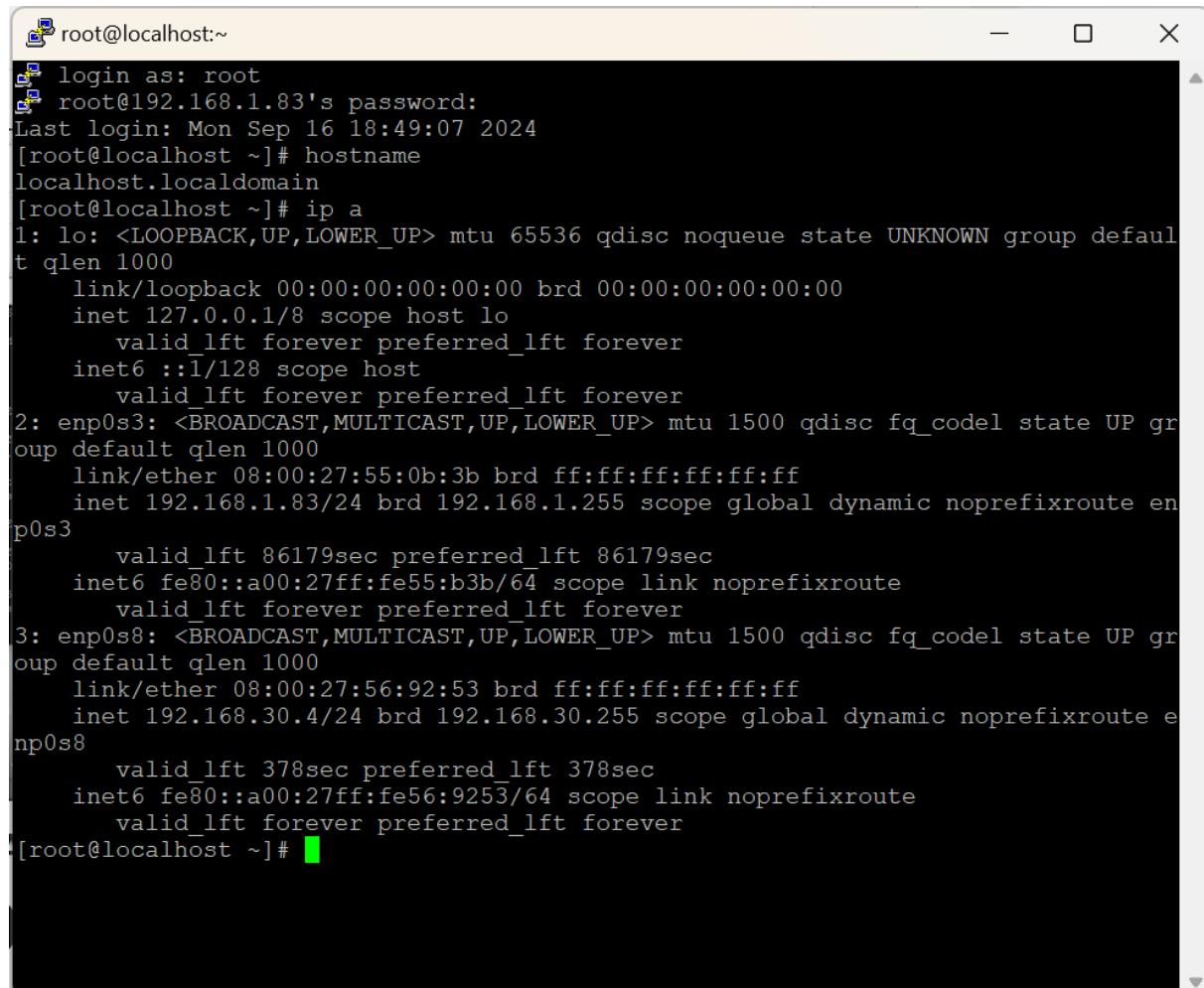
I HAVE FACED WHILE CONNECTING MY VM TO PUTTY BECAUSE THE PROMISCUOUS MODE IS IN DENY LET CHANGE THAT INTO ALLOW ALL



```
192.168.1.83 - PuTTY
login as: root
root@192.168.1.83's password: [REDACTED]

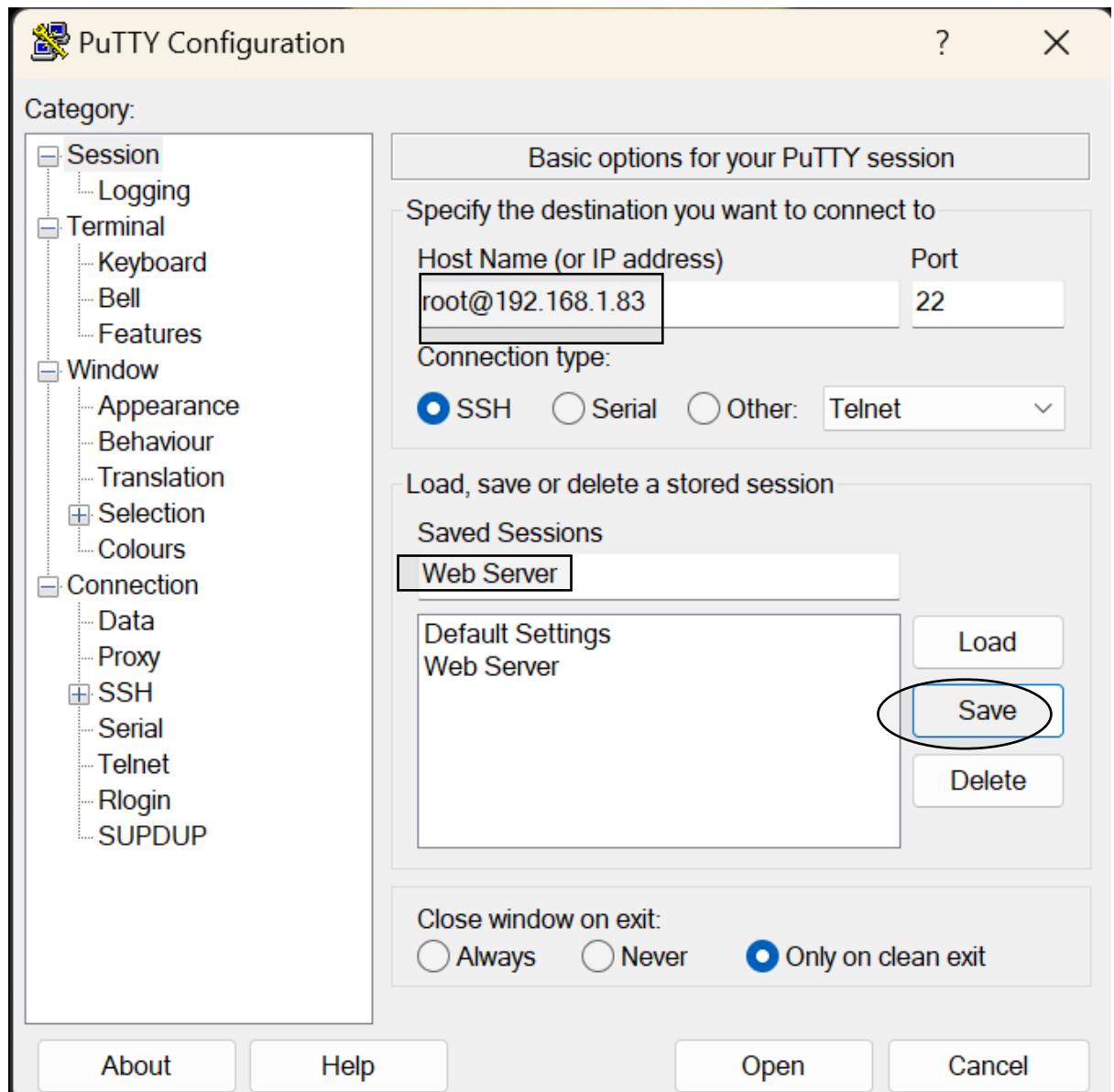
root@localhost:~
login as: root
root@192.168.1.83's password:
Last login: Mon Sep 16 18:49:07 2024
[root@localhost ~]# [REDACTED]
```

## WE HAVE CONNECTED WITH PUTTY TO OUR VM

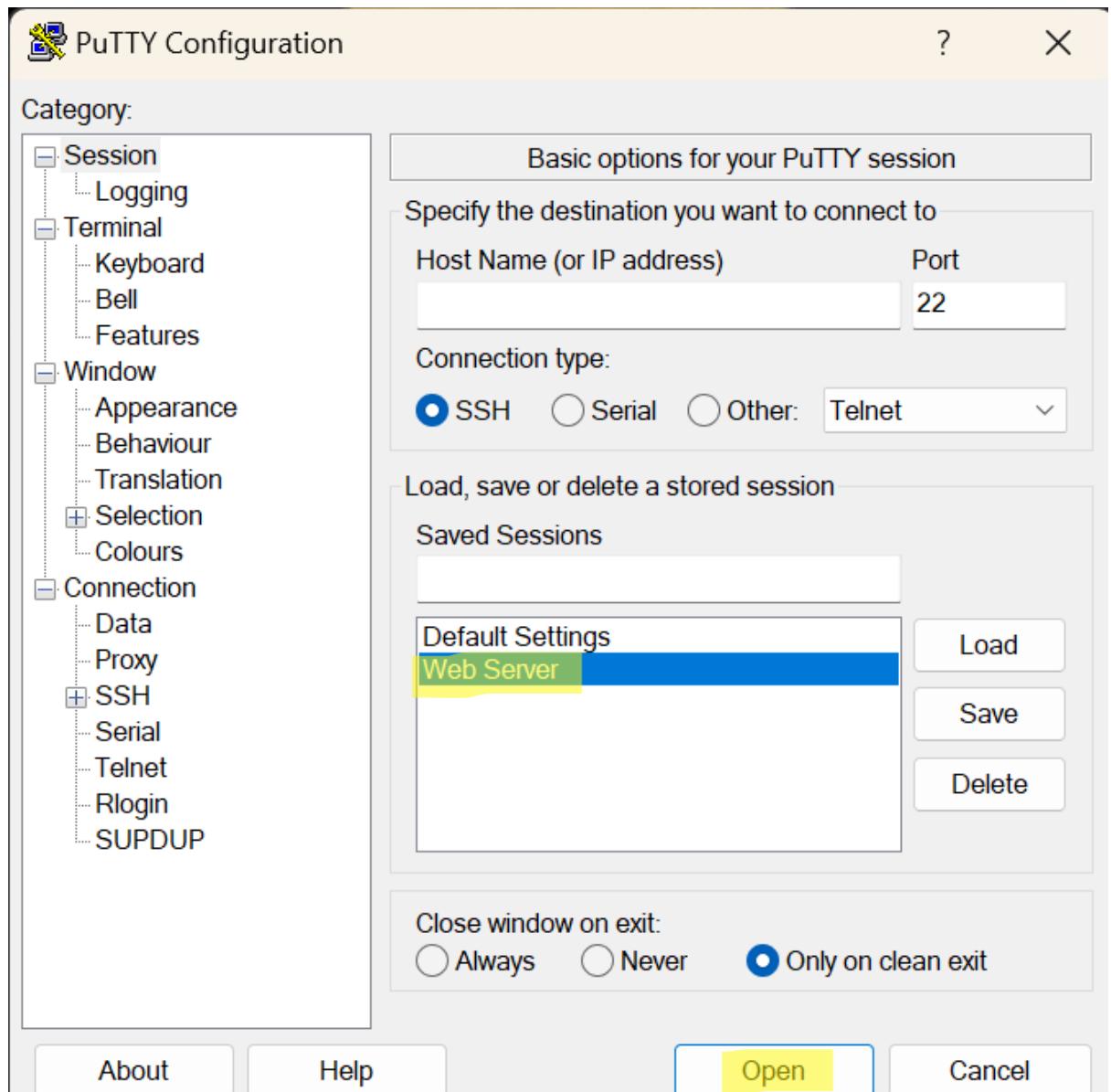


```
root@localhost:~  
root@192.168.1.83's password:  
Last login: Mon Sep 16 18:49:07 2024  
[root@localhost ~]# hostname  
localhost.localdomain  
[root@localhost ~]# ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:55:0b:3b brd ff:ff:ff:ff:ff:ff  
    inet 192.168.1.83/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3  
        valid_lft 86179sec preferred_lft 86179sec  
        inet6 fe80::a00:27ff:fe55:b3b/64 scope link noprefixroute  
            valid_lft forever preferred_lft forever  
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:56:92:53 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.30.4/24 brd 192.168.30.255 scope global dynamic noprefixroute enp0s8  
        valid_lft 378sec preferred_lft 378sec  
        inet6 fe80::a00:27ff:fe56:9253/64 scope link noprefixroute  
            valid_lft forever preferred_lft forever  
[root@localhost ~]#
```

WHEN WE ENTER TO THE PUTTY EVERY TIME WE NEED TO DO ALL THIS STUFF TO STOP THIS WE ARE GOING TO FOLLOW SOME STEPS

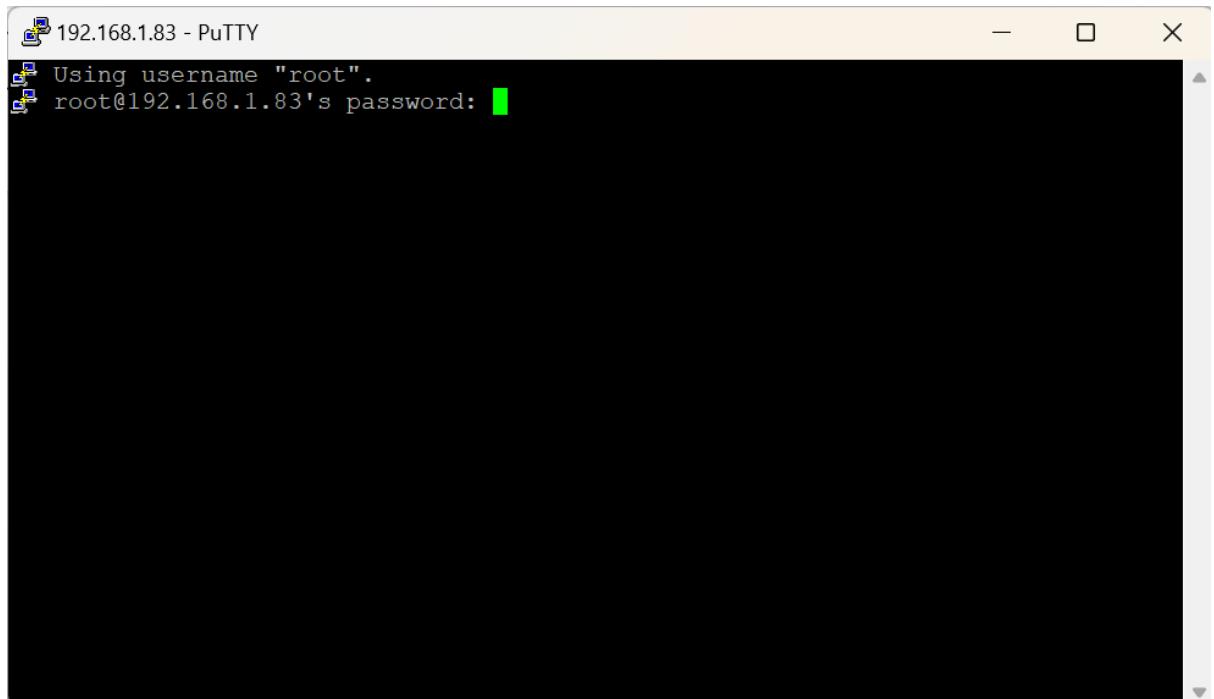


GO



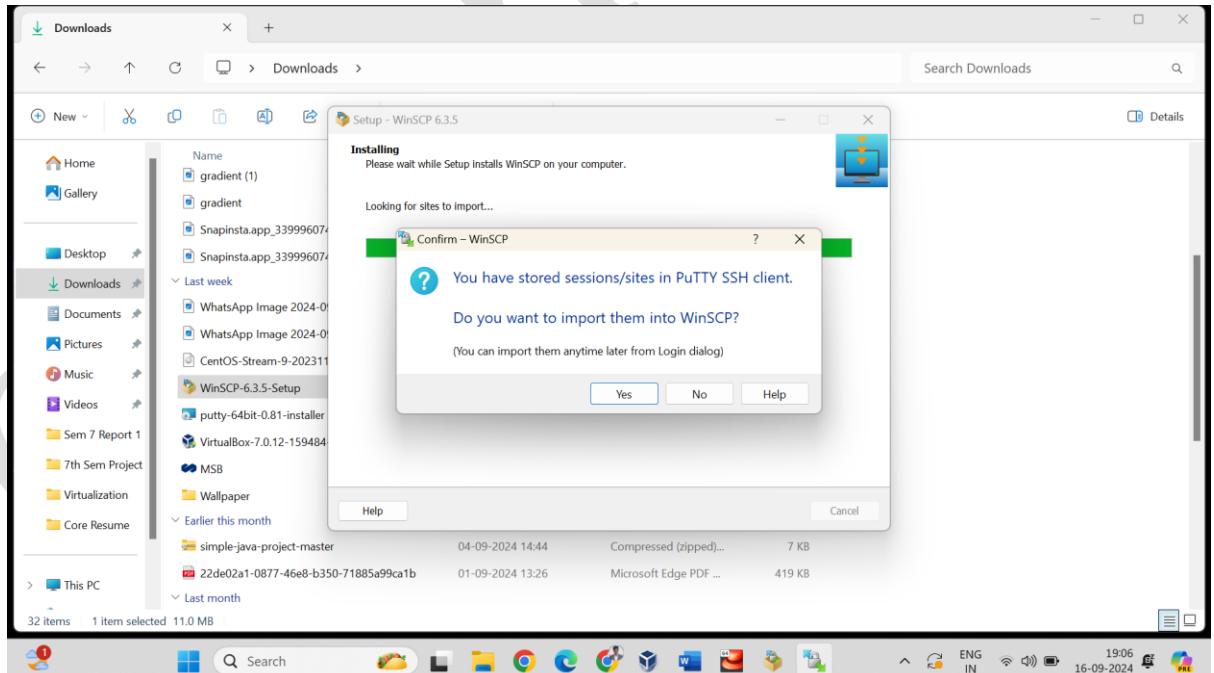
GO

## OUR VM IS OPEN NOW

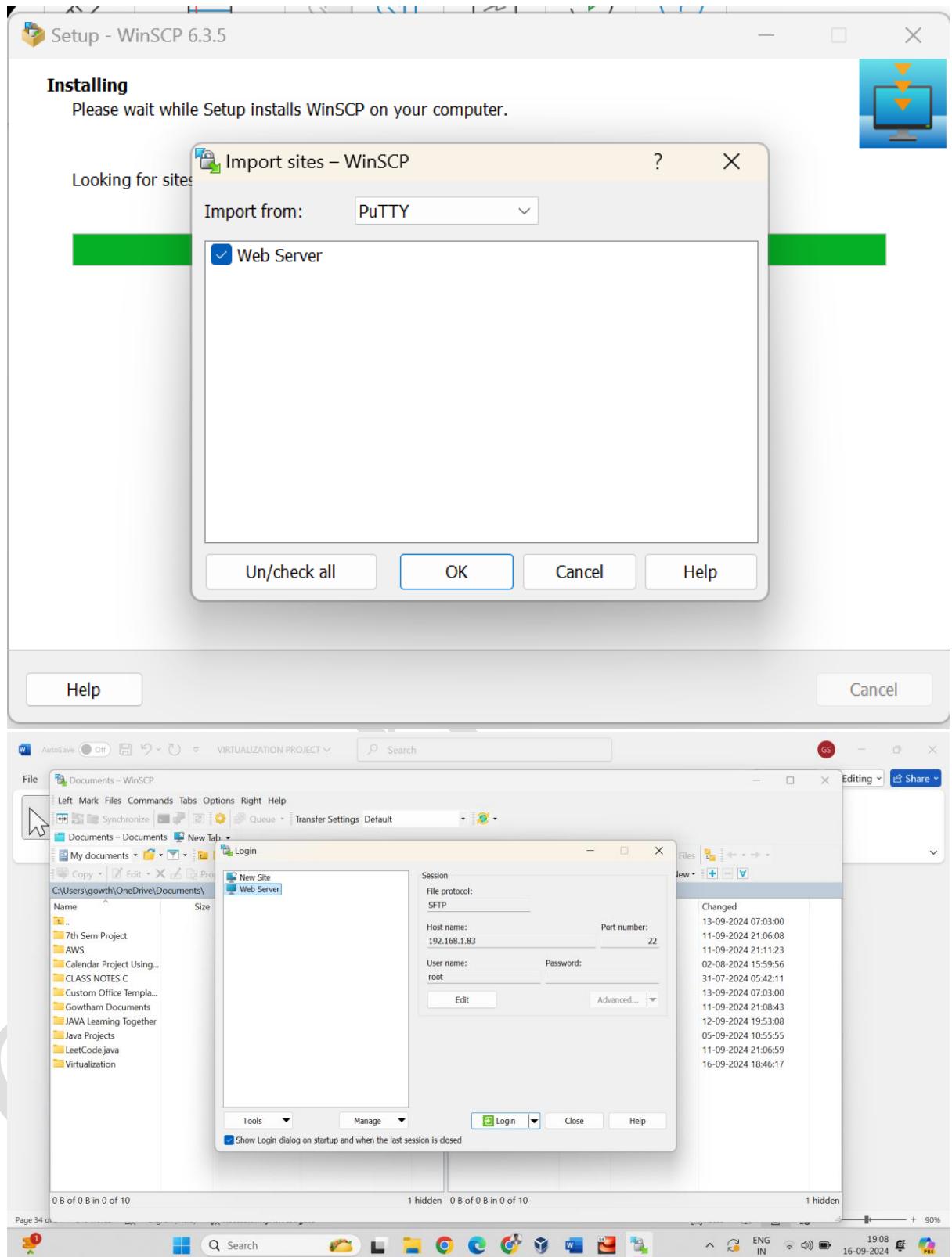


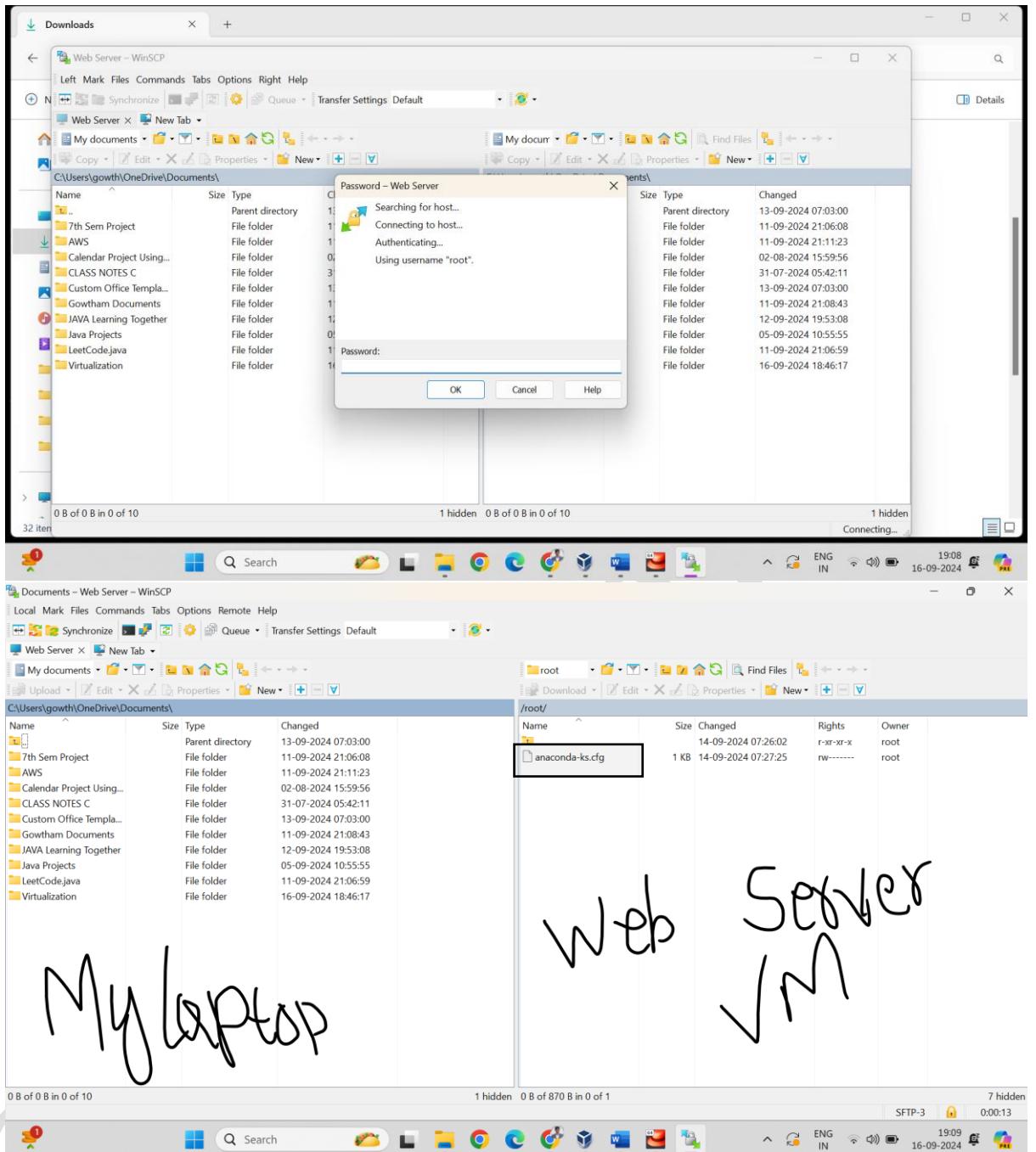
NOW WE HAVE ADDED OUR VM TO THE PUTTY WHICH IS USED TO INTERACT WITH THE SERVER DIRECTLY

NOW WE ARE GOING TO CREATE FTP WITH WINSSCP



THIS HELPS TO DIRECTLY CONNECT OUR VM





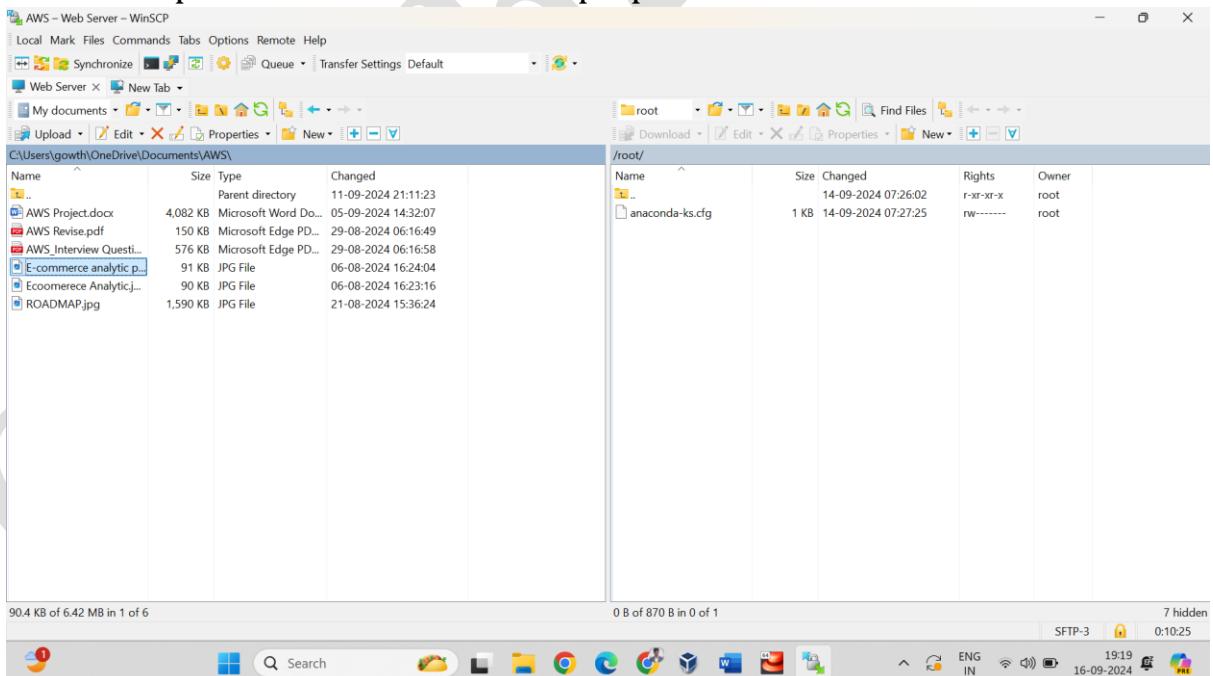
Now let's check it out did the anaconda file is present in our vm

```
CentOS Stream 9
Kernel 5.14.0-505.e19.x86_64 on an x86_64

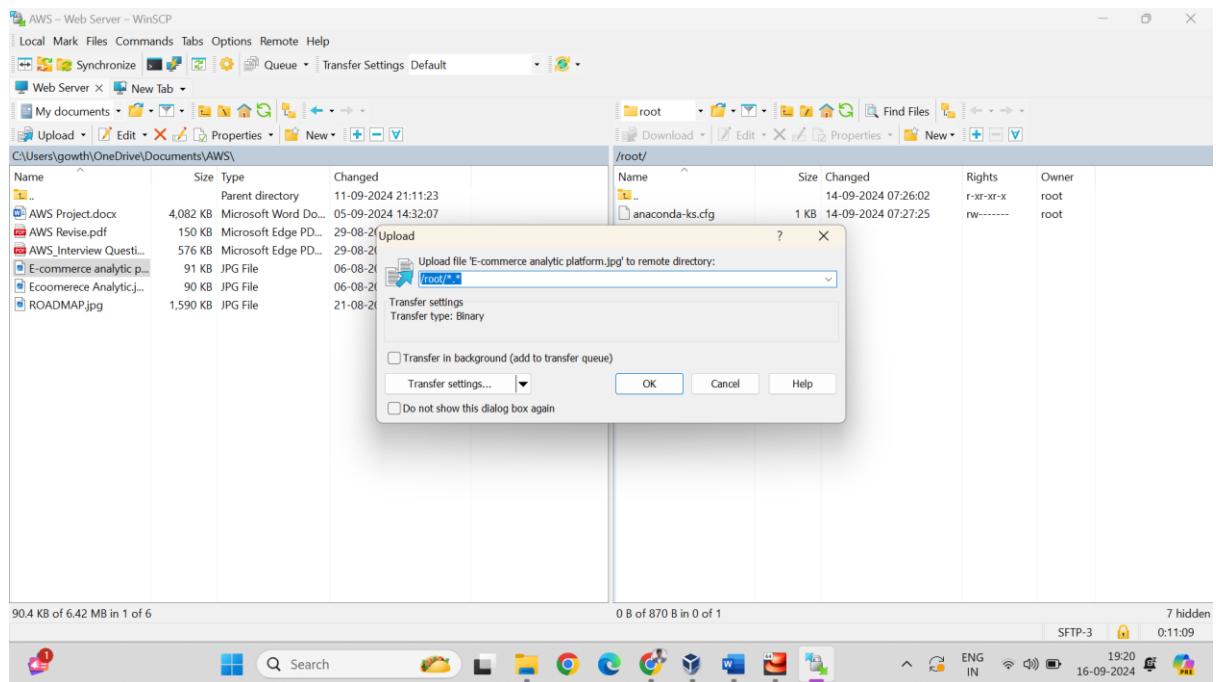
localhost login: root
Password:
Last login: Mon Sep 16 06:27:52 on ttys1
root@localhost ~% ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:00:27:55:0b:3b brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.83/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
        valid_lft 66365sec preferred_lft 66385sec
    inet6 fe80::a00:27ff:fe55:b3b/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:00:27:56:92:53 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.4/24 brd 192.168.0.255 scope global dynamic noprefixroute enp0s8
        valid_lft 584sec preferred_lft 584sec
    inet6 fe80::a00:27ff:fe56:9253/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@localhost ~% ps
-bash: ps: command not found
root@localhost ~% pwd
/root
root@localhost ~% ls
anaconda-ks.cfg
[root@localhost ~]#
```

It is present in our vm

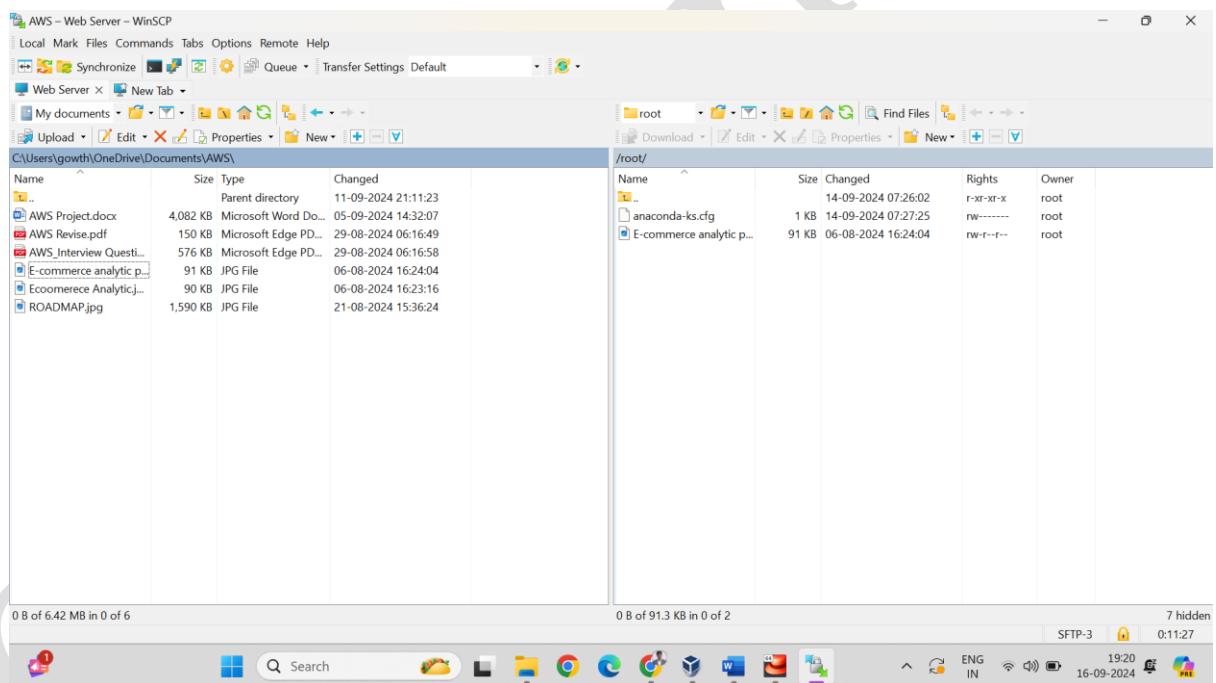
Now lets upload the file from our Laptop to our VM



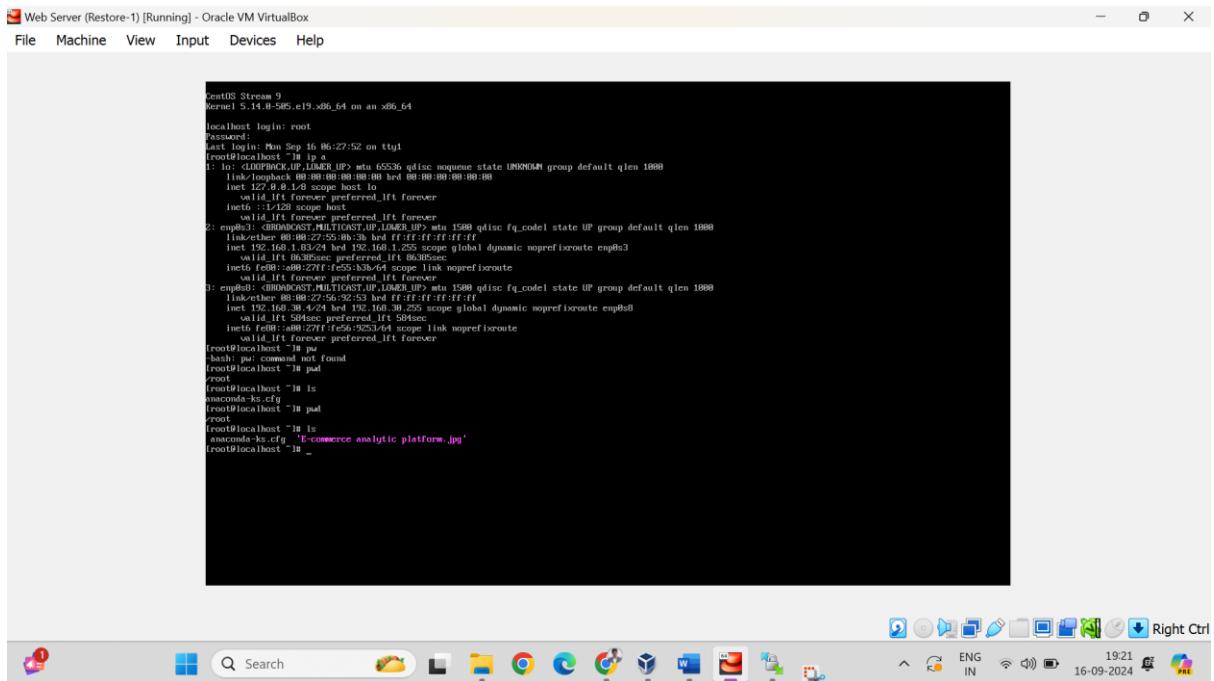
RIGTH CLICK ON THE FILE AND CLICK THE UPLOAD OPTION



NOW WE HAVE UPLOADED THE FILE TO OUR VM



## NOW WE CAN CHECK THIS IN THE VM



CentOS Stream 9  
Kernel 5.14.0-362.e19.x86\_64 on an x86\_64  
localhost login: root  
Password:  
Last login: Mon Sep 16 06:27:52 on ttys1  
root@localhost ~% ip a  
1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
 link/loopback brd 00:00:00:00:00:00 state UNKNOWN group default  
 inet 127.0.0.1/8 brd 00:00:00:00:00:00 scope host lo  
 valid\_lft forever preferred\_lft forever  
 inetet ::1/128 scope host  
 valid\_lft forever preferred\_lft forever  
2: enp0s3: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000  
 link/ether 00:0c:29:00:00:32 brd ff:ff:ff:ff:ff:ff  
 inet 192.168.1.83/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3  
 valid\_lft forever preferred\_lft forever  
 inet6 fe80::0c29:3ff:fe00:32%enp0s3/64 scope link noprefixroute  
 valid\_lft forever preferred\_lft forever  
3: enp0s8: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000  
 link/ether 00:0c:29:00:00:3f brd ff:ff:ff:ff:ff:ff  
 inet 192.168.30.4/24 brd 192.168.30.255 scope global dynamic noprefixroute enp0s8  
 valid\_lft forever preferred\_lft forever  
 inet6 fe80::0c29:3ff:fe00:3f%enp0s8/64 scope link noprefixroute  
 valid\_lft forever preferred\_lft forever  
root@localhost ~% ps aux  
 user pid nice pri size nice pri command  
root 1 0 0 0 0 0 0 /bin/bash  
root@localhost ~% ls  
root@localhost ~% curl -F file=@e-commerce\_analytic\_platform.jpg http://192.168.30.4:8080/upload  
root@localhost ~%

THE FILE HAS BEEN ADDED SUCCESSFULLY

STILL NOW OUR WEB SERVER IS READY

## GOING TO LEARN LINUX TO CONTINUE THE PROJECT

### 1. TEMPORARY HOST NAME SETTING

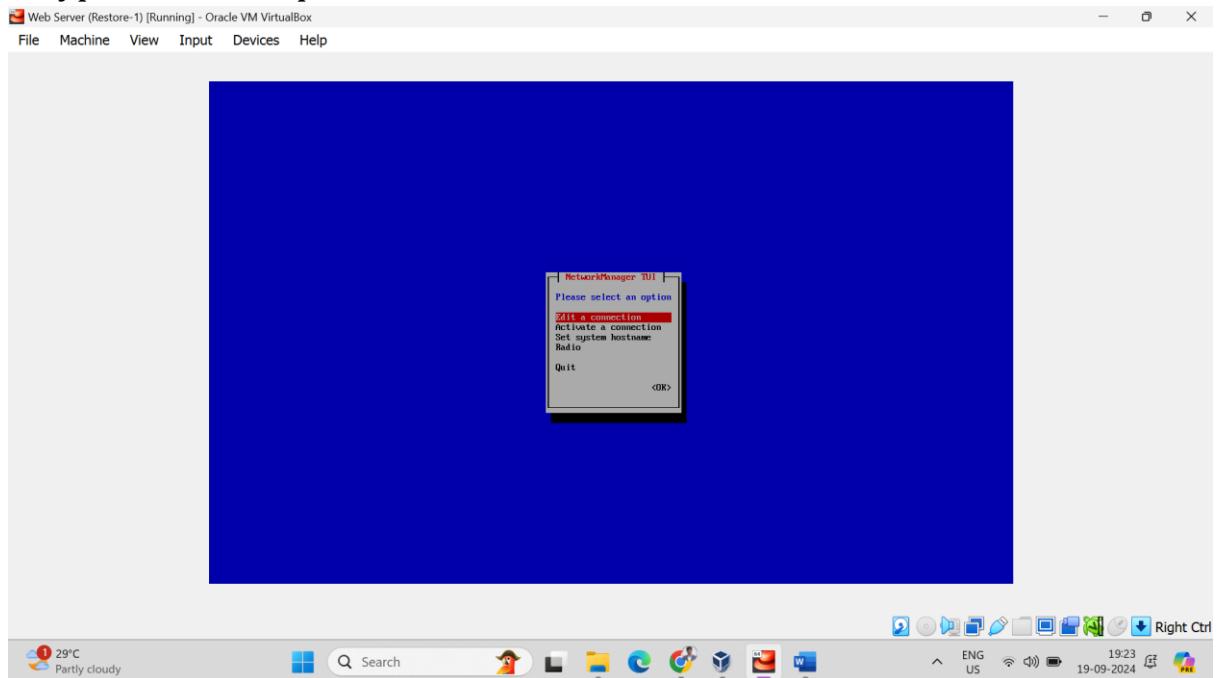
Step1: Login ----->root, password

Step2: Type hostname to see predefined hostname by the computer

Step3: To change the hostname type----->hostname webserver.com

## 2. GOING TO SET OUR OWN IP ADDRESS TO THE VM

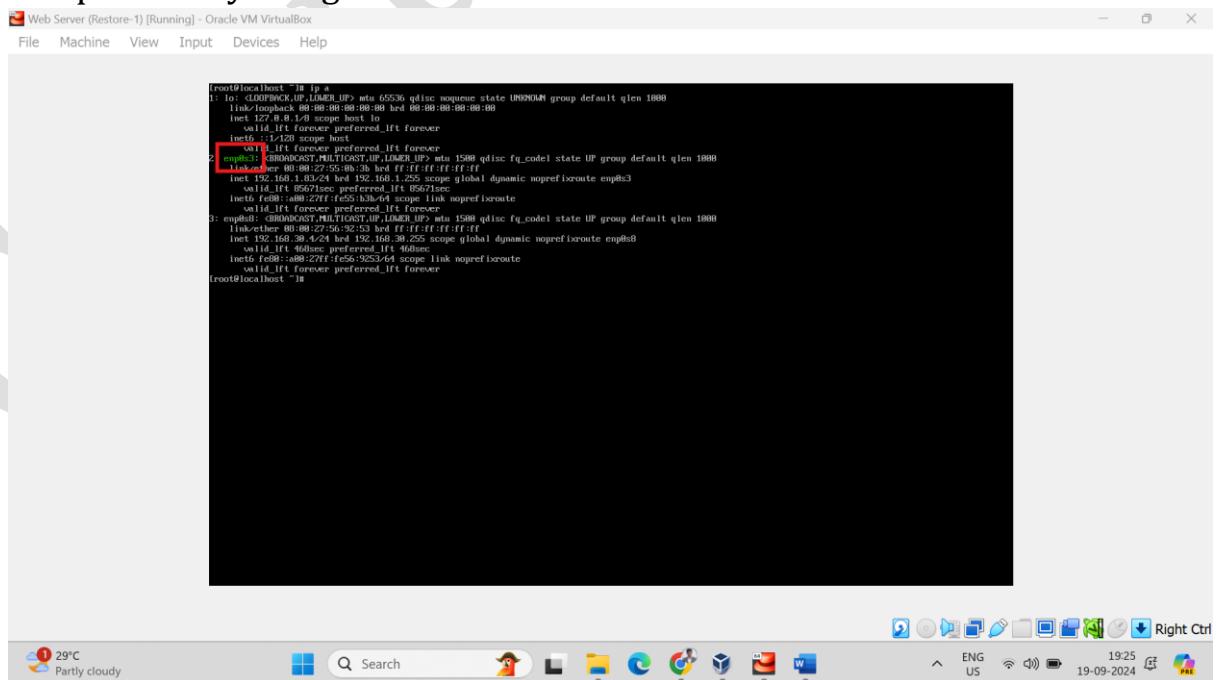
1. To see existing ip address type ---->ip a
- 2.Type nmcli and press enter



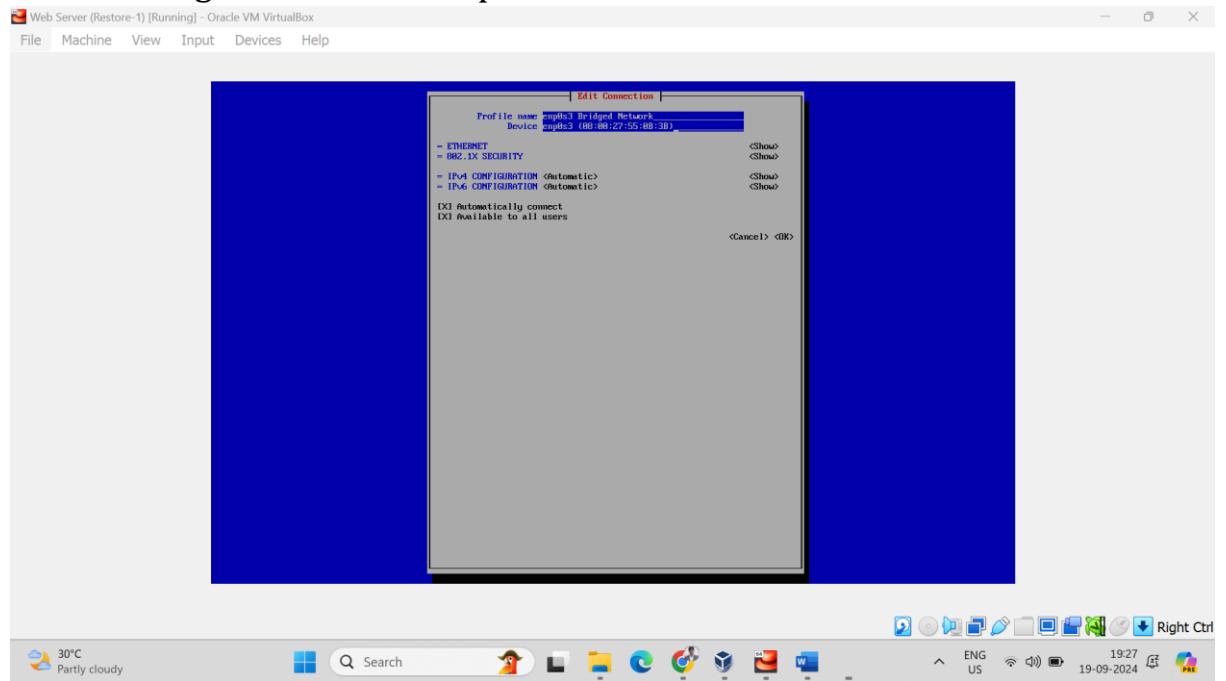
### 3.Press Enter

We are going to edit our Brigid Network so let's check it out again and continue which is bridged network.

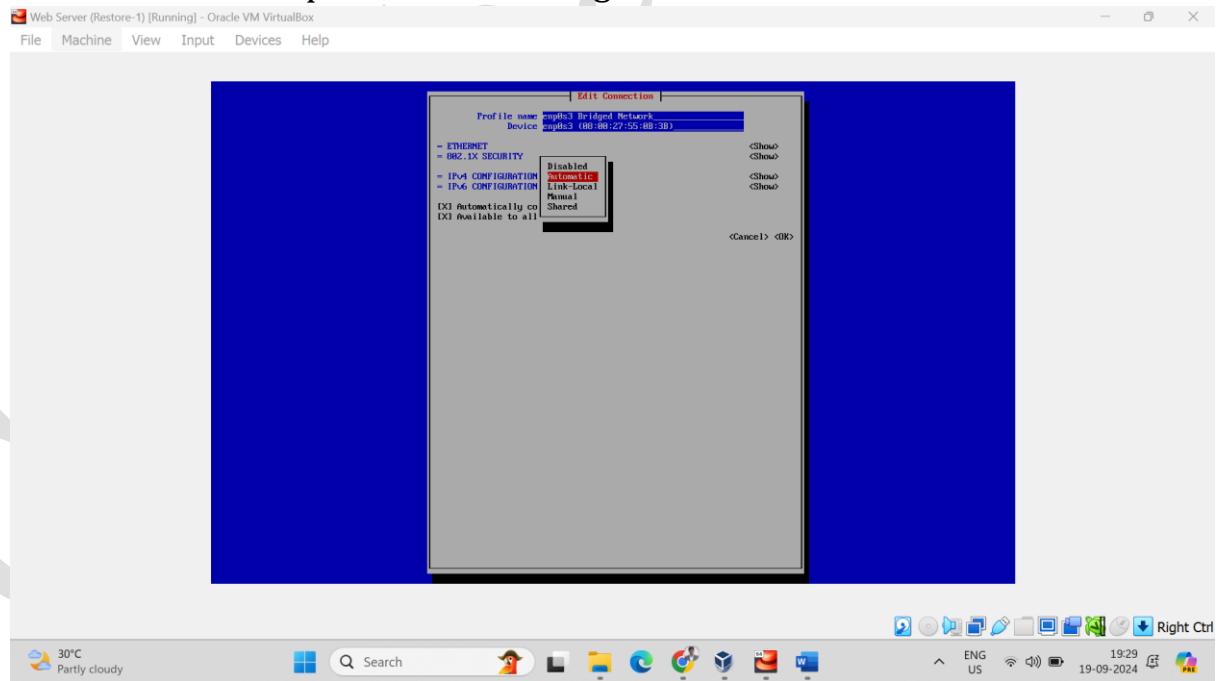
### 4.enp0s3 is my bridged network



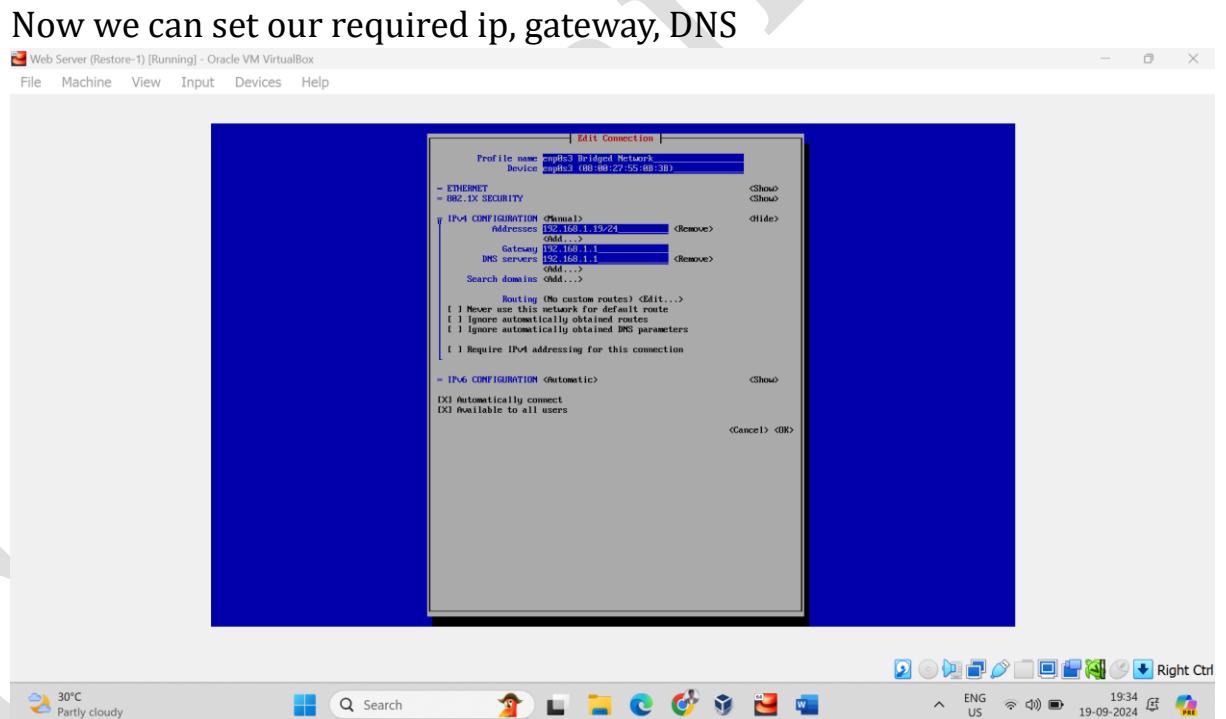
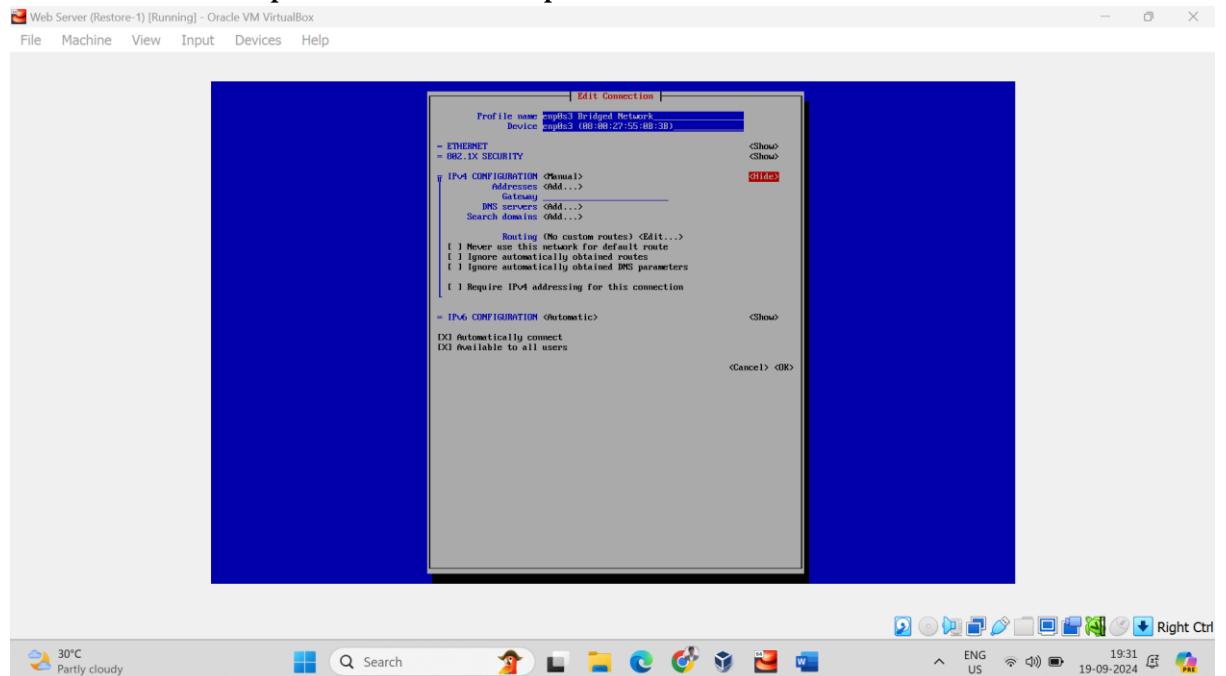
So we can again continue the process



Click the down arrow to continue  
And go to IPV4 configuration  
It is in automatic option we can change that to manual

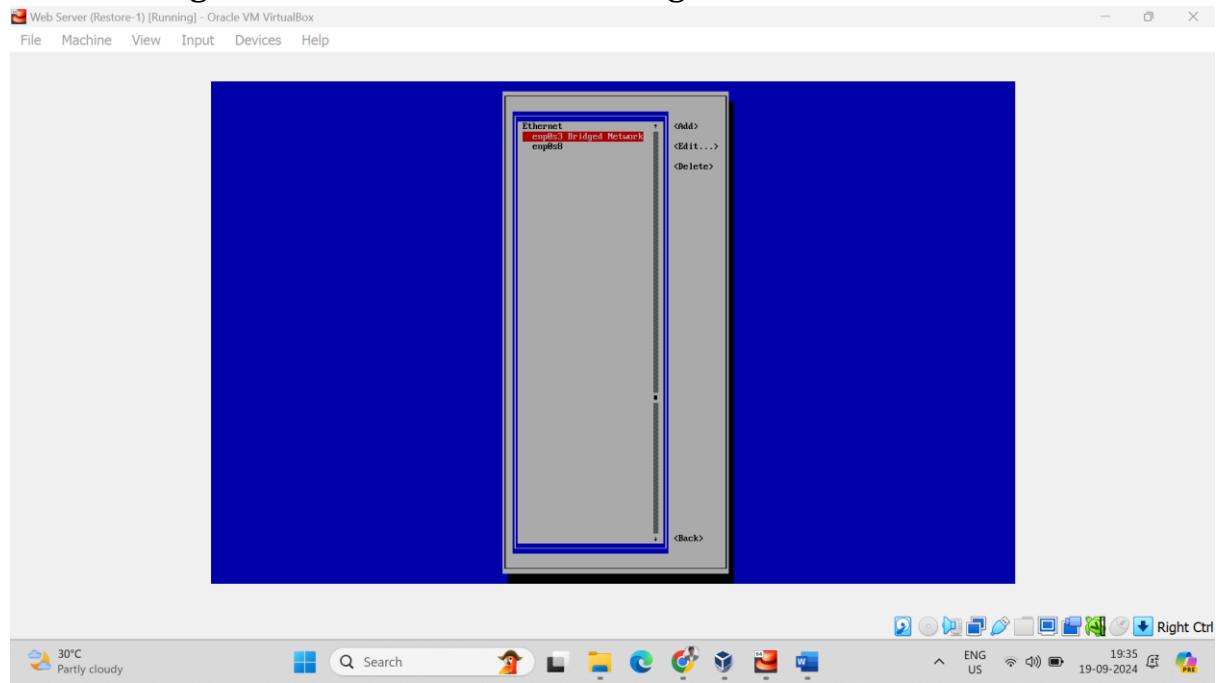


Click the show option to edit the ip address

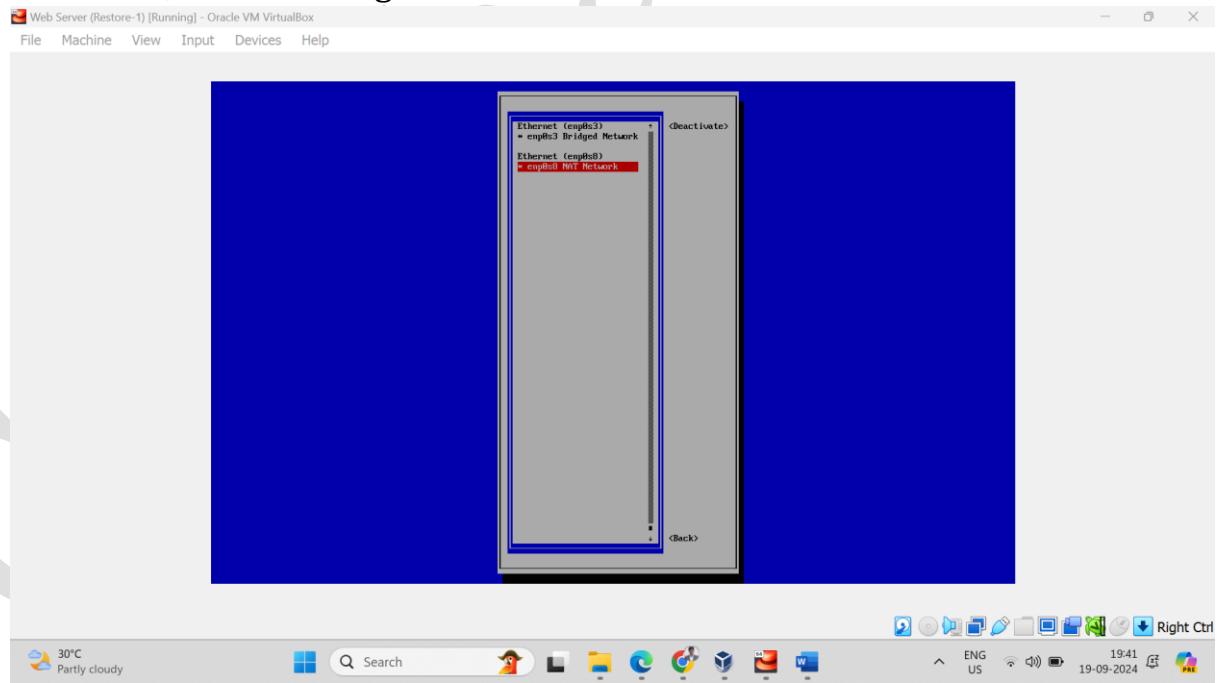


The ip address has been changed so now we can enter ok

As we changed the name it has been changed here also



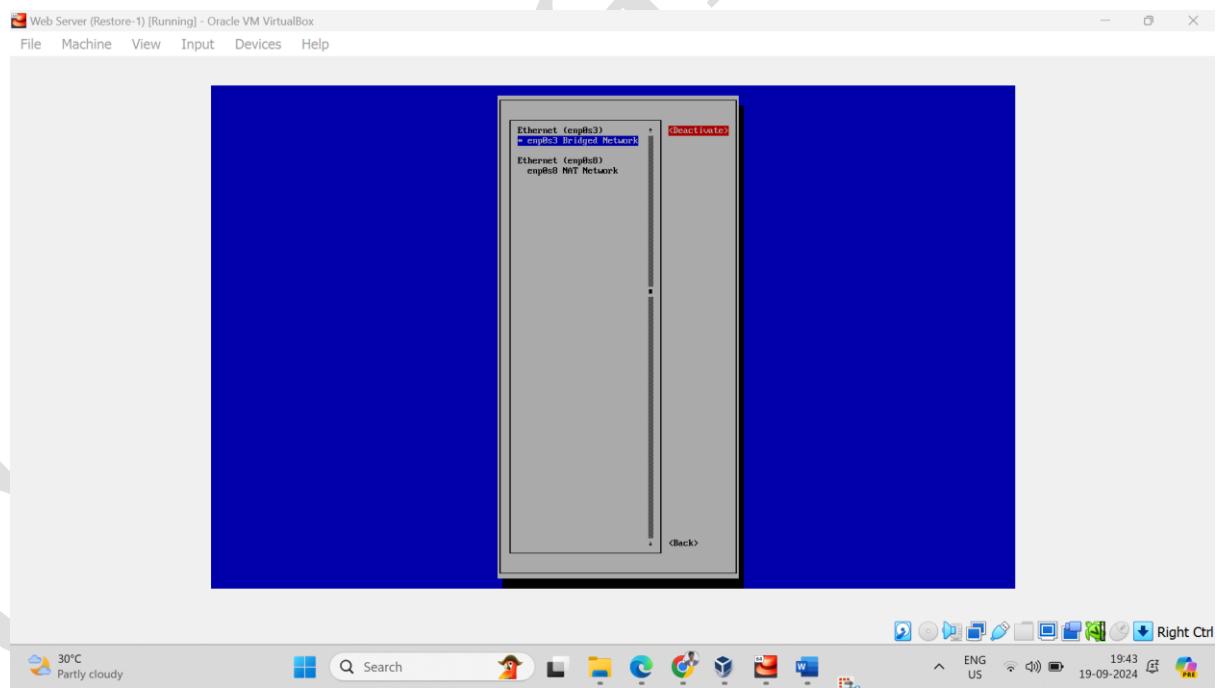
As we made some changes we need to deactivate the vm and again activate it, so that changes will be confirmed.



As both ip are in deactivated if we try to see ip then it will be like this only

```
root@localhost ~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
            valid_lft forever preferred_lft forever
            link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff
2: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0B:0B:0B:0B:0B brd ff:ff:ff:ff:ff:ff
        inet 192.168.1.255/24 brd 192.168.1.255 scope global eth1
            valid_lft forever preferred_lft forever
            link/ether 00:0B:0B:0B:0B:0B brd ff:ff:ff:ff:ff:ff
root@localhost ~# _
```

I have activated it



Now changed ip are allocated

A screenshot of a Linux terminal window titled "Web Server (Restore-1) [Running] - Oracle VM VirtualBox". The terminal displays the output of the 'ifconfig' command, which shows various network interfaces and their configurations. Key details include:

- The 'eth0' interface is set to 'loopback' and has an IP of '127.0.0.1'. It uses the 'qdisc fq\_codel' discipline.
- The 'wlan0' interface has an IP of '192.168.1.19'. It uses the 'qdisc fq\_codel' discipline.
- The 'eth1' interface has an IP of '192.168.3.10'. It uses the 'qdisc fq\_codel' discipline.
- The 'eth2' interface has an IP of '192.168.3.11'. It uses the 'qdisc fq\_codel' discipline.
- The 'eth3' interface has an IP of '192.168.3.12'. It uses the 'qdisc fq\_codel' discipline.

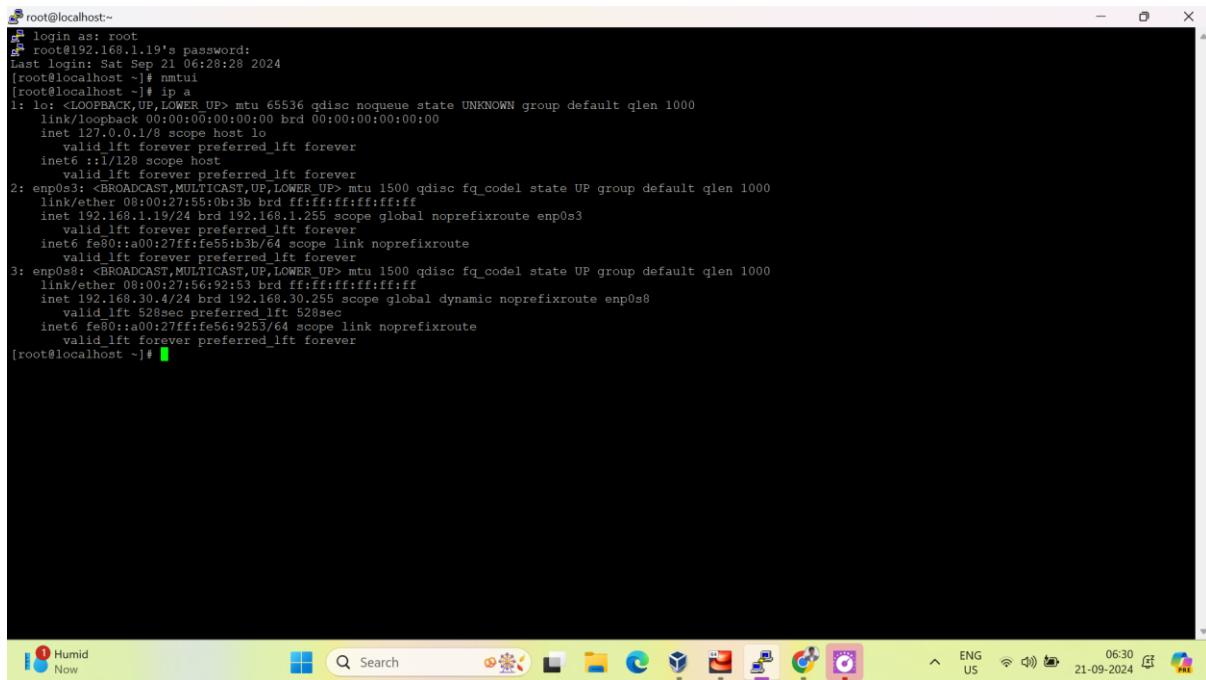
The terminal prompt is 'root@localhost ~ \$'.

WE CAN USE PUTTY SOFTWARE HEREAFTER TO GET GOOD KNOWLEDGE IN IT

### **3. DATE, SYSTEM AND OS INFORMATION**

## 1. TYPE 'date' to see the date

```
[root@localhost ~]# date  
Thursday 19 September 2024 07:56:11 PM IST  
[root@localhost ~]#
```



```
root@localhost:~#
root@92.168.1.19's password:
Last login: Sat Sep 21 06:28:28 2024
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:56:92:53 brd ff:ff:ff:ff:ff:ff
        inet 192.168.30.4/24 brd 192.168.30.255 scope global dynamic noprefixroute enp0s3
            valid_lft 528sec preferred_lft 528sec
        inet6 fe80::a00:27ff:fe56:9253/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:56:92:53 brd ff:ff:ff:ff:ff:ff
        inet 192.168.30.4/24 brd 192.168.30.255 scope global dynamic noprefixroute enp0s8
            valid_lft 528sec preferred_lft 528sec
        inet6 fe80::a00:27ff:fe56:9253/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
[root@localhost ~]#
```

2. To see the processor architecture, kernel, OS we can write a command----> uname -a

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 5.14.0-505.el9.x86_64 #1 SMP PREEMPT_DYNAMIC Thu Sep 5 07:54:07 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]# hostname Gowtham.com
[root@localhost ~]# uname -a
Linux Gowtham.com 5.14.0-505.el9.x86_64 #1 SMP PREEMPT_DYNAMIC Thu Sep 5 07:54:07 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]#
```

3. To see slightly detailed version of above write a command---> cat /proc/version

```
[root@localhost ~]# cat /proc/version
Linux version 5.14.0-505.el9.x86_64 (mockbuild@x86-05.stream.rdu2.redhat.com) (gcc (GCC) 11.5.0 20240719 (Red Hat 11.5.0-2), GNU ld version 2.35.2-54.el9) #1
SMP PREEMPT_DYNAMIC Thu Sep 5 07:54:07 UTC 2024
[root@localhost ~]#
```

4. To see the ram information---->cat /proc/meminfo

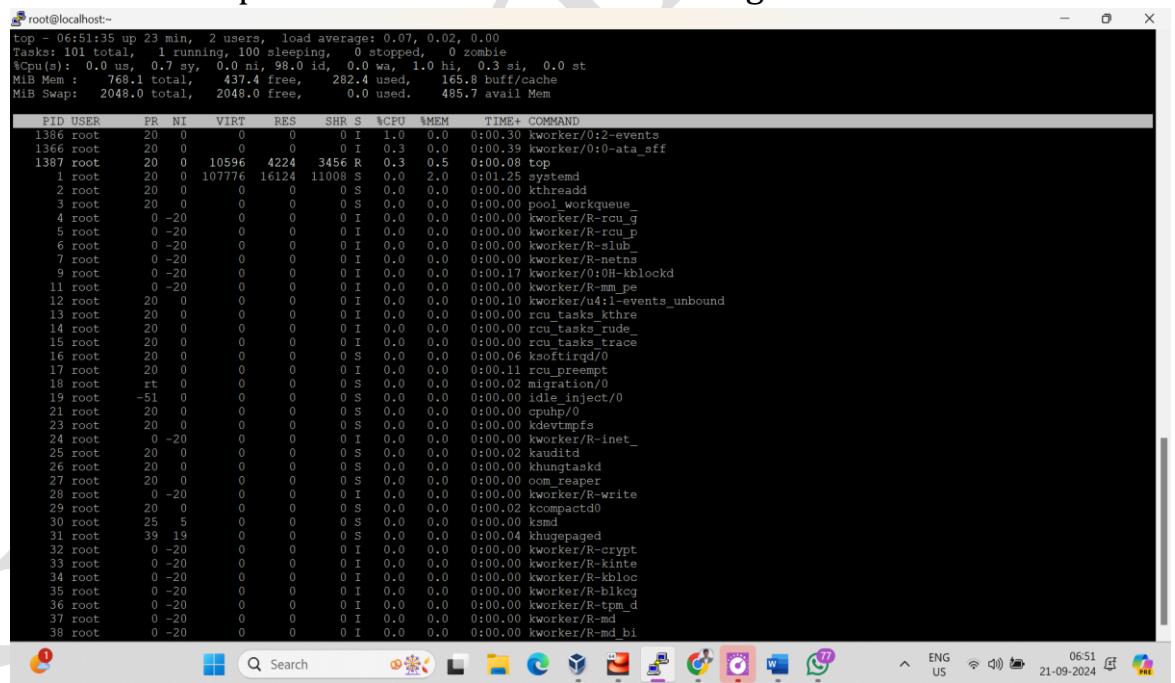
```
[root@localhost ~]# cat /proc/meminfo
MemTotal:       786564 kB
MemFree:        447920 kB
MemAvailable:   497064 kB
Buffers:         2708 kB
Cached:          141088 kB
SwapCached:      0 kB
Active:          145712 kB
Inactive:        63668 kB
Active(anon):    68704 kB
Inactive(anon):  0 kB
Active(file):    77008 kB
Inactive(file):  63668 kB
Unevictable:     0 kB
Mlocked:         0 kB
SwapTotal:       2097148 kB
SwapFree:        2097148 kB
Zswap:           0 kB
Zswapped:        0 kB
Dirty:            0 kB
Writeback:        0 kB
AnonPages:       63596 kB
Mapped:          40544 kB
Shmem:           3120 kB
KReclaimable:    25612 kB
Slab:             54820 kB
SReclaimable:    25612 kB
SUnreclaim:      29208 kB
KernelStack:     1740 kB
PageTables:      1484 kB
SecPageTables:   0 kB
NFS_Unstable:    0 kB
Bounce:           0 kB
WritebackTmp:    0 kB
CommitLimit:     2490428 kB
Committed_AS:    197700 kB
VmallocTotal:    34359738367 kB
VmallocUsed:     22208 kB
VmallocChunk:    0 kB
Percpu:          376 kB
HardwareCorrupted: 0 kB
AnonHugePages:   16384 kB
ShmemHugePages:  0 kB
ShmemPmdMapped:  0 kB
```

```

ShmemHugePages:          0 kB
ShmemPmdMapped:         0 kB
FileHugePages:          0 kB
FilePmdMapped:          0 kB
CmaTotal:                0 kB
CmaFree:                 0 kB
Unaccepted:              0 kB
HugePages_Total:         0
HugePages_Free:          0
HugePages_Rsvd:          0
HugePages_Surp:          0
Hugepagesize:            2048 kB
Hugetlb:                 0 kB
DirectMap4k:              83904 kB
DirectMap2M:              964608 kB
[root@localhost ~]#

```

5. To see the detailed version of our VM we can use a command → top so we can see like a task manager



```

root@localhost:-
top - 06:51:35 up 23 min, 2 users, load average: 0.07, 0.02, 0.00
Tasks: 101 total, 1 running, 100 sleeping, 0 stopped, 0 zombie
%CPU(s): 0.0 us, 0.7 sy, 0.0 ni, 98.0 id, 0.0 wa, 1.0 hi, 0.3 si, 0.0 st
MiB Mem : 768.1 total, 437.4 free, 282.4 used, 165.8 buff/cache
MiB Swap: 2048.0 total, 2048.0 free, 0.0 used. 485.7 avail Mem

      PID USER      PR  NI    VIRT    RES   SHR S %CPU %MEM     TIME+ COMMAND
1386 root      20   0     0   0 I  1.0  0.0  0:00.30 kworker/0:2-events
1366 root      20   0     0   0 I  0.3  0.0  0:00.39 kworker/0:0-ata_sff
1387 root      20   0 10596 4224 3456 R  0.3  0.5  0:00.08 top
  1 root      20   0 107776 16124 11008 S  0.0  2.0  0:01.25 systemd
  2 root      20   0     0   0 S  0.0  0.0  0:00.00 kthreadd
  3 root      20   0     0   0 S  0.0  0.0  0:00.00 pool_queue_
  4 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-rcu_g
  5 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-rcu_p
  6 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-slub_
  7 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-netns
  9 root      0 -20     0   0 I  0.0  0.0  0:00.17 kworker/0:0-hblockd
 11 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-mm_pe
 12 root      20   0     0   0 I  0.0  0.0  0:00.10 kworker/u4:1-events_unbound
 13 root      20   0     0   0 I  0.0  0.0  0:00.00 rcu_tasks_kthre
 14 root      20   0     0   0 I  0.0  0.0  0:00.00 rcu_tasks_rude_
 15 root      20   0     0   0 I  0.0  0.0  0:00.00 rcu_tasks_trace
 16 root      20   0     0   0 S  0.0  0.0  0:00.06 ksoftirqd/0
 17 root      20   0     0   0 I  0.0  0.0  0:00.11 rcu_preempt
 18 root      rt   0     0   0 S  0.0  0.0  0:00.02 migration/0
 19 root      -51   0     0   0 S  0.0  0.0  0:00.00 idle_inject/0
 21 root      20   0     0   0 S  0.0  0.0  0:00.00 cpuhp/0
 23 root      20   0     0   0 S  0.0  0.0  0:00.00 kdevtmpfs
 24 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-inet_
 25 root      20   0     0   0 S  0.0  0.0  0:00.02 kaudited
 26 root      20   0     0   0 S  0.0  0.0  0:00.00 khungtaskd
 27 root      20   0     0   0 S  0.0  0.0  0:00.00 oom_reaper
 28 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-write
 29 root      20   0     0   0 S  0.0  0.0  0:00.02 kompactd0
 30 root      25   5     0   0 S  0.0  0.0  0:00.00 ksm
 31 root      39   19    0   0 S  0.0  0.0  0:00.04 khugepaged
 32 root      0 -20     0   0 S  0.0  0.0  0:00.00 kworker/R-crypt
 33 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-kintte
 34 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-kblloc
 35 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-blkgd
 36 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-tpm_d
 37 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-md_
 38 root      0 -20     0   0 I  0.0  0.0  0:00.00 kworker/R-md_bci

```

6. To see the Hard Disk---->cat /proc/partitions

```
[root@gowtham ~]# cat /proc/partitions
major minor #blocks name
 8        0   20971520 sda
 8        1    1048576 sda1
 8        2   19921920 sda2
11        0    1048575 sr0
253       0   17821696 dm-0
253       1   2097152 dm-1
[root@gowtham ~]#
```

7. To see this in detail write command --> df -h

```
[root@gowtham ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/devtmpfs        4.0M   0  4.0M  0% /dev
tmpfs           385M   0  385M  0% /dev/shm
tmpfs           154M  3.1M 151M  2% /run
/dev/mapper/cs-root  17G  1.4G  16G  8% /
/dev/sda1       960M 230M  731M 24% /boot
tmpfs           77M   0   77M  0% /run/user/0
[root@gowtham ~]#
```

8. Present working directory -> pwd

```
[root@gowtham ~]# pwd
/root
[root@gowtham ~]#
```

The root directory is represented by a single forward slash (/)

9. Change Directory -> cd /

```
[root@gowtham ~]# pwd
/root
[root@gowtham ~]# cd /
[root@gowtham /]# _
```

command cd / is used to change the current working directory to the root directory. The root directory is the top-most directory in the filesystem hierarchy, represented by a single forward slash (/).

10. ls

The ls command in Linux is used to list files and directories within the file system.

```
[root@gowtham ~]# pwd
/root
[root@gowtham ~]# cd /
[root@gowtham /]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@gowtham /]# _
```

/ - slash

o It is the parent directory for all other file and folders. Only root user is having permissions to make changes.

/home

o It contains normal user profiles. By default, the normal users don't have permission to make changes in other folders.

/boot

o It contains Linux OS bootable files. GRUB2 - Boot Loader and VMLINUZ - Kernel files

/bin

o It contains binary files (commands) for the normal users.

/sbin

o It contains binary files (commands) for the root user.

/etc

o It contains system and server configuration files.

/dev

o It contains system's device file informations.

A user in the home directory has permission only in home and he is a normal user not an administrator  
Let see who is inside

```
[root@gowtham ~]# cd
[root@gowtham ~]# pwd
/root
[root@gowtham ~]# cd /
[root@gowtham /]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@gowtham /]# cd /home
[root@gowtham home]# pwd
/home
[root@gowtham home]# ls
gowtham26
[root@gowtham home]# _
```

- /proc
  - It contains systems devices process information
- /mnt
  - It is a temporary mount point for file systems.
- /media
  - It is a temporary mount point for secondary storage devices like pen drive, CD/DVD and external HD
- /tmp
  - This folder will have used system's temporary files and it will get automatically deleted after 30 days.
- /lib
  - This folder contains the process initiator files and device driver files.
- /var
  - This folder contains systems service log files.

## Basic user management:

1. we will create a user
2. We can assign password so that he can login
3. We can assign him to a group

I have switched the user to the normal user

```
[root@gowtham ~]# pwd
/root
[root@gowtham ~]# su gowtham26
[gowtham26@gowtham root]$ _
```

Let us exit from this user

```
[root@gowtham ~]# pwd
/root
[root@gowtham ~]# su gowtham26
[gowtham26@gowtham root]$ exit
exit
[root@gowtham ~]# _
```

Again we came to the root user

## We can create a new user and set a password

```
WebServer login: root
Password:
Last login: Sat Sep 21 11:04:10 on tty1
[root@WebServer ~]# useradd mahi
[root@WebServer ~]# passwd mahi
Changing password for user mahi.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@WebServer ~]# _
```

Switch user -> su mahi

Adding new user -> useradd mahi

Password setting to mahi user -> passwd mahi

Exit command is used to logout from that user account

To see the all-user details there is an command -> cat /etc/passwd

```
[root@WebServer ~]# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin/nologin
daemon:x:2:2:daemon:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:::/sbin/nologin
tss:x:59:59:Account used for TPM access::/usr/sbin/nologin
systemd-coredump:x:999:997:systemd Core Dumper::/sbin/nologin
dbus:x:81:81:System message bus::/sbin/nologin
sssd:x:998:996:User for sssd::/sbin/nologin
chrony:x:997:995:chrony system user:/var/lib/chrony:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/sbin/nologin
gowtham26:x:1000:1000:Gowtham S :/home/gowtham26:/bin/bash
mahima:x:1001:1001::/home/mahima:/bin/bash
mahi:x:1002:1002::/home/mahi:/bin/bash
[root@WebServer ~]# _
```

In Linux we no need to create a group it will automatically create let's check that by command -> cat /etc/group

```
[root@WebServer ~]# cat /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
mem:x:8:
kmem:x:9:
wheel:x:10:
cdrom:x:11:
mail:x:12:
man:x:15:
dialout:x:18:
floppy:x:19:
games:x:20:
tape:x:33:
video:x:39:
ftp:x:50:
lock:x:54:
audio:x:63:
users:x:100:
nobody:x:65534:
utmp:x:22:
utempter:x:35:
ssh_keys:x:181:
tss:x:59:
input:x:999:
kvm:x:36:
render:x:998:
systemd-journal:x:190:
systemd-coredump:x:997:
dbus:x:81:
sssd:x:996:
chrony:x:995:
sshd:x:74:
sgx:x:994:
gowtham26:x:1000:
mahima:x:1001:
mahi:x:1002:
[root@WebServer ~]#
```

Now let's create our own group

```
[root@WebServer ~]# groupadd webserver
[root@WebServer ~]# _
```

Now let's add the user to our webserver group

```
[root@WebServer ~]# groupadd webserver
[root@WebServer ~]# usermod -aG webserver mahi
[root@WebServer ~]#
```

Let's check it has added in group

```
[root@WebServer ~]# groupadd webserver
[root@WebServer ~]# usermod -aG webserver mahi
[root@WebServer ~]# cat /etc/group
root:x:0:
bin:x:1:
daemon:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
mem:x:8:
kmem:x:9:
wheel:x:10:
cdrom:x:11:
mail:x:12:
man:x:15:
dialout:x:18:
floppy:x:19:
games:x:20:
tape:x:33:
video:x:39:
ftp:x:50:
lock:x:54:
audio:x:63:
users:x:100:
nobody:x:65534:
utmp:x:22:
utempter:x:35:
ssh_keys:x:101:
tss:x:59:
input:x:999:
kvm:x:36:
render:x:998:
systemd-journal:x:190:
systemd-coredump:x:997:
dbus:x:81:
sssd:x:996:
chrony:x:995:
sshd:x:74:
sgx:x:994:
gowtham26:x:1000:
mahima:x:1001:
mahi:x:1002:
webserver:x:1003:mahi
[root@WebServer ~]#
```

It has been added but to check that user alone we have another command that help to filter that group -> cat /etc/group | grep webserver

```
[root@WebServer ~]# cat /etc/group | grep webserver
webserver:x:1003:mahi
```

I have added a another user in the webserver group

```
[root@WebServer ~]# cat /etc/group | grep webserver
webserver:x:1003:mahi
[root@WebServer ~]# useradd developer
[root@WebServer ~]# passwd developer
Changing password for user developer.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@WebServer ~]# usermod -aG webserver developer
[root@WebServer ~]# cat /etc/group | grep webserver
webserver:x:1003:mahi,developer
[root@WebServer ~]#
```

## File Management

- Type "pwd" - to check the present working directory.
- Type "ls" - to check files in the directory.
- Type "cd <provide location>" - to move other directory location.
- Type "mkdir <provide location>" - to create new directory.
- Type "touch <name>" - to create empty files.
- Type "cp <file name> /<location>/<new file name>/" - to copy the files to some other location.
- Type "cp -r <folder name> /<location>/<new folder name>/" - to copy the files to some other location.
- Type "mv <file name> /<location>/<new file name>/" - to move the files to some other location.
- Type "rm <file name>" - to delete the file.
- Type "rm -r <folder name>" - to delete the folder.

1. I am going to create a new directory in /  
We can go to the / location

```
[root@WebServer ~]# pwd  
/root  
[root@WebServer ~]# cd /  
[root@WebServer /]# ls  
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var  
[root@WebServer /]# cd /root  
[root@WebServer ~]# ls  
anaconda-ks.cfg 'E-commerce analytic platform.jpg'  
[root@WebServer ~]#
```

2. There is a command to create an new directory -> mkdir

```
[root@WebServer ~]# mkdir testDirectory  
[root@WebServer ~]# ls  
anaconda-ks.cfg 'E-commerce analytic platform.jpg' testDirectory  
[root@WebServer ~]#
```

3. Lets get into testDirectory and check what is inside

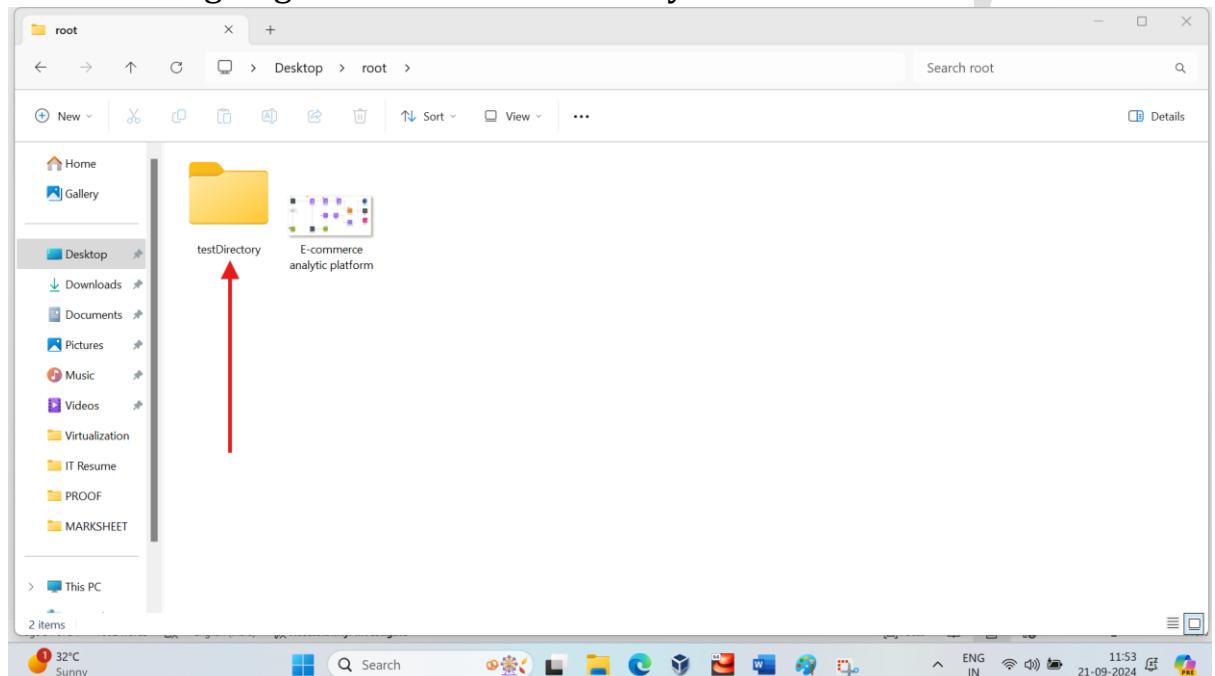
```
[root@WebServer ~]# pwd  
/root  
[root@WebServer ~]# cd testDirectory  
[root@WebServer testDirectory]# pwd  
/root/testDirectory  
[root@WebServer testDirectory]# ls  
[root@WebServer testDirectory]# _
```

There is nothing because we haven't stored any files inside the testDirectory. But if we check what is inside the root directory there, we can see testDirectory

I will explain as in windows to understand  
Lets consider root folder present here as an root directory

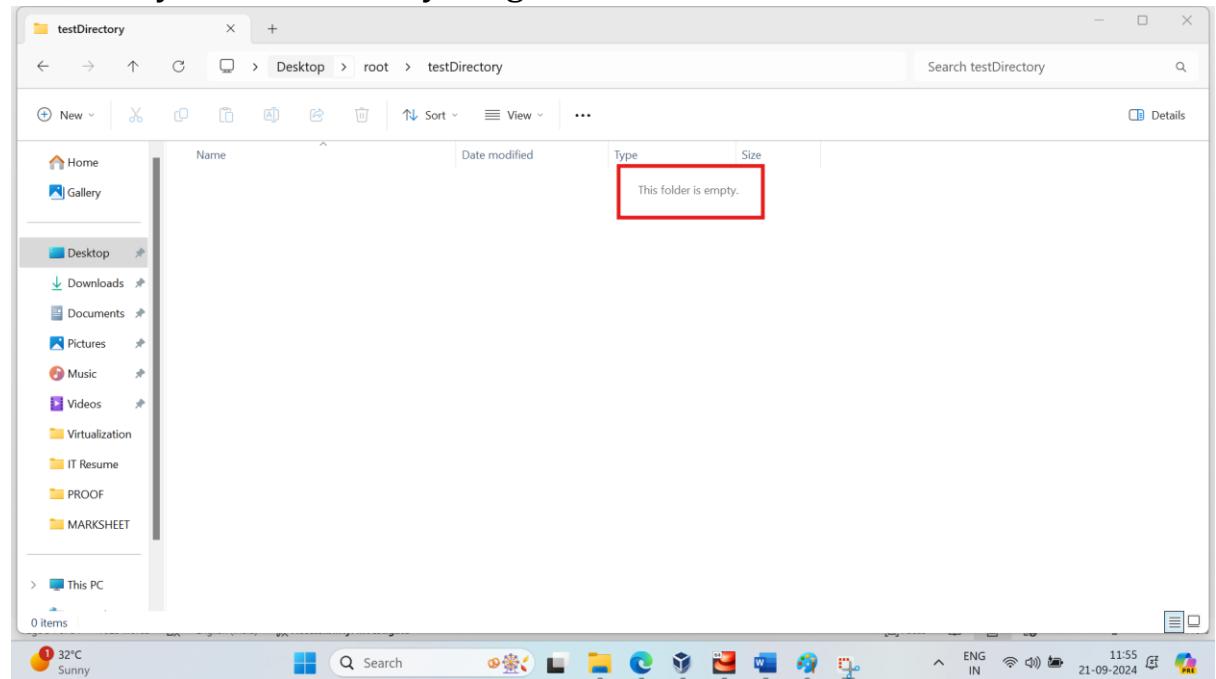
Name	Date modified	Type	Size
JAVA CODE	28-06-2024 19:42	File folder	
root	21-09-2024 11:52	File folder	
CodeBlocks	16-07-2024 10:58	Shortcut	1 KB
Microsoft Edge	27-06-2024 15:09	Shortcut	3 KB
putty	06-04-2024 09:47	Application	1,275 KB
SAB-Proctor	09-07-2024 11:59	Application	269 KB

We are now going inside the root directory



We have two files inside the root directory that are testDirectory  
and the E-commerce analytic platform  
Now lets select the testDirectory

There is an empty folder as like we have in the testDirectory there that's why it cant show anything.



I am going to create an empty file -> touch example.txt test.txt

```
CentOS Stream 9
Kernel 5.14.0-505.e19.x86_64 on an x86_64

localhost login: root
Password:
Last login: Sat Sep 21 11:04:47 on tty1
[root@localhost ~]# pwd
/root
[root@localhost ~]# cd /
[root@localhost /]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@localhost /]# cd root
[root@localhost ~]# ls
anaconda-ks.cfg 'E-commerce analytic platform.jpg' testDirectory
[root@localhost ~]# cd testDirectory
[root@localhost testDirectory]# touch examplelearn.txt examplestudy.txt
[root@localhost testDirectory]# ls
examplelearn.txt examplestudy.txt
[root@localhost testDirectory]#
```

Now the directory is not empty it has two files they are examplelearn.txt and examplestudy.txt

Now we can copy a file to another location  
Let us copy one of the file into the tmp directory  
So command is->cp examplelearn.txt /tmp

```
[root@localhost testDirectory]# ls
examplelearn.txt examplestudy.txt
[root@localhost testDirectory]# cp examplelearn.txt /tmp
```

Let us check did this file is copied to the tmp directory -> ls /tmp

```
[root@localhost testDirectory]# ls
examplelearn.txt examplestudy.txt
[root@localhost testDirectory]# cp examplelearn.txt /tmp
[root@localhost testDirectory]# ls /tmp
examplelearn.txt
systemd-private-8a3ec3116c364755a2e7bf2a17fd75c-chronyd.service-fhsd@0
systemd-private-8a3ec3116c364755a2e7bf2a17fd75c-dbus-broker.service-6MGUx@
[root@localhost testDirectory]#
```

It is present in this directory so it copied

If we need to copy the file from one location to the other location with the new name in it we need to write this command

->cp examplelearn.txt /tmp/new.txt

```
[root@localhost ~]# cd /
[root@localhost ~]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@localhost ~]# cd tmp
[root@localhost tmp]# ls
examplelearn.txt
new.txt
systemd-private-534d9b05e3d24fdb2baaa98d6f1ceae-chronyd.service-MiyTMB
[root@localhost tmp]#
```

If we want to copy the directory then there is a command

->cp -r testDirectory/tmp

```
[root@localhost tmp]# pwd
/tmp
[root@localhost tmp]# cd ..
[root@localhost ~]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@localhost ~]# cd /root
[root@localhost ~]# ls
anaconda-ks.cfg 'E-commerce analytic platform.jpg' testDirectory
[root@localhost ~]# cp -r testDirectory /tmp
[root@localhost ~]# cd ..
[root@localhost ~]# cd tmp
[root@localhost tmp]# ls
examplelearn.txt
new.txt
systemd-private-534d9b05e3d24fdb2baaa98d6f1ceae-chronyd.service-MiyTMB
systemd-private-534d9b05e3d24fdb2baaa98d6f1ceae-dbus-broker.service-4Kymxf
[root@localhost tmp]#
```

We have copied the entire directory to the other directory

Now we can move the examplelearn.txt from the tmp directory to the root directory

->mv /tmp/examplelearn.txt /root

```
[root@localhost tmp]# mv /tmp/examplelearn.txt /root
[root@localhost tmp]# ls /root
anaconda-ks.cfg 'E-commerce analytic platform.jpg' examplelearn.txt testDirectory
[root@localhost tmp]#
```

->mv /tmp/testDirectory/examplelearn.txt /root

Now we can rename a file with same above command

```
->mv /tmp/testDirectory/examplelearn.txt  
/tmp/testDirectory/examlearn.txt
```

```
[root@localhost tmp]# pwd  
/tmp  
[root@localhost tmp]# cd ..  
[root@localhost ~]# ls  
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys [tmp] usr var  
[root@localhost ~]# mv /tmp/testDirectory/examplelearn.txt /tmp/testDirectory/examlearn.txt  
[root@localhost ~]# cd tmp  
[root@localhost tmp]# ls  
new.txt  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-chrony.service-MigTMB systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-kdump.service-SHD2Wb  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-dbus-broker.service-4KymxF systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-systemd-logind.service-YzSOK1  
[root@localhost tmp]# cd testDirectory  
[root@localhost testDirectory]# ls  
examlearn.txt examplestudy.txt  
[root@localhost testDirectory]#
```

Now we can write a command to delete the file examlearn.txt the command is -> rm /tmp/testDirectory/examlear.txt

```
[root@localhost ~]# pwd  
/root  
[root@localhost ~]# cd /  
[root@localhost ~]# ls  
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys [tmp] usr var  
[root@localhost ~]# cd tmp  
[root@localhost tmp]# ls  
new.txt  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-chrony.service-MigTMB systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-kdump.service-SHD2Wb  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-dbus-broker.service-4KymxF systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-systemd-logind.service-YzSOK1  
[root@localhost tmp]# touch creatingDeleting.txt  
[root@localhost tmp]# ls  
creatingDeleting.txt  
new.txt  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-chrony.service-MigTMB systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-kdump.service-SHD2Wb  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-dbus-broker.service-4KymxF systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-systemd-logind.service-YzSOK1  
[root@localhost tmp]# rm /tmp/creatingDeleting.txt  
rm: remove regular empty file '/tmp/creatingDeleting.txt'? y  
[root@localhost tmp]# ls  
new.txt  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-chrony.service-MigTMB systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-kdump.service-SHD2Wb  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-dbus-broker.service-4KymxF systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-systemd-logind.service-YzSOK1  
[root@localhost tmp]# _
```

Now we can delete the directory named testDirectory which is present in the tmp directory

command -> rm -r /tmp/testDirectory

```
[root@localhost tmp]# rm -r /tmp/testDirectory  
rm: descend into directory '/tmp/testDirectory'? y  
rm: remove regular empty file '/tmp/testDirectory/examlestudy.txt'? y  
rm: remove directory '/tmp/testDirectory'? y  
[root@localhost tmp]# ls /tmp  
new.txt  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-chrony.service-MigTMB systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-kdump.service-SHD2Wb  
systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-dbus-broker.service-4KymxF systemd-private-534d9b05e3d24dfdb2baaa98d6f1ceae-systemd-logind.service-YzSOK1  
[root@localhost tmp]# _
```

## File Management – File Editing

Cat -> we can read the file and also we can edit the file lets see how It works.

Command to learn the file -> cat examplelearn.txt

```
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
anaconda-ks.cfg  'E-commerce analytic platform.jpg'  examplelearn.txt  testDirectory
[root@localhost ~]# cat examplelearn.txt
[root@localhost ~]# _
```

There is no output because it is an empty file lets write some thing inside for this there is an command - > cat > examplelearn.txt

After writing this we can give the ctrl + d to save the file lets see did it works

```
[root@localhost ~]# cat > examplelearn.txt
Hi I am Gowtham -> Cloud Engineer...[root@localhost ~]# cat examplelearn.txt
Hi I am Gowtham -> Cloud Engineer...[root@localhost ~]# _
```

Now lets try to add an new sentence or a line in the command and check what happens

```
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
anaconda-ks.cfg  'E-commerce analytic platform.jpg'  examplelearn.txt  I  testDirectory
[root@localhost ~]# cat examplelearn.txt
Hi I am Gowtham -> Cloud Engineer...[root@localhost ~]#
[root@localhost ~]# cat > examplelearn.txt
I am doing my bachelors in bannari amman institute of technology
[root@localhost ~]# cat examplelearn.txt
I am doing my bachelors in bannari amman institute of technology
[root@localhost ~]#
```

Now we have lost our old data new data has replaced the old data

Command to write new sentence without replacing is

->cat >> examplelearn.txt

```
[root@localhost ~]# cat >> examplelearn.txt
I am acquiring a knowledge in the linux
[root@localhost ~]# cat examplelearn.txt
I am doing my bachelors in bannari amman institute of technology
I am acquiring a knowledge in the linux
[root@localhost ~]#
```

Now we can see an another editor called as vi editor to open this write a command called -> vi examplelearn.txt



```
I am doing my bachelors in bannari amman institute of technology
I am acquiring a knowledge in the linux
```

We have opened the editor now we can insert a line by pressing a letter 'i'.



```
I am doing my bachelors in bannari amman institute of technology
I am acquiring a knowledge in the linux
```

-----  
-- INSERT --

After writing press the esc button to come out of editor

```
I am Gowtham  
I am doing my bachelors in bannari amman institute of technology  
I am acquiring a knowledge in the linux  
-- INSERT --
```

```
I am Gowtham  
I am doing my bachelors in bannari amman institute of technology  
I am acquiring a knowledge in the linux  
:wq! <enter>
```

w-> Denotes that I have completed writing

q-> Denotes that to quite the editor

!-> Denotes that a confirmation

```
[root@localhost ~]# cat examplelearn.txt
I am Gowtham
I am doing my bachelors in bannari amman institute of technology
I am acquiring a knowledge in the linux
[root@localhost ~]#
```

Now we can see that data has been added

We can also set the numbers to every line with the help to vi command lets see how it is

The screenshot shows a terminal window with a black background and white text. At the bottom of the screen, there is a red rectangular box highlighting the command ':set nu' which is being typed into the terminal. Above this, the terminal displays three lines of text: '1 I am Gowtham', '2 I am doing my bachelors in bannari amman institute of technology', and '3 I am acquiring a knowledge in the linux'. The terminal prompt '[root@localhost ~]#' is visible at the bottom right.

```
1 I am Gowtham
2 I am doing my bachelors in bannari amman institute of technology
3 I am acquiring a knowledge in the linux
:set nu
```

Now we can search a word in the vi editor with the command -> /\_\_\_\_\_ fill what word need to search in the command

```
I am Gowtham  
I am doing my bachelors in bannari amman institute of technology  
I am acquiring a knowledge in the linux  
I am fond of new technology
```

```
search hit BOTTOM, continuing at TOP
```

I have searched a word called technology then we can see the the next technology word is there by only pressing the 'n'.

```
I am Gowtham  
I am doing my bachelors in bannari amman institute of technology  
I am acquiring a knowledge in the linux  
I am fond of new technology  
  
/technology
```

Now cursor have moved to the next occurrence

## Permissions viewing for users

Paylagam

- How to check a file permission?
  - Type "ls -l <filename>" - to see the file permissions
- How to understand the file permission?
  - -rw-r--r--. 1 root root 11 Oct 14 20:29 new.txt

-	rw-	r--	r--	.	1	root	root	11	Oct 14 20:29	new.txt
File Type	Owner / User Permission	Group Permission	Other User Permission	Special permission	Hard Link number	Owner name	Group name	File Size in bytes	File Modified time	File name

```
/root
[root@localhost ~]# ls -l
total 100
-rw-----. 1 root root 870 Sep 14 07:27 anaconda-ks.cfg
-rw-r--r--. 1 root root 92653 Aug 6 16:24 'E-commerce analytic platform.jpg'
-rw-r--r--. 1 root root 147 Sep 23 12:03 examplelearn.txt
-rw-r--r--. 1 root root 0 Sep 23 11:20 I
drwxr-xr-x. 2 root root 54 Sep 21 13:47 testDirectory
[root@localhost ~]_
```

We can see how to change a file permission

Two types of method are there to change the permission

1. Symbolic method
2. Numeric method

- How to change a file permission?
  - We have two ways to change file permissions
    - Symbolic type | "+" "-" "=" | r=read, w=write, x=execute | u=user, g=group, o=others
      - Example : chmod g-w /home/testuser/test.txt
    - Numeric type | "0=no value, 1=execute, 2=write, 4=read"
      - Example : chmod 641 /home/testuser/test.txt
- How to change a folder permission?
  - Example : chmod -R u-w,g-w <directory location>
  - Example : chmod -R 440 <directory location>

+ -> We can increment the permission

- -> We can decrement the permission

= -> Assign the same permission as it is without any change

If we want to remove the write permission in the group then the command -> chmod g-w /tmp/testDirectory/examplelearn.txt

We can remove the read permission for others in the examplelearn.txt

### Before Removing Permission

```
/root
[root@localhost ~]# ls -l
total 100
-rw----- 1 root root 870 Sep 14 07:27 anaconda-ks.cfg
-rw-r--r-- 1 root root 92653 Aug 6 16:24 'E-commerce analytic platform.jpg'
-rw-r--r-- 1 root root 147 Sep 23 12:03 examplelearn.txt
-rw-r--r-- 1 root root 0 Sep 23 11:20 I
drwxr-xr-x 2 root root 54 Sep 21 13:47 testDirectory
[root@localhost ~]#
```

### After Removing Permission

```
[root@localhost ~]# ls
afs bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
[root@localhost ~]# cd /root
[root@localhost ~]# ls
anaconda-ks.cfg 'E-commerce analytic platform.jpg' examplelearn.txt I testDirectory
[root@localhost ~]# chmod o-r examplelearn.txt
[root@localhost ~]# ls -l
total 100
-rw----- 1 root root 870 Sep 14 07:27 anaconda-ks.cfg
-rw-r--r-- 1 root root 92653 Aug 6 16:24 'E-commerce analytic platform.jpg'
-rw-r----- 1 root root 147 Sep 23 12:03 examplelearn.txt
-rw-r--r-- 1 root root 0 Sep 23 11:20 I
drwxr-xr-x 2 root root 54 Sep 21 13:47 testDirectory
[root@localhost ~]#
```

I have removed the permission for others to read the examplelearn.txt

```
[root@localhost ~]# su gowtham26
[gowtham26@localhost root]$ cat examplelearn.txt
cat: examplelearn.txt: Permission denied
[gowtham26@localhost root]$
```

If a gowtham26 user tried to access the file the user gets a permission denied

Now lets give the permission again with the help of the command

->chmod o+r examplelearn.txt

```
[gowtham26@localhost tmp]$ exit
exit
[root@localhost tmp]# pwd
/tmp
[root@localhost tmp]# ls
examplelearn.txt
new.txt
systemd-private-aec793c9766f44aab3e4e533a29f3f1-chromyd.service-AQgaU
[root@localhost tmp]# chmod o+r examplelearn.txt
[root@localhost tmp]# ls -l
total 4
-rw-r--r-- 1 root root 147 Sep 23 12:03 examplelearn.txt
-rw-r--r-- 1 root root 0 Sep 23 09:03 new.txt
drwx----- 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-chromyd.service-AQgaU
drwx----- 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-dbus-broker.service-Zcx3FI
drwx----- 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-kdump.service-toX8m
drwx----- 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-systemd-logind.service-8U3brh
[root@localhost tmp]# chmod o+r examplelearn.txt
[root@localhost tmp]# su gowtham26
[gowtham26@localhost tmp]$ cat examplelearn.txt
I am Gowtham
I am doing my bachelors in bambali amman institute of technology
I am acquiring a knowledge in the linux
I am fond of new technology
[gowtham26@localhost tmp]$
```

## In Numeric method

- Numeric type | "0=no value, 1=execute, 2=write, 4=read"
  - Example : chmod 641 /home/testuser/test.txt

6 stands for the permission given to the user the permissions include the user has an permission to read and write as  $2 + 4 = 6$ .

4 stand for the permission given to the group the permission include the group has an permission to read alone.

1 stands for permission given to the others the others permission include the other has an permission to execute alone.

Now for the examplelearn.txt file we can give the full permission and the command is -> chmod 777 /tmp/examplelearn.txt

```
[root@localhost tmp]# chmod ??? /tmp/examplelearn.txt
[root@localhost tmp]# ls -l
total 4
-rwxrwxrwx. 1 root root 147 Sep 23 12:03 examplelearn.txt
-rw-r--r--. 1 root root 0 Sep 23 09:03 new.txt
drwx-----. 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-chromod.service-ACQgaU
drwx-----. 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-dbus-broker.service-Zcx3FI
drwx-----. 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-kdump.service-toXr8m
drwx-----. 3 root root 17 Sep 23 13:38 systemd-private-aec793c9766f44aab3e4e533a29f3f1-systemd-logind.service-8U3brh
[root@localhost tmp]# _
```

## **Package Management**

It is installation, updating, removing of a software this is called as package management.

yum and DNF are package management tools for Linux mainly red hat systems like fedora & CentOS.

YUM – yellowdog updater modified

- >Mainly use this
- >It has all the dependencies and required software automatically.

```
yum install package_name.  
Yum update package_name  
Yum remove package_name
```

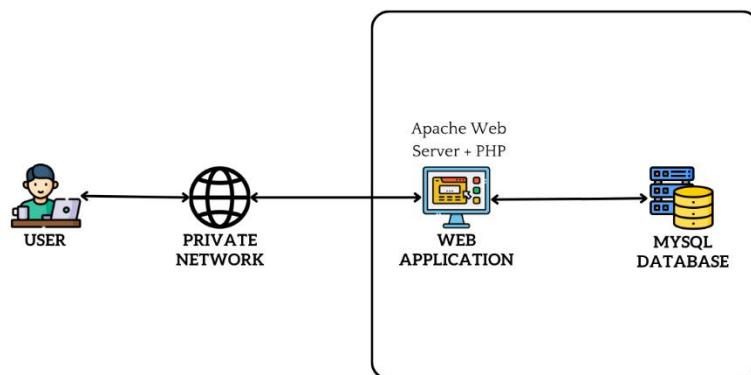
DNF – updated version of YUM – Dandified YUM

- >Faster
- >Made with recent architecture
- >Better to use DNF then yum.

Steps to follow :

- Step 1 : Install and Configure Apache Httpd and PHP service in web server host
- Step 2 : Install and Configure MySQL Database in DB host.
- Step 3 : Design your **Application** in PHP
- Step 4 : Write your Application code in PHP
- Step 5 : Understand, how the Application works in local environment?
- Step 6 : Deploy & Test your Application !

### FLOW CHART FOR THE PROJECT OF USER DETAIL COLLECTOR



### HOW APPLICATION WORKS

- >THERE IS AN DATA ENTRY TEAM WHO ENTERS THE DATA OF THE PATIENTS IN THE HOSPITAL
- >THERE IS A WEWB SERVER WHICH IS IN A PRIVATE NETWORK WHICH MEANS ONLY THE DATA ENTRY TEAM CONNECTED IN THE PRIVATE NETWORK ONLY HAS A PERMISSION TO ACCESS IT.

-> IN ABOVE WE HAVE CREATED TWO VIRTUAL MACHINES IN WHICH ONE IS GOING TO ACT AS A WEB SERVER AND ANOTHER IS GOING TO ACT AS AN DATA BASE SERVER.

-> THE DATA ENTRY TEAM DOES NOT HAVE AN PERMISSION TO ACCESS THE DATABASE SERVER BECAUSE THEY DOESN'T KNOW THE IP ADDRESS OF THE DATA BASE SERVER

-> THEY ENTER THE DATA IN THE WEB SERVER WHICH SEND THE DATA TO THE DATABASE SERVER THEN THIS DATA CAN BE SEEN BY THE DATA ENTRY TEAM WITH THE HELP OF THE WEB SERVER.

FLOW CHART:



# Deploy the Application

Step 1 : Prepare the MySQL DB for our PHP Application.

Step 2 : Ensure the connectivity with PHP - DB.

Step 3 : Paste the PHP code in required location.

Step 4 : Restart the required services and check the service status.

Step 5 : Access the PHP application from user end.

## STARTING THE PROJECT

1. ON THE VIRTUAL MACHINE AND SET THE PERMANENT HOSTNAME.  
FOR THAT WE SHOULD EDIT IN hostname FILE  
I WILL DO THIS THROUGH THE VI EDITOR

```
[root@localhost ~]# hostname www.userdetailscollection.com
[root@localhost ~]# vi /etc/hostname
```

2. INSERT THE HOSTNAME IN THIS FILE  
TO INSERT THE HOSTNAME FIRST PRESS 'i'

-- INSERT --

3. To save this file press esc and write a command ':wq!'.
  4. NOW HOSTNAME IS SET PERMENENTALY
  5. NOW WE CAN INSTALL THE APACHE WEB SERVER  
COMMAND -> yum install httpd -y
  6. THEN WE NEED TO INSTALL THE PHP AND DEPENDENCIES  
Command -> dnf module list php
    - >dnf module -y enable php:8.1
    - >dnf module -y install php:8.1/comman

## 7. WE ALSO NEED TO INSTALL MYSQL AND CONNECT MYSQL WITH THE PHP.

COMMAND-> yum install mysql -y

-> yum install php-mysqcli -y

## I HAVE INSTALLED THE APACHE WEB SERVER

```
Total                                         608 kB/s | 3.7 MB   00:05
CentOS Stream 9 - BaseOS                         112 kB/s | 1.6 KB   00:00

Importing GPG key 0x8483C65D:
Userid      : "CentOS (CentOS Official Signing Key) <security@centos.org>"
Fingerprint: 99DB 70FA E1D7 CE2Z 7FB6 4082 0B5B 55B3 8483 C65D
From        : /etc/pki/rpm-gpg/RPM-GPG-KEY-centosofficial
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing:
  apr-1.7.0-12.el9.x86_64          1/1
  apr-util-bdb-1.6.1-23.el9.x86_64  1/12
  apr-util-openssl-1.6.1-23.el9.x86_64 2/12
  apr-util-1.6.1-23.el9.x86_64      3/12
  httpd-tools-2.4.62-1.el9.x86_64   4/12
  httpd-filesystem-2.4.62-1.el9.noarch 5/12
  httpd-filesystem-2.4.62-1.el9.noarch 6/12
  centos-logos-httpd-90.8-1.el9.noarch 7/12
  mailcap-2.1.49-5.el9.noarch       8/12
  httpd-core-2.4.62-1.el9.x86_64    9/12
  mod_lua-2.4.62-1.el9.x86_64      10/12
  mod_http2-2.0.26-2.el9.x86_64    11/12
  httpd-2.4.62-1.el9.x86_64        12/12
  httpd-2.4.62-1.el9.x86_64        12/12
Running scriptlet: httpd-filesystem-2.4.62-1.el9.noarch
Running scriptlet: httpd-filesystem-2.4.62-1.el9.noarch
Installing : httpd-logos-httpd-90.8-1.el9.noarch
Installing : mailcap-2.1.49-5.el9.noarch
Installing : httpd-core-2.4.62-1.el9.x86_64
Installing : mod_lua-2.4.62-1.el9.x86_64
Installing : mod_http2-2.0.26-2.el9.x86_64
Installing : httpd-2.4.62-1.el9.x86_64
Running scriptlet: httpd-2.4.62-1.el9.x86_64
[ 570.878732] systemd-rc-local-generator[14931]: /etc/rc.d/rc.local is not marked executable, skipping.
Verifying   : mailcap-2.1.49-5.el9.noarch
Verifying   : apr-1.7.0-12.el9.x86_64
Verifying   : apr-util-1.6.1-23.el9.x86_64
Verifying   : apr-util-bdb-1.6.1-23.el9.x86_64
Verifying   : apr-util-openssl-1.6.1-23.el9.x86_64
Verifying   : centos-logos-httpd-90.8-1.el9.noarch
Verifying   : httpd-2.4.62-1.el9.x86_64
Verifying   : httpd-core-2.4.62-1.el9.x86_64
Verifying   : httpd-filesystem-2.4.62-1.el9.noarch
Verifying   : httpd-tools-2.4.62-1.el9.x86_64
Verifying   : mod_lua-2.4.62-1.el9.x86_64
Verifying   : mod_http2-2.0.26-2.el9.x86_64
Installed products updated.

Installed:
  apr-1.7.0-12.el9.x86_64           apr-util-1.6.1-23.el9.x86_64           apr-util-bdb-1.6.1-23.el9.x86_64           apr-util-openssl-1.6.1-23.el9.x86_64
  centos-logos-httpd-90.8-1.el9.noarch  httpd-2.4.62-1.el9.x86_64           httpd-core-2.4.62-1.el9.x86_64           httpd-filesystem-2.4.62-1.el9.noarch
  httpd-tools-2.4.62-1.el9.x86_64      mailcap-2.1.49-5.el9.noarch           mod_http2-2.0.26-2.el9.x86_64           mod_lua-2.4.62-1.el9.x86_64

Complete!
[root@www ~]#
```

## WE CAN LIST THE MODULE IN PHP -> dnf module list php

```
[root@www ~]# dnf module list php
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use "rhc" or "subscription-manager" to register.

Last metadata expiration check: 0:03:48 ago on Wednesday 25 September 2024 06:13:55 AM.
CentOS Stream 9 - AppStream
Name          Stream          Profiles          Summary
php           8.1             common [d], devel, minimal  PHP scripting language
php           8.2             common [d], devel, minimal  PHP scripting language

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

WE HAVE 8.1 LET'S ENABLE THAT -> dnf module -y enable php:8.1

```
[root@www ~]# dnf module -y enable php:8.1
Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use "rhc" or "subscription-manager" to register.

Last metadata expiration check: 0:05:02 ago on Wednesday 25 September 2024 06:13:55 AM.
Dependencies resolved.

=====
Package           Architecture      Version       Repository      Size
=====
Enabling module streams:
php               x86_64          8.1          @System     8.1 KB

Transaction Summary
=====
Complete!
```

WE CAN INSTALL THE PHP 8.1 -> dnf module -y install php:8.1/common

```
Transaction Summary
=====
Install  7 Packages

Total download size: 6.8 M
Installed size: 35 M
Downloading Packages:
(1/7): nginx-filesystem-1.20.1-20.e19.noarch.rpm           25 kB/s | 9.2 kB   00:00
(2/7): libxslt-1.1.34-9.e19.x86_64.rpm                   282 kB/s | 243 kB   00:00
(3/7): php-common-8.1.27-1.module_e19+790+4812d76d.x86_64.rpm 339 kB/s | 692 kB   00:02
(4/7): php-fpm-8.1.27-1.module_e19+790+4812d76d.x86_64.rpm 877 kB/s | 1.8 MB   00:02
(5/7): php-mbstring-8.1.27-1.module_e19+790+4812d76d.x86_64.rpm 432 kB/s | 473 kB   00:01
(6/7): php-xal-8.1.27-1.module_e19+790+4812d76d.x86_64.rpm 119 kB/s | 146 kB   00:01
(7/7): php-cli-8.1.27-1.module_e19+790+4812d76d.x86_64.rpm 666 kB/s | 3.5 MB   00:05

Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing :                                                 1/1
  Installing : php-common-8.1.27-1.module_e19+790+4812d76d.x86_64 1/7
  Running scriptlet: nginx-filesystem-2:1.20.1-28.e19.noarch 2/7
  Installing : libxslt-1.1.34-9.e19.x86_64                  2/7
  Installing : php-xml-0.1.27-1.module_e19+790+4812d76d.x86_64 3/7
  Installing : php-fpm-8.1.27-1.module_e19+790+4812d76d.x86_64 4/7
  Running scriptlet: php-fpm-8.1.27-1.module_e19+790+4812d76d.x86_64 5/7
  Installing : php-cli-8.1.27-1.module_e19+790+4812d76d.x86_64 6/7
  Installing : php-mbstring-8.1.27-1.module_e19+790+4812d76d.x86_64 7/7
  Running scriptlet: php-mbstring-8.1.27-1.module_e19+790+4812d76d.x86_64
[ 1124.572633] systemd-rc-local-generator[11460]: /etc/rc.d/rc.local is not marked executable, skipping.
  Verifying  : libxslt-1.1.34-9.e19.x86_64                1/7
  Verifying  : nginx-filesystem-2:1.20.1-28.e19.noarch    2/7
  Verifying  : php-cli-8.1.27-1.module_e19+790+4812d76d.x86_64 3/7
  Verifying  : php-common-8.1.27-1.module_e19+790+4812d76d.x86_64 4/7
  Verifying  : php-fpm-8.1.27-1.module_e19+790+4812d76d.x86_64 5/7
  Verifying  : php-mbstring-8.1.27-1.module_e19+790+4812d76d.x86_64 6/7
  Verifying  : php-xml-8.1.27-1.module_e19+790+4812d76d.x86_64 7/7
Installed products updated.

Installed:
  libxslt-1.1.34-9.e19.x86_64          nginx-filesystem-2:1.20.1-28.e19.noarch      php-cli-8.1.27-1.module_e19+790+4812d76d.x86_64
  php-common-8.1.27-1.module_e19+790+4812d76d.x86_64  php-fpm-8.1.27-1.module_e19+790+4812d76d.x86_64  php-mbstring-8.1.27-1.module_e19+790+4812d76d.x86_64
  php-xml-8.1.27-1.module_e19+790+4812d76d.x86_64

Complete!
[root@www ~]
```

LETS CHECK THE VERSION OF PHP->php -v

```
[root@www ~]# php -v
PHP 8.1.27 (cli) (built: Dec 19 2023 20:35:55) (NTS gcc x86_64)
Copyright (c) The PHP Group
Zend Engine v4.1.27, Copyright (c) Zend Technologies
```

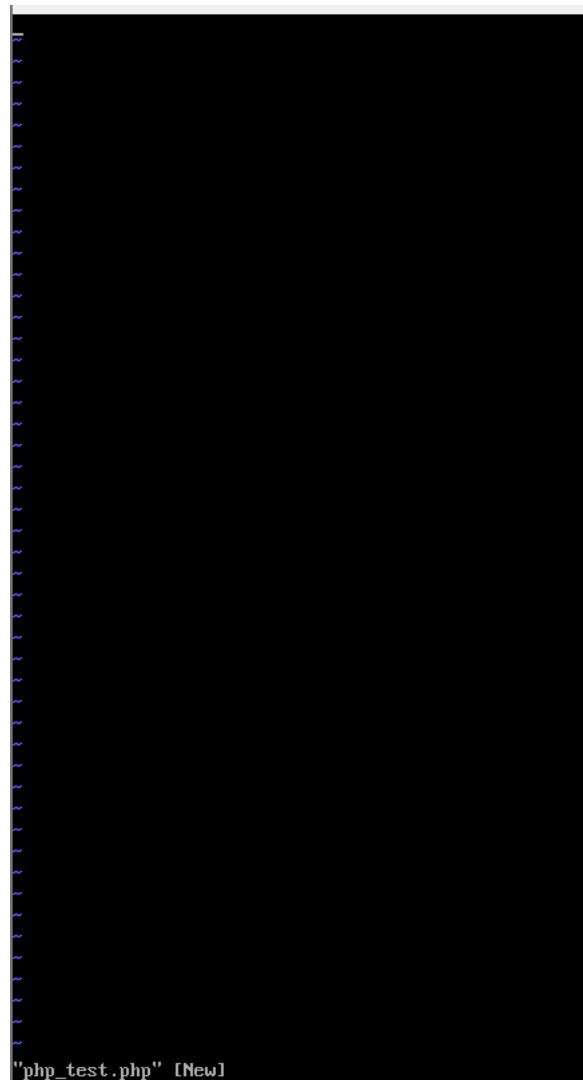
WE CAN ENABLE THIS SERVICE TO MAKE THIS ALIVE WHENEVER WE ON THE SYSTEM -> systemctl enable --now httpd

```
[root@www ~]# systemctl enable --now httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ 1512.214560] systemd-rc-local-generator[11651]: /etc/rc.d/rc.local is not marked executable, skipping.
```

NOW WE NEED TO WRITE A PHP CODE IN ONE FILE THE FILE IS A HTML FILE WHICH IS LOCATED IN -> cd /var/www/html

```
[root@www ~]# cd /var/www/html  
[root@www html]#
```

ONE THE VI EDITOR TO WRITE THE CODE->vi php\_test.php



```
[root@www html]# vi php_test.php
```

WRITE A CODE IN THIS VI

```
<!DOCTYPE html>

<html>
<body>
<h1>My first PHP page</h1>
<?php
echo "Hello World!";
?>
</body>
</html>
```



```
<!DOCTYPE html>
<html>
<body>
<h1>My first PHP page</h1>
<?php
echo "Hello World!";
?>
</body>
</html>
~
```

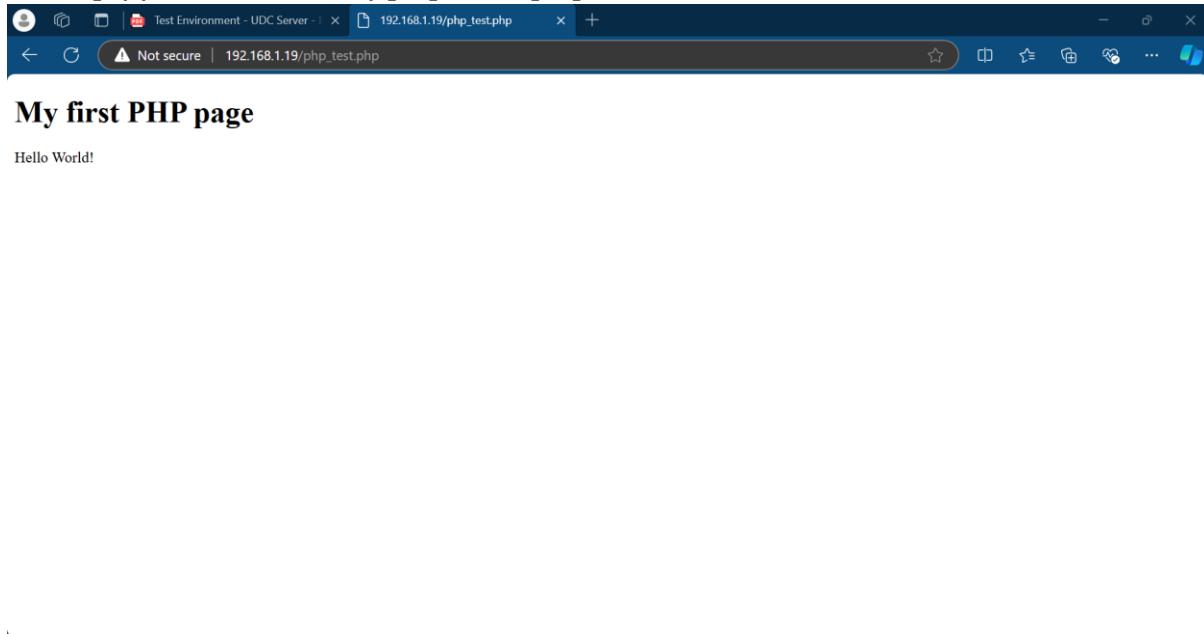
-- INSERT --

WRITE A COMMAND TO STOP THE FIREWALL AS OUR PAGE IS NOT LOADING

```
[root@www html]# setenforce 0  
[root@www html]# systemctl stop firewalld
```

NOW WE CAN SEE IN THE 192.168.1 SERIES WEB BROWSER

-> [http://192.168.1.19/php\\_test.php](http://192.168.1.19/php_test.php)



NOW WE HAVE DEPLOYED THE WEB SERVER WHICH CAN BE VIEWED BY PRIVATE NETWORK.