

****Secure Multi-User File Sharing Server on a LAN****

The **Secure Multi-User File Sharing Server on a LAN** project is designed to simulate a real-world office environment where multiple users, grouped by department (HR, IT, Sales, etc.), securely share files over a local area network. The project is implemented on **CentOS Stream 9**, using fundamental Linux administration techniques such as user and group management, permission controls, secure file transfer, backups, and system monitoring.

The server enables users to securely transfer and access departmental files while restricting unauthorized access using proper **file system permissions**. It also includes scheduled backups using **archiving and compression tools**, and uses **SSH/SCP** for secure communication. System monitoring and package management ensure optimal performance and control.

Tools & Technologies Used

- **Operating System:** CentOS Stream 9
 - **Virtualization:** VirtualBox
 - **Network Configuration:** Internal Network (VirtualBox)
 - **User & Group Management:** useradd, groupadd, passwd, usermod
 - **File Permissions:** chmod, chown, umask
 - **Secure File Transfer:** SSH (sshd), SCP
 - **Archiving & Compression:** tar, gzip
 - **Process Monitoring:** top, free -m, df -h
 - **Package Manager:** dnf (replacement for yum)
 - **Firewall Management:** firewalld
-

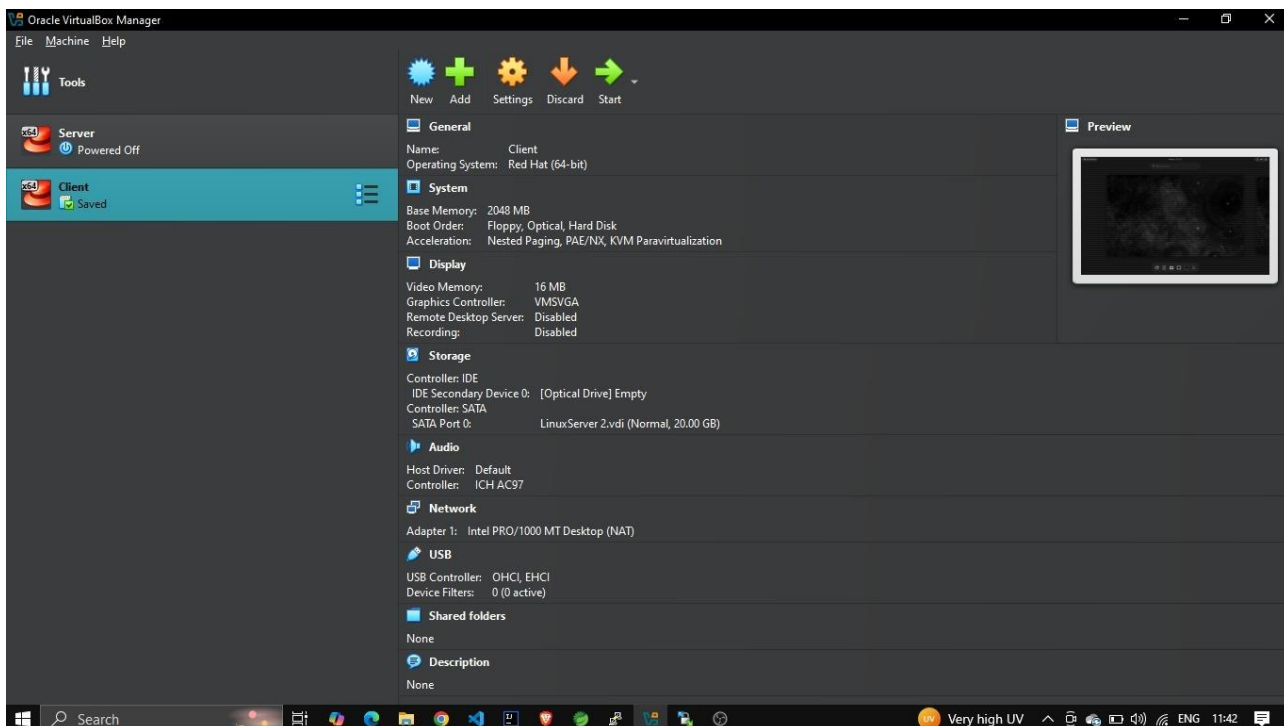
Key Features

- **Multi-user and group-based access:** Users are organized into groups representing departments. File access is restricted based on group membership using Linux permissions.
- **Secure file transfers:** Users connect to the server and transfer files using **SSH** and **SCP**, ensuring encrypted communication.
- **Department-wise shared directories:** Each department has its own folder with read/write access for group members and restricted access for others.

- **System monitoring:** Admins can monitor system health and performance using tools like top, df -h, and free -m.
 - **Package and service management:** Uses yum to install and manage essential services and tools.
 - **Firewall configuration:** Configured to allow SSH access while maintaining a secure internal environment.
-

PART 1: SETUP - 2 Virtual Machines

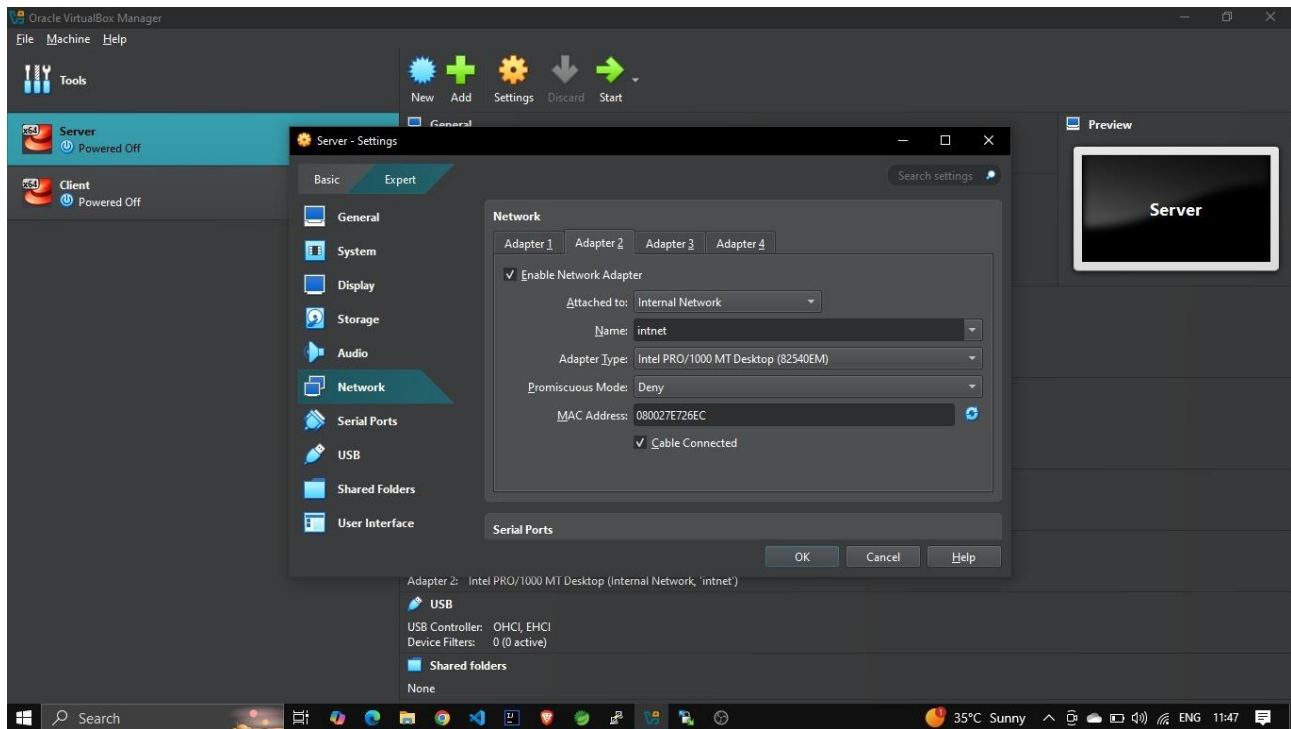
1. Create one CentOS 9 VM (main server)
2. Clone the VM (client machine)



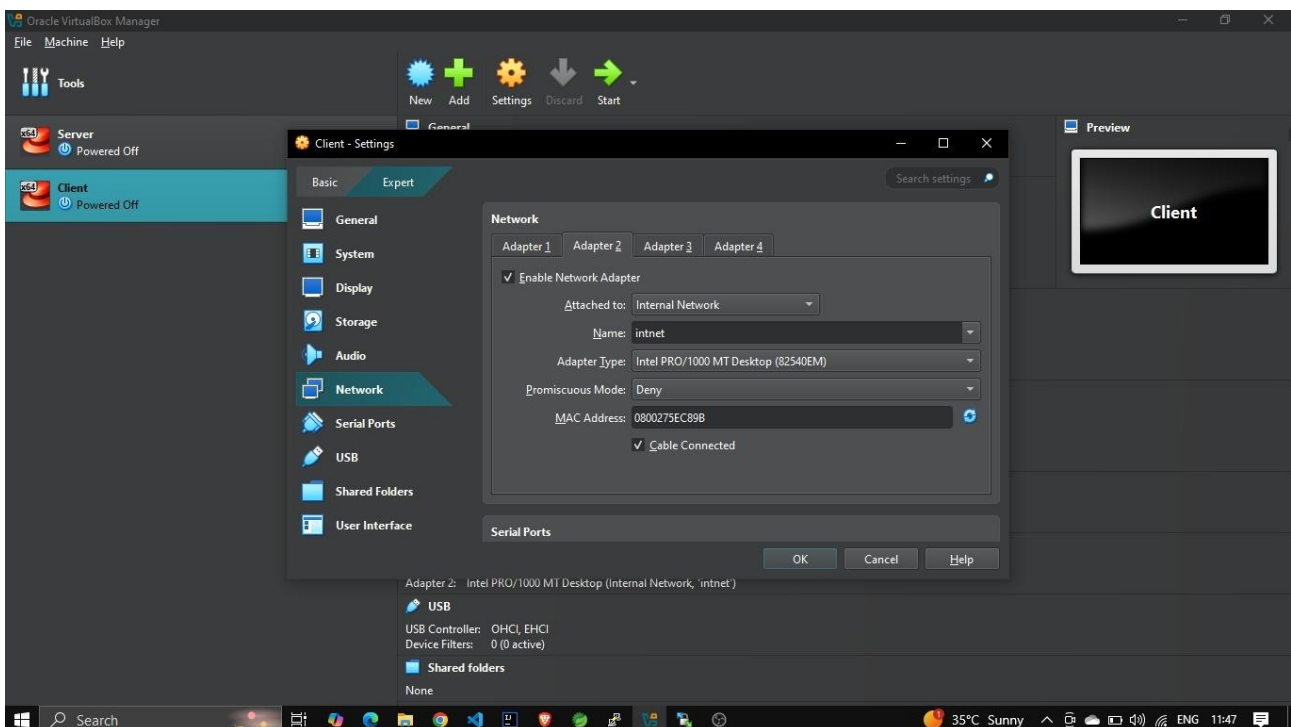
PART 2: CONFIGURE NETWORK (LAN Setup)

- Set both VMs to 'Internal Network' or 'Host-Only Adapter' in VM settings.

Server Machine

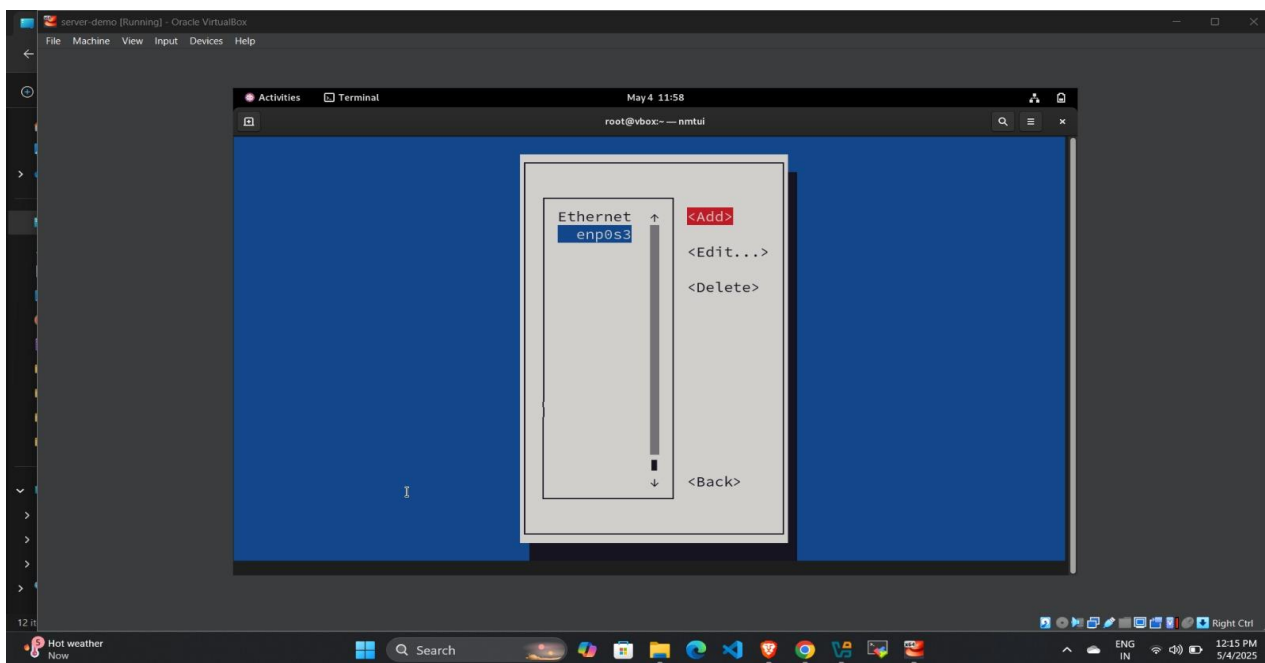
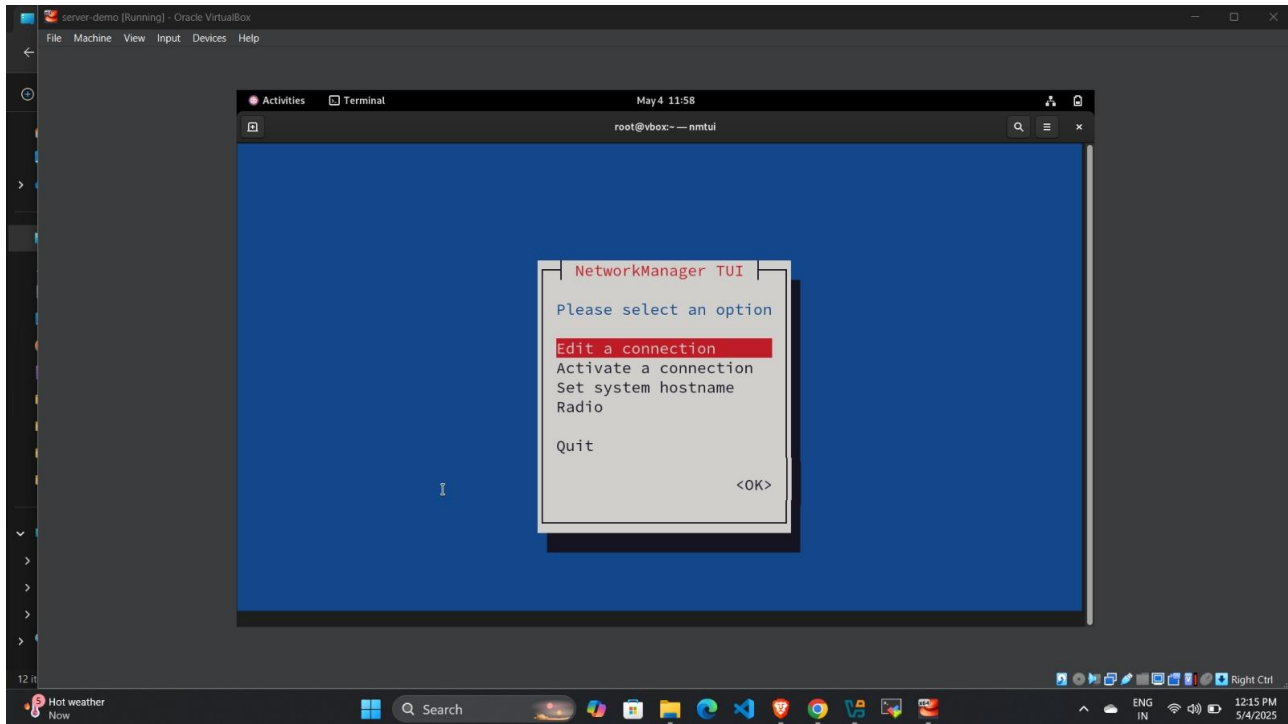


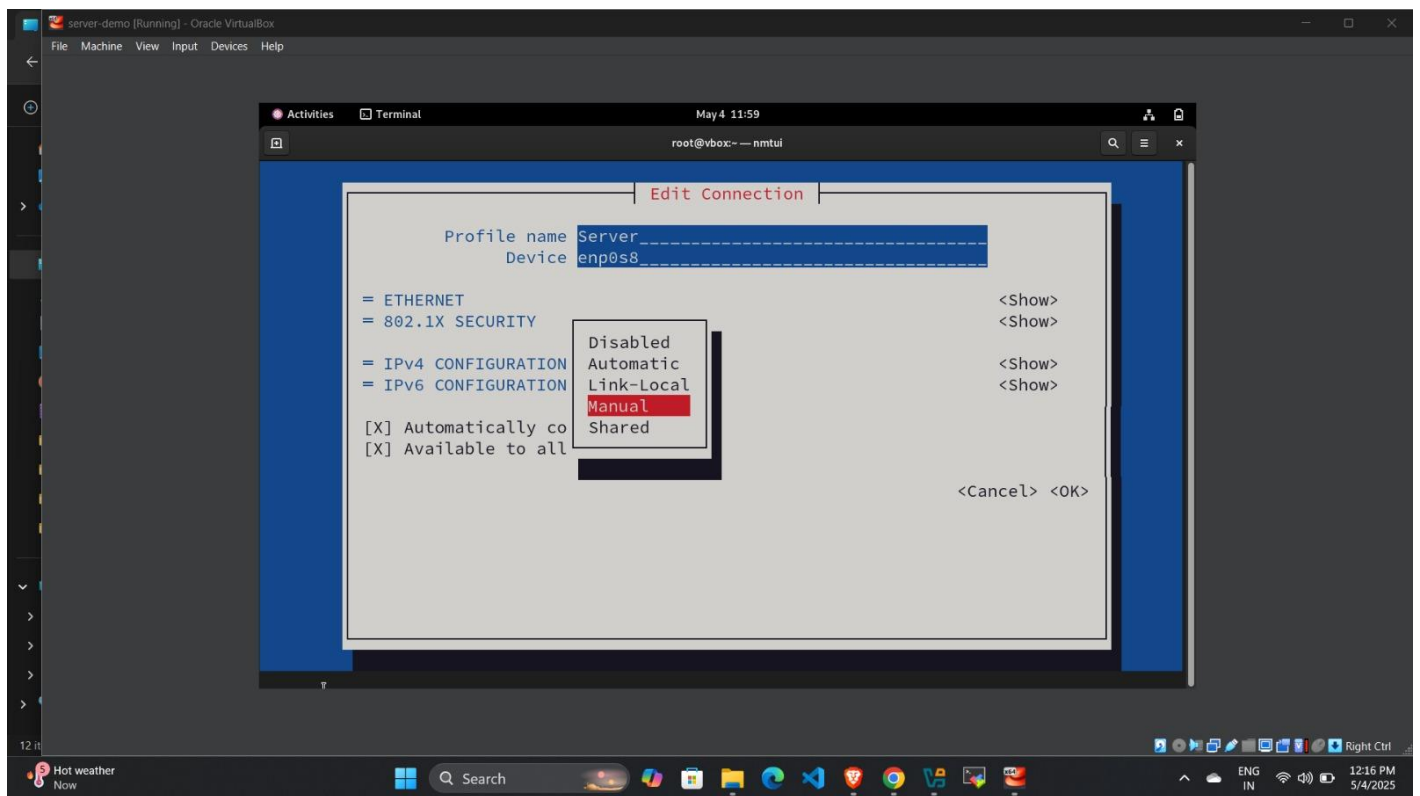
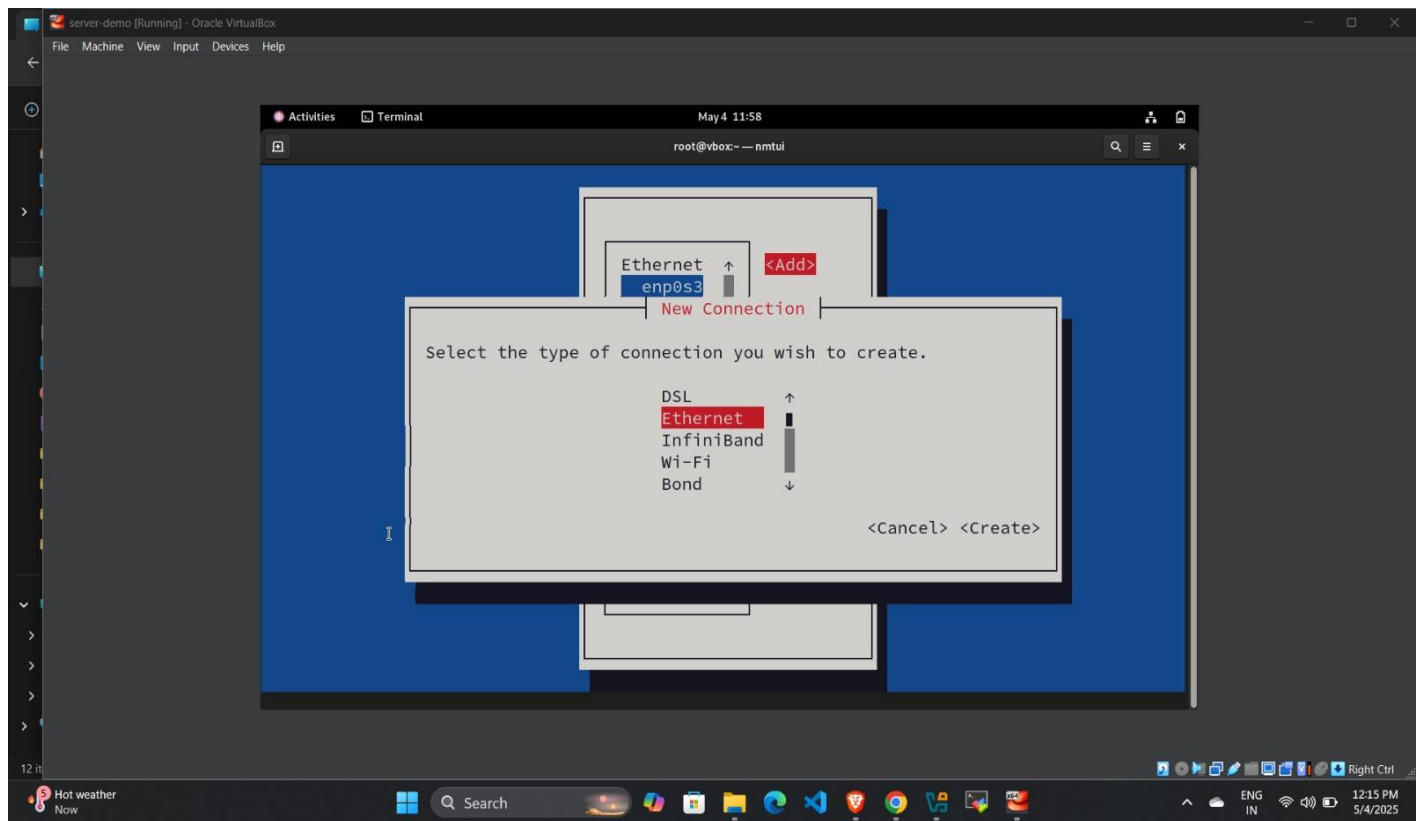
Client Machine

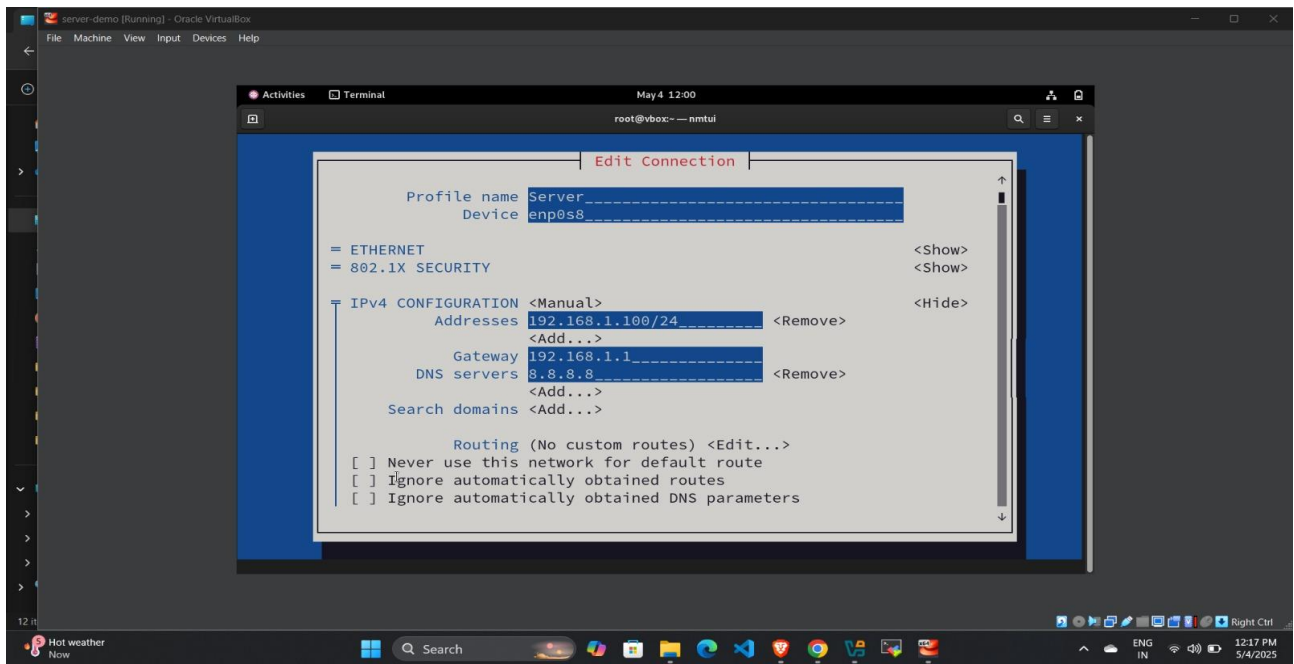


- Use nmtui to assign static IPs:

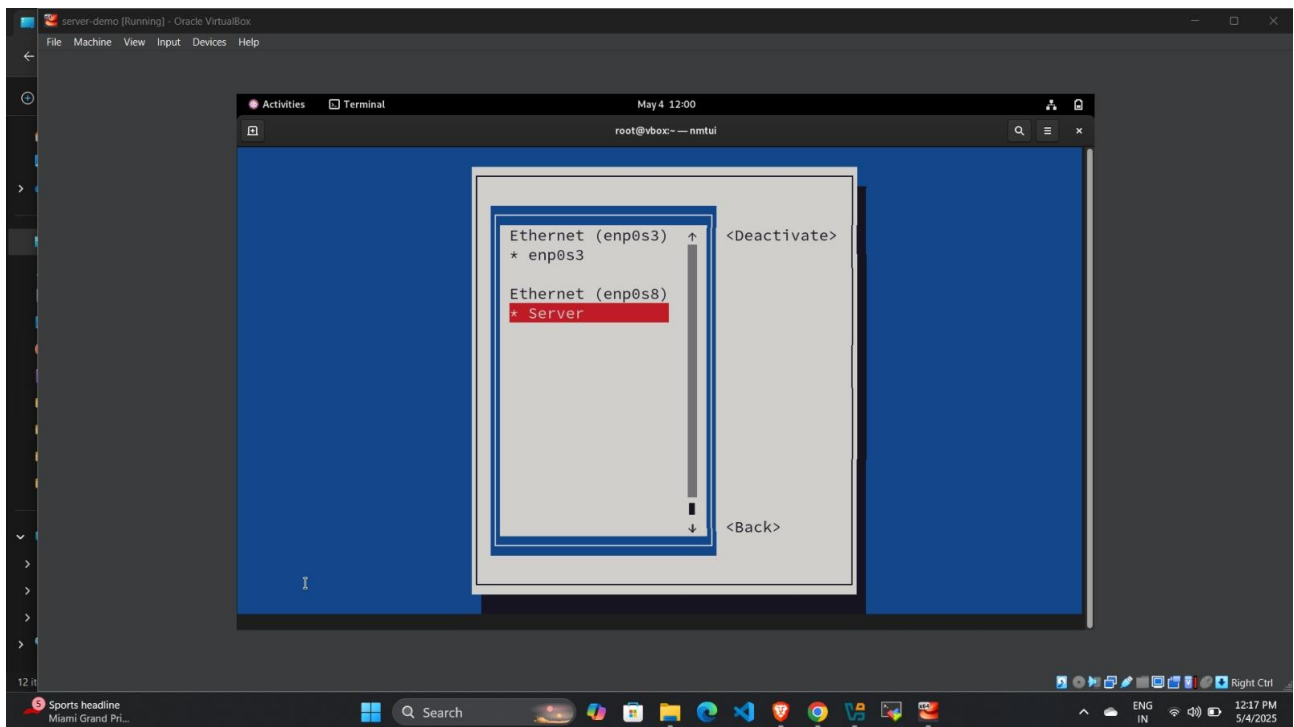
- Server: 192.168.1.100/24



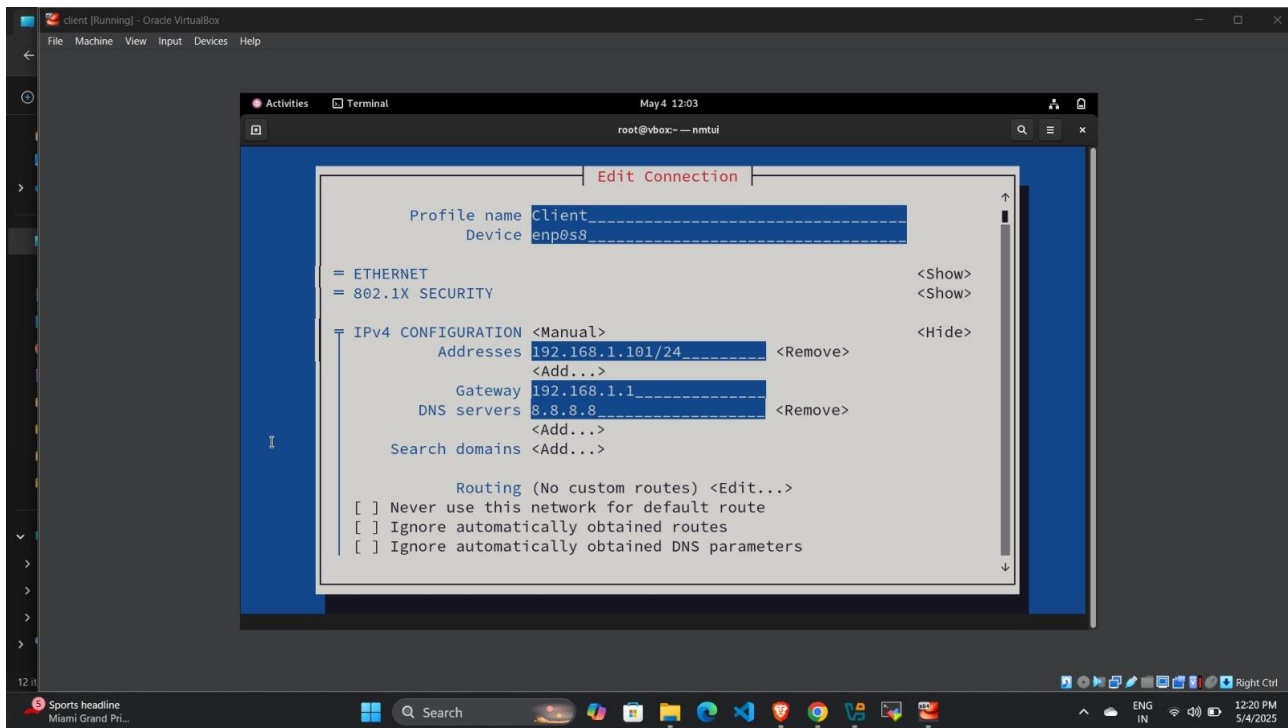




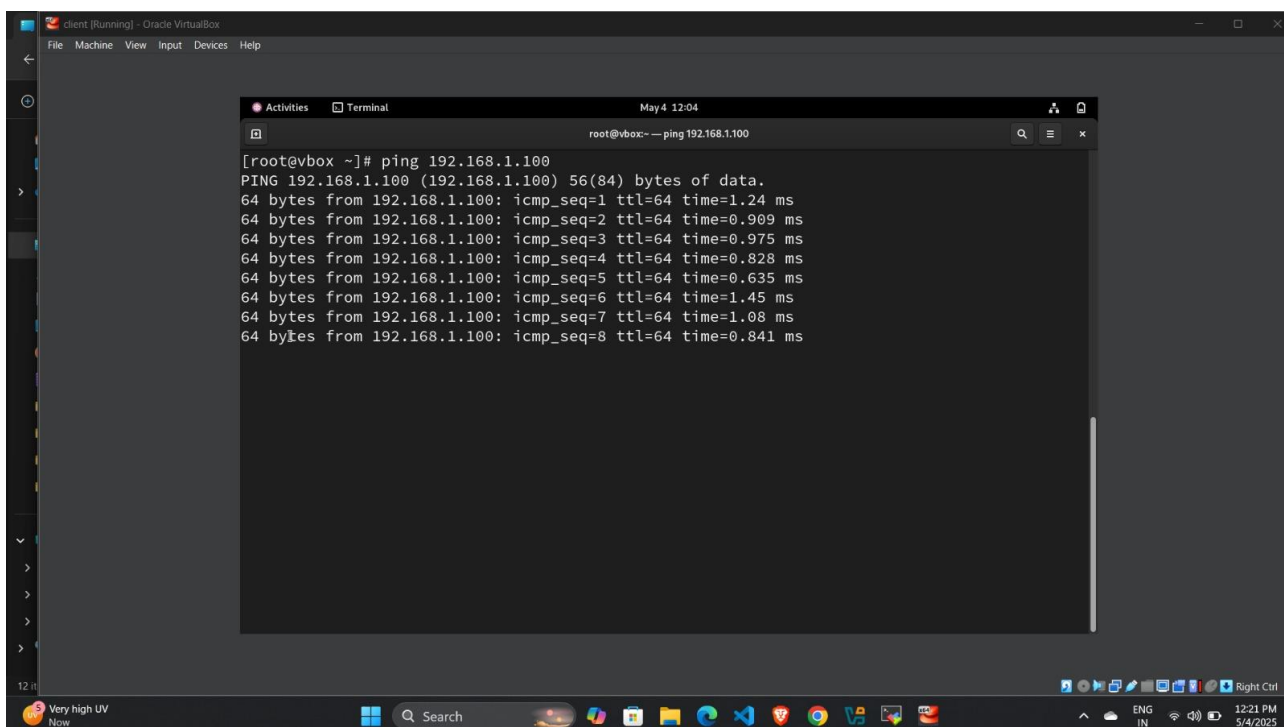
-Here we check activation of server.



- Just like server we do same things with client machine and give it to ip.
- Client: 192.168.1.101/24



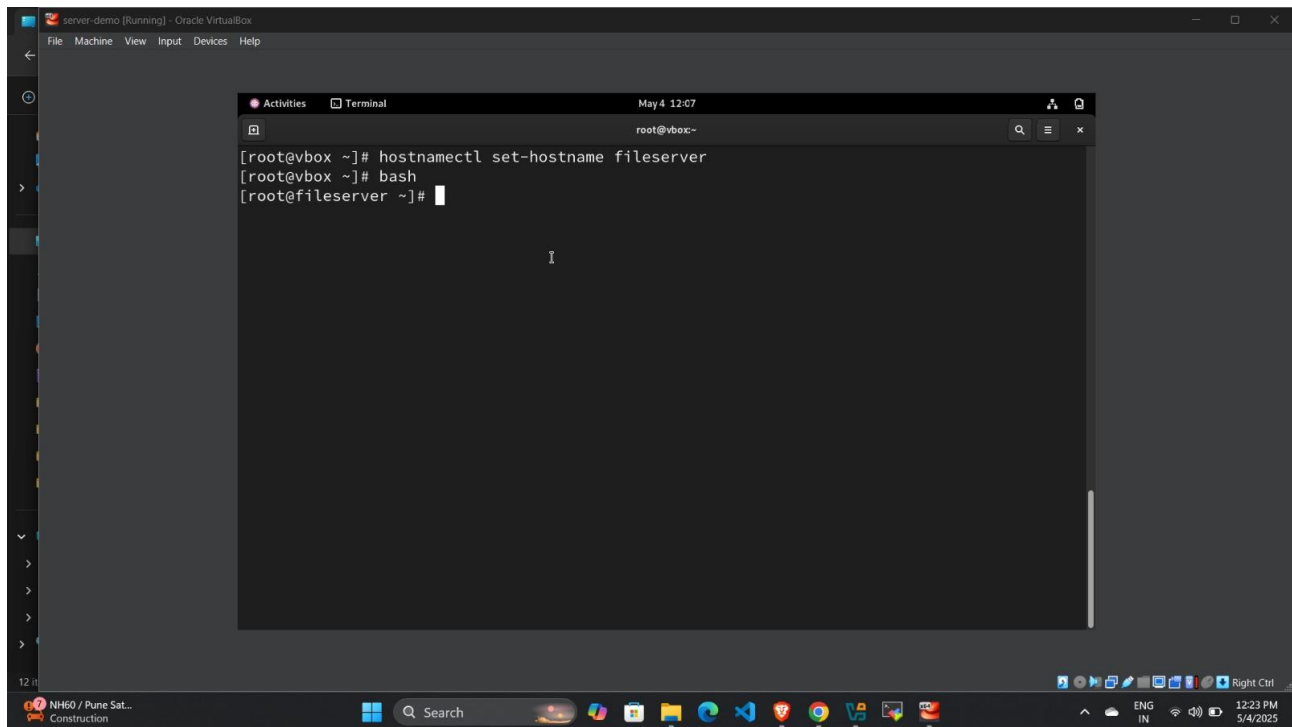
- Here we check connection between server machine and client machine.
- Test connectivity:
ping 192.168.1.100 from client



PART 3: SERVER CONFIGURATION

1. Set Hostname:

```
hostnamectl set-hostname fileserver
```



```
server-demo [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

Activities Terminal May 4 12:07
root@vbox:~

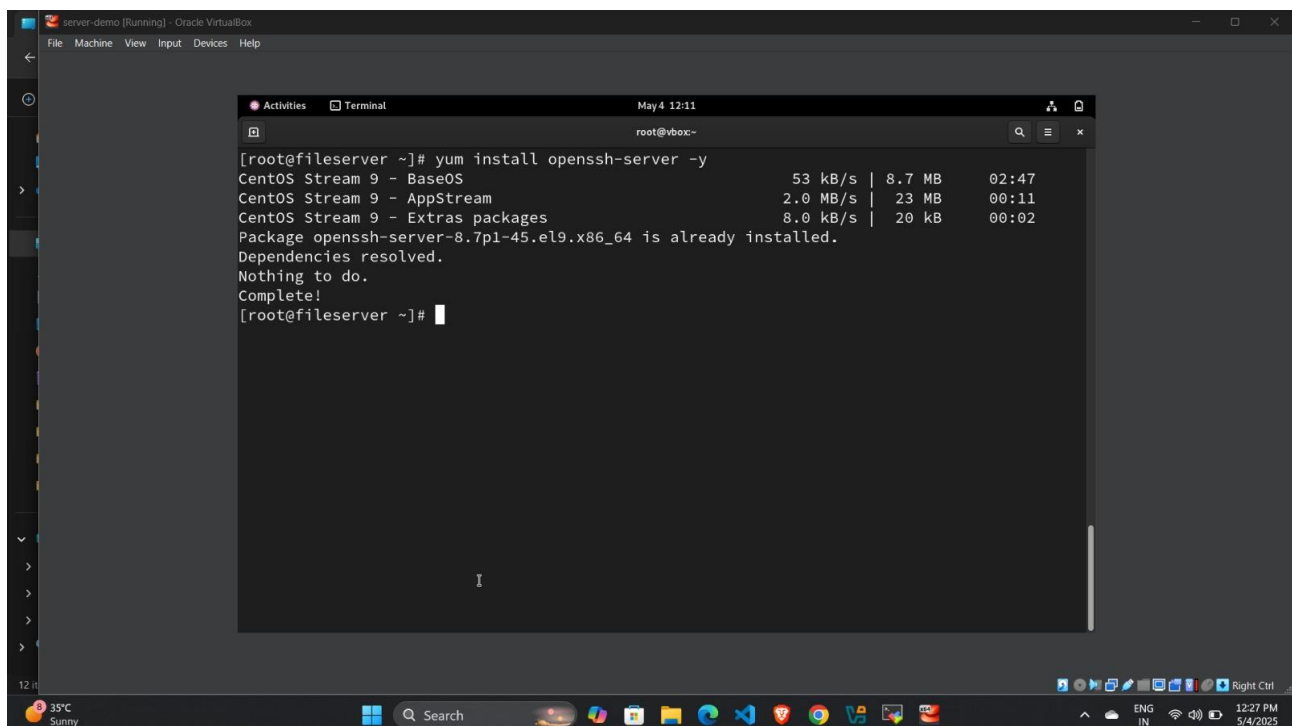
[root@vbox ~]# hostnamectl set-hostname fileserver
[root@vbox ~]# bash
[root@fileserver ~]#
```

2. Install & Start SSH:

```
yum install openssh-server -y
```

```
systemctl enable sshd
```

```
systemctl start sshd
```



```
server-demo [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

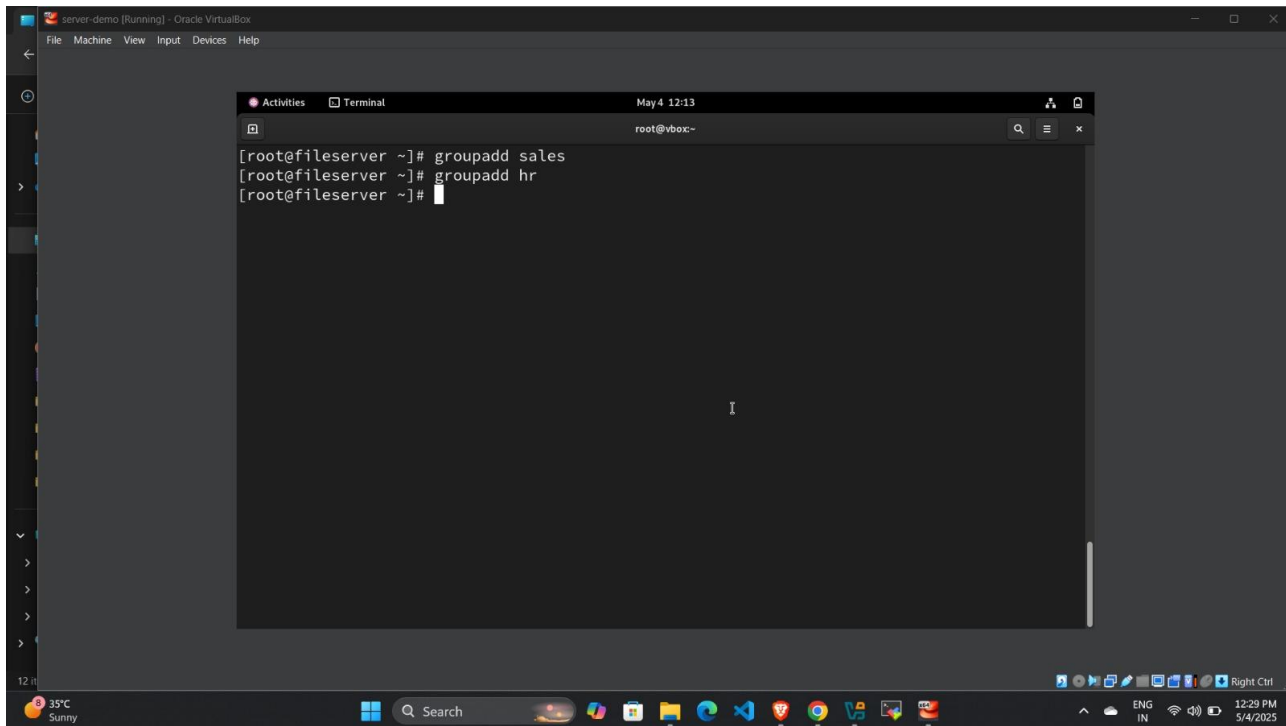
Activities Terminal May 4 12:11
root@vbox:~

[root@fileserver ~]# yum install openssh-server -y
CentOS Stream 9 - BaseOS                    53 kB/s | 8.7 MB    02:47
CentOS Stream 9 - AppStream                  2.0 MB/s | 23 MB    00:11
CentOS Stream 9 - Extras packages            8.0 kB/s | 20 kB    00:02
Package openssh-server-8.7p1-45.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@fileserver ~]#
```


3. Create User Groups:

```
groupadd sales
```

```
groupadd hr
```



The screenshot shows a terminal window titled 'server-demo [Running] - Oracle VM VirtualBox'. The terminal prompt is 'root@vbox:~'. The user has entered the following commands:

```
[root@fileserver ~]# groupadd sales
[root@fileserver ~]# groupadd hr
[root@fileserver ~]#
```

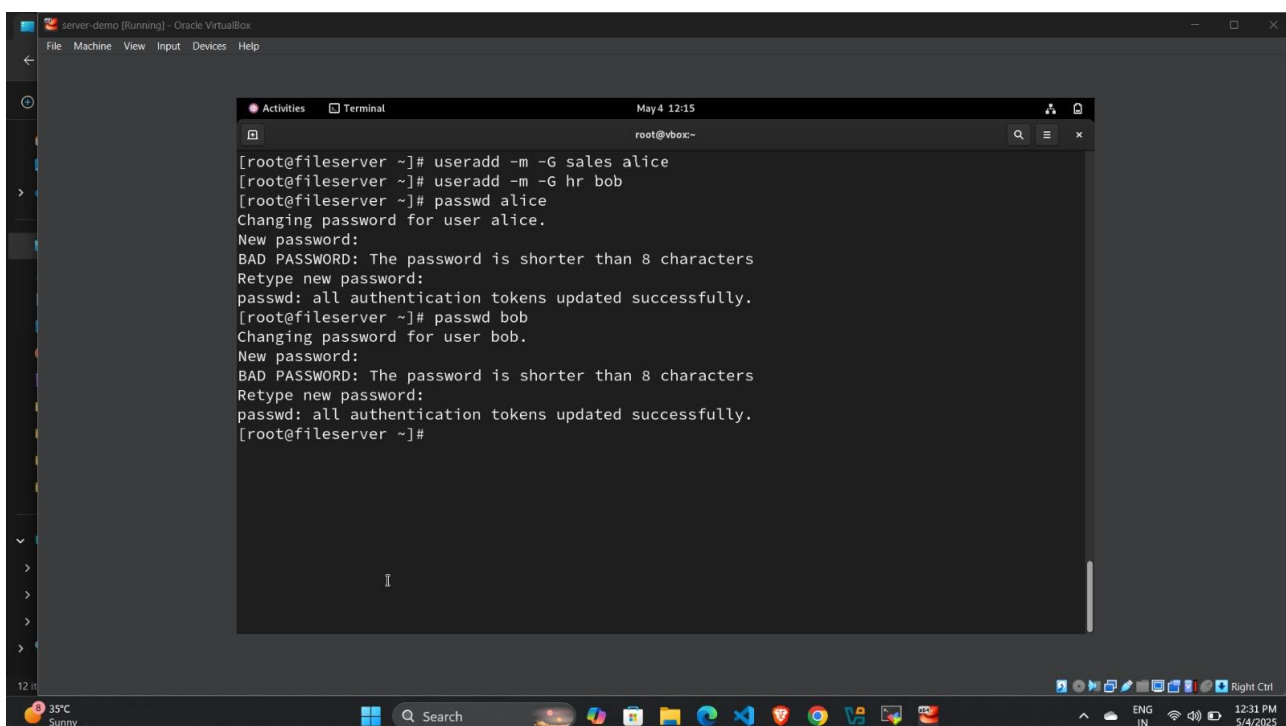
4. Add Users to Groups:

```
useradd -m -G sales alice
```

```
useradd -m -G hr bob
```

```
passwd alice
```

```
passwd bob
```



The screenshot shows a terminal window titled 'server-demo [Running] - Oracle VM VirtualBox'. The terminal prompt is 'root@vbox:~'. The user has entered the following commands:

```
[root@fileserver ~]# useradd -m -G sales alice
[root@fileserver ~]# useradd -m -G hr bob
[root@fileserver ~]# passwd alice
Changing password for user alice.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@fileserver ~]# passwd bob
Changing password for user bob.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@fileserver ~]#
```

5. Create Shared Folders:

```
mkdir /srv/sales
```

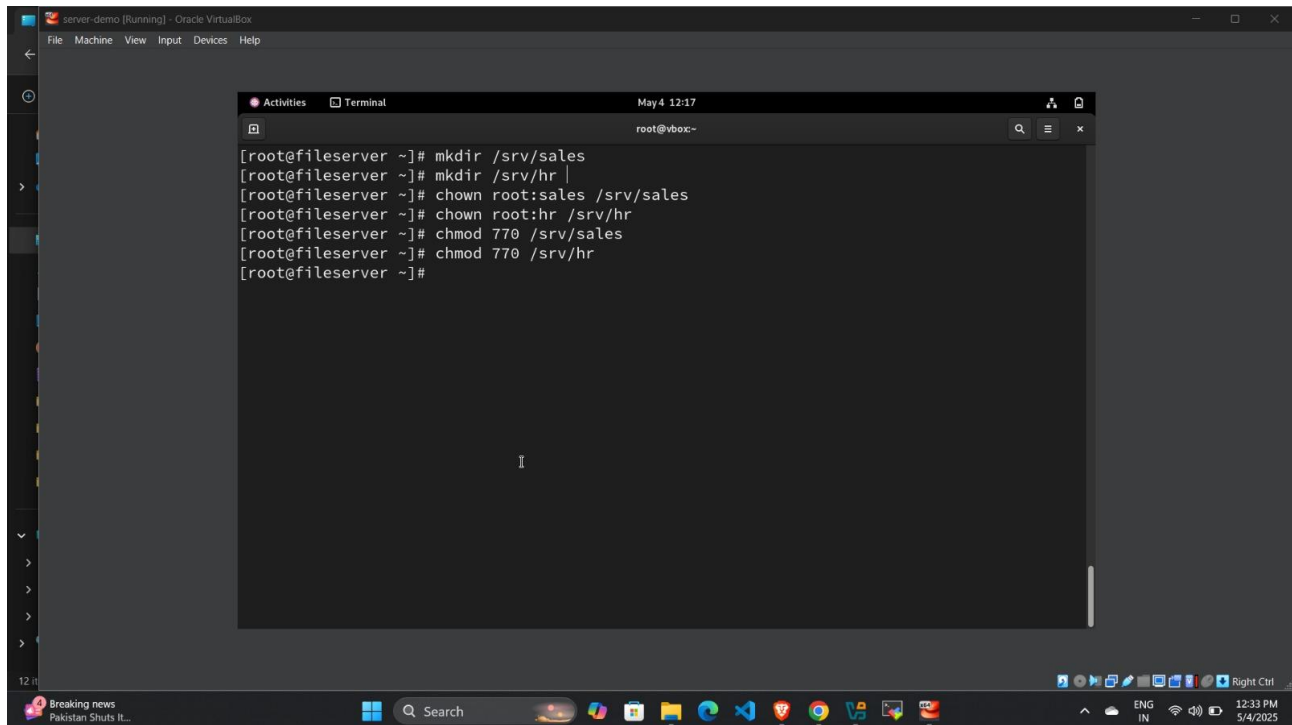
```
mkdir /srv/hr
```

```
chown root:sales /srv/sales
```

```
chown root:hr /srv/hr
```

```
chmod 770 /srv/sales
```

```
chmod 770 /srv/hr
```

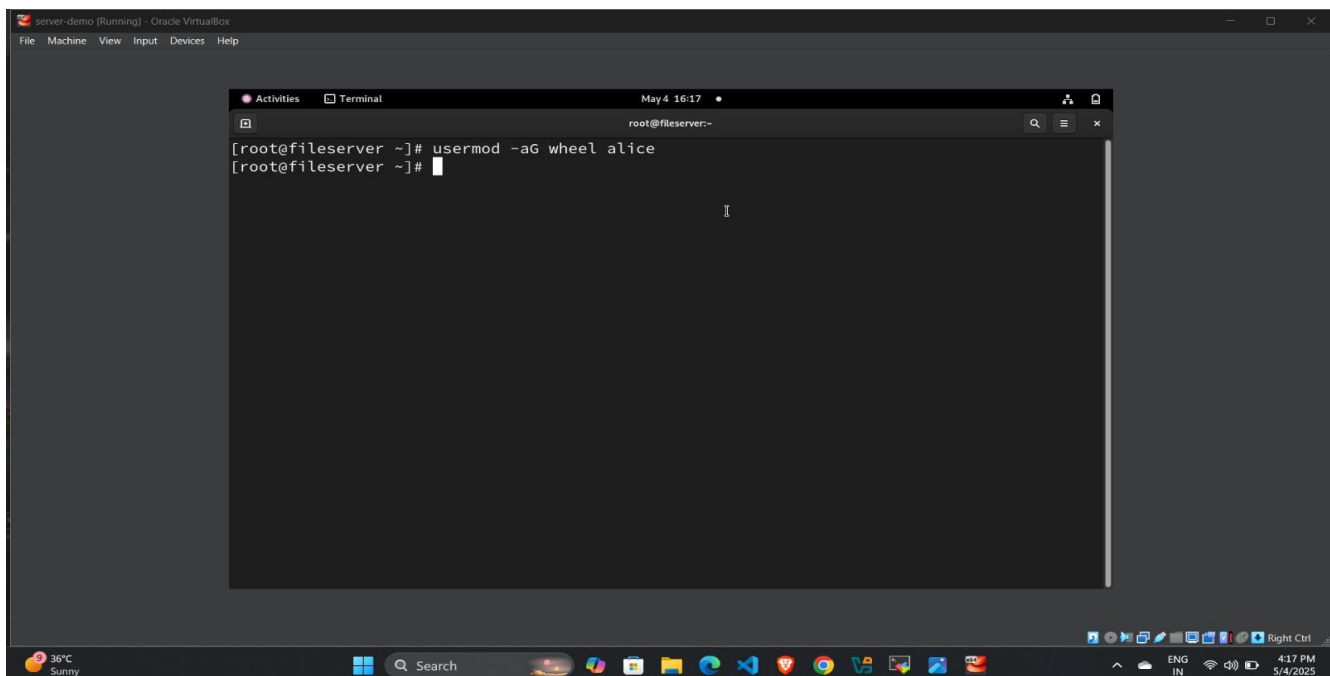


The screenshot shows a terminal window titled "server-demo [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
root@vbox:~  
[root@fileserver ~]# mkdir /srv/sales  
[root@fileserver ~]# mkdir /srv/hr |  
[root@fileserver ~]# chown root:sales /srv/sales  
[root@fileserver ~]# chown root:hr /srv/hr  
[root@fileserver ~]# chmod 770 /srv/sales  
[root@fileserver ~]# chmod 770 /srv/hr  
[root@fileserver ~]#
```

6. Optional: Sudo Access

```
usermod -aG wheel alice
```



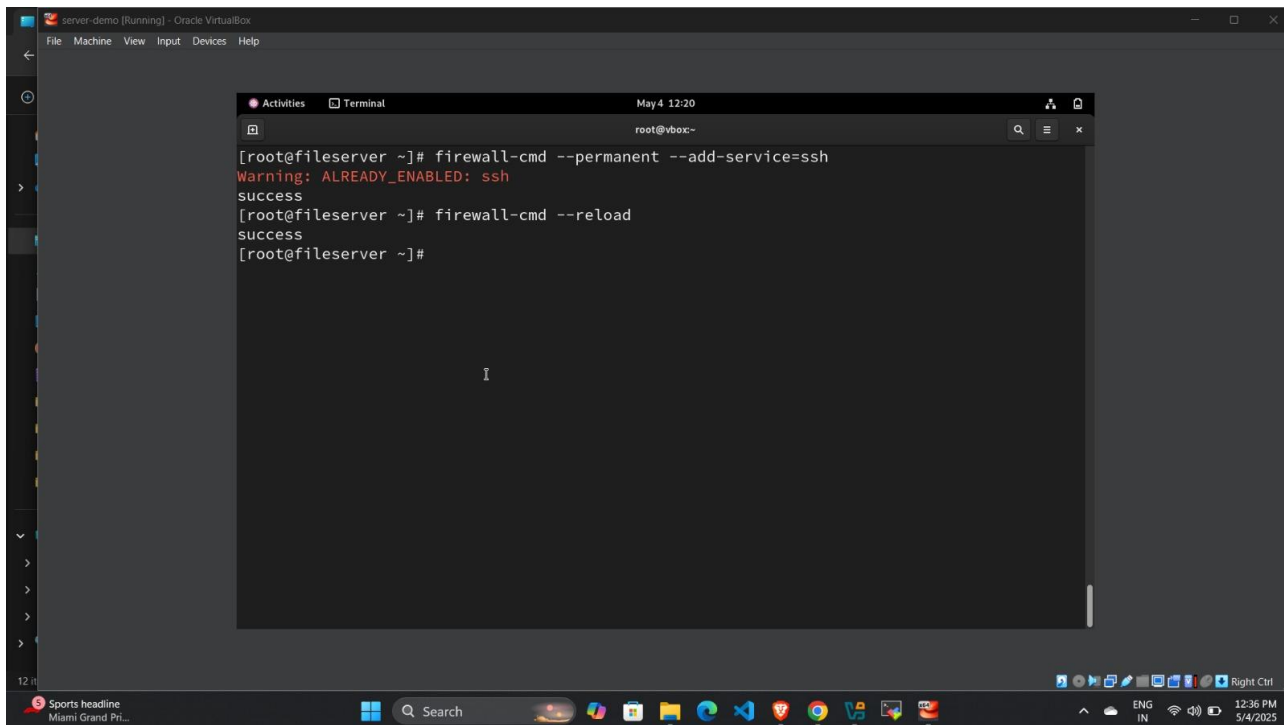
The screenshot shows a terminal window titled "server-demo [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
root@fileserver:~  
[root@fileserver ~]# usermod -aG wheel alice  
[root@fileserver ~]#
```

7. Enable Firewall SSH Port:

```
firewall-cmd --permanent --add-service=ssh
```

```
firewall-cmd --reload
```

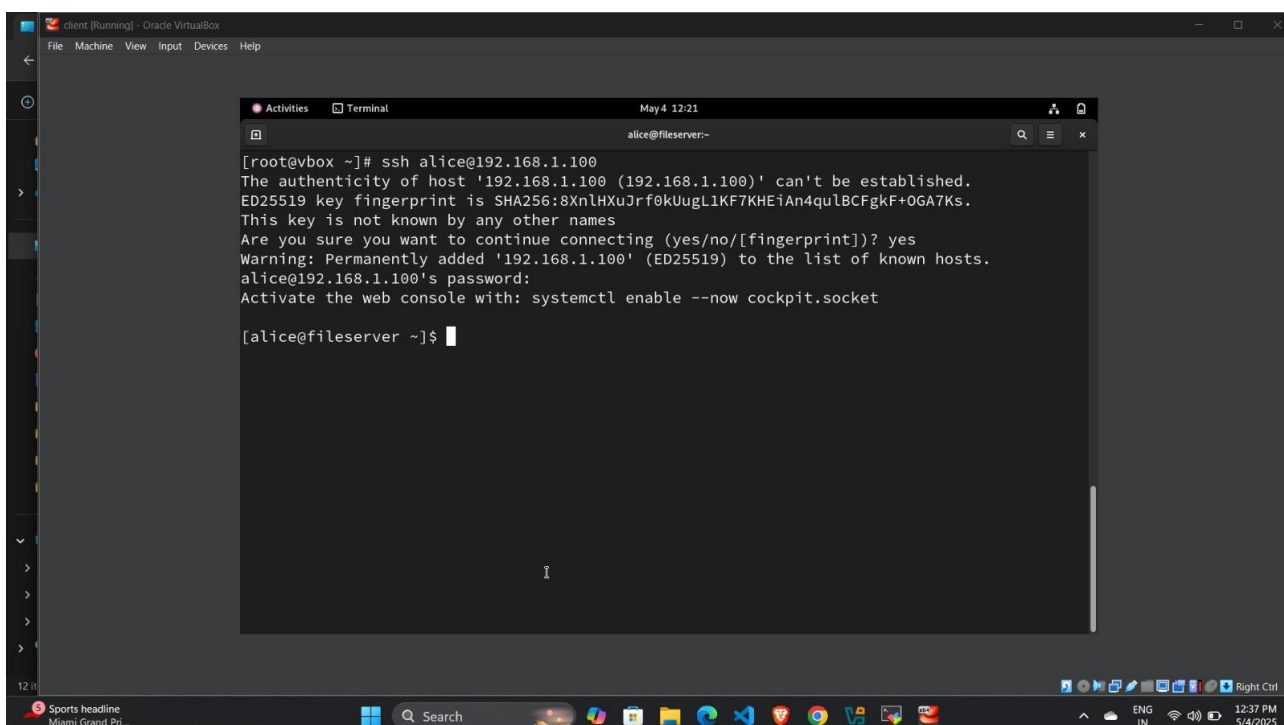


The screenshot shows a terminal window titled 'root@vbox:~' with the following output:

```
[root@fileserv ~]# firewall-cmd --permanent --add-service=ssh
Warning: ALREADY_ENABLED: ssh
success
[root@fileserv ~]# firewall-cmd --reload
success
[root@fileserv ~]#
```

PART 4: CLIENT MACHINE USAGE: Take access of server machine's in client machine.

- Connect via SSH: `ssh alice@192.168.1.100`

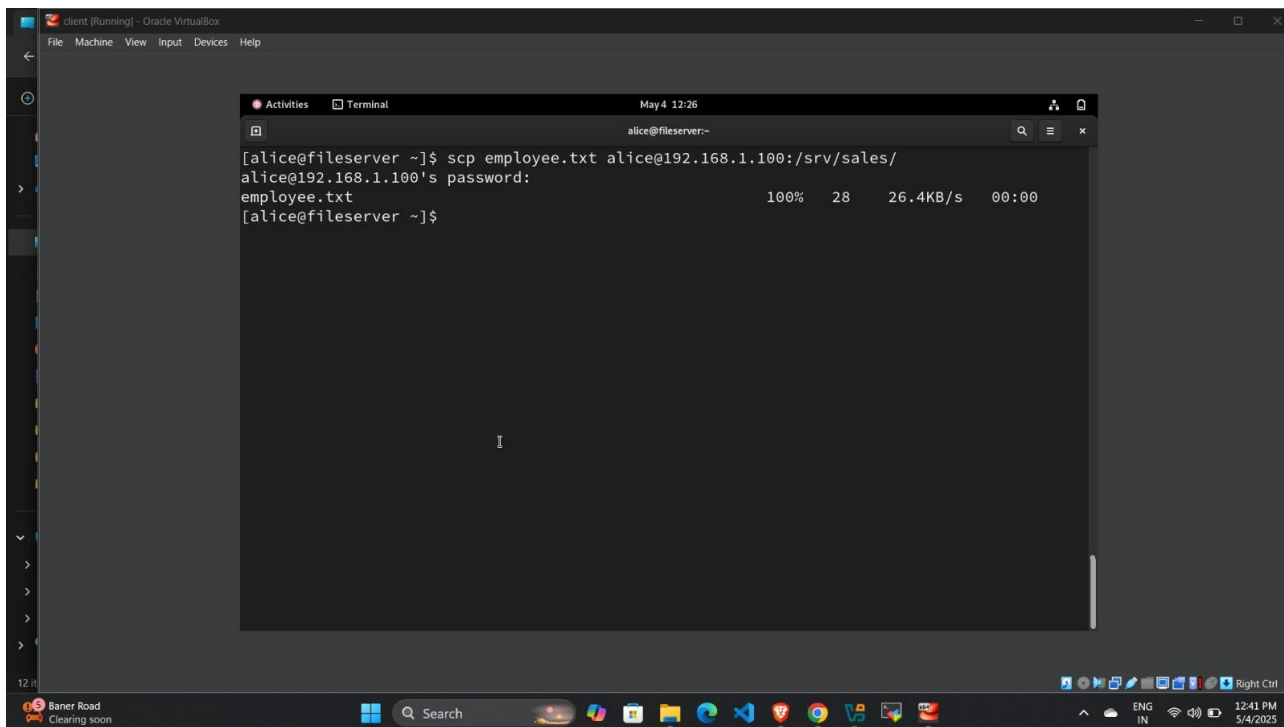


The screenshot shows a terminal window titled 'alice@fileserv:~' with the following output:

```
[root@vbox ~]# ssh alice@192.168.1.100
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be established.
ED25519 key fingerprint is SHA256:8XnlHXuJrf0kUugL1KF7KHEiAn4qu1BCFgkF+OGA7Ks.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.100' (ED25519) to the list of known hosts.
alice@192.168.1.100's password:
Activate the web console with: systemctl enable --now cockpit.socket

[alice@fileserv ~]$
```

- Transfer Files: scp employee.txt alice@192.168.1.100:/srv/sales/
- Here we transfer file from client to server.



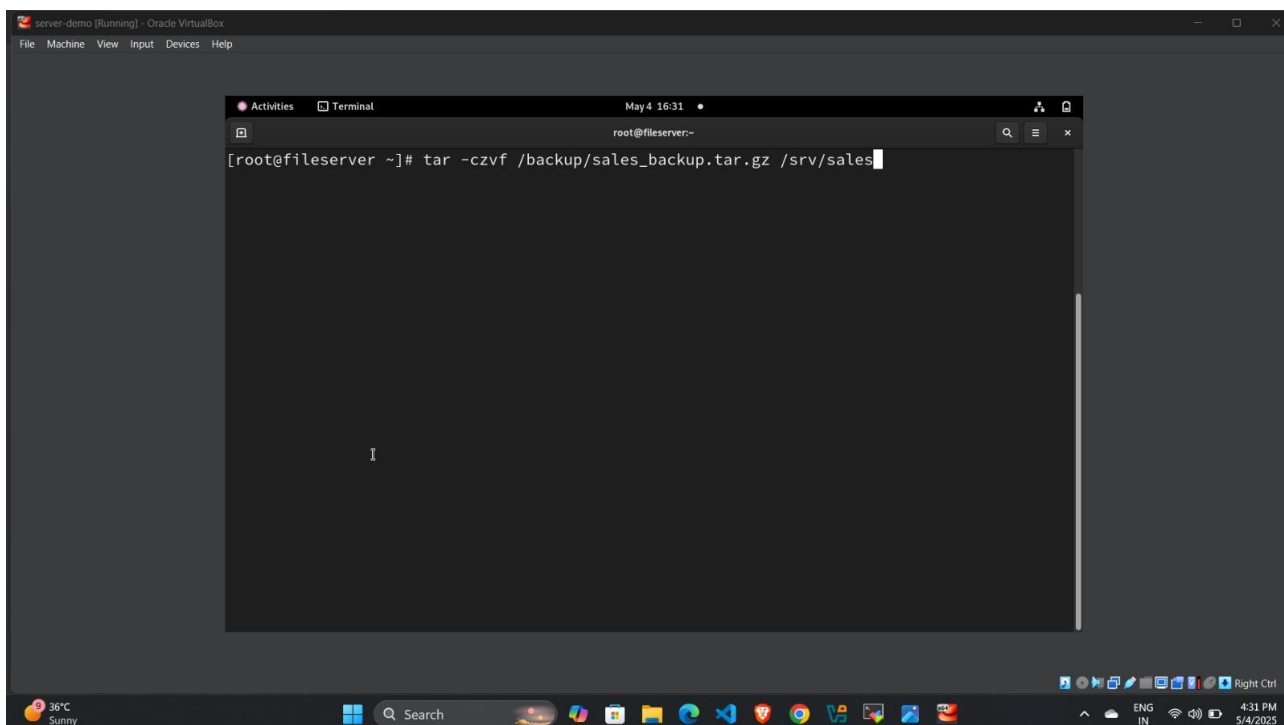
The screenshot shows a terminal window titled "client [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
alice@fileserv:~$ scp employee.txt alice@192.168.1.100:/srv/sales/
alice@192.168.1.100's password:
employee.txt                                100% 28   26.4KB/s   00:00
[alice@fileserv ~]$
```

The terminal window is part of a desktop environment with a taskbar at the bottom showing various application icons and system status information like "12:41 PM 5/4/2025".

PART 5: BACKUP & MONITORING (Server)

- Backup: tar -czvf /backup/sales_backup.tar.gz /srv/sales



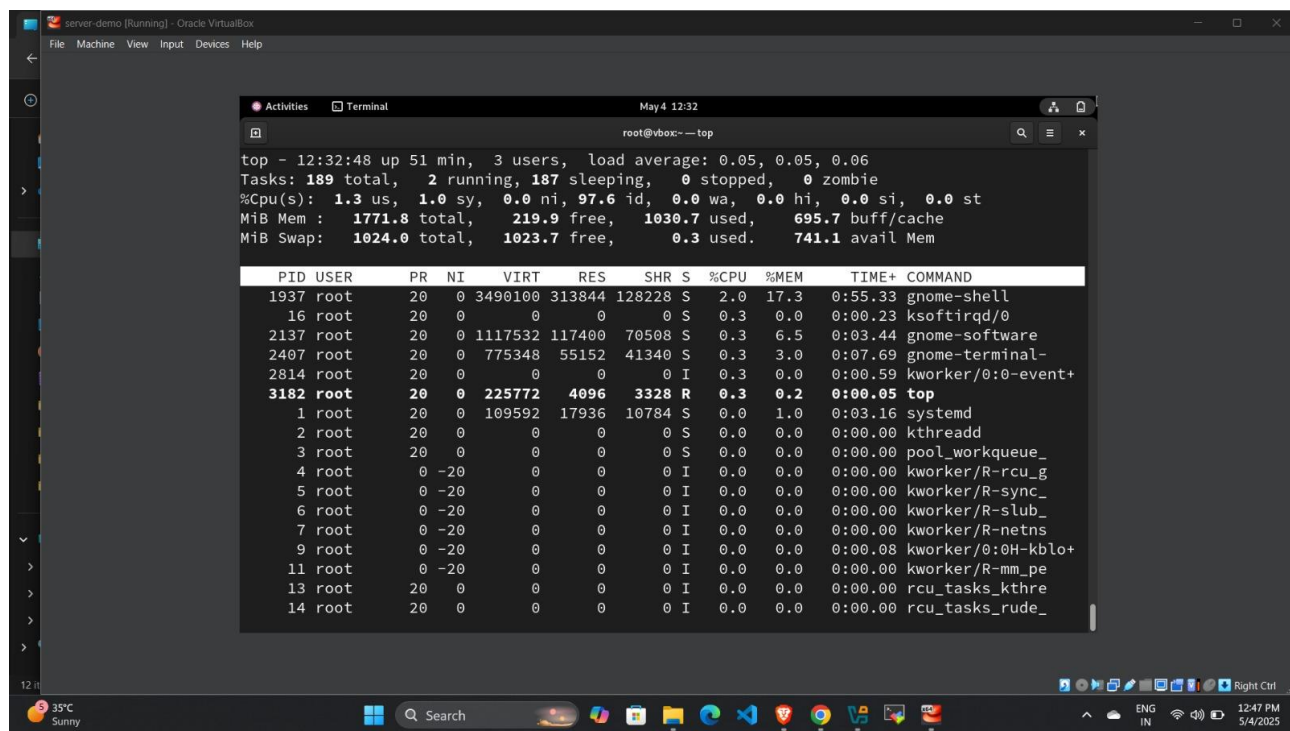
The screenshot shows a terminal window titled "server-demo [Running] - Oracle VM VirtualBox". The terminal output is as follows:

```
root@fileserv:~# tar -czvf /backup/sales_backup.tar.gz /srv/sales
```

The terminal window is part of a desktop environment with a taskbar at the bottom showing various application icons and system status information like "4:31 PM 5/4/2025".

- Monitoring files.

- top

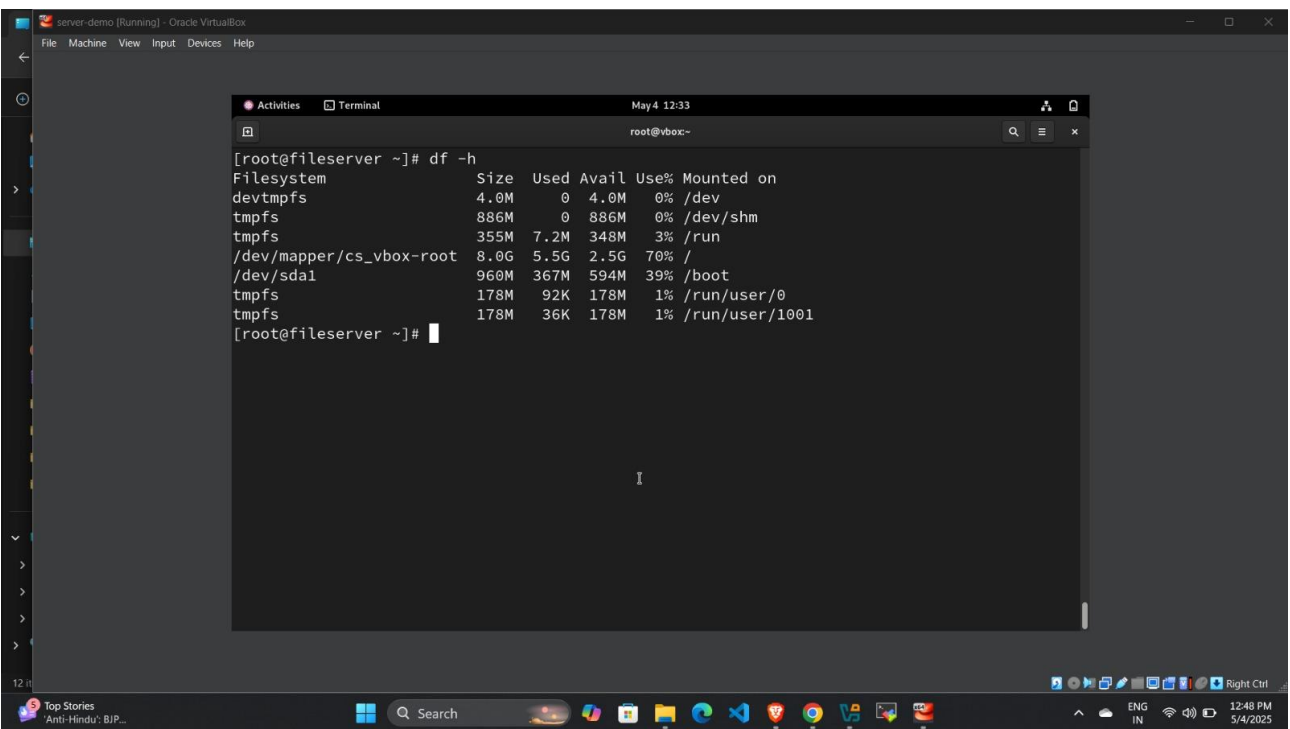


The screenshot shows a terminal window titled "root@vbox:~ - top" with the following output:

```
top - 12:32:48 up 51 min, 3 users, load average: 0.05, 0.05, 0.06
Tasks: 189 total, 2 running, 187 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.3 us, 1.0 sy, 0.0 ni, 97.6 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1771.8 total, 219.9 free, 1030.7 used, 695.7 buff/cache
MiB Swap: 1024.0 total, 1023.7 free, 0.3 used, 741.1 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1937	root	20	0	3490100	313844	128228	S	2.0	17.3	0:55.33	gnome-shell
16	root	20	0	0	0	0	S	0.3	0.0	0:00.23	ksoftirqd/0
2137	root	20	0	1117532	117400	70508	S	0.3	6.5	0:03.44	gnome-software
2407	root	20	0	775348	55152	41340	S	0.3	3.0	0:07.69	gnome-terminal-
2814	root	20	0	0	0	0	I	0.3	0.0	0:00.59	kworker/0:0-event+
3182	root	20	0	225772	4096	3328	R	0.3	0.2	0:00.05	top
1	root	20	0	109592	17936	10784	S	0.0	1.0	0:03.16	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_g
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-sync_
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-slub_
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-netns
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.08	kworker/0:0H-kblo+
11	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-mm_pe
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_kthre
14	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_rude_

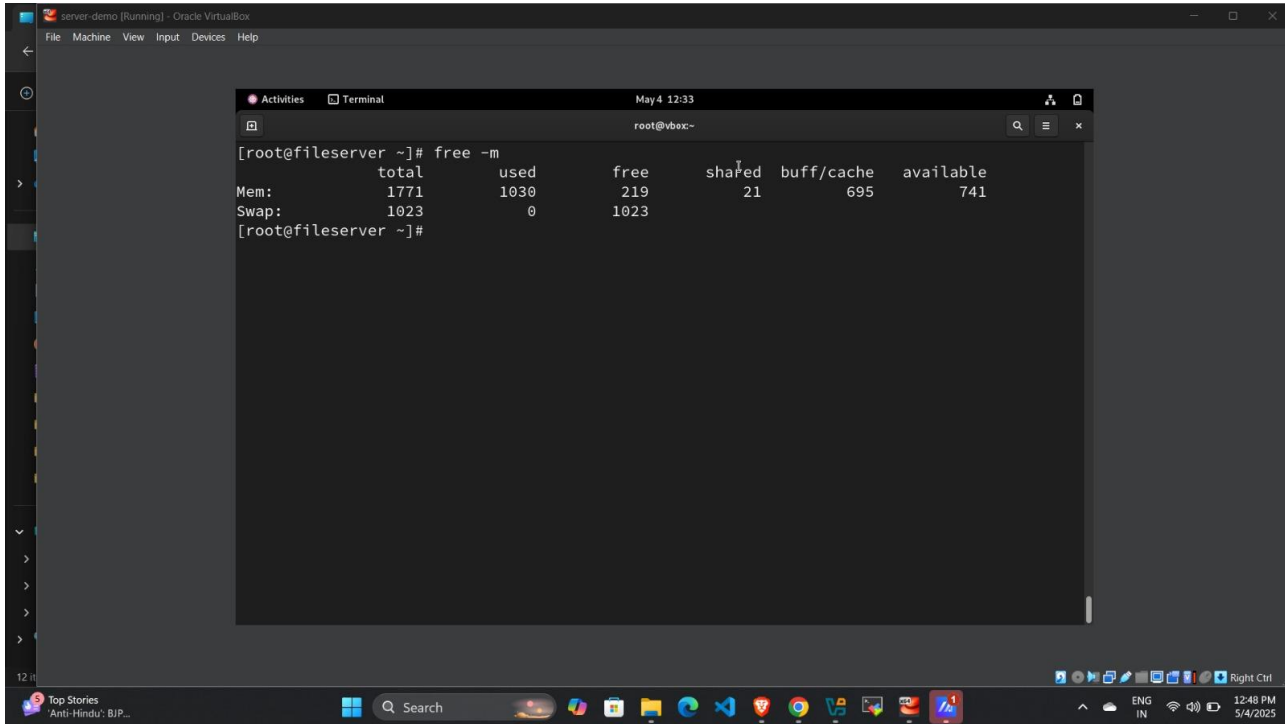
- df -h



The screenshot shows a terminal window titled "root@fileserver ~]# df -h" with the following output:

```
[root@fileserver ~]# df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                   4.0M        0   4.0M   0% /dev
tmpfs                      886M        0   886M   0% /dev/shm
tmpfs                      355M    7.2M   348M   3% /run
/dev/mapper/cs_vbox-root  8.0G    5.5G    2.5G  70% /
/dev/sda1                  960M    367M   594M  39% /boot
tmpfs                      178M     92K   178M   1% /run/user/0
tmpfs                      178M     36K   178M   1% /run/user/1001
```

- free -m

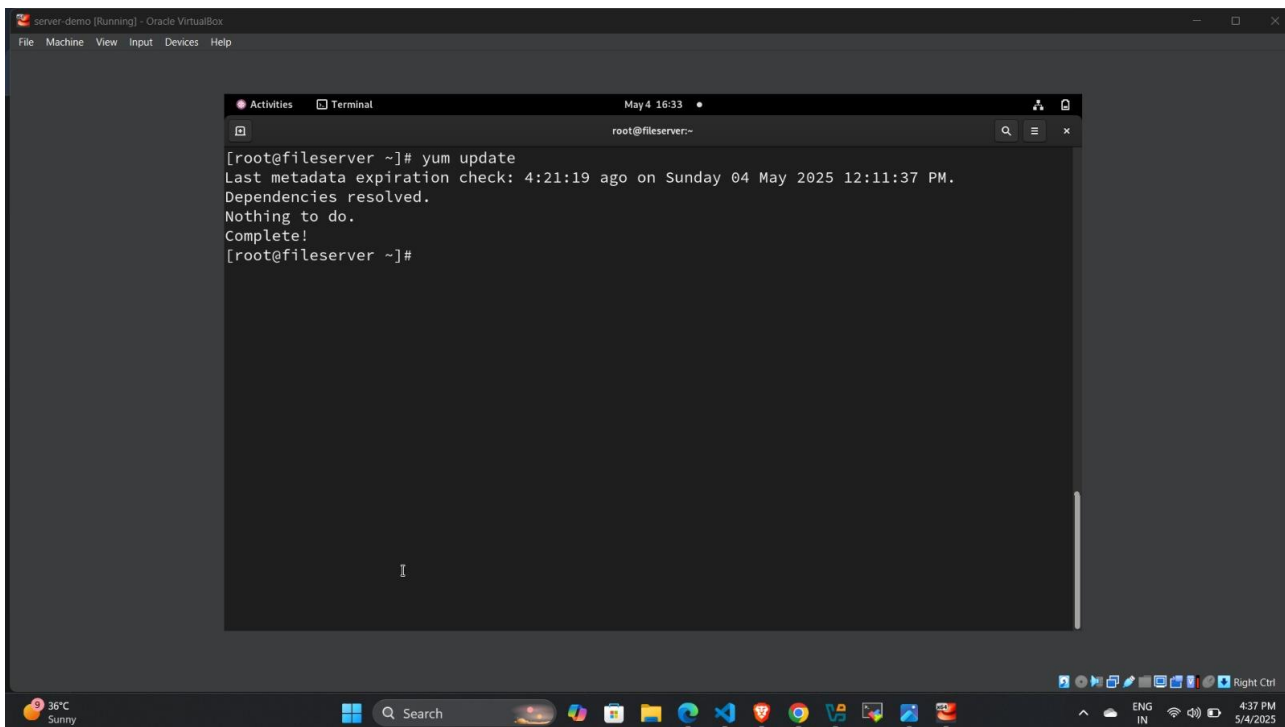


The screenshot shows a terminal window titled 'server-demo [Running] - Oracle VM VirtualBox'. The terminal is running the command 'free -m' as root@fileserver. The output is a table showing memory usage in megabytes.

	total	used	free	shared	buff/cache	available
Mem:	1771	1030	219	21	695	741
Swap:	1023	0	1023			

The terminal prompt is [root@fileserver ~]#.

- yum update



The screenshot shows a terminal window titled 'server-demo [Running] - Oracle VM VirtualBox'. The terminal is running the command 'yum update' as root@fileserver. The output indicates that the metadata expiration check was successful and no updates are available.

```
[root@fileserver ~]# yum update
Last metadata expiration check: 4:21:19 ago on Sunday 04 May 2025 12:11:37 PM.
Dependencies resolved.
Nothing to do.
Complete!
[root@fileserver ~]#
```

The terminal prompt is [root@fileserver ~]#.

