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

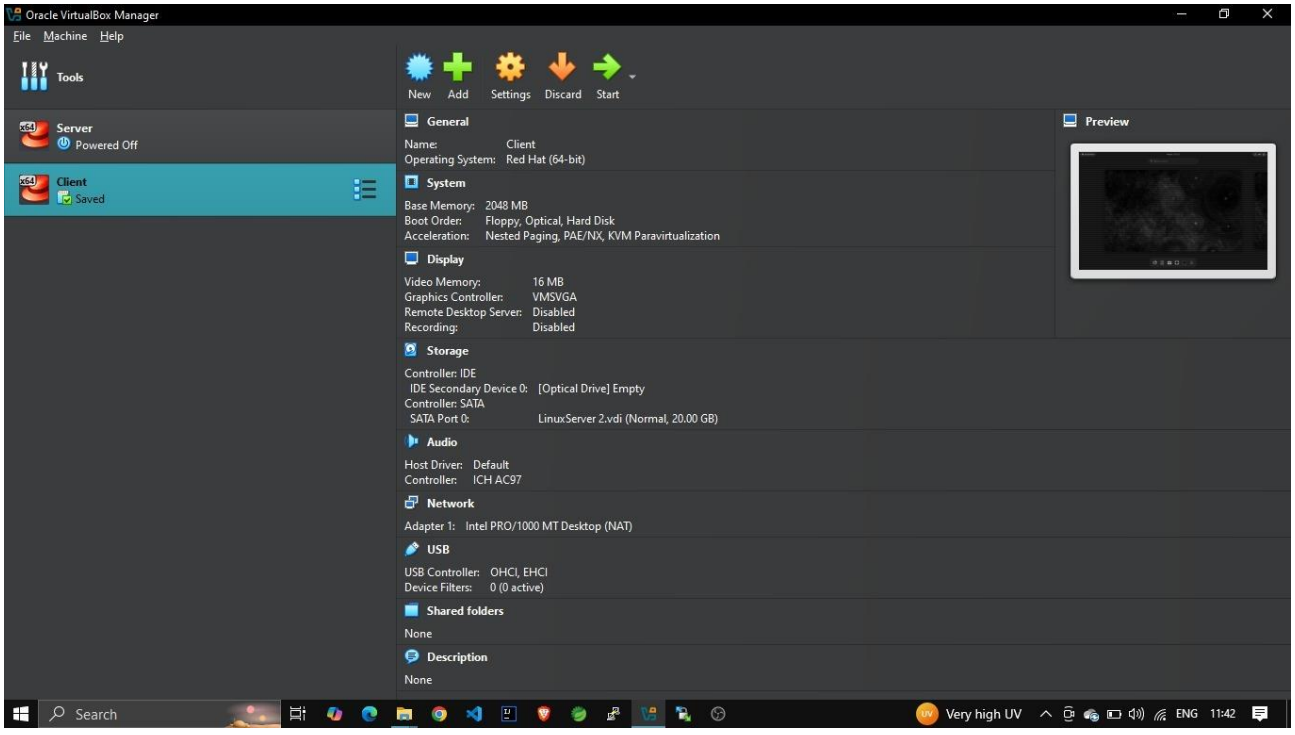
Key Features

- **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

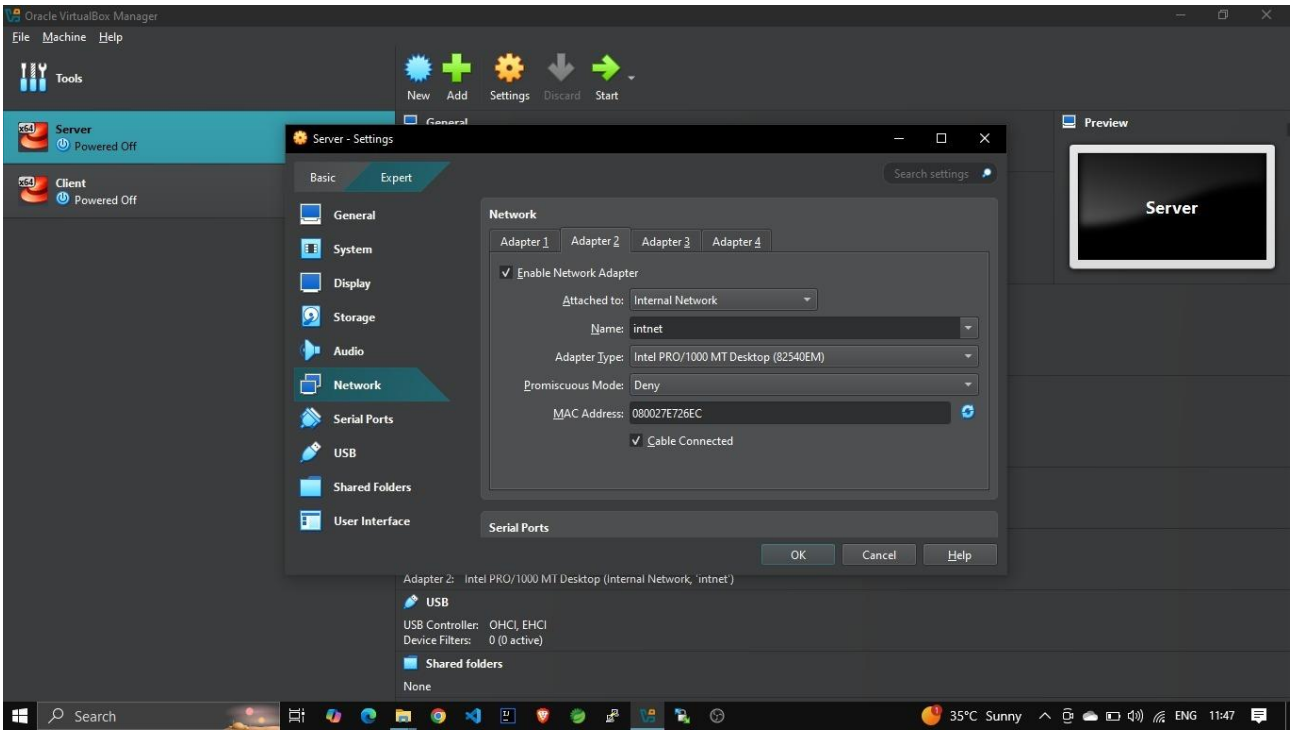
1. Create one CentOS 9 VM (main server)
2. Clone the VM (client machine)



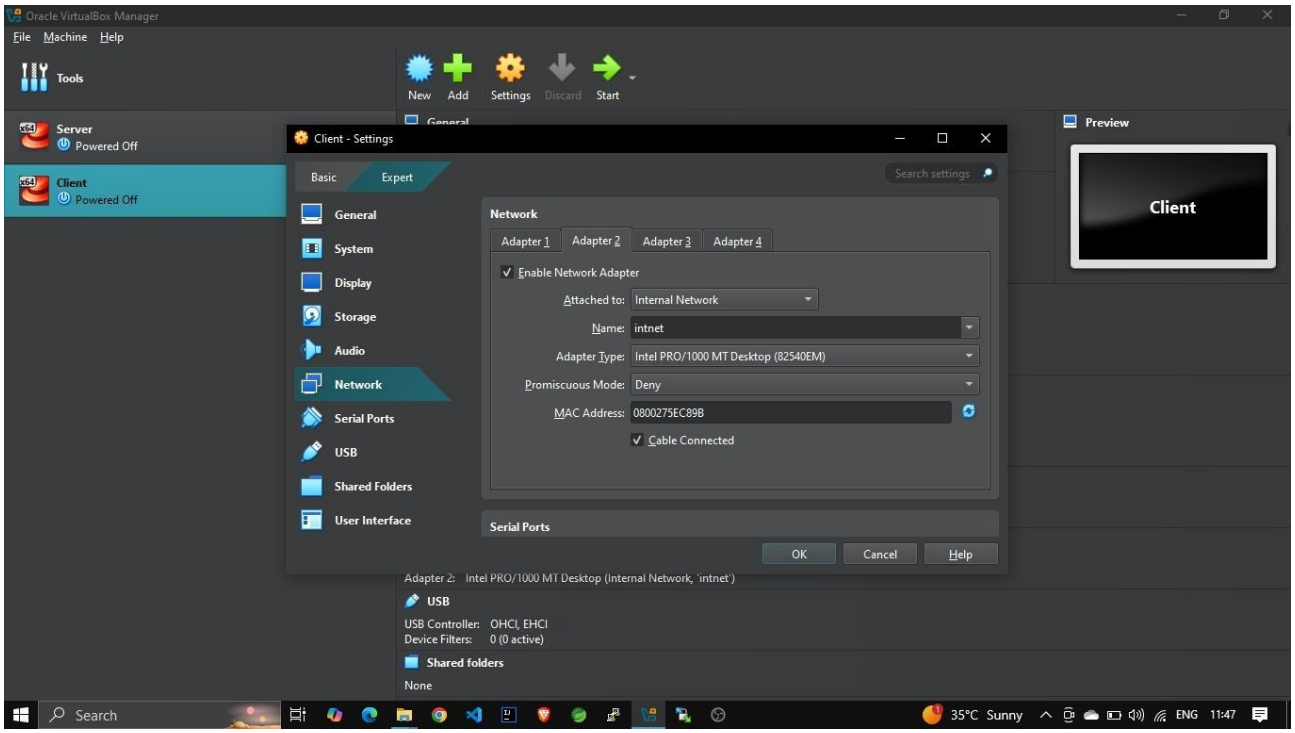
PART 2: CONFIGURE NETWORK (LAN Setup)

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

Server Machine

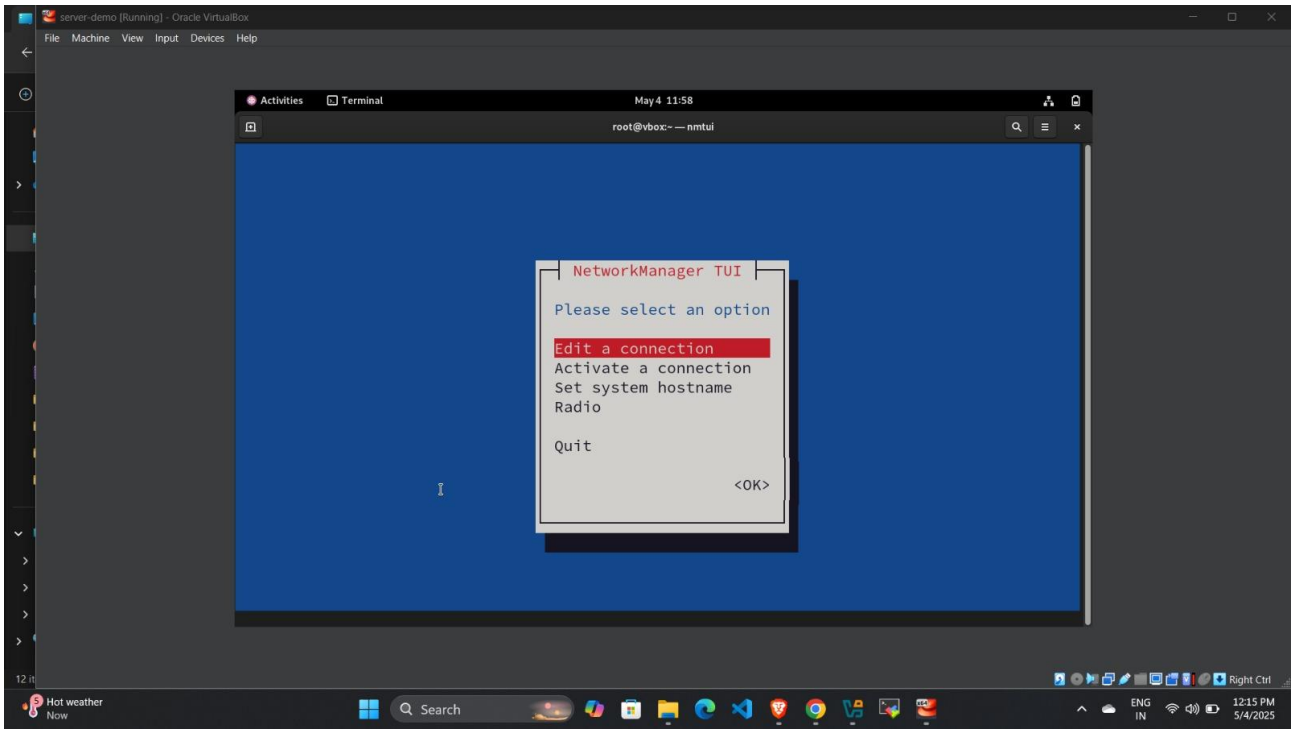


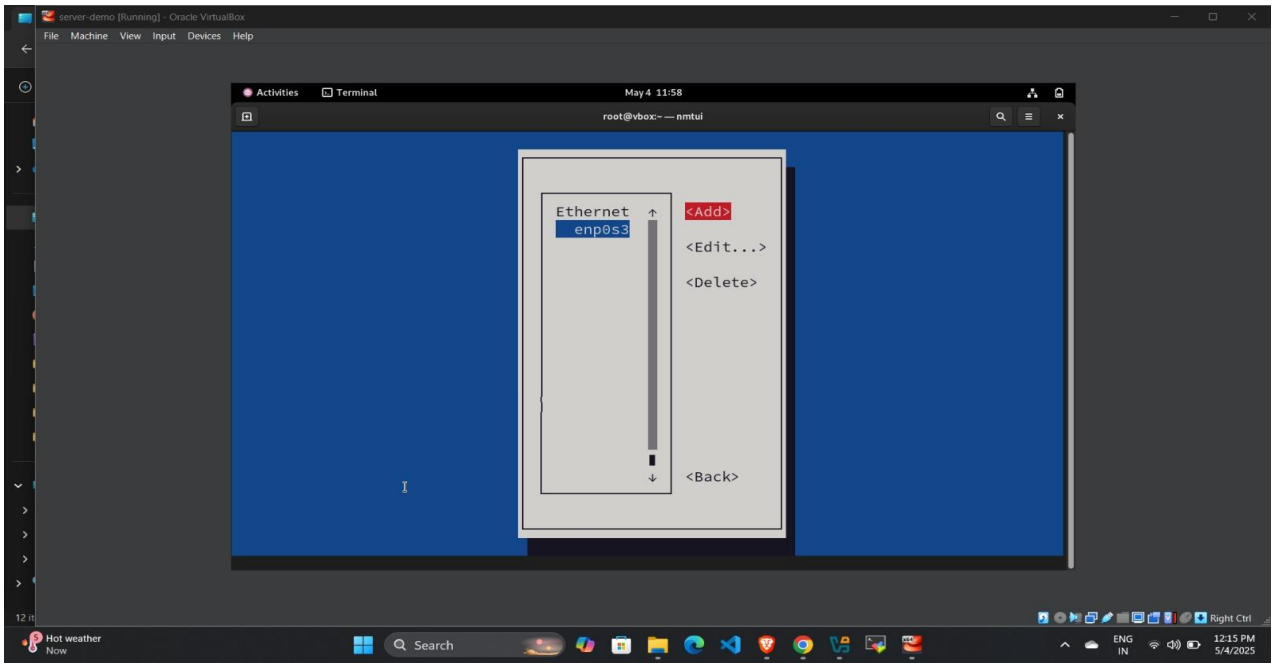
Client Machine

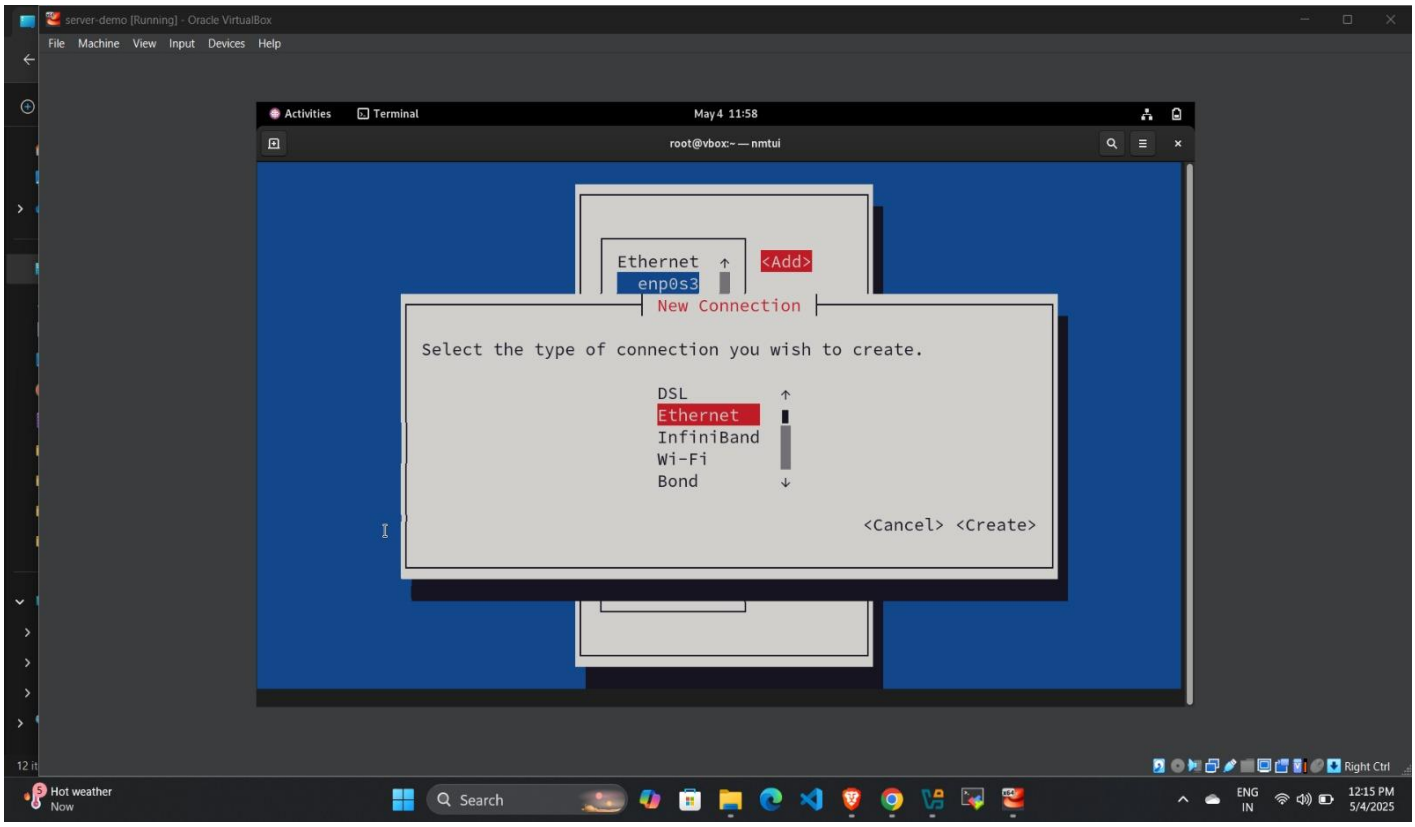


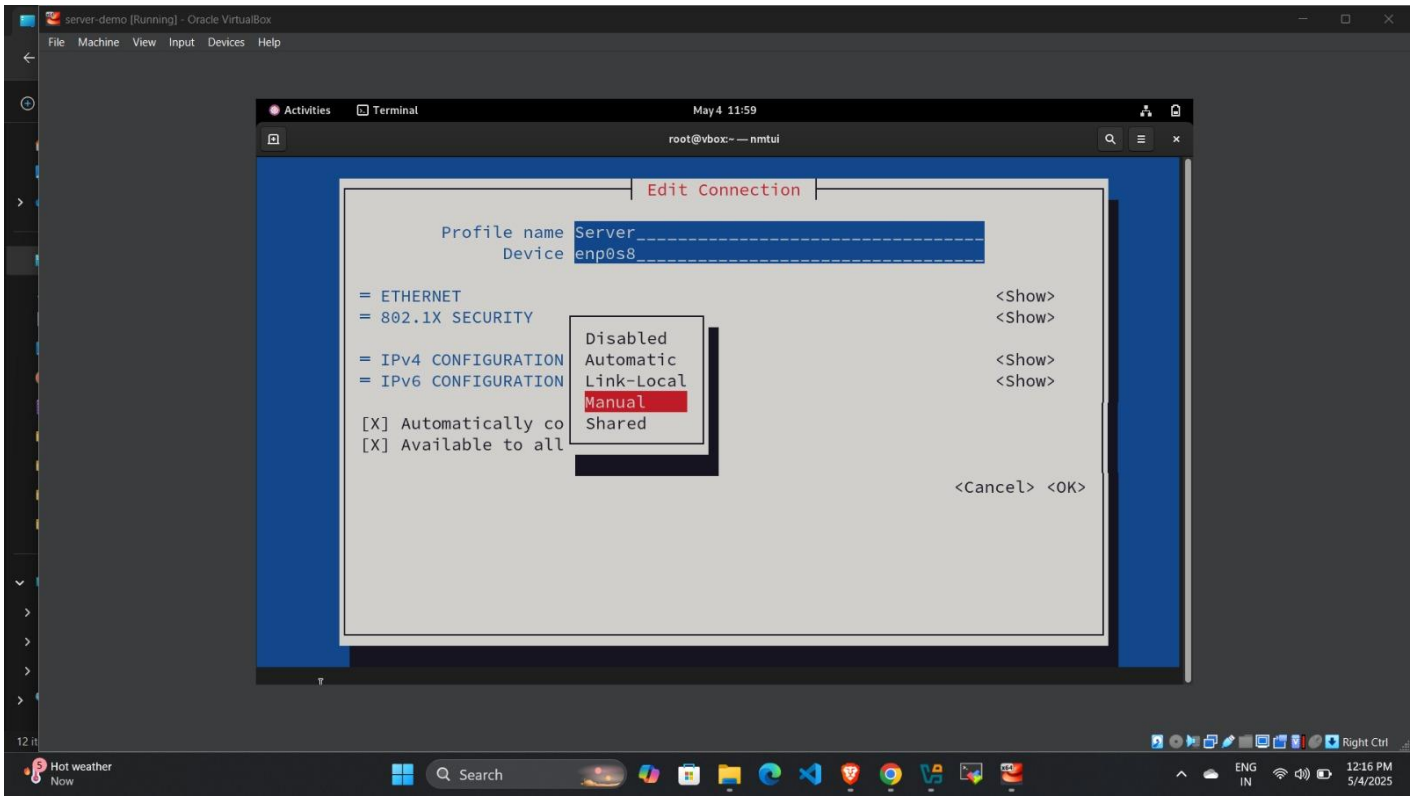
- Use nmtui to assign static IPs:

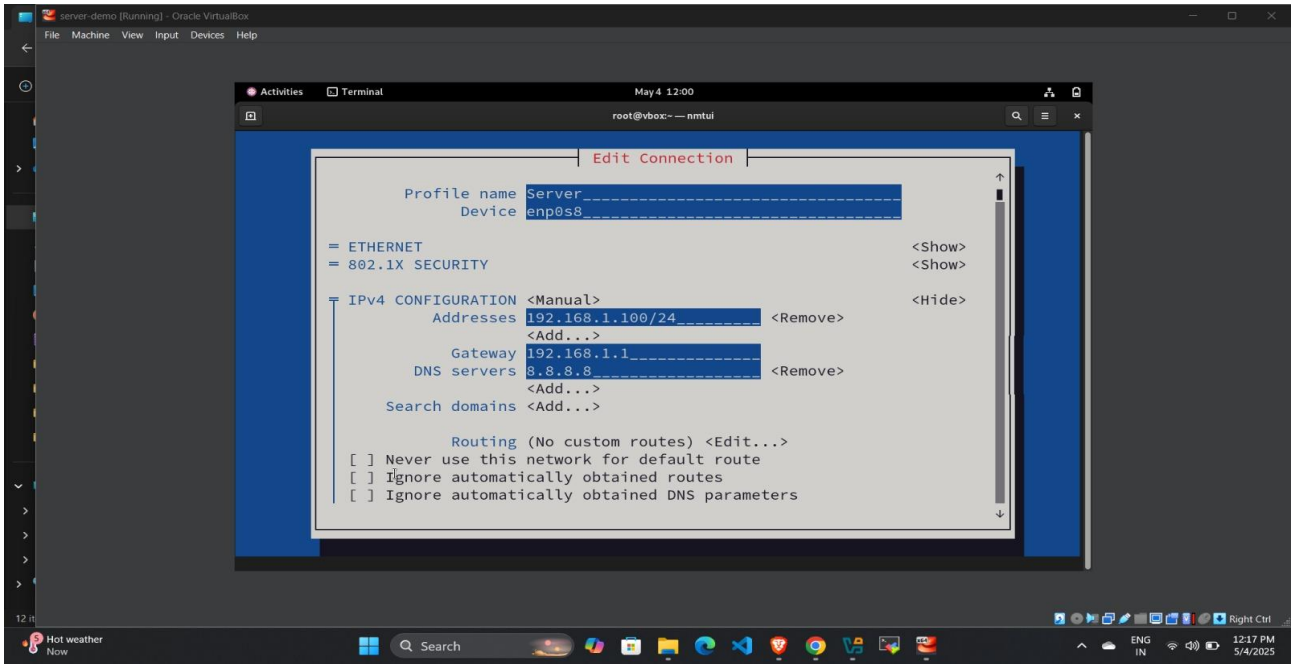
  - Server: 192.168.1.100/24



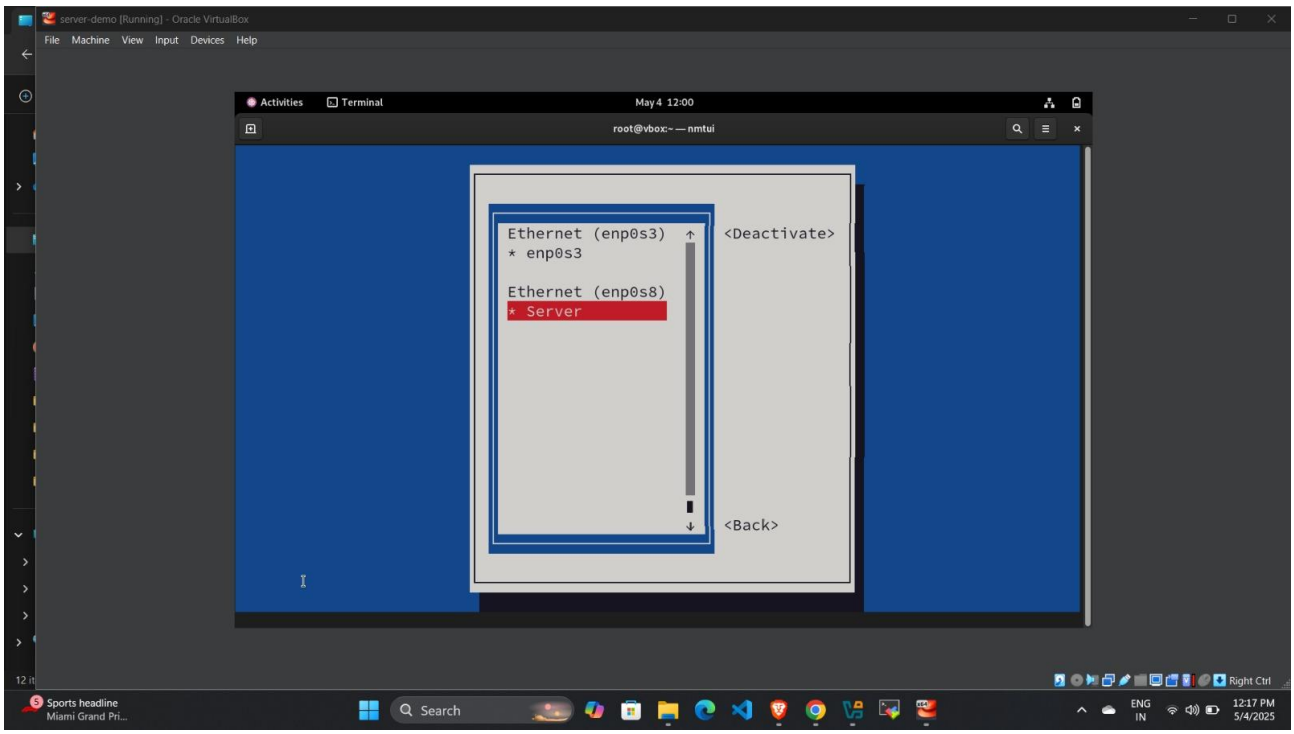



server-demo [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal May 4 11:58
root@vbox:~ — nmtui
Ethernet
enp0s3
<Add>
New Connection
Select the type of connection you wish to create.
DSL
Ethernet
InfiniBand
Wi-Fi
Bond
<Cancel> <Create>
Right Ctrl
Hot weather
Now
Search
ENG
IN
12:15 PM
5/4/2025


server-demo [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Activities Terminal May 4 11:59
root@vbox:~ — nmtui
Edit Connection
Profile name Server
Device enp0s8
= ETHERNET <Show>
= 802.1X SECURITY <Show>
= IPv4 CONFIGURATION <Show>
= IPv6 CONFIGURATION <Show>
Disabled
Automatic
Link-Local
Manual
Shared
[X] Automatically co
[X] Available to all
<Cancel> <OK>
Right Ctrl
Hot weather
Now
Search
ENG
IN
12:16 PM
5/4/2025

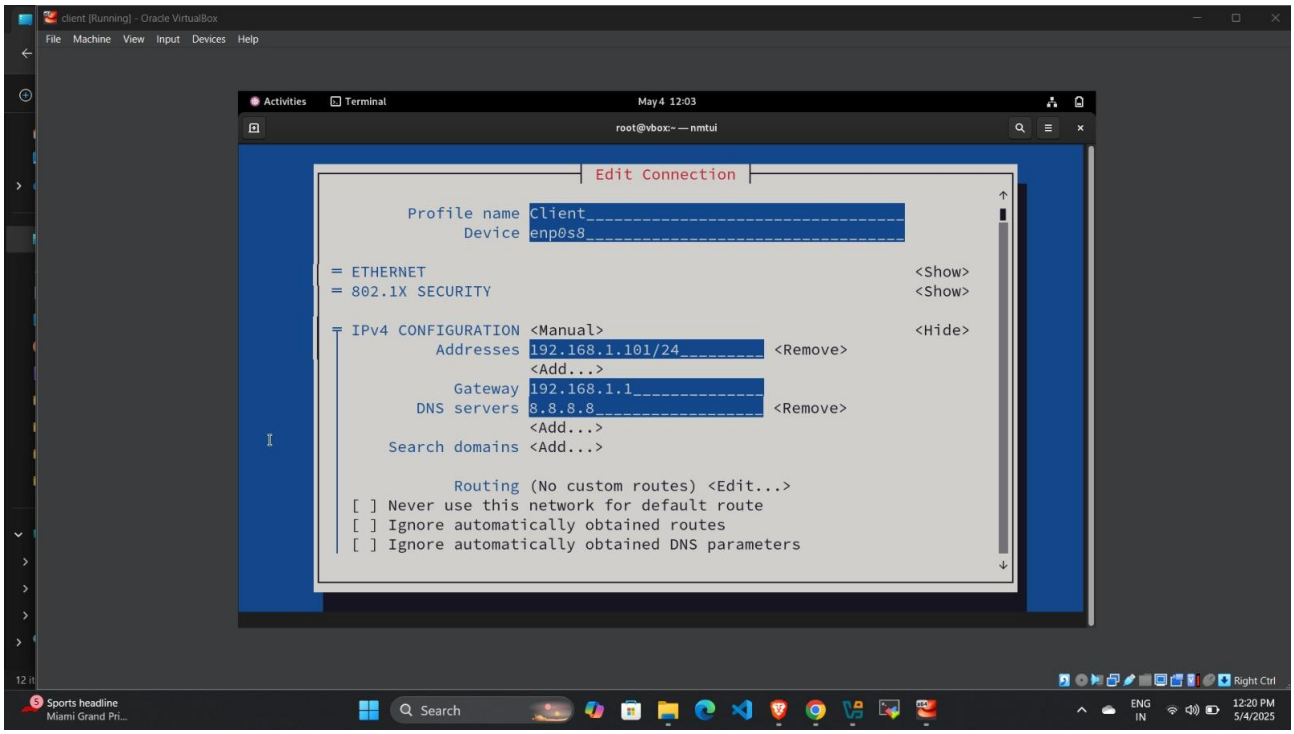


-Here we check activation of server.

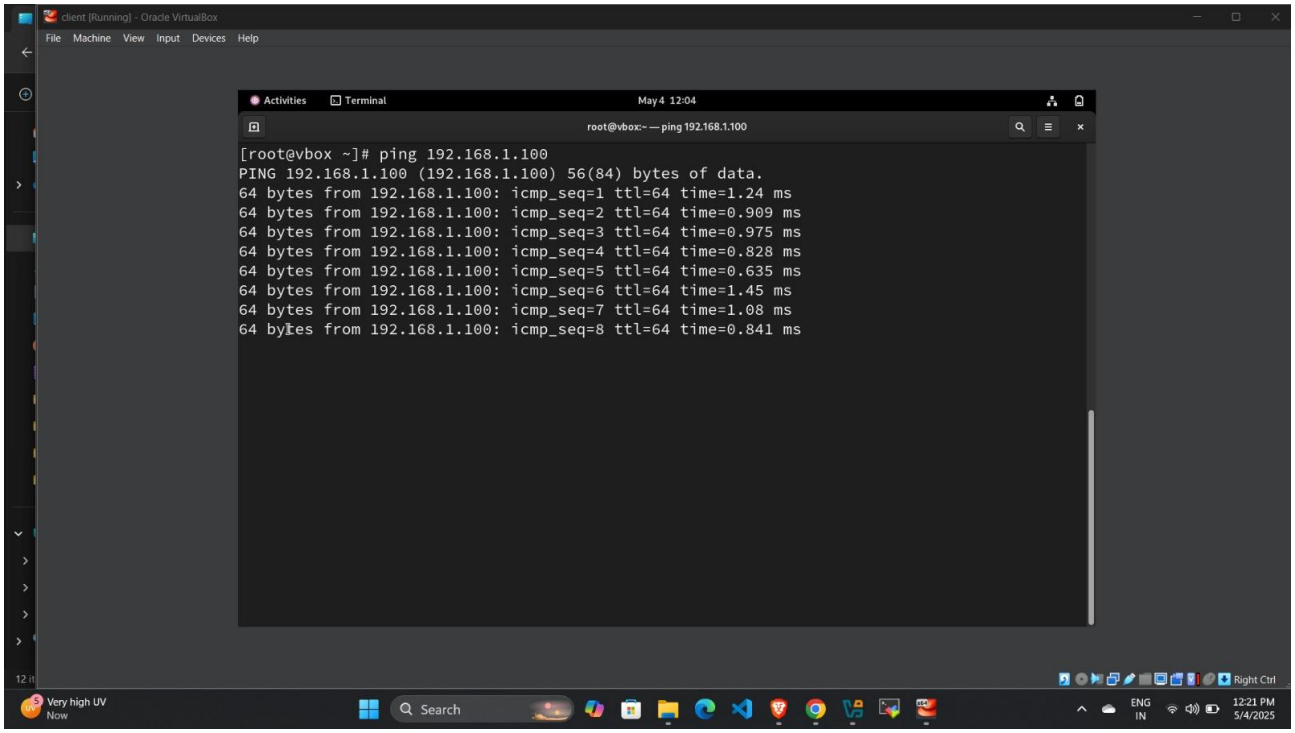


- Just like server we do same things with client machine and give it to ip.

  - Client: 192.168.1.101/24



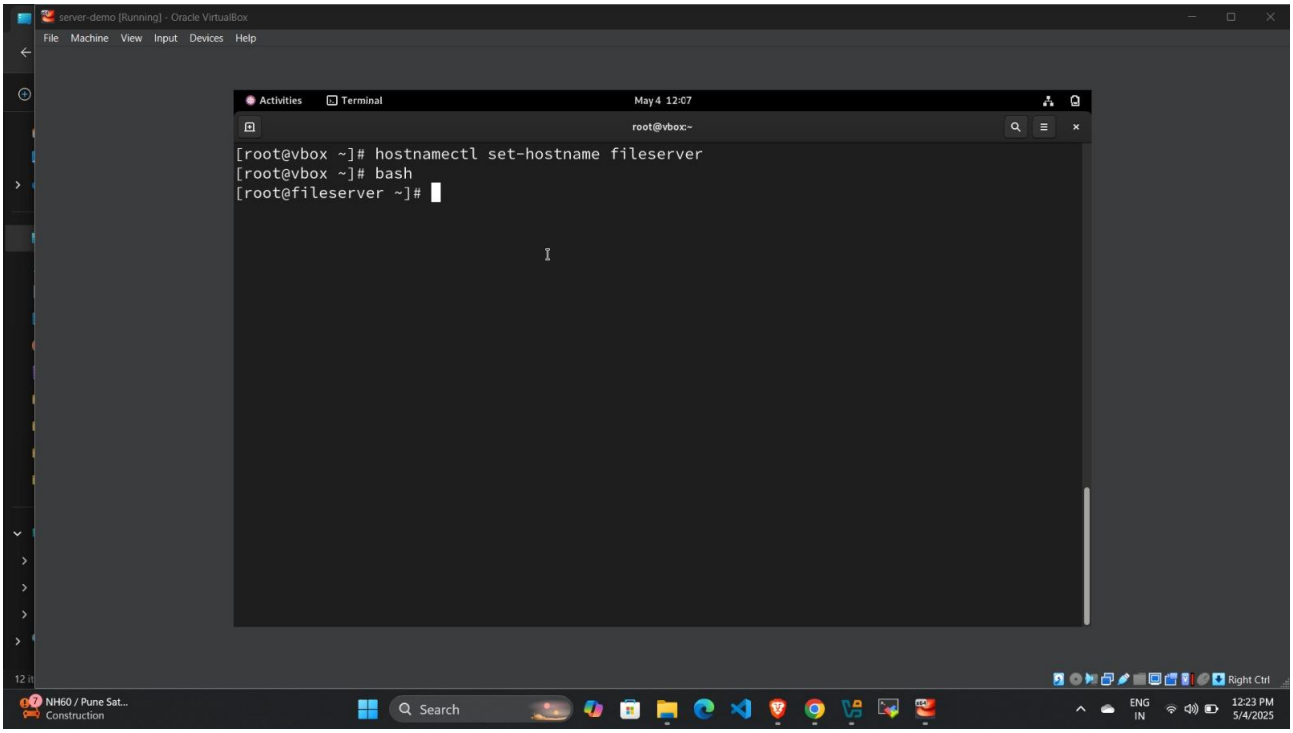
- Here we check connection between server machine and client machine.
- Test connectivity:
  ping 192.168.1.100 from client



PART 3: SERVER CONFIGURATION

1. Set Hostname:

hostnamectl set-hostname fileserver

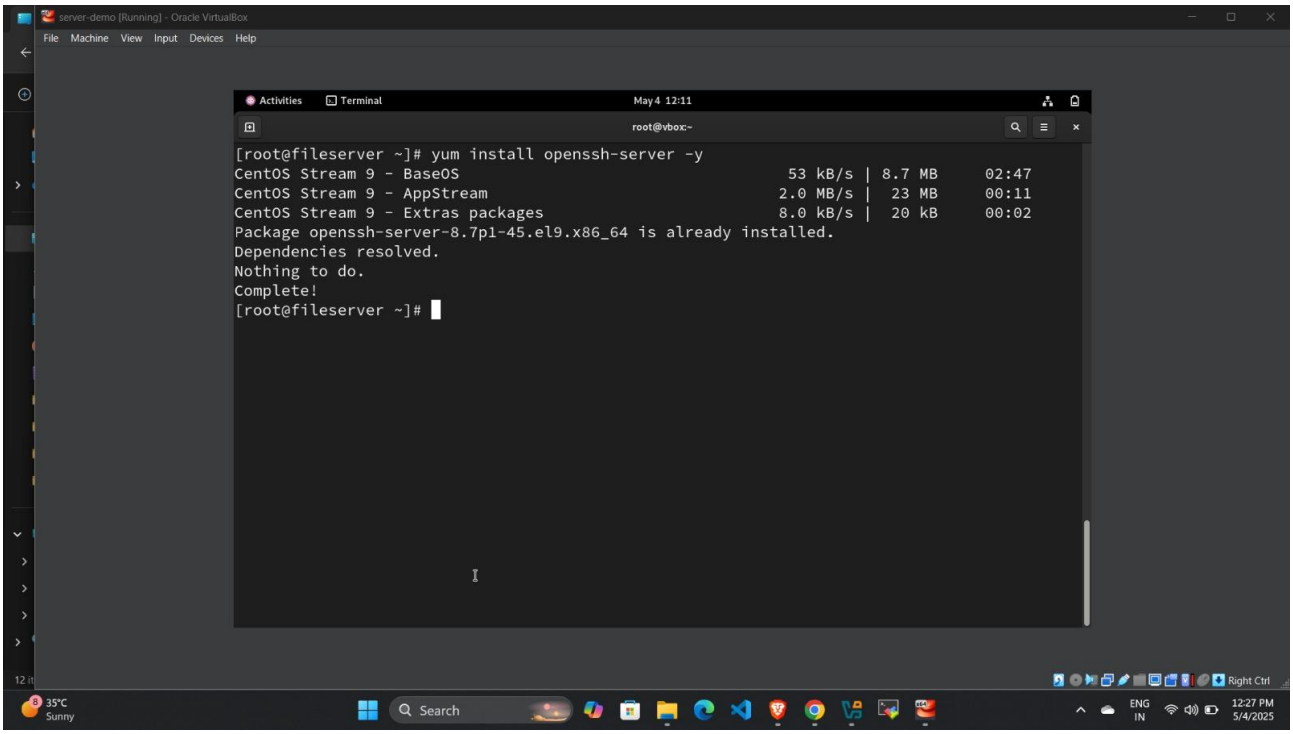


2. Install & Start SSH:

yum install openssh-server -y

systemctl enable sshd

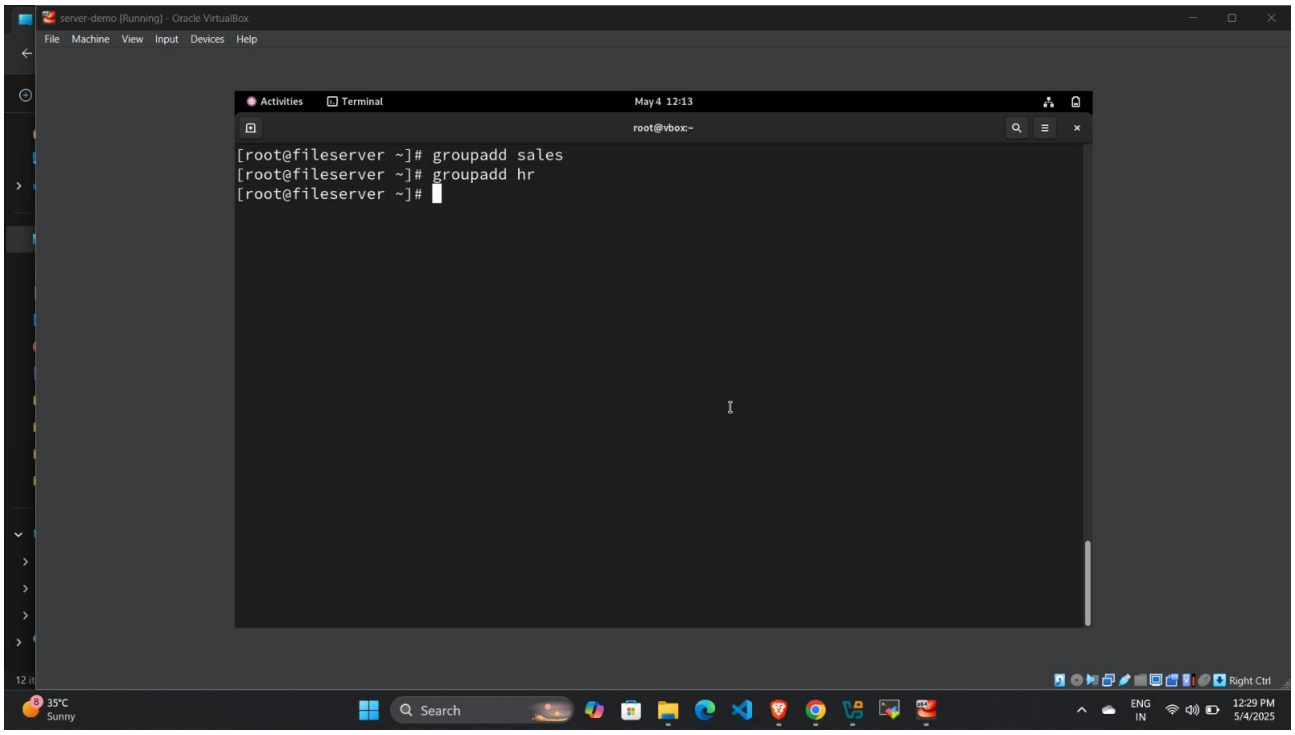
systemctl start sshd



3. Create User Groups:

groupadd sales

groupadd hr



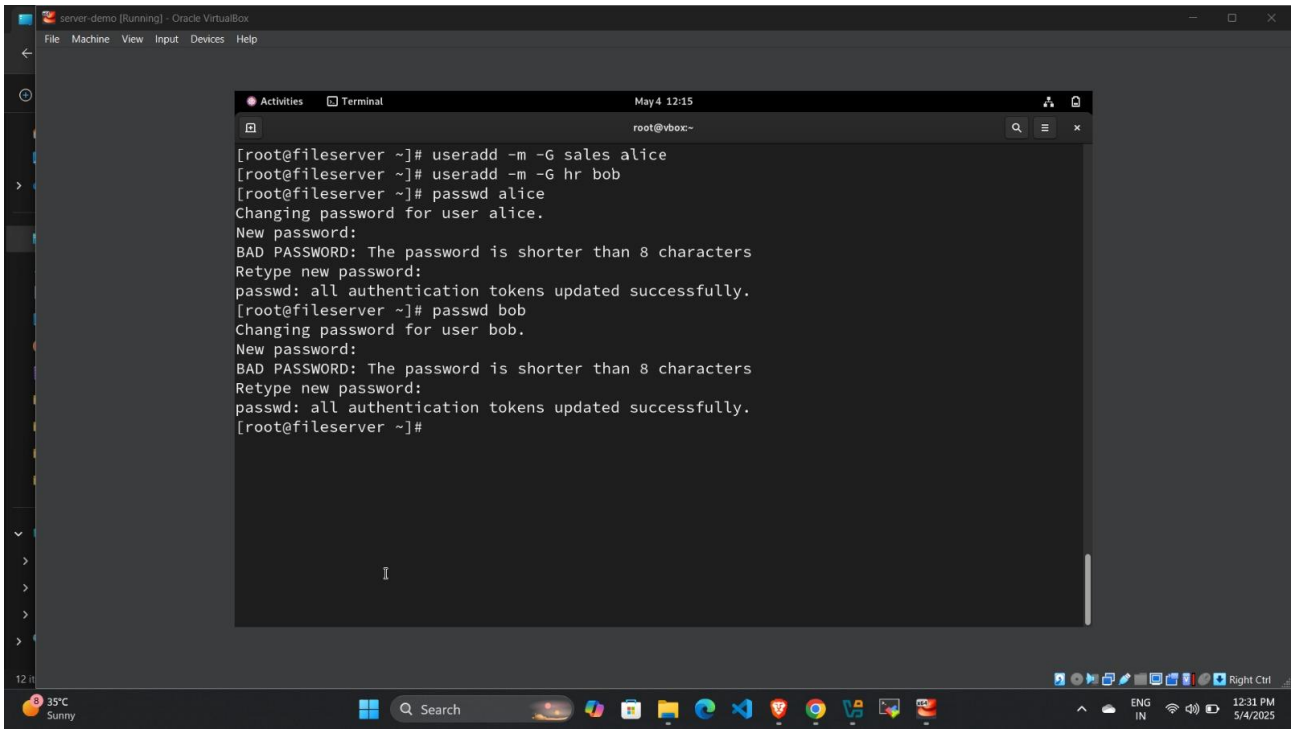
4. Add Users to Groups:

useradd -m -G sales alice

useradd -m -G hr bob

passwd alice

passwd bob



5. Create Shared Folders:

mkdir /srv/sales

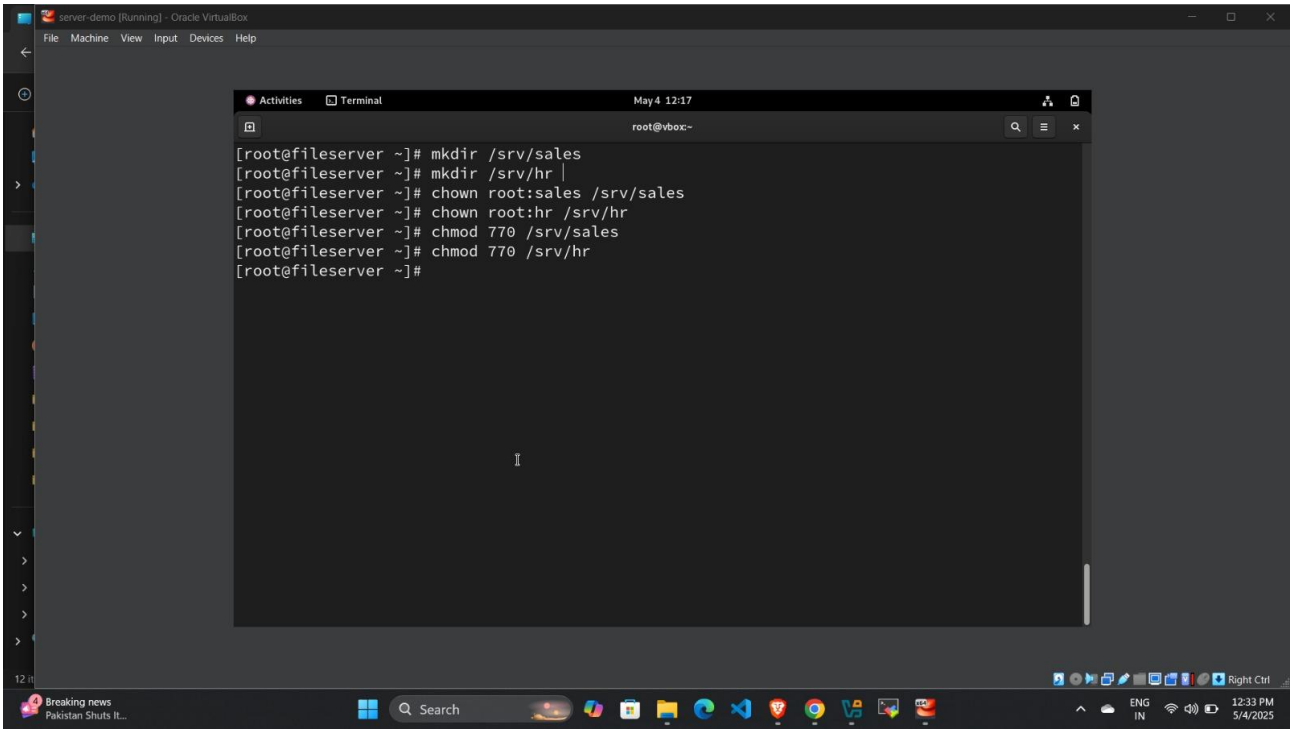
mkdir /srv/hr

chown root:sales /srv/sales

chown root:hr /srv/hr

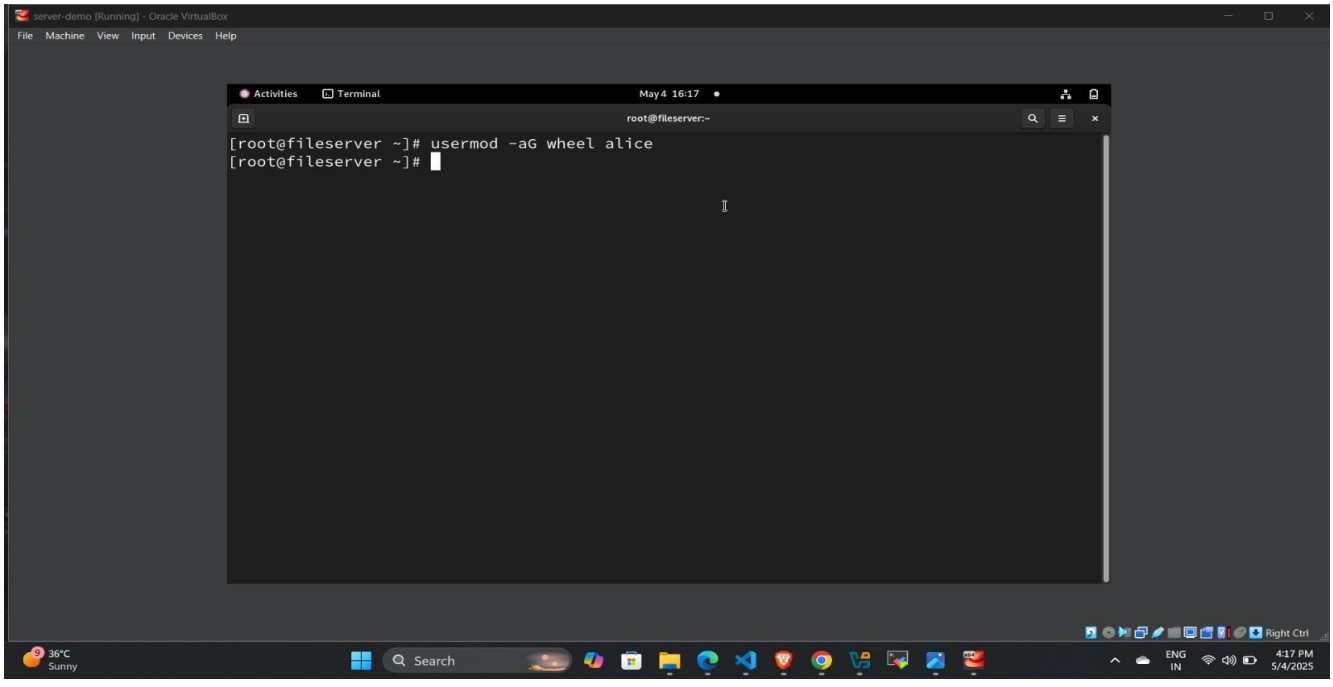
chmod 770 /srv/sales

chmod 770 /srv/hr



6. Optional: Sudo Access

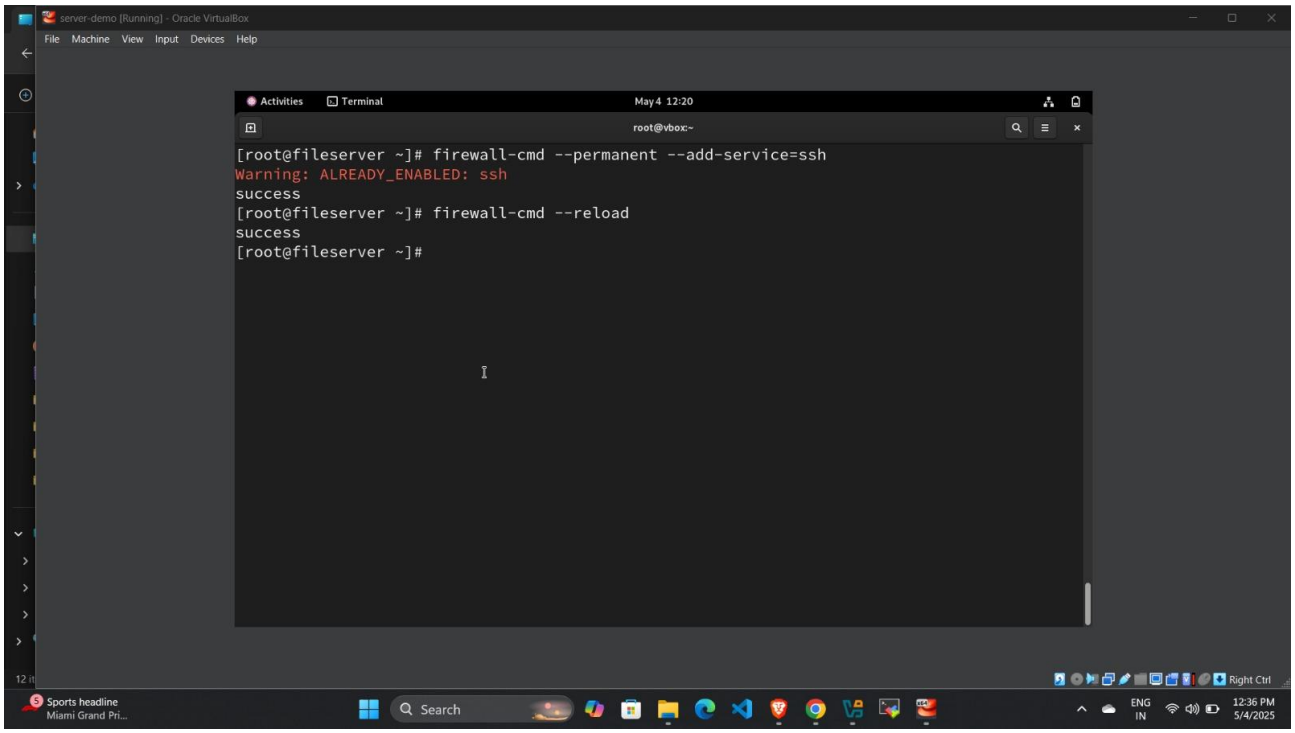
usermod -aG wheel alice



7. Enable Firewall SSH Port:

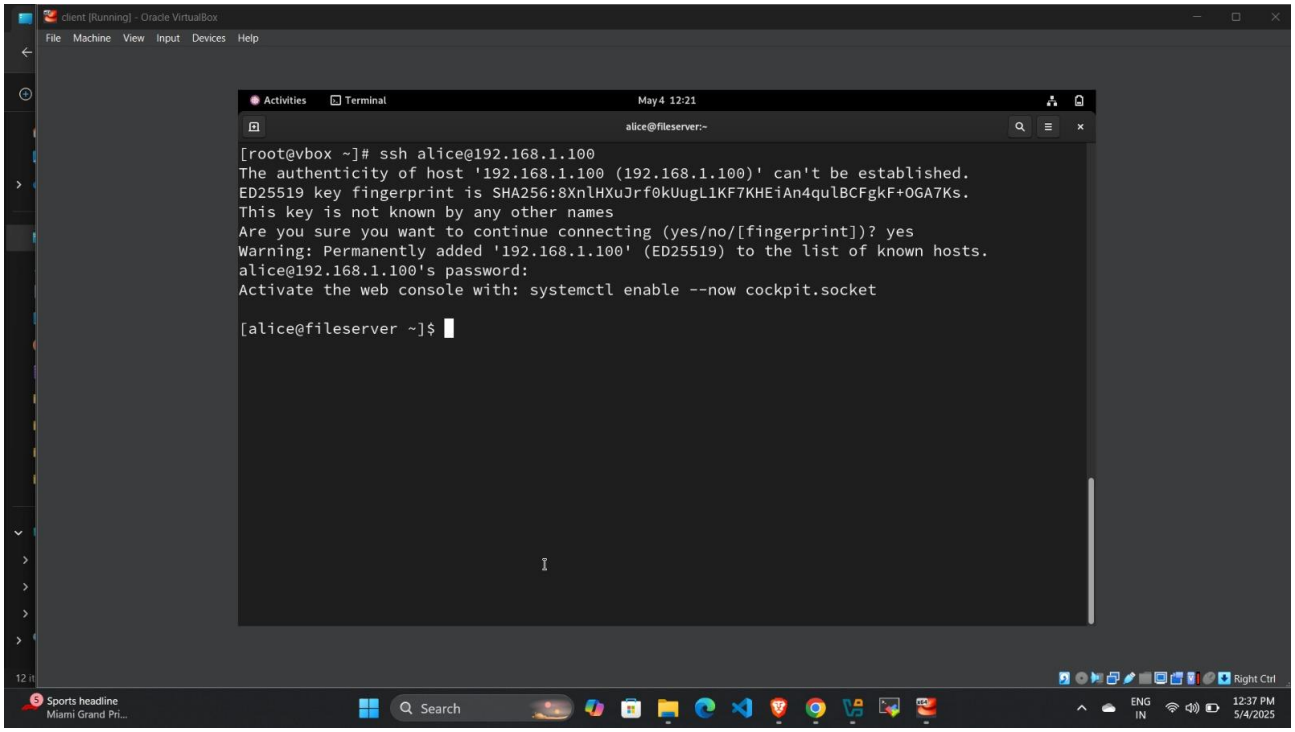
firewall-cmd --permanent --add-service=ssh

firewall-cmd –reload

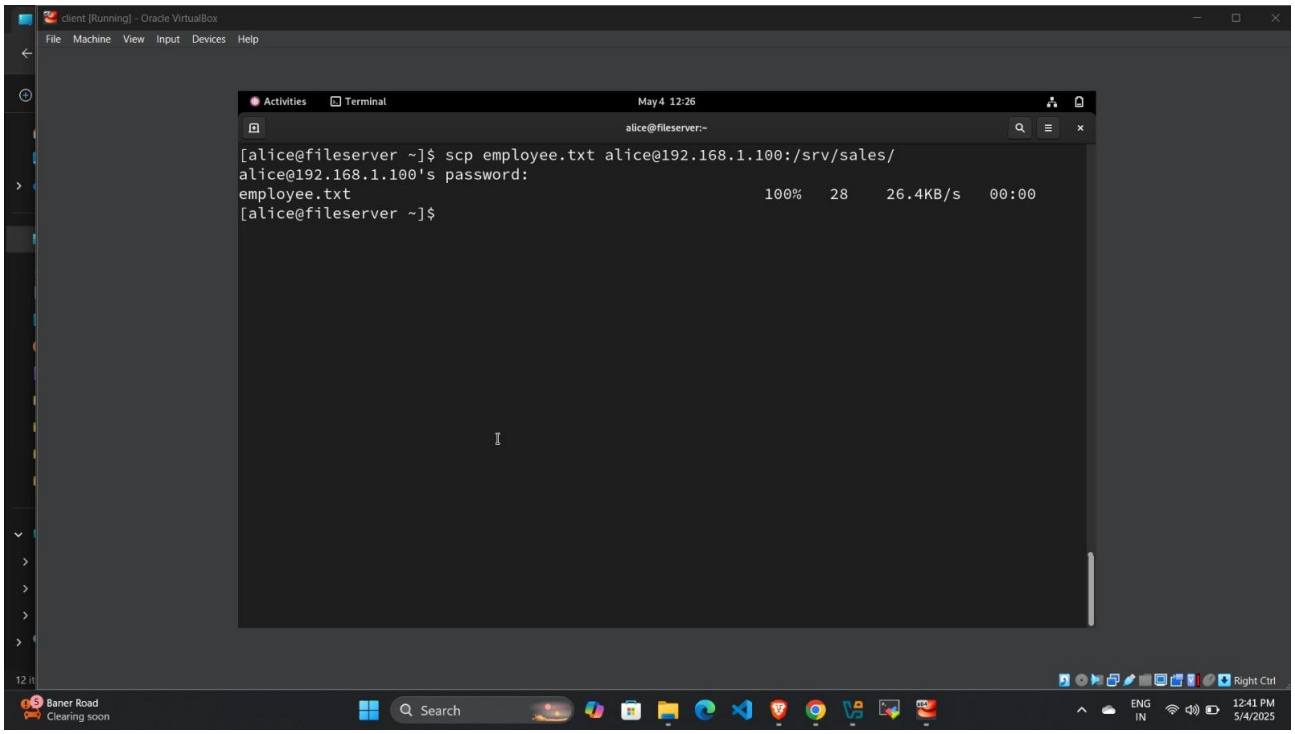


PART 4: CLIENT MACHINE USAGE: Take access of server machine’s in client machine.

- Connect via SSH: ssh alice@192.168.1.100

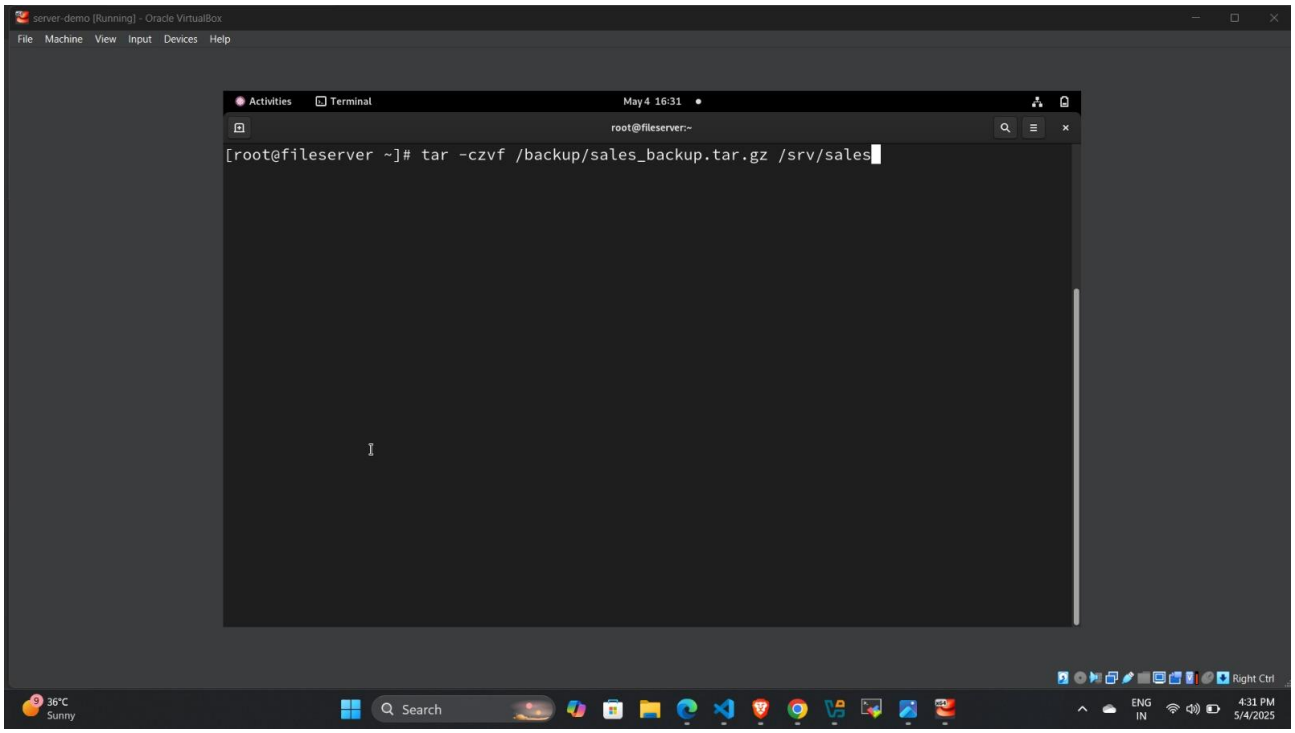


- Transfer Files: scp employee.txt alice@192.168.1.100:/srv/sales/
- Here we transfer file from client to server.



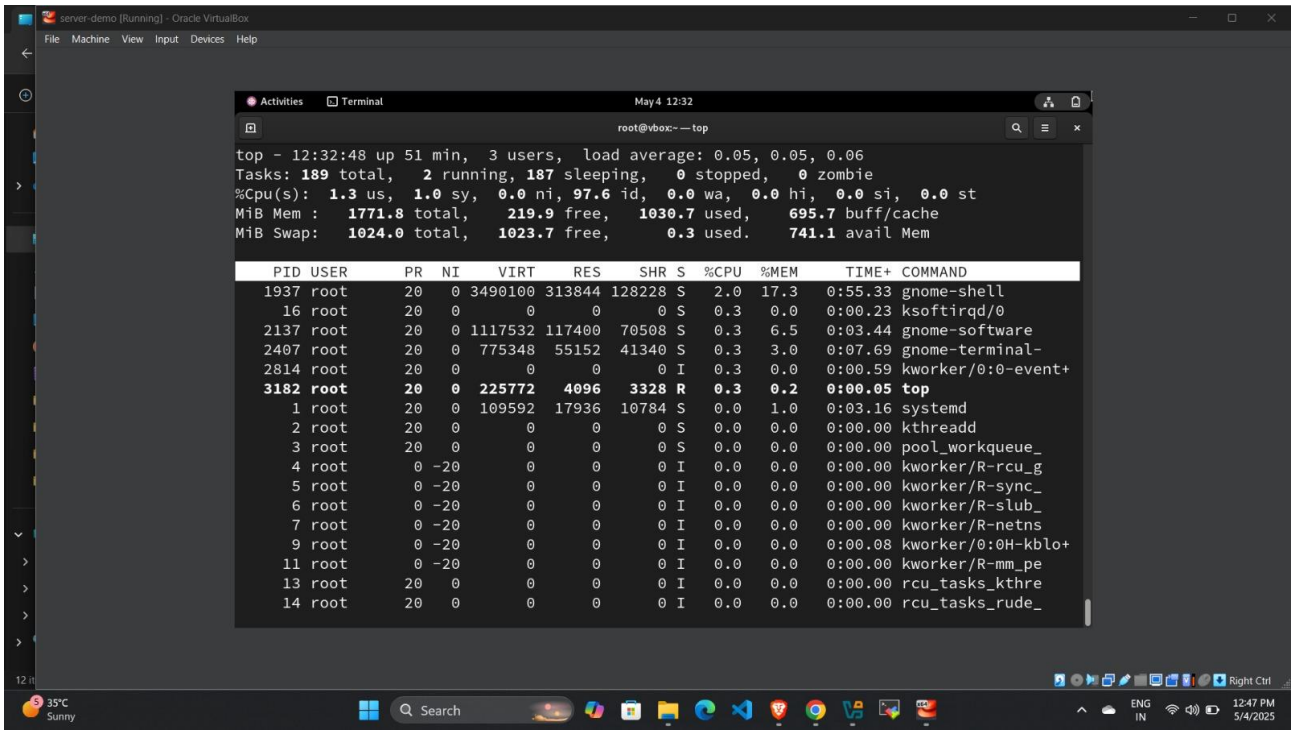
PART 5: BACKUP & MONITORING (Server)

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

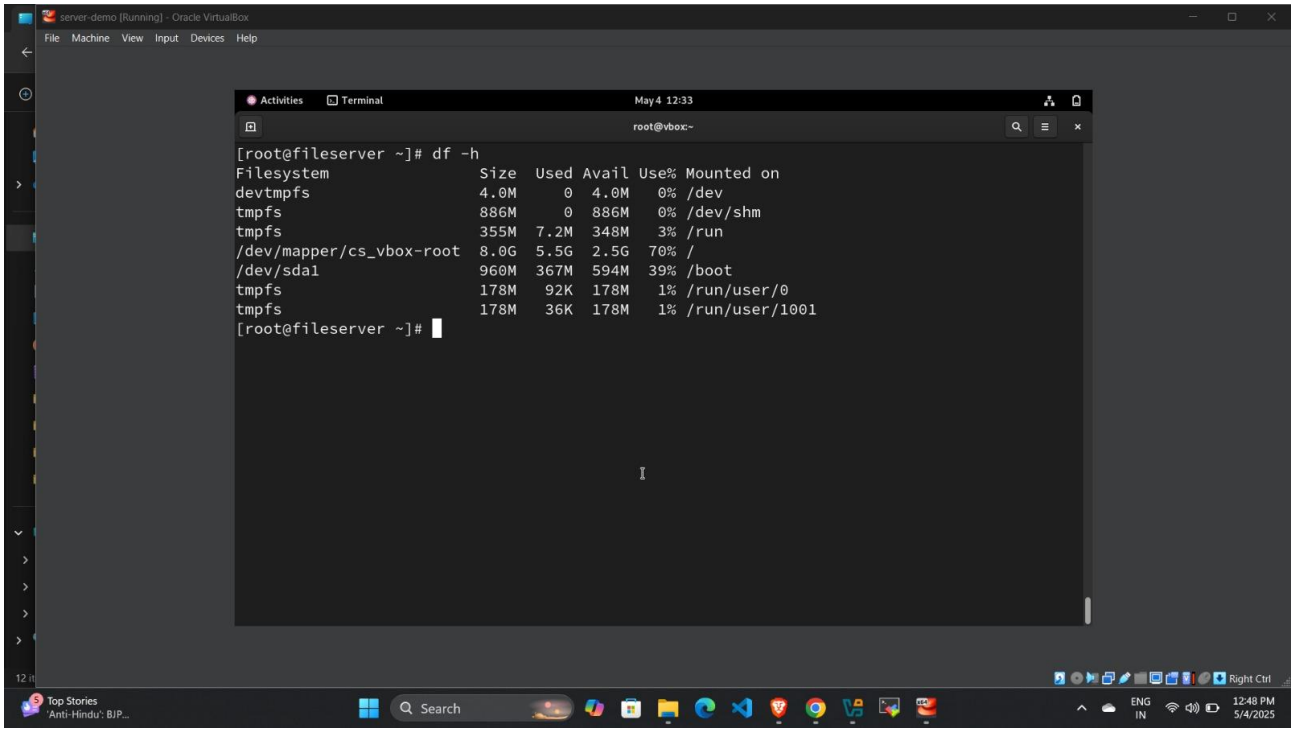


- Monitoring files.

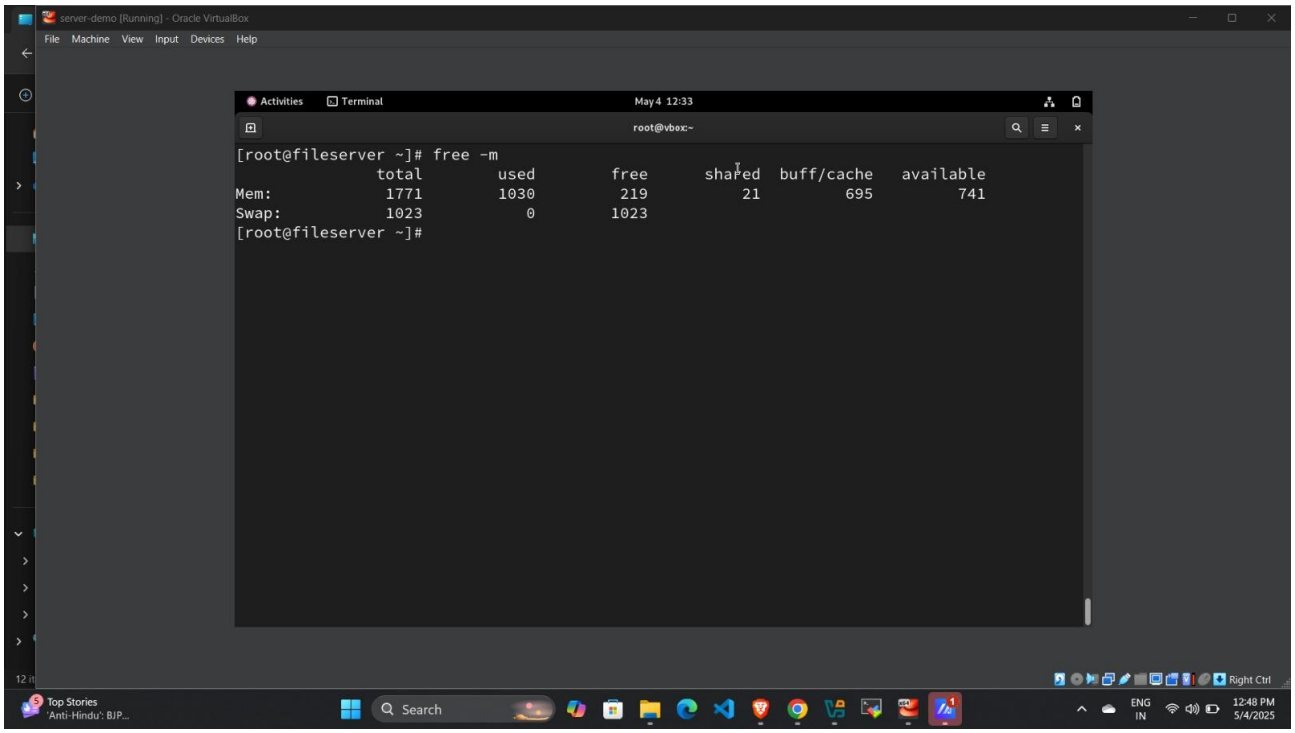
- top



- df -h



- free -m



- yum update