AeroAspire

SDE Intern

Goutham V

Week 5 – Day1 (21st October)

Task:

Install Multipass, Launch Ubuntu VM, SSH into VM, Practice CLI commands (ls, cd, touch, nano, rm, chmod, sudo), Install Python, Git, curl, net-tools, Setup simple folder & run Python script inside.

Reflections,

1. What is the difference between VM and container?

A VM (Virtual Machine) is like a full computer running inside your computer. It has its own operating system, memory, and virtual hardware. It's heavier and slower to start but fully isolated.

A container is like a lightweight package that shares your main OS but has everything it needs to run an application. It starts faster, uses less memory, and is easier to deploy multiple apps.

So basically: VM = full OS, Container = isolated app environment.

2. How do permissions work in Linux? Explain chmod and chown.

Linux files and folders have permissions: read (r), write (w), execute (x) for owner, group, and others.

- **chmod changes who can read/write/execute a file or folder.**
  Example: chmod 755 hello.py → owner can read/write/execute, others can read/execute.
- **chown changes the owner or group of a file/folder.**
  Example: chown gouth:gouth hello.py → sets both owner and group to user gouth.

Permissions ensure security and control over who can do what on your files.

3. Why use apt and sudo? What happens behind the scenes when updating packages?

- **apt is the package manager in Ubuntu. It installs, updates, or removes software from repositories.**
- **sudo allows you to run commands as administrator since installing/updating software needs special permissions.**

When you run sudo apt update:

a) **Your system checks repositories for the latest package info.**
b) **Downloads the package list.**
c) **Updates internal database of packages.**

4. Command flow: create project folder → create script → run it → make it executable → delete folder

```
mkdir my_project       # create folder
cd my_project          # go into folder
nano hello.py          # create script
# write: print("Hello World!") and save
python3 hello.py       # run script
chmod +x hello.py      # make script executable
./hello.py             # run as executable
cd ..                  # go out of folder
rm -r my_project       # delete folder
```

This flow is **simple and clear**, showing how you create, run, and remove projects safely.