

## To change the user name in Linux:

1. check your user name

```
>whoami
```

2. Create a temporary user

```
>sudo adduser tempadmin
```

3. Add the new user to sudoers

```
sudo usermod -aG sudo tempadmin
```

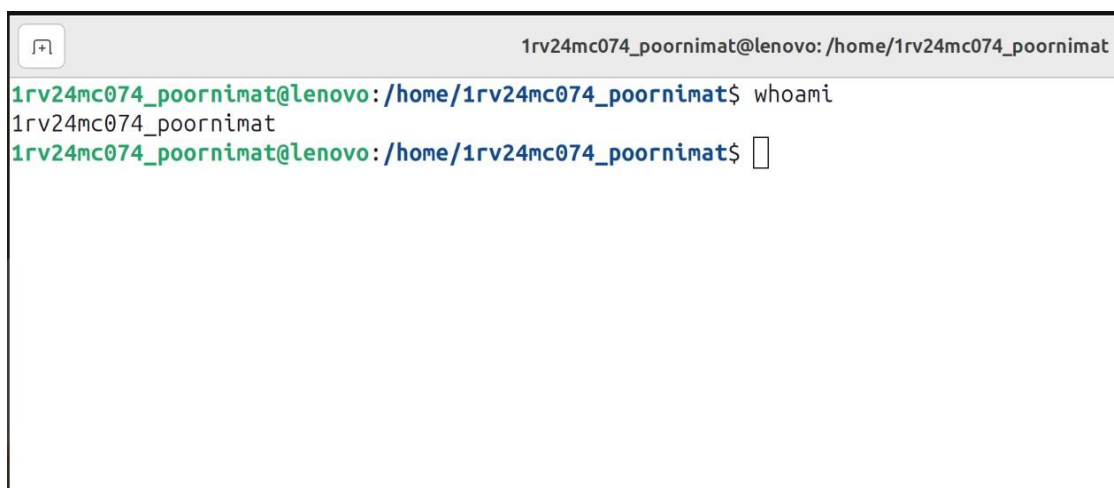
4. Logout from the current user via GUI and login to tempadmin

5. In the terminal of tempadmin,

```
>sudo usermod -l 1RV24MC074_poornimat fullmoon
```

6. Now type the below command in original user after logging again via GUI:

```
>whoami
```

A terminal window screenshot with a title bar showing the user '1rv24mc074\_poornimat@lenovo' and the path '/home/1rv24mc074\_poornimat'. The terminal text shows the command '1rv24mc074\_poornimat@lenovo: /home/1rv24mc074\_poornimat\$ whoami' being executed, followed by the output '1rv24mc074\_poornimat' on the next line. The prompt '1rv24mc074\_poornimat@lenovo: /home/1rv24mc074\_poornimat\$' is shown again on the third line with a cursor.

```
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$ whoami
1rv24mc074_poornimat
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$
```

# TO INSTALL DOCKER USING APT REPO

1. Set up Docker's at repository.

#Add Docker's official GPG key:

```
sudo apt-get update
```

```
sudo apt-get install ca-certificates curl
```

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

#Add the repository to Apt sources:

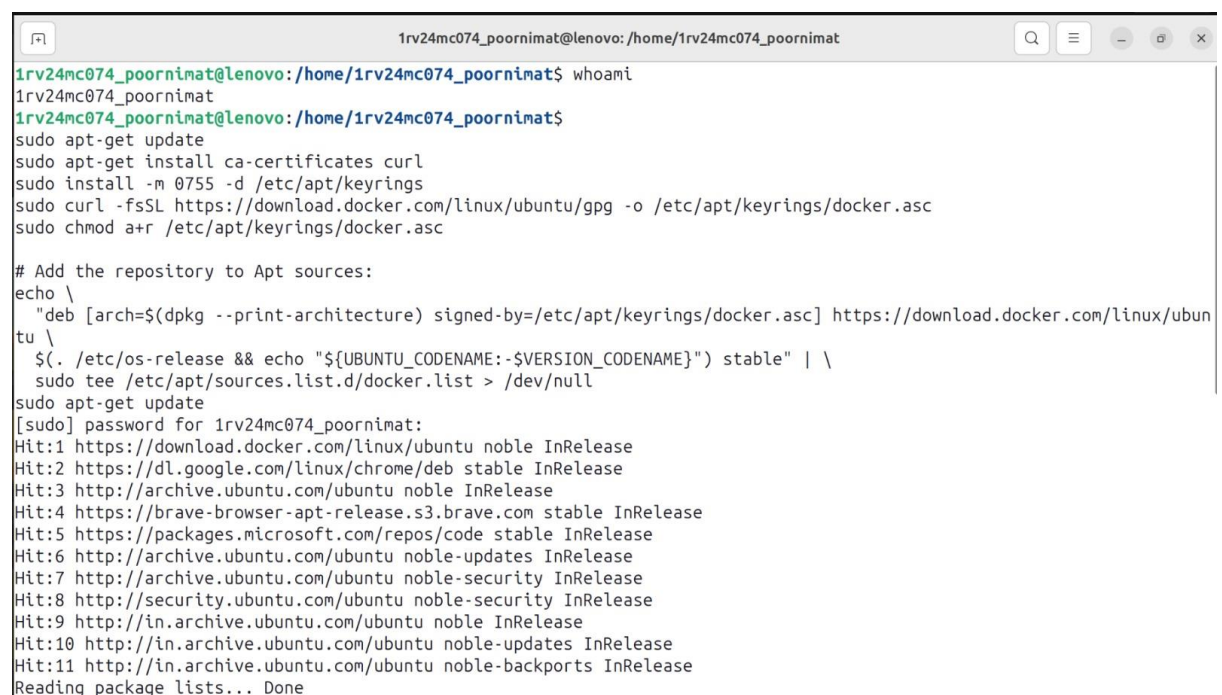
```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
```

```
$(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
```

```
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
```

```
sudo apt-get update
```



A terminal window screenshot showing the installation of Docker on Ubuntu. The terminal output includes the following commands and their results:

```
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$ whoami
1rv24mc074_poornimat
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
[sudo] password for 1rv24mc074_poornimat:
Hit:1 https://download.docker.com/linux/ubuntu noble InRelease
Hit:2 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:3 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:4 https://brave-browser-apt-release.s3.brave.com stable InRelease
Hit:5 https://packages.microsoft.com/repos/code stable InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:7 http://archive.ubuntu.com/ubuntu noble-security InRelease
Hit:8 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:9 http://in.archive.ubuntu.com/ubuntu noble InRelease
Hit:10 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:11 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Reading package lists... Done
```

## 2. Install the Docker packages.

> sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin

docker-compose-plugin

```
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat
Hit:1 https://download.docker.com/linux/ubuntu noble InRelease
Hit:2 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://archive.ubuntu.com/ubuntu noble InRelease
Hit:5 http://in.archive.ubuntu.com/ubuntu noble InRelease
Hit:6 http://archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:7 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:8 http://archive.ubuntu.com/ubuntu noble-security InRelease
Hit:9 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:10 https://packages.microsoft.com/repos/code stable InRelease
Hit:11 https://brave-browser-apt-release.s3.brave.com stable InRelease
Reading package lists... Done
N: Skipping acquire of configured file 'main/binary-i386/Packages' as repository 'https://brave-browser-apt-release.s3.brave.com stable InRelease' doesn't support architecture 'i386'
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$ sudo apt-get install docker-ce docker-ce-cli containerd.io docke
r-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker-ce is already the newest version (5:28.5.1-1~ubuntu.24.04~noble).
docker-ce-cli is already the newest version (5:28.5.1-1~ubuntu.24.04~noble).
containerd.io is already the newest version (1.7.28-1~ubuntu.24.04~noble).
docker-buildx-plugin is already the newest version (0.29.1-1~ubuntu.24.04~noble).
docker-compose-plugin is already the newest version (2.40.0-1~ubuntu.24.04~noble).
The following packages were automatically installed and are no longer required:
  android-libbase android-libboringssl android-libcutils android-liblog android-libziparchive
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 253 not upgraded.
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$
```

## 3. Check docker status

sudo systemctl status docker

```
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat
docker-compose-plugin is already the newest version (2.40.0-1~ubuntu.24.04~noble).
The following packages were automatically installed and are no longer required:
  android-libbase android-libboringssl android-libcutils android-liblog android-libziparchive
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 253 not upgraded.
1rv24mc074_poornimat@lenovo: /home/1rv24mc074_poornimat$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-10-15 16:29:37 IST; 9min ago
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 2061 (dockerd)
      Tasks: 18
     Memory: 97.8M (peak: 100.8M)
        CPU: 836ms
     CGroup: /system.slice/docker.service
             └─2061 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Oct 15 16:29:36 lenovo dockerd[2061]: time="2025-10-15T16:29:36.657088393+05:30" level=info msg="Creating a containerd
Oct 15 16:29:36 lenovo dockerd[2061]: time="2025-10-15T16:29:36.673346799+05:30" level=info msg="[graphdriver] using pr
Oct 15 16:29:36 lenovo dockerd[2061]: time="2025-10-15T16:29:36.674540829+05:30" level=info msg="Loading containers: st
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.541899231+05:30" level=info msg="Loading containers: do
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.558594614+05:30" level=info msg="Docker daemon" commit=>
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.558798077+05:30" level=info msg="Initializing buildkit"
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.568881711+05:30" level=info msg="Completed buildkit ini
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.577038316+05:30" level=info msg="Daemon has completed i
Oct 15 16:29:37 lenovo dockerd[2061]: time="2025-10-15T16:29:37.577192053+05:30" level=info msg="API listen on /run/doc
Oct 15 16:29:37 lenovo systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-22/22 (END)
```

## . Run a sample container

```
1rv24mc074_poornimat@lenovo:~/honey$ cd 1rv24mc074_poornimat  
1rv24mc074_poornimat@lenovo:~/home/1rv24mc074_poornimat$ sudo docker run hello-world
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it