Username Change & Docker Installation

Step 1: Check Current Username

$whoami

sandarsh

Step 2: Create a Temporary Admin User

$sudo adduser admin_temp
123456

set a password and press Enter for other fields.

Step 3: Give Sudo Privileges to the New User

$sudo usermod -aG sudo admin_temp

Step 4: Switch to the New Admin User

$su - admin_temp or through gui

Check your current user:
$whoami

admin_temp

Step 5: Rename the Original User
`sandarsh` → `Sandarsh_J_N_1RV24MC093`

$sudo usermod -l Sandarsh_J_N_1RV24MC093 sandarsh

Step 6: Rename the Home Directory

$sudo mv /home/sandarsh /home/Sandarsh_J_N_1RV24MC093

$sudo usermod -d /home/Sandarsh_J_N_1RV24MC093 -m Sandarsh_J_N_1RV24MC093

Step 7: Fix Ownership Permissions

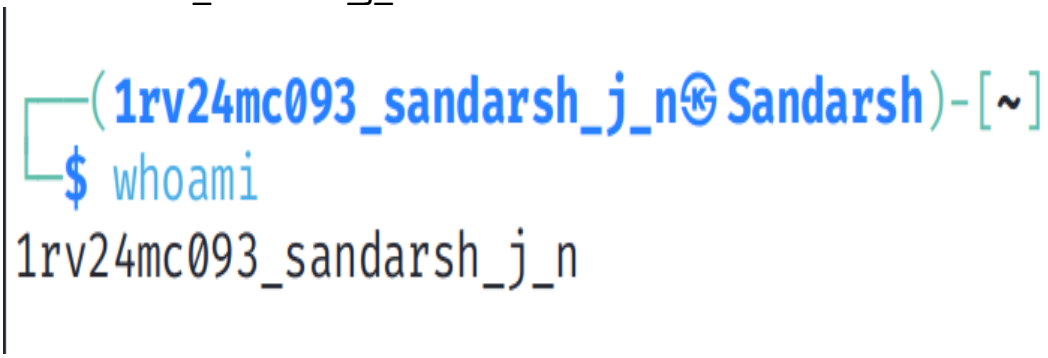
$sudo chown -R Sandarsh_J_N_1RV24MC093:Sandarsh_J_N_1RV24MC093 /home/Sandarsh_J_N_1RV24MC093

Step 8: Test the Username Change
Log out from admin_temp then log in again as:

Username: 1rv24mc093_sandarsh_j_n
Password: ****

Then open a terminal:

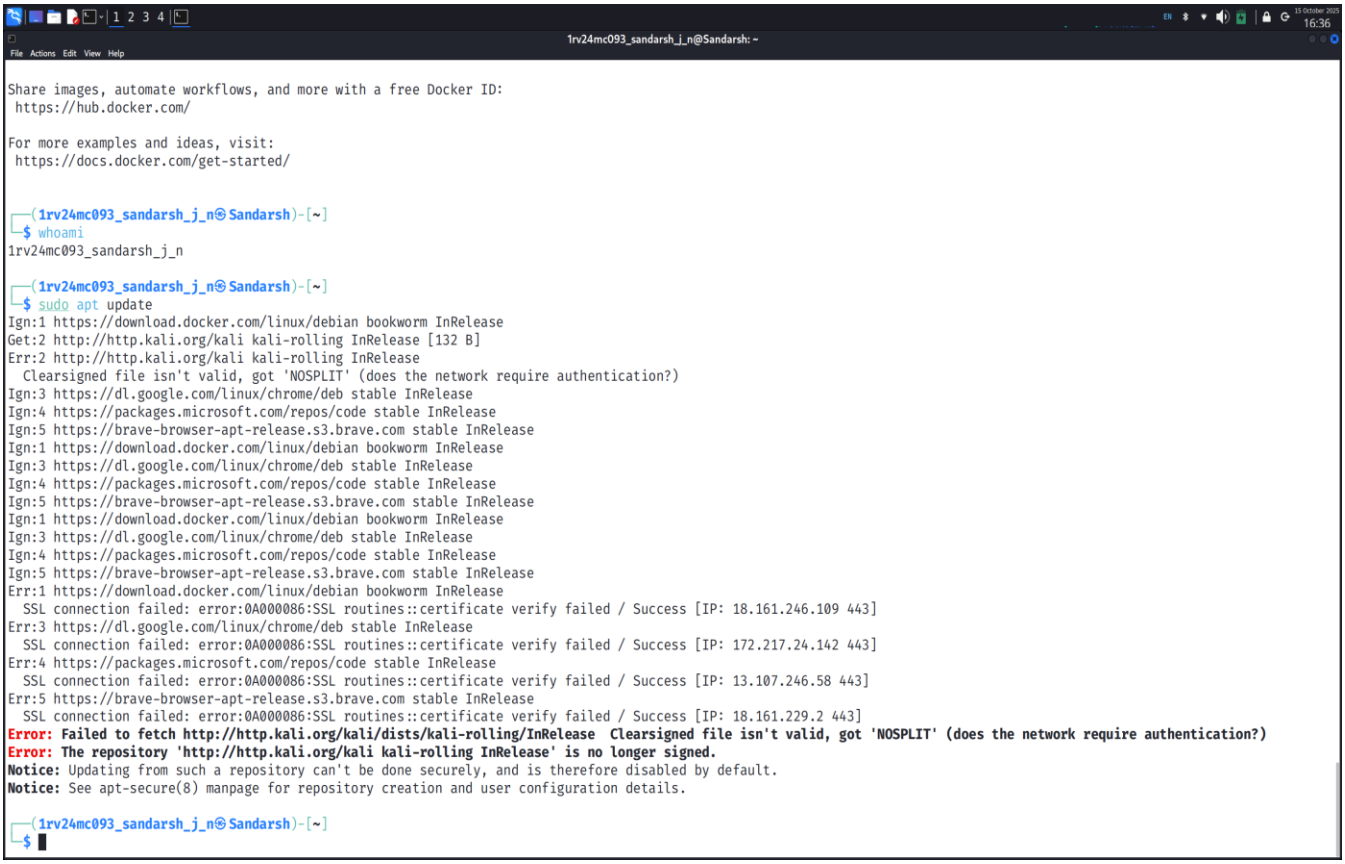
$whoami
1rv24mc093_sandarsh_j_n



Docker Installation and Testing on Kali Linux

Step 1: Update Package List

$sudo apt update



Step 2: Install Required Dependencies

$sudo apt-get install ca-certificates curl

```
(1rv24mc093_sandarsh_j_n㉿Sandarsh)-[~]
$ sudo apt-get install ca-certificates curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20250419).
curl is already the newest version (8.15.0-1).
The following packages were automatically installed and are no longer required:
  docker-ce-rootless-extras libslirp0 pigz slirp4netns
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1409 not upgraded.
```

Step 3: Add Docker’s Official GPG Key

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

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

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

```
┌──(1rv24mc093_sandarsh_j_n㉿Sandarsh)-[~]
└─$ sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

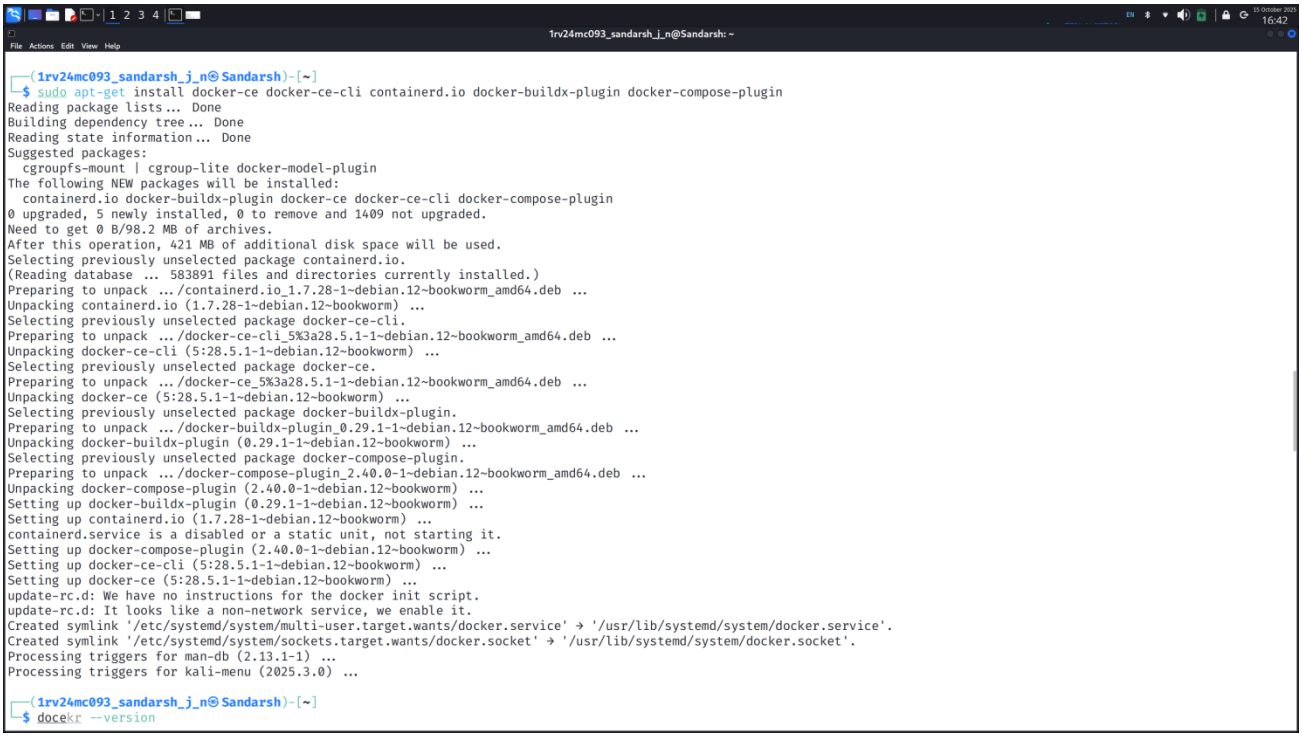
Step 4: Add the Docker Repository

echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] \
https://download.docker.com/linux/debian \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

```
┌──(1rv24mc093_sandarsh_j_n㉿Sandarsh)-[~]
└─$ echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/debian \
  bookworm stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Step 6: Install Docker Engine

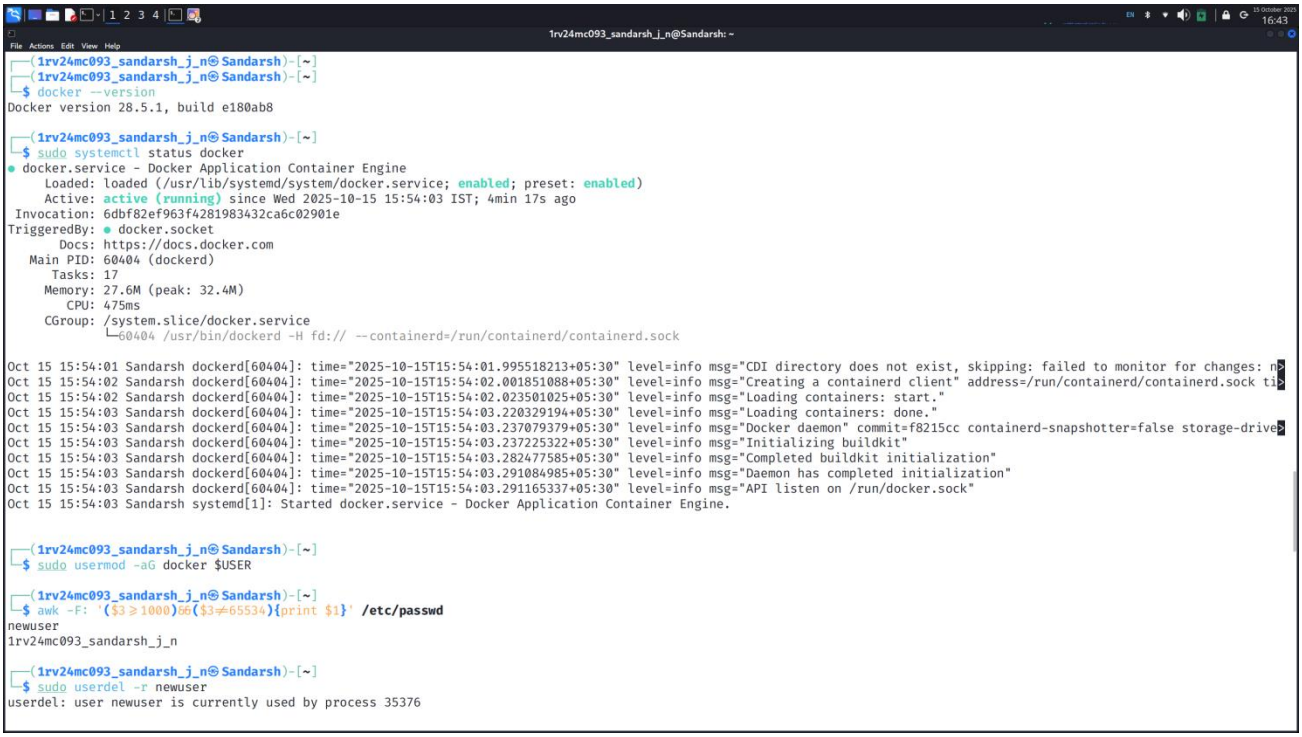
$sudo apt install docker-ce docker-ce-cli containerd.io -y



Step 7: Enable and Start Docker Service

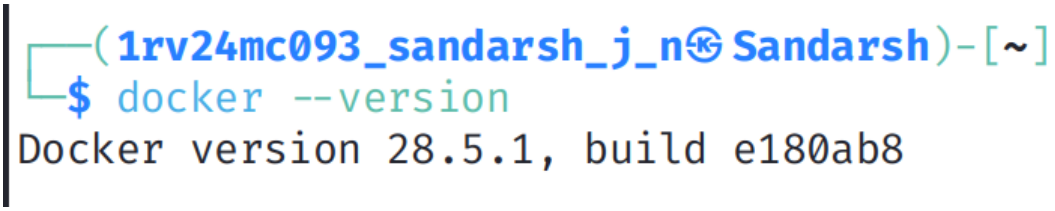
$sudo systemctl enable docker
$sudo systemctl start docker

Check status:
$sudo systemctl status docker



Step 8: Verify Docker Installation

$docker --version



Step 9: Run a Test Docker Container

$sudo docker run hello-world

```
┌──(1rv24mc093_sandarsh_j_n㉿Sandarsh)-[~]
└─$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:6dc565aa630927052111f823c303948cf83670a3903ffa3849f1488ab517f891
Status: Downloaded newer image for hello-world:latest

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
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```