

## Change username on Ubuntu using a temporary user

follow these steps in the terminal:-

The process involves creating a temporary user, logging in with that user, and then renaming your main user.

**Create a temporary user** (since you can't change your username while logged in with it because you have to kill the current working process):

### step 1:-

Open a terminal and run:

**sudo adduser temp**

//This creates a new user//

**sudo usermod -aG temp**

//this makes the temp account an admin

**sudo pkill -u r-mahesh-babu**

//this make sures to stop the user account

### step 2:-

Switch to the temp user account and open the terminal:

**sudo pkill -u r-mahesh-babu**

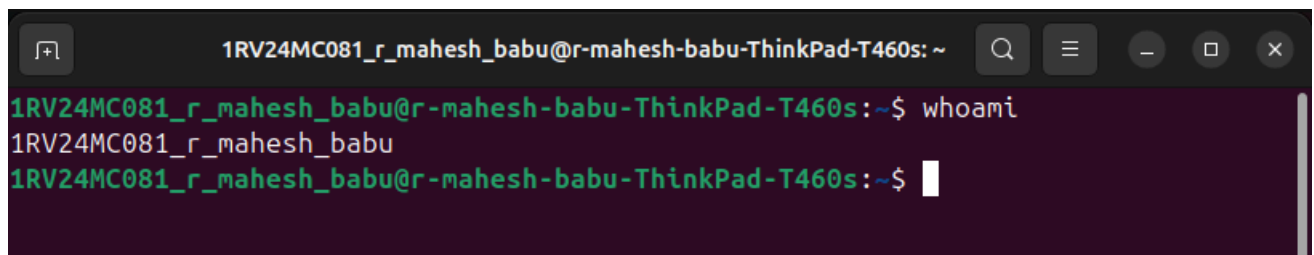
//just to make sure that the process is ended

**sudo usermod -l 1RV24MC081\_r\_mahesh\_babu r-mahesh-babu**

**//newusername**

**//oldusername**

**Whoami** //to display the username changed once you log in to the older user account

A terminal window screenshot with a dark background. The title bar shows the user '1RV24MC081\_r\_mahesh\_babu' and the host 'r-mahesh-babu-ThinkPad-T460s'. The terminal shows three lines of text: the prompt '1RV24MC081\_r\_mahesh\_babu@r-mahesh-babu-ThinkPad-T460s:~\$' followed by the command 'whoami', the output '1RV24MC081\_r\_mahesh\_babu', and the prompt again '1RV24MC081\_r\_mahesh\_babu@r-mahesh-babu-ThinkPad-T460s:~\$' with a cursor.

Now, you should have successfully changed your username!

Now use this command to delete the temp user account:

```
sudo deluser temp
```

## **Now install docker in the linux(ubuntu LTS)**

Open terminal and type the following command:

```
# 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 \
```

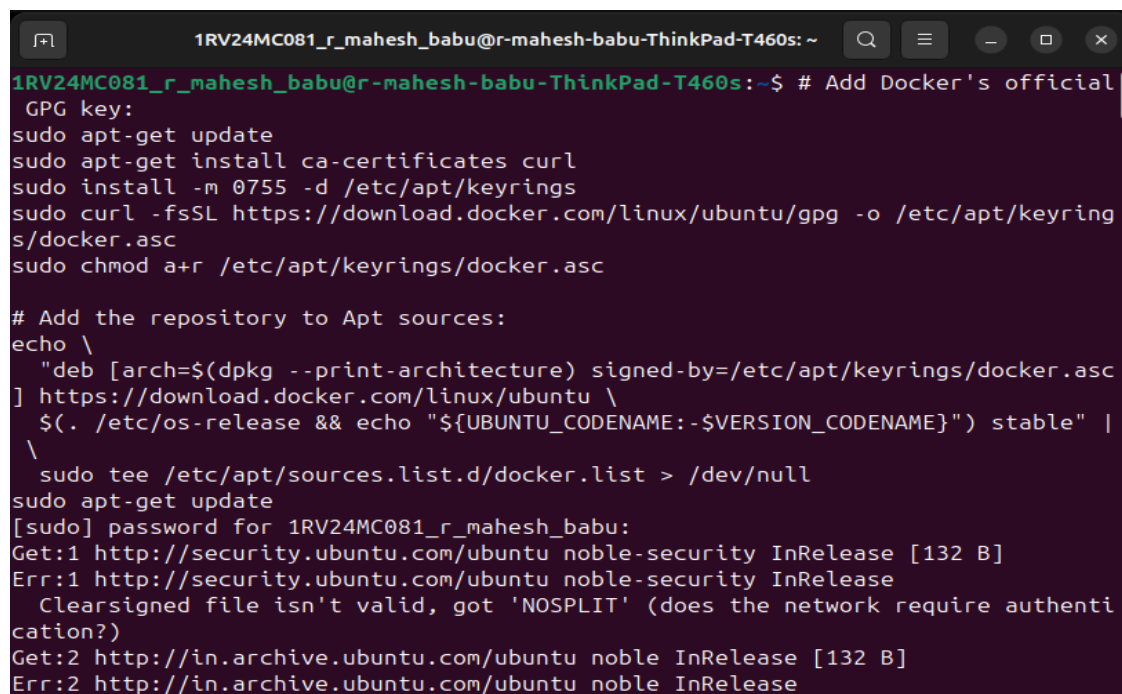
```
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
```

```
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
```

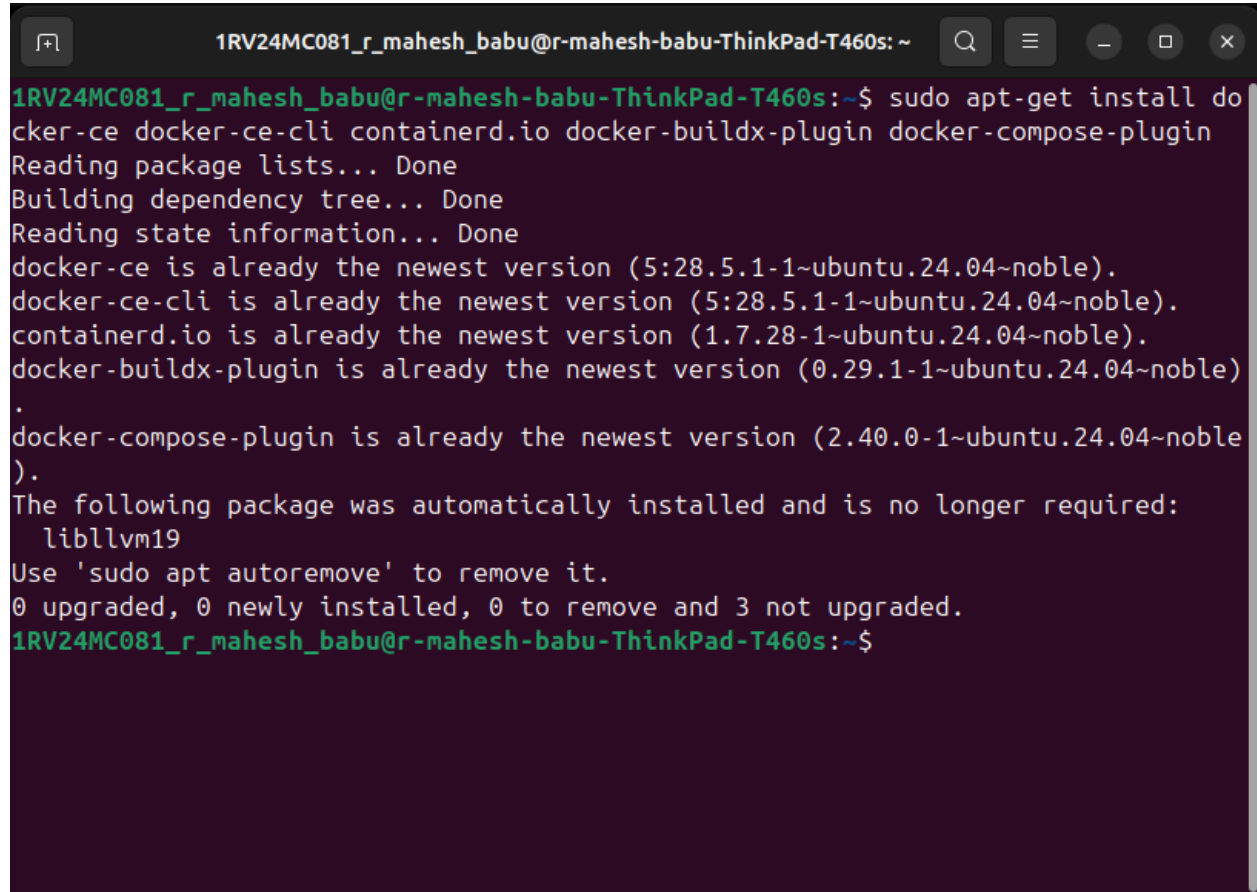
A terminal window screenshot showing the installation of Docker on Ubuntu LTS. The terminal title is '1RV24MC081\_r\_mahesh\_babu@r-mahesh-babu-ThinkPad-T460s: ~'. The user enters the command '# Add Docker's official GPG key:'. The terminal shows the execution of several commands: 'sudo apt-get update', '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', and 'sudo chmod a+r /etc/apt/keyrings/docker.asc'. Then, the user enters '# Add the repository to Apt sources:'. The terminal shows the execution of 'echo \' followed by a multi-line command to add the Docker repository to the sources list. The command is: 'deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \' followed by a line continuation character '\', then '\$(. /etc/os-release && echo "\${UBUNTU\_CODENAME:-\$VERSION\_CODENAME}") stable' followed by another line continuation character '\', and finally 'sudo tee /etc/apt/sources.list.d/docker.list > /dev/null'. The user then enters 'sudo apt-get update'. The terminal shows the output of the update command, including a password prompt for '1RV24MC081\_r\_mahesh\_babu', and two error messages: 'Err:1 http://security.ubuntu.com/ubuntu noble-security InRelease [132 B] Clearsigned file isn't valid, got 'NOSPLIT' (does the network require authentication?)' and 'Err:2 http://in.archive.ubuntu.com/ubuntu noble InRelease [132 B]'. The terminal output is as follows:

```
1RV24MC081_r_mahesh_babu@r-mahesh-babu-ThinkPad-T460s:~$ # 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/keyring
s/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 1RV24MC081_r_mahesh_babu:
Get:1 http://security.ubuntu.com/ubuntu noble-security InRelease [132 B]
Err:1 http://security.ubuntu.com/ubuntu noble-security InRelease
  Clearsigned file isn't valid, got 'NOSPLIT' (does the network require authenti
cation?)
Get:2 http://in.archive.ubuntu.com/ubuntu noble InRelease [132 B]
Err:2 http://in.archive.ubuntu.com/ubuntu noble InRelease
```

To install the latest version, run:

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

A terminal window with a dark background and light green text. The window title bar shows the user '1RV24MC081\_r\_mahesh\_babu' and the machine 'r-mahesh-babu-ThinkPad-T460s'. The terminal displays the output of the command 'sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin'. The output indicates that all requested packages are already at the newest version. It also lists a package 'libllvm19' that was automatically installed and is no longer required, with a suggestion to use 'sudo apt autoremove' to remove it. The summary shows 0 upgrades, 0 new installations, 0 removals, and 3 packages not upgraded.

```
1RV24MC081_r_mahesh_babu@r-mahesh-babu-ThinkPad-T460s: ~  
1RV24MC081_r_mahesh_babu@r-mahesh-babu-ThinkPad-T460s:~$ sudo apt-get install do  
cker-ce docker-ce-cli containerd.io docker-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 package was automatically installed and is no longer required:  
  libllvm19  
Use 'sudo apt autoremove' to remove it.  
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.  
1RV24MC081_r_mahesh_babu@r-mahesh-babu-ThinkPad-T460s:~$
```

## sudo docker run hello-world

```
1RV24MC081_r_mahesh_babu@r-mahesh-babu-ThinkPad-T460s:~$ 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/
```