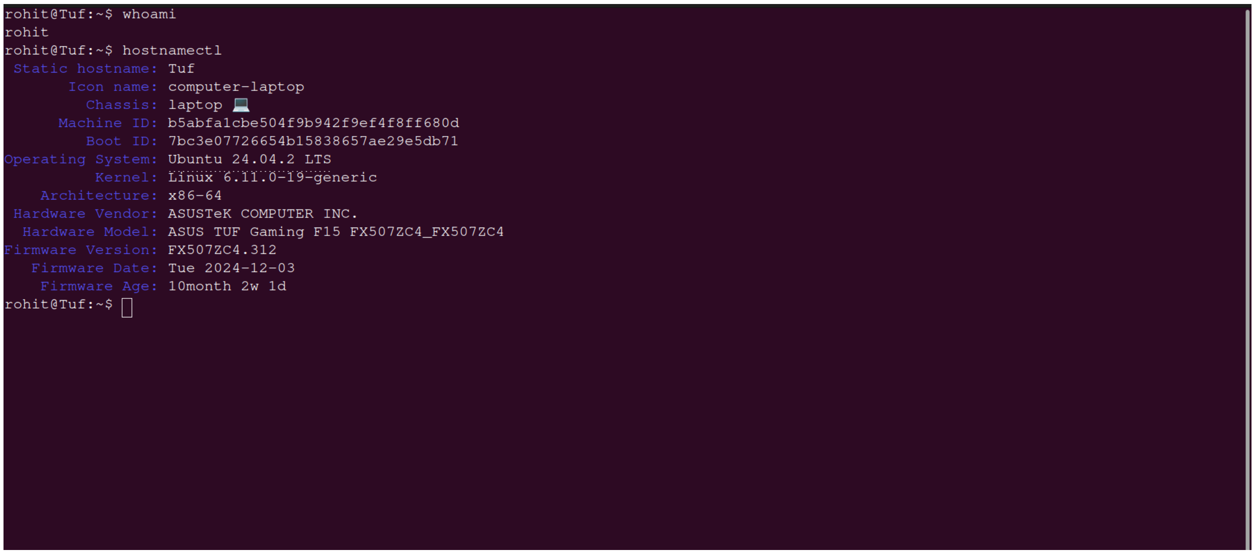
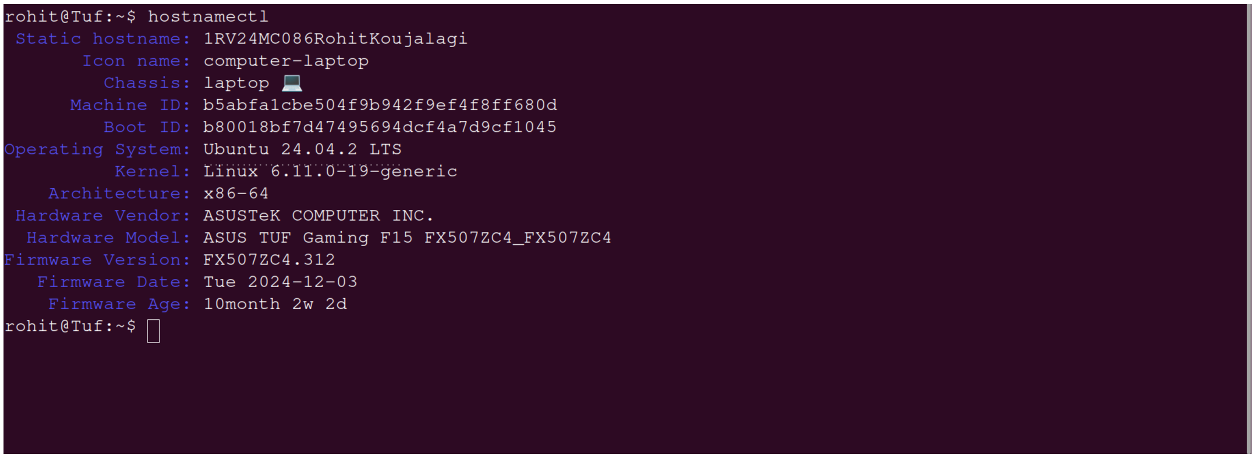
Changing hostname in Ubuntu

To check the current hostname
$ whoami
OR
$ hostnamectl

Commands Used
$ sudo nano /etc/hostname
$ sudo nano /etc/hosts
$ sudo hostname 1RV24MC086RohitKoujalagi



```
rohit@Tuf:~$ whoami
rohit
rohit@Tuf:~$ hostnamectl
 Static hostname: Tuf
       Icon name: computer-laptop
         Chassis: laptop 💻
      Machine ID: b5abfa1cbe504f9b942f9ef4f8ff680d
         Boot ID: 7bc3e07726654b15838657ae29e5db71
Operating System: Ubuntu 24.04.2 LTS
          Kernel: Linux 6.11.0-19-generic
    Architecture: x86-64
 Hardware Vendor: ASUSTeK COMPUTER INC.
  Hardware Model: ASUS TUF Gaming F15 FX507ZC4_FX507ZC4
Firmware Version: FX507ZC4.312
   Firmware Date: Tue 2024-12-03
    Firmware Age: 10month 2w 1d
rohit@Tuf:~$ 
```



```
rohit@Tuf:~$ hostnamectl
 Static hostname: 1RV24MC086RohitKoujalagi
       Icon name: computer-laptop
         Chassis: laptop 💻
      Machine ID: b5abfa1cbe504f9b942f9ef4f8ff680d
         Boot ID: b80018bf7d47495694dcf4a7d9cf1045
Operating System: Ubuntu 24.04.2 LTS
          Kernel: Linux 6.11.0-19-generic
    Architecture: x86-64
 Hardware Vendor: ASUSTeK COMPUTER INC.
  Hardware Model: ASUS TUF Gaming F15 FX507ZC4_FX507ZC4
Firmware Version: FX507ZC4.312
   Firmware Date: Tue 2024-12-03
    Firmware Age: 10month 2w 2d
rohit@Tuf:~$ 
```

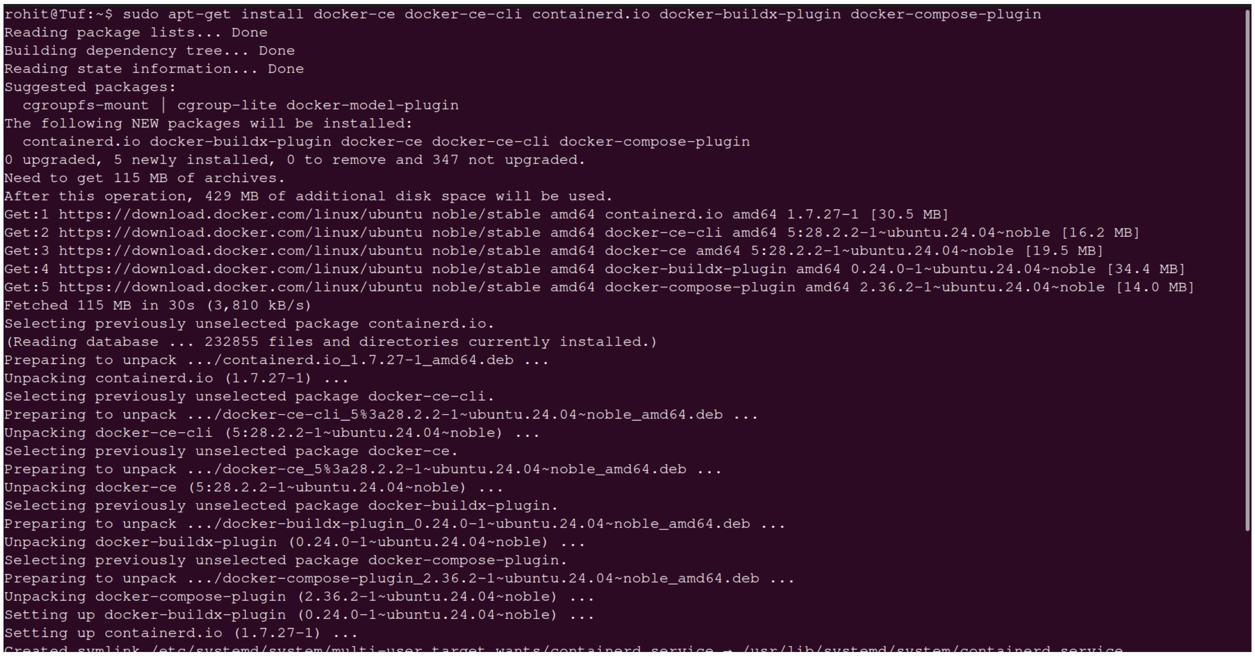
Docker Installation

Part 1: Install Docker via the apt Repository

This method sets up your system to get Docker updates automatically through the standard system update process.

Step 1.1: Install prerequisite packages

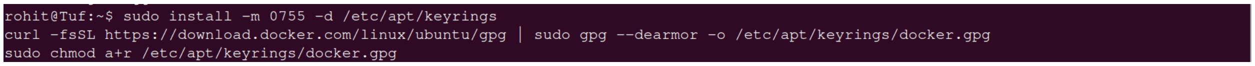
These packages are needed to allow apt to use a repository over HTTPS.
sudo apt-get install ca-certificates curl gnupg



```
rohit@Tuf:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  cgroupfs-mount | cgroup-lite docker-model-plugin
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-compose-plugin
0 upgraded, 5 newly installed, 0 to remove and 347 not upgraded.
Need to get 115 MB of archives.
After this operation, 429 MB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu noble/stable amd64 containerd.io amd64 1.7.27-1 [30.5 MB]
Get:2 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce-cli amd64 5:28.2.2-1~ubuntu.24.04~noble [16.2 MB]
Get:3 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce amd64 5:28.2.2-1~ubuntu.24.04~noble [19.5 MB]
Get:4 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-buildx-plugin amd64 0.24.0-1~ubuntu.24.04~noble [34.4 MB]
Get:5 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-compose-plugin amd64 2.36.2-1~ubuntu.24.04~noble [14.0 MB]
Fetched 115 MB in 30s (3,810 kB/s)
Selecting previously unselected package containerd.io.
(Reading database ... 232855 files and directories currently installed.)
Preparing to unpack .../containerd.io_1.7.27-1_amd64.deb ...
Unpacking containerd.io (1.7.27-1) ...
Selecting previously unselected package docker-ce-cli.
Preparing to unpack .../docker-ce-cli_5%3a28.2.2-1~ubuntu.24.04~noble_amd64.deb ...
Unpacking docker-ce-cli (5:28.2.2-1~ubuntu.24.04~noble) ...
Selecting previously unselected package docker-ce.
Preparing to unpack .../docker-ce_5%3a28.2.2-1~ubuntu.24.04~noble_amd64.deb ...
Unpacking docker-ce (5:28.2.2-1~ubuntu.24.04~noble) ...
Selecting previously unselected package docker-buildx-plugin.
Preparing to unpack .../docker-buildx-plugin_0.24.0-1~ubuntu.24.04~noble_amd64.deb ...
Unpacking docker-buildx-plugin (0.24.0-1~ubuntu.24.04~noble) ...
Selecting previously unselected package docker-compose-plugin.
Preparing to unpack .../docker-compose-plugin_2.36.2-1~ubuntu.24.04~noble_amd64.deb ...
Unpacking docker-compose-plugin (2.36.2-1~ubuntu.24.04~noble) ...
Setting up docker-buildx-plugin (0.24.0-1~ubuntu.24.04~noble) ...
Setting up containerd.io (1.7.27-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service
```

Step 1.2: Add Docker's official GPG key

This adds the official key to verify the authenticity of the Docker packages.
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL
https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg

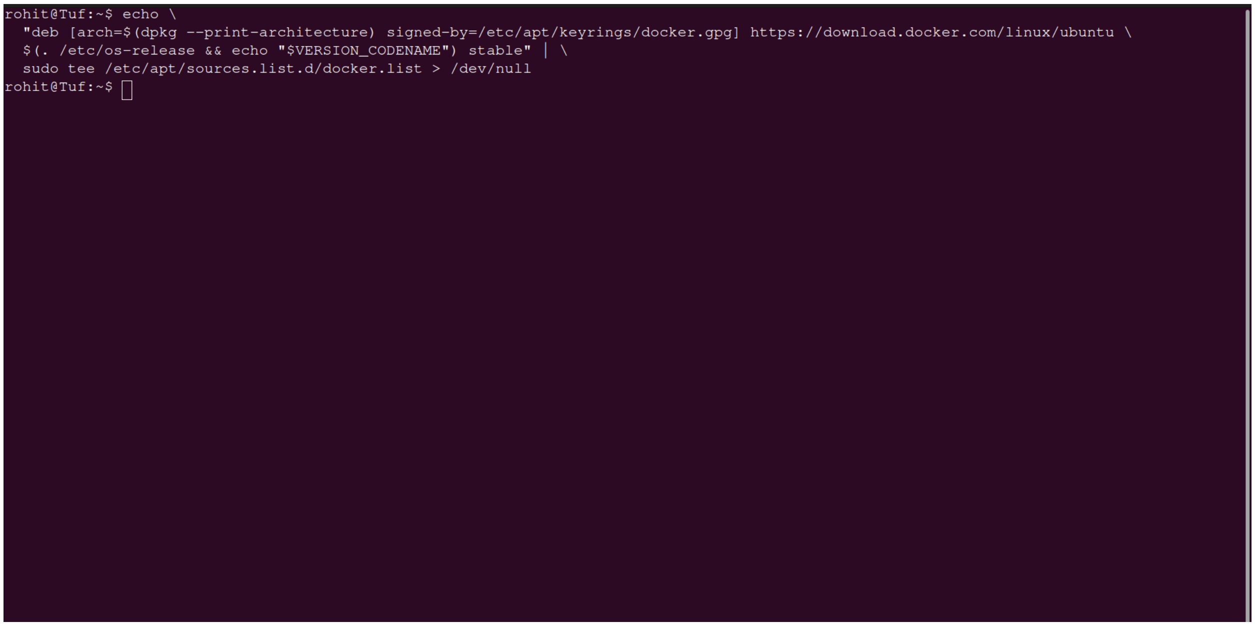


```
rohit@Tuf:~$ sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

Step 1.3: Set up the Docker repository

This command adds the official Docker software repository to your system's sources.

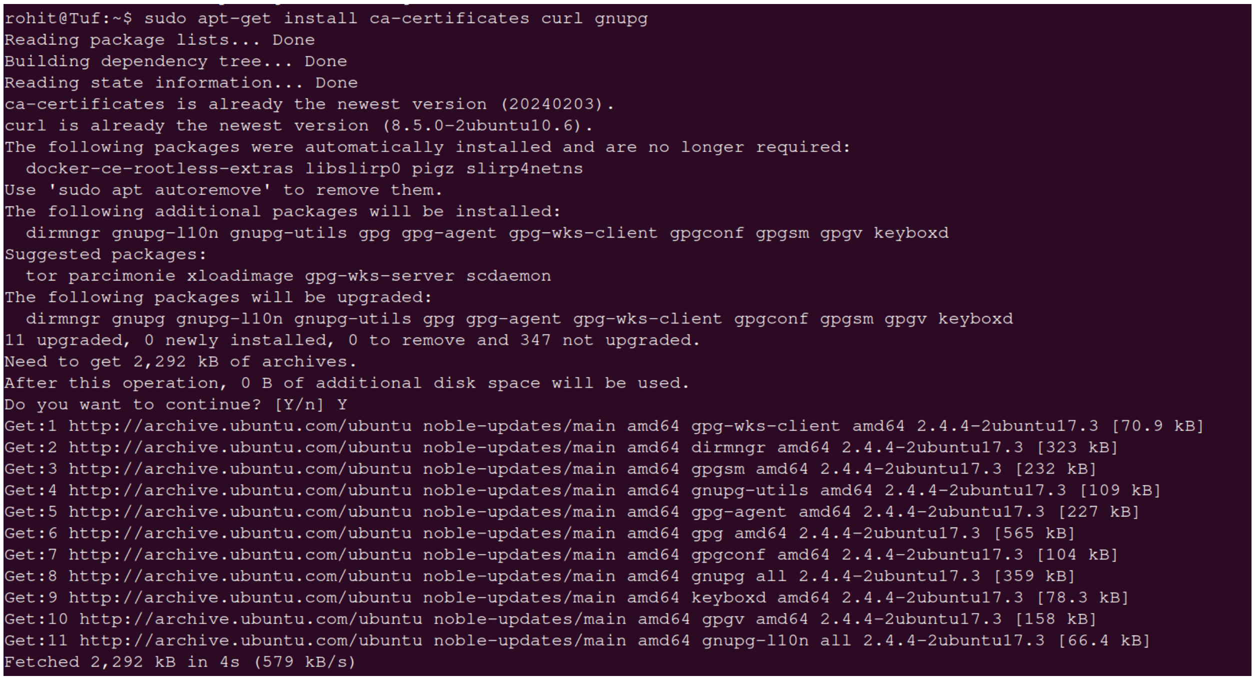
```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
[https://download.docker.com/linux/ubuntu](https://download.docker.com/linux/ubuntu) \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```



```
rohit@Tuf:~$ echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
rohit@Tuf:~$ 
```

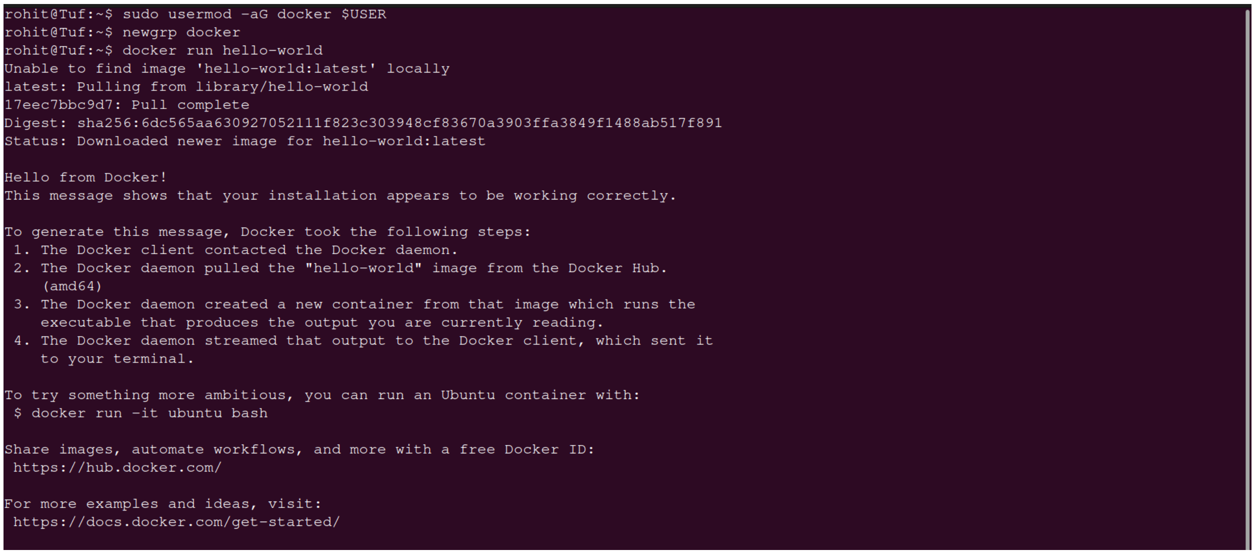
Step 1.4: Install Docker Engine

Update your package list again, then install the latest stable release of Docker.
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin



```
rohit@Tuf:~$ sudo apt-get install ca-certificates curl gnupg
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
curl is already the newest version (8.5.0-2ubuntu10.6).
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.
The following additional packages will be installed:
  dirmngr gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client gpgconf gpgsm gpgv keyboxd
Suggested packages:
  tor parcimonie xloadimage gpg-wks-server scdaemon
The following packages will be upgraded:
  dirmngr gnupg gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client gpgconf gpgsm gpgv keyboxd
11 upgraded, 0 newly installed, 0 to remove and 347 not upgraded.
Need to get 2,292 kB of archives.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpg-wks-client amd64 2.4.4-2ubuntu17.3 [70.9 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 dirmngr amd64 2.4.4-2ubuntu17.3 [323 kB]
Get:3 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpgsm amd64 2.4.4-2ubuntu17.3 [232 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gnupg-utils amd64 2.4.4-2ubuntu17.3 [109 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpg-agent amd64 2.4.4-2ubuntu17.3 [227 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpg amd64 2.4.4-2ubuntu17.3 [565 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpgconf amd64 2.4.4-2ubuntu17.3 [104 kB]
Get:8 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gnupg all 2.4.4-2ubuntu17.3 [359 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 keyboxd amd64 2.4.4-2ubuntu17.3 [78.3 kB]
Get:10 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gpgv amd64 2.4.4-2ubuntu17.3 [158 kB]
Get:11 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 gnupg-l10n all 2.4.4-2ubuntu17.3 [66.4 kB]
Fetched 2,292 kB in 4s (579 kB/s)
```

Part 2: Post-Installation Step



```
rohit@Tuf:~$ sudo usermod -aG docker $USER
rohit@Tuf:~$ newgrp docker
rohit@Tuf:~$ 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/
```