

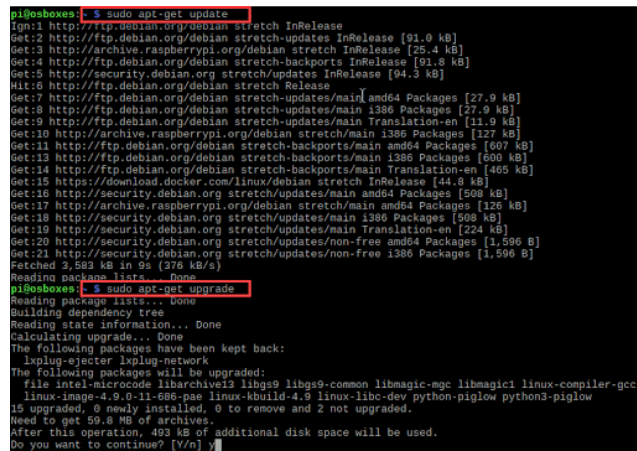
Steps to install Docker and ThingsBoard on Raspberry PI 4

Step 1:

Upgrade and Update

In this step, keep upgrading and updating your system and install the most advanced software version.

Run this command "**sudo apt-get update && sudo apt-get upgrade**" as shown in the below screenshot.



```
pi@osboxes:~$ sudo apt-get update
Ign:1 http://ftp.debian.org/debian stretch InRelease
Get:2 http://ftp.debian.org/debian stretch-updates InRelease [91.9 kB]
Get:3 http://archive.raspberrypi.org/debian stretch InRelease [25.4 kB]
Get:4 http://ftp.debian.org/debian stretch-backports InRelease [91.8 kB]
Get:5 http://security.debian.org stretch/updates InRelease [94.3 kB]
Hit:6 http://ftp.debian.org/debian stretch Release
Get:7 http://ftp.debian.org/debian stretch-updates/main amd64 Packages [27.9 kB]
Get:8 http://ftp.debian.org/debian stretch-updates/main i386 Packages [27.9 kB]
Get:9 http://ftp.debian.org/debian stretch-updates/main Translation-en [11.9 kB]
Get:10 http://archive.raspberrypi.org/debian stretch/main i386 Packages [127 kB]
Get:11 http://ftp.debian.org/debian stretch-backports/main amd64 Packages [607 kB]
Get:12 http://ftp.debian.org/debian stretch-backports/main i386 Packages [600 kB]
Get:13 http://ftp.debian.org/debian stretch-backports/main Translation-en [465 kB]
Get:14 http://download.docker.com/linux/debian stretch InRelease [44.9 kB]
Get:15 http://security.debian.org stretch/updates/main amd64 Packages [508 kB]
Get:16 http://security.debian.org stretch/updates/main i386 Packages [508 kB]
Get:17 http://archive.raspberrypi.org/debian stretch/main amd64 Packages [126 kB]
Get:18 http://security.debian.org stretch/updates/main i386 Packages [508 kB]
Get:19 http://security.debian.org stretch/updates/main Translation-en [224 kB]
Get:20 http://security.debian.org stretch/updates/non-free amd64 Packages [1.596 B]
Get:21 http://security.debian.org stretch/updates/non-free i386 Packages [1.596 B]
Fetched 3,583 kB in 9s (376 kB/s)
Reading package lists... Done
pi@osboxes:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages have been kept back:
lxdm, lxdm-network
The following packages will be upgraded:
lxdm, lxdm-network, lxdm-image
15 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
Need to get 59.3 kB of archives.
After this operation, 493 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Step 2:

Download the right script and install Docker on the Raspberry Pi environment.

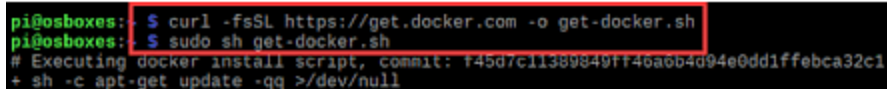
Push this below installation script.

```
curl -fsSL https://get.docker.com -o get-docker.sh
```

Run the script with the help of the below command:

```
sudo sh get-docker.sh
```

This establishes the expected packages on the Raspbian Linux administration environment.



```
pi@osboxes:~$ curl -fsSL https://get.docker.com -o get-docker.sh
pi@osboxes:~$ sudo sh get-docker.sh
# Executing docker install script, commit: f45d7c11389849f146a0b4d94e0dd1ffebe3a32c1
+ sh -c apt-get update -qq >/dev/null
```

Here is the running output of the Docker version as per the above command.

```
Server: Docker Engine - Community
Engine:
  Version:          19.03.5
  API version:      1.40 (minimum version 1.12)
  Go version:       go1.12.12
  Git commit:       633a0ea838
  Built:            Wed Nov 13 07:28:22 2019
  OS/Arch:          linux/amd64
  Experimental:     false
containerd:
  Version:          1.2.10
  GitCommit:        b34a5c8af56e510852c35414db4c1f4fa6172339
runc:
  Version:          1.0.0-rc8+dev
  GitCommit:        3e425f80a8c931f88e6d94a8c831b9d5aa481657
docker-init:
  Version:          0.18.0
  GitCommit:        fec3683
If you would like to use Docker as a non-root user, you should now consider
adding your user to the "docker" group with something like:

    sudo usermod -aG docker your-user

Remember that you will have to log out and back in for this to take effect!

WARNING: Adding a user to the "docker" group will grant the ability to run
containers which can be used to obtain root privileges on the
docker host.
Refer to https://docs.docker.com/engine/security/security/#docker-daemon-attac
k-surface
for more information.
```

Step 3:

Append a non-root user on the Docker group

As per the Raspberry Pi process, whichever user has the administrative rights whom we can consider as root user can execute containers. For example, if a case user is not logged in to the admin root, then they should use the sudo prefix.

We can also add the non-root users to the Docker group, which will enable running the executed docker commands.

Here is the syntax for adding users to the Docker group:

```
sudo usermod -aG docker [user_name]
```

We can consider it the default user for adding the Pi user in Raspbian.

Refer to the below command:

```
sudo usermod -aG docker Pi
```

We need to log out of the system to check the output process.

Step 4:

For checking the Updated Docker Version and info details on your Raspberry Pi, refer to the below command.

docker version

The below output screenshot will show the docker version with all the relevant details.

For getting the system-wide details, which involve the list of containers, kernel version, and docker images, run this below command.

docker info

Step 5:

Execute/run the sample Hello World Container. Here we need to configure and set up this below command.

docker run hello-world

After running the above command, it will contact the Docker daemon and pull the “hello-world” image.

The output will show the “installation appears to be working correctly” message.

```
pi@osboxes:~$ 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.
 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/
```

Docker Raspberry Pi 4

Raspberry Pi 4 can work as a low-cost Docker resolution for application development and various responsibilities. It has the latest 8GB version.

A Raspberry Pi 4 is a single-board processor. To connect Docker on your Raspberry Pi 4, we need the following:

A Raspberry Pi 4 Type-C power accumulation.

A microSD card has a minimum of 32GB storage space, including the Raspberry Pi OS image on it.

Internet connectivity using the Raspberry Pi 4.

A laptop or computer for VNC remote desktop accessibility using SSH way to the Raspberry Pi 4.

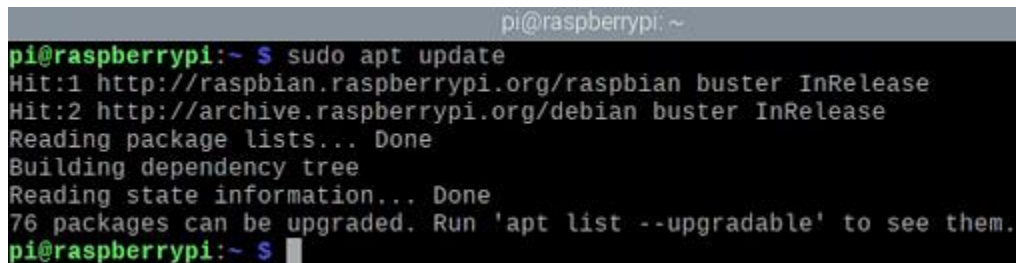
Installing Docker Raspberry Pi 4

Below is the command for updating Raspberry Pi OS.

\$ sudo apt update

A terminal window with a dark background. The prompt 'pi@raspberrypi:~' is shown in green, followed by the command '\$ sudo apt update' in blue.

After running the command, it will show the below output.

A terminal window showing the output of the 'sudo apt update' command. The output includes: 'Hit:1 http://raspbian.raspberrypi.org/raspbian buster InRelease', 'Hit:2 http://archive.raspberrypi.org/debian buster InRelease', 'Reading package lists... Done', 'Building dependency tree', 'Reading state information... Done', and '76 packages can be upgraded. Run \'apt list --upgradable\' to see them.' The prompt 'pi@raspberrypi:~ \$' is visible at the bottom.

Run the below command to update all the Raspberry Pi OS environment packages.

```
$ sudo apt upgrade
```

```

pi@raspberrypi:~ $ sudo apt upgrade
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  agnostics alacarte base-files bind9-host bluez-firmware libbind9-161
  libdns-export1104 libdns1104 libdm-data libdm-extra4 libdm-gtk-data
  libdm-gtk4 libdm-modules libdm4 libgs9 libgs9-common libgssdp-1.0-3
  libgupnp-1.0-4 libisc-export1100 libisc1100 libisc161 libiscscfp163
  liblwres161 libqt5concurrent5 libqt5core5a libqt5dbus5 libqt5gui5
  libqt5network5 libqt5sprintsupport5 libqt5sql5 libqt5sql5-sqlite
  libqt5widgets5 libqt5xml5 libraspberrypi-bin libraspberrypi-dev
  libraspberrypi-doc libraspberrypi10 libvlc-bin libvlc5 libvlccore9 libx11-6
  libx11-data libx11-xcb libx11-xcb1 lxpanel lxpanel-data lxplug-bluetooth
  lxplug-cputemp lxplug-ejector lxplug-network lxplug-ptbatt pcmanfm pipanel
  rprompt qt5-gtk-platformtheme raspberrypi-bootloader raspberrypi-kernel
  rspi-config rc-gui rp-prefapps rpi-chromium-mods rpi-eeeprom vlc vlc-bin
  vlc-data vlc-110n vlc-plugin-base vlc-plugin-notify vlc-plugin-qt
  vlc-plugin-samba vlc-plugin-skins2 vlc-plugin-video-output
  vlc-plugin-video-splitter vlc-plugin-visualization xserver-common
  xserver-xorg-core
76 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 161 MB of archives.
After this operation, 1,795 kB of additional disk space will be used.
Do you want to continue? [Y/n]

```

Raspberry Pi 4 Docker

Raspberry Pi 4 can be applied to build a private cloud resolution with Ansible and Docker, which are great tools used by various large-scale cloud systems that automate the tasks, configure them, and enable containerization for the applications.

After booting your system, the following commands can implement the most advanced Rasbian updates.

```
$ sudo apt updat
```

```
$ sudo apt dist-upgrade
```

To define the hostname, we need to add two files as per the below command.

```
$ sudo nano /etc/hostname
```

```
$ sudo nano /etc/hosts
```

ThingsBoard Installation

Before starting Docker container run following commands to create a directory for storing data and logs and then change its owner to docker container user, to be able to change user, chown command is used, which requires sudo permissions (command will request password for a sudo access):

```
$ mkdir -p ~/.mytb-data && sudo chown -R 799:799 ~/.mytb-data
```

```
$ mkdir -p ~/.mytb-logs && sudo chown -R 799:799 ~/.mytb-logs
```

Execute the following command to run this docker directly:

```
$ docker run -it -p 9090:9090 -p 1883:1883 -p 7070:7070 -p 5683-5688:5683-5688/udp -v  
~/.mytb-data:/data -v ~/.mytb-logs:/var/log/thingsboard --name mytb --restart always  
thingsboard/tb-postgres
```

Where:

docker run - run this container

-it - attach a terminal session with current ThingsBoard process output

-p 9090:9090 - connect local port 9090 to exposed internal HTTP port 9090

-p 1883:1883 - connect local port 1883 to exposed internal MQTT port 1883

-p 7070:7070 - connect local port 7070 to exposed internal Edge RPC port 7070

-p 5683-5688:5683-5688/udp - connect local UDP ports 5683-5688 to exposed internal COAP and LwM2M ports

-v ~/.mytb-data:/data - mounts the host's dir ~/.mytb-data to ThingsBoard DataBase data directory

-v ~/.mytb-logs:/var/log/thingsboard - mounts the host's dir ~/.mytb-data to ThingsBoard logs directory

--name mytb - friendly local name of this machine

--restart always - automatically start ThingsBoard in case of system reboot and restart in case of failure.

thingsboard/tb-postgres - docker image

After executing this command, you can open **http://{your-host-ip}:9090** in your browser. You should see ThingsBoard login page. Use the following default credentials:

System Administrator: sysadmin@thingsboard.org / sysadmin

Tenant Administrator: tenant@thingsboard.org / tenant

Customer User: customer@thingsboard.org / customer

You can always change passwords for each account in account profile page.

You can detach from session terminal with Ctrl-p Ctrl-q - the container will keep running in the background.

To reattach to the terminal (to see ThingsBoard logs) run:

```
$ docker attach mytb
```

To stop the container:

```
$ docker stop mytb
```

To start the container:

```
$ docker start mytb
```