Docker Developer Guide

ID: RK-KF-YF-334

Release Version: V1.2.1

Release Date: 2022-04-13

Security Level: □Top-Secret □Secret □Internal ■Public

Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian,PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document introduces how to build a Docker environment, compile libmali, gstreamer, mpp, xserver, libdrm and other third repositories into deb packages on a PC with Ubuntu system, and then install them into the Debian system.

Packages and System Support

Packages	Version	OS
glmark2	2021.02	Debian
gst-plugins-base	1.14.4/1.18.5	Debian
gstreamer-rockchip	1.14.4	Debian
libdrm	2.4.97/2.4.104	Debian
libdrm-cursor	1.4.0	Debian
libmali	1.9.0	Debian
mpp	1.5.0	Debian
libv4l-rkmpp	1.5.0	Debian
openbox	3.6.1	Debian
pcmanfm	1.2.5	Debian
rga	2.1.0/2.2.0	Debian
rkisp	2.2.0	Debian
rkaiq	5.0	Debian
xserver	1.20.4/1.20.11	Debian
wifibt	1.0.0	Debian
rktoolkit	1.0.0	Debian

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers

Software development engineers

Revision History

Date	Version	Author	Change Description
2019-08-27	V1.0.0	Caesar Wang	Initial version
2020-02-18	V1.0.1	Caesar Wang	Sync the style with release
2021-03-15	V1.0.2	Ruby Zhang	Update product version information
2021-07-19	V1.1.0	Caesar Wang	Update docker for Debian10
2022-09-20	V1.2.0	Caesar Wang	Update docker for Debian11
2023-04-13	V1.2.1	Caesar Wang	Update some repositories for Debian

Contents

1. Rockchip Docker

Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.

In a way, Docker is a bit like a virtual machine. But unlike a virtual machine, rather than creating a whole virtual operating system, Docker allows applications to use the same Linux kernel as the system that they're running on and only requires applications be shipped with things not already running on the host computer. This gives a significant performance boost and reduces the size of the application.

And importantly, Docker is open source. This means that anyone can contribute to Docker and extend it to meet their own needs if they need additional features that aren't available out of the box.

Please refer to the following website for Rockchip Docker:
docker-rockchip.

2. OS Requirements

To install Docker, you need one of the following 64-bit Ubuntu versions:

- Jammy 22.04 (LTS)
- Focal 20.04 (LTS)
- Bionic 18.04 (LTS)
- Xenial 16.04 (LTS)
- Trusty 14.04 (LTS)

Note: the dockerfile is used for arm64 Socs by default.

The following is used for arm32 Socs: cp dockerfile-32 dockerfile

2.1 Install Docker

- Use this command to install the latest version of Docker(replace docker with docker.io in ubuntu 14.04):

```
sudo apt-get install docker qemu-user-static binfmt-support
```

- Start and run Docker deamon:

```
sudo service docker start
```

- Build Docker image by dockerfile:

```
sudo docker build -t rockchip .
```

Now you get a Docker image named "rockchip" which include a Debian multi-arch cross-compiling environment.

2.2 Build application

- Enter docker shell:

```
sudo docker run -it -v <package dir>:/home/rk/packages rockchip /bin/bash
```

- Start build:

For arm 32-bit Socs：

```
cd /home/rk/packages/<package-name>
DEB_BUILD_OPTIONS=nocheck dpkg-buildpackage -rfakeroot -b -d -uc -us -aarmhf
ls ../ | grep *.deb
```

For arm 64-bit Socs：

```
cd /home/rk/packages/<package-name>
DEB_BUILD_OPTIONS=nocheck dpkg-buildpackage -rfakeroot -b -d -uc -us -aarm64
ls ../ | grep *.deb
```

2.3 Modify Image

If you want to modify your Docker image, please open a shell by below command:

```
sudo docker run -it rockchip /bin/bash
```

After exit from the container, you should use below command to save your changes.

```
sudo docker commit  <container_id> rockchip
```

3. Others

To get more informations about dockers, please check below link: https://docs.docker.com

4. Examples

- How to generate the libmali-bifrost-g52-g2p0-x11_1.9-1_arm64.deb on libmali

```
~/work/docker/docker-rockchip$sudo service docker start
~/work/docker/docker-rockchip$sudo docker build -t rockchip .
~/work/docker/docker-rockchip$sudo docker run -it -v
/home/wxt/work:/home/rk/packages rockchip /bin/bash
rk@2888134f9c12:/$ cd /home/rk/packages/docker/libmali
rk@2888134f9c12:~/packages/docker/libmali$ DEB_BUILD_OPTIONS=nocheck dpkg-
buildpackage -rfakeroot -b -d -uc -us -aarm64
```

The above steps will get the debs for ~/packages/docker/

- Rockchip had uploaded some source code for building and generating the deb packages

glmark2
libmali
mpp
rga
rkwifbt
gstreamer-rockchip