

- ①ファイル共有, ②リモート接続,
③x64 コンテナを動かす

ご購入はこちら

ラズパイで試す! コンテナ環境の構築&実行

土屋 健

リスト1 インストール実行結果

```
tsuyo@rp4:~/CQ/Docker $ sh get-docker.sh
# Executing docker install script, commit:
e5543d473431b782227f8908005543bb4389b8de
省略
Client: Docker Engine - Community
Version: 26.0.0
省略
Server: Docker Engine - Community
Engine:
Version: 26.0.0
省略
tsuyo@rp4:~/CQ/Docker $
```

この章ではラズベリー・パイにDocker環境を構築し、コンテナを使ってみます。ここでは、次の動作を確認してみます。

- コンテナを起動してホスト環境とは分離された環境であることを確認
- ホストとコンテナでファイルを共有する
- 外からコンテナ内のサービスにアクセスしてみる
- ホスト環境 (arm64のラズパイ) とは異なるアーキテクチャ (amd64) のプログラムを動作させてみる

ラズベリー・パイにDocker環境を構築

まずは、ラズベリー・パイ4にDockerをインストールしてコンテナを利用できるようにします。ここでは、Docker公式が提供しているインストール・スクリプトを使ってセットアップします。

次の手順でコマンドを実行します。

```
$ mkdir -p $HOME/CQ/Docker
$ cd $HOME/CQ/Docker
$ curl -fsSL https://get.docker.com -o get-docker.sh
$ sh get-docker.sh
$ sudo apt-get install -y uidmap
```

詳細については、下記URLにある公式ドキュメントやプログラムが参考になります。

<https://docs.docker.com/engine/install/raspberry-pi-os/>

リスト2 dockerコマンドがエラーとなる

```
tsuyo@rp4:~/CQ/Docker $ docker ps
permission denied while trying to connect to the
```

<https://github.com/docker/docker-install>

リスト1のようにエラーなくコマンドが終了すればインストールは完了です。

では、Dockerの動作を確認してみます。しかしリスト2のように、権限がなくdockerにアクセスできません。そこで、dockerシステムにアクセスできるように、ユーザをdockerグループに追加します。ユーザ名はご自分のものに読み替えてください。

```
$ sudo usermod -aG docker tsuyo
```

設定を反映させるためにログインし直し、ユーザの状態を確認します。idコマンドで所属グループを確認し、991(docker)のようにdockerグループが表示されれば問題ありません。

```
tsuyo@rp4:~ $ id
uid=1000(tsuyo) gid=1000(tsuyo)
省略
991(docker), 993(gpio), 994(i2c),
995(spi)
```

再度、dockerコマンドを実行すると今度はエラーとならずに次のように表示されます。

```
tsuyo@rp4:~ $ docker ps
CONTAINER ID IMAGE COMMAND
CREATED STATUS PORTS NAMES
```

これでDockerのインストールが完了し、コンテナを使えるようになりました。

コンテナ環境の起動

今回は、Ubuntuのコンテナ・イメージを使用します。コンテナ・イメージの詳細は、次を参照してください。