



Raspberry pi with Docker Swarm 跨網域溝通使用 network

29 DECEMBER 2015 on Raspberry pi, docker, swarm

之前在 [Container Summit 2015: 以 Docker Swarm 打造多主機叢集環境](#) 分享過在一般運用情狀下使用 docker swarm，這次要介紹在 Raspberry pi 如何進行。



© The logos are trade marks of the Raspberry Pi Foundation or Docker, Inc.

首先，在 Raspberry pi 跟在一般 PC 的環境上有所不同，主要還是 ARM 架構，所以需要使用相關的 OS 才能驅動 Docker，當然若已經有人包好就不用在自造輪子，下面將先介紹如何在 Raspberry pi 將 docker 環境安裝完成

hypriot

hypriot 是個已經打包好 docker 的 image

<http://blog.hypriot.com/>

flash

用來安裝 hypriot 所使用的 command line tool

<https://github.com/hypriot/flash>

Hypriot Image Download

下載相關 image 可以在下面連結取得

<http://blog.hypriot.com/downloads/>

一旦所有需要的東西備齊後，就可以開始進行 image 的安裝，首先我們必須要先設定好 wifi 連接方式，若本身 Raspberry pi 已有皆有線網路就不需要，設定內容如下。

occidentalis.txt

```
hostname=hypriot-pi  
wifi_ssid=my-wifi  
wifi_password=12345678
```

準備好設定檔之後，就可以透過 flash 來進行安裝

flash install

若不使用設定檔的話，可以使用下面指令：

```
flash -n hypriot-pi \  
-s TrunkStudio-Internal \  
-p 22019020 \  
hypriot-rpi-20151115-132854.img /dev/disk2s1
```

使用設定檔的情形，指令如下：

```
flash --config occidentalism.txt hypriot-rpi-20151115-132854.img /dev/disk2s1
```

如此安裝完成後就有 docker 可以進行操作

edit wifi

安裝完成後，若要更新 wifi 可以 ssh 到 pi 上面進行檔案修正

ssh into pi: `vim /boot/occidentalism.txt`

一旦更新完後再將 pi reboot 新的設定檔就會生效

到這邊算是把 pi 的 docker 環境準備好了

Hypriot Docker Images

在這裡可以找到使用 ARM 建置完成的 docker images

<https://hub.docker.com/u/hypriot/>

大部分需要的都可以在這邊取得。

再來，我們為了方便進行 docker 操作可以在安裝下列相關工具，分別是 docker-compose 以及 docker-machine

Docker Compose install

<https://github.com/hypriot/arm-compose>

install docker-machine for pi

```
curl -o docker-machine  
http://downloads.hypriot.com/docker-machine_darwin-  
amd64_0.4.1  
  
chmod +x ./docker-machine  
  
alias docker-machine-pi='./docker-machine'
```

有了上述工具的安裝後，我們就可以開始重現 Raspberry pi with Docker Swarm 跨網域溝通使用 network 的建置過程。

建立 consul

```
docker-machine-pi create -d hypriot --hypriot-ip-  
address=192.168.168.120 swl-consul
```

```
docker $(docker-machine-pi config swl-consul) run -d --  
restart=always -p 8500:8500 -h consul nimblestratus/rpi-  
consul -server -bootstrap
```

建立 swarm master

```
docker-machine-pi create -d hypriot \  
--hypriot-ip-address=192.168.168.38 \  
--swarm --swarm-master \  
--swarm-discovery="consul://$(docker-machine-pi ip  
swl-consul):8500" \  
--engine-opt="cluster-store=consul://$(docker-  
machine-pi ip swl-consul):8500" \  
--engine-opt="cluster-advertise=eth0:2376" swl-
```

```
master
```

查看一下目前的 docker info

```
eval $(docker-machine-pi env --swarm swl-master)
docker info
```

```
➔ ~ docker info
Containers: 4
Images: 5
Role: primary
Strategy: spread
Filters: health, port, dependency, affinity, constraint
Nodes: 1
swl-master: 192.168.168.120:2376
  Status: Healthy
  Containers: 4
  Reserved CPUs: 0 / 4
  Reserved Memory: 0 B / 971.8 MiB
  Labels: executiondriver=native-0.2, kernelversion=4.1.12-hypriot-v7+, operatingsystem=Raspbian GNU/Linux 8 (jessie), provider=hypriot, storagedriver=overlay
CPUs: 4
Total Memory: 971.8 MiB
Name: b80d96394545
```

手上 Raspberry pi 有限，所以 master 也是 node 節點，不過可以看到，swarm 叢集已正確建立完成。

建立 overlay network

```
eval $(docker-machine-pi env --swarm swl-master)
docker network create -d overlay pi-net
```

Docker run on Raspberry pi server web

```
docker run -itd --name=web --net=pi-net --
env="constraint:node==swl-master" hypriot/rpi-nano-
httpd
```

client

```
docker run -it --rm --net=pi-net --  
env="constraint:node=swl-master" hypriot/armhf-  
busybox wget -O- http://web
```

運行結果

```
➔ ~ docker info  
Containers: 4  
Images: 4  
Role: primary  
Strategy: spread  
Filters: health, port, dependency, affinity, constraint  
Nodes: 1  
swl-master: 192.168.168.38:2376  
  Status: Healthy  
  Containers: 4  
  Reserved CPUs: 0 / 4  
  Reserved Memory: 0 B / 971.8 MiB  
  Labels: executiondriver=native-0.2, kernelversion=4.1.12-hypriotos-v7+, operatingssystem=Raspbian GNU/Linux 8 (jessie), provider=hypriot, storagedriver=overlay  
CPUs: 4  
Total Memory: 971.8 MiB  
Name: de879320ef7a  
➔ ~ docker ps  
CONTAINER ID        IMAGE                COMMAND              CREATED             STATUS              PORTS              NAMES  
2d4020e85b5e        hypriot/rpi-nano-httpd  "/httpd 80"         About a minute ago  Up About a minute  80/tcp             swl-master/web  
00ef0fc2e431        hypriot/rpi-swarm:latest  "/swarm join --advert"  6 minutes ago      Up 6 minutes       2375/tcp           swl-master/swarm-agent  
de879320ef7a        hypriot/rpi-swarm:latest  "/swarm manage --tls"  6 minutes ago      Up 6 minutes       2375/tcp, 192.168.168.38:3376->3376/tcp  swl-master/swarm-agent-master
```

可以看到運行結果雖然都是在 swl-master 上面，但基本上已經是
在 swarm 叢集下進行。

結論

Raspberry pi 一直以來就很想嘗試看看，但礙於總覺得使用起來很
麻煩，一直遲遲沒有嚐試，有了 docker 之後，可以在本機測試之
後放到 pi 上面運行。

不過對於 Raspberry pi 的運行，也算是第一次嘗試，應該需要再
好好熟悉了。

若對 Raspberry pi 使用有興趣的，參考看看囉！

同場加映：docker 存取 usb

因為目前進行的案子需要用到 serialPort 的操作，也剛好嘗試看看直接存取 usb，compose 設定檔如下

```
web-arm:
  container_name: hmi-web
  image: hypriot/rpi-node:4
  command: "/bin/bash -l -c 'npm start'"
  ports:
    - "3000:3000"
  working_dir: /app
  volumes:
    - ./:/app
  devices:
    - /dev/ttyUSB0:/dev/ttyUSB0
  environment:
    SERIALPORT: '/dev/ttyUSB0'
```

實際測試結果，確實可以對 USB 進行操作，也給需要的人參考

參考資料

<https://gist.github.com/StefanScherer/c3890d8277455e6c257d>

<http://blog.hypriot.com/post/let-docker-swarm-all-over-your-raspberry-pi-cluster/>

<https://github.com/docker/swarm/blob/master/scheduler/filter/README.md>





Share

61



Spooky Xie

Read [more posts](#) by this author.

Share this post



READ THIS NEXT

透過 docker run command restart docker swarm

延續 Raspberry pi with Docker Swarm 跨網域溝通使用 network，在實際上的應用有可能某個 Raspberry pi 重啟或更換，希望可以透過 docker run 的方式讓 docker 重新上線，也因此研究解析了一下...

<https://github.com/docker/swarm>

trunk studio © 2015

YOU MIGHT ENJOY

Container Summit 2015: 以 Docker Swarm 打造多主機叢集環境

最新的 Docker 1.9.0 在 11 月份正式推出，此講題將介紹最新的 Network 的特性，介紹如何透過

Docker Machine 建置多台 Docker Engine，使用 Docker...

Proudly published with Ghost