Lab: docker swarm

請先在 Play with Docker 建立 3 個 instances

分別為 node1 / node2 / node3

建立 swarm manager node

於 node1

使用 docker swarm init 建立 manager node

\$ docker swarm init --advertise-addr 192.168.0.63

Swarm initialized: current node (o2yfoboxvs7uh4btyp5rxd09b) is now a manager.

To add a worker to this swarm, run the following command:

docker swarm join --token SWMTKN-1-3pkc4kkza6unpgkz19zgcuu6g17qdgajts1gekyj03fpmlyxba-6hgyskapvls9dobzc2d4db5g1 192.168.0.63:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

觀察相關資訊

\$ docker node ls

ID HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION o2yfoboxvs7uh4btyp5rxd09b * node1 Ready Active Leader 18.03.1-ce

於 node2 以及 node3

使用剛剛上面的指令加入 docker swarm

\$\foothar{swarm join --token swmtkn-1-3pkc4kkza6unpgkz19zgcuu6g17qdgajts1gekyj03fpmlyxba-6hgyskapvls9dobzc2dddb5g1 192.168.0.63:2377

This node joined a swarm as a worker.

於 node1

觀察相關資訊

\$ docker node ls

ID HOSTNAME **STATUS** AVAILABILITY MANAGER STATUS ENGINE VERSIONo2yfoboxvs7uh4btyp5rxd09b * node1 Ready Active Leader 18.03.1-ce t4e2d444sa1dmognsv18gfihn node2 Ready Active 18.03.1-ce ijg1fo1h6s7hmge5hmho3nlcb Ready Active node3 18.03.1-ce

建立 docker service 測試

\$ docker service create --name cluster --constraint "node.role == worker" -p 80:80 russmckendrick/cluster

使用 visualizer 觀察

\$ docker run -it -d -p 8080:8080 -e HOST=192.168.0.63 -v /var/run/docker.sock:/var/run/docker.sock dockersamples/visualizer

觀察相關資訊

\$ docker service ls

NAME MODE ID MODE 1/1 **REPLICAS IMAGE** PORTShgg0qsiqduf9

cluster replicated russmckendrick/cluster:latest *:80->80/tcp

觀察相關資訊

\$ docker service inspect cluster

測試 service scale

\$ docker service scale cluster=4

cluster scaled to 4

overall progress: 4 out of 4 tasks

verify: Service converged

觀察相關資訊

\$ docker service ls

NAME MODE REPLICAS **IMAGE PORTS** 4/4 russmckendrick/cluster:latest $*:80 \rightarrow 80/tcp$ hgg0qsiqduf9 cluster replicated

可以到 worker node 上面觀察 docker ps

移除服務

\$ docker service rm cluster cluster