

Practica 3

Comenzamos siguiendo los pasos del guion.

The screenshot shows the Docker Playground interface in a web browser. The session is titled 'ccs0orch_ccs0otktugg00f1et70'. The IP address is 192.168.0.23, memory usage is 2.44% (97.41MiB / 3.906GiB), and CPU usage is 1.17%. The URL is https://labs.play-with-k8s.com/p/ccs0orchtugg00f1et6g#ccs0orch_ccs0otktugg00f1et70. The interface includes a sidebar with a clock showing 03:57:06, a 'CLOSE SESSION' button, and a list of instances. The main area displays a terminal with the following content:

```
2. Initialize cluster networking:
kubectl apply -f https://raw.githubusercontent.com/cloudnativelabs/kube-router/master/daemonset/kubeadm-kuberouter.yaml

3. (Optional) Create an nginx deployment:
kubectl apply -f https://raw.githubusercontent.com/kubernetes/website/master/content/en/examples/application/nginx-app.yaml

The PKM team.

[node1 ~]$ ls
anaconda-k8s.cfg
[node1 ~]$ docker swarm init --advertise-addr 192.168.0.23
Swarm initialized: current node (gn3ayf5w3enib3litb10t9chb) is now a manager.

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

    docker swarm join --token SWMTKN-1-5dttxyexjc5jz03eprepu0wtln7hsj5przyj2bvuceelnq2mtp-3gi45jnkz0lwjmr3rpggand 192.168.0.23:2377

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

[node1 ~]$ docker node ls
ID                HOSTNAME    STATUS    AVAILABILITY    MANAGER STATUS    ENGINE VERSION
gn3ayf5w3enib3litb10t9chb * node1      Ready     Active           Leader            20.10.1
[node1 ~]$ docker node ls
ID                HOSTNAME    STATUS    AVAILABILITY    MANAGER STATUS    ENGINE VERSION
gn3ayf5w3enib3litb10t9chb * node1      Ready     Active           Leader            20.10.1
y7ar4y1vseztgafpmw74piwc node2      Ready     Active           Leader            20.10.1
zvhh3s3ux1l15pr101z9ifa node3      Ready     Active           Leader            20.10.1
[node1 ~]$
```

The screenshot shows the Docker Playground interface in a web browser. The session is titled 'ccrat3kh_ccrbcfchtugg00em8ssg'. The IP address is 192.168.0.17, memory usage is 2.23% (89.07MiB / 3.906GiB), and CPU usage is 1.60%. The URL is https://labs.play-with-k8s.com/p/ccrat3khtugg00em8rg0#ccrat3kh_ccrbcfchtugg00em8ssg. The interface includes a sidebar with a clock showing 03:12:00, a 'CLOSE SESSION' button, and a list of instances. The main area displays a terminal with the following content:

```
WARNING!!!!

This is a sandbox environment. Using personal credentials
is HIGHLY discouraged. Any consequences of doing so, are
completely the user's responsibilities.

You can bootstrap a cluster as follows:

1. Initialize cluster master node:
kubeadm init --apiserver-advertise-address $(hostname -i) --pod-network-cidr 10.5.0.0/16

2. Initialize cluster networking:
kubectl apply -f https://raw.githubusercontent.com/cloudnativelabs/kube-router/master/daemonset/kubeadm-kuberouter.yaml

3. (Optional) Create an nginx deployment:
kubectl apply -f https://raw.githubusercontent.com/kubernetes/website/master/content/en/examples/application/nginx-app.yaml

The PKM team.

[node2 ~]$ docker swarm join --token SWMTKN-1-2qxq5ucoegksudrebjx2mlz57wptefw8jafeyyy9rsu4cqhesw-8zopi9sg6nahiku0uy2rrzza 192.168.0.18:2377
This node joined a swarm as a worker.
[node2 ~]$
```

Vemos como desde el nodo 2 nos podemos conectar al host principal (nodo 1)
Se van alternando entre nodo1 nodo2 y nodo 3

Actividades Navegador web Firefox 1 de oct 11:52

SWAD: plataforma de ap: x practica_3_CPD_v1.pdf Docker Playground

https://labs.play-with-k8s.com/p/ccs0orchtugg00f1et6g#ccs0orch_ccs0p0shtugg00f1et7g

03:53:35

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.23 node1

192.168.0.22 node2

192.168.0.21 node3

ccs0orch_ccs0p0shtugg00f1et7g

IP 192.168.0.22 8080

Memory 28.86% (1.127GiB / 3.906GiB) CPU 1.23%

URL lp172-18-0-57-ccs0orchtugg00f1et6g.direct.labs.play-with-k8s.com

DELETE

You can bootstrap a cluster as follows:

1. Initializes cluster master node:
kubeadm init --apiserver-advertise-address \$(hostname -i) --pod-network-cidr 10.5.0.0/16
2. Initialize cluster networking:
kubect1 apply -f https://raw.githubusercontent.com/cloudnativelabs/kube-router/master/daemonset/kubeadm-kuberouter.yaml
3. (Optional) Create an nginx deployment:
kubect1 apply -f https://raw.githubusercontent.com/kubernetes/website/master/content/en/examples/application/nginx-app.yaml

The FWK team.

```
[node2 ~]$ docker swarm join --token SWMTKN-1-5dttxyexjc5jz03epreu0wtln7hsj5przj2bvuceelnq2mtp-3gi45jnkz0lwjmr3rpggand 192.168.0.23:2377
This node joined a swarm as a worker.
[node2 ~]$ curl http://192.168.0.23:8080
node3
[node2 ~]$ curl http://192.168.0.23:8080
node2
[node2 ~]$ curl http://192.168.0.23:8080
node1
[node2 ~]$ curl http://192.168.0.23:8080
node3
[node2 ~]$ docker service ps web
Error response from daemon: This node is not a swarm manager. Worker nodes can't be used to view or modify cluster state. Please run this command on a manager node or promote the current node to a manager.
[node2 ~]$
```

Actividades Navegador web Firefox 1 de oct 11:52

SWAD: plataforma de ap: x practica_3_CPD_v1.pdf Docker Playground

https://labs.play-with-k8s.com/p/ccs0orchtugg00f1et6g#ccs0orch_ccs0otkhtugg00f1et70

03:53:26

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.23 node1

192.168.0.22 node2

192.168.0.21 node3

ccs0orch_ccs0otkhtugg00f1et70

IP 192.168.0.23 8080

Memory 29.05% (1.135GiB / 3.906GiB) CPU 1.67%

URL lp172-18-0-7-ccs0orchtugg00f1et6g.direct.labs.play-with-k8s.com

DELETE

```
[node1 ~]$ ls
anaconda-k8s.cfg
[node1 ~]$ docker swarm init --advertise-addr 192.168.0.23
Swarm initialized: current node (gn3ayf5w3enib31itb10t9chb) is now a manager.
To add a worker to this swarm, run the following command:
    docker swarm join --token SWMTKN-1-5dttxyexjc5jz03epreu0wtln7hsj5przj2bvuceelnq2mtp-3gi45jnkz0lwjmr3rpggand 192.168.0.23:2377
To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
[node1 ~]$ docker node ls
ID                HOSTNAME        STATUS        AVAILABILITY    MANAGER STATUS    ENGINE VERSION
gn3ayf5w3enib31itb10t9chb * node1            Ready         Active           Leader            20.10.1
[node1 ~]$ docker node ls
ID                HOSTNAME        STATUS        AVAILABILITY    MANAGER STATUS    ENGINE VERSION
gn3ayf5w3enib31itb10t9chb * node1            Ready         Active           Leader            20.10.1
7tarayivsoztgfbpmw74plwc node2            Ready         Active           Ready             20.10.1
txvhh3s3uxlxl5pr10iz9lfa node3            Ready         Active           Ready             20.10.1
[node1 ~]$ docker service create --name web --replicas 3 --mount type=bind,src=/etc/hostname,dst=/usr/share/nginx/html/index.html,readonly --publish published=8080,target=80 nginx
7xatub0lxtadast69vulbmk
overall progress: 3 out of 3 tasks
1/3: running [=====>]
2/3: running [=====>]
3/3: running [=====>]
Verify: Service converged
[node1 ~]$ docker service ps web
ID                NAME        IMAGE        NODE        DESIRED STATE    CURRENT STATE    ERROR    PORTS
07rhlgcw3xsl     web.1       nginx:latest node2        Running           Running about a minute ago
94b1n8gjo270     web.2       nginx:latest node3        Running           Running about a minute ago
4wc3aed57gx6     web.3       nginx:latest node1        Running           Running about a minute ago
[node1 ~]$
```

Reducimos el numero de nodos a dos:

The screenshot shows the Docker Playground interface with three nodes listed on the left: node1 (192.168.0.13), node2 (192.168.0.12), and node3 (192.168.0.11). The main panel displays the details for node1, including its IP (192.168.0.13), memory usage (33.75%), CPU usage (8.68%), and URL (ip172-18-0-9-ccs1a8khtugg00f1etvg.direct.labs.play-with-k8s.com). The terminal window shows the output of a 'docker service ps web' command, which lists the containers for the 'web' service across all three nodes.

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
ehdk23bytwh5	web.1	nginx:latest	node1	Running	Running 37 seconds ago		
3bjxer70kf92	web.2	nginx:latest	node2	Running	Running 37 seconds ago		
m7tf8j1t2wkg	web.3	nginx:latest	node3	Running	Running 37 seconds ago		

Si apagamos uno de los dos nodos (nodo 2) se activará el otro nodo automáticamente (nodo 3)

The screenshot shows the Docker Playground interface with three nodes listed on the left: node1 (192.168.0.13), node2 (192.168.0.12), and node3 (192.168.0.11). The main panel displays the details for node1, including its IP (192.168.0.13), memory usage (33.07%), CPU usage (1.27%), and URL (ip172-18-0-9-ccs1a8khtugg00f1etvg.direct.labs.play-with-k8s.com). The terminal window shows the output of a 'docker service ps web' command, which lists the containers for the 'web' service across all three nodes. Node2 is now in a 'Shut-down' state.

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
ehdk23bytwh5	web.1	nginx:latest	node1	Running	Running 5 minutes ago		
3bjxer70kf92	web.2	nginx:latest	node2	Running	Running 5 minutes ago		
m7tf8j1t2wkg	web.3	nginx:latest	node3	Running	Running 5 minutes ago		

Miguel Carracedo Rodríguez

Por ultimo si reactivamos el nodo 2 (docker start) y ajustamos el tamaño en el host principal a 3 vuelve a estar activo el nodo 2:

The screenshot shows the Docker Playground interface in a web browser. The session is titled 'ccs1a8kh_ccs1a9shtugg00f1eu00'. The left sidebar shows a list of instances: node1 (192.168.0.13), node2 (192.168.0.12), and node3 (192.168.0.11). The main panel displays the details of the selected instance, including its IP (192.168.0.13), memory usage (33.10%), CPU usage (1.18%), and a URL. Below this, a terminal window shows the following output:

```
conds ago
(node1 ~)$ docker service scale web=2
web scaled to 2
overall progress: 2 out of 2 tasks
1/2: running [=====]
2/2: running [=====]
verify: Service converged
(node1 ~)$ docker service ps web
IP          NAME          PORTS          NODE          DESIRED STATE  CURRENT STATE
ehdk23bytwh5 web.1         nginx:latest   node1         Running         Running 10 mi
nutes ago
4nxx02pcqlog web.2         nginx:latest   node3         Running         Running 4 min
utes ago
3bjxer70kfg2 \_ web.2      nginx:latest   node2         Shutdown        Shutdown 2 mi
nutes ago
(node1 ~)$ docker service scale web=3
web scaled to 3
overall progress: 3 out of 3 tasks
1/3: running
2/3: running
3/3: running
verify: Service converged
(node1 ~)$ docker service ps web
IP          NAME          PORTS          NODE          DESIRED STATE  CURRENT STATE
ehdk23bytwh5 web.1         nginx:latest   node1         Running         Running 12 mi
nutes ago
4nxx02pcqlog web.2         nginx:latest   node3         Running         Running 7 min
utes ago
3bjxer70kfg2 \_ web.2      nginx:latest   node2         Shutdown        Shutdown 4 mi
nutes ago
pyyuhndehc7b web.3         nginx:latest   node2         Running         Running 8 sec
conds ago
(node1 ~)$
```

En la ultima parte de la practica me da un error (espacio insuficiente) al intentar clonar el repositorio