CPD - Práctica 5

Elena Cantero Molina

Docker Swarm: Combinando múltiples máquinas para la ejecución de contenedores Docker.

- (obligatorio): Realizar diversas capturas donde se muestren:
  - La creación de las máquinas virtuales.
  - El inicio del manager de docker swarm.
  - Capturas de las redes internas.
  - Ejecución del servicio web
  - Cuando los 3 nodos están activos
  - Cuando se cambia de escala a 2
  - Cuando apagamos un nodo activo y sólo ejecuta un nodo,y la activación automática del segundo nodo.
- (opcional): Capturas de diversas ejecuciones en la plataforma Katakoda.

I) Docker-machine

Instalación Docker-Machine

```
elena@elena-X555LDB:~/Escritorio/CPD/Practica 5$ base=https://github.com/docker/mac
hine/releases/download/v0.16.0 &&
>   curl -L $base/docker-machine-$(uname -s)-$(uname -m) >/tmp/docker-machine &&
>   sudo mv /tmp/docker-machine /usr/local/bin/docker-machine &&
>   chmod +x /usr/local/bin/docker-machine
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   617    0   617    0     0    944      0 --:--:-- --:--:-- --:--:--   943
100 26.8M  100 26.8M    0     0  3414k      0  0:00:08  0:00:08 --:--:-- 4493k
[sudo] contraseña para elena:
```

Versión Docker-Machine

```
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine version
docker-machine version 0.16.0, build 702c267f
```

Creación de una máquina con docker-machine

```
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine create m1
Creating CA: /home/elena/.docker/machine/certs/ca.pem
Creating client certificate: /home/elena/.docker/machine/certs/cert.pem
Running pre-create checks...
(m1) Image cache directory does not exist, creating it at /home/elena/.docker/machine/cache...
(m1) No default Boot2Docker ISO found locally, downloading the latest release...
(m1) Latest release for github.com/boot2docker/boot2docker is v19.03.4
(m1) Downloading /home/elena/.docker/machine/cache/boot2docker.iso from https://github.com/boot2docker/boot2docker/releases/download/v19.03.4/b
oot2docker.iso...
(m1) 0%....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%
Creating machine...
(m1) Copying /home/elena/.docker/machine/cache/boot2docker.iso to /home/elena/.docker/machine/machines/m1/boot2docker.iso...
(m1) Creating VirtualBox VM...
(m1) Creating SSH key...
(m1) Starting the VM...
(m1) Check network to re-create if needed...
(m1) Found a new host-only adapter: "vboxnet2"
(m1) Waiting for an IP...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with boot2docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: docker-machine env m1
root@elena-X555LDB:~/Escritorio/CPD/Practica 5#
```

Listado de máquinas, entrada por ssh a la máquina m1 y consulta de la IP de la máquina m1.

```
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine ls
NAME   ACTIVE   DRIVER       STATE     URL                         SWARM   DOCKER      ERRORS
m1     -        virtualbox   Running   tcp://192.168.99.100:2376           v19.03.4
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine ssh m1
   ( '>')
  /) TC (\   Core is distributed with ABSOLUTELY NO WARRANTY.
 (/-_--_-\)           www.tinycorelinux.net

docker@m1:~$ exit
logout
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine ip m1
192.168.99.100
root@elena-X555LDB:~/Escritorio/CPD/Practica 5#
```

II) Evaluando Docker Swarm

Creación de docker swarm en m1 y visulización de los nodos

```
docker@m1:~$ docker swarm init --advertise-addr 192.168.99.100
Swarm initialized: current node (ivyl4ku3p1n66sayzbuccc9hq) is now a manager.

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

    docker swarm join --token SWMTKN-1-3mda9wfil5y4vshdc6bwplbni0blp7rsrmkh2lb60785x3f6g6-4s477n5yy8nvd87kgpgx58u9e 192.168.99.100:2377

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

docker@m1:~$ docker node ls
ID                            HOSTNAME            STATUS              AVAILABILITY        MANAGER STATUS      ENGINE VERSION
ivyl4ku3p1n66sayzbuccc9hq *   m1                  Ready               Active              Leader              19.03.4
docker@m1:~$
```

Entrar por ssh a m2 y añadir los nodos m2 y m3

```
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine ssh m2
   ( '>')
  /) TC (\   Core is distributed with ABSOLUTELY NO WARRANTY.
 (/-_--_-\)           www.tinycorelinux.net

<a9wfil5y4vshdc6bwplbni0blp7rsrmkh2lb60785x3f6g6-4s477n5yy8nvd87kgpgx58u9e 192.168.99.100:2377
This node joined a swarm as a worker.
docker@m2:~$ docker swarm join --token SWMTKN-1-3mda9wfil5y4vshdc6bwplbni0blp7rsrmkh2lb60785x3f6g6-4s477n5yy8nvd87kgpgx58u9e 192.168.99.100:2>
Error response from daemon: This node is already part of a swarm. Use "docker swarm leave" to leave this swarm and join another one.
docker@m2:~$ exit
logout
exit status 1
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine ssh m3
   ( '>')
  /) TC (\   Core is distributed with ABSOLUTELY NO WARRANTY.
 (/-_--_-\)           www.tinycorelinux.net

docker@m3:~$ docker swarm join --token SWMTKN-1-3mda9wfil5y4vshdc6bwplbni0blp7rsrmkh2lb60785x3f6g6-4s477n5yy8nvd87kgpgx58u9e 192.168.99.100:2>
This node joined a swarm as a worker.
docker@m3:~$
```

III) Creamos un servicio

Creación del servicio en m1

```
docker@m1:~$ docker service create --name web --replicas 3 --mount type=bind,src=/etc/hostname,dst=/usr/share/nginx/html/index.html,readonly >
thzufhkcycoxxrghpis2048tf
overall progress: 3 out of 3 tasks
1/3: running   [==================================================>]
2/3: running   [==================================================>]
3/3: running   [==================================================>]
verify: Service converged
docker@m1:~$ exit
logout
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# curl http://192.168.99.102:8080
m1
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# curl http://192.168.99.102:8080
m2
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# curl http://192.168.99.102:8080
m3
root@elena-X555LDB:~/Escritorio/CPD/Practica 5#
```

Comprobación de que el servicio ha lanzado 3 contenedores, uno en cada nodo y reducción del número de nodos

```
docker@m1:~$ docker service ps web
ID                  NAME                IMAGE               NODE                DESIRED STATE       CURRENT STATE           ERROR
 PORTS
iv703m6fa0ub        web.1               nginx:latest        m2                  Running             Running 5 minutes ago

h142n4mvrd0e        web.2               nginx:latest        m3                  Running             Running 5 minutes ago

3qvrj52k8tj9        web.3               nginx:latest        m1                  Running             Running 5 minutes ago

docker@m1:~$ 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
docker@m1:~$ docker service ps web
ID                  NAME                IMAGE               NODE                DESIRED STATE       CURRENT STATE           ERROR
 PORTS
iv703m6fa0ub        web.1               nginx:latest        m2                  Running             Running 5 minutes ago

h142n4mvrd0e        web.2               nginx:latest        m3                  Running             Running 5 minutes ago

docker@m1:~$
```

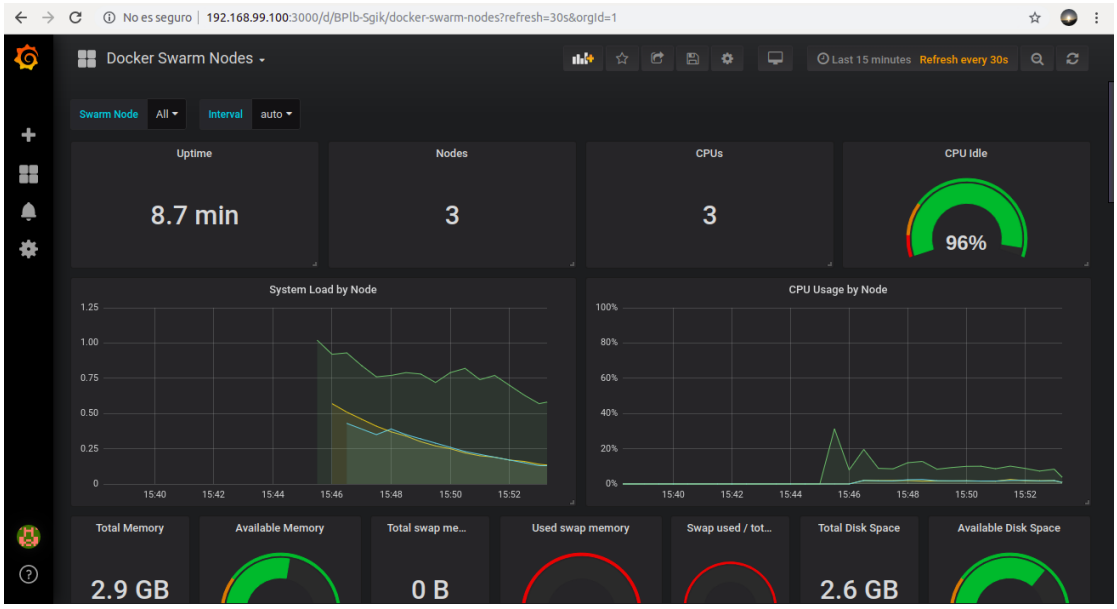
Parada de un nodo

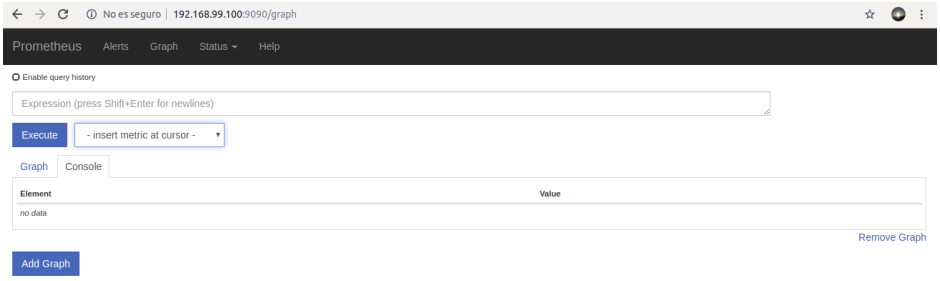
```
root@elena-X555LDB:~/Escritorio/CPD/Practica 5# docker-machine stop m3
Stopping "m3"...
Machine "m3" was stopped.
```

IV) Monitorizar Docker Swarm

```
docker@m1:~$ git clone https://github.com/stefanprodan/swarmprom.git
Cloning into 'swarmprom'...
remote: Enumerating objects: 803, done.
remote: Total 803 (delta 0), reused 0 (delta 0), pack-reused 803
Receiving objects: 100% (803/803), 5.79 MiB | 1.13 MiB/s, done.
Resolving deltas: 100% (460/460), done.
docker@m1:~$ cd swarmprom
docker@m1:~/swarmprom$ ADMIN_USER=admin
docker@m1:~/swarmprom$ -bash: words: bad array subscript
^C
docker@m1:~/swarmprom$ ADMIN_USER=admin
docker@m1:~/swarmprom$ ADMIN_PASSWORD=admin
docker@m1:~/swarmprom$ SLACK_URL=https://hooks.slack.com/services/TOKEN
docker@m1:~/swarmprom$ SLACK_CHANNEL=devops-alerts
docker@m1:~/swarmprom$ SLACK_USER=alertmanager
docker@m1:~/swarmprom$ docker stack deploy -c docker-compose.yml mon
Creating network mon_net
Creating config mon_caddy_config
Creating config mon_dockerd_config
Creating config mon_node_rules
Creating config mon_task_rules
Creating service mon_dockerd-exporter
Creating service mon_cadvisor
Creating service mon_grafana
Creating service mon_alertmanager
Creating service mon_unsee
Creating service mon_node-exporter
Creating service mon_prometheus
Creating service mon_caddy
docker@m1:~/swarmprom$
```

Comprobación de que Docker Swarm funciona bien




No es seguro | 192.168.99.100:9090/graph
Prometheus
Alerts
Graph
Status
Help
Enable query history
Expression (press Shift+Enter for newlines)
Execute
- insert metric at cursor -
Graph
Console
Element
Value
no data
Remove Graph
Add Graph