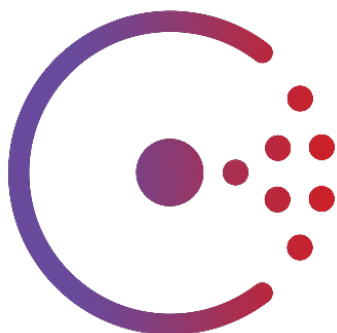


DOCKER-2 Docker Swarm Cluster amb Consul



Continguts

- *Requeriments previs*
 - *Instal·lar el requeriments:*
- *Consul*
 - *Docker Swarm*
 - *Swarm-master*
 - *Swarm-node*
 - *Connexió amb el Swarm-master*

La idea és disposar d'un clúster amb Docker Engine i poder instanciar-hi allà els nostres contenidors Docker.

Docker Swarm: Ens permet crear un clúster Docker. El node màster rebrà i distribuirà els contenidors als nodes secundaris on s'executaran.

Consul: Ens proporcionarà el servei de discovery dels nous nodes que afegim al clúster swarm.

Requeriments previs

- **Docker Engine:** Ens permet crear els contenidors, gestionar-los i instanciar-los
- **Docker Machine:** Ens permet crear i gestionar les màquines virtuals
- **Driver Docker Machine Opennebula:** Ens permet utilitzar el docker-machine al cloud de l'OpenNebula

Instal·lar el requeriments:

Per a poder implementar el **Docker Swarm Clúster** amb l'OpenNebula hem d'instalar: **Docker Engine**, **Docker Machine** i el **Driver de Docker Machine per l'Opennebula**.

Teniu disponible el següent tutorial de com fer-ho: *DOCKER-1 Docker Machine OpenNebula Driver*

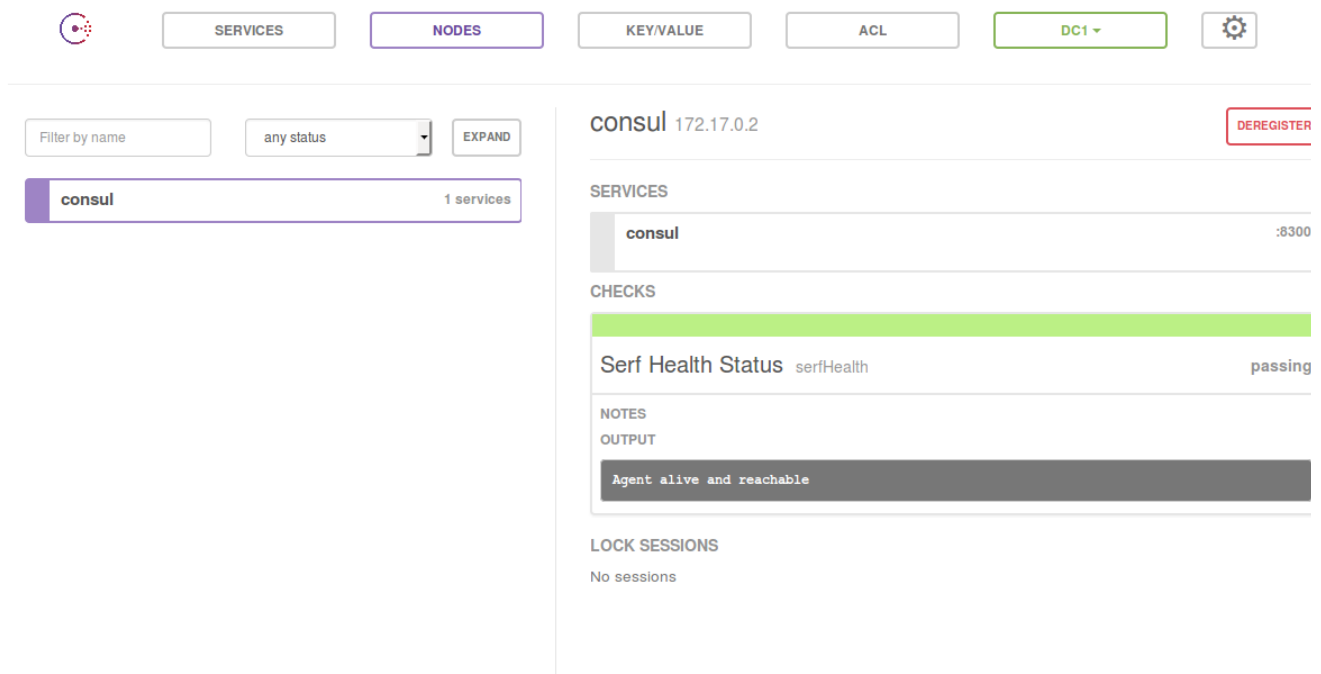
Consul

Disposarem d'una maquina virtual amb Docker Engine que ens proporcionarà el servei de discovery amb *Consul*

Farem servir el driver de Docker Machine per l'OpenNebula, i crearem una màquina virtual anomenada Consul, on després executem un contenidor amb el propi servei de Consul dins d'aquesta.

```
docker-machine create -d opennebula --opennebula-network-id [network_id] --opennebula-image-id [boot2docker_image_id] --opennebula-b2d-size [volatile_disk_size] consul
docker $(docker-machine config consul) run -d -p "8500:8500" -h "consul" progrium/consul -server -bootstrap
CONSUL_IP=$(docker-machine ip consul)
```

Podem verificar que s'ha instal·lat correctament el servei de Consul, accedint a la IP de la màquina, que acabem de crear, pel port 8500.



The screenshot shows the Consul web interface. At the top, there are tabs for SERVICES, NODES, KEY/VALUE, ACL, and DC1. The NODES tab is selected. Below the tabs, there is a search bar and a filter dropdown. The main content area shows a list of nodes, with 'consul' selected. The details for the 'consul' node are displayed on the right, including its IP address (172.17.0.2), status (passing), and output (Agent alive and reachable).

Docker Swarm

Disposarem de dos tipus de màquines virtuals amb Swarm:

- **Swarm Master:** Aquesta màquina amb swarm serà l'encarregada de distribuir les instàncies dels contenidors als diferents nodes que tinguem al clúster.
- **Swarm Node:** Aquesta o aquestes màquines seran les que executaran propiament els contenidors Docker.

Swarm-master

Creem la màquina virtual, amb el driver de Docker Machine, pel node màster dels clúster:

```
docker-machine create -d opennebula --opennebula-network-id [network_id] --opennebula-image-id [boot2docker_image_id] --opennebula-b2d-size [volatile_disk_size] --swarm --swarm-master --swarm-discovery="consul://$CONSUL_IP:8500" --engine-opt cluster-store=consul://$CONSUL_IP:8500 --engine-opt cluster-advertise="eth0:2376" swarm-master
```

Swarm-node

Creem la màquina virtual amb el driver de Docker Machine, pels diferents nodes dels clúster:

```
docker-machine create -d opennebula --opennebula-network-id [network_id] --opennebula-image-id [boot2docker_image_id] --opennebula-b2d-size [volatile_disk_size] --swarm --swarm-discovery="consul://$CONSUL_IP:8500" --engine-opt cluster-store=consul://$CONSUL_IP:8500 --engine-opt cluster-advertise="eth0:2376" swarm-node-01
```

Podem crear tants nodes com vulguem. Només hem d'anar modificant el nom de la màquina virtual.

Connexió amb el Swarm-master

Un cop hem creat les diferents màquines virtuals que componen el clúster, podem connectar-nos al node màster amb la següent comanda:

```
eval $(docker-machine env --swarm swarm-master)
```

Virtual Machines



☐	ID	Owner	Group	Name	Status	Host	IPs	
☐	2265		Docker	swarm-node-01	RUNNING	cluster06		
☐	2264		Docker	swarm-master	RUNNING	cluster04		
☐	2261		Docker	consul	RUNNING	cluster04		

Showing 1 to 3 of 3 entries

Previous

1

Next

10

3 TOTAL 3 ACTIVE 0 OFF 0 PENDING 0 FAILED

NAME	ACTIVE	DRIVER	STATE	URL	SWARM	DOCKER	ERRORS
consul	-	opennebula	Running	tcp://10.00.14.255:2376		v1.10.2	
swarm-master	-	opennebula	Running	tcp://10.00.14.255:2376	swarm-master (master)	v1.10.2	
swarm-node-01	-	opennebula	Running	tcp://10.00.14.255:2376	swarm-master	v1.10.2	

docker info

```
Containers: 3
  Running: 3
  Paused: 0
  Stopped: 0
Images: 2
Server Version: swarm/1.1.3
Role: primary
Strategy: spread
Filters: health, port, dependency, affinity, constraint
Nodes: 2
  swarm-master: 611be10 :2376
    L Status: Healthy
    L Containers: 2
    L Reserved CPUs: 0 / 1
    L Reserved Memory: 0 B / 517.6 MiB
    L Labels: executiondriver=native-0.2, kernelversion=4.1.18-boot2docker, operatingsystem=Boot2Docker 1.10.2 (TCL 6.4
; master : 611be10 - Tue Feb 23 00:06:40 UTC 2016, provider=opennebula, storagedriver=aufs
    L Error: (none)
    L UpdatedAt: 2016-04-06T08:02:43Z
  swarm-node-01: 3fd4a80e :2376
    L Status: Healthy
    L Containers: 1
    L Reserved CPUs: 0 / 1
    L Reserved Memory: 0 B / 1.021 GiB
    L Labels: executiondriver=native-0.2, kernelversion=4.1.18-boot2docker, operatingsystem=Boot2Docker 1.10.2 (TCL 6.4
; master : 611be10 - Tue Feb 23 00:06:40 UTC 2016, provider=opennebula, storagedriver=aufs
    L Error: (none)
    L UpdatedAt: 2016-04-06T08:01:55Z
Plugins:
  Volume:
  Network:
Kernel Version: 4.1.18-boot2docker
Operating System: linux
Architecture: amd64
CPUs: 2
Total Memory: 1.527 GiB
Name: swarm-master
```

Xarxa

Un cop tenim creat el nostre clúster amb Swarm també tenim l'opció de poder crear xarxes privades dins del clúster.

```
docker network create --driver overlay --subnet=10.0.1.0/24 overlay_net
docker network ls
```

NETWORK ID	NAME	DRIVER
6968b8228eb9	swarm-master/none	null
04168ccd6192	swarm-master/host	host
a52245a565f3	overlay_net	overlay
3fd4a80e457b	swarm-master/bridge	bridge
f79e429ab6f1	swarm-node-01/none	null
2a5cd0d44a16	swarm-node-01/host	host
08957e6ef5a6	swarm-node-01/bridge	bridge

Per a més informació : <http://opennebula.org/docker-swarm-with-opennebula/>

Exemple d'ús

Executar contenidors Docker al clúster creat:

- Ara podem començar a fer servir Docker sobre el clúster que acabem de crear:

Utilitzarem un contenidor amb nginx, per exemple:

```
docker pull nginx
docker run --name mynginx -d -p 80:80 nginx
```

```
Using default tag: latest
swarm-node-01: Pulling nginx:latest... : downloaded
swarm-master: Pulling nginx:latest... : downloaded
~$ docker run --name mynginx -d -p 80:80 nginx
a4aa9d56ff83043ebcd99364bf006d39c4d0c05250230853a046f85e7b105f69
```

- Podem comprovar com el contenidor s'està executant en un node del clúster:
Podem observar que el contenidor amb el nostre nginx esta corrent a la màquina swarm-node-01

```
docker ps
```

CONTAINER ID	IMAGE NAMES	COMMAND	CREATED	STATUS	PORTS
a4aa9d56ff83 ->80/tcp, 443/tcp	nginx swarm-node-01/mynginx	"nginx -g 'daemon off'"	12 minutes ago	Up 17 minutes	0.0.0.0:80->80

Per accedir al servei del contenidor hem de posar la direcció IP del node del clúster al navegador.