

Lab 6 – Docker-swarm

Manage cluster of machines

We're going to build a cluster of Docker servers using Docker Swarm. You have 3 Docker servers:

- 1 manager
- 2 workers

Cluster initialization

1. Initialize the swarm cluster from the manager host using its IP from the same range than the 2 workers:

```
vagrant@Manager:~$ docker swarm init --advertise-addr 192.168.205.10
Swarm initialized: current node (tgrnm51domfc5f64gt6tobvak) is now a manager.

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

    docker swarm join --token SWMTKN-1-27y0ngobz8xzty58b0v6nihqxxgq8ovwp55roajbmiqlrey70a6-byk75lw40y382p2uqes3e31cw 192.168.205.10:2377

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

- What does this command generate?

2 tokens

- Check that the cluster has been created and that the manager is the single member.

```
vagrant@Manager:~$ docker node ls
ID                HOSTNAME        STATUS    AVAILABILITY    MANAGER STATUS    ENGINE VERSION
tgrnm51domfc5f64gt6tobvak *   Manager        Ready     Active           Leader             24.0.2
vagrant@Manager:~$
```

- What is the token generated by this command used for? Display the 2 tokens.

```
vagrant@Manager:~$ docker swarm join-token manager
To add a manager to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-27y0ngobz8xzty58b0v6nihqxxgq8ovwp55roajbmiqlrey70a6-emx6fslmb5efb0kj311k8l0b4 192.168.205.10:2377

vagrant@Manager:~$
vagrant@Manager:~$ docker swarm join-token worker
To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-27y0ngobz8xzty58b0v6nihqxxgq8ovwp55roajbmiqlrey70a6-byk75lw40y382p2uqes3e31cw 192.168.205.10:2377

vagrant@Manager:~$
```

2. Join worker1 and worker2 VMs to the cluster (as workers).

```
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh worker1
Last login: Mon Jun 12 13:00:55 2023 from 10.0.2.2
vagrant@worker1:~$ sudo docker swarm join --token SWMTKN-1-27y0ngobz8xzt58b0v6nihxgq8ovwp55roajbmiqlrey70a6-byk75lw40y382p2uqes3e31cw 192.16
8.205.10:2377
This node joined a swarm as a worker.
vagrant@worker1:~$
```

```
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh worker2
Last login: Mon Jun 12 13:05:00 2023 from 10.0.2.2
vagrant@worker2:~$ sudo docker swarm join --token SWMTKN-1-27y0ngobz8xzt58b0v6nihxgq8ovwp55roajbmiqlrey70a6-byk75lw40y382p2uqes3e31cw 192.16
8.205.10:2377
This node joined a swarm as a worker.
vagrant@worker2:~$
```

- Check that the 2 workers are in the cluster and that they are ready.

```
vagrant@Manager:~$ docker node ls
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS	ENGINE VERSION
tgrnm51domfc5f64gt6tobvak *	Manager	Ready	Active	Leader	24.0.2
f9fzgu5dge6wt3xrmad30br6	worker1	Ready	Active		24.0.2
k8qert553j7o2d6vwmihqrouu	worker2	Ready	Active		24.0.2

```
vagrant@Manager:~$
```

3. What networks are created following cluster initialization.

```
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh Manager
Last login: Mon Jun 12 13:12:22 2023 from 10.0.2.2
vagrant@Manager:~$ docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
7938fc3cc0c9	bridge	bridge	local
6285cbd26031	docker_gwbridge	bridge	local
370478b68c6c	dockercoins_default	bridge	local
f01d80fd2323	host	host	local
n2svwazx6rj9	ingress	overlay	swarm
9ab24961264f	none	null	local

```
vagrant@Manager:~$ logout
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh worker1
Last login: Mon Jun 12 13:17:01 2023 from 10.0.2.2
vagrant@worker1:~$ sudo docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
ac2e8ea9077c	bridge	bridge	local
b9e92603da75	docker_gwbridge	bridge	local
bbea0cde38d4	host	host	local
n2svwazx6rj9	ingress	overlay	swarm
04d7d890dcf4	none	null	local

```
vagrant@worker1:~$ logout
brahim@Training:~/devops-lab/vagrant-ubuntu20-multiple$ vagrant ssh worker2
Last login: Mon Jun 12 13:12:30 2023 from 10.0.2.2
vagrant@worker2:~$ sudo docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
80d84ab58222	bridge	bridge	local
b0dea255c1a9	docker_gwbridge	bridge	local
89cdafd89a6e	host	host	local
n2svwazx6rj9	ingress	overlay	swarm
c8b9deb992df	none	null	local

```
vagrant@worker2:~$
```

- Describe them.
 - 1 network overlay: ingress
 - 3 networks bridge: docker_gwbridge

Launch of the Dockercoins application on Swarm

4. Create a network to isolate all services of the application. View details of our network.

```
vagrant@Manager:~$ docker network create --driver overlay dockercoins-overlay
n16xx5zs37ntbvi8ml6yib4ja
vagrant@Manager:~$
vagrant@Manager:~$ docker network inspect dockercoins-overlay
[
  {
    "Name": "dockercoins-overlay",
    "Id": "n16xx5zs37ntbvi8ml6yib4ja",
    "Created": "2023-06-12T13:22:30.50508073Z",
    "Scope": "swarm",
    "Driver": "overlay",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "10.0.1.0/24",
          "Gateway": "10.0.1.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": null,
    "Options": {
      "com.docker.network.driver.overlay.vxlanid_list": "4097"
    },
    "Labels": null
  }
]
vagrant@Manager:~$
vagrant@Manager:~$
```

5. Create the service running the redis database.
 - Wait until the server is in "running"

```
vagrant@Manager:~$ docker service create --name redis --network dockercoins-overlay redis
nk1xxn10f6tprl04agb8vvrhb
overall progress: 1 out of 1 tasks
1/1: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service ps redis
ID            NAME      IMAGE      NODE      DESIRED STATE  CURRENT STATE      ERROR      PORTS
28sle927peze  redis.1   redis:latest  worker2    Running        Running less than a second ago
vagrant@Manager:~$
```

- Once redis is ready, run the hasher, rng, and worker services.

```
vagrant@Manager:~$ docker service create --name woker --network dockercoins-overlay brahimhamdi/worker
afrf4orspumbomn55p4wu6fhu
overall progress: 1 out of 1 tasks
1/1: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service create --name rng -p 8001-8005:80 --network dockercoins-overlay brahimhamdi/rng
xqca13af35jum8xmh9nhkmbj
overall progress: 1 out of 1 tasks
1/1: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service create --name hasher -p 8006-8010:80 --network dockercoins-overlay brahimhamdi/hasher
ugg5b7i17wfxomag3hpdrcq72
overall progress: 1 out of 1 tasks
1/1: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service create --name webui -p 8000:80 --network dockercoins-overlay brahimhamdi/webui
14f2qd4i3hsthdryw94ax7a1v
overall progress: 1 out of 1 tasks
1/1: running [=====>]
verify: Service converged
vagrant@Manager:~$
```

- Check that these services are "running".

```
vagrant@Manager:~$ docker service ls
ID            NAME      MODE      REPLICAS  IMAGE      PORTS
ugg5b7i17wfx  hasher    replicated 1/1        brahimhamdi/hasher:latest  *:8006-8010->80/tcp
nk1xxn10f6tp  redis     replicated 1/1        redis:latest
xqca13af35ju  rng       replicated 1/1        brahimhamdi/rng:latest     *:8001-8005->80/tcp
14f2qd4i3hst  webui     replicated 1/1        brahimhamdi/webui:latest   *:8000->80/tcp
afrf4orspumb  woker     replicated 1/1        brahimhamdi/worker:latest

vagrant@Manager:~$ docker service ps woker rng hasher redis webui
ID            NAME      IMAGE      NODE      DESIRED STATE  CURRENT STATE      ERROR      PORTS
s9vas6ch0y7v  hasher.1  brahimhamdi/hasher:latest  Manager    Running        Running 2 minutes ago
28sle927peze  redis.1   redis:latest  worker2    Running        Running 8 minutes ago
42lsx2rdj6cg  rng.1     brahimhamdi/rng:latest     worker1    Running        Running 3 minutes ago
2yvtyca10lko  webui.1   brahimhamdi/webui:latest   worker1    Running        Running 9 seconds ago
7rwbq2inhsax  woker.1   brahimhamdi/worker:latest  Manager    Running        Running 6 minutes ago
vagrant@Manager:~$
```

- Give the IP address of each container as well as the node on which it is run.

```
vagrant@Manager:~$ docker service inspect woker | grep Addr
"Addr": "10.0.1.5/24"
vagrant@Manager:~$ docker service inspect rng | grep Addr
"Addr": "10.0.0.5/24"
"Addr": "10.0.1.8/24"
vagrant@Manager:~$ docker service inspect hasher | grep Addr
"Addr": "10.0.0.7/24"
"Addr": "10.0.1.11/24"
vagrant@Manager:~$ docker service inspect redis | grep Addr
"Addr": "10.0.1.2/24"
vagrant@Manager:~$ docker service inspect webui | grep Addr
"Addr": "10.0.0.9/24"
"Addr": "10.0.1.13/24"
vagrant@Manager:~$
```

Scaling on the swarm

6. scale out the worker service to 10 replicas.
 - Once all the replicas are "running", display the webui. What is the rate of generation of dockercoins?

```
vagrant@Manager:~$ docker service scale woker=10
woker scaled to 10
overall progress: 10 out of 10 tasks
1/10: running [=====>]
2/10: running [=====>]
3/10: running [=====>]
4/10: running [=====>]
5/10: running [=====>]
6/10: running [=====>]
7/10: running [=====>]
8/10: running [=====>]
9/10: running [=====>]
10/10: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service ps woker
```

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
7rwbq2inhsax	woker.1	brahimhamdi/woker:latest	Manager	Running	Running 12 minutes ago		
yw6m5z9bo9n1	woker.2	brahimhamdi/woker:latest	worker2	Running	Running less than a second ago		
w0yueniojkk5	woker.3	brahimhamdi/woker:latest	worker1	Running	Running less than a second ago		
pp4kejk9i9p	woker.4	brahimhamdi/woker:latest	Manager	Running	Running about a minute ago		
ka54q8jy6qdr	woker.5	brahimhamdi/woker:latest	worker2	Running	Running less than a second ago		
5pkw7yu2obqn	woker.6	brahimhamdi/woker:latest	worker1	Running	Running less than a second ago		
mboylwkk6yl9	woker.7	brahimhamdi/woker:latest	worker2	Running	Running less than a second ago		
yorp5x6f3084	woker.8	brahimhamdi/woker:latest	worker1	Running	Running less than a second ago		
nl4vcji8r4md	woker.9	brahimhamdi/woker:latest	Manager	Running	Running about a minute ago		
021awxoyd45g	woker.10	brahimhamdi/woker:latest	Manager	Running	Running about a minute ago		

```
vagrant@Manager:~$
```

7. We want to run exactly one rng instance on each host. Unfortunately, this scheduling mode cannot be activated / deactivated for a existing service.
So, we have to start by removing the existing rng service, then recreate it with global scheduling

```
vagrant@Manager:~$ docker service rm rng
rng
vagrant@Manager:~$ docker service create --name rng -p 8001-8005:80 --mode global --network dockercoins-overlay brahimhamdi/rng
hotxr79exxyvrtktuqciia87
overall progress: 3 out of 3 tasks
f9fzzgu5dge6: running [=====>]
k8qert553j7o: running [=====>]
tgrnm51domfc: running [=====>]
verify: Service converged
vagrant@Manager:~$ docker service ps rng
```

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	P
ORTS							
h481o4ojgpiti	rng.f9fzzgu5dge6wt3xrmad30br6	brahimhamdi/rng:latest	worker1	Running	Running less than a second ago		
r9rvs3zycwk2	rng.k8qert553j7o2d6vwm1hqrouu	brahimhamdi/rng:latest	worker2	Running	Running less than a second ago		
wzoq2hmlq7zz	rng.tgrnm51domfc5f64gt6tobvak	brahimhamdi/rng:latest	Manager	Running	Running 16 seconds ago		

```
vagrant@Manager:~$
vagrant@Manager:~$ docker service ls
```

ID	NAME	MODE	REPLICAS	IMAGE	PORTS
ugg5b7i17wfx	hasher	replicated	1/1	brahimhamdi/hashier:latest	*:8006-8010->80/tcp
nk1xxn10f6tp	redis	replicated	1/1	redis:latest	
hotxr79exxyv	rng	global	3/3	brahimhamdi/rng:latest	*:8001-8005->80/tcp
14f2qd4i3hst	webui	replicated	1/1	brahimhamdi/webui:latest	*:8000->80/tcp
afrr4orspumb	woker	replicated	10/10	brahimhamdi/woker:latest	

```
vagrant@Manager:~$
```

8. Remove all services of the application.

```
vagrant@Manager:~$ docker service rm woker rng hasher webui redis
woker
rng
hasher
webui
redis
vagrant@Manager:~$
```


Docker stack

9. Now we will combine docker swarm and docker compose to be able to deploy on the cluster using docker-compose.yml file. Check that all services running.

```
version: "3"
services:
  rng:
    image: brahimhamdi/rng
    ports:
      - "8001-8005:80"
  hasher:
    image: brahimhamdi/hasher
    ports:
      - "8006-8010:80"
  worker:
    image: brahimhamdi/worker
  webui:
    image: brahimhamdi/webui
    ports:
      - "8000:80"
  redis:
    image: redis
```

```
vagrant@Manager:~/dockercoins$ docker stack deploy --compose-file docker-compose.yml dockercoins
```

```
Creating network dockercoins_default
Creating service dockercoins_hasher
Creating service dockercoins_worker
Creating service dockercoins_webui
Creating service dockercoins_redis
Creating service dockercoins_rng
```

```
vagrant@Manager:~/dockercoins$ docker stack ls
```

```
NAME SERVICES
```

```
dockercoins 5
```

```
vagrant@Manager:~/dockercoins$ docker service ls
```

ID	NAME	MODE	REPLICAS	IMAGE	PORTS
tdrh87xyccgy	dockercoins_hasher	replicated	0/1	brahimhamdi/hasher:latest	*:8006-8010->80/tcp
wumllde9tu0p	dockercoins_redis	replicated	1/1	redis:latest	
m7dopj2ia6pw	dockercoins_rng	replicated	1/1	brahimhamdi/rng:latest	*:8001-8005->80/tcp
xrw7ib89y2l8	dockercoins_webui	replicated	1/1	brahimhamdi/webui:latest	*:8000->80/tcp
ui35mem8ucel	dockercoins_worker	replicated	1/1	brahimhamdi/worker:latest	

```
vagrant@Manager:~/dockercoins$
```

```
vagrant@Manager:~/dockercoins$ docker stack ps dockercoins
```

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
84ghbc4rwo1o	dockercoins_hasher.1	brahimhamdi/hasher:latest	worker1	Running	Preparing 53 seconds ago		
f0aasj3m41z0	dockercoins_redis.1	redis:latest	worker2	Running	Running 42 seconds ago		
plaza0y8489n	dockercoins_rng.1	brahimhamdi/rng:latest	Manager	Running	Running 33 seconds ago		
jzuqf5irm61	dockercoins_webui.1	brahimhamdi/webui:latest	Manager	Running	Running 44 seconds ago		
y9l0106ak1cp	dockercoins_worker.1	brahimhamdi/worker:latest	worker2	Running	Running 48 seconds ago		

```
vagrant@Manager:~/dockercoins$
```

```
vagrant@Manager:~$ docker stack services dockercoins
```

ID	NAME	MODE	REPLICAS	IMAGE	PORTS
tdrh87xyccgy	dockercoins_hasher	replicated	1/1	brahimhamdi/hasher:latest	*:8006-8010->80/tcp
wumllde9tu0p	dockercoins_redis	replicated	1/1	redis:latest	
m7dopj2ia6pw	dockercoins_rng	replicated	1/1	brahimhamdi/rng:latest	*:8001-8005->80/tcp
xrw7ib89y2l8	dockercoins_webui	replicated	1/1	brahimhamdi/webui:latest	*:8000->80/tcp
ui35mem8ucel	dockercoins_worker	replicated	1/1	brahimhamdi/worker:latest	

```
vagrant@Manager:~$
```

```
vagrant@Manager:~$
```

```
vagrant@Manager:~$ docker service ls
```

ID	NAME	MODE	REPLICAS	IMAGE	PORTS
tdrh87xyccgy	dockercoins_hasher	replicated	1/1	brahimhamdi/hasher:latest	*:8006-8010->80/tcp
wumllde9tu0p	dockercoins_redis	replicated	1/1	redis:latest	
m7dopj2ia6pw	dockercoins_rng	replicated	1/1	brahimhamdi/rng:latest	*:8001-8005->80/tcp
xrw7ib89y2l8	dockercoins_webui	replicated	1/1	brahimhamdi/webui:latest	*:8000->80/tcp
ui35mem8ucel	dockercoins_worker	replicated	1/1	brahimhamdi/worker:latest	

```
vagrant@Manager:~$
```

```
vagrant@Manager:~$
```

```
vagrant@Manager:~$
```

```
vagrant@Manager:~$ docker stack rm dockercoins
```

```
Removing service dockercoins_hasher
```

```
Removing service dockercoins_redis
```

```
Removing service dockercoins_rng
```

```
Removing service dockercoins_webui
```

```
Removing service dockercoins_worker
```

```
Removing network dockercoins_default
```