

TP6 – Docker Compose

Brahim Hamdi

Générer des bitcoins

Le minage c'est le procédé par lequel les transactions Bitcoin sont sécurisées. A cette fin les mineurs effectuent avec leur matériel informatique des calculs mathématiques pour le réseau Bitcoin. Comme récompense pour leurs services, ils collectent les bitcoins nouvellement créés ainsi que les frais des transactions qu'ils confirment.

Les mineurs sont en concurrence et leurs revenus sont proportionnels à la puissance de calcul déployée.

— minage de bitcoins, <https://bitcoin.fr/minage/>

Dockercoins

Notre application exemple est Dockercoins, une application de minage de Dockercoins!

Toutes les ressources de cette application (fictive et complètement inutile) sont disponibles sur le dépôt <https://github.com/brahimhamdi/dockercoins>.

Elle est composée de 4 microservices dans le dossier dockercoins (+une base de données redis):

- rng = un service web(Python) générant en sortie des nombres aléatoires
- hasher = un service web(Ruby) générant en sortie un hash des données qui lui sont envoyées par HTTP POST
- worker = processus(Python) utilisant rng et hasher
- webui = web interface (JS)

Principe :

- worker demande à rng de lui fournir des données aléatoires
- worker injecte ces données dans hasher , hasher génère un hash, récupéré par worker,
- Pour chaque hash commençant par 0, worker génère un DockerCoin
- Le worker stocke les DockerCoins générés dans une base de données Redis,
- La webui affiche le taux d'hashage par seconde.

Préparation de l'environnement

1. Installez la dernière version de Docker Compose sur la VM Manager:

```
vagrant@Manager:~$ sudo apt install docker-compose -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-certifi python3-docker python3-dockerpty python3-docopt python3-dotenv python3-jsschema
```

...

- Vérifiez ensuite que docker-compose est bien installé, et que vous avez la bonne version (>= 1.8):

```
vagrant@Manager:~$ docker-compose version
docker-compose version 1.29.2, build unknown
docker-py version: 5.0.3
CPython version: 3.10.12
OpenSSL version: OpenSSL 3.0.2 15 Mar 2022
vagrant@Manager:~$
```

2. Git clonez l'application *Dockercoins* sur votre hôte Docker:

```
vagrant@Manager:~$ git clone https://github.com/brahimhamdi/dockercoins
Cloning into 'dockercoins'...
remote: Enumerating objects: 218, done.
remote: Counting objects: 100% (218/218), done.
remote: Compressing objects: 100% (142/142), done.
remote: Total 218 (delta 86), reused 200 (delta 68), pack-reused 0 (from 0)
Receiving objects: 100% (218/218), 154.10 KiB | 595.00 KiB/s, done.
Resolving deltas: 100% (86/86), done.
vagrant@Manager:~$
vagrant@Manager:~$ ls dockercoins/
docker-compose.yaml  hasher  rng  webui  worker
vagrant@Manager:~$
```

- Sous le dossier *dockercoins*, créez le fichier *docker-compose.yml* et y ajoutez le contenu suivant :

```
version: "3"
services:
  worker:
    image: worker
    build: ./worker

  rng:
    image: rng
    build: ./rng

  hasher:
    image: hasher
    build: ./hasher

  webui:
    image: brahimhamdi/webui
    ports:
      - 8000:80

  redis:
    image: redis
```

Lancement de l'application

3. Lancez l'application avec la commande ***docker-compose up -d***. Cette commande va déployer l'application Dockercoins en appliquant le contenu de *docker-compose.yaml*

```
vagrant@Manager:~$ cd dockercoins/
vagrant@Manager:~/dockercoins$ ls
docker-compose.yaml  hasher  rng  webui  worker
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker-compose up -d
Creating network "dockercoins_default" with the default driver
Building worker
[+] Building 4.9s (5/9)
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 202B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:alpine
=> [auth] library/python:pull token for registry-1.docker.io
=> [1/4] FROM docker.io/library/python:alpine@sha256:c38ead8bcf521573dad837d7ecfdebbbc87792202e89953ba8b2b83a9c5a520b6
=> => resolve docker.io/library/python:alpine@sha256:c38ead8bcf521573dad837d7ecfdebbbc87792202e89953ba8b2b83a9c5a520b6
=> => sha256:c38ead8bcf521573dad837d7ecfdebbbc87792202e89953ba8b2b83a9c5a520b6 9.02kB / 9.02kB
```

...

4. Une fois l'application est déployée, affichez sur le navigateur de votre machine l'interface web de l'application en tapant l'URL <http://192.168.205.10:8000>



- Qu'affiche l'interface de l'application *Dockercoins* ?

5. Vérifiez avec deux commandes différentes que tous les services sont « up ».

```
vagrant@Manager:~/dockercoins$ docker-compose ps

```

Name	Command	State	Ports
dockercoins_hasher_1	ruby hasher.rb	Up (healthy)	80/tcp
dockercoins_redis_1	docker-entrypoint.sh redis ...	Up	6379/tcp
dockercoins_rng_1	python rng.py	Up	80/tcp
dockercoins_webui_1	node webui.js	Up	0.0.0.0:8000->80/tcp
dockercoins_worker_1	python worker.py	Up	

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

```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
4354f2de6a1c	brahimhamdi/webui	"node webui.js"	3 minutes ago	Up 3 minutes	0.0.0.0:8000->80/tcp	dockercoins_webui_1
0c0264033978	hasher	"ruby hasher.rb"	8 minutes ago	Up 7 minutes (healthy)	80/tcp	dockercoins_hasher_1
11aeb5697e90	worker	"python worker.py"	8 minutes ago	Up 7 minutes		dockercoins_worker_1
2f96326fc274	redis	"docker-entrypoint.s..."	8 minutes ago	Up 7 minutes	6379/tcp	dockercoins_redis_1
2fd312fed625	rng	"python rng.py"	8 minutes ago	Up 7 minutes	80/tcp	dockercoins_rng_1

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

6. Via quel réseau les conteneurs de l'application sont-ils interconnectés ?

```
vagrant@Manager:~/dockercoins$ docker network ls
NETWORK ID        NAME                DRIVER             SCOPE
27b944e52e95      bridge              bridge             local
144e48d8bf31      dockercoins_default bridge             local
4caafbe3fceb      host                host               local
b34fdabfc770      none                null               local
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker network inspect dockercoins_default | grep -A35 Containers
"Containers": {
  "0c026403397830865a1e7c8e1be9d4f19e54872714632670b463b78c7abc58d2": {
    "Name": "dockercoins_hasher_1",
    "EndpointID": "6d53130239325f786881d053541a3e41026421535b786962ca932998955775c1",
    "MacAddress": "02:42:ac:12:00:02",
    "IPv4Address": "172.18.0.2/16",
    "IPv6Address": ""
  },
  "11aeb5697e903332b74f54cab1e7964bf08f3ae76b8802857b2766fa792633e": {
    "Name": "dockercoins_worker_1",
    "EndpointID": "ad18cb89d555bf9b2fdb4dc022e35a3ba331d7b4fabaad449687ffabf6347872",
    "MacAddress": "02:42:ac:12:00:05",
    "IPv4Address": "172.18.0.5/16",
    "IPv6Address": ""
  },
  "2f96326fc274bd7a68f33f318d0cf216703694cd8d6415a4d31242debe43c8d7": {
    "Name": "dockercoins_redis_1",
    "EndpointID": "679abaa96e8f58f165315931d7cc206ad78be34852dbbb9e4d048b68448c0d92",
    "MacAddress": "02:42:ac:12:00:03",
    "IPv4Address": "172.18.0.3/16",
    "IPv6Address": ""
  },
  "2fd312fed6254aed9d66aec81ff9044456bd6ff60074e94da5910b3bfc70e8b3": {
    "Name": "dockercoins_rng_1",
    "EndpointID": "c4cf51474b73f0e37e43a020b11e456d621b8d0ccaa92a91c446db2cc5e44e19",
    "MacAddress": "02:42:ac:12:00:06",
    "IPv4Address": "172.18.0.6/16",
    "IPv6Address": ""
  },
  "4354f2de6a1c52964c2dcb103ce276a3ef6e8e5185c9534da76900e6b1919832": {
    "Name": "dockercoins_webui_1",
    "EndpointID": "a3380acble782efcd1973ad9bb196fc0fe1eb225e0aaabe2f2ac5f1f13a088c3",
    "MacAddress": "02:42:ac:12:00:04",
    "IPv4Address": "172.18.0.4/16",
    "IPv6Address": ""
  }
}
```

7. Vérifiez que les nouvelles images *worker*, *rng* et *hasher* sont bien créées.

```
vagrant@Manager:~/dockercoins$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
hasher               latest             d34d4ed4657b       17 minutes ago     311MB
rng                  latest             4f3fc45d4e04       19 minutes ago     58.2MB
worker               latest             569a3155f726       19 minutes ago     59.2MB
brahimhamdi/demoapp latest             c9e3d3dd9670       14 hours ago       102MB
demoapp              1.0-auto           c9e3d3dd9670       14 hours ago       102MB
demoapp              1.0-usuel           aa65e16535f9       15 hours ago       113MB
redis                alpine             e105cfd78d67       2 weeks ago        41.2MB
redis                latest             f02a7f566928       2 weeks ago        117MB
nginx                latest             3b25b682ea82       2 weeks ago        192MB
mariadb              latest             4b0711c6c639       6 weeks ago        407MB
brahimhamdi/webui    latest             e174d2bf3a61       3 years ago        218MB
brahimhamdi/moby-counter latest             b425a09c8044       4 years ago        80.2MB
openjdk              8-jre-alpine       f7a292bbb70c       5 years ago        84.9MB
```

- Taggez les nouvelles images et poussez-les vers DockerHub.

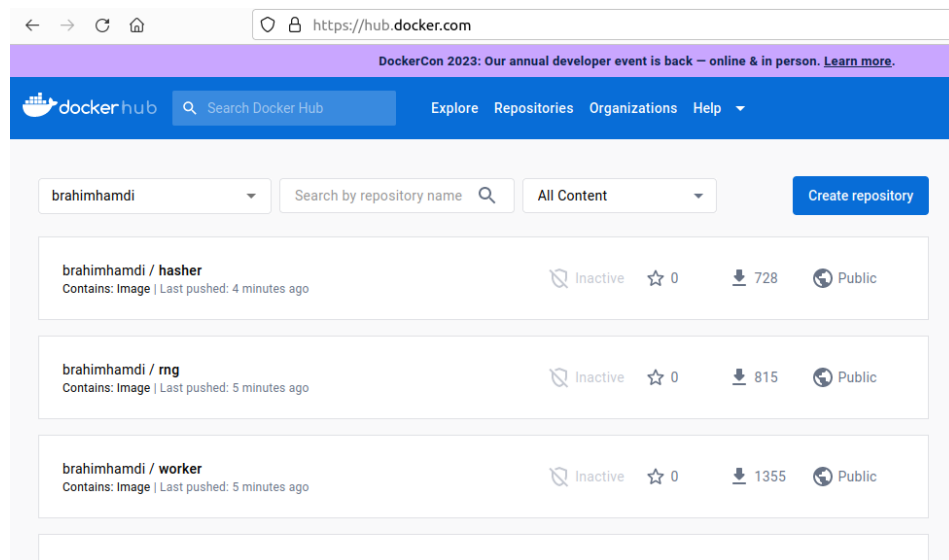
```
vagrant@Manager:~/dockercoins$ docker tag worker:latest brahimhamdi/worker:latest
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker tag hasher:latest brahimhamdi/hasher:latest
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker tag rng:latest brahimhamdi/rng:latest
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
brahimhamdi/hasher  latest             d34d4ed4657b       20 minutes ago     311MB
hasher              latest             d34d4ed4657b       20 minutes ago     311MB
brahimhamdi/rng     latest             4f3fc45d4e04       22 minutes ago     58.2MB
rng                 latest             4f3fc45d4e04       22 minutes ago     58.2MB
brahimhamdi/worker  latest             569a3155f726       23 minutes ago     59.2MB
worker              latest             569a3155f726       23 minutes ago     59.2MB
brahimhamdi/demoapp latest             c9e3d3dd9670       15 hours ago       102MB
```

```

vagrant@Manager:~/dockercoins$ docker push brahimhamdi/worker:latest
The push refers to repository [docker.io/brahimhamdi/worker]
1e42be9d1c21: Pushed
4be87a1c49b6: Pushed
885ee149bbc1: Pushed
b5cb171ffead: Mounted from library/python
16f484505bfb: Mounted from library/python
3f7827bfcbed: Mounted from library/python
63calfbb43ae: Mounted from library/ruby
latest: digest: sha256:3b9232b4d93896516746061f4539fa5d273e8e9e6d4e052c141ea35f9ae57335 size: 1786
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker push brahimhamdi/rng:latest
The push refers to repository [docker.io/brahimhamdi/rng]
9da2296f86a4: Pushed
4276f6184ea6: Pushed
b5cb171ffead: Mounted from brahimhamdi/worker
16f484505bfb: Mounted from brahimhamdi/worker
3f7827bfcbed: Mounted from brahimhamdi/worker
63calfbb43ae: Mounted from brahimhamdi/worker
latest: digest: sha256:2013b65ecfd21c1c546189f3b74bb3e6d16d84884a56db4a45453c85f6df014b size: 1575
vagrant@Manager:~/dockercoins$
vagrant@Manager:~/dockercoins$ docker push brahimhamdi/hashe:latest
The push refers to repository [docker.io/brahimhamdi/hashe]
f487529bb2d7: Pushed
411038d7eee1: Pushed
81876d1817d9: Pushed
89ad3541e6f2: Pushed
d3b1ea227d66: Mounted from library/ruby
9b29296c9fc2: Pushed
d88668927664: Pushed
28ed3d3e77a8: Pushed
63calfbb43ae: Mounted from brahimhamdi/rng
latest: digest: sha256:1532d241dc9ec7a30dd7fd88067215438a2d3ab780ca0bd9593f45950e35bc04 size: 2204
vagrant@Manager:~/dockercoins$

```

- Vérifiez sur *Dockerhub* que ces nouvelles images existent.



Passage à l'échelle

8. Nous voulons augmenter le nombre de *Dockercoins* générés par seconde, pour cela on vous demande de répliquer le service *worker* en augmentant le nombre de conteneurs à 2 (réplicas).

```
vagrant@Manager:~/dockercoins$ docker-compose scale worker=2
WARNING: The scale command is deprecated. Use the up command with the --scale flag instead.
Creating dockercoins_worker_2 ... done
vagrant@Manager:~/dockercoins$ docker-compose ps
```

Name	Command	State	Ports
dockercoins_hasher_1	ruby hasher.rb	Up (healthy)	80/tcp
dockercoins_redis_1	docker-entrypoint.sh redis ...	Up	6379/tcp
dockercoins_rng_1	python rng.py	Up	80/tcp
dockercoins_webui_1	node webui.js	Up	0.0.0.0:8000->80/tcp
dockercoins_worker_1	python worker.py	Up	
dockercoins_worker_2	python worker.py	Up	

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

- Vérifiez maintenant sur la courbe de l'application le taux de génération de dockercoins.



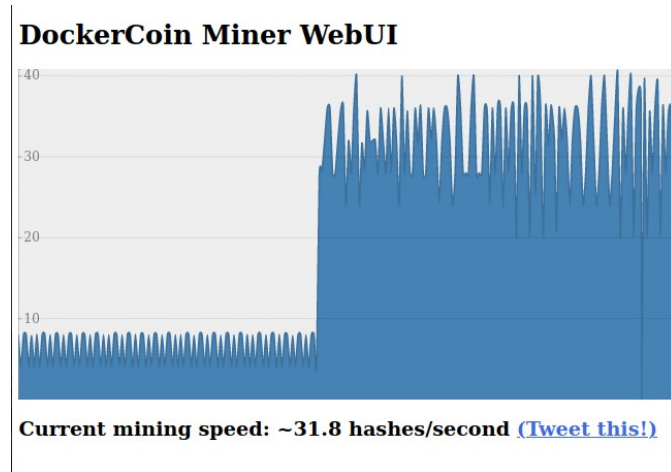
9. Répliquez une autre fois le service *worker* en augmentant le nombre de conteneurs à 10.

```
vagrant@Manager:~/dockercoins$ docker-compose scale worker=10
WARNING: The scale command is deprecated. Use the up command with the --scale flag instead.
Creating dockercoins_worker_3 ... done
Creating dockercoins_worker_4 ... done
Creating dockercoins_worker_5 ... done
Creating dockercoins_worker_6 ... done
Creating dockercoins_worker_7 ... done
Creating dockercoins_worker_8 ... done
Creating dockercoins_worker_9 ... done
Creating dockercoins_worker_10 ... done
vagrant@Manager:~/dockercoins$ docker-compose ps
```

Name	Command	State	Ports
dockercoins_hasher_1	ruby hasher.rb	Up (healthy)	80/tcp
dockercoins_redis_1	docker-entrypoint.sh redis ...	Up	6379/tcp
dockercoins_rng_1	python rng.py	Up	80/tcp
dockercoins_webui_1	node webui.js	Up	0.0.0.0:8000->80/tcp
dockercoins_worker_1	python worker.py	Up	
dockercoins_worker_10	python worker.py	Up	
dockercoins_worker_2	python worker.py	Up	
dockercoins_worker_3	python worker.py	Up	
dockercoins_worker_4	python worker.py	Up	
dockercoins_worker_5	python worker.py	Up	
dockercoins_worker_6	python worker.py	Up	
dockercoins_worker_7	python worker.py	Up	
dockercoins_worker_8	python worker.py	Up	
dockercoins_worker_9	python worker.py	Up	

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


- Quel est le nombre de DockerCoins généré par seconde ?



Arrêt de l'application

10. Arrêtez et supprimez tous les services de l'application en une seule commande (y compris la dizaine de conteneurs worker lancés).

```
vagrant@Manager:~/dockercoins$ docker-compose down
Stopping dockercoins_worker_3 ... done
Stopping dockercoins_worker_10 ... done
Stopping dockercoins_worker_5 ... done
Stopping dockercoins_worker_6 ... done
Stopping dockercoins_worker_9 ... done
Stopping dockercoins_worker_7 ... done
Stopping dockercoins_worker_4 ... done
Stopping dockercoins_worker_8 ... done
Stopping dockercoins_worker_2 ... done
Stopping dockercoins_webui_1 ... done
Stopping dockercoins_hasher_1 ... done
Stopping dockercoins_worker_1 ... done
Stopping dockercoins_redis_1 ... done
Stopping dockercoins_rng_1 ... done
Removing dockercoins_worker_3 ... done
Removing dockercoins_worker_10 ... done
Removing dockercoins_worker_5 ... done
Removing dockercoins_worker_6 ... done
Removing dockercoins_worker_9 ... done
Removing dockercoins_worker_7 ... done
Removing dockercoins_worker_4 ... done
Removing dockercoins_worker_8 ... done
Removing dockercoins_worker_2 ... done
Removing dockercoins_webui_1 ... done
Removing dockercoins_hasher_1 ... done
Removing dockercoins_worker_1 ... done
Removing dockercoins_redis_1 ... done
Removing dockercoins_rng_1 ... done
Removing network dockercoins_default
vagrant@Manager:~/dockercoins$
```