

TP VIRTUALISATION

Création du cluster swarm

```
docker swarm init
```

```
o22202871@o22202871-236:~$ docker node ls
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS	ENGINE VERSION
qsx32c8fzyjntrhgce3khvpai	o22202871-76	Ready	Active		27.3.1
ijdddpyb51e14ljn6zab1mbv5	o22202871-78	Ready	Active		27.3.1
vu9lgxbbmamia8tq4e64utzuu *	o22202871-236	Ready	Active	Leader	27.3.1

Les services

Nous avons besoin de créer les images pour les services à faire.

Site web de **Reprise Contact**

On commence par déployer l'image :

```
docker build -t utilisateurs_web_image:latest ./services/utilisateurs/
```

Ensuite, on crée le réseau traefik public :

```
docker network create --driver overlay traefik-public
```

Et on crée le service :

```
docker network create --driver overlay web-utilisateurs
```

Après cela, on peut déployer le service :

```
docker stack deploy -c docker-compose.yml traefik
```

Nous en faisons alors un curl ensuite pour vérifier que le service est bien en place :

```
o22202871@o22202871-236:~/tp_virtualisation$ curl -v http://utilisateurs.localhost
* Host utilisateurs.localhost:80 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:80...
* Connected to utilisateurs.localhost (::1) port 80
> GET / HTTP/1.1
> Host: utilisateurs.localhost
> User-Agent: curl/8.5.0
> Accept: */*
```

Site web de **Cluster avec Swarm**

On commence par déployer l'image :

```
docker build -t clusterswarm_web_image:latest ./services/clusterswarm/
```

On crée le service :

```
docker network create --driver overlay web-clusterswarm
```

Et on déploie le service à nouveau.

Voici le curl pour vérifier que le service est bien en place :

```
o22202871@o22202871-236:~/tp_virtualisation$ curl -v http://clusterswarm.localhost
* Host clusterswarm.localhost:80 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
* Trying [::1]:80...
* Connected to clusterswarm.localhost (::1) port 80
> GET / HTTP/1.1
> Host: clusterswarm.localhost
> User-Agent: curl/8.5.0
> Accept: */*
```

Application **Fortune Image**

On commence par déployer l'image :

```
docker build -t fortune_web_image:latest ./services/fortune/
```

On crée le service :

```
docker network create --driver overlay web-fortuneapp
```

Et on déploie le service à nouveau.

Voici le curl pour vérifier que le service est bien en place :

```
o22202871@o22202871-236:~/tp_virtualisation$ curl -v http://fortune.localhost
* Host fortune.localhost:80 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:80...
* Connected to fortune.localhost (::1) port 80
> GET / HTTP/1.1
> Host: fortune.localhost
> User-Agent: curl/8.5.0
> Accept: */*
```

Recherche perso

On commence par déployer l'image :

```
docker build -t apache_web_image:latest ./services/apache/
```

On crée le service :

```
docker network create --driver overlay web-apache
```

Et on déploie le service à nouveau.

Voici le curl pour vérifier que le service est bien en place :

```
o22202871@o22202871-236:~/tp_virtualisation$ curl -v http://localhost/apache
* Host localhost:80 was resolved.
* IPv6: ::1
* IPv4: 127.0.0.1
*   Trying [::1]:80...
* Connected to localhost (::1) port 80
> GET /apache HTTP/1.1
> Host: localhost
> User-Agent: curl/8.5.0
> Accept: */*
```