

## TD EDL

# Mémo Docker

## § Définitions

---

**image**

comme l'iso d'une distribution

**container**

comme une installation sur un ordi de l'iso, cette install qui vit sa vie possède un nom aléatoire par défaut : "[adjectif]\_[nom d'un scientifique ou hacker]"

**hôte**

système sur lequel tourne docker (i.e. notre machine)

## Basic

---

- `docker pull debian` récupérer l'image debian
- `docker run -h mon_container -i -t debian /bin/bash` lancer bash en mode interactif dans un container debian dont le host sera appelé mon\_container (le `-t` alloue un pseudo tty, nécessaire pour se comporter comme un shell)
- `docker run --name mon-mongo -v /my/own/datadir:/data/db -d mongo` lancer l'image mongo dans un conteneur nommé mon-mongo avec un répertoire particulier du système monté dans le conteneur dans `/data/db` en tâche de fond (`-d`)
- `docker run --name mynginx1 -p8080:80 -d nginx` on expose le port 80 du conteneur sur le port 8080 de l'hôte
- `docker ps` pour lister ce qui tourne, ex :

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
803657128e43	debian	"/bin/bash"	5 minutes ago
Up 5 minutes		condescending_curran	

- `docker ps -a` pour avoir aussi les containers arrêtés
- `docker inspect condescending_curran` tout savoir sur un conteneur
- `docker diff condescending_curran` permet de voir ce qu'on a changé dans l'exécution nommée condescending\_curran
- `docker logs condescending_curran` liste les commandes passées dans l'exécution condescending\_curran

- `docker stop condescending_curran` arrêter le conteneur `condescending_curran` (utile pour les BDD ou si on a pas la main sur le conteneur)
- `docker rm condescending_curran` supprimer le conteneur `condescending_curran`
- `docker rm -v $(docker ps -aq -f status=exited)` Supprimer les conteneur stoppés
- `docker images` Info sur les images en stock
- `docker rmi nginx` supprimer l'image `nginx`

## Conseils

---

- Docker tourne sur votre poste avec les droits de `root` .
- Les images docker tournent avec les droits de `root`
- Les volumes et points de montages sont accessibles par l'image docker avec le user `root`
- Docker peut modifier vos règles de firewall

Donc :

- Ne pas faire tourner docker par défaut, ne le lancer que lorsque vous en avez besoin et que votre firewall est bien configuré
- Sur le hub, ne se fier qu'aux images officielles
- Lire les dockerfiles des images officielles pour les contrôler, et préférer celles qui font tourner des applis avec un user différent de `root`
- Ne pas mapper le port d'un conteneur sur sa machine (option `-p 80:8080` par exemple), car Docker ouvrira le port en question de votre firewall. + Utiliser docker-compose en définissant un sous réseau particulier pour éviter d'ouvrir des ports à l'extérieur.

---

This page was built using the Antora default UI.

The source code for this UI is licensed under the terms of the MPL-2.0 license.