

Docker

L'essentiel

Les images et les conteneurs peuvent être désignés par leur ID ou par leur nom.

Gestion des images : docker image ...

<code>docker image ls (docker images)</code>	Liste des images disponibles en locale
<code>docker search httpd</code>	Chercher les images dans Docker Hub (registry)
<code>docker image pull debian</code>	Télécharger en local une image depuis un dépôt
<code>docker image push myImage</code>	Uploader une image locale vers un dépôt
FROM debian:latest RUN apt update \ && apt -y full-upgrade \ && apt -y install vim iproute2 WORKDIR /root	Fichier "Dockerfile" pour générer des images personnalisées - Créer un répertoire - Y créer le fichier Dockerfile - docker build...
<code>docker image build -t debian-network .</code>	Créer une image personnalisée à partir de Dockerfile
<code>docker image rm b3a422523a11 (ou nom-image)</code>	Supprimer une image, si aucun conteneur en dépend

Gestion des conteneurs : docker container ...

<code>docker container run --name myCT hello-world</code> <code>docker run -d -t --name myDebian debian</code> <code>-e LANG=C.UTF-8</code> <code>-p 8000:80</code> <code>-v ./conf:/.../rep</code> <code>--rm</code>	Si nécessaire télécharge l'image, crée le conteneur puis le démarre. Options : Démon, terminal TTY Set variable d'environnement Port local → port CT Répertoire local → répertoire CT Supprimer le CT à la fin de l'exécution
<code>docker container create -it --name myDebian debian</code>	Créer un conteneur ; mode interactif + TTY
<code>docker container exec -it myDebian bash</code>	Exécuter une commande, mode interactif dans un tty
<code>docker container ls (docker ps)</code>	Liste des conteneurs actif
<code>docker container ls -a (docker ps -a)</code>	Liste des conteneurs disponibles
<code>docker container start b941ffa9db3c</code> <code>docker container start myDebian</code>	Démarrer un conteneur avec son ID ou son nom restart = stop → start
<code>docker container stop myDebian</code>	Arrêter un conteneur
<code>docker container logs b941ffa9db3c</code>	Logs d'un conteneur by ID
<code>docker container rm b941ffa9db3c</code>	Supprimer un conteneur
<code>docker container stats</code>	Ressources (cpu, mem...) utilisées par les CT

Installation (Debian)

```
$ sudo apt-get install apt-transport-https ca-certificates curl gnupg2 software-properties-common
$ sudo curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o
/usr/share/keyrings/docker-archive-keyring.gpg
$ echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://download.docker.com/linux/debian bookworm stable" | sudo tee
/etc/apt/sources.list.d/docker.list
$ sudo apt update
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
$ sudo systemctl --now enable docker
$ sudo systemctl status docker
$ sudo adduser myLogin docker
```

Validation de l'installation

```
$ docker version
$ docker system info
$ docker run hello-world
```

Orchestration de conteneurs sur un même hôte avec Compose : docker compose...

Exemple avec l'image officiel httpd d'Apache ; Les configurations sont dans "/usr/local/apache2"

<pre>services: httpd: image: httpd:latest container_name: myHTTPd ports: - 8000:80 volumes: - ./html:/usr/local/apache2/htdocs working_dir: /root</pre>	<p>Créer un répertoire contenant le fichier "docker-compose.yml"</p> <p>Translation port hôte → port conteneur http://@hôte:8000</p> <p>Montage répertoire hôte → conteneur</p>
<pre>docker compose up -d</pre>	Créer les conteneurs
<pre>docker compose ps</pre>	Lister les conteneurs
<pre>docker compose stop</pre>	Arrêter les conteneurs

En savoir +

Nouvelles syntaxes de Docker CLI

Des alias existent entre les anciennes syntaxes et les nouvelles.

Ex. : `docker rmi my-image` = `docker image rm my-image`

Quelques commandes utiles

<pre>docker system df</pre>	Espace disque utilisé par les images et les conteneurs
<pre>docker cp name:rep/fichierCT fichierHôte</pre>	Copier fichiers entre hôte et CT
<pre>dpkg-reconfigure tzdata</pre>	Reconfigurer la zone datetime (dans le conteneur)

Installation scriptée : Ce script Docker détecte la distribution Linux pour installer sa dernière version.

```
$ curl -ssl https://get.docker.com/ | sh
$ sudo adduser myLogin docker
$ newgrp docker
```

Accès du client docker par le réseau TCP/IP

```
$ sudo cp /usr/lib/systemd/system/docker.service /etc/systemd/system
ExecStart=/usr/bin/dockerd -H fd:// -H tcp://0.0.0.0:2375
--containerd=/run/containerd/containerd.sock
$ sudo systemctl daemon-reload
$ sudo systemctl restart docker.service
```