

INSTITUE DE FORMATION AUX NORMES ET TECHNOLOGIE DE L'INFORMATIQUE (IFNTI)

on

Simone ADJANAYO et Malik KONDI

11 janvier 2023

1 Docker

Docker est une plateforme de conteneurisation, d'hébergement d'application. Nous avons plusieurs version de docker dont

- Docker Community edition (linux seulement)
- Docker Desktop (Mac ou Windows)
- Docker Enterprise (Linux seulement)

Docker Desktop et Docker Community Edition (CE) sont deux versions de Docker gratuites. Avec les deux solutions, vous aurez un Docker fonctionnel sur votre ordinateur.

Comme tout application, il faut l'installer avant l'usage. Pour installer docker on fait : **sudo apt-get install docker**

2 Image

Une image en docker est un modèle en lecture seul utilisé pour créer des conteneurs docker.

2.1 dockerfile

Un Dockerfile définit une image Docker qui est constituée de couches empilées les unes sur les autres. c'est elle qui nous permettra de créer les images que nous souhaitons. pour effectuer la création de l'image on crée un fichier Dockerfile NB : le fichier doit être forcément nommé dockerfile

Voici un exemple de créer de l'image

```
#syntax=docker/dockerfile:1
FROM python:3.7-alpine
WORKDIR /code
ENV FLASK_APP=app.py
ENV FLASK_RUN_HOST=0.0.0.0
RUN apk add --no-cache gcc musl-dev linux-headers
COPY requirements.txt requirements.txt
RUN pip install -r requirements.txt
EXPOSE 5000
COPY . .
CMD ["flask", "run"]
```

Ici nous cherchons à installer python de version 3.7-alpine sur lequel il va tourner une petite application de flask sur le port 5000.

Voir la documentation de dockerFile pour plus d'explication : **documentation**

Pour construire l'image on fait : **docker build -t name-image** le name-image c'est le nom que nous voulons donner à notre image.

Pour l'exécuter on fait : **docker run nom-image**

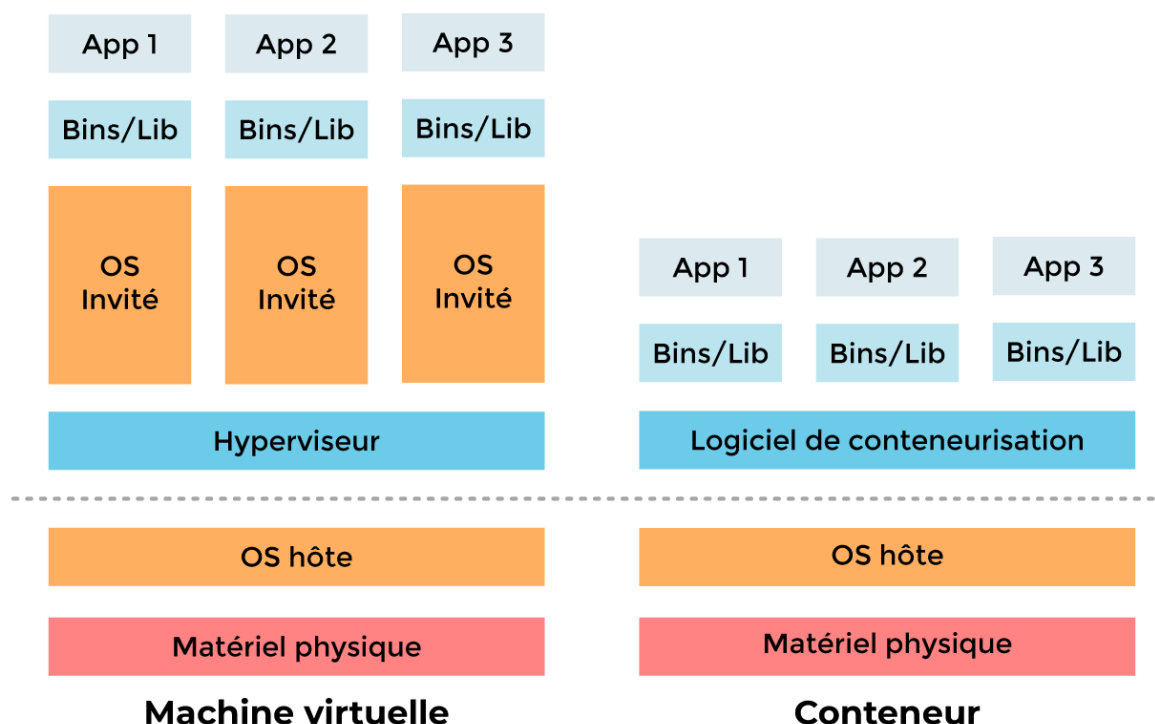
Pour lister les images que nous avons créées on tape : **docker image ls**

Pour inspecter l'image on fait **Docker image inspect <image-id>**

Nous pouvons aussi télécharger une image sans le spécifier dans le docker file on fait : **docker pull <nom-image>** exemple **docker pull ubuntu**. Pour préciser la version on a juste à faire : **docker pull ubuntu :16.04**.

3 Container

Un container docker est en quelque sort comme une machine virtuelle qui nous permettra de faire des hébergement des applications et autre. Mais il y a une différence entre ces deux concepts. Un conteneur a une virtualisation légère, il ne crée qu'une isolation des processus, le conteneur partage donc les ressources avec le système hôte. alors qu'une machine virtuelle est totalement isolée du système hôte, elle a une virtualisation lourde.



Passons maintenant à la création d'un conteneur.

Avant de pouvoir créer un conteneur il faut d'abord avoir l'image, dès que nous sommes rassurés que nous avons l'image avec laquelle nous voulons créer notre conteneur on a juste à effectuer cette commande pour la créer : **docker run -itd --name nom-conteneur nginx :latest**.

Pour avoir la liste des conteneurs que nous avons créés : **docker container ps**.

Pour inspecter le conteneur on fait **docker inspect <nom-conteneur>**, cette commande va nous retourner l'adresse IPv4 de notre conteneur avec laquelle on pourra avoir accès au conteneur.

Nous n'avons plus besoin d'un conteneur comment allons-nous faire pour l'enlever de la liste des conteneurs, on a juste à faire **docker rm <nom-docker>** pour le supprimer.

Pour accéder au conteneur que nous avons créé on fait : **docker exec -ti mon-conteneur bash**

4 volume

Les volumes sont le mécanisme préféré pour conserver les données générées et utilisées par les conteneurs Docker. Pour créer un volume on fait : **docker volume create nom-volume** si nous voulons voir la liste de nos volumes on tape **docker volume ls**. Au cas où nous voulons inspecter(examiner) un volume quelconque on écrit **docker volume inspect nom-volume**. Lorsque nous n'avons plus besoin d'un volume et que nous voulons le supprimer on fait **docker volume rm nom-volume**.

Lorsque nous allons effectu  cette commande elle va nous renvoyer une erreur si le centenaire avec lequel on utilise ce volume est actif. Pour r soudre ou  viter ces il faut d'abord stopper en faisant **docker container stop nom-conteneur**, le fonctionnement du conteneur ensuite supprimer le conteneur **docker container rm mon-conteneur** puis maintenant on peut pass  au volume.

Maintenant que nous avons notre volume pour la persistance de nos donn , nous pouvons activ  notre conteneur comme si :**docker run -itd -v /home/simone/Documents/L3/UE-Libre/public/ :/usr/share/nginx --name test nginx :latest**, **itd** signifie runner en mode interactif et d tacher et **-v** le volume

5 Networking

La mise en r seau Docker vous permet d'attacher un conteneur   autant de r seaux que nous le souhaitons. Nous pouvons  galement joindre un conteneur d j  en cours d'ex cution. pour connecter les conteneur et les service docker entre eux,ou es connecter   des charges de travail non docker.

6 Security

La s curit  de docker es relative aux menaces et attaques qui remettraient en question la s curit  informatiques bas e sur les services de confidentialit .

7 Docker Hub