

Fiche de procédure Docker

1. L'installation de Docker

Pour installer Docker il faut prendre un invité de commande et mettre dedans :

- apt-get update
apt-get install ca-certificates
curl gnupg install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/debian/gpg | gpg --dearmor -o
/etc/apt/keyrings/docker.gpg chmod
a+r /etc/apt/keyrings/docker.gpg echo \
"deb [arch="\$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/debian \
"\$(. /etc/os-release && echo "\$VERSION_CODENAME)" stable" | \
tee /etc/apt/sources.list.d/docker.list > /dev/null apt-get update

puis faire :

- apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
dockercompose-plugin
et

pour finir :

- docker run hello-world

Si l'installation a fonctionné un message s'affiche pour dire que l'installation s'est bien déroulée.

2. Découverte des commandes de base de Docker

Récupération d'une image :

Pour récupérer une image faire :

- docker pull [nom_image] exemple : docker pull nginx

Affichage liste des images :

Pour afficher les images dont celle que vous venez de récupérer faire :

- docker image ls

Lancement d'un conteneur :

Pour lancer un conteneur :

- docker run -p [port_ordinateur]:[port_interne_du_conteneur] [nom_image]

Affichage conteneur sur internet :

Pour afficher le conteneur mettre dans une barre de recherche internet l'adresse ip de l'ordinateur utilisé avec le port de l'ordinateur.

Affichage Liste des conteneurs :

Pour afficher la liste des conteneurs faire :

- docker container ls

Vérification status conteneur :

Pour vérifier si le conteneur est en cours de fonctionnement ou en arrêt faire : -
docker ps

Si le status est en Up cela veut dire que c'est en cours de fonctionnement.

Stopper un conteneur :

Pour stopper un conteneur faire :

- docker stop [nom_conteneur] exemple : docker stop beautiful_sammet

Si le conteneur est bien stoppé normalement le nom du conteneur s'affiche.

Relancer un conteneur :

Pour relancer un conteneur faire :

- docker restart [nom_conteneur_stoppé] exemple : docker restart beautiful_sammet
- Si le conteneur est bien relancé normalement le nom du conteneur s'affiche.

Supprimer un conteneur :

Pour supprimer définitivement un conteneur faire :

- docker rm -f [nom_conteneur] exemple : docker rm -f beautiful_sammet
- Si le conteneur est bien supprimé normalement le nom du conteneur s'affiche.

Supprimer une image :

docker image rm -f [nom_image] exemple : docker image rm -f nginx

Si l'image est supprimé l'invité de commande affiche cela :

```

root@debian-console:~# docker image rm -f nginx
Untagged: nginx:latest
Untagged: nginx@sha256:275c11c8996bacda0ca35e12f42e9696e80032a2b9926b79f0ac58fdecf5de2f
Deleted: sha256:f5a6b296b8a29b4e3d89ffa99e4a86309874ae400e82b3d3993f84e1e3bb0eb9

```

3. Création de vos propres images Docker

Création d'une image :

Pour créer une image il faut :

- créer un dossier exemple : mkdir mon_projet_docker
- se déplacer dans le dossier exemple : cd mon_projet_docker
- créer un fichier dockerfile exemple : touch Dockerfile
- mettre dans le fichier les infos suivantes : exemple : nano Dockerfile
 - # Utilisez Debian comme image de base
 - FROM debian:latest **si pas utilisation de debian remplacé le debian par votre image de base**
 - # Commande qui sera exécutée lorsque le conteneur sera démarré
 - CMD ["echo", "Hello World!"]
- créer l'image docker build -t [nom_image] - < Dockerfile
- lancer l'image docker run [nom_image]

4. Automatisation du déploiement de vos conteneurs

Création fichier de configuration docker compose :

Pour créer un fichier de config docker compose faire :

- créer un fichier exemple : touch docker-compose.yml
- mettre dans ce fichier cela : exemple : nano docker-compose.yml

- version: '3' - services: my-nginx:
image:nginx
ports:
- "[port_ordinateur]:[port_interne_du_conteneur]"

Récupération logs de console du conteneur :

Pour récupérer les logs de console du conteneur faire :

- docker compose logs -f [nom_image]