

Module 2.2 – Manipulation de Docker (TP)

En utilisant Docker cli et en vous appuyant sur la documentation officielle, réaliser les exercices suivants :

Exercice 1 : Télécharger (docker pull) plusieurs images :

- ubuntu
- nginx
- debian
- httpd
- mysql

Exercice 2 : Exécuter plusieurs conteneurs (docker run) :

- 2.1 Exécuter un conteneur basé sur l'image « ubuntu », qui va réaliser un echo de « Mon premier conteneur ».
- 2.2 Exécuter un conteneur basé sur l'image « ubuntu », qui va effectuer une commande « ls »
- 2.3 Exécuter un conteneur basé sur l'image « ubuntu », qui va effectuer une commande ls sur le répertoire /var/
- 2.4 Exécuter un conteneur ubuntu et s'attacher au shell de ce conteneur (utiliser l'option -it et la commande bin/bash)
- 2.5 Exécuter un conteneur nginx qui écoute sur le port 8080 de l'host, le nommer « my-nginx » (utilisez les options -p pour préciser le mapping de port et -d pour le détacher).
 - Lancez votre navigateur sur l'adresse <http://localhost:8080> et vérifiez que vous avez bien un retour de nginx
- 2.6 Attachez vous sur le conteneur « my-nginx » et éditer le fichier index.html dans /usr/share/nginx/html.
 - Vous pouvez mettre un contenu html de votre choix à l'intérieur du fichier index.html.

- Lancez votre navigateur sur l'adresse <http://localhost:8080> et vérifiez que le contenu a bien changé.

Exercice 3 : Gestion des conteneurs (lister, stopper, reprendre, supprimer, créer une image et inspecter)

- 3.1 Lister les conteneurs actifs du système
- 3.2 Lister tous les conteneurs du système
- 3.3 Stopper le conteneur actif « my-nginx »
 - Relancer votre navigateur sur <http://localhost:8080> → vérifier que vous avez bien une erreur car le conteneur est maintenant arrêté.
- 3.4 Relancer le conteneur « my-nginx »
 - Normalement <http://localhost:8080> est à nouveau accessible et avec les modifications apportés (cf 2.6)
- 3.5 Inspecter le conteneur « my-nginx » (docker inspect)
 - cela permet d'analyser des informations sur le conteneur
- 3.6 Créer une image du conteneur my-nginx (docker commit)
- 3.7 Vérifier que la nouvelle image est créée sur votre système (docker images)
- 3.8 Stopper et supprimer le conteneur « my-nginx »
- 3.9 Relancer la commande de l'exercice 2.5 et vous devriez constater que vos modifications sont perdues, ensuite stopper et supprimer à nouveau le conteneur.

- 3.10 Relancer la commande de l'exercice 2.5 en prenant comme image de base celle qui a été créée dans l'exercice 3.6.
- 3.11 En utilisant la commande docker history, vérifier la différence entre les images nginx et celle créée dans l'exercice 3.6. → vous devriez constater un commit supplémentaire qui représente votre différence.
- 3.12 Supprimer tous les conteneurs inactifs du système
- 3.13 Nettoyer le système en purgeant les éléments non utilisés (docker system prune)