

# TP ISEN Docker 2019

## Objectif :

- Revoir toutes les notions Docker :
  - Volumes/Réseau
  - Interaction
  - Construction d'une image
  - Docker Compose

## Exercice 1 :

- Démarrer un shell (bash) dans un container à partir d'une image centos
- Installer wget et netcat dans le conteneur
- Lancer un autre container en partant de la même image de base
- Expliquer pourquoi les logiciels wget et nc ne sont pas présents dans ce container
- Sortir du container
- Lister les containers créés précédemment
- Renommer le premier container créé (celui dans lequel wget et nc ont été installés) en centos-shell-01
- Redémarrer le container centos-shell-01 et vérifier que wget et nc sont présents
- Sortir du container

## Exercice 2 :

- Démarrer un container à partir de l'image nginx:latest
  - en mode daemon
  - en exposant le port 80 du container sur le port 9888 du host
- Vérifier que le nginx est accessible depuis le host
- Se connecter dans le container et vérifier les logs dans /var/log/nginx
- Sortir du container et le supprimer

## Exercice 3 :

- Vérifier que l'image nginx:latest expose le port 80
- Créer un répertoire public\_html
- Créer un fichier index.html avec votre nom et prénom dans le repertoire public\_html et écrire un message gentil pour le prof :)
- Lancer un container à partir de l'image nginx:latest
  - en mode daemon
  - nommer le container isenweb
  - monter le repertoire public\_html à l'emplacement /usr/share/nginx/html dans le container
  - exposer le port 80 du container en laissant docker choisir le port sur le host
- Vérifier que la page index.html accessible est bien celle créée précédemment
- Modifier le fichier index.html depuis l'extérieur du container et vérifier que la modification est bien prise en compte
- Stopper le container superisenweb

#### Exercice 4 :

- Créer un volume nginx-logs
- Démarrer un container à partir de l'image superisenweb
  - en mode daemon
  - nommer le container mynginx
  - exposer le port 80 du container sur le port 7999 du host
  - monter le volume nginx-logs sur le répertoire /var/log/nginx
- Lister les fichiers stockés dans le volume nginx-logs depuis le host
- Démarrer un container nommé log-tailer à partir de l'image alpine:
  - qui monte les mêmes volumes que le container my-nginx
  - qui exécute la commande tail -f /var/log/nginx/access.log
- Accéder à la page index.html dans un browser et vérifier que le container
- Sortir du container log-tailer
- Supprimer le container log-tailer

#### Exercice 5 :

Le but de cet exercice est de créer une image log-tailer qui fonctionne comme le container créé précédemment.

- Partir de l'image alpine:latest
- Positionner un label ayant pour nom maintainers et comme valeur votre adresse
- Ajouter le fichier Dockerfile dans l'image dans le répertoire /etc/docker-image-metadata
- Positionner l'entrypoint à la commande tail -f
- Faire en sorte que par défaut (i.e. quand le container à partir de cette image sera lancé) la commande lancée sera tail -f /var/log/nginx/access.log
- Builder l'image en la taggant my-log-tailer
- Vérifier que l'image est bien présente
- Comme dans l'exercice précédent, démarrer un container appelé log-tailer à partir de cette image qui monte les mêmes volumes que le container my-nginx (qui doit toujours tourner)
- Accéder au nginx dans un navigateur et vérifier dans les logs du container log-tailer que les logs sont bien visibles
- Arrêter et supprimer les containers my-nginx et log-tailer

#### Exercice 6 :

Le but de cet exercice est de démarrer les 2 containers précédents en passant par docker-compose.

- Créer le fichier docker-compose
- Déclarer un service my-nginx:
  - qui utilise l'image nginx:latest
  - qui expose le port 80 du container sur le port 8000 du host
  - qui utilise un volume appelé nginx-logs monté dans /var/log/nginx

- Déclarer un service log-tailer:
  - qui utilise l'image, créée précédemment, log-tailer
  - qui monte le même volume que le service my-nginx
  - qui *suit* le fichier de logs /var/log/nginx/access.log
- Démarrer les 2 services
- Accéder à <http://localhost:8000> dans un navigateur
- Vérifier dans les logs du service log-tailer que les accès au nginx sont bien tracés

Exercice Bonus : (+1 sur la moyenne)

Faire un projet sous Circle Ci en et pousser l'image de l'exercice 1 sur Docker Hub avec un logiciel d'intégration continue.