Département Informatique
RA-4.08 Virtualisation – TP Docker
Responsable : X.Roirand + M. Le Lain
Durée : 180 mn (sur machine)

Introduction Docker (sur 2 séances)

Le but de ce TP va être d'utiliser une technique de containérisation appelée Docker et très répandu dans le cloud ou dans le déploiement d'applications de type cloud.

Attention : certaines commandes comme le démarrage d'un container peuvent prendre parfois jusqu'a 30 secondes, voir un peu plus, soyez patient :)

1) Connection sur la machine distante ou on doit travailler

Pour se connecter sur la machine distante, il faut utiliser une connexion ssh. C'est une connexion sécurisée qui permet de ne pas faire passer le mot de passe en clair sur le réseau.

Vous pouvez utiliser le programme « putty » sur windows ou « ssh » sur linux.

La connexion se fait sur une machine dont l'adresse IP est 195.83.161.135. Pour que chaque étudiant ait sa propre machine, les ports ssh de connexion diffèrent. Lors de la mise en place du TP, l'enseignant vous a donné un numéro. Ce numéro est important car il va être utilisé pour le port ssh de connexion et le mot de passe.

Pour résumer:

IP: 195.83.161.135
port: 122<votre numéro sur 2 digits>
login: student
mot de passe est: ?<votre_numéro>Student_56

Ex:

- si votre numéro est 3 alors le port sera 12203 et le mot de passe ?3Student_56
- si votre numéro est 86 alors le port sera 12286 et le mot de passe ?86Student_56

Une fois sur votre machine distante, vous pouvez l'administrer totalement. Vous devez absolument changer le mot de passe par défaut le plus rapidement possible.

Pour pouvoir noter le TP, il faut que vous mettiez sur la machine mis à disposition, dans un fichier nommé /root/info.txt votre nom et prénom. Attention, pas de fichier = zéro !

Tout ce qui est noté sur fond bleu … (xxx réponse à fournir)…correspond à une question à laquelle vous devez répondre dans le document de rendu.

2) Qu'est ce que Docker ?

Crééz un document qui va contenir les réponses que vous allez donner au fur et à mesure de l'avancement dans le TP (ce document sera à rendre dans la zone de rendu sur Moodle).

A la fin de la 1ère séance, vous devez rendre ce document dans la zone de rendu « Rendu séance 1 », même si vous n'avez pas fini le TP, normal puisqu'il est sur 2 séances.

A ma fin de la 2ème séance, vous devez rendre ce document dans la zone de rendu « Rendu séance 2 ».

La première question à laquelle je vous demander de répondre est « Qu'est ce que Docker » ? Idéalement je voudrais une réponse qui montre que vous avez fait des recherches et qui m'indique aussi pourquoi c'est différent de VMWare, VirtualBox ou KVM par exemple ? Aussi sur quels OS ca tourne, les limitations et en terme de sécurité est-ce que c'est bien ou pas, etc…mes indications ici ne sont pas exhaustives.
Ce qui m'intéresse aussi c'est ce que vous en avez compris, je ferais une recherche sur les phrases que vous indiquerez dans votre réponse, si c'est un copié / collé ca sera considéré comme pas de réponse.
Le temps estimé de travail pour cette partie est d'environ 15mn. (1ère réponse à fournir)

3) Installation de docker :

Vous avez une machine ubuntu 18.04 qui vous est assignée et vous allez devoir installer docker dessus et vérifier qu'ensuite docker fonctionne bien, puis indiquez ce que vous avez fait pour que tout fonctionne dans le document de rendu (2ème réponse à fournir) et comment vous avez fait pour vérifier que docker fonctionne bien sur la machine (3ème réponse à fournir).

Pour cela cherchez sur internet mais attention, tous les packages docker ne fonctionnent pas forcément sur cette distribution là, il faut choisir le bon, donc peut-être parfois désinstaller le mauvais et réinstaller le bon (tip : suivez les préconisations d'installation de Docker sur Ubuntu).

4) Mise en place des droits pour votre utilisateur :

Pour pouvoir utiliser docker sur la machine qui vous est assignée, il faut qu'il y ait un daemon docker qui tourne, afin de…afin de quoi ? Il sert à quoi ce daemon ? (4ème réponse à fournir).

Ce daemon docker tourne toujours en tant que "root". Donc si vous voulez pouvoir faire des commandes plus évoluées, vous serez vite confronté a un problème de droit car vos commandes demanderont une élévation de privilège (sudo). Pour éviter cela, et demander en permanence une élévation de privilège faite……Trouvez la bonne commande qui résoudra ce problème (5ème réponse à fournir).

5) Vérification du bon fonctionnement du daemon docker:

Trouvez la commande systeme qui permet d'afficher l'état du daemon docker et copiez/coller la commande et la sortie écran correspondante (6ème réponse à fournir).

Trouvez la commande système qui permet de stopper le daemon docker et copiez/coller la commande et la sortie écran correspondante (7ème réponse à fournir).

Trouvez la commande système qui permet de redémarrer le daemon docker et copiez/coller la

commande et la sortie écran correspondante (8ème réponse à fournir).

6) Recherche de container/images :

Maintenant que tout est bien installé, vous allez cherchez les différents images Docker des distributions suivantes :

Fedora
CentOS
OpenSuse

On vous demande juste de chercher, pas d'installer.

Indiquez dans le document de rendu la commande utilisée et l'output obtenu (9ème réponse à fournir).

Trouvez à quel endroit ces images sont stockées sur votre système de fichier avec la commande locate (elle n'est peut-être pas installée…).
ATTENTION : les répertoires dans lesquels sont stockés les images étant accessibles uniquement à certais utilisateurs, il faut faire la commande préfixé de « sudo ». D'ailleurs à quoi sert cette commande (10ème réponse à fournir) ? Indiquez dans le document de rendu les commandes utilisées et l'output obtenu (11ème réponse à fournir) pour trouver l'endroit ou les images sont stockées.

7) Manipulation de container:

Au départ aucune image n'existe en local sur votre machine, il faut donc en télécharger une et l'exécuter. Prenez la distribution « alpine » et trouvez la bonne commande pour lancer l'instance docker correspondante, et constatez que lorsque vous lancez le container, sans paramètre supplémentaire, il vous rend la main rapidement. Mettez cette commande et la sortie écran produite lorsque vous avez rentré cette commande dans le document de rendu (12ème réponse à fournir).

En effet, une image Docker est juste un environnement dans lequel il va exécuter la ou les commandes que vous lui donnez, si vous n'en donnez aucune, alors il lance l'image, puis va finir l'exécution, puisqu'il n'y a rien à exécuter. En général les images ont un point d'entrée (de démarrage) mais pour les images simples, basées sur des distributions, ca n'est pas le cas.

Lancer une image docker CentOS sans rien exécuter dans le container docker. Mettez cette commande et la sortie écran produite lorsque vous avez rentré cette commande dans le document de rendu (13ème réponse à fournir).

On va faire la même chose mais la commande à exécuter va être un shell interactif, ce qui permettra, d'être en live dans l'image docker et de faire des choses. Pour lancer un docker avec un shell de manière interactive, trouvez une commande unique (pas avec 2 commandes, et il faut mettre des options) qui permet de le faire. L'image à exécuter est une image de la distribution CentOS (c'est moi qui veut ça).

Mettez cette commande et la sortie écran produite lorsque vous avez rentré cette commande dans le document de rendu (14ème réponse à fournir).

On va tester que la partie interactive fonctionne bien et qu'on arrive bien à utiliser des fonctionnalités qui ne sont pas sur la machine qu'on vous met à disposition mais que les dockers peuvent bien apporter cette fonctionnalité.

Pour chaque élement ci-dessous, vérifiez que votre machine en fournit pas la fonctionnalité/language mais qu'un docker peut le faire. Affichez l'output de la vérification/commande sur votre machine et pareil dans le docker. (15ème réponse à fournir).

Ex avec Java :

Sur la machine : java -help

Command 'java' not found, but can be installed with:

apt install default-jre

apt install openjdk-11-jre-headless

apt install openjdk-8-jre-headless

Avec le bon docker en interactif :

```
root@05e34a9c317d:/# java -h
Usage: java [-options] class [args...]
           (to execute a class)
   or  java [-options] -jar jarfile [args...]
           (to execute a jar file)
where options include:
    -cp <class search path of directories and zip/jar files>
```

[…]

Voici les éléments :

Python3

Nodejs

Tcl

PHP

Lua