

Installation de Docker et Docker-compose sous Windows 10 Pro

N.B. : Pour tous les logiciels à installer ici, vous devez le faire directement sur votre ordinateur et non dans une machine virtuelle. Vous devez avoir Windows Professionnel (7 ou 10). Vous devez désactiver votre parefeu et désactiver votre antivirus.

Deux approches s'offrent à vous. Vous en choisissez une ou l'autre. La première approche peut ne pas fonctionner. Dans ce cas, essayez la deuxième approche.

Approche 1

- Installez Virtualbox 6.0 (version antérieure qu'on [trouve ici](#)).
- Assurez-vous de fermer votre antivirus et parefeu.
- Téléchargez et installez **Docker Toolbox** disponible [ici](#).
- Si la commande n'est pas reconnue sous Powershell, il faut modifier la variable PATH sous Windows.
- Voici les commandes à exécuter dans Powershell pour finaliser l'installation. Les commandes vont créer une machine virtuelle dans Virtualbox qui contiendra Docker:

```
docker-machine create --driver virtualbox box
docker-machine env --shell powershell box
docker-machine env --shell powershell box | Invoke-Expression
```

- On peut vérifier si la machine virtuelle est bien installée par la commande suivante :

```
docker-machine ls
```

- On peut vérifier si la machine virtuelle est active:

```
docker-machine status box
```

- Si la machine est arrêtée, on tape:

```
docker-machine start box
```

- Après votre travail avec Docker, quand on a terminé de faire fonctionner Docker, on tape:

```
docker-machine stop box
```

- Pour connaître l'adresse IP du serveur Docker, on tape:

```
docker-machine ip box
```

L'adresse IP servira à accéder à un serveur installé dans un conteneur Docker comme par exemple, <http://192.168.99.102>

- Vérifiez sur Docker fonctionne bien en tapant par exemple :

```
docker run hello-world
```

Cela ne devrait pas générer d'erreur. Si c'est le cas, il faut recommencer.

- Si votre machine Docker ne fonctionne plus bien, on tape ce qui suit. Ensuite, il faut reprendre les étapes ci-haut pour recréer la machine Docker sous Virtualbox.

```
docker-machine kill box
docker-machine rm box
```

Si la reprise des étapes précédentes ne fonctionnent pas encore, il vous reste à essayer l'approche 2 ci-bas.

Si tout fonctionne, vous aurez accès **docker** et **docker-compose** sous Powershell.

Approche 2 pour Docker

- Installez Virtualbox 6.0 (version antérieure qu'on [trouve ici](#)). Ne pas installer la version 6.1.
- Installez Vagrant [disponible ici](#).
- Ouvrez une console Powershell et vérifiez que la commande **vagrant** est disponible.
- Créez le dossier **c:\bionic** et rendez-vous dans ce dossier via une console Powershell.
- Tapez les commandes suivantes :

```
vagrant init demers/bionic64-dockercompose
vagrant up
```

- La première commande crée un fichier **Vagrantfile** automatiquement dans le dossier courant.
- Cette deuxième commande démarre une machine virtuelle dans Virtualbox.

- Si vous recevez plusieurs fois le message d'erreur suivant :

`default: Warning: Authentication failure. Retrying...`

vous devrez arrêter la processus en tapant les touches CTRL-C (ou attendre pendant quelques minutes). Ensuite, tapez la touche ENTRER.

- Ne fermez pas la console Powershell. Ouvrez l'application Virtualbox. Vous verrez qu'une machine virtuelle est en exécution (en fonction). Cliquez sur cette machine.
- Tapez le bouton « **Afficher** ». Vous verrez un « Login ». Vous entrez « **vagrant** ».
- Le mot de passe est aussi « **vagrant** ».

- Par la suite, tapez la commande suivante (dans la console de la machine virtuelle) :

```
cp -f authorized_keys .ssh
```

- Aucun message de réponse ne devrait apparaître (à part l'invite Linux).
- Retournez dans la console **Powershell** sans arrêter la machine virtuelle et tapez :

```
vagrant halt
```

- La machine virtuelle va s'arrêter. Ensuite, tapez la commande suivante :

```
vagrant reload
```

- Il ne devrait plus y avoir d'erreurs. Sinon, communiquez avec votre professeur.
- Vous pouvez maintenant accéder à la machine virtuelle par la commande :

```
vagrant ssh
```

- Vous serez alors connecter sur votre machine virtuelle dans une console Bash.
- Vous pouvez alors utiliser la commande **docker** et **docker-compose**.
- Le dossier **/vagrant** dans la machine virtuelle Linux vous permet d'accéder aux fichiers qui se trouvent sur votre ordinateur dans **C:\bionic** C'est un dossier de partage.
- Vous quittez la machine virtuelle en tapant **exit**
- Vous arrêtez la machine virtuelle en tapant **vagrant halt**
- Vous repartez la machine virtuelle en tapant **vagrant up**