



Activer WSL et installer Docker

Windows Subsystem Linux

Rédigé par

David ROUMANET
Professeur BTS SIO

Changement

Date	Révision
2020-09-12	Création
2020-09-20	V8 ajout d'utilisation d'un conteneur Docker et information sur proxy (annexe)
2022-11-15	Mise à jour importante (simplification WSL)

Sommaire

A Introduction à WSL.....	1
A.1 Problématique.....	1
A.2 Intérêt.....	1
B Installation.....	2
B.1 Version et prérequis.....	2
B.2 Activation WSL.....	2
B.3 Installation d'une distribution.....	3
B.4 Fonctionnement.....	4
B.5 Quelques instructions utiles.....	4
B.6 Autres fonctions utiles.....	5
B.6.1 Lancer un programme Windows.....	5
B.6.2 Utiliser le notepad windows pour éditer un fichier Linux.....	5
B.6.3 Installer une application sous Linux.....	5
C Docker et WSL2.....	6
C.1 Installation.....	6
C.2 Test d'un projet.....	7
D Annexes.....	9
D.1 Changement d'emplacement.....	9
D.1.1 Problème WSL2.....	9
D.1.2 Proposition sur une machine correctement configurée.....	9
D.2 Docker et Proxy.....	10
D.3 Sources.....	10
D.3.1 Microsoft.....	10
D.3.2 Autres.....	10

A Introduction à WSL

A.1 Problématique

Utiliser un OS, c'est se priver de l'usage d'un autre OS.

La guerre que se livrent, Microsoft (avec Windows) et la communauté Linux (Debian, Ubuntu...) tient au fait que leurs OS respectifs sont incompatibles. Un programme Linux ne fonctionne pas sous Windows et un programme Windows nécessite un "émulateur" pour fonctionner sous Linux.

Quelques sociétés ont donc proposé des solutions, comme VMWare (vSphere), Oracle (Virtualbox) ainsi que Microsoft (Hyper-V).

Ces solutions restent lourdes, car il est nécessaire de virtualiser chaque machine Linux indépendamment : réservation de mémoire vive, cœur de processeur, disque-dur... et cela a un impact important sur les performances. Un autre inconvénient est que les VM sont isolées (disques indépendants) et ne partagent donc pas le système de fichiers : il faut passer par des protocoles réseaux ou des outils particuliers.

De plus, l'utilisation de la solution Docker reste complexe et installe sous Windows.

Microsoft cherche depuis quelques années à réduire cet écart et a conçu WSL puis WSL2 qui est fourni avec la "build" 2004 de Windows 10.

A.2 Intérêt

La plupart du temps, les environnements de développements sont sous Linux, tandis que les entreprises préfèrent l'administration du parc de stations sous Windows.

En proposant un noyau Linux intégré à Windows, Microsoft limite la fuite vers Linux, tout en favorisant l'administration traditionnelle (migration de poste) pour les organisations.

Le noyau fonctionne dans une machine virtuelle très légère, dont les appels systèmes sont directement reliés aux fonctions (API) Windows. Pour caricaturer, le noyau Linux pense écrire sur une partition exfs4 alors que l'API Windows réalise l'écriture sur la partition en NTFS.

Alors que WSL fonctionne correctement mais connaît quelques limitations et problèmes de performances, WSL2 améliore la latence et réduit encore les problèmes de compatibilité.

B Installation

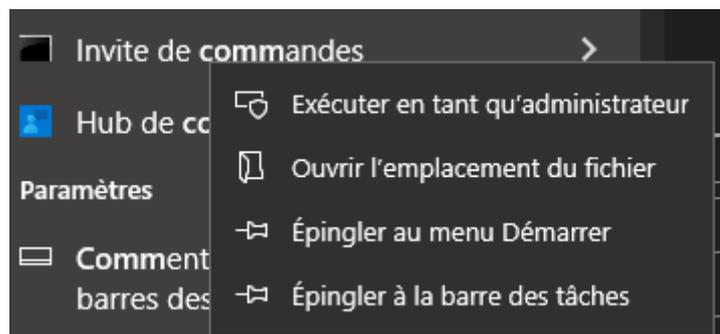
B.1 Version et prérequis

L'installation de WSL2 nécessite quelques prérequis et notamment la mise en œuvre de la mise à jour Windows 10 supérieure ou égale 2004. Pour vérifier votre version, utilisez le raccourci [Win]+[Pause]



B.2 Activation WSL

Lancer une invite de commande en mode administrateur.



Taper la commande suivante :

```
wsl --install
```

Il sera nécessaire de redémarrer durant le processus, mais votre machine devrait ensuite disposer de WSL2.

Anciennement : Le passage à WSL2 peut nécessiter l'installation d'un patch sur les machines ayant déjà WSL installé.

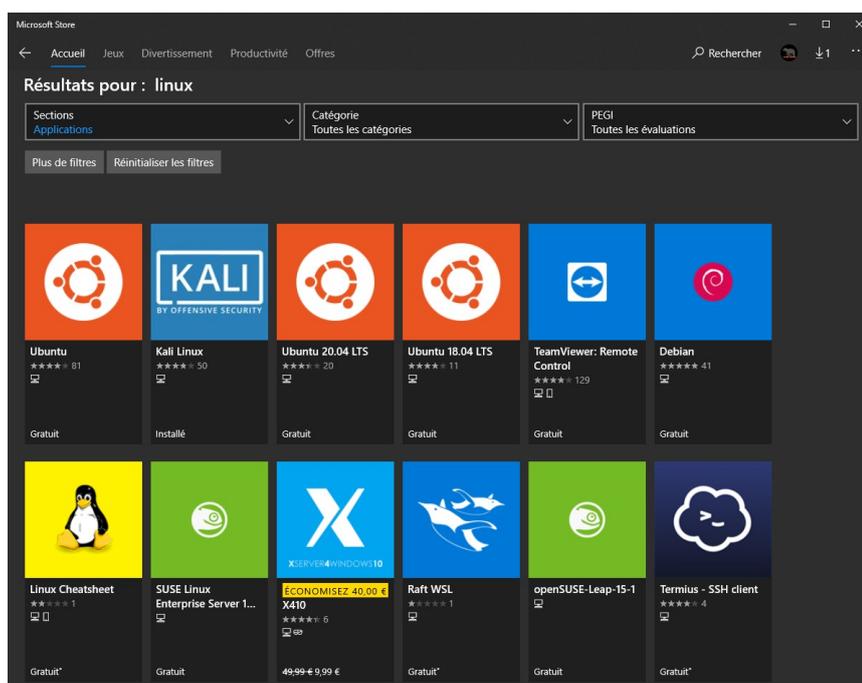
Voici le lien du fichier MSI : <https://aka.ms/wsl2kernel>

B.3 Installation d'une distribution

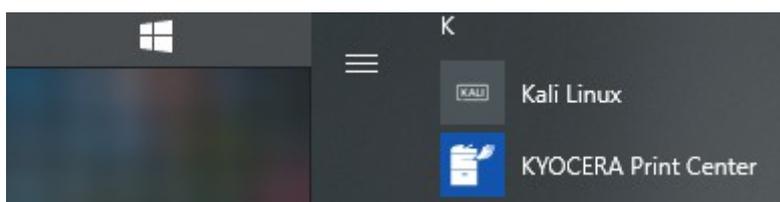
Par défaut, WSL doit installer une distribution Ubuntu. Si ce n'est pas le cas, utilisez la commande suivante :

```
wsl --install -d ubuntu
```

Vous pouvez aussi installer une distribution, via le Windows Store en recherchant le mot-clé 'linux' :



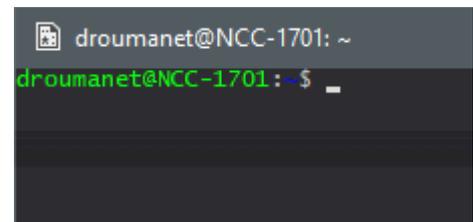
Il suffit de sélectionner une distribution puis de l'installer. Une fois installée, il y a une icône dans le menu Démarrer qu'il suffit de cliquer :



B.4 Fonctionnement

Il ne s'agit pas d'une distribution graphique, mais plutôt d'un mode "core". Dans l'environnement Kali que j'ai choisi, voici l'écran d'arrivée :

Les commandes Bash sont disponibles, comme ls, sudo, etc.



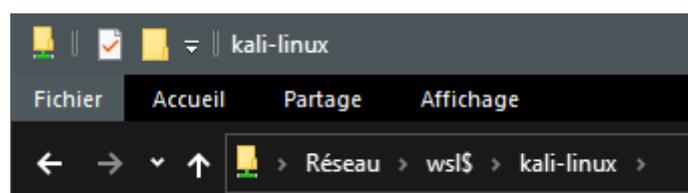
Ce qui est le plus intéressant, c'est que Linux monte automatiquement les disques présents sous Windows :

```
droumanet@NCC-1701:~$ mount
rootfs on / type lxfs (rw,noatime)
none on /dev type tmpfs (rw,noatime,mode=755)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,noatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,noatime)
...
C:\ on /mnt/c type drvfs (rw,noatime,uid=1000,gid=1000,case=off)
D:\ on /mnt/d type drvfs (rw,noatime,uid=1000,gid=1000,case=off)
F:\ on /mnt/f type drvfs (rw,noatime,uid=1000,gid=1000,case=off)
droumanet@NCC-1701:~$
```

Pour avoir les droits d'écriture sur le montage /mnt/c il faut lancer la distribution Linux en mode administrateur (clic droit sur l'icône).

```
droumanet@NCC-1701:~$ cd /mnt/c
droumanet@NCC-1701:/mnt/c$ touch linux.txt
droumanet@NCC-1701:/mnt/c$ nano linux.txt
droumanet@NCC-1701:/mnt/c$ more linux.txt
Hello word !
droumanet@NCC-1701:/mnt/c$
```

À l'inverse, il est possible de voir le disque de la machine Linux en utilisant le partage **\\wsl\$** !



B.5 Quelques instructions utiles

instruction	explication
wsl --list --verbose	Affiche les distributions (compatibles WSL) présentes et actives
wsl --setdefault <DistributionName>	Démarrer WSL avec une distribution par défaut
wsl --user <username>	Lancer la distribution Linux avec un utilisateur particulier

B.6 Autres fonctions utiles

B.6.1 Lancer un programme Windows

Dans une session Linux, il est possible de lancer un exécutable Windows. Pour cela, le nom du fichier doit être suivi de l'extension .exe

```
droumanet@NCC-1701:~$ cd $home
droumanet@NCC-1701:~$ calc.exe
droumanet@NCC-1701:~$
```

B.6.2 Utiliser le notepad windows pour éditer un fichier Linux

Les paramètres de commandes sont compris lors de l'appel d'un exécutable. Par exemple, éditer le fichier

```
droumanet@NCC-1701:~$ notepad.exe ~/.bashrc
```

B.6.3 Installer une application sous Linux

Les commandes Linux s'appliquent dans ce cas. Par exemple, pour installer GIT sous Linux :

```
sudo apt-get update
sudo apt-get install git-all
```

C Docker et WSL2

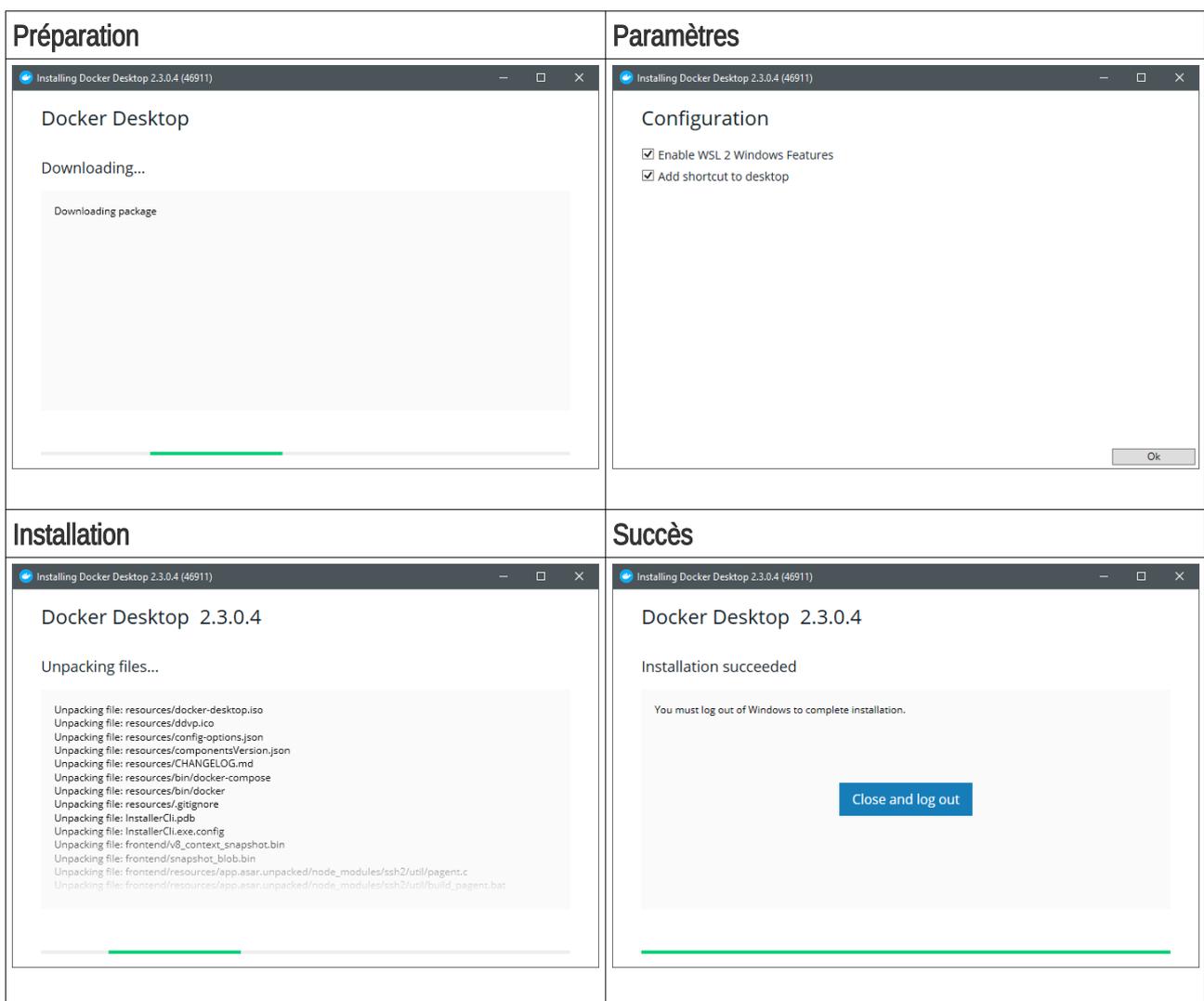
Docker fait figure d'exception, car il n'est pas utile de l'installer sous Windows ET sous Linux.

Docker prend en compte le développement de WSL2, ce qui évite l'installation de Microsoft Hyper-V (qui rend VMWare et Virtualbox inopérants¹).

C.1 Installation

Docker utilisera WSL2, il suffit donc de le télécharger et l'installer sous Windows.

Lien de téléchargement : <https://www.docker.com/products/docker-desktop>



Docker propose de fermer la session, veillez à ne pas avoir de fichiers ouverts si vous cliquez sur le bouton.

1 Voir <https://www.informatiweb.net/tutoriels/informatique/virtualisation/hyper-v-cohabitation-avec-virtualbox-ou-vmware-workstation.html>

C.2 Test d'un projet

Il est possible de tester l'installation correcte de Docker en installant le projet Hello-world :

```
docker pull hello-world
```

L'emplacement du projet sera dans la machine WSL définie par défaut. On peut le vérifier avec :

```
docker info
```

Cette commande retourne les informations de configuration de Docker, dont le chemin par défaut :

```
Client:
 Debug Mode: false

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 1
 Server Version: 19.03.12
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
 Swarm: inactive
 Runtimes: runc
 ...
 Kernel Version: 4.19.128-microsoft-standard
 Operating System: Docker Desktop
 OSType: linux
 Architecture: x86_64
 CPUs: 4
 Total Memory: 9.23GiB
 Name: docker-desktop
 ID: K7VX:I54D:H3MW:QCTQ:C3PU:454A:HHVA:YDSP:U6F5:5LLM:CRJ2:3ZUL
 Docker Root Dir: /var/lib/docker
 Debug Mode: true
  File Descriptors: 41
  Goroutines: 52
  System Time: 2020-09-12T18:23:35.6022198Z
  EventsListeners: 4
 Registry: https://index.docker.io/v1/
 Labels:
 Experimental: false
 Insecure Registries:
  127.0.0.0/8
 Live Restore Enabled: false
 Product License: Community Engine

WARNING: bridge-nf-call-iptables is disabled
WARNING: bridge-nf-call-ip6tables is disabled
```

Votre système d'exploitation est maintenant opérationnel, cependant, nous n'avons fait qu'effleurer les possibilités de Docker.

Il faut maintenant découvrir ce qu'est un conteneur (son mode de fonctionnement, comment en créer un, utiliser une image, etc.), pourquoi Docker est autant utilisé dans les entreprises.

En attendant une activité Docker complète, recherchez ce que Docker peut vous apporter, en termes de déploiement et de développement.

<https://www.docker.com/resources/what-container>

<https://www.docker.com/use-cases>

<https://docs.docker.com/>

D Annexes

D.1 Changement d'emplacement

Par défaut Docker stocke les containers sur la partition `C:` et cela peut poser un problème :

- Pour les machines ayant 2 partitions dont une en SSD (souvent de faible capacité, entre 128 et 256 Go)
- En cas de formatage de la partition contenant l'OS (Windows 10)

Il est donc préférable de configurer Docker pour placer les containers sur un autre disque.

D.1.1 Problème WSL2

La configuration est très liée à la configuration de Microsoft WSL.



La méthode qui consiste à ajouter l'attribut "data-root" dans le fichier JSON, dans Settings > Engine ne fonctionne pas lorsque Docker utilise WSL2. Ne surtout pas tenter de modifier ces paramètres.

Dans les faits, si le disque de destination est compressé ou chiffré, il ne sera pas possible d'exporter et ré-importer un VM WSL2 sur ce disque. Exemple de message d'erreur :

```
Impossible d'effectuer l'opération demandée en raison d'une limitation du système de disque virtuel. Les fichiers de disque dur virtuel doivent être non compressés et non chiffrés, et ne doivent pas être fragmentés.
```

D.1.2 Proposition sur une machine correctement configurée

L'idée est d'exporter puis importer la machine WSL à l'aide des commandes wsl :

```
mkdir d:\WSL
wsl --export kali-linux d:\temp\kali.tar
wsl --unregister kali-linux
wsl --import kali-linux d:\WSL\kali-linux d:\temp\kali.tar
wsl -l
```

Le paramètre `--unregister` permet de libérer le nom et de le ré-enregistrer lors de l'importation.

D.2 Docker et Proxy

Si vous êtes dans une organisation qui utilise un proxy pour accéder à Internet, il faut configurer le fichier `config.json` de Docker pour y ajouter une section proxy.

~/docker/config.json

```
{
  "proxies":
  {
    "default":
    {
      "httpProxy": "http://172.16.0.1:3128",
      "httpsProxy": "http://172.16.0.1:3128",
      "noProxy": "*.sio.local,.example2.com"
    }
  }
}
```

l'exemple est disponible dans la documentation Docker :

<https://docs.docker.com/network/proxy/>

D.3 Sources

D.3.1 Microsoft

FAQ : <https://docs.microsoft.com/fr-fr/windows/wsl/faq>

D.3.2 Autres

Changement disque sans WSL2 : <https://exploringtm1.com/how-to-point-docker-at-a-different-drive/>

vidéo Nick Janetakis : <https://nickjanetakis.com/blog/a-linux-dev-environment-on-windows-with-wsl-2-docker-desktop-and-more>

WSL2 par SiePoint : <https://www.sitepoint.com/wsl2/>