



Chapitre 3

Utilisation de Docker en DevSecOps

1. Introduction à Docker

1.1 L'histoire de la création de Docker

Docker est un logiciel libre utilisé pour lancer des applications à l'intérieur de conteneurs. Cependant, il est intéressant de connaître l'histoire de Docker pour mieux en saisir les différents impacts.

Solomon Hykes, un ancien étudiant de l'école Epitech (Promotion 2001), décide de créer la société DotCloud en 2008 avec Sébastien Pahl. C'est alors en 2010 que Solomon Hykes présente son concept de Docker afin d'améliorer l'efficacité du développement de logiciels et de leur déploiement, peu importe la machine sur laquelle ils sont exécutés.

DotCloud réussit une levée de fonds de 800 000 \$ (notamment avec les investisseurs qui ont été les premiers à croire en Google ou PayPal), et c'est seulement en 2013, après s'être installée à la Silicon Valley, que la société décide d'offrir les sources de sa technologie de conteneurisation à la communauté open source.

Aujourd'hui, Docker, c'est plus de 5 millions d'utilisateurs à travers le monde. L'entreprise Docker Inc. a été valorisée à près de 1,3 milliard de dollars en 2018.

1.2 Les différentes éditions de Docker

La première question que l'on peut se poser c'est pourquoi existe-t-il plusieurs versions de Docker ?

Tout d'abord, parce que Docker n'est pas uniquement un **logiciel**. Docker correspond à un environnement qui est beaucoup plus complet et très volumineux par rapport à ce qu'il était au moment de sa création.

Remarque

Une légende urbaine affirme qu'au tout début de Docker, il ne s'agissait que d'une centaine de lignes de Bash.

Cependant, Docker évolue très vite, et il faut donc toujours s'assurer d'utiliser la version la plus récente qui existe.

Subscription Plan	Price	Description	Call-to-Action
Docker Personal	\$0 /mo	For individual developers who need the essential tools to build and deploy containers.	Get started
Docker Pro	\$11 /mo	For individual professionals who require more advanced features and additional resources.	Buy now
Docker Team	\$16 per user/mo	For small teams that need collaborative tools to make working together more efficient.	Buy now
Docker Business	\$24 per user/mo	For enterprises desiring robust security, control, and compliance features.	Contact sales Buy now

Inclusions:

- Docker Desktop
- Docker Engine + Kubernetes
- Docker Hub
- Docker Scout

Everything in Personal plus:

- Docker Build Cloud
- Testcontainers Cloud
- Synchronized File Shares
- Docker Debug
- Visibility into Scout health scores

Everything in Pro plus:

- Add users in bulk
- Audit logs
- Hub role-based access control
- 2-day support response

Everything in Team plus:

- Hardened Docker Desktop
- Single Sign-On (SSO)
- SCIM user provisioning
- Image and Registry Access Management
- Desktop Insights Dashboard

Concrètement, il existe plusieurs souscriptions possibles à Docker. Vous pouvez retrouver plus d'informations sur le lien suivant : <https://www.docker.com/pricing/>.

Tout d'abord, celle que nous allons utiliser : **Personal**, qui est gratuite.

Elle est utilisable par les petites entreprises (moins de 250 employés et \$10 millions de revenu annuel), pour les utilisations personnelles, à but éducatif, et pour les projets open source non commerciaux.

Chapitre 3

Les autres souscriptions sont destinées aux entreprises et comportent quelques fonctionnalités supplémentaires comme :

- du support commercial ;
- la capacité de faire de l'automatisation via l'intégration continue ;
- des scans de vulnérabilités plus détaillés ;
- la création de dépôts privés pour nos images ;
- une intégration avancée à GitHub ;
- etc.

Cependant, vous verrez qu'il est tout à fait possible d'effectuer la plupart des actions manuellement, avec un tout petit peu de travail.

Docker est compatible avec les trois grands systèmes d'exploitation que sont Linux, Windows et macOS.

Concernant Linux, sachez qu'il s'agit du système de base sur lequel Docker a été développé, et c'était d'ailleurs le seul système supporté jusqu'en 2016. Bien sûr, il s'installe sur la plupart des systèmes Linux : Debian, Ubuntu, CentOS, Red Hat, etc.

Remarque

Il est cependant conseillé d'éviter d'utiliser le paquet par défaut (celui qui s'installe avec les utilitaires comme apt), car vous risquez de ne pas avoir la toute dernière version à jour.

Il est possible d'installer **Docker Desktop for Windows** sur les systèmes Windows. Cependant, il existe quelques conditions :

- Tout d'abord, seules les versions 64 bits sont supportées.
- De manière officielle, il faut utiliser Windows 11 ou Windows 10 en version Home, Pro, Enterprise ou Education.
- Il faut activer la fonctionnalité WSL 2 de Windows.

Remarque

Pas d'inquiétude, il existe des solutions alternatives pour ceux qui ont des versions plus anciennes de Windows.

Concernant **macOS**, à partir de la version 10.15, il n'y a pas de restrictions pour y installer Docker Desktop.

Même si nous venons de parler de systèmes d'exploitation, Docker s'est principalement illustré par une utilisation sur le **Cloud**.

On y retrouve un support par les trois grands acteurs du Cloud que sont Amazon AWS, Microsoft Azure et Google Cloud Platform.

En somme, il est possible d'utiliser Docker peu importe le système sur lequel vous vous trouvez, à condition de pouvoir y effectuer de la **virtualisation**.

1.3 L'utilisation de Docker sur Windows

Beaucoup pensent encore que Docker n'est utilisable que sur Linux ou sur le Cloud, mais c'est tout à fait possible sur Windows. En effet, grâce à de nouvelles versions, la compatibilité est excellente et les performances très intéressantes pour ceux qui ne possèdent pas de machines Linux, et souhaitent quand même s'entraîner à la maîtrise de Docker.

Il faut comprendre qu'il y a différents types de conteneurs sur Windows. Il est bien sûr possible d'y lancer des **conteneurs type Linux** (ceux de base) mais aussi des **conteneurs Windows**.

Dans 99 % des cas, lorsque vous entendez quelqu'un parler de conteneur, c'est toujours de conteneurs Linux dont il est question.

Gardez en tête que Windows a implémenté nativement la gestion de conteneurs sur certaines versions de son système, notamment grâce à **WSLv2** et **Hyper-V**.

La version complètement supportée par Docker correspond à **Docker Desktop for Windows**. Cette version est téléchargeable sur le site officiel de Docker : <https://docs.docker.com/desktop/windows/install/>

Elle nécessite l'activation de WSLv2 selon la procédure détaillée un peu plus loin dans cet ouvrage.

Pour Windows 11 et 10, il faut cependant une machine en **64 bits**, d'une version minimum correspondant à :

- Windows 11-64 bits Home ou version Pro, Enterprise, ou Education 21H2 minimum ;
- Windows 10-64 bits Home, Pro 2004 (build 19041 minimum), Enterprise et Education 1909 (build 18363 minimum).

Il faut également au minimum **4 Go de mémoire RAM** et y installer le **Linux Kernel Update Package**.

■ Remarque

En fonction de la date à laquelle vous lisez cet ouvrage, ces prérequis auront peut-être changé.

■ Remarque

Pour vérifier les informations : <https://docs.docker.com/desktop/windows/install/> puis allez dans la section « System Requirements ».

Si vous possédez une version plus ancienne de Windows, il est toujours possible d'y exécuter du Docker grâce à l'outil **Docker Toolbox**.

En fait, dans cette version, le système reproduit une machine Linux sur laquelle tournent les conteneurs Linux qui sont créés.

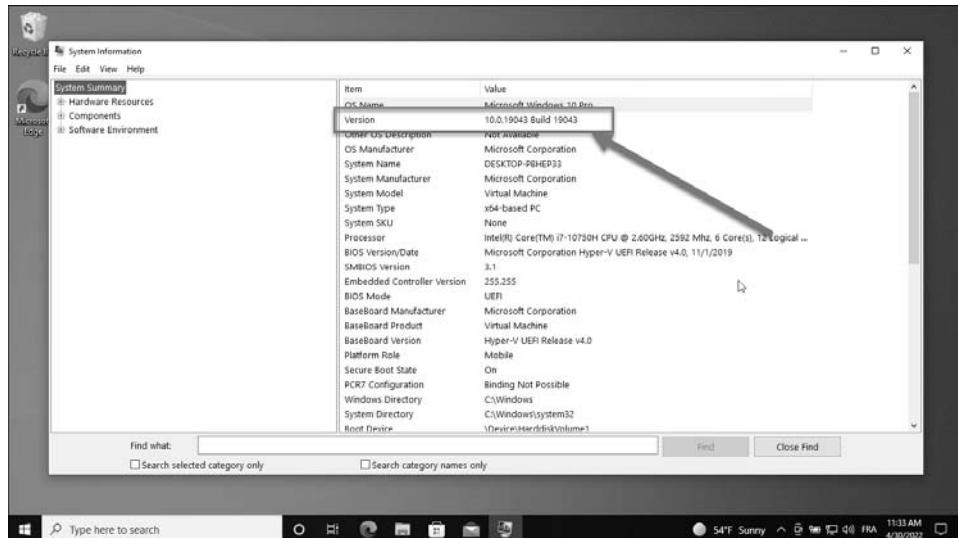
Tout ceci reste assez transparent pour l'utilisateur, et nous verrons étape par étape comment l'installer et l'utiliser de façon simple et intuitive.

1.4 L'installation étape par étape de Docker Desktop sur Windows

1.4.1 Vérification de votre version de Windows

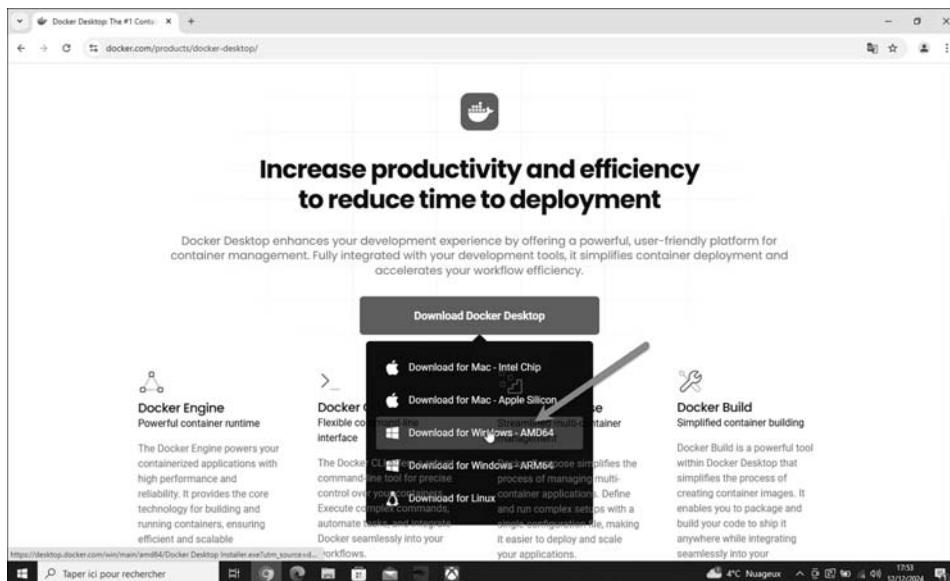
Commençons par vérifier que votre version de Windows est compatible avec Docker Desktop.

- Rendez-vous dans la barre de recherche en bas à gauche et tapez **Information du système**.



Pour rappel, au moment de l'écriture de cet ouvrage, votre version doit être supérieure à :

- Windows 11-64 bits Home ou version Pro, Enterprise, ou Education 21H2 minimum ;
 - Windows 10-64 bits Home, Pro 2004 (build 19041 minimum), Enterprise et Education 1909 (build 18363 minimum).
- Rendez-vous ensuite sur <https://www.docker.com/products/docker-desktop/> puis cliquez sur **Download Docker Desktop** et sélectionnez votre version de **Windows AMD64** ou **ARM64** (si vous ne savez pas, il est très probable que ce soit AMD64).



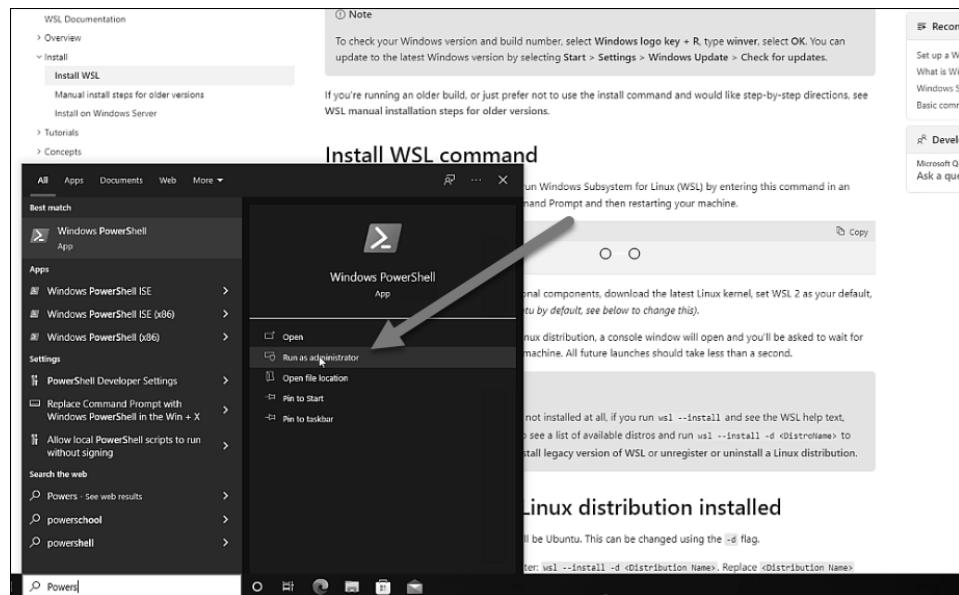
1.4.2 Installation de WSLv2 et du Linux Kernel Package

Une fois la première étape de vérification effectuée, nous devons installer WSLv2.

■ Remarque

Rendez-vous sur <https://docs.microsoft.com/en-us/windows/wsl/install> afin de vérifier que les prochaines étapes n'ont pas changé depuis la rédaction de cet ouvrage.

- Ouvrez **Windows PowerShell** en mode **Administrateur** comme indiqué sur la capture suivante :



- Tapez ensuite la commande suivante :

```
■ wsl -install
```

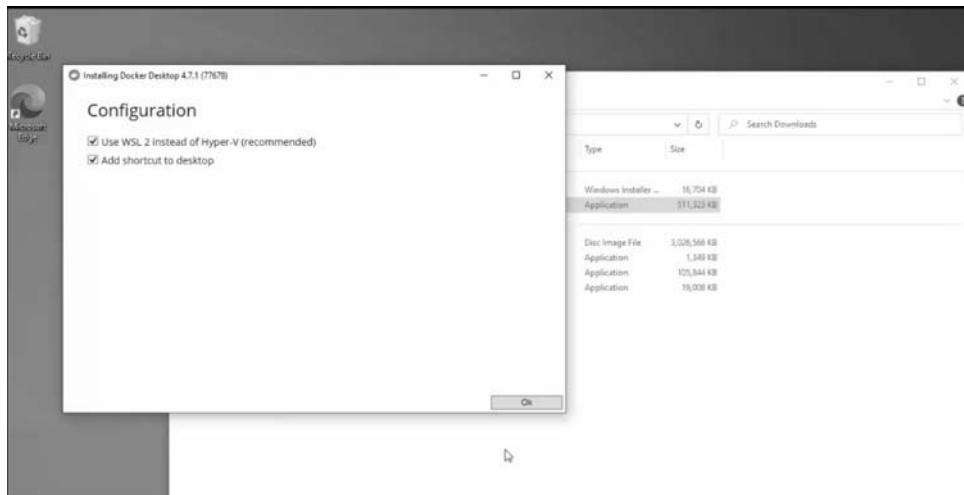
- Redémarrez ensuite l'ordinateur.

■ Remarque

En cas de difficultés lors de l'exécution de la commande, vous aurez peut-être besoin de vous tourner vers l'installation manuelle décrite ici :
<https://learn.microsoft.com/en-us/windows/wsl/install-manual>

1.4.3 Installation de Docker Desktop

- Lancez votre fichier Docker Desktop installer.exe.
- Laissez cochées les cases **Use WSL 2 instead of Hyper-V (recommended)** et **Add shortcut to desktop**.



■ Remarque

En fonction de votre version de Docker Desktop, vous pouvez avoir une vision différente de celle de la capture ci-dessus. Le plus important est de cocher l'option disant d'utiliser WSL 2 plutôt qu'Hyper-V.

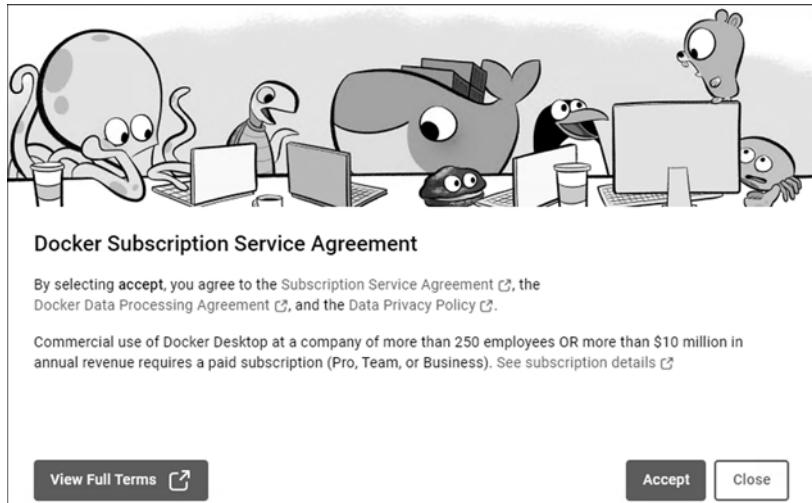
- Continuez l'installation.

■ Remarque

Vous allez sans doute être déconnecté de votre compte utilisateur actuel à la fin de l'installation, c'est tout à fait normal, et vous avez simplement besoin de vous reconnecter pour terminer le processus.

- Cliquez sur l'icône du bureau permettant de lancer Docker Desktop.

- Lisez puis acceptez les termes de la licence qui reprend les éléments dont nous avions parlé dans la section liée aux différentes éditions de Docker (cf. section Les différentes éditions de Docker).



- Cochez la case **Use recommended settings** et cliquez sur **Finish** :

