

Mise en place de Docker sur Linux

Docker est une plateforme de conteneurisation qui encapsule des applications et leurs dépendances dans des conteneurs, assurant une portabilité et une cohérence entre différents environnements. Il utilise des images pour créer ces conteneurs isolés, facilitant le développement, le test et le déploiement d'applications. Docker simplifie la gestion des applications en rendant leur déploiement rapide et fiable, quel que soit l'environnement cible.

- [Qu'est-ce que Docker](#)
- [Installation de Docker sur Linux](#)

Qu'est-ce que Docker



Docker est une plateforme de conteneurisation open source qui permet de développer, déployer et exécuter des applications dans des conteneurs. Un conteneur, dans le contexte de Docker, encapsule une application avec toutes ses dépendances (bibliothèques, outils, fichiers de configuration, etc.) dans un seul paquet ou "image". Cela garantit que l'application s'exécute de manière uniforme et prévisible sur n'importe quel environnement prenant en charge Docker, qu'il s'agisse d'un ordinateur personnel, d'un serveur, ou d'un cloud.

Avantages de Docker



- **Portabilité** : Les conteneurs Docker peuvent être exécutés sur n'importe quel système d'exploitation ou plateforme supportant Docker, ce qui réduit les problèmes de compatibilité.
- **Isolation** : Chaque conteneur fonctionne de manière isolée des autres et du système hôte, ce qui améliore la sécurité et réduit les conflits entre applications.
- **Efficacité** : Docker permet une utilisation plus efficace des ressources système par rapport aux machines virtuelles traditionnelles, car les conteneurs partagent le même noyau de système d'exploitation et isolent uniquement l'application et ses dépendances.
- **Déploiement rapide** : Docker facilite le déploiement rapide des applications en permettant aux développeurs de créer une image de conteneur une seule fois, puis de la déployer partout où Docker est installé.
- **Développement et déploiement cohérents** : Les développeurs peuvent travailler dans des environnements qui correspondent exactement à la configuration de production, ce qui réduit les surprises lors du déploiement.

Composants Clés de Docker

- **Images Docker** : Les modèles en lecture seule utilisés pour créer des conteneurs. Une image contient tout ce qui est nécessaire pour exécuter une application, y compris le code, les bibliothèques et les variables d'environnement.
- **Conteneurs Docker** : Les instances exécutables d'une image. Les conteneurs s'exécutent de manière isolée les uns des autres et du système hôte.
- **[Docker Hub](#)** : Un registre public de Docker où les utilisateurs peuvent télécharger et partager des images Docker. Il existe également des registres privés pour stocker des images de manière sécurisée.
- **Dockerfile** : Un fichier texte contenant des instructions sur comment construire une image Docker spécifique. Les Dockerfiles automatisent le processus de création d'images.

Installation de Docker sur Linux

Installer Docker sur Linux vous permet de lancer des applications dans des conteneurs logiciels légers et portables.

Nous allons ici installer Docker sur une distribution Linux basée sur Debian ou Ubuntu, qui sont parmi les plus populaires.

Étape 1 : Mise à jour du système

Commencez par mettre à jour la liste des paquets et mettez à jour votre système pour vous assurer que toutes les dépendances sont à jour. Ouvrez un terminal et tapez la commande :

```
sudo apt update && sudo apt upgrade -y
```

Étape 2 : Installer les paquets prérequis



Il y a quelques paquets dont Docker a besoin pour s'exécuter correctement. Tapez cette commande afin de les installer :

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common -y
```

Étape 3 : Ajouter la clé GPG officielle de Docker

Pour s'assurer que les paquets téléchargés sont authentiques, ajoutez la clé GPG du dépôt officiel Docker :

```
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

Lors de l'installation de paquets comme Docker, la clé GPG sert à vérifier l'authenticité des paquets téléchargés. En ajoutant la clé GPG du dépôt Docker à votre système, vous dites à votre gestionnaire de paquets de faire confiance aux paquets venant de ce dépôt, assurant qu'ils n'ont pas été altérés et qu'ils sont bien ceux publiés par l'équipe Docker. C'est une mesure de sécurité importante pour éviter l'installation de logiciels malveillants.

Étape 4 : Ajouter le dépôt Docker aux sources APT



Ajoutez le dépôt officiel Docker à vos sources APT pour obtenir les dernières versions de Docker :

```
echo \  
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]  
https://download.docker.com/linux/ubuntu \  
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \  
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Étape 5 : Installer Docker CE (Community Edition)

Mettez à jour à nouveau votre liste de paquets pour inclure les nouveaux paquets du dépôt Docker, puis installez Docker CE :

```
sudo apt-get update  
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-  
compose-plugin
```

Étape 6 : Vérifier l'installation de Docker



Pour vous assurer que Docker est correctement installé et fonctionne, lancez :

```
sudo systemctl status docker
```

```
renard@webserv01:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-04-03 07:05:30 UTC; 1min 22s ago
     TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
    Main PID: 12555 (dockerd)
       Tasks: 10
      Memory: 35.9M
         CPU: 1.023s
      CGroup: /system.slice/docker.service
             └─12555 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

Vous devriez voir que le service est actif. Vous pouvez également exécuter un conteneur de test :

```
sudo docker run hello-world
```

```
renard@webserv01:~$ sudo docker run hello-world
```

```
Hello from Docker!
```

```
This message shows that your installation appears to be working correctly.
```

```
To generate this message, Docker took the following steps:
```

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

```
To try something more ambitious, you can run an Ubuntu container with:
```

```
$ docker run -it ubuntu bash
```

```
Share images, automate workflows, and more with a free Docker ID:
```

```
https://hub.docker.com/
```

```
For more examples and ideas, visit:
```

```
https://docs.docker.com/get-started/
```

Ce conteneur teste votre installation en affichant un message de bienvenue.

C'est tout bon !



Félicitations, vous avez installé Docker sur votre système Linux ! Vous pouvez maintenant commencer à utiliser Docker pour développer, déployer et exécuter des applications dans des conteneurs