

# Docker avec Portainer

Docker peut remplacer une partie des fonctionnalités de Proxmox pour créer et gérer des conteneurs, et **Portainer** fournit une interface graphique qui facilite leur gestion.

- **Installation de Docker et Portainer** : Docker est rapide à installer sur la majorité des VPS, et Portainer ajoute une interface web simple pour gérer les conteneurs, les réseaux, les volumes, etc.
- **Utilisation** : Depuis Portainer, tu peux créer, stopper, et surveiller facilement tes conteneurs et services. Tu peux même définir des "stacks" pour gérer des ensembles de conteneurs avec Docker Compose.
- **Avantages** : L'installation est légère et bien adaptée aux VPS. Portainer permet aussi de gérer Docker Swarm si tu veux répartir des services sur plusieurs VPS.
- [Configuration et installations](#)

# Configuration et installations

## 1. Mettre à jour le système

Assure-toi d'abord que le système est à jour en exécutant les commandes suivantes :

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

## 2. Installer Docker

1 - Installe les dépendances nécessaires pour Docker :

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

2 - Ajoute la clé GPG de Docker :

```
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

3 - Ajoute le dépôt Docker à tes sources de package :

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]  
https://download.docker.com/linux/debian $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list  
> /dev/null
```

4 - Installe Docker :

```
sudo apt update  
sudo apt install docker-ce docker-ce-cli containerd.io -y
```

5 - Vérifie que Docker fonctionne correctement :

```
sudo systemctl start docker  
sudo systemctl enable docker  
sudo docker --version
```

### 3. Installer Portainer

1 - Crée un volume Docker pour stocker les données de Portainer :

```
sudo docker volume create portainer_data
```

2 - Lance Portainer en tant que conteneur Docker :

```
sudo docker run -d -p 8000:8000 -p 9443:9443 --name=portainer --restart=always -v  
/var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
```

### 4. Accéder à Portainer

- Une fois que Portainer est en cours d'exécution, tu peux y accéder via un navigateur en entrant l'adresse suivante : `https://[IP_de_ton_serveur]:9443`.
- Lors de la première connexion, tu seras invité à créer un compte administrateur pour sécuriser l'accès.

### 5. Configuration et Utilisation

- Une fois connecté, tu pourras gérer tes conteneurs Docker, créer de nouveaux services, surveiller les ressources, et configurer des stacks directement depuis l'interface de Portainer.

### 6. (Optionnel) : Vérifier le statut des services

Tu peux vérifier le statut de Docker et Portainer pour confirmer qu'ils fonctionnent correctement avec ces commandes :

```
sudo systemctl status docker  
sudo docker ps # Liste les conteneurs actifs, y compris Portainer
```

C'est tout ! Tu devrais maintenant avoir Docker et Portainer fonctionnels sur ton OS, prêts pour la gestion de conteneurs depuis une interface web.