

Création d'un Cluster Docker Swarm

Créer un cluster Docker Swarm est un processus direct qui peut être accompli en quelques étapes.

Ce tutoriel vous guidera à travers la création d'un cluster Swarm, depuis les prérequis jusqu'à la promotion de nœuds en tant que managers ou workers, en passant par la configuration initiale.

Prérequis

1. **Machines Virtuelles (VMs) ou Serveurs Physiques** : Vous aurez besoin d'au moins une machine pour agir en tant que manager du Swarm et de deux autres machines comme nœuds workers. Trois machines sont un bon point de départ pour un environnement de production minimal pour la tolérance aux pannes. Ces machines doivent être configurées avec Docker.
2. **Docker Installé** : Chaque machine doit avoir Docker installé. Vous pouvez trouver les instructions d'installation sur le site officiel de Docker pour diverses distributions Linux, Windows, et macOS.
3. **Réseau** : Les machines doivent être capables de communiquer entre elles à travers les ports utilisés par Docker Swarm (TCP port 2377 pour la communication entre les managers, et TCP et UDP ports 7946, UDP port 4789 pour le trafic overlay network).

Étape 1 : Initialisation du Swarm

Choisissez une de vos machines pour être le manager initial du Swarm. Connectez-vous à cette machine, puis exécutez :

```
docker swarm init --advertise-addr <MANAGER-IP>
```

Remplacez `<MANAGER-IP>` par l'adresse IP de la machine que vous souhaitez utiliser comme manager. Cette commande initialise un nouveau Swarm et configure la machine actuelle en tant que manager du Swarm.

Étape 2 : Ajout de Nœuds au Swarm

Après avoir initialisé le Swarm, vous obtiendrez un message contenant un token pour joindre d'autres nœuds au Swarm en tant que workers.

Sur chaque machine destinée à être un worker, exécutez la commande fournie, qui ressemblera à ceci :

```
docker swarm join --token SWMTKN-1-xxxxxxxxxx-xxxxxxxxxx-xxxxxxxxxx <MANAGER-IP>:2377
```

- Remplacez `<MANAGER-IP>` par l'adresse IP du manager Swarm. Cette commande permet à la machine d'agir comme un nœud worker dans le Swarm.

Étape 3 : Vérification de l'État du Swarm

Pour vérifier que les nœuds ont bien rejoint le Swarm, exécutez la commande suivante sur le manager :

```
docker node ls
```

Cela affichera la liste de tous les nœuds dans le Swarm, y compris leur rôle (manager ou worker).

Exemple

```
root@EVA-00:~# docker node ls
ID                HOSTNAME          STATUS  AVAILABILITY  MANAGER STATUS  ENGINE VERSION
1p14w09a3a1ysibjegktesxwr * EVA-00  Ready  Active        Reachable       20.10.24+dfsg1
7tnbhyvlbmuv27dnk8d3oqf9b EVA-01  Ready  Active        Leader          20.10.24+dfsg1
5mxg7b4mnkjd3z8vcwo9ygsdy  EVA-02  Ready  Active                          20.10.24+dfsg1
```

Étape 4 : Promotion d'un Nœud Worker en Manager

Si vous souhaitez augmenter la tolérance aux pannes de votre cluster Swarm en ajoutant un manager supplémentaire, vous pouvez promouvoir un nœud worker. Sur un nœud manager, exécutez :

```
docker node promote <WORKER-NODE-ID>
```

Remplacez `<WORKER-NODE-ID>` par l'ID du nœud que vous souhaitez promouvoir, que vous pouvez trouver en exécutant `docker node ls`.

Source :

[Docker Swarm Official](#)

Conclusion

Vous avez maintenant un cluster Docker Swarm fonctionnel composé d'un manager et de plusieurs nœuds workers. Vous pouvez déployer des applications sur ce cluster en utilisant des services Docker Swarm, ce qui permet de gérer facilement la disponibilité, le scaling, et les mises à jour de vos applications conteneurisées.

Ce tutoriel couvre les bases, mais Docker Swarm offre de nombreuses autres fonctionnalités et options de configuration pour optimiser et sécuriser votre cluster.

Revision #5

Created 11 March 2024 13:58:45 by MASSON Romain

Updated 11 March 2024 15:28:32 by MASSON Romain