

Création de Machines Virtuelles avec Docker Machine

Docker Machine permet de créer des **machines virtuelles** (VM) et d'y installer Docker pour faciliter la gestion de vos hôtes Docker, qu'ils soient locaux (par exemple, sur VirtualBox) ou distants (par exemple, sur AWS, Azure, ou Google Cloud). Vous pouvez personnaliser les machines virtuelles en ajustant les ressources, les réseaux, et les paramètres du fournisseur de cloud.

Voici un guide détaillé avec les commandes nécessaires pour créer une machine virtuelle avec Docker Machine.

1. Créer une Machine Virtuelle avec Docker Machine

Étape 1 : Choisir un driver

Docker Machine fonctionne avec différents drivers pour créer des machines virtuelles, selon que vous travaillez sur une machine locale (comme avec VirtualBox) ou sur un service cloud (comme AWS, Google Cloud, etc.).

Voici quelques drivers populaires :

- **virtualbox** : Utilisé pour créer des VMs locales avec VirtualBox.
- **amazonec2** : Utilisé pour créer des instances EC2 sur AWS.
- **azure** : Utilisé pour créer des machines virtuelles sur Microsoft Azure.
- **google** : Utilisé pour créer des instances sur Google Cloud.

Étape 2 : Commandes pour créer une machine virtuelle

1. Créer une machine virtuelle avec VirtualBox (local)

Voici la commande de base pour créer une machine virtuelle Docker sur VirtualBox :

```
docker-machine create --driver virtualbox my-docker-vm
```

- **--driver virtualbox** : Spécifie que Docker Machine doit utiliser VirtualBox pour créer la machine virtuelle.
- **my-docker-vm** : Nom de la machine virtuelle à créer.

2. Personnaliser la machine virtuelle avec VirtualBox

Vous pouvez personnaliser la machine virtuelle en spécifiant plusieurs options comme le nombre de CPU, la quantité de mémoire RAM, ou la taille du disque. Voici un exemple de commande avec ces options :

```
docker-machine create --driver virtualbox --virtualbox-cpu-count 2 --virtualbox-memory 4096 --virtualbox-disk-size 20000 my-docker-vm
```

- **--virtualbox-cpu-count 2** : Alloue 2 cœurs CPU à la VM.
- **--virtualbox-memory 4096** : Alloue 4 Go de RAM à la VM.
- **--virtualbox-disk-size 20000** : Alloue 20 Go de disque dur virtuel.

3. Créer une machine virtuelle sur AWS (cloud)

Si vous souhaitez créer une machine virtuelle Docker sur AWS, vous pouvez utiliser le driver **amazonec2**. Vous devez d'abord configurer vos clés d'accès AWS.

Exemple de commande pour créer une instance EC2 Docker sur AWS :

```
docker-machine create --driver amazonec2 --amazonec2-region us-west-2 --amazonec2-instance-type t2.micro --amazonec2-access-key <AWS_ACCESS_KEY> --amazonec2-secret-key <AWS_SECRET_KEY> my-aws-docker-vm
```

- **--driver amazonec2** : Spécifie que Docker Machine doit utiliser AWS EC2 comme driver.
- **--amazonec2-region us-west-2** : La région AWS dans laquelle l'instance sera déployée.
- **--amazonec2-instance-type t2.micro** : Type d'instance EC2 à créer (par exemple, **t2.micro** pour une petite instance).
- **--amazonec2-access-key** et **--amazonec2-secret-key** : Vos clés d'accès AWS pour l'authentification.

4. Créer une machine virtuelle sur Google Cloud (cloud)

Pour créer une instance sur Google Cloud, vous pouvez utiliser le driver **google** :

```
docker-machine create --driver google --google-project <PROJECT_ID> --google-zone us-central1-a --google-machine-type n1-standard-1 --google-image-family debian-9 --google-image-project debian-cloud my-gcloud-docker-vm
```

- **--driver google** : Spécifie Google Cloud comme fournisseur de cloud.
- **--google-project <PROJECT_ID>** : Votre identifiant de projet sur Google Cloud.
- **--google-zone us-central1-a** : Zone dans laquelle déployer la machine virtuelle.
- **--google-machine-type n1-standard-1** : Type de machine (par exemple, **n1-standard-1**).
- **--google-image-family debian-9** : Image système de la machine (ici Debian 9).
- **--google-image-project debian-cloud** : Projet image Debian sur Google Cloud.

5. Créer une machine virtuelle sur Microsoft Azure (cloud)

Si vous utilisez **Microsoft Azure** comme fournisseur de cloud, vous pouvez créer une machine virtuelle Docker avec la commande suivante :

```
docker-machine create --driver azure --azure-subscription-id
<SUBSCRIPTION_ID> --azure-client-id <CLIENT_ID> --azure-client-secret
<CLIENT_SECRET> --azure-tenant-id <TENANT_ID> --azure-resource-group my-
docker-rg --azure-location westus my-azure-docker-vm
```

- **--driver azure** : Spécifie Azure comme fournisseur de cloud.
- **--azure-subscription-id <SUBSCRIPTION_ID>** : Identifiant de votre abonnement Azure.
- **--azure-client-id, --azure-client-secret, --azure-tenant-id** : Identifiants pour l'authentification via Azure.
- **--azure-resource-group** : Groupe de ressources Azure où déployer la VM.
- **--azure-location westus** : Emplacement où la VM sera déployée.

2. Vérification de la Création de la Machine Virtuelle

Une fois que vous avez créé une machine virtuelle avec Docker Machine, vous pouvez vérifier son état et sa configuration avec la commande suivante :

```
docker-machine ls
```

Cette commande affiche la liste des machines virtuelles créées, avec leur nom, leur adresse IP, leur état et le driver utilisé.

Exemple de sortie :

NAME	ACTIVE	DRIVER	STATE	URL
SWARM	DOCKER	ERRORS		
my-docker-vm	*	virtualbox	Running	
			v19.03.8	
my-aws-docker-vm	-	amazonec2	Running	
			v19.03.8	

- **ACTIVE** : Indique quelle machine virtuelle est actuellement active.
- **STATE** : Indique l'état de la machine (par exemple, "Running").
- **URL** : L'URL à laquelle Docker Machine est accessible.

3. Connexion à la Machine Virtuelle Docker

Pour vous connecter à la machine virtuelle et interagir avec Docker, utilisez la commande **docker-machine ssh** :

```
docker-machine ssh my-docker-vm
```

Cela vous connecte à l'hôte Docker via SSH, vous permettant de travailler directement sur la machine virtuelle.

4. Configurer le Client Docker pour Utiliser un Hôte

Une fois la machine virtuelle créée, vous pouvez configurer votre client Docker pour l'utiliser en définissant les variables d'environnement nécessaires. Utilisez la commande suivante pour configurer automatiquement l'environnement de votre terminal :

```
eval $(docker-machine env my-docker-vm)
```

Cela configure le client Docker pour qu'il interagisse avec l'hôte Docker sur la machine virtuelle **my-docker-vm**. Vous pouvez maintenant exécuter des commandes Docker sur cette machine virtuelle, comme **docker ps**, **docker run**, etc.

Résumé

- **Docker Machine** permet de créer, configurer et gérer des machines virtuelles pour exécuter Docker sur plusieurs plateformes.
- **Commandes** : Utilisez la commande **docker-machine create** pour créer des machines virtuelles avec différents **drivers** (comme VirtualBox, AWS, Azure, Google Cloud).
- **Personnalisation** : Vous pouvez personnaliser les ressources des machines (CPU, RAM, disque, etc.) en spécifiant des options lors de la création.
- **Gestion des machines** : Utilisez **docker-machine ls** pour voir la liste des machines et **docker-machine ssh** pour vous connecter à une machine virtuelle.

Avec Docker Machine, la gestion de plusieurs hôtes Docker devient beaucoup plus simple et centralisée.