[Podman] Cheat-sheet

Introduction

Podman est un gestionnaire de conteneur dit "**deamonless**" (sans démon), ce qui signifie qu'il ne nécessite pas de service systemd ou openrc pour fonctionner et fonctionne même avec les utilisateurs non privilégiés des systèmes Unix.

Hormis, ces différences, il est presque identique à docker dans l'utilisation et le fonctionnement.



Images

Build une image

Après avoir créé un **Dockerfile** (même syntaxe que docker), vous pouvez construire l'image :

```
podman build -t <IMG> .
```

Lister les images

```
podman images
```

Supprimer une image

```
podman rmi <IMG>
```

Afficher l'historique des modifications de l'image

```
podman history <IMG>
```

Supprimer les images inutilisées

```
podman image prune -a
```

Réseaux

Lister les réseaux

```
podman network ls
```

Créer un réseau

```
podman network create <NET_NAME>
```

Connecter un conteneur à un réseau

```
podman run <CT_NAME> --network <NET_NAME> -d <IMG>
```

Afficher les détails d'un réseau

```
podman network inspect <NET_NAME>
```

Volumes

Créer un volume

```
podman volume create <VOL_NAME>
```

Lister les volumes

```
podman volume ls
```

Attacher un volume à un conteneur

```
podman run -v <VOL_NAME>:<PATH> <IMG>
```

Supprimer un volume

```
podman volume rm <VOL_NAME>
```

Supprimer les volumes inutilisés

```
podman volume prune
```

Débogage et monitoring

Afficher les logs d'un conteneur

```
podman logs <CT_NAME>
```

Afficher l'utilisation des ressources des conteneurs actifs

```
podman stats
```

Lancer un shell dans un conteneur

```
podman exec -it <CT_NAME> bash
```

Afficher les informations détaillées d'un conteneur

```
podman inspect <CT_NAME>
```

Divers

Construire une image à partir d'un conteneur existant

```
podman commit <CT_NAME> <IMG>
```

Copier des données du conteneur sur l'hôte

```
podman cp <CT_NAME>:<PATH/TO/FILE> <HOST_PATH>
```

Copier des données de l'hôte sur le conteneur

```
podman cp <HOST_PATH> <CT_NAME>:<PATH/TO/FILE>
```

Nettoyer les ressources inutilisées (volumes, conteneurs, images)

```
podman system prune -a
```

Podman Compose

Podman Compose est équivalent à Docker Compose et permet de construire des stacks à partir d'un fichier **docker-compose.yml**.

Démarrer une stack

```
podman-compose up -d
```

Arrêter une stack

```
podman-compose down
```

Afficher toutes les stacks actives

```
podman-compose ls
```

Afficher les logs de tous les services de la stack

```
podman-compose logs
```

Exécuter une commande unique dans un service

```
podman-compose run <SVC_NAME> <COMMAND>
```

Mettre à jour et reconstruire un service spécifique

```
podman-compose up -d --no-deps <SVC_NAME>
```

Arrêter un service spécifique

```
podman-compose stop <SVC_NAME>
```

Migration docker vers podman

La migration de service docker vers podman est relativement simple dues à la similitudes des commandes, prenez en comptes les deux éléments suivants :

- Le **système de fichiers** utilisé par les **volumes** est différent de celui de docker (surtout en mode rootless).
- Les **réseaux** sont gérés de manière légèrement différente.

> Lors de vos migrations, vous devriez prendre en compte ces deux éléments et peut être effectué des modifications sur la configuration initiale pour migrer correctement vos services.

Podman Desktop

Ce logiciel graphique permet de gérer vos conteneurs, vos volumes et vos images un peu à la même manière que Docker Desktop ou Portainer.

Installation

```
flatpak install --user flathub io.podman_desktop.PodmanDesktop
```

Revision #4
Created 2024-04-22 13:32:17 UTC by Elieroc
Updated 2024-04-22 15:41:42 UTC by Elieroc