

podman 使用笔记

virhuiai

目 录

第 1 章 macOS 上 Podman 的安装配置	3
1.1 前言.....	3
1.2 环境准备：配置文件持久化.....	4
1.3 Podman 安装与初始化.....	4
1.3.1 安装 Podman	4
1.3.2 初始化并启动虚拟机	5

第1章

macOS 上 Podman 的安装配置

1.1 前言

本文档记录了在 macOS 系统上使用 Homebrew 安装并配置 Podman 的完整流程，包括：

- 配置文件持久化（符号链接）
- Podman 虚拟机初始化与启动
- 镜像源加速配置
- 存储路径自定义
- 常见问题与优化建议

适用于 macOS 15.6，Podman 版本 5.6.2。

1.2 环境准备：配置文件持久化

将用户配置文件迁移至外部存储（如移动硬盘），防止系统重装丢失。

```
# 移动 .zshrc 到外部目录
mv ~/.zshrc /Volumes/THAWSPACE/Soft.0k/devEnv/

# 创建符号链接
ln -s /Volumes/THAWSPACE/Soft.0k/devEnv/.zshrc ~/.zshrc

# 创建容器配置目录
mkdir -p /Volumes/THAWSPACE/Soft.0k/devEnv/containers_config
mkdir -p /Volumes/THAWSPACE/Soft.0k/devEnv/containers_share

# 链接容器配置目录
ln -s /Volumes/THAWSPACE/Soft.0k/devEnv/containers_config
→ ~/.config/containers
ln -s /Volumes/THAWSPACE/Soft.0k/devEnv/containers_share
→ ~/.local/share/containers
```

1.3 Podman 安装与初始化

1.3.1 安装 Podman

使用 Homebrew 安装 Podman

```
brew install podman
```

安装输出示例 (Podman 5.6.2):

```
==> Fetching podman
==> Downloading https://ghcr.io/v2/homebrew/core/podman/...
==> Pouring podman--5.6.2.sonoma.bottle.tar.gz
==> Caveats
In order to run containers locally, podman depends on a Linux kernel.
One can be started manually using `podman machine` from this package.
To start a podman VM automatically at login, also install the cask
→ "podman-desktop".
==> Summary
/usr/local/Cellar/podman/5.6.2: 217 files, 79.8MB
```

1.3.2 初始话并启动虚拟机

```
podman machine init --now
```

等价于：

```
podman machine init
podman machine start
```

启动成功后输出：

```
Machine "podman-machine-default" started successfully
API forwarding listening on: /var/run/docker.sock
```

```
Docker API clients default to this address. You do not need to set
↪ DOCKER_HOST.
```

1.4 测试运行容器

```
# 运行 hello-world 测试容器
podman run --rm hello-world
```

输出：

```
!... Hello Podman World ...!
 .--"--
 / - - \
 / (0) (0) \
 ~~~| --(,Y,)=- |
 .---. /` \ |~~
 ~/ o o \~~~~.----. ~~
 | =(X)= |~ / (0 (0) \
 ~~~~~~ ~| =(Y_=)- |
 ~~~~ ~~~| U |~~
Project: https://github.com/containers/podman
Website: https://podman.io
```

1.5 镜像源加速

```
podman pull docker.1ms.run/erlang:28.1.0.0-alpine
```