Rosetta Support

Podman v5.1

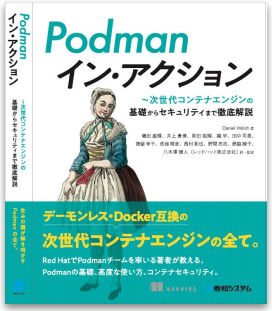
Shion Tanaka
shtanaka@redhat.com

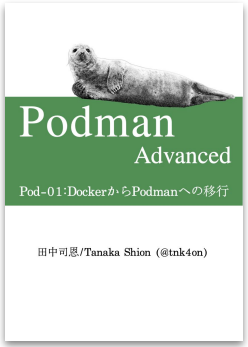
2024/6/4
Podman Community Meeting

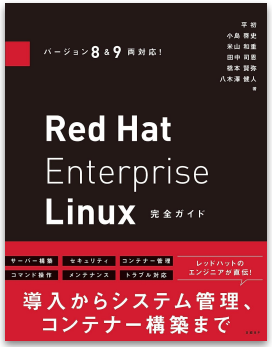
Red Hat

About me

- NAME: Shion Tanaka(@tnk4on)
- Solution Architect at Red Hat in Japan
- Wrote many blog posts, magazines and books on OpenShift and Podman
- Hosted Podman hands-on and webinars for customers and partners
- Hosted community events for Podman and OpenShift







https://www.redhat.com/ja/events/webinar/master-next-gen-container-engine
https://openshift.connpass.com/event/298708/
https://openshift.connpass.com/event/310201/

Red Hat

podmanjp

Podman's unofficial Japanese account

Red Hat

Rosetta in Podman v5.1

Red Hat

Rosetta in Podman v5.1

from Release Notes v5.1.0

- “VMs created by podman machine on macOS with Apple silicon can now use Rosetta 2 (a.k.a Rosetta) for high-speed emulation of x86 code. This is enabled by default. If you wish to change this option, you can do so in containers.conf.”

https://github.com/containers/podman/releases/tag/v5.1.0

What is Rosetta ?

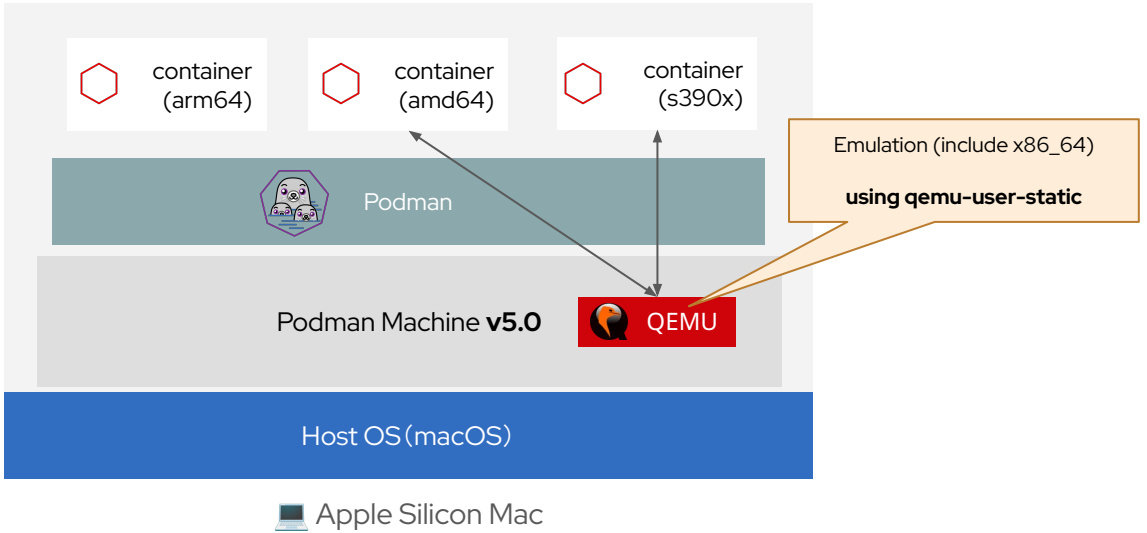
Rosetta is Apple's binary translation technology that provides compatibility between different processor architectures.

- **Rosetta** (first generation): Released in 2006, aimed to run PowerPC applications on Macs with Intel chips.
- **Rosetta 2**: Released in 2020, aimed to run Intel applications on Macs with Apple Silicon chips.
  - In macOS 13 and later, the Virtualization framework supports Rosetta in ARM Linux virtual machines

https://web.archive.org/web/20060113055505/http://www.apple.com/rosetta/
https://support.apple.com/guide/security/rosetta-2-on-a-mac-with-apple-silicon-secebb113be1/web
https://developer.apple.com/documentation/virtualization/running_intel_binaries_in_linux_vms_with_rosetta

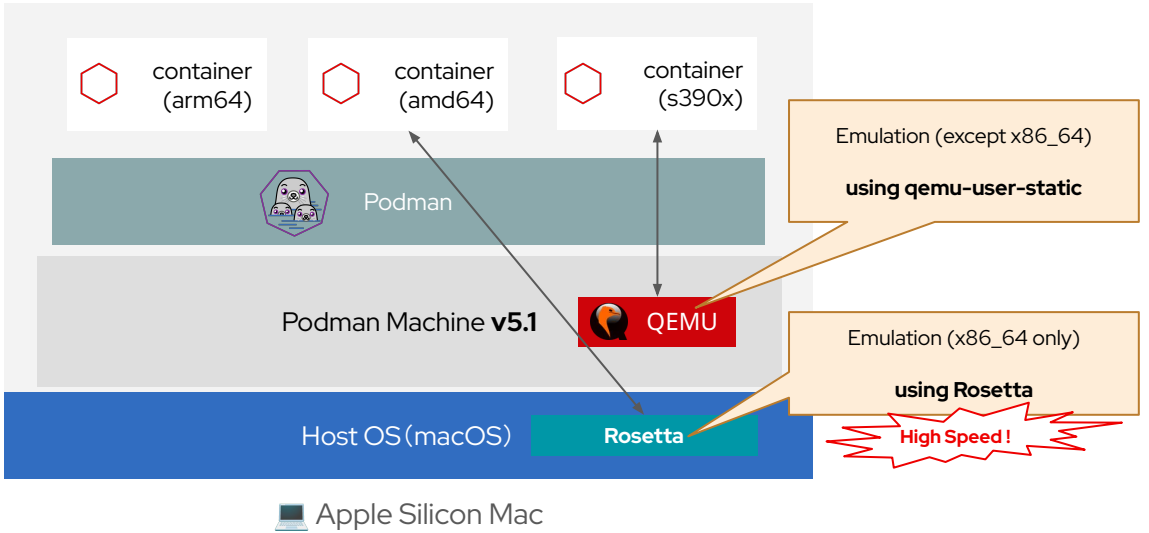
Emulation on macOS (up to v5.0)

Using qemu-user-static for QEMU's User Mode Emulation



Emulation on macOS (after v5.1)

Using Rosetta for x86_64 emulation



How to use Rosetta

Rebuild Podman machine

Rosetta requires Podman machine v5.1 or higher

1. Remove existing Podman machine

```
% podman machine rm -f
```

2. Check Podman CLI version

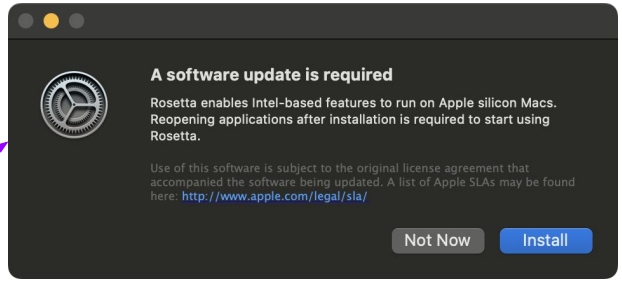
```
% podman -v
podman version 5.1.0
```

3. Initialize and Start Podman machine

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

after start

If Rosetta is not installed



or

manual install via CLI

```
% softwareupdate --install-rosetta --agree-to-license
```

Red Hat

Rebuild Podman machine

Rosetta requires Podman machine v5.1 or higher

4. Check Rosetta flag in machine config

```
% podman machine inspect --format {{.Rosetta}}
true
```

5. Check Rosetta mounts inside Podman machine

```
% podman machine ssh
core@localhost:~$ ls -ld /mnt/rosetta*
-rwxr-xr-x. 1 core core 1660888 Feb 15 11:04 /mnt/rosetta
-rwxr-xr-x. 1 core core  298792 Feb 15 11:04 /mnt/rosettad
core@localhost:~$ ls -ld /proc/sys/fs/binfmt_misc/rosetta
-rw-r--r--. 1 root root 0 Jun  3 23:21 /proc/sys/fs/binfmt_misc/rosetta
```

6. Check qemu-x86_64 (does not exist)

```
core@localhost:~$ ls -ld /proc/sys/fs/binfmt_misc/qemu-x86_64
ls: cannot access '/proc/sys/fs/binfmt_misc/qemu-x86_64': No such file or directory
```

Run containers with Rosetta

1. `podman run` with `--arch` option

```
% uname -m
arm64
% podman run --rm --arch amd64 ubi9 uname -m
x86_64
```

2. Check the process in `podman top`

```
% podman run --rm -d --arch amd64 --name rosetta ubi9 sleep inf
% podman top rosetta x
    PID TTY      STAT   TIME COMMAND
      1 ?        Ss     0:00 /mnt/rosetta /usr/bin/coreutils --coreutils-prog-shebang=sleep /usr/bin/sleep inf
```

Build containers with Rosetta

1. Create the Containerfile

```
% cat > Containerfile <<EOF
FROM ubi9
RUN arch | tee arch.txt
EOF
```

2. Build the Containerfile with `--platform` option

```
% podman build -t test --platform linux/amd64 .
STEP 1/2: FROM ubi9
STEP 2/2: RUN arch | tee arch.txt
x86_64
COMMIT test
--> bdf2bc2ad050
Successfully tagged localhost/test:latest
bdf2bc2ad050eea2aa5b24f5f78a537401f2d6788a423727a266a1cf31ad7b5a
% podman inspect test --format {{.Architecture}}
amd64
```

How to disable Rosetta

You can disable Rosetta via containers.conf.

1. Create containers.conf (if it does not exist) and add `rosetta=false` in the machine section

```
% cat > ~/.config/containers/containers.conf <<EOF
[machine]
rosetta=false
EOF
```

2. Restart the Podman machine. Rosetta will be disabled and qemu-x86_64 will be enabled instead.

```
% podman machine stop
% podman machine start
% podman machine inspect --format {{.Rosetta}}
false
% podman machine ssh
core@localhost:~$ ls -ld /mnt/rosetta*
ls: cannot access '/mnt/rosetta*': No such file or directory
core@localhost:~$ ls -ld /proc/sys/fs/binfmt_misc/rosetta
ls: cannot access '/proc/sys/fs/binfmt_misc/rosetta': No such file or directory
core@localhost:~$ ls -ld /proc/sys/fs/binfmt_misc/qemu-x86_64
-rw-r--r--. 1 root root 0 May 22 03:07 /proc/sys/fs/binfmt_misc/qemu-x86_64
```

How to disable Rosetta

You can disable Rosetta via containers.conf.

3. Check the process in `podman top`

```
% podman run --rm -d --arch amd64 --name qemu ubi9 sleep inf
% podman top qemu x
    PID TTY      STAT   TIME COMMAND
      1 ?        Ss     0:00 /usr/bin/qemu-x86_64-static /usr/bin/coreutils --coreutils-prog-shebang=sleep
/usr/bin/sleep inf
```

Note:

- Podman Desktop does not yet support Rosetta.
- I have opened an RFE Issue.
  - Add Rosetta settings for Apple Silicon Mac · Issue #7367 · containers/podman-desktop
  - https://github.com/containers/podman-desktop/issues/7367

DEMO

Known Issues

How to migrate to Podman machine v5.1?

There is no way to upgrade a Podman machine v5.0 to v5.1.

- Running rpm-ostree upgrade inside a podman machine causes an error (Issue #22678)
  - Because the rpm-ostree reference points to a non-existent URL →quay.io/containers/podman-machine-os:5.0
- Workaround
  - ` *podman machine os apply quay.io/podman/machine-os:5.1*`
- This problem occurs with every Podman machine version upgrade.
  - We should keep discussing this issue.

Can't upgrade core machine · Issue #22678 · containers/podman
https://github.com/containers/podman/issues/22678

Rosetta cannot be used in some cases

Rosetta doesn't translate the following executables

- Kernel extensions
- Virtual Machine apps that virtualize x86_64 computer platforms
- AVX, AVX2, and AVX512 vector instructions

Images that cannot be used (that are known)

- quay.io/fedoraci/fedora:eln
- centos:stream10-development
- quay.io/podman/stable
  - Cannot pull or build inside container

```
% podman run --rm --arch amd64 -it quay.io/fedoraci/fedora:eln
Fatal glibc error: CPU does not support x86-64-v3
```

```
% podman run --rm --arch amd64 -it centos:stream10-development
(response will be stuck)
```

About the Rosetta Translation Environment | Apple Developer Documentation
https://developer.apple.com/documentation/apple-silicon/about-the-rosetta-translation-environment#What-Cant-Be-Translated

Thank you


linkedin.com/company/red-hat


youtube.com/user/RedHatVideos


facebook.com/redhatinc


twitter.com/RedHat

Red Hat