

# podman????

1. docker를 설치하고 실행

1. podman-docker 패키지를 설치하고 docker를 실행  
podman을 사용하여 실행.

2. podman을 사용하여 실행

```
$> yum install podman-docker -y
podman podman 1:40:24 podman: 2022 02 07 ( ) pod 12 21 09.
podman podman.
=====
=====
podman podman podman podman podman podman
=====
=====
podman podman:
podman-docker
noarch 3.3.1-9.0.1.module+el8.5.0+20416+d687fed7 ol8_appstream
56 k

podman podman
=====
=====
podman 1 podman

podman podman podman: 56 k
podman podman : 230
podman podman:
podman-docker-3.3.1-9.0.1.module+el8.5.0+20416+d 288 kB/s | 56 kB 00:00
-----
-----
podman 284 kB/s | 56 kB 00:00
podman podman podman
podman podman podman.
```

```

[[ [[ [[ ]
[[ [[ [[ [[[[[[.
[[ [[ ]
    [[ ]      :
                                     1/1

[[ ]      : podman-docker-3.3.1-9.0.1.module+el8.5.0+20416+d687f
1/1

[[[[[[ [[ ]: podman-docker-3.3.1-9.0.1.module+el8.5.0+20416+d687f    1/1
[/usr/lib/tmpfiles.d/pesign.conf:1] Line references path below legacy
directory /var/run/, updating /var/run/pesign → /run/pesign; please update
the tmpfiles.d/ drop-in file accordingly.

[[ ]      : podman-docker-3.3.1-9.0.1.module+el8.5.0+20416+d687f
1/1

[[[[[[[[[:
    podman-docker-3.3.1-9.0.1.module+el8.5.0+20416+d687fed7.noarch

[[[[[[[[[!

```

```
00 0000 00:
    container-tools/common
```

```
crun-1.0-1.module+el8.5.0+20416+d687fed7.x86_64 5/5
```

```

1/5
2/5
3/5
4/5
5/5

```

容器镜像:

```

crun-1.0-1.module+el8.5.0+20416+d687fed7.x86_64
python3-podman-3.2.0-2.module+el8.5.0+20416+d687fed7.noarch
python3-pytml-0.1.14-5.git7dea353.el8.noarch
python3-pyxdg-0.25-16.el8.noarch
udica-0.2.5-2.module+el8.5.0+20416+d687fed7.noarch

```

容器镜像 !

3. docker 的 socket api 和 podman-docker 的 socket 都在 /var/run/docker.sock 和 /var/run/podman/podman.sock 中。docker-py, docker-compose 使用 docker api 和 podman 使用 podman api。
4. podman 使用 docker 的 socket 和 podman 的 socket，使用 podman 的 socket，使用 podman 的 socket，使用 podman 的 socket。

## 2. Rootless

1. 使用 rootless 容器，使用 rootless 容器，使用 rootless 容器。(root - /var/lib/containers/storage, non root - \$HOME/.local/share/containers/storage)
2. rootless 容器，使用 rootless 容器，使用 rootless 容器。(tcp/80 容器，使用 rootless 容器，使用 rootless 容器，使用 rootless 容器)
3. 1024 容器，使用 rootless 容器，使用 rootless 容器，使用 rootless 容器，使用 rootless 容器，使用 rootless 容器。(rootless 容器，使用 rootless 容器，使用 rootless 容器)

```
$> echo 80 > /proc/sys/net/ipv4/ip_unprivileged_port_start
```

4. 8.1 使用 rootless Containers 容器，使用 rootless 容器，使用 rootless 容器，使用 rootless 容器。(rootless 容器，使用 rootless 容器，使用 rootless 容器)

```
$> podman pull docker.io/library/httpd
Trying to pull docker.io/library/httpd:latest...
Getting image source signatures
Copying blob 5eb5b503b376 done
Copying blob 10c4d45228bf done
Copying blob a43a76ccc967 done
Copying blob 942bd346e7f7 done
Copying blob cdb155854ae6 done
Copying config a8ea074f45 done
Writing manifest to image destination
Storing signatures
a8ea074f4566addcd01f9745397f32be471df4a4abf200f0f10c885ed14b1d28
[user@test~]$ docker image ls
Emulate Docker CLI using podman. Create /etc/containers/nodocker to quiet msg.
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
docker.io/library/httpd	latest	a8ea074f4566	11 days ago	148 MB

### 3. 配置容器镜像仓库

1. registries.conf 配置容器镜像仓库，配置容器镜像仓库，配置容器镜像仓库，配置容器镜像仓库。

```
$> vi /etc/containers/registries.conf
...
unqualified-search-registries = ["container-registry.oracle.com",
"docker.io", "container.test.com"]
...
```

2. test.com 配置 TLS 证书，pull 镜像

```
$> vi /etc/containers/registries.conf
...
[[registry]]
location="container.test.com"
insecure = true
...
```

3. 配置容器镜像仓库，配置容器镜像仓库

```
$> vi /etc/containers/registries.conf
...
[[registry]]
location="container.test.com"
blocked = true
...
```

#### 4. 验证 配置

1. skopeo 验证 配置 是否 正确 。
2. httpd container 验证 配置 是否 正确

```
# skopeo inspect docker://docker.io/library/httpd:latest
{
  "Name": "docker.io/library/httpd",
  "Digest":
"sha256:5cc947a200524a822883dc6ce6456d852d7c5629ab177dfbf7e38c1b4a647705",
  "RepoTags": [
    "2",
    "2-alpine",
    "2-alpine3.13",
    "2-alpine3.14",
    "2-alpine3.15",
    "2-bullseye",
    "2-buster",
    "2.2",
    "2.2-alpine",
    "2.2.29",
    "2.2.31",
    "2.2.31-alpine",
    "2.2.32",
    "2.2.32-alpine",
    "2.2.34",
    "2.2.34-alpine",
    "2.4",
    "2.4-alpine",
    "2.4-alpine3.13",
    "2.4-alpine3.14",
    "2.4-alpine3.15",
    "2.4-bullseye",
```

"2.4-buster",  
"2.4.10",  
"2.4.12",  
"2.4.16",  
"2.4.17",  
"2.4.18",  
"2.4.20",  
"2.4.23",  
"2.4.23-alpine",  
"2.4.25",  
"2.4.25-alpine",  
"2.4.27",  
"2.4.27-alpine",  
"2.4.28",  
"2.4.28-alpine",  
"2.4.29",  
"2.4.29-alpine",  
"2.4.32",  
"2.4.32-alpine",  
"2.4.33",  
"2.4.33-alpine",  
"2.4.34",  
"2.4.34-alpine",  
"2.4.35",  
"2.4.35-alpine",  
"2.4.37",  
"2.4.37-alpine",  
"2.4.38",  
"2.4.38-alpine",  
"2.4.39",  
"2.4.39-alpine",  
"2.4.41",  
"2.4.41-alpine",  
"2.4.43",  
"2.4.43-alpine",  
"2.4.46",  
"2.4.46-alpine",  
"2.4.47",  
"2.4.47-alpine",  
"2.4.48",

```
    "2.4.48-alpine",
    "2.4.48-alpine3.13",
    "2.4.48-alpine3.14",
    "2.4.48-buster",
    "2.4.49",
    "2.4.49-alpine",
    "2.4.49-alpine3.14",
    "2.4.49-buster",
    "2.4.50",
    "2.4.50-alpine",
    "2.4.50-alpine3.14",
    "2.4.50-buster",
    "2.4.51",
    "2.4.51-alpine",
    "2.4.51-alpine3.14",
    "2.4.51-alpine3.15",
    "2.4.51-bullseye",
    "2.4.51-buster",
    "2.4.52",
    "2.4.52-alpine",
    "2.4.52-alpine3.15",
    "2.4.52-bullseye",
    "alpine",
    "alpine3.13",
    "alpine3.14",
    "alpine3.15",
    "bullseye",
    "buster",
    "latest"
  ],
  "Created": "2022-01-26T08:38:51.175633696Z",
  "DockerVersion": "20.10.7",
  "Labels": null,
  "Architecture": "amd64",
  "Os": "linux",
  "Layers": [
    "sha256:5eb5b503b37671af16371272f9c5313a3e82f1d0756e14506704489ad9900803"
  ,
    "sha256:a43a76ccc96739928c7e884f2210dde01ae5b1fd8822f8cdd56e6ba64ec3125a"
  ,
  ]
```



```
    "sha256:942bd346e7f719d26e022dfc42eea7e1fa5cf9ad60ec80ed0ef79ded05288be6"
  ,
    "sha256:cdb155854ae6a9e25834459a7c9dfc7be157a2ebfca5adbdb036aeea43ce3128"
  ,
    "sha256:10c4d45228bf56285a1e2c828d60e34d8413ee80e1abd738ef190be843d9dc1e"
  ],
  "Env": [
    "PATH=/usr/local/apache2/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
    "HTTPD_PREFIX=/usr/local/apache2",
    "HTTPD_VERSION=2.4.52",
    "HTTPD_SHA256=0127f7dc497e9983e9c51474bed75e45607f2f870a7675a86dc90af6d572f5c9",
    "HTTPD_PATCHES="
  ]
```

---

Revision #1

Created 8 June 2022 03:19:55 by artop0420

Updated 24 December 2023 02:58:52 by artop0420