# 1 CheatSheet: CRI-O

CLOUD

Updated: February 24, 2020

- PDF Link: cheatsheet-crio-A4.pdf, Category: Cloud
- Blog URL: https://cheatsheet.dennyzhang.com/cheatsheet-crio-A4
- Related posts: Docker CheatSheet, Kubectl CheatSheet, #denny-cheatsheets

File me Issues or star this repo.

## 1.1 cri-o Basic

Name	Summary	
cri-o	Another container runtime like dockerd. Initialized from Redhat	
cri-o vs dockerd		
Reference	CheatSheet: Docker, CheatSheet: CRI-O, CheatSheet: rkt, CheatSheet: containerd	

# 1.2 cri-o Components

Name	Summary	
crictl	For troubleshooting and working directly with CRI-O container engines	
runc	For running container images	
$\operatorname{podman}$	CLI tool to manage pods and images	
Buildah	For building, pushing and signing container images	
Skopeo	For copying, inspecting, deleting, and signing images	
Quay	Builds, analyzes, distributes your container images. Like CNCF Harbor	

https://raw.githubusercontent.com/dennyzhang/cheatsheet.dennyzhang.com/master/cheatsheet-crio-A4/crio-arch.png

# 1.3 Run cri-o as k8s container engine

Name	Summary
Run cri-o with minikube	minikube startnetwork-plugin=cnicontainer-runtime=cri-o
cri-o conf file	/etc/crio/crio.conf
Configure cri-o image registry	registries = ['docker.io']
Restart cri-o	minikube ssh, then systemctl restart crio
Check status	systemctl status
Reference	Link: Using the CRI-O Container Engine

# 1.4 podman

Name	Summary
podman image registry file	/etc/containers/registries.conf
pull container	podman pull nginx:alpine
List all images	podman images
List all contianers	podman ps

# 1.5 podman online usage

```
$ sudo podman --help
```

WARN[0000] unable to find /etc/containers/registries.conf. some podman (image shortnames) commands may be limentable to find /etc/containers/registries.conf.

podman - manage pods and images

#### USAGE:

podman [global options] command [command options] [arguments...]

## **VERSION:**

0.4.1

### COMMANDS:

Updated: February 24, 2020 attach Attach to a running container commit Create new image based on the changed container build Build an image using instructions in a Dockerfile create create but do not start a container diff Inspect changes on container's file systems Run a process in a running container exec Export container's filesystem contents as a tar archive export Show history of a specified image history list images in local storage images Import a tarball to create a filesystem image import display system information info Displays the configuration of a container or image inspect Kill one or more running containers with a specific signal kill load an image from docker archive load login to a container registry login logout of a container registry logout Fetch the logs of a container logs mount Mount a working container's root filesystem Pauses all the processes in one or more containers pause List containers ps port List port mappings or a specific mapping for the container pull an image from a registry pull push push an image to a specified destination Restart one or more containers restart podman rm will remove one or more containers from the host. The container name or ID of This does not remove images. Running containers will not be removed without removes one or more images from local storage rmi run a command in a new container run

Save image to an archive search registry for image search Start one or more containers start

Display percentage of CPU, memory, network I/O, block I/O and PIDs for one or more cont stats

stop Stop one or more containers

tag Add an additional name to a local image Display the running processes of a container top umount, unmount Unmount a working container's root filesystem Unpause the processes in one or more containers unpause

Display the PODMAN Version Information version

Block on one or more containers wait

help, h Shows a list of commands or help for one command

### GLOBAL OPTIONS:

--cni-config-dir value path of the configuration directory for CNI networks --config value, -c value path of a config file detailing container server configuration options --conmon value path of the conmon binary --cpu-profile value path for the cpu profiling results log messages above specified level: debug, info, warn, error (default), --log-level value --root value path to the root directory in which data, including images, is stored --runroot value path to the 'run directory' where all state information is stored

--runtime value path to the OCI-compatible binary used to run containers, default is /us

--storage-driver value, -s value select which storage driver is used to manage storage of images and cont used to pass an option to the storage driver --storage-opt value

show help --help, -h

--version, -v print the version

## More Resources

https://github.com/containers

https://cri-o.io/

License: Code is licensed under MIT License.

Updated: February 24, 2020