



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'podman-ps.1'

\$ man podman-ps.1

podman-ps(1)() podman-ps(1)()

NAME

podman-ps - Prints out information about containers

SYNOPSIS

podman ps [options]

podman container ps [options]

podman container list [options]

podman container ls [options]

DESCRIPTION

podman ps lists the running containers on the system. Use the --all flag to view all the containers information. By default it lists:

? container id

? the name of the image the container is using

? the COMMAND the container is executing

? the time the container was created

? the status of the container

? port mappings the container is using

? alternative names for the container

OPTIONS

--all, -a

Show all the containers created by Podman, default is only running containers.

Note: Podman shares containers storage with other tools such as Buildah and CRI-O. In some cases these external containers might also exist in the same storage. Use the --external

option to see these external containers. External containers show the 'storage' status.

--external

Display external containers that are not controlled by Podman but are stored in containers storage. These external containers are generally created via other container technology such as Buildah or CRI-O and may depend on the same container images that Podman is also using. External containers are denoted with either a 'buildah' or 'storage' in the COM? MAND and STATUS column of the ps output. Only used with the --all option.

--filter, -f

Filter what containers are shown in the output. Multiple filters can be given with multiple uses of the --filter flag. Filters with the same key work inclusive with the only exception being label which is exclusive. Filters with different keys always work exclusive.

Valid filters are listed below:

??

?Filter ? Description ?

??

?id ? [ID] Container's ID (accepts ?

? ? regex) ?

??

?name ? [Name] Container's name (accepts ?

? ? regex) ?

??

?label ? [Key] or [Key=Value] Label as? ?

? ? signed to a container ?

??

?exited ? [Int] Container's exit code ?

??

?status ? [Status] Container's status: ?

? ? 'created', 'exited', 'paused', ?

? ? 'running', 'unknown' ?

??

?ancestor ? [ImageName] Image or descendant ?

? ? used to create container ?

??

?before ? [ID] or [Name] Containers cre? ?

? ? ated before this container ?

??

?since ? [ID] or [Name] Containers cre? ?

? ? ated since this container ?

??

?volume ? [VolumeName] or [MountpointDes? ?

? ? tination] Volume mounted in con? ?

? ? tainer ?

??

?health ? [Status] healthy or unhealthy ?

??

?pod ? [Pod] name or full or partial ID ?

? ? of pod ?

??

?network ? [Network] name or full ID of ?

? ? network ?

??

--format=format

Pretty-print containers to JSON or using a Go template

Valid placeholders for the Go template are listed below:

??

?Placeholder ? Description ?

??

?.ID ? Container ID ?

??

?.Image ? Image Name/ID ?

??

?.ImageID ? Image ID ?

??

?.Command ? Quoted command used ?

??

?.CreatedAt ? Creation time for container ?

??

?.RunningFor ? Time elapsed since container was ?

? ? started ?

??

?.Status ? Status of container ?

??

?.Pod ? Pod the container is associated ?

? ? with ?

??

?.Ports ? Exposed ports ?

??

?.Size ? Size of container ?

??

?.Names ? Name of container ?

??

?.Networks ? Show all networks connected to ?

? ? the container ?

??

?.Labels ? All the labels assigned to the ?

? ? container ?

??

?.Mounts ? Volumes mounted in the container ?

??

--help, -h

Print usage statement

--last, -n

Print the n last created containers (all states)

--latest, -l

Show the latest container created (all states) (This option is not available with the re?
mote Podman client)

--namespace, --ns

Display namespace information

--noheading

Omit the table headings from the listing of containers.

--no-trunc

Display the extended information

--pod, -p

Display the pods the containers are associated with

--quiet, -q

Print the numeric IDs of the containers only

--sort

Sort by "command, created, id, image, names, runningfor, size, or status", Note: Choosing size will sort by size of rootFs, not alphabetically like the rest of the options Default:

created

--size, -s

Display the total file size

--sync

Force a sync of container state with the OCI runtime. In some cases, a container's state in the runtime can become out of sync with Podman's state. This will update Podman's state based on what the OCI runtime reports. Forcibly syncing is much slower, but can resolve inconsistent state issues.

--watch, -w

Refresh the output with current containers on an interval in seconds.

EXAMPLES

```
$ podman ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
02f65160e14ca	redis:alpine	"redis-server"	19 hours ago	Exited (-1)	19 hours ago	6379/tcp
k8s_podsandbox1-redis_podsandbox1_redhat.test.crio_redhat-test-crio_0						
69ed779d8ef9f	redis:alpine	"redis-server"	25 hours ago	Created		6379/tcp
k8s_container1_podsandbox1_redhat.test.crio_redhat-test-crio_1						

```
$ podman ps -a -s
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES	SIZE
02f65160e14ca	redis:alpine	"redis-server"	20 hours ago	Exited (-1)	20 hours ago	6379/tcp	
k8s_podsandbox1-redis_podsandbox1_redhat.test.crio_redhat-test-crio_0							27.49 MB
69ed779d8ef9f	redis:alpine	"redis-server"	25 hours ago	Created			

ps

Print a list of containers

SEE ALSO

podman(1), buildah(1), crio(8)

HISTORY

August 2017, Originally compiled by Urvashi Mohnani umohnani@redhat.com
[?mailto:umohnani@redhat.com?](mailto:umohnani@redhat.com)

podman-ps(1)