



## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'podman-container-rm.1' command***

***\$ man podman-container-rm.1***

podman-rm(1)                      General Commands Manual                      podman-rm(1)

### NAME

podman-rm - Remove one or more containers

### SYNOPSIS

podman rm [options] container

podman container rm [options] container

### DESCRIPTION

podman rm will remove one or more containers from the host. The container name or ID can be used. This does not remove images. Running or unusable containers will not be removed without the -f option.

### OPTIONS

--all, -a

Remove all containers. Can be used in conjunction with -f as well.

--cidfile=file

Read container ID from the specified file and rm the container. Can be specified multiple times.

--depend

Remove selected container and recursively remove all containers that depend on it.

--filter=filter

Filter what containers remove. 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 (ac? ?

? ? cepts regex) ?

??

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

? ? assigned to a container ?

??

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

??

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

? ? 'created', 'exited', ?

? ? 'paused', 'running', 'un? ?

? ? known' ?

??

?ancestor ? [ImageName] Image or descen? ?

? ? dant used to create container ?

??

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

? ? created before this container ?

??

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

? ? created since this container ?

??

?volume ? [VolumeName] or [Mountpoint? ?

? ? Destination] Volume mounted ?

? ? in container ?

??

?health ? [Status] healthy or unhealthy ?

??

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

? ? ID of pod ?

??

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

? ? network ?

??

--force, -f

Force the removal of running and paused containers. Forcing a container removal also removes containers from container storage even if the container is not known to podman. Containers could have been created by a different container engine. In addition, forcing can be used to remove unusable containers, e.g. containers whose OCI runtime has become unavailable.

--ignore, -i

Ignore errors when specified containers are not in the container store. A user might have decided to manually remove a container which would lead to a failure during the ExecStop directive of a systemd service referencing that container. Further ignore when the specified --cidfile does not exist as it may have already been removed along with the container.

--latest, -l

Instead of providing the container name or ID, use the last created container. Note: the last started container could be from other users of Podman on the host machine. (This option is not available with the remote Podman client, including Mac and Windows (excluding WSL2 machines))

--time, -t=seconds

Seconds to wait before forcibly stopping the container. The --force option must be specified to use the --time option.

--volumes, -v

Remove anonymous volumes associated with the container. This does not include named volumes created with `podman volume create`, or the `--volume` option of `podman run` and `podman create`.

## EXAMPLE

Remove container with a given name

```
$ podman rm mywebserver
```

Remove container with a given name and all of the containers that depend on it

```
$ podman rm --depend mywebserver
```

Remove multiple containers with given names or IDs

```
$ podman rm mywebserver myflaskserver 860a4b23
```

Remove multiple containers with IDs read from files

```
$ podman rm --cidfile ./cidfile-1 --cidfile /home/user/cidfile-2
```

Forcibly remove container with a given ID

```
$ podman rm -f 860a4b23
```

Remove all containers regardless of the run state

```
$ podman rm -f -a
```

Forcibly remove the last created container

```
$ podman rm -f --latest
```

## Exit Status

- 0 All specified containers removed
- 1 One of the specified containers did not exist, and no other failures
- 2 One of the specified containers is paused or running
- 125 The command fails for any other reason

## SEE ALSO

`podman(1)`, `crio(8)`

## HISTORY

August 2017, Originally compiled by Ryan Cole [rycole@redhat.com](mailto:rycole@redhat.com)  
[?mailto:rycole@redhat.com?](mailto:rycole@redhat.com)

`podman-rm(1)`