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

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

$ man podman-unshare.1

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

NAME

podman-unshare - Run a command inside of a modified user namespace

SYNOPSIS

podman unshare [options] [command]

DESCRIPTION

Launches a process (by default, $SHELL) in a new user namespace. The
user namespace is configured so that the invoking user's UID and pri?
mary GID appear to be UID 0 and GID 0, respectively. Any ranges which
match that user and group in /etc/subuid and /etc/subgid are also
mapped in as themselves with the help of the newuidmap(1) and
newgidmap(1) helpers.

podman unshare is useful for troubleshooting unprivileged operations
and for manually clearing storage and other data related to images and
containers.

It is also useful to use the podman mount command. If an unprivileged
user wants to mount and work with a container, then they need to exe?
cute podman unshare. Executing podman mount fails for unprivileged

users unless the user is running inside a podman unshare session.

The unshare session defines two environment variables:

? CONTAINERS_GRAPHROOT: the path to the persistent container's data.

? CONTAINERS_RUNROOT: the path to the volatile container's data.

IMPORTANT: This command is not available with the remote Podman client.

OPTIONS

--help, -h

Print usage statement

--rootless-netns

Join the rootless network namespace used for netavark networking. It can be used to connect to a rootless container via IP address (bridge networking). This is otherwise not possible from the host network name? space.

Exit Codes

The exit code from podman unshare gives information about why the con? tainer failed to run or why it exited. When podman unshare commands exit with a non-zero code, the exit codes follow the chroot standard, see below:

125 The error is with podman itself

```
$ podman unshare --foo; echo $?
Error: unknown flag: --foo
125
```

126 Executing a contained command and the command cannot be invoked

```
$ podman unshare /etc; echo $?
Error: fork/exec /etc: permission denied
126
```

127 Executing a contained command and the command cannot be found

```
$ podman unshare foo; echo $?
Error: fork/exec /usr/bin/bogus: no such file or directory
127
```

Exit code contained command exit code

```
$ podman unshare /bin/sh -c 'exit 3'; echo $?
```

3

EXAMPLE

```
$ podman unshare id
uid=0(root) gid=0(root) groups=0(root),65534(nobody)
$ podman unshare cat /proc/self/uid_map /proc/self/gid_map
         0       1000          1
         1      10000      65536
         0       1000          1
         1      10000      65536
$ podman unshare --rootless-netns ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: tap0: <BROADCAST,UP,LOWER_UP> mtu 65520 qdisc fq_codel state UNKNOWN group default qlen 1000
    link/ether aa:8c:0b:73:98:f6 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.100/24 brd 10.0.2.255 scope global tap0
       valid_lft forever preferred_lft forever
    inet6 fd00::a88c:bff:fe73:98f6/64 scope global dynamic mngtmpaddr
       valid_lft 86389sec preferred_lft 14389sec
    inet6 fe80::a88c:bff:fe73:98f6/64 scope link
       valid_lft forever preferred_lft forever
```

SEE ALSO

podman(1), podman-mount(1), namespaces(7), newuidmap(1), newgidmap(1),
user_namespaces(7)