Rocky Enterprise Linux 9.2 Manual Pages on command 'buildah.1'

$ man buildah.1

buildah(1) General Commands Manual buildah(1)

NAME

Buildah - A command line tool that facilitates building OCI container images.

SYNOPSIS

buildah [OPTIONS] COMMAND [ARG...]

DESCRIPTION

The Buildah package provides a command line tool which can be used to:

* Create a working container, either from scratch or using an image as a starting point.

* Mount a working container's root filesystem for manipulation.

* Unmount a working container's root filesystem.

* Use the updated contents of a container's root filesystem as a filesystem layer to create a new image.

* Delete a working container or an image.

* Rename a local container.

OPTIONS

--cgroup-manager=manager

The CGroup manager to use for container cgroups. Supported values are cgroupfs or systemd. Default is systemd unless overridden in the con?

tainers.conf file.

Note: Setting this flag can cause certain commands to break when called on containers previously created by the other CGroup manager type.

Note: CGroup manager is not supported in rootless mode when using CGroups Version V1.

--log-level value

The log level to be used. Either "trace", "debug", "info", "warn", "er?ror", "fatal", or "panic", defaulting to "warn".

--help, -h

Show help

--registries-conf path

Pathname of the configuration file which specifies which container reg?istries should be consulted when completing image names which do not include a registry or domain portion. It is not recommended that this option be used, as the default behavior of using the system-wide con?figuration (/etc/containers/registries.conf) is most often preferred.

--registries-conf-dir path

Pathname of the directory which contains configuration snippets which specify registries which should be consulted when completing image names which do not include a registry or domain portion. It is not recommended that this option be used, as the default behavior of using the system-wide configuration (/etc/containers/registries.d) is most often preferred.

--root value

Storage root dir (default: "/var/lib/containers/storage" for UID 0, "$HOME/.local/share/containers/storage" for other users) Default root dir is configured in /etc/containers/storage.conf

--runroot value

Storage state dir (default: "/run/containers/storage" for UID 0, "/run/user/$UID" for other users) Default state dir is configured in /etc/containers/storage.conf

--short-name-alias-conf path

Pathname of the file which contains cached mappings between short image

names and their corresponding fully-qualified names. It is used for mapping from names of images specified using short names like "ubi8" which don't include a registry component and a corresponding fully-specified name which includes a registry and any other components, such as "registry.access.redhat.com/ubi8". It is not recommended that this option be used, as the default behavior of using the system-wide cache (/var/cache/containers/short-name-aliases.conf) or per-user cache ($HOME/.cache/containers/short-name-aliases.conf) to supplement system-wide defaults is most often preferred.

--storage-driver value

Storage driver. The default storage driver for UID 0 is configured in /etc/containers/storage.conf ($HOME/.config/containers/storage.conf in rootless mode), and is vfs for other users. The STORAGE_DRIVER envi? ronment variable overrides the default. The --storage-driver specified driver overrides all.

Examples: "overlay", "devicemapper", "vfs"

Overriding this option will cause the storage-opt settings in /etc/con? tainers/storage.conf to be ignored. The user must specify additional options via the --storage-opt flag.

--storage-opt value

Storage driver option, Default storage driver options are configured in /etc/containers/storage.conf ($HOME/.config/containers/storage.conf in rootless mode). The STORAGE_OPTS environment variable overrides the de? fault. The --storage-opt specified options overrides all.

--userns-gid-map mapping

Directly specifies a GID mapping which should be used to set ownership, at the filesystem level, on the working container's contents. Commands run when handling RUN instructions will default to being run in their own user namespaces, configured using the UID and GID maps.

Entries in this map take the form of one or more colon-separated triples of a starting in-container GID, a corresponding starting host-level GID, and the number of consecutive IDs which the map entry repre? sents.

This option overrides the remap-gids setting in the options section of /etc/containers/storage.conf.

If this option is not specified, but a global --userns-gid-map setting is supplied, settings from the global option will be used.

If none of --userns-uid-map-user, --userns-gid-map-group, or --userns-gid-map are specified, but --userns-uid-map is specified, the GID map will be set to use the same numeric values as the UID map.

NOTE: When this option is specified by a rootless user, the specified mappings are relative to the rootless usernamespace in the container, rather than being relative to the host as it would be when run rootful.

--userns-uid-map mapping

Directly specifies a UID mapping which should be used to set ownership, at the filesystem level, on the working container's contents. Commands run when handling RUN instructions will default to being run in their own user namespaces, configured using the UID and GID maps.

Entries in this map take the form of one or more colon-separated triples of a starting in-container UID, a corresponding starting host-level UID, and the number of consecutive IDs which the map entry repre? sents.

This option overrides the remap-uids setting in the options section of /etc/containers/storage.conf.

If this option is not specified, but a global --userns-uid-map setting is supplied, settings from the global option will be used.

If none of --userns-uid-map-user, --userns-gid-map-group, or --userns-uid-map are specified, but --userns-gid-map is specified, the UID map will be set to use the same numeric values as the GID map.

NOTE: When this option is specified by a rootless user, the specified mappings are relative to the rootless usernamespace in the container, rather than being relative to the host as it would be when run rootful.

--version, -v

Print the version

Environment Variables

Buildah can set up environment variables from the env entry in the [en?

gine] table in the containers.conf(5). These variables can be overrid?

den by passing environment variables before the buildah commands.

COMMANDS

Command	Man Page	Description
add	buildah-add(1)	Add the contents of a file, URL, or a directory to the container.
build	buildah-build(1)	Builds an OCI image using instructions in one or more Containerfiles.
commit	buildah-commit(1)	Create an image from a working container.
config	buildah-config(1)	Update image configuration settings.
containers	buildah-containers(1)	List the working containers and their base images.
copy	buildah-copy(1)	Copies the contents of a file, URL, or directory into a container's working directory.
from	buildah-from(1)	Creates a new working container, either from scratch or using a specified image as a starting point.
images	buildah-images(1)	List images in local storage.

??

?info ? buildah-info(1) ? Display Buildah system infor? ?

? ? ? mation. ?

??

?inspect ? buildah-inspect(1) ? Inspects the configuration of ?

? ? ? a container or image ?

??

?login ? buildah-login(1) ? Login to a container registry. ?

??

?logout ? buildah-logout(1) ? Logout of a container registry ?

??

?manifest ? buildah-manifest(1) ? Create and manipulate manifest ?

? ? ? lists and image indexes. ?

??

?mount ? buildah-mount(1) ? Mount the working container's ?

? ? ? root filesystem. ?

??

?prune ? buildah-prune(1) ? Cleanup intermediate images as ?

? ? ? well as build and mount cache. ?

??

?pull ? buildah-pull(1) ? Pull an image from the speci? ?

? ? ? fied location. ?

??

?push ? buildah-push(1) ? Push an image from local stor? ?

? ? ? age to elsewhere. ?

??

?rename ? buildah-rename(1) ? Rename a local container. ?

??

?rm ? buildah-rm(1) ? Removes one or more working ?

? ? ? containers. ?

??

?rmi ? buildah-rmi(1) ? Removes one or more images. ?

??

?run ? buildah-run(1) ? Run a command inside of the ?

? ? ? container. ?

??

?source ? buildah-source(1) ? Create, push, pull and manage ?

? ? ? source images and associated ?

? ? ? source artifacts. ?

??

?tag ? buildah-tag(1) ? Add an additional name to a ?

? ? ? local image. ?

??

?umount ? buildah-umount(1) ? Unmount a working container's ?

? ? ? root file system. ?

??

?unshare ? buildah-unshare(1) ? Launch a command in a user ?

? ? ? namespace with modified ID ?

? ? ? mappings. ?

??

?version ? buildah-version(1) ? Display the Buildah Version ?

? ? ? Information ?

??

Files

storage.conf (/etc/containers/storage.conf)

storage.conf is the storage configuration file for all tools using con?

tainers/storage

The storage configuration file specifies all of the available container

storage options for tools using shared container storage.

mounts.conf (/usr/share/containers/mounts.conf and optionally /etc/con?

tainers/mounts.conf)

The mounts.conf files specify volume mount files or directories that

are automatically mounted inside containers when executing the buildah

run or buildah build commands. Container processes can then use this

content. The volume mount content does not get committed to the final

image.

Usually these directories are used for passing secrets or credentials required by the package software to access remote package repositories. For example, a mounts.conf with the line "/usr/share/rhel/se?crets:/run/secrets", the content of /usr/share/rhel/secrets directory is mounted on /run/secrets inside the container. This mountpoint al?lows Red Hat Enterprise Linux subscriptions from the host to be used within the container. It is also possible to omit the destination if it's equal to the source path. For example, specifying /var/lib/se?crets will mount the directory into the same container destination path /var/lib/secrets.

Note this is not a volume mount. The content of the volumes is copied into container storage, not bind mounted directly from the host.

registries.conf (/etc/containers/registries.conf)

registries.conf is the configuration file which specifies which con?tainer registries should be consulted when completing image names which do not include a registry or domain portion.

registries.d (/etc/containers/registries.d)

Directory which contains configuration snippets which specify reg?istries which should be consulted when completing image names which do not include a registry or domain portion.

SEE ALSO

containers.conf(5), containers-mounts.conf(5), newuidmap(1), newgidmap(1), containers-registries.conf(5), containers-storage.conf(5)

HISTORY

December 2017, Originally compiled by Tom Sweeney tsweeney@redhat.com ?mailto:tsweeney@redhat.com?