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

$ man buildah-push.1

buildah-push(1) General Commands Manual buildah-push(1)

NAME

buildah-push - Push an image, manifest list or image index from local storage to else?

where.

SYNOPSIS

buildah push [options] image [destination]

DESCRIPTION

Pushes an image from local storage to a specified destination, decompressing and recom?

pessing layers as needed.

imageID

Image stored in local container/storage

DESTINATION

DESTINATION is the location the container image is pushed to. It supports all transports

from containers-transports(5) (see examples below). If no transport is specified, the

docker (i.e., container registry) transport is used.

OPTIONS

--all

If specified image is a manifest list or image index, push the images in addition to the

list or index itself.

--authfile path

Path of the authentication file. Default is ${XDG_\RUNTIME_DIR}/containers/auth.json. If

XDG_RUNTIME_DIR is not set, the default is /run/containers/$UID/auth.json. This file is

created using using buildah login.

If the authorization state is not found there, $HOME/.docker/config.json is checked, which

is set using docker login.

Note: You can also override the default path of the authentication file by setting the

REGISTRY_AUTH_FILE environment variable. export REGISTRY_AUTH_FILE=path

--cert-dir path

Use certificates at path (*.crt, *.cert, *.key) to connect to the registry. The default

certificates directory is /etc/containers/certs.d.

--creds creds

The [username[:password]] to use to authenticate with the registry if required. If one or

both values are not supplied, a command line prompt will appear and the value can be en?

tered. The password is entered without echo.

--digestfile Digestfile

After copying the image, write the digest of the resulting image to the file.

--disable-compression, -D

Don't compress copies of filesystem layers which will be pushed.

--encryption-key key

The [protocol:keyfile] specifies the encryption protocol, which can be JWE (RFC7516), PGP

(RFC4880), and PKCS7 (RFC2315) and the key material required for image encryption. For in?

stance, jwe:/path/to/key.pem or pgp:admin@example.com or pkcs7:/path/to/x509-file.

--encrypt-layer layer(s)

Layer(s) to encrypt: 0-indexed layer indices with support for negative indexing (e.g. 0 is

the first layer, -1 is the last layer). If not defined, will encrypt all layers if encryp?

tion-key flag is specified.

--format, -f

Manifest Type (oci, v2s2, or v2s1) to use when pushing an image. (default is manifest type

of the source image, with fallbacks)

--quiet, -q

When writing the output image, suppress progress output.

--remove-signatures

Don't copy signatures when pushing images.

--rm

When pushing a the manifest list or image index, delete them from local storage if pushing

succeeds.

--sign-by fingerprint

Sign the pushed image using the GPG key that matches the specified fingerprint.

--tls-verify bool-value

Require HTTPS and verification of certificates when talking to container registries (de?
faults to true). TLS verification cannot be used when talking to an insecure registry.

EXAMPLE

This example pushes the image specified by the imageID to a local directory in docker for?
mat.

```
# buildah push imageID dir:/path/to/image
```

This example pushes the image specified by the imageID to a local directory in oci format.

```
# buildah push imageID oci:/path/to/layout:image:tag
```

This example pushes the image specified by the imageID to a tar archive in oci format.

```
# buildah push imageID oci-archive:/path/to/archive:image:tag
```

This example pushes the image specified by the imageID to a container registry named reg?
istry.example.com.

```
# buildah push imageID docker://registry.example.com/repository:tag
```

This example pushes the image specified by the imageID to a container registry named reg?
istry.example.com and saves the digest in the specified digestfile.

```
#  buildah  push  --digestfile=/tmp/mydigest imageID docker://registry.example.com/reposi?
tory:tag
```

This example works like docker push, assuming registry.example.com/my_image is a local im?
age.

```
# buildah push registry.example.com/my_image
```

This example pushes the image specified by the imageID to a private container registry
named registry.example.com with authentication from /tmp/auths/myauths.json.

```
#  buildah  push  --authfile   /tmp/auths/myauths.json   imageID   docker://registry.exam?
ple.com/repository:tag
```

This example pushes the image specified by the imageID and puts into the local docker con?
tainer store.

```
# buildah push imageID docker-daemon:image:tag
```

This example pushes the image specified by the imageID and puts it into the registry on
the localhost while turning off tls verification.

```
 # buildah push --tls-verify=false imageID docker://localhost:5000/my-imageID
```

This example pushes the image specified by the imageID and puts it into the registry on the localhost using credentials and certificates for authentication.

```
 # buildah push --cert-dir    /auth  --tls-verify=true  --creds=username:password  imageID
docker://localhost:5000/my-imageID
```

ENVIRONMENT

BUILD_REGISTRY_SOURCES

BUILD_REGISTRY_SOURCES, if set, is treated as a JSON object which contains lists of reg?istry names under the keys insecureRegistries, blockedRegistries, and allowedRegistries.

When pushing an image to a registry, if the portion of the destination image name that corresponds to a registry is compared to the items in the blockedRegistries list, and if it matches any of them, the push attempt is denied. If there are registries in the al?lowedRegistries list, and the portion of the name that corresponds to the registry is not in the list, the push attempt is denied.

TMPDIR The TMPDIR environment variable allows the user to specify where temporary files are stored while pulling and pushing images. Defaults to '/var/tmp'.

FILES

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

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

policy.json (/etc/containers/policy.json)

Signature policy file. This defines the trust policy for container images. Controls which container registries can be used for image, and whether or not the tool should trust the images.

SEE ALSO

buildah(1), buildah-login(1), containers-policy.json(5), docker-login(1), containers-reg?istries.conf(5), buildah-manifest(1)

buildah June 2017 buildah-push(1)