



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 'docker-manifest-push.1'

\$ man docker-manifest-push.1

podman-manifest-push(1)() podman-manifest-push(1)()

NAME

podman-manifest-push - Push a manifest list or image index to a registry

SYNOPSIS

podman manifest push [options] listnameorindexname [destination]

DESCRIPTION

Pushes a manifest list or image index to a registry.

RETURN VALUE

The list image's ID and the digest of the image's manifest.

OPTIONS

--all

Push the images mentioned in the manifest list or image index, in addition to the list or index itself. (Default true)

--authfile=path

Path of the authentication file. Default is `${XDG_RUNTIME_DIR}/containers/auth.json`, which is set using `podman login`. If the authorization state is not found there, `$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 (`*.cert`, `*.key`) to connect to the registry. (Default: `/etc/containers/certs.d`) Please refer to `containers-certs.d(5)` for details. (This option is not available with the remote Podman client)

`--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 entered. The password is entered without echo.

`--digestfile=Digestfile`

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

`--format, -f=format`

Manifest list type (oci or v2s2) to use when pushing the list (default is oci).

`--quiet, -q`

When writing the manifest, suppress progress output

`--rm`

Delete the manifest list or image index from local storage if pushing succeeds.

`--remove-signatures`

Don't copy signatures when pushing images.

`--sign-by=fingerprint`

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

`--tls-verify`

Require HTTPS and verify certificates when talking to container registries. (defaults to true)

DESTINATION

The DESTINATION is a location to store container images

The Image "DESTINATION" uses a "transport":"details" format.

If a transport is not given, podman push will attempt to push to a registry.

Multiple transports are supported:

`dir:path`

An existing local directory path storing the manifest, layer tarballs and signatures as individual files. This is a non-standardized format, primarily useful for debugging or noninvasive container inspection.

```
$ podman manifest push mylist:v1.11 dir:/tmp/mylist
```

`docker://docker-reference`

An image in a registry implementing the "Docker Registry HTTP API V2". By default, uses the authorization state in `$XDG_RUNTIME_DIR/containers/auth.json`, which is set using (pod?)

man login). If the authorization state is not found there, `$HOME/.docker/config.json` is checked, which is set using `(docker login)`.

```
$ podman manifest push mylist:v1.11 docker://registry.example.org/mylist:v1.11  
docker-archive:path[:docker-reference]
```

An image is stored in the docker save formatted file. `docker-reference` is only used when creating such a file, and it must not contain a digest.

```
$ podman manifest push mylist:v1.11 docker-archive:/tmp/mylist  
docker-daemon:docker-reference
```

An image in `docker-reference` format stored in the docker daemon internal storage. `docker-reference` must contain a tag.

```
$ podman manifest push mylist:v1.11 docker-daemon:registry.example.org/mylist:v1.11  
oci-archive:path:tag
```

An image tag in a directory compliant with "Open Container Image Layout Specification" at path.

```
$ podman manifest push mylist:v1.11 oci-archive:/tmp/mylist
```

EXAMPLE

```
podman manifest push mylist:v1.11 docker://registry.example.org/mylist:v1.11
```

SEE ALSO

`podman(1)`, `podman-manifest(1)`, `podman-manifest-add(1)`, `podman-manifest-create(1)`, `podman-manifest-inspect(1)`, `podman-manifest-remove(1)`, `podman-rmi(1)`, `containers-certs.d(5)`
`podman-manifest-push(1)()`