



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

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'flatpak-override.1' command

\$ man flatpak-override.1

FLATPAK OVERRIDE(1) flatpak override FLATPAK OVERRIDE(1)

NAME

flatpak-override - Override application requirements

SYNOPSIS

flatpak override [OPTION...] [APP]

DESCRIPTION

Overrides the application specified runtime requirements. This can be used to grant a sandboxed application more or less resources than it requested.

By default the application gets access to the resources it requested when it is started. But the user can override it on a particular instance by specifying extra arguments to flatpak run, or every time by using flatpak override.

If the application id is not specified then the overrides affect all applications, but the per-application overrides can override the global overrides.

Unless overridden with the --user or --installation options, this command changes the default system-wide installation.

OPTIONS

The following options are understood:

-h, --help

Show help options and exit.

--user

Update a per-user installation.

`--system`

Update the default system-wide installation.

`--installation=NAME`

Updates a system-wide installation specified by NAME among those defined in /etc/flatpak/installations.d/. Using

`--installation=default` is equivalent to using `--system`.

`--share=SUBSYSTEM`

Share a subsystem with the host session. This overrides the Context section from the application metadata. SUBSYSTEM must be one of: network, ipc. This option can be used multiple times.

`--unshare=SUBSYSTEM`

Don't share a subsystem with the host session. This overrides the Context section from the application metadata. SUBSYSTEM must be one of: network, ipc. This option can be used multiple times.

`--socket=SOCKET`

Expose a well-known socket to the application. This overrides to the Context section from the application metadata. SOCKET must be one of: x11, wayland, fallback-x11, pulseaudio, system-bus, session-bus, ssh-auth, pcsc, cups. This option can be used multiple times.

`--nosocket=SOCKET`

Don't expose a well-known socket to the application. This overrides to the Context section from the application metadata. SOCKET must be one of: x11, wayland, fallback-x11, pulseaudio, system-bus, session-bus, ssh-auth, pcsc, cups. This option can be used multiple times.

`--device=DEVICE`

Expose a device to the application. This overrides to the Context section from the application metadata. DEVICE must be one of: dri, kvm, shm, all. This option can be used multiple times.

`--nodevice=DEVICE`

Don't expose a device to the application. This overrides to the

Context section from the application metadata. DEVICE must be one of: dri, kvm, shm, all. This option can be used multiple times.

`--allow=FEATURE`

Allow access to a specific feature. This updates the [Context] group in the metadata. FEATURE must be one of: devel, multiarch, bluetooth, canbus, per-app-dev-shm. This option can be used multiple times.

See flatpak-build-finish(1) for the meaning of the various features.

`--disallow=FEATURE`

Disallow access to a specific feature. This updates the [Context] group in the metadata. FEATURE must be one of: devel, multiarch, bluetooth, canbus, per-app-dev-shm. This option can be used multiple times.

`--filesystem=FILESYSTEM`

Allow the application access to a subset of the filesystem. This overrides to the Context section from the application metadata.

FILESYSTEM can be one of: home, host, host-os, host-etc, xdg-desktop, xdg-documents, xdg-download, xdg-music, xdg-pictures, xdg-public-share, xdg-templates, xdg-videos, xdg-run, xdg-config, xdg-cache, xdg-data, an absolute path, or a homedir-relative path like ~/dir or paths relative to the xdg dirs, like xdg-download/subdir. The optional :ro suffix indicates that the location will be read-only. The optional :create suffix indicates that the location will be read-write and created if it doesn't exist. This option can be used multiple times. See the "[Context] filesystems" list in flatpak-metadata(5) for details of the meanings of these filesystems.

`--nofilesystem=FILESYSTEM`

Undo the effect of a previous `--filesystem=FILESYSTEM` in the app's manifest or a lower-precedence layer of overrides, and/or remove a previous `--filesystem=FILESYSTEM` from this layer of overrides. This overrides the Context section of the application metadata.

FILESYSTEM can take the same values as for `--filesystem`, but the `:ro` and `:create` suffixes are not used here. This option can be used multiple times.

This option does not prevent access to a more narrowly-scoped `--filesystem`. For example, if an application has the equivalent of `--filesystem=xdg-config/MyApp` in its manifest or as a system-wide override, and flatpak override `--user --nofilesystem=home` as a per-user override, then it will be prevented from accessing most of the home directory, but it will still be allowed to access `$XDG_CONFIG_HOME/MyApp`.

As a special case, `--nofilesystem=host:reset` will ignore all `--filesystem` permissions inherited from the app manifest or a lower-precedence layer of overrides, in addition to having the behaviour of `--nofilesystem=host`.

`--add-policy=SUBSYSTEM.KEY=VALUE`

Add generic policy option. For example,

"`--add-policy=subsystem.key=v1 --add-policy=subsystem.key=v2`" would map to this metadata:

```
[Policy subsystem]
```

```
key=v1;v2;
```

This option can be used multiple times.

`--remove-policy=SUBSYSTEM.KEY=VALUE`

Remove generic policy option. This option can be used multiple times.

`--env=VAR=VALUE`

Set an environment variable in the application. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--unset-env=VAR`

Unset an environment variable in the application. This overrides the unset-environment entry in the [Context] group of the metadata, and the [Environment] group. This option can be used multiple times.

`--env-fd=FD`

Read environment variables from the file descriptor FD, and set them as if via `--env`. This can be used to avoid environment variables and their values becoming visible to other users.

Each environment variable is in the form VAR=VALUE followed by a zero byte. This is the same format used by `env -0` and `/proc/*/environ`.

`--own-name=NAME`

Allow the application to own the well-known name NAME on the session bus. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--talk-name=NAME`

Allow the application to talk to the well-known name NAME on the session bus. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--no-talk-name=NAME`

Don't allow the application to talk to the well-known name NAME on the session bus. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--system-own-name=NAME`

Allow the application to own the well known name NAME on the system bus. If NAME ends with `.*`, it allows the application to own all matching names. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--system-talk-name=NAME`

Allow the application to talk to the well known name NAME on the system bus. If NAME ends with `.*`, it allows the application to talk to all matching names. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--system-no-talk-name=NAME`

Don't allow the application to talk to the well known name NAME on the system bus. If NAME ends with `.*`, it allows the application to talk to all matching names. This overrides to the Context section

from the application metadata. This option can be used multiple times.

`--persist=FILENAME`

If the application doesn't have access to the real homedir, make the (homedir-relative) path FILENAME a bind mount to the corresponding path in the per-application directory, allowing that location to be used for persistent data. This overrides to the Context section from the application metadata. This option can be used multiple times.

`--reset`

Remove overrides. If an APP is given, remove the overrides for that application, otherwise remove the global overrides.

`--show`

Shows overrides. If an APP is given, shows the overrides for that application, otherwise shows the global overrides.

`-v, --verbose`

Print debug information during command processing.

`--ostree-verbose`

Print OSTree debug information during command processing.

EXAMPLES

```
$ flatpak override --nosocket=wayland org.gnome.gedit
```

```
$ flatpak override --filesystem=home org.mozilla.Firefox
```

SEE ALSO

flatpak(1), flatpak-run(1)

flatpak

FLATPAK OVERRIDE(1)