



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 'ostree.repo-config.5'

\$ man ostree.repo-config.5

OSTREE.REPO-CONFI(5) ostree.repo-config OSTREE.REPO-CONFI(5)

NAME

ostree.repo-config - OSTree repository configuration

DESCRIPTION

The config file in an OSTree repository is a "keyfile" in the XDG Desktop Entry Specification[1] format. It has several global flags, as well as zero or more remote entries which describe how to access remote repositories.

See ostree.repo(5) for more information about OSTree repositories.

[CORE] SECTION OPTIONS

Repository-global options. The following entries are defined:

mode

One of bare, bare-user, bare-user-only, or archive-z2 (note that archive is used everywhere else.)

repo_version

Currently, this must be set to 1.

auto-update-summary

Boolean value controlling whether or not to automatically update

the summary file after any ref is added, removed, or updated. Other modifications which may render a summary file stale (like static deltas, or collection IDs) do not currently trigger an auto-update.

commit-update-summary

This option is deprecated. Use auto-update-summary instead, for which this option is now an alias.

fsync

Boolean value controlling whether or not to ensure files are on stable storage when performing operations such as commits, pulls, and checkouts. Defaults to true.

If you disable fsync, OSTree will no longer be robust against kernel crashes or power loss.

You might choose to disable this for local development repositories, under the assumption they can be recreated from source. Similarly, you could disable for a mirror where you could re-pull.

For the system repository, you might choose to disable fsync if you have uninterruptable power supplies and a well tested kernel.

per-object-fsync

By default, OSTree will batch fsync() after writing everything; however, this can cause latency spikes for other processes which are also invoking fsync(). Turn on this boolean to reduce potential latency spikes, at the cost of slowing down OSTree updates. You most likely want this on by default for "background" OS updates.

min-free-space-percent

Integer percentage value (0-99) that specifies a minimum percentage of total space (in blocks) in the underlying filesystem to keep free. The default value is 3, which is enforced when neither this option nor min-free-space-size are set.

If min-free-space-size is set to a non-zero value, min-free-space-percent is ignored. Note that, min-free-space-percent is not enforced on metadata objects. It is assumed that metadata objects are relatively small in size compared

to content objects and thus kept outside the scope of this option.

min-free-space-size

Value (in power-of-2 MB, GB or TB) that specifies a minimum space in the underlying filesystem to keep free. Examples of acceptable values: 500MB (524 288 000 bytes), 1GB (1 073 741 824 bytes), 1TB (1 099 511 627 776 bytes).

If this option is set to a non-zero value, and min-free-space-percent is also set, this option takes priority.

Note that, min-free-space-size is not enforced on metadata objects.

It is assumed that metadata objects are relatively small in size compared to content objects and thus kept outside the scope of this option.

add-remotes-config-dir

Boolean value controlling whether new remotes will be added in the remotes configuration directory. Defaults to true for system ostree repositories. When this is false, remotes will be added in the repository's config file.

This only applies to repositories that use a remotes configuration directory such as system ostree repositories, which use /etc/ostree/remotes.d. Non-system repositories do not use a remotes configuration directory unless one is specified when the repository is opened.

payload-link-threshold

An integer value that specifies a minimum file size for creating a payload link. By default it is disabled.

collection-id

A reverse DNS domain name under your control, which enables peer to peer distribution of refs in this repository. See the --collection-id section in ostree-init(1)

locking

Boolean value controlling whether or not OSTree does repository locking internally. This uses file locks and is hence for multiple process exclusion (e.g. Flatpak and OSTree writing to the same

repository separately). This is enabled by default since 2018.5.

lock-timeout-secs

Integer value controlling the number of seconds to block while attempting to acquire a lock (see above). A value of -1 means block indefinitely. The default value is 30.

default-repo-finders

Semicolon separated default list of finders (sources for refs) to use when pulling. This can be used to disable pulling from mounted filesystems, peers on the local network, or the Internet. However note that it only applies when a set of finders isn't explicitly specified, either by a consumer of libostree API or on the command line. Possible values: config, lan, and mount (or any combination thereof). If unset, this defaults to config;mount; (since the LAN finder is costly).

no-deltas-in-summary

Boolean value controlling whether OSTree should skip putting an index of available deltas in the summary file. Defaults to false. Since 2020.7 OSTree can use delta indexes outside the summary file, making the summary file smaller (especially for larger repositories). However by default we still create the index in the summary file to make older clients work. If you know all clients will be 2020.7 later you can enable this to save network bandwidth.

[REMOTE NAME"] SECTION OPTIONS"

Describes a remote repository location.

url

Must be present; declares URL for accessing metadata and content for remote. See also contenturl. The supported schemes are documented below.

contenturl

Declares URL for accessing content (filez, static delta parts). When specified, url is used just for metadata: summary, static delta "superblocks".

branches

A list of strings. Represents the default configured branches to fetch from the remote when no specific branches are requested during a pull operation.

proxy

A string value, if given should be a URL for a HTTP proxy to use for access to this repository.

gpg-verify

A boolean value, defaults to true. Controls whether or not OSTree will require commits to be signed by a known GPG key. For more information, see the ostree(1) manual under GPG.

gpg-verify-summary

A boolean value, defaults to false. Controls whether or not OSTree will check if the summary is signed by a known GPG key. For more information, see the ostree(1) manual under GPG.

tls-permissive

A boolean value, defaults to false. By default, server TLS certificates will be checked against the system certificate store. If this variable is set, any certificate will be accepted.

tls-client-cert-path

Path to file for client-side certificate, to present when making requests to this repository.

tls-client-key-path

Path to file containing client-side certificate key, to present when making requests to this repository.

tls-ca-path

Path to file containing trusted anchors instead of the system CA database.

http2

A boolean value, defaults to true. By default, libostree will use HTTP2; setting this to false will disable it. May be useful to work around broken servers.

unconfigured-state

If set, pulls from this remote will fail with the configured text.

This is intended for OS vendors which have a subscription process to access content.

custom-backend

If set, pulls from this remote via libostree will fail with an error that mentions the value. It is recommended to make this a software identifier token (e.g. "examplecorp-fetcher"), not freeform text ("ExampleCorp Fetcher"). This is intended to be used by higher level software that wants to fetch ostree commits via some other mechanism, while still reusing the core libostree infrastructure around e.g. signatures.

[SYSROOT] SECTION OPTIONS

Options for the sysroot, which contains the OSTree repository, deployments, and stateroots. The following entries are defined:

bootloader

Configure the bootloader that OSTree uses when deploying the sysroot. This may take the values `bootloader=none`, `bootloader=auto`, `bootloader=grub2`, `bootloader=syslinux`, `bootloader=uboot` or `bootloader=zipl`. Default is `auto`.

If `none`, then OSTree will generate only BLS (Boot Loader Specification) fragments in `sysroot/boot/loader/entries/` for the deployment.

If `auto`, then in addition to generating BLS fragments, OSTree will dynamically check for the existence of `grub2`, `uboot`, and `syslinux` bootloaders. If one of the bootloaders is found, then OSTree will generate a config for the bootloader found. For example, `grub2-mkconfig` is run for the `grub2` case.

A specific bootloader type may also be explicitly requested by choosing `grub2`, `syslinux`, `uboot` or `zipl`.

bls-append-except-default

A semicolon separated string list of key-value pairs. For example: `bls-append-except-default=key1=value1;key2=value2`. These key-value pairs will be injected into the generated BLS fragments of the non-default deployments. In other words, the BLS fragment of the

default deployment will be unaffected by `bls-append-except-default`.

/ETC/OSTREE/REMOTES.D

In addition to the `/ostree/repo/config` file, remotes may also be specified in `/etc/ostree/remotes.d`. The remote configuration file must end in `.conf`; files whose name does not end in `.conf` will be ignored.

REPOSITORY URL/CONTENTURL

Originally, OSTree had just a `url` option for remotes. Since then, the `contenturl` option was introduced. Both of these support `file`, `http`, and `https` schemes.

Additionally, both of these can be prefixed with the string `mirrorlist=`, which instructs the client that the target url is a "mirrorlist" format, which is a plain text file of newline-separated URLs. Earlier URLs will be given precedence.

Note that currently, the `tls-ca-path` and `tls-client-cert-path` options apply to every HTTP request, even when `contenturl` and/or `mirrorlist` are in use. This may change in the future to only apply to metadata (i.e. `url`, not `contenturl`) fetches.

PER-REMOTE GPG KEYRINGS AND VERIFICATION

OSTree supports a per-remote GPG keyring, as well as a `gpgkeypath` option. For more information see `ostree(1)`. in the section GPG verification.

PER-REMOTE HTTP COOKIES

Some content providers may want to control access to remote repositories via HTTP cookies. The `ostree remote add-cookie` and `ostree remote delete-cookie` commands will update a per-remote lookaside cookie jar, named `$remotename.cookies.txt`.

SEE ALSO

`ostree(1)`, `ostree.repo(5)`

NOTES

1. XDG Desktop Entry Specification

<http://standards.freedesktop.org/desktop-entry-spec/latest/>