systemd-tmpfiles

- Description
- Commands and options
- Credentials
- Environment
- Unprivileged --cleanup operation
- Exit status
- See Also

systemd

systemd-tmpfiles

8

systemd-tmpfiles

systemd-tmpfiles-setup.service

systemd-tmpfiles-setup-dev-early.service

systemd-tmpfiles-setup-dev.service

systemd-tmpfiles-clean.service

systemd-tmpfiles-clean.timer

Create, delete, and clean up files and directories

systemd-tmpfiles

OPTIONS

CONFIGFILE

System units: `systemd-tmpfiles-setup.service`, `systemd-tmpfiles-setup-dev-early.service`, `systemd-tmpfiles-setup-dev.service`, `systemd-tmpfiles-clean.service`, `systemd-tmpfiles-clean.timer`

User units: `systemd-tmpfiles-setup.service`, `systemd-tmpfiles-clean.service`, `systemd-tmpfiles-clean.timer`

Description

`systemd-tmpfiles` creates, deletes, and cleans up files and directories, using the configuration file format and location specified in `tmpfiles.d(5)`. Historically, it was designed to manage volatile and temporary files, as the name suggests, but it provides generic file management functionality and can be used to manage any kind of files. It must be invoked with one or more commands `--create`, `--remove`, and `--clean`, to select the respective subset of operations.

If invoked with no arguments, directives from the configuration files found in the directories specified by `tmpfiles.d(5)` are executed. When invoked with positional arguments, if option `--replace=PATH` is specified, arguments specified on the command line are used instead of the configuration file <PATH>. Otherwise, just the configuration specified by the command line arguments is executed. If the string `-` is specified instead of a filename, the configuration is read from standard input. If the argument is a file name (without any slashes), all configuration directories are searched for a matching file and the file found that has the highest priority is executed. If the argument is a path, that file is used directly without searching the configuration directories for any other matching file.

System services (`systemd-tmpfiles-setup.service`, `systemd-tmpfiles-setup-dev-early.service`, `systemd-tmpfiles-setup-dev.service`, `systemd-tmpfiles-clean.service`) invoke `systemd-tmpfiles` to create system files and to perform system wide cleanup. Those services read administrator-controlled configuration files in `tmpfiles.d/` directories. User services (`systemd-tmpfiles-setup.service`, `systemd-tmpfiles-clean.service`) also invoke `systemd-tmpfiles`, but it reads a separate set of files, which includes user-controlled files under `~/.config/user-tmpfiles.d/` and `~/.local/share/user-tmpfiles.d/`, and administrator-controlled files under `/usr/share/user-tmpfiles.d/`. Users may use this to create and clean up files under their control, but the system instance performs global cleanup and is not influenced by user configuration. Note that this means a time-based cleanup configured in the system instance, such as the one typically configured for `/tmp/`, will thus also affect files created by the user instance if they are placed in `/tmp/`, even if the user instance's time-based cleanup is turned off.

To re-apply settings after configuration has been modified, simply restart `systemd-tmpfiles-clean.service`, which will apply any settings which can be safely executed at runtime. To debug `systemd-tmpfiles`, it may be useful to invoke it directly from the command line with increased log level (see `$SYSTEMD_LOG_LEVEL` below).

Commands and options

The following commands are understood:

`--create`
If this command is passed, all files and directories marked with `f`, `F`, `w`, `d`, `D`, `v`, `p`, `L`, `c`, `b`, `m` in the configuration files are created or written to. Files and directories marked with `z`, `Z`, `t`, `T`, `a`, and `A` have their ownership, access mode and security labels set.

`--clean`
If this command is passed, all files and directories with an age parameter configured will be cleaned up.

`--remove`
If this command is passed, the contents of directories marked with `D` or `R`, and files or directories themselves marked with `r` or `R` are removed unless an exclusive or shared BSD lock is taken on them (see `flock(2)`).

`--purge`
If this option is passed, all files and directories declared for *creation* and marked with the `$` character by the `tmpfiles.d/` files specified on the command line will be *deleted*. Specifically, this acts on all files and directories marked with `f`, `F`, `d`, `D`, `v`, `q`, `Q`, `p`, `L`, `c`, `b`, `C`, `w`, `e`. If this switch is used at least one `tmpfiles.d/` file (or `-` for standard input) must be specified on the command line or the invocation will be refused, for safety reasons (as otherwise much of the installed system files might be removed).

The primary usecase for this option is to automatically remove files and directories that originally have been created on behalf of an installed package at package removal time.

It is recommended to first run this command in combination with `--dry-run` (see below) to verify which files and directories will be deleted.

Warning! This is usually not the command you want! In most cases `--remove` is what you are looking for.

`--user`
Execute "user" configuration, i.e. `tmpfiles.d/` files in user configuration directories.

`--boot`
Also execute lines with an exclamation mark. Lines that are not safe to be executed on a running system may be marked in this way. `systemd-tmpfiles` is executed in early boot with `--boot` specified and will execute those lines. When invoked again later, it should be called without `--boot` .

`--graceful`
Ignore configuration lines pertaining to unknown users or groups. This option is intended to be used in early boot before all users or groups have been created.

`-n`; `--dry-run`
Process the configuration and print what operations would be performed, but do not actually change anything in the file system. The short option `-n` was added in version 262.

`--inline`
Treat each positional argument as a separate configuration line instead of a file name.

`--prefix=path`
Only apply rules with paths that start with the specified prefix. This option can be specified multiple times.

`--exclude-prefix=path`
Ignore rules with paths that start with the specified prefix. This option can be specified multiple times.

`-E`
A shortcut for `--exclude-prefix=/dev --exclude-prefix=/proc --exclude-prefix=/run --exclude-prefix=/sys`, i.e. exclude the hierarchies typically backed by virtual or memory file systems. This is useful in combination with `--root=`, if the specified directory tree contains an OS tree without these virtual/memory file systems mounted in, as it is typically not desirable to create any files and directories below these subdirectories if they are supposed to be overmounted during runtime.

`--root=root`
Takes a directory path as an argument. All paths will be prefixed with the given alternate <root> path, including config search paths.

When this option is used, the libc Name Service Switch (NSS) is bypassed for resolving users and groups. Instead the files `/etc/passwd` and `/etc/group` inside the alternate root are read directly. This means that users/groups not listed in these files will not be resolved, i.e. LDAP NIS and other complex databases are not considered.

Consider combining this with `-E` to ensure the invocation does not create files or directories below mount points in the OS image operated on that are typically overmounted during runtime.

Note that this is not a sandboxing or security feature. `tmpfiles.d(5)` definitions are potentially not processed in a hermetic fashion to the specified root.

`--image=image`
Takes a path to a disk image file or block device node. If specified all operations are applied to file system in the indicated disk image. This is similar to `--root=` but operates on file systems stored in disk images or block devices. The disk image should either contain just a file system or a set of file systems within a GPT partition table, following the UAPI.2 Discoverable Partitions Specification. For further information on supported disk images, see `systemd-nspawn(1)`'s switch of the same name.

Implies `-E`.

`--replace=PATH`
When this option is given, one or more positional arguments must be specified. All configuration files found in the directories listed in `tmpfiles.d(5)` will be read, and the configuration given on the command line will be handled instead of and with the same priority as the configuration file <PATH>.

This option is intended to be used when package installation scripts are running and files belonging to that package are not yet available on disk, so their contents must be given on the command line, but the admin configuration might already exist and should be given higher priority.

It is possible to combine `--create`, `--clean`, and `--remove` in one invocation (in which case removal and cleanup are executed before creation of new files). For example, during boot the following command line is executed to ensure that all temporary and volatile directories are removed and created according to the configuration file:

```
systemd-tmpfiles --remove --create
```

Credentials

`systemd-tmpfiles` supports the service credentials logic as implemented by `ImportCredential=`/`LoadCredential=`/`SetCredential=` (see `systemd.exec(5)` for details). The following credentials are used when passed in:

`tmpfiles.extra`
The contents of this credential may contain additional lines to operate on. The credential contents should follow the same format as any other `tmpfiles.d/` drop-in configuration file. If this credential is passed it is processed after all of the drop-in files read from the file system. The lines in the credential can hence augment existing lines of the OS, but not override them.

Note that by default the `systemd-tmpfiles-setup.service` unit file (and related unit files) is set up to inherit the `tmpfiles.extra` credential from the service manager.

Environment

Unprivileged --cleanup operation

`systemd-tmpfiles` tries to avoid changing the access and modification times on the directories it accesses, which requires `CAP_FOWNER` privileges. When running as non-root, directories which are checked for files to clean up will have their access time bumped, which might prevent their cleanup.

Exit status

On success, 0 is returned. If the configuration was syntactically invalid (syntax errors, missing arguments, ...), so some lines had to be ignored, but no other errors occurred, `65` is returned (`EX_DATAERR` from `/usr/include/sysexits.h`). If the configuration was syntactically valid, but could not be executed (lack of permissions, creation of files in missing directories, invalid contents when writing to `/sys/` values, ...), `73` is returned (`EX_CANTCREAT` from `/usr/include/sysexits.h`). Otherwise, `1` is returned (`EXIT_FAILURE` from `/usr/include/stdlib.h`).

Note: when creating items, if the target already exists, but is of the wrong type or otherwise does not match the requested state, and forced operation has not been requested with `+`, a message is emitted, but the failure is otherwise ignored.

See Also

`systemd(1)`, `tmpfiles.d(5)`

□□□ #1
□ □□□ □□□ 2026-10-31 15:03:05 UTC
□ □□□ □□□ 2026-10-31 15:03:06 UTC