



Linux Ubuntu 26.04 Manual Pages on command 'systemd-umount.1'

\$ man systemd-umount.1

SYSTEMD-MOUNT(1) systemd-mount SYSTEMD-MOUNT(1)

NAME

systemd-mount, systemd-umount - Establish and destroy transient mount or auto-mount points

SYNOPSIS

systemd-mount [OPTIONS...] WHAT [WHERE]

systemd-mount [OPTIONS...] --tmpfs [NAME] WHERE

systemd-mount [OPTIONS...] --list

systemd-mount [OPTIONS...] --umount WHAT|WHERE...

DESCRIPTION

systemd-mount may be used to create and start a transient .mount or .automount unit of the file system WHAT on the mount point WHERE.

In many ways, systemd-mount is similar to the lower-level mount(8) command, however instead of executing the mount operation directly and immediately, systemd-mount schedules it through the service manager job queue, so that it may pull in further dependencies (such as parent mounts, or a file system checker to execute a priori), and may make use of the auto-mounting logic.

The command takes either one or two arguments. If only one argument is specified it should refer to a block device or regular file containing a file system (e.g. "/dev/sdb1" or "/path/to/disk.img"). The block device or image file is then probed for a file system label and other metadata, and is mounted to a directory below /run/media/system/ whose name is generated from the file system label. In this mode the block device or image file must exist at the time of invocation of the command, so that it may be probed. If the device is found to be a removable block device (e.g. a USB stick), an automount point is created instead of

a regular mount point (i.e. the `--automount=` option is implied, see below). If the option `--tmpfs` is specified, then the argument is interpreted as the path where the new temporary file system shall be mounted.

If two arguments are specified, the first indicates the mount source (the WHAT) and the second indicates the path to mount it on (the WHERE). In this mode no probing of the source is attempted, and a backing device node does not have to exist. However, if this mode is combined with `--discover`, device node probing for additional metadata is enabled, and ? much like in the single-argument case discussed above ? the specified device has to exist at the time of invocation of the command.

Use the `--list` command to show a terse table of all local, known block devices with file systems that may be mounted with this command.

`systemd-umount` can be used to unmount a mount or automount point. It is the same as `systemd-mount --umount`.

OPTIONS

The following options are understood:

`--no-block`

Do not synchronously wait for the requested operation to finish. If this is not specified, the job will be verified, enqueued and `systemd-mount` will wait until the mount or automount unit's start-up is completed. By passing this argument, it is only verified and enqueued.

Added in version 232.

`-l, --full`

Do not ellipsize the output when `--list` is specified.

Added in version 245.

`--no-pager`

Do not pipe output into a pager.

`--no-legend`

Do not print the legend, i.e. column headers and the footer with hints.

`--no-ask-password`

Do not query the user for authentication for privileged operations.

`--json=MODE`

Shows output formatted as JSON. Expects one of "short" (for the shortest possible output without any redundant whitespace or line breaks), "pretty" (for a pretty version of the

same, with indentation and line breaks) or "off" (to turn off JSON output, the default).

`--quiet, -q`

Suppresses additional informational output while running.

Added in version 232.

`--discover`

Enable probing of the mount source. This switch is implied if a single argument is specified on the command line. If passed, additional metadata is read from the device to enhance the unit to create. For example, a descriptive string for the transient units is generated from the file system label and device model. Moreover, if a removable block device (e.g. USB stick) is detected an automount unit instead of a regular mount unit is created, with a short idle timeout, in order to ensure the file-system is placed in a clean state quickly after each access.

Added in version 232.

`--type=, -t`

Specifies the file system type to mount (e.g. "vfat" or "ext4"). If omitted or set to "auto", the file system type is determined automatically.

Added in version 232.

`--options=, -o`

Additional mount options for the mount point.

Added in version 232.

`--owner=USER`

Let the specified user USER own the mounted file system. This is done by appending uid= and gid= options to the list of mount options. Only certain file systems support this option.

Added in version 237.

`--fsck=`

Takes a boolean argument, defaults to on. Controls whether to run a file system check immediately before the mount operation. In the automount case (see `--automount=` below) the check will be run the moment the first access to the device is made, which might slightly delay the access.

Added in version 232.

`--description=`

Provide a description for the mount or automount unit. See `Description=` in

systemd.unit(5).

Added in version 232.

`--property=, -p`

Sets a unit property for the mount unit that is created. This takes an assignment in the same format as `systemctl(1)`'s `set-property` command.

Added in version 232.

`--automount=`

Takes a boolean argument. Controls whether to create an automount point or a regular mount point. If true an automount point is created that is backed by the actual file system at the time of first access. If false a plain mount point is created that is backed by the actual file system immediately. Automount points have the benefit that the file system stays unmounted and hence in clean state until it is first accessed. In automount mode the `--timeout-idle-sec=` switch (see below) may be used to ensure the mount point is unmounted automatically after the last access and an idle period passed. If this switch is not specified, it defaults to false. If not specified and `--discover` is used (or only a single argument passed, which implies `--discover`, see above), and the file system block device is detected to be removable, it is set to true, in order to increase the chance that the file system is in a fully clean state if the device is unplugged abruptly.

Added in version 232.

`-A`

Equivalent to `--automount=yes`.

Added in version 232.

`--timeout-idle-sec=`

Takes a time value that controls the idle timeout in automount mode. If set to "infinity" (the default) no automatic unmounts are done. Otherwise, the file system backing the automount point is detached after the last access and the idle timeout passed. See `systemd.time(7)` for details on the time syntax supported. This option has no effect if only a regular mount is established, and automounting is not used.

Note that if `--discover` is used (or only a single argument passed, which implies `--discover`, see above), and the file system block device is detected to be removable, `--timeout-idle-sec=1s` is implied.

Added in version 232.

`--automount-property=`

Similar to `--property=`, but applies additional properties to the automount unit created, instead of the mount unit.

Added in version 232.

`--bind-device`

Controls whether the generated mount/automount unit shall be bound to the backing device's lifetime. This setting is especially useful in automount mode. If set, the units will be stopped automatically when the backing device vanishes. By default, the automount unit stays around, and subsequent accesses will block until backing device is replugged. This option has no effect in case of non-device mounts, such as network or virtual file system mounts.

If `--discover` is used (or only a single argument passed, which implies `--discover`, see above), and the file system block device is detected to be removable, this option is implied.

Note that mount units by default gain a `Requires=` dependency on the backing device. This behavior can be controlled via `x-systemd.device-bound=` mount option, see `systemd.mount(5)` for details. In particular, `x-systemd.device-bound=no` takes precedence over this option, which suppresses device dependencies both in the generated mount units and what's implied by service manager.

Added in version 232.

`--list`

Instead of establishing a mount or automount point, print a terse list of block devices containing file systems that may be mounted with "systemd-mount", along with useful metadata such as labels, etc.

Added in version 232.

`-u, --umount`

Stop the mount and automount units corresponding to the specified mount points `WHERE` or the devices `WHAT`. `systemd-mount` with this option or `systemd-umount` can take multiple arguments which can be mount points, devices, `/etc/fstab` style node names, or backing files corresponding to loop devices, like `systemd-mount --umount /path/to/umount /dev/sda1 UUID=xxxxxx-xxxx LABEL=xxxxx /path/to/disk.img`. Note that when `-H` or `-M` is specified, only absolute paths to mount points are supported.

Added in version 233.

-G, --collect

Unload the transient unit after it completed, even if it failed. Normally, without this option, all mount units that mount and failed are kept in memory until the user explicitly resets their failure state with `systemctl reset-failed` or an equivalent command. On the other hand, units that stopped successfully are unloaded immediately. If this option is turned on the "garbage collection" of units is more aggressive, and unloads units regardless of whether they exited successfully or failed. This option is a shortcut for `--property=CollectMode=inactive-or-failed`, see the explanation for `CollectMode=` in `systemd.unit(5)` for further information.

Added in version 236.

-T, --tmpfs

Create and mount a new tmpfs file system on `WHERE`, with an optional `NAME` that defaults to "tmpfs".

The file system is mounted with the top-level directory mode determined by the `umask(2)` setting of the caller, i.e. `rw-rw-rwx` masked by the `umask` of the caller. This matches what `mkdir(1)` does, but is different from the kernel default of `"rw-rw-rwxt"`, i.e. a world-writable directory with the sticky bit set.

Added in version 255.

--canonicalize=

Controls whether the specified path shall be canonicalized on the client side before requesting the operation or not. Takes a boolean parameter, defaults to true. Note that for non-local operation (i.e. when `--machine=` or `---host=` are used) canonicalization is implicitly turned off.

Canonicalization of path entails resolving of symlinks, `".."` path elements and `LABEL=/UUID=` style device node expansion. If canonicalization is disabled and the path contains a symlink element, `".."`, or a `LABEL=/UUID=/...` expansion the operation will fail.

Added in version 258.

--user

Talk to the service manager of the calling user, rather than the service manager of the system.

--system

Talk to the service manager of the system. This is the implied default.

-H, --host=

Execute the operation remotely. Specify a hostname, or a username and hostname separated by "@", to connect to. The hostname may optionally be suffixed by a port ssh is listening on, separated by ":", and then a container name, separated by "/", which connects directly to a specific container on the specified host. This will use SSH to talk to the remote machine manager instance. Container names may be enumerated with `machinectl` **-H** HOST. Put IPv6 addresses in brackets.

-M, --machine=

Execute operation on a local container. Specify a container name to connect to, optionally prefixed by a user name to connect as and a separating "@" character. If the special string ".host" is used in place of the container name, a connection to the local system is made (which is useful to connect to a specific user's user bus: "`--user --machine=lennart@.host`"). If the "@" syntax is not used, the connection is made as root user. If the "@" syntax is used either the left hand side or the right hand side may be omitted (but not both) in which case the local user name and ".host" are implied.

-h, --help

Print a short help text and exit.

--version

Print a short version string and exit.

EXIT STATUS

On success, 0 is returned, a non-zero failure code otherwise.

THE UDEV DATABASE

If `--discover` is used, `systemd-mount` honors a couple of additional udev properties of block devices:

SYSTEMD_MOUNT_OPTIONS=

The mount options to use, if `--options=` is not used.

Added in version 232.

SYSTEMD_MOUNT_WHERE=

The file system path to place the mount point at, instead of the automatically generated one.

Added in version 232.

EXAMPLE

Use a udev rule like the following to automatically mount all USB storage plugged in:

```
ACTION=="add", SUBSYSTEMS=="usb", SUBSYSTEM=="block", ENV{ID_FS_USAGE}=="filesystem", \
```

```
RUN{program}+="/usr/bin/systemd-mount --no-block --automount=yes --collect $devnode"
```

SEE ALSO

systemd(1), mount(8), systemctl(1), systemd.unit(5), systemd.mount(5), systemd.automount(5),

systemd-run(1)

systemd 259.5

SYSTEMD-MOUNT(1)