



## ***Rocky Enterprise Linux 9.2 Manual Pages on command 'setpriv.1'***

**C:~>man setpriv.1**

SETPRIV(1) User Commands SETPRIV(1)

### NAME

setpriv - run a program with different Linux privilege settings

### SYNOPSIS

setpriv [options] program [arguments]

### DESCRIPTION

Sets or queries various Linux privilege settings that are inherited across `execve(2)`.

In comparison to `su(1)` and `runuser(1)`, `setpriv(1)` neither uses PAM, nor does it prompt for a password. It is a simple, non-set-user-ID wrapper around `execve(2)`, and can be used to drop privileges in the same way as `setuidgid(8)` from `daemon?` tools, `chpst(8)` from `runit`, or similar tools shipped by other service managers.

### OPTION

`--clear-groups`

Clear supplementary groups.

`-d, --dump`

Dump current privilege state. Can be specified more than once to show extra, mostly useless, information. Incompatible with all other options.

`--groups group...`

Set supplementary groups. The argument is a comma-separated list of GIDs or names.

`--inh-caps (+|-)cap...` or `--ambient-caps (+|-)cap...` or `--bounding-set`

(+|-)cap...

Set the inheritable capabilities, ambient capabilities or the capability bounding set. See `capabilities(7)`. The argument is a comma-separated list of `+cap` and `-cap` entries, which add or remove an entry respectively. `cap` can either be a human-readable name as seen in `capabilities(7)` without the `cap_` prefix or of the format `cap_N`, where `N` is the internal capability index used by Linux. `+all` and `-all` can be used to add or remove all caps. The set of capabilities starts out as the current inheritable set for `--inh-caps`, the current ambient set for `--ambient-caps` and the current bounding set for `--bounding-set`. If you drop something from the bounding set without also dropping it from the inheritable set, you are likely to become confused. Do not do that.

`--keep-groups`

Preserve supplementary groups. Only useful in conjunction with `--rgid`, `--egid`, or `--regid`.

`--init-groups`

Initialize supplementary groups using `initgroups(3)`. Only useful in conjunction with `--ruid` or `--reuid`.

`--list-caps`

List all known capabilities. This option must be specified alone.

`--no-new-privs`

Set the `no_new_privs` bit. With this bit set, `execve(2)` will not grant new privileges. For example, the `set-user-ID` and `set-group-ID` bits as well as file capabilities will be disabled. (Executing binaries with these bits set will still work, but they will not gain privileges. Certain LSMs, especially AppArmor, may result in failures to execute certain programs.) This bit is inherited by child processes and cannot be unset. See `prctl(2)` and `Documentation/prctl/no_new_privs.txt` in the Linux kernel source.

The `no_new_privs` bit is supported since Linux 3.5.

`--rgid gid`, `--egid gid`, `--regid gid`

Set the real, effective, or both GIDs. The `gid` argument can be given as textual group name.

For safety, you must specify one of `--clear-groups`, `--groups`, `--keep-groups`,

or --init-groups if you set any primary gid.

--ruid uid, --euid uid, --reuid uid

Set the real, effective, or both UIDs. The uid argument can be given as textual login name.

Setting a uid or gid does not change capabilities, although the exec call at the end might change capabilities. This means that, if you are root, you probably want to do something like:

```
setpriv --reuid=1000 --regid=1000 --inh-caps=-all
```

--securebits (+|-)securebit...

Set or clear securebits. The argument is a comma-separated list. The valid securebits are noroot, noroot\_locked, no\_setuid\_fixup, no\_setuid\_fixup\_locked, and keep\_caps\_locked. keep\_caps is cleared by execve(2) and is therefore not allowed.

--pdeathsig keep|clear|<signal>

Keep, clear or set the parent death signal. Some LSMs, most notably SELinux and AppArmor, clear the signal when the process' credentials change. Using --pdeathsig keep will restore the parent death signal after changing credentials to remedy that situation.

--selinux-label label

Request a particular SELinux transition (using a transition on exec, not dyntrans). This will fail and cause setpriv(1) to abort if SELinux is not in use, and the transition may be ignored or cause execve(2) to fail at SELinux's whim. (In particular, this is unlikely to work in conjunction with no\_new\_privs.) This is similar to runcon(1).

--apparmor-profile profile

Request a particular AppArmor profile (using a transition on exec). This will fail and cause setpriv(1) to abort if AppArmor is not in use, and the transition may be ignored or cause execve(2) to fail at AppArmor's whim.

--reset-env

Clears all the environment variables except TERM; initializes the environment variables HOME, SHELL, USER, LOGNAME according to the user's passwd entry; sets PATH to /usr/local/bin:/bin:/usr/bin for a regular user and to /usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin for root.

The environment variable PATH may be different on systems where /bin and /sbin are merged into /usr. The environment variable SHELL defaults to /bin/sh if none is given in the user's passwd entry.

-V, --version

Display version information and exit.

-h, --help

Display help text and exit.

## NOTES

If applying any specified option fails, program will not be run and setpriv will return with exit code 127.

Be careful with this tool -- it may have unexpected security consequences. For example, setting no\_new\_privs and then execing a program that is SELinux-confined (as this tool would do) may prevent the SELinux restrictions from taking effect.

## EXAMPLE

If you're looking for behaviour similar to su(1)/runuser(1), or sudo(8) (without the -g option), try something like:

```
setpriv --reuid=1000 --regid=1000 --init-groups
```

If you want to mimic daemontools' setuid(8), try:

```
setpriv --reuid=1000 --regid=1000 --clear-groups
```

## SEE ALSO

runuser(1), su(1), prctl(2), capabilities(7)

## AUTHOR

Andy Lutomirski ?luto@amacapital.net?

## AVAILABILITY

The setpriv command is part of the util-linux package and is available from Linux Kernel Archive ?<https://www.kernel.org/pub/linux/utils/util-linux/>?

util-linux

July 2014

SETPRIV(1)