



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 'kernel-install.8'

\$ man kernel-install.8

KERNEL-INSTALL(8) kernel-install KERNEL-INSTALL(8)

NAME

kernel-install - Add and remove kernel and initrd images to and from
/boot

SYNOPSIS

kernel-install [OPTIONS...] COMMAND KERNEL-VERSION KERNEL-IMAGE
[INITRD-FILE...]

DESCRIPTION

kernel-install is used to install and remove kernel and initrd images [1] to and from the boot loader partition, referred to as \$BOOT here. It will usually be one of /boot/, /efi/, or /boot/efi/, see below. kernel-install will run the executable files ("plugins") located in the directory /usr/lib/kernel/install.d/ and the local administration directory /etc/kernel/install.d/. All files are collectively sorted and executed in lexical order, regardless of the directory in which they live. However, files with identical filenames replace each other. Files in /etc/kernel/install.d/ take precedence over files with the same name in /usr/lib/kernel/install.d/. This can be used to override a

system-supplied executables with a local file if needed; a symbolic link in `/etc/kernel/install.d/` with the same name as an executable in `/usr/lib/kernel/install.d/`, pointing to `/dev/null`, disables the executable entirely. Executables must have the extension `".install"`; other extensions are ignored.

An executable placed in these directories should return 0 on success. It may also return 77 to cause the whole operation to terminate (executables later in lexical order will be skipped).

COMMANDS

The following commands are understood:

```
add KERNEL-VERSION KERNEL-IMAGE [INITRD-FILE ...]
```

This command expects a kernel version string and a path to a kernel image file as arguments. Optionally, one or more initrd images may be specified as well (note that plugins might generate additional ones). `kernel-install` calls the executable files from `/usr/lib/kernel/install.d/*.install` and `/etc/kernel/install.d/*.install` (i.e. the plugins) with the following arguments:

```
add KERNEL-VERSION $BOOT/ENTRY-TOKEN/KERNEL-VERSION/ KERNEL-IMAGE [INITRD-FILE ...]
```

The third argument directly refers to the path where to place kernel images, initrd images and other resources for Boot Loader Specification[2] Type #1 entries (the "entry directory"). If other boot loader schemes are used the parameter may be ignored. The `ENTRY-TOKEN` string is typically the machine ID and is supposed to identify the local installation on the system. For details see below.

Two default plugins execute the following operations in this case:

- ? `kernel-install` creates `$BOOT/ENTRY-TOKEN/KERNEL-VERSION`, if enabled (see `$KERNEL_INSTALL_LAYOUT`).
- ? `50-depmod.install` runs `depmod(8)` for the `KERNEL-VERSION`.
- ? `90-loaderentry.install` copies `KERNEL-IMAGE` to `$BOOT/ENTRY-TOKEN/KERNEL-VERSION/linux`. If `INITRD-FILEs` are provided, it also copies them to

`$BOOT/ENTRY-TOKEN/KERNEL_VERSION/INITRD-FILE`. It also creates a boot loader entry according to the Boot Loader Specification[2]

(Type #1) in

`$BOOT/loader/entries/ENTRY-TOKEN-KERNEL-VERSION.conf`. The title of the entry is the `PRETTY_NAME` parameter specified in `/etc/os-release` or `/usr/lib/os-release` (if the former is missing), or "Linux `KERNEL-VERSION`", if unset.

If `$KERNEL_INSTALL_LAYOUT` is not "bls", this plugin does nothing.

remove `KERNEL-VERSION`

This command expects a kernel version string as single argument.

This calls executables from `/usr/lib/kernel/install.d/*.install` and `/etc/kernel/install.d/*.install` with the following arguments:

remove `KERNEL-VERSION` `$BOOT/ENTRY-TOKEN/KERNEL-VERSION/`

Afterwards, `kernel-install` removes the entry directory

`$BOOT/ENTRY-TOKEN/KERNEL-VERSION/` and its contents, if it exists.

Two default plugins execute the following operations in this case:

? `50-depmod.install` removes the files generated by `depmod` for this kernel again.

? `90-loaderentry.install` removes the file

`$BOOT/loader/entries/ENTRY-TOKEN-KERNEL-VERSION.conf`.

inspect

Shows the various paths and parameters configured or auto-detected.

In particular shows the values of the various `$KERNEL_INSTALL_*` environment variables listed below.

THE `$BOOT` PARTITION

The partition where the kernels and Boot Loader Specification[2] snippets are located is called `$BOOT`. `kernel-install` determines the location of this partition by checking `/efi/`, `/boot/`, and `/boot/efi/` in turn. The first location where `$BOOT/loader/entries/` or `$BOOT/ENTRY-TOKEN/` exists is used.

OPTIONS

The following options are understood:

`-v, --verbose`

Output additional information about operations being performed.

`-h, --help`

Print a short help text and exit.

`--version`

Print a short version string and exit.

ENVIRONMENT VARIABLES

Environment variables exported for plugins

If `--verbose` is used, `$KERNEL_INSTALL_VERBOSE=1` will be exported for plugins. They may output additional logs in this case.

`$KERNEL_INSTALL_MACHINE_ID` is set for the plugins to the desired machine-id to use. It's always a 128-bit ID. Normally it's read from `/etc/machine-id`, but it can also be overridden via `$MACHINE_ID` (see below). If not specified via these methods a fallback value will be generated by `kernel-install`, and used only for a single invocation.

`$KERNEL_INSTALL_ENTRY_TOKEN` is set for the plugins to the desired entry "token" to use. It's an identifier that shall be used to identify the local installation, and is often the machine ID, i.e. same as `$KERNEL_INSTALL_MACHINE_ID`, but might also be a different type of identifier, for example a fixed string or the `ID=`, `IMAGE_ID=` values from `/etc/os-release`. The string passed here will be used to name Boot Loader Specification entries, or the directories the kernel image and initial RAM disk images are placed into.

Note that while `$KERNEL_INSTALL_ENTRY_TOKEN` and `$KERNEL_INSTALL_MACHINE_ID` are often set to the same value, the latter is guaranteed to be a valid 32 character ID in lowercase hexadecimals while the former can be any short string. The entry token to use is read from `/etc/kernel/entry-token`, if it exists. Otherwise a few possible candidates below `$BOOT` are checked for Boot Loader Specification Type 1 entry directories, and if found the entry token is derived from that. If that is not successful,

`$KERNEL_INSTALL_MACHINE_ID` is used as fallback.

`$KERNEL_INSTALL_BOOT_ROOT` is set for the plugins to the absolute path

of the root directory (mount point, usually) of the hierarchy where boot loader entries, kernel images, and associated resources should be placed. This usually is the path where the XBOOTLDR partition or the ESP (EFI System Partition) are mounted, and also conceptually referred to as \$BOOT. Can be overridden by setting \$BOOT_ROOT (see below). \$KERNEL_INSTALL_LAYOUT=bls|other|... is set for the plugins to specify the installation layout. Defaults to bls if \$BOOT/ENTRY-TOKEN exists, or other otherwise. Additional layout names may be defined by convention. If a plugin uses a special layout, it's encouraged to declare its own layout name and configure layout= in install.conf upon initial installation. The following values are currently understood:

bls

Standard Boot Loader Specification[2] Type #1 layout, compatible with systemd-boot(7): entries in \$BOOT/loader/entries/ENTRY-TOKEN-KERNEL-VERSION[+TRIES].conf, kernel and initrds under \$BOOT/ENTRY-TOKEN/KERNEL-VERSION/ Implemented by 90-loaderentry.install.

other

Some other layout not understood natively by kernel-install.

\$KERNEL_INSTALL_INITRD_GENERATOR is set for plugins to select the initrd generator. This may be configured as initrd_generator= in install.conf, see below.

\$KERNEL_INSTALL_STAGING_AREA is set for plugins to a path to a directory. Plugins may drop files in that directory, and they will be installed as part of the loader entry, based on the file name and extension.

Environment variables understood by kernel-install

\$KERNEL_INSTALL_CONF_ROOT can be set to override the location of the configuration files read by kernel-install. When set, install.conf, entry-token, and other files will be read from this directory.

\$KERNEL_INSTALL_PLUGINS can be set to override the list of plugins executed by kernel-install. The argument is a whitespace-separated list of paths. "KERNEL_INSTALL_PLUGINS=" may be used to prevent any

plugins from running.

\$MACHINE_ID can be set for kernel-install to override

\$KERNEL_INSTALL_MACHINE_ID, the machine ID.

\$BOOT_ROOT can be set for kernel-install to override

\$KERNEL_INSTALL_BOOT_ROOT, the installation location for boot entries.

The last two variables may also be set in install.conf. Variables set in the environment take precedence over the values specified in the config file.

EXIT STATUS

If every executable returns 0 or 77, 0 is returned, and a non-zero failure code otherwise.

FILES

/usr/lib/kernel/install.d/*.install /etc/kernel/install.d/*.install

Drop-in files which are executed by kernel-install.

/usr/lib/kernel/cmdline /etc/kernel/cmdline /proc/cmdline

Read by 90-loaderentry.install. The content of the file /etc/kernel/cmdline specifies the kernel command line to use. If that file does not exist, /usr/lib/kernel/cmdline is used. If that also does not exist, /proc/cmdline is used.

\$KERNEL_INSTALL_CONF_ROOT may be used to override the path.

/etc/kernel/tries

Read by 90-loaderentry.install. If this file exists a numeric value is read from it and the naming of the generated entry file is slightly altered to include it as \$BOOT/loader/entries/MACHINE-ID-KERNEL-VERSION+TRIES.conf. This is useful for boot loaders such as systemd-boot(7) which implement boot attempt counting with a counter embedded in the entry file name. \$KERNEL_INSTALL_CONF_ROOT may be used to override the path.

/etc/kernel/entry-token

If this file exists it is read and used as "entry token" for this system, i.e. is used for naming Boot Loader Specification entries, see \$KERNEL_INSTALL_ENTRY_TOKEN above for details.

\$KERNEL_INSTALL_CONF_ROOT may be used to override the path.

/etc/machine-id

The content of this file specifies the machine identification

MACHINE-ID.

/etc/os-release /usr/lib/os-release

Read by 90-loaderentry.install. If available, PRETTY_NAME= is read from these files and used as the title of the boot menu entry.

Otherwise, "Linux KERNEL-VERSION" will be used.

/usr/lib/kernel/install.conf /etc/kernel/install.conf

Configuration options for kernel-install, as a series of KEY=VALUE assignments, compatible with shell syntax, following the same rules as described in os-release(5). /etc/kernel/install.conf will be read if present, and /usr/lib/kernel/install.conf otherwise. This file is optional. \$KERNEL_INSTALL_CONF_ROOT may be used to override the path.

Currently, the following keys are supported: MACHINE_ID=, BOOT_ROOT=, layout=, initrd_generator=. See the Environment variables section above for details.

SEE ALSO

machine-id(5), os-release(5), depmod(8), systemd-boot(7), Boot Loader Specification[2]

NOTES

1. Nowadays actually CPIO archives used as an "initramfs", rather than "initrd". See bootup(7) for an explanation.
2. Boot Loader Specification

https://systemd.io/BOOT_LOADER_SPECIFICATION

systemd 252

KERNEL-INSTALL(8)