Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'systemd-veritysetup-generator.8' command

$ man systemd-veritysetup-generator.8

SYSTEMD-VERITYSETUP-GENERsystemd-veritysetup-gSYSTEMD-VERITYSETUP-GENERATOR(8)

NAME

systemd-veritysetup-generator - Unit generator for verity protected block devices

SYNOPSIS

/usr/lib/systemd/system-generators/systemd-veritysetup-generator

DESCRIPTION

systemd-veritysetup-generator is a generator that translates kernel command line options configuring verity protected block devices into native systemd units early at boot and when configuration of the system manager is reloaded. This will create systemd-veritysetup@.service(8) units as necessary.

Currently, only two verity devices may be set up with this generator, backing the root and /usr file systems of the OS.

systemd-veritysetup-generator implements systemd.generator(7).

KERNEL COMMAND LINE

systemd-veritysetup-generator understands the following kernel command line parameters:

systemd.verity=, rd.systemd.verity=

Takes a boolean argument. Defaults to "yes". If "no", disables the generator entirely. rd.systemd.verity= is honored only by the initrd while systemd.verity= is honored by both the host system and the initrd.

roothash=

Takes a root hash value for the root file system. Expects a hash value formatted in hexadecimal characters of the appropriate length (i.e. most likely 256 bit/64 characters, or longer). If not specified via systemd.verity_root_data= and systemd.verity_root_hash=, the hash and data devices to use are automatically derived from the specified hash value. Specifically, the data partition device is looked for under a GPT partition UUID derived from the first 128bit of the root hash, the hash partition device is looked for under a GPT partition UUID derived from the last 128bit of the root hash. Hence it is usually sufficient to specify the root hash to boot from a verity protected root file system, as device paths are automatically determined from it ? as long as the partition table is properly set up.

systemd.verity_root_data=, systemd.verity_root_hash=

These two settings take block device paths as arguments and may be used to explicitly configure the data partition and hash partition to use for setting up the verity protection for the root file system. If not specified, these paths are automatically derived from the roothash= argument (see above).

systemd.verity_root_options=

Takes a comma-separated list of dm-verity options. Expects the following options ignore-corruption, restart-on-corruption, ignore-zero-blocks, check-at-most-once, panic-on-corruption and root-hash-signature=PATH|base64:HEX. See veritysetup(8) for more details.

usrhash=, systemd.verity_usr_data=, systemd.verity_usr_hash=,

systemd.verity_usr_options=

Equivalent to their counterparts for the root file system as described above, but apply to the /usr/ file system instead.

SEE ALSO

systemd(1), systemd-veritysetup@.service(8), veritysetup(8), systemd-fstab-generator(8)