NAME

cryptsetup-repair − repair the device metadata

SYNOPSIS

cryptsetup *repair* **[<options>] <device>**

DESCRIPTION

Tries to repair the device metadata if possible. Currently supported only for LUKS device type.

This command is useful for fixing some known benign LUKS metadata header corruptions. Only basic corruptions of unused keyslot are fixable. This command will only change the LUKS header, not any keyslot data. You may enforce LUKS version by adding −−type option.

It also repairs (upgrades) LUKS2 reencryption metadata by adding a metadata digest that protects it against malicious changes.

If LUKS2 reencryption was interrupted while writing the reencryption segment, the repair command can perform reencryption recovery so that reencryption can continue later. Repairing reencryption requires verification of the reencryption keyslot, so a passphrase or keyfile is needed.

WARNING: Always create a binary backup of the original header before calling this command.

LUKS keyslots corruption detection

The repair command also checks for detectable corruption of keyslot content. Corruption of a keyslot results in a situation where a known password is no longer accepted. It can happen due to storage media failure or overwriting the keyslot area with other data. Only certain corruptions, usually only a low−entropy area (like zeroed blocks), can be detected.

The detection prints only warnings. It does not modify keyslots. It can also print more specific offsets on the device for detailed manual inspection.

Please note that the warning can be a false positive (no real corruption happened). Conversely, if the keyslot is corrupted, no recovery is possible. You have to use the LUKS header backup.

<options> can be [−−timeout, −−verify−passphrase, −−disable−locks, −−type, −−header, −−key−file, −−keyfile−size, −−keyfile−offset, −−key−slot].

OPTIONS

−−batch−mode, **−q**

Suppresses all confirmation questions. Use with care!

If the −−verify−passphrase option is not specified, this option also switches off the passphrase verification.

−−debug or **−−debug−json**

Run in debug mode with full diagnostic logs. Debug output lines are always prefixed by **#**.

If −−debug−json is used, additional LUKS2 JSON data structures are printed.

−−disable−locks

Disable lock protection for metadata on disk. This option is valid only for LUKS2 and is ignored for other formats.

WARNING: Do not use this option unless you run cryptsetup in a restricted environment where locking is impossible to perform (where /run directory cannot be used).

−−header *<device or file storing the LUKS header>*

Use a detached (separated) metadata device or file where the LUKS header is stored. This option allows one to store the ciphertext and LUKS header on different devices.

For commands that change the LUKS header (e.g., *luksAddKey*), specify the device or file with the LUKS header directly as the LUKS device.

−−help, **−?**
Show help text and default parameters.

−−key−file, **−d** *file*
Read the passphrase from the file.

If the name given is "−", then the passphrase will be read from stdin. In this case, reading will not stop at newline characters.

See section *NOTES ON PASSPHRASE PROCESSING* in **cryptsetup**(8) for more information.

−−keyfile−offset *value*
Skip *value* bytes at the beginning of the key file.

−−keyfile−size, **−l** *value*
Read a maximum of *value* bytes from the key file. The default is to read the whole file up to the compiled−in maximum that can be queried with −−help. Supplying more data than the compiled−in maximum aborts the operation.

This option is useful to cut trailing newlines, for example. If −−keyfile−offset is also given, the size count starts after the offset.

−−key−slot, **−S** *<0−N>*
For LUKS operations that add key material, this option allows you to specify which keyslot is selected for the new key.

The maximum number of keyslots depends on the LUKS version. LUKS1 can have up to 8 keyslots. LUKS2 can have up to 32 keyslots based on keyslot area size and key size, but a valid keyslot ID can always be between 0 and 31 for LUKS2.

−−timeout, **−t** *seconds*
The number of seconds to wait before a timeout on passphrase input via terminal. It is relevant every time a passphrase is asked. It has no effect if used in conjunction with −−key−file.

This option is useful when the system should not stall if the user does not input a passphrase, e.g., during boot. The default is a value of 0 seconds, which means to wait forever.

−−type *type*
Specifies required device type, for more info, read the *BASIC ACTIONS* section in **cryptsetup**(8).

−−usage
Show short option help.

−−verify−passphrase, **−y**
When interactively asking for a passphrase, ask for it twice and complain if both inputs do not match. Ignored on input from file or stdin.

−−version, **−V**

Show the program version.

REPORTING BUGS

Report bugs at cryptsetup mailing list <cryptsetup@lists.linux.dev> or in Issues project section <https://gitlab.com/cryptsetup/cryptsetup/−/issues/new>.

Please attach the output of the failed command with −−debug option added.

SEE ALSO

Cryptsetup FAQ <https://gitlab.com/cryptsetup/cryptsetup/wikis/FrequentlyAskedQuestions>

cryptsetup(8), **integritysetup**(8) and **veritysetup**(8)

CRYPTSETUP

Part of cryptsetup project <https://gitlab.com/cryptsetup/cryptsetup/>.