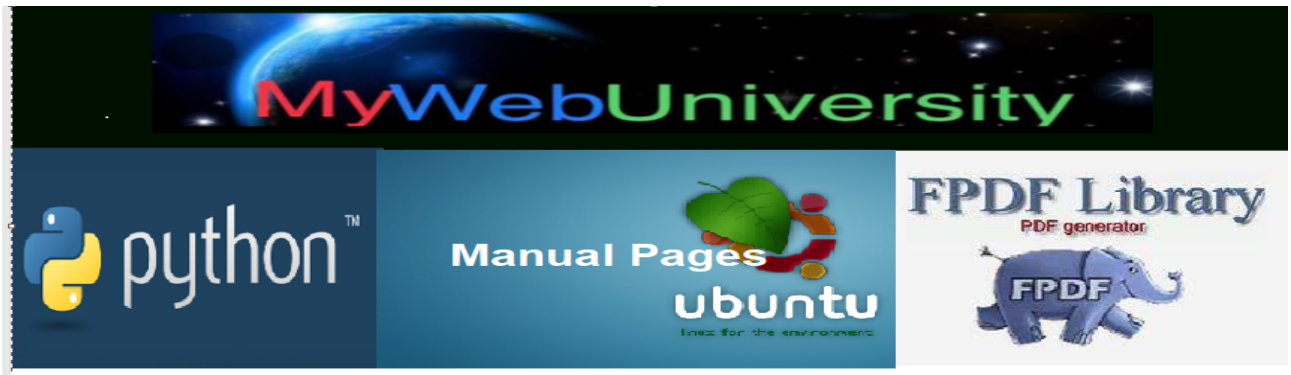

Full credit is given to the above companies including the OS that this PDF file was generated!

Linux Ubuntu 26.04 Manual Pages on command 'rust-shred.1'

$ man rust-shred.1

SHRED(1) General Commands Manual SHRED(1)

NAME

shred - Overwrite the specified FILE(s) repeatedly, in order to make it harder for even very expensive hardware probing to recover the data.

SYNOPSIS

shred [-f|--force] [-n|--iterations] [-s|--size] [-u] [--remove] [-v|--verbose] [-x|--ex?act] [-z|--zero] [--random-source] [-h|--help] [-V|--version] [file]

DESCRIPTION

Overwrite the specified FILE(s) repeatedly, in order to make it harder for even very expen?sive hardware probing to recover the data.

OPTIONS

-f, --force

change permissions to allow writing if necessary

-n, --iterations <NUMBER> [default: 3]

overwrite N times instead of the default (3)

-s, --size <N>

shred this many bytes (suffixes like K, M, G accepted)

-u deallocate and remove file after overwriting

--remove[=<HOW>]

like -u but give control on HOW to delete; See below

Possible values:

? unlink

? wipe

? wipesync

-v, --verbose

show progress

-x, --exact

do not round file sizes up to the next full block; this is the default for non-regu?

lar files

-z, --zero

add a final overwrite with zeros to hide shredding

--random-source

take random bytes from FILE

-h, --help

Print help

-V, --version

Print version

[file]

EXTRA

Delete FILE(s) if --remove (-u) is specified. The default is not to remove the files because it is common to operate on device files like /dev/hda, and those files usually should not be removed.

CAUTION: Note that shred relies on a very important assumption: that the file system over? writes data in place. This is the traditional way to do things, but many modern file system designs do not satisfy this assumption. The following are examples of file systems on which shred is not effective, or is not guaranteed to be effective in all file system modes:

- log-structured or journal file systems, such as those supplied with AIX and Solaris (and JFS, ReiserFS, XFS, Ext3, etc.)
- file systems that write redundant data and carry on even if some writes fail, such as RAID-based file systems
- file systems that make snapshots, such as Network Appliance's NFS server
- file systems that cache in temporary locations, such as NFS version 3 clients
- compressed file systems

In the case of ext3 file systems, the above disclaimer applies (and shred is thus of limited effectiveness) only in data=journal mode, which journals file data in addition to just meta?

data. In both the data=ordered (default) and data=writeback modes, shred works as usual. Ext3 journal modes can be changed by adding the data=something option to the mount options for a particular file system in the /etc/fstab file, as documented in the mount man page (`man mount`).

In addition, file system backups and remote mirrors may contain copies of the file that can? not be removed, and that will allow a shredded file to be recovered later.

VERSION

v(uutils coreutils) 0.8.0

2026-04-16 SHRED(1)