EXAMPLE(8)   System Administration   EXAMPLE(8)

NAME

mkfs.cramfs - make compressed ROM file system

SYNOPSIS

mkfs.cramfs [options] *directory file*

DESCRIPTION

Files on cramfs file systems are zlib-compressed one page at a time to allow random read access. The metadata is not compressed, but is expressed in a terse representation that is more space-efficient than conventional file systems.

The file system is intentionally read-only to simplify its design; random write access for compressed files is difficult to implement. cramfs ships with a utility (mkcramfs) to pack files into new cramfs images.

File sizes are limited to less than 16 MB.

Maximum file system size is a little under 272 MB. (The last file on the file system must begin before the 256 MB block, but can extend past it.)

ARGUMENTS

The *directory* is simply the root of the directory tree that we want to generate a compressed filesystem out of.

The *file* will contain the cram file system, which later can be mounted.

OPTIONS

-v   Enable verbose messaging.

-E   Treat all warnings as errors, which are reflected as command return value.

-b *blocksize*
        Use defined block size, which has to be divisible by page size.

-e *edition*
        Use defined file system edition number in superblock.

-N *big, little, host*
        Use defined endianness. Value defaults to *host*.

-i *file*   Insert a *file* to cramfs file system.

-n *name*
        Set name of the cramfs file system.

-p   Pad by 512 bytes for boot code.

-s   This option is ignored. Originally the -s turned on directory entry sorting.

-z   Make explicit holes. Use of this option will require 2.3.39 kernel, or newer.

-V   Display version information and exit.

-h   Display help and exit.

EXIT STATUS

0   success

8   operation error, such as unable to allocate memory

SEE ALSO

mount(8), fsck.cramfs(8)

AVAILABILITY

The example command is part of the util-linux package and is available from Linux Kernel Archive.