Cramfs - cram a filesystem onto a small ROM

cramfs is designed to be simple and small, and to compress things well.

It uses the zlib routines to compress a file one page at a time, and allows random page access. The meta-data is not compressed, but is expressed in a very terse representation to make it use much less diskspace than traditional filesystems.

You can't write to a cramfs filesystem (making it compressible and compact also makes it _very_ hard to update on-the-fly), so you have to create the disk image with the "mkcramfs" utility.

## Usage Notes

File sizes are limited to less than 16MB.

Maximum filesystem size is a little over 256MB. (The last file on the filesystem is allowed to extend past 256MB.)

Only the low 8 bits of gid are stored. The current version of mkcramfs simply truncates to 8 bits, which is a potential security issue.

Hard links are supported, but hard linked files will still have a link count of 1 in the cramfs image.

Cramfs directories have no `.' or `..' entries. Directories (like every other file on cramfs) always have a link count of 1. (There's no need to use -noleaf in `find', btw.)

No timestamps are stored in a cramfs, so these default to the epoch (1970 GMT). Recently-accessed files may have updated timestamps, but the update lasts only as long as the inode is cached in memory, after which the timestamp reverts to 1970, i.e. moves backwards in time.

Currently, cramfs must be written and read with architectures of the same endianness, and can be read only by kernels with PAGE_CACHE_SIZE == 4096. At least the latter of these is a bug, but it hasn't been decided what the best fix is. For the moment if you have larger pages you can just change the #define in mkcramfs.c, so long as you don't mind the filesystem becoming unreadable to future kernels.

## For /usr/share/magic

```
0       ulelong 0x28cd3d45      Linux cramfs offset 0
>4      ulelong x               size %d
>8      ulelong x               flags 0x%x
>12     ulelong x               future 0x%x
>16     string  >\0             signature "%.16s"
>32     ulelong x               fsid.crc 0x%x
```

```
>36     ulelong x               fsid.edition %d
>40     ulelong x               fsid.blocks %d
>44     ulelong x               fsid.files %d
>48     string  >\0             name "%.16s"
512     ulelong 0x28cd3d45      Linux cramfs offset 512
>516    ulelong x               size %d
>520    ulelong x               flags 0x%x
>524    ulelong x               future 0x%x
>528    string  >\0             signature "%.16s"
>544    ulelong x               fsid.crc 0x%x
>548    ulelong x               fsid.edition %d
>552    ulelong x               fsid.blocks %d
>556    ulelong x               fsid.files %d
>560    string  >\0             name "%.16s"
```

Hacker Notes
------------

See fs/cramfs/README for filesystem layout and implementation notes.