Rocky Enterprise Linux 9.2 Manual Pages on command 'cpio.1'

$ man cpio.1

CPIO(1L) CPIO(1L)

NAME

cpio - copy files to and from archives

__WARNING__

The cpio utility is considered LEGACY based on POSIX specification.
Users are encouraged to use other archiving tools for archive creation.
If you decided to use cpio, you should almost always force cpio to use
the ustar format in copy-out mode by the -H option (cpio -o -H ustar).
This is because the ustar format is well defined in POSIX specification
and thus readable by wide range of other archiving tools (including tar
e.g.).
By default, GNU cpio uses (for historical reasons) the very old binary
format ('bin') which has significant problems nowadays, e.g. with stor?
ing big inode numbers (see the Red Hat bug #952313).
Note also that these days the modern 'pax' archive format should be
considered as the default -- but this format is not implemented in GNU
cpio. You should, again, consider using other archivers (e.g. 'tar
--format=pax').

SYNOPSIS

Copy-out mode

In copy-out mode, cpio copies files into an archive. It reads a list of filenames, one per line, on the standard input, and writes the ar? chive onto the standard output. A typical way to generate the list of filenames is with the find command; you should give find the -depth op? tion to minimize problems with permissions on directories that are un? readable. see ?Options?.

cpio {-o|--create} [-0acvABLV] [-C bytes] [-H format] [-D DIR] [-M mes? sage] [-O [[user@]host:]archive] [-F [[user@]host:]archive] [--file=[[user@]host:]archive] [--format=format] [--warning=FLAG] [--message=message][--null] [--reset-access-time] [--verbose] [--dot] [--append] [--block-size=blocks] [--dereference] [--io-size=bytes] [--rsh-command=command] [--license] [--usage] [--help] [--version] < name-list [> archive]

Copy-in mode

In copy-in mode, cpio copies files out of an archive or lists the ar? chive contents. It reads the archive from the standard input. Any non-option command line arguments are shell globbing patterns; only files in the archive whose names match one or more of those patterns are copied from the archive. Unlike in the shell, an initial `.' in a filename does match a wildcard at the start of a pattern, and a `/' in a filename can match wildcards. If no patterns are given, all files are extracted. see ?Options?.

cpio {-i|--extract} [-bcdfmnrtsuvBSV] [-C bytes] [-E file] [-H format] [-D DIR] [-M message] [-R [user][:.][group]] [-I [[user@]host:]archive] [-F [[user@]host:]archive] [--file=[[user@]host:]archive] [--make-di? rectories] [--nonmatching] [--preserve-modification-time] [--numeric-uid-gid] [--rename] [-t|--list] [--swap-bytes] [--swap] [--dot] [--warning=FLAG] [--unconditional] [--verbose] [--block-size=blocks] [--swap-halfwords] [--io-size=bytes] [--pattern-file=file] [--for? mat=format] [--owner=[user][:.][group]] [--no-preserve-owner] [--mes? sage=message] [--force-local] [--no-absolute-filenames] [--abso?

lute-filenames] [--sparse] [--only-verify-crc] [--to-stdout] [--quiet]
[--ignore-devno] [--renumber-inodes] [--device-independent] [--repro?
ducible] [--rsh-command=command] [--license] [--usage] [--help] [--ver?
sion] [pattern...] [< archive]

Copy-pass mode

In copy-pass mode, cpio copies files from one directory tree to an?
other, combining the copy-out and copy-in steps without actually using
an archive. It reads the list of files to copy from the standard in?
put; the directory into which it will copy them is given as a non-op?
tion argument. see ?Options?.

cpio {-p|--pass-through} [-0adlmuvLV] [-R [user][:.][group]] [-D DIR]
[--null] [--reset-access-time] [--make-directories] [--link] [--quiet]
[--preserve-modification-time] [--unconditional] [--verbose] [--dot]
[--warning=FLAG] [--dereference] [--owner=[user][:.][group]] [--no-pre?
serve-owner] [--sparse] [--license] [--usage] [--help] [--version]
destination-directory < name-list

DESCRIPTION

GNU cpio is a tool for creating and extracting archives, or copying
files from one place to another. It handles a number of cpio formats
as well as reading and writing tar files.

Following archive formats are supported: binary, old ASCII, new ASCII,
crc, HPUX binary, HPUX old ASCII, old tar, and POSIX.1 tar. The tar
format is provided for compatibility with the tar program. By default,
cpio creates binary format archives, for compatibility with older cpio
programs. When extracting from archives, cpio automatically recognizes
which kind of archive it is reading and can read archives created on
machines with a different byte-order.

Main operation mode:

-i, --extract

Extract files from an archive (run in copy-in mode)

-o, --create

Create the archive (run in copy-out mode)

-p, --pass-through

Run in copy-pass mode

-t, --list

Print a table of contents of the input

Operation modifiers valid in any mode:

--block-size=BLOCK-SIZE

Set the I/O block size to BLOCK-SIZE * 512 bytes

-B Set the I/O block size to 5120 bytes. Initially the block size is 512 bytes.

-c Identical to "-H newc", use the new (SVR4) portable format. If you wish the old portable (ASCII) archive format, use "-H odc" instead.

-C, --io-size=NUMBER

Set the I/O block size to the given NUMBER of bytes

-D, --directory=DIR

Change to directory DIR

--force-local

With -F, -I, or -O, take the archive file name to be a local file even if it contains a colon, which would ordinarily indi? cate a remote host name.

-H, --format=FORMAT

Use given archive FORMAT. The valid formats are listed below; the same names are also recognized in all-caps. The default in copy-in mode is to automatically detect the archive format, and in copy-out mode is `bin'.

`bin' The obsolete binary format.

`odc' The old (POSIX.1) portable format.

`newc' The new (SVR4) portable format, which supports file systems hav? ing more than 65536 i-nodes.

`crc' The new (SVR4) portable format with a checksum (Sum32) added.

`tar' The old tar format.

`ustar'

The POSIX.1 tar format. Also recognizes GNU tar archives, which are similar but not identical.

`hpbin'

The obsolete binary format used by HPUX's cpio (which stores de?
vice files differently).

`hpodc'

The portable format used by HPUX's cpio (which stores device
files differently).

--quiet

Do not print the number of blocks copied

-R, --owner=[USER][:.][GROUP]

Set the ownership of all files created to the specified USER
and/or GROUP. Either the user, the group, or both, must be
present. If the group is omitted but the ?:? or ?.? separator
is given, use the given user's login group. Only the super-user
can change files' ownership in copy-in mode.

-v, --verbose

List the files processed, or with `-t', give an `ls -l' style
table of contents listing. In a verbose table of contents of a
ustar archive, user and group names in the archive that do not
exist on the local system are replaced by the names that corre?
spond locally to the numeric UID and GID stored in the archive.

-V, --dot

Print a "." for each file processed

-W, --warning=FLAG

Control warning display. Currently FLAG is one of 'none', 'trun?
cate', 'all'. Multiple options accumulate.

Operation modifiers valid in copy-in and copy-out modes:

-F, --file=[[USER@]HOST:]FILE-NAME

Use this FILE-NAME instead of standard input or output. Optional
USER and HOST specify the user and host names in case of a re?
mote archive

-M, --message=STRING

Print STRING when the end of a volume of the backup media (such
as a tape or a floppy disk) is reached, to prompt the user to

insert a new volume. If STRING contains the string ?%d?, it is replaced by the current volume number (starting at 1).

--rsh-command=COMMAND

Use COMMAND instead of rsh (typically /usr/bin/ssh)

Operation modifiers valid only in copy-in mode:

-b, --swap

Swap both halfwords of words and bytes of halfwords in the data. Equivalent to -sS Use this option to convert 32-bit integers be? tween big-endian and little-endian machines.

-f, --nonmatching

Only copy files that do not match any of the given patterns

-I [[USER@]HOST:]FILE-NAME

Archive filename to use instead of standard input. Optional USER and HOST specify the user and host names in case of a re? mote archive

-n, --numeric-uid-gid

In the verbose table of contents listing, show numeric UID and GID

-r, --rename

Interactively rename files

-s, --swap-bytes

Swap the bytes of each halfword in the files

-S, --swap-halfwords

Swap the halfwords of each word (4 bytes) in the files

--to-stdout

Extract files to standard output

-E, --pattern-file=FILE

Read additional patterns specifying filenames to extract or list from FILE

--only-verify-crc

When reading a CRC format archive, only verify the checksum of each file in the archive, don't actually extract the files

Operation modifiers valid only in copy-out mode:

-A, --append

Append to an existing archive. The archive must be a disk file specified with the -O or -F (-file) option.

--device-independent, --reproducible

Create device-independent (reproducible) archives

--ignore-devno

Don't store device numbers

-O [[USER@]HOST:]FILE-NAME

Archive filename to use instead of standard output. Optional USER and HOST specify the user and host names in case of a re? mote archive

--renumber-inodes

Renumber inodes

Operation modifiers valid only in copy-pass mode:

-l, --link

Link files instead of copying them, when possible

Operation modifiers valid in copy-in and copy-out modes:

--absolute-filenames

Do not strip file system prefix components from the file names

--no-absolute-filenames

Create all files relative to the current directory

Operation modifiers valid in copy-out and copy-pass modes:

-0, --null

Filenames in the list are delimited by null characters instead of newlines, so that files whose names contain newlines can be archived. GNU find is one way to produce a list of null-termi? nated filenames.

-a, --reset-access-time

Reset the access times of files after reading them, so that it does not look like they have just been read.

-L, --dereference

Dereference symbolic links (copy the files that they point to instead of copying the links).

Operation modifiers valid in copy-in and copy-pass modes:

-d, --make-directories

Create leading directories where needed

-m, --preserve-modification-time

Retain previous file modification times when creating files

--no-preserve-owner

Do not change the ownership of the files; leave them owned by the user extracting them. This is the default for non-root users, so that users on System V don't inadvertently give away files. This option can be used in copy-in mode and copy-pass mode

--sparse

Write files with large blocks of zeros as sparse files

-u, --unconditional

Replace all files unconditionally

-?, --help

give this help list

--usage

give a short usage message

--version

print program version

Mandatory or optional arguments to long options are also mandatory or optional for any corresponding short options.

EXAMPLES

When creating an archive, cpio takes the list of files to be processed from the standard input, and then sends the archive to the standard output, or to the device defined by the `-F' option. Usually find or ls is used to provide this list to the standard input. In the follow? ing example you can see the possibilities for archiving the contents of a single directory.

% ls | cpio -ov > directory.cpio

The `-o' option creates the archive, and the `-v' option prints the names of the files archived as they are added. Notice that the options

can be put together after a single `-' or can be placed separately on the command line. The `>' redirects the cpio output to the file `di? rectory.cpio'.

If you wanted to archive an entire directory tree, the find command can provide the file list to cpio:

% find . -print -depth | cpio -ov > tree.cpio

This will take all the files in the current directory, the directories below and place them in the archive tree.cpio. Again the `-o' creates an archive, and the `-v' option shows you the name of the files as they are archived. see ?Copy-out mode?. Using the `.' in the find state? ment will give you more flexibility when doing restores, as it will save file names with a relative path vice a hard wired, absolute path. The `-depth' option forces `find' to print of the entries in a direc? tory before printing the directory itself. This limits the effects of restrictive directory permissions by printing the directory entries in a directory before the directory name itself.

Extracting an archive requires a bit more thought because cpio will not create directories by default. Another characteristic, is it will not overwrite existing files unless you tell it to.

% cpio -iv < directory.cpio

This will retrieve the files archived in the file directory.cpio and place them in the present directory. The `-i' option extracts the ar? chive and the `-v' shows the file names as they are extracted. If you are dealing with an archived directory tree, you need to use the `-d' option to create directories as necessary, something like:

% cpio -idv < tree.cpio

This will take the contents of the archive tree.cpio and extract it to the current directory. If you try to extract the files on top of files of the same name that already exist (and have the same or later modifi? cation time) cpio will not extract the file unless told to do so by the -u option. see ?Copy-in mode?.

In copy-pass mode, cpio copies files from one directory tree to an? other, combining the copy-out and copy-in steps without actually using

an archive. It reads the list of files to copy from the standard in?
put; the directory into which it will copy them is given as a non-op?
tion argument. see ?Copy-pass mode?.

% find . -depth -print0 | cpio --null -pvd new-dir

The example shows copying the files of the present directory, and sub-
directories to a new directory called new-dir. Some new options are
the `-print0' available with GNU find, combined with the `--null' op?
tion of cpio. These two options act together to send file names be?
tween find and cpio, even if special characters are embedded in the
file names. Another is `-p', which tells cpio to pass the files it
finds to the directory `new-dir'.

AUTHOR

Written by Phil Nelson, David MacKenzie, John Oleynick, and Sergey
Poznyakoff.

REPORTING BUGS

Report bugs to <bug-cpio@gnu.org>. Report bugs in this manual page via
https://bugzilla.redhat.com.

COPYRIGHT

SEE ALSO

The full documentation for cpio is maintained as a Texinfo manual. If
the info and cpio programs are properly installed at your site, the
command

info cpio

should give you access to the complete manual.
The online copy of the documentation is available at the following ad?
dress:
http://www.gnu.org/software/cpio/manual