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

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'tar.1' command

$ man tar.1

TAR(1) GNU TAR Manual TAR(1)

NAME

tar - an archiving utility

SYNOPSIS

Traditional usage

tar {A|c|d|r|t|u|x}[GnSkUWOmpsMBiajJzZhPlRvwo] [ARG...]

UNIX-style usage

tar -A [OPTIONS] ARCHIVE ARCHIVE

tar -c [-f ARCHIVE] [OPTIONS] [FILE...]

tar -d [-f ARCHIVE] [OPTIONS] [FILE...]

tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]

tar -r [-f ARCHIVE] [OPTIONS] [FILE...]

tar -u [-f ARCHIVE] [OPTIONS] [FILE...]

tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]

GNU-style usage

tar {--catenate|--concatenate} [OPTIONS] ARCHIVE ARCHIVE

tar --create [--file ARCHIVE] [OPTIONS] [FILE...]

tar {--diff|--compare} [--file ARCHIVE] [OPTIONS] [FILE...]

tar --delete [--file ARCHIVE] [OPTIONS] [MEMBER...]

tar --append [-f ARCHIVE] [OPTIONS] [FILE...]

tar --list [-f ARCHIVE] [OPTIONS] [MEMBER...]

tar --test-label [--file ARCHIVE] [OPTIONS] [LABEL...]

tar --update [--file ARCHIVE] [OPTIONS] [FILE...]

tar --update [-f ARCHIVE] [OPTIONS] [FILE...]

tar {--extract|--get} [-f ARCHIVE] [OPTIONS] [MEMBER...]

NOTE

This manpage is a short description of GNU tar. For a detailed discus? sion, including examples and usage recommendations, refer to the GNU Tar Manual available in texinfo format. If the info reader and the tar documentation are properly installed on your system, the command

info tar

should give you access to the complete manual.

You can also view the manual using the info mode in emacs(1), or find it in various formats online at

http://www.gnu.org/software/tar/manual

If any discrepancies occur between this manpage and the GNU Tar Manual, the later shall be considered the authoritative source.

DESCRIPTION

GNU tar is an archiving program designed to store multiple files in a single file (an archive), and to manipulate such archives. The archive can be either a regular file or a device (e.g. a tape drive, hence the name of the program, which stands for tape archiver), which can be lo? cated either on the local or on a remote machine.

Option styles

Options to GNU tar can be given in three different styles. In tradi? tional style, the first argument is a cluster of option letters and all subsequent arguments supply arguments to those options that require them. The arguments are read in the same order as the option letters. Any command line words that remain after all options has been processed are treated as non-optional arguments: file or archive member names. For example, the c option requires creating the archive, the v option requests the verbose operation, and the f option takes an argument that sets the name of the archive to operate upon. The following command, written in the traditional style, instructs tar to store all files from the directory /etc into the archive file etc.tar verbosely listing the files being archived:

tar cfv etc.tar /etc

In UNIX or short-option style, each option letter is prefixed with a single dash, as in other command line utilities. If an option takes argument, the argument follows it, either as a separate command line word, or immediately following the option. However, if the option takes an optional argument, the argument must follow the option letter without any intervening whitespace, as in -g/tmp/snar.db.

Any number of options not taking arguments can be clustered together after a single dash, e.g. -vkp. Options that take arguments (whether mandatory or optional), can appear at the end of such a cluster, e.g. -vkpf a.tar.

The example command above written in the short-option style could look like:

tar -cvf etc.tar /etc

or

tar -c -v -f etc.tar /etc

In GNU or long-option style, each option begins with two dashes and has a meaningful name, consisting of lower-case letters and dashes. When used, the long option can be abbreviated to its initial letters, pro? vided that this does not create ambiguity. Arguments to long options are supplied either as a separate command line word, immediately fol? lowing the option, or separated from the option by an equals sign with no intervening whitespace. Optional arguments must always use the lat? ter method.

Here are several ways of writing the example command in this style:

tar --create --file etc.tar --verbose /etc

or (abbreviating some options):

tar --cre --file=etc.tar --verb /etc

The options in all three styles can be intermixed, although doing so with old options is not encouraged.

Operation mode

The options listed in the table below tell GNU tar what operation it is to perform. Exactly one of them must be given. Meaning of non-op?

tional arguments depends on the operation mode requested.

-A, --catenate, --concatenate

Append archive to the end of another archive. The arguments are treated as the names of archives to append. All archives must be of the same format as the archive they are appended to, oth? erwise the resulting archive might be unusable with non-GNU im? plementations of tar. Notice also that when more than one ar? chive is given, the members from archives other than the first one will be accessible in the resulting archive only if using the -i (--ignore-zeros) option.

Compressed archives cannot be concatenated.

-c, --create

Create a new archive. Arguments supply the names of the files to be archived. Directories are archived recursively, unless the --no-recursion option is given.

-d, --diff, --compare

Find differences between archive and file system. The arguments are optional and specify archive members to compare. If not given, the current working directory is assumed.

--delete

Delete from the archive. The arguments supply names of the ar? chive members to be removed. At least one argument must be given.

This option does not operate on compressed archives. There is no short option equivalent.

-r, --append

Append files to the end of an archive. Arguments have the same meaning as for -c (--create).

-t, --list

List the contents of an archive. Arguments are optional. When given, they specify the names of the members to list.

--test-label

Test the archive volume label and exit. When used without argu?

ments, it prints the volume label (if any) and exits with status 0. When one or more command line arguments are given. tar compares the volume label with each argument. It exits with code 0 if a match is found, and with code 1 otherwise. No output is displayed, unless used together with the -v (--verbose) option. There is no short option equivalent for this option.

-u, --update

Append files which are newer than the corresponding copy in the archive. Arguments have the same meaning as with -c and -r options. Notice, that newer files don't replace their old archive copies, but instead are appended to the end of archive. The resulting archive can thus contain several members of the same name, corresponding to various versions of the same file.

-x, --extract, --get

Extract files from an archive. Arguments are optional. When given, they specify names of the archive members to be extracted.

--show-defaults

Show built-in defaults for various tar options and exit. No arguments are allowed.

-?, --help

Display a short option summary and exit. No arguments allowed.

--usage

Display a list of available options and exit. No arguments allowed.

--version

Print program version and copyright information and exit.

OPTIONS

Operation modifiers

--check-device

Check device numbers when creating incremental archives (default).

-g, --listed-incremental=FILE

Handle new GNU-format incremental backups. FILE is the name of a snapshot file, where tar stores additional information which is used to decide which files changed since the previous incre? mental dump and, consequently, must be dumped again. If FILE does not exist when creating an archive, it will be created and all files will be added to the resulting archive (the level 0 dump). To create incremental archives of non-zero level N, cre? ate a copy of the snapshot file created during the level N-1, and use it as FILE.

When listing or extracting, the actual contents of FILE is not inspected, it is needed only due to syntactical requirements. It is therefore common practice to use /dev/null in its place.

--hole-detection=METHOD

Use METHOD to detect holes in sparse files. This option implies --sparse. Valid values for METHOD are seek and raw. Default is seek with fallback to raw when not applicable.

-G, --incremental

Handle old GNU-format incremental backups.

--ignore-failed-read

Do not exit with nonzero on unreadable files.

--level=NUMBER

Set dump level for created listed-incremental archive. Cur? rently only --level=0 is meaningful: it instructs tar to trun? cate the snapshot file before dumping, thereby forcing a level 0 dump.

-n, --seek

Assume the archive is seekable. Normally tar determines auto? matically whether the archive can be seeked or not. This option is intended for use in cases when such recognition fails. It takes effect only if the archive is open for reading (e.g. with --list or --extract options).

--no-check-device

Do not check device numbers when creating incremental archives.

--no-seek

Assume the archive is not seekable.

--occurrence[=N]

Process only the Nth occurrence of each file in the archive. This option is valid only when used with one of the following subcommands: --delete, --diff, --extract or --list and when a list of files is given either on the command line or via the -T option. The default N is 1.

--restrict

Disable the use of some potentially harmful options.

--sparse-version=MAJOR[.MINOR]

Set version of the sparse format to use (implies --sparse). This option implies --sparse. Valid argument values are 0.0, 0.1, and 1.0. For a detailed discussion of sparse formats, re? fer to the GNU Tar Manual, appendix D, "Sparse Formats". Using info reader, it can be accessed running the following command: info tar 'Sparse Formats'.

-S, --sparse

Handle sparse files efficiently. Some files in the file system may have segments which were actually never written (quite often these are database files created by such systems as DBM). When given this option, tar attempts to determine if the file is sparse prior to archiving it, and if so, to reduce the resulting archive size by not dumping empty parts of the file.

Overwrite control

These options control tar actions when extracting a file over an exist? ing copy on disk.

-k, --keep-old-files

Don't replace existing files when extracting.

--keep-newer-files

Don't replace existing files that are newer than their archive copies.

--keep-directory-symlink

Don't replace existing symlinks to directories when extracting.

--no-overwrite-dir

Preserve metadata of existing directories.

--one-top-level[=DIR]

Extract all files into DIR, or, if used without argument, into a subdirectory named by the base name of the archive (minus stan? dard compression suffixes recognizable by --auto-compress).

--overwrite

Overwrite existing files when extracting.

--overwrite-dir

Overwrite metadata of existing directories when extracting (de? fault).

--recursive-unlink

Recursively remove all files in the directory prior to extract? ing it.

--remove-files

Remove files from disk after adding them to the archive.

--skip-old-files

Don't replace existing files when extracting, silently skip over them.

-U, --unlink-first

Remove each file prior to extracting over it.

-W, --verify

Verify the archive after writing it.

Output stream selection

--ignore-command-error

Ignore subprocess exit codes.

--no-ignore-command-error

Treat non-zero exit codes of children as error (default).

-O, --to-stdout

Extract files to standard output.

--to-command=COMMAND

Pipe extracted files to COMMAND. The argument is the pathname

of an external program, optionally with command line arguments. The program will be invoked and the contents of the file being extracted supplied to it on its standard input. Additional data will be supplied via the following environment variables:

TAR_FILETYPE

Type of the file. It is a single letter with the follow? ing meaning:

f Regular file

d Directory

l Symbolic link

h Hard link

b Block device

c Character device

Currently only regular files are supported.

TAR_MODE

File mode, an octal number.

TAR_FILENAME

The name of the file.

TAR_REALNAME

Name of the file as stored in the archive.

TAR_UNAME

Name of the file owner.

TAR_GNAME

Name of the file owner group.

TAR_ATIME

Time of last access. It is a decimal number, representing seconds since the Epoch. If the archive provides times with nanosecond precision, the nanoseconds are appended to the timestamp after a decimal point.

TAR_MTIME

Time of last modification.

TAR_CTIME

Time of last status change.

TAR_SIZE

Size of the file.

TAR_UID

UID of the file owner.

TAR_GID

GID of the file owner.

Additionally, the following variables contain information about tar operation mode and the archive being processed:

TAR_VERSION

GNU tar version number.

TAR_ARCHIVE

The name of the archive tar is processing.

TAR_BLOCKING_FACTOR

Current blocking factor, i.e. number of 512-byte blocks in a record.

TAR_VOLUME

Ordinal number of the volume tar is processing (set if reading a multi-volume archive).

TAR_FORMAT

Format of the archive being processed. One of: gnu, oldgnu, posix, ustar, v7.

TAR_SUBCOMMAND

A short option (with a leading dash) describing the oper? ation tar is executing.

Handling of file attributes

--atime-preserve[=METHOD]

Preserve access times on dumped files, either by restoring the times after reading (METHOD=replace, this is the default) or by not setting the times in the first place (METHOD=system)

--delay-directory-restore

Delay setting modification times and permissions of extracted directories until the end of extraction. Use this option when extracting from an archive which has unusual member ordering.

--group=NAME[:GID]

Force NAME as group for added files. If GID is not supplied, NAME can be either a user name or numeric GID. In this case the missing part (GID or name) will be inferred from the current host's group database.

When used with --group-map=FILE, affects only those files whose owner group is not listed in FILE.

--group-map=FILE

Read group translation map from FILE. Empty lines are ignored. Comments are introduced with # sign and extend to the end of line. Each non-empty line in FILE defines translation for a single group. It must consist of two fields, delimited by any amount of whitespace:

OLDGRP NEWGRP[:NEWGID]

OLDGRP is either a valid group name or a GID prefixed with +. Unless NEWGID is supplied, NEWGRP must also be either a valid group name or a +GID. Otherwise, both NEWGRP and NEWGID need not be listed in the system group database.

As a result, each input file with owner group OLDGRP will be stored in archive with owner group NEWGRP and GID NEWGID.

--mode=CHANGES

Force symbolic mode CHANGES for added files.

--mtime=DATE-OR-FILE

Set mtime for added files. DATE-OR-FILE is either a date/time in almost arbitrary format, or the name of an existing file. In the latter case the mtime of that file will be used.

-m, --touch

Don't extract file modified time.

--no-delay-directory-restore

Cancel the effect of the prior --delay-directory-restore option.

--no-same-owner

Extract files as yourself (default for ordinary users).

--no-same-permissions

Apply the user's umask when extracting permissions from the ar?
chive (default for ordinary users).

--numeric-owner

Always use numbers for user/group names.

--owner=NAME[:UID]

Force NAME as owner for added files. If UID is not supplied, NAME can be either a user name or numeric UID. In this case the missing part (UID or name) will be inferred from the current host's user database.

When used with --owner-map=FILE, affects only those files whose owner is not listed in FILE.

--owner-map=FILE

Read owner translation map from FILE. Empty lines are ignored.

Comments are introduced with # sign and extend to the end of line. Each non-empty line in FILE defines translation for a single UID. It must consist of two fields, delimited by any amount of whitespace:

OLDUSR NEWUSR[:NEWUID]

OLDUSR is either a valid user name or a UID prefixed with +.

Unless NEWUID is supplied, NEWUSR must also be either a valid user name or a +UID. Otherwise, both NEWUSR and NEWUID need not be listed in the system user database.

As a result, each input file owned by OLDUSR will be stored in archive with owner name NEWUSR and UID NEWUID.

-p, --preserve-permissions, --same-permissions

extract information about file permissions (default for supe?
ruser)

--same-owner

Try extracting files with the same ownership as exists in the archive (default for superuser).

-s, --preserve-order, --same-order

Sort names to extract to match archive

--sort=ORDER

When creating an archive, sort directory entries according to ORDER, which is one of none, name, or inode.

The default is --sort=none, which stores archive members in the same order as returned by the operating system.

Using --sort=name ensures the member ordering in the created archive is uniform and reproducible.

Using --sort=inode reduces the number of disk seeks made when creating the archive and thus can considerably speed up archiva? tion. This sorting order is supported only if the underlying system provides the necessary information.

Extended file attributes

--acls Enable POSIX ACLs support.

--no-acls

Disable POSIX ACLs support.

--selinux

Enable SELinux context support.

--no-selinux

Disable SELinux context support.

--xattrs

Enable extended attributes support.

--no-xattrs

Disable extended attributes support.

--xattrs-exclude=PATTERN

Specify the exclude pattern for xattr keys. PATTERN is a POSIX regular expression, e.g. --xattrs-exclude='^user.', to exclude attributes from the user namespace.

--xattrs-include=PATTERN

Specify the include pattern for xattr keys. PATTERN is a POSIX regular expression.

Device selection and switching

-f, --file=ARCHIVE

Use archive file or device ARCHIVE. If this option is not given, tar will first examine the environment variable `TAPE'.

If it is set, its value will be used as the archive name. Oth?
erwise, tar will assume the compiled-in default. The default
value can be inspected either using the --show-defaults option,
or at the end of the tar --help output.
An archive name that has a colon in it specifies a file or de?
vice on a remote machine. The part before the colon is taken as
the machine name or IP address, and the part after it as the
file or device pathname, e.g.:
--file=remotehost:/dev/sr0
An optional username can be prefixed to the hostname, placing a
@ sign between them.
By default, the remote host is accessed via the rsh(1) command.
Nowadays it is common to use ssh(1) instead. You can do so by
giving the following command line option:
--rsh-command=/usr/bin/ssh
The remote machine should have the rmt(8) command installed. If
its pathname does not match tar's default, you can inform tar
about the correct pathname using the --rmt-command option.

--force-local

Archive file is local even if it has a colon.

-F, --info-script=COMMAND, --new-volume-script=COMMAND

Run COMMAND at the end of each tape (implies -M). The command
can include arguments. When started, it will inherit tar's en?
vironment plus the following variables:

TAR_VERSION

GNU tar version number.

TAR_ARCHIVE

The name of the archive tar is processing.

TAR_BLOCKING_FACTOR

Current blocking factor, i.e. number of 512-byte blocks
in a record.

TAR_VOLUME

Ordinal number of the volume tar is processing (set if

reading a multi-volume archive).

TAR_FORMAT

Format of the archive being processed. One of: gnu, oldgnu, posix, ustar, v7.

TAR_SUBCOMMAND

A short option (with a leading dash) describing the oper? ation tar is executing.

TAR_FD File descriptor which can be used to communicate the new volume name to tar.

If the info script fails, tar exits; otherwise, it begins writ? ing the next volume.

-L, --tape-length=N

Change tape after writing Nx1024 bytes. If N is followed by a size suffix (see the subsection Size suffixes below), the suffix specifies the multiplicative factor to be used instead of 1024. This option implies -M.

-M, --multi-volume

Create/list/extract multi-volume archive.

--rmt-command=COMMAND

Use COMMAND instead of rmt when accessing remote archives. See the description of the -f option, above.

--rsh-command=COMMAND

Use COMMAND instead of rsh when accessing remote archives. See the description of the -f option, above.

--volno-file=FILE

When this option is used in conjunction with --multi-volume, tar will keep track of which volume of a multi-volume archive it is working in FILE.

Device blocking

-b, --blocking-factor=BLOCKS

Set record size to BLOCKSx512 bytes.

-B, --read-full-records

When listing or extracting, accept incomplete input records af?

ter end-of-file marker.

-i, --ignore-zeros

Ignore zeroed blocks in archive. Normally two consecutive 512-blocks filled with zeroes mean EOF and tar stops reading af? ter encountering them. This option instructs it to read further and is useful when reading archives created with the -A option.

--record-size=NUMBER

Set record size. NUMBER is the number of bytes per record. It must be multiple of 512. It can can be suffixed with a size suffix, e.g. --record-size=10K, for 10 Kilobytes. See the sub? section Size suffixes, for a list of valid suffixes.

Archive format selection

-H, --format=FORMAT

Create archive of the given format. Valid formats are:

gnu GNU tar 1.13.x format

oldgnu GNU format as per tar <= 1.12.

pax, posix

POSIX 1003.1-2001 (pax) format.

ustar POSIX 1003.1-1988 (ustar) format.

v7 Old V7 tar format.

--old-archive, --portability

Same as --format=v7.

--pax-option=keyword[[:]=value][,keyword[[:]=value]]...

Control pax keywords when creating PAX archives (-H pax). This option is equivalent to the -o option of the pax(1) utility.

--posix

Same as --format=posix.

-V, --label=TEXT

Create archive with volume name TEXT. If listing or extracting, use TEXT as a globbing pattern for volume name.

Compression options

-a, --auto-compress

Use archive suffix to determine the compression program.

-I, --use-compress-program=COMMAND

Filter data through COMMAND. It must accept the -d option, for

decompression. The argument can contain command line options.

-j, --bzip2

Filter the archive through bzip2(1).

-J, --xz

Filter the archive through xz(1).

--lzip Filter the archive through lzip(1).

--lzma Filter the archive through lzma(1).

--lzop Filter the archive through lzop(1).

--no-auto-compress

Do not use archive suffix to determine the compression program.

-z, --gzip, --gunzip, --ungzip

Filter the archive through gzip(1).

-Z, --compress, --uncompress

Filter the archive through compress(1).

--zstd Filter the archive through zstd(1).

Local file selection

--add-file=FILE

Add FILE to the archive (useful if its name starts with a dash).

--backup[=CONTROL]

Backup before removal. The CONTROL argument, if supplied, con?

trols the backup policy. Its valid values are:

none, off

Never make backups.

t, numbered

Make numbered backups.

nil, existing

Make numbered backups if numbered backups exist, simple

backups otherwise.

never, simple

Always make simple backups

If CONTROL is not given, the value is taken from the VER?

SION_CONTROL environment variable. If it is not set, existing is assumed.

-C, --directory=DIR

Change to DIR before performing any operations. This option is order-sensitive, i.e. it affects all options that follow.

--exclude=PATTERN

Exclude files matching PATTERN, a glob(3)-style wildcard pat? tern.

--exclude-backups

Exclude backup and lock files.

--exclude-caches

Exclude contents of directories containing file CACHEDIR.TAG, except for the tag file itself.

--exclude-caches-all

Exclude directories containing file CACHEDIR.TAG and the file itself.

--exclude-caches-under

Exclude everything under directories containing CACHEDIR.TAG

--exclude-ignore=FILE

Before dumping a directory, see if it contains FILE. If so, read exclusion patterns from this file. The patterns affect only the directory itself.

--exclude-ignore-recursive=FILE

Same as --exclude-ignore, except that patterns from FILE affect both the directory and all its subdirectories.

--exclude-tag=FILE

Exclude contents of directories containing FILE, except for FILE itself.

--exclude-tag-all=FILE

Exclude directories containing FILE.

--exclude-tag-under=FILE

Exclude everything under directories containing FILE.

--exclude-vcs

Exclude version control system directories.

--exclude-vcs-ignores

Exclude files that match patterns read from VCS-specific ignore files. Supported files are: .cvsignore, .gitignore, .bzrignore, and .hgignore.

-h, --dereference

Follow symlinks; archive and dump the files they point to.

--hard-dereference

Follow hard links; archive and dump the files they refer to.

-K, --starting-file=MEMBER

Begin at the given member in the archive.

--newer-mtime=DATE

Work on files whose data changed after the DATE. If DATE starts with / or . it is taken to be a file name; the mtime of that file is used as the date.

--no-null

Disable the effect of the previous --null option.

--no-recursion

Avoid descending automatically in directories.

--no-unquote

Do not unquote input file or member names.

--no-verbatim-files-from

Treat each line read from a file list as if it were supplied in the command line. I.e., leading and trailing whitespace is re? moved and, if the resulting string begins with a dash, it is treated as tar command line option.

This is the default behavior. The --no-verbatim-files-from op? tion is provided as a way to restore it after --verba? tim-files-from option.

This option is positional: it affects all --files-from options that occur after it in, until --verbatim-files-from option or end of line, whichever occurs first.

It is implied by the --no-null option.

--null Instruct subsequent -T options to read null-terminated names

verbatim (disables special handling of names that start with a

dash).

See also --verbatim-files-from.

-N, --newer=DATE, --after-date=DATE

Only store files newer than DATE. If DATE starts with / or . it

is taken to be a file name; the mtime of that file is used as

the date.

--one-file-system

Stay in local file system when creating archive.

-P, --absolute-names

Don't strip leading slashes from file names when creating ar?

chives.

--recursion

Recurse into directories (default).

--suffix=STRING

Backup before removal, override usual suffix. Default suffix is

~, unless overridden by environment variable SIMPLE_BACKUP_SUF?

FIX.

-T, --files-from=FILE

Get names to extract or create from FILE.

Unless specified otherwise, the FILE must contain a list of

names separated by ASCII LF (i.e. one name per line). The names

read are handled the same way as command line arguments. They

undergo quote removal and word splitting, and any string that

starts with a - is handled as tar command line option.

If this behavior is undesirable, it can be turned off using the

--verbatim-files-from option.

The --null option instructs tar that the names in FILE are sepa?

rated by ASCII NUL character, instead of LF. It is useful if

the list is generated by find(1) -print0 predicate.

--unquote

Unquote file or member names (default).

--verbatim-files-from

Treat each line obtained from a file list as a file name, even if it starts with a dash. File lists are supplied with the --files-from (-T) option. The default behavior is to handle names supplied in file lists as if they were typed in the com? mand line, i.e. any names starting with a dash are treated as tar options. The --verbatim-files-from option disables this be? havior.

This option affects all --files-from options that occur after it in the command line. Its effect is reverted by the --no-verba? tim-files-from} option.

This option is implied by the --null option.

See also --add-file.

-X, --exclude-from=FILE

Exclude files matching patterns listed in FILE.

File name transformations

--strip-components=NUMBER

Strip NUMBER leading components from file names on extraction.

--transform=EXPRESSION, --xform=EXPRESSION

Use sed replace EXPRESSION to transform file names.

File name matching options

These options affect both exclude and include patterns.

--anchored

Patterns match file name start.

--ignore-case

Ignore case.

--no-anchored

Patterns match after any / (default for exclusion).

--no-ignore-case

Case sensitive matching (default).

--no-wildcards

Verbatim string matching.

--no-wildcards-match-slash

Wildcards do not match /.

--wildcards

Use wildcards (default for exclusion).

--wildcards-match-slash

Wildcards match / (default for exclusion).

Informative output

--checkpoint[=N]

Display progress messages every Nth record (default 10).

--checkpoint-action=ACTION

Run ACTION on each checkpoint.

--clamp-mtime

Only set time when the file is more recent than what was given with --mtime.

--full-time

Print file time to its full resolution.

--index-file=FILE

Send verbose output to FILE.

-l, --check-links

Print a message if not all links are dumped.

--no-quote-chars=STRING

Disable quoting for characters from STRING.

--quote-chars=STRING

Additionally quote characters from STRING.

--quoting-style=STYLE

Set quoting style for file and member names. Valid values for STYLE are literal, shell, shell-always, c, c-maybe, escape, lo? cale, clocale.

-R, --block-number

Show block number within archive with each message.

--show-omitted-dirs

When listing or extracting, list each directory that does not match search criteria.

--show-transformed-names, --show-stored-names

Show file or archive names after transformation by --strip and --transform options.

--totals[=SIGNAL]

Print total bytes after processing the archive. If SIGNAL is given, print total bytes when this signal is delivered. Allowed signals are: SIGHUP, SIGQUIT, SIGINT, SIGUSR1, and SIGUSR2. The SIG prefix can be omitted.

--utc Print file modification times in UTC.

-v, --verbose

Verbosely list files processed. Each instance of this option on the command line increases the verbosity level by one. The max? imum verbosity level is 3. For a detailed discussion of how various verbosity levels affect tar's output, please refer to GNU Tar Manual, subsection 2.5.1 "The --verbose Option".

--warning=KEYWORD

Enable or disable warning messages identified by KEYWORD. The messages are suppressed if KEYWORD is prefixed with no- and en? abled otherwise.

Multiple --warning messages accumulate.

Keywords controlling general tar operation:

all Enable all warning messages. This is the default.

none Disable all warning messages.

filename-with-nuls

"%s: file name read contains nul character"

alone-zero-block

"A lone zero block at %s"

Keywords applicable for tar --create:

cachedir

"%s: contains a cache directory tag %s; %s"

file-shrank

"%s: File shrank by %s bytes; padding with zeros"

xdev "%s: file is on a different filesystem; not dumped"

file-ignored

"%s: Unknown file type; file ignored"

"%s: socket ignored"

"%s: door ignored"

file-unchanged

"%s: file is unchanged; not dumped"

ignore-archive

"%s: file is the archive; not dumped"

file-removed

"%s: File removed before we read it"

file-changed

"%s: file changed as we read it"

failed-read

Suppresses warnings about unreadable files or directo?

ries. This keyword applies only if used together with the

--ignore-failed-read option.

Keywords applicable for tar --extract:

existing-file

"%s: skipping existing file"

timestamp

"%s: implausibly old time stamp %s"

"%s: time stamp %s is %s s in the future"

contiguous-cast

"Extracting contiguous files as regular files"

symlink-cast

"Attempting extraction of symbolic links as hard links"

unknown-cast

"%s: Unknown file type '%c', extracted as normal file"

ignore-newer

"Current %s is newer or same age"

unknown-keyword

"Ignoring unknown extended header keyword '%s'"

decompress-program

Controls verbose description of failures occurring when

trying to run alternative decompressor programs. This warning is disabled by default (unless --verbose is used). A common example of what you can get when using this warning is:

$ tar --warning=decompress-program -x -f archive.Z
tar (child): cannot run compress: No such file or directory
tar (child): trying gzip

This means that tar first tried to decompress archive.Z using compress, and, when that failed, switched to gzip.

record-size

"Record size = %lu blocks"

Keywords controlling incremental extraction:

rename-directory

"%s: Directory has been renamed from %s"

"%s: Directory has been renamed"

new-directory

"%s: Directory is new"

xdev "%s: directory is on a different device: not purging"

bad-dumpdir

"Malformed dumpdir: 'X' never used"

-w, --interactive, --confirmation

Ask for confirmation for every action.

Compatibility options

-o When creating, same as --old-archive. When extracting, same as --no-same-owner.

Size suffixes

Suffix	Units	Byte Equivalent
b	Blocks	SIZE x 512
B	Kilobytes	SIZE x 1024
c	Bytes	SIZE
G	Gigabytes	SIZE x 1024^3
K	Kilobytes	SIZE x 1024
k	Kilobytes	SIZE x 1024

M	Megabytes	SIZE x 1024^2
P	Petabytes	SIZE x 1024^5
T	Terabytes	SIZE x 1024^4
w	Words	SIZE x 2

RETURN VALUE

Tar exit code indicates whether it was able to successfully perform the requested operation, and if not, what kind of error occurred.

0 Successful termination.

1 Some files differ. If tar was invoked with the --compare (--diff, -d) command line option, this means that some files in the archive differ from their disk counterparts. If tar was given one of the --create, --append or --update options, this exit code means that some files were changed while being archived and so the resulting archive does not contain the exact copy of the file set.

2 Fatal error. This means that some fatal, unrecoverable error occurred.

If a subprocess that had been invoked by tar exited with a nonzero exit code, tar itself exits with that code as well. This can happen, for example, if a compression option (e.g. -z) was used and the external compressor program failed. Another example is rmt failure during backup to a remote device.

SEE ALSO

bzip2(1), compress(1), gzip(1), lzma(1), lzop(1), rmt(8), symlink(7), xz(1), zstd(1).

Complete tar manual: run info tar or use emacs(1) info mode to read it.

Online copies of GNU tar documentation in various formats can be found at:

http://www.gnu.org/software/tar/manual

BUG REPORTS

Report bugs to <bug-tar@gnu.org>.

COPYRIGHT

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/li?

censes/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.