NAME

mtree - map a directory hierarchy

SYNOPSIS

mtree [-**bCcDdejLlMnPqrStUuWx**] [-**i** | -**m**] [-**E** tags] [-**F** flavor] [-**f** spec] [-**I** tags] [-**K** keywords] [-**K** keywords] [-**N** dbdir] [-**O** onlyfile] [-**p** path] [-**R** keywords] [-**s** seed] [-**X** exclude-file]

DESCRIPTION

The **mtree** utility compares a file hierarchy against a specification, creates a specification for a file hierarchy, or modifies a specification.

The default action, if not overridden by command line options, is to compare the file hierarchy rooted in the current directory against a specification read from the standard input. Messages are written to the standard output for any files whose characteristics do not match the specification, or which are missing from either the file hierarchy or the specification.

The options are as follows:

-b	Suppress blank lines before entering and after exiting directories.
-C	Convert a specification into a format that's easier to parse with various tools. The input specification is read from standard input or from the file given by -f <i>spec</i> . In the output, each file or directory is represented using a single line (which might be very long). The full path name (beginning with "./") is always printed as the first field; -K , -k , and -R can be used to control which other keywords are printed; -E and -I can be used to control which files are printed; and the -S option can be used to sort the output.
-с	Print a specification for the file hierarchy originating at the current working directory (or the directory provided by -p <i>path</i>) to the standard output. The output is in a style using relative path names.
-D	As per -C , except that the path name is always printed as the last field instead of the first.
-d	Ignore everything except directory type files.
-E tags	Add the comma separated tags to the "exclusion" list. Non-directories with tags which are in the exclusion list are not printed with -C and -D .

-е	Don't complain about files that are in the file hierarchy, but not in the specification.
-F flavor	Set the compatibility flavor of the mtree utility. The <i>flavor</i> can be one of mtree , freebsd9 , or netbsd6 . The default is mtree . The freebsd9 and netbsd6 flavors attempt to preserve output compatibility and command line option backward compatibility with FreeBSD 9.0 and NetBSD 6.0 respectively.
-f spec	Read the specification from <i>file</i> , instead of from the standard input.
	If this option is specified twice, the two specifications are compared to each other rather than to the file hierarchy. The specifications will be sorted like output generated using -c . The output format in this case is somewhat reminiscent of comm(1), having "in first spec only", "in second spec only", and "different" columns, prefixed by zero, one and two TAB characters respectively. Each entry in the "different" column occupies two lines, one from each specification.
-I tags	Add the comma separated tags to the "inclusion" list. Non-directories with tags which are in the inclusion list are printed with -C and -D . If no inclusion list is provided, the default is to display all files.
-i	If specified, set the schg and/or sappnd flags.
-j	Indent the output 4 spaces each time a directory level is descended when creating a specification with the -c option. This does not affect either the /set statements or the comment before each directory. It does however affect the comment before the close of each directory. This is the equivalent of the -i option in the FreeBSD version of mtree .
-K keywords	Add the specified (whitespace or comma separated) keywords to the current set of keywords. If 'all' is specified, add all of the other keywords.
-k keywords	Use the type keyword plus the specified (whitespace or comma separated) keywords instead of the current set of keywords. If 'all' is specified, use all of the other keywords. If the type keyword is not desired, suppress it with -R <i>type</i> .
-L	Follow all symbolic links in the file hierarchy.
-l	Do "loose" permissions checks, in which more stringent permissions will match less stringent ones. For example, a file marked mode 0444 will pass a check for mode 0644. "Loose" checks apply only to read, write and execute permissions in

MTREE(8)	FreeBSD System Manager's Manual	MTREE(8)
	particular, if other bits like the sticky bit or suid/sgid bits are set eit specification or the file, exact checking will be performed. This op set at the same time as the -U or -u option.	her in the ption may not be
-M	Permit merging of specification entries with different types, with the taking precedence.	ne last entry
-m	If the schg and/or sappnd flags are specified, reset these flags. Not possible with securelevel less than 1 (i.e., in single user mode or we running in insecure mode). See init(8) for information on security	te that this is only hile the system is levels.
-n	Do not emit pathname comments when creating a specification. N comment is emitted before each directory and before the close of the when using the $-c$ option.	ormally a nat directory
-N dbdir	Use the user database text file <i>master.passwd</i> and group database to from <i>dbdir</i> , rather than using the results from the system's getpwna getgrnam(3) (and related) library calls.	ext file <i>group</i> am(3) and
-O onlypaths	Only include files included in this list of pathnames.	
-P	Don't follow symbolic links in the file hierarchy, instead consider itself in any comparisons. This is the default.	the symbolic link
-p path	Use the file hierarchy rooted in <i>path</i> , instead of the current director	ry.
-q	Quiet mode. Do not complain when a "missing" directory cannot because it already exists. This occurs when the directory is a symb	be created polic link.
-R keywords	Remove the specified (whitespace or comma separated) keywords set of keywords. If 'all' is specified, remove all of the other keywords	from the current ords.
-r	Remove any files in the file hierarchy that are not described in the	specification.
-8	When reading a specification into an internal data structure, sort the will affect the order of the output produced by the -C or -D options affect the order in which missing entries are created or reported whe tree is checked against a specification.	e entries. Sorting a, and will also ten a directory

The sort order is the same as that used by the -c option, which is that entries within

MTREE(8)	FreeBSD System Manager's Manual	MTREE(8)
	the same directory are sorted in the order used by strcmp(3), except the subdirectories sort after other entries. By default, if the -S option is not within the same directory are collected together (separated from entried directories), but not sorted.	at entries for ot used, entries es for other
-s seed	Display a single checksum to the standard error output that represents for which the keyword cksum was specified. The checksum is seeded specified value.	all of the files I with the
-t	Modify the modified time of existing files, the device type of devices, link targets, to match the specification.	and symbolic
-U	Same as -u except that a mismatch is not considered to be an error if i corrected.	t was
-u	Modify the owner, group, permissions, and flags of existing files, the devices, and symbolic link targets, to match the specification. Create directories, devices or symbolic links. User, group, and permissions r specified for missing directories to be created. Note that unless the -i given, the schg and sappnd flags will not be set, even if specified. If - these flags will be reset. Exit with a status of 0 on success, 2 if the file not match the specification, and 1 if any other error occurred.	device type of any missing nust all be option is m is given, e hierarchy did
-W	Don't attempt to set various file attributes such as the ownership, mod time when creating new directories or changing existing entries. This most useful when used in conjunction with -U or -u .	le, flags, or option will be
-X exclude-file	The specified file contains fnmatch(3) patterns matching files to be ex the specification, one to a line. If the pattern contains a '/' character, is matched against entire pathnames (relative to the starting directory); o will be matched against basenames only. Comments are permitted in <i>exclude-list</i> file.	cluded from it will be otherwise, it the
-X	Don't descend below mount points in the file hierarchy.	
G :C /:		

Specifications are mostly composed of "keywords", i.e. strings that that specify values relating to files. No keywords have default values, and if a keyword has no value set, no checks based on it are performed.

Currently supported keywords are as follows:

cksum	The checksum of the file using the default algorithm specified by the cksum(1) utility.
device	The device number to use for block or char file types. The argument must be one of the following forms:
	<i>format,major,minor</i> A device with <i>major</i> and <i>minor</i> fields, for an operating system specified with <i>format</i> . See below for valid formats.
	<i>format,major,unit,subunit</i> A device with <i>major, unit,</i> and <i>subunit</i> fields, for an operating system specified with <i>format.</i> (Currently this is only supported by the bsdos format.)
	number Opaque number (as stored on the file system).
	The following values for <i>format</i> are recognized: native , 386bsd , 4bsd , bsdos , freebsd , hpux , isc , linux , netbsd , osf1 , sco , solaris , sunos , svr3 , svr4 , and ultrix .
	See mknod(8) for more details.
flags	The file flags as a symbolic name. See $chflags(1)$ for information on these names. If no flags are to be set the string 'none' may be used to override the current default. Note that the schg and sappnd flags are treated specially (see the -i and -m options).
ignore	Ignore any file hierarchy below this file.
gid	The file group as a numeric value.
gname	The file group as a symbolic name.
link	The file the symbolic link is expected to reference.
md5	The MD5 cryptographic message digest of the file.
md5digest	Synonym for md5 .
mode	The current file's permissions as a numeric (octal) or symbolic value.
nlink	The number of hard links the file is expected to have.

nochange Make sure this file or directory exists but otherwise ignore all attributes.

optional The file is optional; don't complain about the file if it's not in the file hierarchy.

ripemd160digest

Synonym for rmd160.

rmd160 The RMD-160 cryptographic message digest of the file.

- **rmd160digest** Synonym for **rmd160**.
- sha1 The SHA-1 cryptographic message digest of the file.
- **sha1digest** Synonym for **sha1**.
- sha256 The 256-bits SHA-2 cryptographic message digest of the file.
- sha256digest Synonym for sha256.
- sha384 The 384-bits SHA-2 cryptographic message digest of the file.
- sha384digest Synonym for sha384.
- sha512 The 512-bits SHA-2 cryptographic message digest of the file.
- sha512digest Synonym for sha512.
- size The size, in bytes, of the file.
- **tags** Comma delimited tags to be matched with **-E** and **-I**. These may be specified without leading or trailing commas, but will be stored internally with them.
- timeThe last modification time of the file, in second and nanoseconds. The value should
include a period character and exactly nine digits after the period.
- **type** The type of the file; may be set to any one of the following:
 - blockblock special devicecharcharacter special device
 - dir directory

uname

fifo fifo file regular file link symbolic link socket socket uid The file owner as a numeric value.

The file owner as a symbolic name.

The default set of keywords are flags, gid, link, mode, nlink, size, time, type, and uid.

There are four types of lines in a specification:

- 1. Set global values for a keyword. This consists of the string '/set' followed by whitespace, followed by sets of keyword/value pairs, separated by whitespace. Keyword/value pairs consist of a keyword, followed by an equals sign ('='), followed by a value, without whitespace characters. Once a keyword has been set, its value remains unchanged until either reset or unset.
- 2. Unset global values for a keyword. This consists of the string '/unset', followed by whitespace, followed by one or more keywords, separated by whitespace. If 'all' is specified, unset all of the keywords.
- 3. A file specification, consisting of a path name, followed by whitespace, followed by zero or more whitespace separated keyword/value pairs.

The path name may be preceded by whitespace characters. The path name may contain any of the standard path name matching characters ('[', ']', '?' or '*'), in which case files in the hierarchy will be associated with the first pattern that they match. mtree uses strsvis(3) (in VIS_CSTYLE format) to encode path names containing non-printable characters. Whitespace characters are encoded as '\s' (space), '\t' (tab), and '\n' (new line). '#' characters in path names are escaped by a preceding backslash '\' to distinguish them from comments.

Each of the keyword/value pairs consist of a keyword, followed by an equals sign ('='), followed by the keyword's value, without whitespace characters. These values override, without changing, the global value of the corresponding keyword.

The first path name entry listed must be a directory named '.', as this ensures that intermixing full and relative path names will work consistently and correctly. Multiple entries for a directory named '.' are permitted; the settings for the last such entry override those of the existing entry.

A path name that contains a slash ('/') that is not the first character will be treated as a full path (relative to the root of the tree). All parent directories referenced in the path name must exist. The current directory path used by relative path names will be updated appropriately. Multiple entries for the same full path are permitted if the types are the same (unless -**M** is given, in which case the types may differ); in this case the settings for the last entry take precedence.

A path name that does not contain a slash will be treated as a relative path. Specifying a directory will cause subsequent files to be searched for in that directory hierarchy.

4. A line containing only the string '..' which causes the current directory path (used by relative paths) to ascend one level.

Empty lines and lines whose first non-whitespace character is a hash mark ('#') are ignored.

The **mtree** utility exits with a status of 0 on success, 1 if any error occurred, and 2 if the file hierarchy did not match the specification.

FILES

/etc/mtree system specification directory

EXAMPLES

To detect system binaries that have been "trojan horsed", it is recommended that **mtree** be run on the file systems, and a copy of the results stored on a different machine, or, at least, in encrypted form. The seed for the **-s** option should not be an obvious value and the final checksum should not be stored on-line under any circumstances! Then, periodically, **mtree** should be run against the on-line specifications and the final checksum compared with the previous value. While it is possible for the bad guys to change the on-line specifications to conform to their modified binaries, it shouldn't be possible for them to make it produce the same final checksum value. If the final checksum value changes, the off-line copies of the specification can be used to detect which of the binaries have actually been modified.

The **-d** option can be used in combination with **-U** or **-u** to create directory hierarchies for, for example, distributions.

COMPATIBILITY

The compatibility shims provided by the -F option are incomplete by design. Known limitations are described below.

The **freebsd9** flavor retains the default handling of lookup failures for the **uname** and **group** keywords by replacing them with appropriate **uid** and **gid** keywords rather than failing and reporting an error. The related **-w** flag is a no-op rather than causing a warning to be printed and no keyword to be emitted. The

latter behavior is not emulated as it is potentially dangerous in the face of /set statements.

The **netbsd6** flavor does not replicate the historical bug that reported time as seconds.nanoseconds without zero padding nanosecond values less than 100000000.

SEE ALSO

chflags(1), chgrp(1), chmod(1), cksum(1), stat(2), fnmatch(3), fts(3), strsvis(3), mtree(5), chown(8), mknod(8)

HISTORY

The **mtree** utility appeared in 4.3BSD-Reno. The **optional** keyword appeared in NetBSD 1.2. The **-U** option appeared in NetBSD 1.3. The **flags** and **md5** keywords, and **-i** and **-m** options appeared in NetBSD 1.4. The **device**, **rmd160**, **sha1**, **tags**, and **all** keywords, **-D**, **-E**, **-I**, **-L**, **-I**, **-N**, **-P**, **-R**, **-W**, and **-X** options, and support for full paths appeared in NetBSD 1.6. The **sha256**, **sha384**, and **sha512** keywords appeared in NetBSD 3.0. The **-S** option appeared in NetBSD 6.0.