NAME

strip − discard symbols and other data from object files

SYNOPSIS

strip [**−F** *bfdname* |**−−target=***bfdname*]
[**−I** *bfdname* |**−−input−target=***bfdname*]
[**−O** *bfdname* |**−−output−target=***bfdname*]
[**−s**|**−−strip−all**]
[**−S**|**−g**|**−d**|**−−strip−debug**]
[**−−strip−dwo**]
[**−K** *symbolname*|**−−keep−symbol=***symbolname*]
[**−M**|**−−merge−notes**][**−−no−merge−notes**]
[**−N** *symbolname* |**−−strip−symbol=***symbolname*]
[**−w**|**−−wildcard**]
[**−x**|**−−discard−all**] [**−X** |**−−discard−locals**]
[**−R** *sectionname* |**−−remove−section=***sectionname*]
[**−−keep−section=***sectionpattern*]
[**−−remove−relocations=***sectionpattern*]
[**−−strip−section−headers**]
[**−o** *file*] [**−p**|**−−preserve−dates**]
[**−D**|**−−enable−deterministic−archives**]
[**−U**|**−−disable−deterministic−archives**]
[**−−keep−section−symbols**]
[**−−keep−file−symbols**]
[**−−only−keep−debug**]
[**−−plugin** *name*]
[**−v** |**−−verbose**] [**−V**|**−−version**]
[**−−help**] [**−−info**]
objfile...

DESCRIPTION

GNU **strip** discards all symbols from object files *objfile*. The list of object files may include archives. At least one object file must be given.

strip modifies the files named in its argument, rather than writing modified copies under different names.

OPTIONS

−F *bfdname*
−−target=*bfdname*
Treat the original *objfile* as a file with the object code format *bfdname*, and rewrite it in the same format.

−−help
Show a summary of the options to **strip** and exit.

−−info
Display a list showing all architectures and object formats available.

−I *bfdname*
−−input−target=*bfdname*
Treat the original *objfile* as a file with the object code format *bfdname*.

−O *bfdname*
−−output−target=*bfdname*
Replace *objfile* with a file in the output format *bfdname*.

−R *sectionname*
−−remove−section=*sectionname*
Remove any section named *sectionname* from the output file, in addition to whatever sections would otherwise be removed. This option may be given more than once. Note that using this option

inappropriately may make the output file unusable. The wildcard character ***** may be given at the end of *sectionname*. If so, then any section starting with *sectionname* will be removed.

If the first character of *sectionpattern* is the exclamation point (!) then matching sections will not be removed even if an earlier use of **--remove-section** on the same command line would otherwise remove it. For example:

```
--remove-section=.text.* --remove-section=!.text.foo
```

will remove all sections matching the pattern '.text.*', but will not remove the section '.text.foo'.

--keep-section=*sectionpattern*
When removing sections from the output file, keep sections that match *sectionpattern*.

--remove-relocations=*sectionpattern*
Remove relocations from the output file for any section matching *sectionpattern*. This option may be given more than once. Note that using this option inappropriately may make the output file unusable. Wildcard characters are accepted in *sectionpattern*. For example:

```
--remove-relocations=.text.*
```

will remove the relocations for all sections matching the patter '.text.*'.

If the first character of *sectionpattern* is the exclamation point (!) then matching sections will not have their relocation removed even if an earlier use of **--remove-relocations** on the same command line would otherwise cause the relocations to be removed. For example:

```
--remove-relocations=.text.* --remove-relocations=!.text.foo
```

will remove all relocations for sections matching the pattern '.text.*', but will not remove relocations for the section '.text.foo'.

--strip-section-headers
Strip section headers. This option is specific to ELF files. Implies **--strip-all** and **--merge-notes**.

-s
--strip-all
Remove all symbols.

-g
-S
-d
--strip-debug
Remove debugging symbols only.

--strip-dwo
Remove the contents of all DWARF .dwo sections, leaving the remaining debugging sections and all symbols intact. See the description of this option in the **objcopy** section for more information.

--strip-unneeded
Remove all symbols that are not needed for relocation processing in addition to debugging symbols and sections stripped by **--strip-debug**.

-K *symbolname*
--keep-symbol=*symbolname*
When stripping symbols, keep symbol *symbolname* even if it would normally be stripped. This option may be given more than once.

-M
--merge-notes
--no-merge-notes
For ELF files, attempt (or do not attempt) to reduce the size of any SHT_NOTE type sections by removing duplicate notes. The default is to attempt this reduction unless stripping debug or DWO information.

−N *symbolname*
−−strip−symbol=*symbolname*
Remove symbol *symbolname* from the source file. This option may be given more than once, and may be combined with strip options other than **−K**.

−o *file*
Put the stripped output in *file*, rather than replacing the existing file. When this argument is used, only one *objfile* argument may be specified.

−p
−−preserve−dates
Preserve the access and modification dates of the file.

−D
−−enable−deterministic−archives
Operate in *deterministic* mode. When copying archive members and writing the archive index, use zero for UIDs, GIDs, timestamps, and use consistent file modes for all files.

If *binutils* was configured with **−−enable−deterministic−archives**, then this mode is on by default. It can be disabled with the **−U** option, below.

−U
−−disable−deterministic−archives
Do *not* operate in *deterministic* mode. This is the inverse of the **−D** option, above: when copying archive members and writing the archive index, use their actual UID, GID, timestamp, and file mode values.

This is the default unless *binutils* was configured with **−−enable−deterministic−archives**.

−w
−−wildcard
Permit regular expressions in *symbolname*s used in other command line options. The question mark (?), asterisk (*), backslash (\) and square brackets ([]) operators can be used anywhere in the symbol name. If the first character of the symbol name is the exclamation point (!) then the sense of the switch is reversed for that symbol. For example:

```
−w −K !foo −K fo*
```

would cause strip to only keep symbols that start with the letters "fo", but to discard the symbol "foo".

−x
−−discard−all
Remove non-global symbols.

−X
−−discard−locals
Remove compiler-generated local symbols. (These usually start with **L** or **.**.)

−−keep−section−symbols
When stripping a file, perhaps with **−−strip−debug** or **−−strip−unneeded**, retain any symbols specifying section names, which would otherwise get stripped.

−−keep−file−symbols
When stripping a file, perhaps with **−−strip−debug** or **−−strip−unneeded**, retain any symbols specifying source file names, which would otherwise get stripped.

−−only−keep−debug
Strip a file, emptying the contents of any sections that would not be stripped by **−−strip−debug** and leaving the debugging sections intact. In ELF files, this preserves all the note sections in the output as well.

Note − the section headers of the stripped sections are preserved, including their sizes, but the contents of the section are discarded. The section headers are preserved so that other tools can match up the

debuginfo file with the real executable, even if that executable has been relocated to a different address space.

The intention is that this option will be used in conjunction with **--add-gnu-debuglink** to create a two part executable. One a stripped binary which will occupy less space in RAM and in a distribution and the second a debugging information file which is only needed if debugging abilities are required. The suggested procedure to create these files is as follows:

1. Link the executable as normal. Assuming that it is called `foo` then...
2. Run `objcopy --only-keep-debug foo foo.dbg` to create a file containing the debugging info.
3. Run `strip --strip-debug foo` to create a stripped executable.
4. Run `objcopy --add-gnu-debuglink=foo.dbg foo` to add a link to the debugging info into the stripped executable.

Note---the choice of `.dbg` as an extension for the debug info file is arbitrary. Also the `--only-keep-debug` step is optional. You could instead do this:

1. Link the executable as normal.
2. Copy `foo` to `foo.full`.
3. Run `strip --strip-debug foo`.
4. Run `objcopy --add-gnu-debuglink=foo.full foo`.

i.e., the file pointed to by the **--add-gnu-debuglink** can be the full executable. It does not have to be a file created by the **--only-keep-debug** switch.

Note---this switch is only intended for use on fully linked files. It does not make sense to use it on object files where the debugging information may be incomplete. Besides the gnu_debuglink feature currently only supports the presence of one filename containing debugging information, not multiple filenames on a one-per-object-file basis.

--plugin *name*
Load the plugin called *name* to add support for extra target types. This option is only available if the toolchain has been built with plugin support enabled.

If **--plugin** is not provided, but plugin support has been enabled then **strip** iterates over the files in *${libdir}/bfd-plugins* in alphabetic order and the first plugin that claims the object in question is used.

Please note that this plugin search directory is *not* the one used by **ld**'s **-plugin** option. In order to make **strip** use the linker plugin it must be copied into the *${libdir}/bfd-plugins* directory. For GCC based compilations the linker plugin is called *liblto_plugin.so.0.0.0*. For Clang based compilations it is called *LLVMgold.so*. The GCC plugin is always backwards compatible with earlier versions, so it is sufficient to just copy the newest one.

-V
--version
Show the version number for **strip**.

-v
--verbose
Verbose output: list all object files modified. In the case of archives, **strip -v** lists all members of the archive.

@*file*
Read command-line options from *file*. The options read are inserted in place of the original @*file* option. If *file* does not exist, or cannot be read, then the option will be treated literally, and not removed.

Options in *file* are separated by whitespace. A whitespace character may be included in an option by

surrounding the entire option in either single or double quotes. Any character (including a backslash) may be included by prefixing the character to be included with a backslash. The *file* may itself contain additional @*file* options; any such options will be processed recursively.

SEE ALSO

the Info entries for *binutils*.

COPYRIGHT