



Rocky Enterprise Linux 9.2 Manual Pages on command 'go-build.1'

C:\>man go-build.1

GO-BUILD(1) General Commands Manual GO-BUILD(1)

NAME

go - tool for managing Go source code

SYNOPSIS

go build [-o output] [build flags] [packages]

DESCRIPTION

Build compiles the packages named by the import paths, along with their dependencies, but it does not install the results.

If the arguments are a list of .go files, build treats them as a list of source files specifying a single package.

When the command line specifies a single main package, build writes the resulting executable to output. Otherwise build compiles the packages but discards the results, serving only as a check that the packages can be built.

The -o flag specifies the output file name. If not specified, the name is package.name.a (for a non-main package) or the base name of the first source file (for a main package).

OPTIONS

The build flags are shared by the build, install, run, and test commands:

- a force rebuilding of packages that are already up-to-date.
- n print the commands but do not run them.
- p n the number of builds that can be run in parallel. The default is the number of CPUs available.

- v print the names of packages as they are compiled.
- work print the name of the temporary work directory and do not delete it when exiting.
- x print the commands.
- compiler name
name of compiler to use, as in runtime.Compiler (gccgo or gc)
- gccgoflags 'arg list'
arguments to pass on each gccgo compiler/linker invocation
- gcflags 'arg list'
arguments to pass on each 5g, 6g, or 8g compiler invocation
- ldflags 'flag list'
arguments to pass on each 5l, 6l, or 8l linker invocation
- tags 'tag list'
a list of build tags to consider satisfied during the build. See the documentation for the go/build package for more information about build tags.
For more about specifying packages, see go-packages(7).
For more about where packages and binaries are installed, see go-gopath(1).

SEE ALSO

go-install(1), go-get(1), go-clean(1).

AUTHOR

This manual page was written by Michael Stapelberg <stapelberg@debian.org>, for the Debian project (and may be used by others).

2012-05-13

GO-BUILD(1)