



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

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

\$ man go-gopath-get.1

GO-GOPATH-GET(1) General Commands Manual GO-GOPATH-GET(1)

NAME

go-get - legacy GOPATH go get

SYNOPSIS

The 'go get' command changes behavior depending on whether the go command is running in module-aware mode or legacy GOPATH mode. This help text, accessible as 'go help gopath-get' even in module-aware mode, describes 'go get' as it operates in legacy GOPATH mode.

go get [-d] [-f] [-t] [-u] [-v] [-fix] [build flags] [packages]

DESCRIPTION

Get downloads the packages named by the import paths, along with their dependencies. It then installs the named packages, like 'go install'.

- d The -d flag instructs get to stop after downloading the packages; that is, it instructs get not to install the packages.
- f The -f flag, valid only when -u is set, forces get -u not to verify that each package has been checked out from the source control repository implied by its import path. This can be useful if the source is a local fork of the original.
- fix The -fix flag instructs get to run the fix tool on the downloaded packages before resolving dependencies or building the code.
- t The -t flag instructs get to also download the packages required to build the tests for the specified packages.
- u The -u flag instructs get to use the network to update the named packages and their dependencies. By default, get uses the network to check out missing packages but does not use it to look for updates to existing packages.

-v The -v flag enables verbose progress and debug output.

Get also accepts build flags to control the installation. See go-build(1).

When checking out a new package, get creates the target directory GOPATH/src/<import-path>. If the GOPATH contains multiple entries, get uses the first one. For more details see: 'go help gopath'.

When checking out or updating a package, get looks for a branch or tag that matches the locally installed version of Go. The most important rule is that if the local installation is running version "go1", get searches for a branch or tag named "go1". If no such version exists it retrieves the default branch of the package.

When go get checks out or updates a Git repository, it also updates any git submodules referenced by the repository.

Get never checks out or updates code stored in vendor directories.

For more about specifying packages, see go-packages(7).

For more about how 'go get' finds source code to download, see go-remote(7).

SEE ALSO

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

AUTHOR

This manual page was written by Michael Stapelberg <stapelberg@debian.org> and is maintained by the Debian Go Compiler Team <team+go-compiler@tracker.debian.org> based on the output of 'go help gopath-get' for the Debian project (and may be used by others).

2021-10-15

GO-GOPATH-GET(1)