

# pack install

## In this article

- Synopsis
- Description
- Options

[Experimental] Install dependencies for this pack.

GitHub CodeQL is licensed on a per-user basis upon installation. You can use CodeQL only for certain tasks under the license restrictions. For more information, see "[About the CodeQL CLI](#)." If you have a GitHub Advanced Security license, you can use CodeQL for automated analysis, continuous integration, and continuous delivery. For more information, see "[About GitHub Advanced Security](#)."

This content describes the most recent release of the CodeQL CLI. For more information about this release, see <https://github.com/github/codeql-cli-binaries/releases>.

To see details of the options available for this command in an earlier release, run the command with the `--help` option in your terminal.

## Synopsis

Shell

```
codeql pack install [--force] <options>... -- <dir>
```

## Description

[Experimental] Install dependencies for this pack.

This command installs the dependencies of the pack. If a `codeql-pack.lock.yml` exists, the versions specified in that lock file will be installed if they are compatible with the version constraints specified in the `qlpack.yml` file. If any dependency versions specified in the lock file are incompatible with this pack's constraints, this command will install the latest compatible version of each of those dependencies, updating the lock file with the newly chosen versions. If no lock file is present, this command installs the latest compatible version of all dependencies, creating a new lock file.

Available since `v2.6.0`.

## Options

### Primary Options

**<dir>** [↗](#)

The root directory of the package.

**--format=<fmt>** [↗](#)

Select output format, either `text` (*default*) or `json`.

**-f, --[no-]force** [↗](#)

Allow overwriting already existing packs.

**--[no-]allow-prerelease** [↗](#)

Allow packs with pre-release version qualifiers (e.g., `X.Y.Z-qualifier`) to be used. Without this flag, pre-release packs will be ignored.

Available since `v2.11.3`.

**--lock-override=<file>** [↗](#)

[Advanced] Specifies an alternate lock file to use as the input to dependency resolution.

**--lock-output=<file>** [↗](#)

[Advanced] Specifies an alternate location to save the lock file generated by dependency resolution.

Available since `v2.14.1`.

**--no-strict-mode** [↗](#)

[Advanced] Turn off strict mode to avoid a warning when resolving packages from the `--additional-packs`

and other locally resolved locations. Packages resolved locally are never downloaded and will not be added to the package lock.

**--mode=<mode>** [↗](#)

[Deprecated] Specifies how to resolve dependencies:

`minimal-update` (*default*): Update or create the `codeql-pack.lock.yml` based on the existing contents of the `qlpack.yml` file. If any existing `codeql-pack.lock.yml` does not satisfy the current dependencies in the `qlpack.yml`, the lock file will be updated as necessary.

`upgrade` : Update or create the `codeql-pack.lock.yml` to use the latest versions of all dependencies, subject to the constraints in the `qlpack.yml` file. This mode has been superseded by the 'codeql pack upgrade' command.

`verify` : Verify that the existing `codeql-pack.lock.yml` is still valid with respect to the dependencies specified in the `qlpack.yml` file, or fail the lock file if it does not exist. This mode has been superseded by the 'codeql pack ci' command.

`no-lock` : Ignore the existing `codeql-pack.lock.yml` and perform resolution based on `qlpack.yml` file. Does not create or update the lock file.

`use-lock` : Use the existing `codeql-pack.lock.yml` file to resolve dependencies, or create the lock file if it does not exist.

`update` : [Deprecated] Update or create the `codeql-pack.lock.yml` to use the latest versions of all dependencies, subject to the constraints in the `qlpack.yml` file. This mode has been superseded by the `'codeql pack upgrade'` command.

## Options for resolving QL packs outside of the package registry [↗](#)

`--search-path=<dir>[:<dir>...]` [↗](#)

A list of directories under which QL packs may be found. Each directory can either be a QL pack (or bundle of packs containing a `.codeqlmanifest.json` file at the root) or the immediate parent of one or more such directories.

If the path contains more than one directory, their order defines precedence between them: when a pack name that must be resolved is matched in more than one of the directory trees, the one given first wins.

Pointing this at a checkout of the open-source CodeQL repository ought to work when querying one of the languages that live there.

If you have checked out the CodeQL repository as a sibling of the unpacked CodeQL toolchain, you don't need to give this option; such sibling directories will always be searched for QL packs that cannot be found otherwise. (If this default does not work, it is strongly recommended to set up `--search-path` once and for all in a per-user configuration file).

(Note: On Windows the path separator is `;`).

`--additional-packs=<dir>[:<dir>...]` [↗](#)

If this list of directories is given, they will be searched for packs before the ones in `--search-path`. The order between these doesn't matter; it is an error if a pack name is found in two different places through this list.

This is useful if you're temporarily developing a new version of a pack that also appears in the default path. On the other hand, it is *not recommended* to override this option in a config file; some internal actions will add this option on the fly, overriding any configured value.

(Note: On Windows the path separator is `;`).

## Options for configuring the CodeQL package manager [↗](#)

`--registries-auth-stdin` [↗](#)

Authenticate to GitHub Enterprise Server Container registries by passing a comma-separated list of `<registry_url>=<token>` pairs.

For example, you can pass

```
https://containers.GHEHOSTNAME1/v2/=TOKEN1,https://containers.GHEHOSTNAME2/v2/=TOKEN2
```

 to authenticate to two GitHub Enterprise Server instances.

This overrides the `CODEQL_REGISTRIES_AUTH` and `GITHUB_TOKEN` environment variables. If you only need to authenticate to the `github.com` Container registry, you can instead authenticate using the simpler `--github-auth-stdin` option.

`--github-auth-stdin` [↗](#)

Authenticate to the `github.com` Container registry by passing a `github.com` GitHub Apps

token or personal access token via standard input.

To authenticate to GitHub Enterprise Server Container registries, pass `--registries-auth-stdin` or use the `CODEQL_REGISTRIES_AUTH` environment variable.

This overrides the `GITHUB_TOKEN` environment variable.

## Common options [↗](#)

`-h, --help` [↗](#)

Show this help text.

`-J=<opt>` [↗](#)

[Advanced] Give option to the JVM running the command.

(Beware that options containing spaces will not be handled correctly.)

`-v, --verbose` [↗](#)

Incrementally increase the number of progress messages printed.

`-q, --quiet` [↗](#)

Incrementally decrease the number of progress messages printed.

`--verbosity=<level>` [↗](#)

[Advanced] Explicitly set the verbosity level to one of errors, warnings, progress, progress+, progress++, progress+++. Overrides `-v` and `-q`.

`--logdir=<dir>` [↗](#)

[Advanced] Write detailed logs to one or more files in the given directory, with generated names that include timestamps and the name of the running subcommand.

(To write a log file with a name you have full control over, instead give `--log-to-stderr` and redirect stderr as desired.)

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)