Configuring the dependency graph

In this article

You can allow users to identify their projects' dependencies by enabling the dependency graph.

About the dependency graph

The dependency graph is a summary of the manifest and lock files stored in a repository and any dependencies that are submitted for the repository using the Dependency submission API (beta). For each repository, it shows dependencies, that is, the ecosystems and packages it depends on.

GitHub Enterprise Server does not calculate information about dependents, the repositories and packages that depend on a repository.

For more information, see "About the dependency graph."

Enabling the dependency graph

If the dependency graph is not available in your system, your enterprise owner can enable the dependency graph. For more information, see "Enabling the dependency graph for your enterprise."

When the dependency graph is first enabled, any manifest and lock files for supported ecosystems are parsed immediately. The graph is usually populated within minutes but this may take longer for repositories with many dependencies. Once enabled, the graph is automatically updated with every push to the repository.

Additionally, you can use the Dependency submission API (beta) to submit dependencies from the package manager or ecosystem of your choice, even if the ecosystem is not supported by dependency graph for manifest or lock file analysis. The dependency graph will display the submitted dependencies grouped by ecosystem, but separately from the dependencies parsed from manifest or lock files. For more information on the Dependency submission API, see "Using the Dependency submission API."

Further reading

- "Viewing and updating Dependabot alerts"
- "Troubleshooting the detection of vulnerable dependencies"