Configuring code scanning for your appliance

In this article

You can enable, configure and disable code scanning for your GitHub Enterprise Server instance. Code scanning allows users to scan code for vulnerabilities and errors.

Code scanning is available for organization-owned repositories in GitHub Enterprise Server. This feature requires a license for GitHub Advanced Security. For more information, see "About GitHub Advanced Security."

About code scanning

Code scanning is a feature that you use to analyze the code in a GitHub repository to find security vulnerabilities and coding errors. Any problems identified by the analysis are shown in GitHub Enterprise Server.

You can configure code scanning to run CodeQL analysis and third-party analysis. Code scanning also supports running analysis natively using GitHub Actions or externally using existing CI/CD infrastructure. The bullets below summarize the options available to users when you configure your GitHub Enterprise Server instance to allow code scanning using actions.

- **CodeQL**: Uses GitHub Actions with either default setup (see "Configuring default setup for code scanning") or advanced setup (see "Configuring advanced setup for code scanning"), or runs CodeQL analysis in a third-party continuous integration (CI) system (see "Using code scanning with your existing CI system").
- **Third-party**: Uses GitHub Actions or third-party tools and uploads results to GitHub Enterprise Server (see "Uploading a SARIF file to GitHub").

Checking whether your license includes GitHub Advanced Security

You can identify if your enterprise has a GitHub Advanced Security license by reviewing your enterprise settings. For more information, see "Enabling GitHub Advanced Security for your enterprise."

Prerequisites for code scanning

- A license for GitHub Advanced Security (see "About billing for GitHub Advanced Security")
- Code scanning enabled in the management console (see "Enabling GitHub Advanced Security for your enterprise")
- A VM or container for code scanning analysis to run in.

Running code scanning using GitHub Actions

Setting up a self-hosted runner

GitHub Enterprise Server can run code scanning using a GitHub Actions workflow. First, you need to provision one or more self-hosted GitHub Actions runners in your environment. You can provision self-hosted runners at the repository, organization, or enterprise account level. For more information, see "About self-hosted runners" and "Adding self-hosted runners."

If you are provisioning a self-hosted runner for CodeQL analysis, your runner must use a CodeQL-supported operating system version and CPU architecture. For more information, see the CodeQL system requirements.

If you are using default setup for code scanning, assign the `code-scanning` label to your self-hosted runner. For more information about using labels with self-hosted runners, see "Using labels with self-hosted runners." For more information about using default setup for code scanning analysis of compiled languages, see "CodeQL code scanning for compiled languages."

You must ensure that Git is in the PATH variable on any self-hosted runners you use to run CodeQL actions.

> If you use CodeQL code scanning to analyze code written in Python in your enterprise, you must make sure that your self-hosted runner has Python 3 installed.

Provisioning the actions for code scanning

If you want to use actions to run code scanning on GitHub Enterprise Server, the actions must be available on your appliance.

The CodeQL action is included in your installation of GitHub Enterprise Server. If both GitHub Enterprise Server 3.10 and your GitHub Actions runner have access to the internet, the action will automatically download the CodeQL 2.13.5 bundle required to perform analysis. Alternatively, you can use a synchronization tool to make the latest released version of the CodeQL analysis bundle available locally. For more information, see "Configuring CodeQL analysis on a server without internet access" below.

You can also make third-party actions available to users for code scanning, by setting up GitHub Connect. For more information, see "Configuring code scanning for your appliance" below.

Configuring CodeQL analysis on a server without internet access

If the server on which you are running GitHub Enterprise Server is not connected to the internet, and you want to allow users to enable CodeQL code scanning for their

repositories, you must use the CodeQL action sync tool to copy the CodeQL analysis bundle from GitHub.com to your server. The tool, and details of how to use it, are available at https://github.com/github/codeql-action-sync-tool.

If you configure the CodeQL action sync tool, you can use it to sync the latest releases of the CodeQL action and associated CodeQL analysis bundle. These are compatible with GitHub Enterprise Server.

Configuring GitHub Connect to sync GitHub Actions

1. If you want to download action workflows on demand from GitHub.com, you need to enable GitHub Connect. For more information, see "Managing GitHub Connect."
2. You'll also need to enable GitHub Actions for your GitHub Enterprise Server instance. For more information, see "Getting started with GitHub Actions for GitHub Enterprise Server."
3. The next step is to configure access to actions on GitHub.com using GitHub Connect. For more information, see "Enabling automatic access to GitHub.com actions using GitHub Connect."
4. Add a self-hosted runner to your repository, organization, or enterprise account. For more information, see "Adding self-hosted runners."

Running code scanning using the CodeQL CLI

If you don't want to use GitHub Actions, you should run code scanning using the CodeQL CLI.

The CodeQL CLI is a command-line tool that you use to analyze codebases on any machine, including a third-party CI/CD system. For more information, see "Using code scanning with your existing CI system."