Reading: Using CodeQL on GitHub

Estimated effort: 6 mins

In this reading, you will describe CodeQL and how it works. You will also list the languages it supports.

Prerequisites:

- You should be familiar with using GitHub as a repository for your code.
- You should be familiar with the security lapses that occur and the importance of handling them along with the development.

What is CodeQL?

Ensuring security analysis of your code is of key importance. With the continuous evolution of the software development landscape, it is crucial to not just establish security on the premises but also keep it up-to-date to safeguard the system against any new vulnerabilities. Here, CodeQL in GitHub acts as a solution to security analysis.

How does CodeQL work?

CodeQL treats the code in the repository as data. When you add, push, and commit code into the repository, you add it to the database. Once the security analysis is done, it runs queries on the database, which in this case is the code itself, and the query results are displayed as code scanning alerts. You can either run default queries or configure specific queries according to the project's requirements.

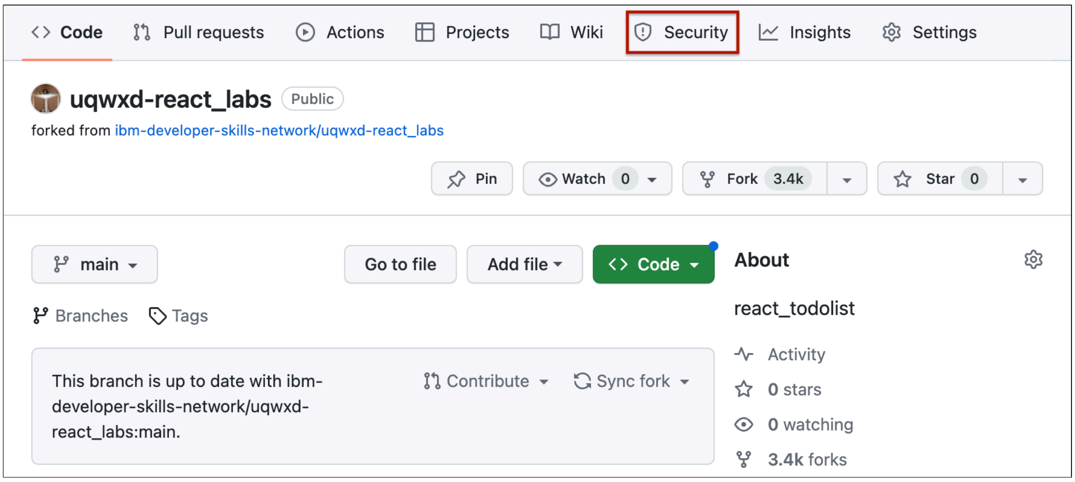
What languages does CodeQL support?

You might be required to use some specific language or more than one language for your code repository. CodeQL has the capability to perform security analysis for various compiled and interpreted languages, such as:

- C
- C++
- C#
- Java and Kotlin
- Go
- Ruby
- Python
- JS/TS
- Kotlin
- Swift

How to use CodeQL?

Once you add, push, and commit the code in the GitHub repository, you can go to the **Security** tab in the repository, where you can configure and view the security analysis.



uqwxd-react_labs Public

forked from ibm-developer-skills-network/uqwxd-react_labs

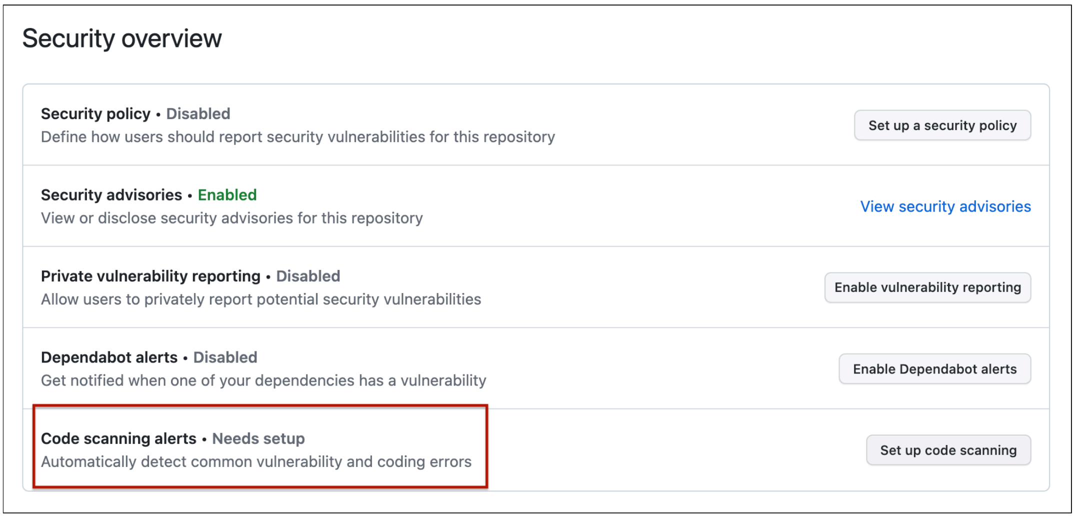
This branch is up to date with ibm-developer-skills-network/uqwxd-react_labs:main.

About

react_todolist

Activity
0 stars
0 watching
3.4k forks

When you initially navigate to the **Security** tab, you will see the **Security Overview**. Here, you will observe that the **Code scanning alerts** requires a setup. It is recommended to have this set up in order to keep a check on the vulnerabilities as you develop the code.



Security overview

Security policy • Disabled
Define how users should report security vulnerabilities for this repository

Security advisories • Enabled
View or disclose security advisories for this repository

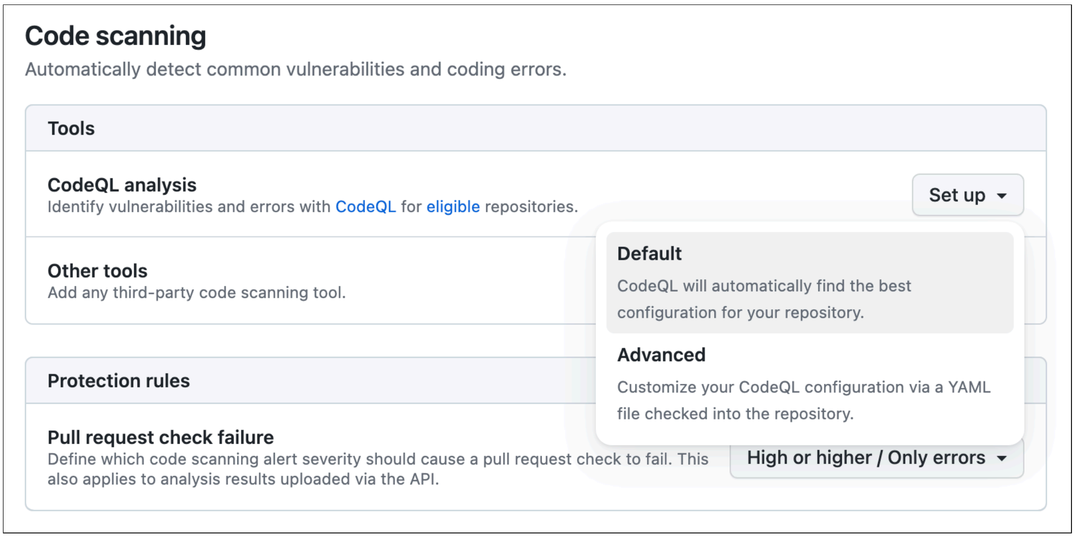
Private vulnerability reporting • Disabled
Allow users to privately report potential security vulnerabilities

Dependabot alerts • Disabled
Get notified when one of your dependencies has a vulnerability

Code scanning alerts • Needs setup
Automatically detect common vulnerability and coding errors

To set up code scanning, you can select `Set up code scanning`. This takes you to a configuration panel. In the configuration panel, you will notice that CodeQL is provided by GitHub for security analysis. You will also have other tools like Snyk, but to use other such tools, you will need to integrate the GitHub repository with the tool. However, CodeQL, being a GitHub tool, can be configured easily. It allows two different modes of configuration:

- Default: CodeQL automatically finds the best configuration for your repository based on the language.
- Advanced: CodeQL allows you to manually create a YAML file and include more security checks that are relevant to the code.



Code scanning

Automatically detect common vulnerabilities and coding errors.

Tools

CodeQL analysis

Identify vulnerabilities and errors with CodeQL for eligible repositories.

Other tools

Add any third-party code scanning tool.

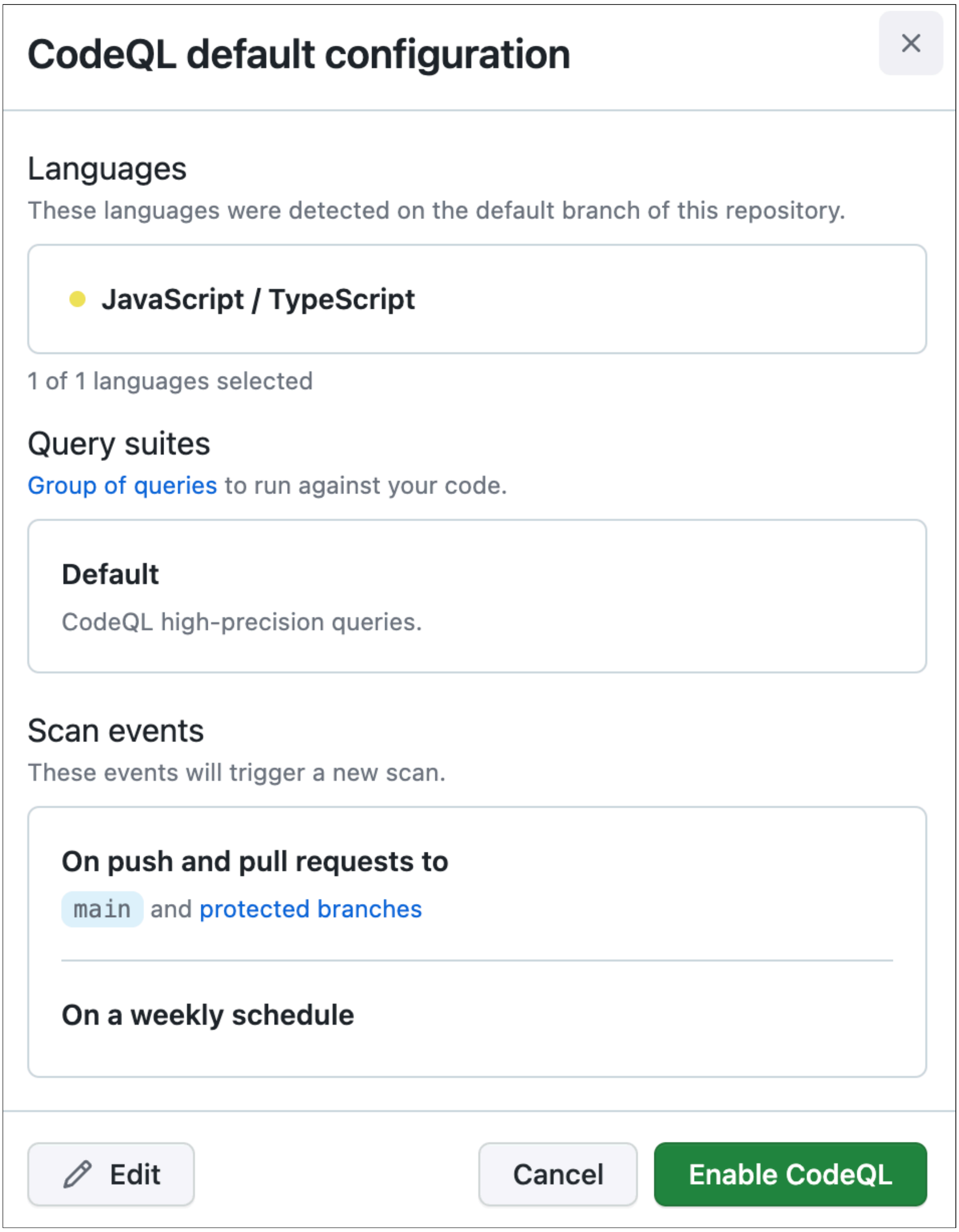
Protection rules

Pull request check failure

Define which code scanning alert severity should cause a pull request check to fail. This also applies to analysis results uploaded via the API.

Set up
Default
CodeQL will automatically find the best configuration for your repository.
Advanced
Customize your CodeQL configuration via a YAML file checked into the repository.
High or higher / Only errors

When you allow CodeQL to automatically find the configuration by choosing `Default`, it detects the language in the code and provides the analysis accordingly. CodeQL runs a set of default queries. In addition, if more queries are required to run, the `security-extended query suite` option can be used. It consists of all the queries in the default suite, along with additional queries with slightly lower precision and severity. You can access more such details through this link.



CodeQL default configuration

Languages

These languages were detected on the default branch of this repository.

● JavaScript / TypeScript

1 of 1 languages selected

Query suites

Group of queries to run against your code.

Default

CodeQL high-precision queries.

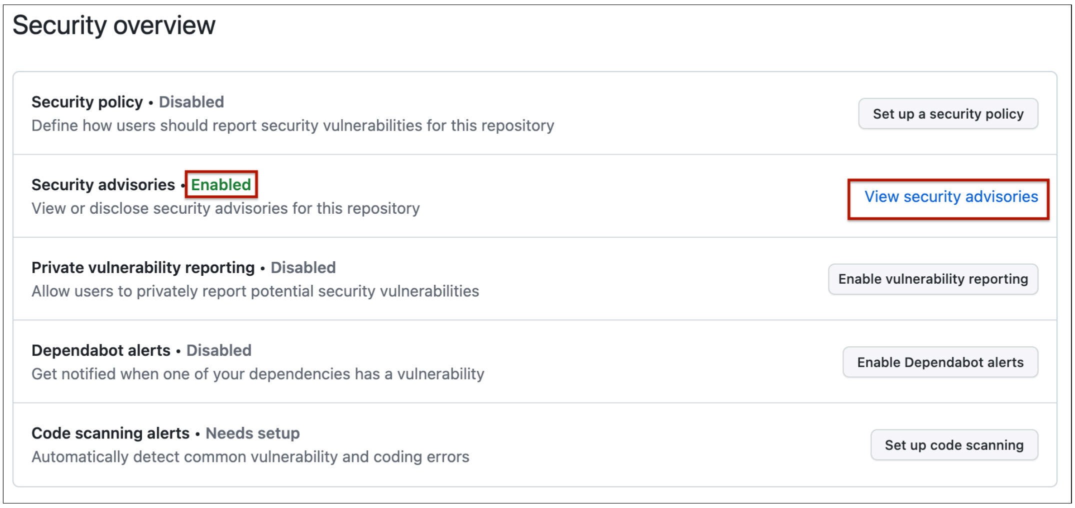
Scan events

These events will trigger a new scan.

On push and pull requests to `main` and protected branches

On a weekly schedule

Once CodeQL analysis is enabled, it runs whenever code is pushed or pulled into the repository. It also runs a routine check on a weekly basis. This aims to include new vulnerability checks in the default query suite from time to time. You will notice that security is enabled for the repository on the overview page. Now, you can view the analysis reports whenever required.



Security overview

Security policy • Disabled
Define how users should report security vulnerabilities for this repository

Security advisories • Enabled
View or disclose security advisories for this repository

Private vulnerability reporting • Disabled
Allow users to privately report potential security vulnerabilities

Dependabot alerts • Disabled
Get notified when one of your dependencies has a vulnerability

Code scanning alerts • Needs setup
Automatically detect common vulnerability and coding errors

Author(s)

Lavanya T S