

Be ambitious.

CodeQL – Azure User Group

March 2023 - How Open Source can protect you from yourself?



Static testing

Examine the source code and the flow for information, compiled artifacts, etc.
a.k.a. White/Gray box testing.

CodeQL
SonarQube
Snyk Code
Qodana



Dynamic testing

Carried out on software during code execution a.k.a. Black box testing.

Burp suite
CrowdStrike
OWASP ZAP
Intruder

Keep in mind when reviewing tools.
Common Weakness Enumeration (**CWE**) is the cause
Common Vulnerability Enumeration (**CVE**) is the effect



Where can you get CodeQL and how does it help?

- Open-Source('ish) tool for code analysis
- Available in GitHub Public Repos, also under license as “Advanced Security” for private/internal repositories
- Can be used for academic research and demos

Insight

Home > Microsoft Defender for Cloud



Microsoft Defender for Cloud | DevOps Security (Preview)

Showing subscription 'Visual Studio Professional Subscription' | PREVIEW

Search

- Overview
- Getting started
- Recommendations
- Security alerts
- Inventory
- Cloud Security Explorer (Preview)
- Workbooks
- Community
- Diagnose and solve problems

Cloud Security

- Security posture
- Regulatory compliance

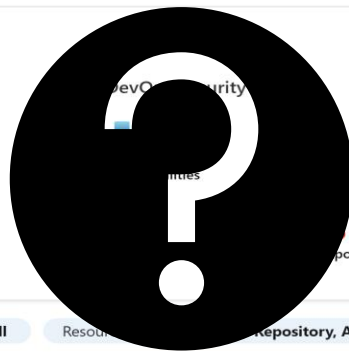
+ Add environment Refresh DevOps workbook Guides and Feedback → Getting Started Configure

Security Overview

DevOps security vulnerabilities



High
0
Medium
0
Low
0



DevOps coverage

0 GitHub Connectors 1 Azure DevOps Connectors

1 Total

GitHub repositories 0 Azure DevOps repositories 1

Search

Subscriptions == All

Resources

Repository, Azure DevOps Repository

☐ Name ↑↓

Pull request status

Total exposed secrets ↑↓

OSS vulnerabilities ↑↓

IaC scanning vulnerab... ↑↓ Total code scanning v... ↑↓

☐ [github-security-examples](#)

Off

N/A - Unspecified

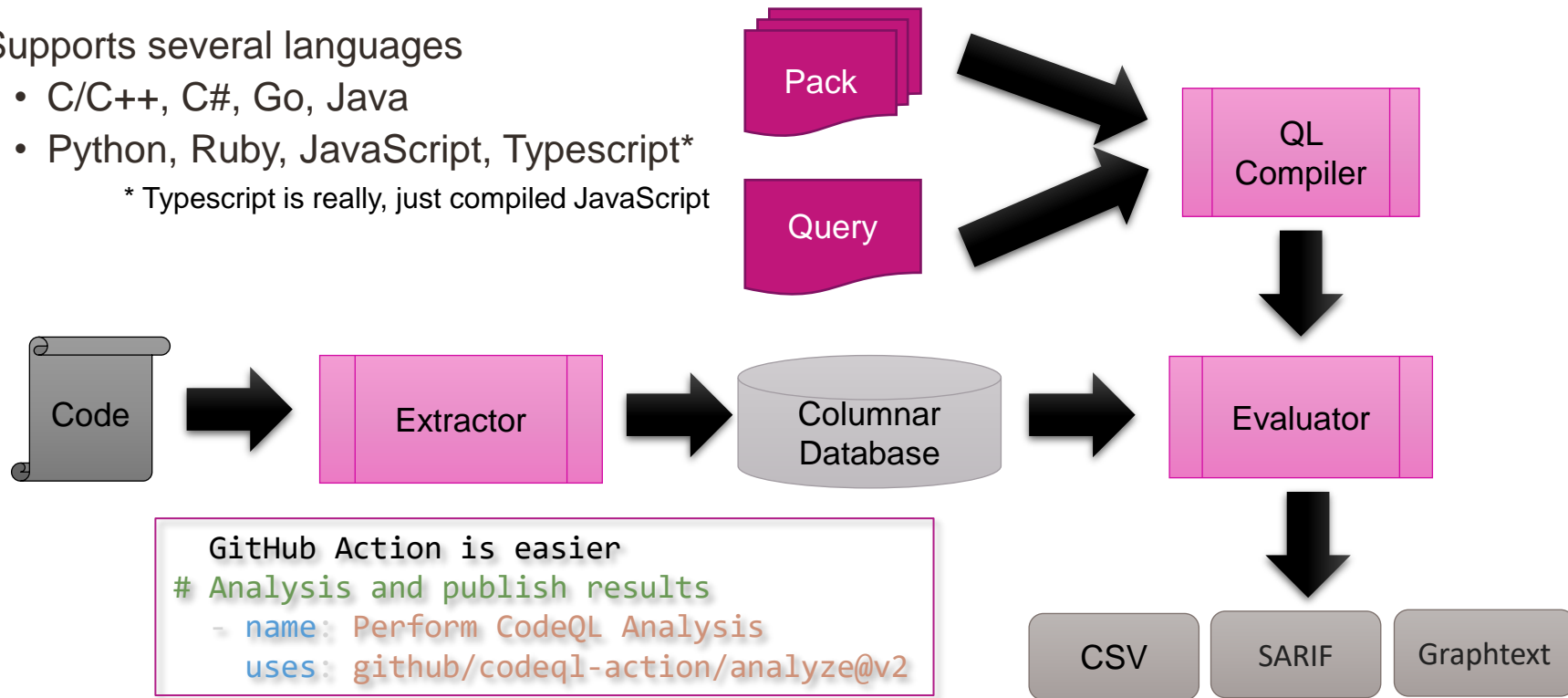
N/A

0

N/A

CodeQL Analysis Process

- Supports several languages
 - C/C++, C#, Go, Java
 - Python, Ruby, JavaScript, Typescript*
- * Typescript is really, just compiled JavaScript



But it does so much more..

- Demo
 - Setup
 - Local execution
 - Accessing existing packs
 - VS Code integration and custom queries
- [Bounty program](#) for new submissions
- Open-Source Community of developers
 - [@GHSecurityLab](#) on Twitter or [Slack](#)
 - A few [capture the flag](#) tests
 - [Events schedule](#)





Final words on providing submissions

- Submit your pull requests in the experimental folder, following the published guidelines.
- Create an issues noting what vulnerability groups you are targeting (CWE)
- Evaluation with take time based on
 - Performance
 - Impact
 - Number of false positives, over several large codebases
- Eventually your submission will be migrated into the full pack (maybe)

Questions?

All code available on GitHub :

<https://github.com/rgreene-public-repos/github-security>

“Coding makes finding bugs fun !?”