CodeQL概览

一、开源部分

开源部分为QL语言查询库，可以进行QL规则编写，进行漏洞挖掘等

github源码地址：https://github.com/github/codeql

官方使用教程：https://codeql.github.com/docs/

相关目录结构：

- 根目录：
  - 配置目录
  - 文档：感兴趣可以都看看
  - 各种语言的查询库：进入特定语言的目录查看，例如Python目录
    - 文档目录
    - 配置目录
    - tools：部分语言存在，辅助工具等
    - ql库目录：每种语言的主要查询库，集成大量内置规则
      - lib库：标准查询库，提供了编写ql语言的大量库，比如数据流、source点、sink点、API图等

        例如内置的Source和sink：

```
/**
 * Helper file that imports all framework modeling.
 */

// If you add modeling of a new framework/library, remember to add it to the docs in
// `docs/codeql/support/reusables/frameworks.rst`
private import semmle.python.frameworks.Aioch
private import semmle.python.frameworks.Aiohttp
private import semmle.python.frameworks.Aiomysql
private import semmle.python.frameworks.Aiopg
private import semmle.python.frameworks.Asyncpg
private import semmle.python.frameworks.ClickhouseDriver
private import semmle.python.frameworks.Cryptodome
private import semmle.python.frameworks.Cryptography
private import semmle.python.frameworks.Cx_Oracle
private import semmle.python.frameworks.data.ModelsAsData
private import semmle.python.frameworks.Dill
private import semmle.python.frameworks.Django
private import semmle.python.frameworks.Fabric
```

      - src：内置ql规则库，包含有限的安全类和非安全类规则

        例如CWE关联规则(ql/src/Security)：

名称	修改日期	类型	大小
CVE-2018-1281	2022/12/9 17:26	文件夹	
CWE-020	2022/12/9 17:26	文件夹	
CWE-020-ExternalAPIs	2022/12/9 17:26	文件夹	
CWE-022	2022/12/9 17:26	文件夹	
CWE-078	2022/12/9 17:26	文件夹	
CWE-079	2022/12/9 17:26	文件夹	
CWE-089	2022/12/9 17:26	文件夹	
CWE-090	2022/12/9 17:26	文件夹	
CWE-094	2022/12/9 17:26	文件夹	
CWE-116	2022/12/9 17:26	文件夹	
CWE-117	2022/12/9 17:26	文件夹	
CWE-209	2022/12/9 17:26	文件夹	
CWE-215	2022/12/9 17:26	文件夹	

        - tools：一些辅助工具
        - 其他目录：略

二、未开源部分

codeql-cli-binaries：CodeQL命令行执行工具，数据库创建、数据库查询等

github下载地址：https://github.com/github/codeql-cli-binaries

官方使用教程：https://codeql.github.com/docs/

相关目录结构：

- 执行入口：codeql（Linux）/codeql.exe（Windows）
    - 详细执行流程分析参考：https://paper.seebug.org/1921/
    - codeql整个调度流程是JAVA语言编写
- 具体语言目录：以Python举例
    - tools：提取器
        - 自动构建脚本
        - xxx.zip：提取器源码，可以解压调试分析，可以发现是生成trap文件的
    - 其他配置：略