FG Software Engineering
Leibniz Universität Hannover
Welfengarten 1
3. Stock, Flur G

Bachelor-Arbeit

- zu vergeben -

Security Code Exporter

Hintergrund

Um das Wissen über existierende Schwachstellen der Allgemeinheit bekannt zu machen, existieren Datenbanken denen dieses Wissen innewohnt. Eine dieser Datenbanken ist die National Vulnerability Database (NVD) oder alternativ die Common Vulnerabilities and Exposures (CVE) [1,2].
Sind diese Einträge Source Code spezifisch, so existieren teilweise Verlinkungen zu Github-Projekten, welche die sicherheitskritischen Stellen im Programcode identifizieren.
Um diese und weitere Security-relevante Source Code Elemente im Github zu finden, identifizieren und herunterzuladen zu können, muss die Konfiguration der Suchoptionen analysiert werden. Um eine weitere automatisierte Verarbeitung der Source Code Elemente zu ermöglichen, müssen diese Daten in einheitlicher Struktur, z.B. in einer Datenbank, abgelegt werden.Für eine weitere Datenverarbeitung ist oft Hashing oder weitere Normalisierung des Source Codes nötig. (Kommentare Entfernen, Variablen Namen ersetzen etc.)
[1] National Vulnerability Database: https://nvd.nist.gov/home.cfm
[2] Common Vulnerabilities and Exposures: https://cve.mitre.org

Aufgabe

Im Rahmen dieser Arbeit soll ein (Java)-Tool entwickelt werden, das die bestehenden CVE-Identifier der Datenbank (NVD oder CVE) herunterlädt. Hierfür existieren bereits Referenzprojekte die diese Funktionalität ebenfalls innehaben.
Die Identifier sollen dann für eine Github-Suche verwendet werden um den mit der CVE-ID in Verbindung stehenden Source Code zu identifizieren und herunterzuladen. Um weiteren Security-relevanten Source Code in Github zu finden, muss die Konfiguration der Suche zusätzlich noch analysiert und erweitert werden (weitere Schlüsselwörter, nur angegebene Programmiersprache berücksichtigen etc.)
Wurde die Suche auf Github durchgeführt und die Einträge herausgeladen, so soll es möglich sein die Einträge noch zu filtern und zu normalisieren. Weiterhin soll es möglich sein das Tool so zu konfigurieren, dass man Whitespaces, Kommentare entfernen kann und Methoden/Variablen Namen mit benutzerdefinierten Platzhaltern ersetzen lassen kann.
Das Ergebnis der Anwendung soll in einer Datenbank abgespeichert werden.

Weitere Anforderungen:
- Analyse der Suchkonfiguration
- Implementieren einer Java-Anwendung
- Es soll um weitere Such- und Filterkriterien erweitert werden können
- Gut strukturierter und kommentierter Programmcode
- Manuelle Adaptierung bei fehlerhaften Einträge

Organisatorisches

Betreuer: M. Sc. Fabien P. Viertel, fabien.viertel@inf .uni-hannover.de, Raum G307
Prüfer: Prof. Dr. Schneider **Beginn**: Ab sofort möglich