API Security Testing

Diplomand

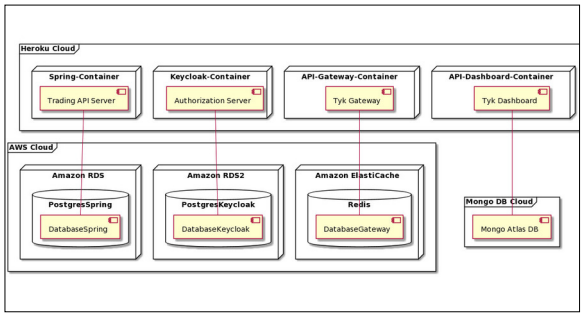
Maximilian Lukas Marxer

Aufgabenstellung: Das Ziel dieser Arbeit war es, ein Konzept zu entwerfen mit dem man APIs auf ihre Sicherheit testen kann. Dazu wurde eine Beispielapplikation (Spring Boot „Mockup“ API) entwickelt, die den Kauf und Verkauf von Wertpapieren simuliert und eine Zugriffskontrolle implementiert. An der API soll demonstriert werden, wie Schwachstellen durch Sicherheitstests entdeckt werden können. Um die Funktionalität und Sicherheit der API zu überprüfen, sollen neben den üblichen Unit- und Integration-Tests auch End-to-End Tests an der API durchgeführt werden. Für diese Tests soll ein geeignetes Tool in der Arbeit bestimmt werden. Die Tests sollen automatisiert in der CI/CD Pipeline ausgeführt werden.

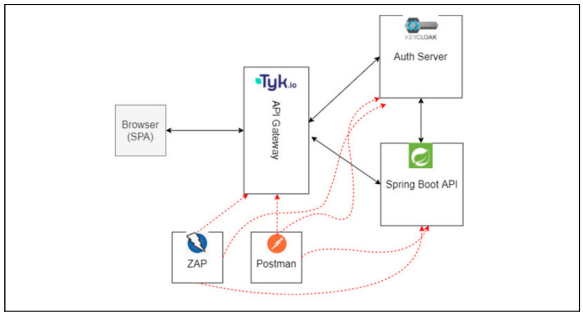
Vorgehen: Zu Beginn der Arbeit wurde eine Tool-Evaluation über aktuelle Tools zum Testen einer API durchgeführt. Durch die Evaluation wurde ein Tool bestimmt werden, mit welchem die End-to-End Tests an der API durchgeführt werden können. Dabei wurden drei verschiedene Tools genauer untersucht und auf ihre Funktionen analysiert. Anhand der gesammelten Erkenntnisse wurde mit Praxispartner zusammen entschieden, dass die Tests mit dem Postman Tool durchgeführt werden. Nach dieser Entscheidung wurde ein Lösungskonzept entwickelt, um Sicherheitsanforderungen der Beispielapplikation und die dazu nötige Architektur zu definieren. Für die definierte Architektur wurde eine Bedrohungsmodellierung durchgeführt, um die Schwachstellen der Beispielapplikation zu identifizieren und zu priorisieren. Für die Analyse der Schwachstellen wurden die OWASP Top 10 Schwachstellen für APIs verwendet. Nach der Analyse konnten die priorisierten Schwachstellen durch Security API Tests überprüft werden.

Ergebnis: Für das Testen der Security wurde eine funktionsfähige API entwickelt. Die Authentifizierung & Autorisierung wurde mit der Keycloak Anwendung umgesetzt und verwendet ein sogenanntes standarisiertes JSON Web Access Token, kurz JWT. Es wurde ein API Gateway aufgesetzt und so konfiguriert, dass alle Anfragen von Benutzern darüber geleitet wird. Er beschränkt die Anzahl der Aufrufe und zeichnet sie auf. Für die Authentifizierung bildet er aus der Signatur des JWT-Tokens einen Opaque-Token. Die Header und Payload Informationen des JWTs bleiben dadurch dem Benutzer verborgen und sind nur dem API Gateway bekannt. Die API wurde mit Unit- und Integration-Tests und End-to-End Tests auf ihre korrekte Funktionalität und Sicherheit getestet. Die Tests wurden automatisiert in die CI/CD Pipeline integriert. Ebenfalls wurde eine statische Code Analyse und ein automatisierter Penetration Test durchgeführt. Die Ergebnisse der Tests haben viele Schwachstellen der Beispielapplikation aufgezeigt. Für einige dieser Schwachstellen wurden Massnahmen getroffen, um diese zu mindern. Die Arbeit zeigt exemplarisch auf, wie eine API auf ihre Sicherheit getestet werden kann.

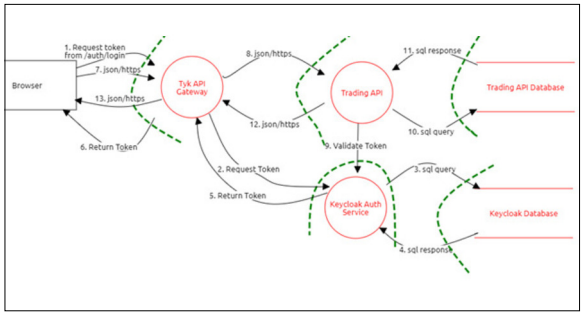
Deployment Diagram
Eigene Darstellung



Architektur Diagram
Eigene Darstellung



Bedrohungsmodell
Eigene Darstellung



Examinatorin
Prof. Dr. Nathalie Weiler

Experte
Cyrill Rüttimann, ipt, Ebertswil, ZH

Themengebiet
Sicherheit, Software