SENSITIVE INFORMATION SCAN

DevSecOps Basics

What is Sensitive Information Scan?

The Sensitive Information Scan (SAS) phase scans the code for sensitive information (e.g. hardcoded password, tokens, secret keys, etc) before pushing the code into code repositories. This makes sure that even if the code falls into wrong hands tomorrow, the sensitive information won't get exposed.

The following components are there in this phase:

- SAS tools i.e. Trufflehog, GitSecrets, Talisman

People involved: Developers

External sources

- Using Trufflehog and GitSecrets https://sweetcode.io/how-use-truffle-hog-git-secrets/

Why is it important in DevSecOps?

The Sensitive Information Scan makes sure that sensitive information is not entering the DevSecops pipeline and code repository, reducing the attack surface. And, as it is performed by tools, it can be automated to run every time the user tries to push the code into the version control system.

What will you learn in this section?

The user will learn to perform the following tasks

- Finding security issues in code using TruffleHog
- Locating security issues in code using GitSecrets
- Using Talisman to find security bugs in the code

Tools Covered

- TruffleHog
- GitSecrets
- Talisman

Labs

- TruffleHog: Locating Sensitive Information
  - A Kali machine is provided to the user with GitSecrets installed on it. The source code for a sample web application is provided in the home directory of the root user.
    Objective: Scan the web application source code with TruffleHog and find sensitive information in the code!
- GitSecrets: Finding Hardcoded Credentials
  - A Kali machine is provided to the user with GitSecrets installed on it. The source code for a sample web application is provided in the home directory of the root user.
    Objective: Scan the source code with the GitSecrets tool and find sensitive information!
- Talisman: Pre-Commit Code Scanning
  - A Kali machine is provided to the user with Talisman installed on it. The source code for three sample web applications is provided in the home directory of the root user.
    Objective: Commit the Source code in the local repository and analyze the talisman report