SEI Cyber Minute Script & visuals

Title: SCAIFE: An Alert Auditing Classification Prototype

Ebonie McNeil
VIDEO/Podcasts/vlogs This video and all related information and materials ("materials") are owned by Carnegie Mellon University. These materials are provided on an "as-is" "as available" basis without any warranties and solely for your personal viewing and use. You agree that Carnegie Mellon is not liable with respect to any materials received by you as a result of viewing the video, or using referenced web sites, and/or for any consequence or the use by you of such materials. By viewing, downloading and/or using this video and related materials, you agree that you have read and agree to our terms of use (http://www.sei.cmu.edu/legal/index.cfm).
DM19-0681
Hello, this is Ebonie McNeil from CERT with your SEI Cyber Minute. Secure software is a top priority for many organizations. Static analysis tools output alerts that identify potential flaws in code. Last year, we publically published SCALe, an aggregator tool that displays various tool alerts and gives auditors an intuitive and user-friendly interface. [Slide 4] Auditing manual alert determinations can often be time consuming. The Source Code Analysis Integrated Framework Environment (commonly known as SCAIFE) is a multi-server software architecture and open source prototype system developed to integrate with SCALe and other similar tools to enable automatic alert classification and advanced prioritization. [Slide 5] The SCAIFE prototype is intended to be used by engineers and analysts who manually audit alerts. SCAIFE provides automatic alert classification using Machine Learning which gives a level of confidence that the alert is true or false. [Slide 6] The SCAIFE prototype also enables organizations to apply formulas that prioritize static analysis alerts by using factors they care about. [Slide 7] SCAIFE relies on REST-API principles to provide an interface for developers to quickly integrate and communicate with the server system. We have published a YAML-formatted file specifying the SCAIFE API, available at the CMU-SEI GitHub site for free downloads by the public. [Slide 8] The YAML specification provides the SCAIFE API definition beta version in a format that developers can easily use to view, modify, and automatically generate code from. Thanks for watching this SEI Cyber Minute. For more information, please visit our website or send me an email at info@sei.cmu.edu.
SCALe Interface

[Image of a software interface showing a list of vulnerabilities and a code section labeled 'SRC']
SCAIFE Architecture

UI Module
- Stores local projects
- Displays project and alert data

User Interface

Registration Module
- Generates registration tokens
- Provides authentication and basic authorization for other servers

Prioritization Module
- Stores prioritization formulas and user-uploaded prioritization fields

Statistics Module
- Creates, runs, and stores classifiers
- Stores adaptive heuristic algorithms
- Stores automated hyperparameter optimization algorithms

DataHub Module
- Stores tool and alert information
- Stores test suite meta-data and alert determinations
- Generates speculative mappings

API Calls

API Calls
Automatic Alert Classification

Select ‘Classify’ button to run the classifier on a project
- Classifier predicts alert determinations
- Meta-alerts will be classified
- Currently, example metrics are loaded for the 'Confidence' field
  - Usability demonstration only
  - Values not currently from classifier
Alert Prioritization

Prioritization schemes with mathematical formulas user can create and/or use

Project: project2

Create New Scheme

Formula for CERT RULES:

\[
\text{IF CWEIS(\text{cert\_likelihood})+IF CERT RULES(\text{cert\_severity}/2+\text{cert\_remediation})/\text{confidence}}^2
\
\]

Prioritization Formula:

\[
\text{IF CWEIS(\text{cert\_likelihood})+IF CERT RULES(\text{cert\_severity}/2+\text{cert\_remediation})/\text{confidence}}^2
\
\]

Surface

Alert Prioritization

Prioritization schemes with mathematical formulas user can create and/or use

Project: project2

Create New Scheme

Formula for CERT RULES:

\[
\text{IF CWEIS(\text{cert\_likelihood})+IF CERT RULES(\text{cert\_severity}/2+\text{cert\_remediation})/\text{confidence}}^2
\
\]

Prioritization Formula:

\[
\text{IF CWEIS(\text{cert\_likelihood})+IF CERT RULES(\text{cert\_severity}/2+\text{cert\_remediation})/\text{confidence}}^2
\
\]
SCAIFE API on GitHub

The YAML file specifies the Source Code Analysis Integrated Framework Environment (SCAIFE) API definition beta version 0.0.2 [1,2,3]. In a format that developers can easily use to view, modify, and automatically generate code from (e.g., with the Swagger-Editor and Swagger-Codegen tools [4]). The YAML file was almost entirely manually created by SEI developers. The only things that were auto-generated by Swagger tools within the YAML file are some of the examples.

SCAIFE is an architecture that supports static analysis alert classification and prioritization. It is designed so a wide variety of static analysis tools can integrate with the system using the API definition we are developing. We expect the API to be of interest to organizations that develop and/or research static analysis tools. Static analysis alert auditing aggregators, and other static analysis alert auditing frameworks. This SCAIFE beta API definition can be referenced by developers to help them estimate development effort that would be required to modify their organization’s tools to make and respond to SCAIFE API calls. Also, this beta API definition is being published with a goal of generating feedback from developers and organizations interested in implementing the SCAIFE API to help improve SCAIFE API v1.0 to become more easily usable by developers for a wide variety of static analysis tools. Compared to the beta API definitions, the published SCAIFE API v1.0 definition will include implementation details, the architecture description, motivations, and a prototype system.