

Software Security by Design (SSD) Platform

Miltiadis Siavvas, CERTH

Kalouptsoglou Ilias, CERTH

11/10/2022

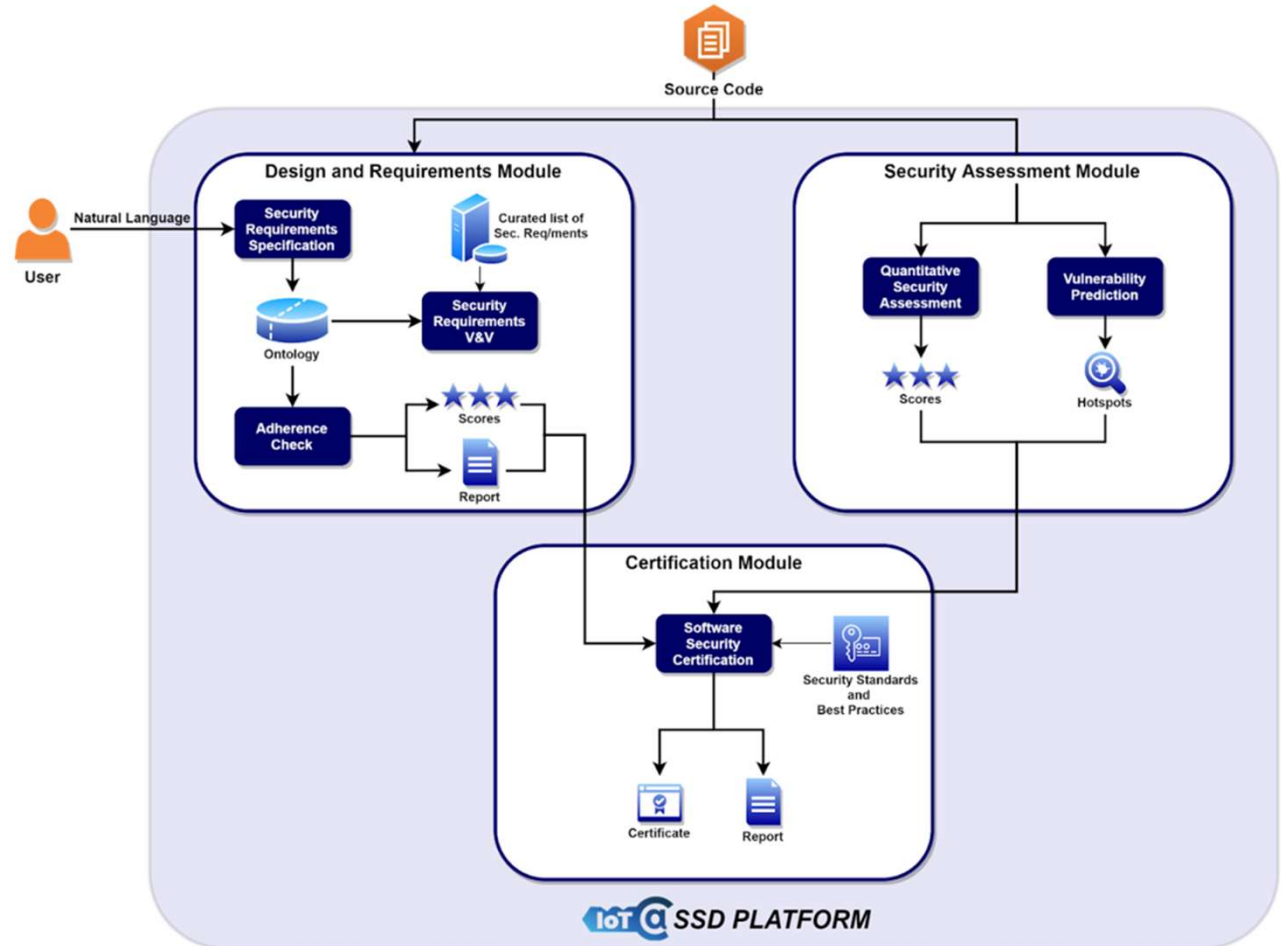




IoTAC: Security by Design IoT Development and Certificate Framework with Front-end Access Control (Horizon 2020)

Three core parts of the SSD:

1. Design & Requirements Module
2. Security Assessment Module
3. Validation/Certification Module
(in a future version)



The screenshot shows the SSD Platform web interface. At the top, there is a navigation bar with 'SSD Platform', 'Home', 'Settings', and 'Documentation'. On the left, there is a sidebar with 'IoT@' logo and menu items: 'Projects', 'Requirements', and 'Security'. The main content area has two status boxes: 'Security Analysis: pending' and 'Vulnerability Analysis: pending', along with a 'RUN CENTRAL ANALYSIS' button. Below this is a 'Projects' section with a '+ NEW PROJECT' button and three project cards: 'librdkafka' (created 9/15/2022, 2:38:33 PM), 'OkHttp' (created 9/15/2022, 2:39:12 PM), and 'WeatherAPISmartHome' (created 10/5/2022, 5:24:59 PM). Each card has 'EDIT' and 'DELETE' buttons. The footer contains social media icons and 'IoTAC Website' link, and a copyright notice: '© 2022 Copyright: MDBootstrap.com'.

Users can select:

- Add new project
- Go to the Requirements page
- Run central analysis
- View the security results

Project Name

Git URL

Git Username

Git Password

Description

Extra info by toolbox

Requirements ▾

Extra info common (for all toolboxes)

```
{"language": "java"}
```

Last modified: 10/5/2022, 5:24:59 PM

CANCEL

SAVE CHANGES

User input:

- Project name
- Project Git url
- Project language
- Basic Authentication credentials (if private repository)

Requirements Module

Specification: Insert, submit requirements and view results

The system could authenticate users. | Authentication
 The system must ensure the storage of cryptographic keys. | Confidentiality
 The system must ensure access to log files. | Non-repudiation
 The system must ensure a least privilege policy for access privilege management. | Authorization
 The system must ensure securely booting operating systems. | Integrity

Check Result: **Successfully validated: 5 requirement(s).**

CHECK **SUBMIT**

Requirements Specification Results

The results produced by employing Syntactic and Semantic analysis.

Show requirements **5**

ID	Requirement	Characteristic
ID: 1	The system could authenticate users .	Authentication

Priority could
Actor system **Action** authenticate **Object** user

V&V: Recommendations for edit or complement

ID: 1 The system could authenticate users . Authentication

Proposed Priority must

Replace Recommendations

↔ The system must authenticate users prior to accessing an application or data . **Authentication**

↔ The system must authenticate users using at least one of the following authentication mechanisms : username , password , digital certificate , secure token or biometrics . **Authentication**

Complement Recommendations

+ The system must authenticate users prior to accessing an application or data . **Authentication**

+ The system must authenticate users using at least one of the following authentication mechanisms : username , password , digital certificate , secure token or biometrics . **Authentication**

ID: 2 The system must ensure the storage of cryptographic keys . Confidentiality

Adherence Check: Questionnaire for experts to check the adherence of the requirements

ID: 1 The system could authenticate users . **Authentication**

Is there any evidence of the existence of appropriate security controls that address this requirement?

Very Low Low Medium High Very High *How certain are you?* **Default**

Have existing security controls been properly implemented?

Very Low Low Medium High Very High *How certain are you?* **Uncertain**

What is the strength level of the existing security controls?

Very Low Low Medium High Very High *How certain are you?* **Certain**

At what level do the controls function and operate as intended?

Very Low Low Medium High Very High *How certain are you?* **Certain**



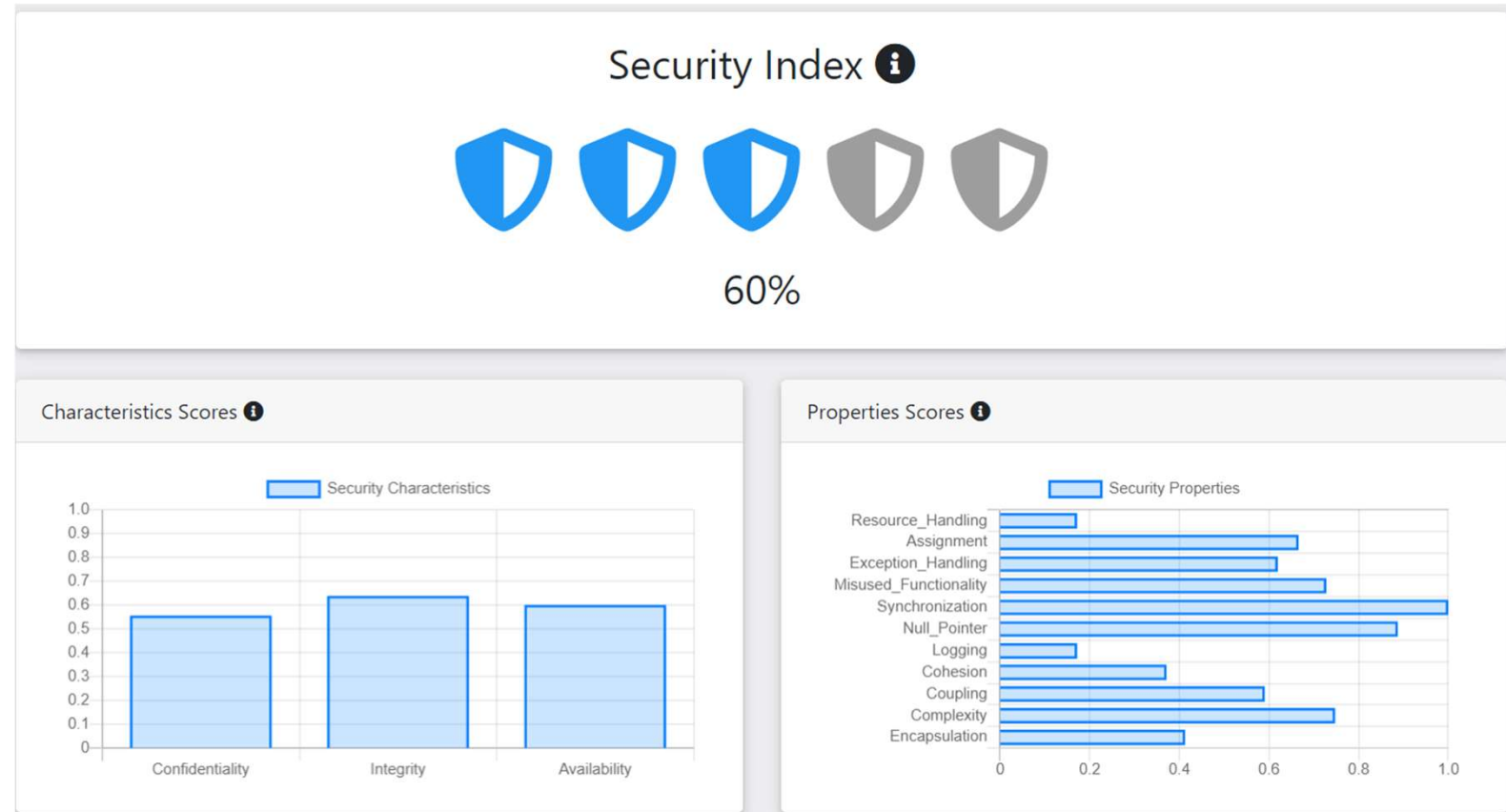
The Fuzzy averaged membership function of the Requirements Adherence Score



Quantitative Security Assessment

A Static Analysis-based tool

- Overall security level of the analyzed project
- Security level corresponding to CIA characteristics
- Properties Scores

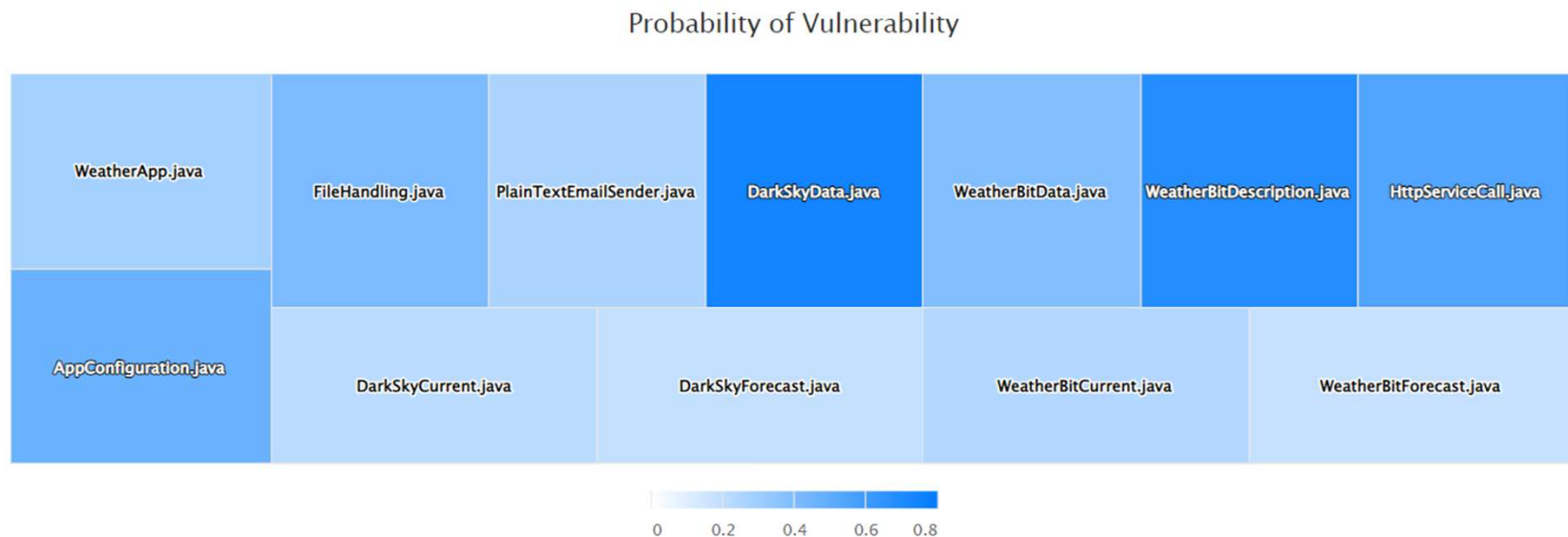


Vulnerability Prediction

Software Vulnerability Prediction tool based on Deep Learning (DL) and Natural Language Processing (NLP)

- Input: Sequences of source code tokens
- Techniques:
 - Word Embedding Practices (i.e., Word2Vec representations)
 - DL model
- Output: Highlights software components that are likely to be vulnerable

Vulnerability Prediction Heatmap ⓘ



THANK YOU!

