Static Analysis Report

# Revision

Version 3

9/8/23 12:22 PM

# SME

Process: Charles Wilson

Report: Marwan Abi-Antoun

# Abstract

This document describes the process used to create a static analysis report.

# Group / Owner

DevOps / Information Systems Security Developer

# Motivation

This document is motivated by the need to have early security-related implementation feedback in the development of software for use within safety-critical, cyber-physical systems for certification of compliance to standards such as **ISO/SAE 21434** and **ISO 26262**.

# License

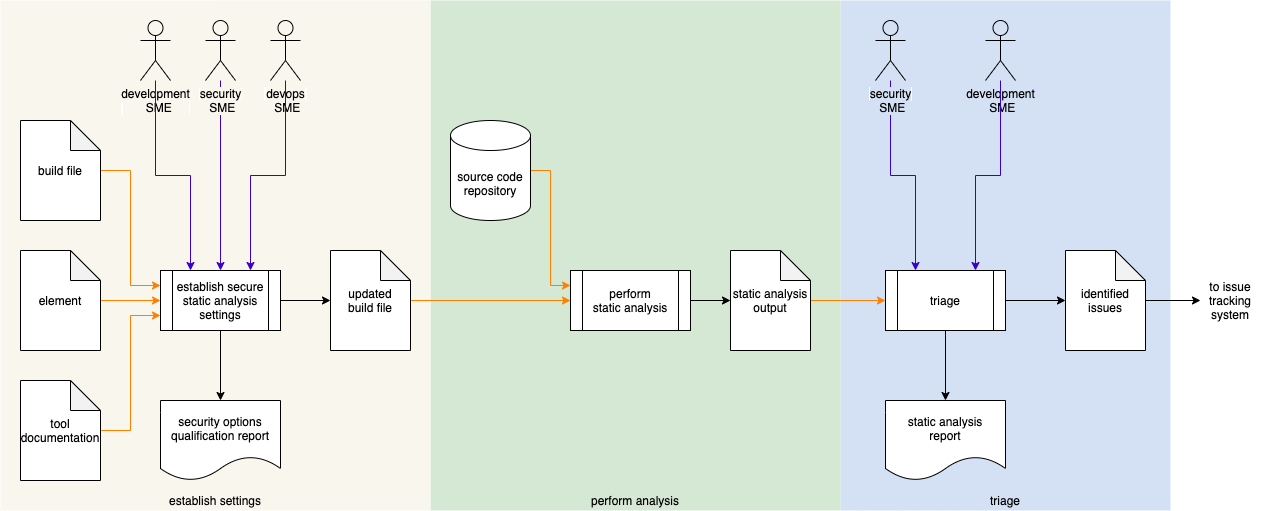
This work was created by **Motional** and is licensed under the **Creative Commons Attribution-Share Alike (CC4-SA)** License.

[**https://creativecommons.org/licenses/by/4.0/legalcode**](https://creativecommons.org/licenses/by/4.0/legalcode)

# Overview

After compiler feedback, static analysis provides the fastest feedback available to developers as to possible security issues within their code.

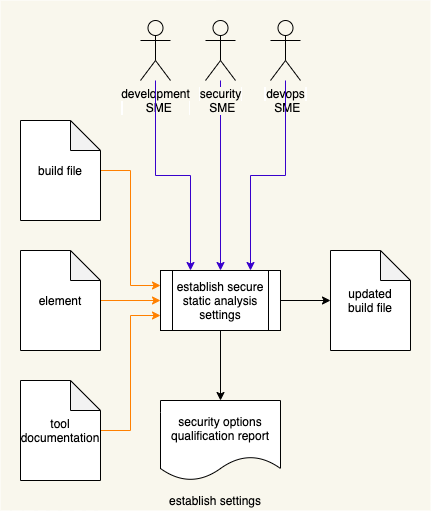
The following diagram illustrates the process to be used:



# Process

## Establish Settings

|  |  |
| --- | --- |
| **Inputs** | Build file  Element  Tool documentation |
| **Outputs** | Updated build file  Security options qualification report |
| **Participants** | Development SME  Security SME  DevOps SME |

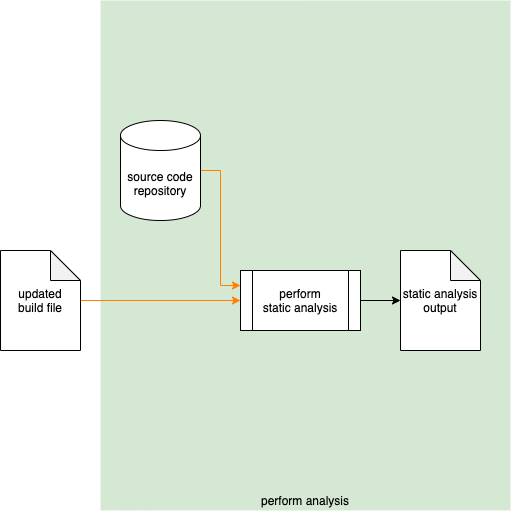


Using the **Element** under consideration, static analysis **Tool Documentation**, and **Build File**; the Development SME, Security SME, and Devops SME will work together following the **Secure Settings Document** **[2]** process to determine what element aspect needs to be tested and what static analysis tool settings are needed to enable that testing. An **Updated Build File** will be produced. A **Security Options Qualifications Report** is generated.

**Note:** The scope of the element may be as small as a single file or as large as the entire project.

## Perform Analysis

|  |  |
| --- | --- |
| **Inputs** | Updated build file  Source code repository |
| **Outputs** | Static analysis output |
| **Participants** | **none** |

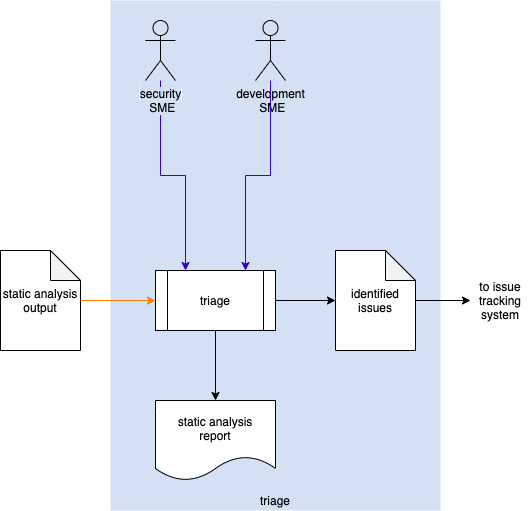


Using the settings from the **Updated Build File**, the **Static Analysis Tool** takes the **Source Code Repository** and performs an analysis on the element. The generated output is captured into **Static Analysis Output**.

**Note:** The scope of the analysis may be as small as a single file or as large as the entire repository.

## Triage

|  |  |
| --- | --- |
| **Inputs** | Static analysis output |
| **Outputs** | Identified issues  Static analysis report |
| **Participants** | Security SME  Development SME |



The Security SME and the Development SME review the **Static Analysis Output** to identify any issues needing further investigation. For any such, an issue will be created in the issue tracking system. A **Static Analysis Report** will be generated.

### Identified Issues

The recommended form of the **Identified Issues** artifact is a Static Analysis Results Interchange Format (**SARIF**) encoded JSON. This document assumes SARIF version 2.1.0 **[1]** or later.

### Static Analysis Report

The **Static Analysis Report** is recommended to be produced from the **Identified Issues** artifact and should detail the issues exposed by the static analysis.

The report contains one or more analysis runs. Each run includes:

* Description of the tool used
* Description of the analyzed element
* Tool invocation settings
* Results of the analysis

The tool description includes:

* Name
* Version
* URI to tool documentation
* Tool rules

The tool rules (one or more) convey the classes of analysis performed. Each includes:

* ID (unique)
* Name
* Short description of the rule
* Full description of the rule
* URI to rule documentation
* Reference (optional) to associated taxonomy entry (Common Weakness Enumeration, …)

The analyzed element description (one or more) provides information related to the element under consideration. Each includes:

* URI to analyzed element
* URI to repository the element came from

The analysis results (one or more) describe the issues exposed by the analysis. Each includes:

* Human-readable description
* Location within the element of the issue
* Severity of the issue
* Reference (optional) to the associated taxonomy entry

# References

1. **Static Analysis Results Interchange Format (SARIF) Version 2.1.0**  
   <https://docs.oasis-open.org/sarif/sarif/v2.1.0/os/sarif-v2.1.0-os.pdf>
2. **Secure Settings Document** (AVCDL secondary document)
3. **Fuzz Testing Report** (AVCDL secondary document)
4. **Dynamic Analysis Report** (AVCDL secondary document)
5. **Security Options Qualification Report** (AVCDL tertiary document)