

STATIC CODE ANALYSIS METHODOLOGY FOR SECURITY

WHITEPAPER

July 2025

© 2025 Software AG. All rights reserved.

softwareAG
A SOFTWARE GMBH BRAND

1 Introduction

Application security is concerned with the privacy and security of the products and web applications of an organization. It is of immense importance to a company to protect itself and its customers from the constant threat of being hacked.

This paper elaborates how Software AG leverages Security static code analysis, the tools used, and the processes followed to improve product security.

2 Security Static Code Analysis

Static analysis involves verifying the application code without executing it. This type of security assessment is complementary to dynamic testing as it has distinct advantages wherein, we identify security weaknesses by scanning the source code directly. This is a highly effective way of identifying the Security flaws in any application. It can be scaled well and could be leveraged effectively for automatically identifying security flaws such as buffer overflows, SQL injection, Code injections and most of the OWASP top rated vulnerabilities.

At Software AG, we use the industry recognized static analysis tool to review the source code and find security flaws. Software AG has chosen an industry standard static source code analysis tool that best fits the organization's requirements and features that make it stand out from other enterprise class products.

To check the capabilities of the tool to analyze the code, please refer to https://docs.checkmarx.com/en/34965-318312-severity-levels.html.

The tool would usually produce results with a high number of false positives and some false negatives. Please refer to section 3.0 to understand how Software AG deals with such limitations.

3 The Workflow

Every product must undergo a pre-setup in the tool by providing its source code location and access details. The tool is also configured with a pre-defined set of product-specific rules to perform the security static code scan.

© 2025 Software GmbH. All rights reserved.

Process steps in detail:

- For the automatic scans to be able to run, every product's current version and optional fixed versions undergo integration set-up either in the build environment or static code scanner. This is performed by the Product team and Security team.
- The build system triggers the static code scanner to perform the scan. In the case of a scheduled scan, the scan is triggered by the static code scanner at the pre-configured time.
- The static code scanner executes the rules on the code.
- The static code scanner loads the latest scan results and processes them.
- The product team is notified of the scan results, and they analyze the reported results from the static code scanner.
- The reported issues are tracked and fixed.
- Results with a high false positive rate are mitigated, marked as false positives, adapt rules and a new scan is triggered.
- Issues with high confidence and low false-positive rate are considered as bugs and tracked in the defects tracking system.
- The scan is configured to perform automatically during the build process.
- The projects are also configured to perform a periodic full scan in between incremental scans for better performance and accuracy.

4 Integrating SAST into the pipelines

Static application security testing (SAST) is the process of examining source code for security defects. SAST is one of the many checks in an application security assurance program designed to identify and mitigate security vulnerabilities early in the development lifecycle.

To build a sustainable and repeated assessment, integrating SAST tools into the development DevOps pipeline is crucial that automates the whole task for efficiency, consistency, and early detection.

The source code of Adabas Natural product suite is configured to be scanned with predefined rulesets and at regular intervals. The integrated tool sets help to support heterogeneous development systems to integrate SAST scans at strategic points during development like the pull requests, pre-commit, build process, etc.

Developers would review the SAST results and take appropriate actions. Once an identified issue is marked as a false positive, the scanner makes sure the same issue does not appear as an issue in the next scan. The scanner has the intelligence to manage and report New, Recurrent and Fixed vulnerabilities based on the latest scan results.

© 2025 Software GmbH. All rights reserved.

In the end, product teams review the SAST reports and complete the Release task on Quality Category.

5 Feedback

If you have feedback or questions on this whitepaper, please send an email to devsecurity@softwareag.com

© 2025 Software GmbH. All rights reserved.