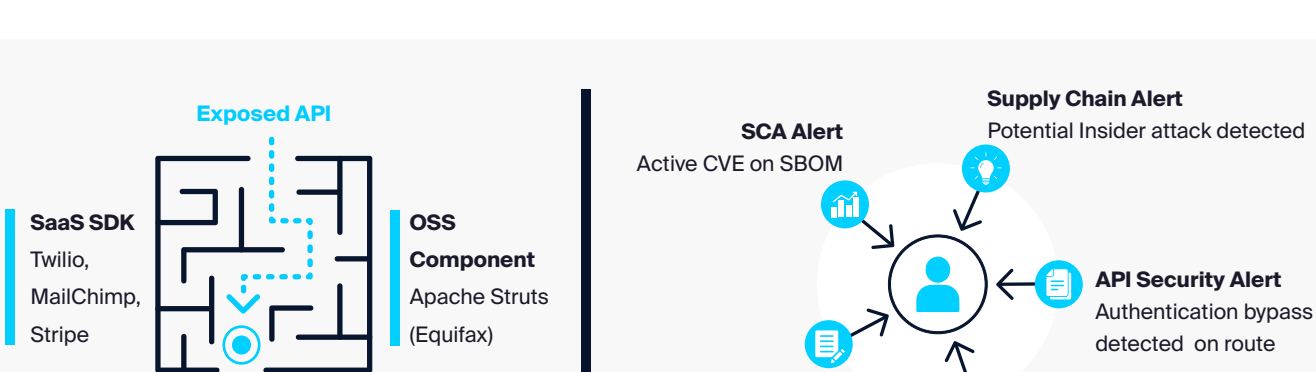
ShiftLeft

# Seeing Attackability in Modern Code

## The Challenge for AppSec Today

The complexity of modern code has outpaced most static analysis tools. Applications today are made up of custom code and open source libraries that behave as though they were a single entity. Instead of recognizing this complexity, vendors have been maintaining siloed tools that analyze custom and open source code separately, in bits and pieces. Because of this, they cannot answer the question, “can an attacker find and exploit this vulnerability given my application’s architecture.”




**Attackers access an application as though it is a single entity, while security tools analyze analyze modern code piecemeal and drown AppSec teams in siloed sets of vulnerabilities.**

## ShiftLeft CORE

ShiftLeft CORE understands attackability in modern code and can quickly deliver findings to agile teams so they can quickly prove and fix security risk in their code. A single scan from ShiftLeft CORE combines static analysis for bugs in custom code, composition analysis for known vulnerabilities in open source code, and secrets detection for leaked credentials.

ShiftLeft CORE is a SaaS platform that creates an intermediate representation of your code and sends that artifact, not your source code, to our servers for analysis. Scans are fast enough to be easily inserted into your CI/CD pipeline for daily or weekly scans or, if your team is ready, to run at every pull request. So feel free to scan early and scan often.

# Modern Source Code Analysis for Earlier Fixes

Developers never understand their code better than while they are writing it. The key to faster fixes is to provide them with accurate security results for the code they are currently working on. ShiftLeft CORE processes code quickly, and can analyze 1M lines-of-code (LOC) in less than 10 minutes; 100K LOC in less than 1min. It is also the most accurate static analysis solution with an OWASP Benchmark score of 75%.

## Bugs in Custom Code

Next-gen static application security testing (NG-SAST) uses graph analysis to quickly produce findings with complete dataflows that allow engineers to prove the attackability of threats. Because scans are quick and can be automated in the CI/CD for frequent testing, teams using ShiftLeft CORE are able to fix 91% of new vulnerabilities within two sprints of discovery[1].

## Security Insights

Security Insights are potential security issues in the code that may not be vulnerabilities today but are bad practices based on industry best-practice. These are conditions that can lead to OWASP Top 10 or CWE Top 25 vulnerabilities.

## Secret Detection

ShiftLeft detects Secrets, or hard-coded values (e.g., client Secrets, username/password combinations) and sensitive information (e.g., phone numbers and addresses). Unlike “grepping” for these patterns that lead to false positives, the use of Code Property Graph identifies when secrets are being leaked without proper transformation or obfuscation.

## CVEs in Open Source

ShiftLeft Intelligent Software Composition Analysis (I-SCA) is the only composition analysis tool to show the attackability of vulnerable packages. Teams triaging open source risk based on the context of their application architecture have reduced critical security tickets by more than 92%.

## ShiftLeft Educate

ShiftLeft Educate provides developers with on demand education in the context of the bug they are looking at and the language they are working in. With a premium upgrade, AppSec Managers can manage training and certification program for developers.

## Coverage

ShiftLeft CORE supports attackability across custom and open source code for Java, C-Sharp, Scala, JavaScript and TypeScript with other languages on the way. NG-SAST also supports Python, Go, and Terraform and covers the OWASP Top 10 and CWE Top 25 along with a range of language specific vulnerability categories.

[1] AppSec Shift Left Progress Report