cycode

Static Application Security Testing Buyer's Guide:

How to Choose the Right Tool for Your Team

Static Application Security Testing Buyer's Guide: How to Choose the Right Tool for Your Team

Tired of chasing vulnerabilities through your codebase? Static Application Security Testing (SAST) can help you catch them before they become a problem. By analyzing your code statically, SAST tools can detect potential security flaws early in the development lifecycle.

But let's be honest: legacy SAST solutions often feel like a drag. Slow scan times, overwhelming false positives, and a frustrating developer experience can hinder your team's productivity and increase your risk exposure.

It's time for a SAST upgrade.

In this guide, we'll explore why modern SAST tools have become indispensable in today's DevOps environments, the must-have features to look for in a tool, and steps to help you choose the right solution for your team.

By the end, you'll understand why companies like PayPal and UBS trust Cycode for their application security, and why a modern, developer-friendly SAST solution is key to securing your code without slowing down your development process.

Looking for a more general primer on SAST? Take a deep dive into what SAST is, how it works, and why it's important.

Refresh: What Is Static Application Security Testing (SAST)?

SAST is a security testing technique that analyzes source code, bytecode, or binaries for vulnerabilities without executing the application. It acts like a constant code reviewer, and is commonly used during the coding or build phases of the software development lifecycle (SDLC) to detect security issues early, when they are easier and cheaper to fix.

By breaking down the source code into tokens and running pattern-matching algorithms, SAST identifies security flaws like SQL injection, cross-site scripting (XSS), and buffer overflows. It checks against coding standards and known vulnerability patterns to flag potential weaknesses. This makes it a key player in the shift-left movement, where security is integrated earlier into development, rather than waiting until after deployment to test.

Importantly, SAST tools are vital for organizations aiming to adopt DevSecOps practices, guiding developers through secure coding while maintaining development velocity.

SAST vs Other Application Security Testing Tools

While SAST is critical for scanning proprietary code, it's one of several application security testing (AST) tools that should be used in a comprehensive security strategy.

There's also Software Composition Analysis (SCA), which scans third-party libraries and open-source dependencies. Together, these two tools provide robust coverage for both internally developed and external components. Learn more about how SAST and SCA work together.

While other tools like Dynamic Application Security Testing (DAST) can complement SAST and SCA by identifying runtime vulnerabilities, **SAST and SCA remain the foundation of any effective AST strategy.** Why? Because they identify vulnerabilities early in the development lifecycle and offer more proactive protection.

Importantly though, AST tools like SAST and SCA should be a part of a broader Application Security Posture Management (ASPM) strategy to deliver continuous monitoring, risk prioritization, and holistic security coverage across the entire SDLC. That's why robust integration capabilities are one of the key features to look for in a modern SAST tool.

Keep reading to discover 6 more must-have features.

Must-Have Features for Modern SAST Tools

Legacy SAST solutions don't align with today's development practices and, as a result, tend to disrupt and slow down development. Modern solutions, on the other hand, improve developer experience (DevEx).

Here's what to look for in a modern solution:



Speed and Accuracy

In today's DevOps environments, speed is critical. But 25% of developers' time is spent waiting on code reviews. That's why developers need quick feedback to fix vulnerabilities before code is merged. Look for SAST tools that provide faster scanning times and focus on reducing false positives, which can lead to wasted time and frustration.



Seamless Integration with DevOps Tools

Modern SAST tools transform the traditional feedback loop into a real-time dialogue between the developer and the security tools, providing immediate, context-sensitive insights exactly where developers need them most. In particular, integration with existing development environments, such as IDEs and CI/CD pipelines is critical. SAST should also work seamlessly with version control systems like GitHub and GitLab, ensuring that security checks happen automatically with every code change and that developers can remediate critical vulnerabilities, fast.



Customization and Configuration Options

You'll want to tailor the analysis to your organization's specific requirements and coding standards. Customization options to adjust scanning rules, set severity thresholds, and define exclusions will be important.



Proprietary Scanners

While open-source scanners may offer initial cost savings, proprietary scanners are continuously updated, ensuring faster detection of vulnerabilities and fewer false positives. Enterprise-grade tools like Cycode's proprietary scanners are optimized for handling large, complex codebases and integrate seamlessly into development environments.



Scalability and Performance

Consider the scalability and performance of the SAST tool, especially for large and complex codebases. The tool should be capable of efficiently analyzing large volumes of code without compromising performance or accuracy.



Reporting and Remediation Support

Look for features that provide actionable insights for remediation, like prioritization of vulnerabilities based on severity, detailed remediation guidance, and integration with issue tracking systems.



AI Capabilities

SAST tools that leverage AI to help with threat detection, suggestions, and resolution are a game-changer. They enhance precision, guide developers through remediation, and prioritize critical vulnerabilities. Cycode's Risk Intelligence Graph (RIG), for example, offers AI-powered context for every security issue, providing developers with precise recommendations on how to fix vulnerabilities efficiently.

Steps to Choose the Right SAST Tool 5

Finding the right SAST solution can be challenging, but by following these steps, you can ensure that the tool fits your organization's needs:

STEP 1

Identify Your Security and Development Requirements

Start by assessing your organization's specific needs, such as supported programming languages, frameworks, and integration with your current CI/CD pipelines. Consider whether your organization needs to meet regulatory requirements like FedRAMP or PCI DSS, and how SAST will fit into your DevSecOps processes.

STEP 2

Evaluate Speed, Accuracy, and Flexibility

Look for a tool that balances speed with accuracy. Legacy SAST tools often suffer from slow scanning speeds and high false-positive rates, which frustrate developers and slow down workflows. Modern SAST solutions, like Cycode's, address these issues with faster scans, lower false positives, and AI-powered suggestions for quicker remediation.

STEP 3

Consider Modern vs. Legacy SAST

Traditional SAST tools have been around for over 25 years but are notorious for generating noisy results that overwhelm teams with non-critical alerts. Modern SAST solutions, in contrast, offer better developer experiences, faster scanning times, and more accurate findings. They also integrate AI to automate remediation, reducing manual effort.

STEP 4

Assess Whether a Point Solution Makes Sense

With the average AppSec team already using 49 tools, it's worth considering whether a point solution is adding complexity rather than solving problems. A standalone SAST tool may solve some problems, but for many organizations, there's a more appealing option: a single platform that covers the entire SDLC, including all components, tools, libraries, languages, CI/CD pipeline, cloud infrastructure, SAST, SCA, and more...all in one. This exists, and it's called Application Security Posture Management (ASPM).

STEP 5

Think About Scalability, Enterprise Support, and a Complete ASPM Approach

If your organization is growing or if you're dealing with complex architectures, scalability is critical. A Complete ASPM platform not only supports larger codebases, but it can integrate with or replace your existing third-party tools, with proprietary scanners like SAST, SCA, and more built in.

This streamlines security management, reduces silos, minimizes alert fatigue, and provides holistic visibility across the SDLC.

Why Cycode?

Cycode goes far beyond offering just a SAST solution.

While its SAST tool addresses the common pain points developers face with traditional static analysis, our Complete ASPM platform bolsters SAST with additional AST tools, pipeline and build security, posture management, and more. The result? Holistic security coverage across the entire SDLC.

Here's how:

Proprietary Scanners:

Unlike many standalone ASPM solutions that rely on third-party scanners or offer limited native capabilities, Cycode's proprietary scanners are built to provide complete visibility into your code. Continuously updated to stay ahead of evolving threats, these scanners offer unparalleled accuracy, fewer false positives, and deeper integration into your security processes. With Cycode's proprietary scanning, you gain a comprehensive view of vulnerabilities, enabling more effective prioritization and remediation.

AI-Driven Remediation:

Powered by the RIG, Cycode uses AI to prioritize vulnerabilities and provide context-aware remediation suggestions. This minimizes the burden on developers and accelerates the fix process, reducing manual effort.

Developer-Centric Design:

Cycode integrates seamlessly with developers' existing workflows, providing real-time feedback within IDEs and CI/CD pipelines without slowing down development. This alignment with DevSecOps principles fosters a culture of security across the development lifecycle.

Have questions or want a custom demo?

Get a personalized demo and learn how you can develop secure software, faster with Cycode.

cycode.