CSO

by John Breeden

9 top SAST and DAST tools

Feature
14 Apr 2022 • 11 mins

Application Security | Enterprise Buyer's Guides | Security

These static application security testing and dynamic application security testing tools can help developers spot code errors and vulnerabilities quicker.

Credit: Sorayut / MF3D / Getty Images

The so-called software supply chain has been generating a lot of buzz these days. It came fully into the spotlight because of the **global intrusion campaign** where attackers used the update process of the popular Orion management software from SolarWinds to upload malicious code. Over 18,000 customers were affected, although the attackers only selectively attacked major corporations and government agencies once their backdoor was installed.

SolarWinds was probably the highest-profile supply chain attack in recent history, but there have **been many others**. The attack led to a reevaluation of who is responsible for security. For example, one of the major responses to the SolarWinds attack was President Biden's **Executive Order** on Improving the Nation's Cybersecurity. Among other things, the order stresses the need for supply chain security. And for the first time, a high-profile government directive specifically mentioned developers' responsibility to deploy secure software.

While the EO only applies to government agencies and those who do business with them, it's becoming apparent that all organizations need to evaluate their software vendors to ensure they are deploying secure code. Whether a company only develops programs and applications for themselves or is part of the software supply chain for others, evaluating and certifying that their code is secure is more critical than ever.

The biggest problem with this effort is that developers have, for many years, almost exclusively been evaluated on how quickly they could code, with security being either an afterthought or someone else's responsibility. Many in the developer community are training in cybersecurity skills but will need help ensuring that they are deploying code that is free from vulnerabilities. That is where SAST and DAST tools can become invaluable assets in helping to secure the software supply chain.

What are SAST and DAST tools?

It's not surprising that both static application security testing (SAST) tools and their close cousins, dynamic application security testing (DAST) tools, have gotten renewed attention with the push to secure the software supply chain. Both can put the power to deploy secure code squarely in the developer's hands, either as part of an **official DevSecOps** program or to help shift more of the responsibility for security closer to where apps are created.

Both SAST and DAST tools have the ultimate goal of making code more secure. Ideally, this will happen long before a program or application makes it into a production environment and before it can become part of the software supply chain. Their goals are the same, but they come at the problem from different angles.

SAST tools analyze the source code of programs and applications still under development. You can integrate some into a continuous integration and continuous delivery (CI/CD) pipeline or set it to activate whenever a developer issues a pull request automatically. That way, SAST tools can ensure that new changes to an app have not unintentionally added vulnerabilities or otherwise broken the program. Some SAST tools can become part of an integrated development environment (IDE), where the platform can warn developers about errors as they work, sort of like how a modern word processor handles spell checking.

On the other hand, you deploy DAST tools after completing and compiling a program. A DAST tool is not so concerned about vulnerabilities hiding within the code, as a SAST tool has (cross fingers) already eliminated them. Instead, a DAST tool acts as an outside tester, trying to hack a program using, for example, exposed HTTP and HTML interfaces. You can also configure some to look for vulnerabilities to the most prevalent attacks in specific industries like finance or retail.

As a result of these differences, SAST tools require specific support for your programming language, whereas DAST tools mostly do not, although they may be able to work with source code as well to pinpoint problems.

While some organizations may exclusively use either a DAST or a SAST tool, these days, it's probably safer for organizations to deploy both, or to work with a tool that has both components. Those that use both SAST and DAST tools can better safeguard their applications and thus also help to protect their links within the software supply chain.

Explore related questions

- **What are the most common software supply chain attacks?**
- **How does President Biden's Executive Order impact software development?**
- **Can SAST tools integrate with CI/CD pipelines?**
- **How do DAST tools identify vulnerabilities in software?**
- **What is the role of DevSecOps in software development?**

Ask a question Ask

The following are some of the top SAST and DAST tools being used today. We tried to find the most popular or highly rated tools to feature, including those that scored highly in other reviews or had very active user groups and install bases. However, there are quite a few choices, so we are sure to have inevitably missed a few good ones. But this list should help get anyone started when trying to pick out a good SAST or DAST tool to help protect their applications and software before deployment.

5 top SAST tools

1. Checkmarx SAST

The **Checkmarx SAST** program combines advanced features with one of the best web-based user interfaces for SAST programs. The interface enables even those new to security concerns in software development to thrive. Checkmarx not only identifies vulnerabilities but goes out of its way to explain why a discovered vulnerability is so risky. And by pushing one “Best Fix Location” button, developers get insight into the easiest and most effective ways of eliminating those problems.

Out of the box, Checkmarx supports over 25 programming languages. You can configure the application to run automatically as part of a CI/CD pipeline or set up custom queries and run as needed. It can also fit into any mainstream IDE or source code management platform.

2. CyberRes Fortify

The **CyberRes Fortify** platform has elements of both SAST and DAST testing. As a SAST product, it uses a clean visual interface to show developers the specific vulnerabilities within code and statistics about the kinds of flaws regularly uncovered, broken down into 810 vulnerability categories. It then directs developers to its gamified training interface, which strives to make learning about security and secure code interesting and fun.

The platform supports 27 programming languages and frameworks and can be deployed on-premises or used as a service. It also can be integrated into most major IDEs such as Eclipse and Visual Studio.

3. Perforce Klocwork SAST

The **Perforce Klocwork SAST** aims for speed in even the largest environments. It works with programs coded in C, C++, Java, JavaScript and Python, even within Docker containers. And it can be integrated into any major IDE like Visual Studio Code, IntelliJ and many others.

Its developers say they designed Klocwork to bridge the gap for SAST tools to enable them to operate in complex environments. You can even use Klocwork to scan truly massive code bases consisting of millions of lines of code. It uses several tricks to cut down those scan times even further, like only scanning the changed areas of code and not the entire program every time.

Klocwork even helps to train developers about security. It fully integrates into the **Secure Code Warrior** training platform, which focuses on security and awareness training. So it can spot problems in code, help fix them, and train developers to become better coders.

4. Spectral SpectralOps Platform

Check Point recently acquired Spectral, but the new company is still actively supporting the **SpectralOps Platform**, likely because of its unique SAST features. SpectralOps uncovers secrets. Specifically, it finds sensitive information like API keys, credentials and tokens that developers often hard-code into programs during development. The idea is to expose those secrets and the security misconfigurations that might allow access to them while a program is still in development. That way, organizations don’t have to worry about malicious users doing the same with a deployed application.

It continually scans at every step along the software development lifecycle, using artificial intelligence to keep track of over 2,000 detection engines. SpectralOps employs other tests to ensure that it’s not dealing with a false positive when it uncovers something suspect. After that, it can report its findings to Slack, issue a JIRA ticket or alert developers using almost any desired communication platform.

5. Veracode Static Analysis SAST

The **Veracode Static Analysis** SAST platform is a cloud service, so it even removes the complexity of maintaining a SAST application within your environment. Veracode embraces the principle of just-in-time learning, meaning that vulnerable code can be flagged as a developer is writing the code. After you fix the code, with help from Veracode, it can generate a report so that organizations can praise their security-aware developers and encourage them with positive reinforcement.

In addition to integration into an IDE, Veracode focuses on speed. Every build of a program or application can be automatically scanned, with an average scan time of just 90 seconds. And the Veracode platform also meticulously tracks what it does, with reports collated in the online portal. That makes passing audits easier, with no surprises, even in highly complex or busy development environments.

4 top DAST tools

1. Acunetix DAST

The **Acunetix DAST platform** uses DAST and IAST (interactive application security testing, which embeds scanning and testing code into a compiled program, similar to **debug symbols**) to look for over 7,000 vulnerabilities in completed code, website designs, applications, etc. By tapping into IAST, Acunetix can launch its scans while a program is actively running, potentially uncovering more vulnerabilities than when looking at an application at rest. IAST should also limit false positives compared to SAST.

The code for the platform is written in C++ to make it speedy. It feels even faster because the platform begins exporting up to 90% of its results while the scan is running and not even halfway complete. Users can set the Acunetix platform to run one time or set up schedules for repeated testing over time. And because the platform is so streamlined, it can even scan multiple environments simultaneously without slowing down.

2. Micro Focus Fortify WebInspect

The Micro Focus **Fortify WebInspect** platform is available as an on-premises installation, a service or a combination of the two in a hybrid environment. While it works as an isolated DAST tool, it integrates into the CI/CD pipeline and can be used by developers, who typically use only SAST tools.

It does this partially by enabling scans looking only for the most critical vulnerabilities. Developers thus get alerted to any really big mistakes and can fix them long before deployment. It can also scan for compliance with various industry and governmental frameworks like NIST 800-53, PCI DSS, OWASP, or HIPAA.

Once a vulnerability is uncovered, the platform uses a graphical interface and step-by-step explanations to reveal the problem and suggest fixes.

3. Synopsys Managed DAST

As the name suggests, the **Synopsys Managed DAST** platform is available as a managed service. Besides the fact that this eliminates the need to maintain and manage the platform internally, another key advantage is that Synopsys provides expert help when needed. If the DAST scan reveals a problem that the development team does not know how to fix, you can tap the experts at Synopsys for help, with subsequent scans verifying mitigation of any issues.

In addition to uncovering all of the common vulnerabilities that plague most programs like SQL injection, cross-site scripting and other security misconfigurations, the Synopsys DAST has a manual scan mode that can look for and discover more complex problems. It can uncover vulnerabilities relating to authentication and session management errors, access control issues, information leakage and others that don’t pop up in a typical scan.

4. Tenable.io Web App Scanning

Tenable has been around for longer than many other cybersecurity companies and has a reputation for providing a robust cloud-based vulnerability management platform for government and private customers. The **Tenable Web App Scanning** application is part of that platform and acts as a capable DAST tool.

The Tenable app only works with web applications, but it performs a deep scan on them. The scope of the scan covers both HTML5 and standard HTML, plus AJAX. The app has a straightforward interface, making it accessible to teams that may not be blessed with professional application security specialists. Setting up the automation is easy, and users can tightly configure which sections of code to scan. For example, you can set the Web App Scanner to only look at parts of an application while, in a possible nod to its government customers, it passes over others.

As a final bonus, you can use the Web App Scanner alone or easily integrate it into any of the other cybersecurity solutions created by Tenable, all of which share a similar interface for easy deployment.

Show me more

Popular *Articles* *Podcasts* *Videos*

01

02

News analysis

Long-running Chinese cyberespionage operation targeted Southeast Asian government

By Lucian Constantin

05 Jun 2024 • 6 mins

Advanced Persistent Threats Hacker Groups Hacking

Podcast

CSO Executive Sessions: The ne

28 May 2024 • 17 mins

CSO and CISO