RAPID7

BUYER'S GUIDE

Application Security Buyer's Guide

15 questions to ask
yourself and your DAST vendor

TABLE OF CONTENTS

An Introduction to the AppSec Market

In today's world of complex, modern web applications, accurate and automated Dynamic Application Security Testing (DAST) tools are rare, but do exist. What characteristics should you look for in a DAST tool to give you greater accuracy and ease of use?

Given the various dimensions upon which you can compare vendors, finding the right DAST tool for you isn't always a walk in the park. The following are common challenges during the buying process:

- Often, when organizations are looking to purchase a DAST tool, they are doing so under a very compressed time frame.
- You need to test a real application with known vulnerabilities; otherwise, it can be difficult to compare one solution's effectiveness against that of another's. By knowing the vulnerabilities present in advance, you can determine which DAST tool is more apt to identifying any and all vulnerabilities. When possible, work with your development team to seed a test application with a variety of SQL injections, XSS, and other vulnerabilities.
- Results from DAST tools will naturally differ quite a bit, due to the differences in configuration, scanning techniques, and reported findings.
- You will be, in certain instances, forced to rely on the word of the DAST vendor. Why? The technology that underlies DAST tools can be a black box.
- It takes a lot of time (time that not many teams have to spare) to check and re-check reports for accuracy.

In this guide, we comprehensively outline the major features and capabilities you should be looking for when selecting a DAST tool. To help you cover all your bases, we've also included some questions and techniques you can leverage to get the most out of your evaluation period. Our goal? To equip you to select the best application security for your organization — one that is automated, accurate, and easy to use. For more advanced application security programs, we've included a few other considerations that will not only improve the effectiveness of your DAST solution, but also its ability to fold seamlessly into the workflows of your development counterparts.

01 Dynamic Application Security Testing Requirements

1 COVERAGE OF MODERN WEB TECHNOLOGIES

Coverage is the first step of accuracy. A DAST tool can't test what it can't find or doesn't understand. Most DAST tools were built to scan HTML, and they do so quite effectively. But times have changed, and in reality very few applications are built solely in HTML. Today's applications have gone beyond static pages to involve advanced web clients and web services that make use of new technologies. These applications are powered by JavaScript and AJAX on the client-side, and often have interfaces built in JSON, REST, and SOAP with CSRF protection thrown in for good measure. Thus, you need a tool that is built to scan apps utilizing modern web technologies, on top of just basic HTML.

Speak with your development team and understand which technologies are used to build the applications you are tasked to secure. Look for DAST tools that explicitly support these technologies out of the box — without a significant amount of training.

2 FUTURE-PROOF STRATEGY

Modern DAST tools need to understand and adapt to new application technologies as they become popular. Inevitably, we will continue to see an increase in application complexity with the emergence of new technologies. While most DAST tools continuously work toward understanding and attacking classic web apps of the past, modern DAST tools need to be architected so that new technologies can be bolted on like drill bits on a drill.

Ask your vendors how their architecture provides the flexibility to handle new technologies.as well, as you upgrade or swap out various tools in your stack.

3 QUICK START CAPABILITIES

The best pen testers love to do things by hand, leading to a comprehensive yet slow, manual process. The reality is that you need those smart pen testers to cover the work that can't be done by automation. A good DAST tool's real value is in its capacity for automation, thus reducing the need for manual testing. The best tool is the one that will work well in a "point and shoot" mode. In many cases, that is all that's possible for understaffed security teams.

Make sure the tool you choose offers a simple, "point and shoot" approach, allowing you to maximize the most of your limited time and resources

4 ARCHITECTURE AND SCALABILITY THAT MEETS YOUR NEEDS

DAST tools can be deployed in a number of ways: on-premise, in the cloud, or as managed services. In addition to where security data is hosted, scalability is also an important component; some organizations may manage just a few dozen web applications, while some organizations can have thousands to tens of thousands of web applications that all need to be secured. Finally, not all organizations are able to execute application security programs in-house due to the lack of staffing, necessitating external consultants to run and manage scans as well as validate vulnerability findings.

Ask your vendors about their different deployment options for DAST, and which would be best for your scalability, staffing, and data handling requirements.

5 AUTHENTICATION AND SESSION MANAGEMENT (DEVELOPER'S FUNZONE, SECURITY'S NIGHTMARE)

Developers seem to revel in creating innovative, complex, and difficult-to-automate schemes for authentication and session management. Your DAST tool needs to have advanced capabilities to authenticate automatically and have backup plans (macros and advanced settings) to tweak, in case there is a clever edge case.

Make sure your tool can automate the login process and maintain a session on your applications. It's also important to make sure the solution has a macro recorder that supports user events so that you can better handle complex authentication scenarios.

6 CUSTOMER SUPPORT AND CUSTOMIZATION

The reality of application scanning today is that your applications are highly customized, making it extremely difficult for DAST tools to address 100% of cases. Each custom application uses unique technical approaches that can trip tools up and cause them to crash. You need to seek out a solution that is flexible and responsive in the face of unique, complex applications.

Find a vendor that has a proven track record of quickly responding to customers' needs and tweaking DAST functionalities to improve testing coverage—especially of complex applications.

7 SOPHISTICATED ATTACK TECHNIQUES

All DAST tools must find a balance between comprehensiveness and performance. In order to improve performance, some DAST tools randomly limit the set of attacks to send based on proprietary choices. Others intelligently profile applications to determine which attacks are useful, and dynamically adjust attacks for each input. This latter approach increases not only the efficiency of the scan, but also its ability to find valid vulnerabilities.

Be sure you understand how your DAST tool selects its attacks, and how configurable the attacks are to fit your needs.

8 REDUNDANT FALSE POSITIVE CHECKING

False positives are simultaneously the bane of automated scanning and a time suck for security teams. Web applications often behave in mysterious ways, and per the nature of the beast, smart DAST tools must check and recheck findings to avoid false positives.

Seek out solutions that provide findings you can be confident in, as well as vendors who are committed to improving the quality of their results based on your feedback.

9 RELEVANT DATA INPUT

During automated scans, there are usually two phases: crawl and attack. During the crawl phase, it is imperative that a tool provide valid data for each input field as expected by the application. For example, if a form is asking for a shipping address, some tools enter random values into each input instead of the expected values. Certain fields such as the ZIP code would be invalid, and the application would subsequently reject the request. In this case, the scan is actually halted, resulting in a less comprehensive scan and the potential for missed vulnerabilities.

Ask vendors what kind of data they use in their attack phase to determine if they are using both expected and unexpected datasets. Furthermore, ask if they are attacking one input at a time.

10 INCLUSION OF EVERY

The point of automation is to handle the repetitive tasks against every input, but this can also lead to slower scan times. To save time, some web application security solutions only check the first several parameters on each page. However, each parameter could use different filters. Why is this important? Tools could be arbitrarily missing vulnerabilities for the sole sake of saving time. Our take? Time savings may not be worth the increased risk.

Make sure the solution you choose checks every parameter on every page for comprehensive identification of application vulnerabilities.

11 SCAN SCHEDULING AND BLACKOUTPERIODS

Continuously assessing your web applications for vulnerabilities is more critical than ever in today's world of rapid development release cycles. In response, automated scan scheduling can be leveraged to help your program stay on top of the vulnerabilities that appear in constantly-evolving applications. Blackout periods can also be useful to ensure scans don't run during times of high activity on an application, and in turn prevent potential negative user impacts.

Check your DAST solution for flexible scan scheduling and blackout period capabilities.

12 INTERACTIVE AND USABLE REPORTING

As you know all too well, "reporting" in most tools takes the form of very, very long PDF files that are difficult to work with. Your team doesn't want to send them, and those in charge of remediation most definitely don't want to open (let alone read) them. A good DAST solution provides you with results that can be used by auditors and developers alike. Reports should be easy to navigate through, and allow you to reproduce the attacks with a few clicks. It should also be easy to understand the context around issues, with the ability to read summaries, drill into details, and view the information in different ways. The unfortunate reality is that developers with limited security training often have a difficult time replicating vulnerabilities, thus slowing down or stopping remediation.

When choosing a solution, make sure the reports are interactive, easy to use, and useful for review and remediation — across all of your stakeholder groups.

13 ATTACK REPLAY

When developers are handed a list of security bugs in their applications, they're often skeptical that the bugs truly exist and aren't just false positives. Some DAST tools offer "Attack Replay" or "Validate" features that enable developers to replay attacks directly within exported vulnerability findings reports. This is game-changing, as developers can now validate that security bugs truly exist, and also test potential source code patches for the vulnerabilities without running another DAST scan.

When evaluating DAST tools, make sure an "Attack Replay" capability is available to reduce friction between security and development teams, and streamline remediation efforts.

14 COMPLIANCE REPORTING

Many organizations will launch application security initiatives in response to regulatory compliance requirements like PCI, HIPAA, and SOX. Often, there are security compliance requirements to adhere to as well, such as the OWASP Top 10. In order to make your life easier, your application security solution should facilitate your journey towards compliance.

The DAST solution you choose should have the ability to generate reports specifically organized and designed around the compliance requirements to which your organization is subject, making it easier for auditors and business stakeholders to understand the compliance risk of your organization's proprietary applications.

15 CUSTOM MOBILE APPLICATIONS,

Custom mobile applications are the new frontier for security teams. They provide native mobile interfaces, but then communicate with web services or APIs (JSON, REST/XML, AMF, etc.) that have the same range of potential vulnerabilities (SQLi, authentication, and session management weaknesses) that web applications have.

Be sure your DAST tool is capable of testing back-end interfaces or APIs: This is where the real weaknesses are likely to be found.

02 DAST Requirements for Advanced Application Security Programs and DevSecOps

1 CONTINUOUS INTEGRATION (CI)

Many organizations are pushing development teams and teams working under the DevSecOps mentality to use Continuous Integration solutions (whether off-the-shelf or home grown) to streamline QA efforts and reduce time-to-market. Security teams are wise to find ways to plug their scanning activities into the CI to ensure every build is security tested before it goes into production. This requires a tool that works well in "point and shoot" mode (see #3), and offers open APIs for running scans.

If your organization is ready to start integrating application security assessment into the Software Development Lifecycle (SDLC), speak with your vendors about how their DAST solutions would fit into your development team's CI toolchain.

2 WAF/IPS LINKING WITH CUSTOM RULES AND QUICK RE-TEST

Due to the volume of applications and vulnerabilities, most organizations are relying on WAFs and IPS devices to protect themselves against vulnerabilities that haven't yet been patched. These WAFs and IPS devices come with default rules, which will not give your custom application all the protection it needs. You will likely require a custom rule that combines knowledge of both the WAF/IPS device and the application.

Be sure you have a sufficient understanding of how rules are created and applied. Look for a solution that goes beyond turning on a default rule from a WAF/IPS to creating truly custom rules for your custom application.

Go Forth and Scan Repeatedly

Each of the recommendations and considerations noted in this guide works toward a simple goal: making sure your DAST solution is lightening your load, not adding onto it. That means automating and streamlining the scanning and reporting process as much as possible. In due time, you'll gain a profound level of visibility including what's wrong and where, see faster remediation times, and most importantly, ensure that applications get and stay secure.

Discover a DAST solution built to help you address modern application security challenges.

Built upon Rapid7's Insight platform, InsightAppSec combines ease-of-use with powerful crawling and attack capabilities. Get unparalleled visibility into your application vulnerabilities within minutes.

Test your own application with a free 30-day trial of InsightAppSec:

Visit www.rapid7.com/try/InsightAppSec to get started today.

About Rapid7

Rapid7 (Nasdaq: RPD) is advancing security with visibility, analytics, and automation delivered through our Insight cloud. Our solutions simplify the complex, allowing security teams to work more effectively with IT and development to reduce vulnerabilities, monitor for malicious behavior, investigate and shut down attacks, and automate routine tasks. Customers around the globe rely on Rapid7 technology, services, and research to improve security outcomes and securely advance their organizations. For more information, visit our website, check out our blog, or follow us on Twitter.

To learn more about Rapid7 or get involved in our threat research,

visit www.rapid7.com.