5 Reasons Why DAST Is Better Than IAST

How DAST goes above and beyond traditional functional testing

What is DAST?

DAST (dynamic application security testing) is the process of analyzing a web application through the front end to find vulnerabilities through simulated attacks. This type of approach evaluates the application from the "outside in" by attacking an application in the way a malicious user would. After a DAST scanner performs these attacks, it looks for results that are not part of the expected result set and identifies security vulnerabilities.

What is IAST?

IAST (interactive application security testing) is an application security testing method that tests the application while it is run by an automated test, human tester, or any activity "interacting" with the application functionality. The core of an IAST tool is sensor modules, which are software libraries included in the application code. These sensor modules keep track of application behavior while the interactive tests are running. The process and feedback happen in real time in your integrated development environment (IDE), continuous integration (CI) environment, quality assurance phase, or in production.

With that in mind, here are five reasons why DAST is better than IAST.

DAST can:

- Go beyond traditional functional tests to find additional vulnerabilities.
- Detect client-side exploits.
- Scan the entire application space, including the environment between the client and server.
- Run independent of the application.
- Find vulnerabilities in modern apps as well as legacy apps.
- Integrate into the CI/CD pipeline.
- Support API discovery and testing.
- Be independent of languages/frameworks.

1. Address the Need for Speed without Sacrificing Coverage and Depth

Due to the increasing number of web-based applications and mobile applications, the need for faster deployment has never been greater than it is now. The robust maintenance of these applications and continuously increasing cyber attacks are some of the major factors driving the growth of the DAST market (Mordor Intelligence DAST Market Trends 2021). But speed of AppSec testing alone can't be traded for depth and quality, and that's where DAST comes into play.

DAST can go beyond the traditional functional tests. IAST depends on some other tool to exercise the application and expose vulnerabilities. With DAST, not only can functional tests be used for exercising the application, but DAST can crawl on its own and find vulnerabilities that IAST isn't exposed to. ScanCentral DAST and the functional application security testing (FAST) proxy functional tests can be used in the same way as an IAST scan. How well do you trust your team in creating these tests, though? With DAST you can start with the functional test, but go deeper and crawl the entire site to find areas not covered in those tests. Also, horizontally scaling DAST scans enables dramatically reduced scan times without permanently dedicating resources. As a result, ScanCentral DAST can be used to scan very large applications in a fraction of the time, which enables integration into CI/CD pipelines and shifting left.

2. IAST Is Limited in What It Can Test

61% of tested apps have at least one high or critical vulnerability not listed in the OWASP Top 10 (2019 Application Security Risk Report, Micro Focus). Because of this, organizations can't limit their testing to just server-side exploits.

DAST can detect client-side exploits. Because IAST inherently instruments the server, it is limited to what it can test. DAST, on the other hand, is scanning as a Pen Tester would. It downloads and executes all the client-side code within the context of the application. This allows for exploiting DOM-based attacks and inspecting the third-party libraries that might be hosted on a CDN or another repository. WebInspect even reports the client-side libraries discovered with its Hacker-Level Insights feature.

3. Replicate What an Attacker Would Do with Outside-in Testing

DAST provides a holistic approach to scanning. The application server is an important part of an application's deployment, but it is just one part. With DAST, the entire application space is scanned. This includes the client-side code and the environment between the client and server (such as load balancers in place, proxies, and Web Application Firewalls), as well as the server and its configuration. This outside-in approach best replicates what an attacker will interact with during an attack and the entire environment is scanned for vulnerabilities. This also allows for targeting the scan to specific scenarios, such as scanning the environment itself, scanning the application server itself, and targeting DOM-based attacks that are client-side only.

Consider a basic weak SSL cipher vulnerability. While static testing and dynamic testing can both detect this weakness, the finding is heavily tied to the application's implementation in production. Dynamic testing will provide a view of the web server configuration for instances where SSL is terminated outside of the application. IAST may give insight into the application server, but it doesn't provide a view into the entire operating environment. Additionally, DAST compliance scans are possible, including GDPR, PCI, and DISA-STIG to name a few.

4. DAST Is a Standalone Tool That Can Scan Any Technology

DAST is independent of the application. IAST requires instrumenting the application server to expose vulnerabilities. While IAST has good coverage of popular application servers, it is still dependent on version parity between the IAST solution and the application server. This creates additional overhead to maintaining the application server, can lock a development team into specific servers and versions due to compatibility, and adds latency to the application server as IAST performs its tests.

DAST is an independent standalone tool that can scan any technology that responds to HTTP traffic. There are no dependencies on frameworks or application versions, and even embedded systems can be scanned with DAST.

5. Scan Both Modern and Legacy Applications

DAST is a very mature security testing method. DAST technology has been around for decades and has matured along with modern web development. This means DAST has the ability to both scan and find vulnerabilities in modern applications, as well as legacy applications. Fortify WebInspect has many thousands of checks and a breadth of scanning technologies that new techniques such as IAST don't have, and ScanCentral DAST supports the volume and velocity of modern application development.

Learn more about Fortify's DAST solution: WebInspect.

Contact us at **CyberRes.com**
Like what you read? Share it.

CyberRes
A Micro Focus Line of Business