opentext™ | Cybersecurity

Position Paper

DAST for the Enterprise—a Shift to the Left

You are probably familiar with launching security scans against just a single web application and generating a report of any vulnerabilities found. This works great if you have a handful of applications to test, but if you have dozens, hundreds, or even thousands of web applications to cover, you need a solution that will scale with your needs. Let's take a step back and understand what's driving the volume of applications that require testing and how security testing fits in.

Table of Contents

Businesses Are Undergoing Digital Transformations

The world runs on software. There's a huge digital transformation going on, accelerated by the COVID-19 pandemic, and it's making us more reliant on digital services. A survey of CIOs by Harvey Nash and KPMG found that tech investment grew at a greater rate in the early months of 2020 than at any other point in history, equal to an extra $15bn per week over this period. And 29% of global CIOs surveyed said that their IT budgets have permanently increased as a result of the pandemic.

This transformation requires organizations to rapidly deliver new functionality through accelerated application delivery. That's something we try to achieve through our software development practices and why software development teams are moving to DevOps. DevOps with Continuous Integration and Continuous Delivery (CI/CD) enables development teams to increase the frequency of code deployments and has other great benefits as well. However, the volume and velocity of applications pushed into production can result in a higher exposure to security risks if application security is unable to keep pace.

Studies have shown that although software defects or flaws are introduced early, they are discovered late in the development lifecycle.

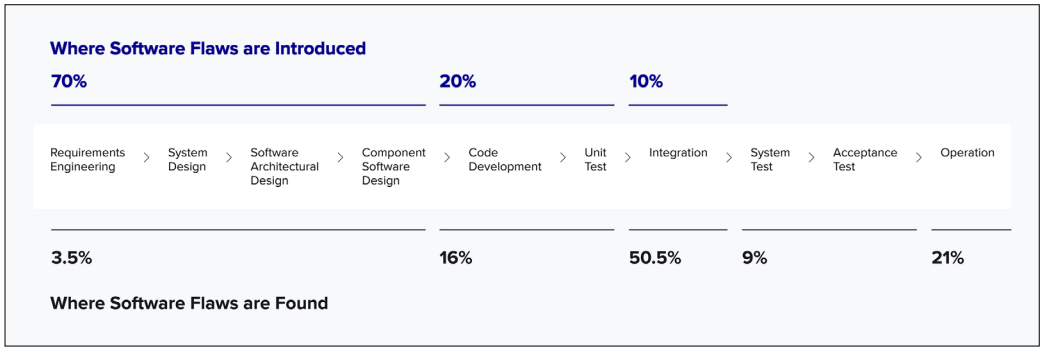
The Value of Application Security

Applications have become a prime target for cyber criminals and other threat actors. The 2021 Verizon Data Breach Investigations Report shows that, once again, web applications remain one of the top attack vectors. For example, according to NTT's 2021 Global Threat Intelligence Report, application-specific and web application attacks together made up 73% of all attacks on the finance sector in 2020. Why are applications such a frequent attack target?

Research and experience have shown that it is extremely difficult to develop vulnerability-free software. Several years ago, a team of CERT researchers established a connection between security vulnerabilities and quality defects[1]. This is worrisome because, according to an SEI study, the average defect level in the US is 0.75 defects per function point or 6,000 per million lines of code (MLOC) for a high-level language[2]. Particularly good levels would be 600 to 1,000 defects per MLOC and exceptional levels would be below 600 defects per MLOC. Thus, software can't always function perfectly as intended. Additionally, SEI research showed that 5% of defects should be categorized as vulnerabilities. And studies have shown that although these software defects or flaws are introduced early, they are discovered late in the development lifecycle.[3]

To minimize the chance of an application being attacked, as well as subsequent damages, application security testing (AST) holds more importance than ever. No high-risk application should be considered ready for public consumption until it has undergone thorough security testing.

1. **https://resources.sei.cmu.edu/asset_files/technicalnote/2014_004_001_428597.pdf**
2. Jones, Caper and Bonssignour, Oliver. The Economics of Software Quality. s.l.: Addison-Wesley Professional, 2011.
3. Woody, C. and Mead, N., 2016: Using Quality Metrics and Security Methods to Predict Software Assurance. Carnegie Mellon University's Software Engineering Institute Blog. **https://insights.sei.cmu.edu/blog/using-quality-metrics-and-security-methods-to-predict-software-assurance/**



Because modern application development leverages a CI/CD approach for faster delivery, the approach to AST requires it to be automated and administered throughout the software lifecycle. As reported in Gartner's 2021 Magic Quadrant for Application Security Testing, a major driver in the evolution of the AST market is the need to support enterprise DevOps initiatives. Organizations require AST offerings that provide high assurance and high-value findings, the report noted, while not slowing down development efforts unnecessarily.

Dynamic analysis is an AST technique that will play an increasingly important role in ensuring that security testing spans the SDLC.

What Is DAST and Why Is It Important?

Dynamic Application Security Testing (DAST) is the process of analyzing a web application through the front end and APIs to find vulnerabilities through simulated attacks on a running application. This type of black box testing approach evaluates the application from the "outside in" by attacking an application as a malicious user would and does not require access to source code. Dynamic analysis can identify runtime vulnerabilities such as logic weaknesses, server misconfiguration, weak authentication, and other problems likely to be encountered after a user is logged into the application.

DAST scanners have long been a favorite tool of enterprise security teams, Quality Assurance (QA) teams, and penetration testers. DAST scans inject real, known attacks into a running application. As the DAST scanner performs these attacks, it looks for results that aren't part of the expected result set and identifies security vulnerabilities. Because these tests get data out of the application and validate the results as unexpected, DAST results provide high confidence of exploitable vulnerabilities and clear surfacing of application security risk. DAST is often used alongside Static Application Security Testing (SAST) and Software Composition Analysis (SCA) tools.

Fortify WebInspect by OpenText™ is Fortify's DAST solution that provides comprehensive vulnerability assessment. WebInspect is available on-premises for Fortify by OpenText™'s customers and is also the tool that supports the Fortify on Demand (FoD SaaS) by OpenText™ DAST solution. A reflection of how Fortify WebInspect has evolved over the last few years

is our attainment of a perfect score of 5.0 in the 2021 Gartner Critical Capabilities report for DAST.

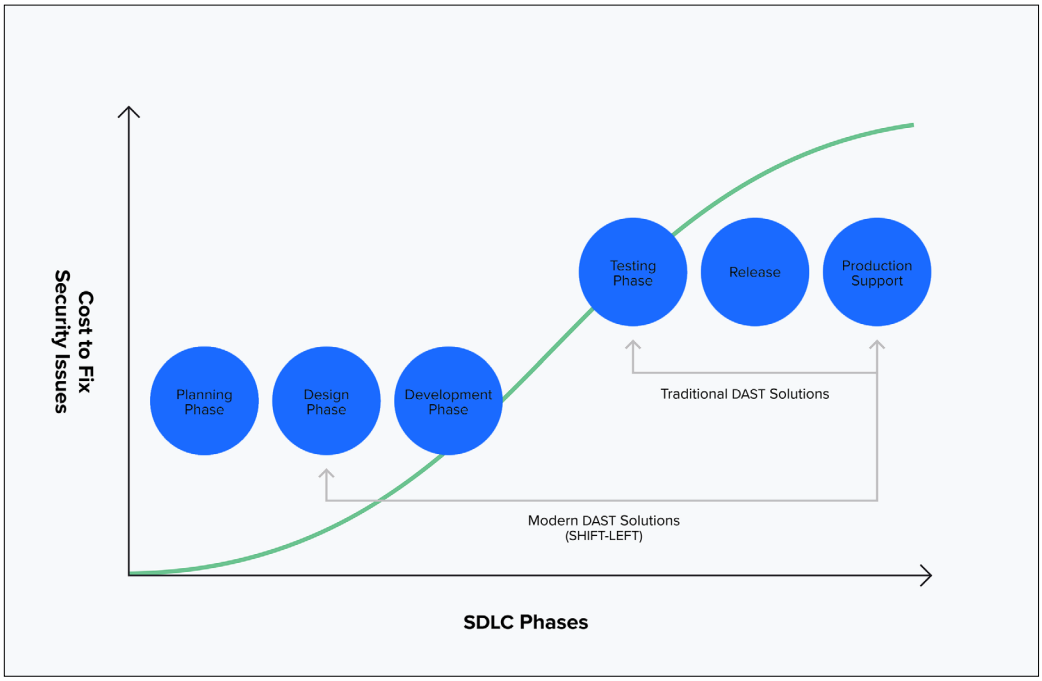
DAST Timing in the SDLC

Because DAST scans are conducted on a running application, they have typically been conducted in parallel with functional testing to gather insights without performing simulated attacks on the running application in production. DAST is often used as a security gate by the Application Security team or QA team to prevent applications that contain critical vulnerabilities from being released into production. For example, WebInspect has prebuilt integrations for OpenText™ Application Lifecycle Management/Quality Center, as well as other security testing and management systems, to work seamlessly with QA teams.

To keep up with the pace of modern development, DAST needs to integrate with processes and pipelines and scale horizontally as needed. Dynamic scans need to deliver results quickly, but also in a smart way that saves time.

In production, security teams use DAST to scan all web applications for vulnerabilities at regular intervals. DAST scan results can also be used in the discovery process of a late lifecycle penetration testing team, making their pentest more efficient, effective, and comprehensive.

However, the days of dedicated security/QA analysts launching manual dynamic scans one application at a time are coming to an end. To keep up with the pace of modern development, DAST needs to integrate with processes and pipelines and scale horizontally as needed. This is especially important when DAST is supporting a security gate. Dynamic scans need to deliver results quickly, but also in a smart way that saves time.



Historically, the turnaround times of DAST scans have precluded their integration into stringent CI/CD workflows. However, we are starting to see developer-driven DAST testing expand, extending the use of Fortify DAST beyond the hands of AppSec/QA and fully within the Dev CI/CD automation pipelines. This enables DAST to be included in Agile, Scrum testing cycles.

With automated security scans in the CI/CD pipeline, it yields many benefits that lead to faster discovery and fixes:

- Developers are alerted to any new vulnerabilities before they hit production, optionally breaking the build to ensure that a review happens before the release.
- Testing can be run against underlying services and APIs instead of being limited to the customer-facing application, leading to faster identification of the underlying issue when a bug is found.
- With DAST scans aligned with functional test scripts, only the portions of the application that are being worked on remain in the context of the code they were working on.

Scans that run automatically and integrate with existing processes and tools keep security and development teams moving quickly. They remain focused on fixing critical issues, not scheduling scans.

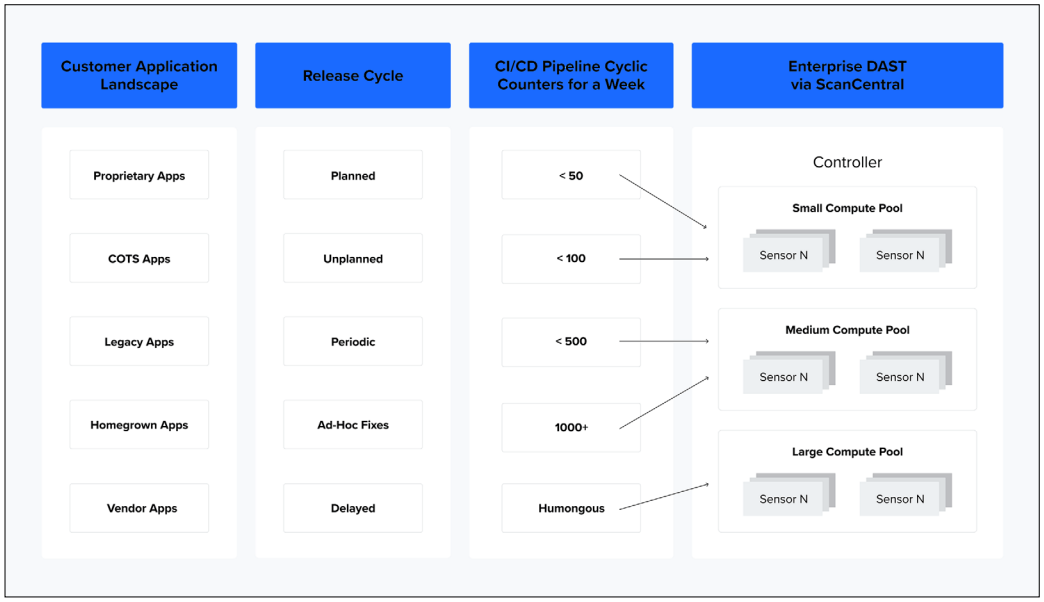
Testing earlier means organizations don't need to re-orient their entire development process to a late-stage security gate as they did before. Shifting DAST left helps bring more stability to the development process.

Customers are leveraging the REST APIs exposed by Fortify WebInspect to run it as a fully automated solution to integrate DAST within the CI/CD pipeline. Testing earlier means organizations don't need to re-orient their entire development process to a late-stage security gate as they did before. Shifting DAST left helps bring more stability to the development process.

ScanCentral DAST

Comprehensive dynamic web application security testing is needed in order to support the volume and velocity of modern applications development. In brief, what's needed is more "agile" and scalable DAST scanning. To meet this need, Fortify WebInspect has been re-engineered for enterprise scalability, including major upgrades to its underlying scan engine.

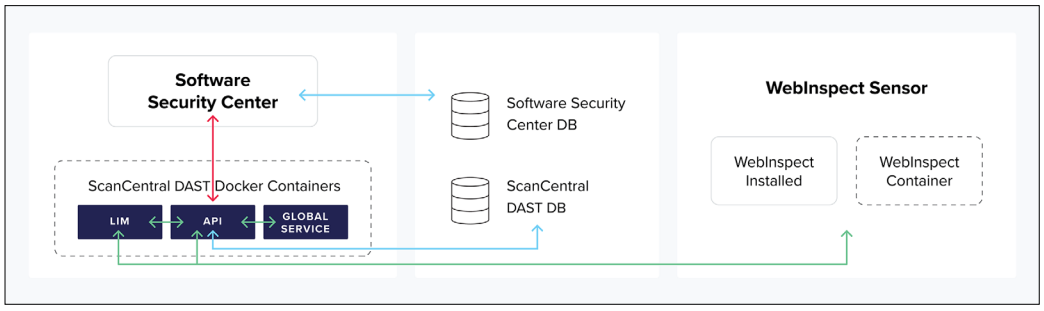
Comprised of the Fortify WebInspect sensor service and other supporting technologies that can be used in conjunction with Fortify Software Security Center Server (SSC) by OpenText™, Fortify ScanCentral DAST enables orchestration and automation of dynamic security scans at a new level. ScanCentral DAST enables the operation of hundreds or even thousands of scans efficiently. We now have a platform that existing WebInspect Enterprise customers can migrate to, as well as interoperate with Fortify ScanCentral SAST. The ability to scan multiple applications at once with ScanCentral further keeps security from being a bottleneck and impacting the velocity of development.



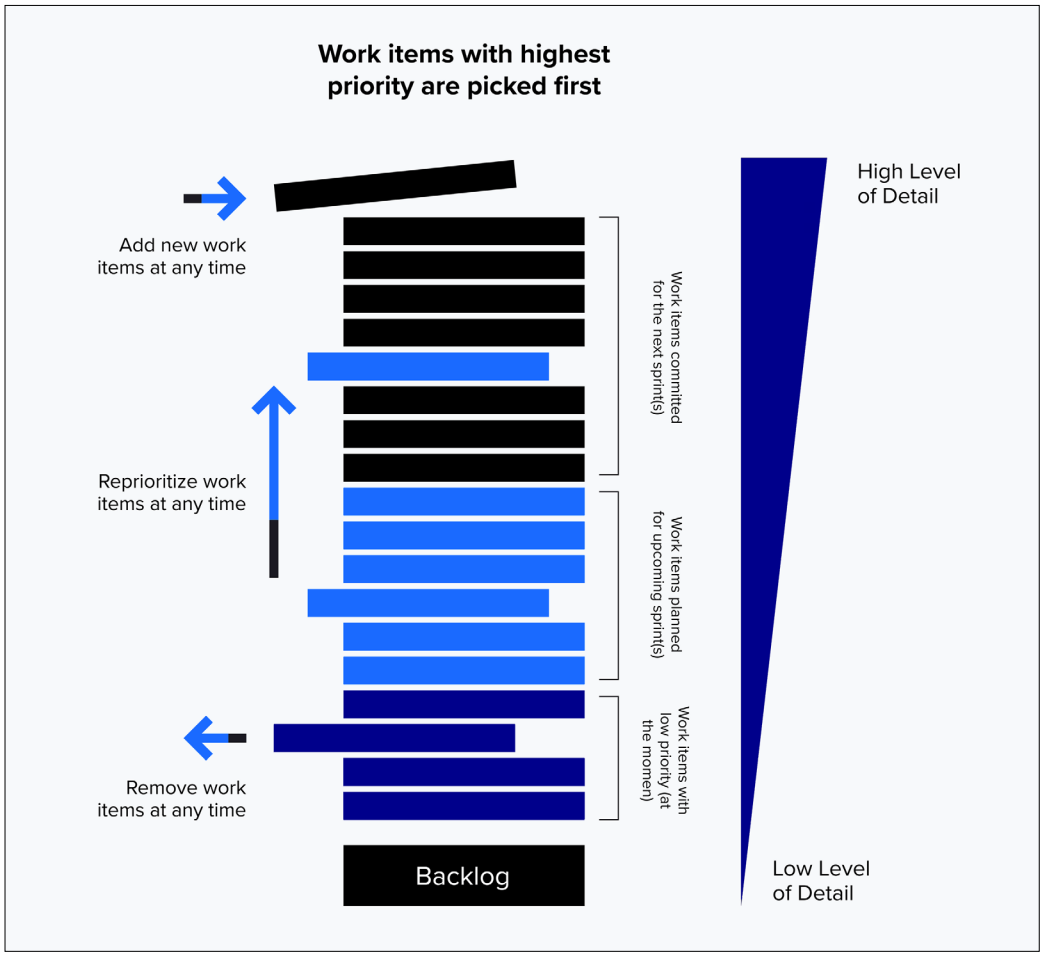
ScanCentral DAST is a scalable architecture that enables Horizontally Scaling through multiple containerized versions of WebInspect (known as Sensors) that can parallel process JavaScript, DOM Rendering, and other activities.

For DAST, scanning Document Object Model (DOM) Rendering and JavaScript execution both consume a large portion of the system resources, as well as a large percentage of scan time. The DOM tree is essentially the tree that contains all the HTML elements (nodes), whereas the render tree is a culmination of the DOM and CSSOM trees. The render tree is the one that is actually rendered onto the page.

ScanCentral DAST is a scalable architecture that enables Horizontally Scaling through multiple containerized versions of WebInspect (known as Sensors) that can parallel process JavaScript, DOM Rendering, and other activities. Horizontally Scaling enables dramatically reduced scan times without permanently dedicating resources. As a result, ScanCentral DAST can scan very large applications in a fraction of the time, which enables integration into CI/CD pipelines and shifting left. SSC can be used for onboarding an application, scheduling scans, and setting limits on scan parameters. All of this is done through a Dockerized deployment, enabling you to have multiple sensors.



SSC provides a single pane of glass for those managing dynamic scans, as well as SAST or open source software (OSS) SCA test results. From a priority perspective, you can set the priority for how the applications are scanned with the sensors. Sensors pick the highest priority scan to work on and higher-priority scans can pause lower-priority scans. Scans can also be moved to other sensors dynamically. The configured scans are applied to sensors that run and get scan results. The results are pushed into SSC for review, comparison, tracking, and trends.



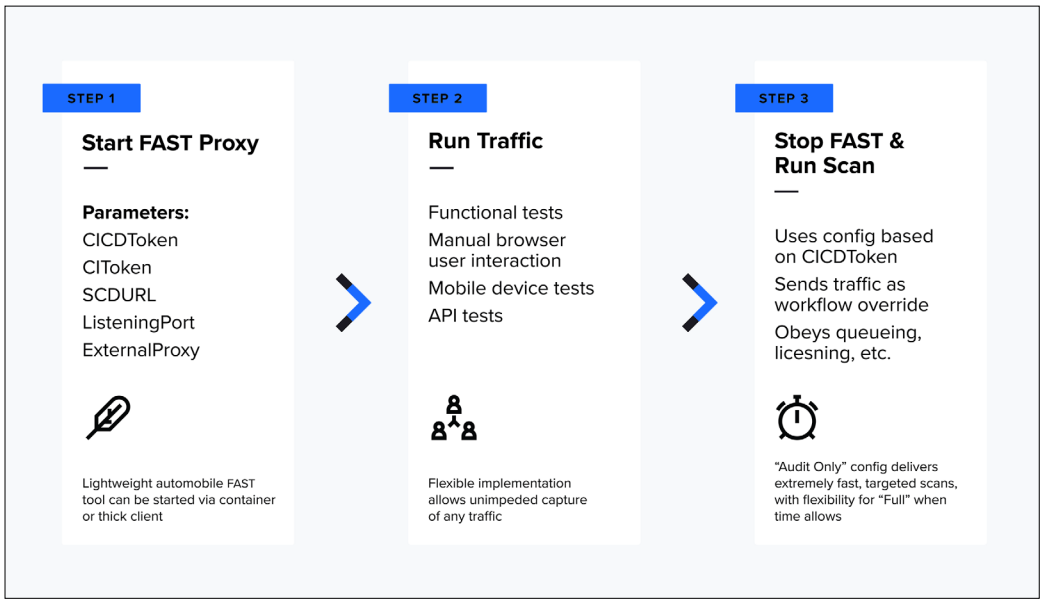
Anything that can be done in ScanCentral through the UI can be done through the API as well. There are also some preconfigured options. For example, with the Azure DevOps plug-in, scans can be easily kicked off from Azure pipelines. The Fortify Jenkins plug-in can be leveraged with the existing build process, so dynamic scans can become part of that, too. You can also integrate ScanCentral DAST with Kubernetes to create a scalable cloud architecture that provides scan scaling functionality, resulting in faster scans of your applications.

WebInspect has augmented the speed improvements of ScanCentral DAST with ease-of-use enhancements such as automated configuration for single-page applications (SPAs), APIs, DOM event parsing, and auto-generated Login Macros. In addition, it now has Redundant Page Detection. This feature is useful for "content sites" such as Amazon or CNN where the same page is displayed over and over and only the pictures and text change. With Redundant Page Detection, WebInspect can compare how similar the pages of an application are and only audit the content once. It does this through the SimHash algorithm, the same technology that Google uses to detect similar pages. This will obviously greatly improve scan times for these types of sites.

FAST technology enables WebInspect users to test the most critical portions of their applications with sub-five-minute scan times, without all the complexity of setup and configuration.

Functional Application Security Testing (FAST) Proxy

DAST is shifting left and Fortify's FAST Proxy makes this real. To realize seamless, fully automated DAST testing as part of CI/CD pipelines, we have fully integrated the FAST proxy with the ScanCentral platform. FAST detects dynamic vulnerabilities based on functional test cases, whether that be Selenium, Cucumber, or any other frameworks that use HTTP to test an application.



FAST technology enables WebInspect users to test the most critical portions of their applications with sub-five-minute scan times, without all the complexity of setup and configuration. The FAST proxy enables DAST to be included in Agile, Scrum testing cycles to align with the features being tested in a sprint.

API Security at Scale

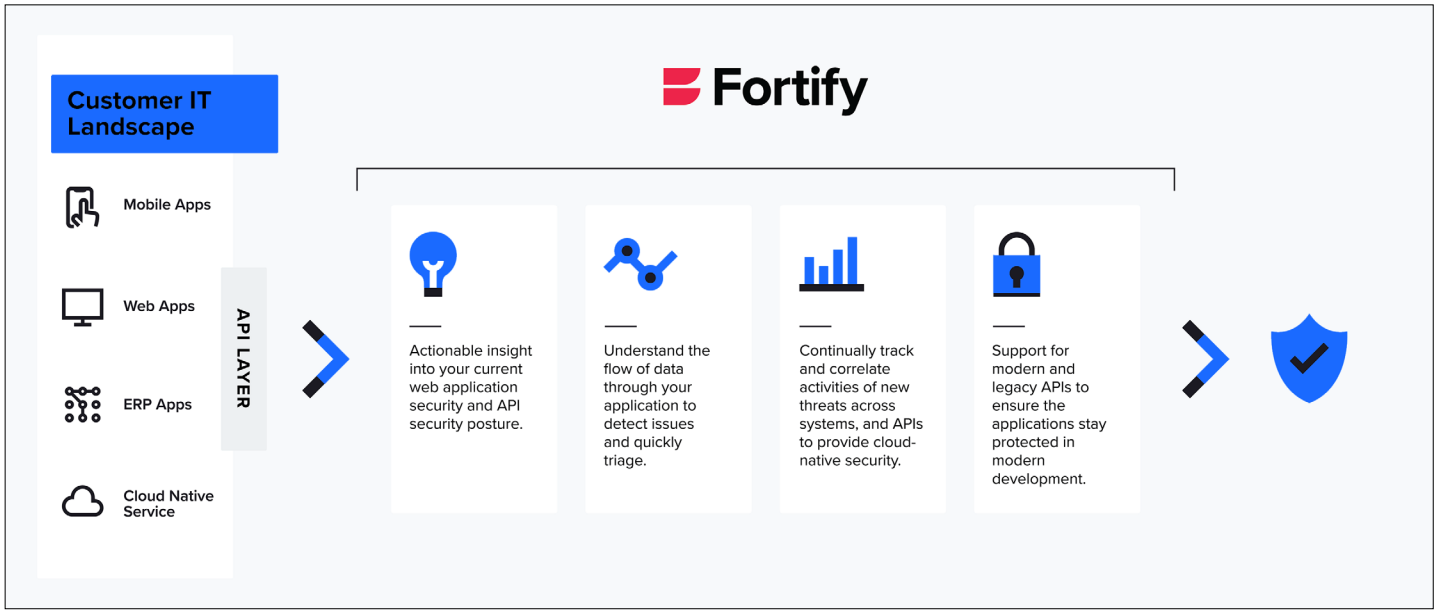
Application Programming Interfaces (APIs) are an essential enabler of innovation in today's digitally-driven world. Applications (or application components) can leverage APIs to connect to other applications and communicate autonomously. APIs are found in customer-facing, partner-facing, and internal applications that support mobile, SaaS, and web applications.

By design, client-side developers need fine-grained access to services and data. Like basic web requests, API calls incorporate URIs, methods, headers, and other parameters. Detailed documentation is usually available for APIs to provide transparency to developers, but (unfortunately) it also provides the blueprint for hackers to utilize for their attacks. APIs define a back door into adjacent systems and applications for those who are intent on gaining access, both legitimately and otherwise. APIs also have the potential to expose application logic and data, therefore providing access to multiple sources of potentially sensitive data and mission-critical services. This, in turn, widens the attack surface exponentially.

According to the 2021 Gartner Magic Quadrant for Application Security Testing, "By 2023, 90% of web-enabled applications will have more surface area for attack in the form of exposed APIs rather than the user interface (UI), up from 50% in 2020. By 2022, API abuses will move from an infrequent to the most frequent attack vector, resulting in data breaches for enterprise web applications." Because APIs are increasingly important and hidden from view, they tend to represent a bigger business risk than other assets.

Most organizations have limited or no awareness of which APIs are exposed by their applications, much less what the correct applied controls are to secure them. Unlike web applications that you can "crawl," APIs have nothing to monitor and can be difficult to discover. Given the increased importance of APIs, it is vital to have better visibility into which APIs exist, who owns them, which port they are listening to, etc. Some organizations, however, are starting to proactively layer in API security controls and leverage API tools to gain visibility. API gateways such as NetIQ Secure API Manager by OpenText™ can create, manage, secure, and measure the APIs in use. Use of tools such as Google's APiGee and API collaboration software such as Swaggerhub or Postman are on the rise and can provide a fuller picture of an API and all its interactions.

> *"By 2023, 90% of web-enabled applications will have more surface area for attack in the form of exposed APIs rather than the user interface (UI), up from 50% in 2020. By 2022, API abuses will move from an infrequent to the most frequent attack vector, resulting in data breaches for enterprise web applications."*
>
> —Magic Quadrant for Application Security Testing



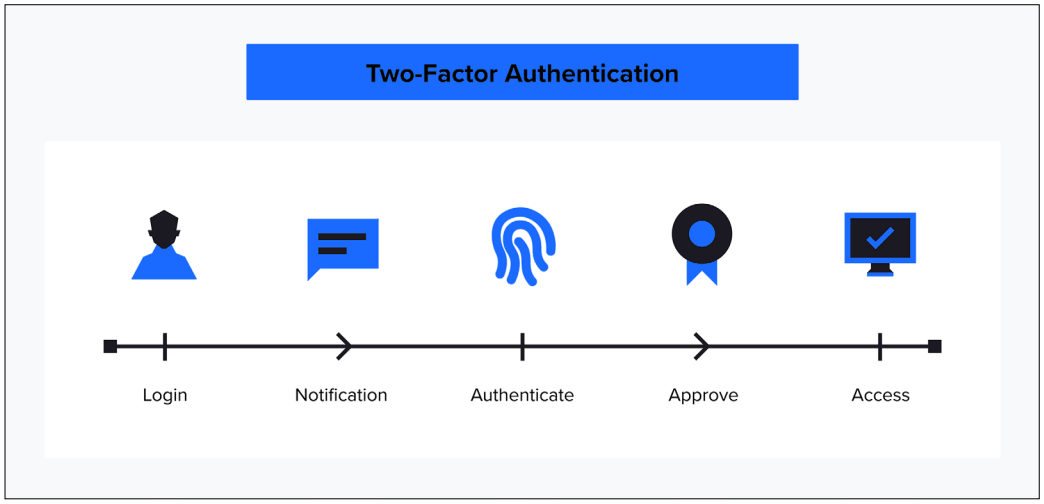
API collaboration tools can be used to provide input into WebInspect for a more complete analysis of exposed APIs. However, with the new API Discovery built into Fortify WebInspect, any Swagger or OpenAPI schema detected during a scan will have its endpoints added to the existing scan. With our automatic state detection, authentication will be applied to the endpoints. In addition, probes will be sent to default locations of popular API frameworks to discover schemas. Fortify WebInspect enables you to scale your scanning of APIs and gain visibility into these potential attack vectors.

Authenticated Scans

DAST scanners can struggle with authentication, leading them to skip restricted sections and leave unchecked vulnerabilities in your environment. While crawling web pages that are accessible to all users is relatively easy, authentication-protected web pages have always posed a challenge for DAST scanners, due to the variety of methods used to authorize page access. Beyond basic login forms, sites might use OAuth, set custom session cookies, require single sign-on (SSO) or two-factor authentication, etc. To complete a scan, DAST scanners might require risky workarounds such as scanning with authentication disabled or scanning manually, which don't scale to enterprise needs. This is especially dangerous considering that pages that require authentication are precisely the ones that attackers are most likely to target.

WebInspect is designed to cope with complicated authentication and custom parameter requirements. Its extensive support for authenticated scanning means you can test applications in their ready-to-run configuration (including almost any authentication), in both staging and production environments.

WebInspect is designed to cope with complicated authentication and custom parameter requirements. Its extensive support for authenticated scanning means you can test applications in their ready-to-run configuration, including almost any authentication, in both staging and production environments.



WebInspect can also handle two-factor authentication by installing a lightweight Android app onto a phone or emulator that can capture SMS and Email tokens and pass them back to the scanner for authentication. Once configured, there is no need for user interaction. WebInspect can scan the entire application in the same way a real-life authenticated attacker would, giving you full confidence in your scan coverage and results.

Shifting DAST Scans Left

With all these new Fortify WebInspect capabilities, the potential use cases for DAST expand greatly. Examples of how you can shift DAST scans left include:

1. **CI/CD + Standard Scan:** ScanCentral DAST automatically runs fast targeted scans triggered by Agile, Scrum tests in CI/CD pipelines and merges those results with slower, more complete DAST scanning configurations that occur at regular intervals.
2. **Hotspot + Notspot:** The hotspot policy is configured to run at a higher priority and the Notspot (less frequently found checks) with a lower priority as sensors are available.
3. **Unauthenticated + Authenticated:** You could scan all your applications without a login macro every month and then run more complete Fortify WebInspect scans that are authenticated once per quarter.
4. **Standard + Adhoc Zero Day:** In addition to regularly scheduled Standard scans, you might want to merge extremely fast Fortify WebInspect scans for very specific zero days. (Apache struts is a great example.)
5. **Functional Test Audit Only + Standard:** You could capture functional tests as workflow macros that are automatically uploaded for fast, audit-only Fortify WebInspect scans and then merge full, standard Fortify WebInspect scans prior to release.

Issue Tracking and Remediation Validation

DAST scans identify high-confidence vulnerabilities in an application and environmental issues. However, to properly remediate these findings, there must be an effective and thorough handoff of security defects to development teams; so, it is critical to integrate DAST with the defect management platforms they are using. This will also enhance traceability within the DevOps teams. By providing development teams with effective and actionable context and reducing the time needed to remediate vulnerabilities, they can make security concerns a higher priority and bring your company closer to a DevSecOps mindset.

WebInspect scan results are initially posted in SSC or the Fortify on Demand (FoD) by OpenText™ portal, which are integrated with defect tracking solutions such as Jira. If it detects critical issues that could prevent a project from moving forward, a build can be made to fail. Or, it can be allowed to continue with a warning/notice if the team is in an early stage of testing so that QA can look at it. That early feedback loop is one of the advantages of this approach.

WebInspect also provides features such as automatic macro generation, macro validation, and fix validation to enable teams to detect and remediate vulnerabilities at scale.

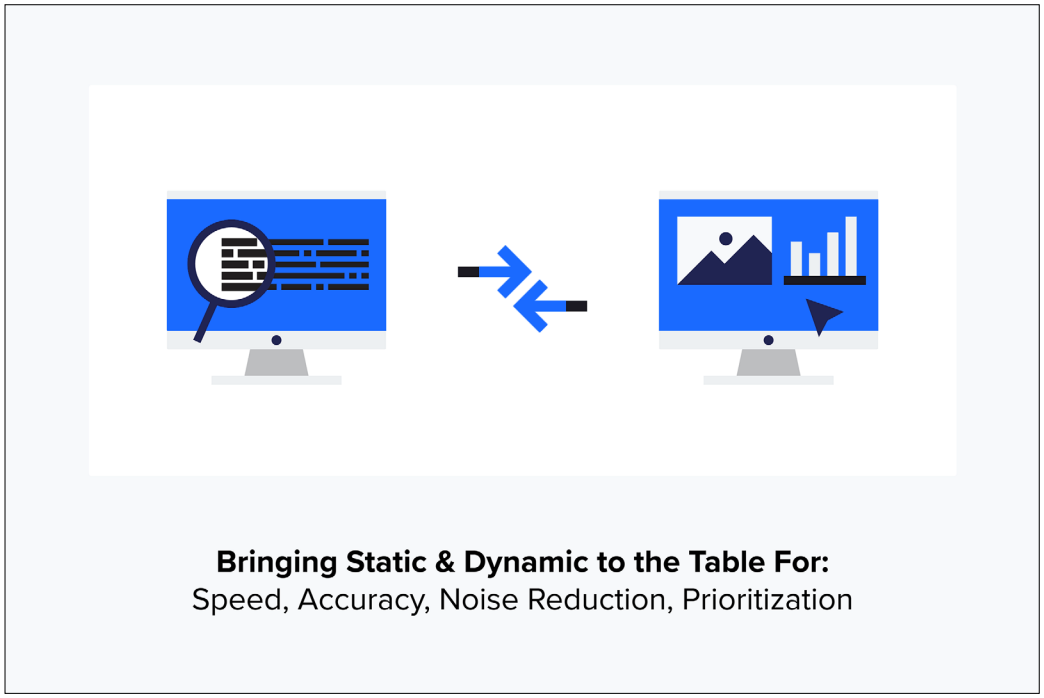
No single testing technique alone is sufficient... By layering dynamic analysis on top of other AST techniques, you gain a valuable additional risk metric that allows you to see a more complete real-world risk picture.

One Team. One Mission. One Fortify

Various AST methods offer different strengths for uncovering security defects. No single testing technique alone is sufficient. As a result, organizations evolve their usage of AST to span development, build, and runtime so that they can see the range of vulnerabilities they need to address. And, just as you need to scale DAST, scaling Static Application Security Testing (SAST) and Software Composition Analysis (SCA) are important as well.

By layering dynamic analysis on top of other AST techniques, you gain a valuable additional risk metric that enables you to see a more complete real-world risk picture. Using Fortify DAST, SAST, and SCA together enables teams to find and flag application security weaknesses and vulnerabilities that require investigation and potential remediation. A smooth, collaborative feedback loop is vital to address issues quickly and for Agile, Scrum Teams to react and minimize the risk of vulnerability as much as possible. A common pattern for this feedback is integrating it into the developer work management system (for example, Azure DevOps or GitHub) and linking alerts or incidents to work items for developers to plan and act upon. This process provides an effective means for developers to resolve issues within their standard workflow, including development, testing, and release.

Correlation of scan results between SAST and DAST scans is available with ScanCentral DAST. Scan results from SCA are passed to Fortify WebInspect and findings that overlap between the scans are displayed as correlated in SSC.



To make it easier to run both Fortify DAST and SAST, the Fortify team has introduced Fortify Scan Machine by OpenText™. With Fortify Scan Machine, you can scan an application using either static or dynamic analysis without a dedicated license. You can scan an application with Fortify SAST and then use the same license to scan the application with Fortify DAST. The license is limited to performing one type of scan at a time, but you aren't restricted to either SAST or DAST, as with our traditional Fortify licenses. You can also buy DAST- or SAST-only licenses to augment the Scan Machine if you need additional scanning in one area.

To learn more about how Fortify helps secure application deployments at scale, visit **www.microfocus.com/en-us/cyberres/application-security**.

Connect with Us

www.opentext.com

opentext™ | Cybersecurity

OpenText Cybersecurity provides comprehensive security solutions for companies and partners of all sizes. From prevention, detection and response to recovery, investigation and compliance, our unified end-to-end platform helps customers build cyber resilience via a holistic security portfolio. Powered by actionable insights from our real-time and contextual threat intelligence, OpenText Cybersecurity customers benefit from high efficacy products, a compliant experience and simplified security to help manage business risk.