CONTRAST SECURITY

WHITEPAPER

INTERACTIVE APPLICATION SECURITY TESTING (IAST)

Software affects virtually every aspect of an individual's finances, safety, government, communication, businesses, and even happiness. Individuals need to trust software — and it makes one feel less safe when it is misused or causes harm to others. So, in response to these concerns, Contrast Security created interactive application security testing (IAST) software called Contrast Assess, that enables software applications to protect themselves against cyberattacks. Contrast Assess is accurate, easy to install, simple to use and scalable.

THE PRIMARY CHALLENGE OF APPLICATION SECURITY

Application vulnerabilities are the leading cause of enterprise breaches and create major headaches for IT organizations. Traditional approaches to the problem, like penetration testing and code review, are too slow and error-prone to be effective in modern high-speed software development processes like Agile and DevOps. Unfortunately, vulnerability scanning tools — both static and dynamic — are spotty and require experts to run (see NIST study).

Contrast Security has invented a new instrumentation technology that uses sensors to passively monitor the behavior of applications and discover vulnerabilities quickly and accurately. Instrumentation provides developers with security feedback as soon as they write their code — not in weeks or months. This paper will explore the Contrast interactive application security testing technique and show how it can help organizations tackle application security without disrupting software development.

The State-of-the-Art

The NSA Center for Assured Software (CAS) Static Analysis Test Results are available at http://appsecusa.org/p/nsacas.pdf. Results from the NIST SAMATE program are available at *https://samate.nist.gov/docs/CAS_2011_SA_Tool_Method.pdf*.

THE IMPORTANCE OF CONTEXT

Since 2002, Contrast Security experts have verified the security of hundreds of millions of lines of source code in thousands of applications, most of which are critical financial, energy, healthcare, defense, and government applications. To provide cost-effective reviews, Contrast invented a highly efficient manual approach that combines the best of threat modeling, architecture review, manual security testing, and security code review techniques. This method is effective because it focuses on extracting the business, technical, and application *context* that is necessary to identify vulnerabilities accurately, quickly, and cost-effectively.

Providing contextual information to static and dynamic scanning tools dramatically improves their performance. Based on this insight, Contrast Security invented a new way to perform fast and fully automated vulnerability analysis from *within a running application*. Contrast technology automatically extracts context and uses that information — along with both static and dynamic techniques — to identify vulnerabilities with accuracy and efficiency. This revolutionary new approach is called *interactive application security testing (IAST)*.

INTERACTIVE APPLICATION SECURITY TESTING (IAST)

Interactive application security testing (IAST) is performed inside the application while it runs and continuously monitors and identifies vulnerabilities. Contrast Security uses aspect-oriented programming techniques[1] to create IAST “sensors” that weave security analysis into an existing application at runtime. These sensors allow Contrast to extract context, data-flow, and control-flow information from within the application and provide access to the actual data values passing through the running code. Because of this wealth of information, Contrast can identify problems that other tools cannot, and achieve an unprecedented level of accuracy without generating false positives.

For example, Contrast can identify credit card numbers extracted from a database and report when these credit cards end up exposed in a log file. It can identify a weak encryption algorithm specified in a properties file, or even data that flows from within an encoded cookie, through a data bean, into a session store, into a JSF component, and finally into a browser — indicating a Cross-site Scripting (XSS) weakness. Contrast can also see vulnerabilities spanning custom code, third party libraries, application frameworks, and the runtime platform itself. Static, dynamic, and even human security analysts have extreme difficulty finding these types of deep security flaws. Through the creation of Contrast Assess rules or “sensors” that become part of the organization’s immune system, Contrast makes it possible to deliver “security as code.” Application security experts can translate their research into new sensors in Contrast Assess, and deploy them into the development process.

Remember the NSA study?

Contrast correctly identifies 74% of the full suite of test cases in the NSA study, and 98% of those focused on web application vulnerabilities with ZERO false alarms. This means that Contrast can identify and provide remediation for vulnerabilities that otherwise may go undetected.

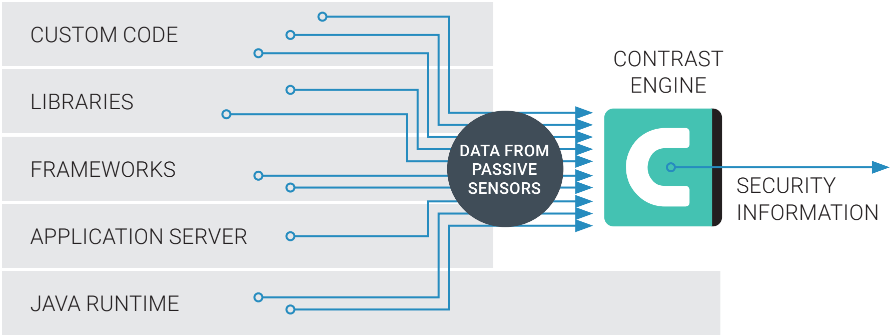


Figure 1. Speed and Accuracy

Contrast’s unique access to information about the application delivers unprecedented levels of speed and accuracy in identifying vulnerabilities as fast as applications run.

[1] *https://en.wikipedia.org/wiki/Aspect-oriented_programming*. Or, for an easy example of how aspect-oriented programming works, see: *http://www.infoworld.com/article/3040557/application-development/my-two-cents-on-aspect-oriented-programming.html*

APPLICATION SECURITY ANALYTICS AT ENTERPRISE SCALE

Getting great results one application at a time isn't good enough. To help organizations meet application security challenges, technology must scale to the entire application portfolio. Contrast brings the power of intrinsic analysis to hundreds of thousands of applications. In some ways, Contrast is like analysis platforms New Relic or Google Analytics. Millions of websites use these powerful tools to extract performance and marketing information from running applications. Both services work by instrumenting running applications, sending findings to a server, and using that data to create useful reports and dashboards.

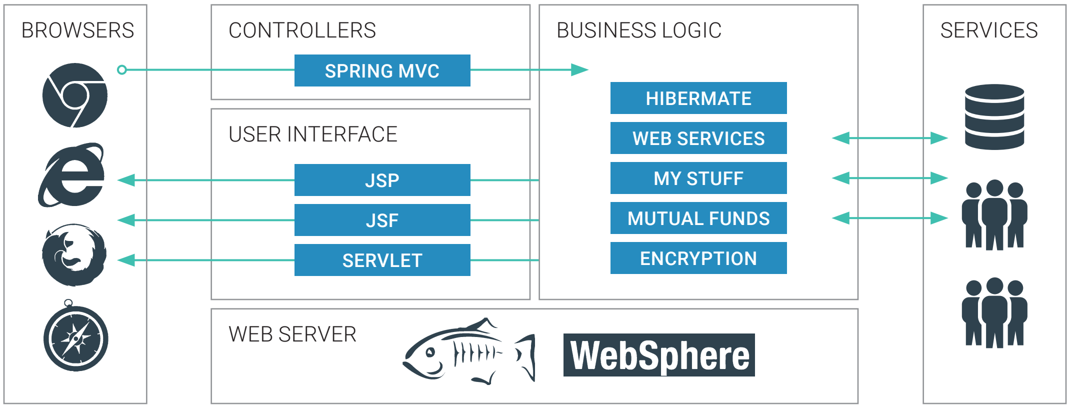
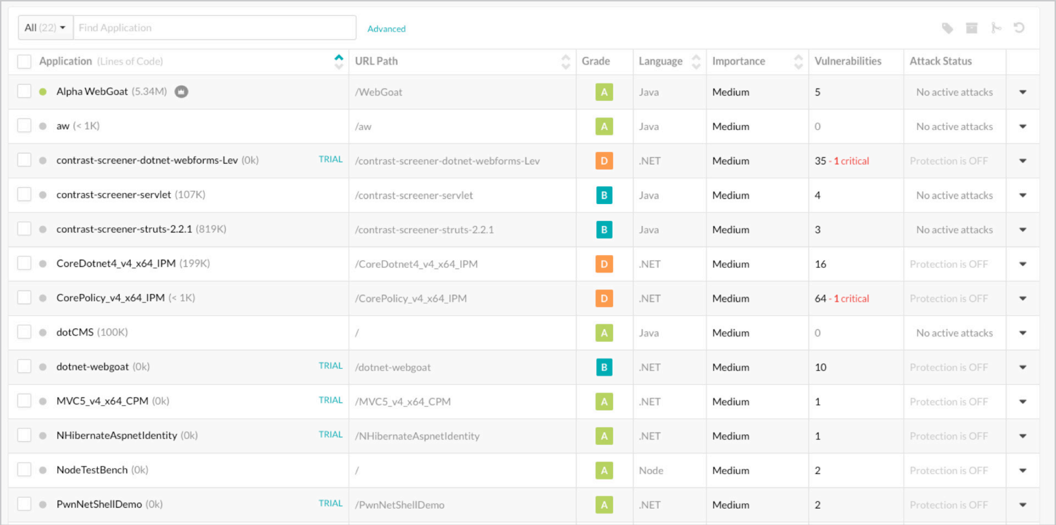


Figure 2. Easy and Scalable

Since Contrast doesn't require a compute farm or large scanning engine, it's easy to add it to all application servers. As applications are tested and run, Contrast reports critical security information over a secure channel to the Contrast Team Server.

Contrast provides application security analytics by employing a similar model. When Contrast's security plugin is installed into application servers, it automatically and invisibly instruments them with simple passive sensors and a powerful rule engine. Getting up and running typically takes less than five minutes and requires no enterprise security skills. As applications run normally during quality assurance and testing, Contrast automatically reports vulnerabilities to the Central Contrast Team Server.



☐	Application (Lines of Code)		URL Path	Grade	Language	Importance	Vulnerabilities	Attack Status
☐	● Alpha WebGoat (5.34M)		/WebGoat	A	Java	Medium	5	No active attacks
☐	● aw (< 1K)		/aw	A	Java	Medium	0	No active attacks
☐	● contrast-screener-dotnet-webforms-Lev (0k)	TRIAL	/contrast-screener-dotnet-webforms-Lev	D	.NET	Medium	35 - 1 critical	Protection is OFF
☐	● contrast-screener-servlet (107K)		/contrast-screener-servlet	B	Java	Medium	4	No active attacks
☐	● contrast-screener-struts-2.2.1 (819K)		/contrast-screener-struts-2.2.1	B	Java	Medium	3	No active attacks
☐	● CoreDotnet4_v4_x64_IPM (199K)		/CoreDotnet4_v4_x64_IPM	D	.NET	Medium	16	Protection is OFF
☐	● CorePolicy_v4_x64_IPM (< 1K)		/CorePolicy_v4_x64_IPM	D	.NET	Medium	64 - 1 critical	Protection is OFF
☐	● dotCMS (100K)		/	A	Java	Medium	0	No active attacks
☐	● dotnet-webgoat (0k)	TRIAL	/dotnet-webgoat	B	.NET	Medium	10	Protection is OFF
☐	● MVC5_v4_x64_CPM (0k)	TRIAL	/MVC5_v4_x64_CPM	A	.NET	Medium	1	Protection is OFF
☐	● NHibernateAspnetIdentity (0k)	TRIAL	/NHibernateAspnetIdentity	A	.NET	Medium	1	Protection is OFF
☐	● NodeTestBench (0k)		/	A	Node	Medium	2	Protection is OFF
☐	● PwnNetShellDemo (0k)	TRIAL	/PwnNetShellDemo	A	.NET	Medium	2	Protection is OFF

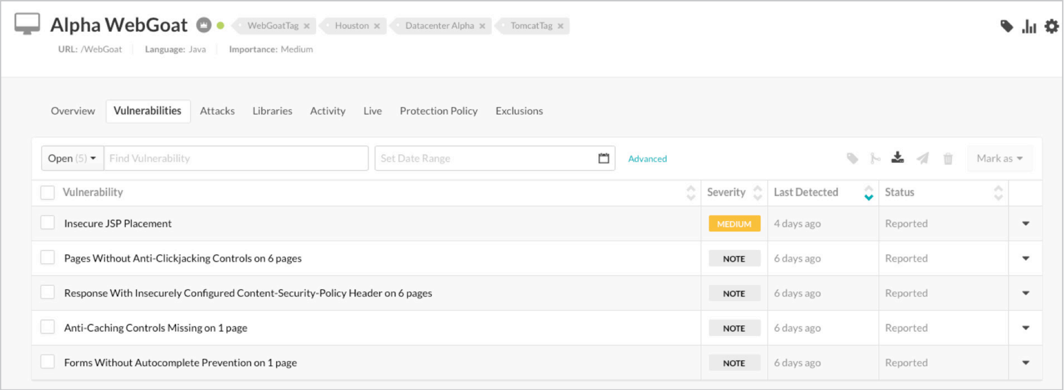
Figure 3. Analytics

All of your applications are presented in a clear, understandable dashboard. Each application also gets its own dashboard with a score for both security and coverage.

Security analysis results appear automatically in a real-time dashboard of critical security information, vulnerabilities, and remediation advice across all of applications. The Contrast dashboard displays charts, trends, metrics, and full vulnerability traces for security, development, and test teams. Each application receives an easy-to-read and understand letter grade for security based on both security and analysis coverage.

The Team Server also explains vulnerabilities to those that need to understand and fix them. Contrast's innovative Security Trace format pinpoints exactly where a vulnerability appears in the code and how it works. All the results above were captured by Contrast after only a few minutes of browsing WebGoat, a deliberately flawed, vulnerable open source application donated to OWASP to assist developers with application security.

The SQL Injection example illustrated above explains to the developer exactly how untrusted data flows through the application and gets embedded in an SQL query without either validation or parameterization. Contrast "speaks the developer's language," and provides remediation guidance that is easy to understand and implement.



Vulnerability	Severity	Last Detected	Status
Insecure JSP Placement	MEDIUM	4 days ago	Reported
Pages Without Anti-Clickjacking Controls on 6 pages	NOTE	6 days ago	Reported
Response With Insecurely Configured Content-Security-Policy Header on 6 pages	NOTE	6 days ago	Reported
Anti-Caching Controls Missing on 1 page	NOTE	6 days ago	Reported
Forms Without Autocomplete Prevention on 1 page	NOTE	6 days ago	Reported

Figure 4. Remediation

Contrast vulnerability reports include all of the details needed to understand the problem, find it in the code, and remediate it correctly. The simple "trace" format shows exactly how the vulnerability works with real data.

AGILE AND WATERFALL COMPATIBLE

Using Contrast doesn't disrupt ordinary software development cycles. Developers receive continuous feedback on the exact code that they are testing in their development environment. QA testers can identify security vulnerabilities and file bug reports without extensive application security experience. Application Security experts can stop wasting time chasing vulnerabilities and false positives and focus on strategic security initiatives. Because Contrast provides continuous vulnerability detection, security analysis does not have to be a large cumbersome effort at the end of the software development lifecycle. Instead, security happens naturally, continuously throughout the development process. Issues are addressed on-the-spot, quickly and efficiently.

OWASP Top Ten

Contrast provides complete coverage of the OWASP Top Ten and beyond. Because Contrast works inside the application, it identifies all complex variants of each vulnerability type.

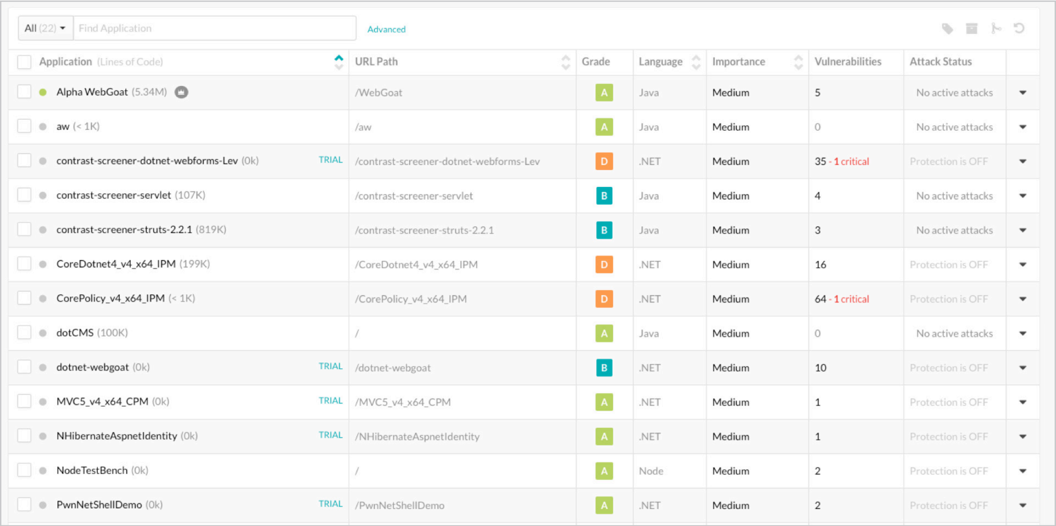
Vulnerability Coverage

Contrast provides coverage over most common vulnerabilities, including the OWASP Top Ten. Unlike tools that claim coverage for a category when they only find a few simple examples, Contrast's coverage is extensive. Note that there are no rules for: Insufficient Transport Layer Protection (A6) and Security Misconfiguration (A9), since these are typically enforced outside of the Java environment.

- SQL injection (A1)
- Blind SQL injection (A1)
- Command injection (A1)
- Reflected XSS (A2)
- Stored XSS (A2)
- Session ID disclosure (A3)
- Path traversal (A8)
- Insecure direct object reference (A4)
- Weak hash algorithm (A7)
- Weak encryption algorithm (A7)
- Authorization missing (A8)
- Arbitrary forward (A10)
- Unchecked redirect (A10)
- No size limit on data read
- File download injection
- HTTP header injection
- And more...

Portfolio Intelligence

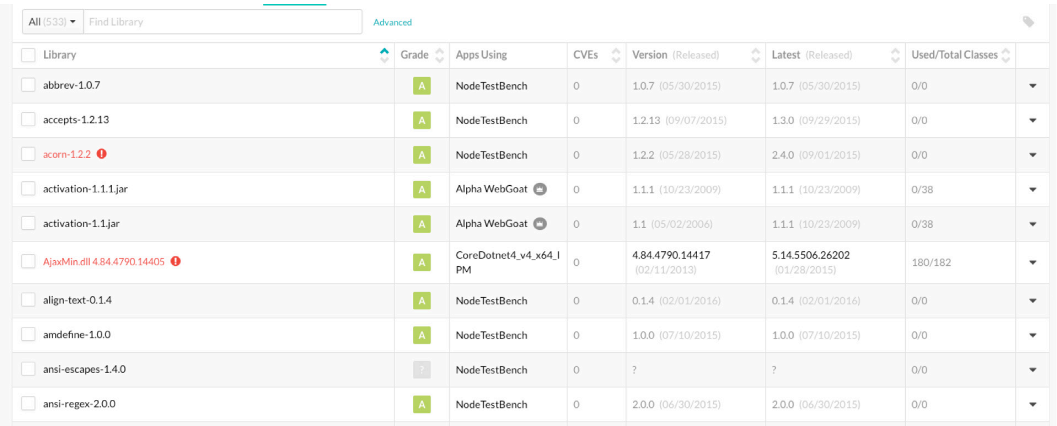
Contrast automatically gathers all application portfolio information that can be so difficult to gather manually. Up-to-date information is available on what applications are in use, including metadata like lines of code, libraries in use, component technologies, architecture, and back end connections.



Application (Lines of Code)	URL Path	Grade	Language	Importance	Vulnerabilities	Attack Status
Alpha WebGoat (5.34M)	/WebGoat	A	Java	Medium	5	No active attacks
aw (< 1K)	/aw	A	Java	Medium	0	No active attacks
contrast-screener-dotnet-webforms-Lev (0k) TRIAL	/contrast-screener-dotnet-webforms-Lev	D	.NET	Medium	35 - 1 critical	Protection is OFF
contrast-screener-servlet (107K)	/contrast-screener-servlet	B	Java	Medium	4	No active attacks
contrast-screener-struts-2.2.1 (819K)	/contrast-screener-struts-2.2.1	B	Java	Medium	3	No active attacks
CoreDotnet4_v4_x64_IPM (199K)	/CoreDotnet4_v4_x64_IPM	D	.NET	Medium	16	Protection is OFF
CorePolicy_v4_x64_IPM (< 1K)	/CorePolicy_v4_x64_IPM	D	.NET	Medium	64 - 1 critical	Protection is OFF
dotCMS (100K)	/	A	Java	Medium	0	No active attacks
dotnet-webgoat (0k) TRIAL	/dotnet-webgoat	B	.NET	Medium	10	Protection is OFF
MVC5_v4_x64_CPM (0k) TRIAL	/MVC5_v4_x64_CPM	A	.NET	Medium	1	Protection is OFF
NHibernateAspnetIdentity (0k) TRIAL	/NHibernateAspnetIdentity	A	.NET	Medium	1	Protection is OFF
NodeTestBench (0k)	/	A	Node	Medium	2	Protection is OFF
PwnNetShellDemo (0k) TRIAL	/PwnNetShellDemo	A	.NET	Medium	2	Protection is OFF

THIRD PARTY CODE ANALYSIS

Like icebergs, 80 percent of the code in modern applications is "beneath the surface," lurking in libraries, frameworks, and other components. Applications often have 50 or more of these libraries, comprising millions of lines of potentially vulnerable code. We released a study[2] detailing findings from 113 million downloads of the 31 most popular Java frameworks and security libraries from more than 60,000 organizations. Contrast Assess automatically analyzes these libraries and provides a detailed dashboard.

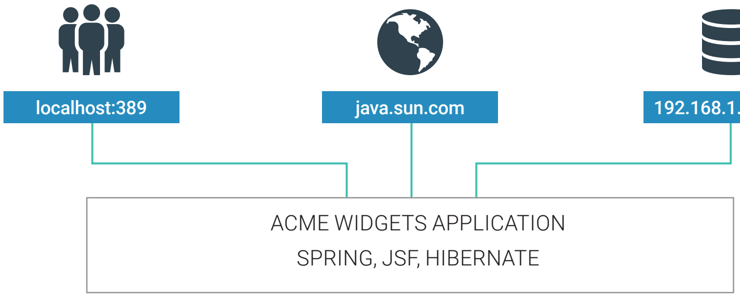


□ Library	Grade	Apps Using	CVEs	Version (Released)	Latest (Released)	Used/Total Classes
□ abbrev-1.0.7	A	NodeTestBench	0	1.0.7 (05/30/2015)	1.0.7 (05/30/2015)	0/0
□ accepts-1.2.13	A	NodeTestBench	0	1.2.13 (09/07/2015)	1.3.0 (09/29/2015)	0/0
□ acorn-1.2.2 ❗	A	NodeTestBench	0	1.2.2 (05/28/2015)	2.4.0 (09/01/2015)	0/0
□ activation-1.1.1.jar	A	Alpha WebGoat	0	1.1.1 (10/23/2009)	1.1.1 (10/23/2009)	0/38
□ activation-1.1.jar	A	Alpha WebGoat	0	1.1 (05/02/2006)	1.1.1 (10/23/2009)	0/38
□ AjaxMin.dll 4.84.4790.14405 ❗	A	CoreDotnet4_v4_x64_I PM	0	4.84.4790.14417 (02/11/2013)	5.14.5506.26202 (01/28/2015)	180/182
□ align-text-0.1.4	A	NodeTestBench	0	0.1.4 (02/01/2016)	0.1.4 (02/01/2016)	0/0
□ amdefine-1.0.0	A	NodeTestBench	0	1.0.0 (07/10/2015)	1.0.0 (07/10/2015)	0/0
□ ansi-escapes-1.4.0	?	NodeTestBench	0	?	?	0/0
□ ansi-regex-2.0.0	A	NodeTestBench	0	2.0.0 (06/30/2015)	2.0.0 (06/30/2015)	0/0

ARCHITECTURAL ANALYSIS

Understanding an application's architecture is extremely helpful when performing security analysis. Contrast gathers information from within the running application about the software architecture and connected components. Contrast automatically generates simple diagrams that illustrate the application's major architectural components. This information helps the developer quickly identify the meaning of a vulnerability that Contrast pinpoints. In this example, Contrast has correctly identified that the WebGoat application has three backend connections: an LDAP directory, a web service, and a database. Contrast lists the frameworks being used within the application: Spring, JSF, and Hibernate. Imagine the benefit of having up-to-date architectural information available, on demand, across entire application portfolio.

[2] Source: *http://cdn2.hubspot.net/hub/315719/file-1988689661-pdf/download-files/The_Unfortunate_Reality_of_Insecure_Libraries.pdf?t=1460664477246*



Unfortunate Reality

In an extensive study, we discovered that 29.8 million (26%) of open source library downloads in 2011 had known vulnerabilities. Further, more than half of the Global 500 use software built using components with vulnerable code. Read the rest of the study at: *https://www.aspectsecurity.com/blog/theunfortunate-reality-of-insecure-libraries*

CONTRAST KEY BENEFITS

Real-Time Vulnerability Detection and Expert Guidance

Contrast monitors Java and .NET code execution, data flow, configurations and more to quickly find dangerous vulnerabilities with virtually no false positives. Code-level pinpointing eliminates guesswork while context sensitive guidance enables quick remediation.

Portfolio-Class Scalability

Contrast transparently automates application security to support application portfolios of virtually any size. New applications are discovered automatically as they are run. Executive-level portfolio dashboards display the entire portfolio security posture in real-time.

SaaS, On-Site and IDE Deployment

It takes minutes to go from zero to resolving application security issues using Contrast's SaaS service. Contrast can also be hosted and administered on-site, enabling a completely administered private service.

Library Inventory and Analysis

As much as 80 percent of software code comes from open source and third-party libraries. Contrast automatically discovers third-party libraries, alerts to the known (and unknown) risks they may bring with them, and provides critical versioning and usage information that helps remediate risks.

Agile Speed and Seamless Automation

Continuous integration and deployment require fast and continuous security. Scriptable silent installers, automated updates, and a REST API enable Contrast to deliver security as fast as applications change.

SUMMARY

Contrast Assess is a new application security solution that provides a fast, accurate, easy, and scalable way eliminate the most serious risks facing enterprises today.

Organizations using Contrast Assess receive continuous, always-on visibility into the security of all their applications. Contrast analyzes every line of code, in every application, for visibility from the inside. By knowing what's happening across the entire application portfolio, organizations can prioritize their development and operations teams to remedy the most critical risks right now, and reduce friction throughout the entire software lifecycle.

Unlike tools that create bottlenecks through periodic or serial application portfolio testing, Contrast uses a highly scalable architecture that empowers every application to analyze, enforce and communicate about application security. Contrast strengthens an organization's immune system to defeat vulnerabilities across the entire application portfolio, rather than only for a chosen few.

CONTRAST SECURITY

240 3rd Street
Los Altos, CA 94022
888.371.1333

121916

Contrast Security is the world's leading provider of security technology that enables software applications to protect themselves against cyberattacks. Contrast's patented deep security instrumentation is the breakthrough technology that enables highly accurate analysis and always-on protection of an entire application portfolio, without disruptive scanning or expensive security experts. Only Contrast has intelligent agents that work actively inside applications to prevent data breaches, defeat hackers and secure the entire enterprise from development, to operations, to production.