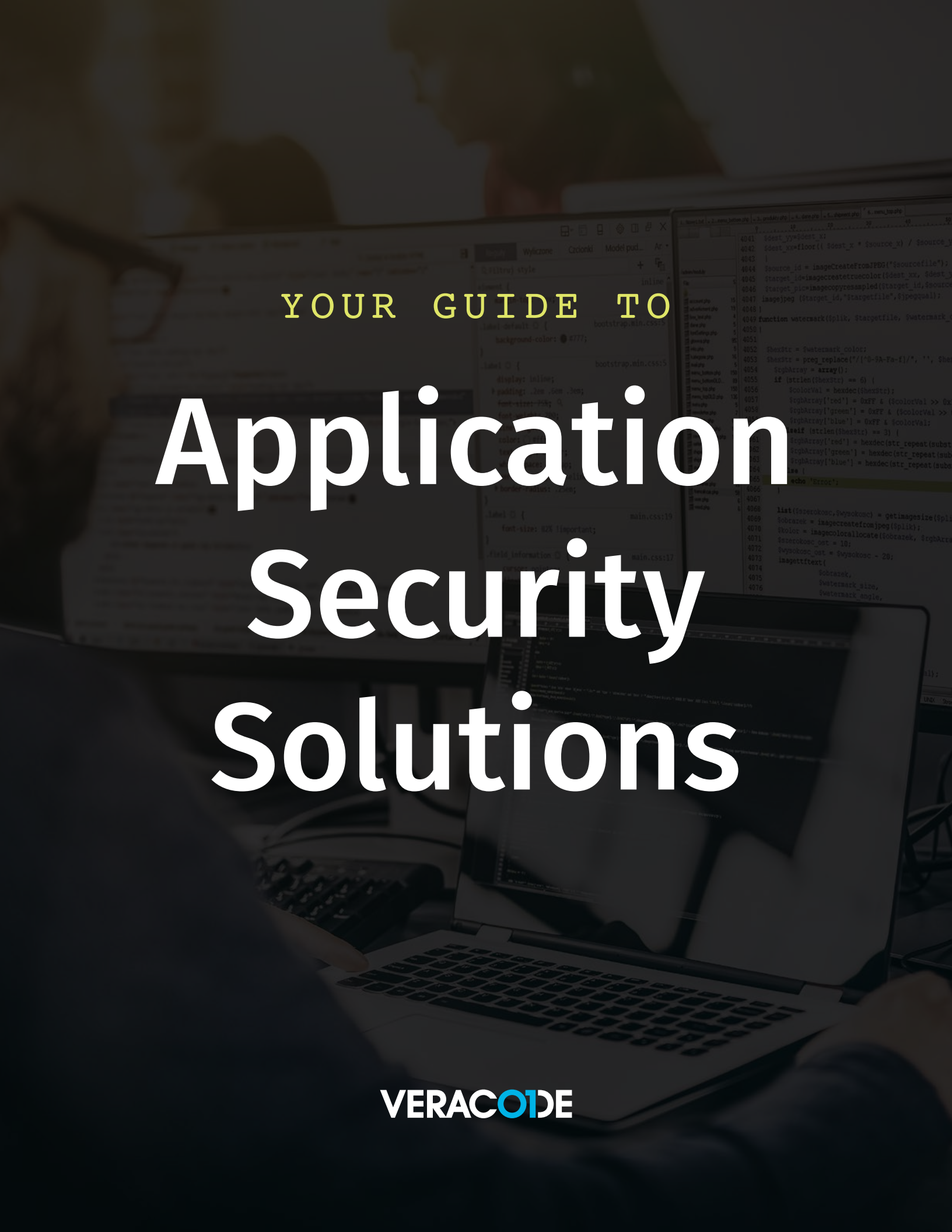

YOUR GUIDE TO

Application Security Solutions

VERACODE

COMMON APPLICATION VULNERABILITIES

SQL Injection

- Allows an attacker to submit a database SQL command, exposing the back-end database.
- Gives an attacker the ability to create, read, update, alter or delete data.
- Found in about 24 percent of all applications.

Cross-Site Scripting (XXS)

- Occurs when "malicious scripts are injected into otherwise benign and trusted websites" (according to OWASP).
- Stems from the security weaknesses of client-side scripting languages.
- Found in nearly half of all applications.

CRLF Injection

- Includes any vulnerability that enables any kind of Carriage Return Line Feed (CRLF) injection attack.
- Encompasses flaws involving improper output neutralization for logs and improper neutralization of CRLF in HTTP headers.
- Found in close to 60 percent of applications.

Why You Need Application Security

Every company is now a software company. Companies of all sizes in all industries are churning out applications more rapidly than ever in order to move faster; better communicate with customers, prospects and partners; and differentiate themselves in this digital world.

To keep pace, organizations are not just developing more apps, but also buying more and supplementing their code more often with open source libraries. In the end, every organization is dependent on applications, and if these applications are insecure and at risk, so is the business.

Yet information security has not kept pace with this new software-driven world. Traditional defenses are proving inadequate in this environment. As users and applications become the risk focal point, there is no hard and fast perimeter security professionals can put a wall around. Consequently, application layers remain insufficiently secured. In fact, application layer attacks are now the most frequent pattern in confirmed breaches (Verizon 2020 Data Breach Investigations Report). And as this threat intensifies, so do the security regulations organizations have to understand and comply with. In this new landscape, application security becomes key.

What is Application Security?

Application security, or "AppSec," is what an organization does to protect its critical data from external threats by ensuring the security of all of the software used to run the business, whether built internally, bought or downloaded.

Application security helps identify, fix and prevent security vulnerabilities in any kind of software application. A software "vulnerability" is an unintended flaw or weakness in the software that leads it to process critical data in an insecure way. By exploiting these "holes" in applications, cybercriminals can gain entry into an organization's systems and steal confidential data. Common software vulnerabilities include SQL injection, CRLF injection and Cross-Site Scripting (XSS). And almost every application has vulnerabilities. Veracode's State of Software Security Report revealed that more than 83 percent of all applications have at least one vulnerability in them; more than 20 percent have at least one critical severity flaw. Commercial software, financial services software, software written by government agencies…all are vulnerable.

Types of Application Security Solutions

Although there are a variety of application security technologies, there is no silver bullet. You need to gather the strengths of multiple analysis techniques along the entire application lifetime — from development to testing to production — to drive down application risk.

In addition, before looking at any AppSec tools or services, organizations should first develop a solid strategy. At a high level, the strategy should address, and continuously improve, these basic steps:

- Assessment of risk
- Identification of vulnerabilities
- Fixing flaws
- Better managing development processes
- Refining the reporting and management of the program

CAPABILITIES	1 STATIC ANALYSIS	2 DYNAMIC ANALYSIS	3 INTERACTIVE APPLICATION SECURITY TESTING	4 SOFTWARE COMPOSITION ANALYSIS	5 MANUAL PENETRATION TESTING
Flaws in Custom Web Apps (CWEs)	✓	✓	✓		✓
Flaws in Custom Non-Web Apps (CWEs)	✓		✓		✓
Flaws in Mobile Apps (CWEs)	✓		✓		✓
Known Vulnerabilities in Open Source Components (CVEs)				✓	✓
Behavioral Issues (CWEs)	✓		✓		✓
Configuration Errors (CWEs)		✓			✓
DOM-Based Cross-Site Scripting	✓	✓			✓
Business Logic Flaws (CWEs)					✓
Coverage of Full Application	✓	✓		✓	✓
Repeatable for Process for Automation	✓	✓	✓	✓	
Scalable to All Corporate Applications	✓	✓	✓	✓	
Scan Speed	Seconds to Hours	Hours	Seconds to Minutes	Seconds to Minutes	Days to Weeks
Cost	$$	$	$$$	$	$$$$

CLOUD VS. ON-PREMISES SOLUTIONS

In addition to investigating the many techniques available to assess the security of their applications, enterprises must also decide whether on-premises tools or a cloud-based service best fits their needs.

With on-premises tools, the enterprise's security team installs and maintains an application security tool and manages the application assessment process in-house. With a cloud-based service, there is no equipment on-site, and a third party manages the enterprise's application security assessment via the Internet.

Although the trend is toward cloud-based application security and away from on-premises tools, some organizations continue to use on-premises tools, or a combination of on-premises tools and a cloud-based service. This more traditional approach typically appeals to organizations uncomfortable uploading code to the cloud to be assessed, or that want or need the control afforded by on-premises tools. However, on-premises solutions require in-house expertise and are cumbersome to scale. This in-house expertise requirement is proving especially difficult for many organizations. (ISC)2's Global Information Security Workforce Study reported that the number of unfilled cybersecurity jobs will rise to a whopping 1.8 million by 2022.

1 Static Analysis

Static application security testing (SAST) is a testing process that looks at the application from the inside out. This test process is performed without executing the program, but rather by examining the source code, byte code or application binaries for signs of security vulnerabilities.

In the static test process, the application data and control paths are modelled and then analyzed for security weaknesses. Static analysis is a test of the internal structure of the application, rather than functional testing.

As the development process has shifted toward the DevOps model and become faster and more incremental, static analysis has had to shift along with it. Today, static analysis needs to integrate early into development processes, while still giving the security team the policy check they need at the end of the development cycle. The most effective static analysis solutions today give developers quick security feedback on small sections of code as they're writing it, and then later give security professionals a pass/fail on whether a full application meets policy.

STRENGTHS OF SAST

- **Find vulnerabilities earlier in software development lifecycle (SDLC).** Remediation is easier and less expensive when flaws are found early in the development process.
- **Find exact location of vulnerability.** Because it analyzes applications' source code, or bytecode or binary code, developers can pinpoint the precise location of the vulnerability.
- **Integrate into development processes.** Static analysis is less disruptive to DevOps development processes.
- **Get 100% coverage.** This test assesses all data paths, regardless of the data or user privileges.
- **Scale more easily.** There are no human resources limitations.

CAUTIONS OF SAST

- **Cannot test for certain vulnerability classes.** For example, SAST won't find authorization issues, business logic problems or problems stemming from misconfiguration of applications at runtime.

PRO TIP

Binary analysis scanners can also follow the data flow through third-party components that you don't have the source code for, reducing the number of false positives because they don't miss sanitization functions in compiled code. Source code scanners don't have this capability. Binary scanners are also better suited to scan legacy applications or third-party applications, for which you may not have access to the source code.

Veracode's analysis of over 85,000 applications found that more than 70 percent have a security flaw in an open source library.

2 Dynamic Analysis

Dynamic application security testing (DAST) looks at the application from the outside in — by examining it in its running state and trying to manipulate it in order to discover security vulnerabilities. The dynamic test simulates attacks against a web application and analyzes the application's reactions, determining whether it is vulnerable.

STRENGTHS OF DAST

- **Finds runtime issues**, such as authentication issues and server misconfiguration issues, that can't easily be found when the code is in its offline state.
- **Can determine the likelihood that a vulnerability will be exploited**, because it analyzes the application's actual response.

CAUTIONS OF DAST

- **Cannot point to the line of code** where a vulnerability originates.
- **Cannot locate business logic flaws and cannot locate the line of code where the vulnerability originates.**
- **Only works in web applications.**

PRO TIPS

- **If you have a large organization,** look for a solution that can scale to scan all of your applications in parallel. DAST scans can run for days to ensure that they don't disrupt a production application, so waiting for each scan to complete before starting the next scan can really throw off your audit deadline schedule.
- **If you have more applications than you can count,** look for a solution that can find all of your company's applications that are exposed to the Internet. You'll be surprised to find all the applications you didn't know about.

SECURE CODING

Ensuring that developers understand and practice secure coding plays a major role in any application security solution. When developers are trained to avoid introducing vulnerabilities, you are effectively stopping the problem at its source. eLearning and hands-on training are two proven options for getting developers up to speed on secure coding and remediation best practices.

Hands-On Training

The pace of software development today requires that security be built into code from the start, during the coding phase, but most developers don't have the tools needed to create secure code. Veracode offers two forms of hand-on, secure code training: Security Labs and the IDE Scan. With Veracode Security Labs, developers build AppSec skills through hands-on-keyboard experience with practical, real-world examples and interactive scenarios so that new skills can be applied immediately. With the IDE Scan, developers receive real-time security feedback while they code, including positive reinforcement, remediation guidance, code examples, and links to Veracode AppSec Tutorials.

eLearning

Research done by Veracode has found that implementing an eLearning program has a big impact on vulnerability remediation, as well as on reduction in overall flaw density. Specifically, we found that development organizations that leverage eLearning see a 19 percent improvement in fix rate.

Remediation Coaching

Our research has also found that organizations using remediation coaching services improve their fix rate by 88 percent (2017 State of Software Security).

"Businesses aren't asking for SAST, SCA, DAST — they're asking, how do I solve my problem? And the right answer is: with a little bit of everything, depending on your environment."

CHRIS WYSOPAL
CTO and Co-Founder, Veracode

3 Interactive Application Security Testing (IAST)

Interactive Application Security Testing (IAST) enables organizations to embed DevSecOps into the pipeline to get fast and accurate security feedback. A single, lightweight IAST agent covers multiple languages to simplify CI/CD tooling and adds only three percent to pipeline timelines. Because vulnerabilities are observed at runtime from within the code, exploitability of a security issue is proven beyond doubt.

STRENGTHS OF IAST

- Offers **fast and accurate results.**
- Veracode Interactive Analysis **simplifies testing by using one, multi-language agent.**
- Although Veracode Interactive Analysis runs locally, **findings and remediation progress can be tracked via Veracode's Analytics.**

CAUTIONS OF IAST

- **Does not scan the entire applications.**
- **Cannot identify every flaw type.**

PRO TIP

Advanced users can **extend functionality by adding vulnerability checks** through patent pending LiveTrack™ technology.

4 Software Composition Analysis (SCA)

With the extreme pressure on developers to get working code delivered quickly, the use of open source libraries has increased. But, thanks to the Heartbleed and Struts-Shock vulnerabilities, many organizations are also now looking for a way to manage and track their library use.

Software composition analysis technologies keep track of which applications are using each library and what versions are being used. With this data, organizations can more easily update libraries to the latest version when new vulnerabilities are discovered.

STRENGTHS OF SCA

- **Identifies vulnerable open source libraries in use.**
- **Is easy to integrate into SDLC.**

CAUTIONS OF SCA

- **Only identifies flaws in open source code.**
- **Cannot locate business logic flaws.**

PRO TIP

Look for a solution that combines SAST and SCA to simplify your life by requiring fewer integration points with your systems and delivering a single report.

5 Penetration Testing

In penetration testing, a security consultant (or pen tester) manually checks an application for security vulnerabilities, typically with no visibility into the inner workings of the application.

STRENGTHS OF PENETRATION TESTING

- **Is a comprehensive method of security testing** because human testers can apply logic and reasoning.
- **Has a very low false-positive rate.**

CAUTIONS OF PENETRATION TESTING

- **The quality is variable** because it depends on environmental factors and on the skill of the human tester, as well as the scope of the project.
- **Reports tend to differ** because even with the same tester and the same scope, the tester may not follow exactly the same path and thought process each time.
- **It's difficult to cover 100 percent of an application's functionality.**
- **Does not scale over large numbers of applications** or across multiple points in the development lifecycle.
- **Is expensive and difficult to schedule.**

PRO TIPS

- **Use automated testing technologies**, such as SAST, SCA and DAST, to find low-hanging fruit before you engage a penetration tester to find the more complex security issues.
- **Find a penetration testing firm that leverages the results of an automated scan** to inform and accelerate the penetration test, so you will get better results in less time (and for less money).

Conclusion

As software increases in importance, and breaches continue to proliferate through the application layer, organizations will need a new approach to security.

An application security program that uses a mix of technologies and services to secure the entire application landscape, and each application throughout its lifecycle, is becoming a necessity. In addition, as application security "shifts left" and moves earlier in the development process, developers will need the right training, coaching and integrated tools to code securely. People, process and technology all need to be addressed to ensure effective application security.

LEARN MORE

Best Practices vs. Practicality: What to Strive for and Where to Start

Veracode is the leading independent AppSec partner for creating secure software, reducing the risk of security breach, and increasing security and development teams' productivity. As a result, companies using Veracode can move their business, and the world, forward. With its combination of process automation, integrations, speed, and responsiveness, Veracode helps companies get accurate and reliable results to focus their efforts on fixing, not just finding, potential vulnerabilities. Veracode serves more than 2,500 customers worldwide across a wide range of industries. The Veracode solution has assessed more than 15 trillion lines of code and helped companies fix more than 51 million security flaws.

Learn more at www.veracode.com, on the Veracode blog, and on Twitter.

VERACODE