STATE OF SOFTWARE SECURITY

Volume 6: Focus on Industry Verticals

JUNE 2015

VERACODE

CONTENTS

Introduction

Veracode has assessed applications for security vulnerabilities on behalf of our customers for over eight years. Our cloud-based platform has analyzed hundreds of thousands of applications and over a trillion lines of code. This has enabled us to amass a great deal of intelligence about the state of software security. This intelligence allows us to diagnose whether an organization is effectively reducing application security risk, especially compared to its peer organizations.

The Veracode *State of Software Security Report* is one way we share this intelligence with the security community. Our report helps CISOs and application security professionals make informed decisions about their application risk. We are often asked by our customers to benchmark their performance. They ask questions such as, "Do I have more serious security vulnerabilities than my peers?" and "What percentage of vulnerabilities do my peers remediate?" In this report we present data that can help you answer those questions for your organization.

One of the unique characteristics of application security is there are no standards defining acceptable security flaw density, or which criticality of defects are acceptable, or what remediation timeframe is adequate. Our data can help you decide if you are doing better than average and hence on the right path or if your application risk is much worse than your peers and in need of a different approach.

The threat space continues to grow in size and sophistication and seemingly no industry is spared. Web application attacks remain one of the most frequent patterns in confirmed breaches and account for up to 35 percent of breaches in some industries according to the 2015 Verizon Data Breach Investigations Report. Yet we still see some organizations only assessing a small percentage of even their Internet-facing applications.

It may be tempting in the face of repeated breaches (OPM, Target and Sony) to throw up one's hands, not to bother building secure applications and to give up on fixing vulnerabilities in the applications you've already deployed. But organizations have not yet begun to seriously address this massive problem. In 2014, according to Gartner, enterprises spent $12 billion securing their network perimeters — but only $600 million securing applications. The data in this report clearly shows that, by addressing the problem systematically and at scale, enterprises can significantly reduce application risk — not by installing more next-generation firewalls, but by remediating application-layer vulnerabilities to reduce enterprise risk.

Security professionals should use the data in this report to understand what level of application risk reduction is possible and what metrics they can strive for to significantly drive down this risk in their organizations.

Chris Wysopal, Veracode CTO and CISO

Executive Summary

Good news: enterprises can reduce application-layer risk via a metrics-driven, policy-based approach.

1.

The problem of reducing application security risk is not intractable; Veracode's customers are actively reducing application-layer risk. Since 2006, Veracode customers have identified 23.3M potential vulnerabilities and fixed 13.7M of them, or about 60 percent. The trend is accelerating; last year, customers found 6.9M and fixed 4.7M vulnerabilities, or almost 70 percent of the vulnerabilities found.

2.

The financial services and manufacturing industries' attention to software security is paying off. In contrast to other sectors, financial services and manufacturing organizations proactively remediate the majority of their vulnerabilities (65 and 81 percent respectively). Based on our knowledge of these organizations, these results are correlated with an emphasis on systematic approaches focusing on centralized policies, KPIs and a culture of continuous improvement.

3.

Government organizations are not sufficiently addressing remediation. Only 27 percent of identified vulnerabilities in government applications get remediated — last among all industry sectors. Plus, government applications have the highest prevalence of SQL Injection, and 3 out of 4 public sector applications fail the OWASP Top 10 when first assessed for risk. Part of the reason for this is that many government agencies still use older programming languages such as ColdFusion, which are known to produce more vulnerabilities.

4.

Healthcare organizations fare poorly. Given the large amount of sensitive data collected by healthcare organizations, it's concerning that 80 percent of healthcare applications exhibit cryptographic issues such as weak algorithms upon initial assessment. In addition, healthcare fares near the bottom of the pack when it comes to addressing remediation, with only 43 percent of known vulnerabilities being remediated.

5.

Significant risk is introduced by the software supply chain. Nearly 3 out of 4 applications produced by third-party software vendors (ISVs) and SaaS suppliers fail the OWASP Top 10 when initially assessed.

6.

Remediation coaching services have a big impact. Lack of in-house expertise is often cited as a barrier to producing more secure code. The data shows that development organizations that leverage external remediation coaching services improve the security of their code by a factor of two and half times compared those that choose to do it on their own. Delivered by world-class security and development experts, Veracode's on-demand advisory services (also known as "readout calls") help developers understand secure coding practices and remediate vulnerabilities more quickly and efficiently.

Overview

The *State of Software Security* is a periodic report that draws on continuously updated analytics from Veracode's cloud-based platform. Unlike a survey, the data comes from actual code-level analysis of billions of lines of code, representing more than 200,000 assessments performed over the past 18 months.

The resulting security intelligence is unique in both the breadth and depth it offers. It represents multiple testing methodologies (binary static analysis, dynamic analysis and manual penetration testing) on the full spectrum of application types (components, shared libraries, web and non-web applications, mobile applications) and programming languages (including Java, C/C++, .NET languages, ColdFusion, PHP, Objective C, COBOL and JavaScript) from every part of the software supply chain (internally developed, open source, outsourced, commercial). For executives, security practitioners and developers who want to understand the root cause of recent breaches, this is essential reading.

This volume captures data collected over the past 18 months from 208,670 application analyses performed via our cloud-based platform (compared to only 22,430 application analyses from a similar 18 month period analyzed in Volume 5 — published in May 2013 — reflecting the rapidly-growing use of Veracode's automated cloud-based service by both enterprises and software vendors). The report looks at differences across industry domains and then looks at remediation trends and practices.

Previous versions of the *State of Software Security* report included extensive analytics about the overall application development landscape, including key metrics such as policy compliance by programming language, flaw density by programming language, and top vulnerability categories by programming language. These are planned to be covered in a subsequent release of the report.

New in this volume is a focus on understanding remediation best practices across industries and across Veracode's customer base. We use two different measures of remediation success, total vulnerabilities fixed and reduced flaw density, to show where Veracode's customers have been successful in reducing application-layer risk.

Spotlight on Industry Performance

When we last looked at how different industries fared in developing secure software, in the 2011 *State of Software Security* Volume 4 report, we focused on software being tested by the government industry sector. This time, we broaden our focus to look at the government sector in context, comparing it to the performance of a total of 34 industries, organized into seven vertical markets. A full listing of all component industries included in the report may be found in the Appendix.

Security of applications by industry vertical

The Veracode platform provides a number of different ways to measure application quality. Here we look at two: compliance with a well-accepted industry standard, and average application flaw density.

In this section, we look at the security quality on initial risk assessment, that is, when the application was first assessed by Veracode. Looking at application scores on initial assessment eliminates any changes in software quality caused by exposure to Veracode's services, and therefore provides a better picture of the quality of the applications before the assessment process begins.

POLICY COMPLIANCE BY INDUSTRY VERTICAL

The *OWASP Top 10* is a list of the most important vulnerability categories in web applications, compiled through community consensus by the security practitioners at the Open Web Application Security Project (OWASP). The OWASP Top 10 is also referenced by industry standards such as PCI-DSS, which sets forth security standards for payment card processing systems. For the purposes of this study, we have defined a policy compliance rule that says that an application must be free of vulnerabilities in the OWASP Top 10 (as found by static analysis, dynamic analysis or manual penetration testing) to pass the OWASP Top 10 policy.

Across our entire data set, we see a low pass rate for the OWASP Top 10 policy (see Security of Applications section below). As might be expected, there is wide variability in this pass rate by industry vertical. Some of this can be explained by language distribution in the industry vertical, but there may be other factors at play.

For instance, the high first-assessment OWASP compliance rate for applications in financial services is higher than can be explained by the disproportionate use of Java or .NET in that industry vertical, given that Java and .NET only have an average pass rate of 24 percent and 27 percent respectively (this data will be presented in a subsequent release of the report that focuses on the application development landscape). We hypothesize that other factors may be at work in financial services applications, such as the impact of regulatory mandates and a bigger focus on continuous improvement processes.

Conversely, the low pass rate (24 percent) in government may be partially explained by the higher use of scripting languages and older languages such as ColdFusion which are known to produce more vulnerabilities, but cannot be entirely ascribed to this. Other factors, such as the lack of regulatory demands that are present in other fields like healthcare, may also contribute to the lower first-pass rate.

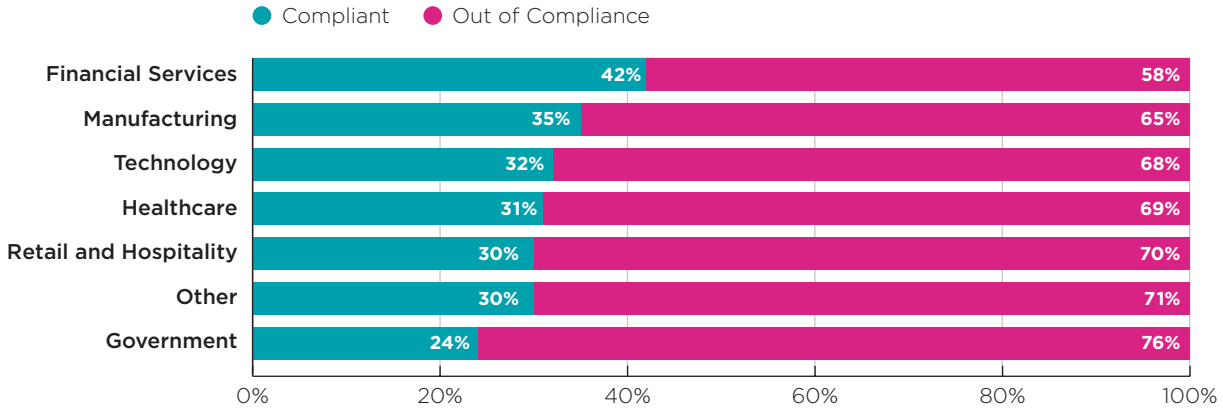


Figure 1: Compliance with OWASP Top 10 Policy on First Risk Assessment, by Industry Vertical

POLICY COMPLIANCE OF COMMERCIAL VS. INTERNALLY-DEVELOPED APPLICATIONS

We are often asked about the relative security of commercial applications produced by third-party software vendors. Since the first volume of the *State of Software Security,* the data has consistently shown that commercial software applications are not significantly more secure than those from other industry sectors, and the data bears this out. Applications from the technology industry vertical, which includes commercial software applications, were only in the middle of the pack with respect to initial software quality.

We also examined the data from a different lens, looking at commercial applications that are submitted on behalf of enterprises and assessed through Veracode's vendor application security testing (VAST) program. This process allows an enterprise to pay for the security assessment of an application that it plans to purchase or has purchased and is deploying. As shown in the chart below, commercial software assessed through Veracode's third-party process had a 9 percent lower OWASP pass rate than internally-developed software. There may be several possible contributing factors to this disparity, including the mix of software languages used for commercial software and the age of the code base.

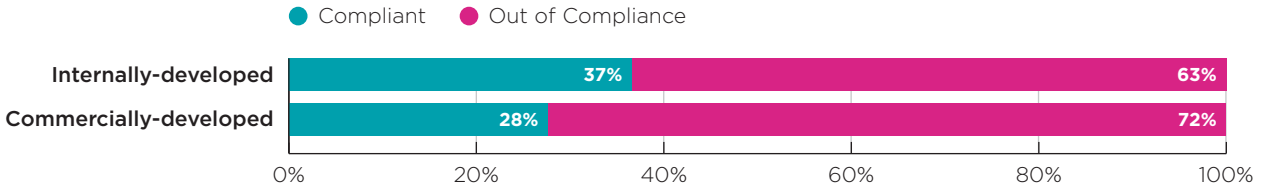


Figure 2: Compliance with OWASP Top 10 Policy on First Risk Assessment, by Commercial vs. Internally-Developed Applications

FLAW DENSITY BY INDUSTRY VERTICAL

Average application flaw density is a measure of average risk per unit of software. It is defined as the number of flaws[1] for an application divided by the size of the application's executable code in megabytes, and has the unit of flaws per megabyte. (Note: In the case of uncompileable scripting languages like JavaScript, PHP or Classic ASP, the density is measured in terms of the size of the source code in megabytes.) Like the similar industry-standard of defects per line of code, flaw density normalizes out application size and allows a side-by-side comparison of application riskiness. Flaw density only includes static assessment flaws, and so this metric was only calculated on the portion of the set that had a static analysis conducted (more than 80 percent of the applications in the set).

We can examine the average flaw density on first assessment by industry vertical. The highest observed average flaw density is in the manufacturing industry vertical, followed by "other" and technology. These results are likely explained by language distribution, as the manufacturing industry vertical has a disproportionately high share of ColdFusion applications, while technology has a disproportionately high share of C/C++ applications.

It is worth noting that flaw density does not necessarily correlate with policy failure. The average flaw density in healthcare is lower than any other industry, but it had a pass rate against the OWASP policy of only 31 percent, lower than financial services or manufacturing.

It is also important to note that flaw density is affected by many factors, most notably by the types and mix of programming languages in use within an organizational group. The information below is therefore only provided for reference purposes and as a way to interpret subsequent analyses of flaw density over time.

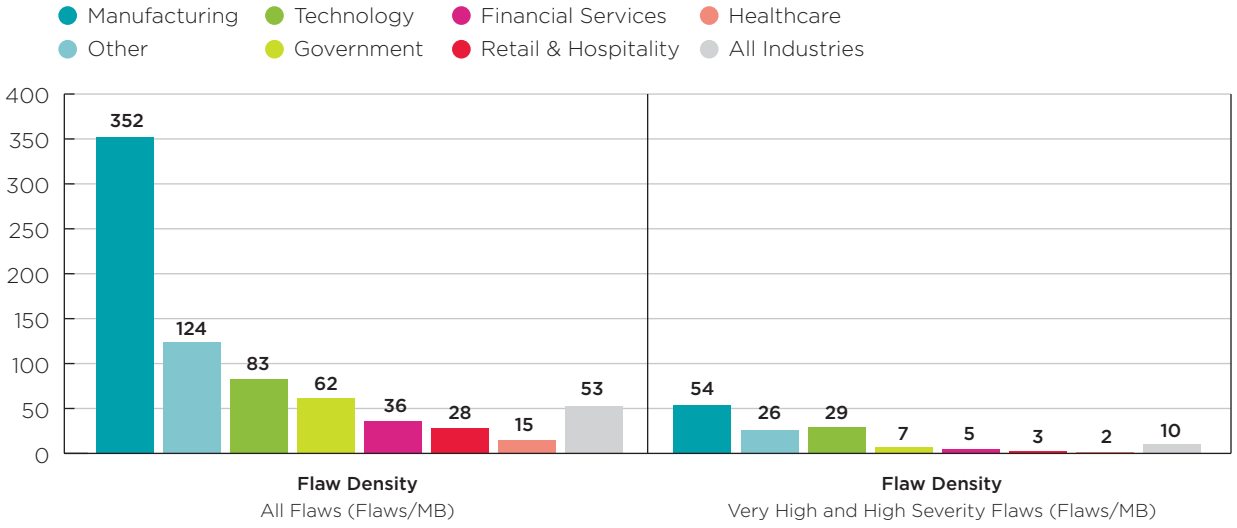


Figure 3: Flaw Density by Industry Vertical

[1] Flaw density is exclusively a measure of flaws found via binary static analysis, which finds potential vulnerabilities caused by flaws in the developer's code. We sometimes refer to these potential vulnerabilities as "flaws" for short.

Remediation by industry vertical — fixed vs. found index

We now turn to remediation progress. We look at a simple measure of risk reduction success — which industries are doing a better job of fixing more of the flaws that they're finding? Here we looked at the number of vulnerabilities fixed within a given industry as a percentage of the total number of vulnerabilities found.

We found that manufacturing and financial services fixed the largest percentage of flaws (81 and 65 percent respectively), compared to government, which fixed just 27 percent of the vulnerabilities identified by Veracode's cloud-based service.

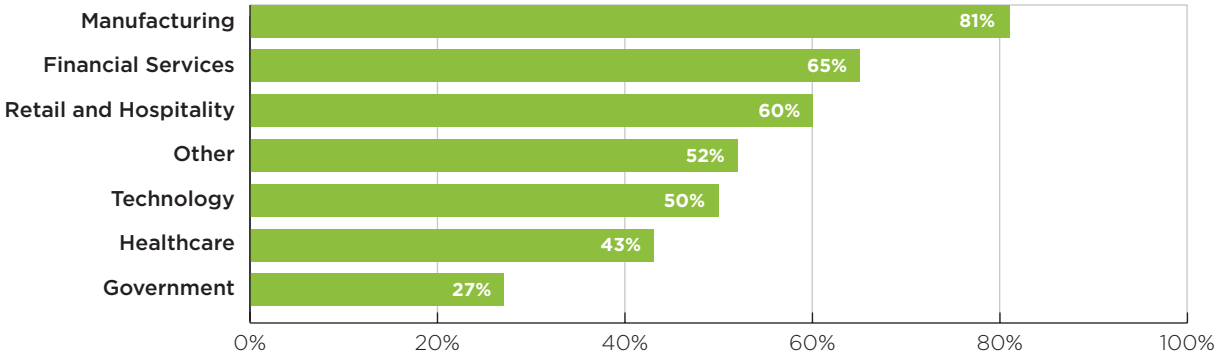


Figure 4: Percent of Flaws Fixed by Industry Vertical

The exceptional performance of the manufacturing vertical is interesting. There are several factors that may be at work here. First, it is important to note that manufacturing, earlier than almost any other industry, has adopted process improvement methodologies as part of the culture of the business, and has also been a leader in implementing supply chain controls for its critical suppliers. As the role of supply chain becomes increasingly digital, we look forward to diving deeper to see which practices manufacturing customers find effective at addressing vulnerabilities in their software supply chains.

Second, it is worth noting that in many cases, differences in security policy can drive significant differences in security program performance. For example, policies that require a large number of vulnerability categories to be fixed seem to discourage developer participation, and paradoxically make the organization less secure. (For a detailed discussion of the effects of policy design, see the November 2012 *State of Software Security* feature supplement, "Enterprise Testing of the Software Supply Chain.") Inspection of the security policies of the companies in this sector may provide additional clues to their success.

Top 10 vulnerability categories by industry vertical

As we have seen in other sections of this report, there are important differences across industries regarding the riskiness of their software. In this section, we look at relative prevalence of key vulnerability categories by industry vertical.

Vulnerability	Financial Services	Government	Healthcare	Manufacturing	Retail & Hospitality	Technology	Other	Rank
Code Quality	65%	70%	80%	56%	68%	70%	65%	1
Cryptographic Issues	60%	66%	61%	51%	63%	62%	59%	2
Information Leakage	58%	62%	60%	49%	55%	62%	53%	3
CRLF Injection	52%	52%	48%	45%	54%	54%	48%	4
Cross-Site Scripting (XSS)	49%	51%	46%	45%	52%	49%	47%	5
Directory Traversal	48%	48%	45%	40%	44%	48%	46%	6
Insufficient Input Validation	41%	45%	43%	33%	44%	37%	37%	7
SQL Injection	29%	40%	32%	31%	25%	30%	34%	8
Credentials Management	25%	20%	26%	24%	24%	28%	32%	9
Time and State	23%	19%	23%	17%	21%	26%	23%	10

Figure 5: Top 10 Vulnerability Categories by Industry Vertical

Prevalence of selected high-profile vulnerabilities by industry vertical

In addition to looking at the overall top 10 vulnerability categories for each industry, we compare the per-industry prevalence of four important categories: SQL Injection, Cross-Site Scripting (XSS), cryptography issues and command injection.

These categories were chosen for their pervasiveness and their severity. For instance, SQL injection was the application vulnerability most often exploited in web application attacks in the 2015 Verizon Data Breach Incident Report,[2] while cross-site scripting ranks in the list of the top vulnerabilities and is far more prevalent overall.

Likewise, OS command injection not only was used in a small percentage of breaches, but more worryingly played a role in 2014's Shellshock vulnerability, in which a vulnerable, commonly used open source component was found to be exploitable in a novel attack that allowed taking over a server to run arbitrary code.

Finally, cryptography issues are highly prevalent across all applications and may be used to allow an attacker to retrieve poorly protected data or hijack communication with an application.

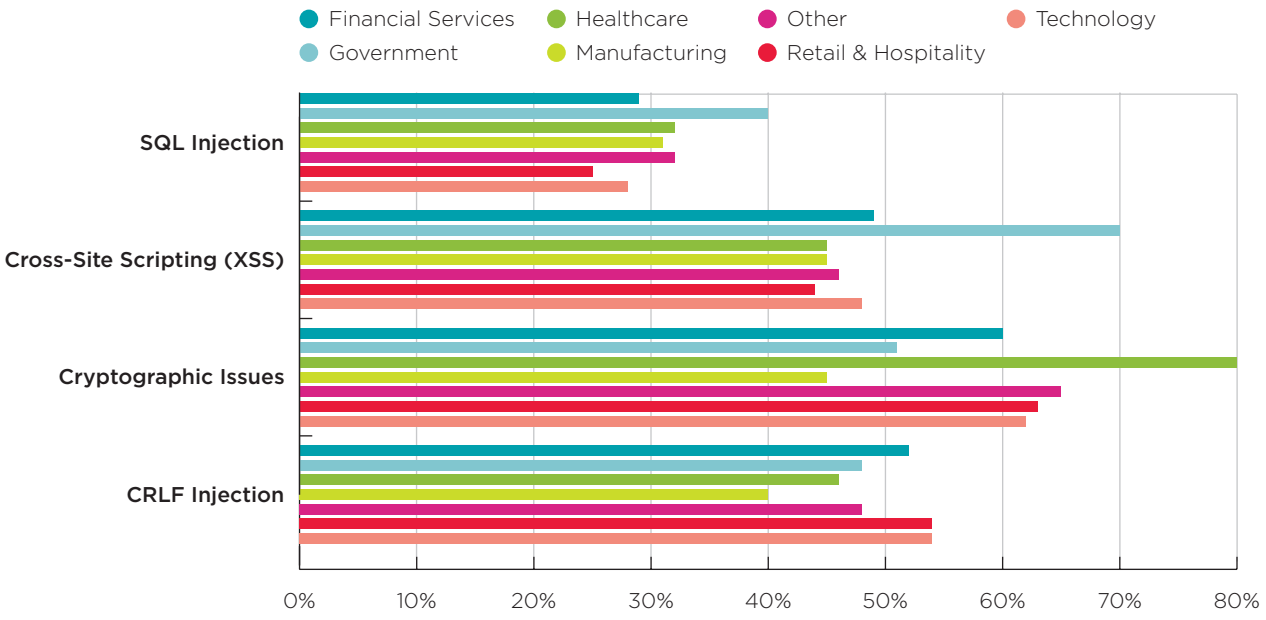


Figure 6: Comparison of High-Profile Vulnerability Prevalence by Industry Vertical

Consistent with their low pass rate for the OWASP Top 10, organizations in the government industry vertical have the highest prevalence of both SQL Injection and Cross-Site Scripting on first assessment, while organizations in retail and hospitality have the lowest. Among other flaw categories, organizations in healthcare have the highest incidence of cryptographic issues — which is concerning given data confidentiality requirements for personal information imposed by HIPAA.

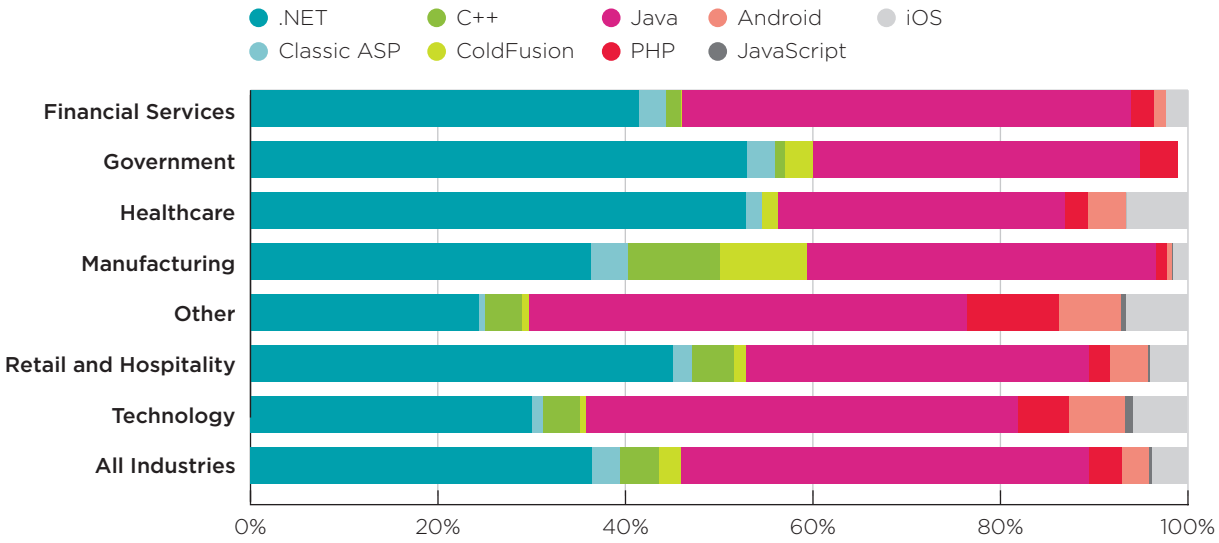
[2] The 2015 Verizon DBIR notes that breach via SQL injection is less common than breach via use of stolen credentials. But how were the credentials stolen in the first place? In the case of the single largest known collection of stolen credentials on the Internet, the 2014 "CyberVar" hack that resulted in the theft of 1.2 billion credentials from various systems, the attackers created a botnet that looked for and exploited SQL injection vulnerabilities across the web. (Source: Nicole Perlroth and David Gelles, "Russian Hackers Amass Over a Billion Internet Passwords," New York Times, August 5, 2014.)

Programming language breakout by industry vertical

Software language choice can have a big effect on the security of applications. Where some languages and programming models completely eliminate some security issues (for instance, buffer management issues common in C/C++ are completely eliminated in Java or .NET), often the choice of programming language is influenced by factors other than security. Availability of skilled developers, programming languages used by suppliers and other points can lead to significant industry differences in programming language choice.

This year's data set bears out this observation. In particular, financial services has a much higher proportional use of Java (48 percent) and .NET (42 percent) and lower use of other languages in the data set. Manufacturing had the highest use of C++ (10 percent), followed by retail and hospitality (5 percent) and technology (4 percent). Healthcare and government consumed a disproportionately high share of .NET applications (53 percent and 52 percent), and manufacturing had the highest use of ColdFusion (9 percent) and Active Server Pages (4 percent).

Lastly, the technology and healthcare industry verticals (along with "other") assessed a proportionally higher share of mobile applications. iOS and Android use were roughly even in most industries, though iOS was slightly higher in healthcare.



	.NET	Classic ASP	C++	ColdFusion	Java	PHP	Android	JavaScript	iOS
Financial Services	42%	3%	2%	<1%	48%	2%	1%	<1%	2%
Government	52%	3%	1%	3%	35%	4%	0%	0%	0%
Healthcare	53%	2%	0%	2%	31%	3%	4%	0%	7%
Manufacturing	36%	4%	10%	9%	37%	1%	<1%	1%	2%
Other	24%	1%	4%	1%	47%	10%	7%	1%	7%
Retail & Hospitality	45%	2%	5%	1%	37%	2%	4%	<1%	4%
Technology	30%	1%	4%	1%	46%	6%	6%	1%	6%
All Industries	37%	3%	4%	2%	44%	4%	3%	<1%	4%

Figure 7: Programming Language Breakout by Industry Vertical

Remediation Analysis

In this section, we examine customer remediation behavior across all applications, rather than considering it by industry segment. Here we seek to establish a few points:

1. What percentage of applications are fixed and reassessed by Veracode customers?
2. How effective are customers at reducing flaw density through the remediation process?
3. What are some factors that help customers fix vulnerabilities?

A few things to note about the data set that are relevant for this section:

1. Because the data set is limited in time, the remediation analysis represents a "point-in-time" assessment. In particular, comparisons of first assessment to final assessment only include applications receiving their first assessment in the six quarters between October 2013 and March 2015.
2. Likewise, the data set ignores any reassessments occurring after March 2015.
3. Lastly, because of the focus on flaw density as a metric of remediation, the analysis ignores testing and remediation based on applications that did not receive a static assessment.

Reassessment statistics as a measure of remediation intent

First we try to evaluate how many applications are reassessed versus being assessed only once. Of the applications assessed statically, about 28 percent of the applications in the sample were assessed once during the time period under study and never reassessed. This may represent two possible scenarios:

- Organization will test again, but did not before the end of the time window of the data set
- Organization only intended to understand the baseline risk of the application and did not attempt to reduce risk by fixing flaws

In the latter use case, there are a number of possible reasons why a customer might not reassess the application:

- Application passed policy on the first assessment — this was true for about 26 percent of the applications that were only assessed once
- Customer no longer has the source code to fix the application and will remediate through some other means, e.g., Web Application Firewall (WAF), or retire the application
- Customer was evaluating the application as part of a merger and acquisition process

The takeaway from this part of the analysis is that the overwhelming majority of applications are assessed more than once, presumably to verify fixes to software vulnerabilities.

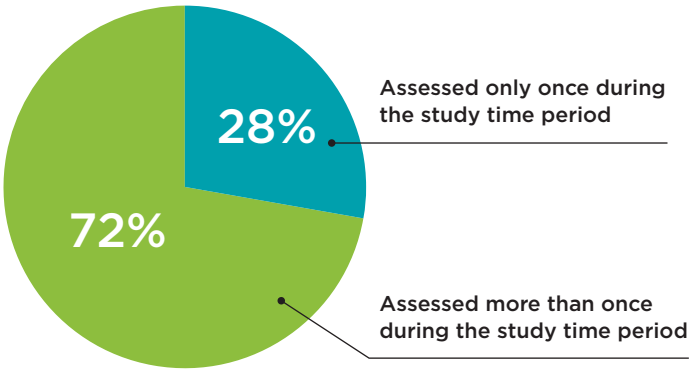


Figure 8: Distribution of Application Scan Frequency During Study Time Period

Reduction in flaw density — first assessment vs. reassessment

In looking at how flaw density changes on average from the first assessment of an application to the most recent assessment, we can use flaw density as a way to understand the improvement in application quality.

Applications that are assessed only one time have, on average, a slightly higher flaw density than applications that are assessed and then reassessed. Over time, customers reduce flaw density for all flaws by an average of 13 percent, but reduce flaw density for high and very high severity flaws by 58 percent. This suggests that customers prioritize remediation of high and very high severity flaws over fixing other flaws, and may suggest that their security policies focus exclusively on high and very high severity flaws.

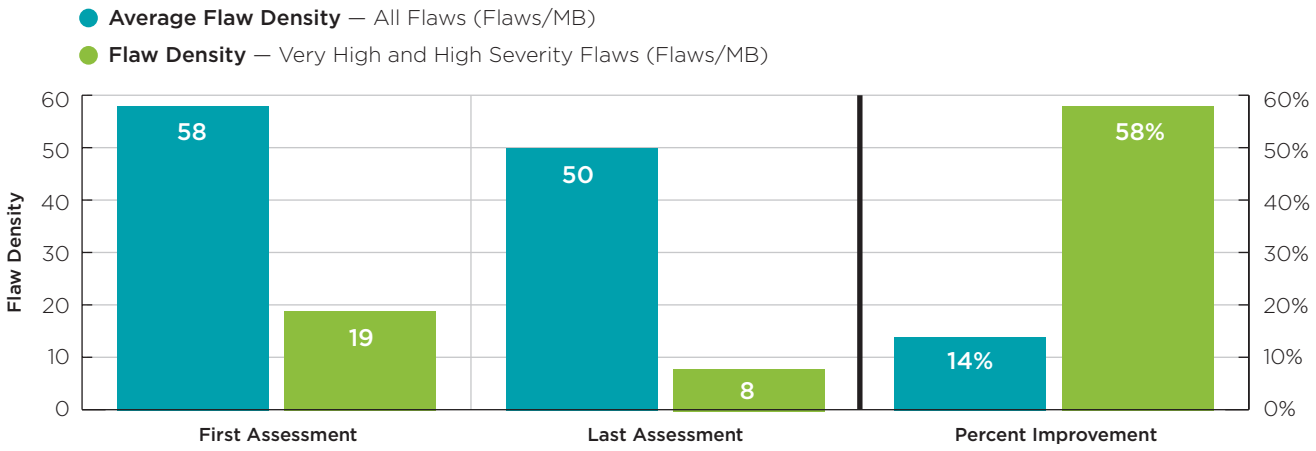


Figure 9: Reduction in Flaw Density, First Assessment vs. Reassessment

Reduction in flaw density via remediation coaching services

Lastly, we turn our attention to a factor that may help organizations fix more vulnerabilities. This section focuses on applications that are statically assessed more than once and looks at remediation coaching, a factor that affects flaw density between the first assessment and the most recent assessment.

Remediation coaching is a standard part of Veracode's cloud-based service that grants developers on-demand access to Veracode application security consultants as part of a process called a readout. During a readout, Veracode application security consultants do the following with the application development team:

- Explain how the testing was performed, including types of tests and the scope of the tests
- Review the findings, including an overview of Veracode platform features for performing triage and documenting mitigations
- Answer questions
- Discuss next steps, including remediation and mitigation plan

Customers can initiate the readout via a request through the Veracode platform or via an out-of-band email request. In cases where a readout is requested through the platform, we are able to see that a readout took place for the application and use that in anonymized data analysis.

The impact of remediation coaching on flaw density is significant, as shown in the data below. Applications remediated without a readout on average achieved 17 percent flaw density reduction, compared to a 42 percent flaw density reduction in applications that did undergo a readout — a 2.5x improvement.

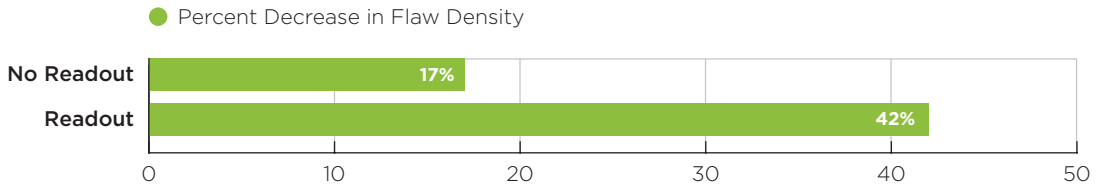


Figure 10: Relative Improvement in Flaw Density via Remediation Coaching (Readout)

Appendix

About the dataset

The data represents 208,670 application assessments submitted for analysis during the 18-month period from October 1, 2013 through March 31, 2015 by large and small companies, commercial software suppliers, open source projects and software outsourcers. In most analyses, an application was counted only once, even if it was submitted multiple times as vulnerabilities were remediated and new versions uploaded. The report contains findings about applications that were subjected to static analysis, dynamic analysis or manual penetration testing through Veracode's cloud-based platform. The report considers data that was provided by Veracode's customers (application portfolio information such as assurance level, industry, application origin) and information that was calculated or derived in the course of Veracode's analysis (application size, application compiler and platform, types of vulnerabilities, Veracode Level (predefined security policies which are based on the NIST definitions of assurance levels)).

INDUSTRY VERTICALS

This report condenses information about applications coming from 34 different industry classifications into seven industry verticals. The component industry classifications come from Data.com via Salesforce.com, but Veracode has created the industry verticals below to simplify the analysis. A mapping of the component industries to industry verticals is provided below.

	Component Industries as Defined in Data.com
Financial Services	Banking, Finance, Insurance
Manufacturing	Manufacturing, Aerospace
Technology	Technology, Telecommunications, Electronics, Software, Security Products and Services, Consulting
Retail & Hospitality	Retail, Hospitality
Government	Government
Healthcare	Healthcare
Other	Other, Biotechnology, Education, Entertainment, Transportation, Not for Profit, Apparel, Communications, Engineering, Media, Media & Entertainment, Food & Beverage, Utilities, Energy, Machinery, Construction, Chemicals, Not Specified, Shipping, Business Services

SAMPLE SIZE

In any study of this size, there is a risk that sampling issues will arise because of the nature of the way the data was collected. For instance, all the applications in this study came from organizations that were motivated enough about application security to engage Veracode for an independent assessment of software risk. We have taken care to only present comparisons where a statistically significant sample size was present.

ABOUT THE FINDINGS

Unless otherwise stated, all comparisons are made on the basis of the count of unique application builds submitted and rated.

VERACODE

The Most Powerful Application Security Platform on the Planet

Veracode is a leader in securing web, mobile, and third-party applications for the world's largest global enterprises. By enabling organizations to rapidly identify and remediate application-layer threats before cyberattackers can exploit them, Veracode helps enterprises speed their innovations to market—without compromising security.

Veracode's powerful cloud-based platform, deep security expertise, and systematic, policy-based approach provide enterprises with a simpler and more scalable way to reduce application-layer risk across their global software infrastructures.

Veracode serves hundreds of customers across a wide range of industries, including nearly one-third of the Fortune 100, three of the top four U.S. commercial banks, and more than 20 of Forbes' 100 Most Valuable Brands.

LEARN MORE AT WWW.VERACODE.COM, ON THE VERACODE BLOG, AND ON TWITTER.