Accelerating Software Development with Secure Open Source Software

BY VERACODE

VERACODE

ACCELERATING SOFTWARE DEVELOPMENT WITH SECURE OPEN SOURCE SOFTWARE

FOCUS ON THE 10% THAT DIFFERENTIATES YOUR BUSINESS

The world of software development has drastically shifted in the past several years, as the demand for software that better solves your customers' needs increases. With every company becoming a software company, and competition in markets becoming fiercer than ever, companies are required to move quickly or get left behind. Thus, engineering groups had to figure out a way to spend most of their time focused on the features that differentiate themselves in the market, and less on the features that are table stakes.

Take user login systems for example, every single SaaS based product has a user account management and login screen, and I've never heard of a customer buying from a vendor because their login screen was better than the competition. This example of a tables takes feature is something that you don't want your company spending a lot of time developing. However, user authentication is absolutely critical to the security of your customer's data, so developers can just do it quickly and move on to other things. That's where Open Source libraries come in to play, giving developers the way to implement table stakes features that actually work.

By adding functionality from open source libraries into their code, developers can spend less time on the things that do not differentiate themselves, and more time on the features that do. In fact, we're seeing that many applications are comprised of 90-95% of open source code! That means, your company needs its developers time spent on that 10%.

What is the 10%?
It's the part of your application that differentiates you from the rest of the market

And what is the 10%? It's the part of your application that differentiates you from the rest of the market. The companies that are able to spend the majority of their time on those pieces, are the ones getting ahead of the competition. They are able to release new features faster than anyone else, because they are spending less time on the trivial pieces of the code.

Explosive growth of open source libraries

The use of open source libraries to assemble applications is accelerating. Not only are more people using open source libraries, but more individual developers, and even companies, are also on a mission to contribute to more open source projects. For Veracode, we're seeing more than 70 percent of our customer base leveraging one or more open source libraries in their applications. And when customers use open source libraries, they use a lot of them. On average, applications are leveraging 49 direct open source libraries – this does not count the indirect libraries, which can be in the hundreds for a single application.

Risk of open source libraries

So what about the risks present in those applications? On average, applications contain seven open source vulnerabilities per application. While this doesn't seem like a big number, it only takes one to compromise an entire application. And of those applications

that contain vulnerabilities, 44 percent of them contain critical vulnerabilities. So what are companies doing to protect themselves? Well, not always enough, as only about one-third of organizations do regular open source vulnerability scanning.

In the case of the Equifax breach, one vulnerability in an open source Struts2 component had huge impacts, including the resignation of the CEO.

So while the use of open source libraries is proliferating, so are the risks. With the reusable code and functionality also comes reusable vulnerabilities, and with very popular libraries used in thousands of applications, a single vulnerability, in a single piece of code, can suddenly makes thousands of applications vulnerable to the same exploit. If we look at Java applications alone, we're seeing 88 percent of Java applications are leveraging one or more libraries containing vulnerabilities.

There is also a pervasive false sense of security around open source libraries, and a general lack of understanding about the attack vectors that open source libraries can open up. While it is true that there are more "eyes on the code" when it comes to open source libraries, there is nobody ensuring that these security fixes are being disclosed, and, of course, nobody to inform you that there was a vulnerability patched. Ultimately, contributing developers are interested in the success of an open source library, not in the success of your business. It's up to the company implementing the open source library to ensure that it is safe to use, and that they are staying up to date on any known vulnerabilities in the library and their fixes.

> Simply **using** open source libraries is **not a security threat to the business.** The real problem is **not knowing that what you are using contains vulnerabilities,** and that they are exploitable in your application.

CHALLENGES IN SECURING OPEN SOURCE

When assessing the posture of your open source risk, you have to ask three key questions:

- Which libraries are we using, and do they contain any vulnerabilities?
- Can I react fast enough to new vulnerabilities that arise?
- Can we implement a program that scales with the needs of our business?

The unknown unknowns – does my code contain any vulnerabilities?

With 70 percent of our customers leveraging open source libraries, and software being comprised of up to 90 percent open source code, the amount of open source code in use is proliferating rapidly. In addition, so are the problems that accompany that code. Development and security leaders alike are asking the question, "Does my code contain any vulnerabilities?" There is a need to both better understand the composition of software, and the vulnerabilities that are present.

There is another fundamental issue that can be broken into two parts:

First, today the most used source of vulnerability data is the National Vulnerability Database (NVD), which contains tens of thousands of vulnerabilities across all different types of applications, including open source libraries. The NVD, unfortunately, is completely overrun with vulnerability submissions. (You can view their dashboard at any time[1].) At the time of this paper, the NVD is receiving about 200 new or modified vulnerabilities each week.

Second, everything submitted to the NVD has to be known and disclosed, but what about those vulnerabilities that few people know about because they haven't been broadly disclosed? Oftentimes, when a vulnerability is disclosed and assigned in the NVD, it has been patched for several months. For example, in August 2018, a new Remote Code Execution vulnerability for the Apache Struts library was disclosed and assigned in the NVD, however this vulnerability was actually found and patched back in April 2018. If anyone had been watching the commit logs of the library, they would have been aware of this potential vulnerability for months before the public was made aware of it. Malicious actors are targeting these libraries.

Exploit attack window is shortening – can I respond fast enough?

Developers not only have to release software faster than ever before, but they also have to respond to security threats faster than ever before. They also must have a tool that helps them maintain their development velocity.

Time between vulnerability disclosure and exploitation is counted in days, sometimes hours. As in many areas of IT security, the attacker is at an advantage.

Attackers are known to look through the commit history of open source libraries to find code changes that remediate a vulnerability. Oftentimes, developers make these changes without publicly disclosing the vulnerability to the NVD. This means open source users are not alerted of security issues nor urged to update their versions. Meanwhile, attackers can develop exploits based on the vulnerability and often start attacking applications with a "spray and pray" approach across the entire Internet. Once they successfully exploit a machine, they can access data or get a session pivot to other parts of the organization's network.

On the flip side, defenders need to be aware of not just this one vulnerability, but of all of the vulnerabilities in their open source code, with the added complexity that they are not all listed in the NVD. In addition, they need to work with the development teams to fix all vulnerabilities that put the organization at risk. Without the right solutions, this is a near impossible task.

Real-life example of the gap between discovery and disclosure

Let's have a look at a real-life example: In August 2018, a public announcement was made of a remote code execution vulnerability in the Apache Struts open source library. However, this vulnerability was actually privately disclosed in April 2018, and finally in June 2018 the vulnerability was patched in the library – two months before the public disclosure. If attackers were monitoring that change, they would have seen the commit, explored if it

[1] https://nvd.nist.gov/general/nvd-dashboard

was a security fix, and then focused their attention on finding applications with out-of-date libraries – thus giving them a place to narrow their focus and efforts.

Now let's look at a different example. The Equifax hack was the result of not patching a publicly disclosed vulnerability, in this case CVE-2017-5638 in the Apache Struts framework. While patching disclosed vulnerabilities is the first step you take to address the problem, it was merely days between the vulnerability being disclosed, and the vulnerability being exploited in an unpatched system. The larger your organization, the harder it can be to patch these systems in time, so companies need a leg up to get ahead of these disclosures. Unfortunately, as we've shown, you cannot rely solely on the CVE system and the National Vulnerability Database for open source security vulnerabilities alone.

Security and development teams need a way to level that playing field, stay ahead of vulnerabilities, and prioritize those that actually affect their code. Developers also need something that provides automated and instant feedback, directly in their CI system – where and how they expect to work. This is the only way they can remediate issues in the moment, and maintain their velocity.

Taking a programmatic approach: scaling with your business

There are many different security tools on the market, however, they are just that – tools. And to the extremely well trained and experienced person, a tool is a wonderful answer. However, in many cases a tool gets the job started until the company realizes that they are either not using the tool properly or that nobody is really leveraging the tools they purchased. We see this problem all the time with customers who come to Veracode from other vendors that only provided them with a tool and not a program. A tool is usually inexpensive, but it relies on the person using it to understand all of the nuances and how to take that tool and turn it into an entire program that is adhered to by the rest of the organization. **This is a really difficult thing to do**.

We encourage anyone looking to purchase any type of security solution (not just AppSec), to not skip out on the services and program management side of security. Security is critical and if it's not being implemented properly, your company may not be any better off than when they started. In fact, implementing security controls incorrectly could lead to a disastrous reduction in the efficiency and production of your entire organization. And when it comes to AppSec, we often hear that the largest complaints come from developers who had to experience AppSec tools implemented without a proper program in place.

Application Security programs should fit to the way your organization develops software, not the other way around. If you don't provide solutions that help your developers produce secure software faster, you will never get the adoption nor support from your engineers. Without development teams on board with AppSec, you will never make any real true progress.

VERACODE

HOW VERACODE CAN HELP

It's so critical to the success of your organization to ship new and differentiating features to the market as fast as possible. The best way to do that is ensure developers are spending most of their times building these differentiating features, and not on recreating table stakes functionality. What's even more critical is the ability to use secure open source libraries so that developers do not have to go back later to fix vulnerabilities after applications make it to market. Simply using open source is not a security threat to the business; it's not knowing that what you are using contains vulnerabilities and that they are exploitable in your application that is the real problem.

The problem we solve with Software Composition Analysis (SCA)

Veracode helps security and development teams answer three fundamental questions: Does my code contain vulnerabilities, do they actually impact my application, and can I react fast enough to new threats?

What are we using and is it safe?	• Comprehensive database of both known vulnerabilities and undisclosed vulnerabilities • Bill of materials for all of your open source libraries • Detailed information on the individual vulnerability and its potential impact
Can I react to new vulnerability quickly?	• Integrated into your CI, automating open source vulnerability scanning • Proprietary Vulnerability Database, leverages data mining and machine learning, to get ahead of vulnerabilities before they are disclosed • Alerting security to new vulnerabilities in your applications without requiring a rescan
How do I scale an actual AppSec Program?	• SaaS based Application Security provider, scales directly with the needs of your business • Years of experience in programmatic approaches to introducing, implementing, and succeeding with scalable AppSec programs • Program managers and application security consultants that can help Security manage the program and give Developers one-on-one coaching to remediate vulnerabilities

VERACODE

How you benefit from Veracode's solution

AppSec programs are only as good as the rate that they are adopted and adhered to. In order to have a successful program, you must balance the needs of the business, the requirements of security, and the desires of the developers. Your company is looking to bring secure software to market as fast as possible, and they're relying on your developers to accomplish this.

Veracode's Software Composition Analysis solution provides you with:

- **Better developer adoption:** Veracode's solutions are built with the Developer in mind. We don't just help you find the issues, we help you remediate them as well. Whether it's our self-serve online resources or our personalized one-on-one consultation calls with your developers to review their code with them, you'll remediate faster with Veracode.
- **Better inventory and awareness of open source risk:** Because of our extensive database of not only known vulnerabilities, but also unknown/undisclosed ones – you can be assured that you get the full view of your risk posture. With a full bill of materials covering your libraries, their versions, any vulnerabilities, and licenses for them, you can feel confident in knowing exactly what is being used.
- **The ability to scale as your company grows:** We're a SaaS based company, which means you don't have any expensive on-premises equipment to maintain. Our cloud scales with the needs of your business, and as you onboard more developers and develop new applications, Veracode has you covered.

But how do you know that your program is working, and that it's successful? You will want to track a few of these metrics as you roll out your program:

- **Reduction in total number of vulnerabilities in your application:** By eliminating vulnerabilities early and often, you will be able to reduce the overall number of vulnerabilities in your applications delivered to production.
- **Number of times code is scanned:** We have evidence that customers who scan more frequently have far fewer vulnerabilities in their applications, and remediate those vulnerabilities at a much faster rate.
- **Number of applications and teams leveraging open source vulnerability scanning:** Your adoption rate is an important measure, and indicator of the future success of your program. Full coverage is critical to ensuring you are releasing secure applications to market, and protecting your customers' data.
- **Time to remediation:** Knowing how fast your developers are remediating their vulnerabilities is an important stat, since it's a very clear indicator of how security is impacting overall developer velocity.
- **Time to approve:** Without technology scanning your code every time it is touched, your security team has to go through extensive work vetting each individual library for developers to use. Reducing this to zero and having the scan occur automatically will accelerate software development.

VERACODE

VERACODE DETAILS

Our teams have worked hard to develop a Software Composition Analysis (SCA) solution that works for organizations attempting to better secure applications against open source risk without reducing the development velocity of the business. In April 2018, Veracode acquired SourceClear, an SCA company, in order to rapidly accelerate roadmap requirement and add industry leading features to the product. Through early 2019 we will be working hard to integrate SourceClear into our existing SCA product, to provide a single unified solution to the market. For information about the languages, technology, and support that we will be adding to Veracode SCA, please review the technical whitepaper for SourceClear at: https://info.veracode.com/whitepaper-solving-your-open-source-risk-with-sourceclear.html

How it Works

Today Veracode offers Software Composition Analysis (SCA) alongside our Static Analysis product. Customers upload their applications to our platform, where we initiate a Static Analysis scan. Part of the pre-process for that scan is to understand the composition of the application, which is where SCA comes into play. We return the SCA results immediately, before the Static scan continues. Customers are able to cover their 1st and 3rd party code with a single upload and a single scan.

Support

Veracode supports the following languages:

Languages
Java, JavaScript, .NET

You can always find the latest up-to-date supported on our help center:
https://help.veracode.com/reader/V_KSmNRUn6rtPwEJ1NXBmQ/b1Qb8qNYlKadYOnAqZkilw

We support a number of systems to initiate your open source vulnerability scans:

CI & Build Systems
Jenkins, Azure DevOps, TFS, TeamCity, Bamboo, Ant, Maven

For customers without a CI system, you can leverage our manual upload and scan process

Manual Upload
SaaS Platform for manual upload and scanning

We support many of the most popular open source licenses – we are adding new licenses all the time, please contact an account or program manager with specific license types and versions to verify if we support it, as the list would be too exhaustive to include here.

Licenses Supported
AAL, AFL, AGPL, Apache, APOL, APSL, Artistic, BSD, BSL, CATOSL, CC, CCDL, CECILL, CNRI, CPAL, CPL, CUA-OPL, ECL, EFL, Entessa, EPL, EUDatagrid, EUPL, Fair, Frameworx, GPL, HPND, Intel, IPA, IPL, ISC, JSON, LGPL, LiLiQ, LPL, MirOS, MIT, Motosoto, MPL, MS-EULA, MS-L, MS-PL, MS-RL, Multics, NASA, Naumen, NCSA, NGPL, Nokia, NPOSL, NTP, OCLC, ODbL, OFL, OGTSL, OSET, OSL, PDDL, PHP, PostgreSQL, Python, QPL, RPL, RPSL, RSCPL, SimPL, SISSL, Sleepycat, Spice, SPL, Unlicense, UPL, VSL, W3C, Watcom, Xnet, Zlib, ZPL

NOBODY DOES IT QUITE LIKE US

While we help our customers protect themselves from open source risks, there are a number of vendors that do the same basic discovery and reporting. However, our focus is on coverage and scalable solutions that help development teams maintain velocity while ensuring more secure applications are delivered to production. The following are the technology aspects that set us apart from the rest of the competition.

NVD vs Our Proprietary Database

Veracode has begun leveraging the proprietary database, from the SourceClear acquisition, as a data source for applications that are uploaded and scanned, one of the first parts of our integration process

We mentioned earlier that the NVD, although very robust, is not able to keep up with the sheer volume of vulnerabilities being disclosed and/or updated daily. One of the problems is that the NVD is for all software vulnerabilities disclosed, not just open source vulnerabilities. Thus, these open source library vulnerabilities get stuck in a log jam behind everything else. The NVD has a pretty slick dashboard on their website showing how many vulnerabilities are coming in every day, week, and month. This screenshot, taken at the time of writing this paper, shows that there were 247 CVEs processed in that week alone.

NVD Dashboard

CVEs Received and Processed

Time Period	New CVEs Received by NVD	New CVEs Analyzed by NVD	Modified CVEs Received by NVD	Modified CVEs Re-analyzed by NVD
Today	0	7	66	0
This Week	247	249	209	28
This Month	851	990	582	33
Last Month	1143	1546	1275	23
This Year	12665	10910	10407	156

However, there is one more issue not being addressed. The only way that a vulnerability makes it into the database is if a software developer or an independent security researcher submits it to the NVD. It is not uncommon for vulnerabilities to be fixed, but never disclosed or submitted to the NVD. We mentioned this example earlier, but as a reminder, the Apache Struts Remote Code Execution vulnerability that was disclosed to the public in August 2018 was actually patched back in April. That means for four months, anyone not on the latest version of the library was potentially vulnerable to this exploit. This, by the way, was the same type of vulnerability that led to the Equifax breach the year before.

To combat this, we have developed our own database that includes all of the open source vulnerabilities in the NVD, as well as our own list of vulnerabilities in open source libraries that have not yet been disclosed to the NVD. In many cases, the vulnerabilities we find and record have either not been disclosed yet and are in the time

between patching and full public disclosure, or in some cases, there was never any intent to disclose the vulnerability and its fix. There is a third category we track, which are "Reserved CVEs." We take the Reserved CVE IDs from the NVD and then find the vulnerabilities in the public repos, in order to give you a head start on the fix prior to full public disclosure.

So how do we discover these vulnerabilities that are out in the wild and are either unknown or undisclosed? That's where our data mining, proprietary machine learning, and security research process come into play.

License Database

In addition to tracking security vulnerabilities, which is the main focus of our technology, we also offer the ability to identify the most common sets of open source licenses that can pose both business and financial risk to organizations.

We are able to help companies identify open source licenses like Apache, MIT, MPL, BSD, CCDL, and GPL to name a small few. We are working to identify and add more license types that our customers are looking to identify in their code.

Complete Bill of Materials

The first step to getting any type of control over your open source environment, is to understand all of the different libraries and their versions that are currently being used across your applications. Veracode provides security and development teams with a complete bill of materials of every library being used by your application, and allows you to review this list in aggregate at the portfolio level or by each individual application.

SaaS Based Solution

When it comes to scaling your AppSec needs, the easiest way to do that is with a SaaS based vendor. Our cloud based AppSec solution ensures that you always have access to scan your applications, and the results, no matter where you are. Your organization does not have to worry about costly on-premises equipment, redundancies, nor backups.

1st & 3rd Party Code Coverage

The greatest strength of Veracode is the ability to cover your entire Software Development Lifecycle (SDLC) from end to end. From scanning the very first few lines of code added by a developer with Greenlight, to scanning a fleet of applications deployed in product with Dynamic Analysis, Veracode provides you the coverage you need. So while it's critical to ensure you are aware of the vulnerabilities in any open source libraries being used, it's equally important to scan your 1st party code for any critical security flaws.

VERACODE

VERACODE EXPERIENCE OVERVIEW

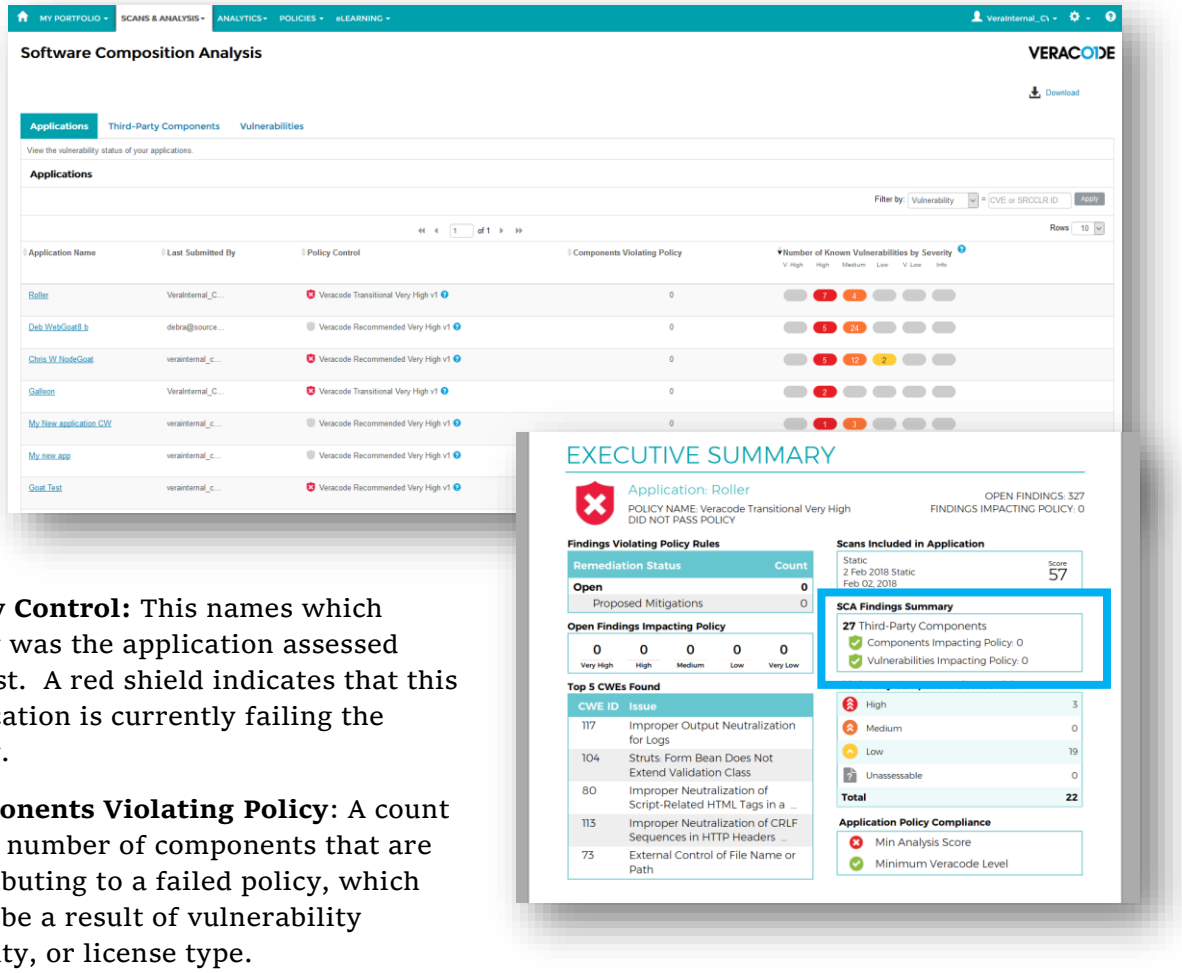
The Veracode platform was built as a single pane of glass for all of your AppSec needs. As an end-to-end Application Security provider, it was critical that security and development alike had a single place to go to, in order to get all of their results for their different scan types. As such, some of what you will see here may encompass more than just SCA scan results and functionality.

Reporting

There are two primary ways to look at results of an SCA scan, actively in the platform, or passively from a downloaded PDF report. Either way, the reports include the ability to view the following pieces of information:

Application: This is the name of your application profile, which serves as the housing place for the application that is uploaded to the platform, the scan results, and all of the users and teams associated with that application.

Last Scheduled: Our platform always contains the date of when the scan was performed, so that security and developers know how relevant the results are.



Policy Control: This names which policy was the application assessed against. A red shield indicates that this application is currently failing the policy.

Components Violating Policy: A count of the number of components that are contributing to a failed policy, which could be a result of vulnerability severity, or license type.

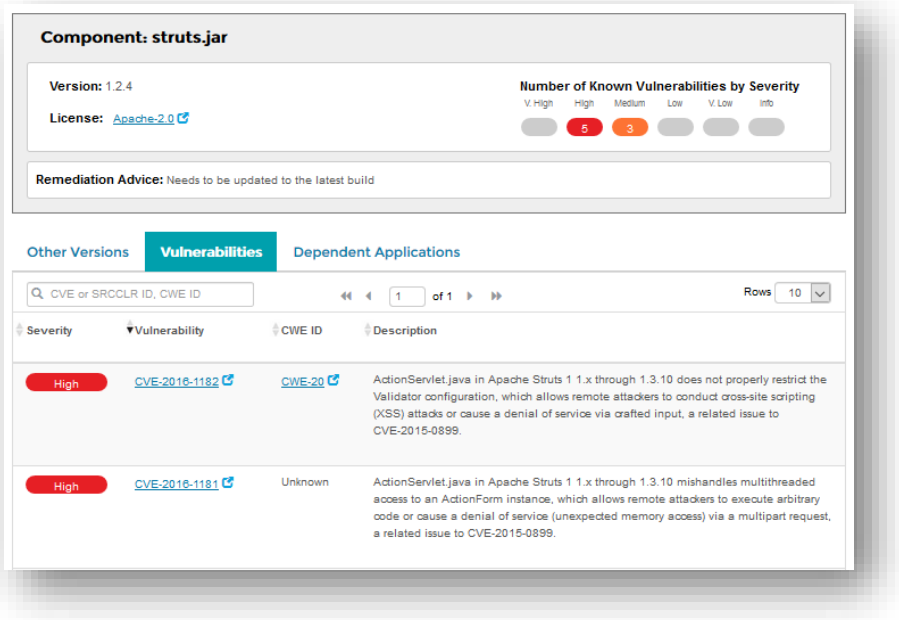
Number of known vulnerabilities by severity: For each application at the portfolio view, we maintain a snapshot view of how many vulnerabilities are in each application and what severity level the vulnerabilities are.

Component Name: We also track the name of the open source library being used in the application, which application is using it, and what the version is of that library.

Other Library Versions: The platform can surface a list of all versions of a library, to allow developers to see ahead of time which version, if they were to upgrade to it, would contain no known vulnerabilities.

You are able to view the results of each scan through a number of lenses, such as the aggregate portfolio level, the individual application level, the vulnerability level, or the component level.

Vulnerability Details

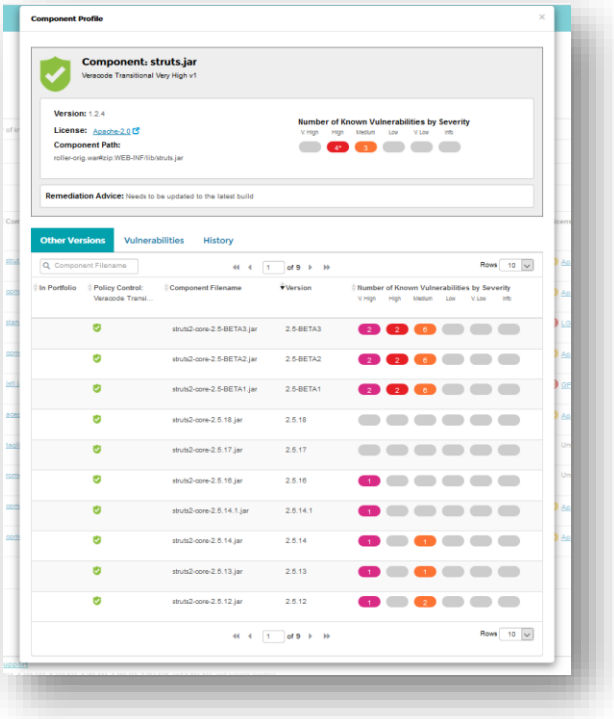


We provide a lot of information on the health of an individual library being used within an application. At a high level we list the name, version, license type, and number of vulnerabilities by type in the library. Below that, we list out each vulnerability, its severity level, a link to the CVE in the National Vulnerability Database, a CWE associated with the vulnerability if present, and a short description on what the vulnerability does. From here, developers can make informed decisions on what is critical for their applications to fix. We also provide a tab showing "Other Versions" that a developer can update too, and the vulnerability status of every version, as well as a tab showing which applications are dependent on this library.

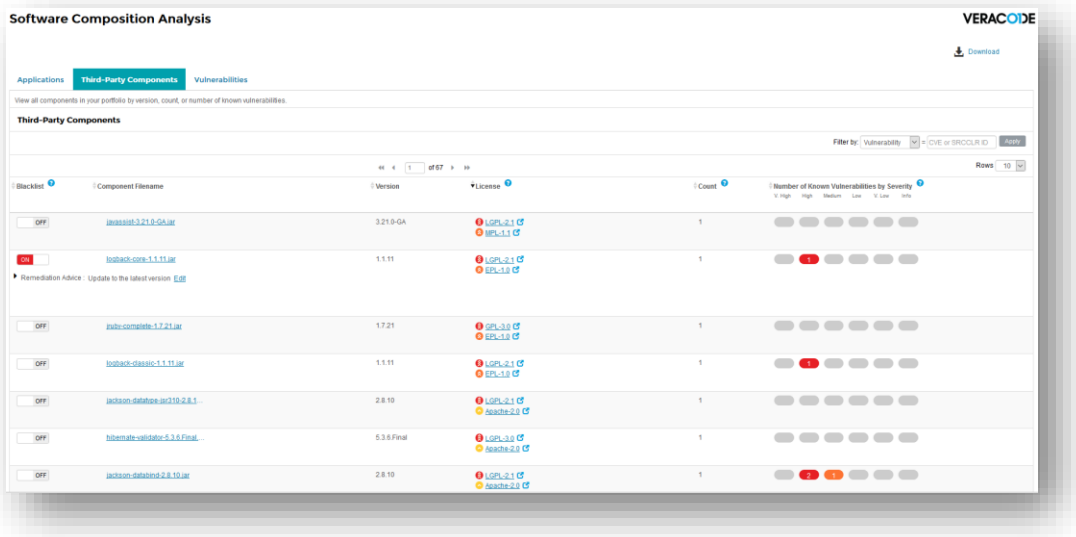
VERACODE

Library Details

As mentioned, we also provide a list of all versions of a library and the vulnerability status of each library. This list is important so that a developer can make an informed decision of which library to update their application to. In many cases, a developer does not want to just update to the next version or even the latest version, as those may contain other vulnerabilities. Instead, they typically want to update to the next safest version, so that there are the fewest disruptions to their application when upgrading modules that may have changed functionality, and the fewest new vulnerabilities.



Bill of Materials

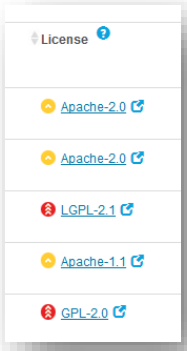


One of the great aspects of our platform is the ability to view your libraries and their vulnerabilities from a number of different lenses. Customers are able to view an entire Bill of Materials for all of their libraries in an aggregate format for all of their applications. There, the license types can be viewed, the versions, and the vulnerabilities present. Customers can drill into each one to see the details page including list of all possible versions, all vulnerabilities, and dependent applications.

VERACODE

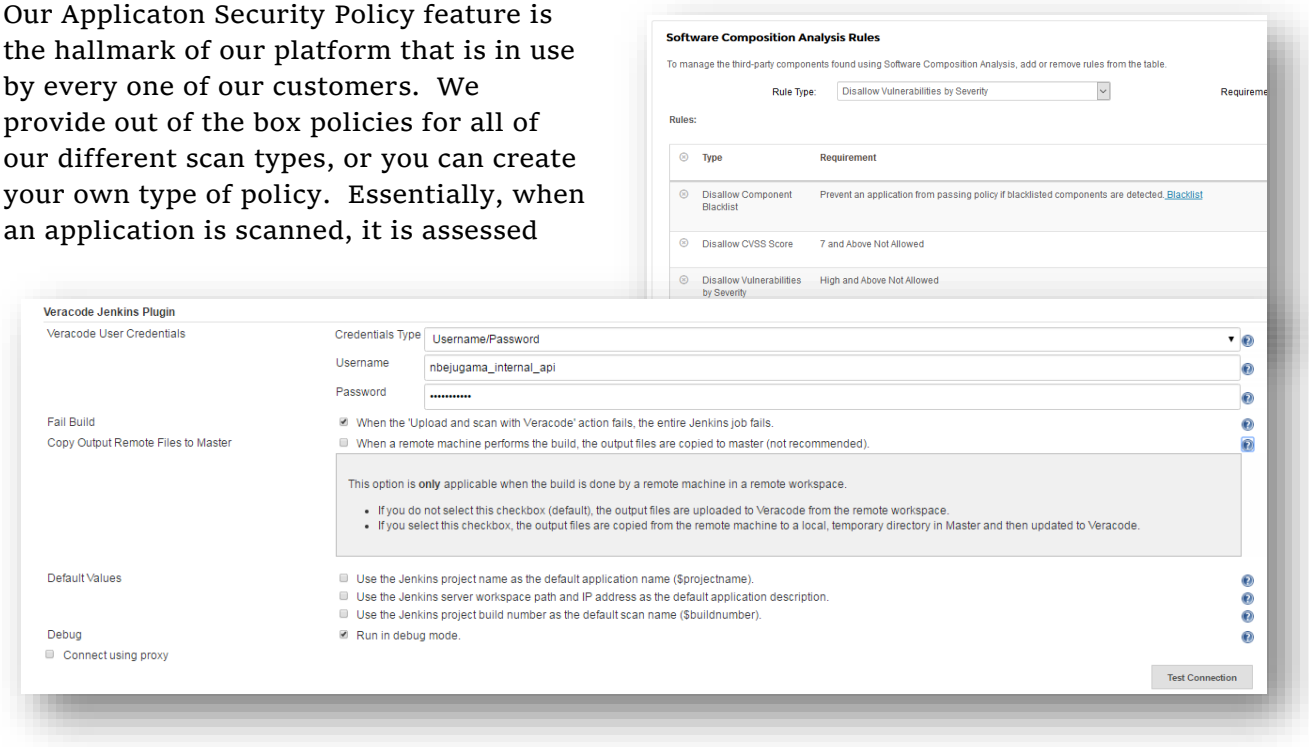
Licenses Found

Again, while not the primary focus of our solution, our customers ask us to provide information on what types of open source licenses are being invoked by the project. We currently check for over 180 of the more popular libraries, and we are adding more of them to our system all of the time. If there is a library we do not cover, we encourage you to let your Security Program Manager know so that we can get it added to our list.



Policies & Integrations (Rules, Filters, & Actions)

Our Applicaton Security Policy feature is the hallmark of our platform that is in use by every one of our customers. We provide out of the box policies for all of our different scan types, or you can create your own type of policy. Essentially, when an application is scanned, it is assessed



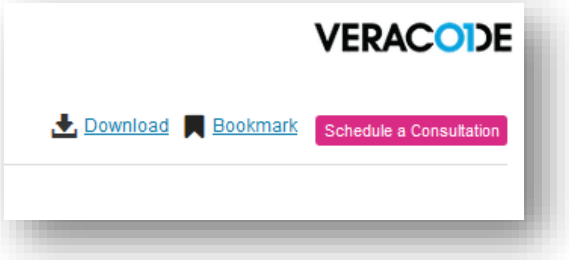
against the chosen policy and, based on the results, determined to have passed or failed the set policy. From there, customers can do a number of things based on a passed/failed flag from the platform. For example, if they integrate with Jenkins, they can fail a build based on a failed policy scan.

VERACODE

Security Alerts

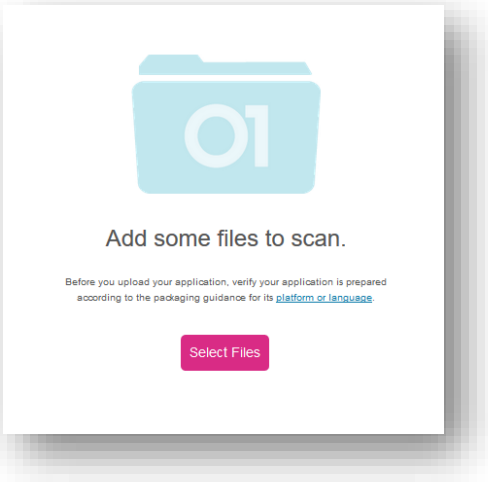
Vulnerabilities can be discovered at any time, and it's important that we keep our customers up to date with the latest information regardless of whether a library is being continuously scanned or not. With Security Alerts, if a vulnerability is added to our database after you have scanned a library, we push the result to your platform, and notify you that a new vulnerability has been discovered, regardless of the last time you scanned that library.

Developer Consultations



The greatest value our customers get out of using Veracode is not just finding their flaws and vulnerabilities, but actually remediating them. We have setup a lot of our platform to be as developer friendly (and we are always improving this aspect of our products) so that engineers can get the information they need as fast as possible, and begin working on the right path to remediate those issues. However, often times a developer hits a roadblock; either unable to understand what a specific reported issue is, or even how to fix/mitigate a problem. That's where our developer consultation calls come in: directly from our platform developers can schedule calls with our experts who can provide over the shoulder readouts of scan results and guidance on how to fix the problems we find. Nobody else has to knowledge and staff on hand to help developers produce secure applications every time.

MATURE YOUR APPSEC PROGRAM WITH VERACODE



There are a number of different places where you could start your application security program, and a lot of different paths to mature your program – but there are not a lot of companies that can help cover your needs from end to end. When assessing your options for your AppSec partners, you need to look for a company that can cover the entire software development lifecycle (SDLC), with a strong focus not only on first- and third-party code, but also the ability to actually implement a mature program. Veracode is the market leader in application security, and our years of experience have shown that those companies that evaluate their first-party code, plus open source libraries, and do so early, midway, and late in the SDLC have the best coverage. With CA Veracode, you can ensure a scalable, cost-effective AppSec program that helps make security part of your competitive advantage.