VansonBourne

ca technologies | VERACODE

The trials and tribulations of component security; are organizations at risk?

Whitepaper

Contents

Introduction

In the technologically advanced era that we live, organizations rely on third-party code more than ever before. As such, increasing the pace of development to create innovative applications and features is absolutely critical in order to stay relevant and provide the best service possible to their customers – if organizations do not do this, then there is an ever-increasing number of competitors waiting to take their business. The only way that many organizations can keep up, is to use third-party code, which allows them to leverage features and functionality that already exists.

With these benefits in mind, many organizations are now using third-party components to increase the speed of development that they can work to. But, these components need to be inventoried stringently and managed appropriately, as if they are not, then organizations may find themselves in a worse position than where they started.

In order to ensure that software components are being utilized safely, security review needs to be a key part of development and procurement processes. Development has always focused primarily on speed of deploying new features, innovation, cost, and other key business objectives – but with security breaches dominating headlines in recent years, it's time it took its place among the rest.

As we will go on to discover, organizations are managing more components and applications than ever before, but security is not keeping up. Action needs to be taken before it's too late. Breaches in third-party code, compliance with customer requirements and regulations are all increasing the need for strong third-party code to be secure. Can organizations afford to fall behind in securing their components and applications? Can you afford to fall behind?

Scope of research

Veracode commissioned independent technology market research specialist Vanson Bourne to undertake the research upon which this whitepaper is based. For this research, a total of 400 application developers were interviewed in February 2018. Interviews took place in the US, the UK and Germany.

Country	Number of interviews
US	200
UK	100
Germany	100

Respondents' organizations could have been from any size or sector, but include a good spread across the following:

- Business and professional services
- Construction and property
- Energy, oil/gas and utilities
- Financial services
- IT, technology and telecoms
- Manufacturing and production
- Media, leisure and entertainment
- Retail, distribution and transport
- Public sector
- Other commercial sectors

All respondents were interviewed using a rigorous multi-level screening process to ensure that only suitable candidates were given the opportunity to participate.

Key findings

Organizations are now most likely (41%) to choose DevOps as their IT methodology, with code being released to customers/production three times per week, on average

93% of organizations use commercial and/or open source components and of those organizations, the average number of components per application is 73

One in four (25%) admit that their organization does not have a formal application security (AppSec) program in place

Of organizations using third-party components in their applications, only 52% update those components when a new security vulnerability is announced

Organisations using an MSP are **more likely** to be **compliant** with the Data Protection Act (94% v 87%)

Only 53% of organizations keep an inventory of all components (top level and sub components)

Introduction to IT and security

More than ever, IT departments and employees are working in different ways across organizations; it's fair to say that the IT department is an area of an organization that is constantly evolving. When it comes to IT methodologies, organizations are now most likely to be using DevOps (41%), followed by agile (33%), with only a small minority (13%) using waterfall. This suggests a move away from more traditional methods and a move toward DevOps; security must be integrated within DevOps – leaving security until the end of the process is no longer good enough. The key to successful a DevOps infrastructure is integrating security within it.

Releasing new code to customers and production is part and parcel of a successful organization and the rate in which this is happening appears to be increasing. On average, organizations are releasing code to customers/production three times per week, although this is only twice per week in the UK. The high frequency with which code is now being released also suggests a move toward DevOps and emphasizes the need for a high level of security in this area.

With code being released faster than ever before, it is possible that development teams are needing to take shortcuts in order to keep up. For example, it is only around seven in ten (71%) organizations who have a formal application security (AppSec) program in place, with a quarter (25%) who admit to not having a program and 4% who do not know, suggesting that many teams ignore security or go around testing. Organizations who don't have this in place could be opening themselves up to cyber-attacks, putting a spotlight on poor practices, compliance issues and ultimately harboring massive risk unnecessarily. Security needs to become a formalized and integrated part of the development process, for organizations to reduce this risk.

Organizations using a formal application security (AppSec) program

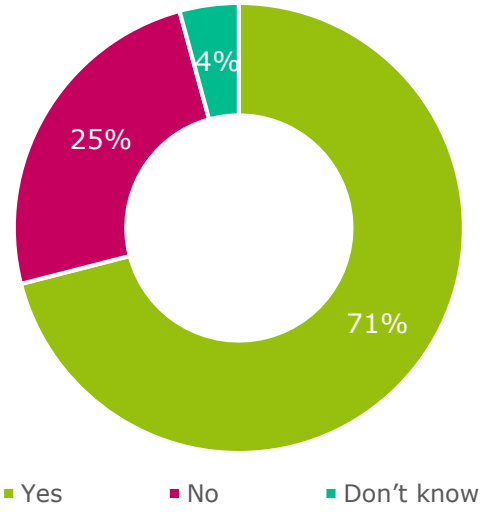

Figure 1: "Does your organization have a formal application security (AppSec) program?" *asked to all respondents (400)*

With the above in mind, surely all organizations should be using best practice security methods? But, when it comes to the most likely methods to secure code, it is fewer than three in five organizations who use web application firewalls (57%) and/or dynamic application security testing (55%). Perhaps even more concerning is that only two in five (40%) report that their organization is using static application security testing. Using a web application firewall is commonly perceived as 'doing enough', but the reality is that this still leaves a lot of holes; organizations can be vulnerable to attacks being missed due to new patterns, application changes and configuration complexity. Organizations need to be using these stronger methods, such as static application security testing.

When it comes to additional technologies that are securing the software development lifecycle, multi-factor authentication (70%), threat analytics (60%) and/or privileged access management (54%) are being used by the majority of organizations. This shows that organizations tend to value those additional layers of protection for their applications. But, it is critical to remember that code and applications still need to be secured. These additional technologies do not replace code security, they are to supplement application security.

It is worrying to see that it is only around eight in ten (82%) who report being at least somewhat familiar with the OWASP Top 10 application risks. It is almost one in five (19%) who are not at all familiar, with those in the US faring slightly better (28% who are totally familiar), but still far lower than it should be. Knowledge also varies depending on the respondent's role: 22% of software designers are not at all familiar, compared to 20% of team leaders and 10% of managers of teams. Overall, this highlights a distinct need for improvements to be made to when it comes to training and awareness of development and security – OWASP is generally seen as the application security standard, yet many fall short when it comes to their knowledge of it.

Perhaps even more concerning, those who do have some awareness still show holes in that awareness. Of those with at least some familiarity of the OWASP Top 10 application risks, only 43% say that they are very aware of OWASP recommendations for preventing the use of components with known vulnerabilities. OWASP recommendations are very much the best practices/standards to follow, so those who are not aware, could be missing out on valuable information and as such, their organization could be less secure than their competitors, putting them at a disadvantage. Ultimately, everyone should be totally aware of the best practice in their field, but for developers, it seems that this is not the case.

Familiarity with OWASP recommendations for preventing the use of components with known vulnerabilities

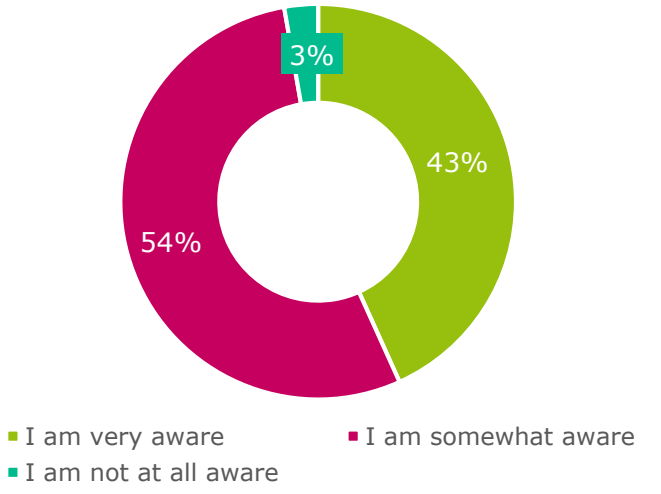

Figure 2: "Are you familiar with OWASP recommendations for preventing the use of components with known vulnerabilities?" *asked to respondents who are at least somewhat familiar with the OWASP Top 10 application risks (326)*

Exploring the use of third-party and commercial components in organizations

In the first section we discovered that organizations are releasing code more quickly than ever and speed is crucial to that. In order to work at maximum velocity, organizations need to use components, otherwise they risk being left behind. It is over nine in ten (93%) organizations who use commercial and/or open source components. Almost two thirds (63%) of organizations use both commercial and open source components in their applications. Under one in five exclusively use commercial (18%) or open source (12%) components. The way in which components are used is mixed across different organizations, but organizations are united in the appreciation that components are required.

Use of components

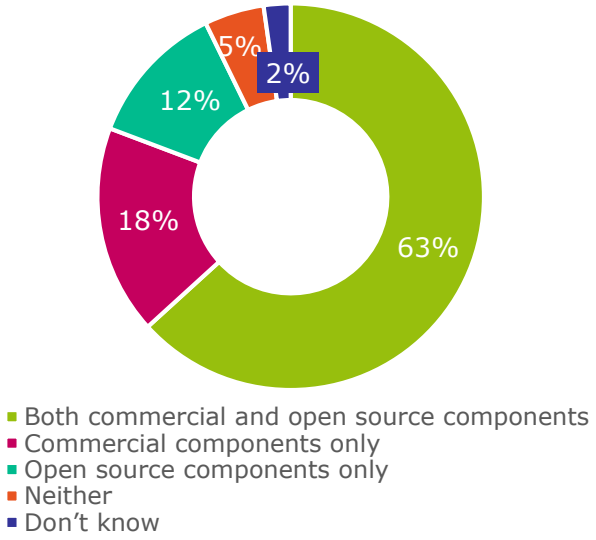

Figure 3: "Does your organization use third-party commercial and/or open source components in your applications?" *asked to all respondents (400)*

When considering the reasons behind not using commercial components, respondents are most likely to put it down to company policy (44%) and/or price (37%), but 28% say that it is due to security concerns. In addition, a concern over security (49%) is the biggest reason for organizations not using open source components, which suggests that there could be a lack of knowledge and awareness around how open source components can be secured and how a trusted partner can be valuable in that process. The right solutions are secure and do exist, but organizations need to familiarize themselves with them.

It's become clear that most organizations are currently utilizing third-party components. Of those respondents whose organization do use third-party components in their applications, some of the most likely benefits that these components can bring are developers working faster (51%) and the ability to use advanced functions that they can't create themselves (47%). Unlocking these benefits illustrate components' value and give the context as to why they are so important.

Reasons to use third-party and/or open source components

Figure 4: "Why does your organization leverage third-party commercial and/or open source components for its applications?" *asked to respondents from organizations that use third-party components in their applications (371)*

The use of components is common place and it's not only in small amounts. Those using third-party components report using 73 of these components per application, on average. This vast quantity is likely to make it incredibly difficult to track and update components. When throwing security into the equation, it points toward a demanding task for organizations to keep their applications secure and using an automated system could be the ideal solution to help with this.

Open source components are often the backbone of most modern, Java software applications. On average it is 42% of components that are open source, when asked to respondents whose organizations are using both commercial and open source components in their applications. This suggests that there is a shift in the way that organizations are using components and we might expect to see this proportion rising in the future.

Organizations are currently using components, but would application developers choose to use them if it were up to them? Just over nine in ten (91%) confirm that they would use components, which suggests that developers realize that it is best practice and beneficial to use components so that they can use existing code. After all, application developers will be some of the biggest beneficiaries of using components and should be able to focus their time elsewhere, such as innovating in new areas of IT and propelling their organization forwards.

The adoption and maintenance of third-party and commercial components

A whole host of third-party components are available and selecting the right ones can be a minefield. So how are organizations selecting which ones to use? Of those using third-party components in their applications, formal team processes (47%) and formal company processes (36%) are the most likely methods to choosing which components to add to applications. There isn't a distinct method which is definitively right or wrong, but it is likely to be a formal process. Ultimately, it doesn't matter which method organizations take, as long as they come to the right result, including the right security being applied.

These formal processes will undoubtedly mean that organizations have a number of facets to consider in selecting the right components. When it comes to the selection of a new open source or third-party commercial component, functionality (62%), performance (48%) and/or cost (44%) are the most likely considerations, followed by security vulnerabilities (42%), highlighting that security is essential but less of a priority. Those in Germany are even less likely (26%) to consider known security vulnerabilities, which could end up being a costly mistake.

Considerations when adopting components

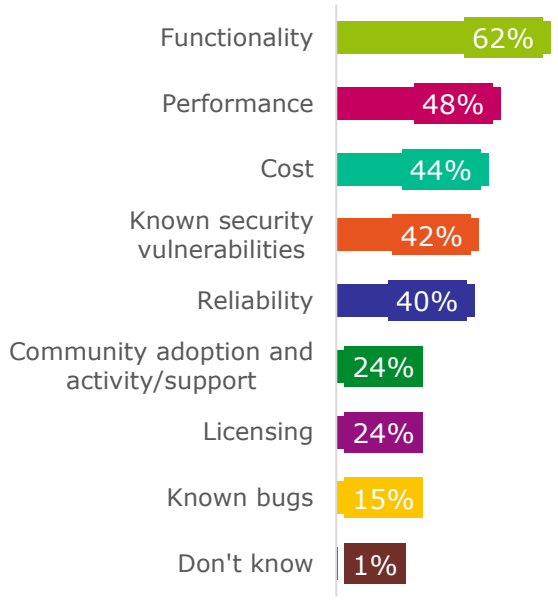

Figure 5: "What priority does your organization place on the following considerations when adopting a new open source or third-party commercial component?" *combination of responses ranked first, second and third, asked to respondents from organizations that use third-party components in their applications (371)*

In addition, individual contributors (38%) and software designers (33%) are less likely to state that known security vulnerabilities are a consideration when adopting a new open source or third-party commercial component (team leader (47%) and manager of a team (45%)). Management appears to be taking security more seriously, but is either team thinking about security as much as they should? The impact of poor security becomes clearer with news of every data breach and it is equally clear that organizations have work to do in shoring up security process and education throughout development teams.

Once these components have been implemented, who holds responsibility for their upkeep? It is the development (44%) or security (31%) teams that are most likely to be responsible for the maintenance of third-party commercial and open source components, which suggests a move toward responsibility for the development team, but, are they prepared for this responsibility? Responsibility should ultimately be taken across the organization rather than just one individual team. If only some segments of the organization take responsibility, then it could lead to other areas creating vulnerabilities, whether knowingly or not.

But, these aren't the only teams officially involved. Around seven in ten (71%) report that their organization has to account or disclose the licenses of their commercial or open source components to their legal team. The involvement of an organization's legal team will only make the processes surrounding components appear even more formal, which of course, is not necessarily a bad thing.

Worryingly, components are not always being updated and managed as frequently as they should; of respondents' organizations using third-party components in their applications, only around half (52%) update those components when a new security vulnerability is announced. This leaves organizations at a massive risk and suggests a distinct lack of focus on security. Many components are initially downloaded with severe vulnerabilities, so not updating them once a new vulnerability is announced makes this threat even bigger and less forgivable to ignore.

Updating components

Figure 6: "When does your organization update third-party commercial and/or open source components?" *asked to respondents from organizations that use third-party components in their applications (371)*

Another opportunity to check for vulnerabilities would be at the time of a new build or release. However, it's only around one in five who say that their organization tests/checks for component vulnerabilities at every build (17%) and/or every release (23%) – in addition, it is only 5% of those in Germany who report testing/checking at every build. This is particularly worrying when remembering that organizations are releasing code to customers/production three times per week, on average. Organizations must have a clear inventory of components or be checking at every release, with doing both of these things the ideal scenario. But, it appears that many organizations are falling short and are finding themselves doing neither, as we will continue to see.

Organizations should be testing far more frequently when considering that there are 71 security vulnerabilities per each application built with third-party commercial and/or open source code components, on average. Perhaps a result of appearing to have a lower sense of priority associated with security, those in Germany report more (83) vulnerabilities, on average, which could leave them at much greater risk if they are not prioritizing security. Furthermore, software designers report an average of 99 security vulnerabilities, compared to individual contributors (60), team leaders (60) and managers of a team (58). It's unclear why this gap exists, but could signal a lack of clearly understood security policies, process, and communication amongst development teams.

Inventory management

Organizations should be keeping on top of their inventory of components, but in reality, there are big gaps in those inventories. Only just over half (53%) of organizations keep an inventory of all components (top level and sub components). Organizations should be aware of their full component list, but many may not have the right systems in place or are not working with the right partners, as they do not have any insight into their full software inventory.

Keeping an inventory of components

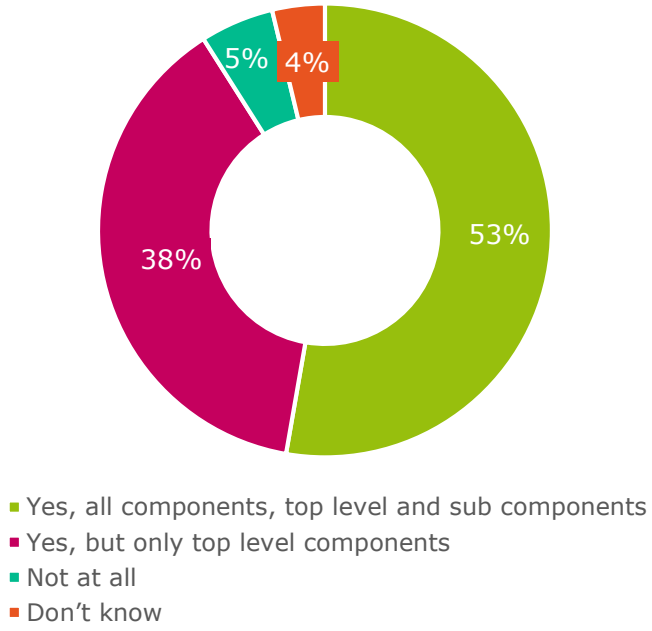

Figure 7: "Does your organization keep an inventory of the components it uses?" *asked to all respondents (400)*

Many of those who do have an inventory in place are not updating it as efficiently as they can be. It's 51% of respondents' organizations who generate and maintain their inventory automatically via a tool like software composition analysis, showing that there is still a large proportion yet to adopt. For the remainder, they're likely to be struggling to keep their inventory up to date, or will be doing so in an inefficient way.

Version information can be tracked in a number of ways. Among those organizations keeping an inventory of the components that they use, around a third do so automatically via Twitter, RSS, or community feeds (36%) or through the National Vulnerability Database (NVD) (33%). There isn't a single source solution which can be adopted across organizations, even though this should be a basic requirement.

Compliance can shape inventory management; for the majority (58%) of organizations, the purpose of their inventory of components is compliance/audit. Just under half say that the purpose is a periodic architectural review (49%), a periodic legal review (49%) and/or developer documentation (47%). Inventory management still tends to be driven by compliance and review, which suggests that this is still an emerging concept and problem for these organizations.

Conclusion

Most organizations are aware that they need to adopt more modern IT and development practices and have left their old methodologies behind. As such, organizations are releasing new code on an increasingly frequent basis, but many are leaving huge security holes and vulnerabilities in their components. Organizations' security is yet to catch up, despite the majority working with third-party components. Perhaps organizations' awareness of security is not as good as they think, or as good as it should be, but ultimately, it is leaving them in danger.

Open source components are being used by the majority of organizations. But, for those who are not, security is the most likely inhibiting factor. These organizations who are not, are missing out on the vast benefits of doing so and could see themselves falling behind their competitors who are. Markets are becoming more competitive than ever and falling behind through a security flaw would be a seemingly needless error.

Furthermore, the development team looks like it's taking a more central role in the responsibility for the maintenance of components than would traditionally be expected, but are often not well informed on security best practice. Have organizations forgotten about security? Are development teams ill-prepared to consider security? Is security suffering as a result? Worryingly, the answer to each of these questions could feasibly be yes.

Another element of this is that organizations do not appear to be tracking vulnerabilities as frequently as they should and could risk huge implications if one of these is breached. The need to track these weaknesses is heightened when considering the vast array of components that are in use and how many potential vulnerabilities each component has. If organizations do not start to track components and vulnerabilities more carefully then they will continue to be operating in an insecure way and will likely be caught out sooner rather than later. This becomes even more pertinent when considering the amount of new components that are being used within organizations, and forgetting about the old components and vulnerabilities becomes a distinct possibility.

However, it's not too late for organizations, regardless of their current methodologies, infrastructure or existing vulnerabilities. The move toward DevOps does not mean that an organization's journey is complete – security is just an extension of that and a natural progression to make. If an organization is prepared to prioritize securing their software and partnering with a reputable security solution provider, then they can pull themselves away from threats, leapfrog the competition and endure a legacy in their industry.

About CA Veracode:

Veracode, CA Technologies application security business, is a leader in helping organizations secure the software that powers their world. Veracode's SaaS platform and integrated solutions help security teams and software developers find and fix security-related defects at all points in the software development lifecycle, before they can be exploited by hackers. Our complete set of offerings help customers reduce the risk of data breaches, increase the speed of secure software delivery, meet compliance requirements, and cost effectively secure their software assets- whether that's software they make, buy or sell.

Veracode serves over a thousand 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, on Twitter and in the CA Veracode Community.

About Vanson Bourne:

Vanson Bourne is an independent specialist in market research for the technology sector. Our reputation for robust and credible research-based analysis, is founded upon rigorous research principles and our ability to seek the opinions of senior decision makers across technical and business functions, in all business sectors and all major markets. For more information, visit www.vansonbourne.com