WHITE PAPER

The Five-Step Road to DevSecOps Success

How AppSec professionals can nurture (and thrive in) their dream team

The working life of a software security professional is many things: challenging, exciting, unpredictable… but rarely is it easy. Even if we reflect on the state of software development many years ago, when the world was demanding far less code, and before there was an app for everything from identifying music in real-time, to telling you the best times to take a bio break during a movie (no, seriously), security professionals were an almost mystic entity, spoken of but scarcely sighted as the software was being created. In most organizations, they were siloed, working separately from operations teams and the developers tasked with creating new applications.

Developers would write their code, ensuring it was functional, beautiful and feature-rich. Once it was complete, they would normally hand it off to an operations team to install in their company's production environment. Security was almost an afterthought most of the time, if directly considered at all. In fact, it wasn't uncommon for development teams to hold back shipping code, shortening the security team's review window so they wouldn't be held up by the potential bad news that their code was, in fact, insecure.

Now, when security flaws and program vulnerabilities were eventually discovered, either by one of those hidden AppSec people - or worse - after a user exploited an application, the operations team would pull the app and send it back to the developers to patch. By then, developers were often already working on new projects, and had to pause those efforts in order to fix old problems. They did their best, but going back and shoehorning security features into an application that a development team completed six months ago is hardly efficient. And sometimes fixing one problem would create many others, and the whole cycle would continue, with security specialists and developers enduring a fairly frosty and misunderstood relationship.

DevOps has arrived, but did the dysfunction leave the building?

Those old-style development processes were flawed, to say the least. Software development was moving too slowly, with potentially unsafe code that exposed organizations to unacceptable risk. The security teams were deployed on rescue missions far too late, with processes that could seriously affect project delivery. Something had to give, and thus, DevOps was born. The DevOps movement was created in order to both restructure the development process and to help make applications more secure.

Taken as a whole, DevOps is a successful movement that, when applied correctly, forges a meaningful collaboration between developers and operations team when creating new software. It certainly streamlines the development process, a necessity these days when organizations are creating hundreds (or even thousands) of new applications every year.

But, alas, DevOps isn't perfect[1]. Those coveted security people are still essentially kept away from the start of the process, and their relationship with developers is still fraught. Having developers and operations teams work together helps to streamline application deployment, but doesn't fully lock down every security issue. Dedicated AppSec professionals are still required, who are trained in how to spot and stop the latest vulnerabilities, security issues, and exploits.

Today (perhaps more than ever), organizations are hyper-aware of their need for this skillset, and AppSec is a fundamental part of the development process. So, with that all front-of-mind, we must be pretty equipped to deal with cybersecurity now and into the future, right?

Not exactly.

> **The DevOps movement was created in order to both restructure the development process and to help make applications more secure.**

[1] Danhieux, Pieter. Why DevOps Implementation is Often Unsuccessful (and How You Can Fix It)

Putting the "Sec" in "DevOps": The first step in shared responsibility for security

We have some organizations that are working within a DevOps environment, but with a separate AppSec team. And some groups have not yet made the plunge to DevOps at all. Perhaps they are still coding using something like Agile, which is efficient, but doesn't take security into account as its primary focus.

No matter what stage an organization finds itself in, the end goal should be evolving into a fully-fledged DevSecOps program. Putting the "Sec" in "DevOps" is where the magic can really happen – as in, remediation of common security vulnerabilities that open up great risk, and a noticeable reduction in time and cost to fix them. A thriving DevSecOps process helps to secure code from the start of production, rather than patching and debugging it at the most expensive stage: after it has already been committed. According to an IBM study, it is thirty times more expensive to fix vulnerabilities in post-release code than if they were found and remediated at the beginning[2].

DevSecOps, which combines the words development, security and operations, has become both a software engineering tactic and a culture that advocates security automation and monitoring throughout the software development lifecycle. The primary goal of DevSecOps is to break down barriers and open collaboration between development, operations and security teams so that all of them can contribute to the creation of new applications, and organizations can deploy new apps with secure, working and efficient code.**But make no mistake, security is the primary goal.**

In these times of disastrous data breaches every other day, deploying a secure application must have equal importance to most organizations as whatever core function the app will be conducting. Coding an application that works fine, but which exposes a business to a potential exploit, is just as much a failure as making an app that doesn't function properly. **And we must get to a point where everyone is responsible for application security, with a clear understanding of the role they play.** This should be music to the ears of scarce, burnt-out and overworked AppSec professionals trying to secure a neverending deluge of code – it's an all-hands problem that requires an all-hands solution (carried out with the direction of those AppSec experts, of course).

[2] IBM Software Group; Minimizing Code Defects to Improve Software Quality and Lower Development Costs

If DevSecOps is the future, then why aren't AppSec teams demanding its implementation?

Unfortunately, moving to an efficient DevSecOps program isn't always easy. It's both an operational and a cultural change for most companies, but the results from successful implementation speak for themselves. In today's cyber environment hackers are continuously scanning for vulnerable code, and insecure apps can be attacked within moments of their deployment.

As an industry, we must do better, and we need to do it soon.

This is not a task for just one person, but AppSec professionals and security-minded development managers just like you can play a fundamental role in staying true to the DevSecOps path, influencing others, and understanding how you can approach and grow your skills for future success.

In this white paper, we will examine some of the key elements that all successful DevSecOps programs share, while providing practical advice for how you, the AppSec specialist (or, indeed, a security-minded development manager), can make the most of this new methodology. With a little bit of work and an open mind, any organization can evolve from whatever processes they are using today, to creating functional, secure applications in an efficient DevSecOps program.

> Any organization can evolve from whatever processes they are using today, to creating functional, secure applications

What does a DevSecOps dream team look like?

Defining a single strategy to create a DevSecOps team that works in every organization and every situation is impossible. It's a bit like examining a winning professional sports team and then trying to mirror their success by building a second one that is exactly the same. There are too many variables for an exact copy to work every time.

But, we can look at winning sports teams and find commonalities that they all share. For example, they will inevitably all have talented players, who have an innate understanding of their own capabilities and that of their teammates. They will have supportive management, a good coach, great training programs, a positive locker room environment, good equipment, an understanding of the rules and strategy of the game, and the ability to work as a team where players mutually support one another toward their common goal of winning championships.

It seems deceptively obvious and simple, but you just have to look at some of the dysfunctional sports teams out there that are light years away from winning the top prize; there will be a key, simple element (or elements) that hinder their success.

DevSecOps teams are the same way. DevSecOps is as much a culture as a methodology, so not every blueprint will work perfectly within every company. However, the majority of successful DevSecOps groups, like winning sports teams, will share many of the same positive elements. Fostering those key elements, modified as appropriate for your organization, will help your company field winning DevSecOps teams that can both improve application development efficiency and security.

The majority of successful DevSecOps groups, like winning sports teams, will share many of the same positive elements

Implementing Successful DevSecOps Elements for Developers and AppSec Professionals

The following are some of the most important elements that highly successful DevSecOps teams share.

1. Successful DevSecOps teams recognize that security is a shared responsibility.
2. DevSecOps teams thrive best in a highly supportive environment where everyone from individual team members to the supervisors and management embrace the culture and support cooperation and collaboration efforts.
3. The top DevSecOps teams train constantly and improve their skills as they work.
4. Successful DevSecOps teams employ a suite of tools that are tailored specifically for the job at hand, and can be understood and used by developers, operations teams and security personnel equally.
5. The most successful DevSecOps teams are transparent with one another. They understand each team's core functions, strengths and limitations, and they play to those strengths.

These aren't a complete list of everything that successful DevSecOps teams share, but are some of the most important. If your company can get AppSec, operations and development teams to work together on these five core goals, it will go a long way to making sure your DevSecOps program starts strong and maintains efficient, secure coding and application development for years to come.

Now, let's take a look at how AppSec specialists can achieve these goals, and integrate into an enviable DevSecOps dream team.

Although creating a DevSecOps operation and culture within your organization is the overall goal, the process by which development and AppSec teams will get there won't be identical. Developers will initially have a completely different outlook and skillset as AppSec personnel. Their end goals with implementing DevSecOps will be similar, but what they have to do in order to achieve success will be, in some cases, vastly different.

The following is some practical advice that AppSec professionals can use to help align their organization's new DevSecOps program for success. By studying tested and proven elements of teams that have effectively employed DevSecOps in the past, you can help to support a smooth transition within your own company.

Are you a developer looking for advice on making the transition to a DevSecOps environment?

This white paper is for you →

1

Successful DevSecOps teams recognize that security is a shared responsibility

Prior to DevSecOps, the process of maintaining the security of applications, especially in organizations that deploy hundreds of apps every year, could be a pretty thankless job. AppSec groups were sometimes the only staff members looking for security problems, and when found, their only recourse was normally to pull an app and send it back to development teams to fix. Depending on the volume of work the developers were doing with new applications, it could be a long wait. And AppSec was almost always seen as the “bad guy” by both developers and operations teams – the wielders of dark arts that could stop code shipment in its tracks.

The prospect of working together with developers on the common goal of security within a DevSecOps program should appeal to AppSec teams. However, they should understand that there may be some fear and resistance to such a move from developers. They should strive to reduce and eliminate both by taking more of a supporting role in DevSecOps. They should no longer be seen as security overseers, but instead champions of their new development team partners.

Like any cultural shift, some careful change management is required. It’s important to listen to developer concerns; invariably, they will want to understand what this new process means for them - and most importantly - how much it will disrupt current projects, or sap precious time from their ever-growing list of priorities. Give them the space to adjust, ask questions and establish yourself as a friendly point of contact, not a school principal. The aim is to reveal the benefits of being a security-aware developer, including the value it could add to their CV. They are the most vital piece of the puzzle when it comes to reducing vulnerabilities from the start; make sure they know that and feel part of the team, not an ongoing rival.

Ultimately, AppSec specialists need to get developer buy-in when it comes to caring about security. You will have to meet them on their level, not shoehorn them into your own. Show a clear understanding of their current situation, and reiterate that they’re not alone in making the switch; after all, that’s what DevSecOps is all about.

2

DevSecOps teams thrive best in a highly supportive environment

AppSec teams should revel in the fact that if a company’s DevSecOps implementation is successful, not only will security improve, but they will no longer be seen as the roadblock to creativity and innovation, one who creates more work for everyone, sometimes long after a job was thought to be over. In fact, it can be argued that AppSec teams have the most to personally gain from a successful DevSecOps program. Everyone will suddenly be working with the same goals, and AppSec will become an ally in that fight. They finally get to be partners with their coworkers, supporting and empowering them in their own security awareness goals.

As such, AppSec teams should do everything they can to foster a supportive environment that is focused on security. That starts with becoming an advocate for the development teams by ensuring that they have all of the tools and training they need to make security a core of the application development process. Even if developers are working hard to support a positive environment, it won’t matter if they don’t have the proper tools for the job, or a basic understanding of security.

Insist on equipping developers with everything they need to make secure coding a priority, including adequate time to learn new processes, train and start upskilling. And for bonus points, bring them on the journey when it comes to choosing the training and tools, within reason. Test a few and get insights from the actual users on what is engaging and useful to their day jobs.

Under DevSecOps, good AppSec teams will offer as much positive feedback, or hopefully more, than negative criticism. In the end, although it might seem like AppSec teams are losing some of their implied authority, what they will gain is a better work environment and more secure applications for their company.

And last but not least, their jobs will become a lot easier overall. AppSec experts are rare, and they can be criminally underutilized. Many are caught up fixing common vulnerabilities over and over again: think cross-site scripting, SQL injection – problems that have existed for decades now, and that developers could easily remedy themselves with the right tools and knowledge. This takes them away from the truly complex security problems that cry out for their expertise.

Sound familiar?

The prospect of working together with developers on the common goal of security within a DevSecOps program should appeal to AppSec teams

Having insights into how best to approach knowledge gaps is far better than making assumptions or judgements about overall skill levels

3

The top DevSecOps teams train constantly and improve their skills as they work

Although continuous security training is good for anyone working in cybersecurity, the bulk of the training efforts when starting up a DevSecOps program will be implemented for the benefit of development teams. As the experts on security, it will be AppSec's responsibility to ensure that developers are offered training that is highly engaging, relevant to their jobs and contextual[3].

Training should never be a once-off exercise, or performed every few months. That is not good enough for a DevSecOps program where developers are charged with coding secure applications every day.

The training should also be supported with good metrics. That way, AppSec teams can see what security vulnerabilities developers are successfully learning about, and what areas still require more work. This also helps immeasurably in strengthening the general relationship between AppSec and development teams; it's always good to approach problems with a healthy dose of empathy and understanding, and having insights into how best to approach knowledge gaps is far better than making assumptions or judgements about overall skill levels.

4

Successful DevSecOps teams employ a suite of tools that are tailored specifically for the job at hand

Your new developer partners in a DevSecOps program are probably going to be very enthusiastic about helping to close the cybersecurity skills gap, especially if you have done a good job of nurturing a positive environment. But, they are not mind readers or magicians: they can only implement changes if they have the knowledge and tools required to carry them out successfully. It's the job of AppSec groups to ensure that developers are empowered to effectively perform their new roles.

It's also important to strike the right balance between training and tools, as they go hand in hand when cultivating a solid DevSecOps program. AppSec must ensure that developers are equipped like an AppSec team, with multiple tools that can scan for vulnerabilities in code, deliver continuous training and enforce company-specific cybersecurity policies.

And in terms of AppSec team tooling? Well, it's common knowledge that there is no "one" Swiss Army Knife multitool that will scan for every vulnerability, in every language, in every conceivable context. A discerning mixture of strategies is required, and scanning (not to mention manual code review) can be a very drawn-out process. Getting the right balance of training and tools across the whole DevSecOps team can save a lot of time for you at the security finish line.

[3] Singh, Jaap. Contextual, Hands-On Learning: The Supercharged Way to Train Your Brain for Security

5

The most successful DevSecOps teams are transparent with one another

While developers will generally be learning how to use new tools and tactics, AppSec teams will already be in familiar territory. It will at least be very similar to other tools and techniques they have previously employed. As such, the job of an AppSec team will be to help train developers how to use their new, transparency-friendly tools.

The one area of concern for AppSec professionals once everyone is working within a fully transparent system of application development, tools and tactics, is to remember that AppSec is still ultimately responsible for security. The developers will be working hard to bake security into their applications, but AppSec will still need to sign off on everything.

The difference is that under DevSecOps, the AppSec teams will generally be finding fewer problems. And instead of sending them back asking for a fix, the transparent environment means that AppSec professionals can actually show developers how to fix a vulnerability using universal coding tools. The ultimate responsibility of AppSec remains unchanged, but the methods where they become mentors and educators, as opposed to overseers, will change.

And that change will be for the best.

The job of an AppSec team will be to help train developers how to use their new, transparency-friendly tools

Are you ready to begin with DevSecOps?

We hope that this white paper has given both developers and AppSec teams a lot to consider about migrating over to an efficient DevSecOps program that supports the creation of secure applications. Lots of organizations have made their move to DevSecOps, but the journey getting there is not always easy.

If you would like to talk about the benefits of baking in security with your own application development, or are looking for a partner in that effort, please contact Secure Code Warrior to see how our tools, technology and expertise can help your organization implement and support a world-class DevSecOps program.

ABOUT SECURE CODE WARRIOR

Secure Code Warrior is the developer-chosen solution for growing powerful secure coding skills. By making security a positive and engaging experience, our human-led approach uncovers the secure developer inside every coder, helping development teams ship quality code faster.

Through inspiring a global community of security-conscious developers to embrace a preventative secure coding approach, our mission is to pioneer a people-first solution to security upskilling, stamping out poor coding patterns for good.

info@securecodewarrior.com

securecodewarrior.com