The SecOps Playbook
How SecOps Enables Secure Code Release, At Scale and At Speed
### TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>
Why SecOps? Why Now?

The term “SecOps” — interchangeable with DevSecOps and SecDevOps — isn’t new. Since its introduction into our vocabulary, a lot has been said about its history and why it matters, but we haven’t actually described ways of implementing it. DevOps itself has now matured to a point where most high-velocity organizations running on modern infrastructure have embraced it or are thinking about implementing it; and as a result, many have felt the struggle that comes from trying to balance speedy DevOps practices with necessary security practices.

This playbook explains how SecOps can enable companies of all sizes and types to continue releasing code regularly without sacrificing security. SecOps requires a shift in mentality and culture, and in this playbook we show you how to achieve it using a real-world action plan.

Let’s start with a short story.

The Briefest History of DevOps

Before the DevOps movement came along (around 2009), development teams and operations teams were incentivized differently. Ops was in charge of ensuring stability while Dev was supposed to be laser-focused on continuous deployment. For a long time, it seemed that these two perspectives were in direct opposition to one another. How can we release stable code if we’re moving so fast?

Fortunately, the two sides eventually realized that they were working toward a set of shared goals — and that their efforts didn’t have to be diametrically opposed. Through the discipline of DevOps, they were able to get on the same page and cooperate so that both teams’ objectives could be achieved sustainably.

That was all well and good, but it left out a key function — you guessed it — security. What good is stable, frequently released code if it contains bugs and vulnerabilities? (Answer: No good. No good at all.)

If you’ve ever worked in a modern, tech-driven organization, then this will probably sound familiar. At one end of the company, DevOps is moving fast; at the other end, security is madly scrambling to keep up. DevOps gets all the glory, while security gets all the blame. Meanwhile, no one is shipping secure code. Enter SecOps.
The Present and Future: SecOps

While we didn’t coin the term alone, Threat Stack has argued for introducing security into the DevOps lifecycle sooner rather than later. Especially in the cloud, SecOps has the power to make continuous deployment a secure process once and for all.

The goal of SecOps is to help companies deliver software more efficiently and more securely. The point is to release code that is well-tested and hardened without slowing the organization down. That's the sweet spot, and only a SecOps approach can achieve this today.

It’s time to move past thought leadership on this subject and bring SecOps into the “real world.” So this playbook is designed to help you make SecOps a reality at your organization.

Who Should Implement SecOps?

My organization?

Yes, yours.

Your organization is a good candidate for SecOps if you:

• Have a development team
• Have an operations team
• Are doing DevOps (or want to be)
• Need to build and release secure code at scale and at speed
• Have sensitive data and/or systems you need to protect
• Have customers who require that you provide both timely releases and absolute security and compliance

We’d be hard pressed to point to a technology-oriented organization anywhere on the small-business-to-enterprise spectrum that isn’t a good candidate for SecOps. It integrates practical security processes into continuous development practices, and many organizations can benefit from it. No matter what resources you do or do not have at hand, including personnel, budget, or tools, SecOps is both critical and achievable.
What are the Challenges of Implementation?

There are several challenges to implementing SecOps. Let’s take a look at them and the ways in which they can be overcome.

**Budget**

One objection we hear a lot is that we don't have enough resources and can't afford to acquire what we need. However, thanks to the large number of open source tools that are cropping up, no matter how big or small your budget, it’s possible to pull together a team and a suite of tools that can make SecOps a reality. You just need a roadmap — which is what you’re reading.

**Talent**

Another objection is the lack of talent. There’s no denying that we are in the midst of a major security talent crunch. There are simply not enough security experts on the market, and it’s particularly hard to find qualified candidates who are skilled in the latest tools and technologies. Even when teams manage to hire good security people, the ones that can code often get stolen by development teams. The good news is that you can make security happen whether you have a full security operations center (SOC) or no infosec employees at all. And the better your tools and processes, the fewer experts you'll need. Moreover, in the modern organization, security should not just be the purview of analysts or even SecOps teams; it should be a team-wide, top-to-bottom effort.

**Tools**

Finally, we hear a lot of people say they don't have the tools that are necessary to make SecOps happen. While that has been a valid complaint historically, these days it holds less water. Between free and open source options that are cropping up and comprehensive solutions like Threat Stack, tools are available to meet nearly every need. Moreover, many of the tools currently being used by DevOps can also be used to ensure security.

Security is People: Powering a Cultural Shift

DevOps itself requires a pretty big cultural shift. If you’ve been on a team that has made the transition or transferred into an organization that is DevOps-oriented from one that was not, then you’ve probably seen the differences and experienced the growing pains firsthand.

One of the chief sticking points centers on what DevOps even means. Is it a job title? A team? A methodology? The answer is all of the above, depending on your organization. But more than anything, it is a culture whose central tenet is continuous deployment that is functional and scalable.

SecOps requires a cultural shift all its own. Once an organization has shifted its mindset to focus on integrated tools and workflows for development and operations, it’s time to pull security in. With the right approach and toolset, it can be easier than you think.

“Once an organization has shifted its mindset to focus on integrated tools and workflows for development and operations, it's time to pull security in.”
Here are the six steps that we consider necessary to succeeding with SecOps.

**1. Get the Right People in the Room**

Before you can get a SecOps program off the ground, you need to figure out who needs to be part of the cultural shift described above. Our recommendation is to take a top-down, bottom-up approach.

That means getting C-levels and executive decision makers on board early. It can be as simple as the CEO telling the team “Make our product more secure.” To be effective, the directive must come from high up (even if it requires some prodding from below). A C-level directive alone may not be actionable, but it’s the first step to changing the culture.

Next, DevOps and security teams both need to buy into the goals of a SecOps transformation. Often, one or both of these teams will be the driving force behind the change, but both need to see how it will solve pain points and bring about commonality (in much the same way that Dev and Ops had to do back around 2009).

Specifically, on the Dev side, it’s key that the release engineering team (which is responsible for shipping code as fast as possible) buys into the value of security and welcomes the security team into their world with open arms. It may sound uncomfortable, and it may require quite a few conversations, but it’s the only way to achieve lasting cultural change.

**2. Educate the Team About SecOps**

Next, make sure everyone is on the same page about what SecOps is and how it will work in your specific organization. This can be tough to do in a vacuum, so we recommend getting out and talking to others who have been there and done that.

A good way to start is by attending conferences. DevOpsDays and the Security BSides are two that often focus on building the bridge between security and DevOps. Find others who have brought security into DevOps. Talk to them about what has worked and not worked in their organizations, and use their real-world experiences to navigate your own hurdles.

Tons of great online resources exist, such as gauntlt.org, which "provides hooks to a variety of security tools and puts them within reach of security, dev and ops teams to collaborate to build rugged
software. It is built to facilitate testing and communication between groups and create actionable tests that can be hooked into your deploy and testing processes.” Gauntlt is a good place to learn about the concept of Rugged DevOps, which means putting better security tools into the same pipelines that developers and operators are using.

Making sure that everyone understands what is meant by SecOps, why the organization is moving in that direction, and how it will benefit everyone in their day-to-day roles is a great way to ensure that cultural change isn’t just something that gets talked about but never acted on.

3. Implement a Buddy System

When it’s time to start implementing security in the DevOps workstream, Threat Stack’s Head of Ops and SecOps veteran, Pete Cheslock, recommends the buddy system. This means that the release engineering team and security team need to work together in one-on-one pairs.

Yes, really. Getting individuals to work together is not only effective from a speed perspective (think: pair programming), but it also gives both sides an opportunity to learn each other’s priorities, perspectives, and tools. Pairing people with complementary skill sets so they can learn from each other along the way is far more effective than sitting both teams down for a presentation that will put everyone to sleep. Sometimes this is called “embedded DevOps.”

On the release engineering side, teach security team members how to use the tools that are being used to ship code. For example, teach configuration management tools to the security team. Another good tool that can be used within the “buddy system” is static analysis like Veracode, which allows teams to test software without executing it.

In much the same way that developers have learned to QA their own code, pairing security and DevOps can teach engineers to test their work for security and enlighten security on how the deployment process really works.

4. Start With Low-Hanging Fruit

So your people have bought in and you’ve got them paired up. Now what? Where do you start?

First, resist the temptation to feel overwhelmed. If your organization has little to no security in place, getting from insecure to secure can feel like a MAJOR undertaking. However, as with anything else in life and business, you have to start somewhere.

Pick a small but meaningful area of security that you want to improve at your business. For example, you could look to address the most common attack vectors for your industry or fix glaringly insecure processes being used at your company. Make it your goal to be five percent more secure than you were yesterday, and distill progress into small projects.

“Make sure everyone is on the same page about what SecOps is and how it will work in your specific organization.”
To identify low-hanging fruit, ask questions like:

- What types of data are we storing and sending?
- What happens if that data is compromised?
- How secure are the vendors we use?
- What do our customers expect of us security wise?

These questions can help you get clear on what the biggest threats to your organization are and how you might begin to protect against them.

Some common areas that many businesses want to tackle early on include:

- Security training (phishing awareness is a good place to start)
- Two-factor authentication and password management
- SSL certificates (tip: self-signed are bad)
- Security alerting (get DevOps and security using the same alert channels)
- Risk assessment
- Policy development
- Incident response plans (look at VictorOps and PagerDuty)
- Encryption (PGP is a good place to start)

Now, almost as important as knowing what you do need early on is knowing what you don’t need. For example, when you are just getting started with security, you don’t need an information technology infrastructure library (ITIL) IR plan.

5. Compile Your Suite of Tools

Once you have clarity on the areas of security you want to start with, you need to build your arsenal of tools. Here’s what you need in that suite:

**Configuration Management**

Developers use configuration management tools to build repeatable systems. They can also be leveraged for security purposes to help you test systems and quickly and easily update important systems when a vulnerability comes to light (e.g., Shellshock or Heartbleed.)

Some examples include Chef (and InSpec), Puppet, and Ansible. Configuration management tools are designed to notify the team if things fall out of line security-wise, so that it’s not someone’s job to manually check in and make sure everything is sympatico.
Incident Management
If you don’t have a repeatable and clearly outlined way to respond to incidents when they happen, you may find yourself flailing at the worst possible moment. Incident response tools like VictorOps and PagerDuty can dramatically simplify this process and make it possible to quickly notify on-call security staff if something goes awry.

Security Monitoring
Next, you want to focus on achieving complete visibility into your systems and data, so you know when vulnerabilities or security threats crop up and you are able to address them as quickly as possible. The best way to do this is to invest in a continuous monitoring tool. Specifically, a host-based continuous monitoring platform like Threat Stack will allow you to monitor all aspects of your systems in real time, and will also provide audit trails, investigatory tools, and flexible alerting to make sure you get the most important alerts delivered in a timely fashion.

6. Climb From Low-Hanging Fruit to the Top of the Trees
We strongly recommend that you put together a "Security Improvement Plan" that ranks action items from simple to complex and most important to least (taking into account the things that can make the biggest difference for your organization). The preceding outline of low-hanging fruit tasks and necessary tools should give you a strong starting place.

When you put together your security improvement plan or roadmap, be sure that it integrates with DevOps processes at every step of the way. Implementing a configuration management tool for security? Pick the one your DevOps team is already using. Want to start tracking logs? Get on board with what the company already has in place. This way, every time you make progress with security, DevOps is right there with you.

It’s important to incorporate continuous monitoring into your stack as early as possible, since this will help your organization achieve a wide range of security objectives all at once. Having a tool like Threat Stack that can pull together many of the tools and processes we’ve discussed in this playbook will streamline security operations and ensure that security is manageable, especially as you grow.

“We strongly recommend that you put together a 'Security Improvement Plan' that ranks action items from simple to complex and most important to least.”
Okay, so you’re putting the above wisdom and actionable tips to work. But how do you know that it’s working? First, you should be able to answer a number of questions (ideally in the affirmative) to indicate that your security program is where it needs to be. Here is a sampling:

- Could you deploy a patch today if you had to?
- How quickly would you be alerted if there was an invalid login attempt?
- How frequently are you able to identify a needed security patch?

We also advocate for developing some metrics or KPIs around your security program, but these need to be customized to your organization. Metrics for a healthcare company’s success with SecOps will be very different from those for an ecommerce company. The healthcare sector should be primarily concerned about protected health information (PHI). Ecommerce companies will naturally be more focussed on credit card data at rest and in transit.

Regardless of your industry or unique requirements, identify the things that are most important for the business and start tracking those before you start making any changes to security. As you pick off the low-hanging fruit and climb the tree to higher-value actions, the metrics should demonstrate improvement.
We hope this playbook has given you some guidance for getting off the ground with a successful SecOps program. If you would like more information, check out the SecOps tag on our blog to access the many posts we have written about why this approach matters and why it works. And of course, we’re always here to answer questions, so feel free to tweet us @ThreatStack, or send us an email.

Select Resources

Check out these other DevOps, SecOps, and Security posts from our blog:

- How to Create a Security-Minded DevOps Organization: Three Best Practices
- DevOpsDays Chicago 2016: Dev, Ops, & the Role of Security
- How to Apply DevOps Culture to Security & Why You Should Do It
ABOUT

Threat Stack enables growth-driven companies to scale with confidence by identifying and verifying insider threats, external attacks, and data loss in real-time. The only fully integrated, cloud-native continuous monitoring solution that gives customers instant visibility and automatically responds to changes in their environment, Threat Stack provides the coverage needed to run secure and compliant, in all environments, without sacrificing speed and efficiency. For more information, or to start a free trial, visit threatstack.com.

CONTACT

1+ (617) 337-4270
threatstack.com
@threatstack
support@threatstack.com sales@threatstack.com

OTHER EBOOKS BY THREAT STACK

The Cloud Security Playbook
The Compliance Playbook for Cloud Infrastructure

LEGAL

Copyright © 2016, Threat Stack, Inc. Threat Stack Cloud Security Platform™ is a trademark of Threat Stack, Inc. All other trademarks, registered trademarks, or product names are the property of their respective owners.