DevSecOps: Another Side of Cybersecurity Culture

Steve Martino, Senior Vice President and Chief Information Security Officer

Companies are offering customers digital experiences where products and services are increasingly powered by mobile, cloud and data analytics capabilities. Developers in turn are moving to Development Operations (DevOps) processes to meet the need for greater agility and scale. To keep pace, CISOs are defining new approaches – building security in and working to shift security to a shared responsibility of the development and IT teams. That requires both technology and organizational culture change.

For more information visit trust.cisco.com.

A new practice of DevSecOps—bridging DevOps workflows with Information Security (InfoSec) Operations—helps drive those changes, blending constructs familiar to both groups. Here are a few tips on how to start a DevSecOps initiative for your organization:

Establish your DevSecOps foundation.

Using clearly defined guiding principles to drive security throughout the development process helps establish mutual trust among the Engineering, Operations and Security teams. This is also the point at which expectations for mutual accountability and high security standards are defined. The devsecops.org manifesto offers a great starting place. Their guidelines can be readily modified to fit your company's unique requirements.

Read the Manifesto →

Prove it out first.

Anyone with IT experience knows it's best to prove ideas manually before automating them. Consider running an Agile security hack-a-thon with participants from the Information Security and application teams to first configure the most important security requirements – what we at Cisco call the guardrails. Start by defining what your guardrails should be in the context of what platform you will use. Our first target environment was built on the Amazon Web Services (AWS) platform, so we defined 10 guardrails for our AWS accounts that fit Cisco's specific requirements. Do the same for your organization and its chosen platform. Then, conduct your hack-a-thon as you would for other Agile development efforts. Post-test readouts will help the entire team be knowledgeable and support users in true DevOps fashion.

Automate Your Guardrails.

Provide an easy way for your teams to apply the guardrails, such as at the time of new account provisioning. Develop simple scripting to retrofit those with existing accounts. This likely will require coordination among multiple teams – InfoSec, IT, Supply Chain, Procurement and possibly others. We achieved the security automation via our own tool we call the Continuous Security Buddy (CSB), which is built on several AWS services.

Continuously Validate.

As new resources are on-boarded or other changes occur, keep guardrails up-to-date with continuous security validation and real-time monitoring of security logs. Consider creating security "health reports" based on specific scoring or grading criteria to send to department tenants on a regular basis. That will empower tenants to address any critical security findings in a timely manner. The cycle of teams continuously integrating and deploying code while getting ongoing security assurance is the Holy Grail of Security!

Lessons Learned

At Cisco, our DevSecOps adoption and the subsequent security improvements actually exceeded our expectations. Within several weeks, our minimal viable tool ran in 72% of accounts hosting Cisco's Cloud offers; 97% of these accounts, on average, received a health score of A or B in their daily report, indicating a healthy security posture relative to the established guardrails.

The whole effort taught us some meaningful lessons:

Cloud is more about doing than telling.

Hack-a-thons enable cross-functional collaboration and deliver on critical security areas defined in the guardrails. They also provide great hands-on learning opportunities for everyone involved.

Timing Matters.

Timing initial launch with other key organizational efforts (like signing major cloud service agreements with a platform vendor) can carry exponential benefits. Try to integrate your efforts to other strategic initiatives.

Start Small and Grow.

Release minimal capabilities first, then iterate based on learnings and user feedback. Continuous visibility via those regular security health reports will enable teams to self-remediate issues and gain confidence in their offering's security posture. Scale over time as you learn more.

Guardrails vs. Just Pass/Fail.

The guardrail approach provides the range of compliance needed based on the situation at hand and allows teams to manage their risks.

Cultivate Partnerships.

Establishing key Operations partnerships with groups like IT, InfoSec, Procurement and Product Operations creates a multiplying effect where the aligned efforts help everyone move faster, and in the same direction.

Credibility built on Trust.

Be open and transparent regarding what you do with the access provided; be available for support if there are any issues. Consider setting up a central online site like a chat room to facilitate easy and fast interaction.

Skill-sets Matter.

Realistically, InfoSec practitioners don't code. Complement your team with other appropriately skilled resources to ensure successfully delivery of your DevSecOps principles and guardrails. The collective skills and knowledge will cross-pollinate.

Taking Risks.

DevSecOps is something new; it requires some risk-taking. Be patient but confident that it will pay off. Bringing teams together guided by a common goal is always a recipe for success.