The shift left in security culture

EXPERT ARTICLE

_DANIEL ESTERMANN
Product Marketing Manager Airlock
Ergon Informatik AG
_ROMAN HUGELSHOFER
Managing Director Application Security
Ergon Informatik AG

Published in SMART insights 2021 magazine

ergon

smart people – smart software®

Development, security and operations form one team to test ideas, quickly, and get feedback early on. They automatically incorporate security tools into every phase of the software development life cycle. The result is secure software that is as fast as Agile and DevOps. Security becomes both economiser and accelerator.

Companies are becoming increasingly agile and customer-centric so that they can respond more swiftly to meet new challenges. Customers today want the best services and features – and they want them to be continuously available, easy to use and secure. Needs will be no less great, the pace no slower and the complexity no simpler in the future. To perform to their best, companies must dismantle their silos and rethink their legacy processes.

Agile and DevOps have already brought significant enhancements to software development, as they make companies faster and more responsive. The next evolutionary phase here is DevSecOps, short for Development, Security and Operations. United and aligned, Agile and DevSecOps achieve their common goals of short deployment cycles and the best possible customer experience.

The power of three

DevSecOps is the natural and necessary progression in approaches to security in software development. It automatically incorporates cybersecurity into each stage, from ideation to integration; testing and deployment; and on to the release of the final software. DevSecOps is an expansion of DevOps. The two methods have their similarities, such as automation and continuous processes, to establish collaborative development cycles. But while DevOps prioritises delivery speed, with DevSecOps it is security that counts. Security is embedded from the start, thereby shifting to the left in the product-development cycle.

In the interests of historical perspective, it is worth pointing out that security used to be tacked on to software at the end of this cycle - almost as an afterthought and tested by an additional team.

That was still manageable in the old days of software updates once or twice a year. Then along came Agile and DevOps practices, targeting rapid releases within days or weeks. The downstream approach to security could no longer keep up.

The aim is to protect applications continuously from the word go, which means bringing security forward. Rather than tackling weaknesses and risks at the end, this approach monitors security right from the start and then throughout the software development process. Security professionals see this as a shift left for security along the timeline, from development to deployment.

Automatically secure

DevSecOps seamlessly integrates application security into Agile and DevOps processes. Security tools automate the provision of secure software without slowing down the development cycle. This anticipates security problems with proprietary or third-party software before they happen. For example, a scanner can check modules automatically for potential weaknesses whenever something changes.

It is generally faster, simpler and cheaper to fix bugs when they occur than to wait for a subsequent security test, or just before going live.

The easier to use, the better the adaption

Despite all due care and attention, security gaps may still be found after the software enters operational use. Incorporating third-party applications and opensource components makes it particularly susceptible to issues emerging or developing over time, necessitating fixes in the live system.

Security tools such as microgateways, which are used in development, enable developers to set security rules easily themselves not only while they are building the application, but also afterwards, when it is live. They do not have

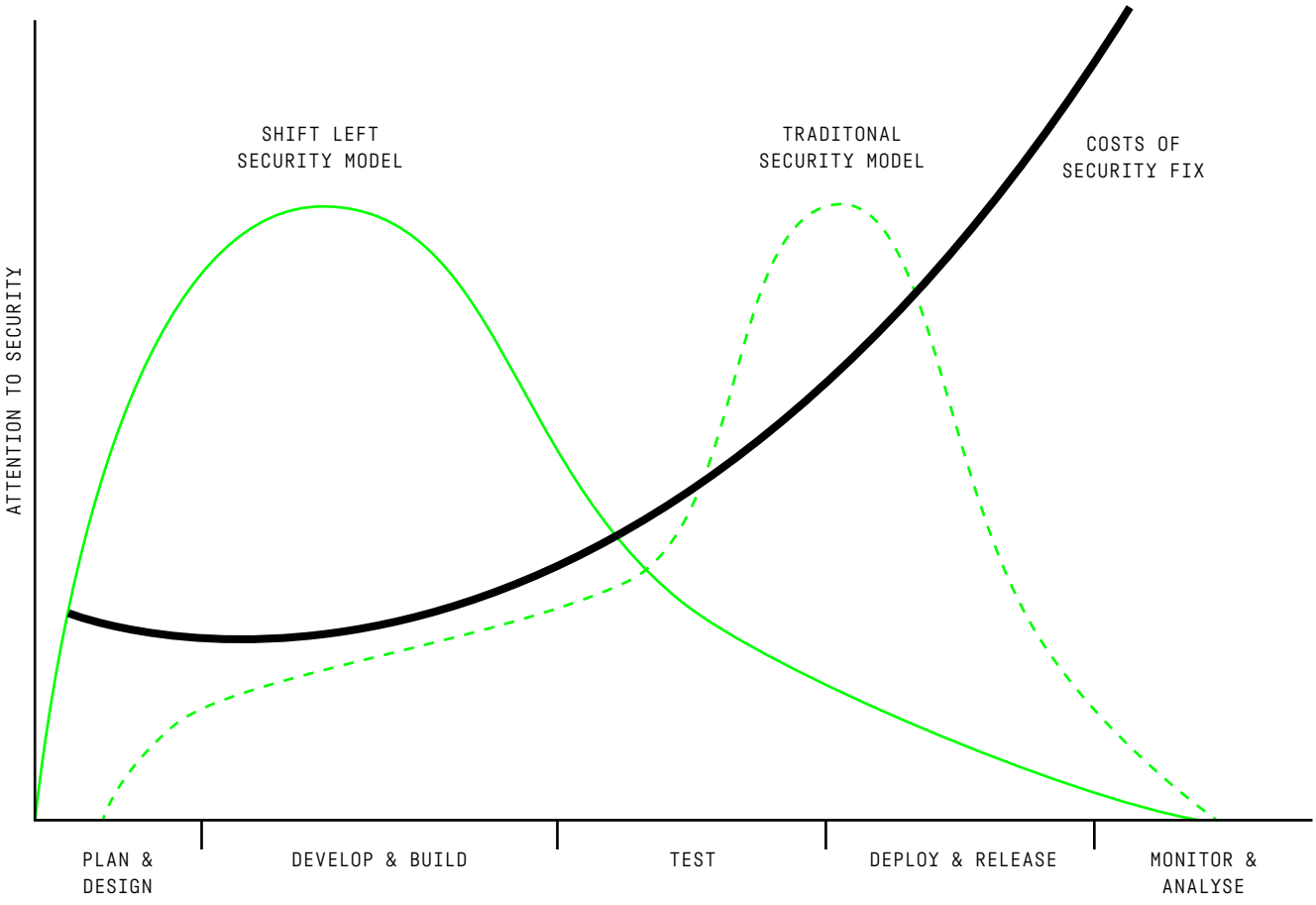
“The aim is to protect applications continuously from the word go, which means bringing security forward.”

_DANIEL ESTERMANN, PRODUCT MARKETING MANAGER AIRLOCK;
DANIEL.ESTERMANN@ERGON.CH

“United and aligned, Agile and DevSecOps achieve their common goals of short deployment cycles and the best possible customer experience.”

_ROMAN HUGELSHOFER, MANAGING DIRECTOR APPLICATION SECURITY, MEMBER OF THE EXECUTIVE BOARD;
ROMAN.HUGELSHOFER@ERGON.CH

Shift left – spotlight on security



Source: devopedia.org/shift-left

to rely on a security professional. That is important because in many cases security tools are not made with developer-friendliness in mind. The right tools improve all-round security-consciousness, which becomes an integral part of the software development process. Since budgets for security tools are often set by security teams, ease of use for developers is a crucial factor in their successful adaptation.

Double the protection, double the security

The DevSecOps approach ensures that security is integrated optimally into applications. It guarantees that cybersecurity keeps pace with the speed of innovation and it begins to build up a culture, not to mention cooperation, between development, security and operations teams. It is impossible to apply security universally throughout a company, however. There may be older unsupported legacy systems, third-party software modules or separate activities by other departments that are out of scope for development teams.

For as long as these legacy applications exist outside of the DevSecOps environment, or DevOps teams neglect to implement security fully from the ground up, conventional tools such as firewalls are still recommended to provide a second line of defence. The double defence tactic is a common one, as most modern organisations work with a mix of old and new IT. This is the case especially with movements such as open banking, in which banks with established legacy environments rely on fast, secure connections with third-party applications. The important thing is to plan moves like this as part of the software life cycle. In development, security rules must also reflect user

Common mistakes when implementing DevSecOps

MISTAKE	TIP
NOT ENOUGH STAMINA	WHERE EXPECTATIONS ARE TOO HIGH, DISAPPOINTMENT IS ALMOST INEVITABLE. DEVSECOPS IS A LONG ROAD AND THERE ARE NO SHORT CUTS.
TOP-DOWN APPROACH	DEVSECOPS CANNOT SIMPLY BE DECREED BY MANAGEMENT. AS WITH ANY CHANGE IN BEHAVIOUR OR AWARENESS, YOU NEED STRUCTURAL ADJUSTMENTS AND CONTINUOUS CHANGE MANAGEMENT THAT PAYS PARTICULAR ATTENTION TO CULTURAL FACTORS.
UNSTRUCTURED APPROACH	FIRST IDENTIFY THE RISKS, PRIORITISE ACTION POINTS AND SET REALISTIC INTERIM TARGETS. PROBLEM AREAS AND STICKING POINTS BETWEEN DEVELOPMENT AND SECURITY ARE THE IDEAL PLACE TO START. ELIMINATE OVERBEARING SECURITY PROCESSES WHEREVER POSSIBLE.
FAILURE TO RECOGNISE THE BENEFITS OF DEVSECOPS	USE STORYTELLING AND INCORPORATE EVERY IMPROVEMENT INTO THE BACKLOG IN THE FORM OF A SECURITY STORY, SIMILAR TO USER STORIES. THIS ALLOWS YOU TO PLAN IMPLEMENTATION AND, MOST IMPORTANTLY, MAKES IT VISIBLE TO ALL STAKEHOLDERS; CREATING TRANSPARENCY AND TRUST. DOCUMENTING CHANGES OF ROLE AND DETERMINING EXPECTATION ON BOTH SIDES IS VITAL TO CLEAR COMMUNICATION SO THE TEAM UNDERSTANDS ITS RESPONSIBILITY.
POORLY AUTOMATABLE SECURITY TOOLS	ENSURE THAT EVERYONE HAS THE TOOLS TO DO THEIR JOB. AUTOMATION IS ALSO GAINING GROUND IN SECURITY TOOLS, MANAGED VIA SCRIPT OR API. GRAPHICAL USER INTERFACES CAN MAKE IT EASIER TO GET STARTED BUT ARE NOT SUITED TO AUTOMATION.
EXCLUSIVE FOCUS ON ANALYSING CODE	APPLICATION SECURITY TESTING CAN IDENTIFY KNOWN ANGLES OF ATTACK AND WEAKNESSES AT AN EARLY STAGE. WEB APPLICATION FIREWALLS ARE STILL A MUST, HOWEVER, AS AN ADDITIONAL LINE OF DEFENCE AGAINST NOVEL OR UNKNOWN TYPES OF ATTACK. IN DEVSECOPS ARCHITECTURES, THIS FUNCTION IS INCREASINGLY BEING PERFORMED BY TOOLS SUCH AS THE AIRLOCK MICROGATEWAY. ITS SECURITY MODEL ENSURES THAT THE ONLY REQUESTS THAT ACTUALLY REACH THE APPLICATION ARE THOSE THAT THE DEVELOPERS HAVE EXPLICITLY CLASSIFIED AS VALID.

requirements at all stages. Security must offer maximum protection but users should scarcely be aware that it is there.

Automation saves costs
The initial investment in automating security should not be underestimated. Each significant change will slow down day-to-day operations at first and naturally also involve costs. That investment will pay off, however. It means fewer time-consuming manual checks and thus a lower error rate, while security and speed both improve.

It clearly makes long-term financial sense to prevent major security incidents and the resulting loss of reputation before they occur. The ability to recognise an attack and to act fast is crucial. In addition, DevSecOps creates a more agile system that can be started and updated more quickly. With the help of DevSecOps engineers, companies can automate their security infrastructures and thus simplify a highly technical, time-consuming and error-prone process.

People, processes and tools
The trinity of people, processes and tools is key to the success of DevSecOps. It takes a culture in which there is no “us” and “them”, just “us” and we all share responsibility for the security of our software. That may sound simple but it demands a whole new way of thinking.

The security team must believe that the developers want to write and deploy secure software. DevOps must in turn recognise that the security professionals are not there to always say “no” and put the brakes on innovation. Instead, their job is to protect companies from security violations and to act as coaches by helping development teams to set up automated security checks. This trains developers in secure programming and draws their attention to all of the possible attack scenarios. These new mindsets demand work, time and cultural change.

It is also worth noting that bought-in security tools are usually provided and approved by the security team under the security budget. If they are to be integrated into the DevSecOps process, they must satisfy more stringent security standards, their user-friendliness must be optimised and they must be customised to the needs of the developers and DevOps engineers. Security providers wanting to support DevSecOps must be aware of these requirements.

Here to stay
DevSecOps is now regarded as the state of the art in product development. Adaptation is still taking its time but, ultimately, the future will always belong to those bold enough to take it.

The two principal advantages are speed and security. Development teams deliver better, more secure, code and they do so quicker and thus at lower cost. DevSecOps makes the development, security and operations teams share responsibility for security. Its guiding principle is that, with the right tools and a shared focus on the user, software will become faster and more secure.

Far from putting the brakes on innovation, this new way of thinking can turbo-charge it. For DevSecOps to succeed, everyone must be aware that it is an interdepartmental endeavour.

By shifting security to the left, companies become more digitally responsive and better equipped for a digital future in the fast lane. />