SECURITY COMPASS WHITEPAPER

Software Security: An Enabler for Development Teams

Security Compass

In many organizations, software security is perceived as a roadblock. Tools that security teams recommend generate a lot of false positives which slows down the software development process. Furthermore, the severity reports generated from these tools is out of context and does not align with business risk thresholds.

Security teams, on the other hand, view things differently. They feel like they have inherited a mess they did not create. Given the small size of many security teams, it is difficult to try and respond to all of the problems.

The apparent challenge is trying to balance the need for both speed and security. Both sides are trying to achieve what is best for the organization. It is possible to reconcile the two sides. A good future state to aim for includes one where security teams are able to propose reasonable alternatives to the business. The recommendations represent risk from a business perspective. Security teams also collaborate with development and process teams. In fact, security is a part of software development.

Actionable steps for integrating security into development

Before any development effort kicks off, security teams should get involved. This means stepping in at sprint zero. Security should seek to understand the priorities of the upcoming sprints and try to build a mutual understanding of the need for both speed and security. Getting involved early provides insights for both security and development teams early.

1. In the requirements stage, use abuse and misuse cases at the requirements stage:

Many development teams may not know how to generate or interpret security requirements. Use cases are a common way for developers to understand functional requirements of a system. Extending this technique a little bit, security teams can introduce abuse and misuse cases to help developers think about security early.

A long-term objective is to capture these security requirements and make them available across multiple teams as common patterns. By facilitating requirements generation sharing across multiple development teams, security teams can help reduce the time it takes for each team to conduct this activity. This also leads to greater consistency and awareness of security.

2. In the design stage, use threat modeling as a way to generate options and a roadmap while helping architects consider alternative scenarios:

The goal of threat modeling is to uncover potential risks to a system. Working with an architect allows security teams to provide an additional perspective which will both train the architect to think about security and also reduce risk for the organization. This can lead to opportunities for developer acceleration by promoting security-hardened architectural components which can be reused by developers. Any new frameworks or technologies can go through a similar threat modeling exercise to drive possible options.

Further refinements and reduction of the load on architects can be done by considering reusable threat modeling insights or starting off from templates. This does not negate the need for a security expert, but serves as an accelerator in the initial stages of the process.

For non-critical systems, it may be sufficient to only consider publicly available threats and mitigations such as those created by MITRE in CVE, CWE, and ATT&CK repositories. This can reduce the time it takes to conduct security activities in the design stage.

3. Use security automation during the coding phase:

A lot of activity happens during the coding phase. Developers not only have to create code, but also test their code and make sure it properly integrates with an existing codebase. Assistive security tools like code scanners can help. As with many automation tools, they need to be configured properly to minimize false positives which slows down the developer. There will always be certain problems that code scanners are not designed to catch. Working with development teams to understand this can help minimize future rework.

Security teams can also help reduce developer load by writing security unit tests. Gradually building a suite of security unit tests that can be shared across teams will help retain the development speed and include security as a normal part of the development process. This type of paired programming or coaching technique can help facilitate a shift in mindset from speed versus security into one that includes both.

4. During the testing stage, recommend security testing frameworks that minimize noise:

Testing frameworks help speed up the testing process. A security team can make recommendations on which security testing frameworks can be used to help reduce the load on testing teams. Integrating these security testing frameworks into the testing process provides a hook into the broader quality assurance practices of the organization. Tapping into the quality assurance processes allows the business, compliance, and security teams to gain visibility into the security posture of software development without being disruptive. Metrics collected at the development level are rolled up into an assurance model for business stakeholders to make informed decisions.

5. At the end of a sprint, involve the business and facilitate a discussion with SDLC teams:

When it comes time to demonstrate the incremental work completed, security teams can help facilitate a discussion between development teams and the business teams such as business unit leads, compliance, and risk teams. This quickly aligns the different teams on next steps while retaining the notion of building security into each release.

Key principles to ensure security

When working with developers to build security into their process, security teams should keep a few things in mind.

1. **Look for a security champion:**
   Being able to scale security activities across multiple teams is difficult with the small team size of security experts. To build this capability across multiple development teams, security teams can work with development teams to designate a security champion. The role of a security champion is to integrate security activities into the development lifecycle workflows.

2. **Apply the right security process based on the type of project:**
   Not every project needs the same level of security rigor. Low risk projects, for example, will slow down developers if the risk to the organization is low. The goal should be to focus on those areas where the business risk is high enough to warrant a fuller set of activities. Taking this approach will help reduce the load across the development lifecycle.

3. **Speak the language used in the respective stage:**
   Every stage of software development has a certain tribal language. Security teams should inject their insights using the normative language in that domain. Forcing development teams to learn a completely new language is cognitively disruptive and difficult to assimilate into their daily workflows.

Build a security fabric through integration

Software developers must move very quickly today. They have to release features faster in order to help their organizations achieve competitive advantage. The activities above help to enable security within this context. There is one challenge, however, that still remains — integration.

Many development teams are operating in an isolated environment. While risk and compliance are important, they are not easily embedded into the value delivery of software development activities. One of the root causes of this is the lack of a security fabric across organizations. A security fabric exposes various development pipelines to business stakeholders. For example, a recommendation from threat modeling can tie into a compliance or regulatory requirement. This makes software development activities immediately relevant in a business context.

Balanced Development Automation tools and platforms help to integrate development activities with other domains like risk, compliance, and security. As such, they act as a strong enabler for achieving integration in the design of a security fabric. They also integrate the data collected from various development tools in order to provide a clearer picture of the security posture. For example, if a developer has completed their security unit testing, a code scanner can review the code and verify whether the work completed meets the security criteria. Both the developer and code scanner completion status are integrated to provide acceptance, rejection, or reconciliation activities.

Understanding between security and development teams

In the end, while many organizations struggle with integrating security practices into development processes, there is a way to reconcile the two and achieve both speed and security.

Security teams have an important role in facilitating a bridge between business and development teams by helping to bridge business concepts like risk with technical concepts like velocity. This shared understanding brings about a richer, shared understanding of security and risk concerns which align with business objectives. In doing so, security will not be an additional set of activities performed in the sidelines but is embedded into our development processes in a way that makes ongoing meaningful business contributions around security.

Automating the balance between speed and security will be key. While many tools exist, they do not integrate well together and the sharing of information is largely manual. Balanced development platforms help to collect and radiate the information in a meaningful way across multiple stakeholders. By including multiple perspectives in this way, there is much tighter alignment between the different teams and that helps us build more secure software.

Security Compass

Go Fast. Stay Safe.

Security Compass, a leading provider of cybersecurity solutions and advisory services, enables organizations to adopt Balanced Development Automation for rapid and secure application development. With their flagship product, SD Elements, the company helps automate significant portions of proactive manual processes for security and compliance that improves time to market for new technology. In addition, they offer advisory services on howorganizations can embrace emerging technologies like cloud to strengthen their security posture. Security Compass is the trusted solution provider to leading financial organizations, technology enablers, and renowned global brands. The company is headquartered in Toronto, with offices in the U.S. and India. Follow Security Compass on Twitter @securitycompass or visit them at securitycompass.com to learn more.

1.888.777.2211
info@securitycompass.com
www.securitycompass.com

@SECURITYCOMPASS
SECURITY COMPASS

OFFICES

GLOBAL HEADQUARTERS
1 Yonge Street
Suite 1801
Toronto, Ontario
Canada M5E 1W7

TORONTO
390 Queens Quay W
2nd Floor
Toronto, Ontario
Canada M5V 3A6

NEW JERSEY
621 Shrewsbury Avenue
Suite 215
Shrewsbury, New Jersey
USA 07702

CALIFORNIA
600 California Street
San Francisco, California
USA 94108

INDIA
#4.07
4th Floor, Statesman House
Barakhamba Road, New Delhi
India 110001