VERACODE

Software Security is a Team Sport

Become an Elite Organization with Secure Cloud Development Collaboration

Table of Contents

Introduction

Building and maintaining secure software is not a one-team effort. It requires the collective strength and collaboration of security, engineering, and operations teams.

By working together in harmony, these teams can achieve remarkable results, including **risk reduction, minimal tech debt, and secure cloud development at scale.**

This e-book is designed to guide you on this collaborative journey, empowering your organization to achieve elite status (based on research benchmarks) in the realm of security.

Let's explore the data-driven strategies, best practices, and tools that enable teams to work together seamlessly, fostering a culture of security and excellence.

It's time to play as a team and win the security game!

Understanding the Roles, Responsibilities, and Workflows

First things first, teams must understand the workflows and objectives of their counterparts. Misunderstandings and misalignments can lead to frustrations, increased security risks, and hindered progress in achieving organizational goals.

Security Team: Their Role, Goals, and Challenges

The security team plays a crucial role in safeguarding the organization's assets, data, and systems from potential threats. Their primary goal is to identify vulnerabilities, implement security controls, and ensure compliance with industry standards and regulations. However, they often face challenges such as limited visibility into the development process, difficulty deciphering which alarms going off are the most important, and the need to balance security with business agility.

Development Team: Their Role, Goals, and Challenges

The software engineering and development team is responsible for creating and maintaining software applications that power the organization. Their goal is to deliver high-quality code that meets functional and security requirements. However, they face challenges such as time constraints, pressure to meet deadlines, and a lack of available security expertise. Balancing the need for speed and security can be a delicate task for development teams.

Operations Team: Their Role, Goals, and Challenges

The operations team ensures the smooth functioning of the organization's infrastructure, systems, and applications. Their goal is to maintain high availability, scalability, and performance. They face challenges such as managing complex environments, handling incidents and outages, and maintaining security while implementing changes. The operations team plays a critical role in ensuring that the organization's systems are secure and operational.

Identifying Common Goals and Aligning Objectives

While each team has its own unique responsibilities and challenges, it is essential to identify common goals and align objectives to foster collaboration. By understanding the roles, goals, and challenges of each team, organizations can bridge the gaps and create a shared vision for security and development.

Common goals may include reducing security vulnerabilities, minimizing tech debt, ensuring compliance, and delivering secure and high-quality software. By aligning objectives, teams can work together towards these shared goals, breaking down silos and fostering a culture of collaboration.

To achieve this alignment, organizations can establish regular (and even informal) communication channels, such as cross-functional meetings and joint planning sessions. This allows teams to share insights, discuss challenges, and collaborate on solutions. Additionally, creating a shared understanding of the importance of security and its impact on the organization's success can help build a strong foundation for collaboration.

3 Tips for Building a Collaborative Culture

1. Gain and Document Buy-in to Develop Trust Among Teams

To foster trust among teams, it is crucial to gain buy-in from counterparts and stakeholders. This can be achieved by conducting testing and proof of concepts to demonstrate the effectiveness of security measures. Engage, solicit feedback, and respond to that feedback. Documenting this buy-in and sharing it with others can help avoid opposition and potential obstacles. Additionally, highlighting the work of specific teams or developers can raise awareness of security initiatives and earn positive recognition, further building trust and promoting collaboration.

2. Encouraging Cross-Team Collaboration and Knowledge Sharing

Collaboration should not be limited to within each team but should extend across security, development, and operations teams. Encouraging cross-team collaboration allows for a holistic approach to security and development, leveraging the expertise and perspectives of each team. Organizations can facilitate cross-team collaboration by organizing joint workshops, hackathons, or cross-functional projects. These activities provide opportunities for teams to work together, exchange knowledge, and learn from each other's experiences. By taking time to understand each other's workflows, you can add more value to the conversation and get more insights into effective collaboration.

3. Establishing a Shared Responsibility Mindset Sharing

To truly foster collaboration, organizations must establish a shared responsibility mindset. This means that security is not solely the responsibility of the security team but is a collective effort of all teams involved in the development and operation of software systems. By instilling a shared responsibility mindset, teams understand that security is an integral part of their roles and responsibilities. This mindset encourages proactive involvement in identifying and addressing security vulnerabilities throughout the development lifecycle. It also promotes accountability and ownership, ensuring that security is not an afterthought but a fundamental consideration in every decision and action. Organizations can reinforce this mindset by providing security training and awareness programs for all team members. This helps to build a common understanding of security best practices and the importance of integrating security into every stage of the development process.

Integrating Security into the Development Process

When security is integrated into each step of the software development lifecycle (SDLC), it becomes a shared responsibility. Developers, testers, architects, and other team members work together to identify and address potential vulnerabilities, implement secure coding practices, and conduct regular security code reviews and testing. This collaborative approach ensures that security is not an afterthought but an integral part of the development process. Here's how to integrate security into the SDLC in 6 steps.

Step 1: **Discover and Assess Risks**

Step 2: **Establish Prevention Methods**

Step 3: **Onboard and Scale Apps**

Step 4: **Set Policies**

Step 5: **Prioritize and Address Findings**

Step 6: **Leverage Reporting and Analytics**

Step 1:
Discover and Assess Risks

- Identify the applications in your portfolio and understand their owners and locations.
- Determine open-source dependencies and assess the use of AI in code generation.
- Consider different application types, elements, development tools, and risk levels.

Step 2:
Establish Prevention Methods

- Implement security controls in the development process to prevent security flaws.
- Utilize code testing tools like Static Application Security Testing (SAST), Container Security, and Software Composition Analysis (SCA).
- Leverage AI-assisted remediation that's responsible-by-design for fast and accurate vulnerability remediation.
- Invest in developer education and training to increase security awareness.

Step 3: Onboard and Scale Apps

- Automate security scans and integrate them into the development process.
- Establish continuous scanning for both legacy and cloud-native applications.
- Gain visibility into security posture and establish a baseline.

Step 4:
Set Policies

- Define and document security policies for applications based on risk tolerance, regulatory requirements, and criticality.
- Establish technical controls to monitor compliance with policies.
- Use security policy recommendations and customize them based on business criticality.

Step 5:
Prioritize and Address Findings

- Categorize and resolve policy-violating flaws through remediation or mitigation.
- Test both first-party and third-party code throughout the SDLC.
- Address security flaws promptly to reduce critical security debt.

Step 6:
Leverage Reporting and Analytics

- Use reporting and analytics to track progress and measure the effectiveness of security measures.
- Establish a unified reporting system through a tool like Longbow to make sense of data from different tools and systems.
- Identify areas for improvement and set quantitative goals.
- Demonstrate compliance with regulatory requirements

By following these six steps, organizations can secure the SDLC, optimize the developer experience, and build a culture of security awareness. Integrating security into each step ensures all teams are playing together in the game of security.

Measuring Success and Demonstrating Risk Reduction

As organizations strive to integrate security into the development process, it is crucial to measure the success of these efforts and demonstrate the reduction in risk. By defining key performance indicators (KPIs) for security, development, and operations, organizations can track progress and align their security practices with their overall goals.

Additionally, establishing metrics to measure risk reduction and technical debt reduction provides quantifiable evidence of the effectiveness of security integration.

To effectively communicate these results, organizations should create dashboards and reports that provide a clear overview of the metrics and KPIs, enabling stakeholders to understand the value of security efforts and make informed decisions. Using a unified platform for your testing solutions makes this step much easier.

Defining KPIs for Security, Development, and Operations

To measure the success of security integration into the development process, it is essential to define KPIs that align with organizational goals. These KPIs should encompass security, development, and operations aspects. Examples of security KPIs include the number of vulnerabilities identified and remediated, the percentage of code coverage by security tests, and the time taken to patch critical vulnerabilities. Development and operations KPIs may include metrics such as deployment frequency, mean time to recovery, and customer satisfaction.

Establishing Metrics to Measure Risk Reduction and Technical Debt Reduction

In addition to KPIs, organizations should establish specific metrics to measure risk reduction and technical debt reduction. These metrics help quantify the effectiveness of security practices and the progress made in reducing vulnerabilities and improving the overall security posture. Examples of risk reduction metrics include the number of critical vulnerabilities mitigated, the percentage of security controls implemented, and the reduction in Mean Time to Remediate (MTTR). Technical debt reduction metrics may include the decrease in the number of known vulnerabilities, the reduction in the time spent on security-related rework, and the improvement in code quality.

Creating Dashboards and Reports to Track Progress and Communicate Results

To effectively track progress and communicate the results of security integration efforts, organizations should create dashboards and reports that provide a clear overview of the metrics and KPIs. These visual representations enable stakeholders to easily understand the current state of security, development, and operations. Dashboards can include real-time data on vulnerability trends, risk reduction, and technical debt reduction. Reports can be generated periodically to provide a comprehensive analysis of the progress made, challenges faced, and recommendations for further improvement. By sharing these dashboards and reports with relevant stakeholders, organizations can demonstrate the value of security integration and the reduction in risk achieved. This step is much easier when your testing solution is an integrated platform.

Becoming an Elite Organization

How do we determine what makes an organization “elite”?

We use data from the State of Software Security 2024 report that analyzed over one million applications across thousands of organizations. This extensive analysis provides us with valuable insights and benchmarks to identify elite organizations in terms of their security practices.

The report reveals that security debt exists in 42% of applications and a staggering 71% of organizations. Even more concerning is the fact that 46% of organizations have persistent, high-severity flaws that constitute ‘critical’ security debt. In a sample of organizations with greater than 7 applications, only 10% of organizations show no security debt. Here’s what we found out about the practices of these elite teams.

Remediation Speed

The analysis reinforced the importance of speed in flaw remediation. Development teams that address flaws promptly can reduce critical security debt by a remarkable 75%. By fixing vulnerabilities quickly, these teams build habits and muscle memory around fixing security flaws, significantly enhancing their security posture and reducing the prevalence of security debt in their applications.

Developer Education

It shows us that among organizations using immersive developer education, Security Labs, 37% have security debt. Compare that to 48% among application teams that don't. The time-to-fix difference is even more significant. Applications developed by teams that aren't using the Labs take seven months longer to reach that 37% mark.

Additional Factors

Additionally, programming languages, first-party vs third-party code, and application age all play a significant role in the accumulation of security debt, though the findings are complex. Read the full report to discover the nuances of these factors.

> **Why Veracode? It's the whole package. It's not only the technology; it's the processes created through the service, the workshops, and the help from highly experienced security professionals.**

-Darius Schaper, Enterprise Architect, HDI Global SE

Conclusion

Security is not the responsibility of a single individual or department but requires the collective effort of all stakeholders. By fostering a culture of collaboration, open communication, and shared responsibility, organizations can effectively address security challenges and mitigate risks. Collaboration enables knowledge sharing, promotes best practices, and ensures that security is integrated into every aspect of the organization's operations.

By working together, organizations can build a strong defense against potential threats, reduce vulnerabilities, and achieve elite status in their security practices. Remember, security is a team effort, and by embracing this mindset, organizations can create a secure and resilient environment for their software development and cloud infrastructure.

Veracode's world-class platform enables you to establish continuous security around your legacy and cloud-native applications. We support you in seamlessly integrating and automating security into your entire SDLC, bringing security and development together, and providing a vehicle to define and implement a set of security policies that align to the business criticality and operating environment of software in production. With Veracode solutions, support, and services, you can avoid and overcome the challenges of securing your software from start to finish.

Click here to schedule a demo of Veracode today and let us show you how easy we can make the team sport of security for your organization.

About Veracode

Veracode is a global leader in Application Risk Management for the AI era. Powered by trillions of lines of code scans and the proprietary AI-generated remediation engine, the Veracode platform is trusted by organizations worldwide to build and maintain secure software from code creation to cloud deployment. Thousands of the world's leading development and security teams use Veracode every second of every day to get accurate, actionable visibility of exploitable risk, achieve real-time vulnerability remediation, and reduce their security debt at scale. Veracode is a multi-award-winning company offering capabilities to secure the entire software development life cycle, including Veracode Fix, Static Analysis, Dynamic Analysis, Software Composition Analysis, Container Security, Application Security Posture Management, and Penetration Testing.

Learn more at www.veracode.com, on the Veracode blog, and on LinkedIn and Twitter.

Copyright © 2024 Veracode, Inc. All rights reserved. Veracode is a registered trademark of Veracode, Inc. in the United States and may be registered in certain other jurisdictions. All other product names, brands or logos belong to their respective holders. All other trademarks cited herein are property of their respective owners.