BLACKDUCK®

Making It All Work

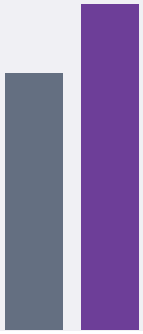
A Practical Guide to Operationalizing the Modern AppSec Framework

Making It All Work

A Practical Guide to Operationalizing the Modern AppSec Framework

Organizations need to develop and deliver secure applications fast. Unfortunately, this need for speed can leave those applications vulnerable to attack. The traditional software development life cycle (SDLC), with its mishmash of groups, orgs, teams, development sections, and so on, no longer works—it creates functional silos between DevOps and SecOps. And it lacks the holistic approach required to solve problems based on business outcome.

Average time to fix high-severity vulnerabilities grew from **194** days at the beginning of 2021 to **246** days at the end of June.



Remediation rates for critical-severity vulnerabilities fell from **54%** at the beginning of 2021 to **48%** at the end of June, and remediation rates for high-severity vulnerabilities decreased from **50%** to **38%** during the same period.

The COVID-19 pandemic required many organizations to very quickly undergo an unprecedented shift to an online business model and remote workforce. This shift led to untested applications being introduced before legacy applications were secured. And in the absence of a streamlined, modern application security (AppSec) program, keeping on top of security became overwhelming for many organizations.

The Modern AppSec Framework provides a better way.

The Modern AppSec Framework

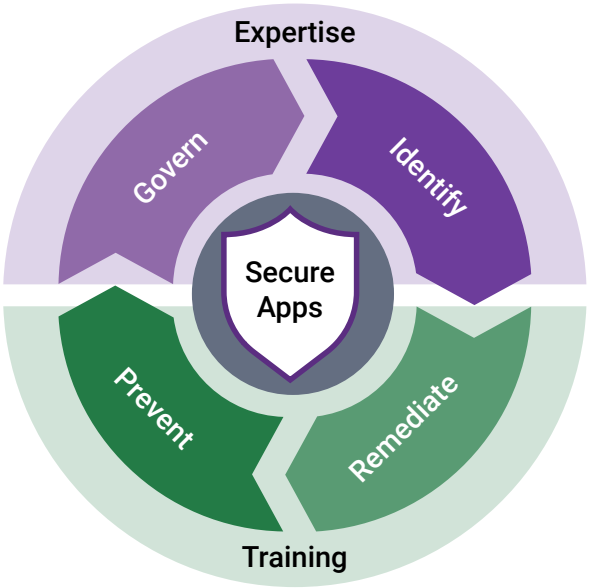
The Modern AppSec Framework is a functional plan that organizations can use to develop and deliver secure applications, regardless of their appetite for risk and where they are in their security or application development journey. It transforms traditional models into four components that focus on business outcomes and correlates those business outcomes with tactical measures and products that can be adopted across an organization.

Govern

- ✓ Asset management and risk ranking
- ✓ Compliance
- ✓ AppSec program best practices
- ✓ AppSec program metrics

Prevent

- ✓ Findings-based training
- ✓ Threat modeling
- ✓ Security frameworks
- ✓ Programming scaling



Identify

- ✓ Attack surface mapping
- ✓ Continuous and automated scanning
- ✓ Point-in-time testing
- ✓ Near-zero false positives

Remediate

- ✓ Verified actionable results
- ✓ Digital acceleration
- ✓ Productivity increase
- ✓ Staff augmentation

As with any new approach, the challenge often arises once an organization begins to marry it with existing teams and processes. So where does an organization begin to operationalize the Modern AppSec Framework in its own processes and structure? It starts with identifying where its program is today, and where it needs to be tomorrow.

Before adopting any new program, organizations should first perform a realistic audit of their current program to account for the tools being used, who is using them, and what processes are currently in place. This discovery process will help establish a starting point from which an organization can begin to operationalize the Modern AppSec Framework. Following this discovery phase, organizations can begin to work through the fundamental challenges associated with each component of the framework and apply an actionable roadmap to take their application security program to the next level.

In this ebook, we will break down each component independently and directly correlate its unique challenges with a roadmap for overcoming them. We will also highlight several key considerations that are critical to the successful implementation of the Modern AppSec Framework, and ultimately, an application security program that empowers both security and development teams to deliver secure applications and APIs at the speed of modern development cycles.

Govern

Organizations must be able to govern their application security program. Overarching factors to consider when defining governance for the program include application asset management and risk ranking, regulatory compliance, best practices, and outlining program metrics proactively to demonstrate success over time.

This table outlines the steps of defining an AppSec program governance and how to perform them.

Step	Description
Inventory Assets	Usually, an organization's security program is based on its knowledge of its own assets, regulatory compliance requirements, and risk ratings. An ideal starting point is to document all assets currently being managed.
Audit Data	Once an organization has an inventory of their assets and how they are being managed, the next step is determining which technology will be used to perform a thorough and exact audit of its data . It remains a fact that to make good decisions, one must have a good source of data. By making data-driven decisions, organizations can view security not as an ad hoc process, rather a scalable, repeatable, and measurable program.
Define Processes	Governing the flow of work between DevOps and SecOps teams will ultimately determine the success or failure of any security program. Identifying the redundancies, bottlenecks, and key touchpoints within this flow will enable organizations to improve their processes. To start, identify a program manager who will act as a solution architect for the organization and closely assist with program integrations. This individual will oversee the AppSec program and its goals, and create an action plan, complete with benchmarks and deadlines, to measure the program's implementation and success.
Set Goals	An organization at this point should be ready to mature its security program. Establishing goals and setting metrics to track progress, as well as implementing best practices to prevent a roll back, will ensure that the organization stays on the path toward a successful framework.

Identify

To ensure that their applications are secure, organizations must first identify security issues in their production environment. These will generally fall into one of two categories: bugs and flaws. Bugs are often identified as code-level security issues, whereas flaws are largely design-level security issues. Regardless of category or the type of application being tested (mobile, API, web application), organizations should be able to know exactly what they have and find security issues—fast.

Continuous and Automated Scanning/Testing

Integrating continuous and automated security scanning and testing into the processes of both DevOps and SecOps is crucial to evaluating and improving the security posture of an organization. Ensuring that only high-fidelity data is shared will improve the security team's credibility with DevOps, and also aid in securing buy-in when establishing new integrations and processes. Continuous and automated scanning enables organization to identify areas of risk due to application vulnerabilities.

Organizations with less mature application security programs can use also automated scanning as a safety net. For example, DevOps might roll back a website and inadvertently reintroduce vulnerabilities. Automated scanning helps ensure that these are caught and the organization is not put at risk.

Scanning also enables new vulnerabilities to be found faster. Organizations can include new vulnerabilities in subsequent scans to ensure constant coverage. If organizations perform checks only whenever there is a new release, they risk being vulnerable until the next release cycle.

Point-in-Time Testing

Vulnerabilities such as business logic flaws that cannot be found by automated testing can be identified by a manual assessment. This type of manual point-in-time testing is intended to complement automated testing with human interaction so that vulnerabilities are identified with context. These manual assessments search for vulnerabilities based on established industry standards set by organizations such as OpenSANS and OWASP.

This table outlines the steps of introducing new AppSec tools and provides some best practices for how to make the process as smooth as possible.

Step	Description
Onboarding	The most effective way to introduce a new security tool is to first have a clear understanding of where the organization stands in terms of its security structure. Additionally, identifying the key users and setting goals will help teams evaluate their performance and overall progress.
Risk Discovery	An organization typically owns thousands of digital assets, each of which needs to be accounted for and scanned. After all, you can only secure what you know you own. Attack surface mapping is a critical component in accurately configuring continuous and automated scanning tools. Traditional risk discovery operations (such as "spidering") can also be automated; however, manual intervention is almost always required to scan assets that do not have a direct entry point.
Setup and Configuration	An organization's ability to effectively and efficiently utilize a scanning solution depends on its team and on the technology. The solution needs to be reliable for both DevOps and SecOps and help streamline communication and eliminate redundancies. Choose a solution that is intuitive and easy to configure.
Integration	Once the organization has the solution configured correctly, the data it provides needs to go to the right people—and in the right format. It's vital to define how the data being produced is • Reaching the team in the right format, through the right technology (Jira, Rally, Basecamp, etc.) • Contextualized for clarity (i.e., where is the vulnerability?) • Providing action items (how to fix it)
Reducing False Positives	One of the major friction points between DevOps and SecOps is the number of false positives in the data. The DevOps team often does not have the bandwidth to go through the thousands of vulnerabilities identified by SecOps. On the other hand, the SecOps team might be smaller and therefore not have the bandwidth to scrub all the results. But in this age of speed, providing clean data is essential. Establishing solid touchpoints within the organization to interact with the scanning solution and ensure the data is validated is key. And documenting these processes will provide beneficial information for others as well.
Production-Safe Testing	A continuous and automated testing solution that is not production-safe defeats the purpose of the tool—because breaches happen in production, not in development. And continuous testing can identify security flaws that might slip through the cracks in a normal development process.

Remediate

Once an organization has generated a comprehensive mapping of its attack surface and identified all security vulnerabilities, it needs to develop a process to keep track of what has been tested, by what means, and when. This will enable security teams to prioritize what needs to be fixed and understand what issues matter most. And it will enable development teams to identify and fix prioritized vulnerabilities first.

The process of vulnerability remediation has several areas that need to be managed. This table outlines those areas and provides some insights for managing them.

Area	Description
Data Transfer	The process of remediating vulnerabilities needs to be seamless. Depending on the maturity level of an organization's vulnerability management program, SecOps might not have the budget or the bandwidth to go through and separate the true positives from the false positives. Providing the right feedback with high-fidelity data at the right time is critical.
False Positives	False positives introduce additional work (and sometimes delays) for DevOps teams. The volume of false positives can increase the noise for teams and may also frustrate already-swamped teams with unproductive tasks, eventually making them insensitive to all information altogether. To help eliminate this, organizations should establish a method of validating true positives so that DevOps can focus on securing code without the burden of fact checking data.
Bandwidth	Injecting security into systems thinking is not a fun exercise. Continuous learning and developing is challenging when teams are lean to begin with. Managing bandwidth to allow improvements is an investment worth making as an organization increases its security acumen.
Expertise	Technical advice on how to deal with a security problem, especially in application security, is valuable. Finding and establishing a process to handle an issue (without taking away bandwidth from other resources) is an important step to programmatically implementing the Modern AppSec Framework.
Proof of Concept	Establish that the solution works within the context of the organization, its processes, and its program. A proof of concept is an extremely valuable resource for organizations looking to adopt new solutions for remediation.

Prevent

The final step of the framework is the prevention. At this point in its maturity, an organization can now determine how to scale its efforts across the entire application security program through a combination of training, threat modeling, security frameworks, and application security solutions.

There are several ways to help prevent vulnerabilities from getting into your code, as outlined in this table.

Method	Description
Relevant Training	Findings-based training should be specific to both an organization and the technology. Ideally, organizations should try to identify vulnerabilities by classes. Then, if an organization has a persistent vulnerability across its assets, it can make better business decisions about which training to prioritize for development teams.
Prioritization	Understanding the high-risk vulnerabilities that affect an organization will help formulate what the team needs to be trained on first. Understanding vulnerability criticality and the practice of risk rating will help organizations understand what business-critical vulnerabilities affect its system.
Security Frameworks and Configuration Standards	Sometimes organizations are not aware of third-party packages and components that their developers are using. The SolarWinds attack is a textbook example of this point. Today's threat landscape has evolved, and the frameworks an organization uses can often be the source of a breach. Having a comprehensive and up-to-date inventory of all the packages and third-party components used across the organization will be helpful in ensuring the security of both those solutions and the organization as a whole.
Problem Approach Methodology and Standardization	Organizations should have standardization criteria for all aspects of their application security program and its processes. For example, an organization could develop and deploy a standard library to validate inputs and identify the exact format of settings used to call that library, and the criteria that team members should use to evaluate the business context. Similarly, organizations should have a template library to standardize output encoding.

Making It All Work

Very little of implementing the Modern AppSec Framework is going to be technically challenging. Rather, it is going to be more of a human and organizational challenge. While determining how to scale efforts across an AppSec program might seem daunting at first, organizations that adopt the Modern AppSec Framework will soon realize the full potential of their application security program.

Black Duck offers several solutions to help organizations mature their application security program and bridge the communication gap between SecOps and DevOps. By providing trustworthy data, organizations can make informed decisions, reduce their risk, and view security not just as an ad hoc process, but rather a scalable, repeatable, and measurable program. Black Duck provides solid, actionable methods that organizations can use to engage DevOps teams and improve security frameworks, while supporting the current skill level within an organization's current infrastructure. Black Duck helps organizations map out a maturity plan for their current application security program and the next steps required to reach their overall security goals.

Regardless of where an organization is on its application security journey, Black Duck can help by delivering the right solutions, services, and expertise that will help it achieve its business goals and drive adoption of a successful application security vision across the entire organization.

About Black Duck

Black Duck® offers the most comprehensive, powerful, and trusted portfolio of application security solutions in the industry. We have an unmatched track record of helping organizations around the world secure their software quickly, integrate security efficiently in their development environments, and safely innovate with new technologies. As the recognized leaders, experts, and innovators in software security, Black Duck has everything you need to build trust in your software. Learn more at www.blackduck.com.

©2024 Black Duck Software, Inc. All rights reserved. Black Duck is a trademark of Black Duck Software, Inc. in the United States and other countries. All other names mentioned herein are trademarks or registered trademarks of their respective owners. August 2024