The events of the past two years have accelerated a near-complete digital transformation for today’s enterprise organizations. The COVID-19 pandemic caused a seismic economic shift. Organizations recognized that in order to stay in business, they needed to quickly pivot almost every aspect of their business online, evolving processes and adopting new technology so that they could succeed in the “new normal.” With a new remote workforce and the coming evolution to a hybrid work model, savvy leaders recognize a competitive advantage in being proactive about application security (AppSec).

However, this accelerated digital transformation has not been without its challenges. As businesses moved quickly to adopt new paradigms and deliver new applications, the security of those applications once deployed has continued to elude even the most mature organizations. The biggest names in business are now measuring their window of exposure (a key metric that indicates the exposure to cyber breaches for software applications) in months, not days.

Organizations need to deliver new applications and APIs quickly. Unfortunately, this “need for speed” can lead to vulnerabilities in software code that can’t be identified until the apps are in production. And that means developers must stop working on new projects to fix the vulnerabilities in released applications.

SecOps and DevOps teams both face roadblocks, because each functional group operates with its own processes, tools, and KPIs. To create a process for developing and deploying secure applications, organizations need to move beyond traditional methodologies and adopt a new approach—one that bridges the gap between security operations and development.

A modern AppSec framework enables organizations to develop and deliver secure applications, regardless of where they are in their security or application development journey. The framework simplifies traditional models into four components that focus on business outcomes and correlates those business outcomes with tactical measures and products that can be adopted cross-functionally. The result is an application security program that empowers both security and development teams to create and deliver secure applications and APIs—fast.

Outcomes and considerations

As organizations focused on the unprecedented shift to an online business model and remote workforce, they also quickly adopted and developed new applications to meet the demands of an online marketplace and workforce. Unfortunately, the focus on speed exacerbated the risk of deploying untested applications. The Synopsys white paper “DAST to the Future” examines the risks associated with siloed programs as well as the fundamental problems that come with a lack of visibility and testing.

“It doesn’t matter if developers ship flawless proprietary code and apply the latest security patches to each open source component—which in 2020, netted-out to an average of 528 per application. Many vulnerabilities found in a production application do not exist in its source code and only arise once deployed into production.”

—“DAST to the Future,” Synopsys, 2022
While these challenges and statistics may seem daunting, it is important to remember that the problem exists because old models were being used to solve very modern problems. As with all things in cybersecurity, there is no silver bullet. It is only by unlocking the full potential of a comprehensive AppSec program that organizations can develop and deploy truly secure applications. So where is an organization to start if it wants a digital future that is free from breaches? At the beginning, of course.

Whether starting an application security program from scratch or maturing a current program, organizations should first establish the goals and desired business outcomes for the program. The first step is to take a strategic look at why it wants to implement an application security program and where limitations exist. Some reasons might include

- **Sales / competitive differentiation:** As today’s organizations face the realities of security breaches such as the recent SolarWinds attack, more customers and businesses are realizing that they are only as secure as the businesses they are connected with. Some potential customers and partners may want to discuss application security processes before agreeing to do business. As more organizations adopt formal application security programs, the ability to secure software may become a competitive differentiator.

- **Compliance:** Many compliance regulations, such as PCI, HIPAA, and GDPR, include application security components. Additionally, a [2021 U.S. executive order](https://www.whitehouse.gov/presidential-actions/executive-order-improved-cybersecurity) outlined a number of standards and requirements that will be adopted to protect federal agencies and their software infrastructure.

- **Risk of attack:** When determining goals for an AppSec program, it is important to remember that breaches exist not only because of vulnerabilities in code. Developers might create completely secure code only to see the application compromised because of how it interacts with other systems. Testing should be planned accordingly for apps at higher risk due to being public-facing or containing sensitive data.

- **Expertise:** All the models and strategies in the world won’t matter if an organization doesn’t have the expertise to put them into practice. A successful application security program requires the right talent and resources, along with the right solutions to run the program. Each functional team needs dedicated resources and business-specific metrics and KPIs. A successful application security program requires
  - Speed and accuracy, and comprehensive testing across the entire application portfolio
  - Access to the right application security talent that can perform high-fidelity, on-demand, manual testing
  - Strong integration with development processes and tools so that security vulnerabilities are fixed faster and prevented in the future

Organizations can take a holistic approach to their AppSec program only when they fully understand their desired business outcomes—and their limitations.

Organizations can take a holistic approach to their AppSec program only when they fully understand their desired business outcomes—and their limitations.

Synopsys offers a modern approach to application security, where security, strategy, and processes are embedded throughout the software development life cycle (SDLC) as part of a programmatic framework that encompasses people, processes, and solutions, and results in successful business outcomes.

**The modern AppSec framework**

Application security is about developing, securing, and maintaining secure software. Gone are the days when this was the job of the security team alone. But it's not the job of the development team alone, either. Today's organizations need to take a holistic and programmatic approach in which application security empowers all teams to work cohesively, and is not siloed by job function. To achieve this, organizations need to evolve past the traditional SDLC model and adopt a fresh approach: A modern AppSec framework.
There are four functional components to a modern AppSec framework: govern, identify, remediate, and prevent. Each component focuses on the outcomes of that function. Once each component is complete, the organization then moves to the next functional component, operating in a cyclical process that leverages training and expertise as the organization evolves.

**Govern**
To ensure its application are secure, organizations must be able to govern the application security program. Important aspects of governance include application asset management and risk ranking, regulatory compliance, and AppSec best practices and program metrics that demonstrate program success over time.

**Asset management and risk rating**
You can’t secure software that you don't know you have. Organizations can start by identifying all web applications, mobile applications, APIs, and cloud applications. Additionally, it's important to create a software Bill of Materials (SBOM)—an up-to-date, comprehensive list of all software components, their dependencies, software versions, and open source.

Next, assign a risk ranking to each piece of software. Consider criteria such as business criticality, data type, and accessibility when assigning the ranking. Important and high-risk applications should go through more (and perhaps different) security activities than less important or lower-risk applications.

**Regulatory compliance**
Many organizations are subject to security requirements because of an application's business function (i.e., payments), type of data stored or processed (i.e., healthcare data), or geographical location (i.e., regional requirements for data privacy and protection). Others may be required to perform specific security activities due to contractual obligations. Some compliance requirements might include certain security controls. Organizations should be aware of each of these and make sure they are meeting the appropriate standards.

**AppSec best practices and metrics**
Best practices in application security are not one-size-fits-all. Standards and controls have been developed after years of practical security experience in real organizations. Organizations need to know how to optimize their own unique program utilizing defined and measurable metrics. For example, security metrics can optimize future application security processes and help answer questions, such as whether to employ penetration testing, using evidence-based information instead of opinion or anecdotes.

**Identify**
Organizations need a way to identify security issues in applications regardless of development methodology—waterfall, agile, DevOps, etc. Security issues generally fall into two categories: bugs and flaws. Bugs are often identified as code-level security issues, and flaws are often categorized as design-level security issues. Regardless of category or the application type (mobile, API, web application), organizations need to find security issues—fast. Some ways to do that include mapping the attack surface, automated and continuous scanning, point-in-time testing, and reducing false positives.

Today’s organizations need to take a holistic and programmatic approach in which application security empowers all teams to work cohesively, and is not siloed by job function.
One of the most prevalent challenges that today’s organizations face is compiling an accurate account of their entire attack surface of exposed assets.

**Attack surface mapping**

The cybersecurity maxim rings true: “You can’t secure what you can’t see.” One of the most prevalent challenges that today’s organizations face is compiling an accurate account of their entire attack surface of exposed assets. So it is crucial that before testing their applications, organizations create a comprehensive map of externally facing assets to ensure that security vulnerabilities in unknown applications do not go unchecked.

**Continuous and automated scanning**

Web application attacks are one of the greatest risks to an organization. Web apps may not be updated regularly or often and may be attacked, changed, or updated with a new feature that was not carefully vetted by security controls.

Web security is always evolving, and organizations need to ensure that their scanning and testing evolves along with it. Continuous production site monitoring and regular security scans provide faster awareness of changes and updates, and increase application security measures.

**Point-in-time testing**

While continuous scanning provides automated testing for potential vulnerabilities, point-in-time testing assesses application security by emulating a real-world attack. When used together, an organization can identify vulnerabilities that exist in the code of an application, while also testing the security of that application as it interacts with other systems. This reduces the risk of attack from multiple vectors.

**Reducing false positives**

Ensuring that your security scans are accurate is crucial. False positives can inhibit productivity and introduce distrust across functional teams. It is therefore crucial that every alerted vulnerability is checked individually to ensure it is not a false positive.

**Remediate**

Once an organization has a comprehensive mapping of its attack surface and identified security vulnerabilities, it needs to develop a process for keeping 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. This will also enable development teams to identify and fix prioritized vulnerabilities first.

For example, many organizations require that issues are fixed within a certain period of time, depending on the criticality of the findings. An eCommerce business might require that critical issues discovered on its customer-facing applications be fixed within 48 hours, high-severity issues be fixed within 10 days, medium-severity issues be fixed within 30 days, and low-severity findings be fixed within 90 days.

By tracking which vulnerabilities are open, which have been addressed, and which are closed, organizations can ensure clear communication and tracking among functional teams. Additionally, summary information can be reported to relevant stakeholders, so everyone is always aware of current status. By developing a process for identifying, tracking, and remediating vulnerabilities in production applications, organizations can increase productivity by fixing vulnerabilities earlier and releasing applications faster.
Developing a process for remediating attacks may seem straightforward, but not all organizations have the expertise, processes, and resources to tackle it on their own. Synopsys offers both in-house services and partnerships that help organizations prioritize and remediate application security vulnerabilities while enabling and developing expertise and best practices.

**Synopsys professional services on-demand and premium support**

Synopsys support includes accelerated and prioritized response times, 24x7 escalation for critical issues, and personalized engagement with a technical account manager to ensure that issues are resolved quickly.

**Staff augmentation:** Organizations can take application security one step further by augmenting their in-house security teams. Joint customers have access to hundreds of application development experts who scan the identified vulnerabilities and implement the fix, saving time and resources.

**eLearning:** Educate your teams through self-paced training on application threat modeling, best coding practices, mitigation, and defensive remediation, all in an easy-to-learn, web-based environment. The information learned in these modules can be used to produce guidelines for consistent secure programming practices throughout your organization.

**Prevent**

Once an organization has determined what is important (the govern component), what needs to be fixed (the identify component), and how it’s going to fix it (the remediation component), the next step is to determine how to scale efforts across the application security program in its entirety.

This is achieved by a combination of training, threat modeling, adhering to security frameworks, and operationalizing application security solutions. It is important that the solutions and processes used throughout the framework complement one another. For example, when used to detect vulnerabilities in applications, Synopsys WhiteHat™ Dynamic can inform both the security and development teams of its findings so that the vulnerability fixed in the current application and future vulnerabilities can be proactively avoided.

**Findings-based training**

The best application security training for developers is based on real security findings, whether they were discovered during an actual security incident or found during manual penetration testing. The OWASP Top 10 is a list of common web application security risks, but each organization will have its own unique top 10 list. Once determined, an organization can and should use this information to prevent entire categories of security vulnerabilities by implementing focused developer training.

**Threat modeling**

Threat modeling is a type of design-level security assessment that examines how an application works and helps identify potential flaws. The process involves analyzing assets, security controls, and threat agents in the context of an application system. When flaws are detected before software implementation, some security problems can be avoided.
Security frameworks and configuration standards

Some security issues can be prevented by using certain security framework and configuration standards. A few examples include CSRF tokens, which prevent cross-site request forgery attacks; content security policies, which identify assets that the browser should allow to load and execute in order to minimize the impact of cross-site scripting exploits; and HSTS, which encrypts data in transit and prevents fallback to non-HTTPS traffic.

Other kinds of security issues can be avoided by securely configuring the software environment, for example, by following the Amazon CIS benchmark to harden AWS accounts and cloud services.

Program scaling

Due to size, complexity, and available resources, some organizations are challenged by scaling their application security program. Development teams are not security experts, and security teams may lack specialized application security knowledge. Organizations that want to mature their AppSec program may not have the resources to lay out a roadmap to the next level.

By leveraging resources and best practices, and performing modeling assessments, organizations can programmatically prevent application breaches by identifying and fixing vulnerabilities across the application security life cycle, whether they exist in development or in deployment.

Putting it all together

A modern AppSec framework enables organizations to expand beyond identifying and remediating application security issues in siloes, and enables them to take a functional and holistic approach to their application security program. Organizations begin by identifying the limitations and goals of their current plan, and then assessing an inventory of their program based on the four components of the framework. For each component, Synopsys offers the right products and expertise to help organizations advance to the next step of the framework. From identifying to remediating, preventing to governing, organizations that adopt a modern AppSec framework can develop and deploy applications fast and succeed in today's rapidly changing marketplace.
Synopsys provides integrated solutions that transform the way you build and deliver software, accelerating innovation while addressing business risk. With Synopsys, your developers can secure code as fast as they write it. Your development and DevSecOps teams can automate testing within development pipelines without compromising velocity. And your security teams can proactively manage risk and focus remediation efforts on what matters most to your organization. Our unmatched expertise helps you plan and execute any security initiative. Only Synopsys offers everything you need to build trust in your software.

For more information, go to www.synopsys.com/software.

Synopsys, Inc.
690 E Middlefield Road
Mountain View, CA 94043 USA

Contact us:
U.S. Sales: 800.873.8193
International Sales: +1 415.321.5237
Email: sig-info@synopsys.com

©2022 Synopsys, Inc. All rights reserved. Synopsys is a trademark of Synopsys, Inc. in the United States and other countries. A list of Synopsys trademarks is available at www.synopsys.com/copyright.html. All other names mentioned herein are trademarks or registered trademarks of their respective owners. September 2022