opentext™ | Cybersecurity

Position Paper

Mature Your AppSec Program

Achieving an optimal security posture for your business happens when technology, automation, infrastructure, architecture, and security policies are in alignment across the company. This paper provides concrete strategies to mature your AppSec program based on your organization's needs.

Table of Contents

Executive Summary

Most businesses today recognize the need for an AppSec program, but many assume that technology alone will resolve their cybersecurity challenges. This assumption distorts the realities and complexities of application security and creates a false sense of cybersecurity.

This paper provides concrete strategies to mature your AppSec program by implementing a Center of Expertise and Operating strategy. This approach gives your company, people, and customers the confidence to meet regulatory, compliance, customer, and business goals. AppSec programs include employees, competitors, processes, vendors, regulations, and practices. Repeatable processes, standards, and remediation plans help organizations move beyond the basics and embed AppSec into their DNA.

Over 60 percent of applications had one or more critical or high-severity security flaws not covered by the OWASP Top 10, according to Fortify on Demand's 2019 Application Security Risk Report.

Introduction

Achieving an optimal security posture for your business happens when technology, automation, infrastructure, architecture, and security policies are in alignment across the company. This paper provides strategies to mature your AppSec program based on your organization's needs.

Why Optimize Your AppSec Program

People, Processes, Tools

After you have integrated and automated AppSec practices, tools, and reporting into your software development processes, you might be wondering, "What is there to optimize? Our AppSec program is performing as expected and catching external and internal security weaknesses in our software. Trying to do more optimization seems disruptive to the business."

Data breaches caused by unprotected applications have affected nearly 75 percent of companies, according to the 2018 Study on Global Megatrends in Cybersecurity* conducted by the Ponemon Institute.

Many organizations rely on the guidance provided by the Open Web Application Security Project (OWASP) Top 10 to identify the critical and high-risk vulnerabilities in their software code. This is a great way to get started, but if you want to avoid becoming the next security breach headline, it requires a more mature and optimized program. Over 60 percent of applications had one or more critical or high-severity security flaws not covered by the OWASP Top 10, according to Fortify's 2019 Application Security Risk Report. Organizations that only test or mitigate security risks found within these Top 10 are still very vulnerable to attacks.

*www.raytheon.com/cyber/cyber_megatrends

Characteristics of a Mature, Optimized AppSec Program

According to the OWASP Software Assurance Maturity Model (SAMM), "There is no single recipe that works for all organizations. A software security framework must be flexible and allow organizations to tailor their choices based on their risk tolerance and the way in which they build and use software."

In addition to technology, automation, and individual capabilities, maturing and optimizing an AppSec program includes the following:

- Identify the maturity level that the business is seeking:
  - Industry
  - Compliance regulations
  - Acceptable business and customer risk levels
- Create a roadmap that establishes achievable program objectives, policies, deployment timelines, scope, and measurements for success.
- Document quick wins and process successes.

Many businesses recognize the need for an AppSec program, but they assume that information technology alone will resolve most cybersecurity challenges. However, this assumption distorts the realities and complexities of application security and creates a false sense of cybersecurity by means of risk mitigation. Statements such as, "We have identified 100 vulnerabilities in our code…" or "Our scanners detect ten weaknesses in our applications every week…" begin to sound like the famous Apollo 13 message, "Houston, we have a problem."

The AppSec maturity evolution is a combination of technology, automation, capabilities, people, processes, and culture.

Rather than trying to force your existing tools into providing everything you need to move from an immature AppSec model to a mature model, it's important to recognize that achieving an optimal security posture for your business happens only when technology, automation, infrastructure, architecture, enforceable security policies, etc., are in alignment across the company.

To determine the AppSec maturity level required for your organization, begin by understanding the acceptable risk levels defined by the industry that you're in. For instance, healthcare providers need to comply with the Health Insurance Portability and Accountability Act (HIPAA) and payment card industry (PCI) regulations. Financial institutions might only need comply with PCI requirements. For them, experiencing an SQL injection could result in significant exploitation.

The AppSec maturity evolution isn't achieved by simply implementing new technology or automating a specific process. It requires a combination of technology, automation, capabilities, people, processes, and culture.

The Maturity Evolution

Accelerate AppSec Maturity through Technology Deployments

Maturing an AppSec program is complex. Moving through the different levels helps you achieve a mature and optimized program. As you progress towards achieving a full maturity model, your business will progress through the different stages outlined in this section. The following diagrams help you envision a roadmap towards maturity, but they don't describe the complexity associated with maturing an AppSec program. Defining the infrastructure is much simpler than identifying your security policies.

Infrastructure alone doesn't cover the architectural needs of the organization.

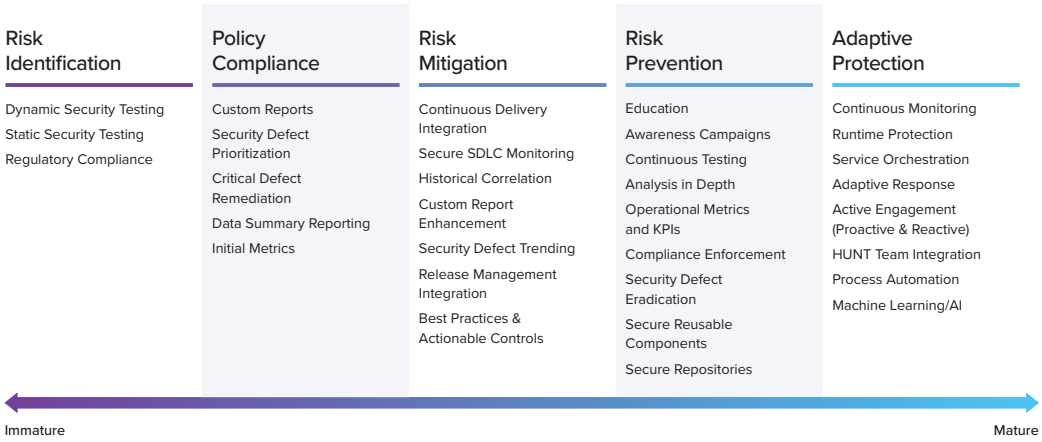

Figure 1. The evolution of AppSec capabilities

Figure 1 shows an AppSec program evolution model from a technology viewpoint; therefore, it doesn't represent a mature or optimized AppSec program. Important controls such as performance, accuracy, or impact level of each of capability area haven't been defined. The requirements, policies, processes, and technologies are simply grouped by capability. When the focus is on the type of technology and the lowest level of complexity, it can create a false sense of security. You need to ask yourself: Do the capabilities shown enable development teams to ensure eradication of weaknesses and vulnerabilities? Can we proactively adapt to the ever-evolving cybersecurity risk landscape by following this model?

In Figure 1, risks are identified during the first phase. You test applications, view results, and run reports that meet compliance requirements. But without the necessary information to fix the vulnerability, holes in your security posture remain, exposing the application to potential exploits.

You might seek to achieve the risk mitigation phase. However, if the policy compliance phase hasn't been completed to address and quantify technical risk, it is difficult to determine which controls should be enforced. Policies must be defined based on the security needs of the business and your customers.

For example, companies that govern mission-critical transactional systems (such as a trillion dollars of cash flow between institutions) need to implement more stringent controls and rigor than applications that manage decision support platforms hosted by cloud service providers for a supply chain company. Businesses need to plan out their security posture and identify acceptable levels of risk in order to develop the right policies.

It is common to focus on the idea that technology alone can accelerate the maturity of your AppSec program. You're told that acquiring certain products and using certain licenses orchestrated in a specific way will help accelerate your AppSec maturity. In this model, businesses assume that maturity acceleration is possible as part of the infrastructure: servers, network, application containers, services stacks, software, and integration between these items. However, infrastructure alone doesn't cover the architectural needs of the organization. Figure 1 provides a basic framework, but it doesn't include any architectural elements, such as operating models, who does what and when, how they get engaged, how systems and coding are monitored, etc.

"Business logic flaws and access control testing is only possible using human assistance."

OWASP Application Security Verification Standard 4.0

Infrastructure identifies the hardware and software necessary to support the development process and meet business requirements such as:

- Applications
- Application containers
- Servers
- Services stacks
- Integration of the technology deployments

Architecture, on the other hand, addresses (but is not limited to) the following:

• **Access to and use of business services**	• **Procedures**
• **Systems operation and integration**	• **Controls**
• **Relationship between deployment and business needs**	• **Staff**
• **Services consumed or provided by the technology deployment**	• **Training**
• **Complexity**	• **KPIs** (key performance indicators)
• **Areas of accountability**	• **KRIs** (key risk indicators)
• **Policies**	

Accelerating the Maturity Process Increases Complexity

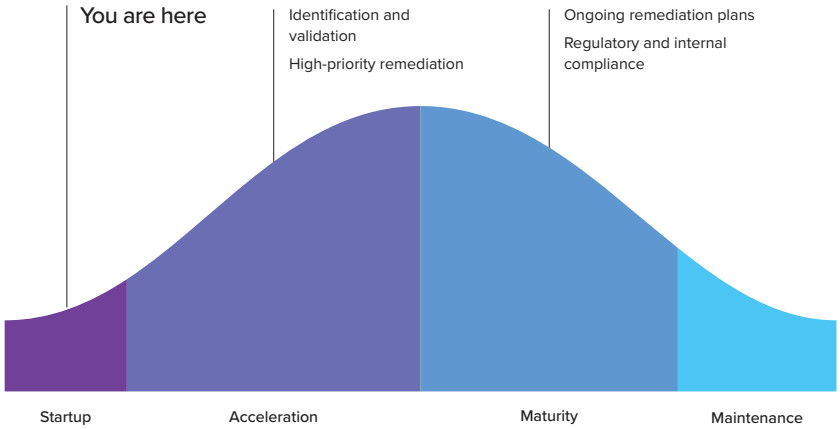

Figure 2. A simple depiction of complex maturity processes

A fully mature AppSec program starts with shifting the business culture.

Diving deeper, Figure 2 highlights the key areas of complexity and prerequisites for acceleration not depicted in the first model:

- Infrastructure
- Architecture
- AppSec Tooling
- Integration

Startup—Acquire technology, identify vulnerabilities. Maturity acceleration through technology includes the following steps:

- Acceleration prerequisites
- Infrastructure
- AppSec Tooling
- Integration

Accelerate—Deploy hardware and software. Vulnerability identification, prioritization, verification, and remediation. (Doesn't include organizational or operating needs.)

Mature—Ongoing remediation, regulatory compliance (internal and external) policies, operating model.

Maintain—Ongoing eradication, system enhancements, maintenance.

Figure 2 is still a very simple diagram of the maturity acceleration and can create a false sense of security. The diagram assumes that the organization, governance, architecture, policy, and compliance are included, but they aren't. The model is primarily focused on

the infrastructure and the necessary software required to identify, prioritize, or remediate vulnerabilities. If technology isn't the complete answer to maturing an AppSec program, what about automation?

Accelerate AppSec Maturity through Automation

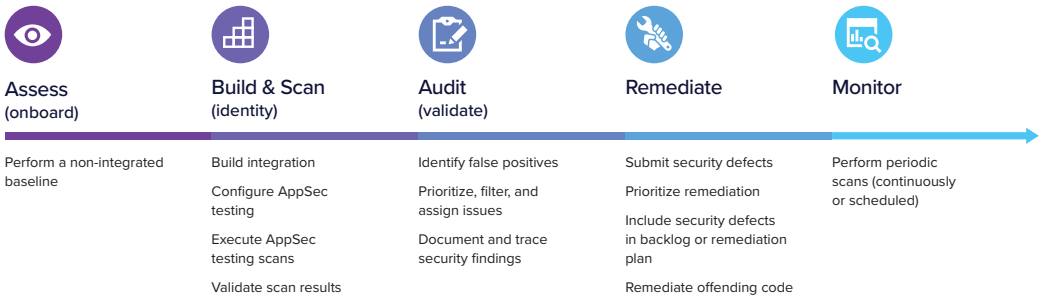

Figure 3. Verification program model

Fortify on Demand's continuous quality and security solutions employ artificial intelligence (AI), machine learning, and advanced analytics to help you make a cultural shift.

If your capabilities and technology deployment haven't achieved maturity, what about using automation? Automation helps improve the use of valuable resources, but a push-button approach that enables the business to release vulnerability-free software applications doesn't equal maturity.

AppSec maturity isn't a one-size-fits-all model. The verification program model in Figure 3 enables developers to test the code, capture the vulnerability, and track, report, and remediate it, but it doesn't define the criticality of the vulnerability. Based on the industry, vulnerabilities should be assigned different risk levels. For example, aerospace might view a vulnerability as critical, but the insurance sector might view the same vulnerability as low risk. The resources to remediate a critical vulnerability for an aerospace environment would be too expensive to remediate in an insurance company.

Automation, like capability and technology deployment, isn't the complete picture of a mature AppSec model. The OWASP Application Security Verification Standard states that automation often can't complete the entire verification process.

Automated tools and online scans are able to complete about half of the verification program without human assistance. If a comprehensive test automation for each build is required, then a combination of custom unit and integration tests, along with build-initiated online scans, are used. Business logic flaws and access control testing is only possible using human assistance. These should be turned into unit and integration tests.

The bottom line is that getting things done in a faster and repeatable way doesn't always mean they are getting done in a better way.

Achieving AppSec Program Maturity

What does it mean to achieve a mature application security model? It can be easy to assume that technology alone will help the organization accelerate your AppSec maturity model. Or, that if all the testing, risk identification, and remediation processes are automated, then maturity is reached. However, a complete maturity model is only attainable when these four areas work together:

- Technology
- Architecture
- Automation
- Operation

To avoid purchasing the technology and then trying to fit it to your requirements, first define what type of technology you need and what you need from it.

Shifting Your Culture

A fully mature AppSec program starts with shifting the business culture. Accelerating a maturity model requires that businesses focus on shifting the culture to adopt security practices and requires incorporating a maturity-centric focus throughout the organization.

However, many businesses concentrate 70 percent of their resources on technology (including training), automation, and processes—reducing maturity concerns to roughly 15 percent of the effort. Figure 4 shows 14 major milestones in a typical Security Development Lifecycle (SDLC), leaving the maturity assessment until the very end of the process.

Figure 4. Fourteen steps in an SDLC

Accelerating AppSec maturity requires integrating security policies and structure into each area. Starting with the level of maturity the business requires is completed during the planning phase and in collaboration with all stakeholders across the company. An AppSec maturity program is included in all the aspects of the SDLC and includes the technology, the automation, and the individual capabilities. When building out a program, define a roadmap that establishes program objectives, deployment timelines, scope, and measurements for success. Keep objectives achievable and ensure that activities are only as complex as they need to be. And be sure to document quick wins and successes.

As part of an AppSec maturity strategy, companies that adopt and establish a Center of Expertise (CoE) accelerate and optimize their AppSec programs in a much shorter amount of time.

Optimize and Mature the AppSec Program

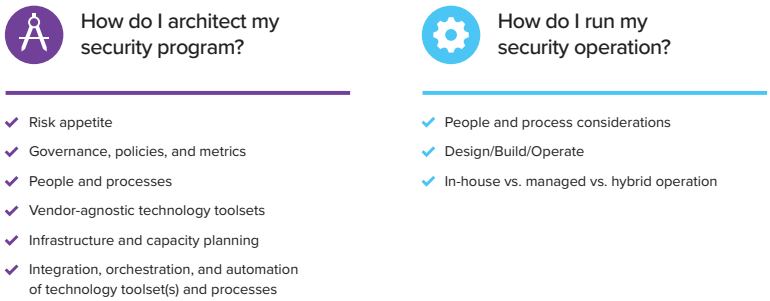

Figure 5. Key areas to address during planning

The AppSec program is in place, so now you need to decide whether the business continues to improve or run. Companies look to improve their security posture for many reasons: compliance, regulatory requirements such as GDPR, financial impact, reputation, etc. The logical next steps are to determine: How are you going to architect this AppSec program? How willing are you to develop a deep understanding of a vulnerability's risk potential? How much risk appetite do you or your customers have? (For instance, is an SQL injection a risk to my customers and business?)

Here are some considerations:

- Make a distinction between the technology and the operating model.
- Choose the technology that enables the right use cases today and tomorrow as you mature your security operations capabilities.
- Maintain the flexibility to support operations through a hybrid or fully managed model.
  - Choose the operational capabilities that you want to maintain in-house (for example, maintain investigations in-house, but rely on managed services to provide low-cost, highly productive level 1 and level 2 ops and take advantage of security expertise).

- Managed security services have the additional benefit of learning from what is happening with customers and industries and applying that intelligence for faster detection, response, and prevention.

If you expect to remain competitive when it comes to building and delivering better software faster, it is no longer a choice between speed, quality, and security. OpenText™'s continuous quality and security solutions employ artificial intelligence (AI), machine learning, and advanced analytics to help you make a cultural shift. Building in comprehensive testing of web, mobile, and enterprise applications from the start is required to quickly bring ideas to life at the pace your industry demands—making users happy and boosting business confidence as a result.

Architecting Your Security Program

Risk appetite drives the architecture of the AppSec program:

1. Governance, policies, and metrics
2. People and processes
3. Vendor-agnostic service stacks and technology toolsets
4. Infrastructure and capacity planning
5. Integration, orchestration, and automation of technology toolset(s) and processes

It's time to select the toolsets that are needed to fulfill the identified business and customer requirements. First, define the governance, policies, service stacks, and business outcomes that must be contained within it to meet your AppSec goals. This varies by industry.

Keep in mind that when technology is acquired before defining the requirements, it's like trying to fit a square peg into a round hole. To avoid such a scenario, first define what type of technology you need and what you need from it. If you need SAST and DAST testing, consider the level of accuracy you need, reporting capabilities, scalability, ramp-up time, training, and technical support. This will help you select the right vendor to provide the necessary technology.

How Do I Run My Security Operations?

Service management. How do we ensure that the AppSec program and services rendered are done in a way that satisfies all the parameters defined by the architecture?

1. People and process considerations
2. Design / Build / Operate
3. In-house, managed, hybrid operation

Do we trust the AppSec program to tell us if the risk posture is working or not working? Are there enough controls in place to ensure that through the design, build, and operate stages?

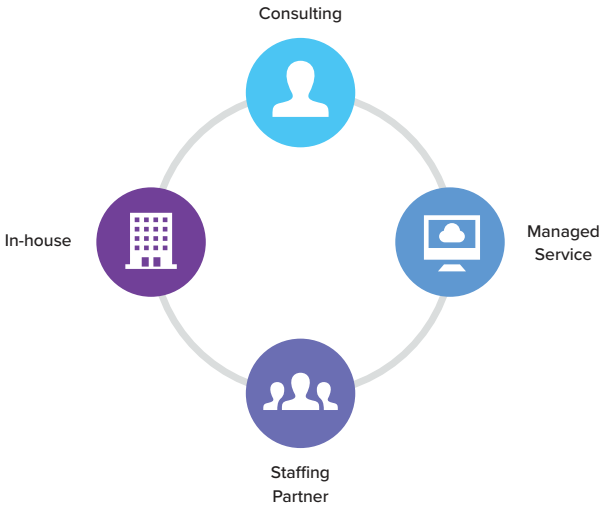

Figure 6. Security operations

Assess the Organization

Assessing the organization on a regular basis helps ensure that you are keeping up with the latest cybersecurity threats in your industry. This ongoing assessment helps you stay current with the following:

- What is happening in our industry? What are our peers doing?
- What are standard practices? Are we performing these practices? If not, why not?
- What technology is deployed? Is the technology supporting the processes and business goals? Are they a function of the business needs?

Means of measuring maturity:

- Implementing a software composition analysis (SCA) strategy and methodology governs how identified vulnerabilities are controlled. The SCA becomes part of the software quality management policies.
- Software application security assurance program controls are how the AppSec program fulfills the SCA and uses the SCA framework to control the quality of the AppSec programs.
- Information Technology, Security, and Assurance enable organizations to achieve their business goals.
- Software security is only one piece of any software development lifecycle (SDLC) and an important component of any software quality assurance strategy.
- The AppSec program maturity assessment takes a holistic look at an organization's AppSec program and uses an interview-based approach to evaluate and rate multiple factors.
- The assessment results in an unbiased report of the organization's strengths, weaknesses, and action plans to address any gaps identified.

AppSec Capabilities Maturity Model

The path to an AppSec capabilities maturity model (CMM) is achieved through vision, culture, communication, and end-to-end processes that include the necessary technology, automation, capabilities, and operations. The following CMM levels move the organization toward their desired goal.

CMM Level	Summary	Attributes	Comments
CMM 1 Initial	Organization's processes are chaotic. Success depends on specific individual and team efforts. Outcomes aren't repeatable or predictable. Processes are ad-hoc and lack documentation. This is a storming phase and not recognized across the enterprise. It might only be focused on highly critical applications. The technology is in place, but is not fully controlled. Lack of standardization leads to redundancy and extra overhead as other teams seek to use some of the processes and structure of the solution.	• Departmental sponsorship. • Ad-hoc secure system development policies; standards and procedures lack enterprise oversight and standardization. • A component developer is accountable for a system malfunction or security vulnerability. • Third-party assessors perform network testing and limited dynamic web application testing on an ad-hoc schedule.	This is the starting point for most organizations trying to implement something new into their environment. Most business seek to achieve CMM 2.
CMM 2 Repeatable	Organization defines, documents, and establishes repeatable processes, policies, sanitation of the data, as well as basic project management routines. All practitioners are using the solution in the same way. Predictability of certain outcomes is possible.	• Executive sponsorship is identified but is limited. • Organization creates an initial draft of its Secure System Development Lifecycle policies, but has no controls in place to enforce them. • Organization is capable of identifying vulnerabilities. • Unable to validate, audit, or remediate vulnerabilities that are found. • Organization understands AppSec as a non-functional concern.	All of the practitioners are using the AppSec program in the same way. This phase is more about how the impacted personnel use program and doesn't yet reflect a level of maturity.
CMM 3 Define	The organization has a standard software development process through attention to documentation, standardization, and integration. A Center of Expertise (CoE) is established.	• Executives understand software security as a program, not just a project. • Has Executive sponsorship to make software security part of the organizational culture. • Organization approves the initial Security SDLC policies and accompanying procedures. • Enforcement is limited to the technical controls in place. • Communities of interest actively participate in security awareness, education, and training events.	

Continued on next page

CMM Level	Summary	Attributes	Comments
CMM 4 Manage	The organization monitors and controls its software development processes, data collection, and data analysis techniques.	• Executives sponsor the AppSec program steering committee to provide oversight and governance. • Software architects integrate functional security requirements into the system design. • Software development community tests, audits, and remediates software security defects. • The organization meets internal and regulatory compliance concerns on time and in full. • Risk management processes determine whether software is released into production.	
CMM 5 Optimize	Continuous improvement of the organization's processes. Monitoring of current processes and introduction of innovative ones. Helps businesses achieve greater confidence, but is not necessary for many companies.	• Organization is actively monitoring the secure SDLC. • Organization periodically reviews AppSec Program policies, standards, and procedures. It makes changes proactively to meet threats, vulnerabilities, and regulations. • Organization understands the risks and liability associated with information security. • Protection of the organization's information assets is of utmost importance.	Applicable for large software or hi-tech companies producing security programs.

Table 1.

Strategies for Maturing and Optimizing an AppSec Program

Strategy 1: Establishing an AppSec Center of Expertise

As part of an AppSec maturity strategy, companies that adopt and establish a Center of Expertise (CoE) are able to accelerate and optimize their AppSec programs in a much shorter amount of time. For example, one business that formed a CoE implemented a fully functioning AppSec program that enabled them to achieve close to 100 percent coverage of their internally developed application portfolio within three years.

The CoE is a group of peers aligned with the AppSec program goals. The team understands the operational concerns, technologies, processes, business problems, and impact that the AppSec program has on business success. They are responsible for the five key areas shown in Table 1. The depth of these tasks is dependent on the organization's AppSec maturity level. CoE practitioners must understand the task to be performed and ensure that it achieves the desired business outcomes.

The CoE acts as the “brains” of the AppSec program and feeds into the second strategy: operating the AppSec program. To be effective, the CoE must have a commitment from the organization and sufficient budget to execute the program. The CoE enables the business to grow the program with funding and resources. It consists of experts from across the organization who support the AppSec maturity program. The CoE is a deliberate strategy that brings together experts from the organization that understand the AppSec technology, are able to identify vulnerabilities, and know how to fix them based on the impact to the business.

Figure 7. Center of Expertise

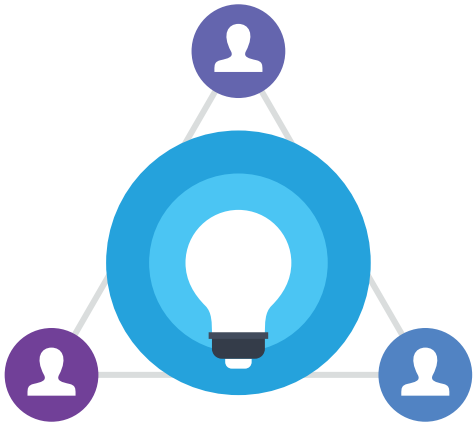

Figure 8. CoE is a team of experts from across the business

Governance—Understand the task to be performed. Ensure that the outcome is guided by the business concerns, policies, rules, objectives, and goals that drive the AppSec program.

- Understand the business problem.
- Identify the task to be performed and the expected outcome to support the business concerns.
- Define and monitor the rules, policies, and goals that drive the AppSec program.
- Measure performance based on established metrics.

Resources—Provide resources, program goals, and objectives. Publish an AppSec maturity assessment, updated policies, standards, and procedures.

- Ensure that resources are avalable across the enterprise, including those beyond the AppSec program.
- Consider the supply chain from a customer's point of view.
- Understand the effect that an exploit would have across the entire ecosystem.

Figure 9. Application exploit at point of sale (POS)

Awareness—Provide corporate-wide communications

- Meet with stakeholders to communicate the organization's goals and objectives.
- Build awareness of new policies, standards, and procedures.
- Communicate the consequences of noncompliance.
- Utilize learning management systems.

Operations—Produce an operational model using an AppSec platform such as Fortify on Demand by OpenText™. The CoE understands operations deliverables. They operate the system using highly skilled experts in the areas of pen testing, auditing, technology, custom rules, and reports to ensure that the workflow is consistent from project to project.

- Use highly skilled staff.
- Manage auditing.
- Define the technology operational model.
- Provison certification enviroments.
- Establish rules and policies.
- Exercise the AppSec program in accordance with the organization's software and systems development.
- Produce a collection of libraries and artifacts to help reduce the attack surface.

Monitoring—Run regular reports on collected data, from the point of inception (vulnerabilities) through evolution. Report on and prescribe remedies to address any goal. Aggregate quality systems, ticketing systems, change logs, and depth-of-scanning cycles to provide a complete picture of the organization.

- Work being done is performed as needed to meet business objectives
- Regular structured data reporting (evolution of vulnerabilities)
- Quality management, tracking and ticketing systems, change logs, deployments
- Depth of scanning cycles
- Enterprise picture of all the activites within the AppSec program

Strategy 2: Operating the AppSec Program

Operating the AppSec program shows the value of supporting continual AppSec program growth. It is not just about pushing a button so that the output comes out as expected based on the input. It demonstrates the value to the business and ensures that the program is growing in a way that justifies the cost. For instance, simply acquiring the technology and letting it sit on the shelf is a disservice to the business, customers, and employees. This aspect of the AppSec program isn't simply a one-time event. AppSec programs require continual oversight in terms of identifying new vulnerabilities, tracking errors in code, managing, and monitoring.

Once the AppSec program begins, it doesn't stop unless there is a concrete business need. This could entail new customer demands, operational needs, new ideas that affect the code and applications, etc.

It is essential that the AppSec program evolves at a pace defined by the CoE. Are the application testing rules regularly updated and kept current? Does the code adhere to the standards and procedures established and agreed to? Do changes in the ecosystem require modifications to the standards and procedures? Any change can cause a weakness in the code and applications.

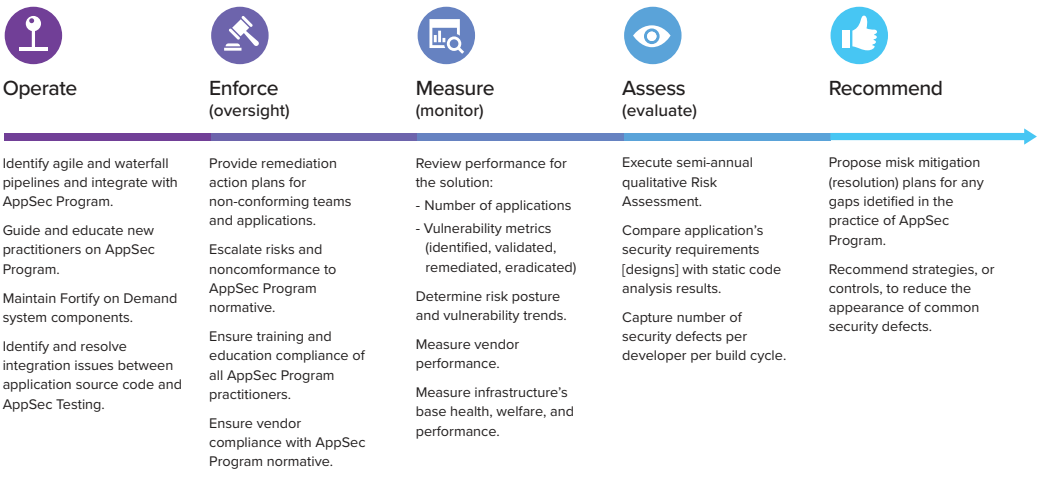

Figure 10. Operating the AppSec model

Operating the AppSec program includes identifying the source of truth for the code pipeline. Rules and enforcement of code that is checked in and checked out of user acceptance testing (UAT) must be defined. Is the code verified for integrity?

Is every instance of code the same version or have changes been made post-submission? Making changes after the code has been signed off and submitted for UAT can introduce points of failure. Unenforced changes to code create potential risks from multiple code revisions. After the code is signed off, it is imperative that it is not contaminated.

Maintaining systems platforms is also part of operating an AppSec program. Aging platforms might not be able to keep up with the latest rules, patterns, and signatures required to eliminate high-risk vulnerabilities and perform as expected. For example, Fortify on Demand is driven by rules, but if it is running on a Windows XP platform that has not been updated since 2010, the scan tests are ineffective against vulnerabilities that have entered since then. The company and its customers are exposed to potential exploits because security signatures aren't available for vulnerabilities affecting this older platform.

The CoE operates the AppSec program according to the directives established earlier in the process, but the operation of your AppSec program could come at a higher cost. Highly-skilled, third-party professionals such as Saltworks can enable the operation, oversight, control enforcement, monitoring and reporting, control evaluations, recommendations—reducing internal resource needs.

A mature AppSec program requires the entire enterprise to understand the risks that can be lurking inside neatly written code. AppSec needs to be part of the business's DNA. It is an orchestrated effort between the organizations that are involved.

Moving AppSec maturity from one CMM level to the next requires that your AppSec programs are built on small wins. Success builds upon success. The entire enterprise needs to embrace AppSec programs. It is an orchestrated effort and higher levels of maturity are achieved when the effort is focused, concentrated, properly aligned, and organized.

Implementing the CoE and executing the Operate strategies can accelerate AppSec maturity when there is deliberation about what is allowed, verified, and signed off in the code before moving from a developer's workstation to production.

CMM levels four and five are achievable, but they require a higher level of investment. Typically, these levels are meant for companies that must meet rigid regulatory standards and need the highest level of application security for their business. For example, an airline could put hundreds of lives at risk if an application fails because of a vulnerability. Similarly, a hospital that performs heart surgeries could be at the mercy of a hacker that exploits a known vulnerability in a critical application.

Operate
- Identify SDLC pipeline and integration with the AppSec program.
- Guide and educate new practitioners on the AppSec program.
- Maintain security testing platform components.
- Resolve integration issues between the application source code and AppSec test results.

Enforce
- Provide remediation plans for non-conforming teams and applications.
- Escalate risks and non-conformance to AppSec program standards.
- Ensure training and education compliance for all AppSec practitioners.
- Ensure vendor compliance with AppSec program requirements.

Measure
- Review performance goals for the solution:
- Number of applications
- Vulnerability metrics (identified, validated, remediated, eradicated)
- Determine risk posture and vulnerability trends.
- Measure vendor performance.
- Measure the infrastructure's base-health, welfare, and performance.

Assess
- Execute a semi-annual qualitative risk assessment.
- Compare the application's security requirements (designs) with static code analysis results.
- Capture the number of security defects per developer per build cycle.

Recommend
- Propose risk mitigation plans for any gaps identified in the AppSec program.
- Recommend strategies or controls to reduce the occurrence of common security defects.

Conclusion

As you review the CMM model, how would you assess the maturity level of your organization? Moving from one CMM level to the next takes time and resources. Maturing an AppSec program can take years, but implementing the type of Center of Expertise and Operating strategy outlined in this paper can help to accelerate your AppSec maturity model beyond the basic level driven by technology. It can give your company, people, and customers the confidence to meet regulatory, compliance, customer, and business goals. Repeatable processes, standards, and remediation plans help organizations move beyond the basics. AppSec needs to be in the company's DNA. AppSec programs include employees, competitors, processes, vendors, regulations, and practices. But achieving AppSec maturity isn't a one-time event. It's an ongoing journey that requires full cooperation from the executives to the developers.

Contact the Fortify on Demand team to help you get started on this exciting, always-evolving journey toward 100 percent coverage of your AppSec programs.

Learn more at
www.microfocus.com/en-us/cyberres/saas/application-security

762-000025-004 | O | 01/24 | © 2024 Open Text