EVERYTHING YOU NEED TO KNOW ABOUT

APPLICATION SECURITY POLICIES

Application security policies can make or break an enterprise security initiative. Here's how you can adopt a best-practice framework.

ca technologies | VERACODE

POLICIES MATTER

There's growing recognition that **application security** (AppSec) is critical to protecting valuable enterprise resources. Used effectively — and in conjunction with other security solutions — AppSec can detect vulnerabilities and identify risks before they lead to full-blown breaches and breakdowns. However, like any tool or methodology, AppSec requires a strong structural framework to deliver maximum results. This means that organizations must establish a governance model — and introduce effective policies — in order to achieve maximum protection. What's more, as we move toward **DevSecOps**, the challenges are magnified.

It's an issue no enterprise can afford to overlook. **Lacking strong application security policies, your developers and security teams may find themselves overwhelmed by alerts, notifications and general "noise."** Of course, not every issue or problem deserves the same level of attention — or the same level of response. Teams can easily become overwhelmed chasing down every flaw and fix. Unfortunately, a broadly defined and unfocused AppSec program can lead to the opposite of what's intended: overwhelmed developers and security teams who don't take threats seriously.

Ultimately, there's a need to fully **understand risks, balance priorities and focus resources** in the most effective way possible. There's no cookie-cutter method to designing a strong application security policy framework for your organization. It's a matter of setting the bar at the right risk and protection level, determining which flaws really matter, understanding remediation and mitigation, and keeping an eye on third-party applications and open source component use. When you can balance these needs — and maintain a focus on standards like the **OWASP Top 10** or **SANS 25** — you'll be positioned for maximum performance and protection.

BLOG POST

Learn how policies impact an AppSec program. Read "Three Reasons AppSec Policies Matter" for a more detailed examination of the topic.

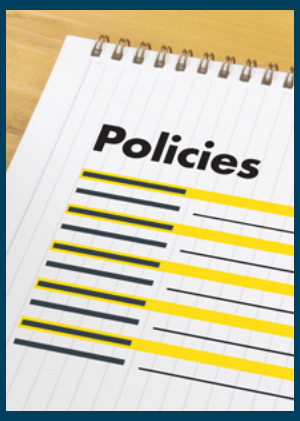
WHAT'S AT STAKE

Today's business environment swirls around a few basic things: speed, flexibility and innovation. Organizations that assemble the pieces of the puzzle effectively reap significant gains — and even become disruptors. As a result, many organizations have turned to DevOps and continuous integration/delivery (CI/CD) methodologies to introduce a more agile framework. For some organizations, AppSec has never been a priority and the emerging digital framework demands changes. For others, existing AppSec policies were never designed to address DevOps and CI/CD — they were written with entirely different needs and requirements in mind.

The result is an environment where tools and policies don't match the organization's business challenges — or its development framework. This may ultimately lead to slowdowns, bottlenecks and an inability to use automated tools that speed detection and remediation. Within this environment, your teams may find it impossible to meet objectives and hit key metrics, or they may not address crucial requirements, including adhering to regulatory frameworks such as PCI or tackling OWASP requirements. Morale may take a hit while overall performance and security lag.

EFFECTIVE POLICIES REQUIRE EFFECTIVE INPUT. HERE ARE SOME OF THE KEY CONSTITUENCIES:

- Executive board/ C-suite
- Development team
- Marketing and communications
- Procurement groups
- Legal team



BLOG POST

For a deeper look into the growing challenge of meeting DevOps and DevSecOps challenges, check out our blog post, "Application Security Policy: Might Need to Revisit as DevOps Emerges."

RETHINK POLICIES

Organizations that are implementing or updating an AppSec program or a policy framework typically benefit by identifying what's critical, what's achievable and what's desirable (but not essential). It's important to strike a balance because the groups that must adhere to the new policy possibly haven't faced a formal framework in the past. **If AppSec policies are too onerous or unrealistic, developers and security teams may feel overwhelmed and give up before the initiative takes off.** In addition, they must often learn about new requirements and face new processes and workflows.

An effective way for your organization to approach the challenge is to identify the most serious flaws and vulnerabilities while hammering out simple policies that are most achievable. This allows your teams to familiarize themselves with an AppSec methodology, build on their success and score key wins. As developers adapt and adjust — and as your teams become more adept at addressing vulnerabilities and flaws — your organization can introduce more stringent or comprehensive policies. This makes it easier for developers to gradually build more robust security into their workflows. It also helps your organization adopt a more seamless and effective security framework.

Many application security policies were built when we did not have fast, automated security tools that could be plugged into the SDLC ... It is important to revisit and build new policies that work with, and not against, the developer goal of "getting good code out quickly."

Pejman Pourmousa
VP, Program Management, CA Veracode

ARTICLE

Learn how to put automation to work and comprehensive controls in place by reviewing Streamline Compliance with Industry Regulations.

REFINE POLICIES

One key to building a better AppSec program is to clearly define what tools, controls and systems are required. This task must span both technology and processes — and tie both together seamlessly. This encompasses assessment tools, including static, dynamic and composition analysis. Your enterprise must also define key criteria, such as the acceptable time span to fix a flaw based on whether it falls into a low-, medium- or high-risk category. For instance, your organization may want to address a high critical flaw within five days, a medium critical flaw within 15 days, and a low critical flaw at an unspecified later date. In some cases, a policy may allow a low critical flaw to go unaddressed entirely.

It's also important to right-size your policies. All apps and tasks are not created equal, and internal and external apps may have very different requirements. As the pace of development accelerates, there's an urgent need to ensure that changes to policies don't hinder your teams and slow software development. On a practical level, this means embedding key security protections into the development cycle, and inserting code reviews and scans at optimal points and times. For instance, conducting a penetration test at each release or at the end of a release cycle may result in delays. By changing the requirement to a more suitable time frame — quarterly or semi-annually — and conducting daily static scans, your teams may radically speed development without boosting risks.

BLOG POST

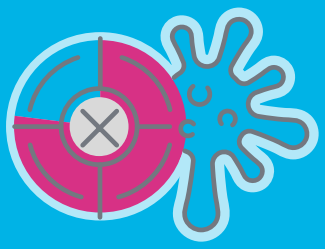
Understand more about how to embed security into the development cycle by reading "Top 4 Ways Veracode Integrations Make Security's Job Easier."

JUST THE FACTS

Our 2017 review of the applications we scanned revealed that:

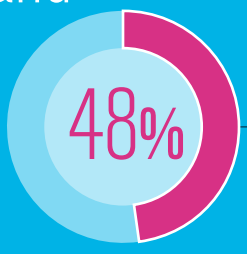
70% of previously untested software **fails to pass the OWASP Top 10 Policy list.**

77% of untested software has **at least one vulnerability.**



12% of previously untested software has **at least one high or very high severity vulnerability.**

Developers scanning code early and often fix more flaws.



Mature AppSec programs have a 35% higher pass rate than new programs.

Source: CA Veracode, *2017 State of Software Security*, 2017.

FOCUS ON GOVERNANCE

At the center of any effective AppSec program is governance. **Without an ability to track performance and policy adherence, analyze behavior and actions, and enforce policies, an initiative is likely to devolve into chaos and confusion.** Security specialists are likely to spend even more time chasing down flaws and attempting to plug vulnerabilities. A strong governance framework can help ensure that your organization is in sync and on track to minimize risk and maximize protection. It creates a consistent and uniform approach across applications, portfolios and systems.

A STRONG GOVERNANCE FRAMEWORK:

- Incorporates input from multiple constituencies and varied sources to address practical requirements and actual needs.
- Tackles both internal and external challenges by placing the focus squarely on security rather than program participation.
- Plugs in industry standards such as the OWASP Top 10, SANS 25, HIPAA or PCI to address relevant risks.
- Concentrates efforts on real-world vulnerabilities rather than flaws. This slides the dial from abstract threats to actual dangers.
- Weighs remediation versus mitigation. This helps ensure that your resources and budgets are used in a practical and outcome-oriented way.
- Measures results using metrics and key performance indicators (KPIs). This helps your organization track compliance, flaw prevalence, fix rates, flaw densities, and business- and goal-specific performance metrics. When it's time to report to senior leaders on these metrics, a recent Veracode blog post, "When You Need to Report a Single AppSec Metric: Our Recommendation," offers guidance on consolidating the numbers for executive consumption.
- Delivers insight into when you should update policies. With the right monitoring systems in place, it's possible to spot gaps that can be addressed through changes to policies and procedures.

POLICY BEST PRACTICES

Here are three ways to get the most out of AppSec governance and policies:

Ensure that policies complement processes and workflows, rather than introducing friction and potential roadblocks to security.

Too often, well-intentioned and seemingly sound policies boost risks rather than reduce them. Only with input from different constituencies and a thorough review of workflows is it possible to ensure that your policies match your organizational requirements.

Introduce incentives and avoid punishments.

A "no judgement" approach will motivate your development teams to comply with policies rather than fight them or unconsciously undermine them. When you identify gaps and problems, be sure to address them through **training** and non-punitive feedback.

Don't set the bar too high.

Although the goal is to reduce, if not eradicate, coding vulnerabilities, unrealistic or seemingly unachievable metrics will likely lead to low morale and diminished effort to deal with issues and problems. So it's wise to avoid policies that are overly stringent. Introduce policies that lead to achievable results at first, and then, as teams gain proficiency, add more stringent controls.

GUIDE

For a more thorough look at how to formulate an effective governance strategy and specific policies, review our guide "Policy Matters: How to Build a Robust Application Security Framework."

PRIORITIZE VULNERABILITIES

It's one thing to recognize that all vulnerabilities and risks aren't the same. It's an entirely other thing to build a framework — with the right policies — that focuses your enterprise resources and money appropriately. According to Gartner,[2] three factors are at the center of developing sound policies and addressing AppSec challenges:

Threat-Centric

Identify vulnerabilities in the wild, including those targeted by malware, ransomware, exploit kits and threat actors. These are the starting point for formulating effective policies.

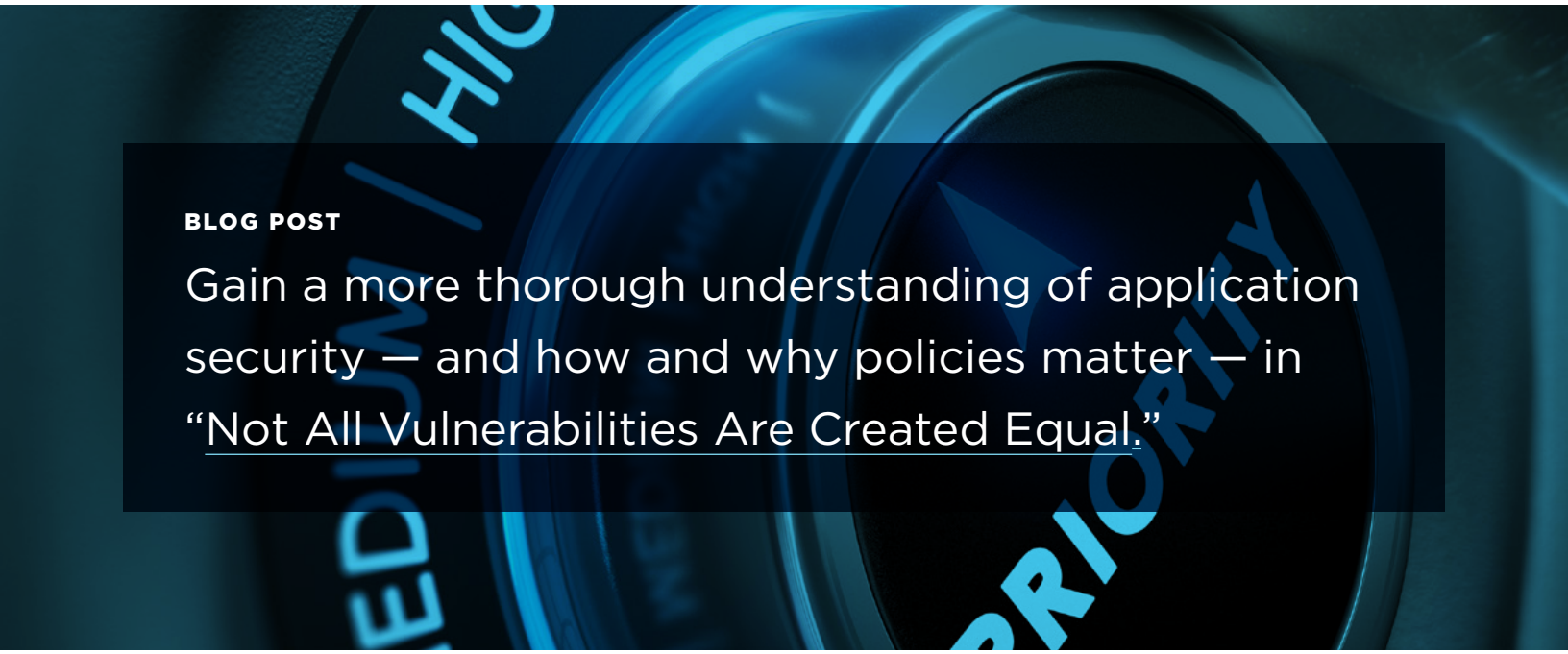
Vulnerability-Centric

Prioritize vulnerabilities according to the criticality of the vulnerability, such as ease of exploitation, exploitation impact and whether a public exploit exists.

Asset-Centric

Finally, prioritize vulnerabilities associated with critical assets, and address the most serious risks first. As money and resources permit, work your way down the list.

2 Gartner, *Incorporate Application Security Throughout the Application Life Cycle*, November 2017.



BLOG POST

Gain a more thorough understanding of application security — and how and why policies matter — in "Not All Vulnerabilities Are Created Equal."

CRAFT POLICIES USING A BEST-PRACTICE APPROACH

Governance isn't the end goal — it's the mechanism for putting policies into motion and managing them effectively. Your organization's ability to design the right policies for its industry, business and risk model is crucial. Here are some key factors that can tip the scale toward success:

Focus on program participation.

When an enterprise sets the bar at the appropriate level and makes results achievable, it's possible for developers and security teams to hit the sweet spot on the performance-protection continuum. Ideally, every policy should revolve around this concept. It's the foundation of an effective AppSec program.

Concentrate on flaws rather than vulnerabilities.

Not every flaw is a vulnerability. The upshot? Your organization should focus on the right risks, including those that fall into the OWASP Top 10 and SANS Top 25. With this knowledge — and the appropriate analysis — it's possible to translate risks into definable criteria and actions. In some cases, a slight risk could lead to enormous damage, while a major risk may only result in minor problems.

Understand remediation and mitigation.

A blunt force approach results in wasted resources, frustrated developers and diluted security. Your policies must take a nuanced approach to addressing problems. This means applying the right policies and solutions to a problem. In some cases, fixing the problem is essential (remediation), while in other instances compensatory controls or reducing the risk through patches or other fixes is preferable. Occasionally, your organization may opt to accept the risk and potential consequences.

Consider third-party applications when designing policies.

It's important to ensure that third-party applications meet internal AppSec requirements. This may necessitate creating policies that address procurement processes, as well as APIs and middleware that connect systems. It can also touch on the cloud and managed applications and services that intersect with internal code. Open source components are yet another risk that must be addressed. One Veracode analysis found that open source components introduce an average of 24 vulnerabilities into each application in which they're used. Therefore, consider adding guidelines for where and how developers use open source components in your policy.

HARD CODING A SECURITY STRATEGY

In the final analysis, it's clear that a sound AppSec strategy goes a long way toward producing best-practice results. **At the center of a strong application security program are effective policies. They define how an organization acts and reacts in the face of risk. What gets measured is what gets done.** Today, application security is not an afterthought, nor is it a tool that your organization can simply toss at cybersecurity challenges and expect outstanding results. Instead, there's a need to understand the power of policies, how they influence behavior and actions within your enterprise, and how a strong governance structure and policies can help dramatically reduce your risk. **As your organization seeks to move faster and smarter** — with DevOps driving the business — a comprehensive application security framework is paramount.

A basic fact of application security is that any policy should be only as complicated as it needs to be to deliver the necessary results, but no more than that.

BLOG POST

For a look at how to get your development team on board with AppSec and a more comprehensive policy framework, read "3 Ways to Get Your Development Team on Board with Application Security."

Make sure your AppSec policy is propelling your program forward rather than holding it back.

Contact us for help developing or refining your application security policy.

RESOURCE ROUNDUP

- **Learn how policies impact an AppSec program:** "Three Reasons AppSec Policies Matter"
- **Get a deeper look into the growing challenge of meeting DevOps and DevSecOps challenges:** "Application Security Policy: Might Need to Revisit as DevOps Emerges"
- **Learn how to put automation to work and comprehensive controls in place:** "Streamline Compliance with Industry Regulations"
- **Understand more about how to embed security into the development cycle:** "Top 4 Ways Veracode Integrations Make Security's Job Easier"
- **Find out how to formulate an effective governance strategy and specific policies:** "Policy Matters: How to Build a Robust Application Security Framework"
- **Gain a more thorough understanding of application security — and how and why policies matter:** "Not All Vulnerabilities Are Created Equal"
- **Discover how to get your development team on board with AppSec and a more comprehensive policy framework:** "3 Ways to Get Your Development Team on Board with Application Security"

ABOUT CA VERACODE

Veracode, CA Technologies' application security business, is a leader in helping organizations secure the software that powers their world. Veracode's SaaS platform and integrated solutions help security teams and software developers find and fix security-related defects at all points in the software development lifecycle, before they can be exploited by hackers. Our complete set of offerings help customers reduce the risk of data breaches, increase the speed of secure software delivery, meet compliance requirements, and cost effectively secure their software assets – whether that's software they make, buy or sell. Veracode serves over a thousand customers across a wide range of industries, including nearly one-third of the Fortune 100, three of the top four U.S. commercial banks and more than 20 of the Forbes 100 Most Valuable Brands. Learn more at **veracode.com**, on the **CA Veracode blog**, and on **Twitter**.

ca technologies | VERACODE