Cigital
BUILDING SECURITY IN

How to Build a Rock-Solid Software Security Initiative in 5 Steps

(Plus 10 reasons why you even need one.)

What You'll Find in This eGuide

This year you tested 46 web applications, 19 mobile apps and 20 client-server apps. You purchased a new application security testing tool. You found 112 vulnerabilities. You're feeling pretty good.

But before you get too excited, ask yourself this: Did you reduce risk significantly? At all? Did you leave critical vulnerabilities unaddressed? Does your Board understand the importance of what you're doing and the impact of what you did?

If you aren't sure of the answers to these questions, you may have a software security testing *plan*, but you don't have a software security *strategy*.

If you've made an investment in application security testing already, then you're on the right rack to lowering risk. Now, however, it's time to take it to the next level: turn your application security activities from a cost center to a competitive advantage for your organization by creating a software security initiative (SSI).

This guide is for you if you've ever...

❑ Relied solely on your instincts to decide where to invest your security budget

❑ Struggled with a development team over prioritizing and repairing security problems

❑ Had challenges communicating security requirements and results to executive leadership or other departments

❑ Found the same security defects again and again—from the same team

❑ Scrambled to find resources to address capacity issues, changing development schedules or regulatory changes

❑ Had last-minute requests to test applications for vulnerabilities that delayed your product launch

❑ Heard about breaches in the news (for example, Target, Wal-Mart, Sony and Neiman Marcus) and thought "Could this happen at my company?"

❑ Been burnt by poor security planning in the past

❑ Had a deal delayed while a client demanded concessions because of your lack of a secure SDLC or because you didn't know which of your vendors used good secure software practices

❑ Been in a position to come under scrutiny by the Federal Trade Commission or other regulatory body

Be honest. We aren't recording your answers.

If even one of these sounds familiar, read on to learn proven steps to building and evolving a software security initiative that will turn your current security efforts into a structured, strategic, rock-solid program.

Hey, we do application security testing, isn't that enough?

In a word, no.

We regularly see application security testing presented as the *de facto* software security technique in case studies and whitepapers—a kind of magic bullet organizations use to show they take security seriously.

Application security testing is a critical and necessary component of every security program. However, "penetrate and patch" application testing alone is not a security strategy at all. *Application security testing is a starting block, not the finish line.*

Application security testing is a starting block, not the finish line.

> “Proactive security saves time and money, but it is not going to be enough. A security PROGRAM is what you need to put in place to lower your exposure across the board.”
>
> **Tyler Shields**
>
> Senior Analyst, Forrester Research, Inc.

Top 10 Reasons You Need a Software Security Initiative

Establishing a software security initiative has many benefits, including:

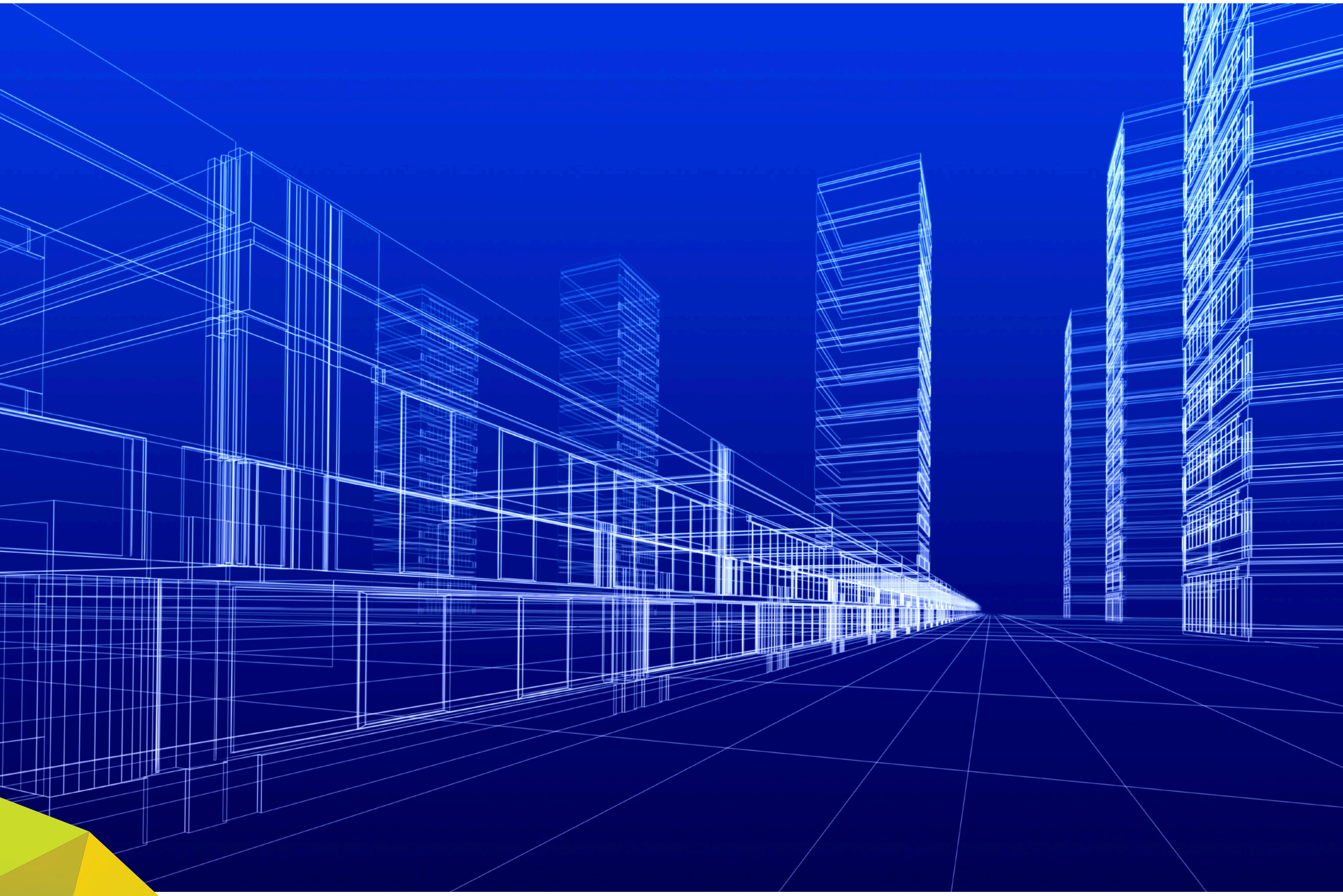
1. Ensuring you address unacceptable risk as a priority.
2. Providing developers a path to create secure software with minimal disruption, which will improve productivity.
3. Giving a specific person or group the responsibility for reducing software security risk so that the hard work actually gets done.
4. Building a formal bridge between security and development teams with shared priorities, responsibilities and incentives to reduce confusion and help everyone work together more efficiently.
5. Documenting and harmonizing software security requirements for product managers, architects, developers, testers and all other stakeholders to create organizational alignment.
6. Providing consistent expectations for everyone in your software supply chain, including internal teams and external vendors, so you can trust that software is built securely no matter where it originates.
7. Providing a 'center of excellence' for all software security needs, such as policy, standards, tools, experts and so on, so that people have a place to get answers and improve their skills.
8. Enabling you to measure and communicate success to customers, partners and the Board.
9. Ensuring consistent outreach to and training for every stakeholder in the software development chain, strengthening a culture that prioritizes security.
10. Meeting the changing needs of development teams while managing risk.

Your Blueprint for a Rock-Solid Software Security Initiative

The most effective SSIs are fine-tuned to fit an organization and built to scale around your staff, processes and software portfolio. They help you "show your work" by providing a clear and understandable methodology for reducing risk and explaining how you've made investment decisions.

We believe the best way to set a solid foundation for a software security initiative (or revive a moldy one) is a five-pronged approach:

1. Build
2. Measure
3. Verify
4. Improve
5. Manage



BUILD

To set the right foundation for your SSI, you're going to need a few things: some key pieces of information to set priorities, a governance structure, and training and tools that build security into development cycles.

Let's explore each of these in more detail.

5 Things Every Security Leader Ought to Know

To start setting priorities for application security activities, you need to fully understand the scope of your challenge.

Do you know...

1. What development projects are in progress and their deadlines?
2. Which teams are touching which applications?
3. What code is developed in-house vs. COTS or third-party?
4. Where your greatest technical risk exists?
5. What is in your application inventory and which have the greatest impact on your business?

Go forth and find out.

It's fine if you don't have all the answers to these questions right away. According to a recent SANS Institute study, more than a quarter of respondents didn't know how many applications their organization used or managed.

Get started with what you know today and continue to build on your inventory of knowledge.

Risk = Likelihood x Impact

(We know, you've seen this hundreds of times...but we're including it anyway. Why? Because it's true.)

Factors that contribute to an application's business impact include:

- Relationship to revenue
- Effect on business continuity
- Compliance or regulatory requirements
- The audience it serves
- How much sensitive data it stores or accesses
- Methods of access
- Connection/integration with other systems
- Human safety
- National security

One way to start is by giving each factor a point value. Add up the points for each of your applications. Group your applications into "high," "medium" and "low" business risk categories to help prioritize your efforts.

The Secret to a Rock-Solid SSI

The secret to a successful SSI is governance. Because governance establishes responsibility for security activities and expectations for behavior, it's a necessary step toward creating or right-sizing a foundation for a sustainable SSI.

It makes no difference whether governance is established by a centralized security group through formal policies, explicit standards, and some systematic processes, or by a scrum master through technology and coding standards for a set of application teams. The fact is, someone must be in charge and there must be some mandatory expectations related to software security—if not, you don't have a 'secure SDLC'.

A word of caution! Don't create security policies in a vacuum.

Ultimate responsibility may rest with the security head, but application security must be widely discussed with other company leaders and distributed throughout the organization. Most importantly, bring the development team in early and make sure they share ownership over the creation and execution of policies.

Most Firms Don’t Have a Secure SDLC

That’s because, despite many claims to the contrary, a majority of firms don’t have software security governance, nor systematic control over the security posture of their application portfolio.

Don’t be one of them.

Customers don’t like it, insurance companies don’t like it, regulatory bodies don’t like it, and—coming very soon to firms everywhere—executives and Boards of Directors don’t like it.

By way of analogy, look at how a lack of “security” policy affects things.

57

Percent of U.S. businesses that don’t have a cyber security policy.[1]

70

Percent of companies where security policy was poorly understood that had staff-related breaches (vs. 41% where the policy was well understood.)[2]

$1.7 Million

Cyber crime “cost savings” for companies that invest in adequate resources, appoint a high-level security leader, and employ certified or expert staff.[3]

1. Security Tracker

2. Department of Business Innovation & Skills

3. 2014 Cost of Cyber Crime Study

5 Key Security Policies

1. **Software security**. Communicate the high-level expectations for getting software security done in your firm and build security into product requirements, implementation, procurement, deployment and operations. Address at least the following topics:

   ❑ **Secure SDLC**. Use is not optional.

   ❑ **Application risk ranking**. Give clear guidance on determining which applications are most important to the business.

   ❑ **Application design**. Require security controls to be built into your system design.

   ❑ **Application development**. Require specific technology stacks and mandatory coding standards, and provide developers clear guidance and pre-built secure-by-design modules.

   ❑ **Application testing**. Determine which applications must be tested, which gates they must pass and set schedules for testing intervals.

   ❑ **Software projects impact rankings**. Define impact rankings for software projects and outline how the rankings drive associated assurance efforts.

   ❑ **Defect severity and remediation**. Set rules on how bug and flaw severities will be set and the timelines for when coding bugs and design flaws must be fixed.

   Ensure harmony between software security policies and other policies.

2. **Network security**. Determine protocols and authorization levels that help application security.

3. **Data security**. Identify and classify your valuable IP and sensitive customer data to help developers apply the correct security features.

4. **Physical security**. Govern access control and secure your physical infrastructure.

5. **Disaster recovery**. Determine steps to take in the event of an attack, including reporting, recording and resolution for attacks against applications.

For incentives to be taken seriously and valued as part of a developer's career path, they must be built into performance evaluations and compensation.

The Trick to Creating Training That Sticks

Everyone involved in the software development lifecycle must know how to perform the software security duties associated with their role. This includes executive management, middle management, product owners, testers, system architects, developers and everyone else.

Why developers?

Every three years the Open Web Application Security Project (OWASP) publishes a list of the top 10 web application security vulnerabilities in an effort to raise security awareness. Well-known security vulnerabilities such as SQL injection and cross-site scripting have made the list year after year. Yet, software developers will still code those vulnerabilities into applications a thousand times today.

An effective SSI must address application security at its core—the point when code is written. The earlier bugs and flaws can be removed from an application build, the less need there is for time-consuming and expensive remediation in the QA stage, which means secure applications can reach the market faster, perhaps even as a competitive advantage.

It's essential that you build incentives into your SSI plan to motivate developers to improve their ability to create secure code—not just deliver features. You can support developers by providing both in-person and online training opportunities. That said, for those incentives to be taken seriously and valued as part of a developer's career path, they must be built into performance evaluations and compensation.

Tools That Put Security in the Path of Development

Tools for dynamic analysis, static analysis, fuzzing and others help security teams consistently identify a broad range of vulnerabilities. That said, these types of tools are used to find issues in applications that are already live or in pre-production builds when it is already expensive and time-consuming to fix them.

Look for tools that help you to move security "left" in the secure SDLC. The earlier you can address security tests and fixes, the more cost-effective and productive you'll be.

You can help developers create secure code from the start by integrating security tools directly into their workflow and into technologies they already use (e.g., integrated development environments). If a new tool requires developers to change their process or takes them away from their preferred systems, you'll have an uphill battle getting them to use it.

> "We get a 15 percent gain in productivity because defects are prevented early."
>
> -Jim Routh,
> Aetna Chief Information
> Security Officer

MEASURE

Many people believe you can't measure software security. How do you measure the absence of something happening?

Just because we don't yet know that something exists (like a security breach), it doesn't mean that it doesn't. And just because something such as a security breach hasn't happened yet, it doesn't mean it can't happen in the future.

Even if you can't prove that you prevented a hacker from penetrating your organization, there are many ways you can demonstrate results of your software security initiative.

Internal metrics help make continuous improvement toward business goals

When you set objectives for your SSI, tie them to underlying business goals. This way, when you share results, you'll be able to show how the SSI has fundamentally changed the way your organization operates.

By demonstrating to your Board that you're not only improving operational processes, but you're also getting software to market faster and saving money, you'll turn your security program from a series of "check the box" activities to an essential business function.

> "Absence of evidence is not evidence of absence!"
>
> \- Carl Sagan, Astronomer

10 Important Measurements That Can Demonstrate Continuous Software Security Improvement

1. Currency of software inventory, including robust characteristic and risk data for each entry
2. Percentage of all applications that are tested, whether as needed or periodically
3. Number of applications that undergo each type and each level of risk-based testing, from none to lightweight to in-depth
4. Number of variances required due to not meeting software security policy or compliance requirements
5. Time to fix various types of security defects
6. Number of security bugs and design flaws that make it all the way to production
7. Percentage of software projects, whether development or procurement, that go through all the secure SDLC gates
8. Time developers could have spent on activities other than fixing vulnerabilities
9. Frequency of delays, from requirements through production stages, stemming from software security issues
10. Number of applications that meet or exceed compliance requirements
11. Number of software security stakeholders that have the appropriate skill levels for their job

OK, so we turned it up to 11. But, hey, they're all important.

How to Speak the Language of the C-Suite

When you focus on measurements that executive leadership understands and values, you'll be more likely to get continued support for your software security initiative—or make an argument for more resources.

External metrics allow you to make comparisons to a broader universe of SSIs.

In addition to demonstrating internal improvements, you can also give your C-suite a broader perspective on your progress by comparing your SSI to that of other organizations. Let's face it: seeing what others do can be a powerful incentive for company leadership to take security seriously.

The leading industry-wide model for assessing and planning a software security initiative is the Building Software Security In Maturity Model (BSIMM). The BSIMM project benchmarks software security practices used by all types of organizations and proven to enhance software security. It provides a comparison of your program against a data-backed security industry standard.

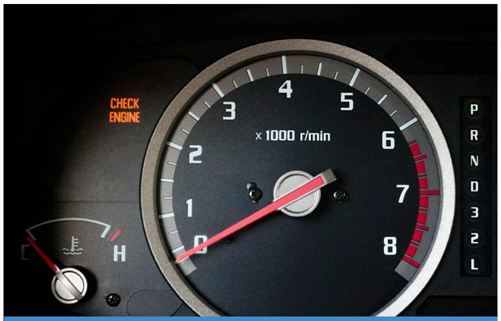
Consider conducting a BSIMM assessment and joining the BSIMM community. In addition to seeing how you measure up, you'll also have ongoing access to a group of folks who have built SSIs. You can learn from their experiences as your initiative evolves.

Learn more about BSIMM ▶

VERIFY

Now that you have policies and a measurement plan in place, you can set up checkpoints to verify whether the activities and requirements set forth in your SSI are being done and are having the impact you expect.

Think of the verification step this way: Your car needs to pass its safety inspection every year or two, but, if you see a "check engine" light turn on, you'll bring it into the shop sooner and run some tests to find out what's going on.



Defect discovery can be just like the "check engine" light in your car. It warns you when you need to address a problem with the system ASAP.

Instead of waiting until the end of a development cycle to squeeze in a complex security testing regimen, you can execute small tests along the way. In other words, you can change the philosophy of your testing from, "let's see whether this software is too horrible to release" to "let's verify that this software turned out as intended."

For organizations using Waterfall development, that means adding tests in the requirements phase, architecture phase, coding phase and QA phase. For Agile shops, that means building security into user stories and making sure developers can find and fix issues seamlessly.

You'll know if your SSI is working because your pre-launch security stage will be much, much shorter. No longer will you have a mass discharge of security issues that strain your capacity, delay launch and cause everyone heartache.

Change the philosophy of your testing from, "let's see whether this software is too horrible to release" to "let's verify that this software turned out as intended."

The Value of an External Perspective

If you're running your assessment and remediation work internally, it's good practice to get an external perspective from time to time. An external testing partner can give you an expert opinion to track whether your testing results are accurate and if your underlying system is able to defend against attacks.

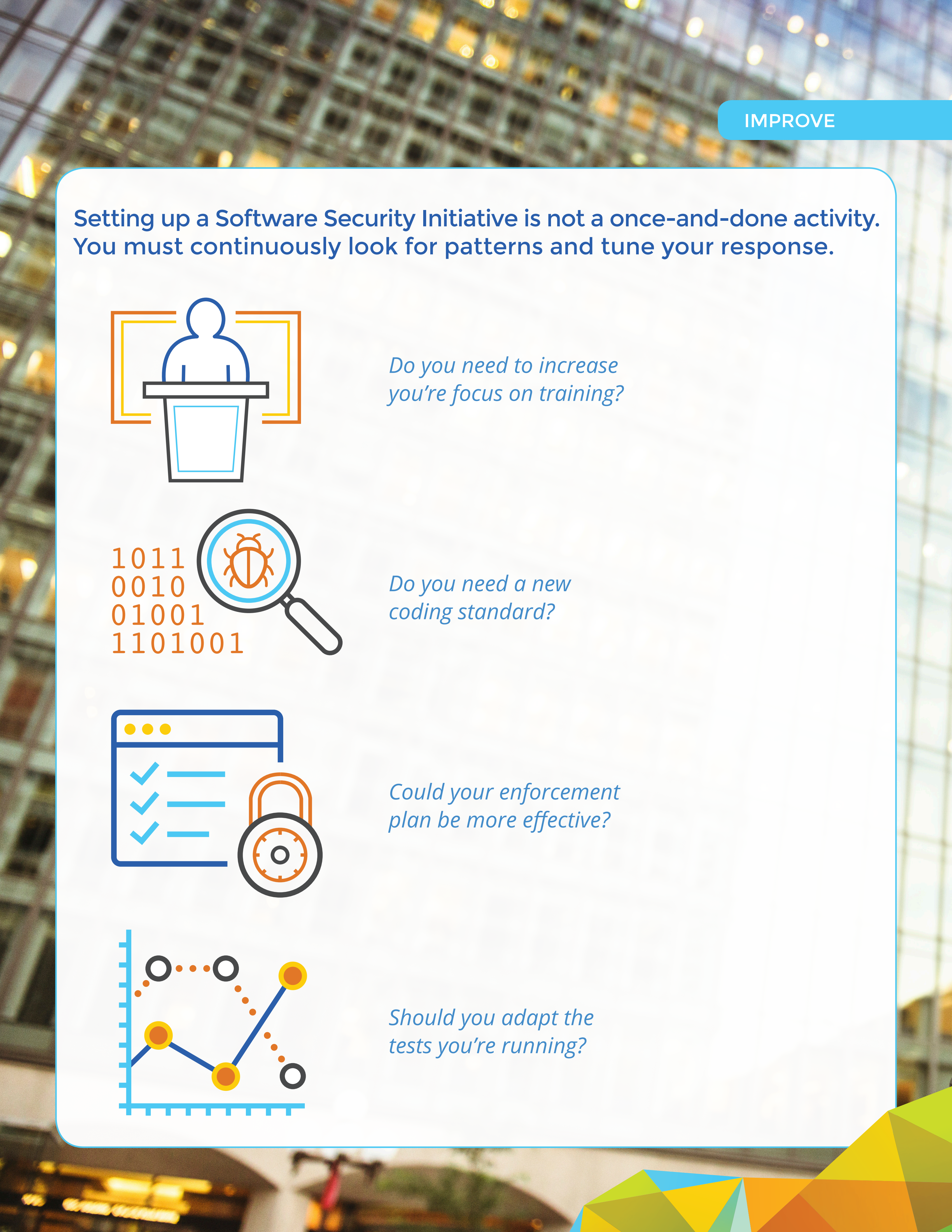
External application testing vendors have the necessary tools and manual testing strategies that help catch vulnerabilities your internal tools may miss. They can combine results to confirm suspicions and eliminate false positives. Most importantly, they can interpret results to help your team remediate any issues they find.

WARNING

Don't stop here!

If defect discovery IS your SSI, you'll never get better.

IMPROVE



Setting up a Software Security Initiative is not a once-and-done activity. You must continuously look for patterns and tune your response.

Do you need to increase you're focus on training?

Do you need a new coding standard?

Could your enforcement plan be more effective?

Should you adapt the tests you're running?

Setting Up an SSI Is As Easy As 1-2-3

Setting up an SSI is not a once-and-done activity. As you work with your initial SSI structure in the real-life environment of day-to-day pressures, you'll inevitably find areas for improvement. Remember, attackers are improving every day as are tools and techniques available to you.

1. **Watch Out for Patterns.** If the same security issues keep showing up in your verification process, you may need to tune your standards, increase your training, or provide more effective tools to developers. Or, you may find that vulnerabilities stem from fundamental design flaws and you'll have to go back to the design team for architecture changes.

2. **Prepare for Elastic Capacity.** Your SSI must be a living, breathing program that responds to changes in the number and type of applications in your portfolio, organizational shifts, updated compliance requirements and new attack vectors.

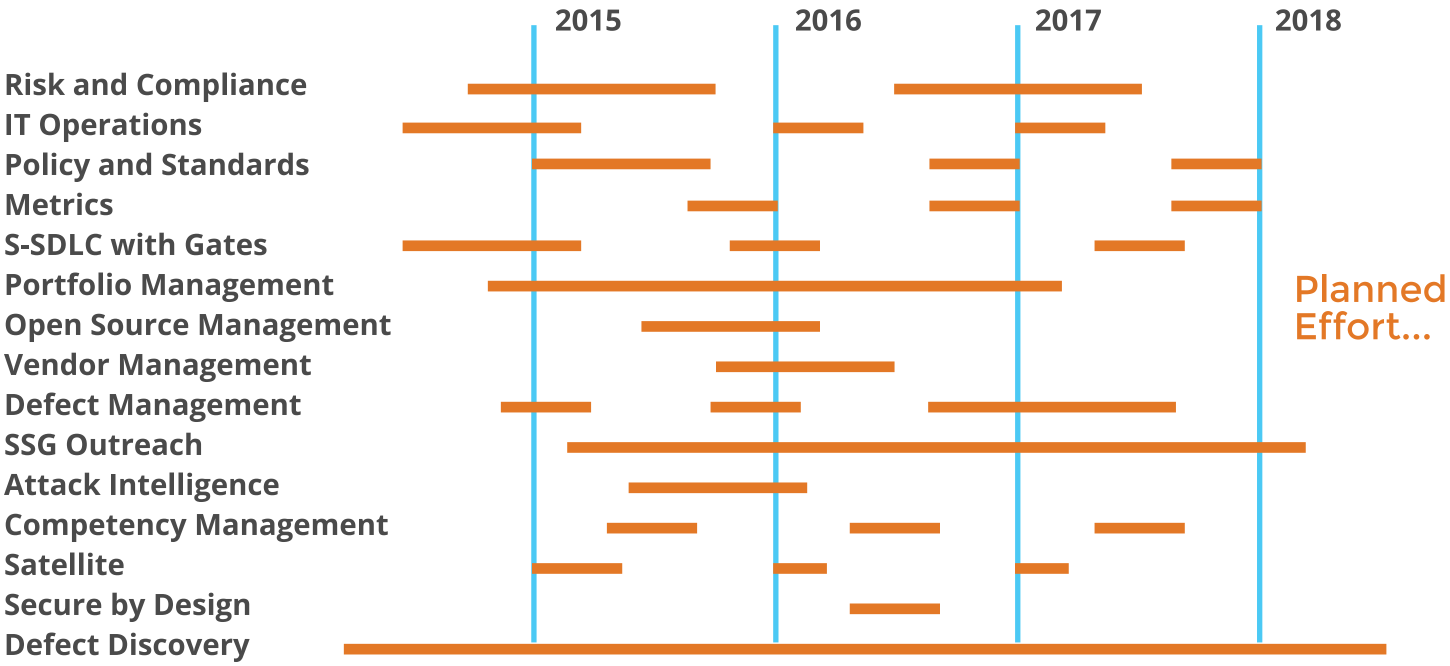
   Inevitably there will be times when demand for application testing outstrips your internal capacity. Finding the right testing partner can alleviate the burden from your internal team and help you maintain consistent testing and mitigation across all applications.

3. **Build a Roadmap for the Future.** Imagine a world in which your SSI is firing on all cylinders—with expertise in all software security capabilities, some of which are:

   - Risk and Compliance
   - Open Source Management
   - Vendor Management
   - Secure Architecture Design Review
   - Application Security Testing
   - Secure Code Review
   - Defect Management

You don't need to build expertise in every capability at the same rate, but you'll want to make sure you are making progress on a regular basis, setting the expectation that your SSI will continue to evolve as a business priority.

A Sample Roadmap Just for You



We've reached the next and final step of a successful SSI...Management.

MANAGE

As you have read so far, an effective SSI will have a lot of moving parts and many different people and departments involved. As the person at the helm, your job is to steer the ship so all of you stay the course.

To maintain structure and give you insight into each security activity, put in place a robust project management system that is responsive to your goals.

Look for a system that makes it easy to match your security activities to security gates in your development lifecycle and your timeline for software launches and upgrades. Compared with a generic project management tool, a security-specific tool will save you time and ensure you don't miss any key elements managed within the SSI.

A security-specific tool will save you time and ensure you don't miss any key elements managed within the SSI.

The right system will make it easy for you to run comparisons across time, application types, business units and specific projects. At a glance, you'll be able to track your progress, see where you may be lagging behind goals and which areas need additional attention.

Plus, you'll have timely, actionable data to report to company leadership.

In Conclusion

Every SSI will reflect its parent organization's structure and culture. Some firms will centralize management while others will federate. Some will rely on outsourced resources while others will hire new staff. Some will rely on managed services while others will grow their own technical teams.

This five-step process will set you on the path to success: You'll have greater alignment across all stakeholders in your development cycle, you'll demonstrate impact against business goals, and you'll have a rock-solid program that is built for the long term.

Cigital

SOFTWARE SECURITY
INITIATIVE IN-A-BOX

Ready to launch a Rock-Solid SSI, but still not sure how to start?

We've got everything you need all tied up with a bow. It's called Software Security Initiative In-a-Box (SSI-In-a-Box). Learn all about it.

I want to learn more about SSI-In-a-Box ▶

SSI CHEAT SHEET

5 Steps to a Rock-Solid Software Security Initiative

1. Build

❑ Gather information on your application portfolio, compliance requirements and areas of technical and business risk.

❑ Set up a governance structure, including ownership responsibilities and policies backed by leadership.

❑ Communicate broadly across internal teams and third-party vendors.

❑ Bring in the right internal and external resources to perform assessments and remediation activities defined in your SSI.

❑ Structure opportunities for staff to improve security skills and incentivize them to do so.

2. Measure

❑ Determine measurements to demonstrate continued progress and that are linked to underlying business goals.

❑ Compare your security practices to the **Building Security In Maturity Model (BSIMM)** and join the community.

3. Verify

❑ Build in defect discovery checkpoints throughout the development process, not just at the end.

❑ Compare your internal results with an external analysis to ensure accuracy and reduce false positives.

4. Improve

❑ Identify patterns and areas for additional resources, training and ongoing investment.

❑ Set up a roadmap to build expertise in software security capabilities.

5. Manage

❑ Set up a security-specific project management tool to help you manage and steer your SSI.

❑ Analyze and compare performance of teams and application types.

❑ Share progress with executive management and all stakeholders throughout the organization.

About Cigital

Cigital is one of the world's largest application security firms. We go beyond traditional testing services to help organizations find, fix and prevent vulnerabilities in the applications that power their business. Our holistic approach to application security offers a balance of managed services, professional services and products tailored to fit your specific needs. We don't stop when the test is over. Our experts also provide remediation guidance, program design services and training that empower you to build and maintain secure applications.

Our proactive methods helps clients reduce costs, speed time to market, improve agility to respond to changing business pressures and threats, and focus resources where they are needed most. Cigital's managed services maximize client flexibility, while reducing operational friction and cost. Cigital gives organizations of any size access to the scale, security expertise, and practices needed to build a successful software security initiative.

For more information, visit us at **www.Cigital.com**.

Interested in learning more? Check out these additional resources.

- **3 Reasons Software Security Governance is Essential to Your Business**
- **Building Meaningful Security Metrics**
- **Risk Ranking Your Applications: A Method to the Madness**
- **Scaling Automated Code Review**
- **Why a Software Security Group is Needed**