VERACODE EBOOK

TOP 6 TIPS

for Explaining Why Your Application Security Journey Is Just Beginning

VERACODE

AppSec Not a One-and-Done Project

Web application attacks are now the most frequent pattern in confirmed breaches, yet application security spending remains only a small fraction of overall security spending.*

Why?

Misconceptions abound regarding what application security is, and what makes it most effective.

NOT EFFECTIVE

A one-time scan or pen test of a handful of business-critical apps is *not effective* application security.

EFFECTIVE

A program that continuously assesses the applications an organization builds, buys or assembles — from inception to production — *is effective* application security.

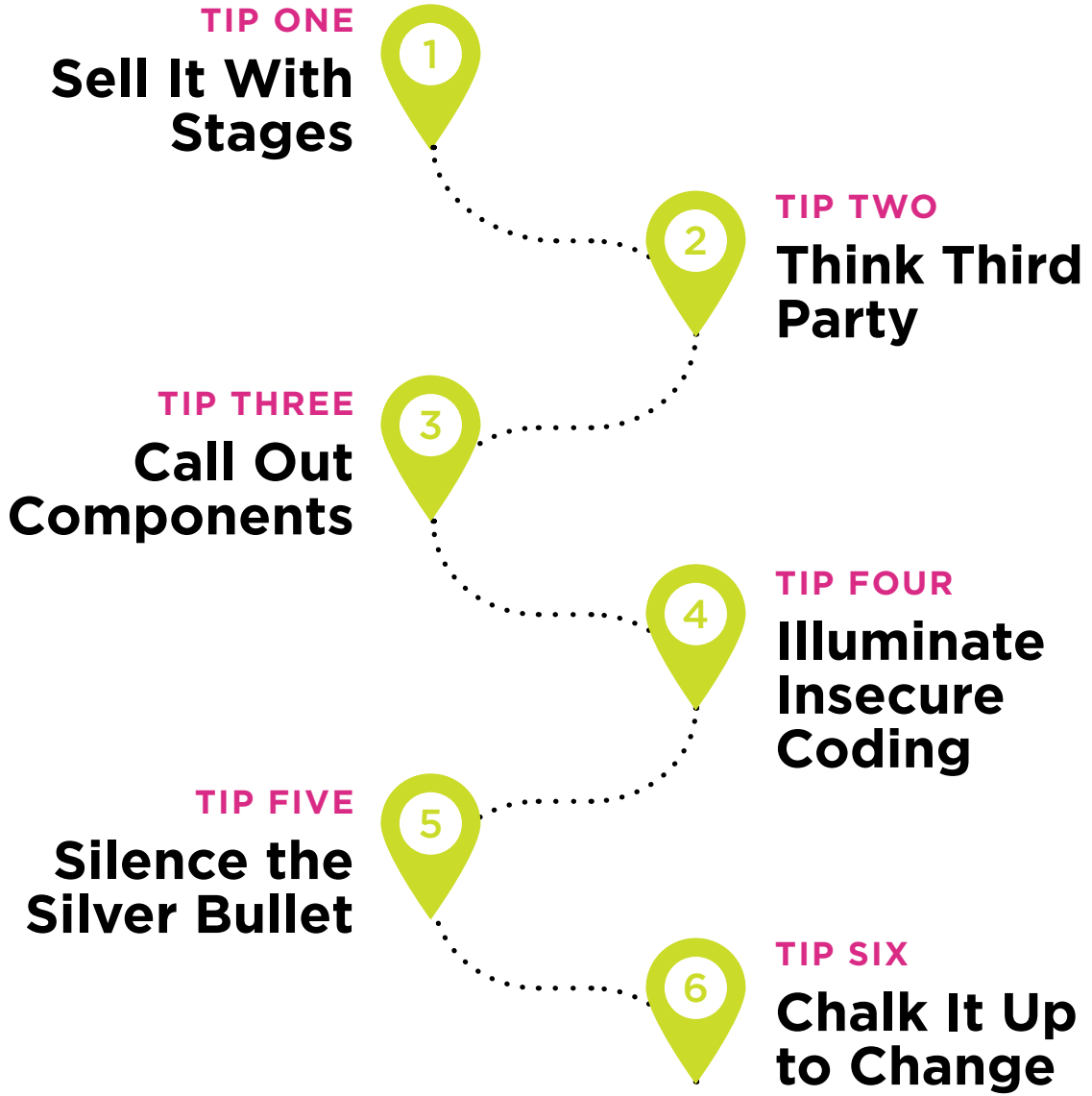
But this program can also seem unattainable.

Security professionals are often faced with the task of explaining why their application security initiative is insufficient and why a comprehensive program is both feasible, and critical.

*SOURCE: 2018 VERIZON DATA BREACH INVESTIGATIONS REPORT

USE THE FOLLOWING TIPS

To help *explain to expand* to anyone from a board member to a developer.



TIP ONE

Sell It With Stages

If the end goal overwhelms, sell it in stages. There are an established series of steps most organizations take when developing an application security program.

Divide the goal up into these manageable steps, then create and share a road map.

TIP IN PRACTICE

RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

The following phases, which will deliver a quick win, and then steady progress:

✓ **Phase 1: Pilot a program.**
Start small to demonstrate value. This phase includes getting a quick win by:

- Running a discovery scan of your web perimeter.
- Gaining an inventory of the most critical and easily exploitable vulnerabilities.
- Patching vulnerable sites or eliminating sites that are no longer in use.
- Measuring success and laying out the plan for scaling the program.

✓ **Phase 2: Set policies and metrics.**
Consider focusing on the OWASP Top 10 vulnerabilities, or reducing your flaw density by a set percentage.

✓ **Phase 3: Scale**
to assess all internally developed applications and integrate in the SDLC.

✓ **Phase 4: Address externally developed code.**
Create a strategy for assessing third-party applications and components.

SAY IT WITH STATS

Web application attacks are now the most frequent pattern in confirmed breaches.

2018 VERIZON DATA BREACH INVESTIGATIONS REPORT

FIND OUT MORE

Ultimate Guide to Getting Started With Application Security

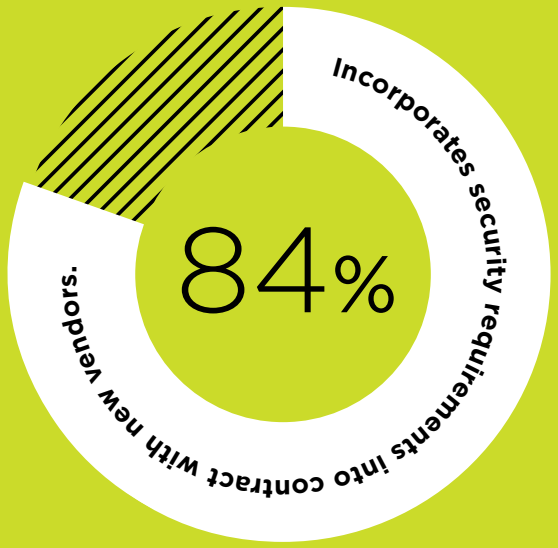
TIP TWO

Think Third Party

Securing your most business-critical applications is a good place to start, not to stop.

In fact, some of the most damaging recent breaches stemmed from vulnerabilities in third-party software. For instance, JPMorgan suffered a major breach through a third-party website created for the bank's annual charity race.

SAY IT WITH STATS



We recently conducted a survey that found that, when doing business witha new software vendor, 84%of respondents' organizations always or frequently incorporate security requirements into the contract.

SECURITY AS A COMPETITIVE ADVANTAGE

FIND OUT MORE

Look for software vendors with security certifications like our Verified program.

TIP IN PRACTICE

RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

✓ Policies that require third-party software to adhere to the same standards as internally developed software.

TIP THREE

Call Out Components

With the extreme pressure on developers to get working code delivered quickly, it's a common development practice to use pre-built open source software components and code.

But, as we learned from Heartbleed and Shellshock, these components often contain serious vulnerabilities that expose organizations to significant risk.

TIP IN PRACTICE

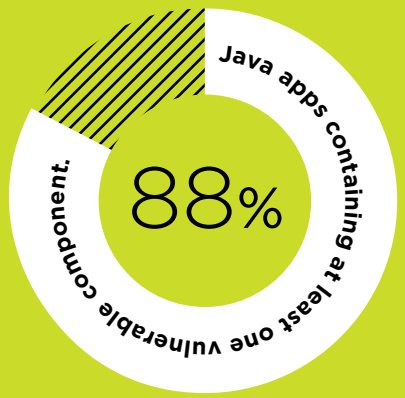
RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

Technologies to keep track of which applications are using each component and what versions are being used. This gives your organization an easy way to update a component to the latest version if a vulnerability is discovered.

FIND OUT MORE

A Best Practice Guide to Managing Your Open Source Risk

SAY IT WITH STATS



Our recent State of Software Security data found that 88% of Java apps contain at least one vulnerable component.

TIP FOUR

Illuminate Insecure Coding

Many organizations narrow their application security focus to scanning and patching code at the end of their development cycle. But this expensive, time-consuming model is not sustainable.

By expanding application security into earlier phases of development, you can address the development behaviors, policies and habits that can lead to vulnerable code in the first place. You can also remediate vulnerabilities in less time and with less money than you would later in the cycle. By stopping the problem at its source in this way, you will eventually begin producing more secure code more quickly.

TIP IN PRACTICE

RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

- ✓ Secure programming education. Recent Veracode research revealed that implementing an eLearning program has a big impact on vulnerability remediation, as well as on reduction in overall flaw density.
- ✓ A systematic process for assessing code and fixing vulnerabilities during the development stage, when flaws are easier and less expensive to fix, rather than at the end of the cycle.

SAY IT WITH STATS

30x

More expensive to fix a vulnerability during post-production than during earlier stages.

THE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST)

FIND OUT MORE

Secure Coding Best Practices Handbook ›

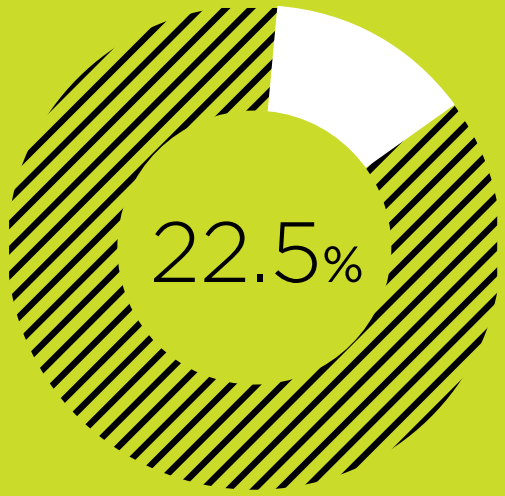
TIP FIVE

Silence the Silver Bullet

Some organizations don't expand their application security programs beyond one technology, because they are under the impression that one technology or testing method is all they need.

However, there is no AppSec silver bullet, and a truly effective application security program uses the strengths of multiple testing techniques. Each analysis technology has its own strengths. For example, static analysis (SAST) can find vulnerabilities earlier in the development cycle, when they're easier and less expensive to fix. But dynamic analysis (DAST) finds runtime issues, such as authentication issues and server misconfiguration issues, that can't easily be found when the code is in its offline state.

SAY IT WITH STATS



Only 22.5% of applications are compliant with OWASP Top 10 standards when initially assessed for security.

READ FULL REPORT

TIP IN PRACTICE

RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

- ✓ Static and dynamic analysis
- ✓ Software composition analysis
- ✓ Manual penetration testing

FIND OUT MORE

Your Guide to Application Security Solutions

TIP SIX

Chalk It Up to Change

Thinking of application security as a one-and-done project leaves you open to attack. Hacking isn't a one-and-done activity — cyberattackers spend all their time and resources looking for holes in your code; if they're thwarted in one pursuit, they'll come up with another one. And software itself isn't stagnant either — code is constantly being changed and updated.

You can't stop code from changing or hackers from plotting, but you can make it harder for them.

SAY IT WITH STATS

In a recent study, one in five business leaders indicated that their software budget had increased 50 percent or more over the past three years.

SECURING THE DIGITAL ECONOMY

TIP IN PRACTICE

RECOMMEND THAT YOUR APPLICATION SECURITY PROGRAM INCLUDE:

The ability to assess the security of code multiple times, and at different stages of development — including when changes are made post-production.

FIND OUT MORE

How Do Vulnerabilities Get Into Software?

CONCLUSION

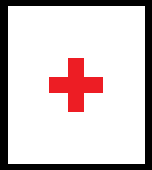
The application layer is far-reaching and fluid; you can't put up a wall or turn on a device to secure your apps.

Application security involves multiple:

- ✓ Departments
- ✓ Development stages
- ✓ Coding methods
- ✓ Vulnerability types

Application security won't be sufficiently addressed with a one-off project:

Forward-thinking organizations are reducing their risk and moving their businesses forward with ongoing, comprehensive application security programs.



Get more details on the application security journey, from someone who's lived it, in our new guide, **From Ad Hoc to Advanced Application Security: Your Path to a Mature AppSec Program.**