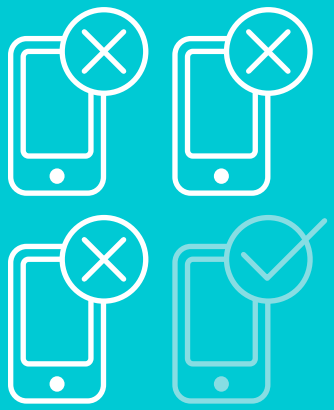
VERACODE GBOOK

Combating the **Top 4 Sources** of Software Vulnerabilities

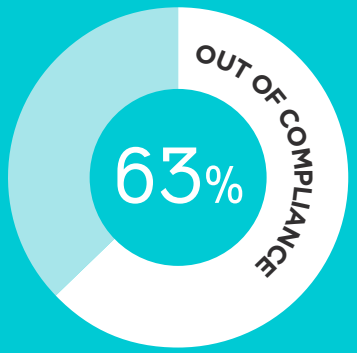
VERACODE

Vulnerable software is leading to breaches...



3 out of 4 applications produced by software vendors fail OWASP Top 10 when first assessed. Applications produced in-house don't fare much better.

VERACODE'S STATE OF SOFTWARE SECURITY, VOLUME 6



63% of internally developed applications are out of compliance with OWASP Top 10 when first assessed.

VERACODE'S STATE OF SOFTWARE SECURITY, VOLUME 6

How important are these software vulnerabilities?

In a word: Very.

90%

of security incidents result from exploits against defects in software.

THE U.S. DEPARTMENT OF HOMELAND SECURITY

Top 4 Sources

OF SOFTWARE VULNERABILITIES

1. Insecure coding practices
2. The ever-shifting threat landscape
3. The reuse of vulnerable components and code
4. Idiosyncrasies of programming languages

SOURCE #1

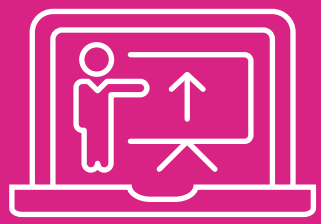
Insecure coding practices

How to combat:

- Create a process for assessing code during the development stage.
- Gain developer buy-in during the planning phase of your application security program.
- Use an eLearning tool to help increase developer understanding of secure coding practices.

Nearly one third of companies never scan for vulnerabilities during code development.

INTERNATIONAL INFORMATION SYSTEMS SECURITY CERTIFICATION CONSORTIUM (ISC)[2]



Development organizations that leverage eLearning see a 30% improvement in fix rate compared to those that do not.

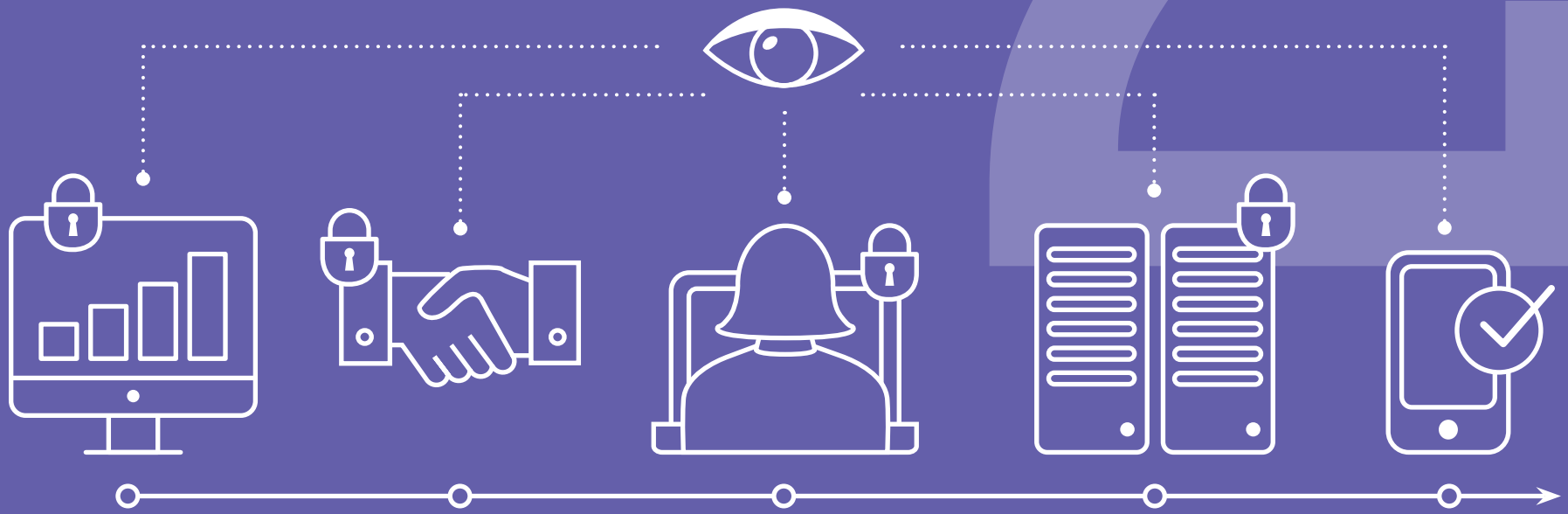
VERACODE'S STATE OF SOFTWARE SECURITY, VOLUME 6

SOURCE #2

The ever-shifting threat landscape

How to combat:

Assess for security at different stages of the development processes — including when post-production updates are made.



SOURCE #3

The reuse of vulnerable components and code

How to combat:

Use technologies to track which applications are using which components and what versions are being used.

Components introduce an average of

24

known vulnerabilities into each application.

VERACODE'S ANALYSIS OF MORE THAN 5,300 ENTERPRISE APPLICATIONS UPLOADED TO ITS PLATFORM

Remember Heartbleed and Shellshock?

That's the end result of vulnerabilities in software components.

SOURCE #4

Idiosyncrasies of programming languages

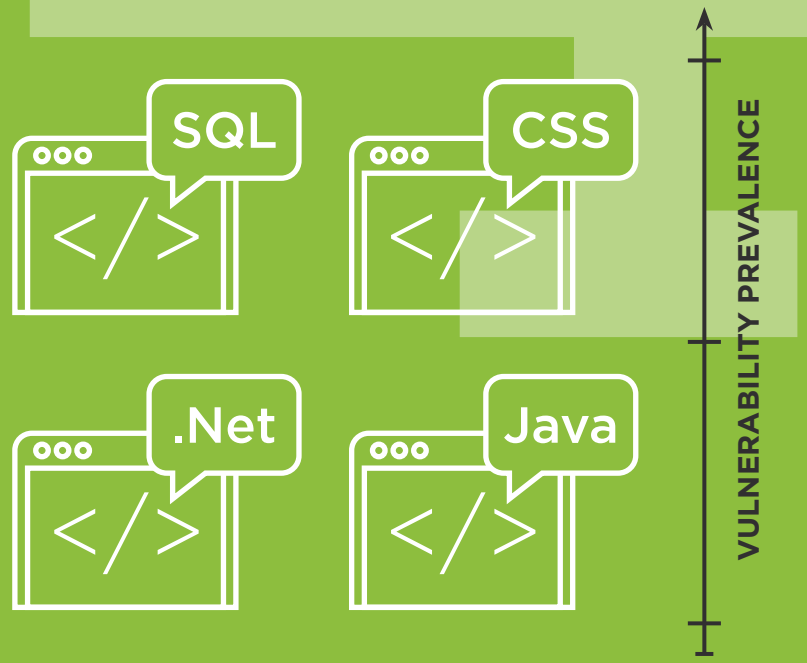
How to combat:

Understand the vulnerability trends associated with the languages dev is using, and architect and test accordingly.

Each programming language is susceptible to different kinds of vulnerabilities.

For example: Applications written in web scripting languages have a higher prevalence rate of vulnerability classes like SQL injection and Cross-Site Scripting than applications written in .Net or Java.

VERACODE STATE OF SOFTWARE SECURITY REPORT, VOLUME 6



> When organization are starting new development projects and selecting languages and methodologies, the security team has an opportunity to anticipate the types of vulnerabilities that are likely to arise and how to best test for them.

CHRIS WYSOPAL

FIND OUT MORE ABOUT MAKING THE CASE FOR AN APPLICATION SECURITY PROGRAM WITH OUR NEW GUIDE:

TOP 6 TIPS FOR EXPLAINING WHY YOUR APPLICATION SECURITY JOURNEY IS JUST BEGINNING.