



WhiteSource

# The State of Open Source Security Vulnerabilities

---

WhiteSource Annual Report 2020

John Timberlake  
October 8th, 2020

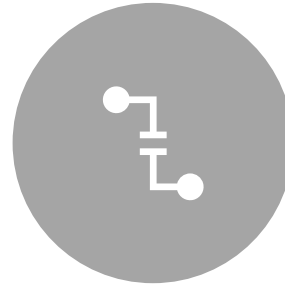


# TOPICS



## VULNERABILITY TRENDS

*Trends & number of open source vulnerabilities published this year*



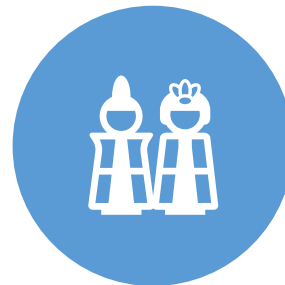
## DEVSECOPS INSIGHTS

*Security vs. Developers:  
The DevSecOps Showdown*



## MOST COMMON CWES

*Open source security vulnerabilities in popular programming languages*



## ADDITIONAL RESOURCES

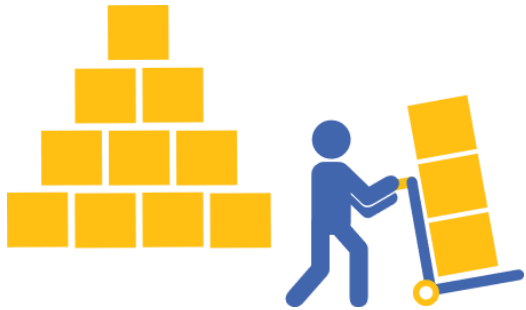
# VULNERABILITY TRENDS



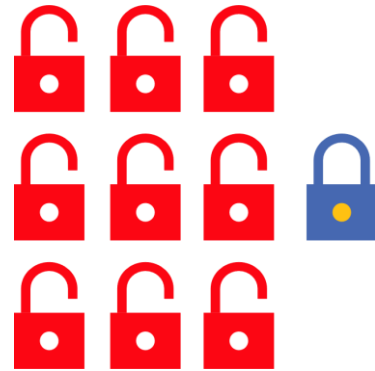


Don't Worry –  
It's free right?

# LIFE WITHOUT OSS MANAGEMENT AUTOMATION



**Over 75%** were aware of only 50% of their open source inventory.



90% of apps have at least **1 vulnerability**, over 45% have 5 or more.



**Broken dialog with Dev.** Tough to explain where vulnerabilities are, and where the risk is.

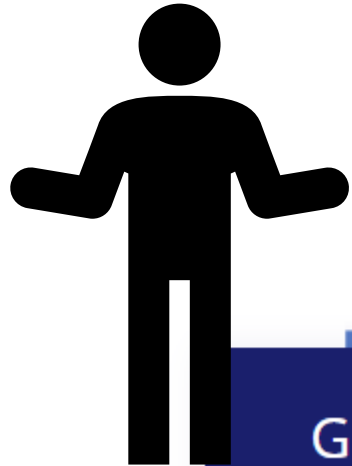


At least **1 license** that doesn't meet company policy on average.

# WHO IS ON THE HOOK TO FIX?



Legal



Security?



Governance Solutions

Development?



Developers Tools

# Chris Roberts

*Go follow him!*

- 'Security can't happen in a silo.'
- 'Stop trying to figure out who to blame and figure out what we can do better everyday.'

<https://www.linkedin.com/in/sidragon1/>



## POLL:

# What are the biggest challenges in implementing & running your AppSec program?

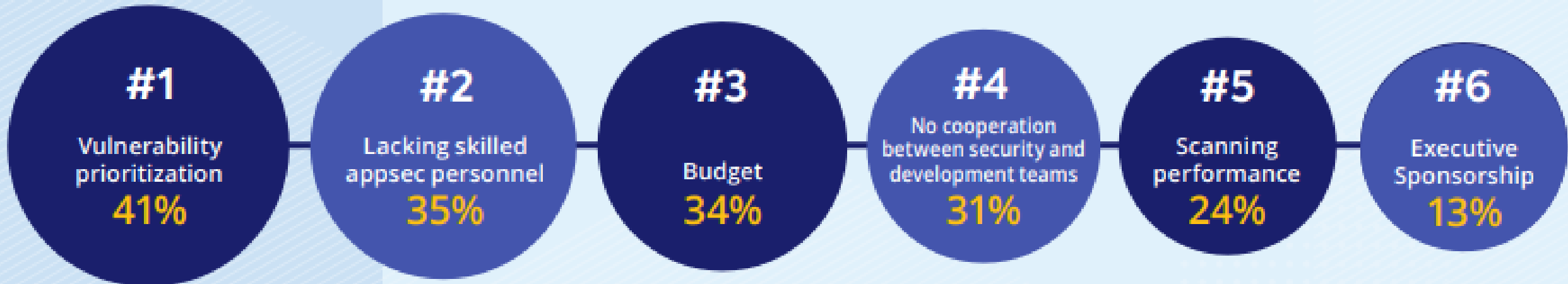
- Scanning Performance
- Budget
- Lacking Skilled AppSec Personnel
- Vulnerability Prioritization
- No Cooperation between security & development teams





# Survey Results

What are the biggest challenges in implementing and running your AppSec program?





Vulnerability  
Detection



Vulnerability  
Remediation

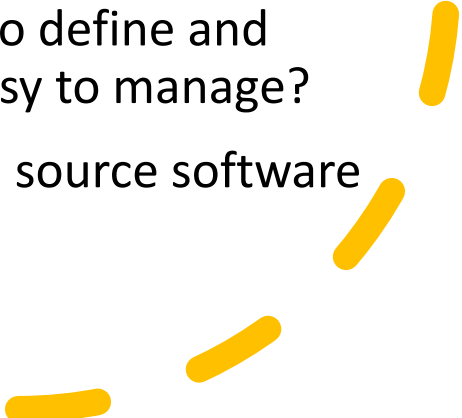


Inventory  
Management

Open Source License  
Compliance



# 7 Questions to Ask When Evaluating SCA

- **Question 1:** Does the SCA solution scale to meet your enterprise needs by offering both governance and developer tools?
  - **Question 2:** Does the SCA solution offer vulnerability prioritization advice and minimal false positives to reduce the number of alerts?
  - **Question 3:** Does the SCA solution automatically remediate vulnerabilities?
  - **Question 4:** Does the SCA solution support all the programming languages you currently use or plan to use?
  - **Question 5:** Does the SCA solution integrate into your DevOps pipeline?
  - **Question 6:** Does the SCA solution allow you to define and automate policies that are both robust and easy to manage?
  - **Question 7:** Does the SCA solution cover open source software in containers?
- 



WHY NOW?





# OPEN SOURCE SECURITY VULNERABILITIES ARE ON THE RISE

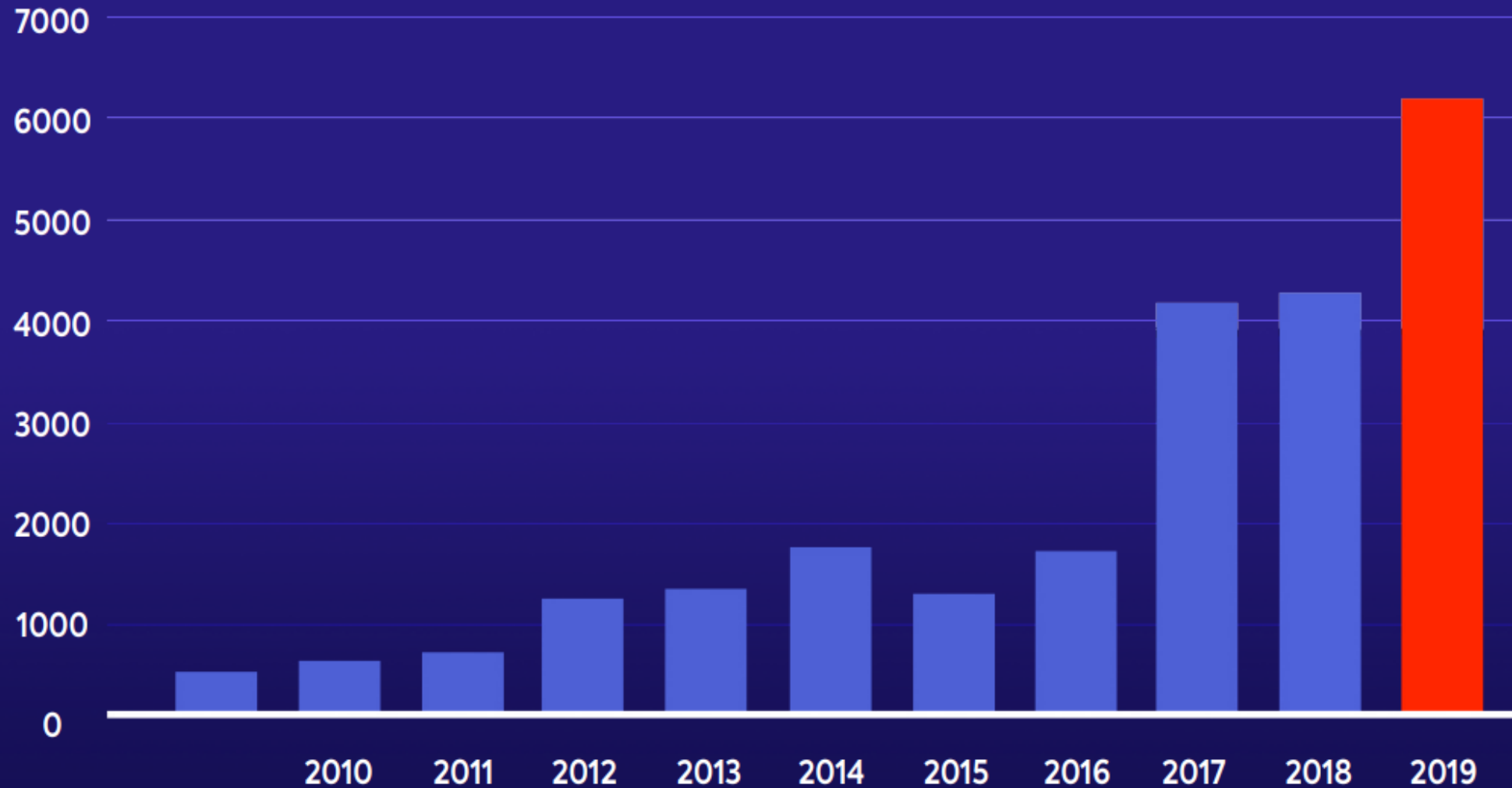
## **Key Takeaway:**

A significant rise in the number of open source vulnerabilities presents a serious challenge to development and security teams striving to meet security objectives.

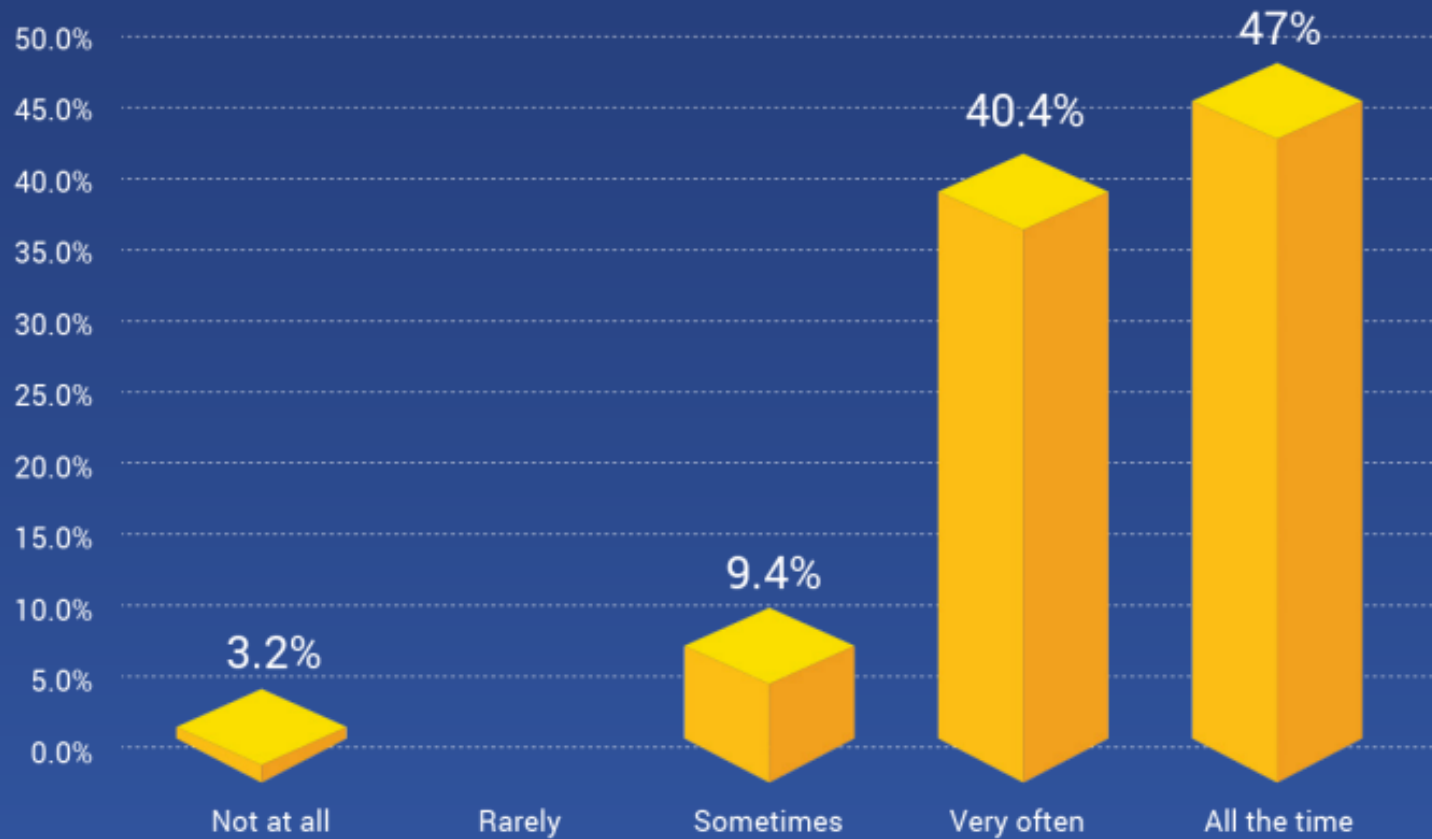


# Open Source Security Vulnerabilities per Year

\*Source: 2020 WS Annual Report



## FREQUENCY OF USE OF OPEN SOURCE COMPONENTS



\* Source: 2018 WS Annual Report



Over **85%** of open source security vulnerabilities are disclosed with a fix already available.

---

Only **84%** of known open source vulnerabilities eventually appear in the NVD.

---

\* Source: 2020 WS Annual Report



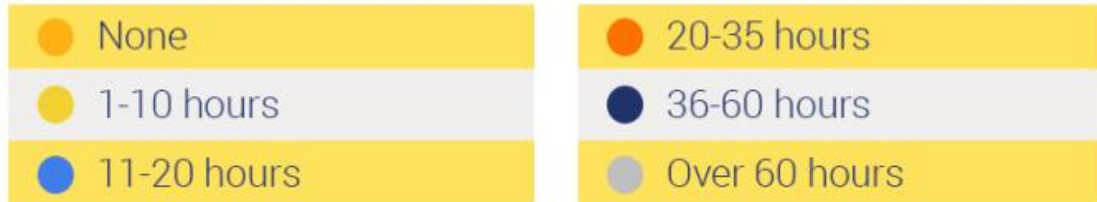
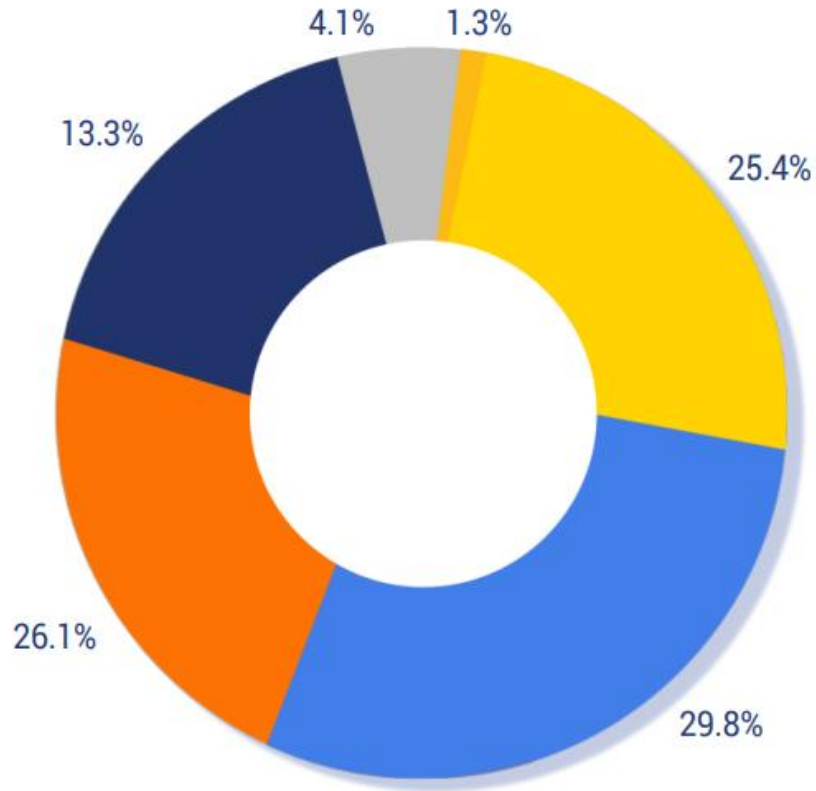


# DEVELOPERS ARE NOT EFFICIENTLY MANAGING OPEN SOURCE VULNERABILITIES

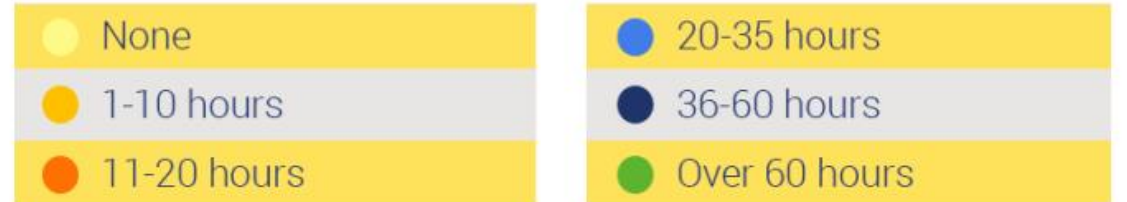
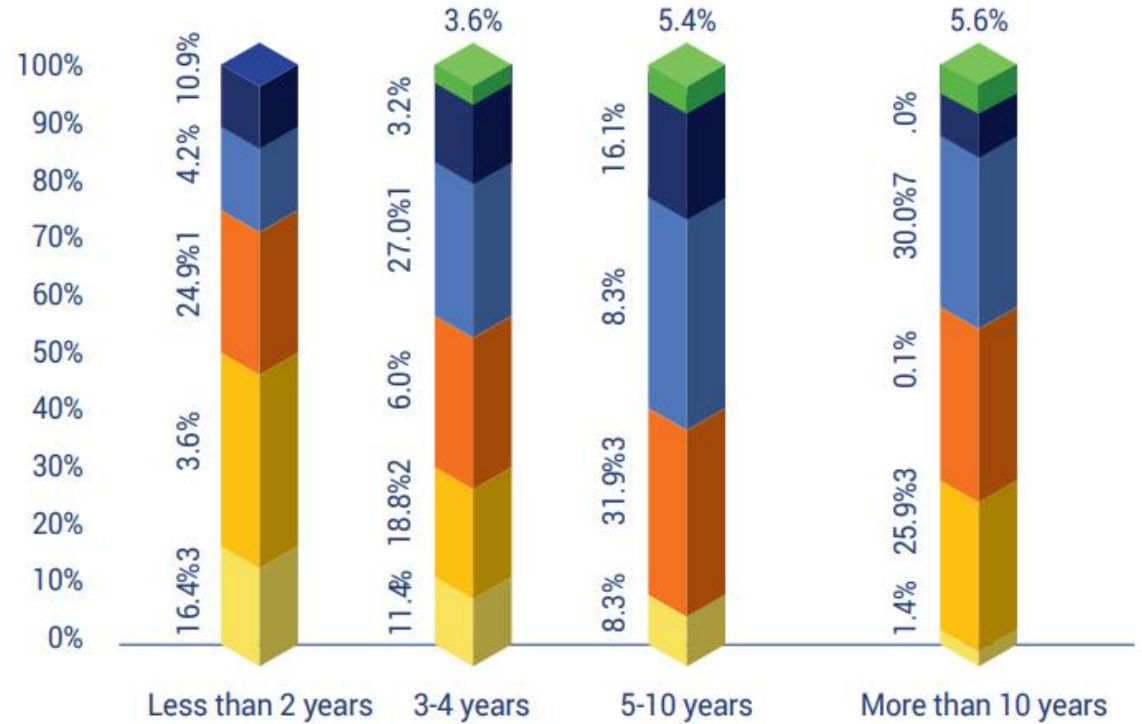
## **Key Takeaway:**

Developers spend a lot of time addressing open source vulnerabilities, but the absence of standard practices and developer-focused tools result in an inefficient use of time.

## HOURS SPENT PER MONTH HANDLING OPEN SOURCE VULNERABILITIES



## HOURS SPENT ON OPEN SOURCE VULNERABILITIES PER DEVELOPERS' EXPERIENCE



\* Source: 2018 WS Annual Report



# PRIORITIZATION IS KEY TO OPEN SOURCE VULNERABILITY MANAGEMENT

## **Key Takeaway:**

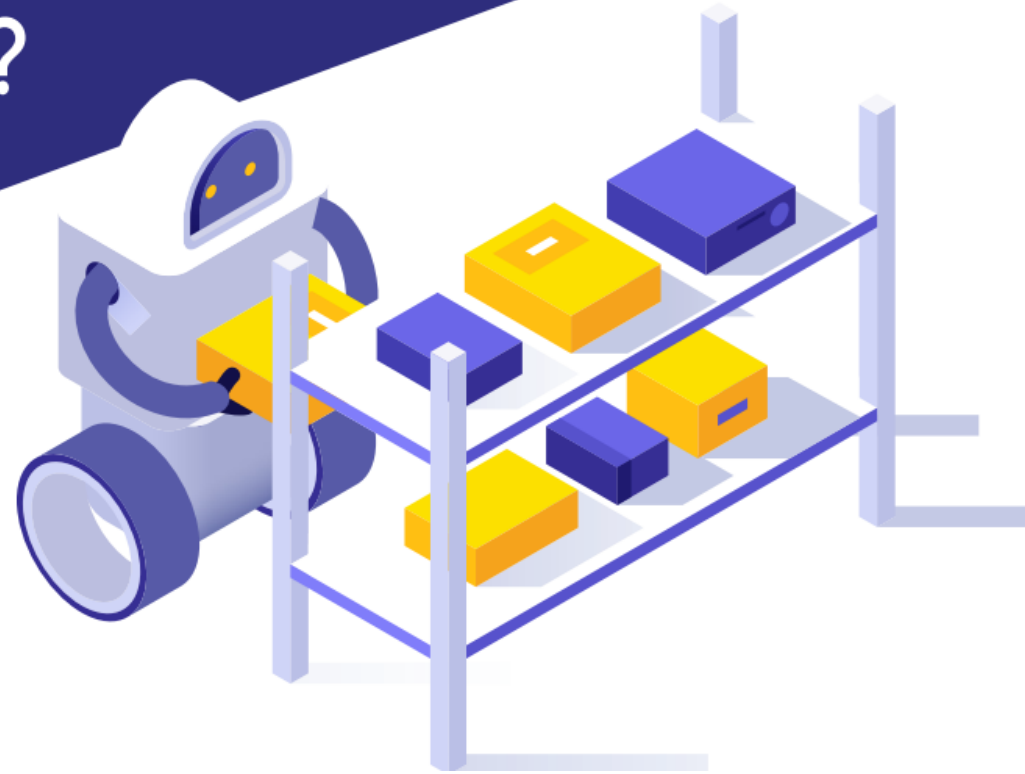
A prioritization strategy for open source vulnerabilities is critical to ensure companies address the most critical issues on time.



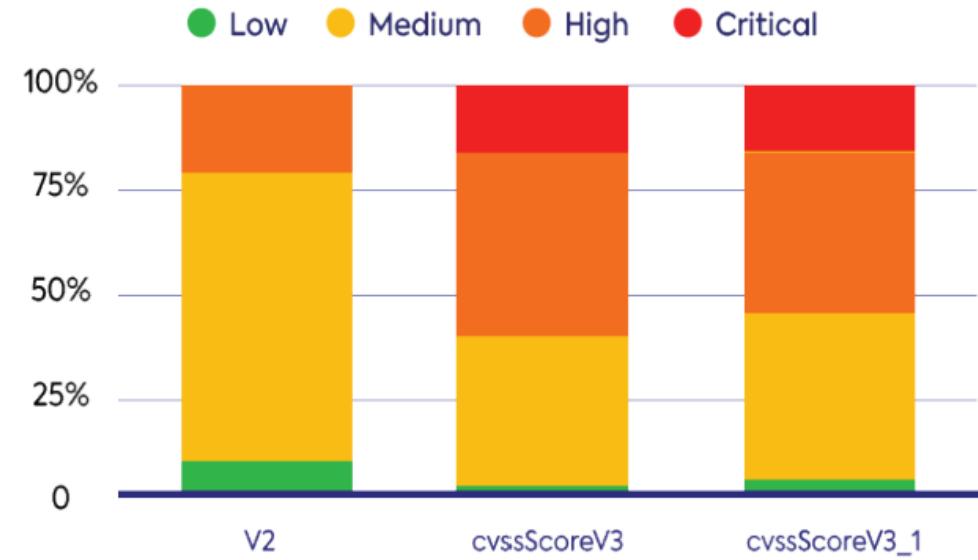
# VULNERABILITY SEVERITY SCORING: AN OBJECTIVE PRIORITIZATION STANDARD?

---

The rising number of reported vulnerabilities demands that development teams quickly prioritize their security alerts. The CVSS (Common Vulnerability Scoring System) score is usually the go-to parameter for remediation prioritization, but should it be?



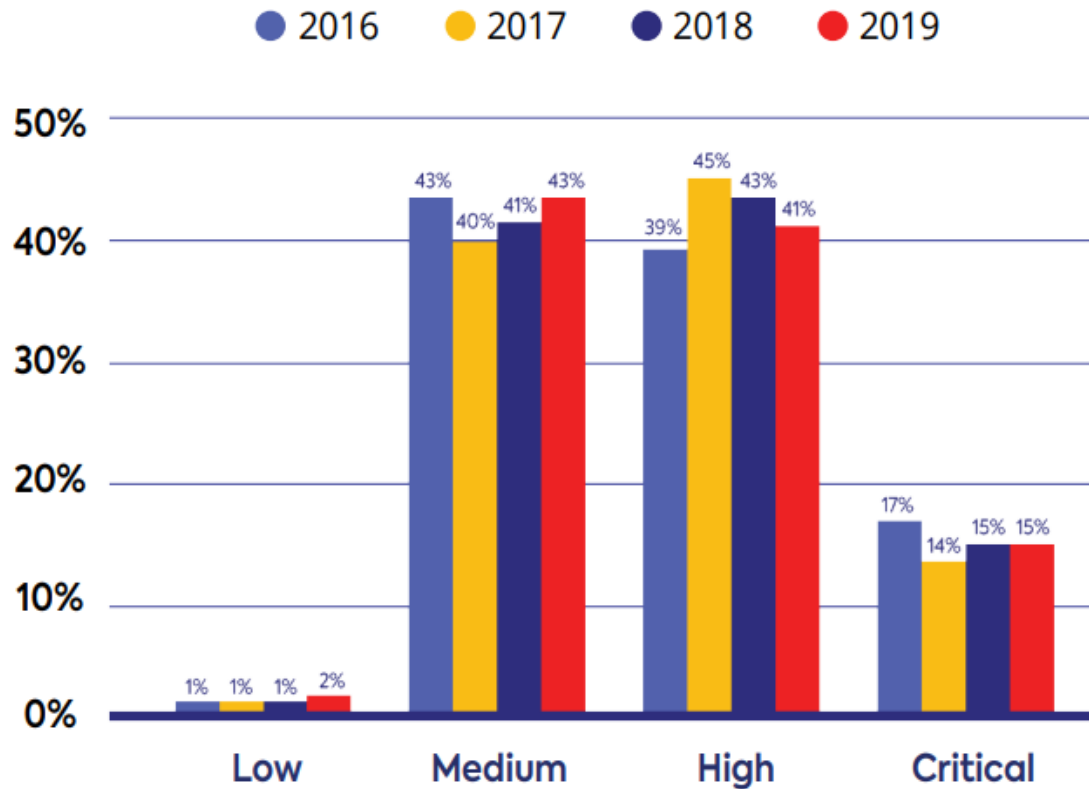
## Severity Break-down: CVSSv2.0, vs. CVSSv3.0, vs. CVSSv3.1



CVSS v2.0 Ratings		CVSS v3.0 Ratings	
Low	0.0-3.9	Low	0.1-3.9
Medium	4.0-6.9	Medium	4.0-6.9
High	7.0-10.0	High	7.0-8.9
		Critical	9.0-10.0

\* Source: 2020 WS Annual Report

## CVSS v3.x Severity Breakdown over time



How can we expect teams to prioritize vulnerabilities efficiently when over 55% are high-severity or critical?

\* Source: 2020 WS Annual Report

The 5 Most common practices to prioritize remediation are:

---

Severity

---

Application Type

---

Popularity

---

Disclosure Date

---

Ease of Remediation



# Diving Into the Evil Internet:

Vulnerability Prioritization Through The Eyes of Hackers



David Habusha,  
VP Product  
WhiteSource



Paulo Shakarian,  
CEO  
CYR3CON





# Two Prioritization Strategies

## 1. Prioritizing based on CWE Type

- The more common a CWE, the more hackers will study it, learn to exploit it, and discuss it.

## 2. Identifying Effective vs. Non-Effective Vulnerabilities

- Is the proprietary code making a call to the vulnerable function?



# Which CWE's Do We Need To Watch Out For In 2020?

---



# Most Common CWE's in 2019

\*Source: 2020 WS Annual Report



	1	2	3	4	5
2019	CWE-79 Cross-site scripting (XSS)	CWE-20 Improper Input Validation	CWE-119 Buffer Errors	CWE-125 Out-of-bounds Read	CWE-200 Information Exposure
2018	CWE-79 Cross-site scripting (XSS)	CWE-119 Buffer Errors	CWE-20 Improper Input Validation	CWE-125 Out-of-bounds Read	CWE-200 Information Exposure
2017	CWE-119 Buffer Errors	CWE-125 Out-of-bounds Read	CWE-79 Cross-site scripting (XSS)	CWE-200 Information Exposure	CWE-20 Improper Input Validation
2016	CWE-119 Buffer Errors	CWE-20 Improper Input Validation	CWE-200 Information Exposure	CWE-264 Permissions, Privileges, and Access Control	CWE-284 Improper Access Control
2015	CWE-119 Buffer Errors	CWE-79 Cross-site scripting (XSS)	CWE-264 Permissions, Privileges, and Access Control	CWE-200 Information Exposure	CWE-20 Improper Input Validation

\* Source: 2020 WS Annual Report

# Most Common CWE's per Year 2014-2019

\*Source: 2020 WS Annual Report



1

CWE-79

Cross-site  
Scripting

CWE-79

Cross-site  
Scripting

CWE-79

Cross-site  
Scripting

CWE-79

Cross-site  
Scripting

CWE-79

Cross-site  
Scripting

CWE-119

Improper  
Restriction of  
Operations within  
the Bounds of a  
Memory Buffer

2

CWE-20

Improper Input  
Validation

CWE-20

Improper Input  
Validation

CWE-20

Improper Input  
Validation

CWE-20

Improper Input  
Validation

CWE-200

Information  
Exposure

CWE-125

Out-of-bounds  
Read

3

CWE-200

Information  
Exposure

CWE-200

Information  
Exposure

CWE-200

Information  
Exposure

CWE-200

Information  
Exposure

CWE-20

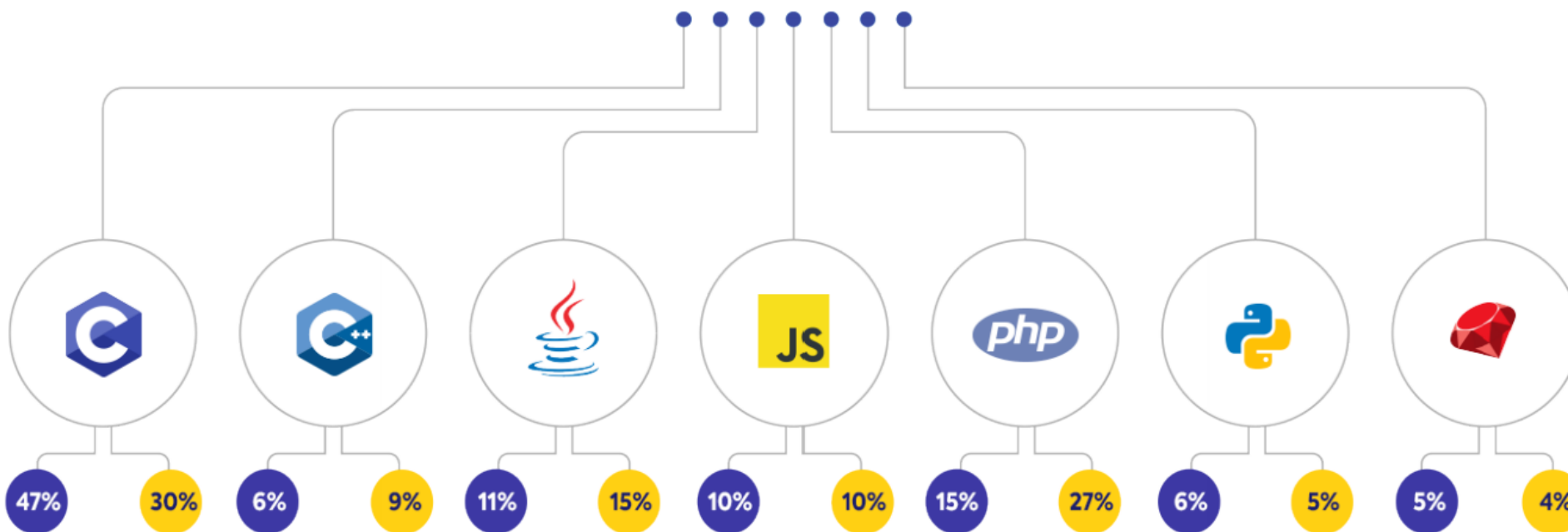
Improper Input  
Validation

CWE-476

NULL Pointer  
Dereference

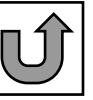
# Which Programming Languages Are MOST SECURE?

Open Source Vulnerabilities per Language, 2019 vs. 2009-2018



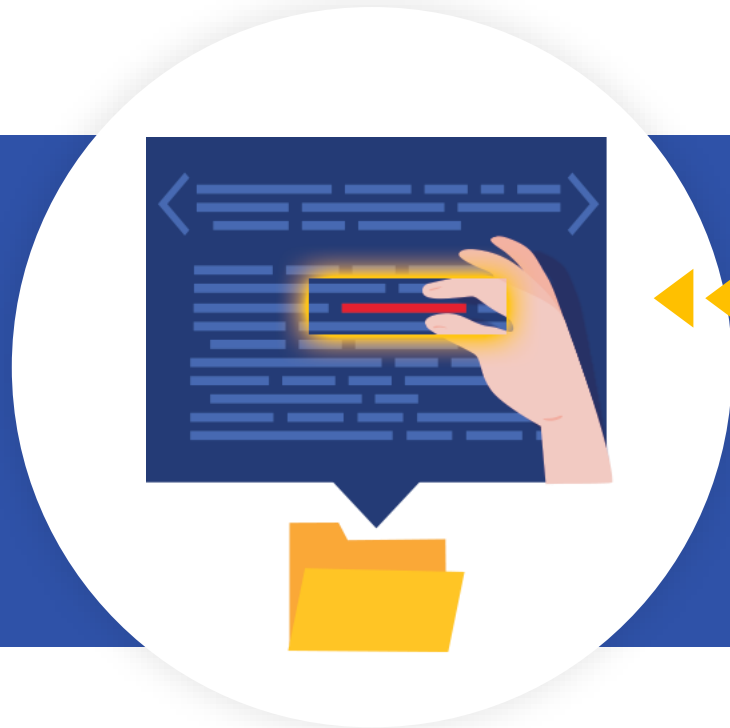
● 2009-2018 ● 2019

\* Source: 2020 WS Annual Report



# WHY FIX ALL VULNERABILITIES

WHEN ONLY **15%-30%** IMPACT YOUR PRODUCTS?



EFFECTIVE  
VS  
INEFFECTIVE





# PRIORITIZE BASED ON EFFECTIVENESS

Top Alerts 13  All  Security Ignore Selected

<input type="checkbox"/>	Library	Type	Description	Occurrences
<input type="checkbox"/>	<span style="color: red;">●</span> jackson-databind-2.9.2.jar	Security Vulnerability	<span style="color: red;">✖</span> High: 3 (2?) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> plexus-archiver-3.4.jar	Security Vulnerability	<span style="color: red;">✖</span> High: 1 (1) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> spring-core-4.3.1.RELEASE.jar	Security Vulnerability	<span style="color: red;">✖</span> High: 1 (1) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> junrar-0.7.jar	Security Vulnerability	<span style="color: red;">✖</span> Medium: 1 (1) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> spring-web-4.3.1.RELEASE.jar	Security Vulnerability	<span style="color: red;">✖</span> Medium: 2 (1) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> zip4j-1.3.2.jar	Security Vulnerability	<span style="color: gray;">?C</span> Medium: 1 (0?) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> bcprov-jdk15on-1.50.jar	Security Vulnerability	<span style="color: gray;">?C</span> High: 7 (0?...) Medium: 4 (0?...) ...	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> commons-collections-3.2.1.jar	Security Vulnerability	<span style="color: green;">✔</span> High: 3 (0) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> guava-20.0.jar	Security Vulnerability	<span style="color: green;">✔</span> Medium: 1 (0) <a href="#">details</a>	1 project <a href="#">details</a>
<input type="checkbox"/>	<span style="color: red;">●</span> vertx-web-3.5.0.jar	Security Vulnerability	<span style="color: green;">✔</span> High: 1 (0) <a href="#">details</a>	1 project <a href="#">details</a>

[Show Reported Vulnerabilities](#) [Show Only Effective Vulnerabilities](#) [View All Alerts](#)





# OPTIMIZE REMEDIATION PROCESSES

Traces		Trace View	
Selected Reference: (1) - <b>com.fasterxml.jackson.databind.deser.BeanDeserializerFactory ()</b>			
Caller Traces (3)			
Trace	Caller Type	Caller ID (hover for full text)	
1	EXTENSION	(9)com.fasterxml.jackson.databind.deser.BeanDeserializerFactory:createBuilderBasedDeserializer (...\\deser\\BeanDeserializerFactory.class:188)	↑
1	↑ EXTENSION	(8)com.fasterxml.jackson.databind.deser.DeserializerCache:_createDeserializer (...\\deser\\DeserializerCache.class:318)	
1	↑ EXTENSION	(7)com.fasterxml.jackson.databind.deser.DeserializerCache:_createAndCache2 (...\\deser\\DeserializerCache.class:264)	
1	↑ EXTENSION	(6)com.fasterxml.jackson.databind.deser.DeserializerCache:_createAndCacheValueDeserializer (...\\deser\\DeserializerCache.class:228)	
1	↑ EXTENSION	(5)com.fasterxml.jackson.databind.deser.DeserializerCache:findValueDeserializer (...\\deser\\DeserializerCache.class:139)	
1	↑ EXTENSION	(4)com.fasterxml.jackson.databind.DeserializationContext:findRootValueDeserializer (...\\databind\\DeserializationContext.class:477)	
1	↑ EXTENSION	(3)com.fasterxml.jackson.databind.ObjectMapper:_findRootDeserializer (...\\databind\\ObjectMapper.class:4173)	
1	↑ EXTENSION	(2)com.fasterxml.jackson.databind.ObjectMapper:_readMapAndClose (...\\databind\\ObjectMapper.class:3986)	
1	↑ EXTENSION	(1)com.fasterxml.jackson.databind.ObjectMapper:readValue (...\\databind\\ObjectMapper.class:2890)	
1	↑ APPLICATION	(0)org.whitesource.fs.configuration.ConfigurationSerializer:load (...\\configuration\\ConfigurationSerializer.class:54)	↓

# WHITESOURCE DEVSECOPS INSIGHTS

Security vs. Developers: The DevSecOps Showdown

Round 2!  
INSIGHTS



01

Most security professionals and developers feel forced to compromise on security in order to meet deadlines.

02

AppSec tools are purchased to 'check the box', disregarding developers' needs and processes.

KEY  
INSIGHTS

03

Huge gaps in AppSec knowledge and skills among developers are neglected by organizations.

04

Security professionals' top challenge is vulnerability prioritization, but the lack of standardized processes leads to friction with developers.

01

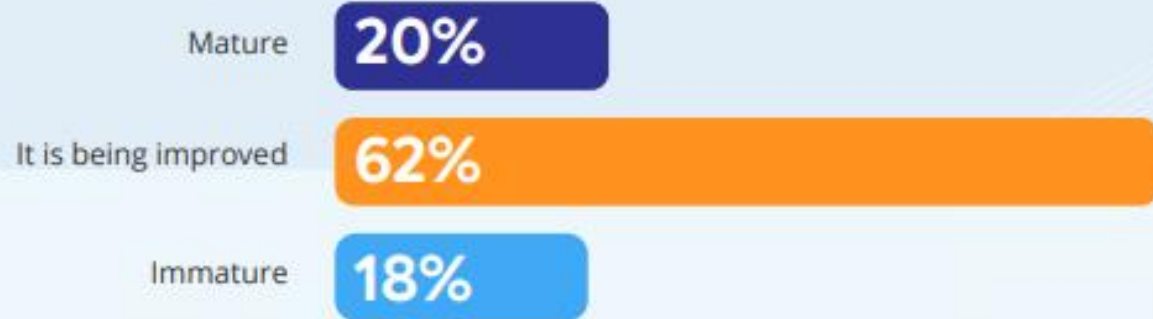
73%

OF SECURITY PROFESSIONALS AND DEVELOPERS FEEL FORCED TO COMPROMISE ON SECURITY

Most respondents think that they are in the process of DevSecOps maturity

Most security professionals and developers believe their organizations are in the process of adopting DevSecOps tools and practices.

How would you describe the maturity of your organization's DevSecOps practices?



Are you forced to compromise on security to meet short deployment cycles?

YES  
73%

Which feature is the most important when it comes to developers adopting certain AppSec tools?

● Security ● Developers

Ease of integration



Accuracy -- I don't like wasting time



Easy to use



Native integrations into development environments



Real-time feedback



Remediation advice



**POLL:**

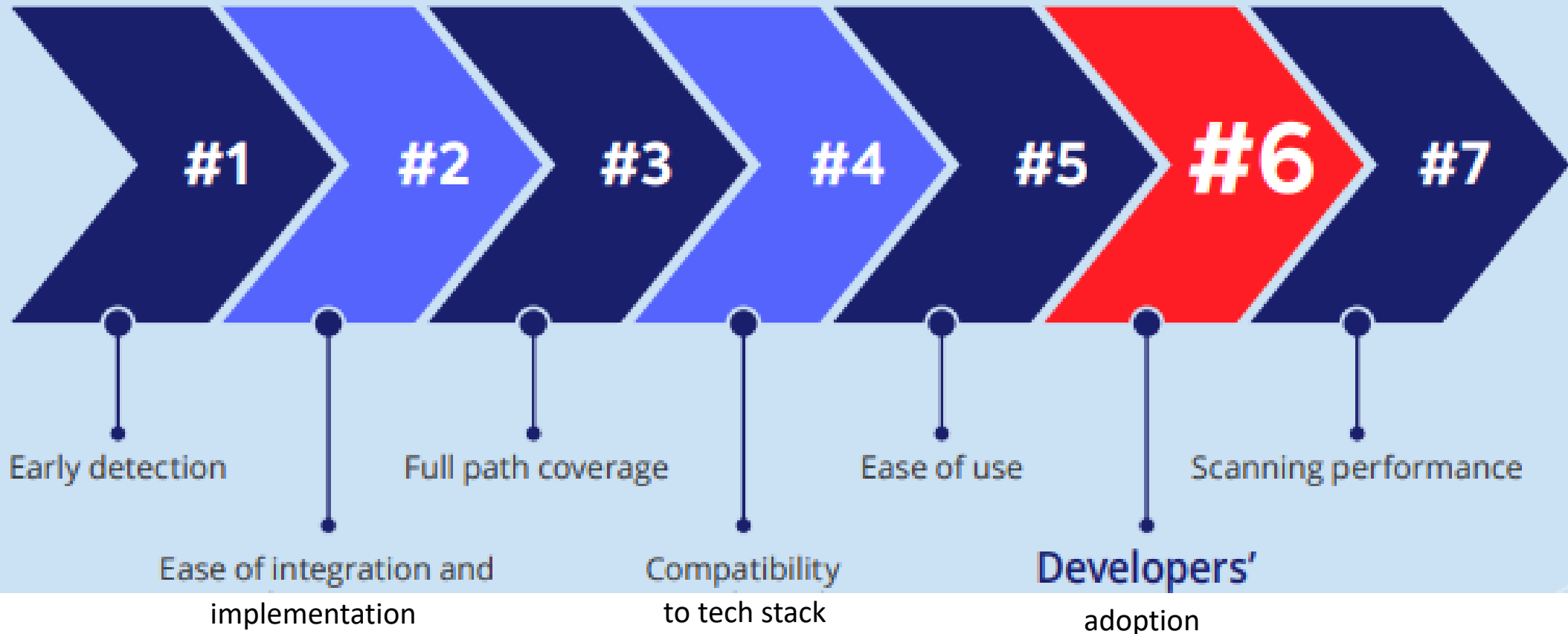
**What feature is MOST important to security?**

- Compatibility with tech stack
- Developers Adoption
- Early Detection
- Full Path Coverage
- Ease of Integration & implementation



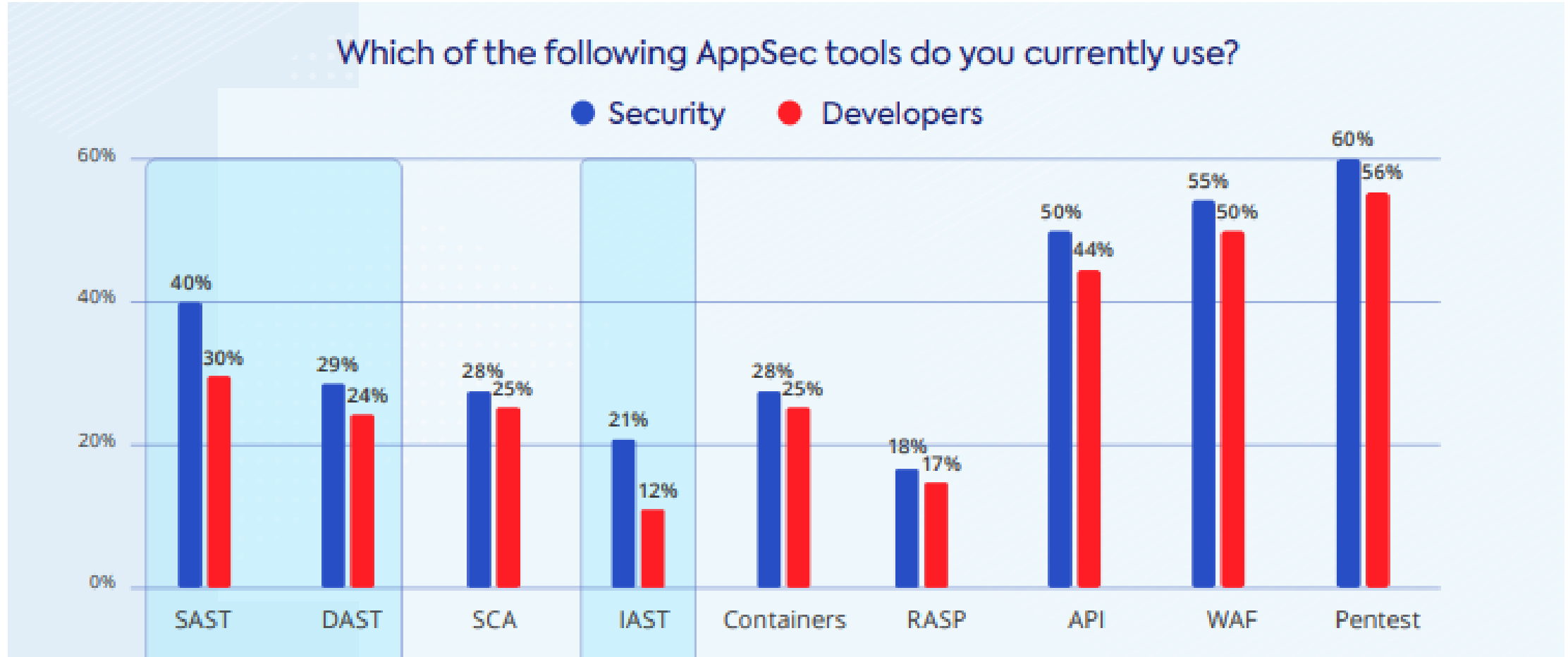
# WHAT IS IMPORTANT TO SECURITY

When considering an AppSec tool, which of the following are most important to you?



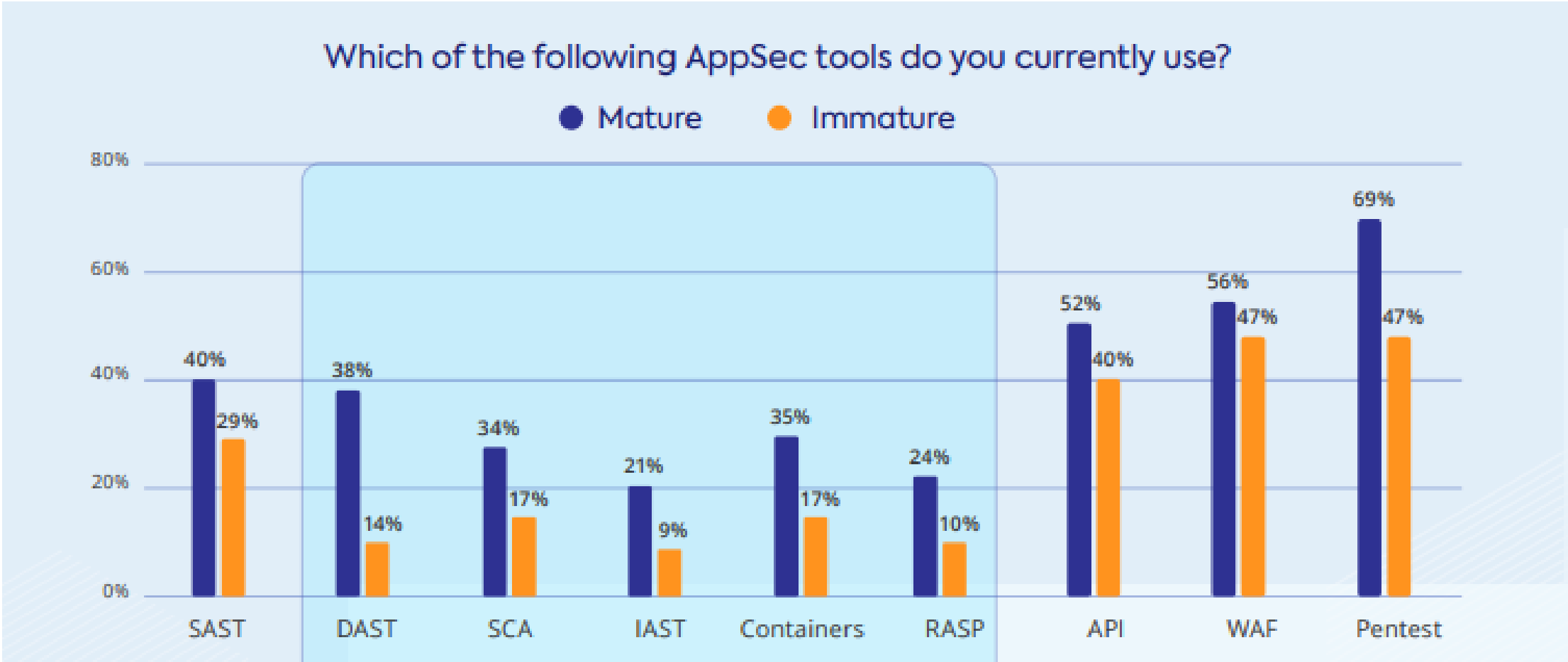
# WHO IS USING YOUR TOOLS?

**SAST:** Static application security testing  
**DAST:** Dynamic application security testing  
**SCA:** Software composition analysis  
**IAST:** Interactive application security testing  
**RASP:** Runtime application self-protection  
**API:** Application programming interact  
**WAF:** Web application firewall





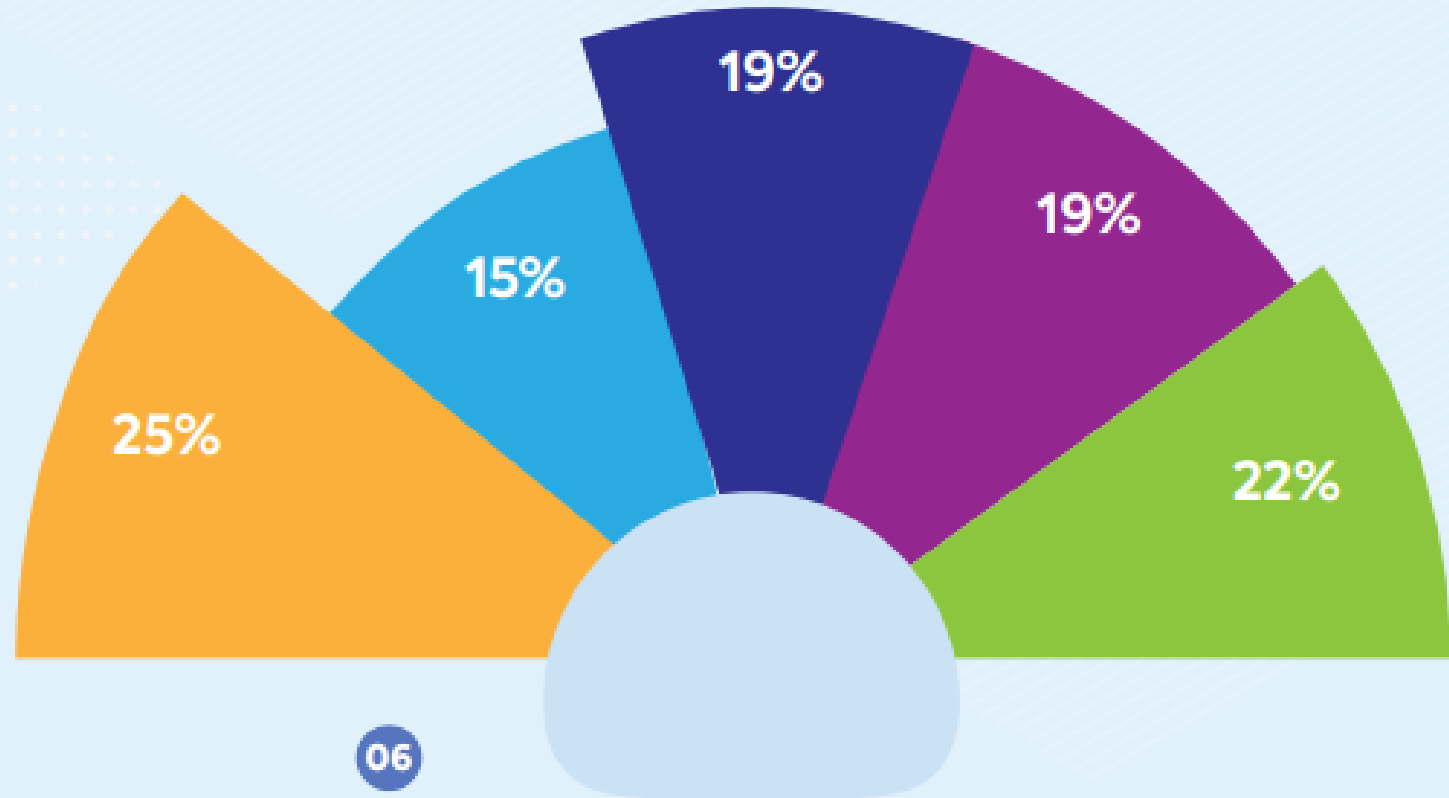
# MATURE = MORE TOOLS



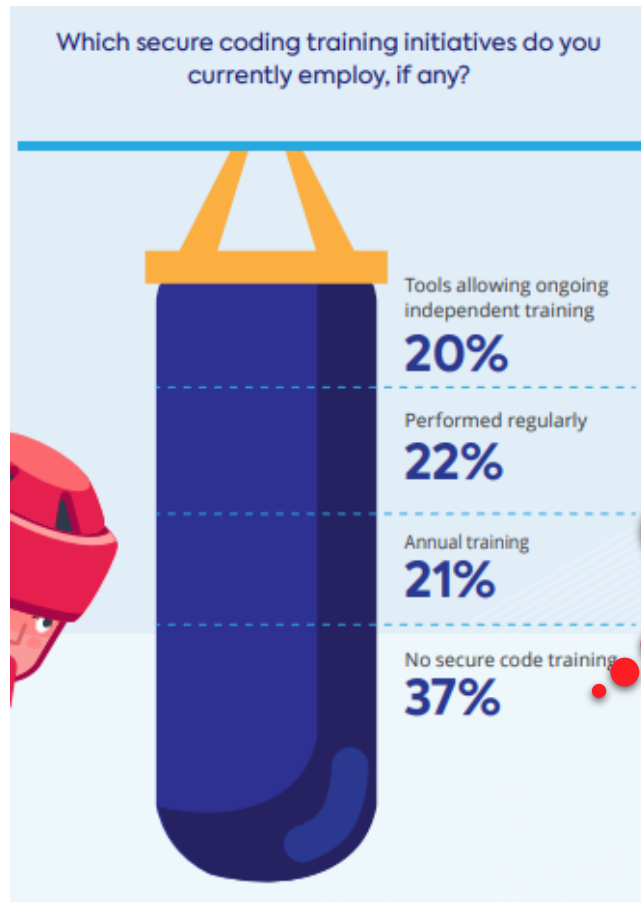
# WHY DO COMPANIES ACQUIRE TOOLS?

How do you justify purchasing new application security tools?

- Meeting industry-specific regulations (HIPAA, PCI etc.)
- Direct response to security audit findings
- Using well-known public incidents to demonstrate benefit (or risk)
- Including AppSec costs in general IT security spending
- Compliance with industry standards such as ISO/IEC 27034



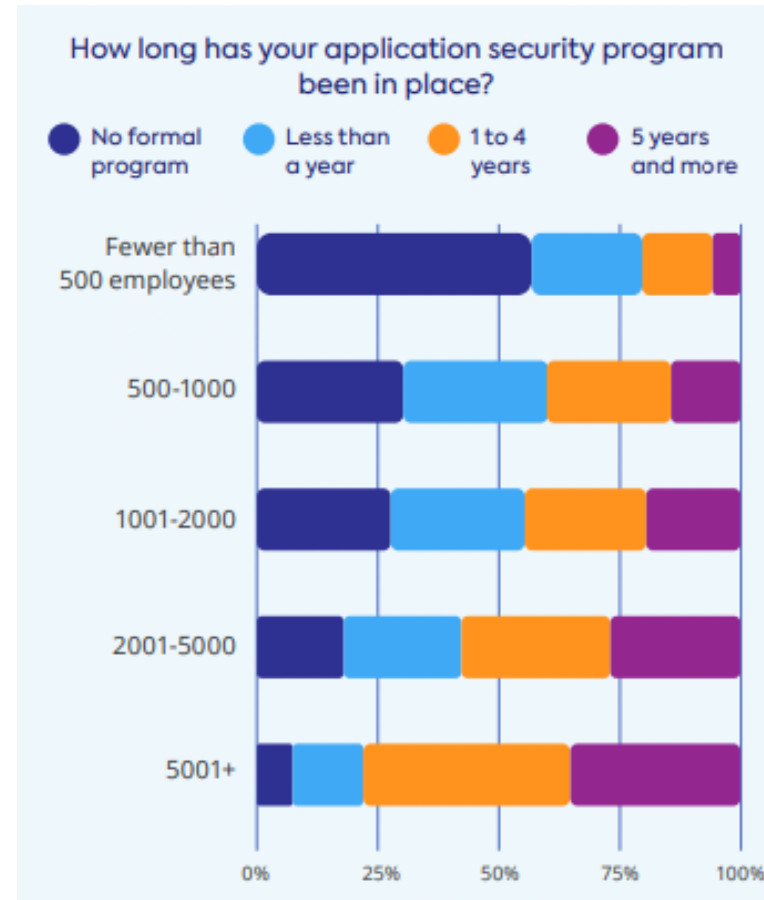
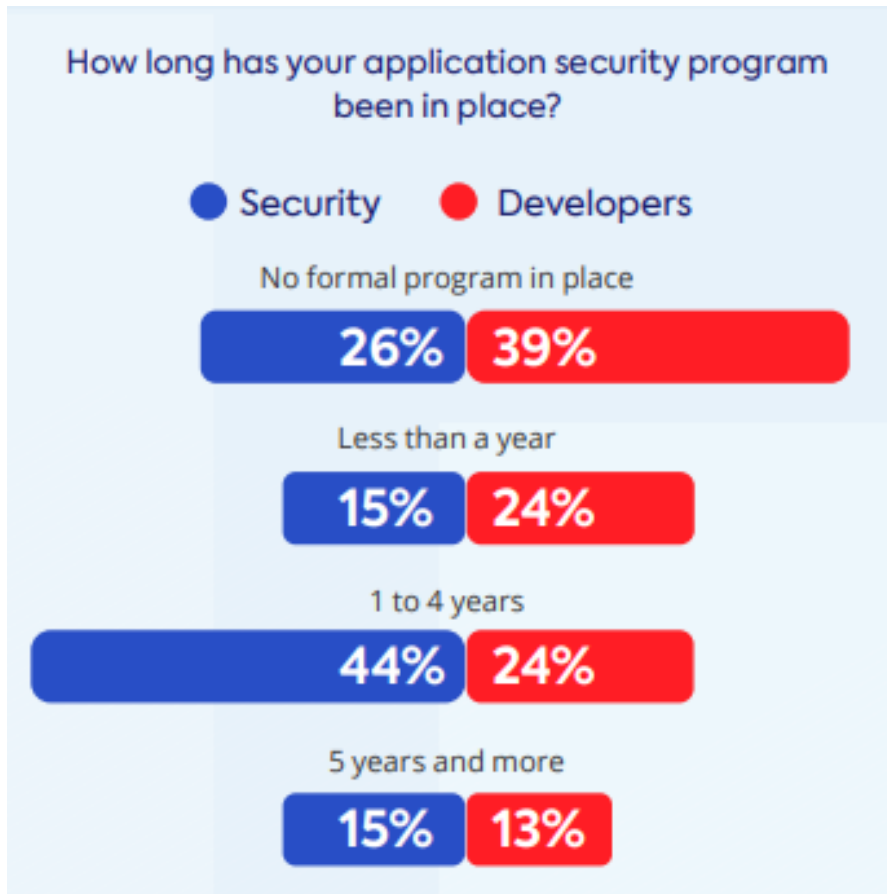
# DO YOU NEED TO TRAIN YOUR DEVS?



Yikes!



# DEVELOPER KNOWLEDGE GAP



# IS EVERYONE ON THE SAME PAGE?

To what extent do the security team and development team in my organization agree on which application vulnerabilities need to be fixed?

We have an agreed-upon process to determine priorities

31%

We sometimes agree, but we follow ad hoc practices and separate guidelines

58%

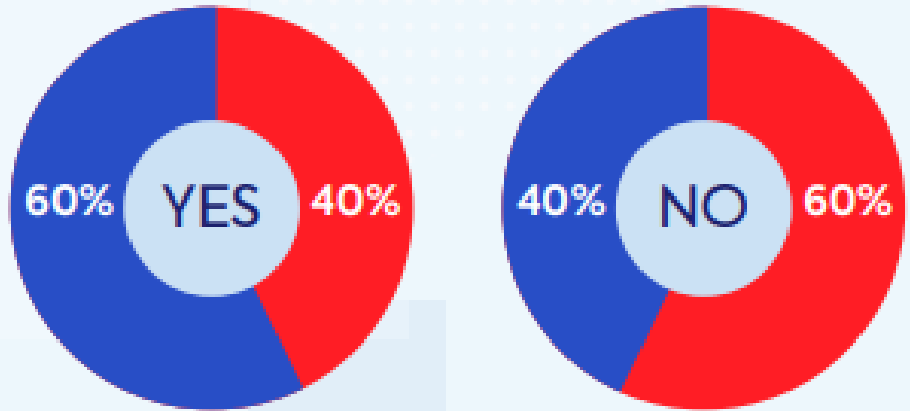
We rarely agree

11%

# DEVSECOPS CHAMPION

Do you have an assigned champion in your development team who takes the lead on AppSec objectives?

● Security ● Developers



To what extent do the security team and development team in my organization agree on which application vulnerabilities need to be fixed?

● We have an agreed-upon process to determine priorities ● We sometimes agree, but we follow ad hoc practices and separate guidelines ● We rarely agree



TRAINING

APPSEC PROGRAM

PRIORITIZATION

AUTOMATION

SHARED STANDARDS

SHIFT LEFT

APPSEC CHAMPION

COMMUNICATION

# ACHIEVING DEVSECOPS MATURITY





# CALL TO ACTION!



Bolt

**Free!** – Great for small teams & limited scans

<https://bolt.whitesourcesoftware.com/>



Renovate

**Free!** – Powerful Dependency updates

<https://renovate.whitesourcesoftware.com/>




Core

**Free Trial** – Flagship WhiteSource Solution

<https://www.whitesourcesoftware.com/free-trial/>



# BOLT (Azure DevOps)



## WhiteSource Bolt

WhiteSource | 18,827 installs | ★★★★★ (18) | Free

Detect & fix security vulnerabilities, problematic open source licenses.

[Get it free](#)

[Overview](#) | [Q & A](#) | [Rating & Review](#)

We help you harness the power of open source without compromising on security or agility!

WhiteSource Bolt for Azure DevOps is a FREE extension, which scans all your projects and detects open source components, their license and known vulnerabilities. Not to mention, we also provide fixes.

We've got you covered with support for most common programming languages and continuous tracking of multiple open source vulnerabilities databases like the NVD, security advisories, peer-reviewed vulnerability databases, and popular open source projects issue trackers.



# BOLT (GITHUB)



Application

## WhiteSource Bolt

[Set up a plan](#)

Verified by GitHub  
GitHub confirms that this app meets the requirements for verification.

Categories

Continuous integration

Security | Free

Supported languages

C, C#, C++  
and 7 other languages supported

Developer

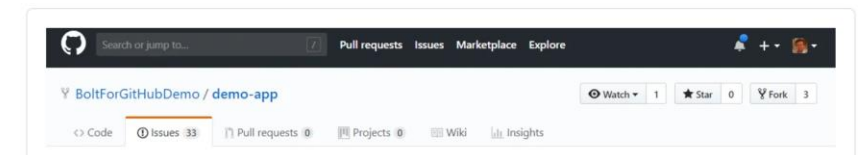


We'll help you harness the power of open source without compromising on security or agility!

WhiteSource Bolt for GitHub is a FREE app, which continuously scans all your repos, detects vulnerabilities in open source components and provides fixes. It supports both private and public repositories.

We've got you covered with over 200 programming languages support and continuous tracking of multiple open source vulnerabilities databases like the NVD and additional security advisories.

[Read more...](#)





✓ Verified by GitHub  
GitHub confirms that this app meets the requirements for verification.

#### Categories

Dependency management

Security **GitHub Enterprise**

Free

#### Supported languages

Dockerfile, Go, Gradle  
and 7 other languages supported

#### Customers



#### Developer

Application

# Renovate

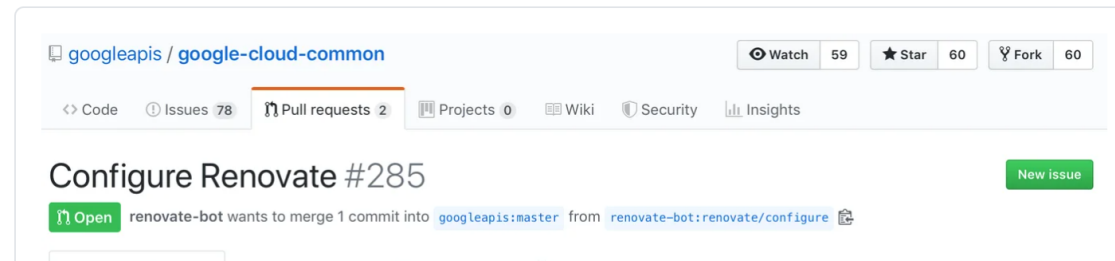
Set up a plan

## Multi-language Dependency Automation

- Automatically update dependencies using convenient Pull Requests
- Supports a multitude of languages including JavaScript, Java, Ruby, PHP, Python, Go, Cargo, Elixir, Docker, etc.
- Extensive configurability. WhiteSource Renovate will fit in with your workflow, including custom grouping and schedules
- Supports shared presets as code, similar to eslint shared configs

[View WhiteSource Renovate website](#)

[Read more...](#)



# Renovate (Github)



# CORE FREE TRIAL?

<https://www.whitesourcesoftware.com/free-trial/>

OR

For more info please contact me:  
[john.timberlake@whitesourcesoftware.com](mailto:john.timberlake@whitesourcesoftware.com)

**WhiteSource | Cooking Corp.**

Home | Dashboards | Organizations | Products | Projects | Policies | Reports | Integrate | Maya R | Contact Support | Admin

### Top Alerts (277)

Library	Type	Description	Occurrences	Library Type	Modified Date
jackson-databind-2.9.2.jar	Security Vulnerability	High: 12 Medium: 1 details	2 projects details	Java	17-04-2019 ignore
Django-1.8.tar.gz	Security Vulnerability	High: 6 Medium: 9 Low: 1 details	1 project details	Python	11-04-2019 ignore
commons-collections-3.2.jar	Security Vulnerability	High: 4 details	6 projects details	Java	20-03-2019 ignore
commons-collections-3.0.jar	Security Vulnerability	High: 4 details	3 projects details	Java	20-03-2019 ignore
mysql-connector-java-5.1.18.jar	Policy Violation	Reject GPLs	5 projects details	Java	06-02-2019 ignore
commons-beanutils-1.8.3.jar	Security Vulnerability	High: 1 details	7 projects details	Java	06-02-2019 ignore
commons-fileupload-1.2.2.jar	Security Vulnerability	High: 4 Low: 1 details	5 projects details	Java	06-02-2019 ignore
awork-core-2.3.31.jar	Security Vulnerability	High: 2 details	5 projects details	Java	06-02-2019 ignore
spring-web-3.1.1.RELEASE.jar	Security Vulnerability	High: 1 (0) Medium: 4 (0) details	5 projects details	Java	06-02-2019 ignore
spring-core-3.1.1.RELEASE.jar	Security Vulnerability	Medium: 1 (1) details	4 projects details	Java	06-02-2019 ignore

**Top 10 Products (23)**

Product	Projects	Libraries	Vulnerable Libraries	Licenses
HR v3.0	4	300	High: 13 Medium: 14	28
ERP-1.0_Repo	1	318	High: 10 Medium: 13	25
ERP-1.0_Build	1	318	High: 10 Medium: 13	25
ERP-1.0_Prod	1	318	High: 10 Medium: 13	25
CRM_Prod	4	81	High: 6 Medium: 1	19
Test	1	54	High: 11 Medium: 4 Low: 1	13
EUA	1	39	High: 8 Medium: 1	9
EUA_2019	2	36	High: 2 Medium: 3	8
My Product	5	24	High: 5 Medium: 4	11
NewProj				

### Library Vulnerability

Reported Library Vulnerability (HIGH) | Effective Library Vulnerability

Library Statistics: 337 Outdated, 53 Outdated & Vulnerable, 100 Vulnerable

Analysis Statistics: 14% Analysis Coverage, 286 / 286 Effective or Non-Analyzed, 0 / 286 Non-Effective

### License Distribution

Total License Types: 28

- GPL 3.0
- Eclipse 1.0
- GPL 2.0 Classpath
- GPL 2.0
- LGPL 3.0
- CDDL 1.0
- LGPL 2.1
- Apache 2.0
- MIT
- MIT
- BSD 3
- Public Domain
- BSD 2
- Common Public 1.0
- Mozilla 2.0
- LGPL
- BSD
- CDL 1.1
- Others

# WE ARE HIRING!

<https://www.whitesourcesoftware.com/careers/>



## Current Open Positions (10)

Job Title

Algorithms Developer

Customer success manager

Demand Generation Manager (ABM/Intent)

Development Team Leader-Enterprise Team

Global Communications Marketing Manager

IT Help Desk Specialist

QA Team Leader

Regional Sales Engineer

Sales Development Representative

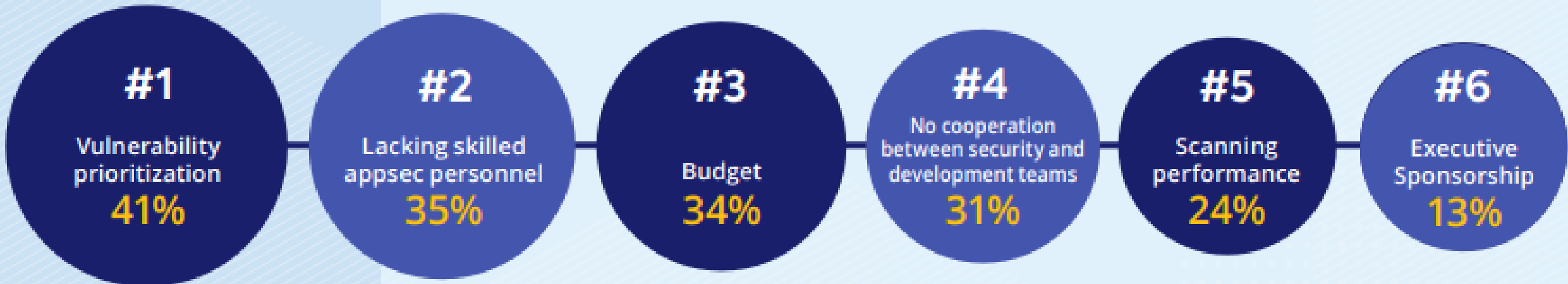
Senior Product Manager

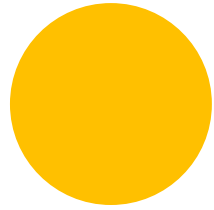
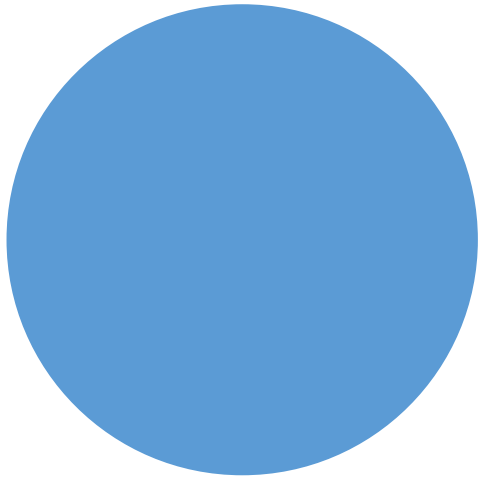
# More Sources

- **Diving Into the Evil internet - Vulnerability Prioritization Through the Eyes of Hackers**
  - <https://resources.whitesourcesoftware.com/wistia-webinars/diving-into-the-evil-internet-vulnerability-prioritization-through-the-eyes-of-hackers>
- **2020 WhiteSource DevSecOps Insights**
  - <https://www.whitesourcesoftware.com/whitesource-devsecops-insights/>
- **2020 WhiteSource Annual Report**
  - [https://www.whitesourcesoftware.com/wp-content/media/2020/03/Annual\\_Report\\_2020\\_12.03.20.pdf](https://www.whitesourcesoftware.com/wp-content/media/2020/03/Annual_Report_2020_12.03.20.pdf)
- **2020 WhiteSource April Report**
  - [https://resources.whitesourcesoftware.com/blog-whitesource/april-open-source-security-vulnerabilities-snapshot?utm\\_origin=social&utm\\_from=linkedin&utm\\_campaign=Blog](https://resources.whitesourcesoftware.com/blog-whitesource/april-open-source-security-vulnerabilities-snapshot?utm_origin=social&utm_from=linkedin&utm_campaign=Blog)
- **2018 WhiteSource Annual Report**
  - <https://www.whitesourcesoftware.com/wp-content/uploads/2018/10/The-State-of-Open-Source-Vulnerabilities-Management-2018.pdf>

# Survey Results:

What are the biggest challenges in implementing and running your AppSec program?



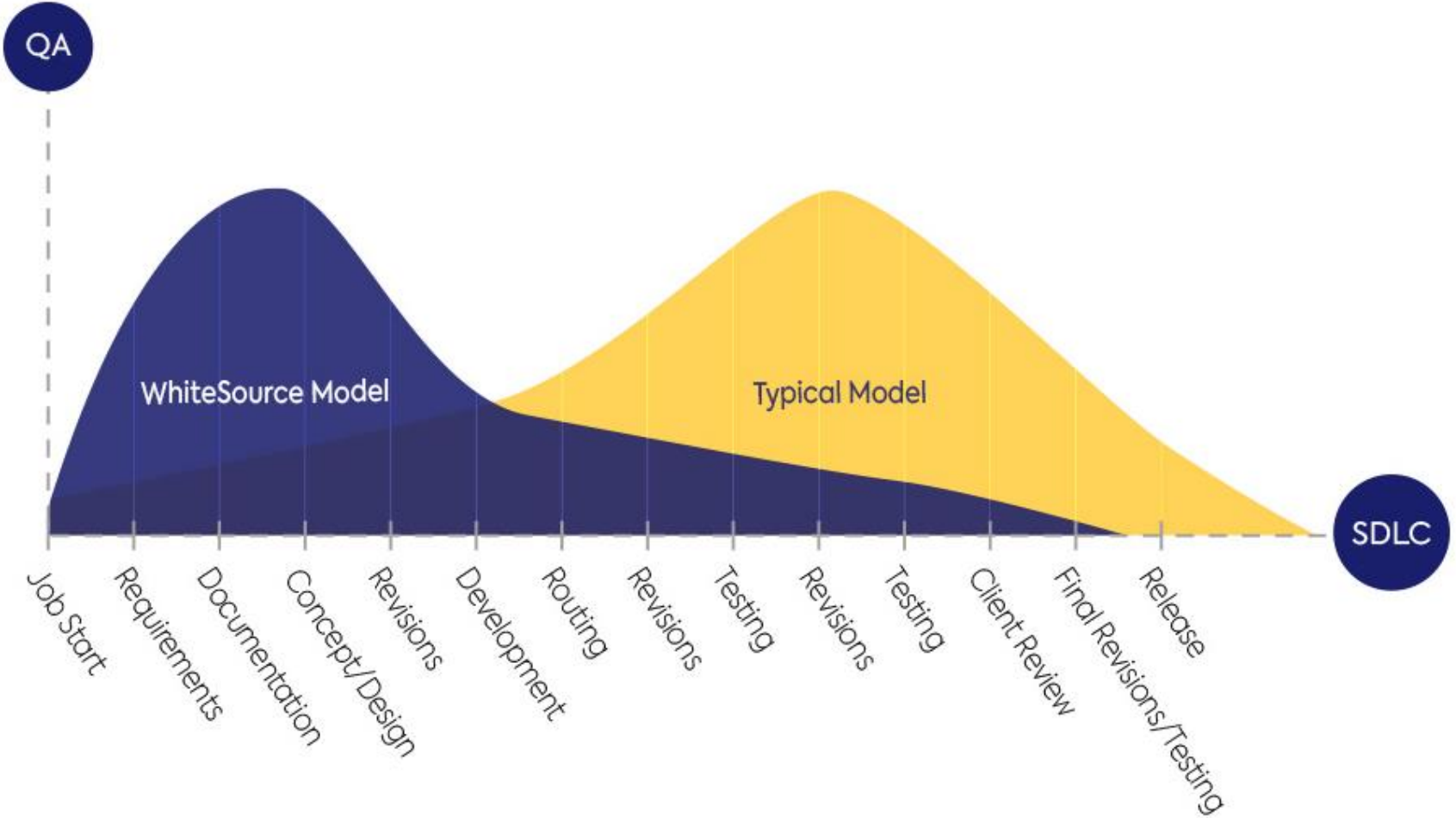


# DEMO

Dima Gorbonos  
Sr. Solution Architect

John Timberlake  
Director of Sales PNW

# Traditional vs. Developer focused SCA Model





# THANK YOU!

For more info please contact us: [john.timberlake@whitesourcesoftware.com](mailto:john.timberlake@whitesourcesoftware.com)



[WhiteSourceSoftware.com](http://WhiteSourceSoftware.com)

