Secure Dev Ops
Testing in the Continuous Delivery Pipeline

About me

Donat Hauser
Principal Consultant

donat.hauser@zuehlke.com

linkedin.com/in/dhauser/

Recent Cyber Attacks

12.02.2019 11:03 Uhr | Security

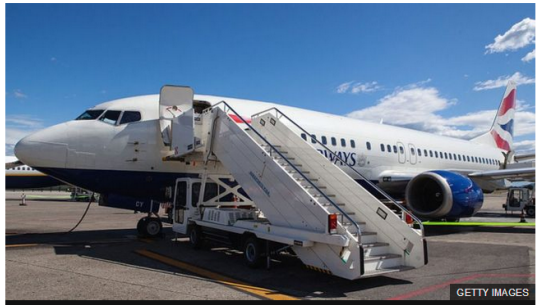
Gehackte Websites: 620 Millionen Accounts zum Verkauf im Darknet

Eine riesige Datenbank mit Mail-Adressen und Passwörtern steht zum Verkauf. Bei einigen Websites war bislang nicht bekannt, dass sie gehackt wurden.

Von Dennis Schirrmacher 127

British Airways breach: How did hackers get in?

7 September 2018 Share



It isn't clear how hackers boarded BA's website and app - but cyber-security experts have some suggestions

British Airways has revealed that hackers managed to breach its website and app, stealing data from many thousands of customers in the process.

14.09.2017 13:29 Uhr | Security

Equifax-Hack: Angreifer über Apache-Struts-Lücke eingestiegen

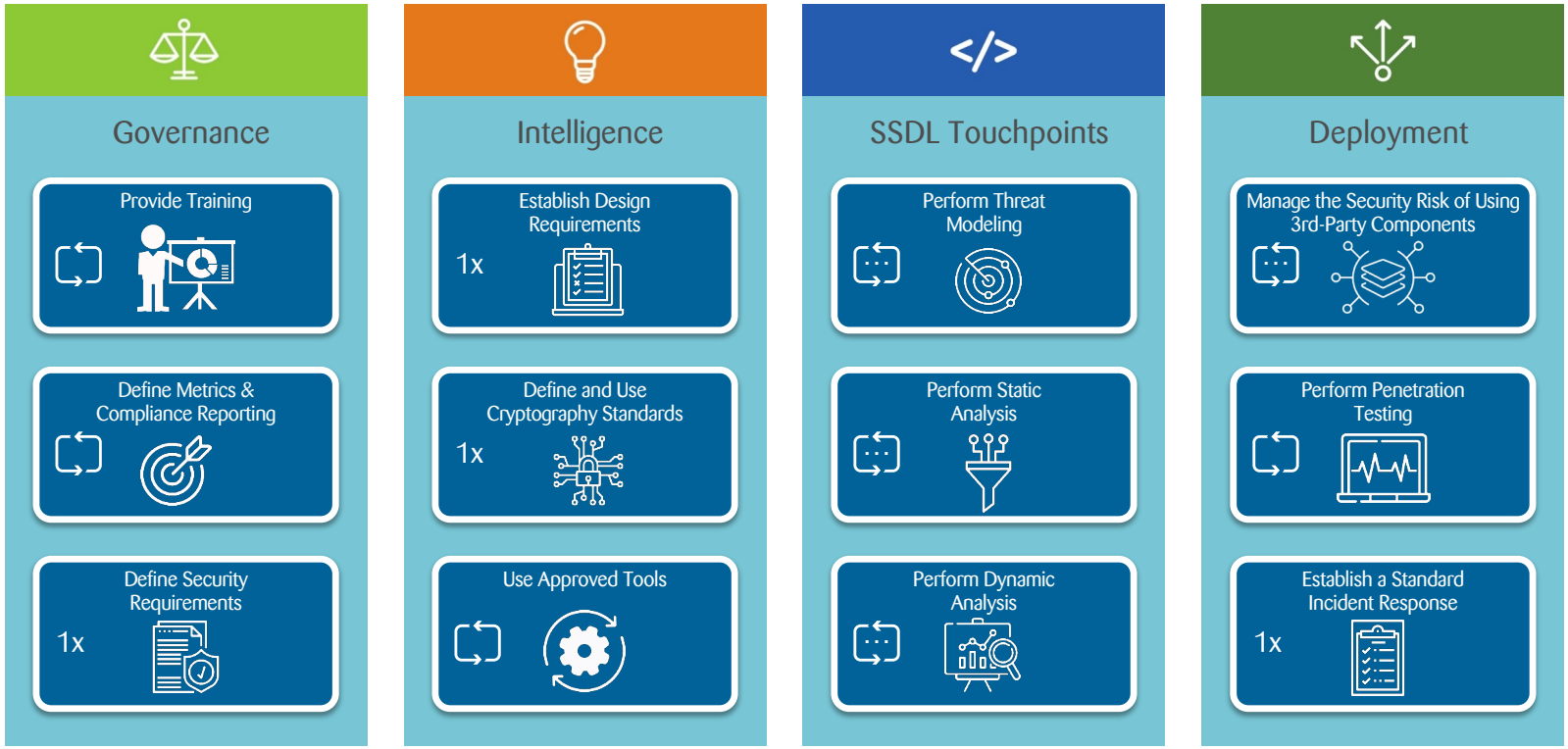
Untersuchungen zeigen, dass Equifax es offensichtlich versäumt hat, Sicherheitsupdates für eine kritische Lücke zu installieren. Darüber hinaus ist es zu einem weiteren Datenleck gekommen.

Von Dennis Schirrmacher 34

"75% of security breaches happen at the application level"
Gartner

92% of reported vulnerabilities are in applications not in networks.
NIST

Microsoft SDL Practices

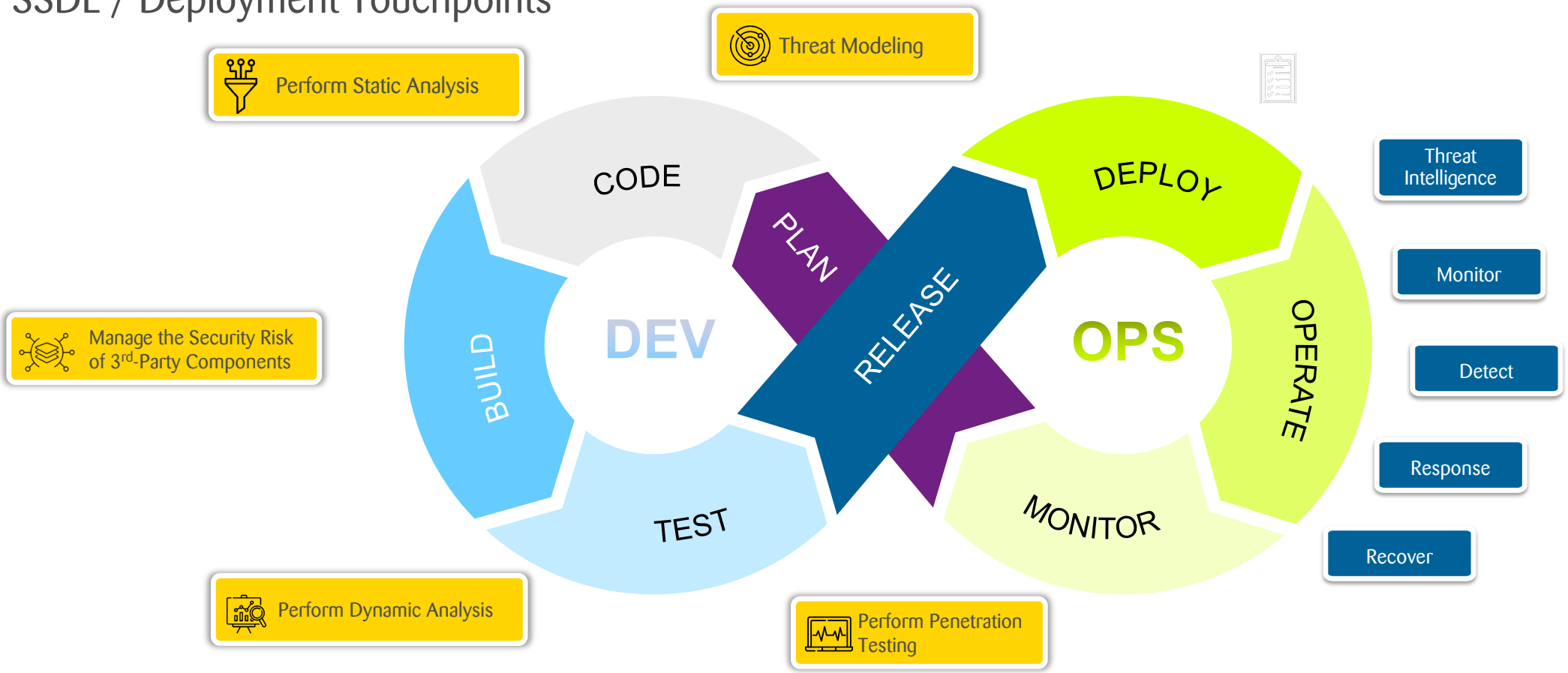


Source: https://www.microsoft.com/en-us/securityengineering/sdl/practices

1x one-time regular every sprint

Secure Dev Ops

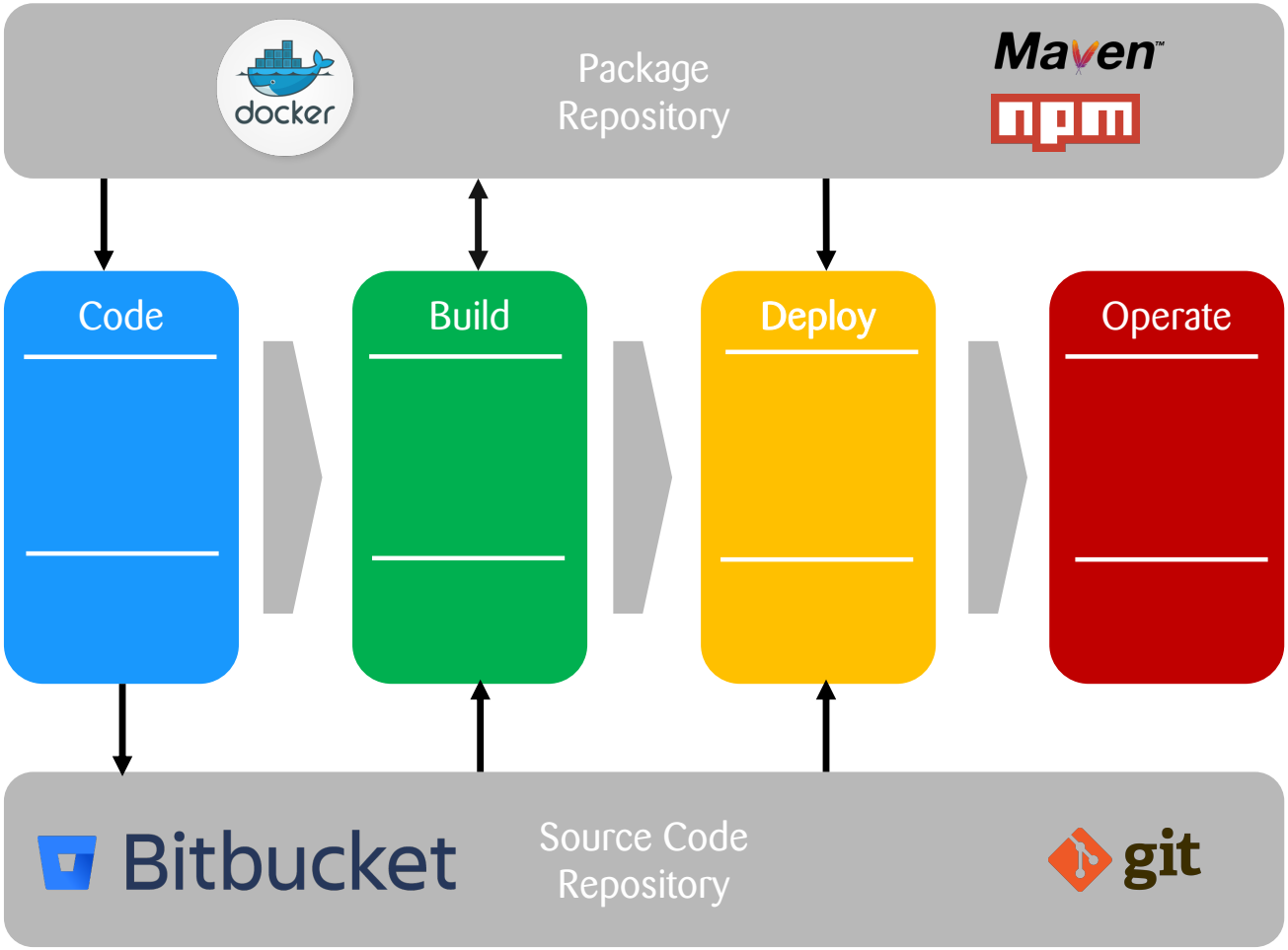
SSDL / Deployment Touchpoints

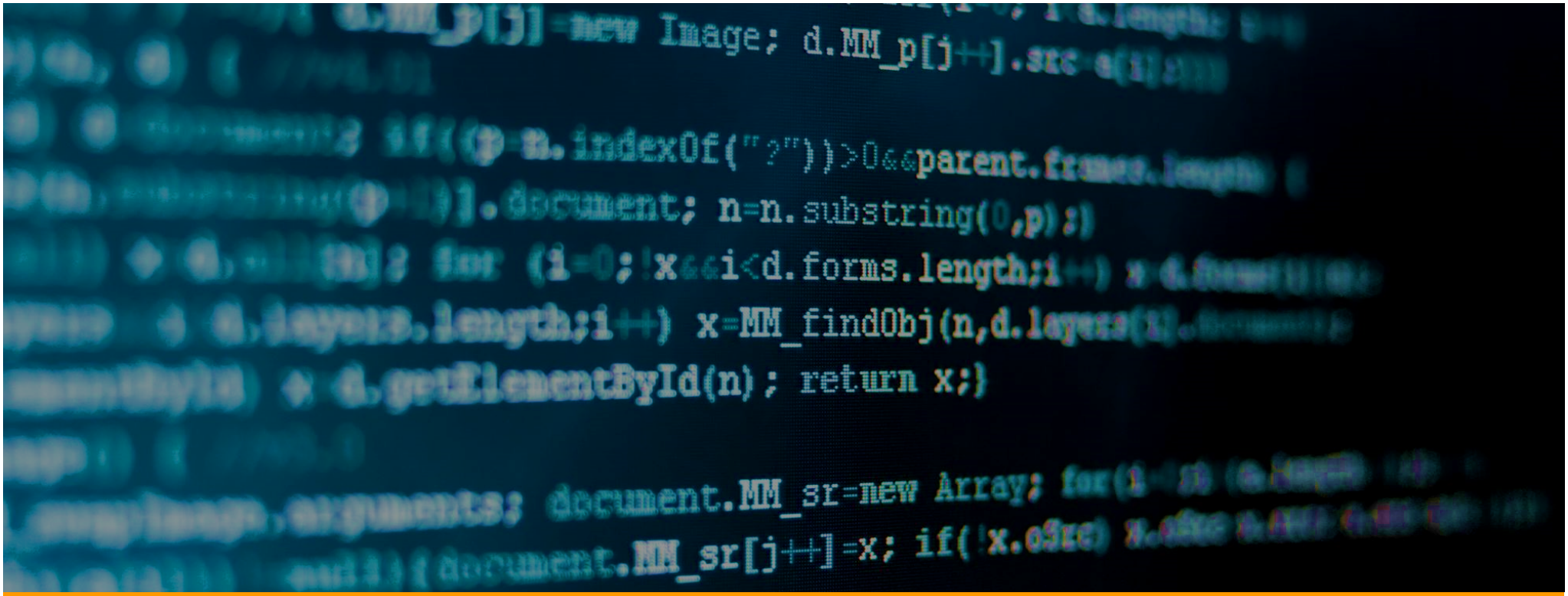


Delivery Pipeline

Success factors

- Automate
- Shift left
- Developer acceptance

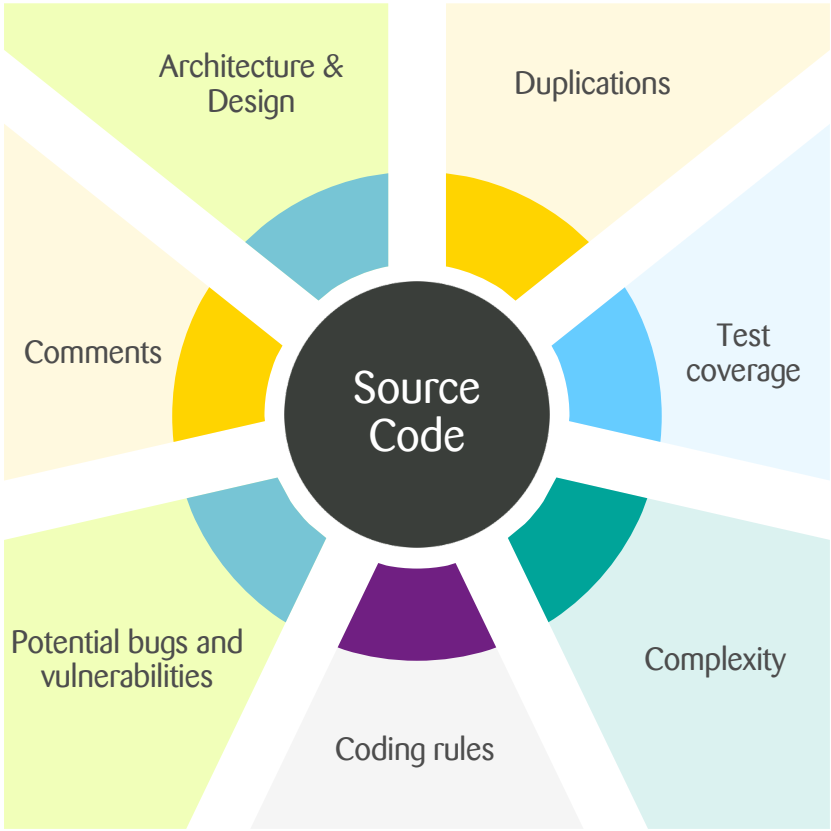




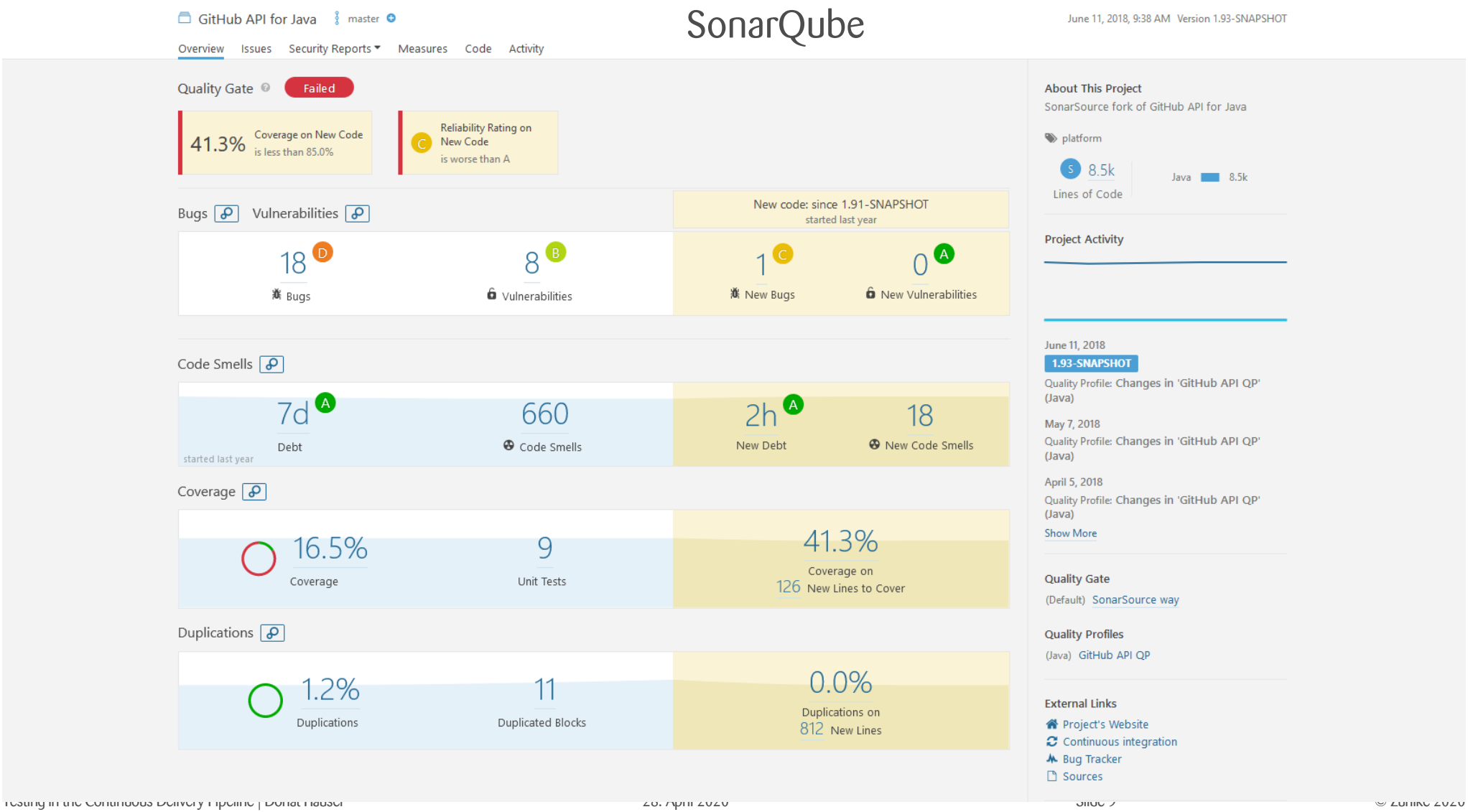
Perform Static Analysis

Static Application Security Testing (SAST)

- Continuous inspection of code quality
- Quality gates based on metrics
- Tracking of code quality over time
- Many supported Languages
- Integration with:
  - Build systems: Maven, Gradle, MSBuild, NPM, etc.
  - CI Servers: Bamboo, Jenkins, etc.
  - IDE: Visual Studio, IntelliJ, Eclipse, etc.



Checkmarx VERACODE MICRO FOCUS sonarqube


GitHub API for Java
master
SonarQube
June 11, 2018, 9:38 AM Version 1.93-SNAPSHOT
Overview
Issues
Security Reports
Measures
Code
Activity
Quality Gate
Failed
41.3% Coverage on New Code
is less than 85.0%
C Reliability Rating on New Code
is worse than A
Bugs
Vulnerabilities
New code: since 1.91-SNAPSHOT
started last year
18 D
Bugs
8 B
Vulnerabilities
1 C
New Bugs
0 A
New Vulnerabilities
Code Smells
7d A
Debt
started last year
660
Code Smells
2h A
New Debt
18
New Code Smells
Coverage
16.5%
Coverage
9
Unit Tests
41.3%
Coverage on
126 New Lines to Cover
Duplications
1.2%
Duplications
11
Duplicated Blocks
0.0%
Duplications on
812 New Lines
About This Project
SonarSource fork of GitHub API for Java
platform
S 8.5k
Lines of Code
Java 8.5k
Project Activity
June 11, 2018
1.93-SNAPSHOT
Quality Profile: Changes in 'GitHub API QP' (Java)
May 7, 2018
Quality Profile: Changes in 'GitHub API QP' (Java)
April 5, 2018
Quality Profile: Changes in 'GitHub API QP' (Java)
Show More
Quality Gate
(Default) SonarSource way
Quality Profiles
(Java) GitHub API QP
External Links
Project's Website
Continuous integration
Bug Tracker
Sources

Static Code Analysis Tools

Effectiveness of static code analysis for finding security vulnerabilities

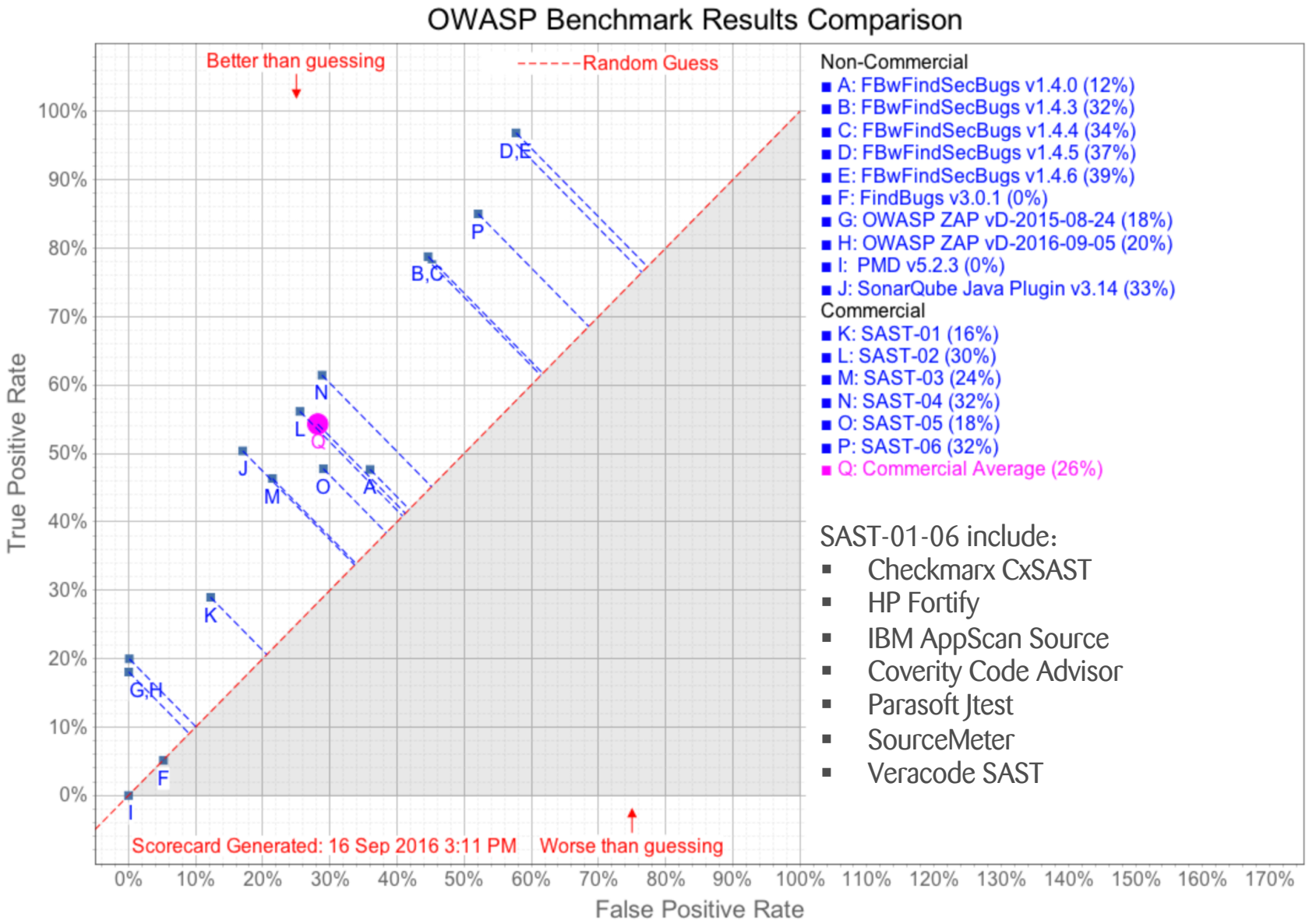
OWASP Benchmark Project

- Goal: verify the speed and accuracy of vulnerability detection tools
- Developed initially for SAST

Background

- Commercial static code analysis tools are expensive
- There is some evidence that suggest that these tools are not always effective:

  https://owasp.org/www-project-benchmark/

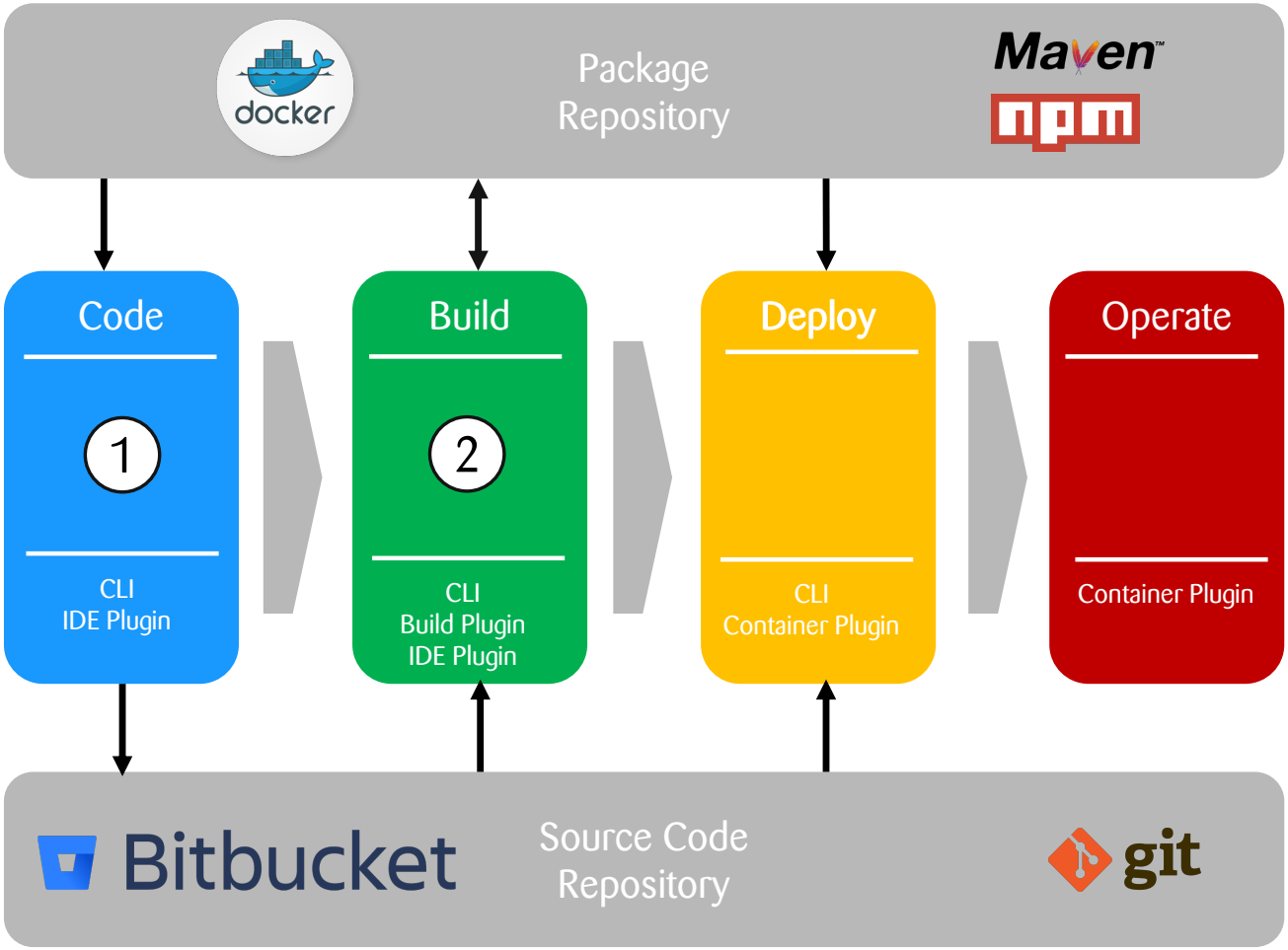

OWASP Benchmark Results Comparison
Better than guessing
Random Guess
Worse than guessing
Scorecard Generated: 16 Sep 2016 3:11 PM
True Positive Rate
False Positive Rate
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
110%
120%
130%
140%
150%
160%
170%
Non-Commercial
A: FBwFindSecBugs v1.4.0 (12%)
B: FBwFindSecBugs v1.4.3 (32%)
C: FBwFindSecBugs v1.4.4 (34%)
D: FBwFindSecBugs v1.4.5 (37%)
E: FBwFindSecBugs v1.4.6 (39%)
F: FindBugs v3.0.1 (0%)
G: OWASP ZAP vD-2015-08-24 (18%)
H: OWASP ZAP vD-2016-09-05 (20%)
I: PMD v5.2.3 (0%)
J: SonarQube Java Plugin v3.14 (33%)
Commercial
K: SAST-01 (16%)
L: SAST-02 (30%)
M: SAST-03 (24%)
N: SAST-04 (32%)
O: SAST-05 (18%)
P: SAST-06 (32%)
Q: Commercial Average (26%)
SAST-01-06 include:
Checkmarx CxSAST
HP Fortify
IBM AppScan Source
Coverity Code Advisor
Parasoft Jtest
SourceMeter
Veracode SAST

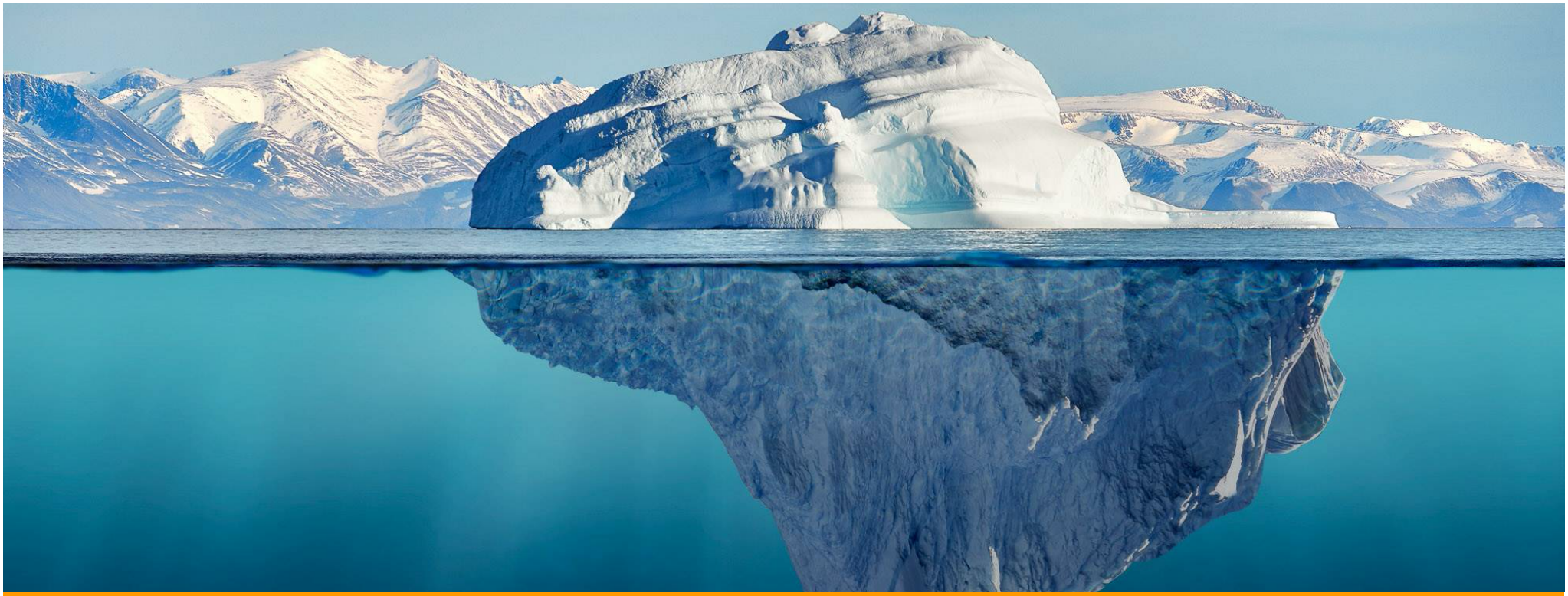
Static Code Analysis Tools

Recommendation:

SonarQube / Sonar Cloud

- Operates at Code and Build level
- Focuses on measuring and tracking quality over time
- Is getting better at detecting security vulnerabilities

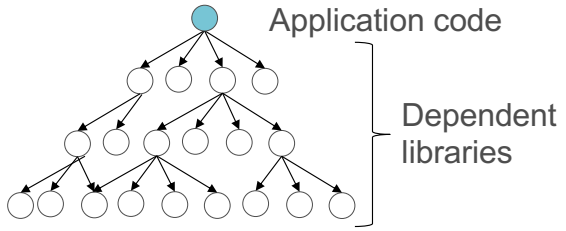




Manage the security risk of 3rd party components

Vulnerabilities of 3rd party components

Application development is increasingly relying on open-source products. That's why it's essential to manage the security of third-party components.



Solution: Software Composition Analysis (SCA)

WhiteHat SECURITY

BLACKDUCK

WhiteSource

JFrog Xray

VERACODE

Twistlock

snyk

NeuVector

Mitigation of OWASP TOP 10 - Risk A9 "Using Components with Known Vulnerabilities"

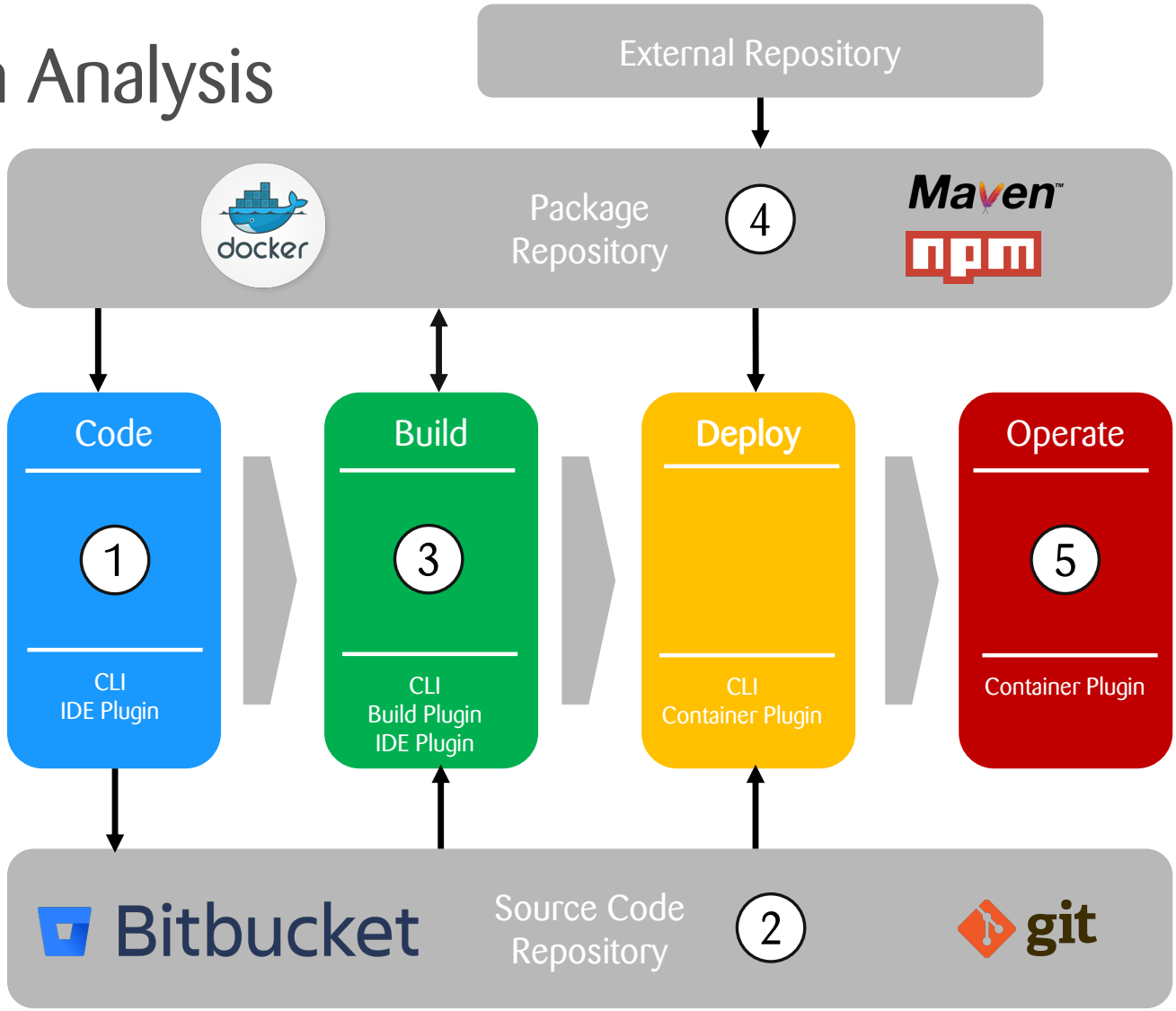
sonatype

Software Composition Analysis

CD Integration

1. Depencency Checker, JFrog Xray, Snyk
2. Whitesource, Snyk
3. Twistlock, JFrog Xray
4. JFrog Xray, Nexus
5. Twistlock, NeuVector

➔ Integration on the Source Code Repository level is the way to go to detect application vulnerabilities



Detecting known vulnerabilities in containers

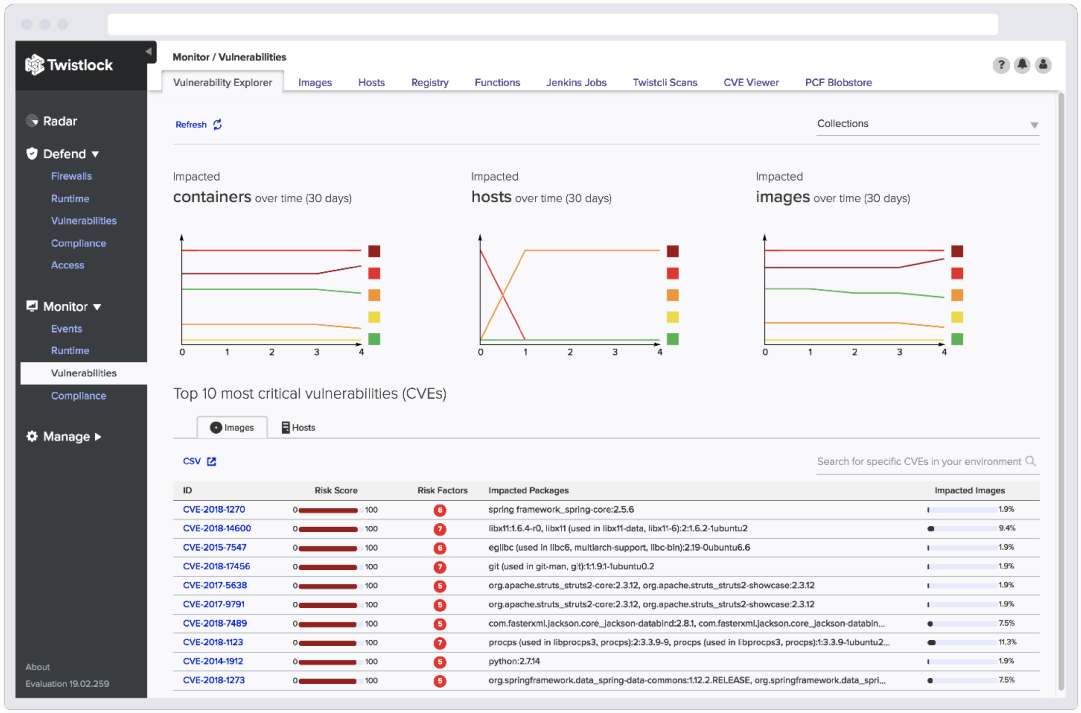
Identifying vulnerabilities in Docker containers requires a new breed of vulnerability management solutions:

Products on the market:

- Twistlock
- NeuVector
- StackRox

Features:

- Deep recursive scanning of dependencies and containers
- Integration into managed container services, such as Kubernetes, Openshift, as well as Cloud platforms.



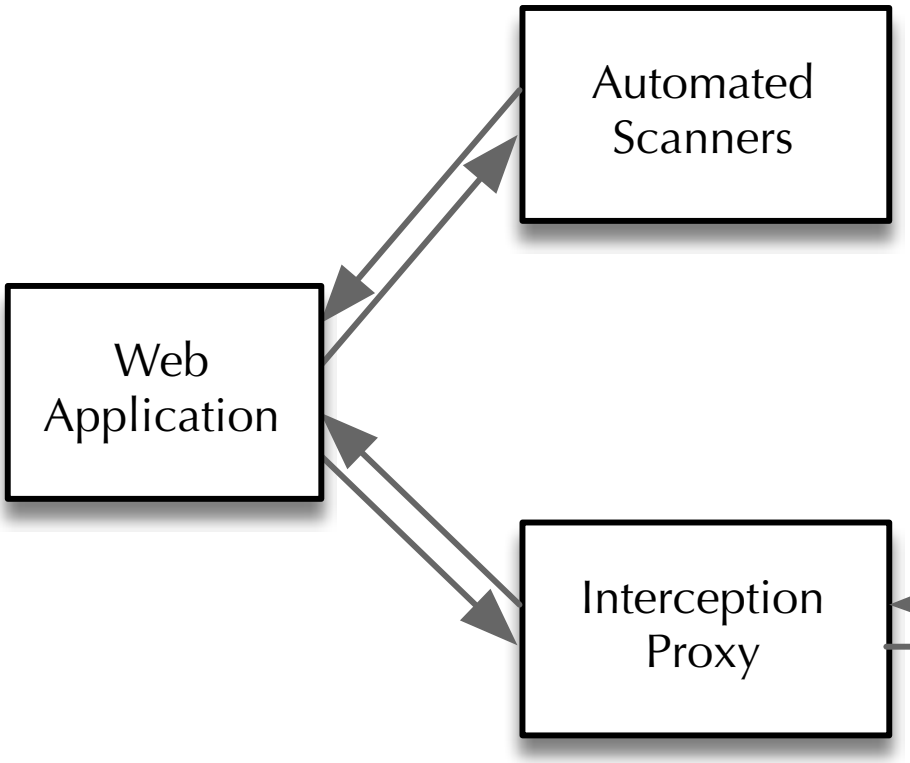
Perform Dynamic Analysis

Dynamic Application Security Testing (DAST)

Approaches

- Vulnerability Scanner
- Fuzzer

Vulnerability Scanners



The Testing Guinea Pig

OWASP Juice Shop an intentionally insecure web application. It contains 38+ hacking challenges of varying difficulty.

HTML 5
Client

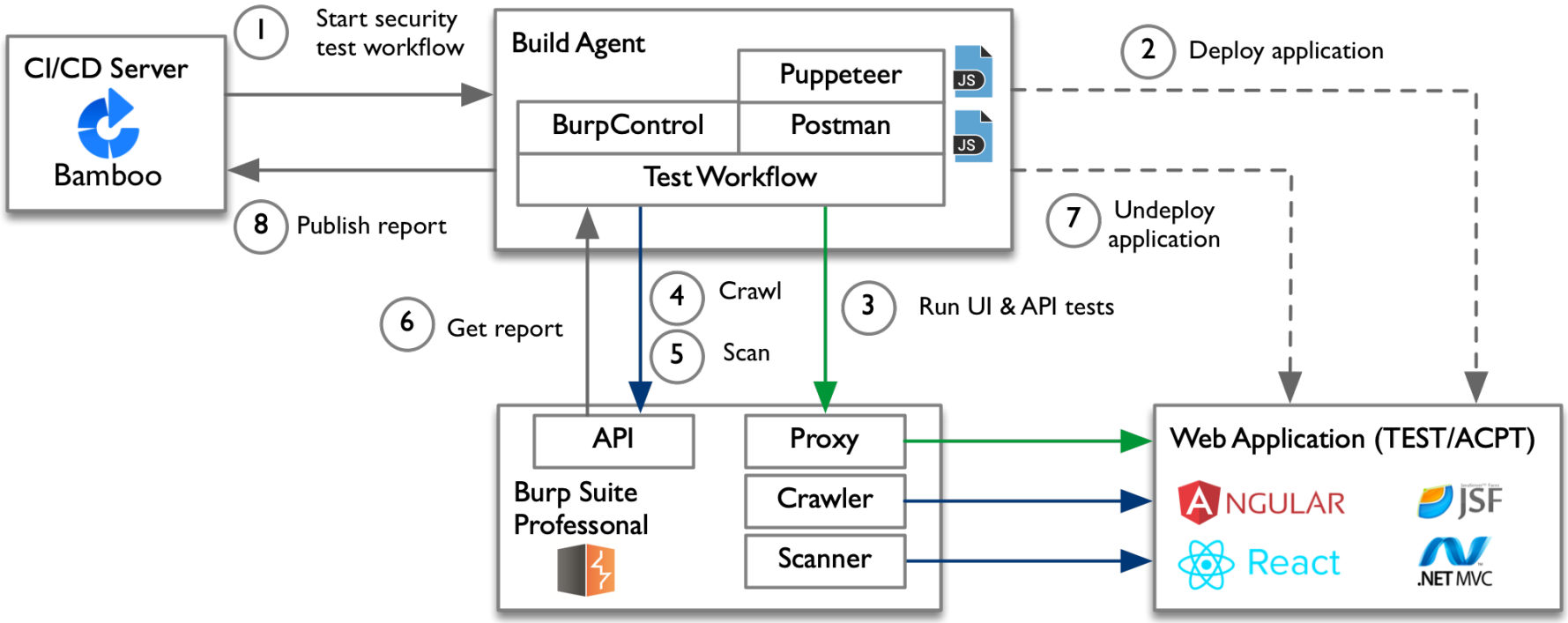
Backend

Database

Vulnerability	Acunetix	Arachni	Burp Suite Pro.	GoLismero	Nessus Pro.	Netsparker Desktop	ZAP
Autocomplete enabled for passwords		●	●			●	●
Content sniffing not disabled			●				●
Cross-site scripting		●	●		●	●	
CSP not implemented					●	●	
Email disclosure			●			●	
HSTS policy not enabled		●	●	●	●	●	
Information leakage via error message	●	●	●			●	
Insecure cookie	●	●	●			●	●
Insecure frame		●	●			●	●
Internal IP disclosure			●			●	●
Misconfigured allow origin header	●	●	●	●		●	
Missing CSRF protection	●	●				●	
Missing X-XSS-protection header				●		●	●
Open redirection (reflected)		●	●				
Outdated frameworks			●			●	
Password transmitted over query string			●			●	
Software versions revealed			●	●	●	●	
SQL injection	●	●	●				

Vulnerability Scanner

Integration into the CI/CD Pipeline

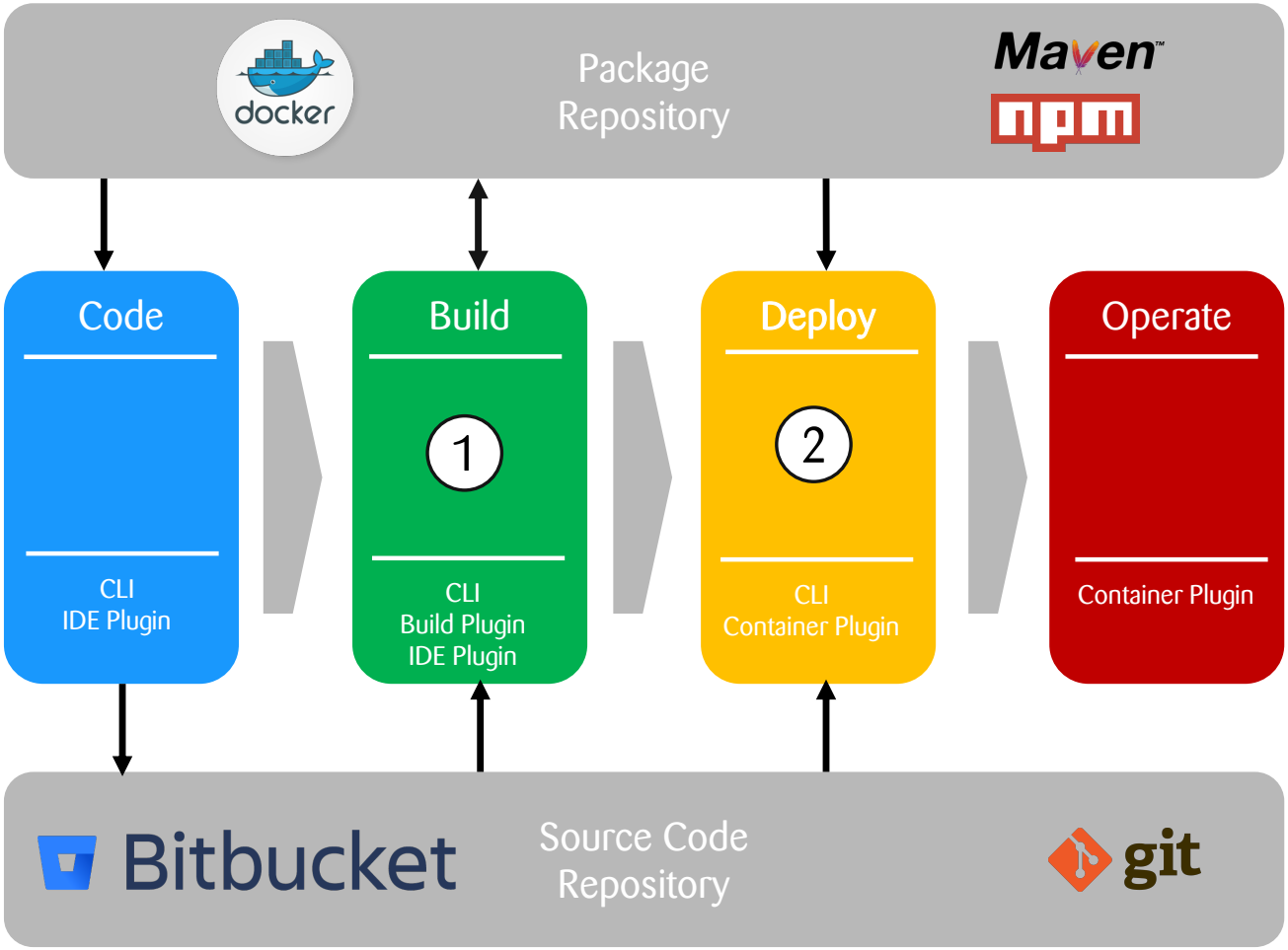


Dynamic Application Security Testing (DAST)

1. Fuzzer
2. Vulnerability Scanner

➔ Vulnerability Scanner should be more Test Case driven to replace penetration testing partially

➔ Fuzzer are currently getting new attention



Threat Modeling

Threat Modeling

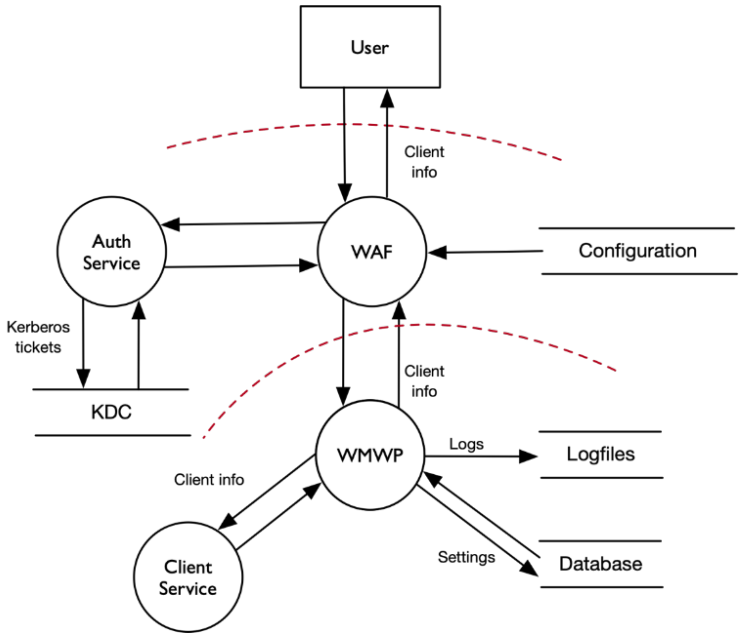
Methodology

Threat Modeling (aka Architectural Risk Analysis) is the activity of identifying and managing application risks.

Threats are identified and evaluated based on the application architecture. Security features are then selected to establish appropriate mitigations.

➔ Difficult to implement and maintain

➔ Little help from tools/automation



Threat Modeling approach by Microsoft based on STRIDE

Threat Modeling – Security Baseline

Alternative approach

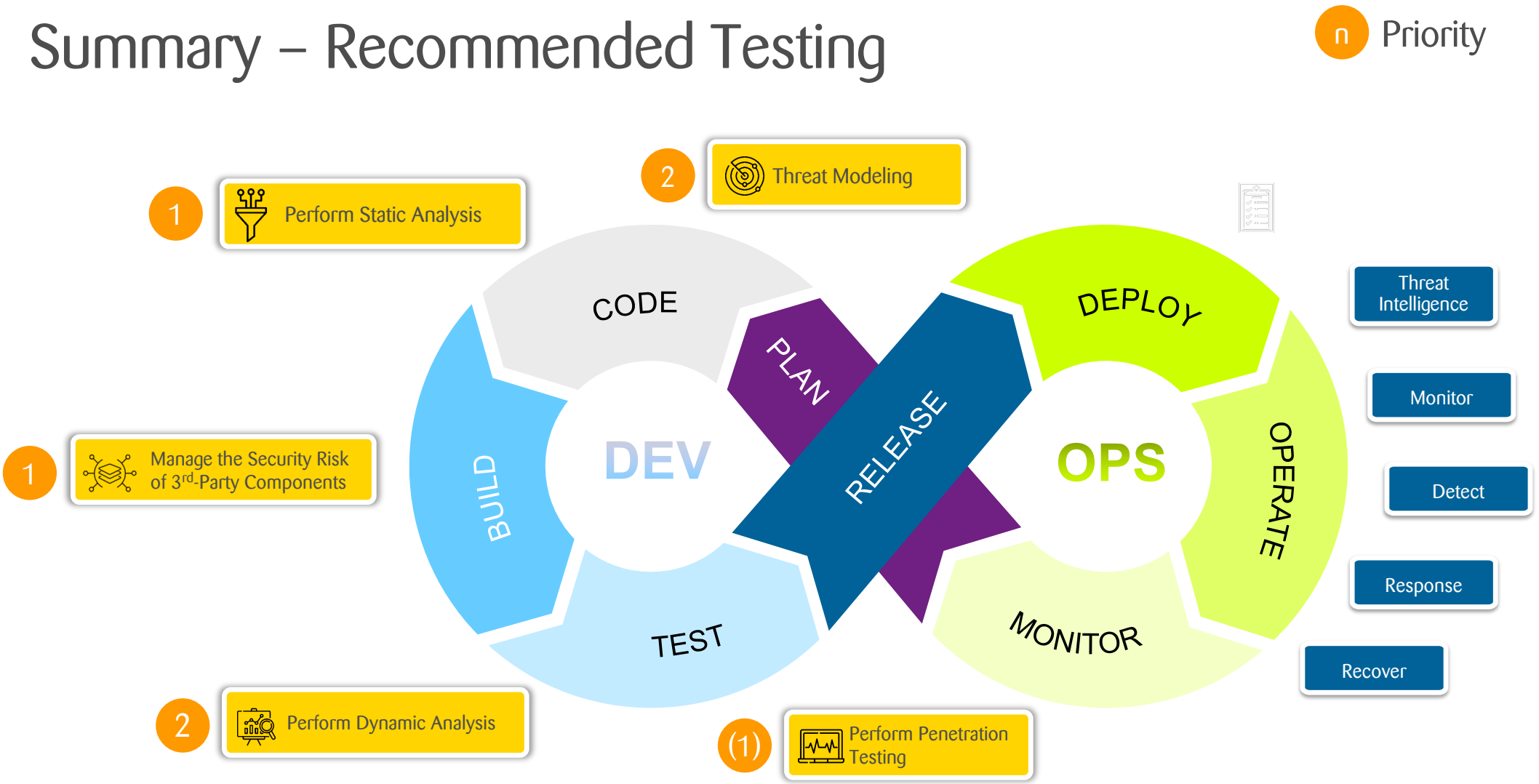
OWASP Application Security Verification Standard 4.0 (ASVS)

- L1: First steps, automated, or whole portfolio view (bare minimum)
- L2: Most applications
- L3: High value, high assurance, or high safety

OWASP Mobile Application Security Verification Standard 1.1 (MASVS)

- L1: Standard Security
- L2: Defense-in-Depth
- R: Resiliency Against Reverse Engineering and Tampering

Summary – Recommended Testing



Donat Hauser
Principal Consultant

donat.hauser@zuehlke.com

linkedin.com/in/dhauser/