Software Composition Analysis

OWASP Stammtisch

19.11.2019

München

Stanislav Sivak

Agenda

- Introduction
- Challenges
- Approaches
- Integration
- Q & A

Agenda

- **Introduction**
- Challenges
- Approaches
- Integration
- Q & A

Introduction

Disclaimer:

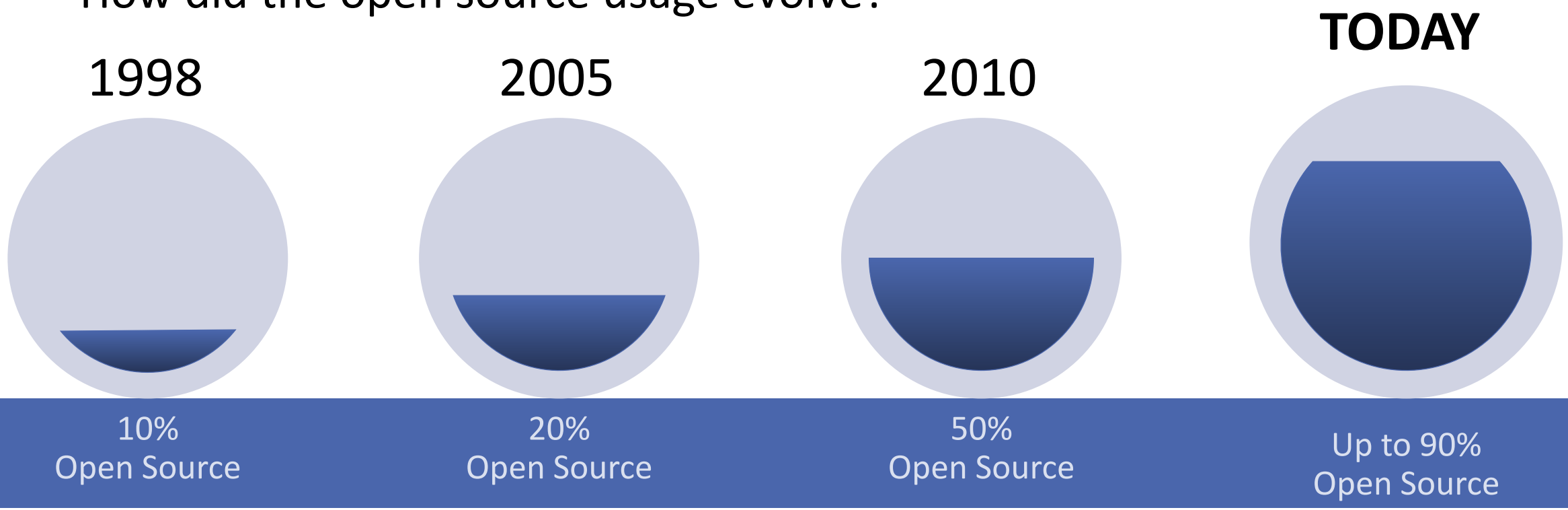
This is my personal presentation and represents neither my current employer nor any other organization.

Introduction

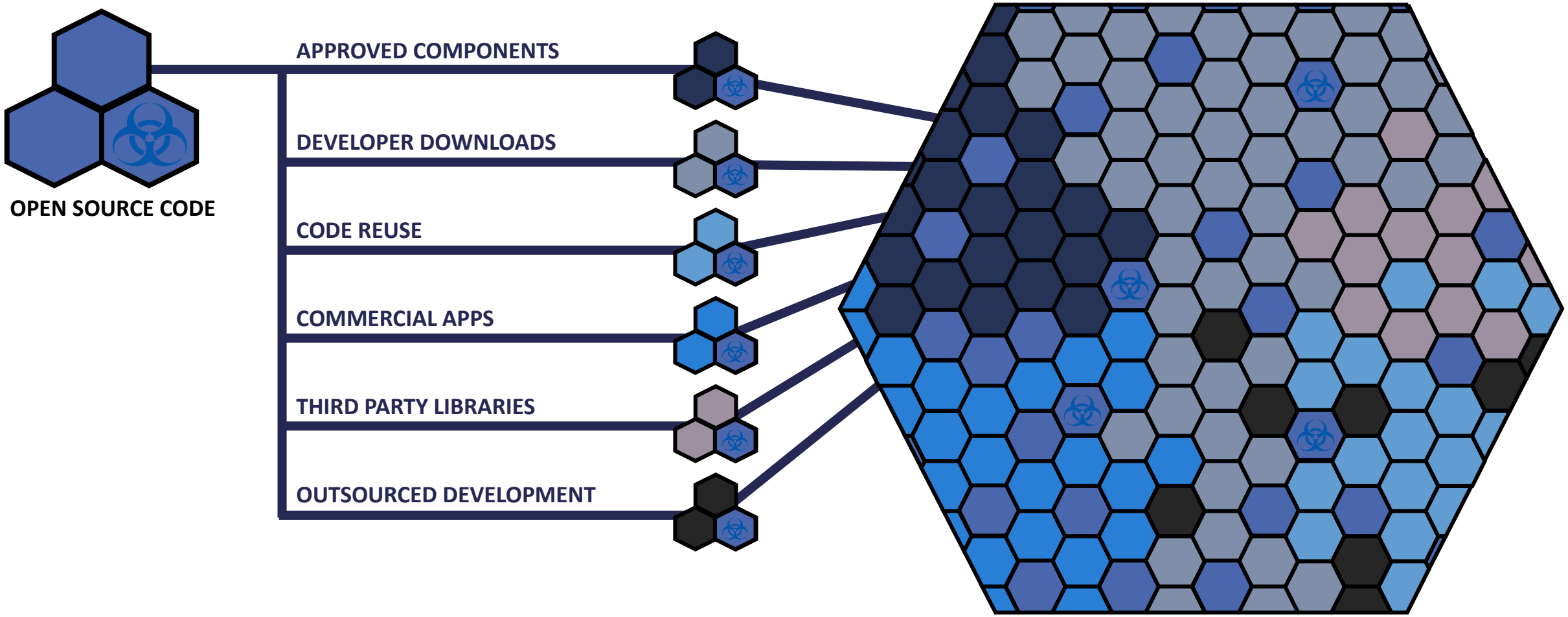
- Senior security consultant at Synopsys
- Working on various AppSec related projects mainly in Germany in the areas of
  - How to secure SDLC with focus on:
    - Threat modelling
    - Application security testing
    - Security in CI/CD
- Previously worked as web developer, security administrator, pentester

Open Source Software

- How did the open source usage evolve?



It enters your code through many channels…



…and open source vulnerabilities can come with it.

State of open source 2018 (1/2)

96%

Black Duck On-Demand audits found open source components in **96%** of the applications scanned, with an average **257** components per application.

134%

The number of open source vulnerabilities per codebase grew by **134%**.

CVE	Percent
CVE-2018-7489	12%
CVE-2017-7525	11%
CVE-2017-15095	11%
CVE-2015-6420	10%
CVE-2014-0050	9%
CVE-2017-15708	9%
CVE-2014-0107	9%
CVE-2016-3092	6%
CVE-2016-8735	5%
CVE-2014-3567	5%

60%

Open source represented 60% of the code analyzed in 2018, up from 57% in 2017

17%

17% of the codebases contained a highly publicized vulnerability such as Heartbleed, Logjam, Freak, Drown, and Poodle.

https://www.synopsys.com/content/dam/synopsys/sig-assets/reports/rep-ossra-19.pdf

Based on over 1,200 commercial applications analyzed by Black Duck On-Demand in 2018

State of open source 2018 (2/2)
Most seen open-source components

jQuery

BootStrap

jQuery UI

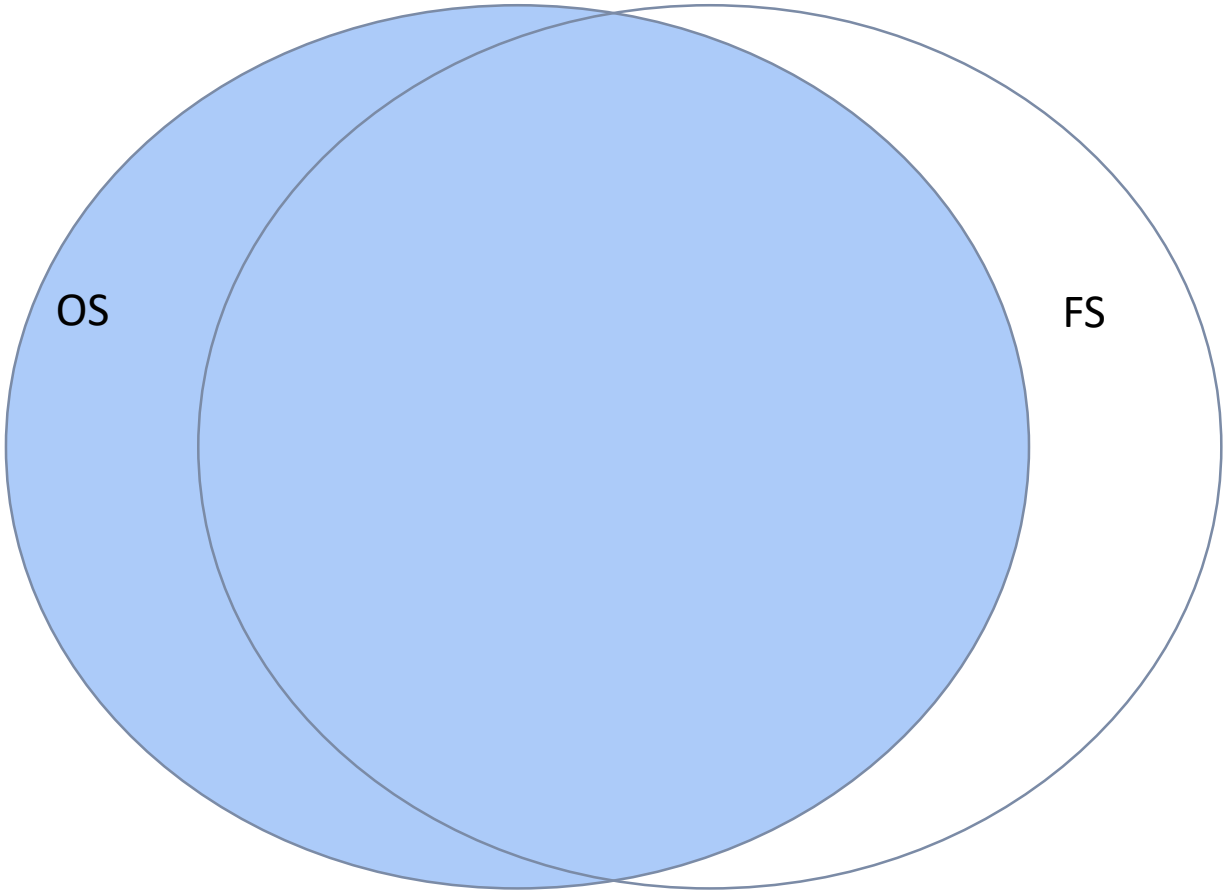
Font Awesome

Moment

What software is in scope?

FOSS – Free and open-source software

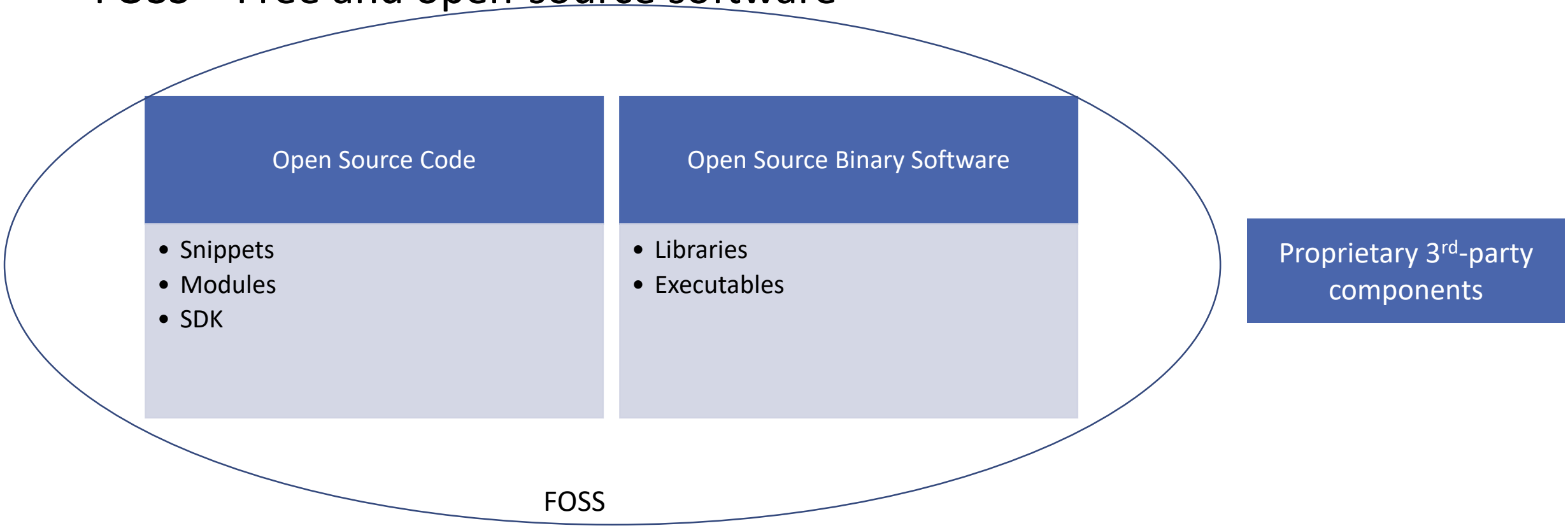
License type approved by Open Source Initiative



License type approved by Free Software Foundation

What software is in scope?

- FOSS – Free and open-source software



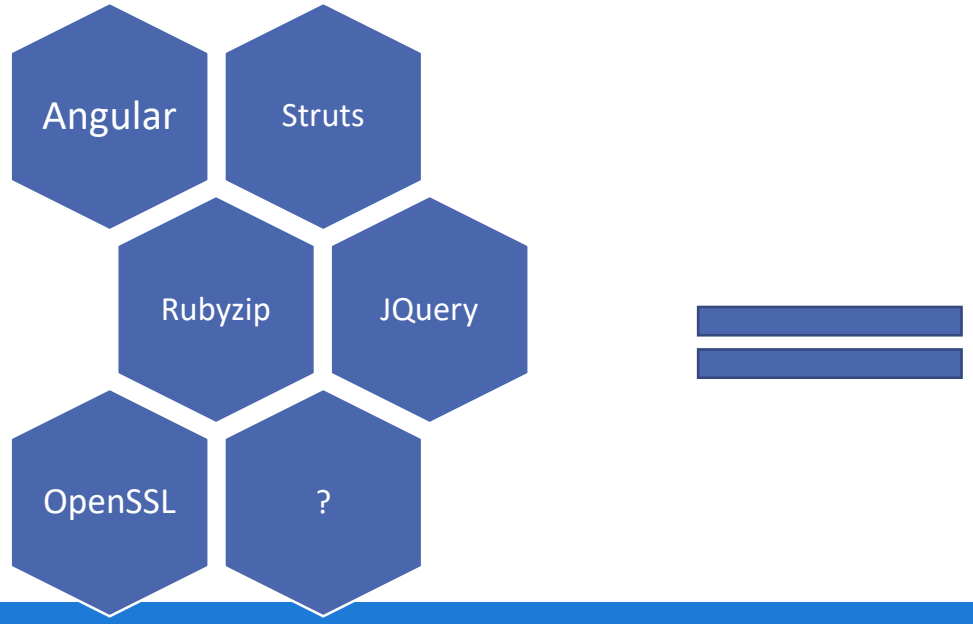
Agenda

- Introduction
- **Challenges (No challenge-> no fun)**
- Approaches
- Integration
- Q & A

Challenge No.1 - Assets

- *I need further information to our application inventory…*
- *How much open source do we use?*
- *How is the use of open source governed in our company?*

CIO



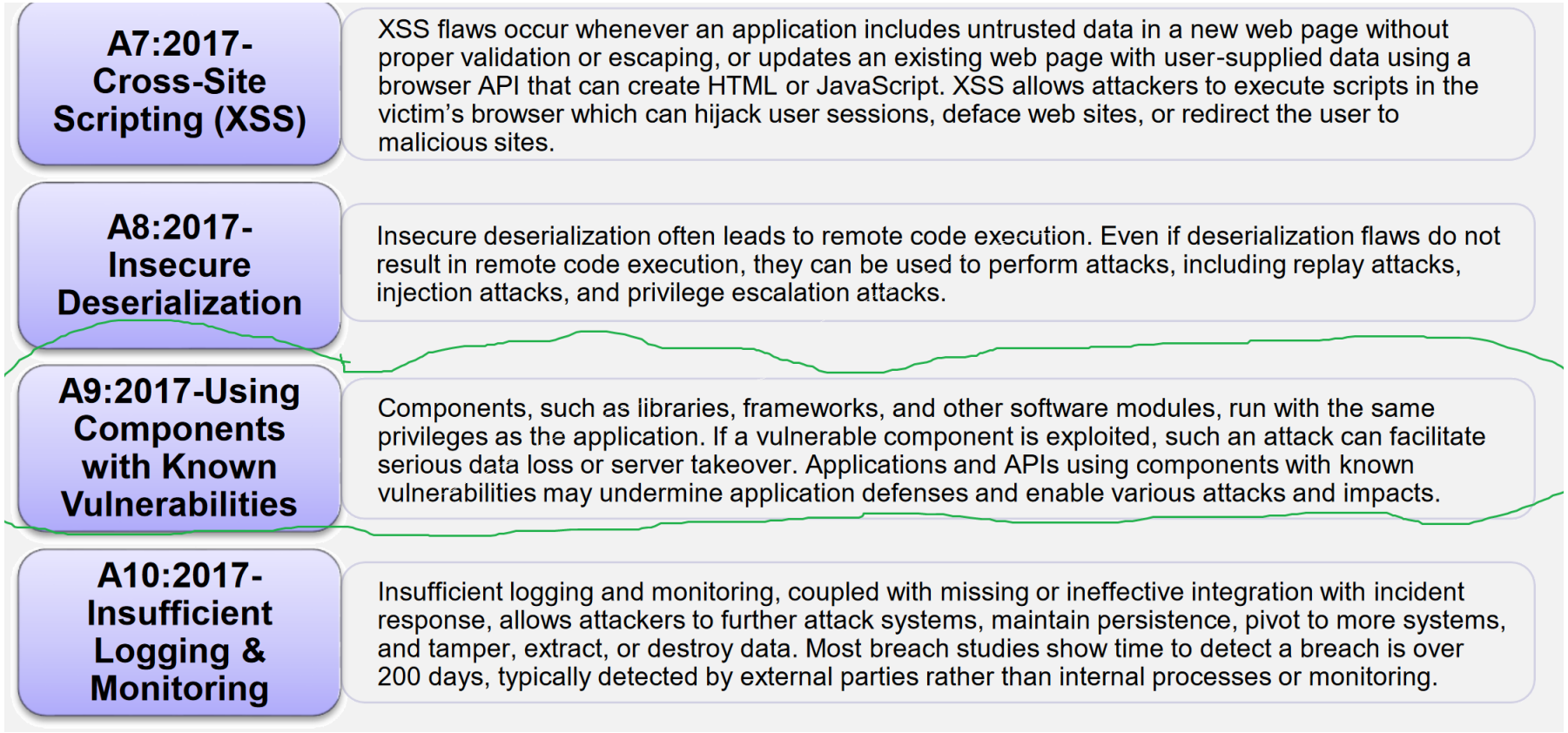
Bill of Material (BoM)

Challenge No.2 - Security

- *Which our projects have known open-source vulnerabilities?*
- *Do we have any components with critical and high vulnerabilities?*
- *Do our projects have the **XXX** vulnerable component?*

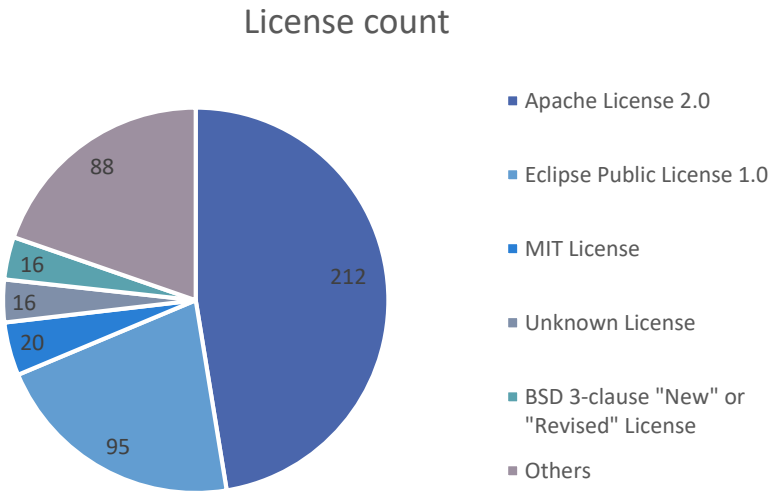
CISO/Security Manager

Challenge No.2 - Security



Challenge No.3 - Licensing

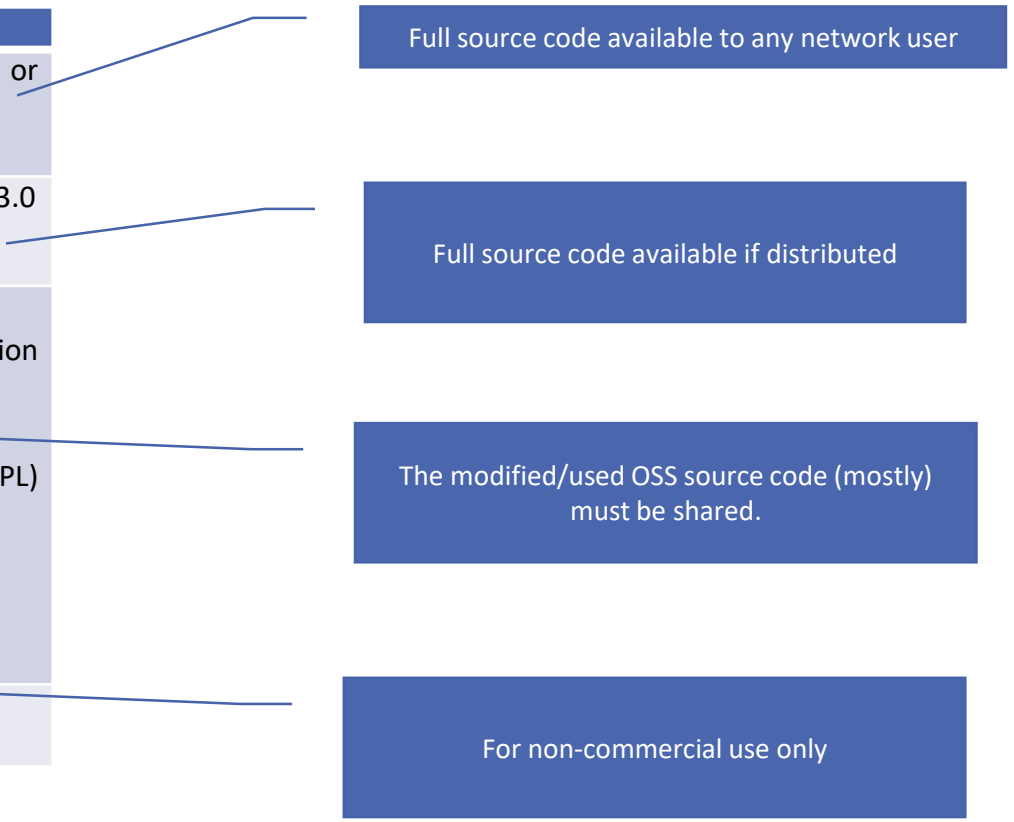
- *Are we allowed to share/distribute my software in its current form?*
- *Do we have any licenses non-compliant with our internal FOSS policy?*
- *Do we distribute any software with a copyleft license?*



Lawyers

Beware of these license families

Licensing scheme	License Family	Examples
Copyleft	Affero General Public License (AGPL)	GNU Affero General Public License v3 or later
Copyleft	Reciprocal	GNU General Public License (GPL) 2.0 or 3.0 Sun GPL with Classpath Exception v2.0
Copyleft	Weak Reciprocal	Code Project Open License 1.02 Common Development and Distribution License (CDDL) 1.0 or 1.1 Eclipse Public License GNU Lesser General Public License (LGPL) 2.1 or 3.0 Microsoft Reciprocal License Mozilla
Non-commercial use	Non-commercial	AFPL JRL

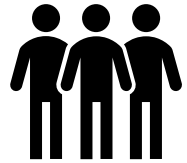


License breach – is it really suable?

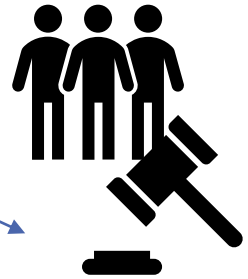
2017 - Artifex Software, Inc. versus Hancom, Inc.

Artifex Software

Hancom Inc.



- 1. Developed open-source PDF interpreter
- 2. The interpreter has a dual license: either GPL or commercial
- 3. Used the interpreter in the commercial Office software
- 4. Hancom neither paid for the commercial license nor published the custom software as open-source -> license infringement



US District Court

5. GPL can be treated like a legal contract

https://www.linux.com/blog/artifex-v-hancom-open-source-now-enforceable-contract

Challenge No.4 – Operational risks

- *How well is the component maintained?*
- *Is there any support?*
- *Are security vulnerabilities/bugs fixed within tolerable time?*
- *How large is the community?*
- *What is plan B if there is no new update?*

Developers, architects

Challenge No.5 – Date protection

Does any of my open-source components access sensitive data and if yes, what happens with that data?

- User tracking
- Data collection
- GDPR

Data protection officer

Who wins?

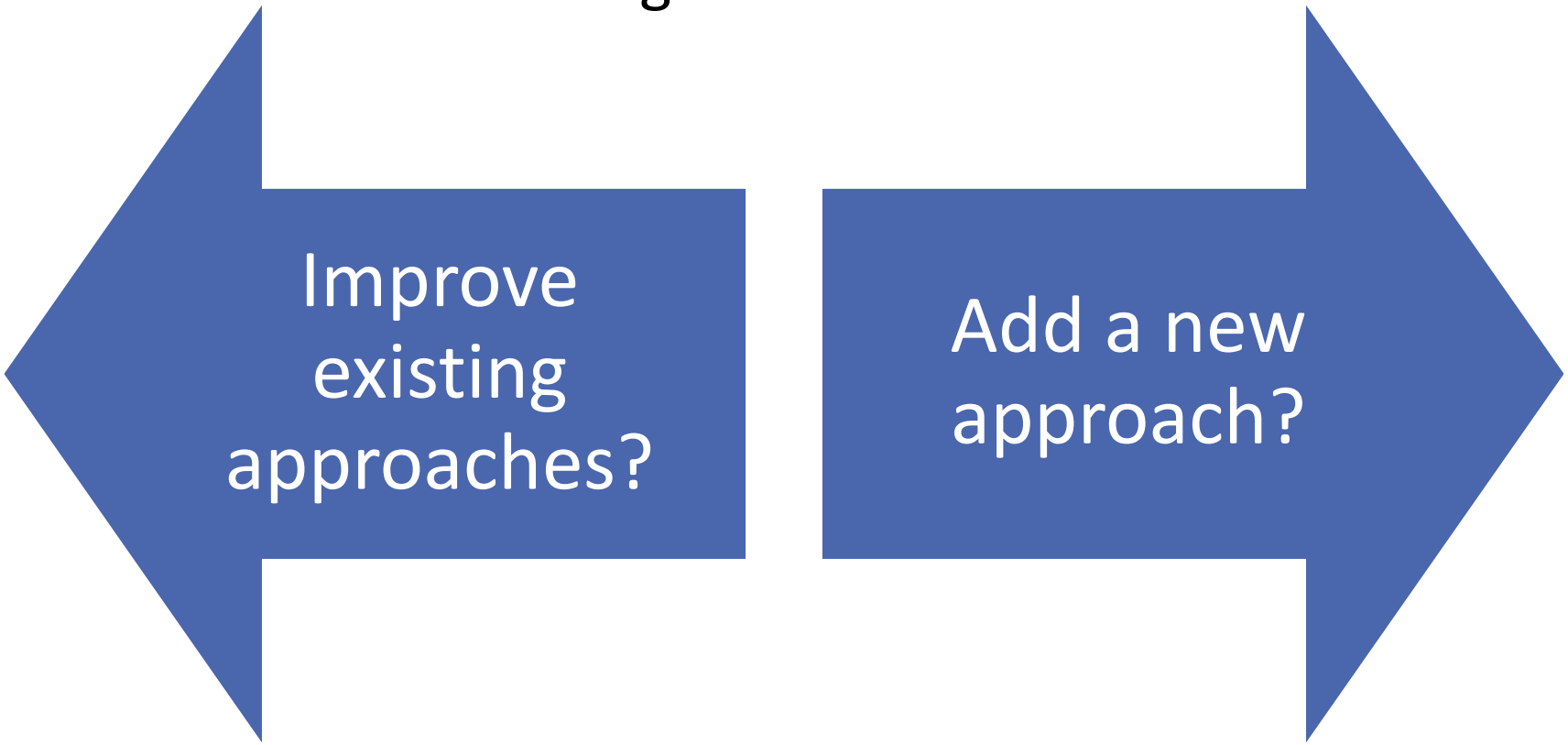
FOSS advantages

Agenda

- Introduction
- Challenges
- **Approaches**
- Integration
- Q & A

Approach

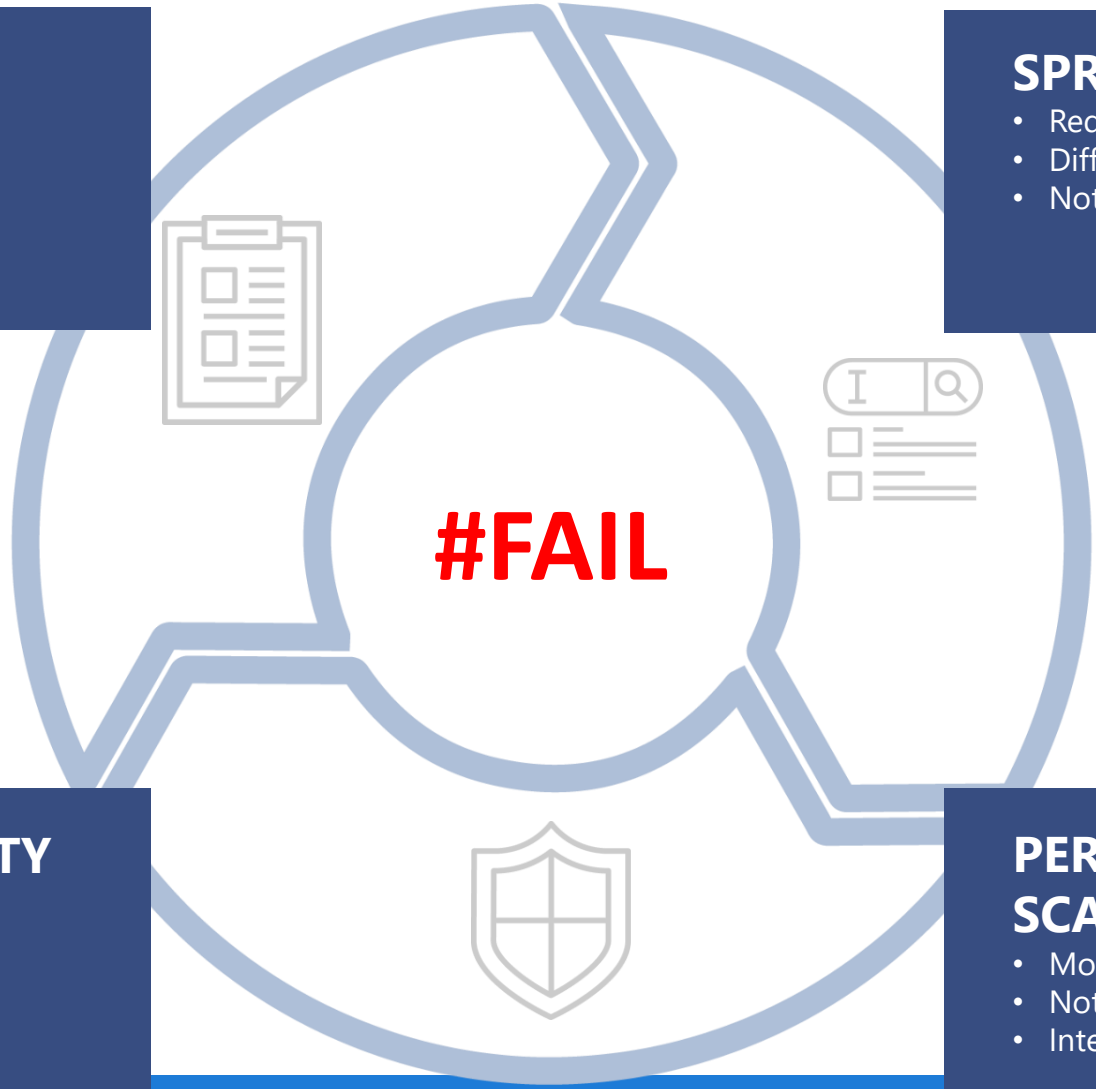
How to deal with our 4 challenges?



Common manual approaches

MANUAL DISCOVERY
- Cumbersome processes
- Occurs at end of SDLC
- High effort and low accuracy
- No ongoing controls

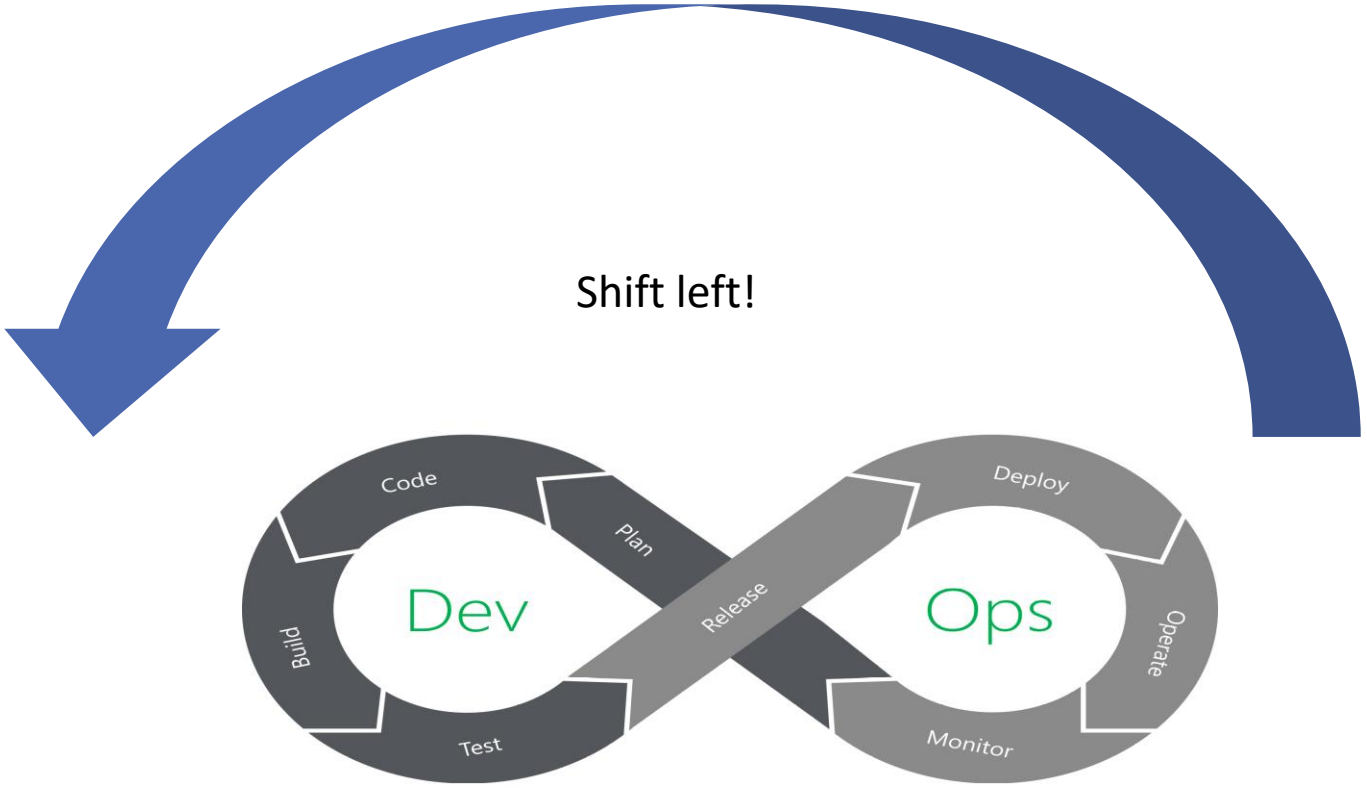
SPREADSHEET INVENTORY
- Requires consistent developer input
- Difficult to maintain and scale
- Not a full/accurate list of actual usage



SPORADIC VULNERABILITY TRACKING
- No single responsible entity
- Labor intensive manual effort
- Unmanageable (~11 new vulns/day)

PERIODIC VULNERABILITY SCANNING
- Monthly/quarterly vulnerability assessments
- Not aimed at open source vulnerabilities
- Integrated later in the SDLC

Common automated approaches



Common automated SCA approaches (1/5)

Source code repository checks

+ Examines open source components automatically – no triggered scan needed
+ Known FOSS security vulnerabilities with CVE are reported
+ Visualisation
+ Often easy remediation in the repository -> replacement of the vulnerable component
+ Alerts sent and displayed for new vulnerabilities
+ Continuous analysis

- Focus on dependencies but no code snippets or modified files/directories
- Often no licenses overview
- Reporting

Common automated SCA approaches (1/5)

Using GitHub source code repository checks

E-mail Alert

We found a potential security vulnerability in a repository for which you have been granted security alert access.

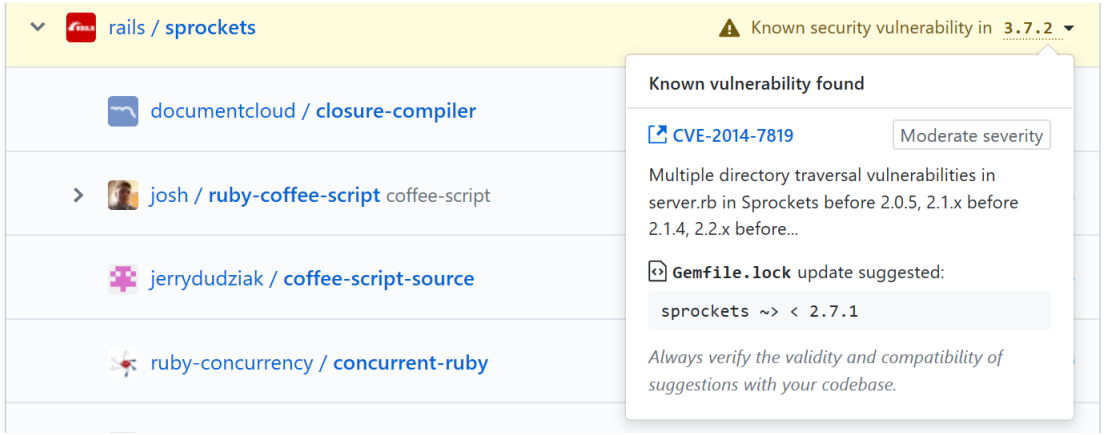
wakeman83/Dependencies

Known **moderate severity** security vulnerability detected in `sprockets >= 2.6.0` defined in `Gemfile.lock`.

`Gemfile.lock` update suggested: `sprockets ~> < 2.7.1`.

Always verify the validity and compatibility of suggestions with your codebase.

Dependency Graph



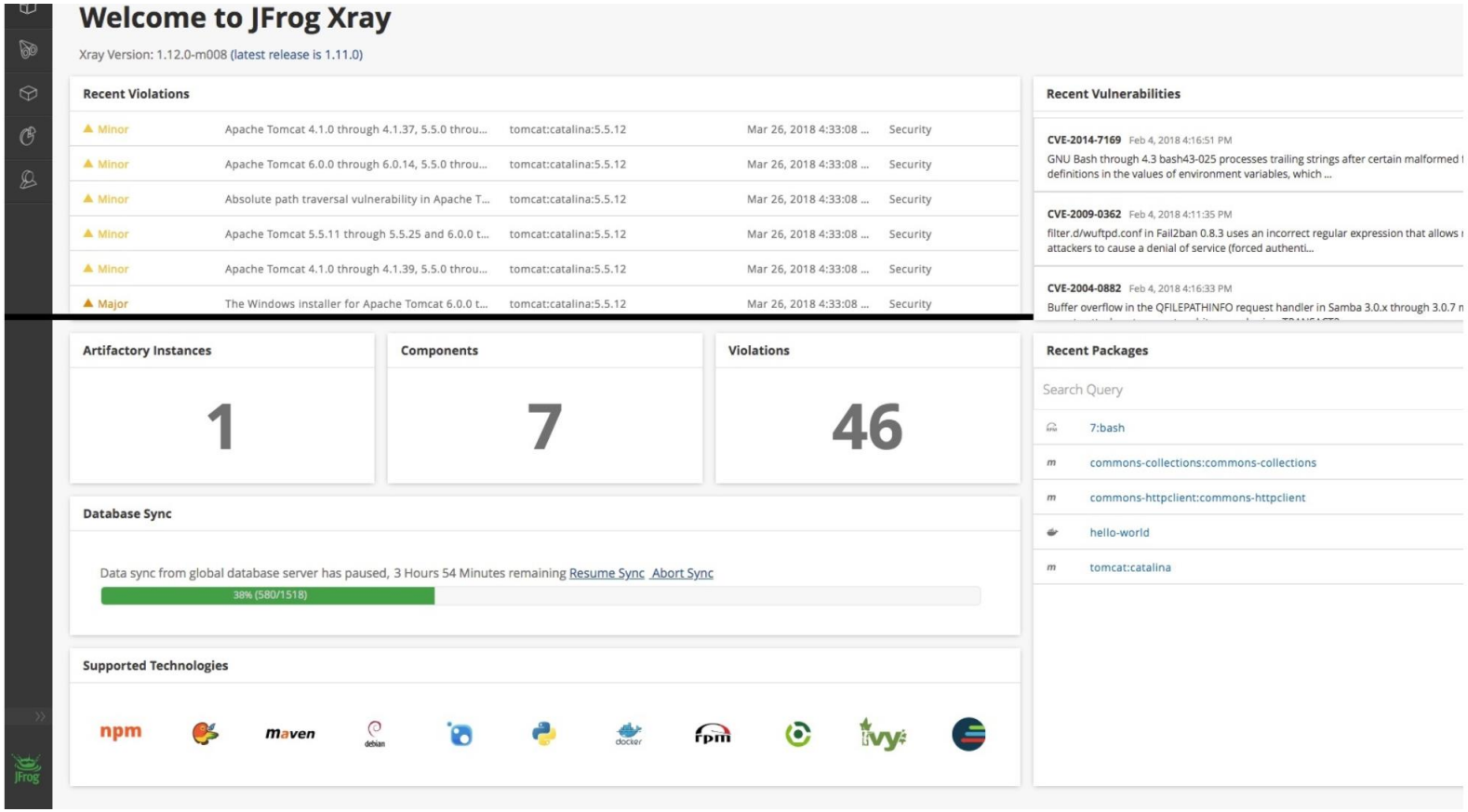
Common automated SCA approaches (2/5)

Binary repository manager checks

+ Examines all binary components known for open-source vulnerabilities
+ Easy access to artifacts
+ Can be triggered on-demand or automatically when new artifacts appear
+ Easy implementation of approved artifacts only (due to licensing, whitelisting,…)
+ Dependency graph
+ Easy integration
+ Continuous analysis
- Only successful if all artifacts stored there -> single source of truth
- Can miss references in Source Code repositories
- Licensing information?
- Reporting

Common automated SCA approaches (2/5)

Binary repository manager checks - example



Common automated SCA approaches (3/5)

SAST

Static Application Security Testing

- Analyzes any source code, not only FOSS specific
- Finds common vulnerability patterns:
  - SQL injection
  - Cross-site scripting
  - Buffer overflows, etc.

+ Finds both publicly known and unknown security vulnerabilities in the source code

+ No additional tool/stage needed

+ SAST can be performed in various pipeline stages

+ SAST tools can have a separate module that inspects software composition

- Limited insight into Software Composition Analysis
- No Bill of Material
- No licensing information
- Results represent a point in time

Common automated SCA approaches (4/5)

DAST

Dynamic Application Security Testing

- Tests running apps
- Finds vulnerable app behavior:
  - Misconfigurations
  - Authentication issues

+ Finds both publicly known and unknown security vulnerabilities

+ No additional tool/stage needed

+ Fewer false positives than SAST

- Limited insight into Software Composition Analysis as it examines running software from outside

- Runs later in a later pipeline stage

- Very incomplete Bill of Material

- No licensing information

- Results represent a point in time

Common automated SCA approaches (5/5)

SCA Testing

Software Composition Analysis (Testing)

- Scans for open source
- Provides Bill of Material
- Finds Open Source licenses
- Finds open source vulnerabilities:
  - Detects known vulns
  - Works through full SDLC
  - Monitors for new vulns

+ Focused on Open Source Components

+ Few false positives due to several ways of identifying FOSS components

+ Both compiled and uncompiled code can be analysed

+ Usually faster in scanning FOSS components

+ Public and private vulnerability databases

+ Can integrate with other application security testing metrics

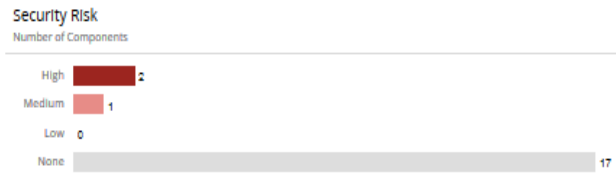
- Yet another stage/tool to implement
- Does not find publicly unknown vulnerabilities, so need to be complemented with SAST/DAST

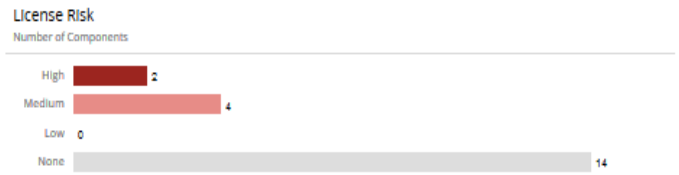
Software composition analysis

SCA is a process that can determine all underlying components of a software and identify at least the public known (open-source) components.

A well defined process is consistent, automated and measurable.

Commercial SCA tools (1/2)

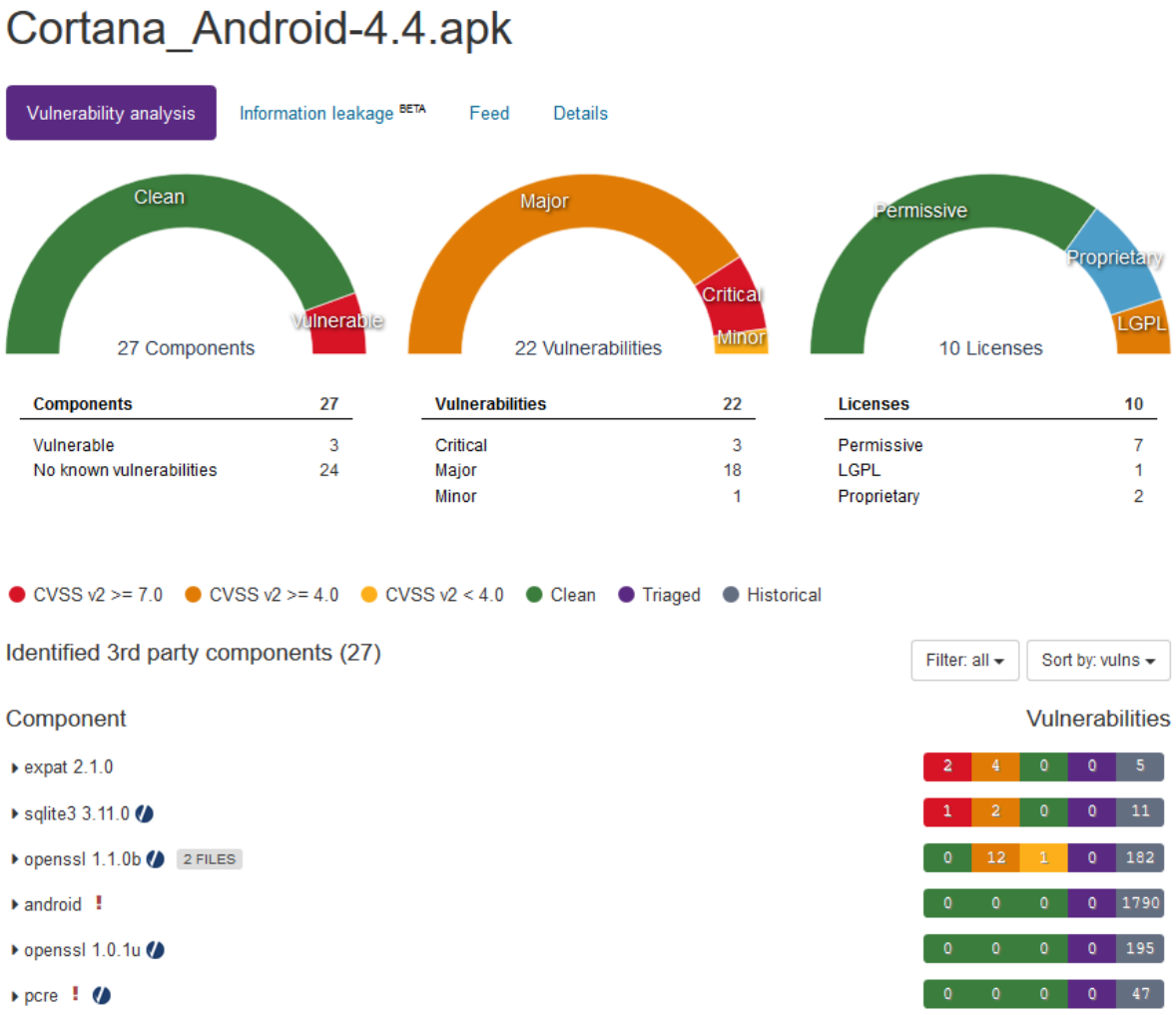




Component List | Add | Compare to... | Print...

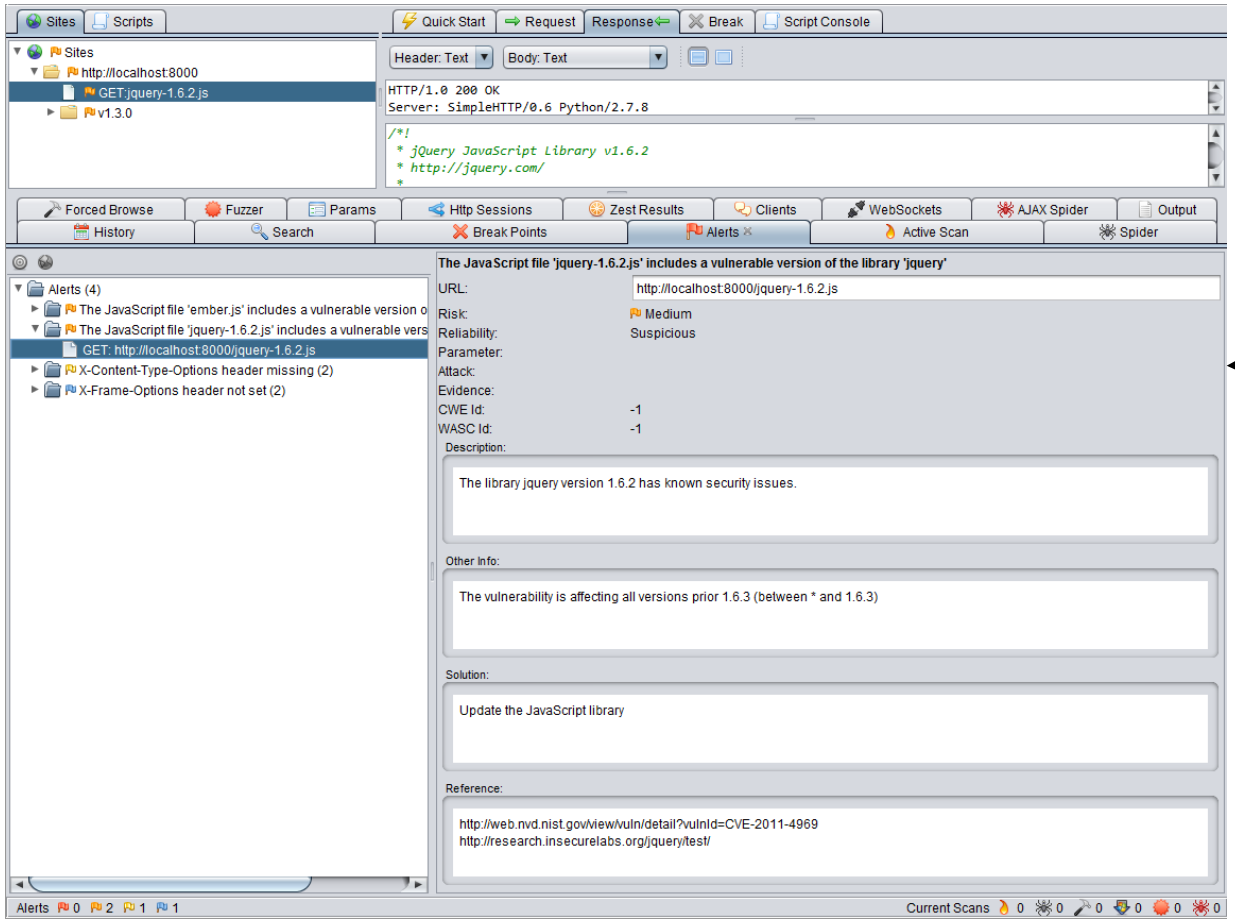
Component	Source	Match Type	Usage	License	Security Risk	Operational Risk
Geronimo Eclipse Plugin :: Assembly 2.2.1	4 Matches	Exact Directory	Dynamically Linked	Apache-2.0		Medium
Geronimo Framework, Modules :: Common 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Crypto 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Deploy Config 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Deploy JSR-88 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Deployment 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Kernel 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: Plugin 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
Geronimo Framework, Modules :: System 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
geronimo-j2ee-schema 2.2.1	1 Match	Exact Directory	Dynamically Linked	Apache-2.0		High
OpenSSL 1.0.2n		Manually Added	Dynamically Linked	M SSLeay License	4 5	Low
OpenSSL 1.0.1l	12 Matches	Exact Directory, Files Modified	Dynamically Linked	M OpenSSL and 1 more...	11 45 16	Medium

Commercial SCA tools (2/2)



Open Source SCA tools (1/5)

RetireJS – JavaScript dependencies



Integration:

- A command line scanner
- A grunt plugin (NPM)
- A Chrome extension
- A Firefox extension
- Burp and OWASP Zap plugin
- Eclipse plugin

https://github.com/retirejs/retire.js/

Open Source SCA tools (2/5)

NPM Audit

```
=== npm audit security report ===

# Run  npm update tough-cookie --depth 6  to resolve 1 vulnerability
```

High	**Regular Expression Denial of Service**
Package	tough-cookie
Dependency of	@npm/spife
Path	@npm/spife > chokidar > fsevents > node-pre-gyp > request > tough-cookie
More info	https://nodesecurity.io/advisories/525

```
[!] 1 vulnerability found - Packages audited: 918 (466 dev, 87 optional)
    Severity: 1 High
```

- A command line scanner
- Focuses on NPM packages
- Suggest fixes -> easy remediation
- Package signing checks in the future?

https://blog.npmjs.org/post/173719309445/npm-audit-identify-and-fix-insecure

Open Source SCA tools (3/5)

OWASP Dependency Check

DEPENDENCY-CHECK

Dependency-Check is an open source tool performing a best effort analysis of 3rd party dependencies; false positives and false negatives may exist in the analysis performed by the tool. Use of the tool and the reporting provided constitutes acceptance for use in an AS IS condition, and there are NO warranties, implied or otherwise, with regard to the analysis or its use. Any use of the tool and the reporting provided is at the user's risk. In no event shall the copyright holder or OWASP be held liable for any damages whatsoever arising out of or in connection with the use of this tool, the analysis performed, or the resulting report.

Project: Demo Insecure Project

Scan Information (show all):

- *dependency-check version*: 1.3.1
- *Report Generated On*: Nov 3, 2015 at 23:20:33 EST
- *Dependencies Scanned*: 14
- *Vulnerable Dependencies*: 3
- *Vulnerabilities Found*: 13
- *Vulnerabilities Suppressed*: 0
- ...

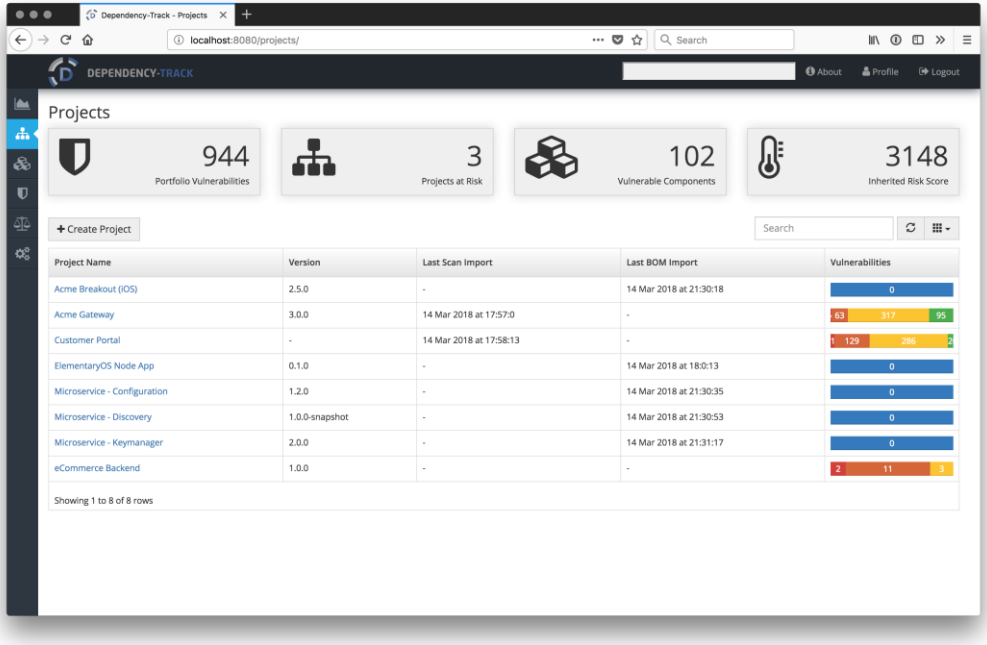
Display: Showing Vulnerable Dependencies (click to show all)

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count
commons-fileupload-1.3.jar	cpe:/a:apache:commons_fileupload:1.3	commons-fileupload:commons-fileupload:1.3	Medium	1	HIGHEST	29
struts2-core-2.3.15.3.jar	cpe:/a:apache:struts:2.3.15.3	org.apache.struts:struts2-core:2.3.15.3	High	6	HIGHEST	25
xwork-core-2.3.15.3.jar	cpe:/a:apache:struts:2.3.15.3	org.apache.struts.xwork:xwork-core:2.3.15.3	High	6	HIGHEST	24

Dependencies

https://jeremylong.github.io/DependencyCheck/

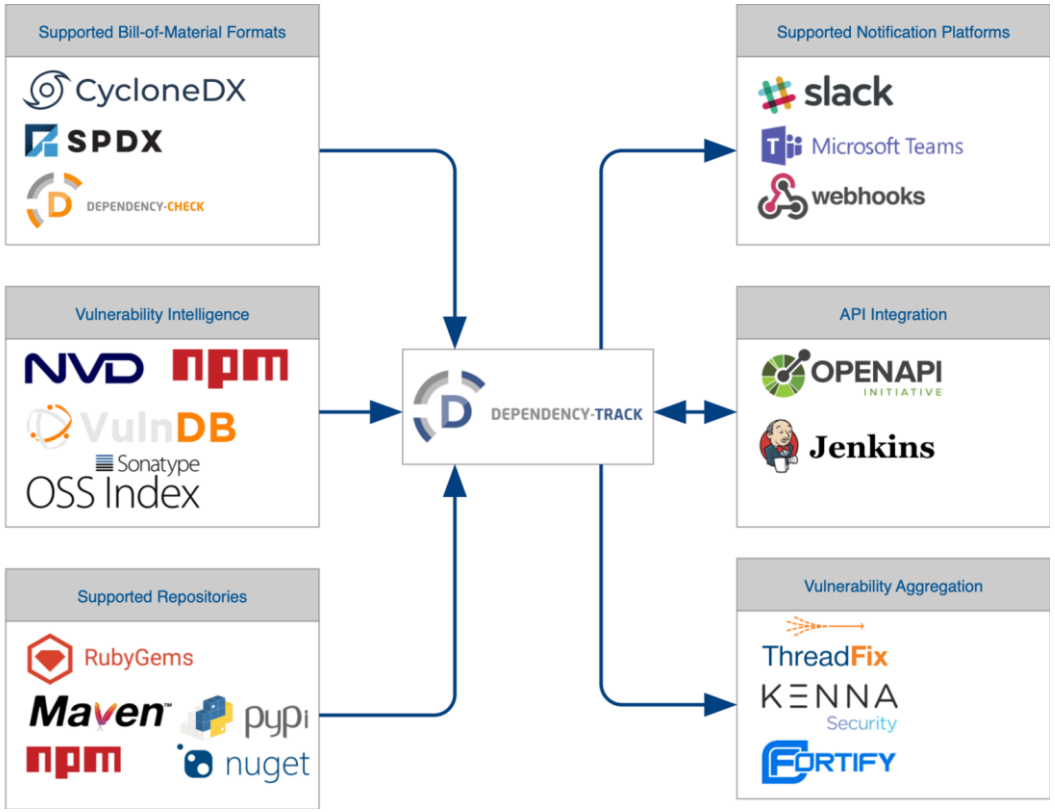
OWASP Dependency Track



https://www.owasp.org/index.php/OWASP_Dependency_Track_Project

Open Source SCA tools (4/5)

Dependency Track – THE open source tool for SCA

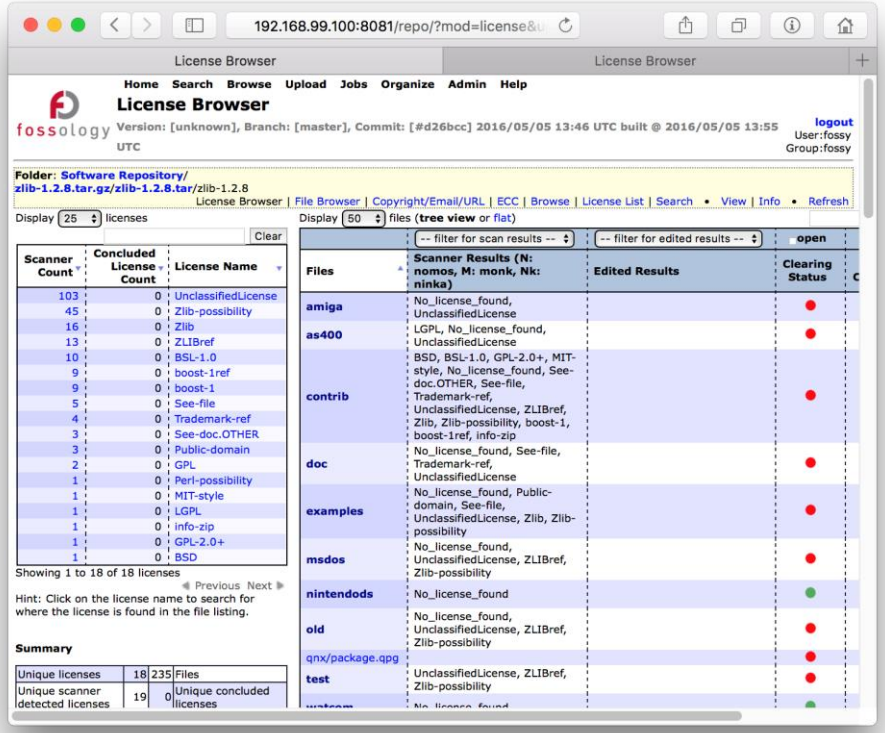


https://www.owasp.org/index.php/OWASP_Dependency_Track_Project

Open Source SCA tools (5/5)

Open-Source tools examples for finding licensing issues

- OSS Review Toolkit https://github.com/heremaps/oss-review-toolkit
- Fossology https://www.fossology.org/
- SW360 https://sw360.github.io/



SCA decision table

Profile	Recommendation
Developer startup with JS frameworks	Use technology-specific tools such as RetireJS, npm audit,…
SMB with multiple technologies and powerful development teams	Use binary repository manager add-ons or source control versioning mechanisms
SMB with multiple technologies at SCA beginning with focus on security	Use or start with OWASP Dependency Track
SMB with multiple technologies at SCA beginning with focus on compliance	Use open-source tools such as Fossology/OSS Review Toolkit
Enterprises with clear SCA requirements and multiple stakeholders: CISO, Legal, Developers, Open-Source Officers	Start with OWASP Dependency Track and/or Evaluate commercial SCA tools

KEYS TO open source security management

1. Contextual identification
2. Complete vulnerability and legal data
3. Zero-day notification
4. Timely remediation
5. Efficient policy management
6. Integrate and automate

Agenda

- **Introduction**
- Challenges
- Approaches
- **Integration in SDLC**
- Q & A

Requirements

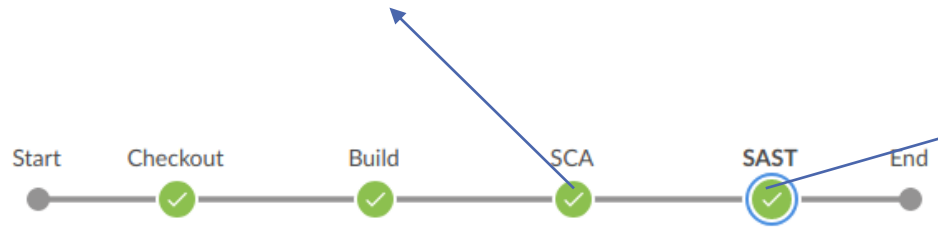
CI/CD

- Automatable
- User-friendly
- Actionable
- Flexible/Open
- Easy to integrate

Application security pipeline

CI/CD Pipeline

```
2018-09-14 14:25:13 INFO [main] --- Starting the Hub signature scans
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- Starting the signature scan of /var/lib/jenkins/workspace/test_pipeline1
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- Hub CLI command :
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- /var/lib/jenkins/blackduck/tools/Hub_Scan_Installation/scan.cli-4.8.2/jre/bin/java
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- -Done-jar.silent=true
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- -Done-jar.jar.path=/var/lib/jenkins/blackduck/tools/Hub_Scan_Installation/scan.cli-4.8.2/lib/cache/scan.cli.impl-standalone.jar
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- -Xmx4096m
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- -jar
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- /var/lib/jenkins/blackduck/tools/Hub_Scan_Installation/scan.cli-4.8.2/lib/scan.cli-4.8.2-standalone.jar
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --no-prompt
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --scheme
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- https
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --host
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- hubsig.blackducksoftware.com
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --port
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- 443
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- -v
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --logDir
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- /var/lib/jenkins/blackduck/scan/HubScanLogs/2018-09-14_12-25-13-022_17
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --statusWriteDir
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- /var/lib/jenkins/blackduck/scan/HubScanLogs/2018-09-14_12-25-13-022_17
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --project
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- WebGoat
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --release
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- 8.0
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- --name
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- test_pipeline1/WebGoat/8.0 scan
2018-09-14 14:25:13 INFO [pool-2-thread-1] --- /var/lib/jenkins/workspace/test_pipeline1
```




```
 INFO] Sensor Black Duck Hub Plugin for SonarQube [hubsonarqube]
[INFO] Successfully connected to https://hubsig.blackducksoftware.com
[INFO] Gathering local component files...
[INFO] Gathering Hub component files...
[INFO] Getting matched files for Apache Ant...
[INFO] Getting matched files for Apache Commons Compress...
[INFO] Getting matched files for Apache Maven 2...
[INFO] Getting matched files for Apache Tomcat...
[INFO] Getting matched files for Bootstrap (Twitter)...
[INFO] Getting matched files for Bouncy Castle...
,,,
[INFO] Getting matched files for Spring Data Commons...
[INFO] Getting matched files for Spring Framework...
[INFO] Getting matched files for Spring Security...
[INFO] Getting matched files for Spring TestContext Framework...
[INFO] Getting matched files for Spring Transaction...
[INFO] Getting matched files for XStream...
[INFO] --> Number of local files matching inclusion/exclusion patterns: 8
[INFO] --> Number of vulnerable Hub component files matched: 8
[INFO] Comparing local components to Hub components...
```

Interesting Links

- Copyright trolling https://blog.fossa.io/patrick-mchardy-and-copyright-profiteering-44f7c28c0693
- GitHub and SCA https://www.dev-insider.de/security-alerts-auf-github-nutzen-a-758877/
- Open Source Metadata https://clearlydefined.io/about

Q&A

Stanislav.Sivak@synopsys.com