IT-SA 2019

Mitigating the risk of using Open Source code in Application Development

@d3v_rand0m
Julian Totzek-Hallhuber

Explosive Grow of Open Source

31M+

Developers on GitHub

2M+

Organizations

40+% grow in 2018

100M+

Repositories as of April 2019

200M+

Pull Requests in 2018

Source: octoverse.github.com

App Development with Open Source

830K +

NPM Modules with 476 new/day

280K +

Maven Central Modules with 182 new/day

225K +

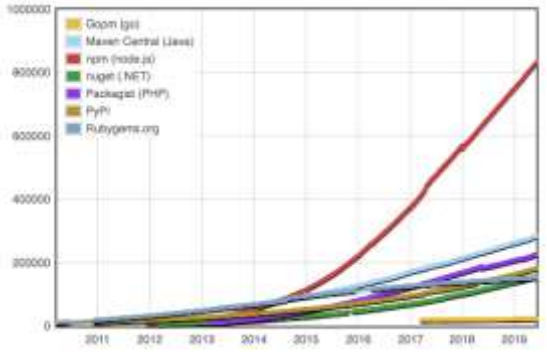
Packagist Modules with 153 new/day

180K +

PyPI Modules with 116 new/day

150K +

Nuget Modules with 121 new/day



Famous Attacks related to Open Source Libraries

Heartbleed

OpenSSL Library

Thousands of affected including JP Morgan, Routers and Canadian Tax Agency

Shellshock

Unix Bash shell

Thousands affected via bots creating DDoS

StageFright

7 Vulnerabilities on Android OS for remote code execution, affected

most Android devices in 2015

Apache Struts

Remote code execution exposing data from 144 million Equifax customers. $700 million settlement

Major Challenges Solving OSS Security

Silent Fixes

Risk prioritization

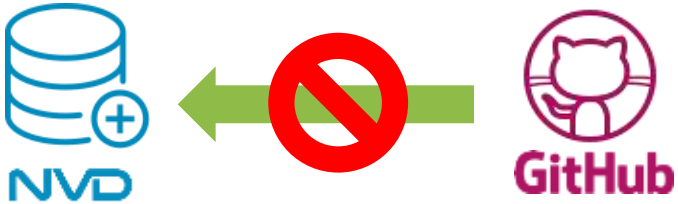
Transitive vulnerabilities

Speed of DevOps

Open Source Is Breaking the NVD Model

Silent Fixes

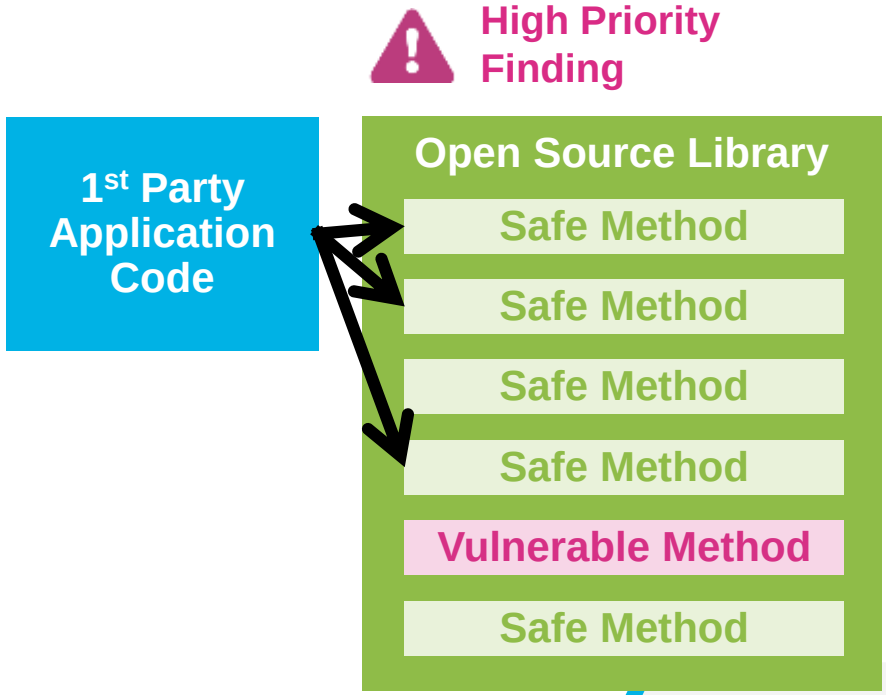
- NVD was designed for a different era
  - Fewer large commercial vendors
  - Manual, tightly controlled process
- OSS development embrace DevOps
  - NVD cannot cope with velocity and volume of submissions
- NVD CVEs do not provide exact library, vulnerable versions, and vulnerable code segment
- Hackers are watching OSS commits for silent fixes of vulnerabilities they can exploit in the wild



Not Every OSS Vulnerability Is Exploitable

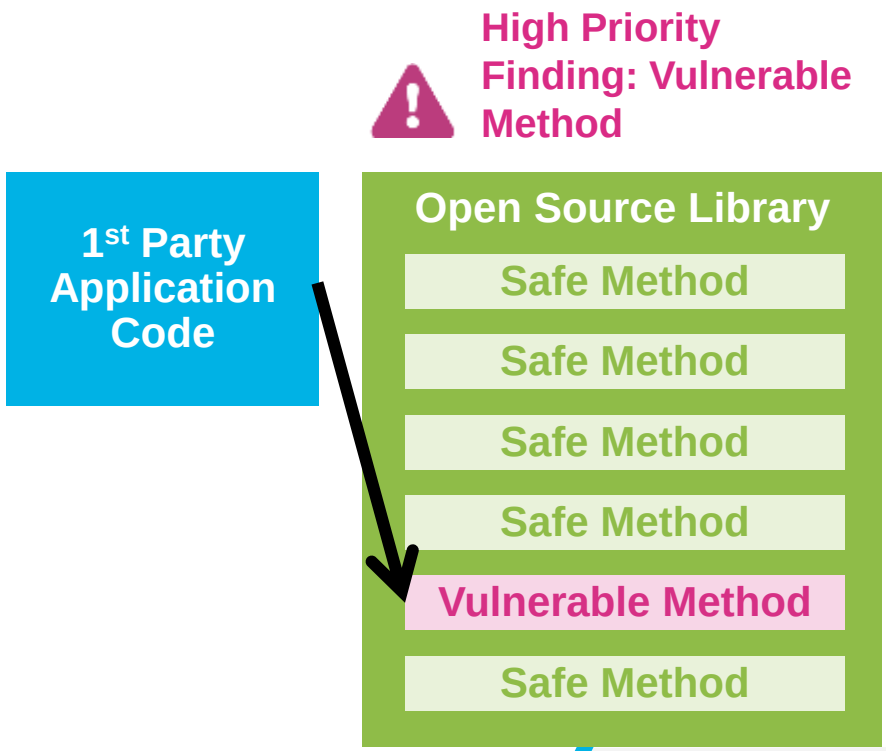
Risk prioritization

- Each OSS library may have 100s of functions and methods
- Vulnerability are usually only tied to one of these
- Your first party code only calls only a handful of methods in the library
- Most solutions don't allow you to prioritize applications where the vulnerable function is being called



Vulnerable Methods Analysis

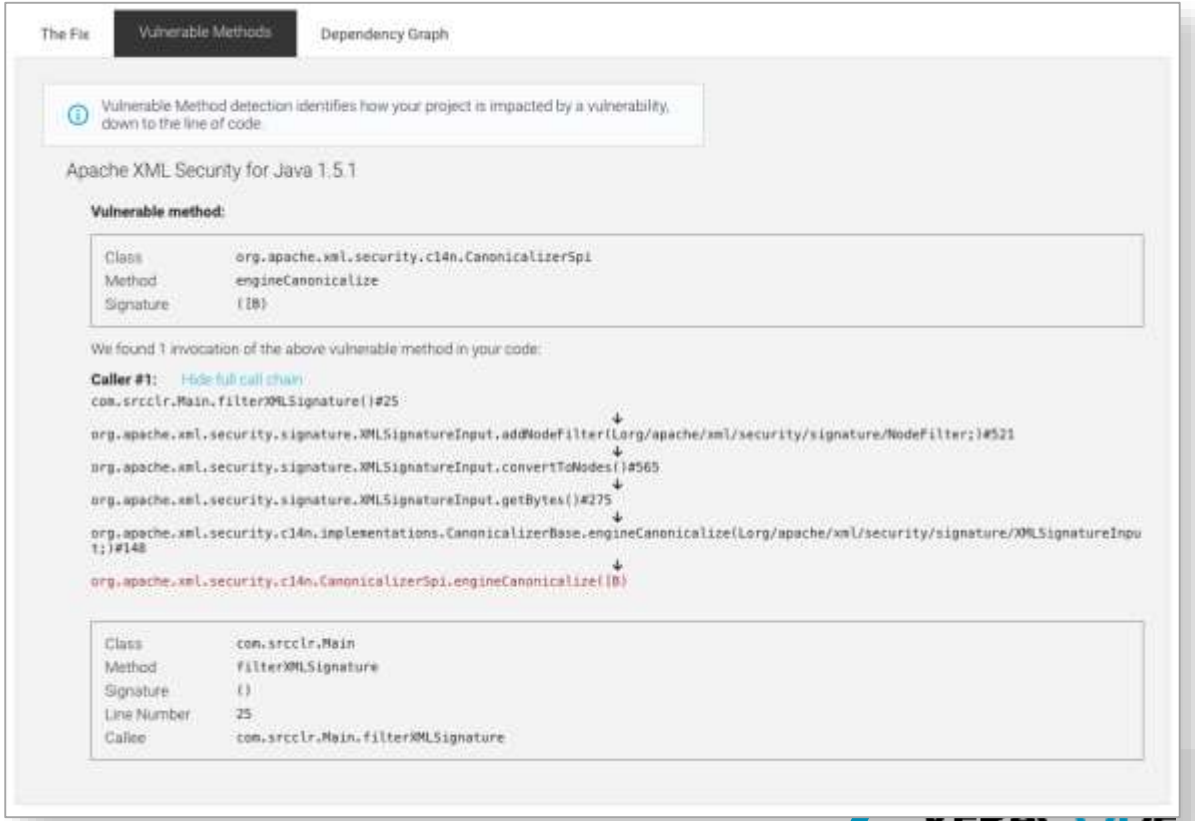
- Which method contains vulnerabilities down to the exact line of code, reducing remediation work
- Curation and expert annotations on vulnerable methods to provide actionable results
- Directly or Indirectly using a vulnerable method in your code makes you vulnerable
- Available in Java, .NET, Python, and Ruby



* Requires agent-based scanning

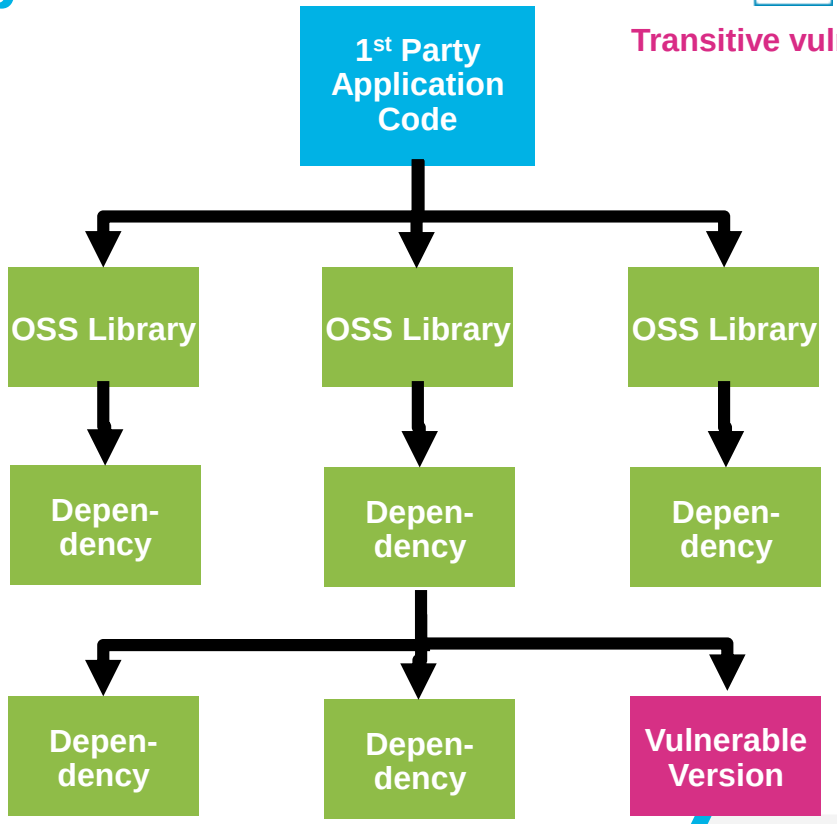
Example: Vulnerable Method

- Call graph analysis
- Pinpoint vulnerable method
- Pinpoint used vulnerability
- Allows you to prioritize exploitable vulnerabilities



Risk Can Hide Layers Deep

- Vulnerabilities may lie in direct dependencies or much deeper – in dependencies of dependencies
- Developers do not test transitive dependencies
- It's easy to miss vulnerabilities hidden in transitive dependencies



Full Language Support

For Application Upload and Agent-based scanning

JavaScript/Node.js

C/C++

.NET/C#

PHP

Objective-C

Swift

Ruby

Java

Python

Golang

Scala

- C/C++ in roadmap for App Upload scan

Supported Build Tools & Package Managers

- **C#:** NuGet, Chocolatey
- **C/C++:** Makefile
- **Golang:** Trash, Glide, GoVendor, GoDep, GoGet, Dep
- **Java:** Maven, Gradle, Ant, Ivy, Jar
- **JavaScript/ Node.js:** NPM, Yarn, Bower
- **Objective-C:** Cocoapods
- **PHP:** Packagist/ Composer
- **Python:** Pip
- **Ruby:** RubyGems
- **Scala:** SBT

Update Advisor

- Update Advisor checks the most suitable safe versions to update to for all vulnerable direct libraries, and adds an “Update Advisor” section in the report at the end of a scan.
- Each item shows the vulnerable library, the safe version that Update Advisor finds most suitable to update to, and possibly-breaking update for Java, Python and Ruby
- For direct libraries only
- srcclr scan <directory> --update-advisor

Update Advisor

Library Name & Version	Safe Version	Breaking Update
Neo4j - JMX support 1.3	3.0.0-M05	No
H2 Database Engine 1.3.176	1.4.198	No
JavaMelody - Core 1.59.0	1.74.0	No
OrientDB Server 2.1.9	2.1.11	No
Keycloak SAML Core 1.8.1.Final	2.5.5.Final	No
Apache Sling Engine Implementation 2.0.4-incubator	2.4.6	No
Spring Web 3.1.1.RELEASE	4.3.20.RELEASE	Yes

Auto Pull Request

- Automatic pull requests to update vulnerable libraries:
  - Scan
  - Report Vulnerabilities
  - Report Fix
  - Create automatically a pull request with the fix
- Auto Pull Request helps users remove vulnerable libraries in their projects by modifying the package dependency files and creating pull requests.
- Auto Pull Request is built on top of Update Advisor, using those recommendations to create the Git pull requests.
- Auto Pull Request supports direct libraries found on projects in GitHub, GitHub Enterprise, and GitLab. srcclr scan <directory> --pull-request


VERACODE
You change the world, we'll secure it.