

ScanCode to reuse FOSS safely, with FOSS

ScanCode

Agenda

- About me, AboutCode, and nexB
- Why should we scan code?
 - Licenses change, vulnerabilities present, quality issues, regulations
- How to communicate? Open Standards:
 - Package-URL (PURL), Vers, SBOMs (SPDX, CycloneDx), VDRs, VEX
- Problems and solutions
 - Knowing your full list of exact dependencies
 - what can you do to avoid issues
 - Hard to identify all packages, too much vulnerabilities
- Why use FOSS tools for code scanning?
- Questions

About me

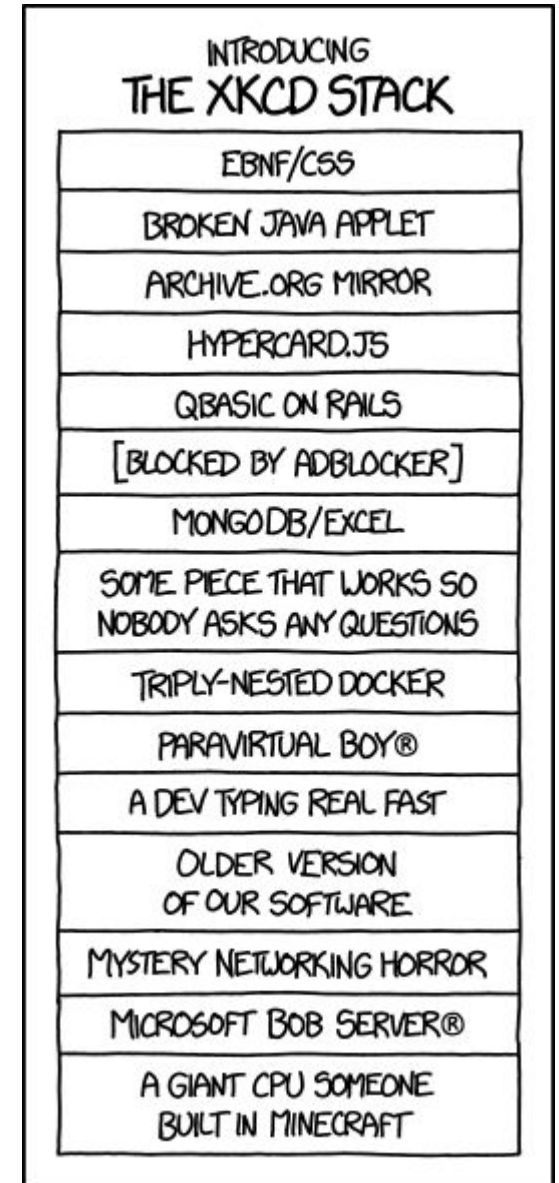
- Core maintainer of [ScanCode](#) (scancode-toolkit and scancode.io)
 - Contributes to and helps maintain other AboutCode tools: [license-expression](#) [licenseDB](#) [scancode-workbench](#) [PURLdb](#)
- Working primarily on license detection, package identification and data summarization
- Google Summer of Code Mentor at AboutCode from 2021
 - participant in GSoC2020 and GSoD2019
- Ayan (pronounced like: awyon):
 - asmahapatra@aboutcode.org
 - GitHub: <https://github.com/AyanSinhaMahapatra/>
 - LinkedIn: <https://www.linkedin.com/in/ayansinhaju/>

AboutCode and nexB

- AboutCode's FOSS-first mission: FOSS for FOSS
 - Open source tools and open knowledge base (AboutCode stack)
 - Simple and practical standards (Package-URL)
 - Applications for Legal Business users (DejaCode, also FOSS) with APIs
- Trusted experts on Software Composition Analysis (SCA) since 2007
 - Creator of Package-URL: <https://github.com/package-url>
 - Co-founders of SPDX: <https://spdx.org>
 - Contributors to CycloneDX: <https://cyclonedx.org>
 - Co-founders of ClearlyDefined: <https://clearlydefined.io>
- nexB: professional services for SCA
 - 800+ SCA projects completed to-date
 - Sponsored development for AboutCode projects
 - Technical support and advisory for SCA process, and deployments

Modern software ecosystem

- FOSS software packages are reused
 - small apps routinely embed 500 FOSS packages
 - large apps: 10,000+!
- Everyday, you have new vulnerabilities, license problems and package updates in your package dependency trees
 - Impossible to check this manually!
- Goal: Discover the problems and help alleviate the pain



Source: <https://xkcd.com/1636/>

Why is Software License important?

- FOSS: Freedom
- Freedom and Responsibilities
 - Can we use the software in different scenarios?
 - Can we modify and redistribute freely, under my choice of terms?
 - Give credit, generate attribution
- See [License categories](#) for more details
- Copyrights:
 - Copyright notices often have to be included and redistributed
- [History of Litigation](#)
 - GPL based court rulings to Distribute Source Code

Why is identification important?

- Modifications can be released under different terms
- License could change between versions
 - packages/products often decide to change their license
 - <https://redis.com/blog/redis-adopts-dual-source-available-licensing/>
 - <https://www.elastic.co/blog/elastic-license-update>

Redis' License is BSD and will remain BSD



Yiftach Shoolman
August 22, 2018



Redis Adopts Dual Source-Available Licensing



Rowan Trollope
March 20, 2024



Why is identification important?

- Vulnerabilities are introduced and fixed by versions (or not!)

VulnerableCode.io Packages Vulnerabilities Documentation

Search for vulnerabilities ?

GHSA-jfh8-c2jp-5v3q

Vulnerability details: **VCID-bk15-3vac-aaaj**

Essentials **Fixed by** packages (50) **Affected** packages (463)



Vulnerability ID	VCID-bk15-3vac-aaaj
Aliases	CVE-2021-44228 GHSA-jfh8-c2jp-5v3q
Summary	Remote code injection in Log4j
Severity score range	0.1 - 10.0
Status	Published

GitHub Advisory Database / GitHub Reviewed / CVE-2021-44228

Remote code injection in Log4j

Critical severity GitHub Reviewed Published on Dec 10, 2021 to the GitHub Advisory Database • Updated on Feb 6

Vulnerability details Dependabot alerts 0

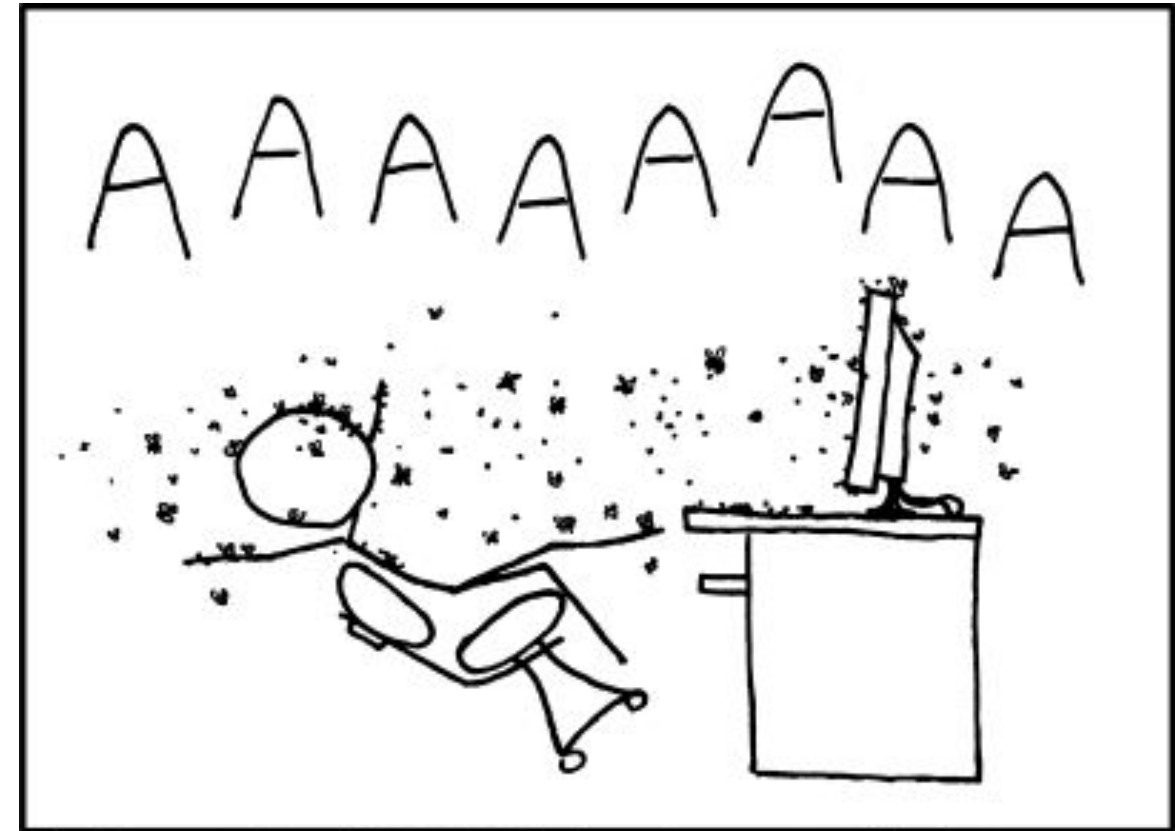
Package	Affected versions	Patched versions
 com.guicedee.services:log4j-core (Maven)	<= 1.2.1.2-jre17	None
 org.apache.logging.log4j:log4j-core (Maven)	>= 2.13.0, < 2.15.0 >= 2.4, < 2.12.2 >= 2.0-beta9, < 2.3.1	2.15.0 2.12.2 2.3.1

Sources:

<https://public.vulnerablecode.io/vulnerabilities/VCID-bk15-3vac-aaaj?search=GHSA-jfh8-c2jp-5v3q>
<https://github.com/advisories/GHSA-jfh8-c2jp-5v3q>

Why is software quality important?

- Better maintained: more secure
 - at least a correlation!
- Code review, branch protection and other quality checks are important
- Great FOSS projects on quality:
 - [OpenSSF Scorecard](#)
 - [endoflife.date](#)
 - [CHAOSS: auger](#), [grimorelab](#)
- Still lots of context not captured by metrics, not clearly actionable

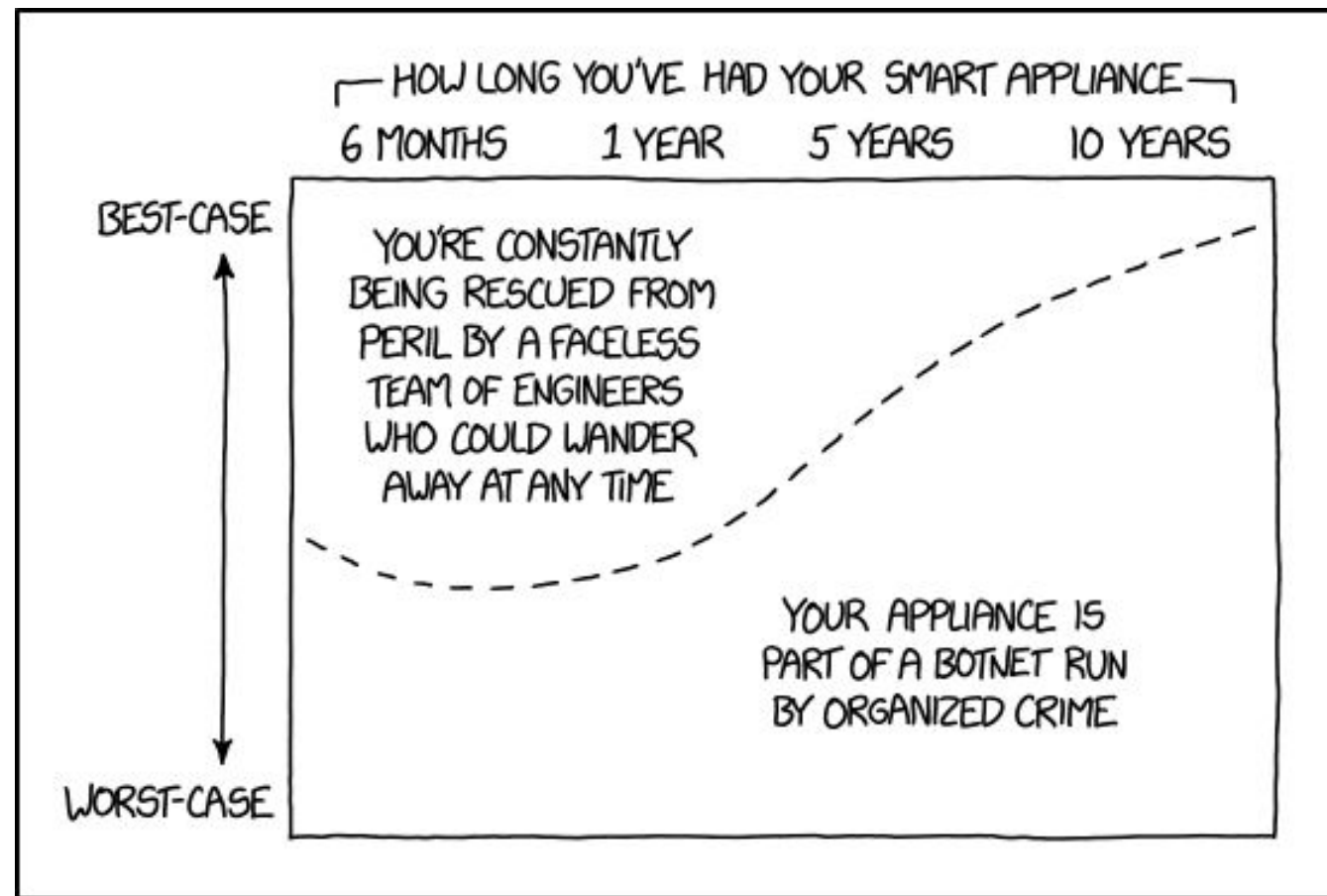


MY PACKAGE MADE IT INTO DEBIAN-MAIN BECAUSE IT LOOKED INNOCUOUS ENOUGH; NO ONE NOTICED "LOCUSTS" IN THE DEPENDENCY LIST.

Source: <https://xkcd.com/797/>

How to communicate? SBOMs, VDRs

- How to disclose security vulnerabilities in my software?
 - lots of legacy software used all around us
- What are the software licenses for all the packages used?
- Machine readable!
- Transparency



Source: <https://xkcd.com/1966/>

And really why?

In the US , Europe, and India, it's the law.

- US: [executive order 14028](#)
 - SBOM for any software business with the government.
 - Europe: CRA ([Cyber Resilience Act](#))
 - Maintainers, Open Source Stewards, Manufacturers
 - <https://github.com/orcwg/cra-hub/blob/main/faq.md>
 - India:
 - [CERT-In SBOM guidelines](#)
 - [CSCRF for SEBI REs](#)
- Often required by companies: using a product/acquiring company
 - Similar legislation/requirements likely in everywhere else

What are the key ingredients?

- standard package identifiers: PURL
- standard license identifiers: SPDX
- standard vulnerability identifiers: CVE, GHSA etc
- standard documents to exchange all this information:
 - SBOMs: CycloneDx, SPDX
 - VEX: CycloneDx VEX, OpenVEX, CSAF VEX

LEAKED LIST OF MAJOR 2018 SECURITY VULNERABILITIES

- CVE-2018-????? APPLE PRODUCTS CRASH WHEN DISPLAYING CERTAIN TELUGU OR BENGALI LETTER COMBINATIONS.
- CVE-2018-????? AN ATTACKER CAN USE A TIMING ATTACK TO EXPLOIT A RACE CONDITION IN GARBAGE COLLECTION TO EXTRACT A LIMITED NUMBER OF BITS FROM THE WIKIPEDIA ARTICLE ON CLAUDE SHANNON.
- CVE-2018-????? AT THE CAFE ON THIRD STREET, THE POST-IT NOTE WITH THE WIFI PASSWORD IS VISIBLE FROM THE SIDEWALK.
- CVE-2018-????? A REMOTE ATTACKER CAN INJECT ARBITRARY TEXT INTO PUBLIC-FACING PAGES VIA THE COMMENTS BOX.
- CVE-2018-????? MYSQL SERVER 5.5.45 SECRETLY RUNS TWO PARALLEL DATABASES FOR PEOPLE WHO SAY "S-Q-L" AND "SEQUEL."

Source: <https://xkcd.com/1957/>

Package-URL

Started in ScanCode to uniquely identify packages.

- pkg:type/namespace/name@version?qualifiers#subpath
 - Specification: <https://github.com/package-url/purl-spec>
- PURL examples:
 - pkg:deb/debian/curl@7.50.3-1?arch=i386&distro=jessie
 - pkg:github/package-url/purl-spec
 - pkg:pypi/django@1.11.1
 - pkg:rpm/fedora/curl@7.50.3-1.fc25
 - pkg:golang/google.golang.org/genproto#googleapis/api/annotations
- Vers: <https://github.com/aboutcode-org/univers/>

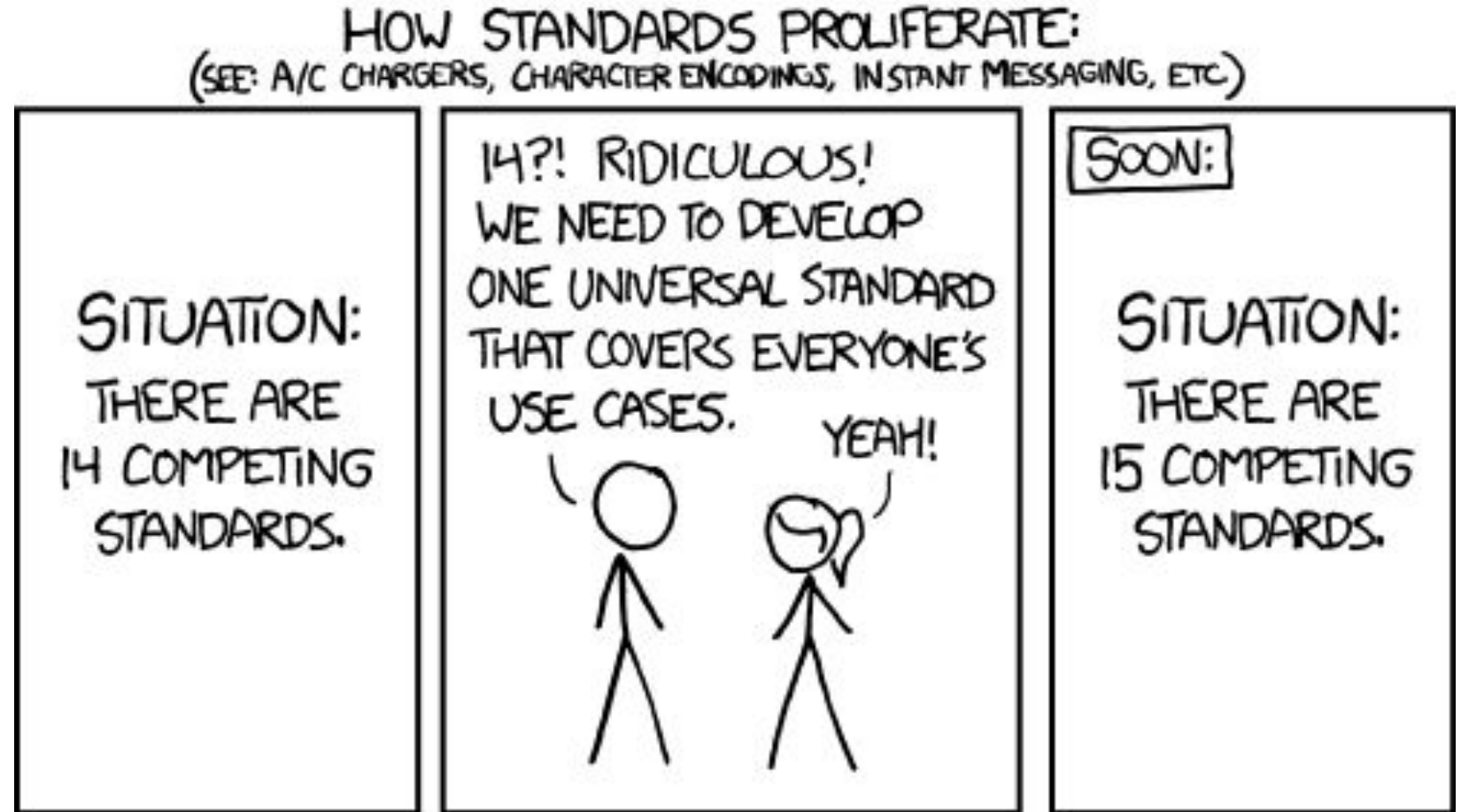
Are these just more standards?

Package-URL (PURL):

An identifier to uniquely identify and download packages

Vers:

Version range specification for package requirements



Source: <https://xkcd.com/927/>

Who is using Package-URL and Vers?

Everyone!

- [GitHub Dependency Submission API](#)
- [OWASP Dependency-Track](#)
- Two major SBOM standards: [CycloneDX](#) and [SPDX](#)
- [OSS Index](#)
- [OSV Schema](#) and [OSV.dev](#) (Google)
- AboutCode tools: [Scancode Toolkit](#) [scancode.io](#) [dejacode](#) [vulnerablecode](#)
- [ORT: OSS Review Toolkit](#), [Osselot](#)
- Anchore, Trivy, Microsoft and GitHub, Chainguard, Snyk
- cve.org, NVD (soon)
- Vers is used at [vulnerablecode](#), Google [OSV](#), AppThreat [vulnerability-db](#)

4 Fs of Open Source

- [The Three Fs of Open Source Puppy Care: Michael Winser](#)
- Complete dependency graph
- Fix
 - Engage with maintainers, open Issues/PRs
- Fork
 - Maintain your version, apply patches
- Forget
 - Use something else!
- Fund!
- relationships with dependencies are important!

Package identification can be hard

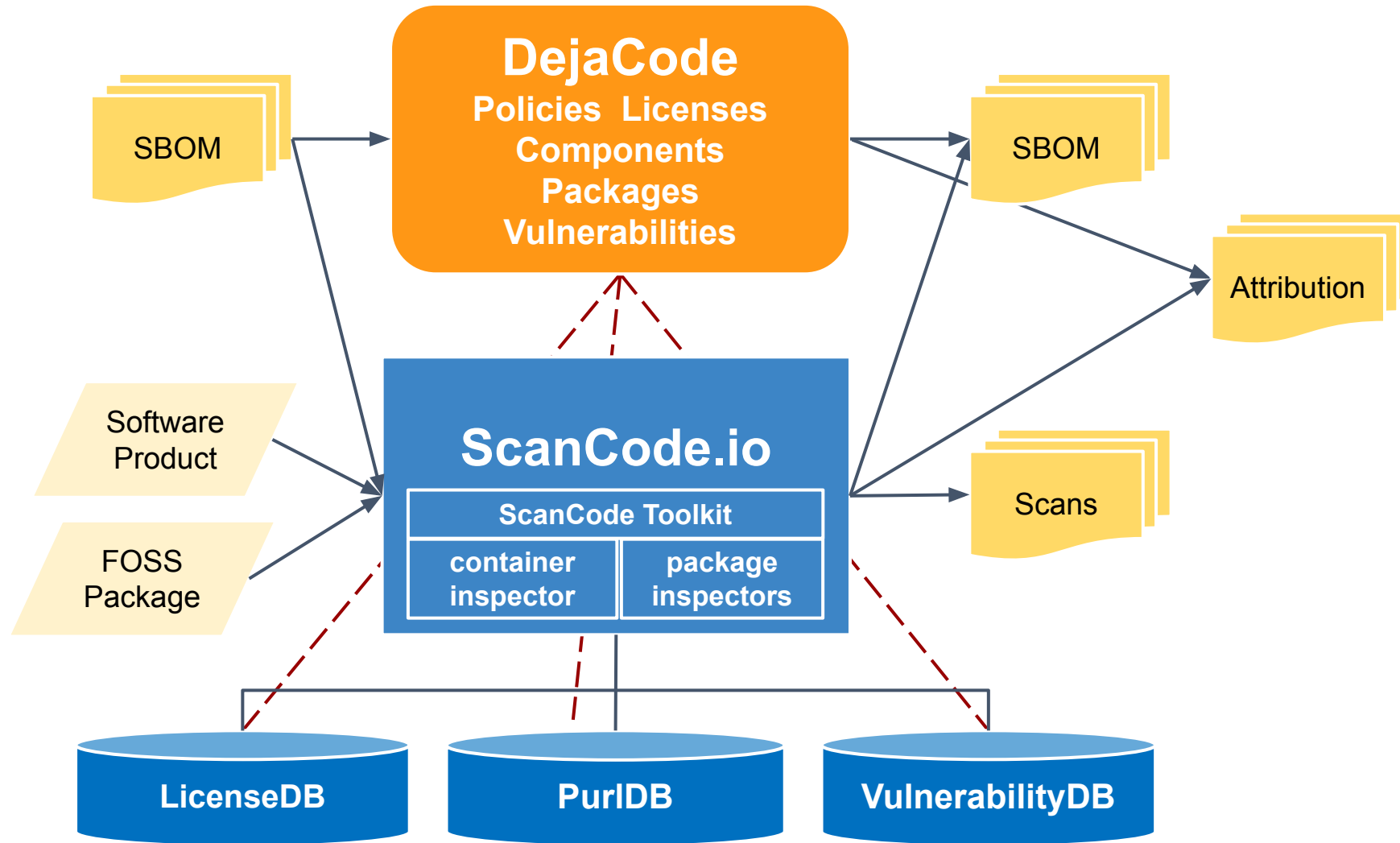
- Binaries have limited metadata
- Most scanners only scan package manifests!
- Code included from different origins
 - vendored (copied partially/fully)
 - distributed with binaries (maven uberjars, jars inside jars)
 - Code matching (MatchCode and PurlDB)
 - Exact archive and file matching
- Finding source repo is not trivial:
 - metadata on source repo missing/incorrect, monorepos
- Customized build systems + metadata formats together
 - build systems have most information, create SBOMs there

- overwhelmed by data!
- Transitive dependencies can be exploited
- dependency graph: how was a specific package version introduced
- might not be applicable
 - vulnerable code not used (reachability) or not deployed (test/docs)
 - only applicable in specific types of deployments/environments/builds
- has this vulnerability been exploited
- score/severity can be good indication, but not absolute
- CI/CD or build dependencies can be problematic too
- update packages regularly
 - if you cannot, you must be more proactive there about issues

What AboutCode is doing differently

- Fully open source
- options: CLI tool, Github action, web app, scans: containers, source/binary etc
- supports and working with package ecosystems
 - to build better metadata, more transparency
 - solve ecosystem wide problems at once
- Don't scan twice
 - Open data on Licensing, Vulnerabilities
 - Reuse Scan Results
 - Only scan the different parts
- Large community
 - OSPOs, security, lawyers, specification groups, developers

The AboutCode stack:



AboutCode: Who is using it?

(based on public data)

Most FOSS Orgs, many commercial and open source SCA providers use our libraries or standards

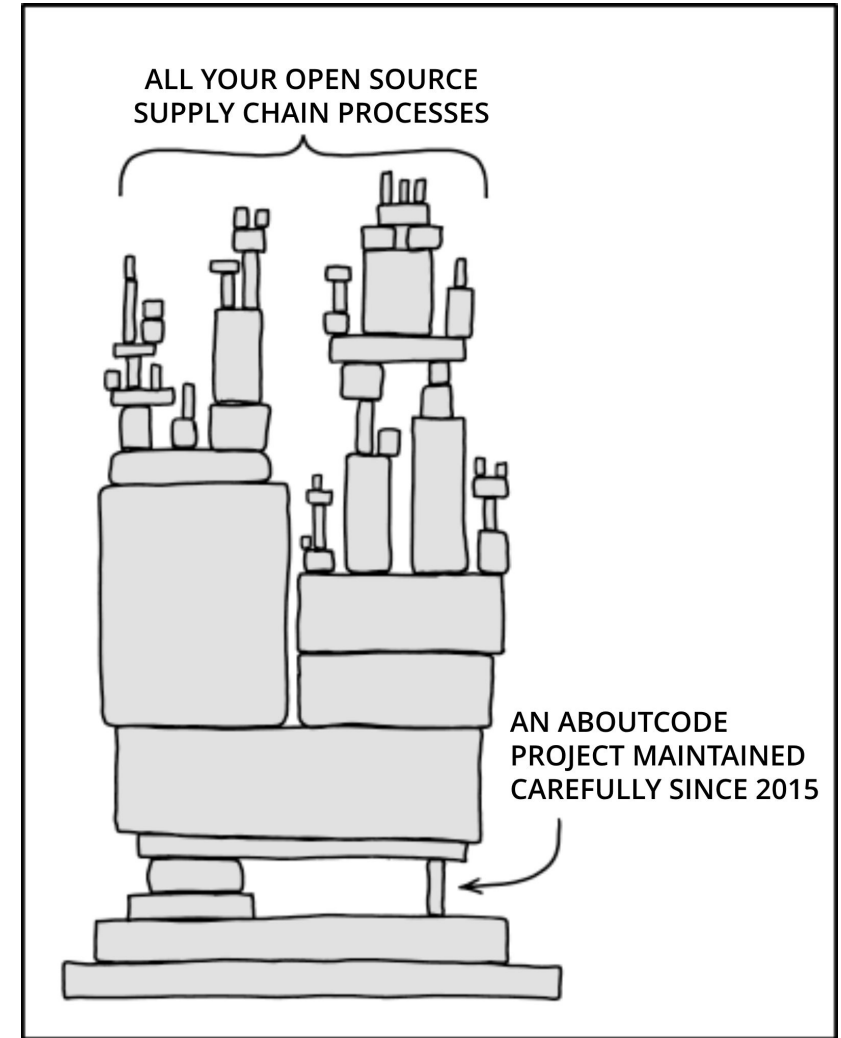
- Most FOSS Foundations
- Five of the top big tech companies
- A leading database company, a leading Linux company
- 2 leading code hosting companies
- European and US government agencies
- All major European car manufacturers and most of their vendors
- Major US chip and microprocessor providers
- All SBOM and VEX standards
- Used to create a database of permissive code to train an open code LLM
 - See <https://huggingface.co/blog/starcoder2>

Other FOSS SCA tools and projects

- ORT: OSS Review Toolkit (Uses ScanCode)
- FOSSology (Uses ScanCode)
- TERN (Uses ScanCode)
- ClearlyDefined (Uses scancode)
- OWASP DependencyTrack
- DepScan (and other AppThreat projects)
- CycloneDx cdxgen
- Anchore: syft, gype

AboutCode also needs your help!

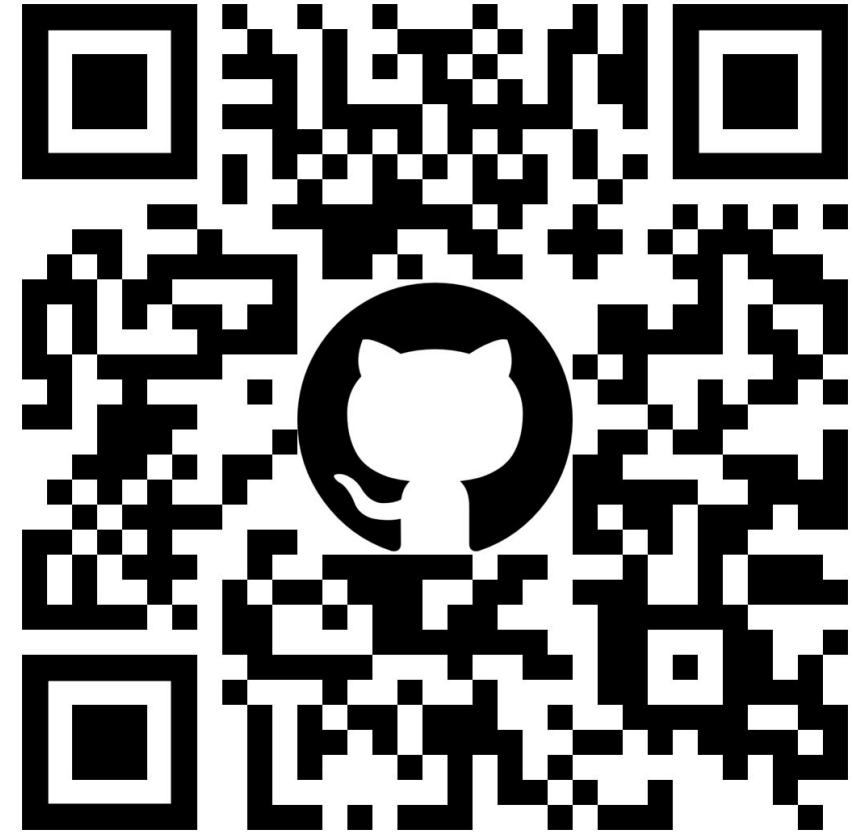
- Contribute to an AboutCode project with code, documentation, use cases, bug reports
 - <https://github.com/aboutcode-org>
- Sponsor AboutCode project maintainers, accelerate development of new features
 - <https://github.com/sponsors/aboutcode-org>
- Buy support, implementation, and advisory services from nexB to pay the maintainers
- Join the community:
 - <https://www.aboutcode.org/>
 - <https://matrix.to/aboutcode-org> discuss



"Dependency" by xkcd, Modified text from original

Questions?

AboutCode



Note: QR Codes are without any tracking

Credits

Special thanks to all the people who made and released these excellent free resources:

- ▷ All the open source software authors that make AboutCode possible
- ▷ [xkcd](#) comics under [cc-by-nc-2.5](#)
- ▷ Presentation template by [SlidesCarnival](#)