FOSS-first mission: Make it easier to reuse open source, safely and efficiently, with open source code and open data



Philippe Ombredanne

pombredanne@aboutcode.org
https://github.com/pombredanne
https://www.linkedin.com/in/philippeombredanne

- Lead maintainer of AboutCode
    - Open source code, data, and standards to automate and secure software supply chains with transparency and confidence
    - https://aboutcode.org
- Creator of PURL (Package-URL) and VERS, co-founder of SPDX and ClearlyDefined, and core contributor to CycloneDX
- CTO and co-founder of nexB
    - Trusted SCA experts since 2007 providing SCA services and AboutCode support
    - https://nexb.com

© AboutCode / CC-BY-SA-4.0 / aboutcode.org



Sampurna Pyne

samk@aboutcode.org
https://github.com/Samk1710
https://www.linkedin.com/in/samk1710

- Open Source Contributor, AboutCode
  - Open source code, data, and standards to automate and secure software supply chains with transparency and confidence
  - https://aboutcode.org
- Final year at HITK, Computer Science Engineering
  - Just wrapped up my GSoC 2026 with AboutCode

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Agenda

- Why vulnerability data is a mess
- Think package-first
- Give packages a proper identity
- Make versions comparable
- How VulnerableCode helps
- Overview of VulnerableCode and related projects

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Software is made of components, like LEGO(™)

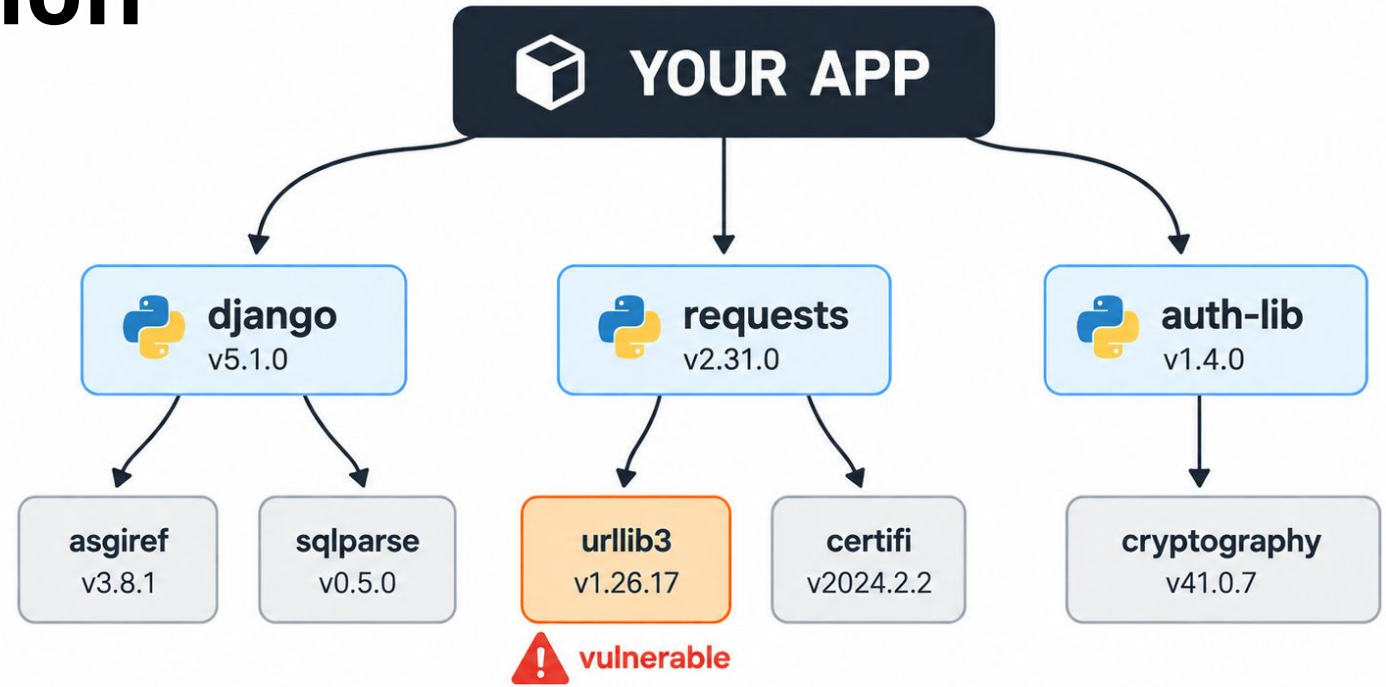
Modern software is assembled, not written from scratch. (Or AI-generated?)



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

1,000's to 10,000's of dependencies fetched in minutes

The dependency explosion



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

The weight of one package



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

The CVE mess

- NVD is DOA, defunded
- CPE does not help
- LLMs find CVEs for breakfast
- Advisories disagree on versions
- When is a CVE exploitable, reachable?

Meanwhile, attackers have walk in the park



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

VulnerableCode's approach: Package first



Look up vulnerabilities?

Look up packages.

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Package-URL (PURL): One package, one identity

Requests

- Which one?
- Which ecosystem?
- Which version?

pkg:pypi/requests@2.31.0

```
pkg:type/namespace/name@version?qualifiers#subpath
```

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Who's using PURL?

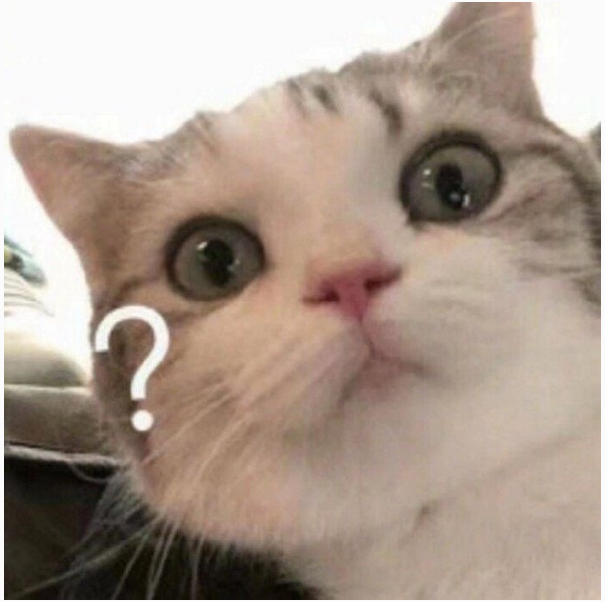
- CycloneDX and SPDX SBOMs
- CSAF and OpenVEX VEX documents
- CVEs! PURL is part of the CVE schema since v5.2.0!
- Most software vulnerability databases such as OSV, Sonatype OSS Index, and VulnerableCode
- Many package repositories, such as Crates.io, Maven.org and Packagist
- All SCA tools that speak SBOMs and vulnerabilities



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

What versions are vulnerable?

Maven:	[2.4.0] , [2.13.0,2.15.0)
nuget:	[1.0.0+A,)
RPM:	2.4.1~rc1, 2.4.1^a-1
npm:	1.x || >=2.5.0 || 5.0.0 - 7.2.3
Gem:	~> 3

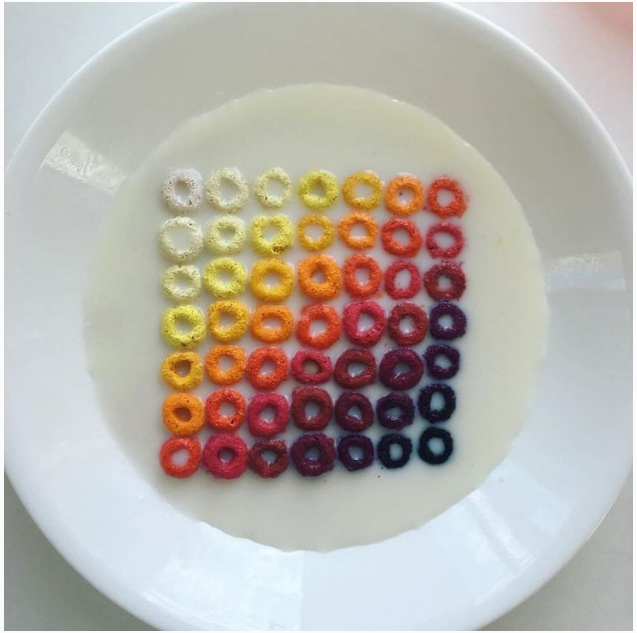


Different ecosystems, different version syntax, different interpretations

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Enter VERS

- VErsion Range Specifier
  - **vers:pypi/>=1.0.0|<2.0.0**
- Implemented via univers



- https://github.com/package-url/vers-spec
- https://github.com/aboutcode-org/univers

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Now we have the answers

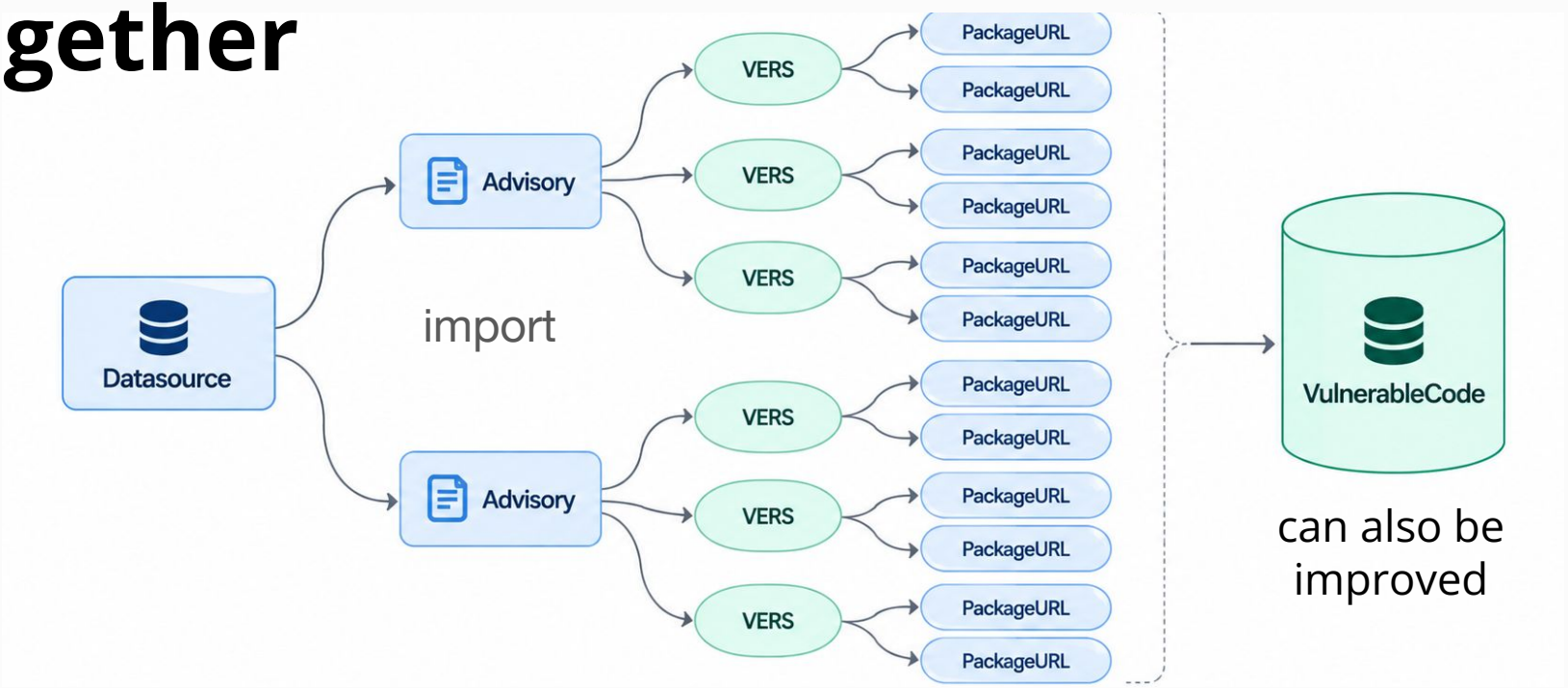
- PURL representation of a Package
    - What package is affected?
- Affected VERS
    - What version range is affected?
- Fixed VERS
    - If a fix exists, which version fixes the given vulnerability?

Just investigate if you are using the affected version range

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Putting it all together

Try: public.vulnerablecode.io



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

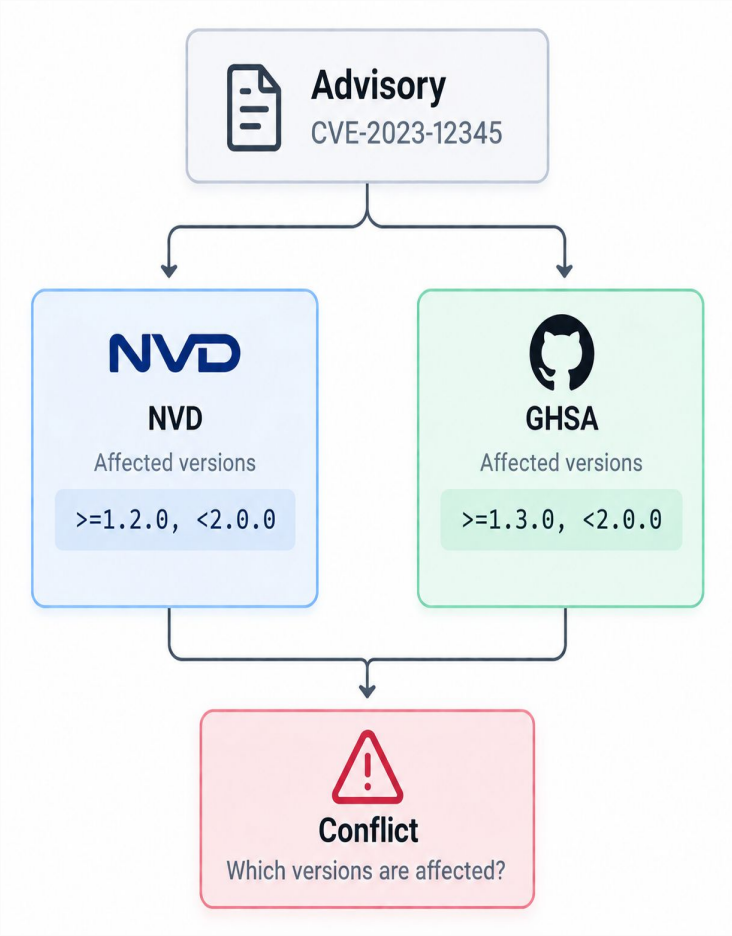
Advisory curation

Q: What happens when upstream sources disagree ?

A: Manual review to improve the quality and accuracy of VulnerableCode data

WIP:

https://public.vulnerablecode.io/advisories/todos/



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Advisory Package Curation

CVE-2026-44248

Progress 1 / 1

Advisory Summaries

github_osv/GHSA-jfg9-48mv-9qgx

Netty MQTT: Resource exhaustion in MqttDecoder
Impact
The MQTT 5 header Properties section is parsed and buffered _before_ any message size limit is applied.

Specifically, in `MqttDecoder`, the `decodeVariableHeader()` method is called before the `bytesRemainingBeforeVariableHeader > maxBytesInMessage` check. The `decodeVariableHeader()` can call other methods which will call `decodeProperties()`. Effectively, Netty does not apply any limits to the size of the properties being decoded.

Additionally, because `MqttDecoder` extends `ReplayingDecoder`, Netty will repeatedly re-parse the enormous Properties sections and buffer the bytes in memory, until the entire thing parses to completion.

This can cause high resource usage in both CPU and memory.

pkg:maven/io.netty/netty-codec-mqtt

Advisories disagree on whether the following version(s) are affected: 4.2.0-Alpha.1.
Advisories disagree on whether the following version(s) contain the fix: 4.1.133.Final, 4.1.133, 4.2.13, 4.2.13.Final.

Package Versions	Curation	github_osv/GHSA-jfg9-48mv-9qgx	gitlab/maven/io.netty/netty-codec-mqtt/CVE-2026-44248
	Reset	Pick This	Pick This
Show Version Ranges			
Show Consensus Range (148 versions)			
4.1.133.Final	Select value	FIXED	UNAFFECTED
4.1.133	Select value	UNAFFECTED	FIXED
4.2.0-Alpha.1	Select value	UNAFFECTED	AFFECTED
Show Consensus Range (23 versions)			
4.2.13	Select value	UNAFFECTED	FIXED
4.2.13.Final	Select value	FIXED	UNAFFECTED

Previous Item

Submit

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

VulnerableCode Insights

- Improve VulnerableCode data quality
- Find out our drawbacks
- Make Vulnerablecode more transparent to its users

- WIP: https://vulnerablecode.readthedocs.io/en/latest/advisory-todos.html

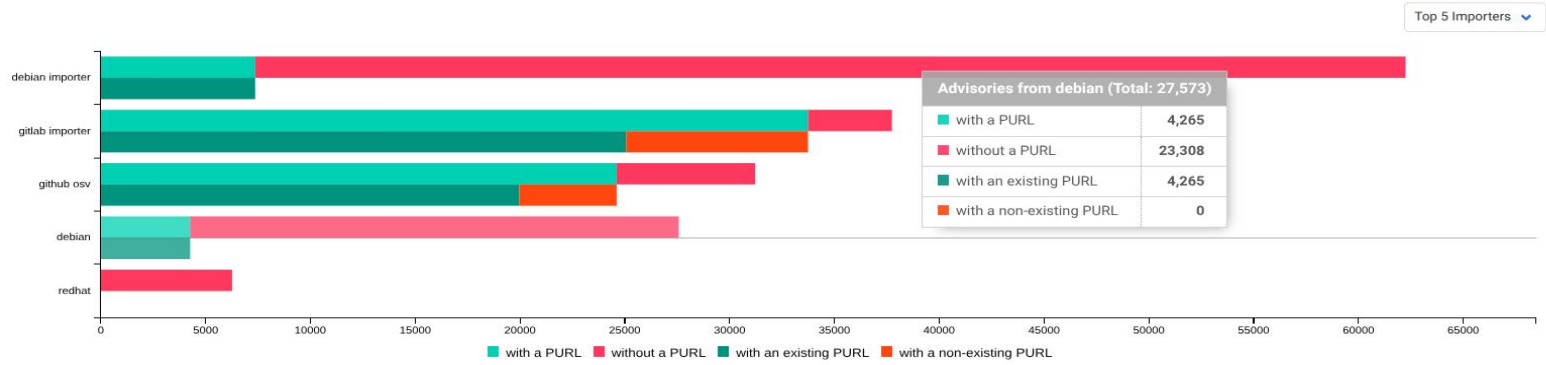
© AboutCode / CC-BY-SA-4.0 / aboutcode.org

INSIGHTS DASHBOARD

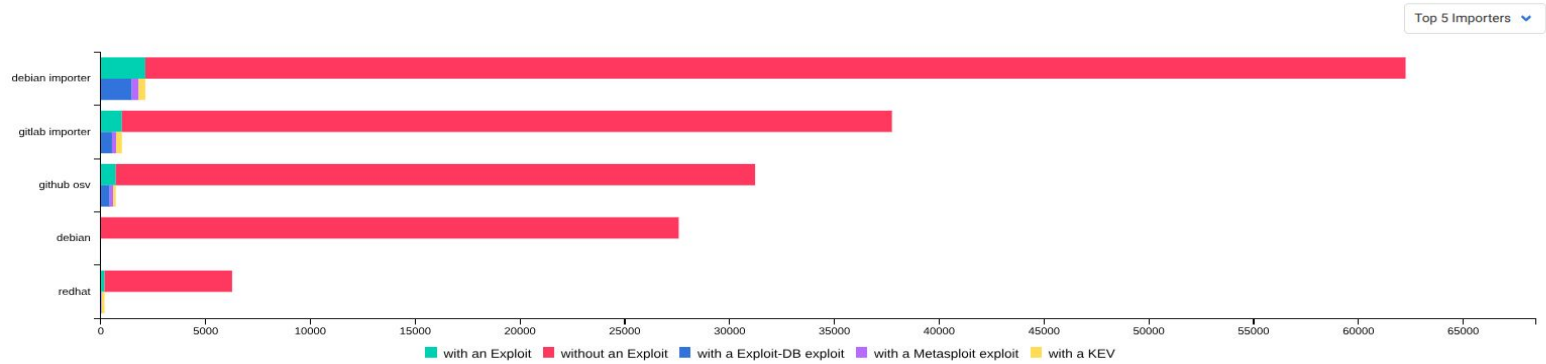
- Overview
- Package Analytics
- Severity Analytics
- Importer Analytics
- Data Quality Analytics

Last Snapshot At:
2026-08-11 22:33 UTC

PURL Coverage across Importers



Exploit Coverage across Importers



© AboutCode / CC-BY-SA-4.0 / aboutcode.org

How can you help?

- Use it: this is FOSS, free and open code AND open data!
- Help make it better
    - Better code, in particular spot data quality issues
    - Better data, in particular fix data quality issues
- Help us fix the issues once upstream, ecosystem-wide:
    - Apache Maven, Rust and nixpkgs campaigns in progress
    - Working with ASF (apache.org) and Linux (kernel.org)
- Make this known in your FOSS community

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Join the community and contribute!

AboutCode

GitHub: https://github.com/aboutcode-org

VulnerableCode:
https://github.com/aboutcode-org/vulnerablecode

Slack: https://join.slack.com/t/aboutcode-org/shared_invite/zt-1paqwxccw-luafuiAvYJFkTqGaZsC1og

Community calls:
Mondays at 15:00 UTC at
https://meet.jit.si/AboutCode

TC 54 PURL + VERS

https://github.com/package-url/purl-spec

https://archive.fosdem.org/2018/schedule/event/purl/

Join #purl channel on AboutCode slack for latest updates

Community calls:
Biweekly on Wednesdays at 16:00 UTC at
https://meet.google.com/ryq-aimn-ghd

© AboutCode / CC-BY-SA-4.0 / aboutcode.org

Come talk to us at the conference!

Any questions?

https://aboutcode.org



Come talk to us at the conference!

Link to slides

https://docs.google.com/presentation/d/1MDuU7G7pQY_QXCW0oV8M5H0PUsXoORk9



© AboutCode / CC-BY-SA-4.0 / aboutcode.org