AboutCode

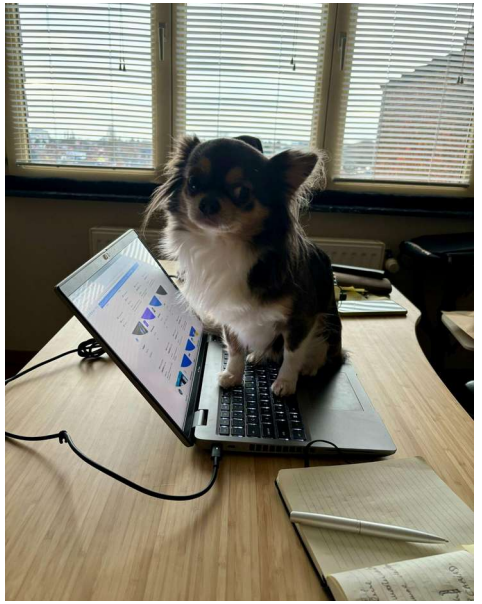
NVDR
Non Vulnerable
Dependency Resolution

Agenda

- About me, about AboutCode
- Keeping the barbarians out
- The problem with dependencies
- Dependency resolution?
- Vulnerable version ranges?
- Package URL aka. PURL
- VERS: version range spec
- Python dependencies resolver
- Aggregated and correlated Vulnerabilities DB
- NVDR: keep the barbarians at the gate!
- Questions

About me

- On a mission to enable easier and safer to reuse FOSS code with best-in-class open source Software Composition Analysis (SCA) tools, data, and standards for open source discovery, license & security compliance
- Lead maintainer of AboutCode projects (ScanCode, DejaCode, VulnerableCode, Package URL, and others)
- CTO and co-founder of nexB, Inc.
  - pombredanne@nexb.com
  - GitHub: https://github.com/pombredanne
  - LinkedIn: https://www.linkedin.com/in/philippeombredanne
  - Assisted by a CTA (Canine Technical Advisor)



About AboutCode

AboutCode

- **AboutCode's FOSS-first mission: FOSS for FOSS**
  - **Open source tools AND open knowledge base (AboutCode stack)**
  - Simple and practical standards (**Package-URL / PURL** https://github.com/package-url)
  - Applications for Legal & Business users (**DejaCode**) with APIs for everything
  - Co-founders of **SPDX**: https://spdx.org
  - Contributors to **CycloneDX**: https://cyclonedx.org
  - Co-founders of **ClearlyDefined**: https://clearlydefined.io
  - Anchors for a community of SCA tools user and developers
  - Supported by contributors, nexB and others generous sponsors and supporters!

AboutCode

keeping the ~~barbarians~~ vikings down at the gate

The problem with dependencies

AboutCode

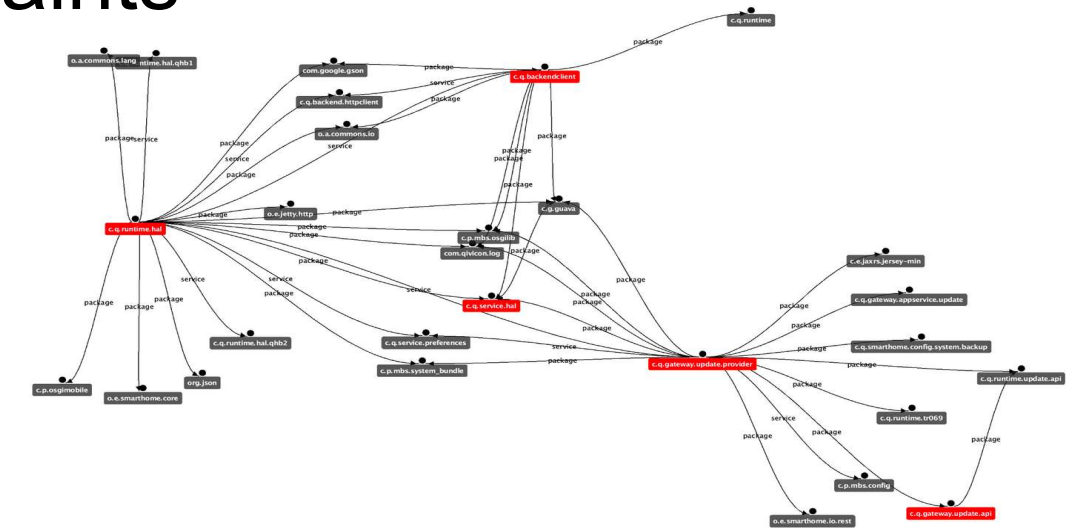
Package dependency management is the discipline to pick and install random packages and pretend things will be OK

- **License?**
- **Security?**
- **But also quality, sustainability and more!**

Dependency resolution?

Inputs are "requirement" or "constraints"

Install log4j from maven, v2.0.0 or higher



- *Many of these form a complex graph*
- *Package managers "resolve" these to concrete versions that satisfies all the constraints (dnf, apt, npm, pip, mvn, bundler, etc.)*
- *They fetch **version lists**, use sat solvers to **find solutions***
- *This is great .... but also the source of so many security issues*
  - *What's not to like when you install random package versions?*
  - *Or the latest version? xz anyone?*

Vulnerable version ranges?

OpenSSL is vulnerable to CVE-2023-3817

Higher than 1.0.1, from 3.0.0 to 3.0.9, but excluding 3.0.0-FIPS

- *Many such constraints may exist*
- *Tools try to determine if a version falls is a range*
- *WTF sidebar :* ***many VDB do not agree on vulnerable ranges ?##&?!***
- *See VulnerableCode's* ***VulnTotal***
  https://github.com/nexB/vulnerablecode/pull/801
  https://github.com/nexB/vulnerablecode/blob/main/vulntotal

Package URL aka. PURL

AboutCode

A mostly universal package identifier

pkg:npm/file@1.9.1

pkg:pypi/django@1.11.1

- *Designed to be obvious and **decentralized***
- *Adopted by tools and specs: CycloneDX, SPDX, CSAF, Google/OSSF OSV, Sonatype OSS Index, OpenVEX, Tern, ORT, Anchore, DependencyTrack, DependencyCheck, Appthreat, Microsoft SBOM tool, and most open (and proprietary such as Snyk) SCA and Infosec/Appsec tools. Started in VulnerableCode and ScanCode. Yeah!*
- *The GLUE between many software supply chain security tools*
- *Project: https://github.com/package-url*
- ***Recent proposal to add purl to NVD:***

https://owasp.org/blog/2022/09/13/sbom-forum-recommends-improvements-to-nvd.html

"Component verification and vulnerability reporting are supported by some SBOM data formats today. Globally unique identifiers is a work in process supported by the leading data formats for package URLs (PURLs)."

https://linuxfoundation.org/wp-content/uploads/LFResearch_SBOM_Report_final.pdf
Software Bill of Materials (SBOM) and Cybersecurity Readiness

Stephen Hendrick, VP Research, The Linux Foundation

VERS: version range spec

AboutCode

- A unified notation for any and all version ranges
  For dependency ranges AND **vulnerable ranges**

  vers:npm/1.2.3|>=2.0.0|<5.0.0

- *Defines how to sort and compare two versions*
- *For all package ecosystems*
- *Part of the PURL project, same overall shape*
  https://github.com/package-url/purl-spec/pull/139

Python-inspector: Python resolver

AboutCode

- **Lightweight Python dependency resolver**
  PURL ***and*** *VERS* ***as an input***

More inspectors in the works for all ecosystems

- *Simulate the resolution of a dependency tree on demand*
- *... to explore the graph and understand transitive dependencies*
- *Do What-if? scenarios by tuning the requirements and constraints and resolve a new graph*
- *Useful before trying to update and other analysis purpose*

Aggregated & correlated Vulnerabilities DB

AboutCode

VulnerableCode is aggregated known Vulnerability DB, BUT packages first

Keyed by PURL
Vulnerable ranges stored as VERS

- *Collect and aggregate vulnerability data from many public sources*
  - *GitHub, Linux Distros, NVD, OSV, Package managers and many more*
  - *Focus is on upstream project feeds (the source of the source)*
- *Discover relations (and inconsistencies) between vulnerabilities and packages from mining the graph:* ***VulnTotal***

- *Available at: https://public.vulnerablecode.io/*
- *Code and data dumps at https://nexb.com/vulnerablecode/*

NVDR: keep the barbarians at the gate!

AboutCode

NVDR: keep the barbarians at the gate!

AboutCode

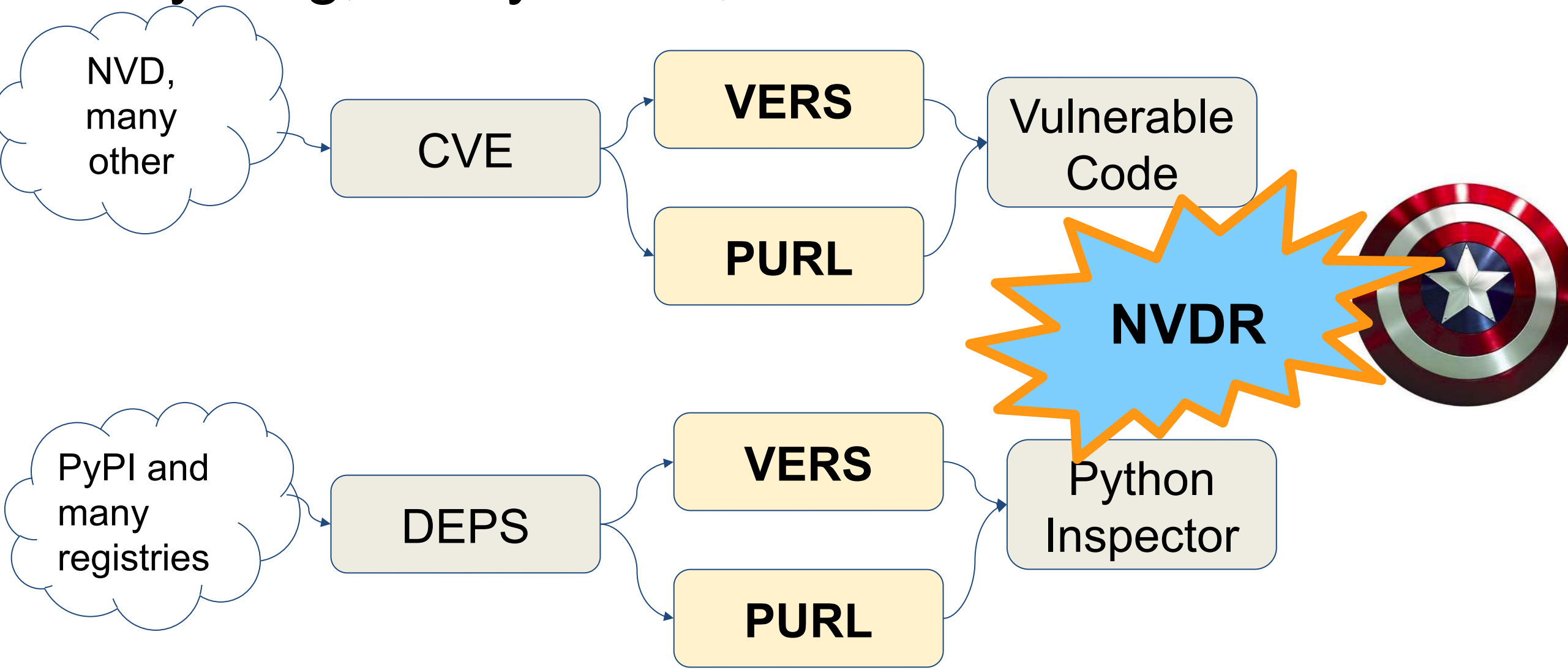
- If you could blend
  - **Functional dependency constraints**
  - **Known vulnerable ranges**
- And inject these in a package dependency resolver then you get

Non Vulnerable Dependency Resolution!

- Working PoC implemented in **python-inspector** tool and paper at https://www.tdcommons.org/dpubs_series/5224/

Everything, everywhere, all at once

AboutCode



What to make of this... and beyond

AboutCode

- What if scenarios?
  - Before selecting a package
  - For remediation
- Vulnerable is a spectrum not just a state
  - Include exploitability, reachability
  - Other criteria.... End-of-life? Out of maintenance? License?
- Extend Non-vulnerable dependency resolution
  - Beyond Python - add Java and JS
  - Create generic dependency resolvers
- Federated, decentralized database of FOSS package data and vulns

AboutCode

Questions?

AboutCode

Bonus

AboutCode: Who is using it?

Many organizations, and most SCA providers use AboutCode tools, libraries or standards:

- Most free software and open source foundations
- Five of the top big tech companies
- A leading database company and a leading Linux company
- European and US government agencies
- All major European car manufacturers and most of their vendors
- Major US chip and microprocessor providers
- Four leading European industrial companies
- All SBOM and VEX standards
- All open source SCA and SBOM tools
- Most proprietary SCA, SBOM or code hosting tools

SCA Tools

Management Apps

Open Knowledge Base

The AboutCode stack: SCA Tools

- ScanCode, industry-leading scanning engine
  - Scripted scan pipelines for large codebase, containers, VMs, and deployed binary-to-source analysis
- Code matching integrated with the open knowledge base
- Many other libraries and tools
  - ABOUT files for curations/corrections stored in the codebase
  - Inspectors for packages and dependencies
  - univers: parse and compare package versions and version ranges
  - license-expression: parse and compare License expressions
- package-url (PURL) adopted by CycloneDX, CSAF, SPDX and the whole SCA ecosystem

SCA Tools

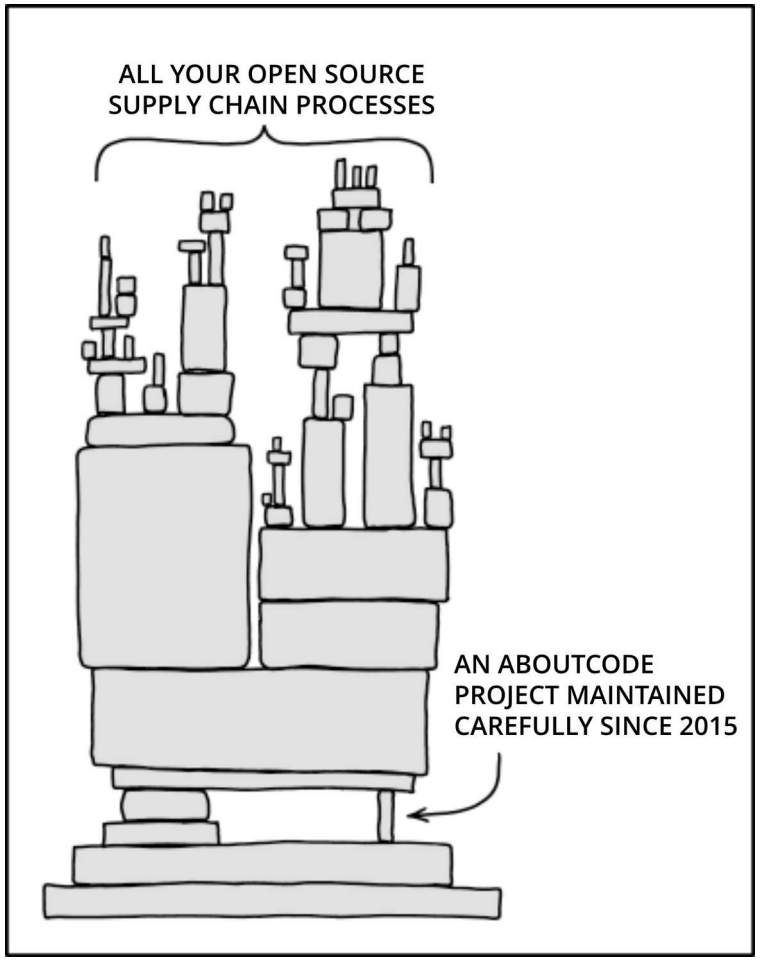
Why AboutCode?

AboutCode

- Free and open source software AND free and open data
  - FOSS for FOSS
  - Open knowledgebase with open data for licenses, packages and vulnerabilities
- Modular and integrated best-in-class SCA tools for developers
  - Tackling the harder code analysis problems so you do not have to
  - PURL-based for easier integration in/out
- Bespoke pipelines enable true end-to-end automation
  - Working towards management by exception to focus on the complex cases of origin and license
  - Decentralized analysis, close to the developers
- Management web app for centralized policies, curations and compliance workflows and data
  - Supports engineering, business and legal stakeholders with features tailored for each using common/shared information

AboutCode also needs your help!

- Contribute to an AboutCode project with code, documentation, use cases, bug reports
  - https://github.com/nexB
- Join the community:
  - https://www.aboutcode.org/
  - https://gitter.im/aboutcode-org/discuss
- Sponsor AboutCode project maintainers
  - Accelerate development of new features and fund contributors
  - Buy support, implementation, retainers and advisory services to pay the maintainers



"Dependency" by xkcd, used under CC BY-NC 2.5 / Modified text from original