AboutCode

How to use open source tools and open data to automate and secure your software supply chains

Philippe Ombredanne,
Lead maintainer of AboutCode

Agenda

1. Introductions
   - Philippe and AboutCode
   - Open source and compliance
2. Overview of FOSS tools for cybersecurity and compliance
   - Discovering and identifying third-party code
   - Discovering, triaging, and managing vulnerabilities
   - Standards for tool interoperability
   - Open license, package, and vulnerability databases
   - Automating compliance processes
3. AboutCode stack for software supply chain automation
   - Massive speed and accuracy improvements in ScanCode
   - Code and snippet matching backed by actual open data in MatchCode
   - PURLs are everywhere
   - Sharing all the scans in PurlDB
   - Vulnerability management for VEX and CRA compliance with VulnerableCode
4. So, what's next?
   - Get involved
   - Future of AboutCode

About Philippe and AboutCode

- FOSS-first mission: Make it easier to reuse open source, safely and efficiently, with open source code and open data
  - Creator of Package-URL (PURL), co-founder of SPDX and ClearlyDefined, contributor to CycloneDX, and trusted SCA expert since 2007
    - pombredanne@aboutcode.org
    - https://github.com/pombredanne
    - https://www.linkedin.com/in/philippeombredanne
- Lead maintainer of AboutCode: https://aboutcode.org
  - Open source tools and open knowledge base: ScanCode, VulnerableCode
  - Simple and practical standards: PURL
  - Apps for legal, security, and business users with APIs for everything: DejaCode

SCA = Software Composition Analysis

AboutCode

- SCA is essential to know what components are actually in the software
  - First step for software supply chain management
  - Includes processes to identify components, their licensing, and known vulnerabilities (like the AboutCode stack), and evaluate the quality of a software unit (like the CHAOSS project)
    - Read "SCA the FOSS Way": https://www.nexb.com/software-composition-analysis/
- SCA needs to be a core competency for any software development organization, especially with CRA and other regulations
  - Embed in the software development workflow from design through release,
    - Similar to manufacturing
  - The choice of SCA tools will depend on your platform, stack and product

Compliance = critical

- FOSS compliance is licensing AND security
  - Identifying licenses and license compliance still a problem at scale
  - Very difficult to track all open source and third-party components - including dependencies, licensing, and compliance obligations - with the high volume and rate of change
- Always important, now urgent with CRA and other regulations and more cybersecurity attacks
  - Disproportionate effect on SMEs, nonprofits and other organizations with same compliance needs as big companies and governments but without the resources
    - No dedicated security teams (usually) or budgets for expensive tooling and processes
- Must automate compliance processes (when possible) for efficiency
  - Imperative to balance compliance efforts and shipping products
  - Critical to ensure software supply chain security and integrity

Proprietary != scalable and effective

AboutCode

- Commercial tools for security are cost-prohibitive and not efficient
  - Increasing expensive with surge of interest in SBOMs and developer-based pricing
    - Gold rush from commercial vendors to sell anything related to CRA, SBOM, compliance, vulnerability, cybersecurity
  - Not efficient for compliance tooling and processes
    - Cost of scan curation is prohibitive with high false positive rates and poor origin and license detection accuracy
- Proprietary data for FOSS is wrong
  - Most current data about FOSS packages and vulnerabilities is proprietary
    - Vendors may offer some free or open source tools but must pay for access to their data
  - Vulnerability and security data about open source must be free and open
    - Security is a fundamental right
    - Safe open source software is a public good

Overview of FOSS tools for cybersecurity and compliance

AboutCode

Modern software requires FOSS for FOSS tools and open data

LEGACY

- Vulnerability-centric
- Proprietary data
- Siloed
- Vendor-driven
- Centralized
- Security team
- Reactive

FUTURE = Open source

- Package-centric
- Open data
- Interoperable
- Community-driven
- Decentralized, federated
- Security team + developers
- Proactive

Identify third-party code

AboutCode

1. Scan code
    - Based on package manifests, and other clues present locally in the code
2. Match code
    - Based on content and fuzzy fingerprints matched to an external open knowledge base
    - PURL-based
3. Identify license, copyright, other origin clues
    - Including binary analysis and build tracing

Many tools, but still "unsolved"

- Recent study to compare commercial and FOSS SCA tools for containers was ... sad
    - More on this later
- Email pombredanne@aboutcode.org for the sanitized report

AboutCode

FOSS tools to identify third-party code

FOSS Tool	Scanning	Matching	Other origin clues
Google OSV			
SCANOSS		(source only)	
ORT			
Syft	(mostly containers)		
Trivy	(mostly containers)		
BANG			(including binary)
ScanCode			(including binary)
MatchCode		(including binary)	
Many other tools			

Triage vulnerabilities

AboutCode

1. Lookup (open) vulnerability databases
2. Rank severity and exploitability
3. PURL-based
4. VEX export

AboutCode

Package-URL (PURL) enables tool interoperability

- Critical for managing software supply chain security and integrity
  - And imperative for actionable SBOMs!
- URL string to identify and locate software packages across various ecosystems and repositories, adopted by:
  - All SBOM and VEX standards including CycloneDX, SPDX, CSAF, and OpenVEX
  - All open source SCA and SBOM tools and most proprietary SCA, SBOM, and code host tools
  - Most open vulnerability databases (part of CVE specification v5.1)
  - Recommended by US CISA, German BSi and the CERT-India
    - Read more: https://nexb.com/purl-universal-software-package-identification/
- In the process of Ecma standardization: https://tc54.org/purl/

FOSS tools to triage vulnerabilities

AboutCode

FOSS Tool	Lookup vulnerability databases	Rank severity and exploitability	PURL-based	VEX export
DependencyTrack				
DefectDojo				
DejaCode CRAVEX				

Need more (and better) tools with more capabilities, especially for mitigating and managing vulnerabilities

And we don't need more vulnerability databases

- We need just one good open package-based vulnerability database
  - Federated with projects submitting vulnerabilities
  - Keyed by PURL to ensure tool interoperability
- Not dependent on US NVD for severities

Peter J. Yost (© 2021) "The One Ring made from scratch in Blender 3D software." (CC BY-SA 4.0). https://upload.wikimedia.org/wikipedia/commons/d/d4/One_Ring_Blender_Render.png

AboutCode

Open vulnerability databases

Open vulnerability database	Open source code	Open infrastructure	PURL-based	Updated data	Scope
US NVD				(delayed)	System + app package + prop
Google OSV			(mostly)		System + app package
GitHub Advisories			(compatible)		App package
GitLab Advisories			(mostly)	(1 month delay)	App package
VulnerableCode					System + app package
Linux distro advisories			(compatible)		System
Ecosystem advisories			(compatible)		App package

Manage compliance

AboutCode

1. Aggregate SBOMs
2. Export VEX and SBOMs
3. PURL-based
4. Dependency updates and remediation

FOSS tools to manage compliance

AboutCode

FOSS Tool	Aggregate SBOMs	Export VEX, SBOMs	PURL-based	Dependency updates and remediation
AboutCode stack (WIP)				
OCCTET (WIP)				
DependencyTrack				
RenovateBot				
DependaBot				

Need more (and better) tools with more capabilities, especially for compliance automation

FOSS tools still have work to do

AboutCode

- The state of SCA tooling accuracy is not great
  - Recent large scale comparison of both FOSS and commercial container scanners using SBOMs to compare scans of the same container images
    - Commercial tools made up packages and PURLs
    - Several tools created invalid SBOMs
    - Most only looking at package manifests and DB
    - Beyond package origin, quality of report licenses is bad and misleading
    - In most cases, this is a grep on the declared license of package manifests
- We can do better!
  - FOSS tools performed better than commercial
  - Still many functionality missing to complete end-to-end automation of compliance processes

Feedback from FOSDEM Fringe workshop

AboutCode

- Adaptable SBOM tooling
  - Robust validation techniques
  - Visualization capabilities
  - Digital signatures
- Interoperability across tools and ecosystems
  - And standardization
- Collaborative knowledge sharing
  - Developing curated databases
  - Federated system for vulnerabilities
  - Establishing benchmark repositories
  - Continuous improvement of standards and practices
- Support open source development

Photo by Salve J. Nilsen at FOSDEM 2025 Fringe: FOSS license and security compliance tools workshop in Brussels, Belgium on January 31, 2025, used under CC BY-SA 4.0

AboutCode

The AboutCode Stack

Essentials for compliance automation

AboutCode

1) Identify third-party code

- Scan code
  - Based on package manifests, and other clues present locally in the code
- Match code
  - Based on content and fuzzy fingerprints matched to an external open knowledge base
  - PURL-based
- Identify license, copyright, origin clues
  - Including binary analysis and build tracing

2) Triage vulnerabilities

- Lookup (open) vulnerability databases
- Rank severity and exploitability
- Remediate and mitigate vulnerabilities
- PURL-based

3) Manage compliance

- Aggregate SBOMs
- Export VEX and SBOMs
- PURL-based
- Dependency updates and remediation

And the AboutCode stack

1) Identify third-party code with **ScanCode** and **MatchCode**
   - Scan code
     - Based on package manifests, and other clues present locally in the code
   - Match code
     - Based on content and fuzzy fingerprints matched to an external open knowledge base
     - PURL-based
   - Identify license, copyright, origin clues
     - Including binary analysis and build tracing

2) Triage vulnerabilities with **VulnerableCode**
   - Lookup (open) vulnerability databases
   - Rank severity and exploitability
   - Remediate and mitigate vulnerabilities
   - PURL-based

3) Manage compliance with **DejaCode**
   - Aggregate SBOMs
   - Export VEX and SBOMs
   - PURL-based
   - Dependency updates and remediation

AboutCode

SCA Tools

Management Apps

Open Knowledge Base

AboutCode

SCA Tools

ScanCode, MatchCode

- Scan
- Match
- Analysis pipelines
- Binary analysis
- Dependency analysis

Management Apps

DejaCode

- Policies
- Curations
- Software inventory
- Workflows
- SBOMs
- Custom reports

Open Knowledge Base

- Licenses
- Packages
- Vulnerabilities

Who is using the AboutCode stack?

Many organizations and most SCA providers use AboutCode tools, libraries, and 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, and most proprietary SCA, SBOM or code hosting tools
- Responsible AI systems and LLMs, including the BigCode project from Hugging Face and ServiceNow Research based on the Software Heritage Archive

The AboutCode stack: ScanCode

- **ScanCode Toolkit** is the industry-leading code scanner for software component, package, and dependency identification, and license detection
  - Multiple techniques to identify third-party code, determine license and origin
  - Accuracy is paramount
    - https://github.com/aboutcode-org/scancode-toolkit
- **ScanCode.io** is a web-based scanning server to automate SCA
  - Powered by ScanCode Toolkit
  - Specialized pipelines for customized analysis, including container and VM scanning, and deployment analysis using binary analysis
  - Integrated enrichment of the open knowledge base
    - https://github.com/aboutcode-org/scancode.io/

SCA Tools

Faster scans

- **FastScan** improves the performance for ScanCode Toolkit and ScanCode.io to detect origin, license, and known vulnerabilities
  - Establish and publish a baseline benchmark and profile hotspots performance
  - Improve the installation performance of ScanCode Toolkit and reduce the time-to-scan from a download packaging ScanCode toolkit as a standalone executable
- The project will massively speed up ScanCode scans
  - EU-funded through the NGI0 Core fund with financial support from the European Commission's Next Generation Internet programme
    - https://nlnet.nl/project/FastScan/

And more accurate scans

Software Heritage + AboutCode

- **Massive FOSS Scan** is a collaborative project with Software Heritage to run a massive license scan on the whole Software Heritage archive
  - Over 20 billion unique source code files from more than 327 million projects
  - Also includes the PurlDB index of all major package registries and Linux distributions
  - And working with HuggingFace to train LLMs on code scanned by ScanCode for responsible AI
- The project will generate a massive open database to improve FOSS code matching and discovery at an unprecedented scale
  - Commons reference DB for faster (future) scanning and matching with accurate license data
  - Collection of fingerprints to enable approximate code matching at scale
  - EU-funded through the NGI0 Commons fund with financial support from the European Commission's Next Generation Internet programme
    - https://nlnet.nl/project/MassiveFOSSscan

NGI ZERO CORE

nlnet FOUNDATION

European Commission

The AboutCode stack: More SCA tools

- **MatchCode** is a new code matching server
  - Smart matching approach in multiple steps with whole tree, exact file, approximate tree and file matching
    - Planned: Snippet matching with AI-Generated Code Search
    - https://github.com/aboutcode-org/matchcode
- **TraceCode** traces software components, packages and files between development and deployment codebases
  - Reconstruct the build graph and then scan and match the source subset
    - https://github.com/aboutcode-org/tracecode-toolkit
- **Inspectors** are tech-specific tools and dependency resolvers
  - nuget-inspector, python-inspector: resolve dependencies
  - container-inspector: analysis tool for Docker and other images
    - More libraries at https://github.com/aboutcode-org

SCA Tools

Matching, backed by (actually) open data

- **AI-Generated Code Search** offers a new, faster, and entirely open approach for approximate code matching to efficiently and effectively return accurate results
  - Works for whole file code trees and snippet matching, and detecting AI-generated code
  - Fingerprint-based matching helps scale the index but also scale the query as a whole codebase (GBs) is the query
    - Tunable fingerprint at query time for precision and recall
  - Approximate, fuzzy fingerprinting enables matching code that was never indexed
  - EU-funded through the NGI Search fund with financial support from the European Commission's Next Generation Internet programme
    - https://github.com/aboutcode-org/ai-gen-code-search

Open Knowledge Base: Standards

- **Package-URL (PURL)** is a URL string to identify and locate packages across various ecosystems and repositories
  - Critical for managing software supply chain security and integrity because PURL enables SCA and SBOM tool interoperability, adopted by:
    - All SBOM and VEX standards including CycloneDX, SPDX, CSAF, and OpenVEX
    - All open source SCA and SBOM tools, most proprietary SCA, SBOM, and code host tools, and most open vulnerability databases
    - Recommended by US CISA, German BSi and the CERT-India
  - In the process of Ecma standardization: https://tc54.org/purl/
    - https://github.com/package-url
- **Version Range Specifier (VERS)** stores and compares package versions
  - https://github.com/package-url/purl-spec/blob/master/VERSION-RANGE-SPEC.rst

Open Knowledge Base

Actionable SBOMs with validated PURLs

- **PURLValidator** validates the PURL syntax against any known PURLs by exposing PurlDB's reference data of 20M+ PURL
  - Accessible, single source of truth to the security and SBOM ecosystem at large and improve the quality and accuracy of PURLs in use, imperative for CRA compliance
  - EU-funded through the NGI0 Commons fund with financial support from the European Commission's Next Generation Internet programme
    - https://nlnet.nl/project/purlvalidator/

- **Better PURL** project extends the syntax validation and verifies the existence of a known PURL, exposed as tools and utilities
  - Ensures package reference data across SBOMs and formats are consistent
  - Funded through corporate sponsors

Open Knowledge Base: License Data

- **ScanCode LicenseDB** includes 2,000+ licenses and 35,000 rules
  - Basic license data in ScanCode LicenseDB
    - No known alternative with comparable depth and breadth
    - https://scancode-licensedb.aboutcode.org/
  - License detection rules in ScanCode Toolkit
  - DejaCode is synchronized with LicenseDB and adds License Conditions

Open Knowledge Base

Open Knowledge Base: Package Data

- **PurlDB** includes 21M+ packages and, files and their fingerprints
  - Database of software package metadata keyed by Package-URL
  - Includes all major ecosystems and distributions - sources AND binaries - with built-in mining of all package ecosystems and on-demand data collection
    - Public PurlDB available at: https://public.purldb.io/api/packages/
- Other package databases:
  - ClearlyDefined (OSI), deps.dev (Google, proprietary)
  - Centralized and too big to share
  - No on-premises option for private operations (too big again)

Open Knowledge Base

Sharing all the scan data

- OCCTET is a FOSS compliance toolkit to simplify compliance for SMEs
  - Collaboration between Eclipse Foundation, Bitsea, EXPERTWARE, the EUROPEAN DIGITAL SME ALLIANCE, AboutCode, RED ALERT LABS, and DoubleOpen
  - EU-funded through the Digital Europe Programme (DIGITAL) / European Cybersecurity Competence Centre (ECCC)
    - https://occtet.eu/
- Integrated open data and open tools, based on the AboutCode stack, and open CRA compliance process guide with:
  - All the raw scan data for packages and vulnerabilities
  - Reduced compliance costs and efforts
    - Streamlined conformity assessment and reduced due diligence burden via shared database of conformity assessment attestation of FOSS

Open Knowledge Base: Vulnerability Data

- **VulnerableCodeDB** includes 760K+ packages and 240K+ vulnerabilities
  - PURL-based database aggregating and correlating data from all major ecosystems and vulnerability databases
  - Discover relations (and inconsistencies) in data from mining the graph
    - Public VulnerableCodeDB available at: https://public.vulnerablecode.io/
- Other vulnerability databases:
  - Google OSV (reuses some AboutCode code too), GitHub, GitLab, NVD
  - Often contain conflicting data for vulnerable ranges, fixed versions or affected packages
  - Comparison made possible with VulnTotal to query vulnerable version ranges given a PURL

Open Knowledge Base

Management Apps: DejaCode

Integrate all tools and data in one web-based app for SCA and compliance management and legal, security, and business users

- Manage product and component Inventories
- Curate code origin and licenses
- Define and apply license policies
- Launch scans and access the knowledge base
- Identify package vulnerabilities
- Integrated with AboutCode SCA Tools and Open Knowledge Base
- Consume and enrich SBOMs (CycloneDX or SPDX)
- Generate FOSS compliance documents, such as product Attribution Notices and SBOMs (CycloneDX or SPDX)
- Standard and custom reports
- JSON API and webhooks
- Built-in basic workflows

Management Apps

Efficient vulnerability management

- **Cyber Resilience Application for Vulnerability Exploitability Exchange (CRAVEX)** automates vulnerability triage and compliance reporting
  - Built for open source projects and small businesses as a free and open solution to comply with the emerging regulatory mandates (SBOMs, CRA) with minimal friction and costs
  - EU-funded through the NLnet and NGI0 funds with financial support from the European Commission's Next Generation Internet programme
    - https://nlnet.nl/project/CRAVEX
- Extension of the AboutCode stack, integrated with open data
  - Web-based, database-backed application to collect, track, and triage FOSS package vulnerabilities, determine their exploitability, and generate reporting as VEX documents
    - Package- and software product-centric management of vulnerabilities

Why the AboutCode stack?

- Makes open source safer to use with open source tools and open data
  - Modular and integrated best-in-class open source SCA tools adaptable to development team processes, tools and environment with coverage for all languages and frameworks
    - Tackling the harder code analysis problems so you don't have to
- Ensures open compliance and improves software supply chain security
  - Management web app for centralized policies, curations, and compliance workflows and data shared among business, legal, engineering and security teams
    - Share licensing and vulnerability risk management and license, security, and regulatory compliance responsibilities across the organization efficiently
  - Bespoke pipelines enable true end-to-end automation of SCA and compliance processes
- Engages with active community of contributors and users
  - All open source and most proprietary SCA, SBOM, and code hosting tools use AboutCode open source tools and libraries, and open data
  - Professional technical support and advisory services also available

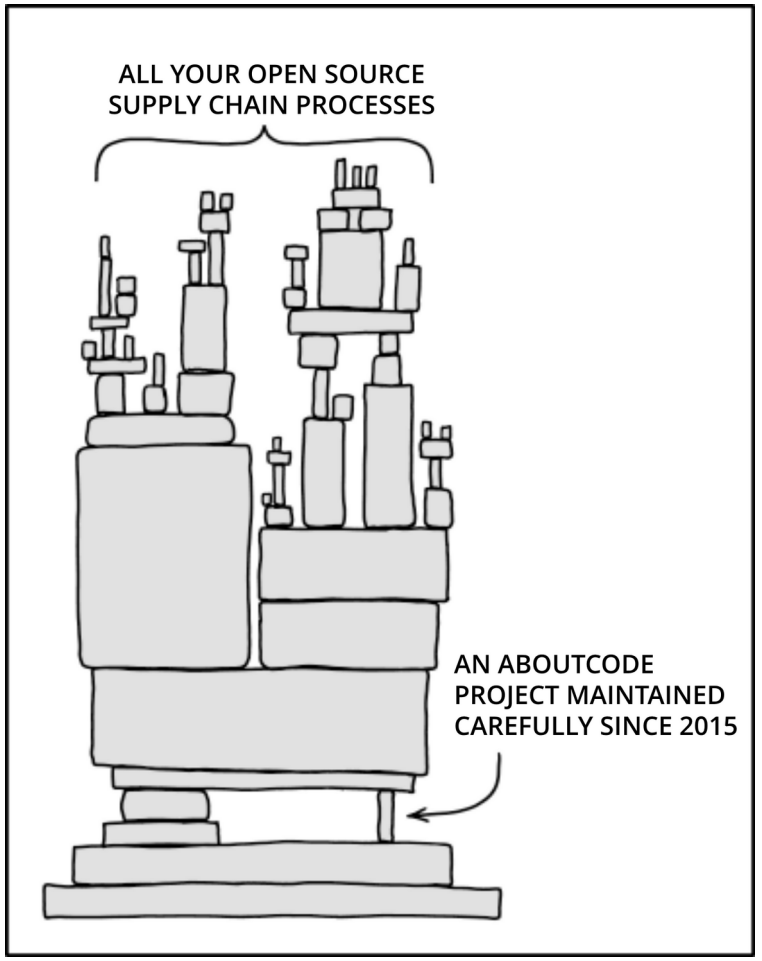
Solve the problem(s) with open source tools and open data

AboutCode

- More work to build a complete end-to-end compliance solution:
  - Compliance of open source projects against the CRA compliance
  - Security by design and by default
- Start small and avoid complexity
  - Waste of resources
- Contribute to open source projects
  - https://github.com/aboutcode-org
  - https://github.com/Open-Source-Compliance
- Engage with the community
  - AboutCode Slack: https://join.slack.com/t/aboutcode-org/shared_invite/zt-2hjzc448i-SZULSuI0~h6YNSUnBWlAqA
  - Join OpenChain activities: https://openchainproject.org/participate

AboutCode also needs your help!

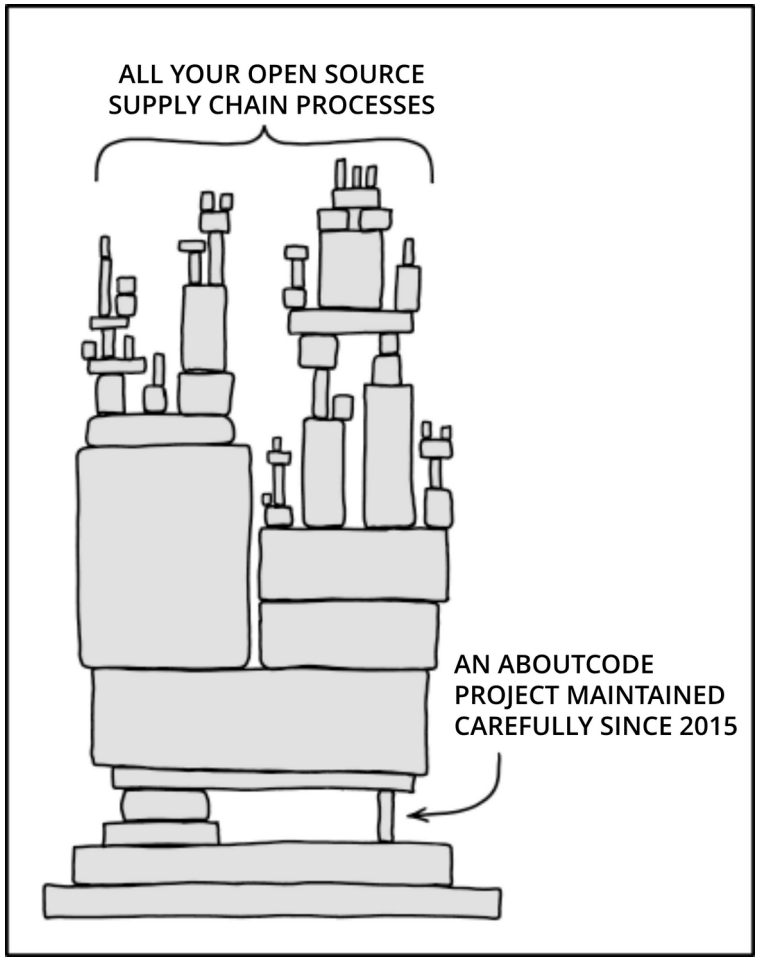
- Contribute code, documentation, bug reports
  - https://github.com/aboutcode-org
- Sponsor project maintainers
  - Accelerate development of new features and fund contributors
    - https://github.com/sponsors/aboutcode-org
  - Buy support, implementation, and advisory services to pay the AboutCode maintainers
    - Email pombredanne@aboutcode.org
- Join the community:
  - https://join.slack.com/t/aboutcode-org/shared_invite/zt-2hjzc448i-SZULSuI0~h6YNSUnBWlAqA
  - https://gitter.im/aboutcode-org/discuss



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

AboutCode also needs your help!

- The future? Let's work together!
  - New foundation to help you secure and automate your software supply chains with open tools, open data, and open standards
  - Goal = Resolve code origin, licensing, and vulnerabilities as a sustained and federated effort, so:
    - You can focus on the processes, policies, and automation – not data corrections and low-level rescanning
    - Software teams can better support and secure their software supply chains efficiently with reliable automation
  - Founding members to be announced soon!
    - Email hello@aboutcode.org for more information



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

Questions?

Connect on LinkedIn!

Philippe Ombredanne

Lead Maintainer, AboutCode