Why Is There No Free Software Vulnerability Database?

Open Source Summit + ELC North America 2020
June 29, 2020

AboutCode.org and nexB Inc.

Abstract

- Databases of known FOSS software vulnerabilities are mostly proprietary and privately maintained.
- Why not open data? Open like FOSS code.
- Find how we are working to build new FOSS tools to:
    - Aggregate and publish software component vulnerability data from multiple sources and
    - Automate the search for FOSS component security vulnerabilities.
- The benefit will be improved security of software applications with **open tools and open data for everyone**.

Background

- "Using Components with Known Vulnerabilities" is one of the OWASP Top 10 Most Critical Web Application Security Risks.
- Identifying vulnerable components is currently hindered by data structures and tools that are:
  - Designed primarily for proprietary software components,
  - Not comprehensive, and
  - Too dependent on voluntary submissions to the National Vulnerability Database.
- With the explosion of FOSS usage we need a new approach to efficiently identify FOSS security vulnerabilities.
- That approach should be based on open data and FOSS tools.

National Vulnerability Database (NVD)

- Maintained by the US Department of Commerce
- Data formats reflect commercial vendor-centric point of view
  - Predates explosion of FOSS software usage
  - Difficult to automatically relate to software components (CPE problem)
  - Also includes hardware (less interesting for FOSS community)
  - Represents only a subset of known vulnerabilities
    - Other sources not always covered (bug trackers, etc.)
    - Fragmented data sources led to the emergence of a commercial vulnerability data aggregation industry.

Solution

- ▷ Independently aggregate many software vulnerability data sources that can easily be recreated in a decentralized fashion
- ▷ Implement uniform software package identification based on **package-url** as the main searchable item
- ▷ Automated search for known package vulnerabilities
- ▷ Later: Crowdsourcing and peer-review classification

Solution

- FOSS tool to automate vulnerability search
  - Based on package data found in package manifests or installed package databases
- Leverage any tools that can detect and report FOSS packages using a **package-url**
  - ScanCode Toolkit scanning of package manifest files
  - Or OWASP tools, Sonatype and more.
- Later
  - Prototype discovery of new correlations between vulnerabilities and software packages from mining the graph

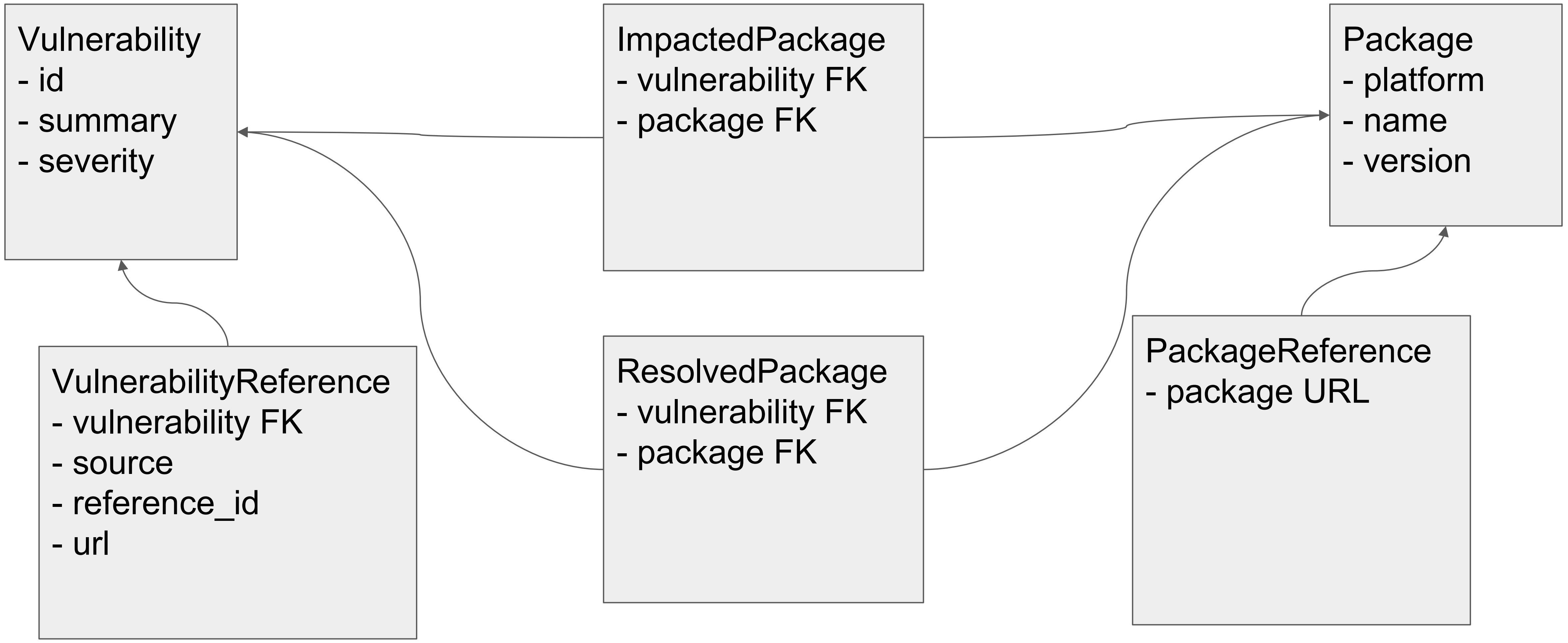
package-url (purl)

- Problem: Each package manager, platform, type or ecosystem has its own conventions and protocols to identify, locate and provision software packages
- Solution
  - An expressive and simple **package-url**, minimalist yet obvious
  - Identify & locate software packages reliably across tools and languages.
    - **pkg:npm/foobar@12.3.1**
    - **pkg:pypi/django@1.11.1**
  - Adopted or included in OWASP, ORT, ScanCode and more
  - Under consideration by the US NTIA as a possible CPE replacement
  - See https://github.com/package-url

Aggregation

- Collect and parse many sources
  - Common data model
  - Cross-references to create a graph
- Linux distro trackers (Debian, Ubuntu, RedHat, SUSE, Gentoo, ...)
  - Custom or standard formats (CVRF, OVAL)
- Application package trackers
  - NuGet, Rust, RubyGems, npm, ...
- Project-specific trackers
  - Apache, OpenSSL, ...
- NVD, Bug trackers, CHANGELOGs.

Data model


Vulnerability
- id
- summary
- severity
ImpactedPackage
- vulnerability FK
- package FK
Package
- platform
- name
- version
VulnerabilityReference
- vulnerability FK
- source
- reference_id
- url
ResolvedPackage
- vulnerability FK
- package FK
PackageReference
- package URL

VulnerableCode

- Primary current project is VulnerableCode
  - Project started by nexB / AboutCode.org
  - Code is at https://github.com/nexB/VulnerableCode
  - Discussion is at https://gitter.im/aboutcode-org/vulnerablecode
- Initial grant from the European Union and NLNet.nl (a non-profit foundation)
- Supported by internships through Google Summer of Code

Search

- Questions to answer
- Is foo@1.0 known to be vulnerable?
  - What are the vulnerabilities?
  - What is the severity of the vulnerability?
  - Which version has a fix?
- Future
  - Which commit introduced the bug? Which has the fix?
  - Is this code or binary vulnerable? (YARA rules)

Curation

- In the future, we will expose a public data curation queue for community review
- Key curation items
  - Validation of the vulnerability
  - Validation of package-urls
  - Severity and scoring
  - Actual commits
  - YARA Rules

Challenges

- Many data sources - redundant, unstructured, messy, incomplete
  - We appreciate the complexity of the task and why commercial vendors currently dominate the space
- Old, obsolete, or less useful data
  - More is not always better - e.g. old vulnerabilities on Windows 95
  - Commercial-only software (Windows, etc.) or hardware is excluded

Future plans

- More data sources
- Establish website and API for data consumption
- AI/ML for data quality improvements
- Community peer curation system
- Outreach to more FOSS projects

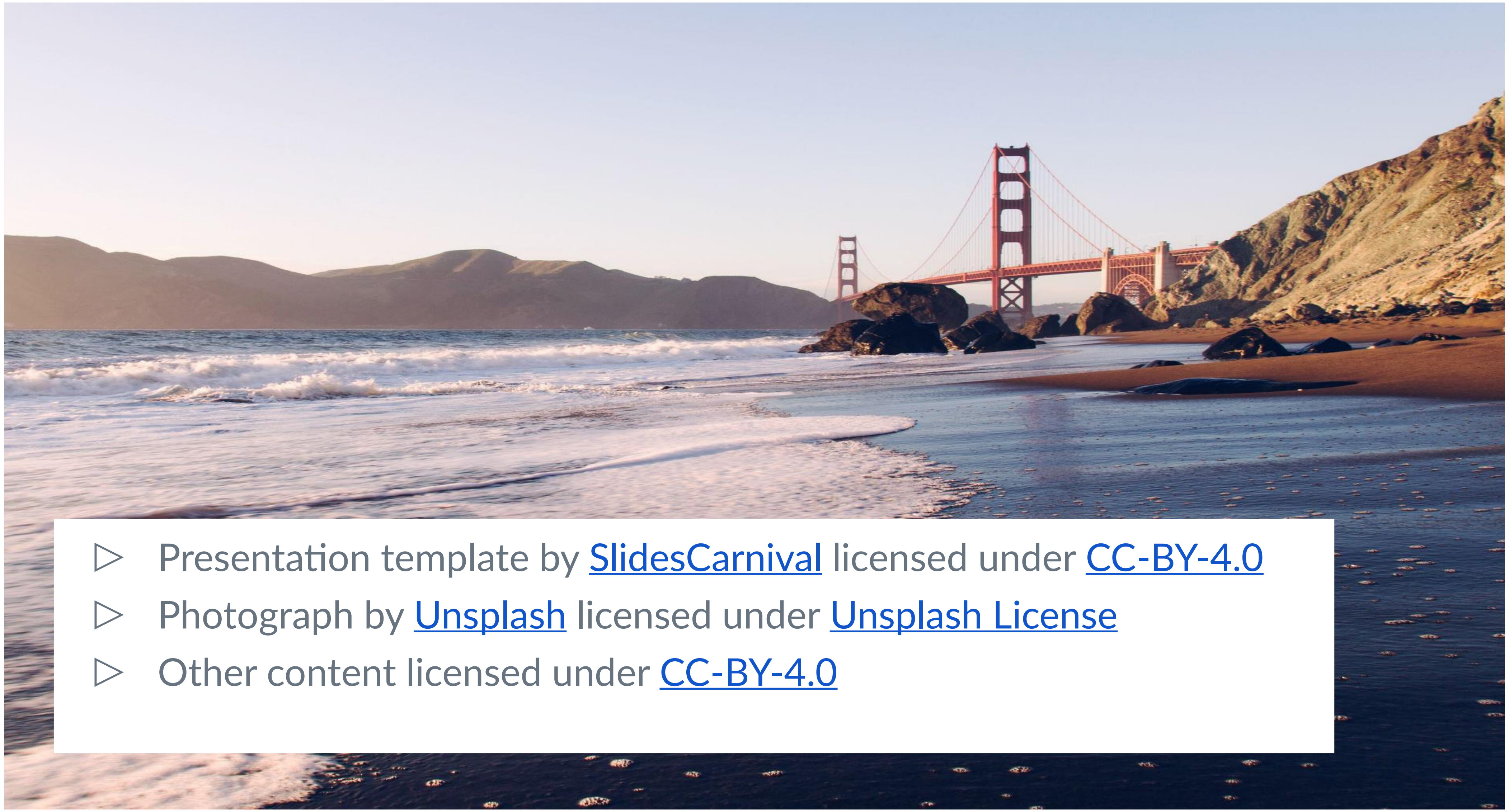
Sustainability

- Need to build a consortium to make open data sustainable
  - Not only for vulnerabilities - also for other SCA (Software Composition Analysis) data
- Starting to establish some collaboration with other projects (FASTEN, Eclipse Steady), others will include OWASP, upstream and package management communities
- **Join us to build the security commons!**

About nexB

- ▷ Focused on FOSS compliance since 2007
- ▷ Hybrid solution for FOSS governance
  - ○ Business applications for Legal/Business
  - ○ Open source tools for Developers
  - ○ APIs in-between
- ▷ Overview of our FOSS projects at www.aboutcode.org
- ▷ Our FOSS tools are at https://github.com/nexB
- ▷ Co-founders of SPDX - http://spdx.org/
- ▷ Co-founders of ClearlyDefined - https://clearlydefined.io/

Credits



▷ Presentation template by SlidesCarnival licensed under CC-BY-4.0

▷ Photograph by Unsplash licensed under Unsplash License

▷ Other content licensed under CC-BY-4.0