Operationalizing SBOM

With CycloneDX and Dependency-Track
• Security Engineer @ PayONE
• OWASP Dependency-Track Project Co-Lead
• OWASP CycloneDX contributor (I maintain the go stuff)

Niklas Düster

niklas.duester@owasp.org
github.com/nscur0
twitter.com/nscur0
# IS MY NISSAN AFFECTED?

Due to the defective Takata airbag installed in the models below, you and others in your vehicle are at risk of serious injury and death from this problem and we need you to take urgent action.

<table>
<thead>
<tr>
<th>Model</th>
<th>Airbag Affected</th>
<th>Vehicle build date</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-TRAIL (T30)</td>
<td>Passenger</td>
<td>2001-2007</td>
</tr>
<tr>
<td>Pulsar Sedan (N16)</td>
<td>Passenger</td>
<td>2001-2005</td>
</tr>
<tr>
<td>Pulsar Hatch (N16)</td>
<td>Passenger</td>
<td>2001-2005</td>
</tr>
<tr>
<td>Patrol Wagon (Y61)</td>
<td>Passenger</td>
<td>2001-2016</td>
</tr>
<tr>
<td>Patrol Cab Chassis (Y61)</td>
<td>Passenger</td>
<td>2006-2016</td>
</tr>
<tr>
<td>Navara (D22)</td>
<td>Passenger</td>
<td>2002-2015</td>
</tr>
<tr>
<td>Maxima (J31)</td>
<td>Passenger</td>
<td>2003-2008</td>
</tr>
<tr>
<td>Tiida (C11 Thailand build)</td>
<td>Driver &amp; Passenger</td>
<td>2006-2012</td>
</tr>
</tbody>
</table>

Source: www.nissan.com.au
Hard to swallow pills

YOU HAVE DEFECTIVE AIRBAGS IN YOUR SUPPLY CHAIN

BUT YOU DON'T KNOW ABOUT IT
97% of commercial codebases contain OSS¹

78% of code in commercial codebases is made up of OSS¹

[¹] Synopsys 2022 Open Source Security and Risk Analysis Report
SBOM
Executive Order on Improving the Nation’s Cybersecurity

MAY 12, 2021 • PRESIDENTIAL ACTIONS
Elements of SBOM

**Bare Minimum**¹
- Supplier Name
- Component Name & Version
- Other Unique Identifiers
- Dependency Relationships
- SBOM Author
- Timestamp

**We probably also want**
- Hashes
- Licenses
- Provenance
- Pedigree
- (Much, much more tbh)

¹ ntia.gov/files/ntia/publications/sbom_minimum_elements_report.pdf
This is not an SBOM talk

SBOM SmackDown: Conquer Dragons in the Shadows with OWASP CycloneDX
Steve Springett @ OWASP AppSec USA 2021
youtube.com/watch?v=vNpj6ogouIY
So we share SBOMs now

```
ACME App 1.0.0

- acme-web-framework 3.1.4
  - spring-web 5.3.7
    - spring-core 5.3.7
  - acme-logging 0.4.0
    - log4j-core 2.14.1

- acme-persistence-framework 1.1.0
  - hibernate-core 5.6.9.Final

- guava 30.0-jre
```
So we share SBOMs now

ACME App 1.0.0

- acme-web-framework 3.1.4
  - spring-web 5.3.7
  - spring-core 5.3.7
- acme-logging 0.4.0
  - log4j-core 2.14.1
- acme-persistence-framework 1.1.0
  - hibernate-core 5.6.9.Final
- guava 30.0-jre
(1) A certification that each item listed on the submitted bill of materials is free from all known vulnerabilities or defects affecting the security of the end product or service identified in—

(A) the National Institute of Standards and Technology National Vulnerability Database; and
(B) any database designated by the Under Secretary, in coordination with the Director of the Cybersecurity and Infrastructure Security Agency, that tracks security vulnerabilities and defects in open source or third-party developed software.

Source: congress.gov/bill/117th-congress/house-bill/7900
VEX
Minimum Elements of VEX

✅ **Metadata.** *What is this? Who created this and when?*

✅ **Product Details.** *What product are we talking about?*

✅ **Vulnerability Details.** *Which vulnerability are we talking about?*

✅ **Vulnerability Status.** *Is the product affected? Do I have to do something about it?*

---

Vulnerability Exploitability eXchange (VEX) – Use Cases

cisa.gov/sites/default/files/publications/VEX_Use_Cases_April2022.pdf
Affected not?

💀 Affected
✌️ Not Affected
✅ Fixed
🚧 Under Investigation

Source: cisa.gov/sites/default/files/publications/VEX_Use_Cases_April2022.pdf
Not affected?

❌ Uhh, uhm, so, I mean, ...
❌ Dave said so, but he’s on vacation right now
❌ Just trust me OK, why would I lie?

✅ Component not present
✅ Vulnerable code not present
✅ Vulnerable code not in execution path
✅ Vulnerable code not controllable by adversary
✅ Inline mitigations exist

Source: cisa.gov/sites/default/files/publications/VEX_Status_Justification_Jun22.pdf
CycloneDX

- Bill of Materials Standard for Cybersecurity Use Cases
  - ✔ Lightweight: Simplicity > Complexity
  - ✔ Optimized for Automation
- Lead by Steve Springett, Patrick Dwyer, Jeffry Hesse

🚀 OWASP Standards Flagship Project
🚀 Recommended by multiple government orgs worldwide
🚀 Used in production at an estimated 100k organizations
<table>
<thead>
<tr>
<th>Metadata</th>
<th>Supplier</th>
<th>Authors</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Manufacturer</td>
<td>Tools</td>
<td></td>
</tr>
<tr>
<td>Components</td>
<td>Application</td>
<td>Framework</td>
<td>Operating System</td>
</tr>
<tr>
<td></td>
<td>Library</td>
<td>Container</td>
<td>Device</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>File</td>
</tr>
<tr>
<td>Services</td>
<td>Provider</td>
<td>Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Endpoints</td>
<td>Trust Boundary</td>
<td></td>
</tr>
<tr>
<td>Dependencies</td>
<td>Components</td>
<td>Services</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Compositions</td>
<td>Compositions</td>
<td>Services</td>
<td>Dependencies</td>
</tr>
<tr>
<td>Vulnnerabilities</td>
<td>Details</td>
<td>Source</td>
<td>Exploitability</td>
</tr>
<tr>
<td></td>
<td>Advisories</td>
<td>Risk Ratings</td>
<td></td>
</tr>
<tr>
<td>Extensions</td>
<td>Properties</td>
<td>Per Organization</td>
<td>Per Team</td>
</tr>
<tr>
<td></td>
<td>Formal Taxonomy</td>
<td>Per Industry</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

High-level data model of CycloneDX
Source: cyclonedx.org
Use Cases

The following examples provide guidance as to the minimal fields required to achieve specific use cases. Ideally, all optional fields would be populated in order to achieve all use cases. Many of the cases highlighted are directly or closely related to security.

Inventory

A complete and accurate inventory of all first-party and third-party components is essential for risk identification. BOMs should ideally contain all direct and transitive components and the dependency relationships between them.

CycloneDX is capable of describing the following types of components:

<table>
<thead>
<tr>
<th>COMPONENT TYPE</th>
<th>CLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Component</td>
</tr>
<tr>
<td>Container</td>
<td>Component</td>
</tr>
<tr>
<td>Device</td>
<td>Component</td>
</tr>
<tr>
<td>Library</td>
<td>Component</td>
</tr>
</tbody>
</table>

- Inventory
- Known vulnerabilities
- Integrity verification
- Authenticity
- Package evaluation
- License compliance
- Assembly
- Dependency graph
- Provenance
- Pedigree
- Service definition
- Properties / name-value store
- Packaging and distribution
- Composition completeness
- OpenChain conformance
- Vulnerability remediation

CycloneDX Use Cases
cyclonedx.org/use-cases/
- Intelligent Component Analysis Platform
- Ideal for Procurement and 🌟DevSecOps🌟
- Lead by Steve Springett and yours truly 👍😎👍

🚀 OWASP Tools Flagship Project
🚀 20B Components Analyzed each Month
🚀 >7M Docker Pulls
SBOM Production  
Generated during CI/CD or acquired from suppliers

SBOM Ingestion  
Via REST API, Web Interface, Jenkins Plugin, GitHub Action

SBOM Analysis  
Component analysis for security, operational, and license risk

Continuous Monitoring  
Continuous analysis of portfolio for risk and policy compliance

Intelligence Streams  
Real-time analysis and security events delivering actionable findings to external systems

Intelligent Response  
Events delivered via Webhooks or ChatOps and findings published to risk management and vulnerability aggregation platforms
Get Involved!

cyclonedx.org/about/participate/
github.com/DependencyTrack/dependency-track/CONTRIBUTING.md
But wait, there is more!

[Click on the image to access the OWASP Software Component Verification Standard (SCVS) page.]

[QR Code: OWASP Software Component Verification Standard (SCVS) page]

[OWASP FOUNDATION owasp.org]
Your turn.