SBOM Know How

Release 0.1

Ivana Atanasova

Mar 23, 2023

CONTENTS

CHAPTER

ONE

SBOM KNOW HOW

SBOM Know How is a documentation project to bring "What you need to know about SBOMs" into one place, including specifications, tools and useful references.

It's available in a more read-friendly format on https://sbom-know-how.readthedocs.io.

CHAPTER

TWO

CONTENT LIST

2.1 Existing SBOM Specifications and Advisories

2.1.1 Software Package Data Exchange (SPDX)

SPDX is an open standard for communicating SBOM information, including components, licenses, copyright, and security references. It was initiated as a part of the Linux Foundation's Open Compliance Program and is an official ISO-approved standard.

For full detail, please see the SPDX specification documentation.

Latest ISO Approved Version

SPDX 2.2 is currently the latest ISO approved version.

Latest version

SPDX 2.3 is the latest published version of the spec.

Upcoming

The upcoming SPDX model updates can be found in the SPDX 3 model GitHub repository. Profiles within SPDX v3+ are considered valid SPDX documents and there is no operational restriction on how one may choose to combine them.

SPDX Lite

SPDX supports a Lite version which is a a subset of the SPDX specification. The SPDX Lite profile consists of mandatory fields from the Document Creation and Package Information sections and other basic information.

2.1.2 CycloneDX

The CycloneDX specification is initiated by OWASP and is focused on creating security context. It allows identifying known vulnerabilities in components using CPE, SWID, and PURL fields.

For more detail, please see the OWASP Vulnerability Naming Schemes and CycloneDX official documentation.

2.1.3 Software Identification (SWID)

Software Identification (SWID) Tags are defined by the ISO/IEC 19770-2:2015 standard. They provide a transparent way for organizations to track installed software on managed devices. The SWID Tags contain information for specific releases of a software product.

The SWID standard defines a lifecycle where a SWID Tag is added to an endpoint as part of the software product's installation process and deleted by the product's uninstall process.

For full detail, please see Guidelines for the Creation of Interoperable Software Identification (SWID) Tags.

A list of SWID Tag Tools can be found in NIST SDWID Tools.

2.1.4 Vulnerability Exploitability eXchange (VEX)

The primary use cases for VEX are to provide users (e.g. operators, developers, and services providers) additional information on whether a product is impacted by a specific vulnerability in an included component and, if affected, whether there are actions recommended to remediate.

To reduce effort spent by users investigating non-exploitable vulnerabilities that don't affect a software product, suppliers can issue a VEX. A VEX is an assertion about the status of a vulnerability in specific products. The status can be:

- Not affected – No remediation is required regarding this vulnerability.
- Affected – Actions are recommended to remediate or address this vulnerability.
- Fixed – Represents that these product versions contain a fix for the vulnerability.
- Under Investigation – It is not yet known whether these product versions are affected by

the vulnerability. An update will be provided in a later release.

(Quoted from NTIA VEX One-page Summary)

For latest updates, please refer to VEX Status Justifications, June 2022.

Useful References

- VEX Guide from Rezilion.
- *VEX Tools*

2.1.5 SBOM Artifact Image Specification

The SBOM Artifact Image Spec defines how to bundle SBOMs as OCI images. SBOM Artifact Image consist of one or more SBOM files, with annotations to indicate which other OCI artifacts, or parts of OCI artifacts, they are intended to cover.

The spec is not restricted to SBOMs generated from OCI images.

The SBOM OCI Artifact Specification defines a method of storing SBOM files which makes them easy to store and distribute, alongside the OCI artifacts they refer to.

Read about using *cosign* for attaching SBOMs to OCI images in the Cosign SBOM Spec.

2.1.6 Package URL

PURL is a mini spec used in *CycloneDX*, *SPDX* and CSAF *VEX*.

It is is a standardization attempt to reliably identify and locate software packages with the existing approaches. A purl is a URL string used to identify and locate a software package in a mostly universal and uniform way across programing languages, package managers, packaging conventions, tools, APIs and databases.

Companion open source vulnerability databases

- VulnerableCode available at https://public.vulnerablecode.io is keyed by purl. It is an open source code and open data correlated and aggregated vulnerability database.
- purldb is a companion database of all the purls listed in the repo.

2.2 Existing SBOM Tools

CycloneDX Tool Center

2.2.1 Tools Classification

ID	Tool	Generation	Consumption	Transformation	Cyclonedx	Spdx	Vulnerabilty Scanning	Licensing	Sbom Quality
TOOL1	apko	yes		yes	yes	yes			
TOOL10	SBOM Operator	yes	yes		yes	yes	yes		
TOOL11	ScanCode	yes	yes		yes	yes			
TOOL12	SPDX SBOM Generator	yes				yes			

continues on next page

Table 1 – continued from previous page

ID	Tool	Generation	Consumption	Transformation	Cyclonedx	Spdx	Vulnerabilty Scanning	Licensing	Sbom Quality
TOOL13	Syft	yes			yes	yes			
TOOL14	Syft	yes			yes	yes			
TOOL15	Bomber		yes		yes	yes	yes		
TOOL16	Dagger-Board		yes		yes	yes	yes		
TOOL17	Dependency-Track		yes		yes		yes	yes	
TOOL18	SBOM Scorecard		yes		yes	yes			yes
TOOL19	FOSSology		yes			yes		yes	
TOOL2	CycloneDX Tool Center	yes			yes				
TOOL20	Grype		yes		yes	yes	yes		
TOOL21	Hoppr Cop		yes		yes		yes		
TOOL22	SBOM Diff Action		yes		yes	yes			
TOOL23	SBOM Utility		yes		yes	yes			yes
TOOL24	Scan-Code.io	yes	yes		yes	yes	yes	yes	
TOOL25	Trivy	yes	yes		yes	yes	yes	yes	
TOOL26	Vulnerability Operator		yes		yes	yes	yes		
TOOL27	CDX2SPDX	no	no	yes	yes	yes			
TOOL28	Dagger-Board		yes		yes	yes	yes		
TOOL29	Dagger-Board		yes		yes	yes	yes		
TOOL3	Docker SBOM	yes			yes				
TOOL30	SBOM Quality Scoring		yes		yes	yes			yes

continues on next page

Table 1 – continued from previous page

ID	Tool	Generation	Consumption	Transformation	Cyclonedx	Spdx	Vulnerabilty Scanning	Licensing	Sbom Quality
TOOL4	Fat-BOM	yes		yes		yes			
TOOL5	KubeClarity	yes	yes		yes	yes	yes		
TOOL6	K8s BOM	yes	yes			yes			
TOOL7	OSS Review Toolkit	yes	yes		yes	yes	yes	yes	
TOOL8	Pkgconf bomtool	yes				yes			
TOOL9	Salus	yes				yes			

2.2.2 SBOM Generation Tools

Tool data: **apko** *TOOL1*

tool: apko
generation: yes
transformation: yes
cyclonedx: yes
spdx: yes

apko

apko provides SBOM support by producing SBOM documents for OCI images.

Tool data: **CycloneDX Tool Center** *TOOL2*
tool: CycloneDX Tool Center generation: yes cyclonedx: yes

CycloneDX Tool Center

Generates CycloneDX format SBOMs. Full list of tools can be found in the CycloneDX Tool Center.

Tool data: **Docker SBOM** *TOOL3*
tool: Docker SBOM generation: yes cyclonedx: yes

Docker SBOM

Generates SBOMs for Docker images with the currently experimental `docker sbom` command. Based on *Syft*. For more detail, please visit the Docker SBOM Documentation.

Tool data: **FatBOM** *TOOL4*
tool: FatBOM generation: yes transformation: yes spdx: yes

FatBOM

FatBOM generates SBOMs via *Syft*, *Salus*, *SPDX SBOM Generator* and *K8s BOM* and composes them into a single SPDX SBOM in JSON format. Full details can be found in the FatBOM GitHub repository.

Tool data: **KubeClarity** *TOOL5*
tool: KubeClarity generation: yes consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

KubeClarity

KubeClarity uses *Syft* and Cyclonedx-gomod (*CycloneDX Tool Center*) to generate SBOMs and offers *SBOM scanning*.

Tool data: **K8s BOM** *TOOL6*
tool: K8s BOM generation: yes consumption: yes spdx: yes

K8s BOM

K8s BOM generates SBOMs from files, images, and docker archives and supports pulling images from remote registries. The SBOM data can be exported to an in-toto provenance attestation. For SBOM scanning details, please see the *K8s BOM consumption tools* section.

Tool data: **OSS Review Toolkit** *TOOL7*
tool: OSS Review Toolkit generation: yes consumption: yes vulnerabilty_scanning: yes licensing: yes cyclonedx: yes spdx: yes

OSS Review Toolkit

The OSS Review Toolkit's Reporter generates SBOMs in *CycloneDX* or *SPDX* format.

Tool data: **Pkgconf bomtool** *TOOL8*
tool: Pkgconf bomtool generation: yes spdx: yes

Pkgconf bomtool

Bomtool is a feature of pkgconf and can be used for generating SBOMs for C/C++ packages under Alpine. Usage: `` `bash $ bomtool <package_name> ` `` where package name should be linked in *pkgconf*.

Tool data: **Salus** *TOOL9*
tool: Salus generation: yes spdx: yes

Salus

Salus is an Open Source SBOM generation tool implemented by Microsoft. It allows build-time generation from source and packages, as well as CI/CD pipelines integration via GitHub Actions and Azure DevOps Pipelines.

Tool data: **SBOM Operator** *TOOL10*
tool: SBOM Operator generation: yes consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

SBOM Operator

SBOM Operator uses *Syft* to generate SBOMs from each image deployed in a Kubernetes cluster. Relies on go-containeregistry for downloading images. Allows *analysis*.

Tool data: **ScanCode** *TOOL11*
tool: ScanCode generation: yes consumption: yes cyclonedx: yes spdx: yes

ScanCode

ScanCode is an OSS tool from AboutCode that generates SBOMs for containers, system packages, and many language packages. Supports both *SPDX* and *CycloneDX*. It's embedded in *ORT*, *Tern*, *FOSSology*, Fosslight, Barista, Philips software license-scanner, and others. It provides a ScanCode.io (CLI, web UI and REST API) to read and write SPDX and CycloneDX.

Tool data: **SPDX SBOM Generator** *TOOL12*
tool: SPDX SBOM Generator generation: yes spdx: yes

SPDX SBOM Generator

The SPDX SBOM Generator generates SBOMs from source code. The supported package managers can be found the the tool Overview.

Tool data: **Syft** *TOOL13*
tool: Syft generation: yes cyclonedx: yes spdx: yes

Syft

Syft generates SBOMs from container images and file systems. It provides both a CLI tool and a Go library. Supported ecosystems are available in the tool documentation.

Tool data: **Syft** *TOOL14*
tool: Syft generation: yes cyclonedx: yes spdx: yes

Tern

Tern is a software package inspection tool that generates SBOMs for container images and Dockerfiles. Supports both *SPDX* and *CycloneDX*, *SWID*.

2.2.3 SBOM Consumption Tools

Tool data: **Bomber** *TOOL15*
tool: Bomber consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

Bomber

Bomber is an application that scans SBOMs for security vulnerabilities. Works with *CycloneDX* JSON and XML, as well as *SPDX* and *Syft* JSON.

Tool data: **DaggerBoard** *TOOL16*
tool: DaggerBoard consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

DaggerBoard

DaggerBoard is a vulnerability scanning tool, based on ingesting SBOM files (CycloneDX,SPDX), that outputs results in a human-readable format.

Tool data: **Dependency-Track** *TOOL17*
tool: Dependency-Track consumption: yes vulnerabilty_scanning: yes licensing: yes cyclonedx: yes

Dependency-Track

Dependency-Track <https://github.com/DependencyTrack/dependency-track> `_ uses :ref:`CycloneDX <cdx> SBOMs to monitor component usage across all versions of the application in its portfolio, in order to identify and reduce risk in the software supply chain.

Tool data: **FOSSology** *TOOL19*
tool: FOSSology consumption: yes licensing: yes spdx: yes

FOSSology

FOSSology is a compliance scanner tool for license, copyright and export control. Documentation can be found on the official web site.

Tool data: **Grype** *TOOL20*
tool: Grype consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

Grype

Grype is a vulnerability scanner for container images and file systems. If scans for vulnerabilities for both operating system and language-specific packages. Supports Docker, OCI and Singularity image formats, as well as consumes SBOM attestations.

Tool data: **Hoppr Cop** *TOOL21*
tool: Hoppr Cop consumption: yes vulnerabilty_scanning: yes cyclonedx: yes

Hoppr Cop

Hoppr Cop generates vulnerability information from CycloneDX SBOMs. It's available both as a CLI and a python library.

KubeClarity

KubeClarity detects and manages SBOMs and vulnerabilities of container images and file systems. It can also scan K8s runtime to detect vulnerabilities discovered post-deployment. It uses Grype and Dependency-Track for vulnerability scanning. More detail can be found in the KubeClarity documentation.

K8s BOM

K8s BOM offers drawing a structure of an SPDX document and serves for verification.

OSS Review Toolkit

The OSS Review Toolkit provides a list of tools, including Analyzer for dependencies of projects and their metadata, Downloader for fetching source code and dependencies, Scanner for detecting license / copyright findings from source code, Advisor for retrieving security advisories for used dependencies, and others.

Tool data: **SBOM Diff Action** *TOOL22*
tool: SBOM Diff Action consumption: yes cyclonedx: yes spdx: yes

SBOM Diff Action

SBOM Diff Action is a GitHub integration tool that creates diffs for SBOMs from PR changes.

SBOM Operator

The SBOM Operator allows checks for changed images and pods within a cluster. Provides vulnerability scans via the *Vulnerability Operator*. For more detail, please refer to the SBOM Operators Analysis-Trigger section.

Tool data: **SBOM Scorecard** *TOOL18*
tool: SBOM Scorecard consumption: yes cyclonedx: yes spdx: yes sbom_quality: yes

SBOM Scorecard

SBOM Scorecard is a tool for providing metrics for SBOM quality, including spec compliance, generation information and package ids, licensed and version.

Tool data: **SBOM Utility** *TOOL23*
tool: SBOM Utility consumption: yes cyclonedx: yes spdx: yes sbom_quality: yes

SBOM Utility

SBOM Utility is a CycloneDX and SPDX SBOM validation tool.

Tool data: **SBOM Quality Scoring** *TOOL30*
tool: SBOM Quality Scoring consumption: yes cyclonedx: yes spdx: yes sbom_quality: yes

SBOM Quality Scoring

sbomqs provides comprehensive quality scoring for your sboms. It provide a quick compliance check of your sboms with NTIA minimum elements. It uses license, spec compliance, data quality to help generate an accurate score for your sbom generator. Supports all SPDX, CycloneDX and SWID spec formats.

Tool data: **ScanCode.io** *TOOL24*
tool: ScanCode.io generation: yes consumption: yes vulnerabilty_scanning: yes licensing: yes cyclonedx: yes spdx: yes

ScanCode.io

ScanCode.io is a CLI, web UI and REST API that can read and write *SPDX* and *CycloneDX*. It embeds scancode-toolkit and can scan for origin, vulnerabilities and license a large range of codebase including first class support for Linux containers and docker images, VM Images, Windows containers, Windows VM images as well as packages and codebase with pre-defined configurable pipelines. It detects all archives, installed and embedded formats for packages from Maven, Pypi, Ruby, Rust cargo, Go, NuGet, Alpine, Debian and derivative, RPM distributions, Windows, npm and yarn, Bower, Chef, Cocoapods, conda, cran, haxe, MSI, opam, pubspec. Both ScanCode toolkit and ScanCode.io are extensively based on and use Package URL.

Tool data: **Trivy** *TOOL25*
tool: Trivy generation: yes consumption: yes vulnerabilty_scanning: yes licensing: yes cyclonedx: yes spdx: yes

Trivy

Trivy scans container images, file systems, Git repositories, and Kubernetes clusters or resources for open source packages and dependencies, CVEs, IaC misconfigurations, and sensitive information. It generates SBOMs in the scanning process. Trivy also allows signing and verifying SBOM attestations.

Tool data: **Vulnerability Operator** *TOOL26*
tool: Vulnerability Operator consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

Vulnerability Operator

The vulnerability-operator uses Grype for scanning SBOMs and exports all found vulnerabilities into a JSON format.

2.2.4 SBOM Transformation Tools

apko

apko produces SBOM documents and provides an SBOM composition functionality

Tool data: **CDX2SPDX** *TOOL27*
tool: CDX2SPDX generation: no consumption: no transformation: yes cyclonedx: yes spdx: yes

CDX2SPDX

CDX2SPDX is a Java tool that converts *CycloneDX* SBOMs to *SPDX*.

Tool data: **DaggerBoard** *TOOL28*
tool: DaggerBoard consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

SBOM Composer

SBOM Composer is a tool that serves for composing *SPDX* SBOM files into a single SPDX document. Not restricted by the contents of the composable SBOMs, as long as they are valid SPDX. The version of the final document is the latest amongst all composed.

Tool data: **DaggerBoard** *TOOL29*
tool: DaggerBoard consumption: yes vulnerabilty_scanning: yes cyclonedx: yes spdx: yes

Tejolote

Tejolote is a tool that consumes SBOMs and generates SLSA provenance attestations about build runs.

2.2.5 SBOM Parsers

CycloneDX Parsers

CycloneDX parsers can be found in the CycloneDX Tool Center's Library section.

SPDX Parsers

SPDX has the following language-specific tools:

- SPDX Online Tool for SBOM validation
- tools-golang
- tools-java
- spdx-tools-js
- tools-python
- spdx-tools in pypi
- opensbom-generator/parsers provides parsers for a wide range of package managers

Conversion tools

- cdx2spdx converts CycloneDX SBOMs to SPDX. The conversion is done based on the SPDX-CycloneDX Mapping.

2.2.6 SWID Tools

A list of SWID Tag Tools can be found in NIST SDWID Tools.

2.2.7 VEX Tools

VEXctl

VEXctl is a tool to apply and attest VEX data. It can "turn off" alerts of vulnerabilities known not to affect a product. It allows both creating VEX statements and VEXing a results set.

Vexy

Vexy is a Python-based CLI for generating VEX in CycloneDX format.

2.3 Useful References

2.3.1 Awesome SBOM

Awesome SBOM is created to collect and share a curated list of SBOM tools, frameworks and publications.