Main Keyword	SBOM Tools
Search volume	1000
Length in Words	1k

1. Agile SEO Research

Relevant Sub-Keywords to Include in the Article

Keyword	Search volume	Used
sbom generation tools	110	yes

Top Search Results - URLs On First Page of Google

1. https://www.wiz.io/academy/top-open-source-sbom-tools
2. https://github.com/awesomeSBOM/awesome-sbom
3. https://www.cybeats.com/blog/top-7-sbom-generation-tools-and-how-to-choose
4. https://mergebase.com/blog/best-tools-for-generating-sbom/
5. https://github.com/microsoft/sbom-tool
6. https://anchore.com/sbom/how-to-generate-an-sbom-with-free-open-source-tools/
7. https://blog.sonatype.com/5-tools-to-automate-sbom-creation
8. https://spectralops.io/blog/top-10-sbom-tools-in-2023/
9. https://www.csoonline.com/article/573225/8-top-sbom-tools-to-consider.html
10. https://www.techtarget.com/searchsecurity/tip/SBOM-tools-to-start-securing-the-software-supply-chain

---

2. Proposed Outline

- What Are SBOM Tools?
- Why Are SBOM Tools Important?
- Types of SBOM Standards
    - CycloneDX
    - SPDX
    - Software Identification (SWID) Tags
- Best SBOM Tools
    - Aqua Security
    - Syft
    - The SBOM Tool
    - Tern

    - CycloneDX Generator
- How to Choose SBOM Generation Software
    - Accuracy
    - Scalability
    - Data privacy and Security
    - Support

==========

Article Info	Content
Page Main KW	SBOM Tools
Site Section	Cloud Native Wiki
Super Cluster	
Topic Cluster	Vulnerability management
Role in Cluster	Supporting Page
Cluster Plan	https://docs.google.com/spreadsheets/d/12ZiThH4vZEh-UJYw0alAafaWUn-M1ZEJUU993sJrxCY/edit
Meta Title (64)	SBOM Tools: The Basics and 5 Free Tools to Get You Started
Meta Description (155)	Software Bill of Materials (SBOM) tools are designed for the detailed listing of components in a software system.
Planned Length	1k
Actual Length (excl. product content)	

—

SBOM Tools: The Basics and 5 Free Tools to Get You Started

What Are SBOM Tools?

Software Bill of Materials (SBOM) tools are designed for the detailed listing of components in a software system. These tools automate the process of identifying and documenting every piece of external code or internal module. The goal is to provide a clear catalog of software components to ensure transparency in software development and maintenance.

The adoption of SBOM tools facilitates software supply chain risk management. By offering a comprehensive view into the software components, these tools allow for better tracking, analysis, and control over the software's components, making the software development lifecycle more secure and manageable.

This is part of a series of articles about vulnerability management.

In this article:

Why Are SBOM Tools Important?

SBOM tools are crucial in enhancing cybersecurity and compliance. They help organizations understand their software dependencies, making it easier to identify and remediate vulnerabilities rapidly. This increased visibility into software components aids in preventing potential security breaches by ensuring all pieces of the software are up-to-date and secure.

In addition, SBOMs are becoming essential for compliance. Several government frameworks are now encouraging or requiring the use of SBOMs:

- **US Cybersecurity and Infrastructure Security Agency (CISA):** Recommends using SBOMs as part of guidelines for secure software development
- **Executive Order 14028:** Directs NIST to develop guidelines for creating and publishing SBOMs and establish criteria for using SBOMs in federal procurement processes.
- **US National Telecommunications and Information Administration (NTIA):** Defines minimum elements that should be included in an SBOM.
- **EU Agency for Cybersecurity (ENISA):** Provides Guidelines for Securing the Supply Chain for the Internet of Things, which include the use of SBOMs.
- **UK National Cyber Security Centre (NCSC):** Recommends that organizations use SBOMs to reduce the risk of software components they use.

Types of SBOM Standards

SBOMs are created according to industry-standard specifications. There are several standards used to create SBOMs; we'll review three common standards.

CycloneDX

CycloneDX is a lightweight SBOM standard designed for use in application security contexts and supply chain component analysis. It focuses on the accurate and reproducible identification of components in the software supply chain. CycloneDX supports a range of content types, allowing for comprehensive coverage of software components.

SPDX

Software Package Data Exchange (SPDX) is another SBOM standard that provides a uniform way to document the components in a software package. It aims at standardizing the way software components are identified, attributed, and audited, making the data easily shareable across different platforms and tools. SPDX is widely adopted due to its flexible and adaptable nature.

SPDX supports comprehensive documentation of components, including licensing and security vulnerabilities associated with each component. This ensures a better understanding and management of software licenses and security risks.

Software Identification (SWID) Tags

Software Identification (SWID) tags offer a method for uniquely identifying software products and components. They provide vital product metadata that can be used for inventory and asset management, helping organizations track and manage their software assets effectively. SWID tags support automation and provide a consistent identification mechanism that aligns with ISO/IEC standards.

The use of SWID tags facilitates compliance with licensing terms and regulatory requirements by ensuring accurate and up-to-date software inventory. It enhances security measures by enabling swift identification and response to vulnerabilities or software discrepancies.

Best SBOM Tools

Trivy

Trivy is an open-source vulnerability scanner that supports the generation of SBOMs. It is designed for containers and other artifacts. It is developed by Aqua Security and provides comprehensive insights into software components, helping in maintaining secure and compliant software environments.

GitHub repo: https://github.com/aquasecurity/trivy

Key features of Trivy:

- **Vulnerability scanning:** Performs thorough vulnerability scans on container images, file systems, and Git repositories, detecting vulnerabilities across a wide range of programming languages and operating system packages.
- **SBOM generation:** Generates SBOMs that offer detailed visibility into the software components within container images and other artifacts. This capability aids in identifying and managing software dependencies effectively.
- **Integration capabilities:** Integrates seamlessly with CI/CD pipelines, making it easy to incorporate into automated workflows for continuous security checks and compliance validation.
- **Support for multiple formats:** Supports multiple output formats for SBOMs, including CycloneDX and SPDX, allowing for flexibility in how the information is used and shared.
- **Ease of Use:** Offers a user-friendly command-line interface, enabling quick adoption and straightforward usage without extensive configuration.

CycloneDX Generator

CycloneDX Generator, known as cdxgen, is a CLI tool, library, REPL, and server designed for generating CycloneDX-compliant Software Bills of Materials (SBOMs) for a variety of programming languages and platforms. Its primary goal is to aggregate all project dependencies into a CycloneDX BOM in JSON format, supporting languages including C/C++, Node.js, PHP, Python, Ruby, Rust, Java, .Net, Dart, Haskell, Elixir, and Go, among others.

GitHub repo: https://github.com/CycloneDX/cdxgen

Key features of CycloneDX Generator:

- **Comprehensive language support:** Creates SBOMs for a range of programming languages and package formats, with support for transitive dependencies and component evidence.
- **CycloneDX specification compliance:** Ensures generated SBOMs are compliant with CycloneDX specification versions 1.4 to 1.6, making it suitable for a variety of compliance and security applications.

- **Plugin extensions for enhanced capabilities:** With plugins, cdxgen can generate OBOMs for Linux docker images, VMs running Linux or Windows, and includes tools like evinse for generating component evidence, CBOM, and SaaSBOM for supported languages.
- **Deep dependency inspection:** Performs deep inspection into project dependencies, making it effective in complex enterprise environments with multiple repositories and intricate build requirements.
- **Installation and usage flexibility:** Offers various installation methods including NPM, Homebrew, Deno, and Docker, along with detailed documentation for getting started and utilizing the tool effectively across different environments.

Syft

Syft is a command-line interface (CLI) tool and Go library designed for creating a Software Bill of Materials (SBOM) from container images and filesystems. It offers vulnerability detection capabilities, especially when paired with a vulnerability scanner like Grype. Syft is built to support a range of ecosystems and package formats.

GitHub repo: https://github.com/anchore/syft

Key features of Syft:

- **Generates SBOMs:** Capable of generating SBOMs from container images, filesystems, and archives, enabling discovery of packages and libraries across a variety of sources.
- **Supports multiple image formats:** Compatible with OCI, Docker, and Singularity image formats, providing flexibility in working with different container technologies.
- **Linux distribution identification:** Efficiently identifies the Linux distribution within container images or filesystems, aiding in targeted vulnerability scanning and analysis.
- **Integration with Grype:** Designed to work seamlessly with Grype, a fast vulnerability scanner, for vulnerability detection and management.
- **Signed SBOM attestations:** Offers the capability to create signed SBOM attestations using the in-toto specification, enhancing the integrity and trustworthiness of SBOMs.
- **SBOM format conversion:** Allows for easy conversion between different SBOM formats, including CycloneDX, SPDX, and Syft's own format.

Tern

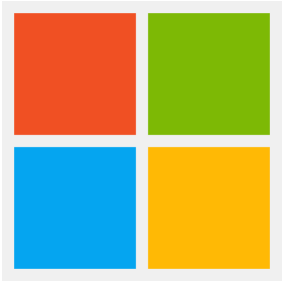
Tern is a tool designed to inspect software packages within container images and generate detailed Software Bills of Materials (SBOMs). Developed in Python3 and supplemented with shell scripts, Tern is optimized for dissecting container layers to reveal the intricacies of their software composition.

GitHub repo: https://github.com/tern-tools

Key features of Tern:

- **SBOM generation for containers:** Generates SBOMs for container images to provide transparency into the software components and their metadata, aiding in security, compliance, and management efforts.
- **Layer-by-layer analysis:** Executes an analysis of each container image layer, starting from the base layer upwards, gathering information on distribution type, package formats, and package managers.
- **Command library scripts:** Utilizes a library of command scripts executed in a chroot environment to extract details on packages installed within each layer.
- **Diverse report formats:** Supports multiple output formats including human-readable text, JSON, HTML, YAML, SPDX (tag-value and JSON), and CycloneDX JSON, catering to various needs for SBOM consumption and integration.
- **Extensible with plugins:** Offers extensions like Scancode for license analysis and cve-bin-tool for vulnerability scanning, allowing for a more comprehensive container analysis.
- **Integration-ready for CI/CD pipelines:** Facilitates integration into CI/CD workflows, including a GitHub Action for automated container image scanning.

Microsoft SBOM Tool



The SBOM Tool, an open source tool created by Microsoft, is an enterprise-level, scalable tool designed to generate SPDX 2.2 compatible Software Bills of Materials (SBOMs) for a range of artifacts. It utilizes Component Detection libraries for component identification and the ClearlyDefined API for populating license information.

GitHub repo: https://github.com/microsoft/sbom-tool

Key features of the SBOM tool:

- **SPDX 2.2 compatibility:** Generates SBOMs that are compatible with the SPDX 2.2 standard, enabling interoperability across different platforms and tools.
- **Comprehensive artifact coverage:** Capable of creating SBOMs for a diverse array of artifacts, enhancing visibility and management of software components and their licenses.
- **Component detection:** Employs libraries to automatically detect software components, streamlining the SBOM generation process.
- **License information with ClearlyDefined API:** Integrates with the ClearlyDefined API to populate license information for detected components, aiding in compliance and risk management.
- **Docker image support:** Offers instructions for building the SBOM tool as a docker image, facilitating use in containerized environments and ensuring compatibility with modern CI/CD pipelines.

How to Choose SBOM Generation Software

Here are key considerations for choosing an SBOM generation solution:

- **Standards compatibility:** Ensure the tool aligns with standards like SPDX, CycloneDX, or SWID. This guarantees interoperability and adherence to best practices in generating and using SBOMs.

- **Language and ecosystem support:** Choose software that is compatible with the programming languages, package managers, and development ecosystems used in your projects. This ensures comprehensive and accurate component analysis.
- **Integration and workflow compatibility:** The software should integrate smoothly into your CI/CD pipelines, development workflows, and existing security tools. This enables real-time alerts and seamless vulnerability management.
- **Depth of analysis:** Look for tools offering deep dependency inspection to uncover transitive dependencies and component evidence, providing a full picture of your software's security posture.
- **Scalability:** Verify that the software can handle your organization's current and future scale, whether it involves single applications or complex, multi-repository projects.
- **Reporting and insights:** Choose software that provides meaningful reports and alerts, tailored to your compliance and security requirements, including licensing information and detailed component metadata.
- **Automation and efficiency:** Tools with automation capabilities, such as automatic component detection or plugin extensions, can accelerate SBOM generation and keep it up-to-date.
- **Security features:** Look for features like signed attestations and integration with vulnerability scanners to ensure integrity, trustworthiness, and comprehensive risk assessment.

Add SBOMs to Cloud Native Security with Aqua

[please add product content - the old product section from the SBOM article appears to be outdated]