SEAL
SECURITY ALERTING

SEAL APPLICATION - SBOM MODULE

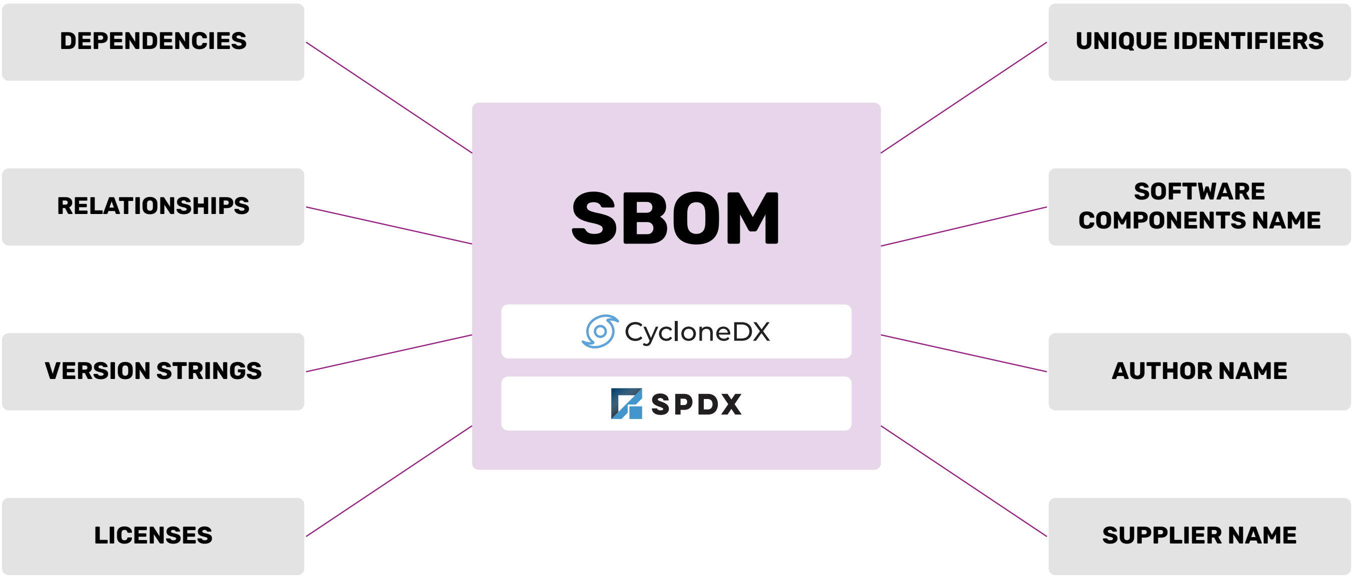
„Develop and use secure software without vulnerabilities.“

The **SBOM (Software Bill of Materials)** module of the SEAL application introduces a modern approach to software inventory management. **This module enables suppliers and users of applications to gain a deeper understanding and control over the software components they use.** This facilitates the efficient management of security risks and increases the transparency of the entire software supply chain.

Today's software typically utilizes hundreds of third-party libraries, within which issues and vulnerabilities are continuously discovered, and new versions are released. Manually monitoring this without the aid of tools is nearly impossible. A well-known example is the Log4j library and its Log4Shell vulnerability. The SBOM module in the SEAL application allows for **easy, automatic, and regular checks to determine if any of the libraries used in the application contain vulnerabilities**. If a vulnerability is discovered, an email notification can be sent. The application provides details, whether new, patched versions are available, allows for comments on vulnerabilities, etc.

Main advantages and features of the SBOM module include:

- **Automated import of SBOM documents:** Enables easy import of SBOM documents through a web interface or HTTP API.
- **Vulnerability assessment:** Regular automatic evaluation of component vulnerabilities from publicly available sources, and the option for manual impact analysis by development teams.
- **Customizable email notifications:** Informs users or development teams about new vulnerabilities, enabling quick response and problem resolution.
- **Increased transparency and trust:** Ability to share SBOM documents and vulnerability impact analyses with end-users. Selected project versions can be published using a unique link, including information on components, vulnerabilities, and their impact assessment.
- **Integration with CI/CD solutions:** Supports integration with existing Continuous Integration and Continuous Delivery processes, automating and reducing the workload on development teams.
- **Management of multiple applications and projects:** Ability to manage various projects, their versions, and set necessary access permissions for team members.
- **Flexible role and permission system:** Custom roles with varying permission levels can be created and assigned to users. This allows the SBOM module to serve larger development teams where different groups work on different projects without everyone seeing everything.



The module not only provides an overview of the used components and their security status but also tools for their effective management and mitigation of potential risks, thereby increasing the overall security and reliability of delivered applications.

UTILIZATION OF THE SBOM MODULE

FOR APPLICATION SUPPLIERS (DEVELOPMENT COMPANIES)

The average number of components in current applications is in the order of hundreds, primarily due to recursive dependencies (so-called indirect dependencies). Monitoring the security of such a high number of components without a specialized Software Component Analysis (SCA) solution like the SBOM module of the SEAL application is almost impossible.

The SBOM module allows **importing of existing SBOM documents and monitoring of component vulnerabilities across all applications of the organization**. The application evaluates component vulnerabilities from publicly available data at regular intervals, allowing for subsequent manual impact analysis by the application's development team.

With the help of **HTTP API, existing Continuous Integration (CI) / Continuous Delivery (CD) solutions can be integrated**, automatically uploading current versions of SBOM documents immediately after each build. The application only works with the SBOM document and **does not require access to the application's source code**.

It provides a single location to get an overview of the status of all components across applications, aiding development teams in maintaining a current and secure state of applications.

Development team notifications about new vulnerabilities are automated through customizable email alerts, eliminating the need for regular manual checks of component status in the application or monitoring of repositories, pages and discussion forums for used components.

Utilizing licensing policies can help capture a component with an inappropriate software license (e.g., copyleft like GPL).

The SBOM document can be published or shared with the customer, including the conducted impact analysis, thus increasing the transparency and trustworthiness of the delivered application.

FOR APPLICATION USERS (END CUSTOMERS)

Importing SBOM documents of used applications provides an **overview of the software components** used across the organization, **identifying the presence of vulnerable components** in the organization's infrastructure in time. This reverses the traditional dependency of the consumer on the supplier – in this case, the application user can learn about vulnerabilities earlier than the supplier and can shut down the affected applications or tighten access to them using a firewall.

IT administrators automatically learn about new vulnerabilities through customizable email alerts without having to regularly check the component status in the SBOM module.

APPLICATION DEVELOPMENT AND THE RELATIONSHIP WITH SBOM AND VEX

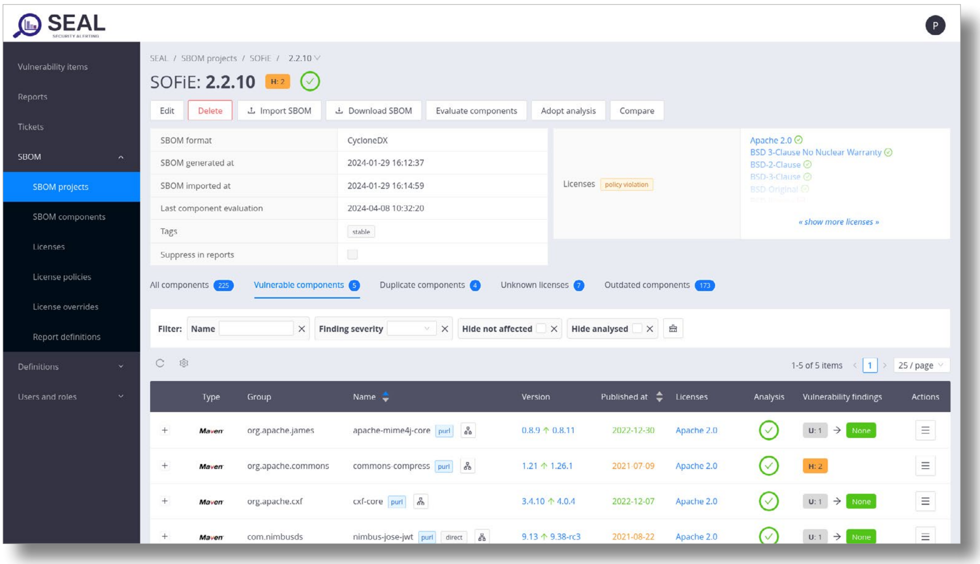
USE OF OPEN-SOURCE LIBRARIES IN APPLICATIONS

According to a study by Synopsis, practically all current applications incorporate freely available open-source components in the form of packages or libraries. Most contain at least one vulnerable component, and nearly half contain at least one component with a high security risk.

Applications are delivered to consumers as closed, complete solutions, and the user is unaware of the components contained and their security risks. In the case of widely known vulnerabilities (e.g., Log4Shell), application suppliers are overwhelmed with requests for statements on whether their software contains the specific vulnerable component – even in cases where the application runs on a completely different technological platform.

SBOM AND VEX

The time between the discovery and exploitation of vulnerabilities has significantly decreased in recent years, making it more necessary than ever to monitor the vulnerability of used components in real-time. This process must not be limited to application suppliers but must also be possible for their users. The so-called SBOM document (Software Bill Of Materials), a machine-readable list of all components used in an application, aids in this.



SBOM format	CycloneDX
SBOM generated at	2024-01-29 16:12:37
SBOM imported at	2024-01-29 16:14:59
Last component evaluation	2024-04-08 10:32:20
Tags	stable
Suppress in reports	□

Licenses policy violation

Apache 2.0 ✓
BSD 3-Clause No Nuclear Warranty ✓
BSD-2-Clause ✓
BSD-3-Clause ✓
BSD Original ✓
« show more licenses »

	Type	Group	Name	Version	Published at	Licenses	Analysis	Vulnerability findings	Actions
+	Maven	org.apache.james	apache-mime4j-core purl	0.8.9 ↑ 0.8.11	2022-12-30	Apache 2.0	✓	U: 1 → None	☰
+	Maven	org.apache.commons	commons-compress purl	1.21 ↑ 1.26.1	2021-07-09	Apache 2.0	✓	H: 2	☰
+	Maven	org.apache.cxf	cxf-core purl	3.4.10 ↑ 4.0.4	2022-12-07	Apache 2.0	✓	U: 1 → None	☰
+	Maven	com.nimbusds	nimbus-jose-jwt purl direct	9.13 ↑ 9.38-rc3	2021-08-22	Apache 2.0	✓	U: 1 → None	☰

The concept of the SBOM document was initiated in 2018 by the American federal agency NTIA, and later, oversight was taken over by the American federal agency for cybersecurity, CISA. Within CISA, the concept of the so-called VEX document (Vulnerability Exploitability Exchange), a machine-readable document containing the application supplier's statement on component vulnerability, was also developed. **In 2021, the American federal government issued the "Improving the Nation's Cybersecurity" (14028) order, which includes the recommendation to supply SBOM documents for all software.**

Currently, the most used implementations of the SBOM document are CycloneDX and SPDX. For VEX documents, CycloneDX and OpenVEX are used. Both implementations have commercial and freely distributable tools for creating SBOM documents from application source codes.

VEX documents are the result of vulnerability analysis of components and are created by SCA tools (Software Component Analysis) like the SEAL application and SBOM module. The analysis must be performed manually, carefully considering all possible implications.

SBOM AND REGULATORY REQUIREMENTS

EU NIS2 DIRECTIVE

According to Article 21 (Cybersecurity risk-management measures), essential and important entities must take appropriate and proportionate technical, operational, and organizational measures to manage the risks posed to the security of networks and information systems.

The measures shall be based on an "all-hazards approach" that aims to protect network and information systems and the physical environment of those systems from incidents and shall include "at least" the following:

(d) supply chain security, including security-related aspects concerning the relationships between each entity and its direct suppliers or service providers;

(e) security in network and information systems acquisition, development and maintenance, including vulnerability handling and disclosure;

https://www.nis-2-directive.com/NIS_2_Directive_Article_21.html

US GOVERNMENT – VIA CISA

The Executive Order on Improving the Nation's Cybersecurity was issued by the US government in May 2021, emphasizing the importance of SBOM in enhancing the security of the software supply chain.

https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/

https://www.cisa.gov/sbom

Cloud service SaaS

FREE TRIAL Request a FREE TRIAL