

# CAST SBOM Manager

## Installation & User Guide

**Version:** 2.0.0

**Date:** July, 2024

# Contents

Introduction .....	4
Pre-requisites .....	4
Overview of the Process .....	5
1. Installation.....	5
Windows installation .....	5
Linux installation.....	6
2. Application Initialization.....	7
3. Creating a SBOM .....	9
Step 1 – Details.....	9
Step 2 – Files.....	10
Step 3 – Packages .....	11
Step 4 – Scanners .....	12
Step 5 – Summary .....	13
4. Browsing and Editing a SBOM.....	14
Home Screen .....	14
Projects .....	15
Bill of Materials.....	16
Dashboards .....	17
Components.....	18
File Map.....	20
SBOM Export .....	21
Catalog .....	22

Component Categories.....	23
Propagate component changes in SBOMs .....	25
Find component usage across SBOMs.....	25
Licenses .....	26
Create/Edit a license .....	27
Create/Edit a license policy.....	29
Vulnerabilities .....	31
Edit a vulnerability .....	33
Find vulnerable SBOMs.....	34
Log a vulnerability .....	35
Preferences & Configuration.....	36

## Introduction

CAST SBOM Manager is a complementary product to CAST Highlight dedicated to Software Composition Analysis (SCA) and Bill of Materials (SBOM) management. This new product is designed to facilitate the following use cases:

1. Enhance SBOM reviews and documentation by recording user modifications (e.g., choice of a license for a dual-licensed component, internal component reviews, etc.) and retrieving this metadata for future SBOMs.
2. Cataloguing and identification of proprietary components (i.e., developed in house) to automatically detect them along with their metadata (custom licenses, copyright information, etc.) in future SBOMs.

## Pre-requisites

CAST SBOM Manager has the following pre-requisites. Please ensure your machine meets these requirements prior to installing and running the product.

- Internet Access: Since the product interacts with CAST Highlight's SCA database, the machine where the product will be running requires access to the Internet.
- Operating System: Windows (10 preferred), Linux
- Java Runtime Environment: JRE 11 (required for Linux)
- Local Port 9001 should be available
- RAM/Memory: 16GB recommended
- Storage Drive Space: 200MB minimum

## Overview of the Process

Following is a general overview of each major step in the process for using the SBOM Manager:

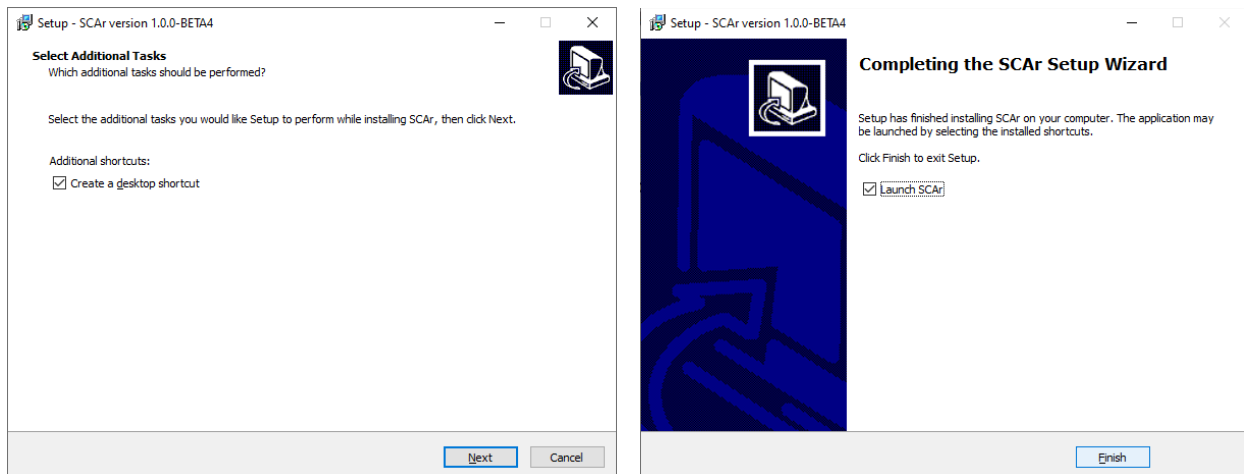
1. Installation
2. Application Initialization
3. Creating a SBOM
4. Browsing and editing a SBOM

## 1. Installation

### Windows installation

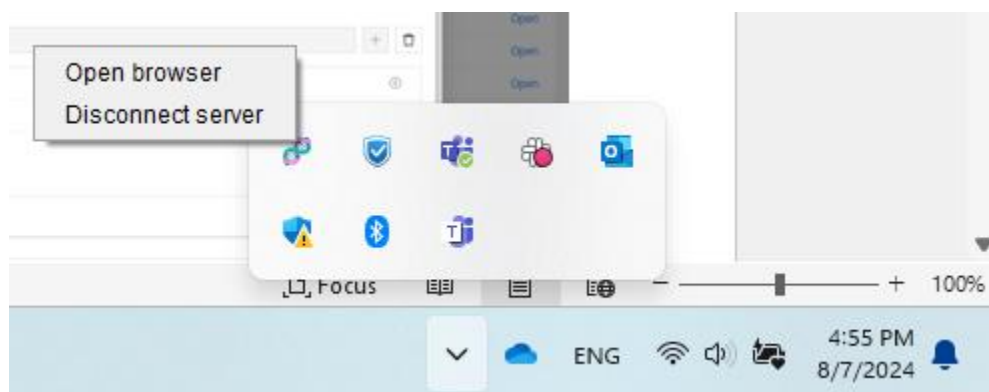
To install the SBOM Manager, simply follow the steps below:

- Execute the installer .exe (e.g., sbom-manager-2.0.0-RC4-setup.exe)
- Select your preferred language and other preferences for the installation



- Once the installation is finished, a browser automatically opens the application for the initialization step. If the browser doesn't open automatically, go directly to <http://localhost:9001>

- If the browser doesn't open automatically, open the system tray icons, right click on the SBOM Manager icon and click on "Open browser"



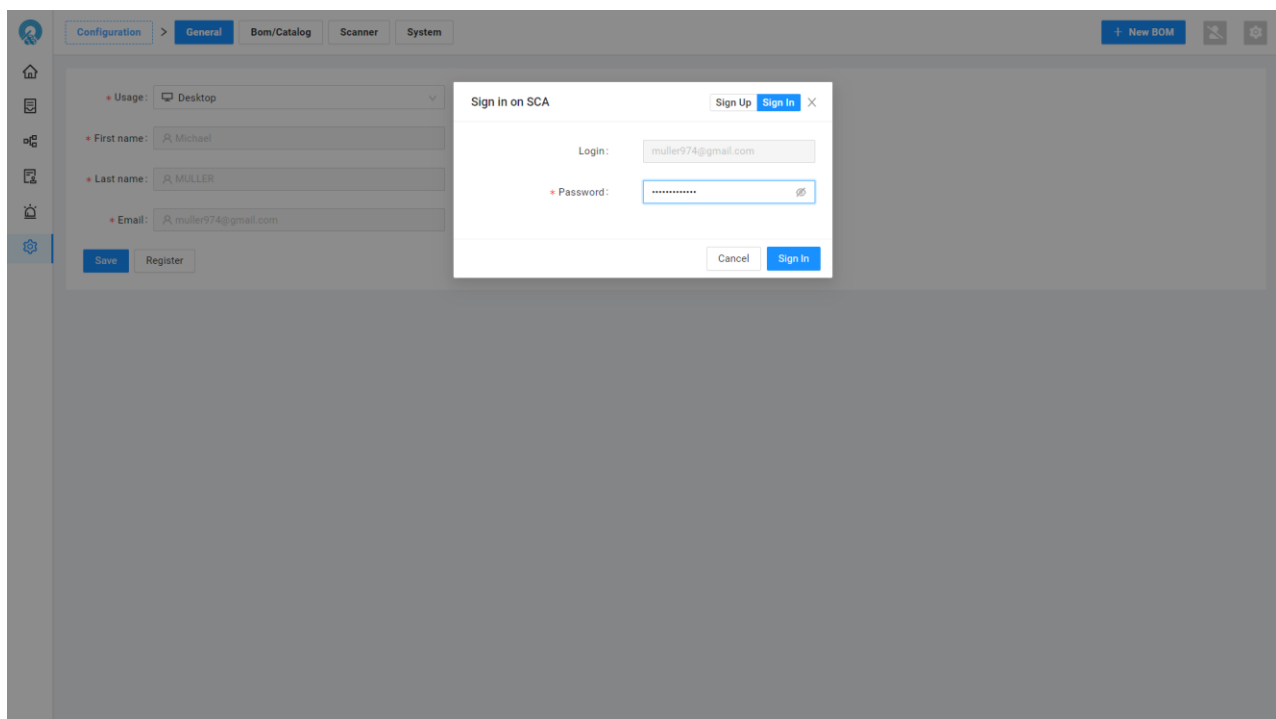
- By default, the application uses port 9001. If there is an issue, first check for a possible conflict on this port from another application running on the machine.
- If this port is already in use by another application, you can change it by following these steps (under Windows):
  - locate the sbom-manager.exe folder location (generally %USER%\AppData\Local\Programs\s bom-manager\)
  - create a file called `sbom-manager.l4j.ini` next to `sbom-manager.exe`
  - Add into it the line: `-Dserver.port=9091` (choose the port you prefer)

## Linux installation

- Download the SBOM installation ZIP file
- Ensure GUI/XDRP is installed/enabled
- Run the following commands
  - `unzip sbom-manager-2.0.0-RC3-linux.zip -d sbom-manager-2.0.0-RC3-linux`
  - `cd sbom-manager-2.0.0-RC3-linux/bin`
  - `chmod 755 *` *## make sure sbom-manager.sh have execute permission*
  - `./sbom-manager.sh` *##will launch the sbom manager process*

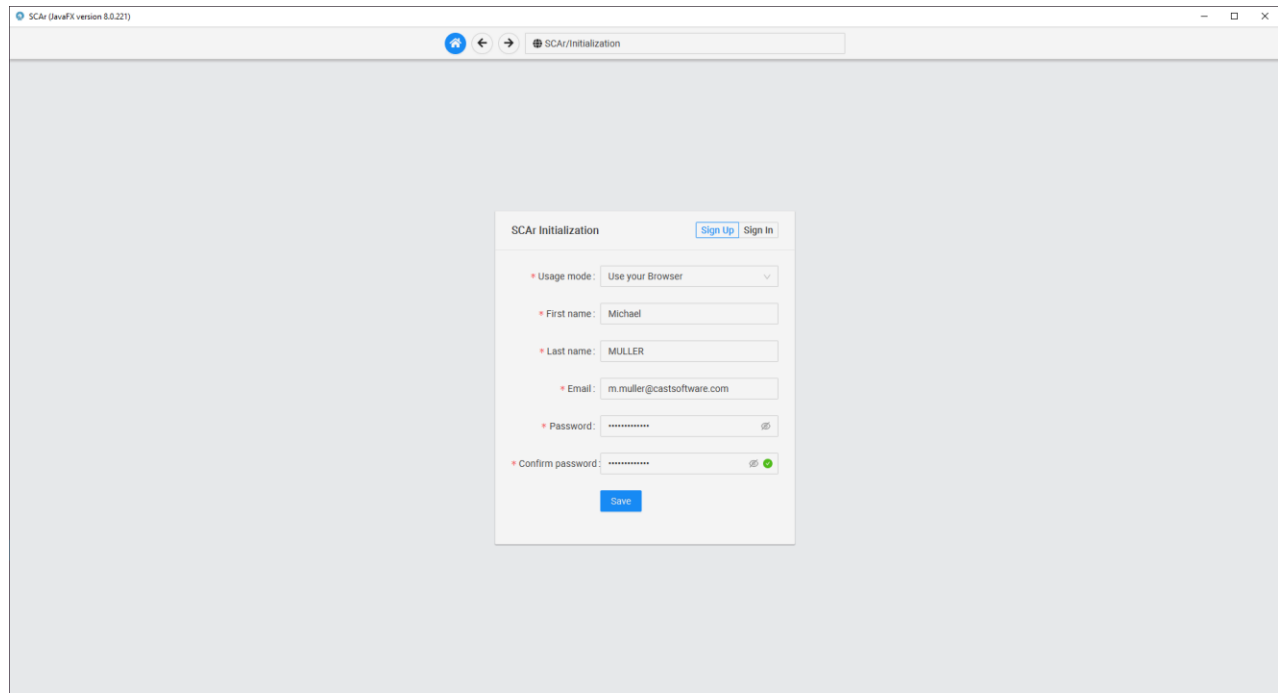
- `ps -ef|grep som` ##check if process(/usr/bin/java -Xms2048m -Xmx8192m -jar /home/hluser/sbom-manager-2.0.0-RC3-linux/bin/../lib/sbom-manager.jar) running
- `netstat -plten|grep 9001` ##process is running on 9001 port
- Open the browser and type in <https://localhost:9001>

## 2. Application Initialization



To initialize the application and set up your user account, provide the following information:

- First name and last name: this information remains local to your machine.
- E-mail address: This should be a valid professional e-mail address.
- Password: Define your password that will be associated with your e-mail address to sign in.

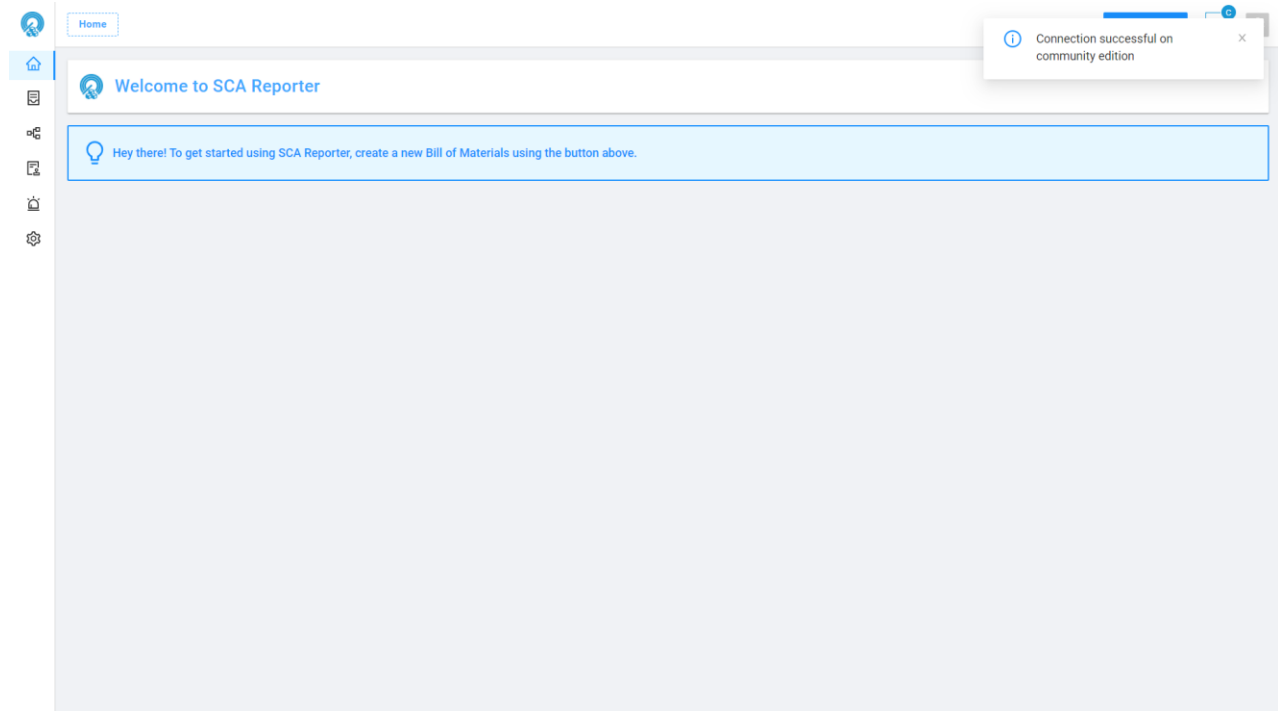


The screenshot shows a web browser window titled "SCAr (JavaFX version 8.0.221)" with the address bar displaying "SCAr/Initialization". The main content area features a form titled "SCAr Initialization" with a "Sign Up" button and a "Sign In" button. The form contains the following fields:

- Usage mode: A dropdown menu set to "Use your Browser".
- First name: A text input field containing "Michael".
- Last name: A text input field containing "MULLER".
- Email: A text input field containing "m.muller@castsoftware.com".
- Password: A password input field with a strength indicator icon.
- Confirm password: A password input field with a strength indicator icon.

A "Save" button is located at the bottom of the form.

Once you click on “Save”, the screen closes and automatically launches the application after a couple seconds.





### 3. Creating a SBOM

To create a SBOM, click on the “+ New SBOM” button on top right of the screen. A wizard opens to let you specify required information and options.

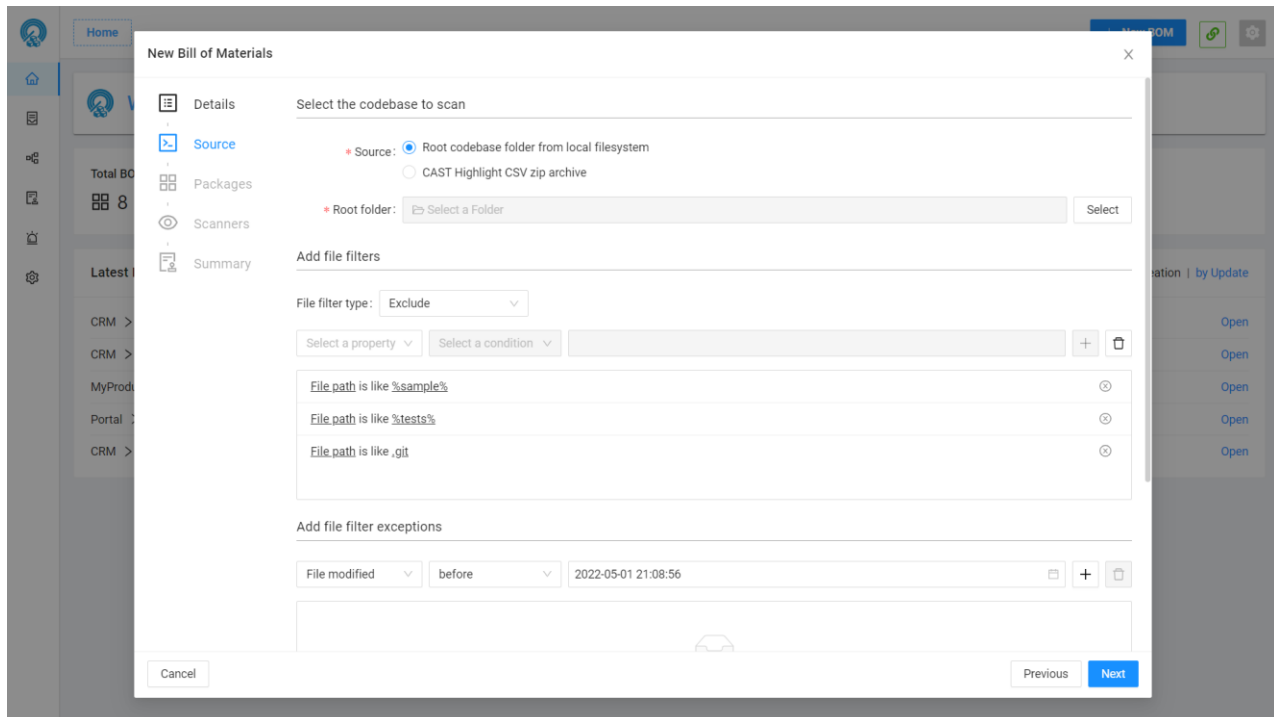
The screenshot shows the 'BOM Creation' wizard in the CAST SBOM Manager application. The wizard is a modal dialog with a sidebar on the left containing five steps: 1 Details, 2 Files, 3 Packages, 4 Scanners, and 5 Summary. The 'Details' step is currently selected. The main content area is titled 'Name your Bill of Materials' and contains three input fields: 'Product' with the value 'MyCustomApp', 'Project' with the value 'API', and 'Version' with the value '1.0.0'. Below these fields is a section titled 'Choose your license policy' with a dropdown menu. The dropdown is open, showing three options: 'My Policy' (which is highlighted), 'GPL v3 compliance', and 'ASL 3rd party compliance'. At the bottom of the dropdown, there is a small button labeled 'My Policy'. At the bottom of the wizard, there is a blue informational bar that reads: 'The following steps will guide you through the process of creating a new Bill of Materials. This help panel provides additional guidance/context for each of these steps.' Below this bar are 'Cancel' and 'Next' buttons.

#### Step 1 – Details

Provide product, project, and version names.

Choose the license policy you want to apply for this SBOM. See how to create/edit a license policy later in this document.

Click on “Next” to go to the next step.



## Step 2 – Files

Specify the root folder where source code of the application is located on your machine.

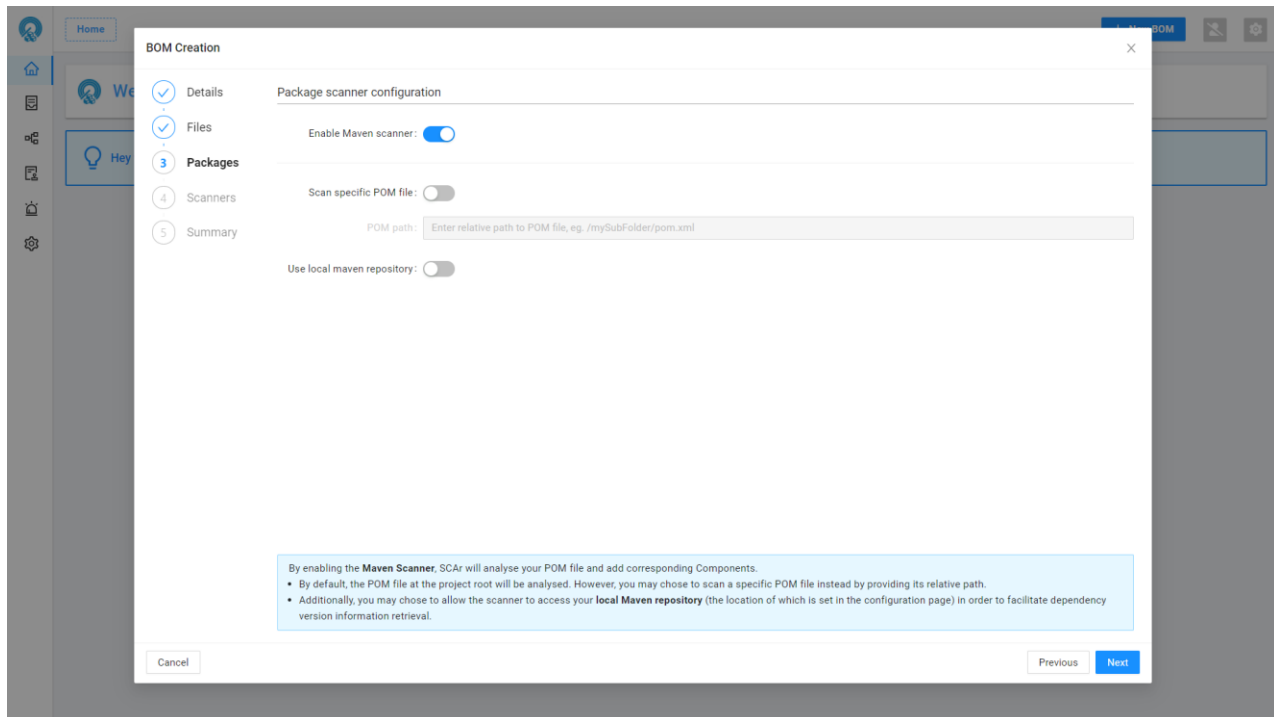
From this step, you can also specify file exclusion or inclusion rules, based on:

- File name
- File path
- File size
- File modified date

File name and path filters use standard Shellsript pattern matching:

- Multiple patterns can be separated by semi-colons
- \* matches 0 or more characters
- ? matches precisely one character
- [!abcABC] matches any character except a,b,c and their uppercase counterparts

Alternatively, you can specify a result .zip file produced by CAST Highlight's agent which contains result CSV files for an application already analyzed by CAST Highlight.



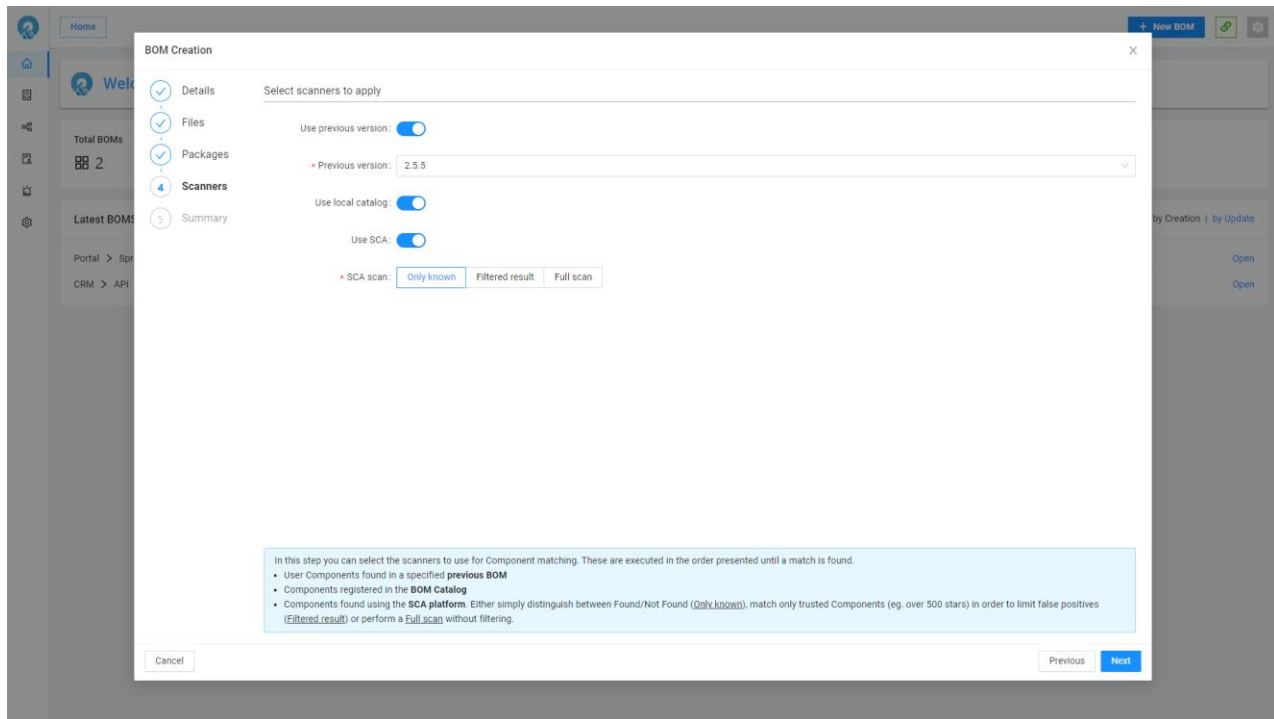
## Step 3 – Packages

For Maven-based applications, you can enable the Maven scanner which will detect components referenced in your pom.xml.

By enabling the Maven Scanner, the SCAR SBOM Manager will analyze your POM file and add corresponding Components.

- By default, the POM file at the project root will be analyzed. However, you may choose to scan a specific POM file instead by providing its relative path.
- Additionally, you may choose to allow the scanner to access your local Maven repository (the location of which is set in the configuration page) to facilitate dependency version information retrieval.

## Step 4 – Scanners

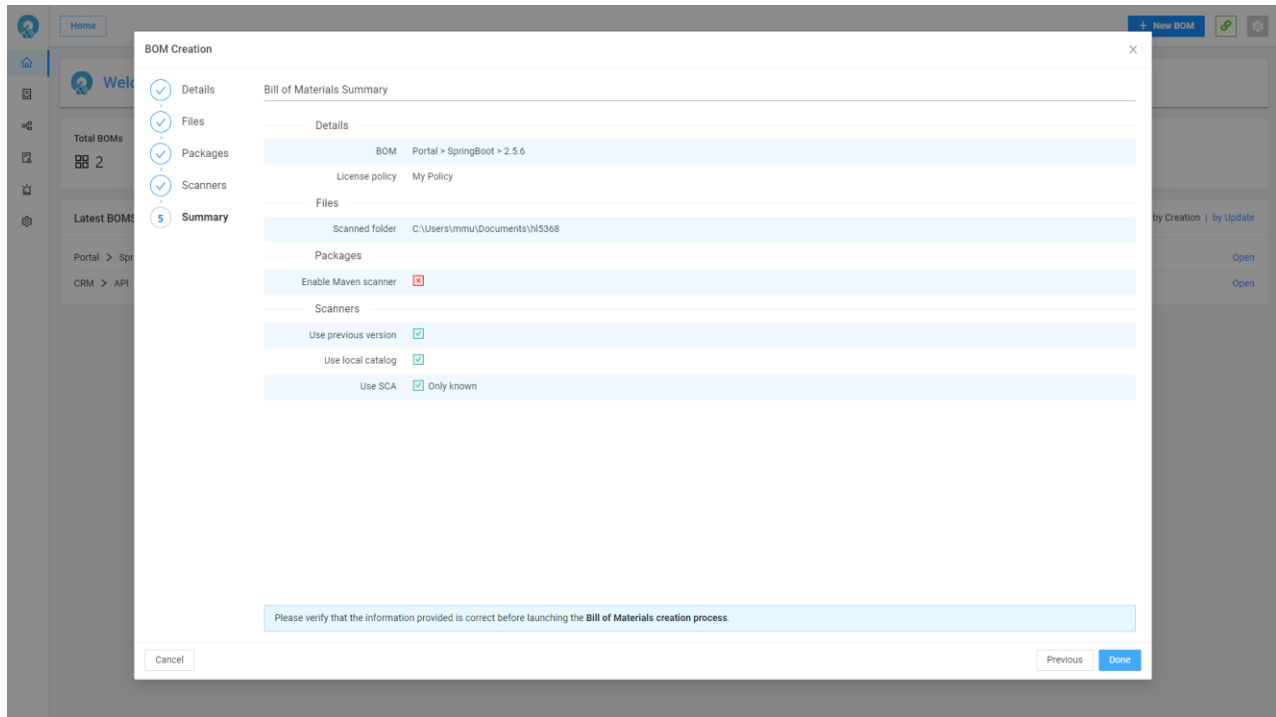


In this step you can select the scanners to use for Component matching. These are executed in the order presented until a match is found.

- **Use previous version:** Scans for components included in a specified previous SBOM (select the desired version in the “Previous Version” field)
- **Use local catalog:** Scans for components already existing from previous scans and registered manually in the SBOM Catalog previously
- **Use SCA:** Scans for components in the CAST Highlight component knowledgebase and allows user to set a filter option:
  - Only known: this mode will let you know whether scanned files are known vs. not known in CAST’s SCA knowledgebase
  - Filtered Result: this mode will retain component identifications that correspond to specific filters (e.g., at least 500 stars, exclusion of file fingerprints below 100 bytes, etc.) to reduce noise
  - Full scan: all file fingerprints will be processed for component identification, without any filter

## Step 5 – Summary

From this step, review the configuration that was set up for the scan. To change it, you can go back to previous steps by clicking on the “Previous” button at the bottom of the screen.



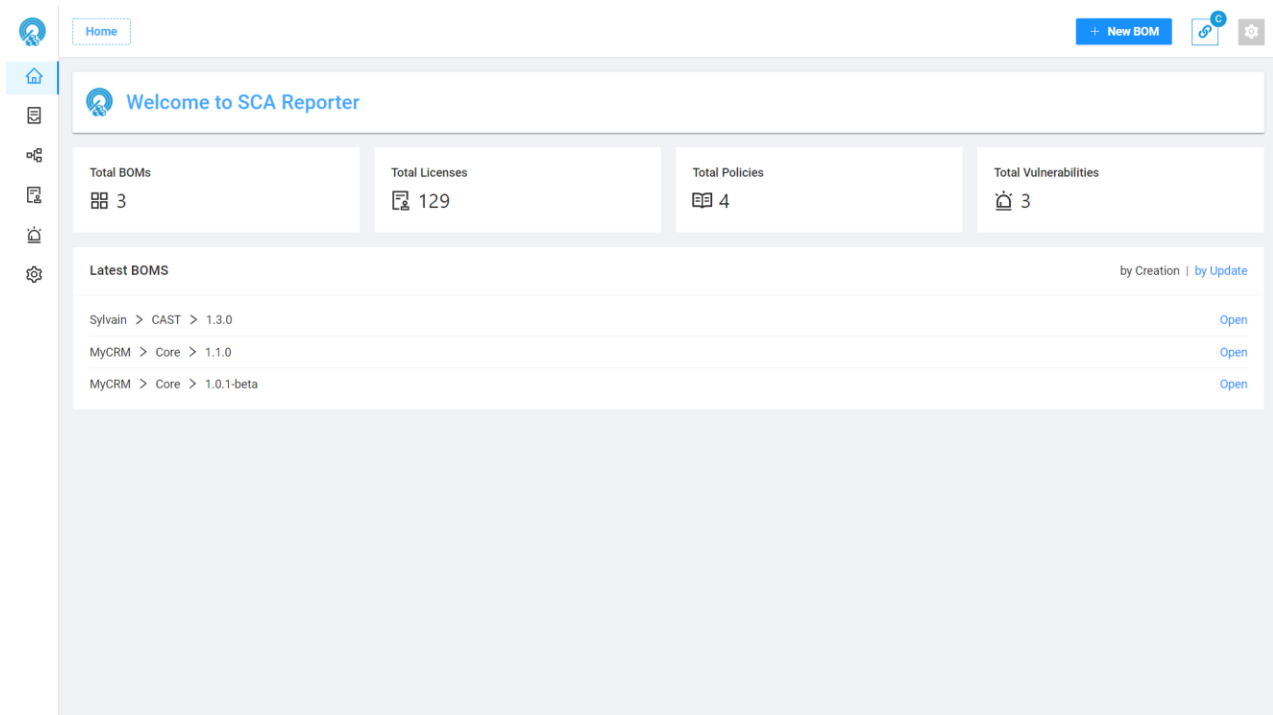
## 4. Browsing and Editing a SBOM

The sections below detail the primary product capabilities.

### Home Screen

From the homepage, you can retrieve the key information for your SBOMs:

- Total SBOMs created
- Total licenses detected in SBOMs
- Total license policies available
- Total vulnerabilities detected in SBOMs
- A list of recent SBOMs for quick access to a specific SBOM



## Projects

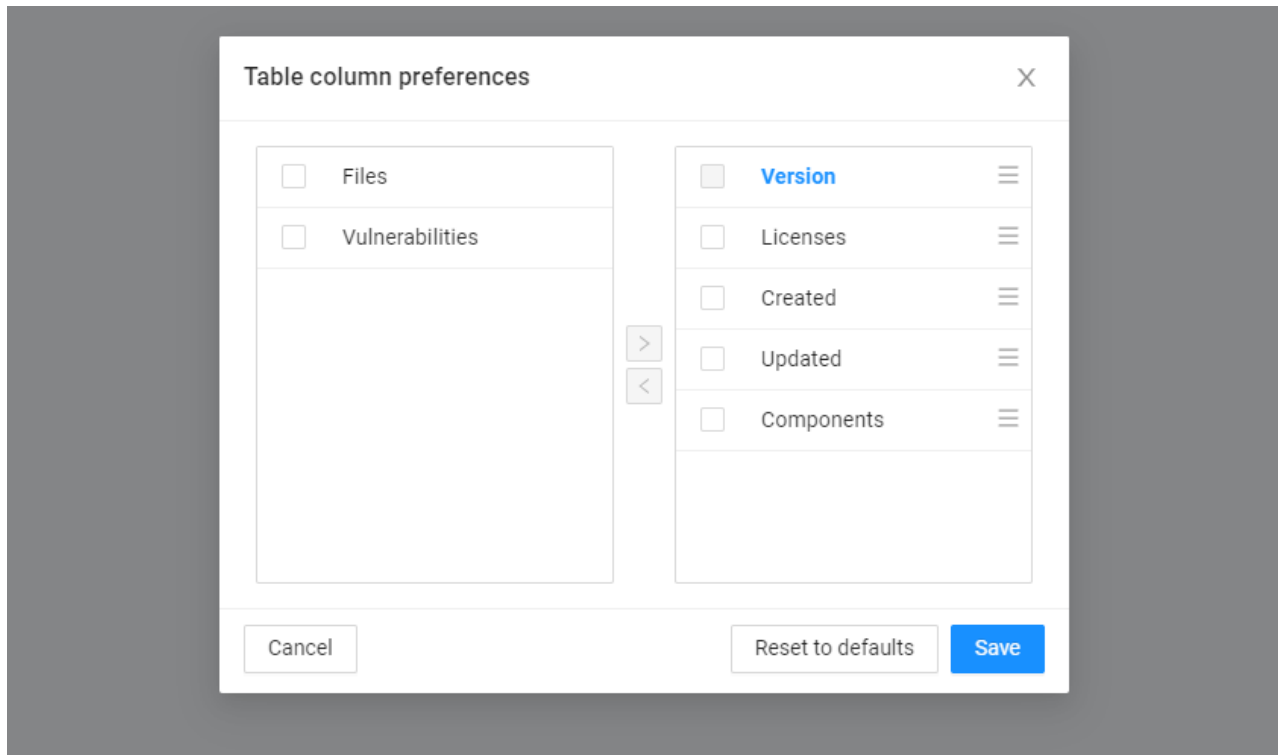
In this “Projects” view, you can retrieve all the SBOMs which have been created, organized as a tree with the following structure:

- Product
  - Project
    - Version

This structure allows users to organize SBOMs in a flexible way. For instance, a product can be an application and a project can be a specific component of an application.

Clicking on a specific project will display the list of versions with corresponding SBOMs available along with some high-level metrics such as the number of scanned files, number of third-party components detected, number of distinct licenses, and date of creation/update.

You can customize this view by showing/hiding some columns of the table. To do so, click on the column icon on top right of the table and configure the columns.



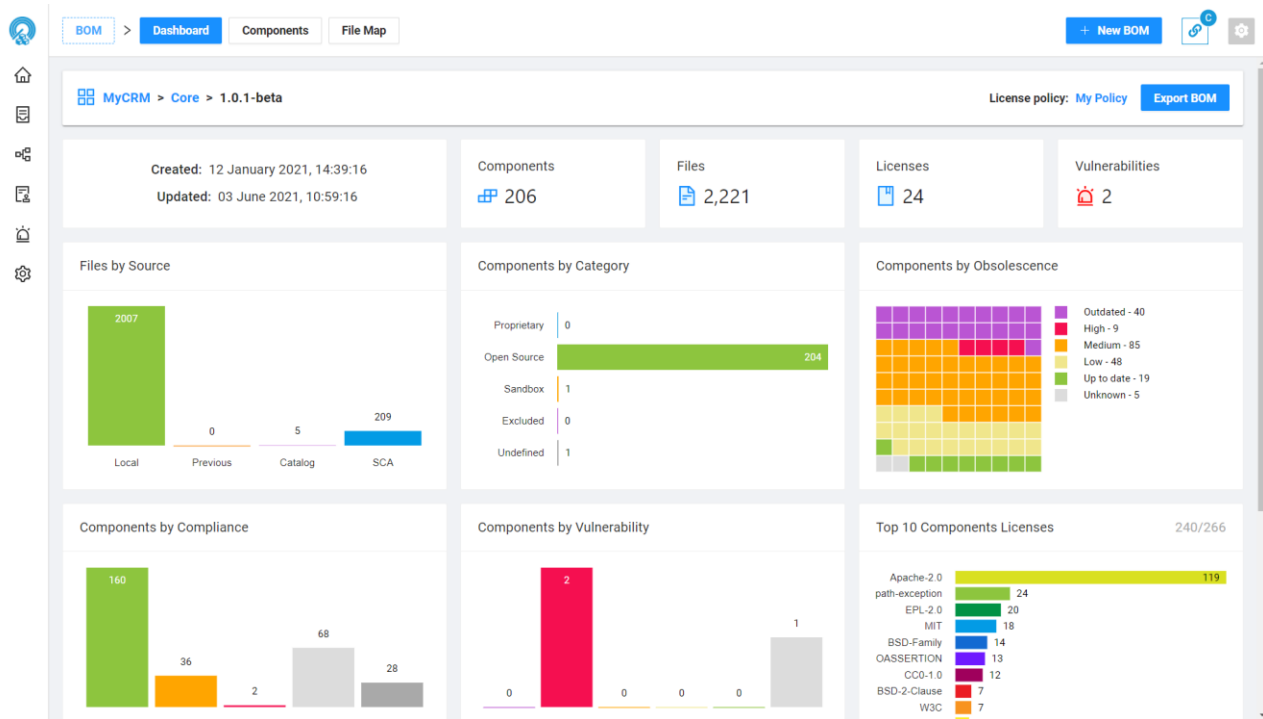
You can also delete a SBOM by clicking on the red trash icon. Finally, you can open the SBOM by clicking on the blue arrow in the last column.

## Bill of Materials

In CAST Highlight's SCAR SBOM Manager, the bill of materials is the central place from which you can see and manipulate scan results with regards to components, license policies, obsolescence, and vulnerabilities.



## Dashboards



This page summarizes the different KPIs and metrics of a SBOM:

- Create and update dates
- Number of components detected
- Number of files scanned
- Number of licenses detected
- Number of vulnerabilities detected
- Distribution of files by origin (i.e. whether they've been associated with a component from a previous scan, the component catalog, the SCA database or local)
- Distribution of components by category (i.e. proprietary, Open Source, excluded, etc.)
- Distribution of component by obsolescence level, based on the detected version and the latest known version of each component
- Distribution of components by license compliance

- Distribution of components by vulnerability level (e.g. critical, high, medium, etc.)
- Distribution of components by license (e.g. MIT, Apache-2.0, etc.)
- Distribution of components by programming language (Java, Cobol, etc.)
- Distribution of components by topics (e.g. html5, devops, Angular, visualization, etc.)
- Distribution of components by file extensions of scanned files

## Components

	Name	Versions	Latest version	Obsolescence	Vulnerabilities	Licenses	Files	Path	
<input type="checkbox"/>	Piero_93/progetto-prb-ambo-raspi			Unknown			1	/Hygieia-master/Ui-tests/	actions
<input type="checkbox"/>	gridstack	0.2.6	4.2.6	High		MIT	1	/Hygieia-master/.../etc/	actions
<input type="checkbox"/>	tinymce/tinymce	2.0.7	5.8.1	High	CVE-2020-17468		1	/Hygieia-master/.../dashboards/	actions
<input type="checkbox"/>	CapitalOneCanada/capitalonecanada.github.io			Unknown			2	/Hygieia-master/.../src/	Show Files
<input type="checkbox"/>	codeheroics/nope			Unknown			1	/Hygieia-master/.../fonts/	Edit Component
<input type="checkbox"/>	vburley/markdown	v0.6	v2.1	High			1	/Hygieia-master/.../fonts/	Change Category
<input type="checkbox"/>	roccojanse/efocus-boilerplate			Unknown			1	/Hygieia-master/.../fonts/	Add to Catalog
<input type="checkbox"/>	dcode-study/ethereum/truffle-react-bootstrap-box			Unknown			1	/Hygieia-master/.../fonts/	Split Components
<input type="checkbox"/>	neocities/neocities	2.0.0	3.0.0	Medium		BSD-2-Clause	4	/Hygieia-master/.../fonts/	actions
<input type="checkbox"/>	jderrick/dropwizard-metrics-cloudwatch	0.1.3	0.1.6	Outdated		Apache-2.0	1	/Hygieia-master/	actions
<input type="checkbox"/>	VirtualEngine/AppProvider	v0.6.0.4	v0.6.2.8	Outdated		Apache-2.0	1	/Hygieia-master/	actions
<input type="checkbox"/>	angular-ads	1.0.0	1.5.0	Outdated		MIT	1	/Hygieia-master/.../src/	actions
<input type="checkbox"/>	xiyouMc/ncmbot		v0.1.6	Unknown			1	/Hygieia-master/	actions
<input type="checkbox"/>	brettbouquin/maven-wrapper	maven-wrapper-0.1.3	maven-wrapper-0.4.0	High		Apache-2.0	3	/Hygieia-master/	actions
<input type="checkbox"/>	Sling-Li/bug	nodearm	nodearm	Outdated			1	/Hygieia-master/.../images/	actions
<input type="checkbox"/>	Hygieia/Hygieia	v3.1.0	v3.1.0	Up to date		Apache-2.0	471	/Hygieia-master/	actions

From this Components view, you can see the list of components that have been detected in a SBOM along with their associated information. Note that columns of this table can also be configured (hide/show).

- Name: name of the component
- Versions: detected version(s) of the component

- Latest Version: latest known version of the detected component
- Obsolescence: level of obsolescence of a given component based on the gap between the detected version and the latest known version of this component
- Vulnerabilities: list of known vulnerabilities for a given component version
- Licenses: license(s) of the component with a color indication of its compliance with the policy used for the current SBOM
- Path: path of the folder where the component has been detected
- SCA Id: technical identifier of the component as it appears in the CAST Highlight SCA database

From this view, the user can launch some actions for each component:

- Show Files: this will show the files that have been mapped to this component based on SHA256 fingerprints.
- Edit Component: this will open a modal where the user can edit information for a component (e.g. add/change a license, add a vulnerability, leave a comment, etc.) – Refer to the Component Edit section
- Change Category: this will change the category of a component based on user input. By default, known third-party components are categorized as being Open Source, unknown files fall into the “Uncategorized” category. However, users can change the category of a component such as Proprietary, Excluded, Sandbox.
- Add to Catalog: this will add the selected component to the catalog so that this component can be shared and retrieved across other SBOMs. Refer to the Catalog section.

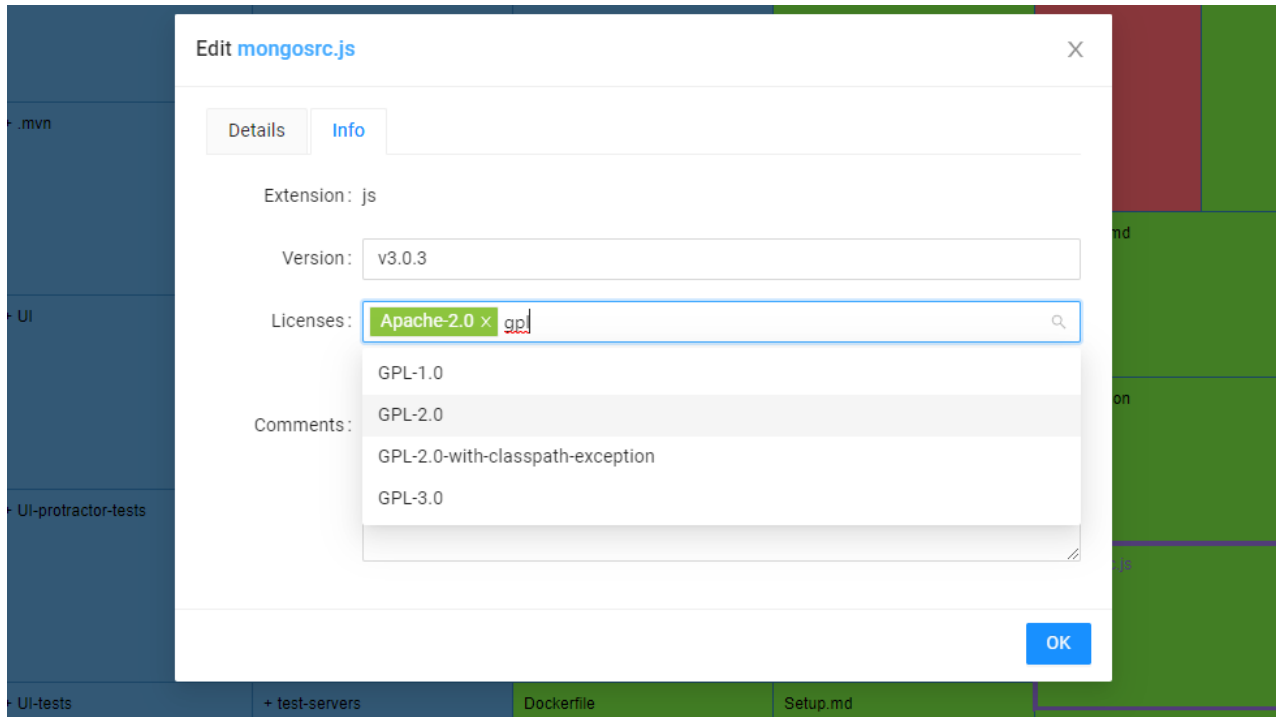
## File Map

The screenshot displays the 'File Map' view in the CAST SBOM Manager. The top navigation bar includes 'BOM', 'Dashboard', 'Components', and 'File Map'. The main area shows a hierarchical tree on the left for 'Hygieia-master' and a grid of files on the right. The grid cells are color-coded: blue for source code, green for documentation, red for license-related files, and grey for configuration files. A bottom panel shows details for 'mongosrc.js', including its path, component, and license (Apache-2.0).

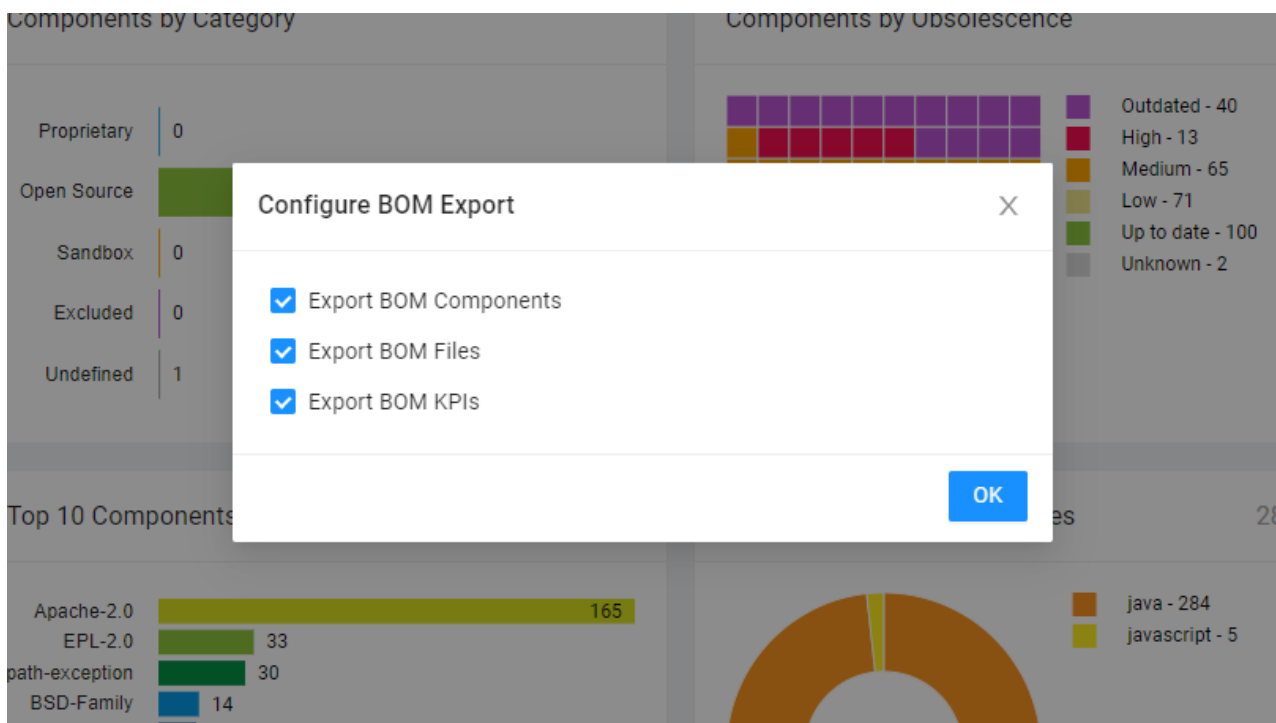
File	Component	License
Hygieia-master	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
github	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
certs	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
travis-utilities	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Hygieia2.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Ticket_26665_source	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
checkstyle.xml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
circle.yml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
docker-compose.override.yml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
docker-compose.yml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
mvn	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
db	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
gitignore	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
LICENSE	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
features.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
mvnw.cmd	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
package-lock.json	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
pmd.xml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
UI	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Hygieia2.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
MAINTAINERS	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Hygieia.json	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Hygieia2.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
README.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
CHANGELOG.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
travis.yml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
travis-utilities	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
gitignore	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
travis.yml	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
CHANGELOG.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Dockerfile	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Hygieia2.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
LICENSE	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
MAINTAINERS	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
README.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Setup.md	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)
Ticket_26665_sourceCodeWithInfractions	Hygieia-master	Hygieia/Hygieia (OPENSOURCE)

This view allows users to navigate through the folder structure of the SBOM to visualize scanned files with their corresponding component information and license compliance.

You can edit file-level information and add a comment by selecting a file and clicking on the “Edit” button.



## SBOM Export



You can export SBOM information from any view of the SBOM/Project section. To do so, click on the “Export” button on the left of the header menu and select the content to be exported:

- Component information
- File information
- KPI information

ID	Component	Version	License	Other Info
25	SCA	central	Apache-2.0	2.12.5
26	SCA	central	Apache-2.0	2.6.1
27	SCA	central	Apache-2.0	2.12.5
28	SCA	central	Apache-2.0	1.9.13
29	SCA	central	CC-BY-SA-3.0	2.0.1
30	SCA	central	LGPL-2.0	5.1.2.Final
31	SCA	central	EPL-2.0	3.27.0
32	SCA	central	BSD-Family	1.06
33	SCA	central	Apache-2.0	2.5.0
34	SCA	central	MIT	3.18.1
35	SCA	central	BSD-Family	7.1
36	SCA	central	Apache-2.0	2.5.0
37	SCA	central	CC-BY-4.0   MIT   OFL-1.1	5.15.4
38	SCA	central	Apache-2.0	6.2.4
39	SCA	central	Apache-2.0   SAX-PD   W3C	1.4.01
40	SCA	central	BSD-3-Clause	2.3.5
41	SCA	central	Apache-2.0	1.5.24
42	SCA	central	Apache-2.0	2.1.0.RELEASE
43	SCA	central	Apache-2.0	5.3.10
44	SCA	central	Apache-2.0	5.5.4
45	SCA	central	Apache-2.0	5.3.10
46	SCA	central	Apache-2.0	5.3.10
47	SCA	central	Apache-2.0	5.14
48	SCA	central	BSD-2-Clause	42.2.20
49	SCA	central	CC-0	1.0.3
50	SCA	central	Apache-2.0	1.5.1
51	SCA	central	NOASSERTION   Apache-2.0	1.2
52	SCA	central	Apache-2.0	3.14.15
53	SCA	central	Apache-2.0	3.14.15
54	SCA	central	NOASSERTION   Apache-2.0	2.2.14
55	SCA	central	Apache-2.0	3.14.15
56	SCA	central	EPL-2.0   GPL-2.0-with-classpath-exception	2.33
57	SCA	central	LGPL-2.1   MPL-1.1   Apache-2.0	3.27.0-GA
58	SCA	central	CDL-1.1   GPL-2.0-with-classpath-exception	2.3.1
59	SCA	central	GPL-2.0-with-classpath-exception	1.0
60	SCA	central	GPL-2.0-with-classpath-exception	1.6.7

The SBOM export is currently available in Excel format.

## Catalog

In the SCAR SBOM Manager, the catalog is a central repository which allows the sharing of components across SBOMs. Using the catalog will help users capitalize on the qualification effort spent for each SBOM, including for managing proprietary components, component exclusions, component approval processes, etc.

	Name	Versions	Latest version	Obsolescence	Vulnerabilities	Licenses	SCA Id	actions
<input type="checkbox"/>	com.thoughtworks.xstream.xstream-jmh	1.4.18	1.4.18	Up to date		BSD-3-Clause	com.thoughtworks.xstream.xst	actions
<input type="checkbox"/>	org.springframework.spring-context	5.3.10	5.3.12	Low		Apache-2.0	org.springframework.spring-cor	actions
<input type="checkbox"/>	org.springframework.boot.spring-boot-autoconfigure	2.5.5	2.5.6	Up to date		Apache-2.0	org.springframework.boot.sprin	actions
<input type="checkbox"/>	org.springframework.security.extensions.spring-security-saml2-core	1.0.3.RELEASE	1.0.10.RELEASE	Low		Apache-2.0	org.springframework.security.e	actions
<input type="checkbox"/>	org.springframework.integration.spring-integration-jmx	5.5.4	5.5.5	Up to date		Apache-2.0	org.springframework.integration	actions
<input type="checkbox"/>	org.springframework.integration.spring-integration-http	5.5.4	5.5.5	Up to date		Apache-2.0	org.springframework.integration	actions
<input type="checkbox"/>	org.apache.velocity.velocity	1.7	1.7	Outdated		Apache-2.0	org.apache.velocity.velocity	actions
<input type="checkbox"/>	com.googlecode.libphonenumber.libphonenumber	8.11.1	8.12.35	Low		Apache-2.0	com.googlecode.libphonenumber	actions
<input type="checkbox"/>	net.java.dev.jna.platform	3.4.0	3.5.2	Outdated		LGPL-2.1	net.java.dev.jna.platform	actions
<input type="checkbox"/>	com.sun.mail.pop3	1.6.7	1.6.7	Up to date		GPL-2.0 with-classpath-exception	com.sun.mail.pop3	actions
<input type="checkbox"/>	neocities/neocities	2.0.0	3.0.0	Medium		BSD-2-Clause	neocities/neocities	actions
<input type="checkbox"/>	gridstack	0.2.6	4.2.6	High		MIT	gridstack.npm	actions

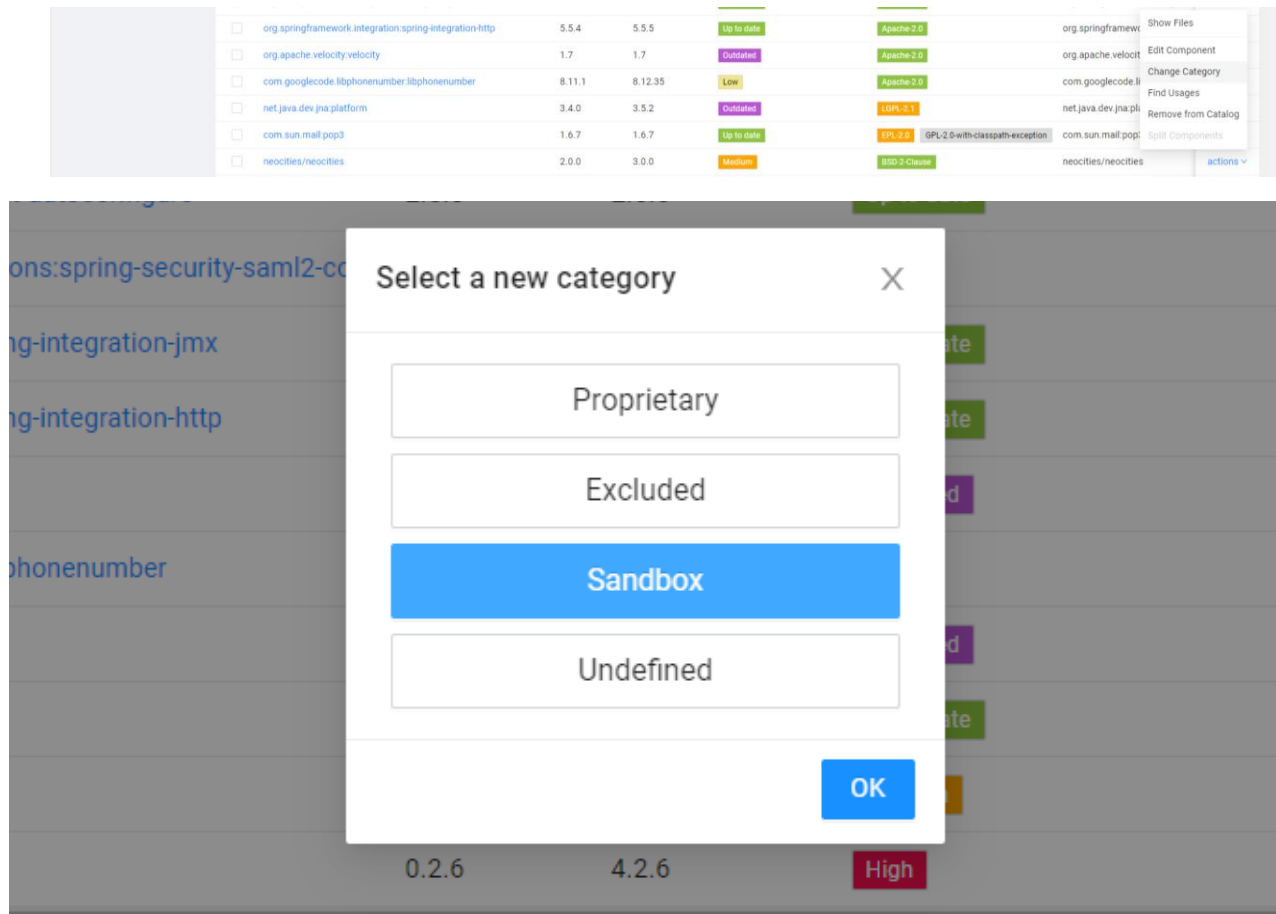
## Component Categories

This view lists all components which have been previously added to the catalog, distributed by component category:

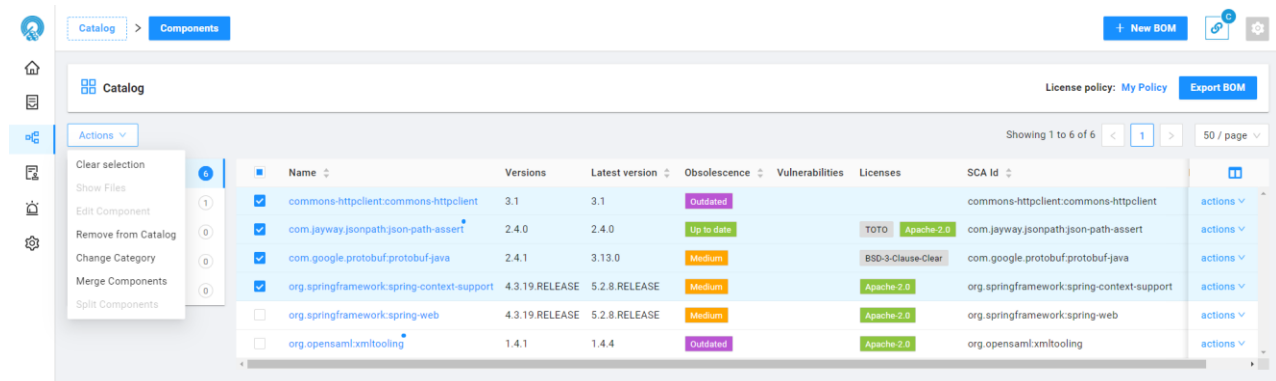
- Open Source
- Proprietary
- Excluded
- Sandbox
- Undefined

Clicking on a category on the top left list will refresh the component table.

To change the category of a component, click on “actions” for a given component, then click on “Change Category”. A modal will open to select the new category to be applied to the component.



You can also change the category of multiple components by checking component boxes and clicking on “Actions” on top left of the screen.



All component changes (description, licenses, vulnerabilities, versions, etc.) made from the Catalog view will be saved for further SBOMs when the catalog option is active during the SBOM creation.



## Propagate component changes in SBOMs

You also have the option to propagate component information changes to existing SBOMs that have this component. To do so, check the box below prior to clicking to the “Save” button.

Vulnerabilities: CVE-2019-8331 x CVE-2020-17480 x CVE-2021-23926 x CVE-2021-28168 x

Licenses: Apache-2.0 x

☐ Apply license changes to files

Comments:

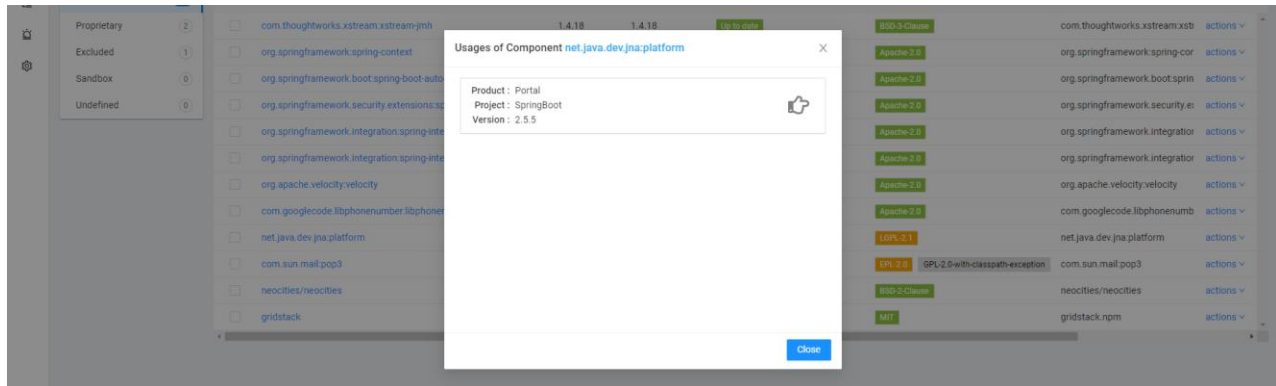
☒ Apply changes to linked bom component

## Find component usage across SBOMs

You can easily retrieve all SBOMs that have a specific component from the catalog. To do so, click on actions > Find Usages for a given component.

<input type="checkbox"/>	com.thoughtworks.xstream:xstream-jmh	1.4.18	1.4.18	Up to date	BSD-3-Clause	com.thoughtworks.xstream:xstr	actions v
<input type="checkbox"/>	org.springframework:spring-context	5.3.10	5.3.12	Low	Apache-2.0	org.springframework	Show Files
<input type="checkbox"/>	org.springframework.boot:spring-boot-autoconfigure	2.5.5	2.5.6	Up to date	Apache-2.0	org.springframework	Edit Component
<input type="checkbox"/>	org.springframework.security.extensions:spring-security-saml2-core	1.0.3.RELEASE	1.0.10.RELEASE	Low	Apache-2.0	org.springframework	Change Category
<input type="checkbox"/>	org.springframework.integration:spring-integration-jmx	5.5.4	5.5.5	Up to date	Apache-2.0	org.springframework	Find Usages
<input type="checkbox"/>	org.springframework.integration:spring-integration-http	5.5.4	5.5.5	Up to date	Apache-2.0	org.springframework	Remove from Catalog
							Split Components

Click on a SBOM to directly access it.



## Licenses

The SBOM Manager enables users to visualize, edit and even create licenses along with updating/defining license properties for each.

The main view lists supported licenses that are automatically detected in SBOMs:

- SPDX identifier
- License title
- Properties: permissions, constraints, and other properties of a license, summarized with icons
- Actions: edit or delete the license

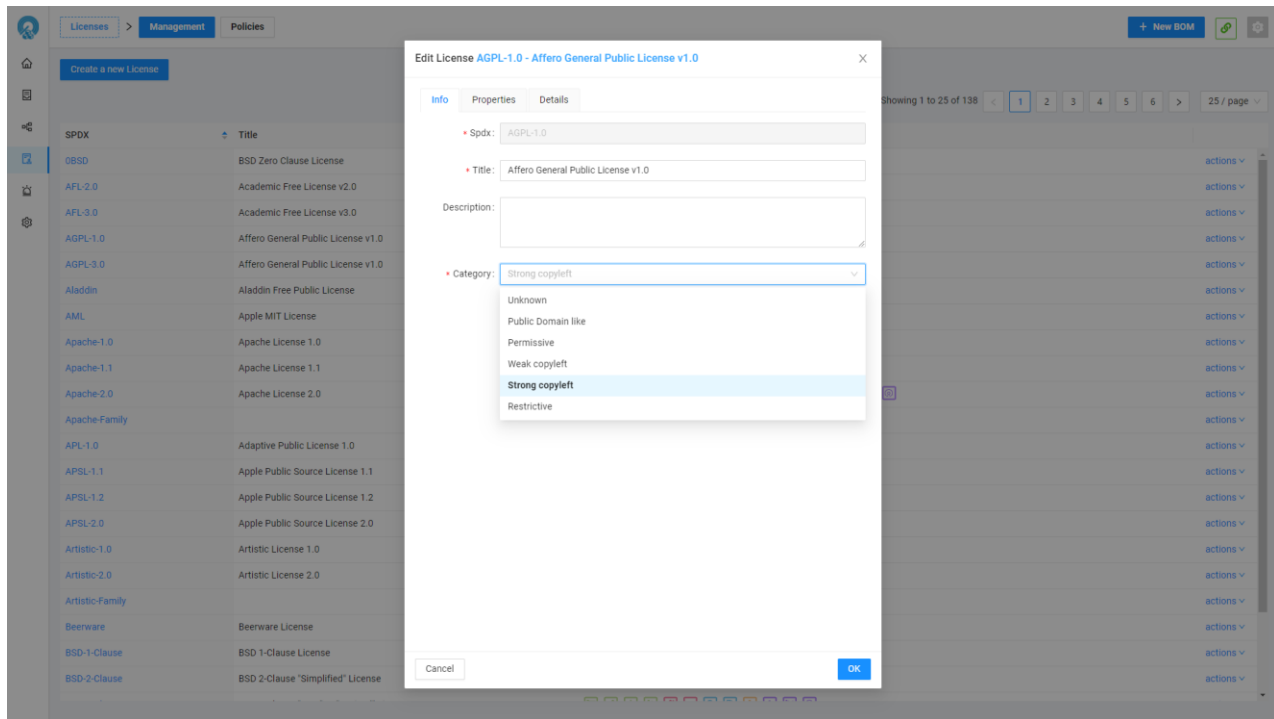
The screenshot displays the 'Licenses' management interface in CAST SBOM Manager. The top navigation bar includes 'Licenses', 'Management', and 'Policies' tabs. A 'Create a new License' button is located in the top left. The main content area shows a table of licenses with columns for 'SPDX', 'Title', and 'Properties'. A pagination bar at the top right indicates 'Showing 1 to 25 of 129' items, with page numbers 1 through 6. The table lists various licenses, including BSD Zero Clause License, Academic Free License v2.0, Academic Free License v3.0, Affero General Public License v1.0, Aladdin Free Public License, Apple MIT License, Apache License 1.0, Apache License 1.1, Apache License 2.0, Adaptive Public License 1.0, Apple Public Source License 1.1, Apple Public Source License 1.2, Apple Public Source License 2.0, Artistic License 1.0, Artistic License 2.0, Beerware License, and BSD 1-Clause License. Each row has an 'actions' dropdown menu, which is currently open for the 'AGPL-1.0' row, showing 'Edit License' and 'Delete License' options.

To edit a license, click on its SPDX identifier or select “Edit License” from the “actions” menu.

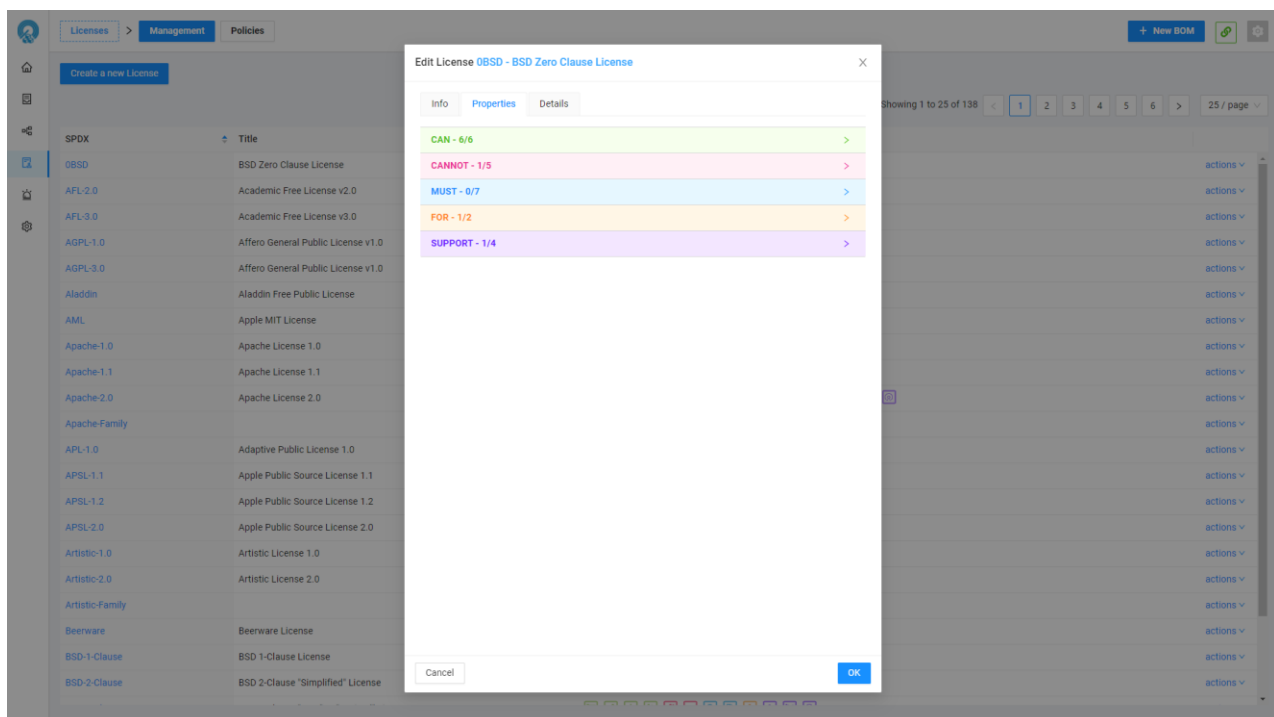
## Create/Edit a license

From the “Info” tab, you can modify:

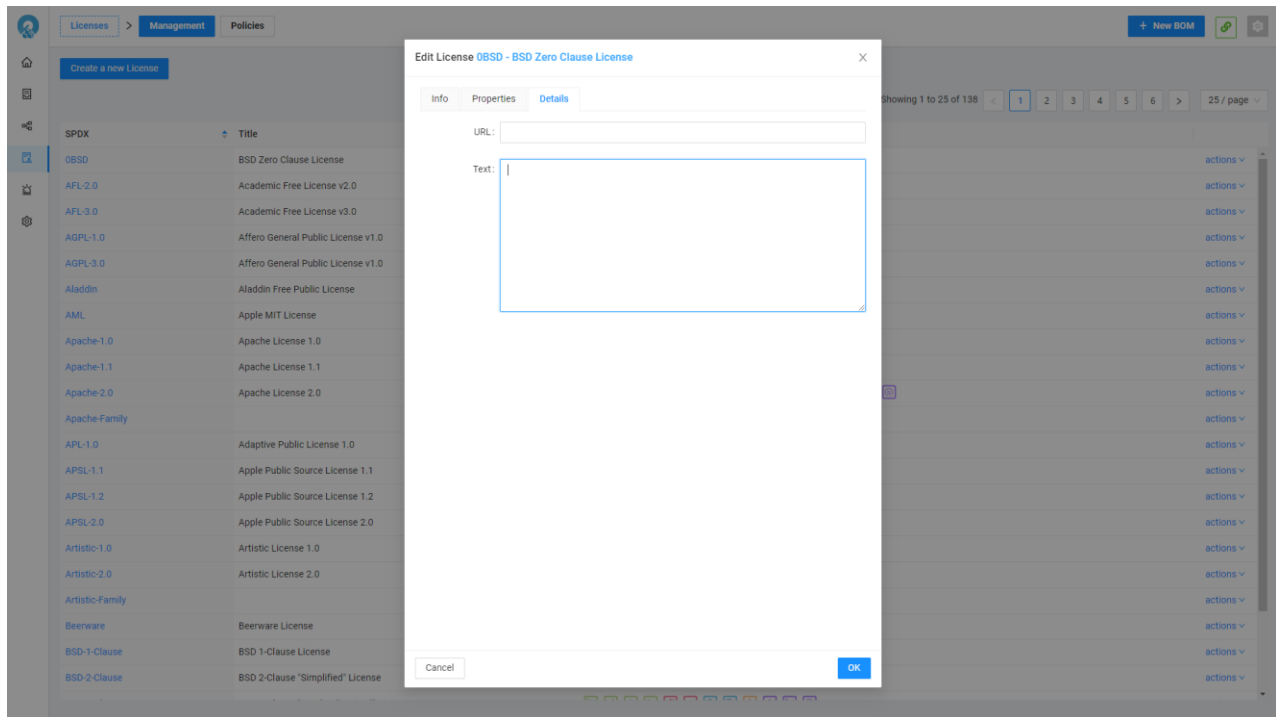
- The license title
- The license description
- The license category (Public Domain Like, Permissive, Weak Copyleft, Strong Copyleft, Restrictive...)



From the “Properties” tab, you can modify the different properties of the edited license. Check or uncheck the properties you want to edit and click “OK” to save your changes for a given license.



From the “Details” tab, you can modify the license url and eventually paste the original license text.



## Create/Edit a license policy

You can define or edit license policies. A license policy is a set of rules that will define the level of risk or acceptance for a given list of licenses.

From the main License view, click on “Policies” in the breadcrumb on top of the page.

You can switch between policies and view the corresponding compliance table for each license.

License compliance has four different properties:

- Compliant
- Partially compliant, under conditions
- Not compliant
- Undefined compliance

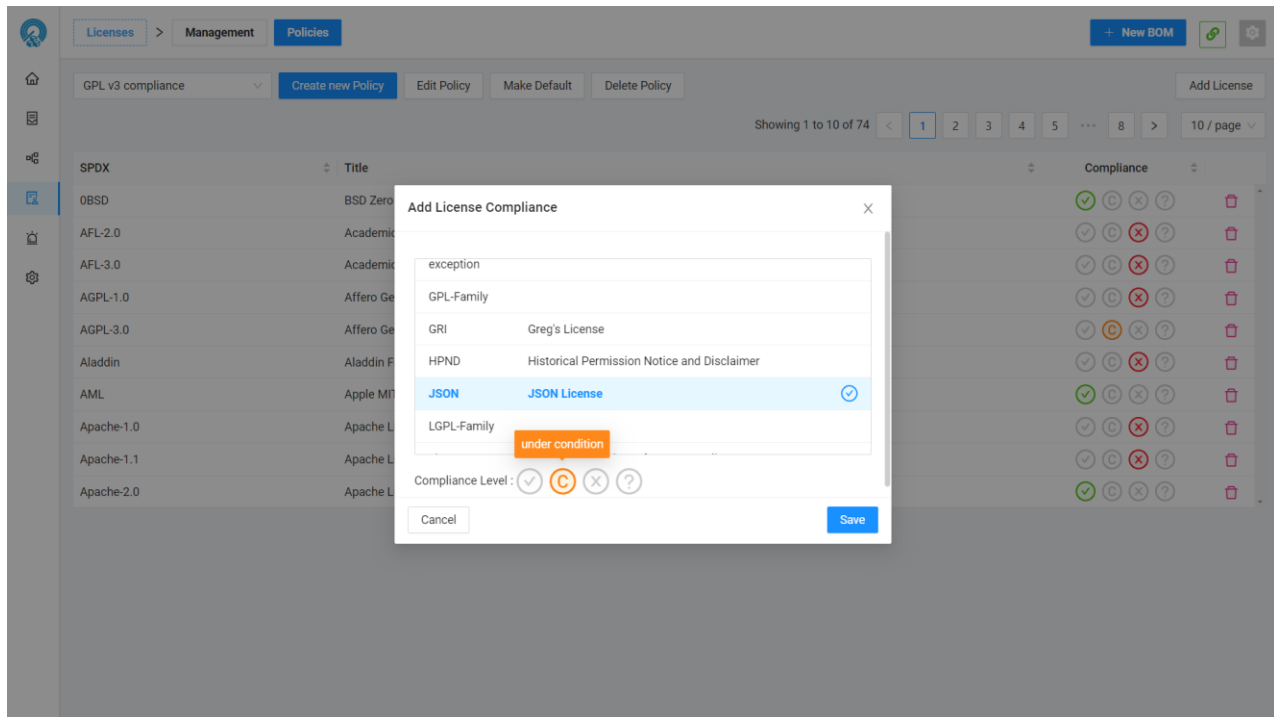
The screenshot shows the 'Policies' management interface in CAST SBOM Manager. A dropdown menu is open for 'GPL v3 compliance', showing options like 'ASL 3rd party compliance', 'GPL v3 compliance', 'My App License Policy', and 'My Policy'. Below the dropdown, there are buttons for 'Create new Policy', 'Edit Policy', 'Make Default', and 'Delete Policy'. A table lists various licenses with their titles and compliance status. The table has columns for 'Title' and 'Compliance'. The 'Compliance' column contains icons for checkmark, copy, delete, and question mark. The table lists licenses such as BSD Zero Clause License, Academic Free License v2.0, Academic Free License v3.0, Affero General Public License v1.0, Aladdin Free Public License, Apple MIT License, Apache License 1.0, Apache License 1.1, and Apache License 2.0. The 'Add License' button is located in the top right corner.

Title	Compliance
BSD Zero Clause License	✓
Academic Free License v2.0	✓
Academic Free License v3.0	✓
Affero General Public License v1.0	✓
Affero General Public License v1.0	✓
Aladdin Free Public License	✓
Apple MIT License	✓
Apache License 1.0	✓
Apache License 1.1	✓
Apache License 2.0	✓

To edit compliance properties of a license in a license policy, simply check/uncheck compliance property icons.

To remove a license from a license policy, click on the trash icon.

To add a license to a license policy, click on the “Add License” button on top right of the screen.



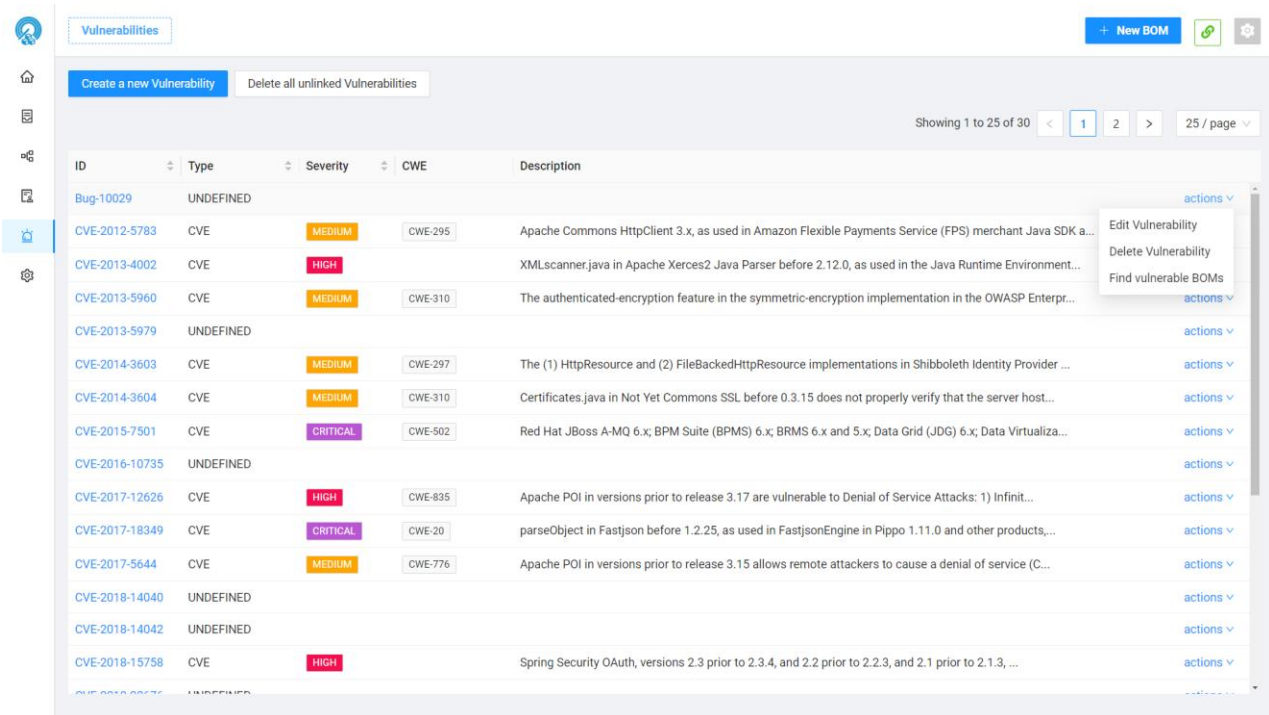
Select the license you want to add and define the compliance level, then click on the “Save” button.

You can make a specific license policy the default policy to apply when creating a SBOM, by clicking on the “Make Default” button on top of the screen.

Finally, you can completely remove a license policy by clicking on the “Delete Policy” button on top of the screen.

## Vulnerabilities

CAST Highlight’s SBOM Manager automatically detects CVEs (Common Vulnerabilities & Exposures) in scanned third-party components, but also allows users to manually log vulnerabilities in Open Source or even internal components, such as security issues or bugs. This section details how it works.

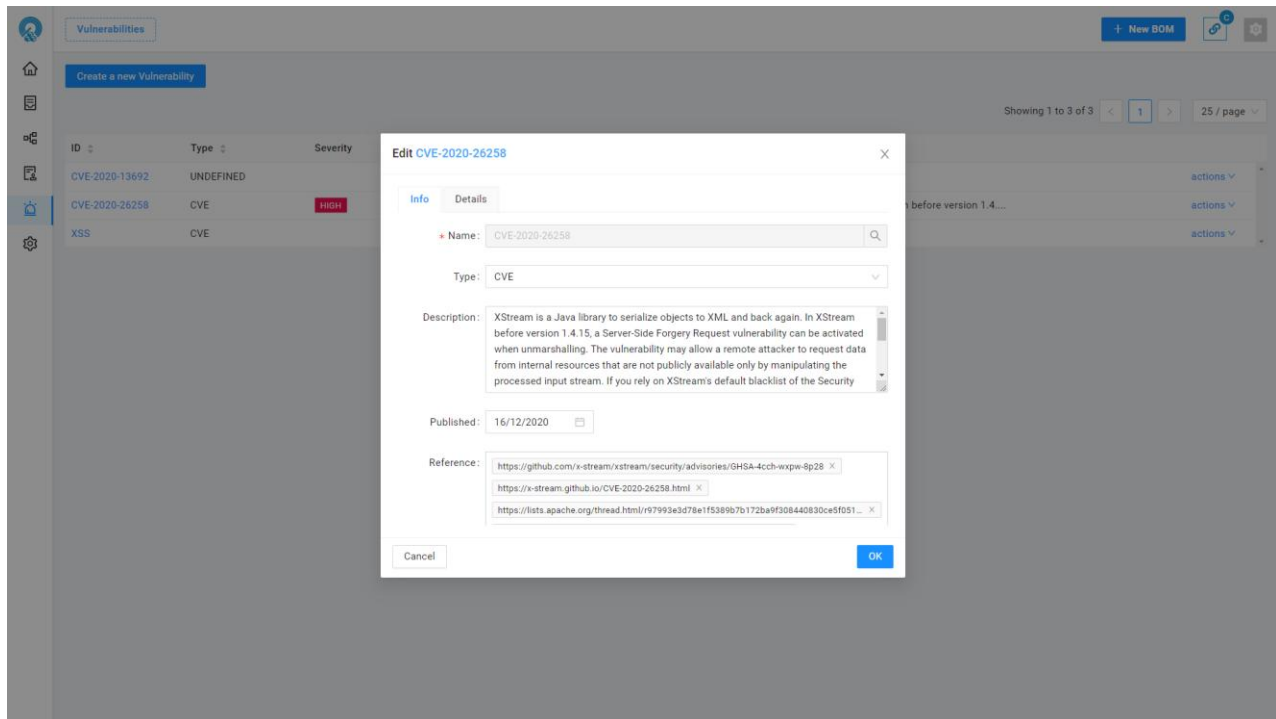


ID	Type	Severity	CWE	Description	actions
Bug-10029	UNDEFINED				actions
CVE-2012-5783	CVE	MEDIUM	CWE-295	Apache Commons HttpClient 3.x, as used in Amazon Flexible Payments Service (FPS) merchant Java SDK a...	actions
CVE-2013-4002	CVE	HIGH		XMLScanner.java in Apache Xerces2 Java Parser before 2.12.0, as used in the Java Runtime Environment...	actions
CVE-2013-5960	CVE	MEDIUM	CWE-310	The authenticated-encryption feature in the symmetric-encryption implementation in the OWASP Enterpr...	actions
CVE-2013-5979	UNDEFINED				actions
CVE-2014-3603	CVE	MEDIUM	CWE-297	The (1) HttpResource and (2) FileBackedHttpResource implementations in Shibboleth Identity Provider ...	actions
CVE-2014-3604	CVE	MEDIUM	CWE-310	Certificates.java in Not Yet Commons SSL before 0.3.15 does not properly verify that the server host...	actions
CVE-2015-7501	CVE	CRITICAL	CWE-502	Red Hat JBoss A-MQ 6.x; BPM Suite (BPMS) 6.x; BRMS 6.x and 5.x; Data Grid (JDG) 6.x; Data Virtualiza...	actions
CVE-2016-10735	UNDEFINED				actions
CVE-2017-12626	CVE	HIGH	CWE-835	Apache POI in versions prior to release 3.17 are vulnerable to Denial of Service Attacks: 1) Infini...	actions
CVE-2017-18349	CVE	CRITICAL	CWE-20	parseObject in Fastjson before 1.2.25, as used in FastjsonEngine in Pippo 1.11.0 and other products,...	actions
CVE-2017-5644	CVE	MEDIUM	CWE-776	Apache POI in versions prior to release 3.15 allows remote attackers to cause a denial of service (C...	actions
CVE-2018-14040	UNDEFINED				actions
CVE-2018-14042	UNDEFINED				actions
CVE-2018-15758	CVE	HIGH		Spring Security OAuth, versions 2.3 prior to 2.3.4, and 2.2 prior to 2.2.3, and 2.1 prior to 2.1.3, ...	actions
CVE-2018-15758	CVE	HIGH		Spring Security OAuth, versions 2.3 prior to 2.3.4, and 2.2 prior to 2.2.3, and 2.1 prior to 2.1.3, ...	actions

The main Vulnerabilities view lists all vulnerabilities detected across the different SBOMs. This data table contains:

- The identifier of the vulnerability
- The type (CVE, undefined, etc.)
- The severity (based on CVSS scores): low, medium, high, critical
- The nature (CWE) of the vulnerability
- A short description of the vulnerability
- Possible actions associated with a vulnerability (edit, delete, find vulnerable SBOMs)





## Edit a vulnerability

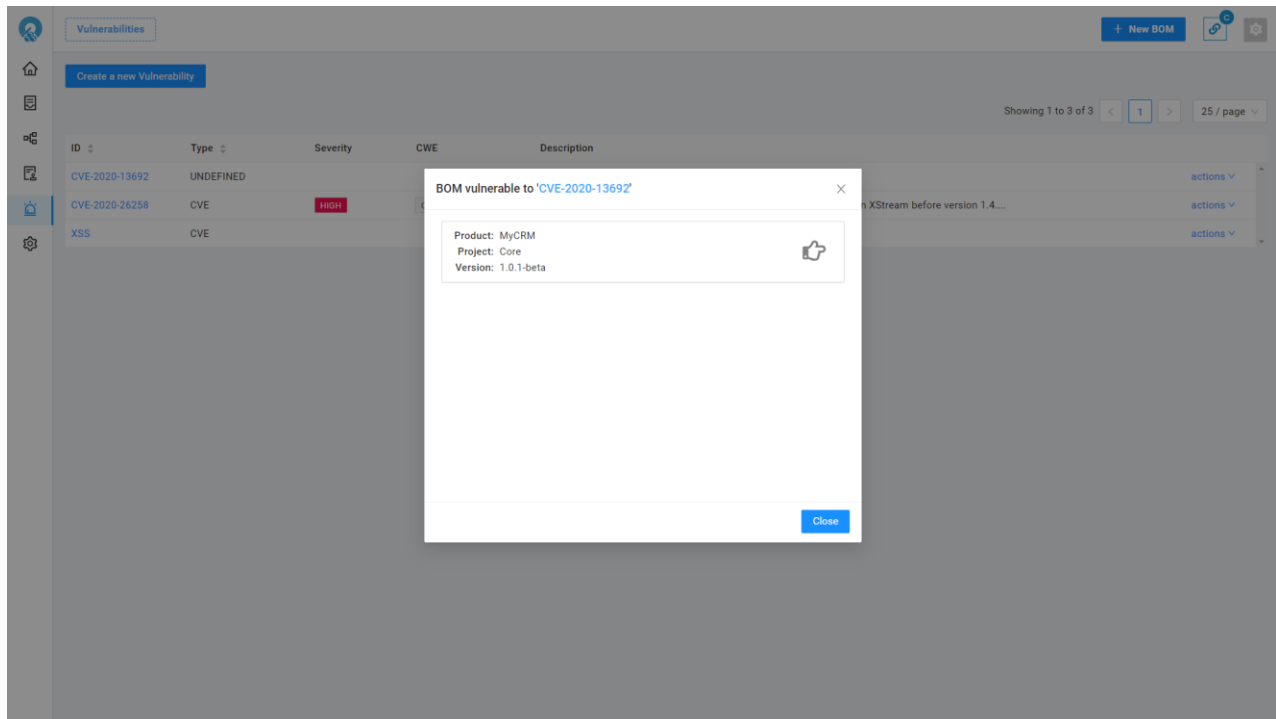
To edit a vulnerability, click on its identifier, severity level or “Edit Vulnerability” from the Actions menu. A modal opens where you can edit the different properties of a vulnerability:

- The type (CVE, Undefined)
- The description
- The publish date
- The references (url to external websites)

From the “Details” tab, you can also edit:

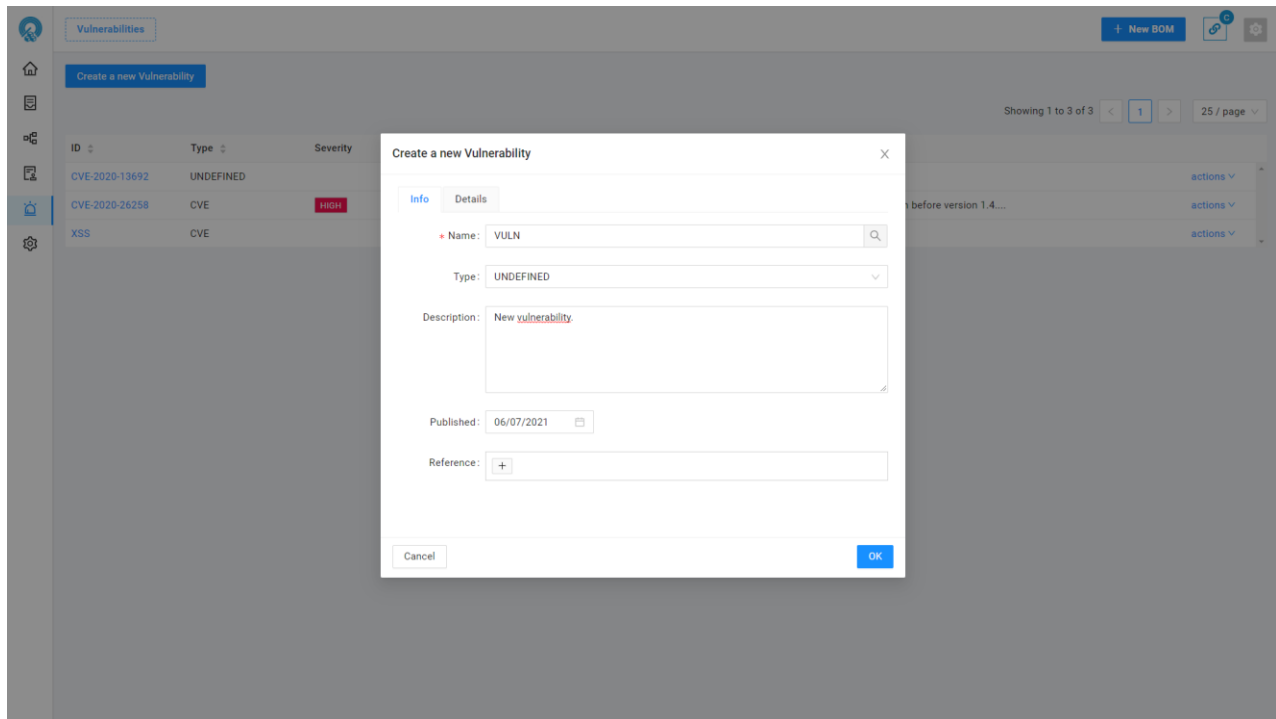
- The different CWEs associated with a vulnerability
- Indicate parameters of CVSS scores (versions 2 and 3) that are used to calculate the severity level of the vulnerability

Click on the “OK” button to save the modified vulnerability information.



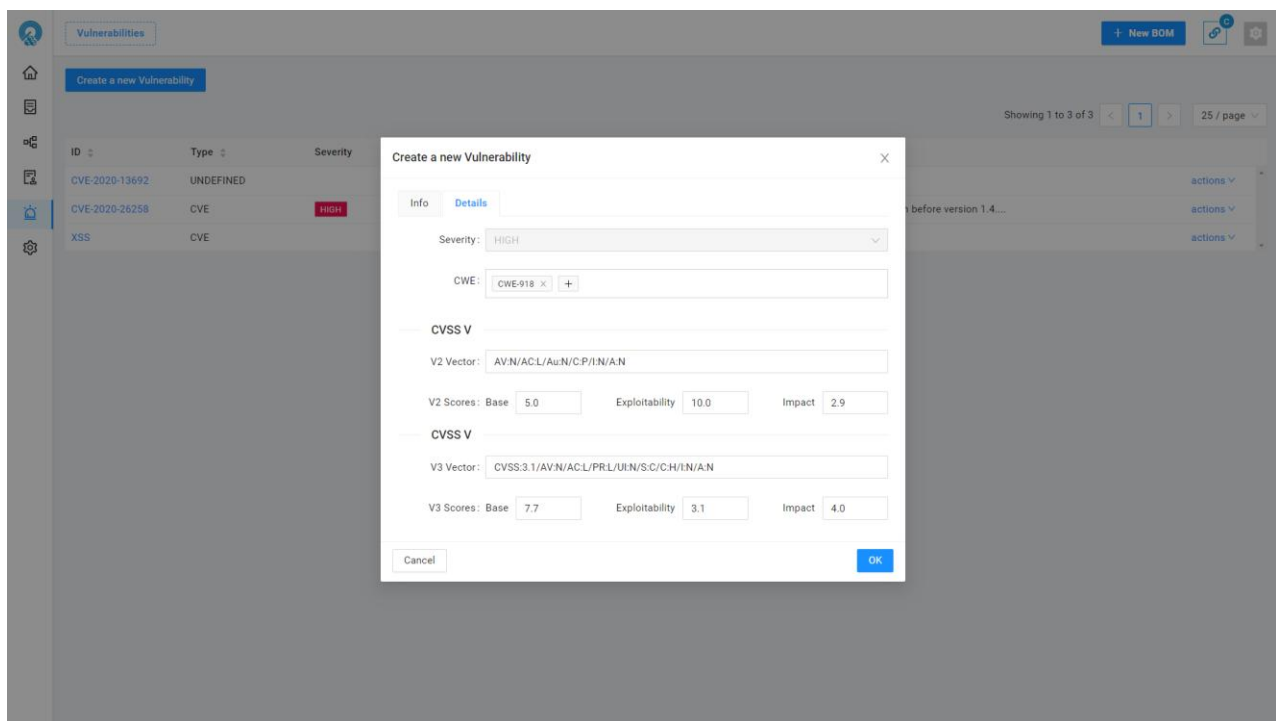
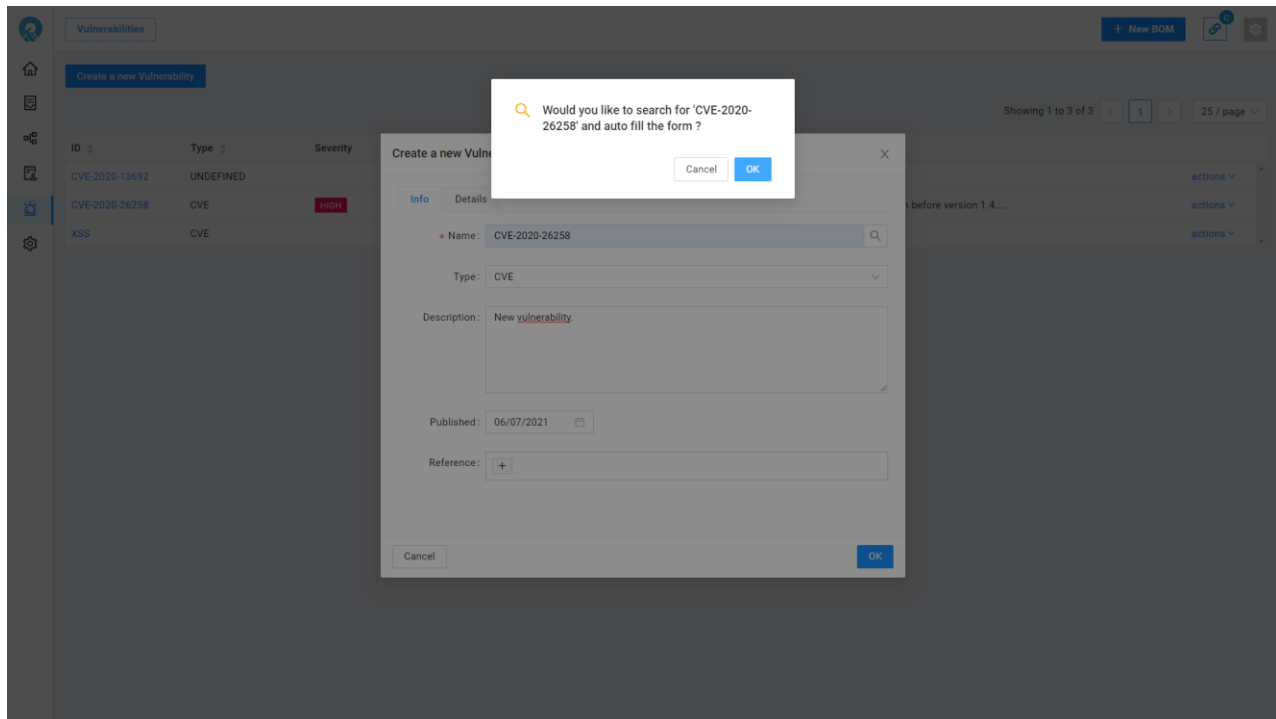
## Find vulnerable SBOMs

You can easily find vulnerable SBOMs which have a given vulnerability by clicking “Find Vulnerable SBOMs” from the Actions menu. The list of SBOMs will be displayed with a direct link to each.



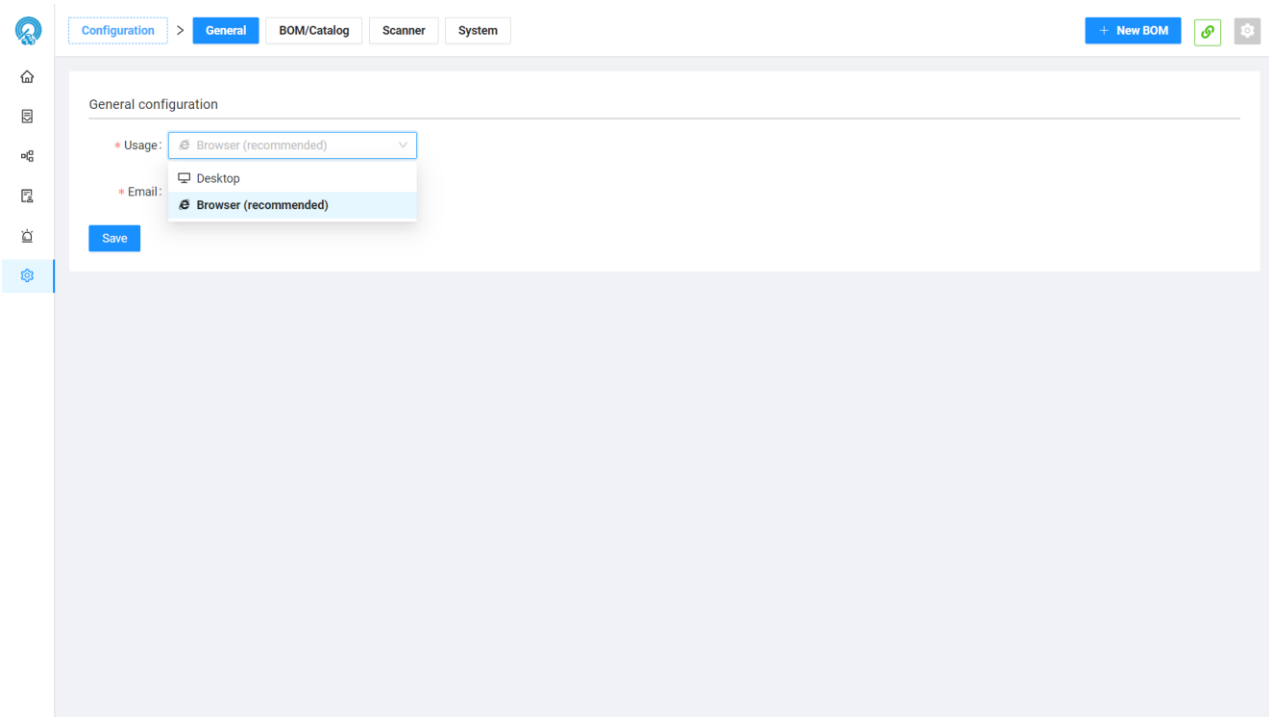
## Log a vulnerability

You can manually log a vulnerability that will be associated with a component in SBOMs. It is also possible to automatically populate vulnerability information by entering the vulnerability identifier, in case this is a CVE from the National Vulnerability Database (NVD). Type the identifier (e.g., CVE-2020-26258) and click on the magnifying glass icon. Click on the “OK” button to save this vulnerability.



## Preferences & Configuration

You can set a series of options and preferences for the SBOM Manager.



From the General element of the breadcrumb, you can retrieve the e-mail address associated with your user account. You can also switch from the Desktop mode to the Browser mode (recommended for better experience).

The screenshot shows the 'Configuration' page with the 'BOM/Catalog' tab selected. The page is divided into two main sections: 'Bill of Materials preferences' and 'Catalog preferences'. Each section contains four toggle switches for different settings. In the 'Bill of Materials preferences' section, all four toggles are turned off. In the 'Catalog preferences' section, the 'Apply changes to linked BOM components by default' toggle is turned on, while the others are off. Below each section is a blue informational box explaining that the settings apply to editing components and files, and that the last two settings specify whether to propagate license changes by default. A 'Save' button is located at the bottom left of the configuration area.

Configuration > General BOM/Catalog Scanner System + New BOM

### Bill of Materials preferences

Move split components to the 'Sandbox' category: ☐

Move merged component to the 'Sandbox' category: ☐

Apply component license changes to associated files by default: ☐

Apply added file licenses to parent component by default: ☐

These settings apply to editing components and files associated with a particular BOM. The last two settings respectively specify whether to enable by default the option to propagate license changes made to BOM components and files.

### Catalog preferences

Move split components to the 'Sandbox' category: ☐

Move merged component to the 'Sandbox' category: ☐

Apply changes to linked BOM components by default: ☒

Apply component license changes to associated files by default: ☐

Apply added file licenses to parent component by default: ☐

These settings apply to editing components and files associated with the Catalog. The last two settings respectively specify whether to enable by default the option to propagate license changes made to Catalog components and files.

Save

From the SBOM/Catalog breadcrumb element, you can define some options when creating a SBOM (e.g., moving a split component to the 'Sandbox' category by default) or using the Component Catalog (e.g., applying changes to a component or license ... in SBOMs when this object is updated in the catalog).

The screenshot shows the 'Configuration' page with the 'Scanner' tab selected. The page displays 'Scanner preferences' with two folder selection fields: 'Root folder' and 'Maven repository folder'. Each field has a 'Select a Folder' button and a 'Select' button. Below each field is a blue informational box explaining the purpose of the folder. The 'Root folder' box states it is used as the root of the file-system when selecting the codebase to scan. The 'Maven repository folder' box states it is used to facilitate the retrieval of component information when using the Maven scanner and must point to the folder where all the project artifacts are stored (e.g., C:/Users/John/m2/repository). A 'Save' button is located at the bottom left of the configuration area.

Configuration > General BOM/Catalog Scanner System + New BOM

### Scanner preferences

Root folder:  Select

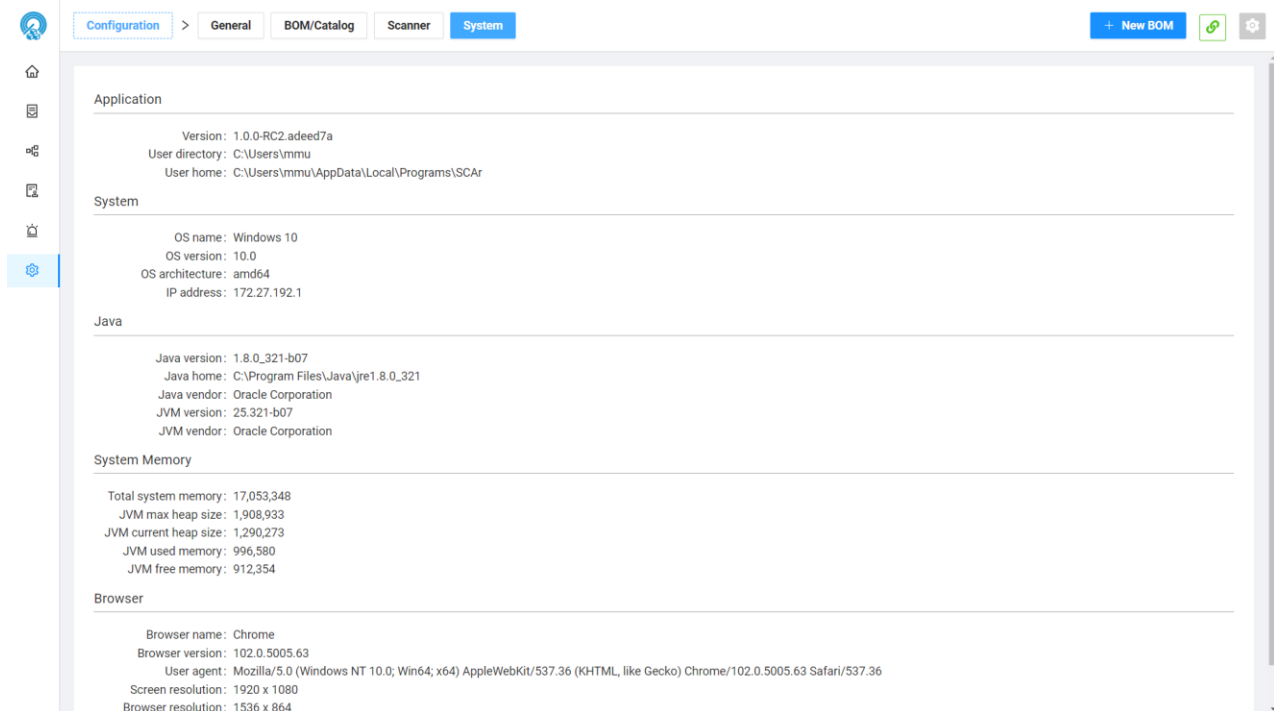
This path is used as the root of the file-system when selecting the codebase to scan.

Maven repository folder:  Select

This path is used to facilitate the retrieval of component information when using the Maven scanner and must point to the folder where all the project artifacts are stored (eg. C:/Users/John/m2/repository).

Save

From the Scanner element of the breadcrumb, you can define preferences related to scans (e.g., define the default root folder for scans, Maven repository folder).



Finally, the last element of the breadcrumb recaps the different details of your SBOM Manager installation and system supporting it (Operating System, Java version installed, system memory, browser, etc.).