revenera™

# Code Insight 2021 R4

## User Guide

# Legal Information

| | |
|---|---|
| **Book Name:** | Code Insight 2021 R4 User Guide |
| **Part Number:** | RCI-2021R4-UG00 |
| **Product Release Date:** | November 2021 |

## Copyright Notice

## Intellectual Property

## Restricted Rights Legend

# Contents

# 1

# Code Insight 2021 R4 User Guide

Code Insight empowers organizations to take control of and manage their use of open source software (OSS) and third-party components. It helps development, legal, and security teams use automation to create a formal OSS strategy that balances business benefits and risk management.

The *Code Insight User Guide* describes how to use Code Insight to realize these benefits. The guide includes the following sections.

**Table 1-1 ▪** Code Insight User Guide Navigation Table

| Topic | Content |
|---|---|
| **Using Code Insight** | “How to” information for using Code Insight functionality. |
| **Performing Advanced Searches** | Overview and procedures for using advanced searches to find specific inventory. |
| **Exporting and Importing Project Data** | Explanation and procedures for exporting and importing project data. |
| **Automated Analysis** | Information about Code Insight automated analysis tools and features. |
| **Performing a Remote Scan** | Information about the Code Insight agent scan, performed remotely. |
| **Configuring Source Code Management** | Procedures for using Source Code Management (SCM) systems with Code Insight. |
| **Pages and Panels** | Reference to field descriptions on the pages, panes, tabs, and dialogs used in the Code Insight user interface. |
| **Code Insight User Roles and Permissions** | A reference to the various user roles and permissions available in Code Insight to control access to Code Insight functionality. |

# Intended Audience

The *Code Insight User Guide* is intended for anyone who uses Code Insight for scanning, analyzing, and reviewing project codebases.

# Product Support Resources

The following resources are available to assist you with using this product:

- Revenera Product Documentation
- Revenera Community
- Revenera Learning Center
- Revenera Support

## Revenera Product Documentation

You can find documentation for all Revenera products on the Revenera Product Documentation site:

https://docs.revenera.com

## Revenera Community

On the Revenera Community site, you can quickly find answers to your questions by searching content from other customers, product experts, and thought leaders. You can also post questions on discussion forums for experts to answer. For each of Revenera's product solutions, you can access forums, blog posts, and knowledge base articles.

https://community.revenera.com

## Revenera Learning Center

The Revenera Learning Center offers free, self-guided, online videos to help you quickly get the most out of your Revenera products. You can find a complete list of these training videos in the Learning Center.

https://learning.revenera.com

## Revenera Support

Customers who have purchased a maintenance contract for their product(s) can submit a support case or check the status of an existing case by making selections from the **Get Support** menu in the Revenera Community (https://community.revenera.com).

# Contact Us

Revenera is headquartered in Itasca, Illinois, and has offices worldwide. To contact us or to learn more about our products, visit our website at:

http://www.revenera.com

You can also follow us on social media:

- Twitter
- Facebook
- LinkedIn
- YouTube
- Instagram

# 2

# Using Code Insight

This chapter provides basic information about Code Insight that will enable you to start using the product effectively. The following topics are covered in this section:

- Opening Code Insight
- About Roles and Permissions in Code Insight
- About Code Insight Projects
- About Code Insight Scans
- Creating a Project
- Applying a Scan Profile to the Project
- Uploading a Project Codebase (for Server Scans)
- Scanning the Codebase (Server Scans)
- Overview of Scan Results
- Auditing Scan Results in the Analysis Workbench
- Reviewing Published Inventory for a Project
- Working with Security Vulnerabilities
- Managing Security Vulnerability Alerts
- Creating and Editing Custom Components
- Creating and Editing Custom Licenses
- Managing Custom Detection Rules
- Finalizing the Notices Text for the Notices Report
- Generating Reports for a Project
- Managing Scan Queues Across All Scan Servers
- Viewing Inventory Across All Projects

- Accessing Projects in Code Insight
- Managing Projects
- Managing Policy Profiles
- Managing Authorization Tokens
- Downloading Code Insight Log Files

# Opening Code Insight

Code Insight runs in your web browser. This section explains how to start up Code Insight, opening to the Code Insight dashboard.

***Note* ▪ *If this is the first time you have opened Code Insight or if you have recently upgraded Code Insight or shut down your Tomcat server, you must start up the Tomcat server with the startup command before opening Code Insight. For more information, see "Starting and Stopping Tomcat" in the "Installing Code Insight" chapter in the "Code Insight Installation and Configuration Guide".***

***Task*** ***To open Code Insight, do the following:***

1. Launch a web browser and navigate to: `http://<your_server_host_name>:8888/codeinsight`.

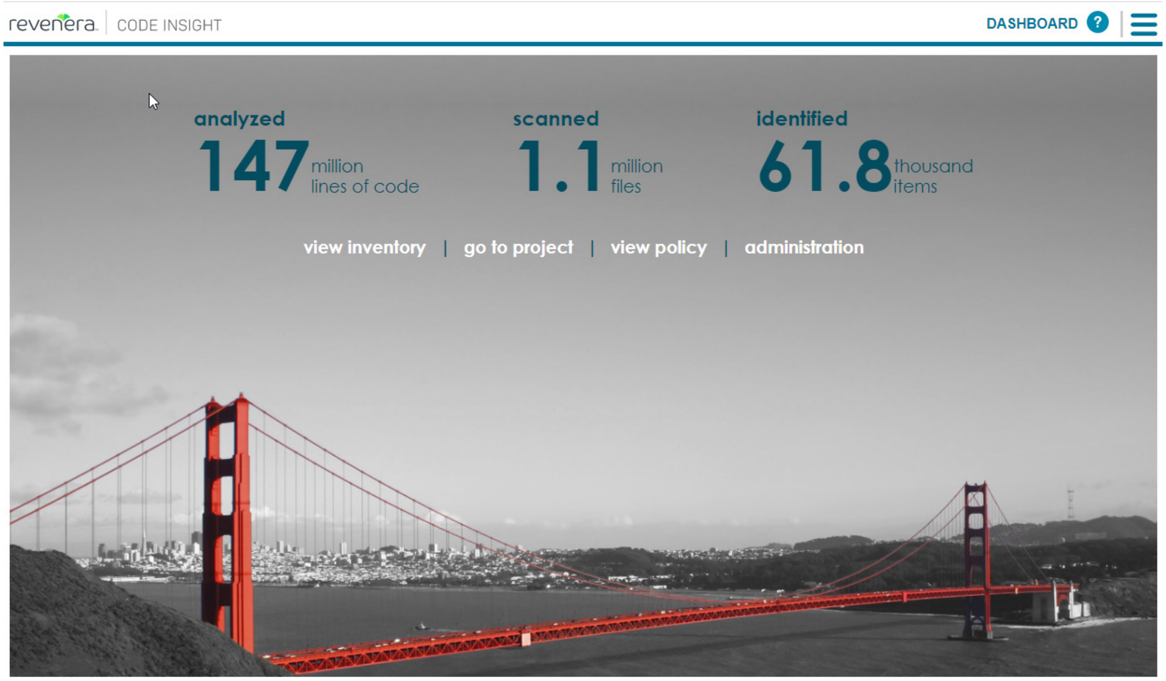
   If you are unsure about your server host name, contact your site's system administrator or the Code Insight System Administrator for guidance.

2. Enter your Code Insight credentials in the **Username** and **Password** fields.

   ***Note* ▪ *The default login name is* `admin`*; the default password is* `Password123`*. Your installation might require a different login name and password. If you are unsure about what credentials to enter, contact the Code Insight System Administrator for guidance.***

3. Click **Login**.

   The the Code Insight dashboard is displayed. This dashboard shows statistics from the most recent codebase scans performed across all Code Insight projects and provides entry points to other parts of the Code Insight Web UI



4. From the Code Insight dashboard, navigate anywhere in Code Insight that you have permission to access, as described in the following table. (Options are not displayed for those areas to which you do not have access permission.) For more information about the dashboard, see Import Project Data Dialog.

| Click this option... | ...to go here |
|---|---|
| **view inventory** | The **Inventory** view, which provides a compilation of inventory across all current Code Insight projects. The list can be filtered at a basic level to show inventory for all projects, only your projects, or for a specific project. Filtering can be further refined by vulnerability severity, review status, and many other criteria. For more information, see Viewing Inventory Across All Projects. |
| **go to project** | The **Projects** view, which provides access to all current projects in Code Insight. For more information, see Accessing Projects in Code Insight. |
| **view policy** | The **Policy** page, where you have access to all policies that automate the review process of project inventory when it is published. For more information, see Managing Policy Profiles. Access to this page requires Manage Policy permissions. |
| **administration** | The **Administration** page, where you have access to Code Insight administrative functionality. See the *Code Insight Installation and Configuration Guide* for a description of administrative tasks. Access to this page requires Code Insight System Administrator permissions. |

## Viewing Online Help and Online Guides

Code Insight provides online help topics and online versions of its guides so you can find answers to your questions about the product while you are using it.

***Task*** ***To access online help and guides, do the following:***

- To access the online help for the current Code Insight page, including the Code Insight dashboard, click the Help icon (?) in the upper right corner of the Code Insight Web UI.
- To access the Code Insight online user guides, click the icon ≡ in the upper right corner of the Code Insight Web UI, including the Code Insight dashboard. From the Code Insight main menu that opens, select **HELP**. The **Help** page is displayed, providing a list of links to the available online documentation.

## Changing Your Password

If you want to change your current Code Insight password, use this procedure.

***Task*** ***To access:***

1. Click the icon ≡ in the upper right corner of the Code Insight Web UI, including the Code Insight dashboard. The Code Insight main menu is displayed.
2. Select **Preferences** from the menu to open the **Preferences** page.
3. Enter your new password in **New Password**, and then reenter it in **New Password Confirm**.
4. Click **Update Password**.

# About Roles and Permissions in Code Insight

Code Insight offers a set of user roles and permissions that enables your site to control access to Code Insight features and functionality.

The initial Code Insight System Administrator, identified during Code Insight installation, can assign users to system-level roles for managing Code Insight policies and creating Code Insight projects. The System Administrator can also create other System Administrators and define default Project Administrators, Analysts, and Reviewers that are automatically assigned to projects when they are created.

At the project level, a project creator automatically becomes the Project Contact as well as a Project Administrator (among other roles) for the project. A Project Administrator can assign users to project roles that enable these users to analyze and review project scan results. The administrator can also remove a user from any project role as needed, whether the user was manually assigned the role or had inherited it.

***Note** ▪ When a project is migrated from a previous Code Insight version (2020 R3 or earlier), by default the Project Owner becomes the Project Contact and is assigned to the Project Administrator and Analyst roles.*

For more about the management of Code Insight roles and permissions, refer to the following:

- The Assigning and Removing Project Users describes the assignment of users to project roles.
- The Code Insight User Roles and Permissions appendix serves as a reference to the various Code Insight roles available and the permissions granted to each role. As you prepare use the Code Insight, refer to this appendix to determine the roles required to perform certain Code Insight functionality and the permissions the roles enable.

# About Code Insight Projects

A project in Code Insight represents a version or release of an application or application module on whose codebase you run a Code Insight scan and analysis. The scan discovers of evidence of open-source software (OSS) and third-party code in the application files and generates an inventory of this software that you can then review and remediate within the project. The project also provides facilities that enable you to perform your own thorough audit of the scanned files to validate the generated inventory and create or modify inventory items if needed.

Typically, you would create a project for each one of your products or services, but you might also create projects to review vendor code for security or licensing issues, to screen an open-source component you are considering using, or to prepare for an open-source contribution.

For added flexibility, you can group projects in project folders that represent business units, teams, product lines, tools, or any other groupings that help you locate projects more easily. The folders can be nested to the desired level.

All projects have the same basic key elements but use various configurations, as described in these next sections:

- Key Project Elements
- Common Project Configurations
- Legacy Projects

***Important** ▪ You can create projects only if the Code Insight System Administrator has granted you permission to do so, as described in the “Code Insight Installation and Configuration Guide”.*

## Key Project Elements

The following key elements of a project are important to keep in mind when creating, configuring, and organizing projects:

- **Materials to scan or analyze**—Each project has an uploaded codebase or a configured remote scan location (such as on a build server, artifact repository, or version control system).
- **Scan profile**—Each project has an associated scan profile with a set of scan settings that are applied when the project is scanned. (The profile can be one of the default scan profiles or a custom scan profile).

- **Policy profile**—Each project has an associated policy profile with a set of intellectual property or security policies that are applied when project inventory is published to the project (such as during the first scan or when manually published by a project user).
- **Project visibility**—Each project has an associated visibility configuration that specifies which logged-in users have the ability to view or change the project.

# Common Project Configurations

These are some examples of common project configurations:

- **Project that manages server scan results**—A project is configured for server scan (that is, a Scan Server scan) on one or more application source codebases that are uploaded to the Scan Server or synchronized to the server through a Source Control Management application. The scan profile for a given server scan can perform a full analysis of the source—including searches for exact matches of entire OSS or third-party files and for partial source-code matches (fingerprints)—and process files inside archives.
- **Project that manages remote-scan results**—The project manages the results of remote scans performed on one or more remote server codebases—each codebase typically consisting of built artifacts residing on a build server (for example, a Jenkins or GitLab server or another supported build server, artifact repository, or version control system). A given remote scan is performed by a Code Insight scan-agent plugin, which sends the scan results to the project. (Currently, only OSS or third-party license evidence that the scan discovers in the codebase files is viewable in the project.)
- **Project for manage results of server and remote scans**—The project is configured to perform server scans on the application's source code (either uploaded or synchronized to the Scan Server), *and* it manages the results of remote scans performed on build server codebases.
- **Security-focused project**—The policy profile selected for the project triggers an automatic review of project inventory based on the presence of security vulnerabilities, on the CVSS scores and severity of these vulnerabilities, and on other criteria.
- **Project focused on intellectual-property protection**—The policy profile selected for the project triggers an automatic review of project inventory based on the presence of allowed and not-allowed components, version ranges, and licenses.

# Legacy Projects

The following section describes the changes that were introduced for the Code Insight project in the 2020 R3 release:

- Projects Prior to Code Insight 2020 R3
- Projects in Code Insight 2020 R3 and Later
- Resource for Additional Information

## Projects Prior to Code Insight 2020 R3

Prior to Code Insight 2020 R3, you had the option of creating one of two types of projects:

- **Standard**—This project type was reserved for server scans only (that is, scans performed by the Scan Server and that require that codebase files be uploaded or synchronized to the Scan Server in order to be scanned.) The standard project enabled users to audit the OSS and third-party evidence discovered in the scanned source files and then review, remediate, and finalize an inventory of OSS and third-party components used by the application.
- **Inventory Only**—This project type was used for remote scans. Remote scans are performed by scan-agent plugins that scan remote server codebases—typically containing an application's built artifacts—and then send the scan results to a project on the Code Insight server. Previously, the plugin was designed to send the resulting inventory of OSS and third-party software to a project created in Code Insight as "Inventory Only". Users could then review, remediate, and finalize the inventory. However, because no file information was sent to the project, users could not audit the scanned codebase files.

## Projects in Code Insight 2020 R3 and Later

Starting in Code Insight 2020 R3, only one type of project is supported—a *unified* project used by both server and remote scans. The unified project can manage the scan results of one or more server scans, one or more remote scans, or a combination of both scan types for a given application. It also manages remote-file information sent by the scan agents. Thus, in a single project, you can audit an application's codebase files loaded on the Scan Server and its remote codebase files—as well as review, remediate, and finalize the complete inventory of OSS and third-party software for the application, as captured from all server and remote scans.

If you are upgrading from a pre-2020 R3 Code Insight release, existing projects are handled in the upgrade as such:

- Standard projects are automatically migrated to the new release as unified projects, enabling you to add the results of remote scans to the previous server scan results in these projects if you want.
- Inventory-only projects are not automatically migrated to the new release as unified projects and will continue to be supported in future releases for a limited time. These legacy projects support only 2020 R2 or earlier scan-agent plugins and allow imports only from other legacy projects. You might consider manually migrating legacy projects to unified projects *now* even if you intend to continue to use these projects for remote scans only (see Resource for Additional Information).

In the user documentation for Code Insight 2020 R3 and later, a unified project is simply called a *project*. Previous inventory-only projects will be referred to as *legacy projects*.

## Resource for Additional Information

For complete information about the concept of a unified project, its impact on projects existing before the 2020 R3 release, and the migration of the previous projects to the unified project type, see the following Knowledge Base article in the Revenera Customer Community:

https://community.flexera.com/t5/FlexNet-Code-Insight-Customer/Code-Insight-2020-R3-Changes-to-Projects/ta-p/160059

# Creating a Project

You must create the project before running a scan on a codebase—whether the codebase resides locally on a Scan Server or is a remote codebase to be scanned by a scan agent.

The following procedure focuses on creating a public project, which is the default project type. However, you can use this same procedure to create a private project, which has limited user access. For more information about creating a private project, see Creating a Private Project.

Any user in the system has read-only access to a public project. To what degree a user can interact with the project depends on whether the user has a project role and what the role is.

***Important*** ▪ *You can create projects only if the Code Insight System Administrator has granted you permission to do so, as described in the "Configuring Code Insight" chapter in the "Code Insight Installation and Configuration Guide". The* ***Add New*** *button and* ***Create New Project*** *right-click menu option referenced in the following procedure are available only if you have this permission.*

***Task*** ***To create a project, do the following:***

1. Ensure that you are in the **Projects** view in the Code Insight Web UI. See Opening the Projects View if you need instructions.
2. In the **Projects** pane on the left, use one of the following methods to create the project. (The display of projects in the **Projects** pane is either in a tree or plain-list format. See Select the Projects Display Format.)
    - (Project tree only) To create a project *under a specific folder*, use either method:
        - Right-click the specific folder or any project directly under that folder, and select **Create New Project | At This Level**.
        - Select the specific folder or any project directly under that folder; then click the **Add New** button (located at the top of the **Projects** pane), and select **Project** from the associated dropdown menu.

      Once the project is defined, it will be stored under the selected folder or the folder to which the selected project belongs.
    - To add a project to the plain list or to store a project *at the root level* of the project tree, use either method:
        - Click the **Add New** button, and select **Project** from the associated dropdown menu.
        - (Plain list only) Right-click anywhere in the plain list, and select **Create New Project**.
        - (Project tree only) Right-click anywhere in the project tree, and select **Create New Project | At Root Level**.

      Once the project is defined, it will be added to plain list or stored at the root level of project tree.

   The **Add Project** dialog is displayed.
3. Complete the following fields on the **Add Project** dialog:
    - **Name**—Type a name for the new project.

- **Project Visibility**—Select **Public** to allow general access to the project. All users in the system can view a public project. To what degree a user can interact with a public project depends on the project role of the user. (To create a project with the **Private** option to limit its visibility, see Creating a Private Project.)

  ***Note*** *▪ The **Project Visibility** setting can be later changed through the **Edit Project** option on the **Manage Project** menu on the **Summary** tab. For more information, see Editing the Project Definition and General Settings.*

- **Scan Server**—Select the Scan Server for this project. Even if the project will contain the results of only remote scans, you still need to specify a Scan Server. In this way, it is available should you need to perform a deep analysis evidence in the codebase files.

4. Click **Save** to save the new project.

   As project creator, you automatically become the Project Contact and are assigned to the Project Administrator, Analyst, and Reviewer roles. These roles enable you to initially manage the project and its users, analyze the project codebase, and review project inventory.

5. (Optional) Assign project roles to users who will interact with the project. You can also remove yourself and others from any roles as needed. For more information, see Assigning and Removing Project Users.

The new project appears in the list of projects under the appropriate folder or at the root level of the list. At this point, the new project's dashboard in the right pane does not contain information about the project. The project dashboard will be populated once a scan is run on the project codebase files.

# About Code Insight Scans

A Code Insight scan processes codebase files to identify evidence of OSS and third-party code. The scan results are then processed by Code Insight, which creates inventory of the detected OSS and third-party components, detects licenses and security vulnerabilities, applies policies for automated review, and creates review and remediation tasks per project configuration.

The following describes the types of Code Insight scans and the analysis techniques used by scans:

- Scan Types
- Scan Analysis Techniques
- Scan Profiles

## Scan Types

Code Insight performs two type of codebase scans—server and remote. A single Code Insight project for a given application can contain of the results of either or both types of scans.

### Server Scans

Server scans are performed by the Code Insight Scan Server. A server scan requires that the codebase files you want to scan reside on the Scan Server. These files are placed on the server either by uploading them (manually or through Code Insight) or by synchronizing one or more repositories in a Source Control Management (SCM) system, such as Git or Perforce, to the server. The complete codebase for a server scan typically represents the source code for a given application. Once the scan is complete, you can review and remediate the inventory of discovered open-source and third-party software, as well as audit the scanned files to verify the inventory findings and customize inventory as needed. For more information about configuring a project for server scans and performing these scans, see the following:

- Uploading a Project Codebase (for Server Scans)
- Configuring Source Code Management
- Scanning the Codebase (Server Scans)

### Remote Scans

Remote scans are performed by a Code Insight scan-agent plugin, which is installed and configured on a remote instance to perform a scan within the context of an Engineering build server on that instance (for example, an IDE, an artifact repository, a CI product, or an product to build, test, or install your application).

The plugin allows a scan of built artifacts and source files and then sends the results to Code Insight as inventory for review and remediation. A representation of the scanned remote files is also available in the project, enabling you associate these files with inventory, mark them as reviewed, or perform other file-related actions.

For more information about configuring Code Insight scan-agent plugins and remote scanning, see the following:

- Performing a Remote Scan
- *Code Insight Plugins Guide* (available for download in the Revenera Customer Community)

## Scan Analysis Techniques

The Code Insight scan performs a static analysis of files of any type (source or binary) to find open source and third-party components, licenses, and security vulnerabilities and, depending on the scan profile, to identify file-level and snippet-level evidence to aid users in determining the origin of every file in the codebase. The end goal of the Code Insight scan is to build an accurate Bill of Materials and to eliminate any security and intellectual property (IP) risk associated with the materials.

During a codebase scan, Code Insight processes every file in the materials, regardless of programming language or file type. It processes source materials, scripts, object code, binaries, images, icons, and documents to identify both open source and closed source components, licenses, and security vulnerabilities.

Code Insight identifies these elements using a combination of Automated Analysis and Advanced Analysis techniques:

- **Automated Analysis**—The Scan Server uses automated detection rules to identify components, versions, licenses, and security vulnerabilities. In applying these rules, the Scan Server automatically generates inventory items that make up the Bill of Materials. The rules are found in the *Code Insight data library*, which is updated on your Code Insight server through both an internal process and as part of the weekly Electronic Update. For more about Automated Analysis, see the Automated Analysis chapter.

- **Advanced Analysis**—The Scan Server uses Advanced Analysis techniques to detect copyrights, emails, URLs, search terms, and source code of actual OSS and third-party software. This level of analysis requires the Code Insight Compliance Library (CL), downloaded from the Product and License Center. The CL is a database containing the source code and other elements found in OSS and third-party software. Advanced Analysis attempts to match the source code in the CL database with entire files and source-code fingerprints (snippets) in the scanned files to generate evidence of OSS and third-party software on which you can take action.

  Currently remote scans do not support the use of the CL.

## Scan Profiles

A scan profile controls aspects of the scan behavior, such as the level of inventory dependencies to scan, whether to perform package discovery or license detection within archives, the type source-code matching to be performed against the Code Insight Compliance Library, and other behavior configuration. For more information about scan profiles, Applying a Scan Profile to the Project.

# Applying a Scan Profile to the Project

Code Insight supports scan profiles for abstracting and reusing scan settings. Often, organizations are concerned about consistent scan or audit practices across their enterprise, and scan profiles support that need. The following describes scan profiles and how to create one:

- About Scan Profiles
- Applying a Scan Profile

***Note** ▪ Currently, remote scans do not support scan profiles.*

## About Scan Profiles

Code Insight includes the following default scan profiles:

- **Basic Scan profile (without a CL)**—Used to produce automated findings along with string-based third-party indicators at a file level. This profile disables both exact-file and source-code matching, and therefore does *not* require a Compliance Library (CL).
- **Standard Scan profile**—Expands the file-level third-party indicators with exact-file matches based on the Compliance Library.
- **Comprehensive Scan profile**—Further expands the file-level third-party indicators with exact file-level and source-code matches based on the Compliance Library.

Additional scan profiles can be defined by the Code Insight System Administrator for use across projects, as described in the *Code Insight Installation & Configuration Guide*.

# Applying a Scan Profile

The scan profile is used to abstract and reuse scan settings across projects. The scan profile currently selected for a project shows in the **Scan Settings** section on the **Summary** tab. The scan settings specified in the current scan profile are applied for each project scan. However, if you want to apply a different scan profile to the project, follow these steps.

***Note*** ▪ *Currently, remote scans do not support scan profiles.*

***Task*** ***To select a new scan profile, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. From the list of projects, select the project for which you want to apply a scan profile.

   (Optional) Click the **My Projects** toggle at the top of the list to show only those projects with which you are associated as Project Contact or through a project role. (For details, see Showing Only Your Projects.) You can also search projects by name, inventory, or security vulnerability as described in Searching Across All Projects the System.
3. Do one of the following to open the project:
   - Click the project name (in the example, *New Project*) in the title bar of the right panel.
   - Click the **Open Project** icon ().
4. Click the **Summary** button at the top of the window to open the **Summary** tab for the project.
5. Click **Manage Project**, and select **Edit Project** from the dropdown menu.
6. From the **Edit Project** dialog, navigate to the **Scan Settings** tab, and select the desired scan profile for your project. (Click the information icon next to a selected scan profile to open a read-only view of its attributes.)

# Uploading a Project Codebase (for Server Scans)

For a server scan, the codebase files that you want to scan for a project must reside on the Scan Server before you can perform the scan. One way to place these files on the Scan Server is upload them through Code Insight, as describe in this section. You can upload multiple codebases for a single project scan.

Supported archive types for uploading a codebase include `.zip`, `.tar`, `.tar.gz`, and `.7z`.

The following topics describe the codebase upload process:

- Performing the Codebase Upload
- Supported Archive Types for Expansion
- More About Archive Expansion Behavior During Codebase Uploads

Refer to the Code Insight User Roles and Permissions appendix for role requirements to upload the codebase.

As an alternative to (or in addition to) uploading codebases for a project, you can obtain codebase files for the project by synchronizing a Source Control Management codebase repository to the Scan Server. For more information, see Configuring Source Code Management. The complete codebase for a project can consist of files that were both uploaded and synchronized to the Scan Server.

# Performing the Codebase Upload

The Scan Server to which you are uploading the codebase must be running (that is, the Tomcat server installed on the same instance as the Scan Server must be running).

The following describes how to upload the project codebase. You can repeat these steps to upload additional codebases for a project.

***Note*** ▪ *The initial maximum size for a codebase upload is 10 GB. However, the Code Insight System Administrator can configure Code Insight to increase or decrease this size as needed.*

***Task*** ***To upload a project codebase, do the following:***

1. Navigate to the **Summary** tab for the project for which you are uploading a codebase. (If necessary, see Opening the Project Summary Tab).
2. Click the **Upload Project Codebase** button to open the **File Upload** dialog.
3. Click **Select Archive File** to browse for the archive containing your codebase.
4. (Optional) Select **Delete existing project codebase files** to have Code Insight delete previously uploaded codebase files.

   ***Note*** ▪ *If you select to delete existing codebase files, a* ***Warning*** *dialog appears, asking you to confirm the deletion. Be aware that all existing codebase files for the project will be permanently removed from the Scan Server during the upload. If you rescan the project without replacing these files via a new upload, the scan results for the removed files will be permanently deleted.*

5. For **Archive File Expansion Options**, select the level of archive expansion you want to perform on the codebase:

   - **Uploaded file only**—Extract the files from the uploaded archive only. Any extracted archives are not expanded.
   - **Uploaded file and first-level archives only**—Extract the files from the uploaded archive and expand all first-level archives in the codebase. See the next step for additional configuration available when you select this option.
   - **Uploaded file and all contained archives**—Extract the files from the uploaded archive and expand archives at all levels (that is, archives within archives) in the codebase. See the next step for additional configuration available when you select this option.

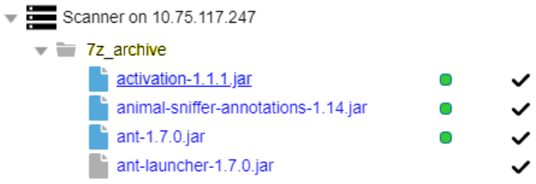
   For each expanded archive, the upload process extracts the archive contents to a folder automatically created with the archive name.

6. Configure settings that define the behavior of the upload process once archives are expanded. These settings are optional and are enabled only if **Uploaded file and first-level archives only** or **Uploaded file and all contained archives** has been selected.

    - **Delete archive files after expansion**—Remove those archives that have been expanded during an upload. (The archive is removed from the uploaded codebase after the upload is finished.) If you leave this option unselected, the archive is retained as an additional file directly under its parent folder. For examples of codebase trees that result based on how this option is configured, see More About Archive Expansion Behavior During Codebase Uploads.

    - **Append value to expanded archive directory name**—(Optional) Define a string to append to the name of any folder automatically created during the upload to store an archive's contents. After a scan, this appended string helps you to identify those folders in the codebase tree whose contents were extracted from archives, especially if the original archives were removed from the codebase during the upload (see the previous option).

      For example, suppose the appended value is `_archive`, and the upload process extracts an archive called `7z.zip`. After the upload process expands the archive, the name of the folder containing the archive contents becomes `7z_archive`, as shown in this example. Note that the example also shows that the 7z.zip archive has been removed due to the selection of **Delete archives after expansion**.



Scanner on 10.75.117.247
7z_archive
activation-1.1.1.jar ● ✓
animal-sniffer-annotations-1.14.jar ● ✓
ant-1.7.0.jar ● ✓
ant-launcher-1.7.0.jar ✓

      The appended value has a maximum of 20 characters and does not support certain special characters. (Hover over the ⓘ icon for a list of unsupported characters.)

7. Click **Upload**. Code Insight uploads your codebase file and attaches it to the selected project. You can now scan the uploaded codebase.

# Supported Archive Types for Expansion

The archive that you upload must be one of these types:

- `.zip`
- `.tar`
- `.tar.gz`
- `.7z`

The following archive types within the upload archive can be expanded either at the first-level only or recursively, depending on the **Expand Archive** option you select:

**Table 2-1 ▪** Expandable Archives

- `.7z`
- `.cpio`
- `.tar.xz`
- `.tgz`

**Table 2-1 ▪** Expandable Archives (cont.)

- `.tar`
- `.tar.bz2`
- `.tar.gz`
- `.apk`
- `.txz`
- `.tar.lzma`
- `.zip`

# More About Archive Expansion Behavior During Codebase Uploads

The following topics describe important information about how archives are expanded when a codebase is uploaded

- Handling of the Archives After Their Expansion
- Expansion of Archives Containing an Intermediary .tar File
- Multiple Codebase Uploads to the Same Project

## Handling of the Archives After Their Expansion

When an archive is expanded, its contents are extracted to a folder automatically created (with the archive's name) directly under the archive's parent folder. What happens to the archive once it is expanded depends on how the **Delete Archive Files After Expansion** is configured.

For example, suppose the archive `AppsSport.zip` is located in the `coreApps` directory in your codebase. The `AppsSport.zip` archive contains the files `hockey1.exe` and `tennis.exe`.

### Archive Retention Configured

If **Delete Archive Files After Expansion** is *not* selected, the archive `AppsSport.zip` is retained in its parent folder, `coreApps`, once it is expanded. The resulting codebase tree looks like this, where both `AppsSport.zip` and a folder `AppsSport`, containing the archive contents, are found directly under `coreApps`:

```
coreApps
---AppsSport
-----hockey1.exe
-----tennis1.exe
---AppsSport.zip
```

Note that, when an archive is retained, both the archive and its extracted files are processed during a codebase scan.

### Archive Removal Configured

If **Delete Archive Files After Expansion** is selected, the archive `AppsSport.zip` is removed once it is expanded, resulting in the following codebase tree:

```
coreApps
---AppsSport
```

```
-----hockey1.exe
-----tennis1.exe
```

## Expansion of Archives Containing an Intermediary .tar File

The `.tar.gz`, `.tgz`, `.txz`, and `.tar.xz` archive types and similar archives contain an intermediary `.tar` archive. The codebase upload extracts the intermediary `.tar` file from the archive, but applies the **Archive Expansion Options** configuration starting with the expansion of the intermediary `.tar` file, not the initial archive. The following example demonstrates this expansion behavior.

Suppose the archive `jars.tar.gz` has these contents, where the intermediary file is `jar.tar`:

```
jars.tar.gz
--jars.tar
----file-1.txt
----file-2.txt
----jar.zip
------abc.jar (color)
------xyz.jar
------classes.zip
--------corporation.class
--------employee.class
```

The uploaded codebase looks like this if **Uploaded File Only** for **Archive Expansion Options** is applied. The `jars.tar` is extracted from `jars.tar.gz`. The `jars.tar` archive is then expanded, but the `jar.zip` file (contained in `jars.tar`) is not expanded. Keep in mind that the original archive files are retained.

```
file-1.txt
file-2.txt
jar.zip
```

The uploaded codebase looks like this if the **Uploaded file and first-level archives only** option for **Archive Expansion Options** is used. The `jars.tar` is extracted from the initial `jars.tar.gz`. Once the `jars.tar` archive is expanded, the first-level `jar.zip` file (contained in `jars.tar`) is also expanded. However, the second-level `classes.zip` (contained in `jar.zip`) is not expanded.

```
file-1.txt
file-2.txt
\jar
---abc.jar
---xyz.jar
---classes.zip
jar.zip
```

The uploaded codebase looks like this if the **Uploaded file and all contained archives** option for **Archive Expansion Options** is used. The `jars.tar` is extracted from the initial `jars.tar.gz`. Once `jars.tar` archive is expanded, the first-level `jar.zip` file and the second-level `classes.zip` file are also expanded.

```
jars.tar
file-1.txt
file-2.txt
\jar
---abc.jar
---xyz.jar
---\classes
------Corporation.class
------Employee.class
```

```
---classes.zip
jar.zip
```

### Multiple Codebase Uploads to the Same Project

If multiple codebases are uploaded to the same codebase path for a given project (and existing codebase files are not deleted), the archives within all the codebases for the project are expanded based on the current **Archive Expansion Options** configuration. The following process demonstrates this behavior:

1. Create project1 and upload the codebase `codefiles1.zip`, using the **Uploaded file only** option. The contents of `codefile1.zip` are extracted. Archives in these contents are not expanded.
2. Upload `codefile2.zip` to the same project (at the same codebase path), this time using the **Uploaded file and all contained archives**. (Keep in mind that `codebase1.zip` was previously expanded with no further expansion of any archives in its contents.) Now all archives at all levels are expanded within the `codefile1` and `codefile2` codebases.
3. Upload `codefile3.zip` to the same project, this time using **Uploaded file and first-level archives only**. Now *only the first-level archives* in all three codebases are expanded.

If you upload multiple codebases to the same project, best practice is to keep track of the **Archive Expansion Options** configuration for each upload so that you can apply an appropriate configuration for the subsequent upload.

# Scanning the Codebase (Server Scans)

After a project's codebase has been uploaded to (or synchronized to) the Scan Server and the appropriate scan profile is selected, you can perform a server scan the codebase. The Scan Server must be running (that is, the Tomcat server installed on the same instance as the Scan Server must be running). The following instructions describe how to start scan on your codebase.

Refer to the Code Insight User Roles and Permissions appendix for role requirements to scan a codebase.

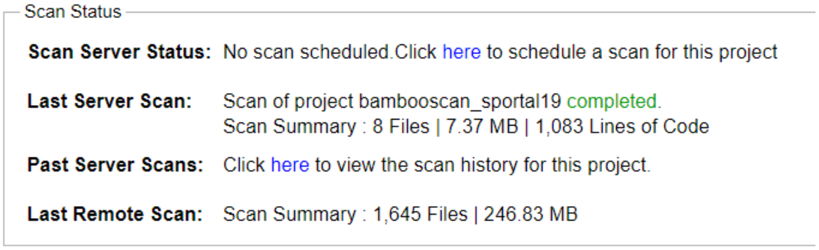
For information about the differences between server and remotes scans, refer to About Code Insight Scans.

***Task*** ***To start the scan, do the following:***

1. Navigate to the **Summary** tab for the project that you want to scan. (If necessary, see Opening the Project Summary Tab).
2. Click the **Start Scan** button (or the link in **Scan Status**) to start the scan. If other scans are running, the scan is queued and will automatically run based on queue order. (Click the link in **Past Server Scans** to view details about the scheduled scan.)

   ***Note*** ▪ *If the **Start Scan** button is disabled, see Actions to Take When the Start Scan Button is Disabled.*

   Information about the scan's progress appears in the **Scan Status** section on the **Summary** tab.



Scan Status

| | |
|---|---|
| **Scan Server Status:** | No scan scheduled.Click here to schedule a scan for this project |
| **Last Server Scan:** | Scan of project bambooscan_sportal19 completed.<br>Scan Summary : 8 Files \| 7.37 MB \| 1,083 Lines of Code |
| **Past Server Scans:** | Click here to view the scan history for this project. |
| **Last Remote Scan:** | Scan Summary : 1,645 Files \| 246.83 MB |

When the scan completes, **Last Server Scan** will display one of the following messages:

- **Completed**—The scan succeeded with no warnings during scan or analysis. This message appears on the screen in green.
- **Completed with warnings**—The scan succeeded but the analysis produced warnings. For more information, check the **scanEngineDetail** log for the Scan Server.
- **Failed**—The scan failed. This message appears on the screen in red. For more information, see Scan Failure Reasons and Troubleshooting Measures.

For an overall understanding of the scan results, see Overview of Scan Results.

3. Do any of the following:

   - Manage the project. For example, you can assign users to project analyzer or reviewer roles, define the project’s scan settings, configure an automated review and remediation workflow, configure a connection to a remote data source such as Perforce or Jira, and more. See Managing Projects for details.
   - Analyze the scan results, as described in Auditing Scan Results in the Analysis Workbench.
   - Generate the following standard reports and any applicable custom reports that have been added:
     - Project Report
     - Audit Report
     - Notices Report

## Actions to Take When the Start Scan Button is Disabled

The **Start Scan** button on the **Summary** page for a project is disabled if the Scan Server associated with the project is not available for scanning. In the button is disabled, check with the Code Insight System Administrator to determine the actual status of the server. If the administrator determines that the server is temporarily shut down, you can use the link in **Scan Status** on the **Summary** page to queue the scan. The scan will automatically run based on queue order once the server is active again.

However, if the server is disabled, you will need to create a new project for the codebase and associate it with an enabled Scan Server.

## Scan Failure Reasons and Troubleshooting Measures

The following lists possible causes and troubleshooting help for the failures of a server scan.

**Table 2-2 ▪** Scan Failure Causes and Troubleshooting Measures

| Scan Failure Cause | Troubleshooting Measures |
|---|---|
| **Scan server is not accessible** | Verify that the correct hostname and port for the selected Scan Server have been identified in Code Insight. |
| **Scan server is unable to access or read the CL files** | Verify that the correct Compliance Library (CL) path has been identified for the Scan Server. |
| **Scan server ran out of memory** | Ensure that the JVM heap (memory) size is adequate for running the Scan Server. (Recommended JVM heap sizes are listed in the *Code Insight Installation and Configuration Guide*.) |
| **Codebase file(s) are not accessible and cannot be read** | Verify (and adjust if necessary) the codebase file permissions. |
| **Codebase file(s) are encrypted and cannot be read** | Attempt to open the codebase files in **7-zip** or **winzip**. This application might provide a clearer description of the error than the scan process can. |
| **Codebase file(s) are corrupted and cannot be read** | Attempt to open the files in an external text editor. The editor might provide a clearer description of the error than the scan process can. |
| **Codebase file(s) contain unparseable characters** | This type of error is rare. Should it occur, verify that your database character set and collation settings are correct and that they match the requirements listed in the *Code Insight Installation and Configuration Guide*. |
| **Indexing of the scanned codebase files and results failed** | To help you identify the problem and troubleshoot, review the **scanEngineDetail** log for the Scan Server. |

**Table 2-2 ▪** Scan Failure Causes and Troubleshooting Measures (cont.)

| Scan Failure Cause | Troubleshooting Measures |
|---|---|
| **Unable to communicate with CodeAware** | This scan failure can occur when both of these conditions exist:<br>● Code Insight is running in a proxy-enabled environment.<br>● The Scan Server is running under its fully qualified domain name.<br>The Scan Server must call Code Insight Automated Analysis to analyze the codebase files. If the time required by Automated Analysis to analyze files exceeds the proxy server “read timeout” limit, the scan fails (even though Automated Analysis might still finish).<br>Try either of these methods to resolve this scan-failure issue:<br>● If the Core Server and Scan Server are running on the same instance, change the Scan Server hostname to `localhost`. (If you are running Code Insight in SSL mode, ensure that the SSL certificates accommodate the hostname change.)<br>● If the Core Server and Scan Server are not running on the same instance, try excluding the Scan Server from the proxy by adding its hostname to the `http.nonProxyHosts` property in the proxy details.<br>The *Code Insight Installation and Configuration Guide* provides information about configuring Code Insight to run in SSL mode or in a proxy-enabled environment. |
| **No alternative DNS name found that matches localhost** | This scan failure occurs when all these conditions exist:<br>● Code Insight is running in a proxy-enabled environment.<br>● The Core Server and Scan Server are installed on separate instances.<br>● Both servers are configured for SSL.<br>Try these methods to resolve the scan-failure issue:<br>● Ensure that the Secure Site SSL certificate on each instance has been properly configured.<br>● Try excluding the Scan Server from the proxy by adding its hostname to the `http.nonProxyHosts` property in the proxy details.<br>The *Code Insight Installation and Configuration Guide* provides information about configuring Code Insight to run in SSL mode or in a proxy-enabled environment. |

**Table 2-2 ▪** Scan Failure Causes and Troubleshooting Measures (cont.)

| Scan Failure Cause | Troubleshooting Measures |
|---|---|
| **Unable to find valid certificate** | This scan failure can occur when both of these conditions exist:<br>● The Core Server and Scan Server are installed on separate instances.<br>● Both servers are configured for SSL.<br>The scan fails when the Core Server is unable to communicate with the Scan Server.<br>Ensure that the Secure Site SSL certificate on each instance is valid and has been properly imported. (The *Code Insight Installation and Configuration Guide* provides information about procuring and importing these certificates as part of the SSL configuration for Code Insight.) |

# Overview of Scan Results

This section provides an overview of the following basic scan results, which can be examined in the **Analysis Workbench** and on the **Project Inventory** tab.Inventory

- Inventory
- Scan Evidence (**Analysis Workbench** only)
- License Information

Details about how these scan results pertain specifically to Analysts (who examine the results using the **Analysis Workbench**) and Reviewers (who examine the results using the **Project Inventory** tab) are found in later sections, Auditing Scan Results in the Analysis Workbench and Reviewing Published Inventory for a Project.

## Inventory

System-generated inventory is created by Code Insight during a scan and is available for view in the **Analysis Workbench** and, if automatically published, on the **Project Inventory** tab. An inventory item represents an explicit finding in the scanned codebase and can represent any of the following: top-level component, bundled component, component found inside an archive, or direct or transitive dependency component.

***Note ▪** Alternatively, you can review the published inventory across all projects. For details, see Viewing Inventory Across All Projects.*

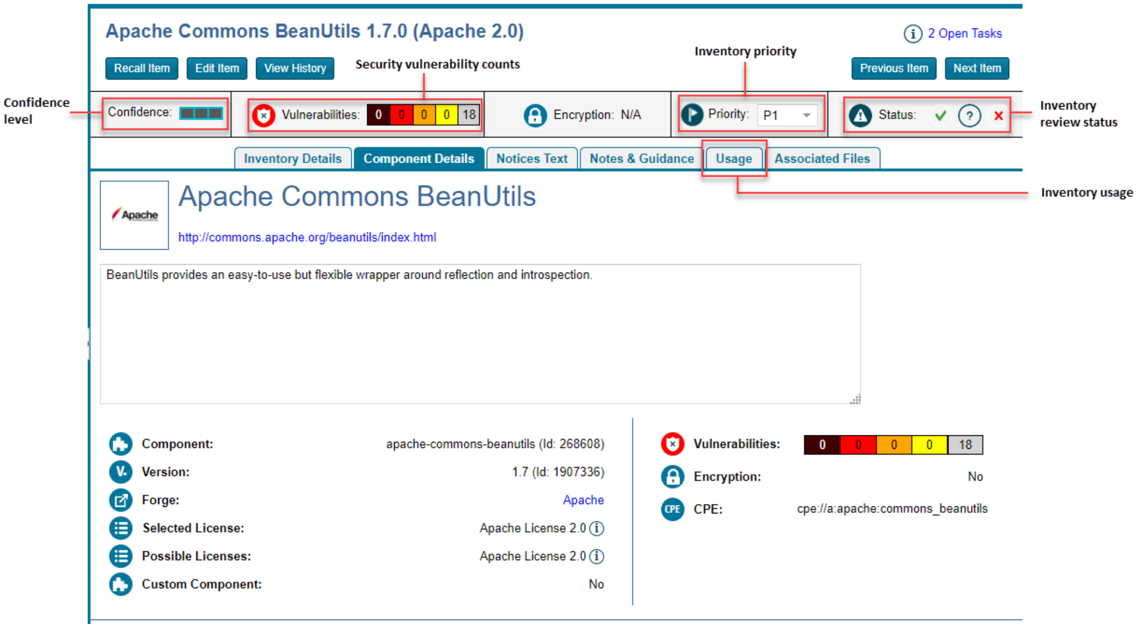
An inventory item typically has an associated component, version, license and list of security vulnerabilities, as well as other details about these elements. See Inventory Details Tab in the Analysis Workbench for a full description of the information collected by the scan.

These are some important elements about an inventory item that you can view at a glance:

- Review Status of Inventory
- Inventory Priority

- Inventory Confidence
- Security Vulnerabilities Associated with Inventory
- Inventory Usage Information

The following example highlights these elements for a given inventory item on the **Project Inventory** tab. Many of these same elements are available in the inventory view in the **Analysis Workbench**. (For more information about the **Analysis Workbench**, see Auditing Scan Results in the Analysis Workbench. For information about the **Project Inventory** tab, see Reviewing Published Inventory for a Project.)



Apache Commons BeanUtils 1.7.0 (Apache 2.0)

2 Open Tasks

Confidence: ▮▮▮ | Vulnerabilities: 0 0 0 0 18 | Encryption: N/A | Priority: P1 | Status: ✓ ? ✕

Apache Commons BeanUtils

http://commons.apache.org/beanutils/index.html

BeanUtils provides an easy-to-use but flexible wrapper around reflection and introspection.

| | |
|---|---|
| Component: | apache-commons-beanutils (Id: 268608) |
| Version: | 1.7 (Id: 1907336) |
| Forge: | Apache |
| Selected License: | Apache License 2.0 ⓘ |
| Possible Licenses: | Apache License 2.0 ⓘ |
| Custom Component: | No |

| | |
|---|---|
| Vulnerabilities: | 0 0 0 0 18 |
| Encryption: | No |
| CPE: | cpe://a:apache:commons_beanutils |

## Review Status of Inventory

During a scan, all inventory is checked against existing policies as defined in the Policy Profile. As a result, inventory is either automatically approved or rejected by policy or unaffected by policy. If inventory is not affected by policy, it should be manually reviewed, and the Policy Profile should be updated to reflect the review decision for future scans. The manual review process is described in detail in Reviewing Published Inventory for a Project.

For information about setting up policies that automate the inventory review process, see Managing Policy Profiles.

***Note** ▪ Unaffected inventory is labeled as **Draft** in the **Review Status** field in the **Analysis Workbench** and is represented as a circled X (for **Not Reviewed**) in the **Status** field on the **Project Inventory** tab.*

## Inventory Priority

The priority of an inventory item is meant to highlight the importance of that item during the inventory review process. Code Insight uses the following algorithms determine the default priority of an inventory item.

You can manually change the inventory priority by simply selecting a different priority from the **Priority** dropdown either in the **Analysis Workbench** or on the **Project Inventory** tab.

### For a "Component" Inventory Type

Code Insight sets the inventory priority to P1 if any of these circumstances exist:

- The inventory item has at least one associated security vulnerability with a severity of High (for CVSS v2.0) or Critical (for CVSS v3.x).
- The **Selected License** priority is P1 (see License Priority).
- No licenses are found (that is, the **Selected License** value is **I don't know** and no evidence of other licenses is found in the files associated with the inventory item).

Otherwise, when the user or system selects a component-version-license triad, the inventory priority is based on the license priority or highest associated security vulnerability severity, *unless* that would mean lowering an existing inventory priority.

***Note ▪** If the **Selected License** value for an inventory item is **I don't know** but evidence of other licenses is found in the files associated with inventory item, the inventory priority is based on the highest priority among the found licenses or the highest associated vulnerability severity.*

### For a "License-Only" Inventory Type

When a user selects a license for a license-only inventory item, the inventory priority is set to the license priority (see License Priority) *unless* that would mean lowering an existing inventory priority.

***Note ▪** Due to the algorithm used to calculate the priority, the system-generated inventory priority will never be lowered by the system. It can only be lowered explicitly by the user.*

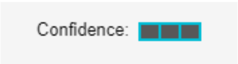
## Inventory Confidence

The Automated Analysis portion of the Code Insight Scan Server uses a variety of techniques to identify inventory items from the scanned code base. The Confidence level (High, Medium, or Low) of an inventory item is a measure of the strength of the discovery technique used to generate the inventory item and the certainty of the finding. It is derived by assigning a score to the following elements:

- The strength of the analysis technique that provided the metadata on the inventory item.
- The existence of this inventory item in the Code Insight data library: items that have matching components in the data library have higher levels of confidence.

The Confidence level is represented as a simple three-segment graph for each inventory item in the **Analysis Workbench** or on the **Project Inventory** tab. Three shaded segments indicate High confidence, two indicate Medium, and one indicates Low.

The following **Confidence** graph shows High confidence (with all three segments shaded):



The Confidence level is also available as a search criterion on the **Project Inventory** tab and can be used to quickly identify items that may require additional triage or review.

The following describes the Confidence levels:

- **High confidence**—An inventory item of High confidence means that either the item was identified with a specific and highly targeted rule or from the processing of a structured manifest file from a package manager (such as `pom.xml` for the maven package manager and `package.json` for the npm package manager). A High-confidence inventory item almost always matches with a component in the Code Insight data library and rarely requires further triage or review by the Analyst.
- **Medium confidence**—An inventory item of Medium confidence means that the item was identified using a more generic technique or by the processing of a secondary indicator to produce an inventory item. A Medium-confidence inventory item might or might not have a match to a component in the Code Insight data library and might require triage or review in order to be validate or further refine the finding.
- **Low confidence**—An inventory item of Low confidence means that the inventory item was identified using a very generic rule or an exploratory detection technique, and thus might represent a component of unknown origin. Inventory of Low confidence rarely have a match to a component in the Code Insight data library and should be further triaged and reviewed by an Analyst for accuracy and completeness.

The table below summarizes the various detection techniques and the corresponding confidence value:

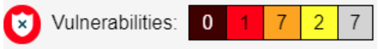
**Table 2-3 ▪** Confidences Levels Associated with Various Detection Techniques

| Detection Technique | Rule or Configuration File Used | Confidence Level |
|---|---|---|
| **Analyzers** | Primary | Hight |
| **Analyzers** | Secondary | Medium |
| **Search term analysis** | Rules with versions | High |
| **Search term analysis** | Component-only rules | Medium |
| **File name analysis** | Specific rules | High |
| **File name analysis** | Generic rules of certain type of components | Medium |
| **File name analysis** | Generic rules | Low |
| **Direct dependencies** | Based on package manager files (`pom.xml`, `package.json`, and so forth) | Low by default, but can increase to Medium if matching component + version is found in the Code Insight data library |
| **Transitive dependencies** | Based on lookups against respective repositories (maven, npm, and so forth) | Low by default, but can increase to Medium if matching component + version is found in the Code Insight data library |

# Security Vulnerabilities Associated with Inventory

Code Insight uses data from the National Vulnerability Database (NVD) and other advisories such as RubySec to report security vulnerabilities associated with your inventory items. The information from these sources is used to create vulnerability rankings and alerts.

The **Vulnerabilities** bar graph shows the current security-vulnerability counts by severity level for a given inventory item listed in the **Analysis Workbench** and on the **Project Inventory** tab (and in other locations):

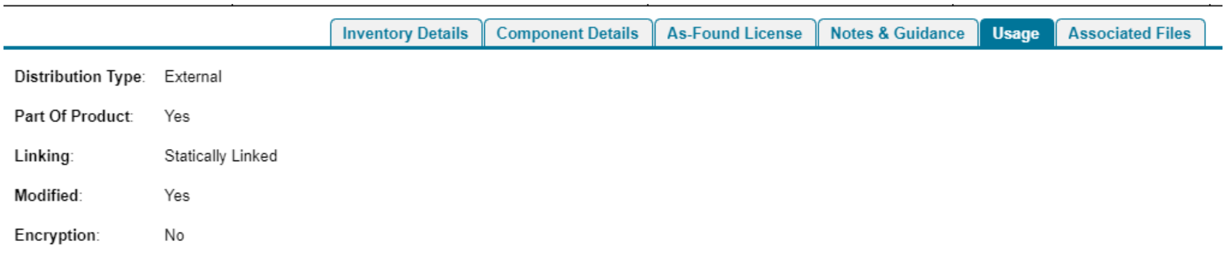

Vulnerabilities: 0 1 7 2 7

For more information about how to explore the security vulnerabilities associated with inventory, see Working with Security Vulnerabilities. This same section also describes how to suppress a vulnerability in your Code Insight instance if, for example, you have taken steps to protect your code against the vulnerability or if the vulnerability proves to be a "false positive".

# Inventory Usage Information

Code Insight provides the ability to see and edit usage information for a given OSS or third-party component associated with an inventory item. Usage information describes how a software package developed in your organization uses the OSS or third-party component. This information is important because it aids auditors and reviewers in determining how closely to monitor an inventory item for intellectual property (IP) and security risks and whether to approve or reject the item, create tasks for its remediation, and issue alerts and notifications pertaining to the item. Usage properties can also help users determine whether an inventory item should be included in Third-Party Notices and what steps need to be taken to satisfy license obligations and conditions of use. Finally, usage information can help to identify license conflicts and compatibility issues.

The inventory usage fields are available on the **Usage** tab for a given inventory item, as found both on the **Project Inventory** tab (shown below) and in the inventory view in the **Analysis Workbench**. You can update these fields when you manually create or edit inventory items.



| | |
|---|---|
| Distribution Type: | External |
| Part Of Product: | Yes |
| Linking: | Statically Linked |
| Modified: | Yes |
| Encryption: | No |

- **Distribution Type**—Indicates how you are distributing the OSS or third-party component associated with the inventory item. The distribution type can affect license priority and obligations.
    - **Externally** with your product, shipped to customers (outside of your organization, including a private cloud deployment at the customer's site)
    - As an application **hosted** in your company's data center (such as a SAAS application)
    - **Internally** only (such as an internal test framework included in the codebase but not distributed with the product)
    - Distribution method **unknown**

- **Part of Product**—Indicates whether the OSS or third-party component is part of the core product or an infrastructure piece such as a build or test tool. This information can affect whether third-party notices are required for this item.
- **Linking**—Indicates how your software package links to libraries in the OSS or third-party component—statically (the component is included in the materials), dynamically (the component is brought in at runtime), or not linked at all. Linking can affect license priority and obligations.
- **Modified**—Indicates whether code from the OSS or third-party package has been modified for use by your organization.
- **Encryption**—Indicates whether the component provides encryption capabilities used in the product. Encryption can affect export controls.

For explicit directions on viewing or editing inventory usage either in the **Analysis Workbench** or on the **Project Inventory** tab, see the following:

- Viewing Security Vulnerabilities for Inventory in the Analysis Workbench
- Viewing Usage Information for Project Inventory
- Editing Inventory from the Project Inventory Tab

# Scan Evidence

Scan evidence is generated by Code Insight during a scan and is available for view in the **Analysis Workbench** to any analyst assigned to the project. Scan evidence is typically an indicator of open-source or third-party content in the codebase. It can be useful for verifying system-generated inventory, identifying and creating additional inventory not discovered during scan, finding embedded licenses and copyrights in bundled code or archives, determining file origin, and locating stolen or borrowed code.

You can quickly view filter on and view the following evidence for codebase files in the **Analysis Workbench**. (For more details about examining evidence in the **Analysis Workbench**, see Examining and Managing Open-Source Evidence for a Given File and Viewing a Summary of Evidence Detected Across the Codebase.)

- **Exact Matches**—A whole-file match to a file in the Compliance Library
- **Source Matches**—Snippet-level matches to files in the Compliance Library
- **Copyrights**—Third-party copyright statements detected in the code
- **Emails/URLs**—Third-party emails and URLs detected in the code
- **Licenses**—Licenses detected in the code based on custom license patterns supplied by Electronic Update
- **Search Terms**—String matches based on pre-configured search terms provided by Code Insight and on custom search terms added by the user as part of the Scan Profile

## Scan Evidence from Scan-Agent Plugins

For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported in Code Insight.

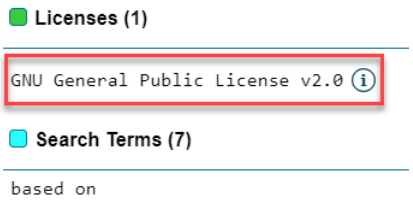
# License Information

The Code Insight scan detects license text and references to licenses in your codebase and enables you to examine this information is various ways, such as viewing the license or license-reference text highlighted in the codebase file itself (see Examining Evidence of Open-Source Copyrights, Email Addresses, URLs, Licenses, and Search Terms in a Given Non-Binary File). The scan can also generate inventory to which it associates a license based on the open-source or third-party component. The following topics provide an overview of other license-related information you can examine or manage:

- License Details from the Code Insight Data Library
- License Priority
- Reporting of Detected License Text through the As-Found Text Inventory Field
- Notices Text

## License Details from the Code Insight Data Library

The Code Insight scan can use information in the Code Insight data library to automatically select a license for a codebase file based on evidence found in the file. The scan also uses the data library to select a license for an automatically generated inventory item based on the inventory's open-source or third-party component. You can view details from the data library for this license by simply clicking ⓘ next to any license reference in the **Analysis Workbench** or the **Project Inventory** tab.



🟩 **Licenses (1)**

GNU General Public License v2.0 ⓘ

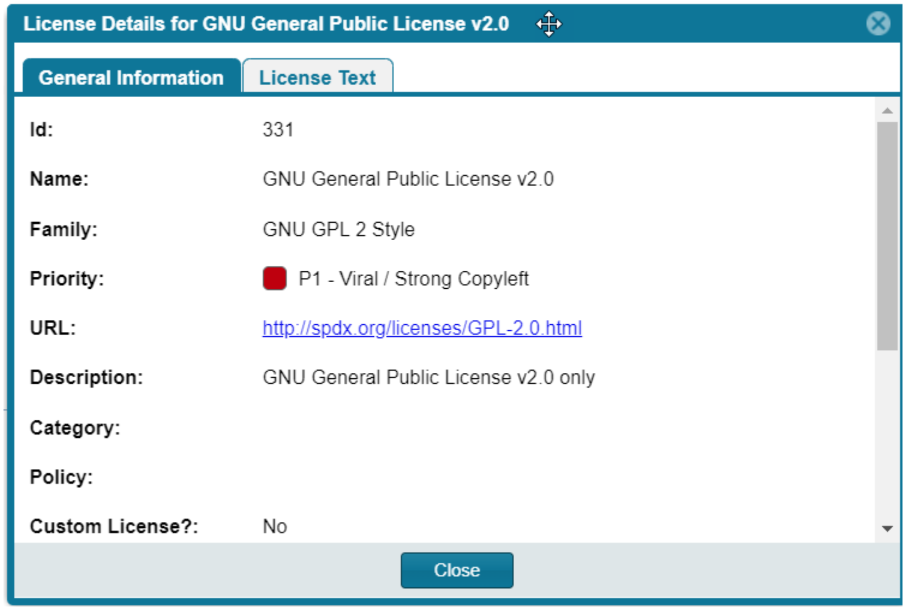
🟦 **Search Terms (7)**

based on

The **License Details** window is displayed containing the following information:

- A **General Information** tab that lists details such as the name of the license, its family, and the license priority assigned by Code Insight.
- A **License Text** tab that displays the complete license text (representing the external forge license text).

For descriptions of these fields in this window, see License Details Window.



License Details for GNU General Public License v2.0

Id: 331

Name: GNU General Public License v2.0

Family: GNU GPL 2 Style

Priority: 🟥 P1 - Viral / Strong Copyleft

URL: http://spdx.org/licenses/GPL-2.0.html

Description: GNU General Public License v2.0 only

Category:

Policy:

Custom License?: No

The following lists locations in Code Insight user interface where you can access the **License Details** window:

- In **Analysis Workshop**, you can view this information by codebase file or inventory item, as described in Viewing Details for Licenses Associated with Codebase Files and Viewing Details About Licenses Associated with Inventory in the Analysis Workbench, respectively.
- From **Project Inventory**, you view this information by inventory item, as described in Viewing Details About the Licenses Associated with Project Inventory.
- Whenever you create or edit an inventory item or preform a Component Lookup in the **Analysis Workbench** or from **Project Inventory**.
- Next to each License policy listed in the **Licenses** section on the Policy Details Window for a given policy profile. For more information, see Managing Policy Profiles.

## License Priority

You want to understand the priority of licenses in your codebase so you can handle them based on your corporate policies. Code Insight uses a default license priority to highlight which inventory items are more important than others, helping to define day-one work items.

Each license referenced in the **Analysis Workbench** and on the **Project Inventory** tab has one of the following priority values:

**Table 2-4 ▪** License Priorities

| Priority | Characteristics | Icon | Description |
|---|---|---|---|
| **P1** | Viral/Strong Copyleft | 🟥 | Usually, P1 licenses require immediate attention due to the possibility of tainting proprietary application code, an issue that can have significant business impact. |

**Table 2-4 ▪** License Priorities (cont.)

| Priority | Characteristics | Icon | Description |
|---|---|---|---|
| **P2** | Weak Copyleft/ Commercial/ Uncommon | 🟨 | The typical P2 license requires legal review and guidance based on corporate policies about the proper use of these types of licenses in your organization. |
| **P3** | Permissive/Public Domain | 🟩 | In general, P3 licenses are allowed and have minimal impact to an organization as long as license obligations are satisfied. The most common license obligation is properly attributing the use of an open source component to its author. This is the default priority. |

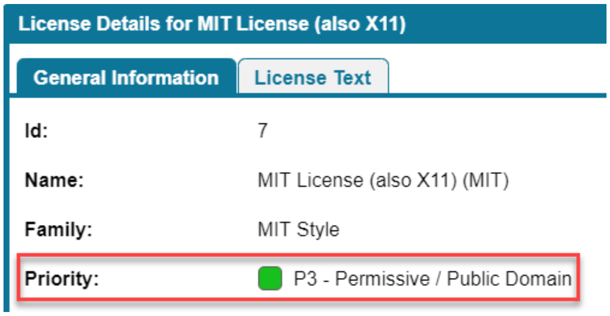
Inventory priority (see Inventory Priority) is a risk metric for the inventory item that takes license priority into account as one of the contributing factors. Inventory priority is set at scan time when the inventory item is created by the system or during inventory review. You can set or override the inventory priority at any time. License priority, on the other hand, is static and never changes. The license priority is supplied by the Electronic Update.

Inventory priority typically defaults to the license priority value unless a critical vulnerability exists or you manually override the inventory priority value (as described in Inventory Priority).

***Note*** ▪ *Code Insight REST APIs that reference the license entity, such as the Component Lookup API, include the license priority in the API response body.*

## Viewing the License Priority

You can view the license priority from the **License Details** window associated with the license. See License Details from the Code Insight Data Library for details on accessing this window.

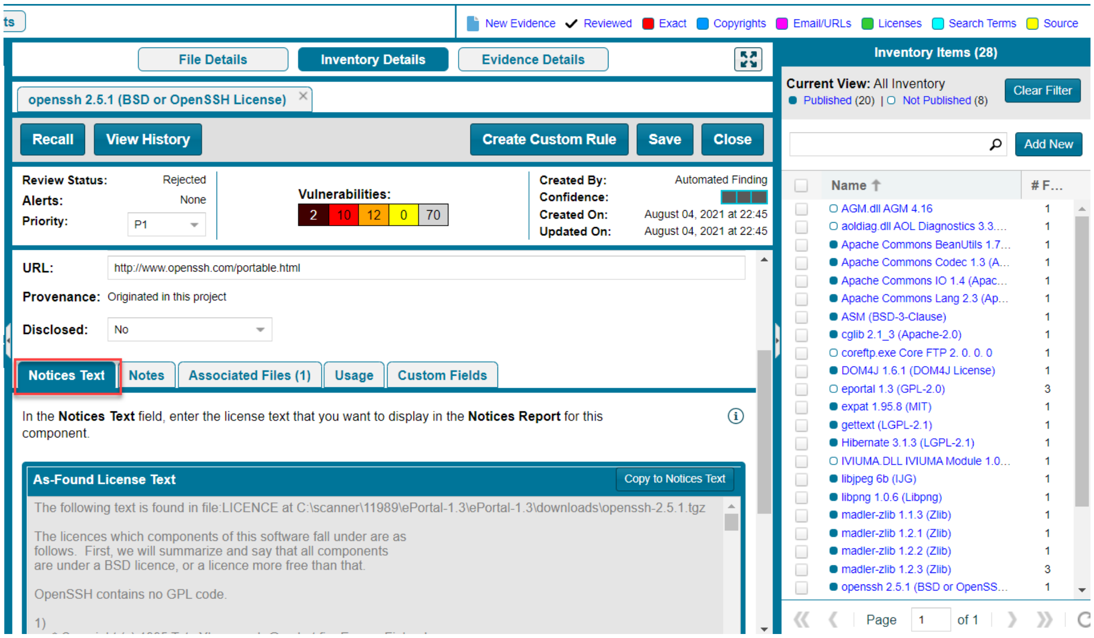

License Details for MIT License (also X11)

Id: 7
Name: MIT License (also X11) (MIT)
Family: MIT Style
Priority: 🟩 P3 - Permissive / Public Domain

# Reporting of Detected License Text through the As-Found Text Inventory Field

The **As-Found License** field (on the **Notices Text** tab for a selected inventory item in the **Analysis Workbench** and in **Project Inventory**) shows the license text or license references found in the scanned codebase.

The following shows the **Notices Text** tab in the **Analysis Workbench**.



openssh 2.5.1 (BSD or OpenSSH License) ×

Current View: All Inventory
● Published (20) | ○ Not Published (8)

Review Status: Rejected
Alerts: None
Priority: P1

Vulnerabilities: 2 10 12 0 70

Created By: Automated Finding
Confidence:
Created On: August 04, 2021 at 22:45
Updated On: August 04, 2021 at 22:45

URL: http://www.openssh.com/portable.html
Provenance: Originated in this project
Disclosed: No

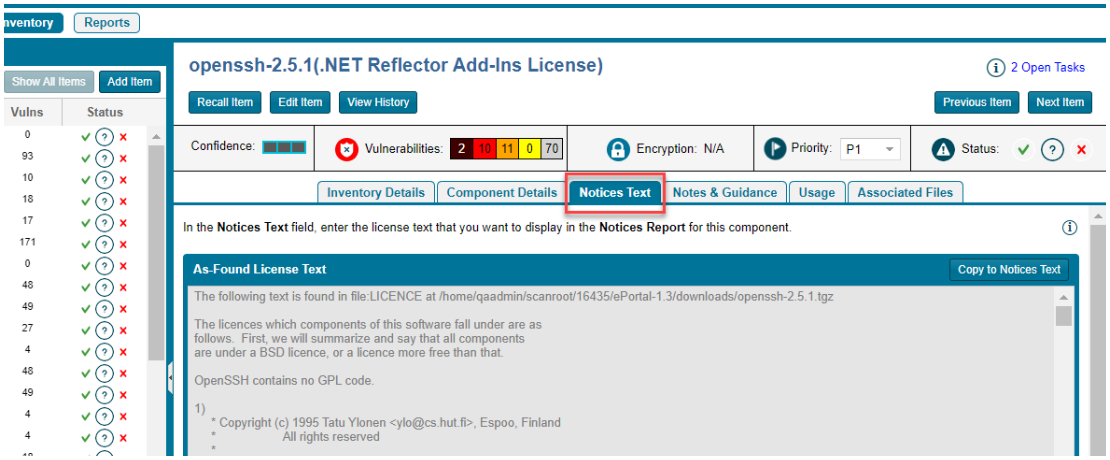
In the **Notices Text** field, enter the license text that you want to display in the **Notices Report** for this component.

As-Found License Text

```
The following text is found in file:LICENCE at C:\scanner\11989\ePortal-1.3\ePortal-1.3\downloads\openssh-2.5.1.tgz

The licences which components of this software fall under are as
follows.  First, we will summarize and say that all components
are under a BSD licence, or a licence more free than that.

OpenSSH contains no GPL code.

1)
```

| Name | # F… |
|---|---|
| ○ AGM.dll AGM 4.16 | 1 |
| ○ aoldiag.dll AOL Diagnostics 3.3.… | 1 |
| ● Apache Commons BeanUtils 1.7… | 1 |
| ● Apache Commons Codec 1.3 (A… | 1 |
| ● Apache Commons IO 1.4 (Apac… | 1 |
| ● Apache Commons Lang 2.3 (Ap… | 1 |
| ● ASM (BSD-3-Clause) | 1 |
| ● cglib 2.1_3 (Apache-2.0) | 1 |
| ○ coreftp.exe Core FTP 2. 0. 0. 0 | 1 |
| ● DOM4J 1.6.1 (DOM4J License) | 1 |
| ○ eportal 1.3 (GPL-2.0) | 3 |
| ● expat 1.95.8 (MIT) | 1 |
| ● gettext (LGPL-2.1) | 1 |
| ● Hibernate 3.1.3 (LGPL-2.1) | 1 |
| ○ IVIUMA.DLL IVIUMA Module 1.0… | 1 |
| ● libjpeg 6b (IJG) | 1 |
| ● libpng 1.0.6 (Libpng) | 1 |
| ● madler-zlib 1.1.3 (Zlib) | 1 |
| ● madler-zlib 1.2.1 (Zlib) | 1 |
| ● madler-zlib 1.2.2 (Zlib) | 1 |
| ● madler-zlib 1.2.3 (Zlib) | 3 |
| ● openssh 2.5.1 (BSD or OpenSS… | 1 |

This shows the **Notices Text** tab in **Project Inventory**.



openssh-2.5.1(.NET Reflector Add-Ins License)

Confidence:
Vulnerabilities: 2 10 11 0 70
Encryption: N/A
Priority: P1
Status: ✓ ? ✕

In the **Notices Text** field, enter the license text that you want to display in the **Notices Report** for this component.

As-Found License Text

```
The following text is found in file:LICENCE at /home/qaadmin/scanroot/16435/ePortal-1.3/downloads/openssh-2.5.1.tgz

The licences which components of this software fall under are as
follows.  First, we will summarize and say that all components
are under a BSD licence, or a licence more free than that.

OpenSSH contains no GPL code.

1)
     * Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland
     *                    All rights reserved
     *
```

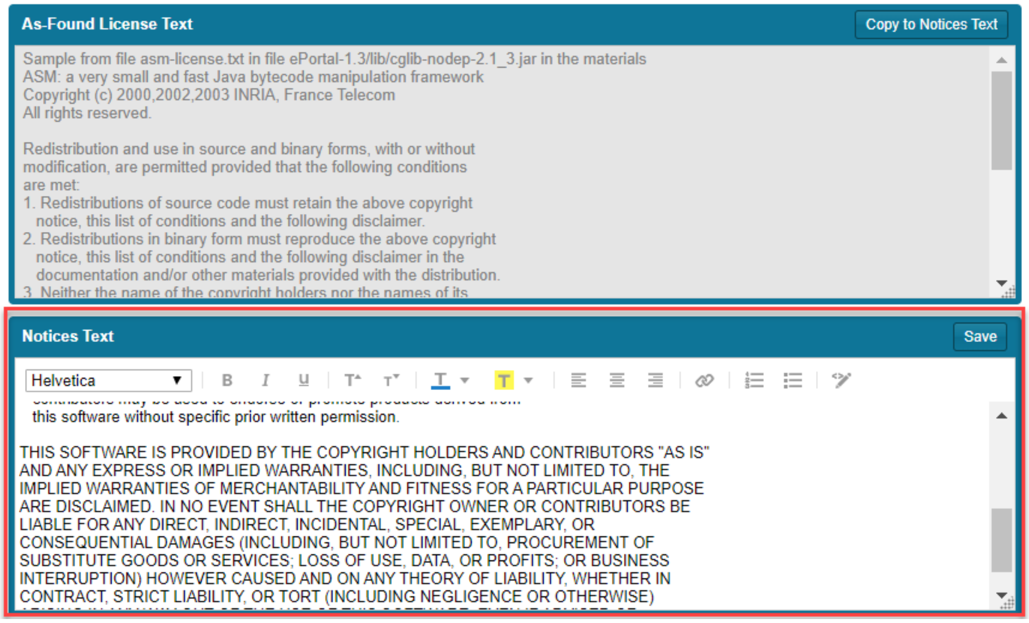
| Vulns | Status |
|---|---|
| 0 | ✓ ? ✕ |
| 93 | ✓ ? ✕ |
| 10 | ✓ ? ✕ |
| 18 | ✓ ? ✕ |
| 17 | ✓ ? ✕ |
| 171 | ✓ ? ✕ |
| 0 | ✓ ? ✕ |
| 48 | ✓ ? ✕ |
| 49 | ✓ ? ✕ |
| 27 | ✓ ? ✕ |
| 4 | ✓ ? ✕ |
| 48 | ✓ ? ✕ |
| 49 | ✓ ? ✕ |
| 4 | ✓ ? ✕ |
| 4 | ✓ ? ✕ |

The **As-Found License Text** content cannot be edited, but you can copy it to the **Notices Text** field (also on the **Notices Text** tab) if you need to modify it. The text in the **Notices Text** field is considered final and is included in the Notices report. Otherwise, if the **Notices Text** field is empty, Code Insight uses the contents of the **As-Found License Text** field as the license text for the inventory item in the report. If both fields are empty, the report uses the license content from Code Insight data library.

For more information about finalizing license text for the Notices report, see Finalizing the Notices Text for the Notices Report.

## Notices Text

The **Notices Text** field (on the **Notices Text** tab for a selected inventory item in the **Analysis Workbench** and in **Project Inventory**) can be used to finalize the license text for use in the Notices report. For example, you can copy the contents of the **As-Found License Text** field to this field and modify the text as needed. Alternatively, you can simply provide your own Notices content in the **Notices Text** field.



```
Sample from file asm-license.txt in file ePortal-1.3/lib/cglib-nodep-2.1_3.jar in the materials
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000,2002,2003 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
```

```
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
```

When the Notices report is run, the content of the **Notices Text** field item is pulled into the report if this field contains information. If the **Notices Text** field is empty, the content of the **As-Found License Text** field is used in the report. If both fields are empty, the report uses the license content from Code Insight data library.

For more information about finalizing license text for the Notices report, see Finalizing the Notices Text for the Notices Report.

# Auditing Scan Results in the Analysis Workbench

After a scan has been performed on a project codebase uploaded to a Scan Server, an Analyst for the project analyzes, or *audits*, the results of the scan in the **Analysis Workbench**. The following sections describe the role of the Analyst and the tasks involved with a scan audit:

- Role of an Analyst
- Opening the Analysis Workbench
- Searching the Codebase Files
- Examining and Managing Open-Source Evidence for a Given File
- Viewing a Summary of Evidence Detected Across the Codebase
- Managing the Codebase Files

- Managing Inventory in the Analysis Workbench

A *remote scan* performed using a Code Insight scan-agent plugin on a remote build server generates only inventory that is then sent to a project on the Code Insight Core Server.

# Role of an Analyst

The role of a project Analyst in Code Insight is to transform the evidence uncovered by the Scan Server into an *inventory item*. Analysts create inventory items that associate files in your codebase to open-source and third-party projects, called *components* in Code Insight. For example, Analysts might evaluate files with a copyright of “Copyright (c) 2015 to 2021 Mark Smith” and a license match to the license used by the “zlib” component. The Analyst would then associate these files with an inventory item for the “zlib” open-source component and mark the files as *reviewed* to register progress.

The Analyst will evaluate all of the evidence within a codebase, create inventory items where appropriate, mark the analyzed files as reviewed, and finally *publish* them. The remaining sections in Auditing Scan Results in the Analysis Workbench describe these tasks.

Once published, the inventory will be available for reporting and review by Legal, Security, and Development teams, as described in Reviewing Published Inventory for a Project. The ultimate goal of both the audit and the review/remediation processes is to produce a complete and accurate inventory of open-source and third-party code within your products—sometimes referred to as a Bill of Materials (BOM).

Refer to the Code Insight User Roles and Permissions appendix for more information about Analyst role required to access the **Analysis Workbench** and to analyze and act on scan results.

# Opening the Analysis Workbench

Use the following procedure to open the **Analysis Workbench**.

***Task*** ***To open the Analysis Workbench, do the following:***

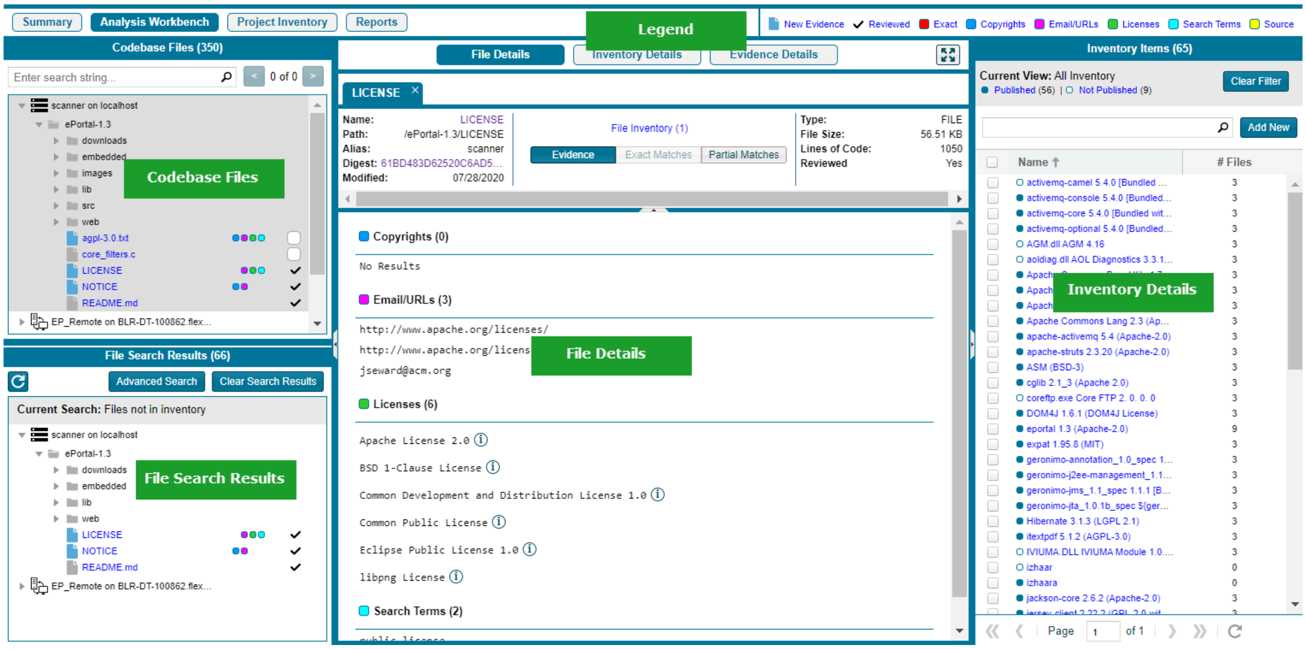
1. Open a project from the **Projects** view. (For instructions, see Opening a Project.)

   The project opens to either its **Project Inventory** or **Summary** tab. If you have Analyst permissions, the **Analysis Workbench** tab is available next to either or both of these tabs.

2. Open the **Analysis Workbench** tab to begin the analysis process. See the next section, The Analysis Workbench Layout.

## The Analysis Workbench Layout

The following is a view of the Code Insight **Analysis Workbench**, showing the various areas of the page.



After you click **Analysis Workbench**, the following information appears in various panes and tabs in the workbench:

- Codebase Files Pane (Top Left Pane)
- File Search Results Pane (Bottom Left Pane)
- Files Details Tab (Middle Pane)
- Inventory Details Tab (Middle Pane)
- Evidence Details Tab (Middle Pane)
- Inventory Items Pane
- Legend

## Codebase Files Pane (Top Left Pane)

This pane lets you browse a codebase tree listing the project’s scanned files that you uploaded or synchronized to the Scan Server or that were scanned remotely by a Code Insight scan-agent plugin. The codebase tree provides the following:

- Scan Server’s Base Node
- Scan Agent’s Base Node
- Types of Evidence Found in a File
- Review Indicator
- Access to File Details

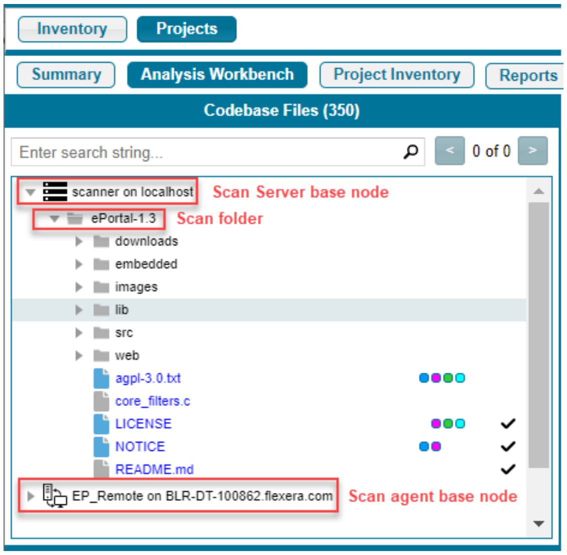
### Scan Server's Base Node

The project's codebases scanned by a Scan Server are listed under the *Scan Server's base node*, which is identified both by the Scan Server's unique alias and by the name of instance on which the server is hosted. This base node has the format <scanServerAlias> on <scanServerHost > (such as **Scanner03 on localhost**).

### Scan Agent's Base Node

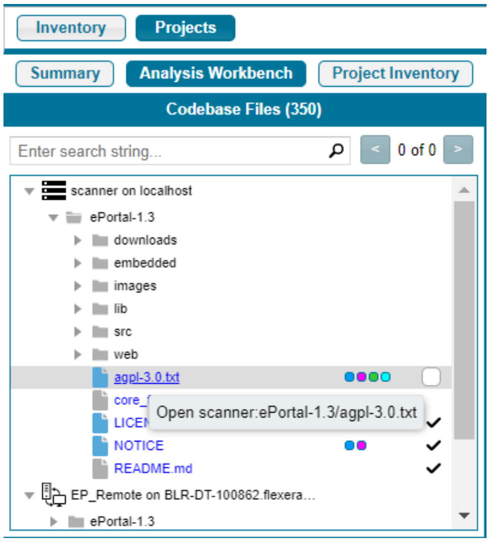
The remote codebases scanned by a scan-agent plugin are listed under the *scan agent's base node*, identified both by a unique alias for the scan agent and by the instance on which the agent is hosted. This base node has the format <scanAgentAlias> on <scanAgentHost> (such as **EP_Remote on BLR-DT-100555.ECompany.com**).

The unique, user-defined alias provided during scanner setup (for either a Scan Server or a remote scan agent) is a descriptive name used to represent the scan-root container for the scanner. The base node then—as a combination of both the alias and the host instance name—provides a more meaningful representation for the absolute scan-root path for the scanner. (The actual absolute scan-root path for each scanner associated with the project is available on the project's **Summary** tab.)



When the **Analysis Workbench** for a given project first opens, the codebase tree expands only the first base node. Under that node, only the first top-level (scan) folder is expanded, showing the first-level codebase folders and files directly under that scan folder. These first-level folders as well as all other base nodes and folders are collapsed and need to be expanded manually as needed.

When you hover over a file name in the codebase tree, the name is shown in an <alias>:<relativeFilePath> format, where <alias> is the alias of the Scan Server or scan agent and <relativeFilePath> is the file path relative to the absolute scan-root path on host instance. (See the following example where, when a user hovers over the codebase file **agpl-3.0.txt**, located directly under the scan folder **ePortal-1.3**, the file name is shown as **scanner:ePortal-1.3/agpl-3.0.txt**.)



### Types of Evidence Found in a File

The types of evidence found in a given file show as color-coded icons to the right of the file name. The color coding is identified in the legend located in the right side of the **Analysis Workbench** header. (See Legend.)

Note the following:

- For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported in Code Insight (indicated by the green icon 🟩 for those files that contain license evidence). No other type of evidence is reported for such files at this time.
- Some source files contain indications that they are data files, generated code, or common code that is widely used in many open source projects. In those cases, Code Insight records the fact that source matches exist but does not store all of the source match data. These files are indicated in the Analysis Workbench with an icon (☒).

### Review Indicator

A checkmark at the end of file row indicates that the file has been reviewed.

### Access to File Details

When you click a file, its metadata, content, and evidence is shown in the middle pane (**File Details** pane).

## File Search Results Pane (Bottom Left Pane)

This pane lists the results of file searches against the codebase. The results are shown in a codebase tree that has the same format, properties, and behavior as the codebase tree in the **Codebase Files** pane (see Codebase Files Pane (Top Left Pane)). For more information about file searches, see Searching the Codebase Files.

## Files Details Tab (Middle Pane)

This tab lists a summary of evidence found in the file currently selected in the codebase tree. For non-binary files, the tab can show the actual file content, including evidence highlighted in color. For a binary file, the tab can show strings of possible third-party evidence. From this tab, you can research the source of the possible third-party code and, if necessary, ultimately create an inventory item explaining the scan findings.

***Note*** ▪ *The middle pane toggles between the **File Details** tab, the **Inventory Details** tab, and the **Evidence Details** tab, depending on whether an codebase file or an inventory item is selected or if you explicitly click one of these tabs.*

## Inventory Details Tab (Middle Pane)

This tab shows information about the inventory item selected in the **Inventory Items** pane (see Inventory Items Pane). This information includes component and license details, inventory priority, inventory confidence level, as well as auditing and guidance notes, associated files, and the third-party notices associated with the inventory item. From this tab, you can edit the inventory item, recall it from its published state, and create a custom rule based on your findings that future scans on other projects can use to automatically generate inventory.

## Evidence Details Tab (Middle Pane)

This tab displays all instances of copyright, license, email, URL, and search-string evidence uncovered by the scan across all files in the project. (You must explicitly click the **Evidence Details** tab to see this information.) The list of evidence instances is organized by evidence type and is sortable.

***Note*** ▪ *For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported in Code Insight. The **Evidence Details** pane visibly shows any license evidence found in remotely scanned files.*

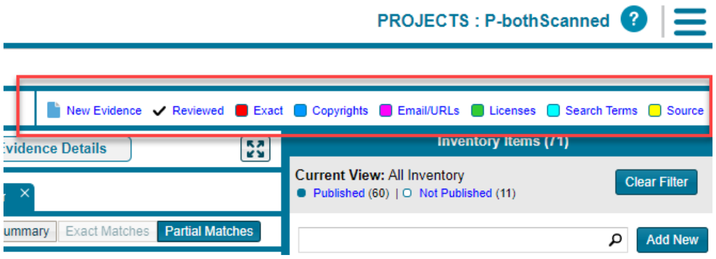
To filter the files in the **File Search Results** pane to focus attention on those files containing particular evidence (such as specific copyright), select one or more evidence instances (rows) on the **Evidence Details** tab, and click **Search Files**. The **File Search Results** pane shows the files containing that evidence. For more information, see Filtering the Codebase by a One or More Specific Instances of Evidence.

## Inventory Items Pane

(Right pane) This pane lists all the inventory currently identified in the codebase. Click any inventory item in the list to display the its details on the **Inventory Details** tab.

## Legend

(Right side of the Analysis Workbench header) This legend provides a color key for the various types of evidence and for file-review status referenced in the **Analysis Workbench**. The **Legend** is interactive. You can click it to filter what appears in the **File Search Results** pane.For more details, see Using the Filter Legend Options to Filter the Codebase.



# Searching the Codebase Files

Code Insight offers various methods for searching the list of scanned codebase file:

- Searching for Codebase Files Based on Name
- Searching for Codebase Files Based on Search Criteria
- Creating and Editing File Searches
- Using the Filter Legend Options to Filter the Codebase
- Filtering the Codebase by a One or More Specific Instances of Evidence

## Searching for Codebase Files Based on Name

You can perform a file search to find files based on the file name to concentrate the **Analysis Workbench** display on files of interest for conducting your analysis of the scan results. The following limitations apply:

- There is no support for wildcard specifications. The comparison is a case-insensitive filename containing the complete search string.
- Only the first 1,000 matching files are returned by the file search.

This specific search highlights the search results in the **Codebase Files** pane, unlike the other file searches (described in the sections that follow) that display the search results in the **File Search Results** pane.

*Task* ***To perform a file search based on name, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the search text box in the **Codebase Files** pane, enter the partial or full name of the file or folder that you want to search and press **Enter**. You must type at least three characters to initiate the filename search. The text box is highlighted with a red border if you enter fewer than three characters, and an error message is shown in a tooltip.
3. When a match is found, the tree in the **Codebase Files** pane is expanded as much as necessary to highlight the matching file. The file details are not open until you click on the file in the tree.
4. Select the **Next Match** (**>**) and **Previous Match** (**<**) buttons next to the search string box to navigate the results of the search.
    - **Files**—If the Previous or Next match button reaches a file, that file will be highlighted in the codebase tree, and the search term will be highlighted in yellow.
    - **Folders**— If the Previous or Next match button reaches a folder, that folder will be highlighted in the codebase tree and the search term will be highlighted in yellow. The folder will also be automatically expanded one level so that you can see its child items.

    The counter between the buttons indicates the total number of matches and the current match number.
5. (Optional) Click the name of a file to display its contents in the **File Details** tab.
6. (Optional) Click the **X** to clear the search string.

# Searching for Codebase Files Based on Search Criteria

You can perform a file search to find files based on the file name to concentrate the **Analysis Workbench** display on files of interest for conducting your analysis of scan results.

*Task* ***To perform a file search by criteria, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Click **Advanced Search** in the **File Search Results** pane. The **Advanced File Search** dialog appears.
3. Select an existing search filter or add a new one:
   - To select an existing search filter, click the name of the filter; and then click **Search** to begin the search with the selected filter. (For a list of built-in search filters that Code Insight provides, see Codebase Filters Provided with Code Insight.)
   - To create and run a new search filter, see Creating and Editing File Searches.

   Results are listed in the **File Search Results** pane.

## Codebase Filters Provided with Code Insight

For your convenience, Code Insight provides the following built-in search filters by which to search codebases. You can copy these filters to create custom filters; or, if necessary, you can edit or delete them, as described in Creating and Editing File Searches.

**Table 2-5 ▪** Codebase Filters Provided by Code Insight

| Code Insight Predefined Filter | Filters to... |
|---|---|
| **Files that require additional analysis** | Files that contain evidence but are *not* reviewed or associated with inventory. |
| **Files with new evidence requiring another look** | Files that contain new evidence but are already marked as reviewed or are currently associated with inventory. |
| **Files that need to be marked as reviewed** | Files that are associated with inventory but are not marked as reviewed. |
| **Files with possible commercial content** | Files containing copyrights that include commercial names. |
| **Files not in inventory** | File that are not associated with any inventory item. |

# Creating and Editing File Searches

You can supplement the built-in filters with custom filters to focus on scan data that are important to you. For example, you can create a new file search from scratch or from a copy of an existing search. You can also edit searches.

Refer to the following topics:

- Creating a New File Search
- Editing a File Search
- Copying a File Search
- Deleting a File Search
- Available Search Criteria for Building Codebase Filters

Any new searches you create or any copies or edits you make are available to all users in your Code Insight system. Likewise, any searches that you delete are no longer available to users in the system.

## Creating a New File Search

Use either procedure to create a new file search:

- Create a File Search from Scratch
- Create a File Search from a Copy

### Create a File Search from Scratch

Use this procedure to create a file search from scratch.

***Task*** ***To create a new search from scratch, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **File Search Results** pane, click **Advanced Search**. The **Advanced File Search** dialog appears.
3. Click **Add New**. The **Create Filter** dialog appears.
4. In the **Name** field, type a name for the search.
5. (Optional) In the **Description** field, type a description of the search. For example, type text that explains what the filter will search for.
6. Use this procedure to enter values in the **Criteria** fields. (For details about the available criteria, see Available Search Criteria for Building Codebase Filters.)
    a. Select a criterion from the drop-down **Select Search Field** menu.

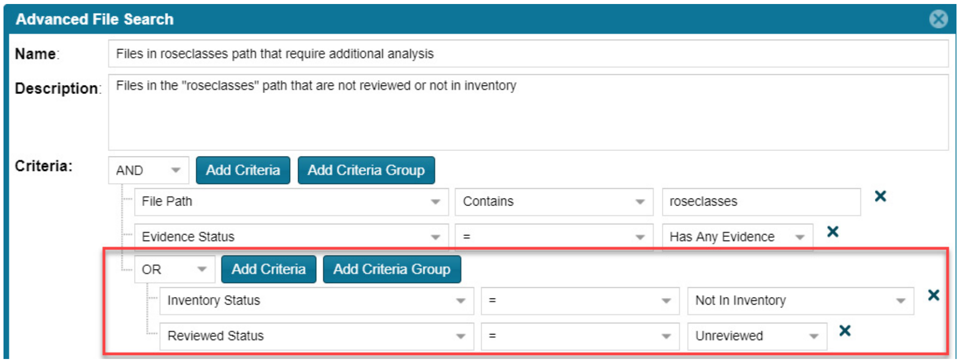
    ***Note*** ▪ *When creating a new filter, consider that a Code Insight scan-agent plugin on a remote system currently reports only license evidence for its scanned files. The fields applicable for searching such files are limited to the following:* ***File Size****,* ***File Path****,* ***File Digest****,* ***Review Status****,* ***Inventory Status, Evidence status, Has license matches, Does not have license matches,*** *and* ***License****.*

    b. If applicable, select a search operation (for example, **Contains** or **=**) and provide a search string or value.

c. To add another criterion, click **Add Criteria**, select a Boolean value to define how the criteria is applied, and repeat the previous steps to define the criterion. Repeat for each criterion added.

d. To add a group of criteria that serves as a criterion, click **Add Criteria Group**, and repeat steps a through c to create the group.

The following shows an example of a criteria group.



Advanced File Search

Name: Files in roseclasses path that require additional analysis

Description: Files in the "roseclasses" path that are not reviewed or not in inventory

Criteria: AND | Add Criteria | Add Criteria Group

| | | | |
|---|---|---|---|
| File Path | Contains | roseclasses | × |
| Evidence Status | = | Has Any Evidence | × |

OR | Add Criteria | Add Criteria Group

| | | | |
|---|---|---|---|
| Inventory Status | = | Not In Inventory | × |
| Reviewed Status | = | Unreviewed | × |

7. Determine how you want to proceed:

- **Save**—Save your search but do not execute it.
- **Save and Search**—Save your search filter and then execute it.
- **Search without Saving**—Execute the search without saving it.
- **Cancel**—Do not execute the search or save it.

## Create a File Search from a Copy

Use this procedure to create a file search from a copy of an existing search. Using a copy keeps the existing search in tact and provides a template for creating the new one.

*Task* ***To create a search from a copy of an existing one, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **File Search Results** pane, click **Advanced Search**. The **Advanced File Search** dialog appears.
3. Locate the search you want to copy, and click 🗋 in its entry. A new file search is created with “Copy of...” in its title.
4. In the entry for the copy, click ✎ to open the filter properties.
5. In the **Name** field, type a new name for the search.
6. Modify the filter criteria as needed and save the changes. See Create a File Search from Scratch for any additional instructions.

## Editing a File Search

Use these instructions to edit an existing file search.

***Task*** ***To edit a file search, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **File Search Results** pane, click **Advanced Search**. The **Advanced File Search** dialog appears.
3. In the entry for the file search you want to edit, click ✎ to open the filter properties.
4. Modify the filter criteria as needed and save the changes. See Create a File Search from Scratch for any additional instructions.

## Copying a File Search

Use these instructions to create a copy an existing file search (as a backup or a basis for creating a new search, for example).

***Task*** ***To make a copy of an existing file search, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **File Search Results** pane, click **Advanced Search**. The **Advanced File Search** dialog appears.
3. Locate the search you want to copy, and click 🗋 in its entry. A new file search entry is created with “Copy of...” in its title.

## Deleting a File Search

Use these instructions to delete a file search. When you delete the search, it is removed from the system and no longer available to users.

***Task*** ***To delete an existing file search, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **File Search Results** pane, click **Advanced Search**. The **Advanced File Search** dialog appears.
3. Locate the search you want to delete, and click ✖ in its entry.

   A message is displayed to confirm that you want to delete the search.
4. Click **Yes** to remove the file search from the system.

## Available Search Criteria for Building Codebase Filters

Code Insight provides the following search criteria on which to build codebase search filters.

***Note*** ▪ *A Code Insight scan-agent plugin on a remote system currently reports only license evidence for its scanned files. The fields applicable for searching these scanned files are limited the following:* ***File Size****,* ***File Path****,* ***File Digest****,* ***Review Status****,* ***Inventory Status, Evidence status, Has license matches, Does not have license matches, License****.*

**Table 2-6** ▪ Available Criteria for Building Codebase Search Filters

| Criterion Type | Available Criterion | Operation | Criterion Value | Criterion will filter to those files with... |
|---|---|---|---|---|
| **File Properties** | File Size (in KB) * | Select **<** or **>**. | Enter or select the file-size. | A size is less than or greater than the specified size. |
| | File Path* | Select **Contains**, **Ends With**, or **Doesn't Contain**. | Enter the file-path string or partial string. | A path containing a match to the specified string. |
| | Reviewed Status * | Only **=** is available. | Select **Reviewed** or **Unreviewed**. | The selected review status. |
| | File Digest * | Only **=** is available. | Enter the file's MD5 value. | The exact MD5 specified. |
| | Evidence Status | Only **=** is available. | Select **Has Evidence**, **Has New Evidence** (since last scan), or **Has No Evidence**. | The evidence status specified. |
| | Inventory Status * | Only **=** is available. | Select one status:<br>● **In Inventory**—Files associated with inventory.<br>● **Not in inventory**—Files not associated with inventory.<br>● **Low Confidence File Inventory**—Files associated with low-confidence inventory. See Inventory Confidence for details. | The selected inventory status. |

**Table 2-6 ▪** Available Criteria for Building Codebase Search Filters (cont.)

| Criterion Type | Available Criterion | Operation | Criterion Value | Criterion will filter to those files with... |
|---|---|---|---|---|
| | Scan Status | Only **=** is available. | Select one:<br>● **Successfully Scanned**—Files that were successfully scanned in the most recent scan. Conversely, this criterion is helpful in determining which files were *not* scanned in certain situations (for example, if you were forced to stop the scan before it finished or if the Scan Server crashed).<br>● **Skipped Source Matching**—Files that were ignored during source-code matching. | The selected scan status. |
| **File Evidence—Source Matches** | Has Source Matches | — | — | Source-code snippets that match snippets of open-source or third-party files stored in the Code Insight Compliance Library. |
| | Does Not Have Source Matches | — | — | No evidence of such source-code snippets. |
| **File Evidence—Search Term Matches** | Has Search Term Matches | — | — | The specified search-term string. (Search terms, defined in the scan profile, are used to identify open-source or third-party evidence in codebase files.) |
| | Does Not Have Search Term Matches | — | — | No evidence of the specified search term. |
| | Search Term | Select **=** or **Contains**. | Enter the full search-term or a partial search-term string. | — |

**Table 2-6 ▪** Available Criteria for Building Codebase Search Filters (cont.)

| Criterion Type | Available Criterion | Operation | Criterion Value | Criterion will filter to those files with... |
|---|---|---|---|---|
| **File Evidence—License Matches** | Has License Matches | — | — | Evidence of the open-source or third-party license selected for **License**. |
| | Does Not Have License Matches | — | — | No evidence of the selected license. |
| | License | Only **=** is available. | Select the open-source or third-party license by which to filter codebase files. | — |
| **File Evidence—Exact Matches** | Has Exact Matches | — | — | Entire content that exactly matches the content of open-source or third-party files stored in the Code Insight Compliance Library. |
| | Does Not Have Exact Matches | — | — | No exact match to the entire content of any open-source or third-party file in the Compliance Library. |
| **File Evidence—Email/URL Matches** | Has Email/URL Matches | — | — | Evidence of the open-source or third-party email address or URL specified for **Email/URL**. |
| | Does Not Have Email/URL Matches | — | — | No evidence of the specified email or URL. |
| | Email/URL | Select **=** or **Contains**. | Enter the open-source or third-party email or URL (or partial value) by which to filter codebase files. | — |

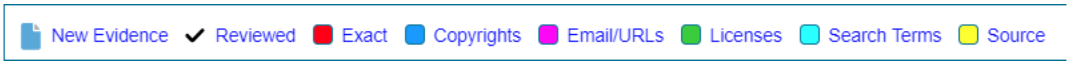
**Table 2-6 ▪** Available Criteria for Building Codebase Search Filters (cont.)

| Criterion Type | Available Criterion | Operation | Criterion Value | Criterion will filter to those files with... |
|---|---|---|---|---|
| **File Evidence—Copyright Matches** | Has Copyright Matches | — | — | Evidence of the copyright holder (specified for **Copyright Holder**) or copyright statement (specified for **Copyright Statement**). |
| | Does Not Have Copyright Matches | — | — | No evidence of the specified copyright or copyright holder. |
| | Copyright Holder | Select = or **Contains**. | Enter the open-source or third-party copyright holder (or partial value) by which to filter codebase files. | — |
| | Copyrights | Select = or **Contains**. | Enter the open-source or third-party copyright statement (or partial value) by which to filter codebase files. | — |

***** Criterion currently supported for searches on scanned remote files (that is, files scanned by a Code Insight scan-agent plugin on a remote system).

## Using the Filter Legend Options to Filter the Codebase

The codebase filter legend in the ribbon at the top right of the **Analysis Workbench** provides a means of filtering the codebase by evidence type or by files with a “Reviewed” status. For example, by simply clicking an icon (or its label), you can filter to all files containing copyright or email-address evidence or that are exact matches to third-party files.



The following describes the filter legend options:

**Table 2-7 ▪** Filter Legend

| Icon | Label | Filters to files... |
|---|---|---|
| 📄 | **New Evidence** | ...containing any evidence that the previous scan did *not* detect but that the most recent scan *did*. |
| ✓ | **Reviewed** | ...marked as “reviewed”. |

**Table 2-7 ▪** Filter Legend (cont.)

| Icon | Label | Filters to files... |
|---|---|---|
| 🟥 | **Exact** | ...that are exact matches to known third-party files. |
| 🟦 | **Copyrights** | ...containing copyright information. |
| 🟪 | **Email/URLS** | ...containing email addresses or URLs. |
| 🟩 | **Licenses** | ...containing license information. |
| 🟦 | **Search Terms** | ...containing search terms defined in the scan profile. |
| 🟨 | **Source** | ...containing code-snippet matches (fingerprints) of known third-party code. |

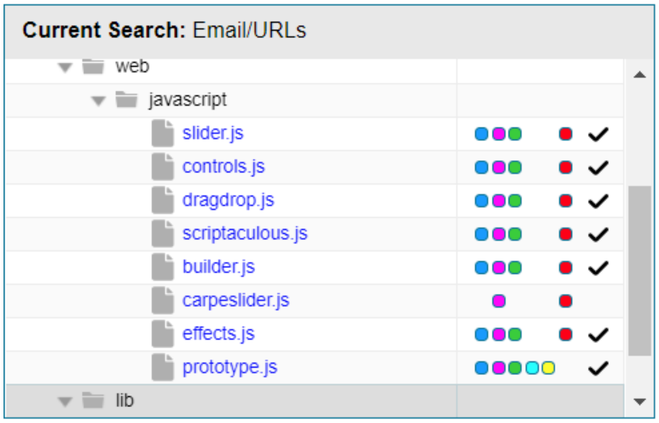
The color theme used for evidence types in this legend is also used to indicate the types of evidence found in a given file in the **Codebase Files** and **File Search Results** lists (see the following procedure) and on the **File Details** tab (see Examining and Managing Open-Source Evidence for a Given File).

***Note ▪*** *For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported. Therefore, the* ***Licenses*** *filter is the only applicable filter for locating such files.*

***Task*** ***To filter the codebase using the filter legend options, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Click the option in the filter legend to identify how you want to filter the codebase files. Results are listed in the **File Search Results** pane.
3. Navigate to the **File Search Results** pane, which now shows a codebase tree containing the files that meet your criterion.
4. Drill down in the codebase tree to view the files.

   Note that each file entry is flagged not only with a icon that matches the filter-legend criterion you selected but also with icons representing all evidence or attributes associated with this file.



**Current Search:** Email/URLs

- web
  - javascript
    - slider.js ✓
    - controls.js ✓
    - dragdrop.js ✓
    - scriptaculous.js ✓
    - builder.js ✓
    - carpeslider.js
    - effects.js ✓
    - prototype.js ✓
- lib

5. Select a file a from the filtered codebase list.

   Refer to these later sections for different ways to analyze and act on third-party evidence discovered in the files:

   - Examining and Managing Open-Source Evidence for a Given File
   - Examining and Managing Open-Source Evidence for a Given File
   - Managing the Codebase Files

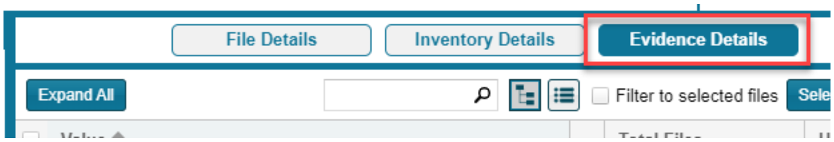
## Filtering the Codebase by a One or More Specific Instances of Evidence

You can filter the codebase to show only those files that contain a one or more *specific* instances of copyright, email, URL, license, or search-term evidence. To do so, use the **Evidence Details** tab in the **Analysis Workbench** to set up a search of these instances in the codebase. This tab lists the actual instances of the various types of evidence found in the codebase and shows the total number of files that contain each instance.

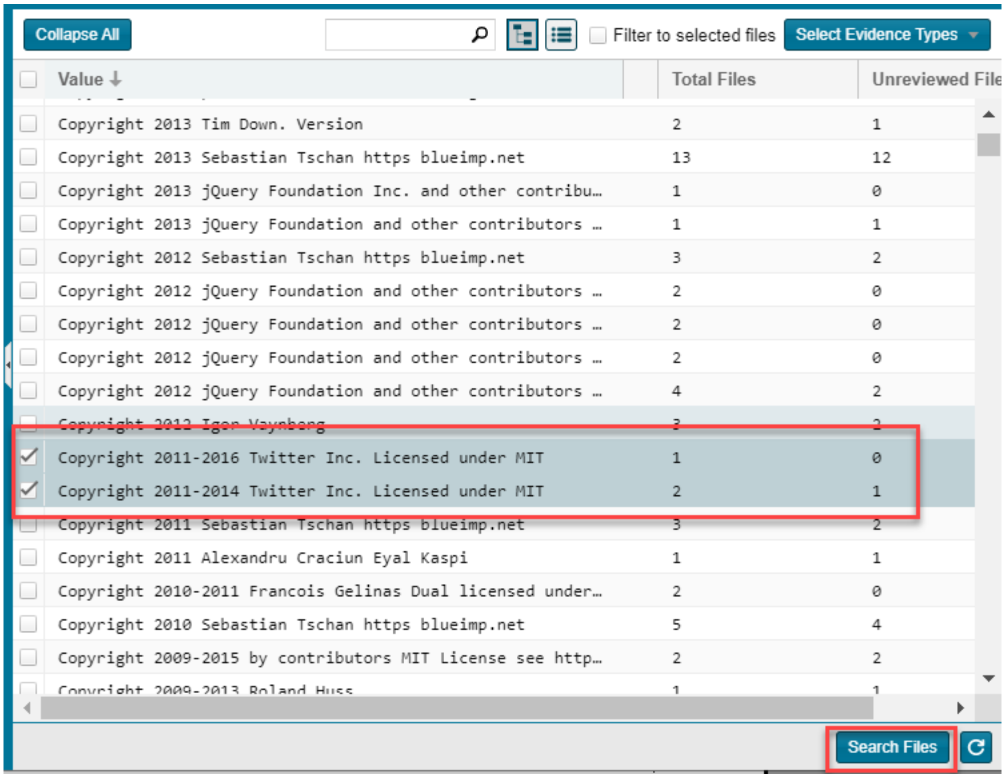
For example, suppose the **Evidence Details** tab indicates that a certain number of codebase files contain evidence of specific Twitter copyrights, and you want to know which codebase files contain this evidence. From the **Evidence Details** tab, you select the evidence instances—that is, the specific Twitter copyrights found in the codebase—by which to filter the codebase. When the search is complete, the files containing any of these copyrights are listed in the **File Search Results** pane.

***Task*** ***To search the codebase for the files containing specific evidence instances, do the following.***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Analysis Workbench**, click **Evidence Details** in the middle pane to open the tab.



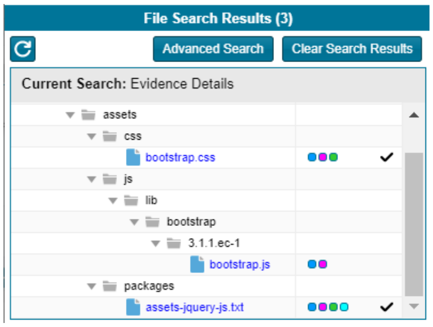
3. Select the checkbox to the left of one or more evidence instances in the list by which you want to search the codebase. (When you select multiple evidence instances, the search uses OR logic to obtain the results.) For example, you might want search for those files containing any of the two selected Twitter copyrights. Note that these copyrights are found in a total of three files.



| | Value ↓ | Total Files | Unreviewed File |
|---|---|---|---|
| □ | Copyright 2013 Tim Down. Version | 2 | 1 |
| □ | Copyright 2013 Sebastian Tschan https blueimp.net | 13 | 12 |
| □ | Copyright 2013 jQuery Foundation Inc. and other contribu… | 1 | 0 |
| □ | Copyright 2013 jQuery Foundation and other contributors … | 1 | 1 |
| □ | Copyright 2012 Sebastian Tschan https blueimp.net | 3 | 2 |
| □ | Copyright 2012 jQuery Foundation and other contributors … | 2 | 0 |
| □ | Copyright 2012 jQuery Foundation and other contributors … | 2 | 0 |
| □ | Copyright 2012 jQuery Foundation and other contributors … | 2 | 0 |
| □ | Copyright 2012 jQuery Foundation and other contributors … | 4 | 2 |
| □ | Copyright 2012 Igor Vaynberg | 3 | 2 |
| ☑ | Copyright 2011-2016 Twitter Inc. Licensed under MIT | 1 | 0 |
| ☑ | Copyright 2011-2014 Twitter Inc. Licensed under MIT | 2 | 1 |
| □ | Copyright 2011 Sebastian Tschan https blueimp.net | 3 | 2 |
| □ | Copyright 2011 Alexandru Craciun Eyal Kaspi | 1 | 1 |
| □ | Copyright 2010-2011 Francois Gelinas Dual licensed under… | 2 | 0 |
| □ | Copyright 2010 Sebastian Tschan https blueimp.net | 5 | 4 |
| □ | Copyright 2009-2015 by contributors MIT License see http… | 2 | 2 |
| □ | Copyright 2009-2013 Roland Huss | 1 | 1 |

4. Click **Search Files** in the lower right of the tab.

   The files containing one or more of the selected evidence instances are listed in the **Files Search Results** pane. In the following example, the three files containing the Twitter copyrights are listed.



# Examining and Managing Open-Source Evidence for a Given File

The **File Details** tab provides metadata about a selected codebase file and detailed information about open-source and other third-party evidence detected in the file:

- File Metadata
- Examining a Codebase File Exactly Matching an Open-Source File
- Examining a Codebase File Containing Partial Matches to Open-Source File Content

## File Metadata

The **File Details** tab includes a expandable header that lists metadata about the selected codebase file, as well as the three tabs—**Evidence**, **Exact Matches**, and **Partial Matches**—available to examine the file's open-source or third-party evidence. (These tabs are described in the procedural sections that follow.)

By default, the header is collapsed.

**Table 2-8** ▪ Codebase File Metadata

| File Property | Description |
|---|---|
| <alias>:<relativeFile Path> name | (Displays only when the header is collapsed) The file identified in an <alias>:<relativeFilePath> format, where <alias> and <relativeFilePath> are defined below for **Alias** and **Path**.<br>As an example, if you selected the codebase file **agpl-3.0.txt**, located directly under the scan folder **ePortal-1.3**, which in turn is directly under the scan-root path for the scanner whose alias is "EP_remote", the file name shown here would be **EP_remote:ePortal-1.3/agpl-3.0.txt**. |
| **Name** | The name of the selected file. |
| **Path** | The file's path relative to the scan-root path on instance hosting the scanner (Scan Server or remote scan agent). |
| **Alias** | The unique name defined during scanner setup to represent the scanner's scan-root path containing the codebase where the file is located. The alias provides a more descriptive name for the scan-root path. |

**Table 2-8 ▪** Codebase File Metadata (cont.)

| File Property | Description |
|---|---|
| **Digest** | The MD5 value for the file.<br>***Note ▪*** *If SHA-1 support is enabled for your Code Insight instance, you can view the file's SHA-1 digest by using the Code Insight* ***Get details of a file by ID*** *or* ***Fetch all scanned files for a project*** *REST API. For more information about these APIs, refer to the Code Insight Swagger documentation, available from the* ***Help > REST API Guide*** *option* on the ≡ menu*. The SHA-1 digest does not display along with the MD5 digest in this metadata section. (To determine whether SHA-1 support is enabled, contact your Code Insight database or system administrator.)* |
| **Modified** | The value of the file's "Date Modified" property at the time of the most recent scan. This value shows the date when the file was last modified in the file system. |
| **Type** | The file format type of the scanned file, such FILE or ARCHIVE_BINARY. |
| **File Size** | The size of the file. |
| **Lines of Code** | The number of lines of code in the file. |
| **Reviewed** | The Yes or No indicator showing whether the file has been reviewed. |

## Examining a Codebase File Exactly Matching an Open-Source File

If your project is configured for exact-match scanning, the scan will identify files in the codebase whose content exactly matches files in the Compliance Library (CL). Follow these steps to examine a scanned codebase file whose content exactly matches one or more open-source or other third-party files (called *remote files*) in the CL. The **Exact Matches** tab for a given codebase file shows the matching remote files, along with the open-source or third-party component versions and licenses associated with each.

***Note ▪*** *By default, Code Insight does not perform source-code matching on files that are exact matches to CL files. However, you can enable your project scan to force source-code matching on files that are also exact matches. See Updating Scan Settings for a Project. For information about the results of source-code matching, see Examining Evidence of Open-Source Code in a Given Non-Binary File. Currently, exact matching is not available for files that are scanned by a scan agent plugin.*

*Task* ***To examine a codebase file that exactly matches one or more remote files, do the following:***

1. Ensure that you have run a scan with the **Comprehensive Scan Profile** selected for the desired project (or a custom scan profile with the Exact Matches feature enabled). For more information, see Updating Scan Settings for a Project.
2. Open the **Analysis Workbench** for the project. (For instructions, see Opening the Analysis Workbench.)
3. Click the **Exact** link in the legend at the top right of the page to find all files with exact matches (see Using the Filter Legend Options to Filter the Codebase). Results are listed in the **File Search Results** pane.
4. Select a codebase file from the **File Search Results** list, and select the **Exact Matches** tab.

   Three Remote Files panels are displayed:

   - The information in the **Remote Files** panel on the left consists of a set of files from the open-source community that are an exact match to the scanned file. This means that the scanned file in the codebase likely originated from outside the organization, and thus its origin needs to be identified.
   - The **Components** panel lists the open-source or third-party components associated with each remote file.
   - The **Licenses** panel lists the licenses normally associated with each component.

   See the More About the “Remote Files” Panels on the Exact or Partial Matches Tabs for more information about the functionality available from the three panels.
5. Select a remote file in the **Remote Files** panel to see the associated component and license information (on the **Components** and **Licenses** panels, respectively).
6. (Optional) Associate the codebase file to an inventory item based on the open-source or third-party component associated with a matching remote file. See Adding a Codebase File to Inventory Associated with a Remote File’s Open-Source Component for details.

# Examining a Codebase File Containing Partial Matches to Open-Source File Content

The following sections describe how to examine a given codebase file that contains code snippets or textual-string evidence that partially matches open-source or other third-party file content that is stored in the Code Insight data library or in the Compliance Library (CL):

- Viewing a Summary of Open-Source Evidence in a Given File
- Viewing Details for Licenses Associated with Codebase Files
- Examining Open-Source Evidence in a Given Binary File
- Examining Evidence of Open-Source Copyrights, Email Addresses, URLs, Licenses, and Search Terms in a Given Non-Binary File
- Examining Evidence of Open-Source Code in a Given Non-Binary File
- More About the “Remote Files” Panels on the Exact or Partial Matches Tabs
- Adding a Codebase File to Inventory Associated with a Remote File’s Open-Source Component

***Note** ▪ Currently, for files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is reported in Code Insight. The **Evidence Summary** tab (described in this section) for such a file will list any license evidence discovered in the file as part of the remote scan. However, both the **Exact Matches** and **Partial Matches** tabs (also described in this section) are disabled for the file, as exact and partial match information is available for only scans performed by a Scan Server.*

## Viewing a Summary of Open-Source Evidence in a Given File

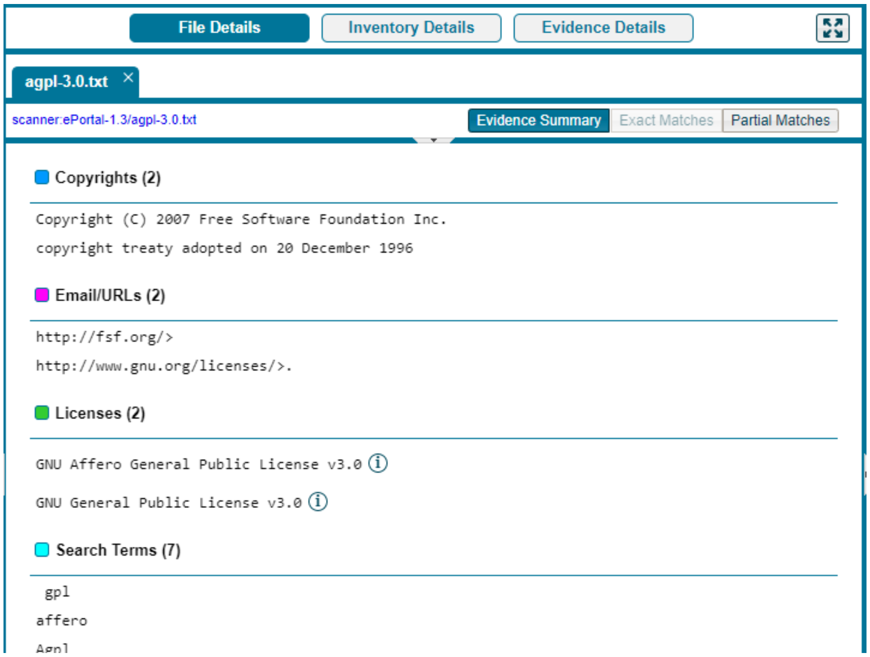
The **Evidence Summary** tab on the **File Details** tab provides a *summary* of the open-source and third-party evidence identified for a given scanned file (binary or non-binary). You can use this information to write review comments in new or existing inventory items associated with this file. The **Evidence Summary** tab lists the string-based scan results (and result totals) for each of the following evidence types:

- Copyrights
- Emails/URLs
- Licenses
- Search Terms

This listing is especially useful for examining a concise view of the open-source and third-party evidence in a binary file (such as an object file, image, executable, and so forth).

***Task*** ***To view the evidence summary, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Select a file in the **Codebase Files** panel.
3. Select the **File Details** tab.
4. Select the **Evidence Summary** tab. Summary information about the selected file appears in the center pane:
5. (Optional) To view additional information for the selected file, click the expand arrow (▾). The top portion of the tab expands to show details about the file.



🟦 Copyrights (2)

```
Copyright (C) 2007 Free Software Foundation Inc.
copyright treaty adopted on 20 December 1996
```

🟪 Email/URLs (2)

```
http://fsf.org/>
http://www.gnu.org/licenses/>.
```

🟩 Licenses (2)

```
GNU Affero General Public License v3.0 ⓘ
GNU General Public License v3.0 ⓘ
```

🟦 Search Terms (7)

```
 gpl
affero
Agpl
```

## Viewing Details for Licenses Associated with Codebase Files

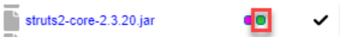
In the **Analysis Workbench**, you can view details about the licenses discovered in codebase files. When you click the ⓘ icon next to a license reference on the **Evidence Summary** tab on the **File Details** tab, detailed information for a given license is displayed on the following tabs in the **License Details** window:

- A **General Information** tab that lists details such as the name of the license, its family, and the license priority assigned by Code Insight.
- A **License Text** tab that displays the complete license text (representing the external forge license text).

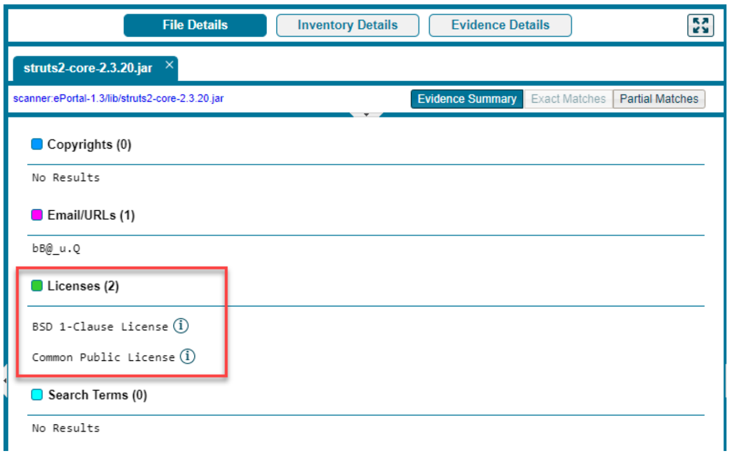
The license information shown in the **License Details** window is pulled from the Code Insight data library.

***Task*** ***To view details for a license, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. (Optional) To make file selection easier, you can filter the codebase files to only those containing license evidence. See Using the Filter Legend Options to Filter the Codebase.
3. In the **Codebase Files** pane or **File Search Results** pane, select the codebase file containing the license evidence you want to review. A file with license evidence will show a green icon in its entry:



4. Locate a license reference on the **File Details** tab, as in this example of the **Evidence Summary** subtab on **File Details** tab.



scanner:ePortal-1.3/lib/struts2-core-2.3.20.jar

🟦 Copyrights (0)

No Results

🟪 Email/URLs (1)

bB@_u.Q

🟩 Licenses (2)

BSD 1-Clause License ⓘ

Common Public License ⓘ

🟦 Search Terms (0)

No Results

***Note** ▪ License references are also displayed when create or edit an inventory item or perform a Component Lookup from the **Inventory Details** tab.*

5. Click the information icon (ⓘ) next to the license name. The **License Details** window appears with the **General Information** tab in focus.

   For descriptions of these fields on this tab, see License Details Window. Also see License Priority for background on how the license priority is used.

6. Select the **License Text** tab to view the license text.

7. When you have finished examining the license details, click **Close**.

## Examining Open-Source Evidence in a Given Binary File

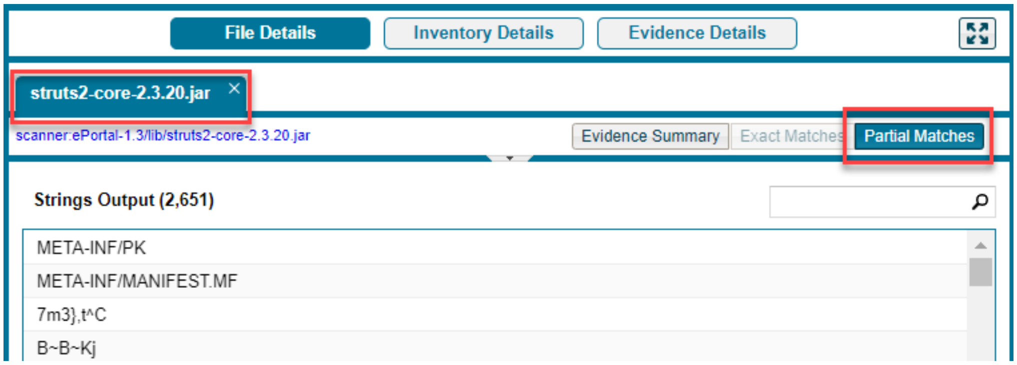
In the content of a given binary file (such as an object file, image, executable, and so forth) in the codebase, Code Insight can locate textual strings that might be evidence of open-source or other third-party code. Each string, consisting of at least three consecutive printable characters, might part of a comment, copyright, URL, email address, and another evidence type. Code Insight simply lists these strings on the **Partial Matches** tab; it does not show them highlighted within the context of the actual binary code (unlike non-binary files, in which Code Insight highlights evidence within the actual file content).

***Task** To examine open-source evidence present in a binary file, do the following:*

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)

2. Select a binary file in the **Codebase Files** panel, and click **File Details**.

3. Click **Partial Matches**. The **File Details** panel displays the strings that are output.

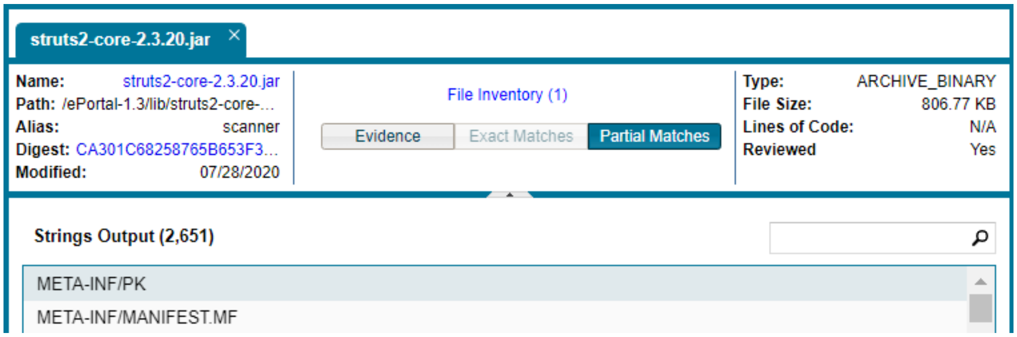


scanner:ePortal-1.3/lib/struts2-core-2.3.20.jar

Strings Output (2,651)

```
META-INF/PK
META-INF/MANIFEST.MF
7m3},t^C
B~B~Kj
```

4. (Optional) Click the expand arrow (▾), to view additional options. The top portion of the tab expands to show details about the binary file.



Name: struts2-core-2.3.20.jar
Path: /ePortal-1.3/lib/struts2-core-...
Alias: scanner
Digest: CA301C68258765B653F3...
Modified: 07/28/2020

Type: ARCHIVE_BINARY
File Size: 806.77 KB
Lines of Code: N/A
Reviewed Yes

Strings Output (2,651)

```
META-INF/PK
META-INF/MANIFEST.MF
```

## Examining Evidence of Open-Source Copyrights, Email Addresses, URLs, Licenses, and Search Terms in a Given Non-Binary File

You can view open-source or third-party evidence of copyrights, URLs, licenses, email addresses, or search terms (or a combination of these) as highlighted within the content of a given non-binary file.

***Task*** ***To view copyright, email, URL, and license content in a file, do the following:***

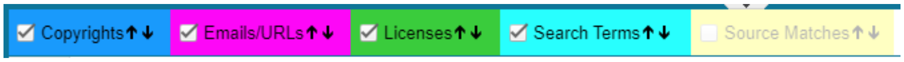
1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)

2. In the **Codebase Files** pane or **File Search Results** pane, select the codebase file containing the evidence you want to review. (Optionally, to make file selection easier, you can filter the codebase files to only those containing a specific type of evidence. See Using the Filter Legend Options to Filter the Codebase.)

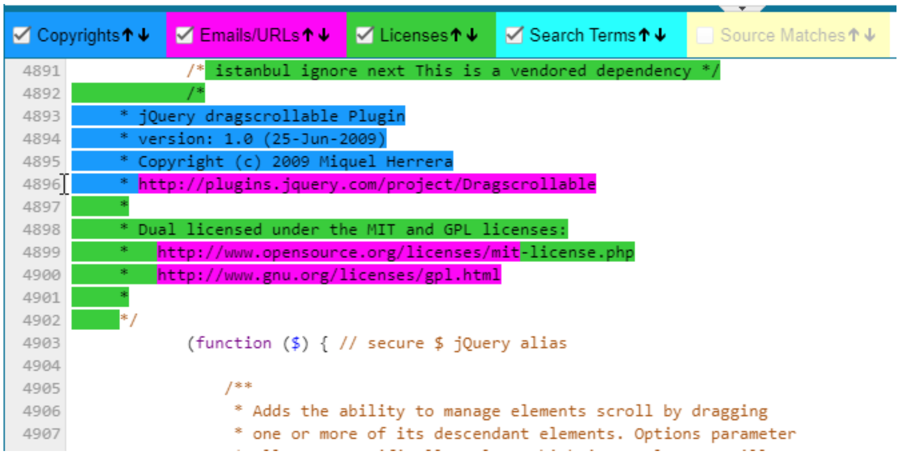
3. Click **File Details**.

4. Select the **Partial Matches** tab to show the contents of the file.

   Color-coded selection boxes at the top of the **Partial Matches** tab are used to indicate the type of evidence you want to highlight in the file. (Based on your screen size, labels on these selection boxes might not be visible. In this case, hover over a box to see its label.) Depending on the types of evidence existing in the file, certain selection boxes might already be selected; others might be disabled.



5. If necessary, select (or unselect) one or more selection boxes to highlight the evidence you want to view in the file. For example, the following selections will highlight instances of copyright, email, URL, license, and search-term evidence in the file:



```
          /* istanbul ignore next This is a vendored dependency */
          /*
       * jQuery dragscrollable Plugin
       * version: 1.0 (25-Jun-2009)
       * Copyright (c) 2009 Miquel Herrera
       * http://plugins.jquery.com/project/Dragscrollable
       *
       * Dual licensed under the MIT and GPL licenses:
       *   http://www.opensource.org/licenses/mit-license.php
       *   http://www.gnu.org/licenses/gpl.html
       *
      */
          (function ($) { // secure $ jQuery alias

              /**
               * Adds the ability to manage elements scroll by dragging
               * one or more of its descendant elements. Options parameter
```

## Considerations for Viewing License Evidence

When a given source or text file in the **Codebase Files** list contains license evidence (as indicated by a green icon 🟩 in the file entry and by the one or more licenses listed on **Evidence Summary** tab), the **Partial Matches** tab usually shows the specific evidence for each license highlighted in green within the file content. However, the following exceptions can occur:

- **Licenses are detected but not highlighted in the file**—Cases can occur during a scan when a license is discovered in the scan results and listed on the **Evidence Summary** tab, but no associated license text is highlighted on the **Partial Matches** tab. The lack of highlighting occurs because the scanner is unable to calculate the offsets for license text it cannot explicitly identify in the file.
- **All content is highlighted, including large sections of non-license-related text**—If a file containing license evidence uses a non-supported file extension, all content in the file is highlighted in green, including large sections of non-license-related information. (This is different from the scenario in which all or nearly all the content in a file *is* the license text, and thus the entire file is highlighted as such.)

# Examining Evidence of Open-Source Code in a Given Non-Binary File

If your project scan is configured to perform source-code matches, the scan will identify source-code snippets (also called *fingerprints*) in your non-binary code that match open-source and other third-party code stored in the Compliance Library (CL). The **Partial Matches** tab for a given codebase file shows the snippet matches as highlighted within the actual file content. This tab also includes a list of the CL files (called *remote files*) associated with the discovered snippets. When you select one of these remote files, the source-code highlights are refreshed to highlight only those snippets associated with the remote file.

***Note** ▪ The size limit for a file that you open in the **Partial Matches** tab is 2 MB. If the file you want to inspect is too large, you can download and open it outside of Code Insight to inspect it manually for evidence.*

***Task*** ***To view source matches, do the following:***

1. Ensure that you have run a scan with **Comprehensive Scan Profile** selected in the desired project (or a custom scan profile with the Source Code Matches feature enabled). For more information, see Updating Scan Settings for a Project.
2. Open the **Analysis Workbench** for the project. (For instructions, see Opening the Analysis Workbench.)
3. Click the **Source** link in the legend at the top right of the page to filter to all files with source-code matches (see Using the Filter Legend Options to Filter the Codebase). Results are listed in the **File Search Results** pane.
4. Click a codebase file in the list in **File Search Results**, and select the **Partial Matches** tab.
5. On the **Partial Matches** tab, click the **Source Matches** selection box at the top of the tab to enable *source code fingerprint match* results.

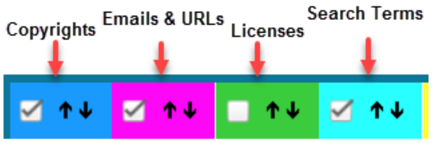
   Three Remote Files panels are displayed:

   - The information in the **Remote Files** panel on the left consists of a set of files identified in the open-source community (whose information is stored in the Code Insight Data Library) that contain code snippets identical to code snippets detected in the scanned file. This matching code can indicate that the scanned file in the codebase contains content that originated from outside the organization, and its origin needs to be identified.
   - The **Components** panel lists the open source or third-party components associated with the remote file.
   - The **Licenses** panel lists the licenses normally associated with the component.

   See the More About the "Remote Files" Panels on the Exact or Partial Matches Tabs for details about the functionality available from the three panels.
6. Select a remote file in the **Remote Files** panel on the left to highlight the source-code snippet matches in the scanned file that match those in the remote file and to view the lists of associated component and license information (on the **Components** and **Licenses** panels, respectively).

   Note that the **Remote Files** panel will additionally contain the following CodeRank™ values:

   - **CodeRank (CR%)**: A composite heuristic comprised of Coverage, Clustering, and Uniqueness values. The higher the number, the stronger the match confidence.
   - **Coverage (CV%)**: The percentage of remote-file content contained in your scanned file.
   - **Clustering (CL%)**: The density or proximity of remote-file matches within your scanned file.
   - **Uniqueness (U%)**: An indication of how often the remote-file matches detected in the scanned file occur in the Compliance Library (CL).
   - **Matches**: The number of unique matches in the scanned file.
7. To view the instances of other types of evidence (for example, copyrights, licenses, URLs, email addresses, and search terms) in the codebase file, click the appropriate color-coded selection boxes at the top of the **Partial Matches** tab.



Each instance of evidence is highlighted in the same color as its corresponding selection box.

## More About the “Remote Files” Panels on the Exact or Partial Matches Tabs

When you open the **Exact Matches** tab or the **Partial Matches** tab (and select the **Partial Matches** checkbox) for a codebase file selected in the **Analysis Workbench**, a **File Details** view is shown in the center of the screen with the following panels:

- Remote Files Panel
- Components Panel
- Licenses Panel

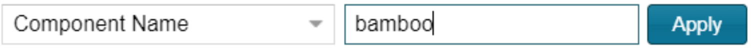
### Note About Filtering in the Panels

The items in each panel can be filtered in these ways:

- When you a select a specific item in one panel, the items in the other panels area filtered to show only those items associated with the selected item.

  For example, when you select a specific remote file (that is, a file found in the Compliance Library that matches a codebase file) in the **Remote Files** panel, the **Components** list is filtered to show only items associated with the remote file, and the **Licenses** list is filtered to show only items associated with the items now listed in the **Components** panel. Similarly, if you select a specific component in the **Components** list, the **Remote Files** and **Licenses** lists are filtered to show only those items associated with the selected component.

- You can filter the items in a given panel by entering a search sting to show only items in that panel containing the string. When the filter is applied, the other panels are automatically filtered to show only items associated with the items now listed in the panel filtered by the search string.


Component Name ▾ | bamboo | Apply

### Remote Files Panel

This panel initially lists all the remote files from the Compliance Library (CL) that are either a perfect match (exact match) or contains partial-match content (source-code fingerprint match) to the scanned file. The partial-match content also ranks the remote files by CodeRank™ values, described in the previous section, Examining Evidence of Open-Source Code in a Given Non-Binary File.

The remote files list can be filtered as discussed in Note About Filtering in the Panels.

### Components Panel

This panel initially lists all the component versions that contain the remote files listed in the **Remote Files** panel. The list can be filtered as discussed in Note About Filtering in the Panels.

You can perform the following operations for a given component in the **Components** panel:

- To review the path of a remote file within a component, select the file in the **Remote Files** panel, and then click the **Remote File Paths** icon in the component row. A remote file is a file found within an open source component release that is either identical to the scanned file, or contains similar partial content as the scanned file. The remote file path is important because similar file structures between the scanned codebase and the remote file content is a potential strong indicator of code reuse from an open source project.
- To view information about the component, click the **Information** icon ⓘ.
- To add the selected codebase file to an inventory item associated with the component, click the **Add File to Inventory** icon. For more information, see Adding a Codebase File to Inventory Associated with a Remote File's Open-Source Component.

### Licenses Panel

This panel lists all the licenses associated with the component versions listed in the **Components** panel but can be filtered as discussed in Note About Filtering in the Panels.

You can view information about the license by clicking the **Information** icon ⓘ in the license entry.

## Adding a Codebase File to Inventory Associated with a Remote File's Open-Source Component

When a given codebase file exactly or partially matches a remote file (that is, a file in the Compliance Library), you can use the following procedure to easily add the codebase file to an inventory item based on an open-source or third-party component associated with the remote file.

***Task*** ***To add codebase file to an inventory item based on the remote file's open-source or third-party component, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Click the **Exact** or **Source** matches link in the legend at the top right of the workbench to search for codebase files that are exact or partial matches to files in the Compliance Library. Results are listed in the **File Search Results** pane.
3. From the list in **File Search Results**, locate and click the codebase file you want to add to an inventory item based on a specific component version associated with the file.
4. Open the **File Details** tab, and, at the top of the tab, select the **Exact Matches** or **Partial Matches** tab.

   Additionally, if you are on the **Partial Matches** tab, select the **Source Matches** checkbox.
5. From the **Remote Files** panel, select the remote file associated with the component on which the inventory item to which you want to add the file is based (or will be based if you need to create an inventory item).
6. In the **Components** panel, locate the component version that you believe is the origin of the matching code in the scanned codebase file, and click the **Add File to Inventory** icon in that component row.

Code Insight searches for existing inventory items associated with the given component version. If one or more inventory items exist, the **Add to Inventory** dialog is displayed, showing the list of available inventory items. Continue with Step 7.

Otherwise, if no inventory items are currently associated with the given component version, the **Lookup Component** window is displayed, showing the given component version. From this window, you can register an instance for the component version (by selecting a license), register a new component version, search for a new component altogether, or create a custom component. Once you click **Use This Instance** for a component version in the **Lookup Component** window, Code Insight creates the inventory item based on the selected component instance and associates the selected file with the inventory item. The **Inventory Details** tab is opened for the inventory item. (Ignore the remaining steps in this procedure.)

7. Click the checkbox next to the inventory item to which you want to add the file.
8. (Optional) To mark the selected codebase file as reviewed, click **Mark file as reviewed**.
9. Click **Submit**. Code Insight adds the codebase file to the inventory item.

# Viewing a Summary of Evidence Detected Across the Codebase

The **Evidence Details** pane in the **Analysis Workbench** enables you to view the list of open-source and third-party textual evidence detected across the codebase during the scan. The list shows instances of evidence for the following entities and, for each instance, includes the total number of files in which the instance was found and the number of those files that are not marked as reviewed:

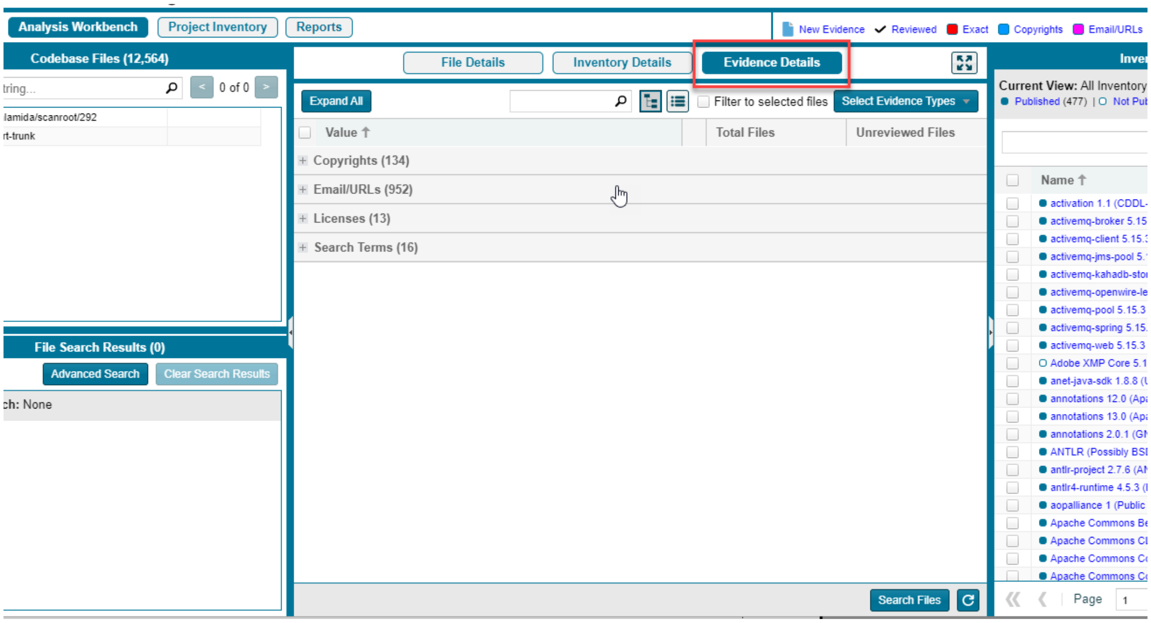
- **Copyrights**—The copyright text of potential third-party software code found in your codebase.
- **Email/URLs**—Email addresses and website URLs of potential owners of third-party software found in your codebase.
- **Licenses**—Third-party licenses in your codebase that should be reviewed for IP compliance.
- **Search Terms**—Terms related to open-source or third-party software in your codebase (based on the terms defined in the Scan Profile).

***Note** ▪ Currently, for files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is reported in Code Insight. The **Evidence Details** pane will list any license evidence found in such files as part of the remote scan.*

***Task*** ***To view all open-source or third-party copyrights, email addresses, URLs, licenses, and search terms in the codebase, do the following:***

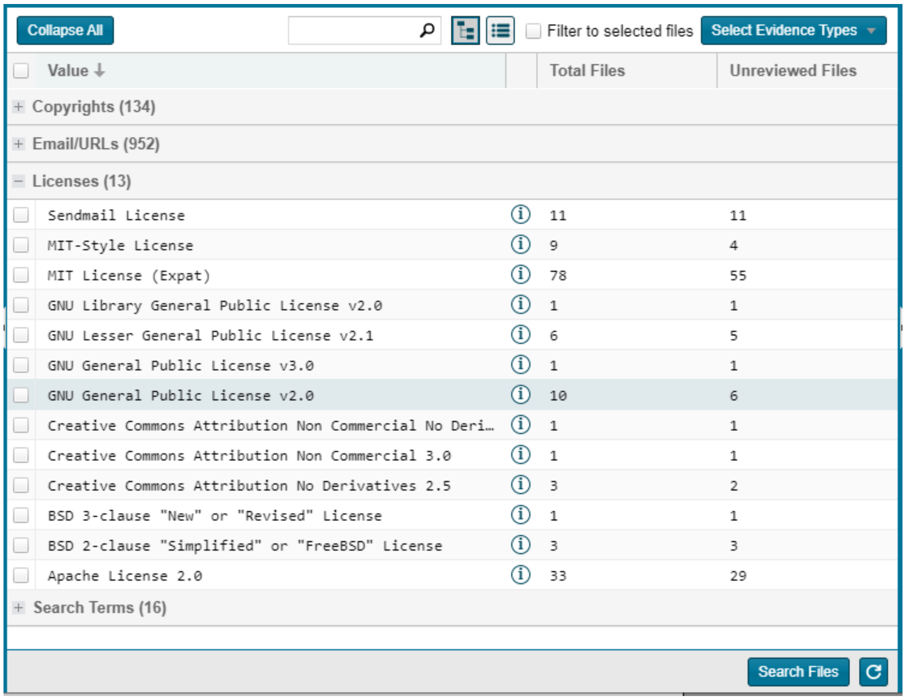
1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Open the **Evidence Details** tab in the center pane.

   By default, the evidence is displayed in a tree-view, where you can expand or collapse the evidence instances (entries) under each evidence category (**Copyrights**, **Email/URLs,** and so forth).



| Value | Total Files | Unreviewed Files |
|---|---|---|
| Copyrights (134) | | |
| Email/URLs (952) | | |
| Licenses (13) | | |
| Search Terms (16) | | |

3. View the list of evidence instances under specific categories as needed. (Alternatively, to see all evidence instances, click **Expand All**; click **Collapse All** to return the display to only collapsed category headings.)



| Value | Total Files | Unreviewed Files |
|---|---|---|
| Copyrights (134) | | |
| Email/URLs (952) | | |
| Licenses (13) | | |
| Sendmail License | 11 | 11 |
| MIT-Style License | 9 | 4 |
| MIT License (Expat) | 78 | 55 |
| GNU Library General Public License v2.0 | 1 | 1 |
| GNU Lesser General Public License v2.1 | 6 | 5 |
| GNU General Public License v3.0 | 1 | 1 |
| GNU General Public License v2.0 | 10 | 6 |
| Creative Commons Attribution Non Commercial No Deri… | 1 | 1 |
| Creative Commons Attribution Non Commercial 3.0 | 1 | 1 |
| Creative Commons Attribution No Derivatives 2.5 | 3 | 2 |
| BSD 3-clause "New" or "Revised" License | 1 | 1 |
| BSD 2-clause "Simplified" or "FreeBSD" License | 3 | 3 |
| Apache License 2.0 | 33 | 29 |
| Search Terms (16) | | |

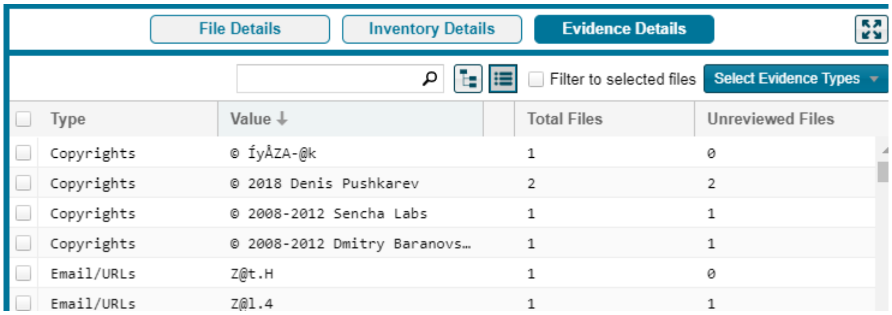
In each category list, you can view the total number of files containing evidence at the category level, the total number of files containing evidence for each instance, and the number of those files that are not marked as reviewed. For each instance of license evidence, you can click the ⓘ icon to view information about the license.

4. (Optional) Configure the **Evidence Details** tab to show alternative views of the available evidence. See the next section, Configuring Various Views of Evidence Details.

## Configuring Various Views of Evidence Details

The **Evidence Details** tab provides options to configure various views of the available evidence. You can use a combination of these configurations to obtain the view you want.

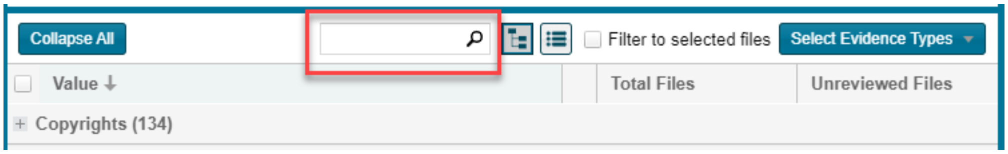
- **Display the evidence as a simple list instead of in a tree-view**—Click ☰ at the top of the **Evidence Details** tab. The list is reformatted without expandable and collapsible categories. A **Type** column is added to show the category for each evidence instance.



| □ | Type | Value ↓ | Total Files | Unreviewed Files |
|---|---|---|---|---|
| □ | Copyrights | © ÍyÅZA-@k | 1 | 0 |
| □ | Copyrights | © 2018 Denis Pushkarev | 2 | 2 |
| □ | Copyrights | © 2008-2012 Sencha Labs | 1 | 1 |
| □ | Copyrights | © 2008-2012 Dmitry Baranovs… | 1 | 1 |
| □ | Email/URLs | Z@t.H | 1 | 0 |
| □ | Email/URLs | Z@1.4 | 1 | 1 |

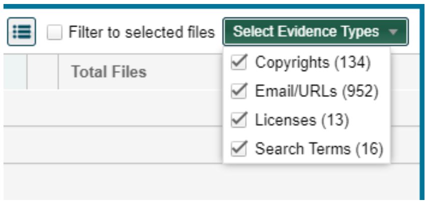
  To return to the tree-view list, click ⊞ at the top of the **Evidence Details** tab.

- **Search for evidence that contains a specific string**—Enter the string in the search box at the top of the **Evidence Details** pane, and then click the **Refresh** ⟳ button in the lower right of the pane.



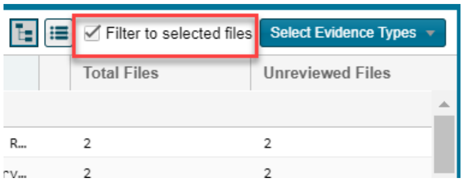
  The list is refreshed to show only evidence containing the string.

- **Show only specific categories of evidence**—Click the **Select Evidence Types** dropdown at the top of the **Evidence Details** tab, and select the categories you want to display.



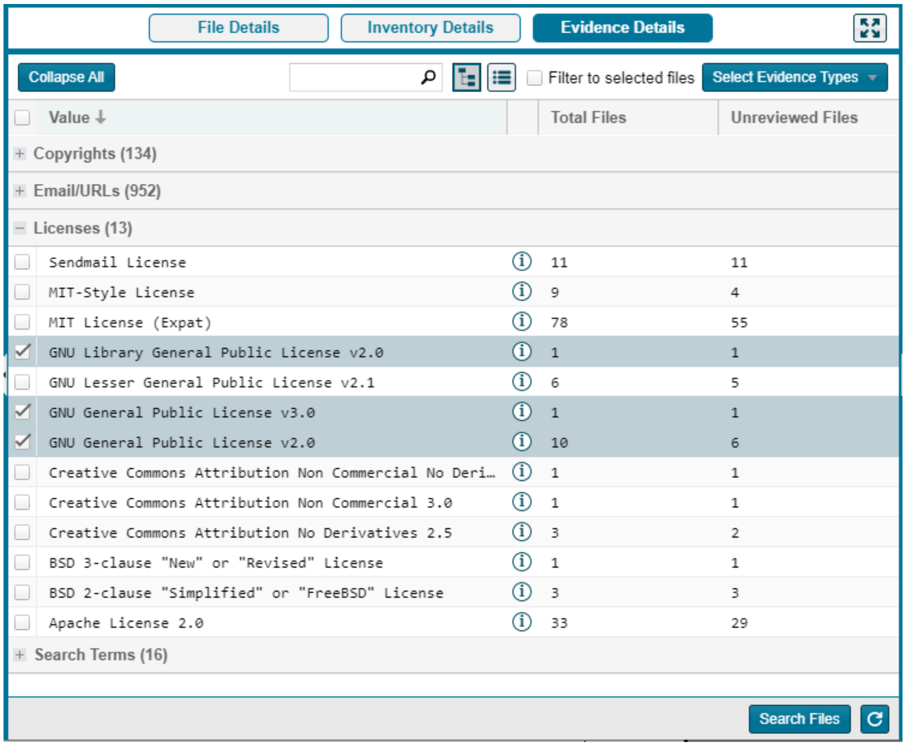
☑ Copyrights (134)
☑ Email/URLs (952)
☑ Licenses (13)
☑ Search Terms (16)

- **List the evidence contained in selected codebase files only**—Select one or more files in the **Codebase Files** pane, and click **Filter to Selected Files** on the **Evidence Details** tab. The evidence instances on the **Evidence Details** tab filters to only those instances found in the selected files.



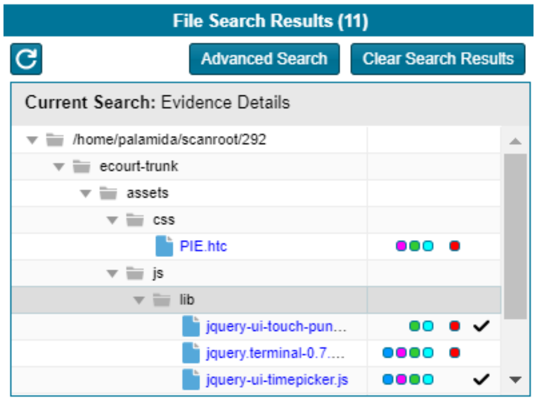
| | Total Files | Unreviewed Files |
|---|---|---|
| R… | 2 | 2 |
| ry… | 2 | 2 |

- **Filter to a list of codebase files in the File Search Results pane that contain only selected evidence**—Select the checkbox to the left of one or more evidence instances in the list on the **Evidence Details** tab, and click **Search Files** in the lower right of the tab. (When you select multiple evidence instances, the search uses OR logic to obtain the results.)



| ☐ Value ↓ | Total Files | Unreviewed Files |
|---|---|---|
| + Copyrights (134) | | |
| + Email/URLs (952) | | |
| − Licenses (13) | | |
| ☐ Sendmail License | 11 | 11 |
| ☐ MIT-Style License | 9 | 4 |
| ☐ MIT License (Expat) | 78 | 55 |
| ☑ GNU Library General Public License v2.0 | 1 | 1 |
| ☐ GNU Lesser General Public License v2.1 | 6 | 5 |
| ☑ GNU General Public License v3.0 | 1 | 1 |
| ☑ GNU General Public License v2.0 | 10 | 6 |
| ☐ Creative Commons Attribution Non Commercial No Deri… | 1 | 1 |
| ☐ Creative Commons Attribution Non Commercial 3.0 | 1 | 1 |
| ☐ Creative Commons Attribution No Derivatives 2.5 | 3 | 2 |
| ☐ BSD 3-clause "New" or "Revised" License | 1 | 1 |
| ☐ BSD 2-clause "Simplified" or "FreeBSD" License | 3 | 3 |
| ☐ Apache License 2.0 | 33 | 29 |
| + Search Terms (16) | | |

A list of only those files that contain the selected evidence appears in a tree view in the **File Search Results** pane.



**Current Search:** Evidence Details

```
▼ /home/palamida/scanroot/292
  ▼ ecourt-trunk
    ▼ assets
      ▼ css
          PIE.htc
      ▼ js
        ▼ lib
            jquery-ui-touch-pun...
            jquery.terminal-0.7....
            jquery-ui-timepicker.js
```

# Managing the Codebase Files

The following topics describe basic operations you can perform on one or more selected codebase files in the **Codebase Files** and **File Search Results** panes:

- Showing Inventory Associated to Files Selected in the Codebase List

- Adding Files to Inventory From the Codebase List
- Listing Copyright, Email, URL, License, and Search-Term Evidence for Files Selected in the Codebase List
- Marking Codebase Files as Reviewed
- Reverting Codebase Files to Unreviewed Status
- Downloading a Codebase File
- Copying Codebase File and Folder Paths

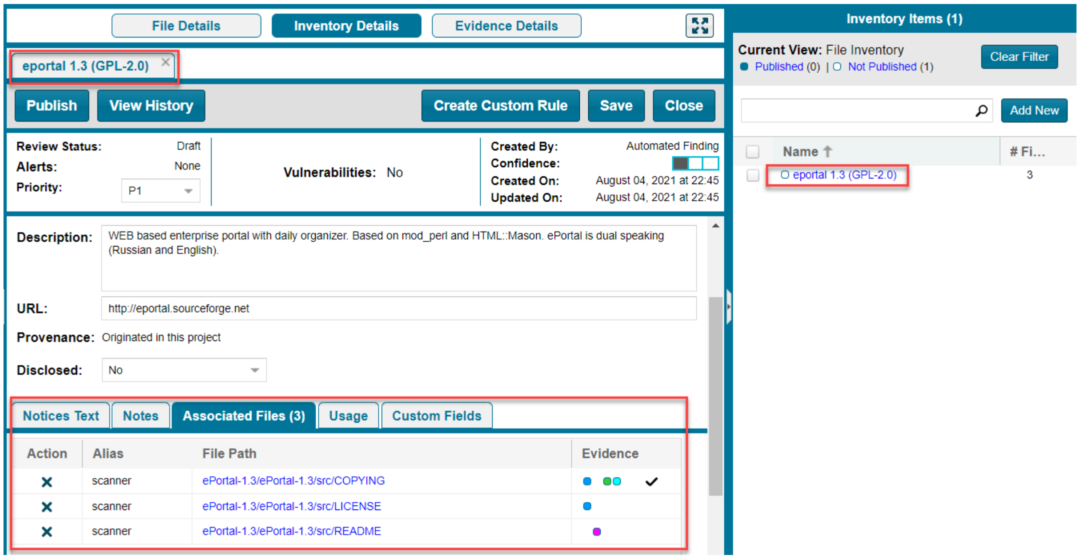
As an alternative to performing these operations on files listed in the **Codebase Files** or **File Search Results** pane, you can also perform them on the files listed on the **Associated Files** tab for a selected inventory item in the **Inventory Items** pane in the **Analysis Workbench**. See Viewing Associated Files.

## Showing Inventory Associated to Files Selected in the Codebase List

You can filter inventory items in the **Inventory Items** pane in the **Analysis Workbench** to only those items associated with the codebase files you have selected in the **Codebase Files** or **File Search Results** pane. This procedure is helpful in quickly locating the inventory items to which a codebase file is associated.

***Task*** ***To show only inventory items associated with files selected in the codebase, do the following:***

1. Open the **Analysis Workbench** for a project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, select and right-click an individual file or a set of files to view the inventory items with which the files are associated. (You can also right-click a directory to select all files in that directory and its subdirectories.)
3. From the pop-up menu, select **Show file inventory**. The **Inventory Items** pane on the right side of the **Analysis Workbench** filters to all inventory items to which the selected files are associated.
4. (Optional) To view all files associated with a displayed inventory item, do the following:
    a. Select the inventory item in the **Inventory Items** pane. The **Inventory Details** tab for that item opens in the middle pane of the **Analysis Workbench**.
    b. Within the **Inventory Details** tab for the inventory item, click the **Associated Files** tab to see all the files associated with the inventory item. (For more information about the file list and the options available on the **Associated Files** tab, see Viewing Associated Files for Inventory in the Analysis Workbench.)



**Current View:** File Inventory
● Published (0) | ○ Not Published (1)

| | Name | # Fi... |
|---|---|---|
| □ | ○ eportal 1.3 (GPL-2.0) | 3 |

**Review Status:** Draft
**Alerts:** None
**Priority:** P1
**Vulnerabilities:** No
**Created By:** Automated Finding
**Confidence:**
**Created On:** August 04, 2021 at 22:45
**Updated On:** August 04, 2021 at 22:45

**Description:** WEB based enterprise portal with daily organizer. Based on mod_perl and HTML::Mason. ePortal is dual speaking (Russian and English).

**URL:** http://eportal.sourceforge.net

**Provenance:** Originated in this project

**Disclosed:** No

| Action | Alias | File Path | Evidence |
|---|---|---|---|
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/COPYING | ● ●● ✓ |
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/LICENSE | ● |
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/README | ● |

# Adding Files to Inventory From the Codebase List

This section describes how to “inventory” selected codebase files from the **Codebase Files** or **File Search Results** list in the **Analysis Workbench**, either by associating these files with existing inventory or by creating a new inventory item with which to associate them.

***Task*** ***To create or update an inventory item with files from the Codebase Files list, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)

2. (Only when adding files to a single existing inventory item) Navigate to the **Inventory Items** pane, and select the inventory item to open its **Inventory Details** tab.

3. On the left side of the **Analysis Workbench**, navigate to the **Codebase Files** pane or the **File Search Results** pane.

4. Select one or more codebase files that you want to add to inventory (or select one or more folders whose files you to add).

   ***Note*** *▪ You cannot select the base node for a Scan Server or a remote scan agent for this function.*

5. Proceed with either procedure:

   - If adding files to the inventory item whose **Inventory Detail**s tab you opened in Step 2, drag and drop the selected files to the **Associated Files** tab on that tab.

   - If adding the selected files to multiple existing inventory items or if creating an inventory item using the selected files as associated files:

     a. Right-click the selected files to open a pop-up menu, and select **Add to Inventory** to open the **Add to Inventory** dialog.

b. Continue with either Add Selected Codebase Files to Existing Inventory or Create a New Inventory Item with Which to Associate Selected Codebase Files.

### Add Selected Codebase Files to Existing Inventory

This procedure describes how to use the **Add to inventory** dialog to add the selected codebase files to one or more existing inventory items in the **Analysis Workbench**. (Steps to access the **Add to inventory** dialog are described in the preceding main procedure, Adding Files to Inventory From the Codebase List.)

*Task* ***To add the selected codebase files to existing inventory, do the following:***

1. In the **Add to inventory** dialog, select one or more inventory items to which to add the selected codebase file or files. (You can use the search field to search for the inventory.)
2. (Optional) Click **Mark files as reviewed**.
3. Click **Submit** to add codebase files to the **Associated Files** tab for each selected inventory item in the **Analysis Workbench**.

### Create a New Inventory Item with Which to Associate Selected Codebase Files

This procedure describes how to use the **Add to inventory** dialog to create a new inventory item with which to associate the selected codebase files in the **Analysis Workbench**. (Steps to access the **Add to inventory** dialog are described in the preceding main procedure, Adding Files to Inventory From the Codebase List.)

*Task* ***To create a new inventory item with which to associated selected codebase, do the following:***

1. In the **Add to inventory** dialog, click **Add New**. A new inventory item “candidate”, showing default values, opens in its own tab on the **Inventory Details** tab.

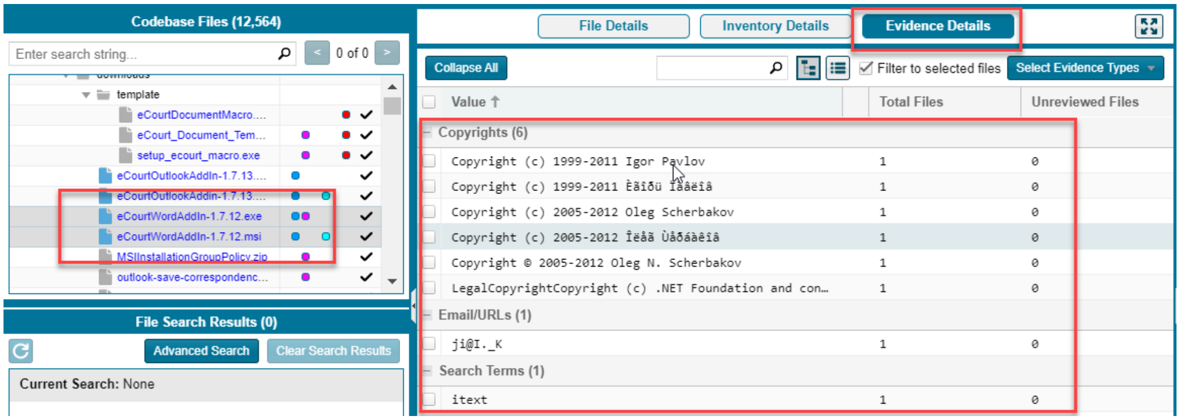
   Note that the selected codebase files for which you are creating the inventory item are automatically added to the **Associated Files** tab for the new inventory item in the **Analysis Workbench**.
2. Complete the fields to define the new inventory item, as described in the later section, Creating Inventory from the Inventory Items List.
3. (Optional) Drag and drop one or more additional files from the **Codebase Files** list (or **File Search Results** list) to the **Associated Files** tab.
4. When you completed the details for the new inventory item, click **Save**. The name of the inventory item appears in the **Inventory Items** pane.

## Listing Copyright, Email, URL, License, and Search-Term Evidence for Files Selected in the Codebase List

Use the following procedure to view a list of all copyright, email, URL, license, and search-term evidence found in one or more files selected in the **Codebase Files** or **File Search Results** pane. The **Evidence Details** tab, which by default lists such evidence for the entire codebase, is filtered to list evidence for the selected files only.

***Task*** ***To list the evidence for one or more codebase files, do the following:***

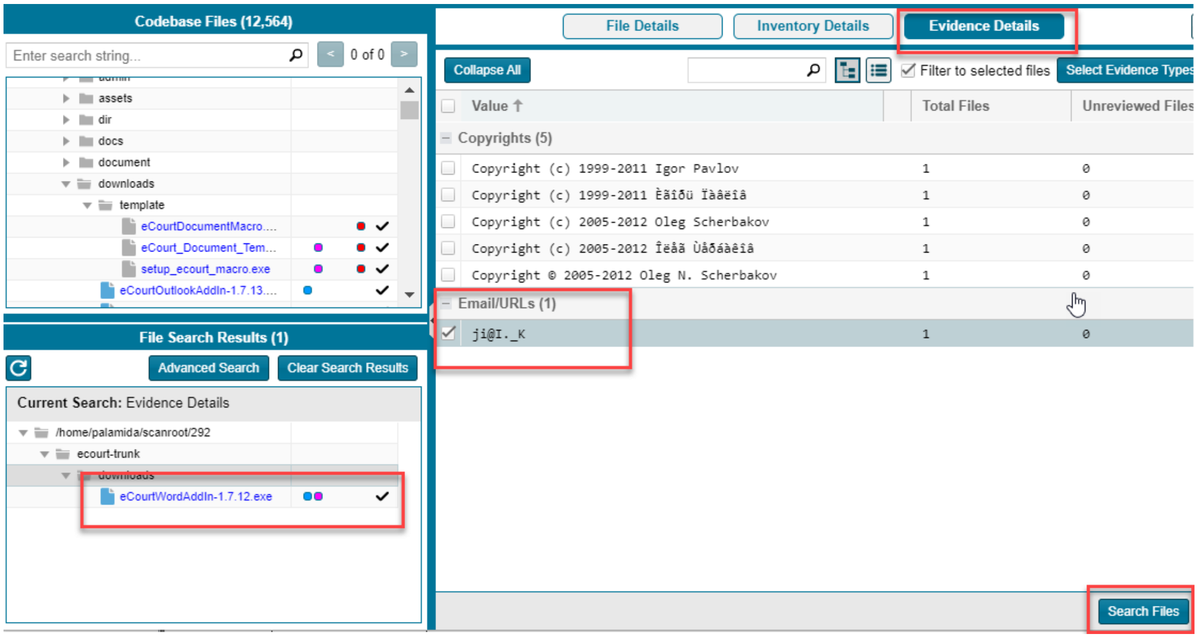
1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, select and right-click an individual file or a set of files whose evidence you want to view. (You can also right-click a directory to select all files in that directory and its subdirectories.)
3. From the pop-up menu, select **Show file evidence**. The **Evidence Details** tab opens in the center of the **Analysis Workbench**, listing the evidence found in the selected files.



| Value ↑ | Total Files | Unreviewed Files |
|---|---|---|
| Copyrights (6) | | |
| Copyright (c) 1999-2011 Igor Pavlov | 1 | 0 |
| Copyright (c) 1999-2011 Èãîðü Ïàâëîâ | 1 | 0 |
| Copyright (c) 2005-2012 Oleg Scherbakov | 1 | 0 |
| Copyright (c) 2005-2012 Îëåã Ùåðáàêîâ | 1 | 0 |
| Copyright © 2005-2012 Oleg N. Scherbakov | 1 | 0 |
| LegalCopyrightCopyright (c) .NET Foundation and con… | 1 | 0 |
| Email/URLs (1) | | |
| ji@I._K | 1 | 0 |
| Search Terms (1) | | |
| itext | 1 | 0 |

4. (Optional) To determine which of the selected files contains a specific instance of evidence (for example, a specific copyright or search term), select the checkbox next to the instance on the **Evidence Details** tab, and click **Search Files** (bottom right of the tab).

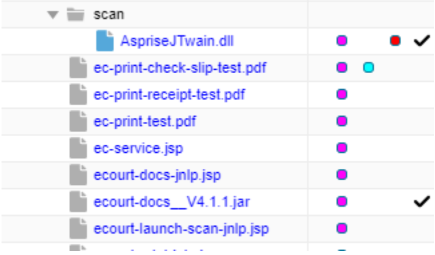
   The associated files are listed in the **Files Search Results** pane. (You can also select multiple evidence instances in this step.) For more details about using the **Evidence Details** tab, see Examining Evidence of Open-Source Copyrights, Email Addresses, URLs, Licenses, and Search Terms in a Given Non-Binary File.



| Value | Total Files | Unreviewed Files |
|---|---|---|
| Copyrights (5) | | |
| Copyright (c) 1999-2011 Igor Pavlov | 1 | 0 |
| Copyright (c) 1999-2011 Èãîðü Ïàâëîâ | 1 | 0 |
| Copyright (c) 2005-2012 Oleg Scherbakov | 1 | 0 |
| Copyright (c) 2005-2012 Îëåã Ùåðáàêîâ | 1 | 0 |
| Copyright © 2005-2012 Oleg N. Scherbakov | 1 | 0 |
| Email/URLs (1) | | |
| ji@I._K | 1 | 0 |

# Marking Codebase Files as Reviewed

It is important to keep track of which files have been audited by marking files as reviewed when you are finished auditing them. If necessary, you can use the **Advanced Search** button on the **File Search Results** pane to filter to only un-reviewed files to see what is left to evaluate. You can also see the progress of the audit on the **Summary** tab. Files that have been marked as reviewed show a checkmark to the right of the file name in the **Codebase Files** and **File Search Results** panes.


scan
AspriseJTwain.dll ✓
ec-print-check-slip-test.pdf
ec-print-receipt-test.pdf
ec-print-test.pdf
ec-service.jsp
ecourt-docs-jnlp.jsp
ecourt-docs__V4.1.1.jar ✓
ecourt-launch-scan-jnlp.jsp

When all files have been marked as reviewed, an overview-style audit can be considered completed.

***Task*** ***To mark files as reviewed, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, select and right-click an individual file or a set of files that you want to mark as reviewed. (You can also right-click a directory to select all files in that directory and its subdirectories.)

***Note ▪** You cannot select the base node for a Scan Server or a remote scan agent for this function.*

3. From the pop-up menu, select **Mark as reviewed**. A checkmark is added to the right of each selected file to indicate that it now has a reviewed status.

   ***Note ▪** If you enabled the project scan setting that automatically publishes inventory, you can also enable the setting that automatically marks files associated with this inventory as reviewed. For more information about these settings, see Edit Project: Scan Settings Tab.*

## Reverting Codebase Files to Unreviewed Status

In some cases, a files marked as reviewed might need to be reverted to an unreviewed status. This can happen, for example, if new evidence or security vulnerabilities require more investigation of the file contents.

***Task*** ***To revert reviewed files to unreviewed status, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, select and right-click an individual file or a set of files that you want to revert to unreviewed. (You can also right-click a directory to select all files in that directory and its subdirectories.)

   ***Note ▪** You cannot select the base node for a Scan Server or a remote scan agent for this function.*

3. From the pop-up menu, select **Mark as unreviewed**. The checkmark to the right of each selected file is removed.

## Downloading a Codebase File

You can download an individual codebase file to your browser's default download location.

***Note ▪** Currently, this option is available only for files scanned by the Scan Server, not for files scanned by a Code Insight scan-agent plugin on a remote system.*

***Task*** ***To download a codebase file, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, select and right-click the file that you want to download.

***Note ▪** You can select only a single file for this function, not a folder or the base node for a Scan Server or a remote scan agent.*

3. From the pop-up menu, select **Download File**. The file is downloaded to your browser's default location.

## Copying Codebase File and Folder Paths

You can copy the complete paths of files and folders listed in **Codebase Files** or **File Search Results** pane, enabling you to paste accurate path information to other parts of the project (for example, to the **Audit Notes** field), between projects, or in personal locations. The following describes the format in which the paths are copied:

- The output for a file-path copy uses the format <alias>:<relativeFilePath>, where <alias> is the meaningful name given to a scanner (Scan Server or remote scan agent) to represent its scan-root container, and <relativeFilePath> is the file path relative to the absolute scan-root path on host instance. For example, if you copy the codebase file **agpl-3.0.txt**, located directly under the scan folder **ePortal-1.3**, which in turn is directly under the scan-root path for the Scan Server whose alias is "EP_remote", the output for the copy is **EP_remote:ePortal-1.3/agpl-3.0.txt**.
- Likewise, the output for a folder-path copy uses a similar format, <alias>:<relativeFolderPath>. For example, if you copy the **src** folder, located directly under the scan folder **ePortal-1.3**, which in turn is directly under the scan-root path for the Scan Server whose alias is "EP_remote", the folder for the copy is **EP_remote:ePortal-1.3/src**.

***Note ▪** When you copy a folder path, only the path for the folder is copied, not the paths for the files within the folder.*

***Task*** ***To copy codebase file and folder paths, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. In the **Codebase Files** or **File Search Results** pane of the **Analysis Workbench**, perform any of these tasks to copy codebase file or folder paths to the operating system Clipboard:
    - **To copy the path for a single file**—Right-click the file, and select **Copy File Path** from the pop-up menu.
    - **To copy paths for multiple files**—Select the files (using the Ctrl or Shift key), right-click anywhere in the group, and select **Copy Paths** from the pop-up menu.
    - **To copy the path for a single folder**—Right-click the folder, and select **Copy Folder Path** from the pop-up menu.
    - **To copy paths for multiple folders**—Select the folders (using the Ctrl or Shift key), right-click anywhere in the group, and select **Copy Paths** from the pop-up menu.

    When selecting multiple codebase items to copy, select only all files or all folders. Do not select a combination of files and folders. If you do select a combination, the **Copy Paths** option is disabled. Additionally, you cannot select the base node for a Scan Server or a remote scan agent for this function.

3. Use Ctrl + v to paste the copied path or paths to the desired location.

# Managing Inventory in the Analysis Workbench

The **Inventory Details** tab allows you to manage details for a selected inventory item:

- Using the Inventory Items Context Menu
- Viewing Security Vulnerabilities for Inventory in the Analysis Workbench
- Viewing Details About the Component Associated with Inventory in the Analysis Workbench
- Viewing Details About Licenses Associated with Inventory in the Analysis Workbench
- Viewing Associated Files for Inventory in the Analysis Workbench
- Viewing or Editing Inventory Usage Information from the Analysis Workbench
- Viewing and Updating Detection and Auditing Notes in the Analysis Workbench
- Searching Components
- Creating an Inventory Item from the Analysis Workbench
- Editing Inventory from the Analysis Workbench
- Publishing or Recalling Inventory from the Analysis Workbench
- Viewing the Update History for an Inventory Item in the Analysis Workbench

## Using the Inventory Items Context Menu

The **Inventory Items** pane has a context menu containing shortcuts to common inventory tasks. The following tasks are available on the context menu:

- **Publish Inventory**—Select inventory items that you would like to publish, right-click, and choose **Publish Inventory** to quickly publish your selected items. Publishing an inventory item makes it visible in the **Project Inventory** view.
- **Recall Inventory**—Select published inventory items that you would like to recall back to an unpublished state, right-click, and choose **Recall Inventory**. The selected items are removed from the **Project Inventory** view and are only visible in the **Analysis Workbench**.

  ***Note*** ▪ *Editing an inventory item does not require a recall of the inventory item. The item's field values may be edited from the* ***Analysis Workbench*** *or the* ***Project Inventory*** *view at any time, even if the item has already been published.*

- **Show Inventory Files**—To see files associated with the selected inventory items, select the list of inventory items, and right-click and choose **Show Inventory Files**. The associated files will be shown in the **File Search Results** pane.
- **Delete Inventory**—Select inventory items that you want to delete, right-click, and select **Delete Inventory**. The selected items will be deleted from the project.

*__Note__ ▪ When you republish an inventory item by selecting the Recall and Publish tasks, the published date on the item is reset. This action in turn affects the age of the inventory item. Republished items are treated as newly published items.*

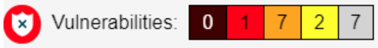
# Viewing Security Vulnerabilities for Inventory in the Analysis Workbench

Code Insight uses data from the National Vulnerability Database (NVD), Secunia advisories (as published by the Secunia Research team from Revenera), and other advisories such as RubySec to report security vulnerabilities associated with your inventory items. The vulnerabilities information from these sources is used to create vulnerability rankings and alerts.

Use this procedure to access details about the security vulnerabilities associated with an inventory item in the **Analysis Workbench**.

***Task*** ***To view security vulnerabilities for an inventory item, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Details** tab for a selected inventory item in the **Analysis Workbench**, locate the **Vulnerabilities** graph. (No graph is displayed if the inventory item has no known associated security vulnerabilities.)

   The severities depicted on the graph differ depending on the CVSS version Code Insight is using (see Working with Security Vulnerabilities). This example shows vulnerability severity counts using CVSS v3.x.
3. Click any of the counts in the graph to open the **Security Vulnerabilities** window, which lists the current security vulnerabilities for the inventory item.

   *__Note__ ▪ Suppressed vulnerabilities are neither reflected in the counts on __Vulnerabilities__ bar graph nor are they visible on __Securities Vulnerabilities__ window.*

   For more information about vulnerabilities, see Working with Security Vulnerabilities.
4. When you have finished viewing the reported vulnerabilities, click **OK** to return to the **Inventory Items** list.

# Viewing Details About the Component Associated with Inventory in the Analysis Workbench

You can view details about the OSS or third-party component with which an inventory item is associated.

***Task*** ***To view details for the component with which an inventory item is associated, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Details** tab for a selected inventory item in the **Analysis Workbench**, locate the **Component** field. This field lists the OSS or third-party component with which the selected inventory item is associated.
3. Click the information icon (ⓘ) next to the **Component** value. The **Component Details** window is displayed, showing information about the component.

   For descriptions of these fields on this tab, see Component Details Window.
4. When you have finished examining the details, click **Close**.

# Viewing Details About Licenses Associated with Inventory in the Analysis Workbench

You can view more details about the licenses associated with the OSS or third-party component on which an inventory item is based. This information is pulled from the Code data library and is displayed on the **License Details** window, which includes the following:

- A **General Information** tab that lists details such as the name of the license, its family, and the license priority assigned by Code Insight.
- A **License Text** tab that displays the complete license text (representing the external forge license text).

While the forge license text is what is likely to be used by the component, the scan might have found actual license text associated with this component in your codebase that can be different from the forge version. To view the exact license text, if any, that the scan discovered and to prepare the final license text, when necessary, to include in the Notices report for distribution to customers, navigate to the **Notices Text** tab. Refer to Finalizing the Notices Text for the Notices Report for more information.

The following procedure describes how to access the **License Details** window from the **Inventory Details** tab in the **Analysis Workbench**.

***Note*** *▪ This window is also accessible when create or edit an inventory item or perform a Component Lookup from the* ***Analysis Workbench****.*

***Task*** ***To view details for the inventory license, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Details** tab for a selected inventory item in the **Analysis Workbench**, locate the **License** field. This field lists the license currently that is associated with the component identified for the inventory item.

3. Click the information icon (ⓘ) next to the **License** value. The **License Details** window appears with the **General Information** tab in focus.

   For descriptions of these fields on this tab, see License Details Window. Also see License Priority for background on how the license priority is used.

4. Select the **License Text** tab to view the license text.

5. When you have finished examining the license details, click **Close**.

## Viewing Associated Files for Inventory in the Analysis Workbench

Use this procedure to view the codebase files that have been automatically or manually associated with the inventory item currently selected in the **Inventory Items** pane in the **Analysis Workbench**.

*Task* ***To view the codebase files currently associated with a given inventory item, do the following:***

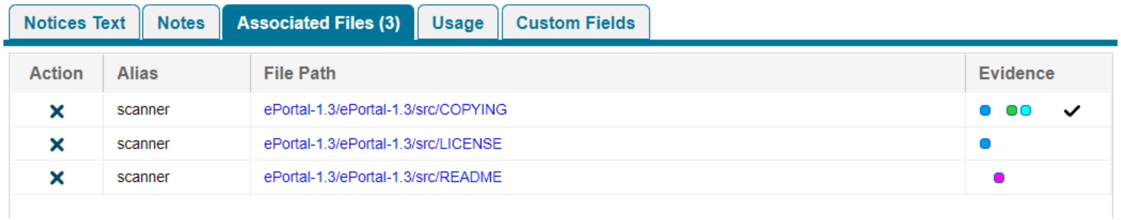
1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)

2. Select an inventory item from the **Inventory Items** pane in the **Analysis Workbench**.

3. In the **Inventory Details** tab that is opened in the middle pane for the inventory item, click the **Associated Files** tab. The tab lists the codebase files currently associated with the selected inventory item. Each file entry shows the following details. (Note that you cannot sort the file list.)

   - **Action**—Icons that you can click to perform certain actions on the file. Currently, only the ✕ icon shows, enabling you to disassociate the file from the inventory item.
   - **Alias**—The unique, user-defined name provided during scanner setup (for a Scan Server or a remote scan agent) to represent its scan-root path containing the codebase in which the file is located. The alias provides a name that is more meaningful than the scan-root path. (The actual absolute scan-root path for each scanner associated with the project is available on the project's **Summary** tab.)
   - **File Path**—The file's path relative to the scan-root path on instance hosting the scanner. You can click the file-path link to open the **File Details** tab for that file in the Codebase Files view.
   - **Evidence**—The color-coded icons representing the types of open-source or third-party evidence found in the file (see Using the Filter Legend Options to Filter the Codebase for a description of the icons). A check mark indicates that the file has been reviewed.

   ***Note** ▪ Currently, license evidence is the only type of open-source and third-party evidence reported for files scanned by a Code Insight scan-agent plugin on a remote system.*



| Action | Alias | File Path | Evidence |
|---|---|---|---|
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/COPYING | ● ●● ✓ |
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/LICENSE | ● |
| ✕ | scanner | ePortal-1.3/ePortal-1.3/src/README | ● |

4. (Optional) Right-click a file entry for a list of options that enable you to perform certain operations on the file, such as marking it as reviewed, reverting its reviewed status to unreviewed, and other operations. See Managing the Codebase Files for details about these same options that are also available from the **Codebase Files** and **File Search Results** panes in the **Analysis Workbench**.

## Viewing or Editing Inventory Usage Information from the Analysis Workbench

Inventory usage information is important because it aids users in determining how closely to monitor inventory items for intellectual property (IP) and security risk and in taking appropriate action to approve or reject inventory, create tasks for remediation, and issue alerts and notifications. Usage fields can determine whether the item should be included in Third-Party Notices and what steps need to be taken to satisfy license obligations and conditions of use. They can also help to identify license conflicts and compatibility issues.

***Task*** ***To view or edit inventory usage information, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Details** tab for a selected inventory item in the **Analysis Workbench**, select the **Usage** tab.
3. View and, if necessary, edit the usage fields.

   For details about the inventory usage fields and how they are used, see Inventory Usage Information.

## Viewing and Updating Detection and Auditing Notes in the Analysis Workbench

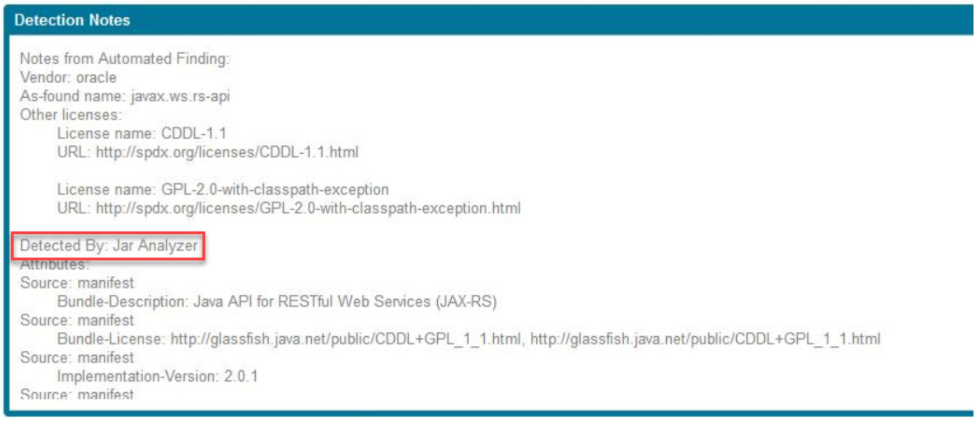
The **Notes** tab can provide information about automated and manual analysis of codebase as it relates to an inventory item. This can help you in your analysis of your product’s use of the OSS or third-party software identified by the inventory item.

***Task*** ***To view notes, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Details** tab for a selected inventory item in the **Analysis Workbench**, select the **Notes** tab.
3. Review or update content in the following fields as needed:

   - **Detection Notes**—Information generated during the scan to explain the means by which the scan detected OSS or third-party software in the codebase. The **Detected By** attribute in the **Detection Notes** content indicates the automated detection technique(s) responsible for the inventory finding. The following example shows that the detection technique used to find the inventory item was Jar Analyzer.

     The **Detection Notes** content is not editable.



```
Notes from Automated Finding:
Vendor: oracle
As-found name: javax.ws.rs-api
Other licenses:
        License name: CDDL-1.1
        URL: http://spdx.org/licenses/CDDL-1.1.html

        License name: GPL-2.0-with-classpath-exception
        URL: http://spdx.org/licenses/GPL-2.0-with-classpath-exception.html

Detected By: Jar Analyzer
Attributes:
Source: manifest
        Bundle-Description: Java API for RESTful Web Services (JAX-RS)
Source: manifest
        Bundle-License: http://glassfish.java.net/public/CDDL+GPL_1_1.html, http://glassfish.java.net/public/CDDL+GPL_1_1.html
Source: manifest
        Implementation-Version: 2.0.1
Source: manifest
```

    - **Audit Notes**—Information recorded about the manual analysis of the codebase associated with this software. You can add your own notes. For example, you might indicate that you needed to create this inventory item manually.

4. Click **Save** in the upper right corner of the **Inventory Details** tab to save any updates to the **Audit Notes** field.

# Searching Components

Component Lookup is the Code Insight feature used to find more information about the open-source or third-party components available for inventory. This information can include security vulnerabilities and potential license issues associated with the component (and hence your inventory), as well as other data. The following sections describe how to use Component Lookup:

- Guidelines for Component Lookup
- Component Lookup Results
- Performing Component Lookup

## Guidelines for Component Lookup

When possible, use the **Forge** or **URL** search for the most targeted search results, and use the **Keyword** search in other cases.

- Use the **Forge** option if you know the forge (that is, the third-party project repository) of the component. For example, Github, NuGet Gallery, and PyPI are forges.
- Use the **URL** option if you know project URL or the forge URL. For example, https://github.com/jquery/jquery or http://jqueryui.com.
- Use the **Keyword** option to search all the component names in the Code Insight data library. The component name is a unique identifier that may be based on the project name, package name, gem name, or other convention such as author and repository. The following are common conventions for component names:
    - **Github**— <AUTHOR>-<REPOSITORY_NAME>, for example “jquery-jquery-ui”
    - **NuGet Gallery**— <PACKAGE_NAME>, for example “newtonsoft.json”
    - **Apache**— <PROJECT_NAME>, for example “apache-batik”

  - **Pypi**—<PACKAGE_NAME>, for example “hash_ring”
  - **RubyGems**— <GEM_NAME>, for example “x-editable-rails”
  - **GitLab**—<AUTHOR/ORGANIZATION>-<REPOSITORY_NAME>, for example:
    - “cryptsetup-cryptsetup” (as found in component URL: `https://gitlab.com/cryptsetup/cryptsetup`)
    - “redhat-bison” (as found in component URL: `https://gitlab.com/redhat/centos-stream/rpms/bison`)
  - **Other**— <PROJECT_NAME>, for example “openssl”

- If you cannot locate the component by keyword, select a **Forge** or **URL** search. If you are still unable to locate the component, the component might not exist in the Code Insight data library. In this case, you have options to do one of the following:
  - Create your inventory item as **Work in Progress** and name it using the convention *<COMPONENT> <VERSION> (<LICENSE>)*. For example, `myComponent 1.2 (MIT)`. (You can later edit this inventory item to convert it to one of the other inventory types—**Component** or **License**.)
  - Create a custom component. See Creating and Editing Custom Components for details.

## Component Lookup Results

Component Lookup search results are prioritized in the following order:

1. **Registered Components**—Components with a history of use (one or more instances of the component are registered for use in the system).
2. **Important Components**—Components that are marked by Code Insight as important due to popularity or presence of security vulnerabilities.
3. **All other Components**—Components that are neither registered nor important.

If no results are returned, the component might not exist in the Code Insight data library. See the previous section for options in dealing with components not available in the library.

## Performing Component Lookup

Use this procedure to search for a specific component.

*Task* ***To perform a Component Lookup, do the following:***

1. Open the desired project and navigate to the **Analysis Workbench** (or the **Project Inventory** tab). If necessary, see Opening a Project.
2. Select an inventory item from the list in the **Inventory Items** pane. The item appears in the **Inventory Details** pane. (For an item selected from the **Project Inventory** tab, you need to also click **Edit Item** to proceed with the next step.)
3. From the **Type** dropdown, select **Component** and click **Lookup Component**. The **Lookup Component** window appears.

4. Select the search type (**Keyword**, **URL**, or **Forge**), and enter the required search criteria. See Guidelines for Component Lookup.

   Click **Search** to find components matching your search criteria. For information about the results of the Component Lookup, see Component Lookup Results.

# Creating an Inventory Item from the Analysis Workbench

When you identify third-party code in your codebase, you should create an inventory item to record it. Inventory items contain information critical for review and approval. The process for creating inventory in the **Analysis Workbench** proceeds in the following way:

**Phase 1**—Filter files that contain evidence of third-party code, such as copyright text or content from an open source license. See Searching for Codebase Files Based on Search Criteria and Viewing a Summary of Evidence Detected Across the Codebase.

**Phase 2**—Research the findings and identify the origin of the files.

**Phase 3**—Create an inventory item with details about the origin of the code. This is typically an open source project, such as zlib, OpenSSL, or ReactJS.

If you do not know code’s origin, you have options to create either a **License Only** inventory item (if the codebase files are governed by a common license) or a **Work In Progress** inventory item to serve as placeholder until you obtain more information. Inventory types are described in more detail in the procedure below.

**Phase 4**—When all of the evidence is explained in the files you are looking at (bearing in mind that some files might have code from several origins), mark the files as “reviewed”.

**Phase 5**—When you are finished creating inventory items, publish the ones you would like to report on. You can choose not to publish internal or test tools.

For more details about creating inventory items in the **Analysis Workbench**, see the following sections:

- Creating Inventory from the Inventory Items List
- Creating Inventory with Associated Files from the Codebase Lists

## Creating Inventory from the Inventory Items List

This section describes how to create inventory from the **Inventory Items** list in the **Analysis Workbench**. (For instructions on creating inventory items for *codebase files* in **Codebase Files** list or **File Search Results** list in the **Analysis Workbench**, see Creating Inventory with Associated Files from the Codebase Lists.)

*Task* ***To create inventory from the Inventory Items list, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. Navigate to the **Inventory Items** list.
3. Click **Add New** at the top of the **Inventory Items** list. A new item, showing default values, opens in its own tab in the **Inventory Details** pane.

4. For the **Name** field, perform the appropriate step, based on the inventory type you intend to select for the **Type** field (see the next step):

    - For inventory of the type **Work in Progress**, specify a name for the inventory item. Best practice is to provide a name in the following conventional syntax used by Code Insight, even if the elements represented in the name are not available in the data library:

        ```
        <COMPONENT_NAME> <VERSION> (LICENSE_NAME)
        ```

    - For inventory of the type **Component** or **License only**, leave the **Name** field blank. The field will be automatically populated based on the registered component or license instance.

5. From the **Type** dropdown, select the type of inventory item you want to create and perform the related step or steps:

    - **Work in Progress**—Create this type of inventory item if you want to quickly represent third-party code or an artifact without having to select an associated component, version, or license from the data library. (You can later edit this inventory item to convert it to one of the other inventory types.) This option is typically used if you need a placeholder or cannot find the associated element in the data library. Items of type **Work in Progress** are not affected by policies and do not receive vulnerability updates or alerts.

    - **Component**—Create this type of inventory item if you know the component, version, and license for the third-party code or artifact *and* either you are able to locate it in the Code Insight data library using the Component Lookup feature or you need to create it as a custom component because it is not in the library. This type of inventory is associated with a registered component instance—that is, a unique component-version-license combination—and is affected by policies and receives vulnerability updates and alerts.

        The Component Lookup feature, made available when you select the **Component** type, enables you to associate the inventory item with an existing registered component instance (or a new instance that you create) or to create the custom component and instance to associate with the item.

        The following are basic steps for using Component Lookup. For details, see Searching Components.

        a. Click the **Lookup Component** button to locate the component of interest (as described in the next steps) or to create a custom component and instance to associate with the inventory item (see Creating and Editing Custom Components for continued steps).

        b. In the list of results, navigate to the appropriate component, and click **Show Versions** to display the list of registered instances for that component.

        c. Click **Use This Instance** next to an existing registered instance to associate that instance with the inventory item.

        or

        Click **Register New Instance** to create a new instance. Complete the registration by selecting an existing component version (or choosing the **Create Custom Version** value to specify a new version) and then selecting the license to associate with the instance. Click **Use This Instance** next to the new instance to associate it with the inventory item. (If you register a new component instance when creating inventory, the registered instance becomes available for selection across the system.)

        The **Name** and **Description** editable fields for the inventory item are automatically populated with information based on the registered instance you selected. Additionally, the **Component** and **License** fields are displayed, showing the component, its version, and the license for the instance.

- **License Only**—Create this type of inventory item if you know the license for the third-party code or artifact but do not know the component. (You can later edit this inventory item to convert it to one of the other inventory types.) This type of inventory is typically used for groups of files of unknown origin that are governed by a specific license. The inventory is affected by policies.

  Simply select the appropriate license from **License Only** dropdown, which is enabled when you select this type.

  The **Name** field for the inventory item is automatically populated with the name **Files under <LICENSE_NAME> License**, where <LICENSE_NAME> is license you selected.

6. Update the remaining fields if appropriate. For a description of each field, see Inventory Details Tab in the Analysis Workbench.

7. When you completed the details for the new inventory item, click **Save**. The name of the inventory item appears in the **Inventory Items** pane.

8. (Optional) To report on newly created or edited inventory items, click **Publish**.

## Creating Inventory with Associated Files from the Codebase Lists

You can create a new inventory item based on files you have selected in the **Codebase Files** or **File Search Results** list in the **Analysis Workbench**. For more information, see Adding Files to Inventory From the Codebase List.

# Editing Inventory from the Analysis Workbench

Use the following steps to edit an inventory item from the **Analysis Workbench** as needed.

*Task* ***To edit an inventory item in the Analysis Workbench, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)

2. Navigate to the **Inventory Items** list in the right pane.

3. Select the inventory item that you want to edit.

   A new tab, labeled with the inventory name and showing information about the inventory item, is opened within the **Inventory Details** tab.

4. Make changes to the fields as needed. Refer to Creating Inventory from the Inventory Items List for field descriptions and additional steps required when updating the inventory type.

   Note the following:

   - For a **Component** inventory item, you can use the Component Lookup feature to select a different registered instance (or create a new one) or to create a custom component and instance to associate with the inventory item. The **Name**, **Component**, and **License** fields are updated accordingly. See Creating Inventory from the Inventory Items List for complete information.

   - You can convert a **Work In Progress** or **License Only** inventory item to a different inventory type, using the **Type** field and performing the additional steps required for the type. The **Name** and other appropriate fields are updated accordingly.

- If you need to update the component version or associated license only, simply click the Edit (✎) icon next to the **Component** or **License** value, and select a different license or version or both. (This avoids performing a Component Lookup process to edit these elements.)
- Update any of the other fields as necessary. For a description of each field, see Inventory Details Tab in the Analysis Workbench.

5. Click **Save** to change the changes to the inventory item.

## Associating Codebase Files with Inventory in the Analysis Workbench

For instructions on associating codebase files with existing or newly created inventory, see Adding Files to Inventory From the Codebase List.

## Publishing or Recalling Inventory from the Analysis Workbench

If you have performed manual work on your inventory items, you must publish the items to **Project Inventory** before anyone can review your work. Likewise, you can recall a published inventory item (that is, remove it from Project Inventory) for further auditing.

In the **Analysis Workbench**, you publish or recall inventory from either the **Inventory Details** tab or the **Inventory Items** pane:

- Publish or Recall an Inventory Item from the Inventory Details Tab
- Publish or Recall Inventory from the Inventory Items Pane

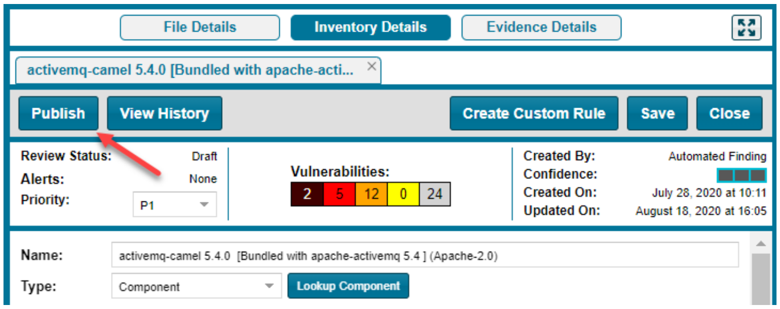
***Note*** ▪ *If you enabled the auto-publish feature in the project scan settings, you do not need to perform the steps below because system-created inventory items are automatically published.*

### Publish or Recall an Inventory Item from the Inventory Details Tab

You these steps to publish or recall an inventory item from the Inventory Details tab.

***Task*** ***To publish or recall an inventory item from its Inventory Details tab, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. For the unpublished inventory item currently in focus on the **Inventory Details** tab in the **Analysis Workbench**, click the **Publish** button. The newly published item now appears in the **Inventory Items** list with a filled box icon before its name (and is now listed in **Project Inventory**).



Review Status: Draft
Alerts: None
Priority: P1

Vulnerabilities: 2 | 5 | 12 | 0 | 24

Created By: Automated Finding
Confidence:
Created On: July 28, 2020 at 10:11
Updated On: August 18, 2020 at 16:05

Name: activemq-camel 5.4.0 [Bundled with apache-activemq 5.4 ] (Apache-2.0)
Type: Component

Conversely, for a published inventory item currently in focus, click the **Recall** button. The item now appears in the **Inventory Items** list with a clear box icon before its name (and is no longer listed in **Project Inventory**).

## Publish or Recall Inventory from the Inventory Items Pane

Use the following procedure to publish or recall one or more inventory items from the **Inventory Items** pane.

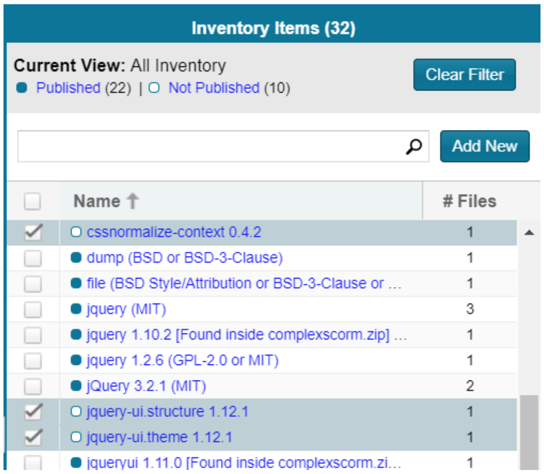
***Task*** ***To publish or recall inventory from the Inventory Items pane, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Items** pane of the **Analysis Workbench**, select the items to publish so that a checkmark appears in front of each item.

   or

   Select the published items you want to recall so that a checkmark appears in front of each item.

   ***Note*** ▪ *If you do not see an inventory item you want to publish or recall, enter a term to search and click the search magnifier button.*



**Current View:** All Inventory
● Published (22) | ○ Not Published (10)

| ☐ | Name ↑ | # Files |
|---|---|---|
| ☑ | ○ cssnormalize-context 0.4.2 | 1 |
| ☐ | ● dump (BSD or BSD-3-Clause) | 1 |
| ☐ | ● file (BSD Style/Attribution or BSD-3-Clause or ... | 1 |
| ☐ | ● jquery (MIT) | 3 |
| ☐ | ● jquery 1.10.2 [Found inside complexscorm.zip] ... | 1 |
| ☐ | ● jquery 1.2.6 (GPL-2.0 or MIT) | 1 |
| ☐ | ● jQuery 3.2.1 (MIT) | 2 |
| ☑ | ○ jquery-ui.structure 1.12.1 | 1 |
| ☑ | ○ jquery-ui.theme 1.12.1 | 1 |
| ☐ | ● jqueryui 1.11.0 [Found inside complexscorm.zi... | 1 |

3. Right click to open the context menu, and choose either **Publish Inventory** or **Recall Inventory**.

    - If you selected **Publish Inventory**, the newly published items appear in the **Inventory Items** list with a filled box icon before their names (and are now listed in **Project Inventory**).
    - If you selected **Recall Inventory**, the recalled items appear in the **Inventory Items** list with a clear box icon before their name (and are no longer listed in **Project Inventory**).

    ***Note*** ▪ *During the scan, inventory item priorities for auto-published inventory are automatically assigned based on the associated license.*

## Viewing the Update History for an Inventory Item in the Analysis Workbench

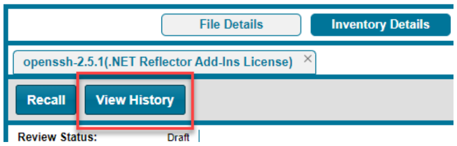
From the **Analysis Workbench**, you can view a history of the updates made to a specific inventory item.

***Note*** ▪ *You have access to this same history from the* ***Project Inventory*** *tab. See Viewing the Update History for an Inventory Item in Project Inventory.*

***Task*** ***To view the update history of an inventory item, do the following:***

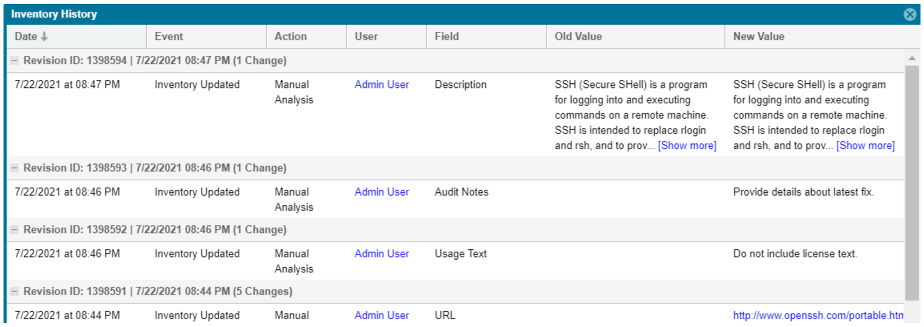
1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Items** pane of the **Analysis Workbench**, select the inventory item whose update history you want to view.

    The **Inventory Details** pane is opened (or refreshed) with information about the selected inventory item.
3. In the **Inventory Details** pane, click the **View History** button.

    Review Status: Draft

    The **Inventory History** window is opened, showing the list of updates made to the inventory item. By default, the updates are listed in descending order by date so that you see the most recent updates first. Each update record identifies—among other details—the update type, the user who made the update, and the before-and-after values in the update. For a description of all features on this window, see Inventory History Window.



Inventory History

| Date ↓ | Event | Action | User | Field | Old Value | New Value |
|---|---|---|---|---|---|---|
| Revision ID: 1398594 \| 7/22/2021 08:47 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:47 PM | Inventory Updated | Manual Analysis | Admin User | Description | SSH (Secure SHell) is a program for logging into and executing commands on a remote machine. SSH is intended to replace rlogin and rsh, and to prov... [Show more] | SSH (Secure SHell) is a program for logging into and executing commands on a remote machine. SSH is intended to replace rlogin and rsh, and to prov... [Show more] |
| Revision ID: 1398593 \| 7/22/2021 08:46 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:46 PM | Inventory Updated | Manual Analysis | Admin User | Audit Notes | | Provide details about latest fix. |
| Revision ID: 1398592 \| 7/22/2021 08:46 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:46 PM | Inventory Updated | Manual Analysis | Admin User | Usage Text | | Do not include license text. |
| Revision ID: 1398591 \| 7/22/2021 08:44 PM (5 Changes) | | | | | | |
| 7/22/2021 at 08:44 PM | Inventory Updated | Manual | Admin User | URL | | http://www.openssh.com/portable.htm |

# Reviewing Published Inventory for a Project

The **Project Inventory** tab shows a list of all the inventory items that have been published for the current project, either automatically by the system or manually by a Reviewer or Analyst. From the **Project Inventory** tab, users can view details for the inventory item, and designated reviewers for the project can manage existing inventory (that is, set the status, change inventory priority, edit details, create and inventory, and manage review and remedial tasks).

Refer to the Code Insight User Roles and Permissions appendix for the project roles (in addition to the Reviewer and Analyst roles) required to review and act on published project inventory.

The following topics describe the various actions you can perform review and manage project inventory:

- Goal of the Reviewer
- Displaying Project Inventory
- Searching Published Inventory
- Viewing Security Vulnerabilities for Project Inventory
- Viewing Details About the Licenses Associated with Project Inventory
- Viewing and Updating Notes and Guidance
- Viewing Usage Information for Project Inventory
- Viewing Associated Files
- Creating Inventory from the Project Inventory Tab
- Editing Inventory from the Project Inventory Tab
- Approving or Rejecting Inventory Items
- Creating and Managing Tasks for Project Inventory
- Creating and Viewing External Work Items for a Project Inventory Task
- Recalling a Published Inventory Item
- Viewing the Update History for an Inventory Item in Project Inventory

# Goal of the Reviewer

The goal of the inventory review is to assess every inventory item and categorize it as *approved* or *rejected* for use in the current project based on your company policy. To review inventory, the user first must be assigned the role of Reviewer (or a role with Reviewer permissions). See Assigning and Removing Project Users.

# Displaying Project Inventory

When an inventory item has been published, it can be reviewed, updated, and reported on from the **Project Inventory** tab. Use this procedure to display the **Project Inventory** tab.

***Task*** ***To view project inventory, do the following:***

1. Open the project whose published inventory you want to review. (For instructions, see Opening a Project.)
2. Click the **Project Inventory** tab. For details about this tab, see Project Inventory Tab.
3. From the **Inventory Items** list on the left, select the inventory item you want to review.

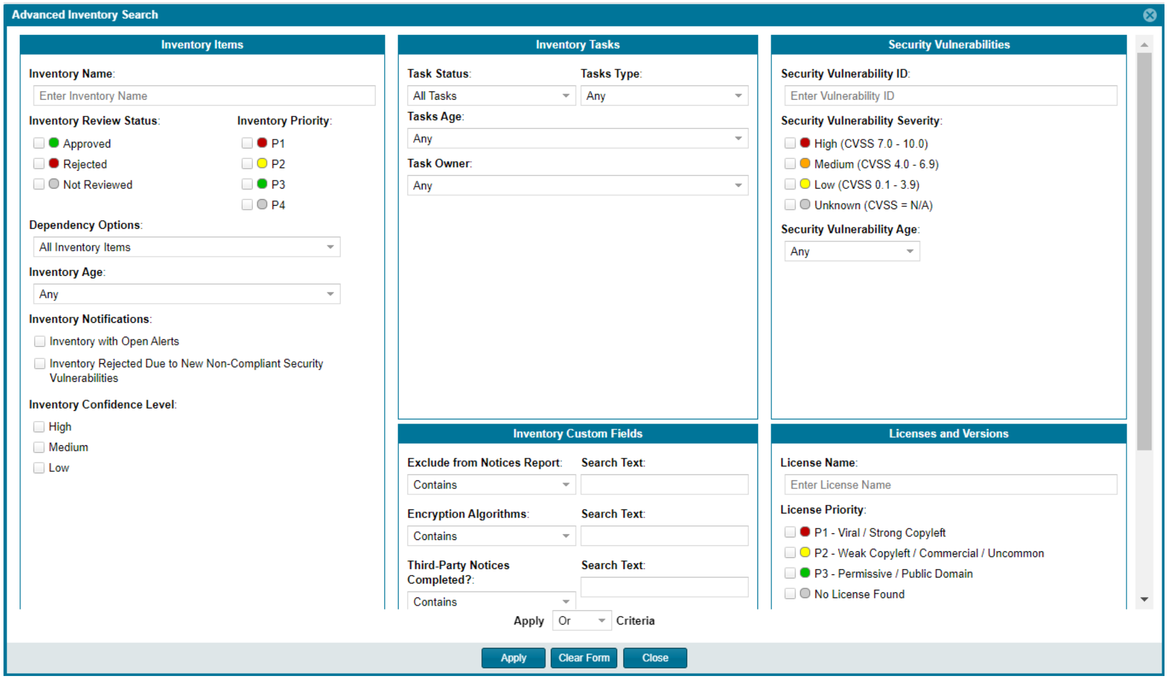
   Details for the selected inventory item populate the **Project Inventory Details** pane on the right. From this pane, you can edit inventory properties, recall an inventory item, finalize the third-party Notices content, set up review and remediation tasks, and provide audit, usage-guidance, and remediation notes—with the ultimate goal of approving or rejecting the item for the Bill of Materials. For complete information about this pane, see Project Inventory Details Pane.

# Searching Published Inventory

Code Insight provides the **Advanced Search** dialog to enable you to quickly filter the list of published inventory items to those of interest based on many available criteria—inventory attributes, selected license attributes, and associated security vulnerabilities, tasks, and security alerts. In this way, you can easily focus on only those inventory items in which you are interested within the list of published items. The following procedure shows you how to access and use this dialog. Refer also to the Performing Advanced Searches chapter for practical applications of this search feature.

***Task*** ***To filter published inventory, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory). The **Inventory Items** pane appears, showing the list of inventory items.
2. Click the **Advanced Search** button at the top of the list to open the **Advanced Inventory Search** dialog.



Advanced Inventory Search

**Inventory Items**

Inventory Name: Enter Inventory Name

Inventory Review Status: □ ● Approved □ ● Rejected □ ● Not Reviewed

Inventory Priority: □ ● P1 □ ● P2 □ ● P3 □ ● P4

Dependency Options: All Inventory Items

Inventory Age: Any

Inventory Notifications: □ Inventory with Open Alerts □ Inventory Rejected Due to New Non-Compliant Security Vulnerabilities

Inventory Confidence Level: □ High □ Medium □ Low

**Inventory Tasks**

Task Status: All Tasks

Tasks Type: Any

Tasks Age: Any

Task Owner: Any

**Security Vulnerabilities**

Security Vulnerability ID: Enter Vulnerability ID

Security Vulnerability Severity: □ ● High (CVSS 7.0 - 10.0) □ ● Medium (CVSS 4.0 - 6.9) □ ● Low (CVSS 0.1 - 3.9) □ ● Unknown (CVSS = N/A)

Security Vulnerability Age: Any

**Inventory Custom Fields**

Exclude from Notices Report: Contains Search Text:

Encryption Algorithms: Contains Search Text:

Third-Party Notices Completed?: Contains Search Text:

**Licenses and Versions**

License Name: Enter License Name

License Priority: □ ● P1 - Viral / Strong Copyleft □ ● P2 - Weak Copyleft / Commercial / Uncommon □ ● P3 - Permissive / Public Domain □ ● No License Found

Apply Or Criteria

3. From this dialog, select search criteria as needed from the following categories. For a detailed description of the search criteria, see Advanced Inventory Search Dialog.

    - **Inventory Items**—Search for inventory items that have a certain name (or string), priority, review status, or age or that have open vulnerability alerts and work items. (For details on alerts and work items, see Managing Security Vulnerability Alerts and Creating and Viewing External Work Items for a Project Inventory Task.)
    - **Inventory Tasks**—Search for inventory items that have been assigned tasks. You can refine the search to locate inventory with open or closed tasks, tasks of a certain age or type (such as manual reviews or source-code remediation), or tasks assigned to a specific user.
    - **Inventory Custom Fields**—Search for inventory whose custom inventory fields contain the value you specify as criteria. Custom inventory fields are defined specifically for your site. If no such fields have been defined this section is not visible.
    - **Security Vulnerabilities**—Search for inventory items that have vulnerabilities of a certain vulnerability ID, CVSS severity, or age. (Note that list of available severities for **Security Vulnerability Severity** varies depending on the CVSS version being used by Code Insight. The picture above shows the severities for CVSS v3.x. See Working with Security Vulnerabilities for details.)
    - **Licenses**—Search for inventory items that have licenses of a certain of a certain name or license priority.

4. Select **And** or **Or** from the **Apply Criteria** field.

5. Click **Apply** to filter the inventory to display only those inventory items that meet the selected criteria.

6. To refresh the list to show all inventory items, click **Show All Items**.

# Viewing Security Vulnerabilities for Project Inventory

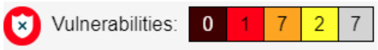
Code Insight uses data from the National Vulnerability Database (NVD), Secunia advisories (as published by the Secunia Research team from Revenera), and other advisories such as RubySec to report security vulnerabilities associated with your inventory items. The vulnerabilities information from these sources is used to create vulnerability rankings and alerts.

Use this procedure to access details for the vulnerabilities associated with an inventory item on the **Project Inventory** tab.

***Task*** ***To view security vulnerabilities for an inventory item, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Click a published inventory item from the **Inventory Items** list. The Project Inventory Details Pane on the right opens to the **Inventory Details** tab.

   If known security vulnerabilities exist for the inventory item, the **Vulnerabilities** graph is displayed:

   The severity levels depicted in the graph differ depending on the version of CVSS Code Insight is using (see Security Vulnerabilities Associated with Inventory). This example shows vulnerability severity counts using CVSS v3.x.
3. Click any of the counts in the graph to open the **Security Vulnerabilities** window, which lists current security vulnerabilities for the inventory item.

   ***Note*** ▪ *Suppressed vulnerabilities are neither reflected in the counts on **Vulnerabilities** bar graph nor are they visible on **Securities Vulnerabilities** window.*

   For more information about how to use this dialog to obtain details about the vulnerabilities, see Working with Security Vulnerabilities.
4. When you have finished viewing the reported vulnerabilities, click **OK** to return to the **Inventory Items** list.

# Viewing Details About the Licenses Associated with Project Inventory

You can view more details about the licenses associated with the OSS or third-party component on which the current inventory item is based. This information is pulled from the Code data library and is displayed on the **License Details** window, which includes the following:

- A **General Information** tab that lists details such as the name of the license, its family, and the license priority assigned by Code Insight.
- A **License Text** tab that displays the complete license text (representing the external forge license text).

While the forge license text is what is likely to be used by the component, the scan might have found actual license text associated with this component in your codebase that can be different from the forge version. To view the exact license text, if any, that the scan discovered and to prepare the final license text, when necessary, to include in the Notices report for distribution to customers, navigate to the **Notices Text** tab. Refer to Finalizing the Notices Text for the Notices Report for more information.

The following procedure describes how to access the **License Details** window from the **Component Details** tab on the **Project Inventory** tab.

***Note** ▪ This window is also accessible when create or edit an inventory item or perform a Component Lookup from the **Project Inventory** tab.*

***Task*** ***To view details for the inventory license, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Select a published inventory item from the **Inventory Items** list.
3. From the Project Inventory Details Pane on the right, do either:
    - For a component-based inventory item, click the **Component Details** tab. Then click information icon (ⓘ) next to the **Selected License** value (the license currently associated with the component) or the **Possible Licenses** (other valid license candidates with which you could associate the inventory item).
    - For a License Only inventory item, click the **License Details** tab, and then click the information icon (ⓘ) next to the license name.

    The **License Details** window appears with the **General Information** tab in focus. For descriptions of these fields on this tab, see License Details Window. Also see License Priority for background on how the license priority is used.
4. Select the **License Text** tab to view the license text.
5. When you have finished examining the license details, click **Close**.

# Viewing and Updating Notes and Guidance

The **Notes & Guidance** tab can provide notes about the automated and manual analysis performed on the codebase as it relates to the current inventory item. The tab can also include guidance on how to remediate issues associated with your product’s use of the OSS or third-party software identified by the inventory item.

***Task*** ***To view notes and guidance, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Select an inventory item from list.
3. From the Project Inventory Details Pane on the right, select the **Notes & Guidance** tab.

4. Review or update content in the following fields as needed. All information is editable except for the information in the **Detection Notes** field:
    - **Detection Notes**—Information generated during the scan to explain the means by which OSS or third-party component was detected in the codebase. This information is not editable. For more details, see Viewing and Updating Detection and Auditing Notes in the Analysis Workbench.
    - **Audit Notes**—Information recorded about the analysis of the code associated with the component in your codebase. For example, these notes might indicate that the inventory item for the component needed to be manually created based codebase evidence that was not detected in scan.
    - **Usage Guidance**—Two kinds of Information: 1) Information propagated from policies that rejected or approved the inventory during the automatic review process that occurred when the inventory was published. This content can explain why the item was rejected or provide requirements and recommendations for using those items that were approved. You cannot edit this content. 2) Reviewers' own notes and concerns about the use of the component in your product software. This information is editable.
    - **Remediation Notes**—A description of items to be addressed or actions to be taken before the use of this software in your product is acceptable from a legal or security standpoint.
5. Click **Save** in any field in which you have made changes.
6. When you have finished with this tab, navigate to another tab for the inventory item, or select another inventory item.

# Viewing Usage Information for Project Inventory

Inventory usage information is important because it aids users in determining how closely to monitor inventory items for intellectual property (IP) and security risk and in taking appropriate action to approve or reject inventory, create tasks for remediation, and issue alerts and notifications. Usage fields can determine whether the item should be included in Third-Party Notices and what steps need to be taken to satisfy license obligations and conditions of use. They can also help to identify license conflicts and compatibility issues.

***Task*** ***To view inventory usage information, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Click an inventory item from the **Inventory Items** list.
3. From the Project Inventory Details Pane on the right, select the **Usage** tab in the inventory details. For details about the inventory usage fields and how they are used, see Inventory Usage Information.

# Viewing Associated Files

Associated files are files in your codebase that have been automatically or manually associated with the inventory item selected in the **Inventory Items** pane due to evidence found in the files.

*Task* ***To view associated files, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Click an inventory item from the **Inventory Items** list.
3. From the Project Inventory Details Pane on the right, select the **Associated Files** tab in the inventory details to view the list of files associated with the selected inventory item. Each file entry shows the following:
    - **Alias**—The unique, user-defined name provided during scanner setup (for a Scan Server or a remote scan agent) to represent its scan-root path containing the codebase in which the file is located. The alias provides a name that is more meaningful than the scan-root path.
    - **File Path**—The file's path relative to the scan-root path on instance hosting the scanner.

      If you have Analyst permissions, the path is hyperlinked to open to the file's **File Details** tab in the **Analysis Workbench**, where you can view file evidence. If necessary, while in the **Analysis Workbench**, you can also add or remove files associated with the inventory. If you do not have Analyst permissions, the path remains in plain text.
4. When you have finished viewing associated files, select another tab or click another item listed in the **Inventory Items** pane.

# Creating Inventory from the Project Inventory Tab

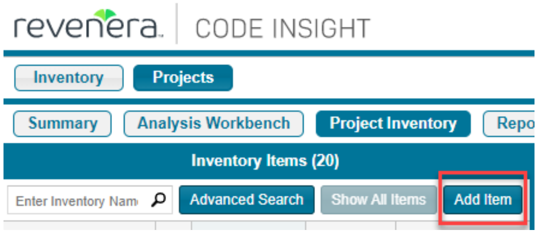
Reviewers can create an inventory item to represent any third-party code or artifact that is not automatically detected by the system.

Use the following steps to create an inventory item from the **Project Inventory** tab as needed. Note the following:

- When you save the inventory item, it is automatically published.
- No files can be associated with an inventory item when it is created from the **Project Inventory** tab.
- If you register a new component instance (a unique component-version-license combination) when creating inventory, the registered instance becomes available for selection across the system.
- Inventory of type **Work in Progress**, **Component**, or **License Only** can be created.

*Task* ***To create an inventory item from the Project Inventory tab, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Click **Add Item** at the top of the **Inventory Items** list.



The **New Inventory** dialog opens.

3. For the **Name** field, perform the appropriate step, based on the inventory type you intend to select for the **Type** field (see the next step):

    - For inventory of the type **Work in Progress**, specify a name for the inventory item. Best practice is to provide a name in the following conventional syntax used by Code Insight, even if the elements represented in the name are not available in the data library:

      ```
      <COMPONENT_NAME> <VERSION> (LICENSE_NAME)
      ```

    - For inventory of the type **Component** or **License only**, leave the **Name** field blank. The field will be automatically populated based on the registered component or license instance.

4. From the **Type** dropdown, select the type of inventory item you want to create and perform the related step or steps:

    - **Work in Progress**—Create this type of inventory item if you want to quickly represent third-party code or an artifact without having to select an associated component, version, or license from the data library. (You can later edit this inventory item to convert it to one of the other inventory types.) This option is typically used if you need a placeholder or cannot find the associated element in the data library. Items of type **Work in Progress** are not affected by policies and do not receive vulnerability updates or alerts.

    - **Component**—Create this type of inventory item if you know the component, version, and license for the third-party code or artifact *and* you either are able to locate it in the Code Insight data library using the Component Lookup feature or you need to create it as a custom component because it is not in the library. This type of inventory is associated with a registered component instance—that is, a unique component-version-license combination—and is affected by policies and receives vulnerability updates and alerts.

      The Component Lookup feature, made available when you select the **Component** type, enables you to associate the inventory item with an existing registered component instance (or a new instance that you create) or to create the custom component and instance to associate with the item.

      The following are basic steps for using Component Lookup. For more details, see Searching Components.

      a. Click the **Lookup Component** button to locate the component of interest (as described in the next steps) or to create a custom component and instance to associate with the inventory item (see Creating and Editing Custom Components for continued steps).

      b. In the list of results, navigate to the appropriate component,  and click **Show Versions** to display the list of registered instances for that component.

      c. Click **Use This Instance** next to an existing registered instance to associate that instance with the inventory item.

         or

         Click **Register New Instance** to create a new instance. Complete the registration by selecting an existing component version (or choosing the **Create Custom Version** value to specify a new version) and then selecting the license to associate with the instance. Click **Use This Instance** next to the new instance to associate it with the inventory item.

         The **Name** and **Description** editable fields for the inventory item are automatically populated with information based on the registered instance you selected. Additionally, the **Component** and **License** fields are displayed, showing the component, its version, and the license for the instance.

- **License Only**—Create this type of inventory item if you know the license for the third-party code or artifact but do not know the component. (You can later edit this inventory item to convert it to one of the other inventory types.) This type of inventory is typically used for groups of files of unknown origin that are governed by a specific license. The inventory is affected by policies.

  Simply select the appropriate license from **License Only** dropdown, which is enabled when you select this type.

  The **Name** field for the inventory item is automatically populated with the name **Files under <LICENSE_NAME> License**, where <LICENSE_NAME> is license you selected.

  A **License Details** tab is added, enabling to view details about the license selected for the inventory item.

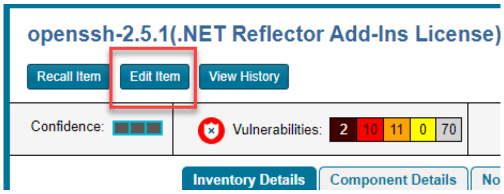
5. Update the remaining fields if appropriate. For a description of each fields, see Project Inventory Details Pane.
6. Click **Save**. The name of the inventory item is added to the **Inventory Items** list.
7. (Optional) If you created a License Only inventory item, view details about the license selected for the new inventory item on the **Licenses Details** tab in the right pane.

# Editing Inventory from the Project Inventory Tab

Use the following steps to edit an inventory item from the **Project Inventory** tab for a project as needed.

*Task* ***To edit an inventory item from the Project Inventory tab, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. In the **Inventory Items** list, select the inventory item that you want to edit. Information about the inventory item is displayed in the right pane.
3. In the header on the right pane, click the **Edit Item** button next to the component name.



   The **Edit Inventory** dialog opens.

4. Make changes to the fields as needed. Refer to Project Inventory Details Pane for field descriptions and additional steps required when updating the inventory type. Note the following:
   - For a **Component** inventory item, you can use the Component Lookup feature to select a different registered instance (or create a new one) or to create a custom component and instance to associate with the inventory item. The **Name**, **Component**, and **License** fields are updated accordingly.
   - You can convert a **Work In Progress** or **License Only** inventory item to a different inventory type, using the **Type** field and performing the additional steps required for the type. The **Name** and other appropriate fields are updated accordingly.

- If you need to update the component version or associated license only, simply click the Edit (✎) icon next to the **Component** or **License** value, and select a different license or version or both. (This avoids performing the longer Component Lookup process to edit these elements.)
- No additional files can be associated with an inventory item from **Project Inventory** tab.

5. Click **Save** to change the changes to the inventory item.

# Approving or Rejecting Inventory Items

The next step in the Code Insight workflow is to have security and legal experts review all published inventory and categorize them as approved or rejected for use in the software project. To approve or reject an inventory item, perform the following steps.

***Task*** ***To approve or reject inventory items, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. In the row for the inventory item you want to approve or reject, click the green checkmark to approve the item or the red X to reject the item.

   A circle appears around the status icon to indicate it has been selected. A circle around the question mark indicates that no status selection has been made (that is, the inventory item requires further review to determine its status).

   Note that, depending on the inventory review and remediation options defined for the project, selecting the **Reject** status can automatically create a Remediate Inventory task. For more information, see Updating Inventory Review and Remediation Settings for a Project.

# Creating and Managing Tasks for Project Inventory

The following sections describe the types of tasks that can be associated with project inventory and how to manage them:

- Task Types
- About External Work Items
- Manually Creating a Task
- Opening the Tasks List
- Editing a Task
- Closing or Reopening a Task Directly from the Tasks List
- Effects of Closing Manual Review Tasks on Inventory Status

## Task Types

Users with access to the project inventory (and edit privileges) can create and manage one or more tasks for a given inventory item. Tasks can be one of three types:

- **Manual Review Inventory**—A task to track the manual review of an inventory item, typically an inventory item that has not already been auto-reviewed by policy. A **Manual Review Inventory** task alerts the assignee to the need to review the inventory item within its current context. Closing this type of task with an **Approve** or **Reject** resolution can automatically approve or reject the inventory item. See Effects of Closing Manual Review Tasks on Inventory Status for details.
- **Remediate Inventory**—A task to track a remediation effort on the inventory item (typically a rejected item). A remediation task signals to the assignee to perform some action to make the inventory item acceptable for use (for example, to upgrade to a new version due to discovered vulnerabilities or to use a specific license and to comply with license obligations). Closing a remediation task does not automatically change the inventory review status.
- **Miscellaneous**—A task to track any other effort for an inventory item. Closing a **Miscellaneous** task does not automatically change the inventory review status.

Note that a task can also be created automatically in an automated workflow process (along with external work items) based on review and remediation options up for the project, as described in Updating Inventory Review and Remediation Settings for a Project. Users with proper permissions can then manage these and manually created tasks, using the procedures described in this section.

## About External Work Items

If the project is configured to connect to an external ALM (application lifecycle management) system such as Jira, each task can also have one or more associated work items that correspond to issues in the external ALM system. Work items are useful for tracking work that needs to be performed outside of Code Insight. A work item can be created manually using the **Create Work Item** option or automatically based on the current project settings (as described in Updating Inventory Review and Remediation Settings for a Project). You can create work items only if the project is associated to an ALM instance, which, in turn, defines a set of attributes used to connect to the ALM system and to set up and assign issues. The Project Administrator configures one or more global ALM instances; but, once the project is associated with one of these instances, you can customize the instance to address the needs of the project.

See for ALM Settings details about associating a project with an ALM instance. For more information about managing external work items, see Creating and Viewing External Work Items for a Project Inventory Task.

Currently, Code Insight supports the creation of issues on a Jira server only.

## Manually Creating a Task

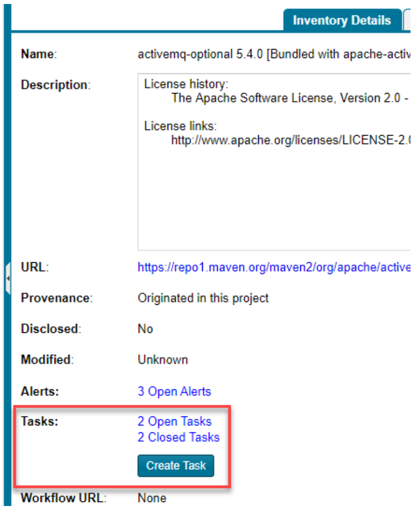
The following procedure describes the manual process for creating a task from the **Inventory Details** tab for a selected inventory item on the **Project Inventory** tab. You can also create a task from the **Tasks** list (see Opening the Tasks List).

***Note*** ▪ *A task can also be created automatically in an automated workflow process (along with external work items) based on review and remediation options up for the project, as described in Updating Inventory Review and Remediation Settings for a Project. Users with proper permissions can then manage these tasks and manually created tasks, using the procedures described in this section.*

***Task*** ***To create a task manually, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Select the inventory item to which you want to add a task. Alternatively, to help you locate the inventory item, click the **Advanced Search** button to open the **Advanced Inventory Search** dialog. From here you can filter inventory items accordingly.

   When you select the specific inventory item, the Project Inventory Details Pane on the right is populated with information about the inventory item. The **Inventory Details** tab in focus.
3. In the **Tasks** section on the tab, click the **Create Task** button to open the **Create Task** dialog.



Name: activemq-optional 5.4.0 [Bundled with apache-activ

Description: License history:
The Apache Software License, Version 2.0 -
License links:
http://www.apache.org/licenses/LICENSE-2.(

URL: https://repo1.maven.org/maven2/org/apache/active

Provenance: Originated in this project

Disclosed: No

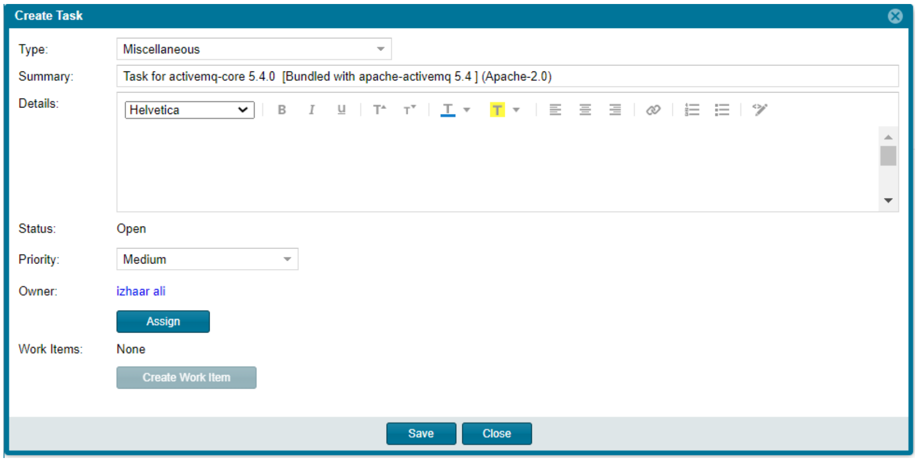
Modified: Unknown

Alerts: 3 Open Alerts

Tasks: 2 Open Tasks
2 Closed Tasks

Workflow URL: None

The **Create Task** window is opened.



Type: Miscellaneous

Summary: Task for activemq-core 5.4.0 [Bundled with apache-activemq 5.4 ] (Apache-2.0)

Details: Helvetica

Status: Open

Priority: Medium

Owner: izhaar ali

Work Items: None

4. Select the type of task you want to create—**Manual Inventory Review**, **Remediate Inventory**, or **Miscellaneous**. (See Task Types.)

5. Complete the following fields as needed:

   - In the **Summary** field, provide a summary or title for the task.
   - In the **Details** field, provide instructions or requirements for completing this task (or provide any information that will be useful to the reviewer).
   - In the **Priority** field, assign a **High**, **Medium**, or **Low** priority to the task.
   - Keep the **Status** as **Open** for a new task.

6. To change the task owner, click the **Assign** button under the **Owner** field, and select a new owner.

   The initial task owner defaults to one of the following contacts, depending on the task type:

   - The Project Contact for **Miscellaneous** tasks
   - The project's Legal Contact for **Manual Inventory Review** tasks
   - The project's Developer Contact for **Remediate Inventory** tasks

   For more information about the these contacts, see Summary Tab.

7. To create an external work item associated with the task, click the **Create Work Item** button. (See Creating and Viewing External Work Items for a Project Inventory Task for details.)

   A "Success" message is displayed if the work item is created successfully on the corresponding ALM system (currently, a Jira server) associated to this project.

   You can repeat this step to create another work task.

8. Click **Save** to create the task. The **Tasks** list opens, showing the task you created.

## Opening the Tasks List

Use the following procedure to open the **Tasks** list, which shows the open and closed tasks associated with the current inventory item. From this list, you create a task or open the **Task Details** dialog for an existing task to edit its status and other task attributes. Alternatively, you can change the status for a task directly from the list.

From the **Tasks** list, you can also click a link to send an email to the owner or creator of a given task or to the user who closed a task.

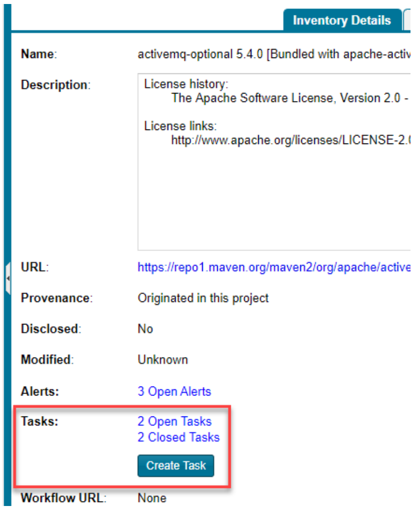
***Task*** ***To open the Tasks list, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).

2. Select the inventory item to which the task you want to edit is associated. Alternatively, click the **Advanced Search** button to open the **Advanced Inventory Search** dialog, where you can select filters that help you pinpoint the inventory item. For example, you can select to filter on those inventory items associated with tasks that are assigned to a specific user or created within a certain date range.

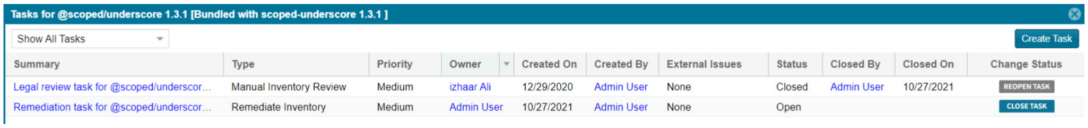
   When you select the specific inventory item, the Project Inventory Details Pane on the right is populated with information about the inventory item. The **Inventory Details** tab in focus.

3. In the **Tasks** section on the tab, click the ***x*** **Open Tasks** or ***x*** **Closed Tasks** link.



| | |
|---|---|
| **Name**: | activemq-optional 5.4.0 [Bundled with apache-activ |
| **Description**: | License history:<br>The Apache Software License, Version 2.0 -<br>License links:<br>http://www.apache.org/licenses/LICENSE-2.( |
| **URL**: | https://repo1.maven.org/maven2/org/apache/active |
| **Provenance**: | Originated in this project |
| **Disclosed**: | No |
| **Modified**: | Unknown |
| **Alerts:** | 3 Open Alerts |
| **Tasks:** | 2 Open Tasks<br>2 Closed Tasks |
| **Workflow URL**: | None |

The **Tasks** list for the inventory item is displayed.



Tasks for @scoped/underscore 1.3.1 [Bundled with scoped-underscore 1.3.1 ]

| Summary | Type | Priority | Owner | Created On | Created By | External Issues | Status | Closed By | Closed On | Change Status |
|---|---|---|---|---|---|---|---|---|---|---|
| Legal review task for @scoped/underscor... | Manual Inventory Review | Medium | izhaar Ali | 12/29/2020 | Admin User | None | Closed | Admin User | 10/27/2021 | REOPEN TASK |
| Remediation task for @scoped/underscor... | Remediate Inventory | Medium | Admin User | 10/27/2021 | Admin User | None | Open | | | CLOSE TASK |

4. If needed, use the search filter at the top of the list to show open, closed, or all tasks.

5. Perform any of the following:

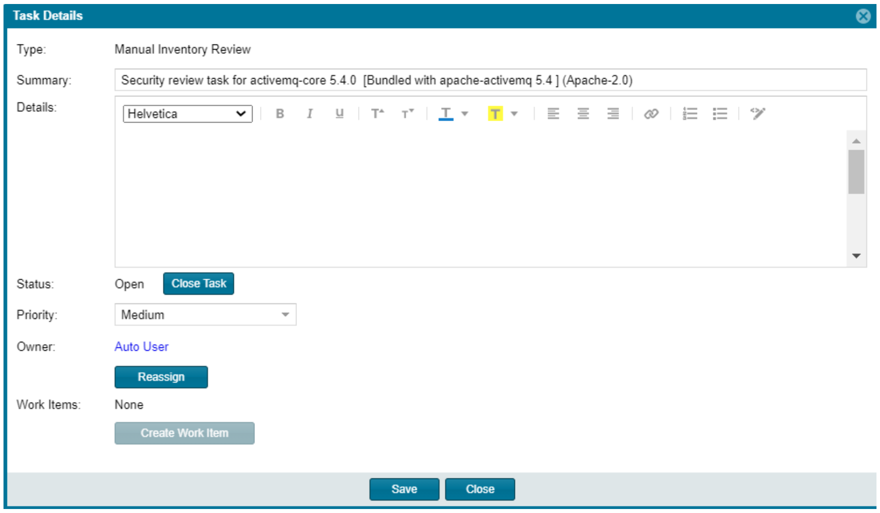
   - Change the status of a task directly from the **Tasks** list without having to open a task. See Closing or Reopening a Task Directly from the Tasks List.
   - Open a task to edit its details, including its status. See Editing a Task.
   - Click **Create Task** to create another task. See Manually Creating a Task for details about completing the fields on the **Create Task** dialog that opens.
   - Send an email to the task **Owner** or **Created By** user by clicking linked name in either column.
   - Click the **Closed By** link to send an email to the user who closed the task. (If the task is open, the **Closed By** and **Closed On** values are blank.)

# Editing a Task

The following describes how to edit and change the open or closed status of a task associated with a given inventory item on the **Project Inventory** tab.

***Task*** ***To edit a task, do the following:***

1. For the selected inventory item on the **Project Inventory** tab, open the **Tasks** list. (See Opening the Tasks List.) The list shows all review tasks associated with the inventory item.
2. Locate the appropriate task from the **Tasks** list, and click the link in the **Summary** column for the task to open the **Task Details** dialog.



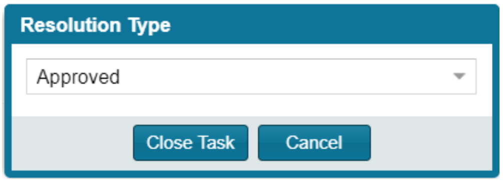
Type: Manual Inventory Review
Summary: Security review task for activemq-core 5.4.0 [Bundled with apache-activemq 5.4 ] (Apache-2.0)
Details:
Status: Open
Priority: Medium
Owner: Auto User
Work Items: None

3. To update the task fields or add an external work item, refer to the descriptions in Manually Creating a Task.
4. To change the status of the task, go to the **Status** section of the window, and do either of the following, depending on the current task status:

   - To *close* an open task, click the **Close Task** button.

     If the task type is **Remediate Inventory** or **Miscellaneous**, the task is immediately closed. Closing either of these tasks types has no effect on the current status of the inventory item. If you need to change the inventory status upon closing the task, do so manually.

     If the task type is **Manual Inventory Review**, select **Approved** or **Rejected** from the **Resolution Type** pop-up to indicate the task resolution. Then click **Close Task** from the pop-up. (To under stand how the resolution affects the status of the inventory item, see Effects of Closing Manual Review Tasks on Inventory Status.)

     Approved

   - To *reopen* a closed task, click the **Reopen** button. This does not affect the current status of the inventory item.

5. Click **Save** to save the updates and return to the **Tasks** list.

If you closed the task, its entry in the **Tasks** list shows your user ID and the date of task closure in the **Closed By** and **Closed On** fields, respectively. If you reopened a task, these values are blank.

## Closing or Reopening a Task Directly from the Tasks List

You can close or reopen a task directly from the **Tasks** list without having to open the task to edit it, as described in the following procedure.

***Task*** ***To close or reopen a task directly from the Tasks list for a given project inventory item, do the following:***

1. For a given inventory item on the **Project Inventory** tab, open the **Tasks** list from the **Inventory Details** tab on the Project Inventory Details Pane on the right. This list shows all review tasks associated with the inventory item. To complete this step, follow the procedure in Opening the Tasks List.
2. In the **Tasks** list, locate the task whose status you want to change. (If necessary, use the search filter at the top of the list to show open, closed, or all tasks.)
3. In the **Change Status** column for the task, click the available status button:
    - **CLOSE TASK**—If the task type is **Remediate Inventory** or **Miscellaneous**, the task is immediately closed. Closing either of these tasks types has no effect on the current status of the inventory item. If you need to change the inventory status based on closing the task, do so manually.

      If the task type is **Manual Inventory Review**, select **Approved** or **Rejected** from the **Resolution Type** pop-up to indicate the task resolution. Then click **Close Task** from the pop-up. (To understand how the selected resolution can automatically change the status of the inventory item, see Effects of Closing Manual Review Tasks on Inventory Status.)

      Once the **Tasks** list is refreshed, it shows your user ID and the date of task closure in the **Closed By** and **Closed On** fields for the task.
    - **REOPEN TASK**—The task is immediately reopened. This does not affect the current status of the inventory item.

      Once the **Tasks** list is refreshed, the **Closed By** and **Closed On** fields for the task show blanks.
4. Repeat these steps to change the status of other tasks from the **Tasks** list.

## Effects of Closing Manual Review Tasks on Inventory Status

When you can close a **Manual Inventory Review** task, you must select an **Approve** or **Reject** resolution which, in turn, has an effect on the status of the inventory item, as follows:

- If the inventory item has only one review task associated with it, the **Approve** or **Reject** status of the task sets the inventory item status to **Approve** or **Reject** accordingly.
- If the inventory item has two or more review tasks associated with it, the **Reject** status of a single review task automatically sets the inventory item status to **Reject**. All review tasks are closed but can be reopened for further investigation.
- If an inventory item has two or more review tasks associated with it and these tasks are a combination of open tasks and tasks with an **Approve** status, the inventory item retains its **Not Reviewed** status.

Note that, depending on the inventory review and remediation options defined for the project, the **Reject** status that is automatically set when you close a **Manual Inventory Review** task can automatically create a **Remediate Inventory** task. For more information about these options, see Updating Inventory Review and Remediation Settings for a Project.

# Creating and Viewing External Work Items for a Project Inventory Task

Users with access to the project inventory (and edit privileges) can create one or more external work items for a task associated with a given inventory item. Each work item in Code Insight contains a corresponding ALM (application lifecycle management) issue on the ALM system (such as Jira) configured for the project.

The following topics are described in this section:

- Prerequisite
- Manually Creating a Work Item
- Viewing a Work Item

## Prerequisite

You can create work items only if you project has been associated with an ALM instance. See for ALM Settings details about defining this association. Currently, Code Insight supports the creation of issues on a Jira server only.

## Manually Creating a Work Item

The following procedure describes the manual process for creating an external work item for a task.

Note that an external work item can also be created automatically in an automated workflow process based on options that you can set up for the project. See Updating Inventory Review and Remediation Settings for a Project for details on editing the automated workflow options and Edit Project: Review and Remediation Settings Tab for field descriptions.

***Task*** ***To create a work item manually, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).
2. Select the inventory item associated with the task to which you want to add a work item. Alternatively, click the **Advanced Search** button to open the **Advanced Inventory Search** dialog, where you can select filters that help you pinpoint the inventory item. For example, you can select to filter on those inventory items associated with tasks that are assigned to a specific user or created within a certain date range.

   When you select the specific inventory item, the Project Inventory Details Pane on the right is populated with information about the inventory item. The **Inventory Details** tab in focus.
3. On the **Inventory Details** tab, click the ***x*** **Open Tasks** link to view the list of open tasks for the inventory item.

4. Locate the appropriate task from the **Tasks** list, and click the link in the **Summary** column to show the **Task Details** dialog.

5. Click the **Create Work Item** button. The **New Work Item** page is displayed.

   ***Note*** ▪ *The **Create Work Item** button is enabled only if the project has been associated with an ALM instance, as described in ALM Settings.*

6. Complete the fields to define the work item. See the inline help for field descriptions.

   This page might already contain default field values based on the project or global application defaults; you can override these values as needed.

7. Click **Create Work Item**.

   A "Success" message is displayed if the work item is created successfully on the corresponding ALM system (currently, a Jira server) associated to this project.

   You are returned to the **Task Details** dialog.

8. Verify that the item was created successfully by clicking the **# Open Work Items** link in the **Work Items** section on the **Task Details** dialog. Then click the **External ID** link for the issue. The link should connect you with the external Jira server and open the issue that corresponds to the work item.

## Viewing a Work Item

An inventory item containing one or more work items displays an information icon in the upper right-hand corner of its **Inventory Details** tab. The **External Issues** field contains links to the open and closed work items.

***Task*** ***To view a work item, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory).

2. Select the inventory item associated with the task containing the work item. The Project Inventory Details Pane on the right is populated with information about the inventory item. The **Inventory Details** tab is in focus.

3. On the tab, click the **# Open Tasks** link. The **Tasks** list is displayed.

4. In the **External Issues** column for the task containing the work item, click either the **# Open Work Items** or **# Closed Work** Items link. The **Work Items** window is displayed.

5. Use the search filter at the top of the window to show **All**, **Open**, or **Closed** work items.

6. Click the **External ID** link for the work item to open the issue in Jira.

# Recalling a Published Inventory Item

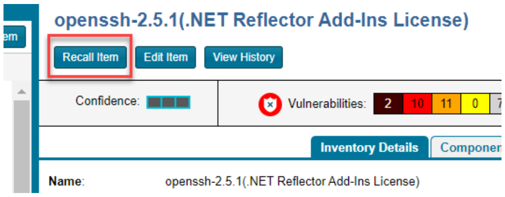
You can recall (remove) a published inventory item from **Inventory Items** list if it does not fit the criteria for inclusion. Recalling the item and publishing it again will affect the publish date on the item as well as the age of the inventory item. A Recall is not required to make edits to the inventory item.

***Task*** ***To recall a published inventory item, do the following:***

1. Open the **Project Inventory** tab for the desired project, and select the inventory item that you want to recall (see Displaying Project Inventory).

   Information about the inventory item is displayed in the right pane.

2. In the pane header, click **Recall Inventory Item**.



openssh-2.5.1(.NET Reflector Add-Ins License)

Name: openssh-2.5.1(.NET Reflector Add-Ins License)

   The item is removed from the **Inventory Items** list.

# Viewing the Update History for an Inventory Item in Project Inventory

From the **Project Inventory** tab, you can view a history of the updates made to a specific inventory item.

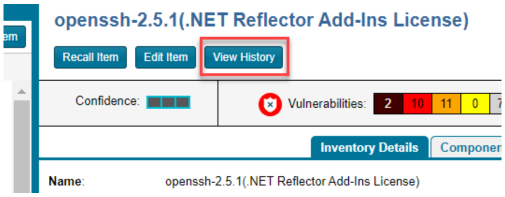
***Note*** ▪ *You have access to this same history from the **Analysis Workbench**. See Viewing the Update History for an Inventory Item in the Analysis Workbench.*

***Task*** ***To view the update history of an inventory item, do the following:***

1. Open the **Project Inventory** tab for the desired project, and select the inventory item whose history you want to view (see Displaying Project Inventory).

   Information about the inventory item is displayed in the right pane.

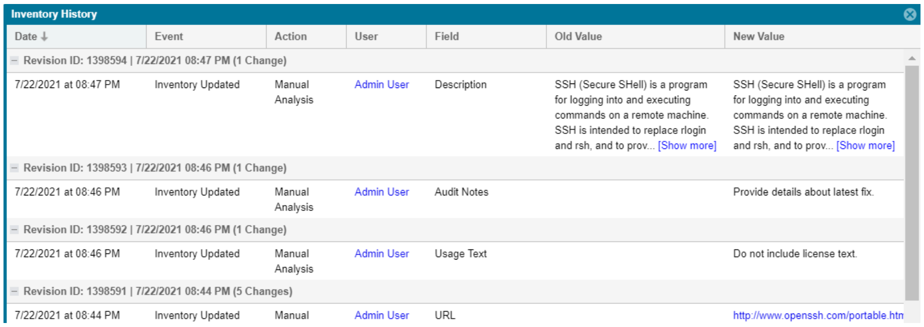
2. In the pane header, click the **View History** button.



openssh-2.5.1(.NET Reflector Add-Ins License)

Name: openssh-2.5.1(.NET Reflector Add-Ins License)

   The **Inventory History** window is opened, showing the list of updates made to the inventory item. By default, the updates are listed in descending order by date so that you see the most recent updates first. Each update record identifies—among other details—the update type, the user who made the update, and the before-and-after values in the update. For a description of all features in this window, see Inventory History Window.



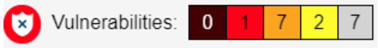
Inventory History

| Date ↓ | Event | Action | User | Field | Old Value | New Value |
|---|---|---|---|---|---|---|
| Revision ID: 1398594 \| 7/22/2021 08:47 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:47 PM | Inventory Updated | Manual Analysis | Admin User | Description | SSH (Secure SHell) is a program for logging into and executing commands on a remote machine. SSH is intended to replace rlogin and rsh, and to prov... [Show more] | SSH (Secure SHell) is a program for logging into and executing commands on a remote machine. SSH is intended to replace rlogin and rsh, and to prov... [Show more] |
| Revision ID: 1398593 \| 7/22/2021 08:46 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:46 PM | Inventory Updated | Manual Analysis | Admin User | Audit Notes | | Provide details about latest fix. |
| Revision ID: 1398592 \| 7/22/2021 08:46 PM (1 Change) | | | | | | |
| 7/22/2021 at 08:46 PM | Inventory Updated | Manual Analysis | Admin User | Usage Text | | Do not include license text. |
| Revision ID: 1398591 \| 7/22/2021 08:44 PM (5 Changes) | | | | | | |
| 7/22/2021 at 08:44 PM | Inventory Updated | Manual | Admin User | URL | | http://www.openssh.com/portable.htm |

# Working with Security Vulnerabilities

Code Insight uses data from the National Vulnerability Database (NVD) and other advisories such as RubySec to report security vulnerabilities associated with your inventory items. The information from these sources is used to create vulnerability rankings and alerts.

The **Vulnerabilities** bar graph shows the current security-vulnerability counts by severity level for a given inventory item:



The graph is shown on the **Analysis Workbench**, **Project Inventory** tab, Inventory View, and Lookup Component Window (for a given component version).

The following sections provide more information about exploring the details for a security vulnerability so that you can better address the vulnerability's impact on your product code and take remedial action if necessary:

- Understanding Severity Levels for Security Vulnerabilities
- Examining Security Vulnerability Details
- Suppressing/Unsuppressing Security Vulnerabilities

# Understanding Severity Levels for Security Vulnerabilities

Code Insight obtains the severity level of a security vulnerability from the advisory database used to identify the vulnerability. The severity is based on the vulnerability's CVSS (Common Vulnerability Scoring System) score, which can have two different values depending on the scoring system used to calculate it—CVSS v2.0 or v3.x. Code Insight supports both systems for displaying the scores and severities of security vulnerabilities. The Code Insight System Administrator determines which scoring system your system uses.

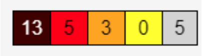
## CVSS v3.x Scoring System

When Code Insight is configured to report security vulnerabilities using the CVSS v3.x scoring system, the color-coded segments in **Vulnerabilities** bar graph represent the following severity levels:

- **Dark brown**—Critical severity (CVSS score 9.0 - 10.0)
- **Red**—High severity (CVSS score7.0 - 8.9)
- **Gold**—Medium severity (CVSS score 4.0 - 6.9)

- **Yellow**—Low severity (CVSS score 0.1 - 3.9)
- **None**—No severity available (N/A)

The following **Vulnerabilities** graph reflects vulnerability counts for an inventory item when CVSS v3.x scoring is used. (The counts are based on vulnerability scores in all CVSS v3 systems supported by Code Insight, currently v3.1 and v3.0. A given vulnerability can have only one v3 score—either a v3.1 or v3.0 score, not both.) This specific graph indicates 13 vulnerabilities of critical severity, 5 of high severity, 3 of medium severity, 0 of low severity, and 5 of unknown severity:

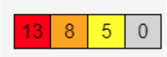


## CVSS v2.0 Scoring System

When Code Insight is configured to use the CVSS v2.0 scoring system, the color-coded segments in graph represent the following severity levels:

- **Red**—High severity (CVSS score 7.0 - 10.0)
- **Gold**—Medium severity (CVSS score 4.0 - 6.9)
- **Yellow**—Low severity (CVSS score 0.1 - 3.9)
- **Gray**—Unknown severity (N/A)

The following **Vulnerabilities** graph reflects vulnerability counts for the same inventory item referenced in the previous section, but in this case CVSS v2.0 scoring is used. Note that the graph shows the same total number of vulnerabilities as the previous graph shows, but the severity distribution is different. In this case, the graph indicates 13 vulnerabilities of high severity, 8 of medium severity, 5 of low severity, and 80 of unknown severity:



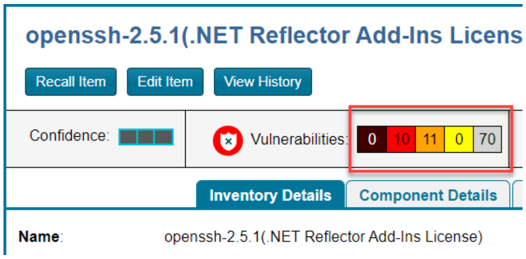
# Examining Security Vulnerability Details

The following procedure explains how to use **Vulnerabilities bar** graph to obtain details about the security vulnerabilities associated with the inventory item.

***Task*** ***To view security vulnerabilities for an inventory item, do the following:***

1. For a specific inventory item in the **Analysis Workbench** or in **Project Inventory** (or for a component version in the **Lookup Component** window accessed within the context of an inventory item), click anywhere on the associated **Vulnerabilities** bar graph.

(The graph is displayed only if vulnerabilities exist for the inventory item or component version.)



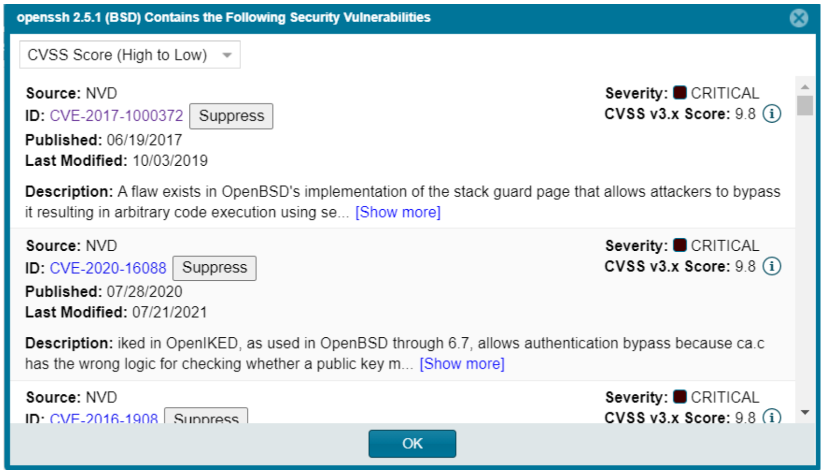
openssh-2.5.1(.NET Reflector Add-Ins Licens

Confidence:

Vulnerabilities: 0 | 10 | 11 | 0 | 70

**Name**: openssh-2.5.1(.NET Reflector Add-Ins License)

The **Security Vulnerabilities** window is displayed. (This example uses the CVSS v3.x scoring system.)



**openssh 2.5.1 (BSD) Contains the Following Security Vulnerabilities**

CVSS Score (High to Low)

**Source:** NVD
**ID:** CVE-2017-1000372
**Published:** 06/19/2017
**Last Modified:** 10/03/2019
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

**Description:** A flaw exists in OpenBSD's implementation of the stack guard page that allows attackers to bypass it resulting in arbitrary code execution using se... [Show more]

**Source:** NVD
**ID:** CVE-2020-16088
**Published:** 07/28/2020
**Last Modified:** 07/21/2021
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

**Description:** iked in OpenIKED, as used in OpenBSD through 6.7, allows authentication bypass because ca.c has the wrong logic for checking whether a public key m... [Show more]

**Source:** NVD
**ID:** CVE-2016-1908
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

***Note* ▪ *When a security vulnerability is suppressed for the component version associated with the current inventory item, the vulnerability is neither reflected in the counts on* Vulnerabilities *bar graph nor is it visible on* Securities Vulnerabilities *window.***

2. Examine the vulnerabilities in the **Security Vulnerabilities** list. For a description of the details shown for each vulnerability entry, see the Description of Security Vulnerability Properties. Note the following general aspects about the list itself:

   - Each entry in the **Security Vulnerabilities** list identifies a specific security vulnerability associated with the selected inventory item (or component version). A vulnerability can be reported by the NVD (National Vulnerability Database) as a CVE (Common Vulnerabilities and Exposures), by Secunia Research, or by another research organization.
   - The list shows only *explicit* CVEs and advisories—that is, only those vulnerabilities that are directly mapped to the component version identified by the inventory item—and lists them in their proper hierarchical position (see the next bulleted item).
   - In the list hierarchy, Secunia and other advisories are at the top level; any CVEs referenced by these advisories are at the secondary level. This structure is in place because advisories are often well-researched and provide additional information above what is provided by the NVD. CVEs that are not referenced by any advisories also appear at the top-level of the hierarchy. The hierarchy view is no more than two levels deep.

- A CVE that is referenced by one or more advisories for the given inventory item is shown in the secondary list under each of the top-level advisory entries with which the CVE is associated. However, the vulnerability itself will count only *once* in places where vulnerability totals are listed—on the project dashboards and **Vulnerabilities** bar graphs in the Web UI, as well as in API responses and reports (Project and Audit).
- All top-level entries (CVEs and advisories) are sorted by CVSS score. Similarly, CVE vulnerabilities in a secondary list under a top-level advisory entry are sorted by CVSS score within the secondary list.
- In some cases, the score of a vulnerability is unknown (and reported as **N/A** in the list), resulting in the severity level of the vulnerability to be reported as **None** or **Unknown**. (For more information about the vulnerability score and severity, see Description of Security Vulnerability Properties.)

  ***Note** ▪ Your feedback is welcome on how Code Insight should handle the severity and scoring of currently unscored vulnerabilities. The Code Insight team will do its best to incorporate the results of this feedback into the Code Insight vulnerability database. Contact Revenera Support with your suggestions.*

3. When you have finished with the **Security Vulnerabilities** window, click **OK** to close the window.

## Description of Security Vulnerability Properties

The following describes the properties shown for each security vulnerability in the **Security Vulnerabilities** list. These properties are not editable.

**Table 2-9** ▪ Description of Security Vulnerability Properties

| Property | Description |
|---|---|
| **Source** | The research system or organization that has reported the security vulnerability (for example, **NVD**, **Secunia**, or another advisory entity). |
| **ID** | The ID of the security vulnerability in the format of the advisory organization that reported it:<br>● For a vulnerability reported by the NVD, the ID uses the CVE (Common Vulnerabilities and Exposures) format.<br>● For a vulnerability reported by Secunia Research, the ID uses the SA (Secunia Advisory) format.<br>● For a vulnerability reported by another research organization, the ID uses the format specific to that organization.<br>Optionally, you can click the hyper-linked CVE ID in an entry to view the vulnerability details found on the NVD or other website:<br> <br>**Source:** NVD<br>**ID:** CVE-2017-1000372<br>Access this link especially if you are conducting a deep research of the vulnerability. The linked site can provide referenced CVEs (those that are not explicitly mapped to the component version but might be indirectly related). |

**Table 2-9 ▪** Description of Security Vulnerability Properties (cont.)

| Property | Description |
|---|---|
| **Published** | The date on which the vulnerability was originally published, as captured from its source (NVD, Secunia, or another advisory). [1] |
| **Last modified** | The date on which the vulnerability was last revised, as captured from its source (NVD, Secunia, or another advisory). If vulnerability was not modified, the field displays the vulnerability's published date. [1] |
| **Severity** | The severity level of the vulnerability (**CRITICAL**, **HIGH**, **LOW**, or other). This level depends on the vulnerability's CVSS score. For details about the relationship between severity levels and CVSS scoring systems, see Understanding Severity Levels for Security Vulnerabilities. |

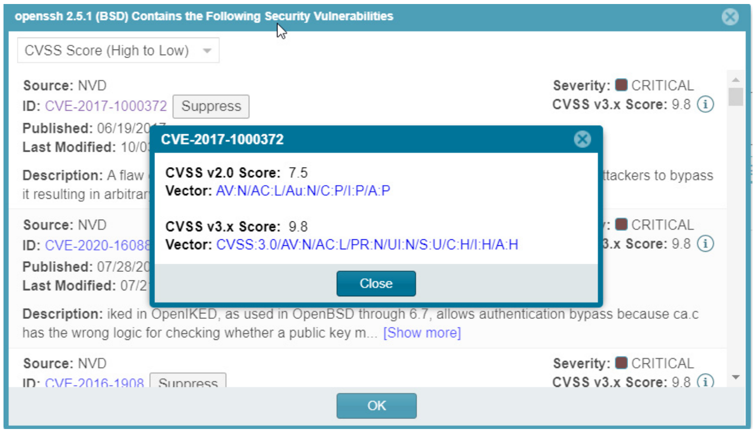
**Table 2-9 ▪** Description of Security Vulnerability Properties (cont.)

<table>
<tr><th>Property</th><th>Description</th></tr>
<tr><td><b>CVSS &lt;version&gt; Score</b></td><td>

The vulnerability’s CVSS (Common Vulnerability Scoring System) score, which can have two different values depending on the scoring system used to calculate it—either CVSS v2.0 or v3.x (specified in the property label). For details about scoring system versions, see Understanding Severity Levels for Security Vulnerabilities.

In some cases, the advisory CVSS score (or other type of vulnerability score) is unknown for a vulnerability because it has not been provided by the supplier. Code Insight reports the score for such a vulnerability as **N/A**.

If you click the ⓘ icon next to the score, the resulting popup lists the v3.x and the v2.0 score for the vulnerability:



openssh 2.5.1 (BSD) Contains the Following Security Vulnerabilities

CVSS Score (High to Low)

**Source:** NVD
**ID:** CVE-2017-1000372 Suppress
**Published:** 06/19/20
**Last Modified:** 10/0
**Description:** A flaw
it resulting in arbitrar

**Severity:** 🟥 CRITICAL
**CVSS v3.x Score:** 9.8 ⓘ

ttackers to bypass

CVE-2017-1000372

**CVSS v2.0 Score:** 7.5
**Vector:** AV:N/AC:L/Au:N/C:P/I:P/A:P

**CVSS v3.x Score:** 9.8
**Vector:** CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Close

**Source:** NVD
**ID:** CVE-2020-16088
**Published:** 07/28/20
**Last Modified:** 07/2

: 🟥 CRITICAL
3.x Score: 9.8 ⓘ

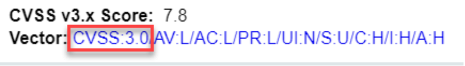
**Description:** iked in OpenIKED, as used in OpenBSD through 6.7, allows authentication bypass because ca.c has the wrong logic for checking whether a public key m... [Show more]

**Source:** NVD
**ID:** CVE-2016-1908 Suppress

**Severity:** 🟥 CRITICAL
**CVSS v3.x Score:** 9.8 ⓘ

OK

The associated **Vector** value for a v3.x vulnerability has the specific score version—3.0 or 3.1—embedded in the value.



**CVSS v3.x Score:** 7.8
**Vector:** CVSS:3.0/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

The **Vector** value is available only if the vulnerability is found in the NVD. (Otherwise, the field shows **N/A**.) This linked value is a compressed textual representation of the values used to derive the score. When you click the link, the appropriate NVD Common Vulnerability Scoring System Calculator is opened, showing you the environmental and temporal factors that determine the score. You can use the calculator to tweak these factors as necessary to calculate another score that is more realistic for your software product. (Instructions are provided with the calculator.) This adjusted score can then be used internally to direct your review and remediation processes.

</td></tr>
<tr><td><b>Description</b></td><td>A description of the vulnerability.</td></tr>
</table>

**Table 2-9 ▪** Description of Security Vulnerability Properties (cont.)

| Property | Description |
|---|---|
| **Suppress** | (Available only to Code Insight System Administrators) Click this button next to a given security vulnerability to suppress—that is, hide—the vulnerability for selected component versions. For more information, see Suppressing/ Unsuppressing Security Vulnerabilities. |

[1] If you have migrated from a pre-2021 R3 Code Insight release to the current release, you must run an Electronic Update to obtain the latest date information.

# Suppressing/Unsuppressing Security Vulnerabilities

For various reasons, your site might want to suppress—that is, hide—a security vulnerability that is associated with one or more component versions used by your inventory. Once suppressed, the vulnerability is no longer published in reports, counted in vulnerability totals for inventory in projects, or automatically associated with inventory during future project scans in your Code Insight instance. For example, you might choose to suppress a vulnerability if you have taken remedial steps to protect your code against the vulnerability or if the vulnerability has proven to be a “false positive” (that is, is associated with an incorrect component version).

Likewise, you might want to unsuppress a security vulnerability that you have previously suppressed so that it is again visible and counted for inventory in projects.

Only a Code Insight System Administrator can suppress security vulnerabilities, as well as monitor currently suppressed vulnerabilities and unsuppress them as needed.

Any type of security vulnerability can be suppressed (and then unsuppressed when necessary):

- Vulnerabilities retrieved from the Code Insight data library during scans or as open alerts
- Custom vulnerabilities

The following sections provide more information about suppressing and unsuppressing security vulnerabilities:

- Suppressing Security Vulnerabilities
- Viewing Suppressed Security Vulnerabilities
- Unsuppressing Security Vulnerabilities

## Suppressing Security Vulnerabilities

The System Administrator can suppress a security vulnerability at the Code Insight system level for one or more (or all) component versions associated with the vulnerability. The following provides more details:

- Effects of Suppressing a Security Vulnerability
- Suppressing a Security Vulnerability

## Effects of Suppressing a Security Vulnerability

The suppression of a security vulnerability has an impact on your Code Insight system. That is, once a vulnerability is suppressed for a specific component version, it is no longer counted in vulnerability totals or is visibly listed at the project, inventory, and component-version levels. The count reduction is evident on the project dashboards and on the **Vulnerabilities** bar graphs in the Web UI, as well as in subsequently generated API responses and reports (Project and Audit). Likewise, the actual vulnerability is no longer visible in the list of vulnerabilities on the **Security Vulnerabilities** window (which is opened when you click a **Vulnerabilities** graph) or in API responses or reports.

***Note*** ▪ *The **Vulnerabilities** graph in the UI is shown on the **Inventory** view, in the **Lookup Component** window for a specific component version, and in **Inventory Details** for a given inventory item (both on the **Project Inventory** tab and in the **Analysis Workbench**).*

The following describes the impact that a security vulnerability suppressed for a specific component version has on other features of Code Insight:

- **Advanced Search on the Project Inventory tab and Inventory View**—When an inventory search is based the vulnerability name or severity, the results do not show inventory items that are related to the suppressed vulnerability.
- **Alerts**—Any open alerts for the suppressed vulnerability are automatically closed, and the open and closed alert counts are adjusted on the **Project Inventory** tab, in the **Analysis Workbench**, and on the **Inventory** view.
- **Policies**—Once a security vulnerability is suppressed, no changes are initially propagated to those review policies that are based on vulnerabilities. However, each time one of these policies is triggered thereafter (that is, when an inventory item is published), the policy ignores the suppressed vulnerability when determining whether to automatically approve or reject the published inventory item.

  Additionally, a change in policy due to the suppression of a vulnerability does not change the existing approval/rejection status of a published inventory item unless the item is manually recalled and then republished.
- **Subsequent scans and rescans**—Once a vulnerability is suppressed, it is not reflected in the results of subsequent rescans and initial scans, whether incremental or full.

## Suppressing a Security Vulnerability

The following procedure is used to suppress a security vulnerability for one or more (or all) versions of an OSS or third-party component associated with your inventory.

***Note*** ▪ *A System Administrator can also suppress a security vulnerability using the **Suppress vulnerability** REST API. For more information about this API, see the Code Insight Swagger documentation, available from the **Help** > **REST API Guide** option on the Code Insight ☰ menu.*

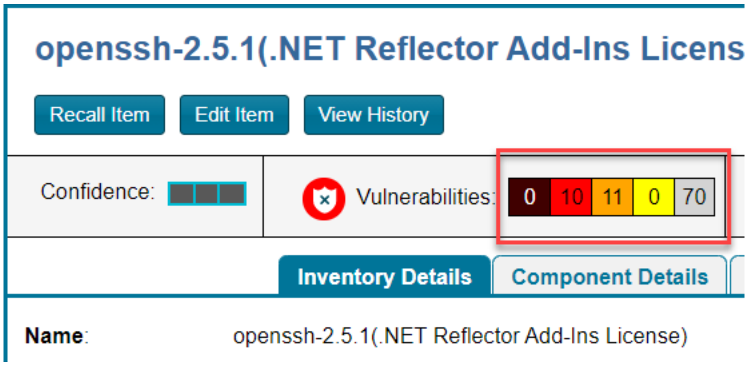
***Task*** ***To suppress a security vulnerability, do the following:***

1. Locate the **Vulnerabilities** bar graph within the context of a given inventory item or component version.

   ***Note*** ▪ *The **Vulnerabilities** graph in the UI is shown on the **Inventory** view, in the **Lookup Component** window for a specific component version, and in **Inventory Details** for a given inventory item (both on the **Project Inventory** tab and in the **Analysis Workbench**).*

   The graph is displayed only if vulnerabilities exist for the inventory item or component version.

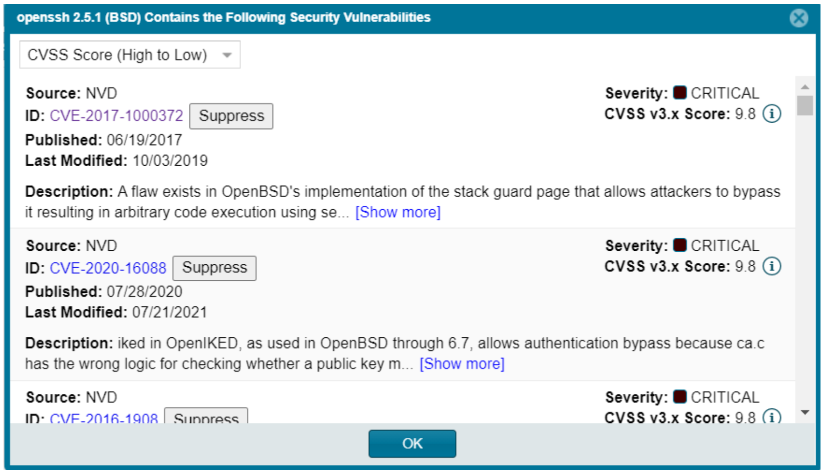
2. Click anywhere on the **Vulnerabilities** bar graph.



openssh-2.5.1(.NET Reflector Add-Ins Licens

**Name**: openssh-2.5.1(.NET Reflector Add-Ins License)

   The **Security Vulnerabilities** window is displayed. (This example uses the CVSS v3.x scoring system.)



openssh 2.5.1 (BSD) Contains the Following Security Vulnerabilities

**Source:** NVD
**ID:** CVE-2017-1000372 Suppress
**Published:** 06/19/2017
**Last Modified:** 10/03/2019
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

**Description:** A flaw exists in OpenBSD's implementation of the stack guard page that allows attackers to bypass it resulting in arbitrary code execution using se... [Show more]

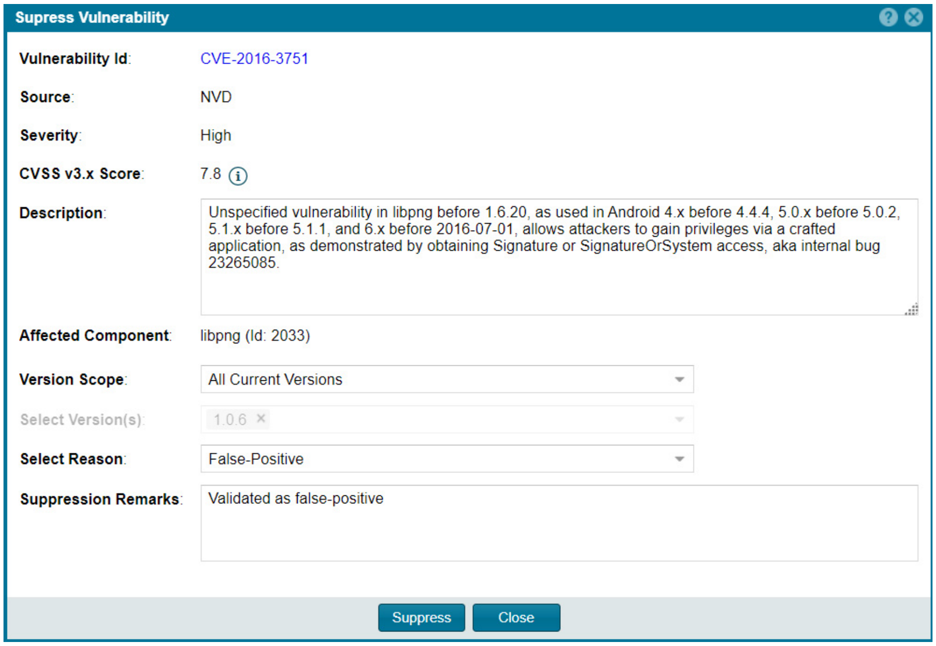
**Source:** NVD
**ID:** CVE-2020-16088 Suppress
**Published:** 07/28/2020
**Last Modified:** 07/21/2021
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

**Description:** iked in OpenIKED, as used in OpenBSD through 6.7, allows authentication bypass because ca.c has the wrong logic for checking whether a public key m... [Show more]

**Source:** NVD
**ID:** CVE-2016-1908 Suppress
**Severity:** CRITICAL
**CVSS v3.x Score:** 9.8

3. Locate the security vulnerability that you want to suppress, and click the **Suppress** button next its **ID** field.

   The **Suppress Vulnerability** window is displayed.



| | |
|---|---|
| Vulnerability Id: | CVE-2016-3751 |
| Source: | NVD |
| Severity: | High |
| CVSS v3.x Score: | 7.8 |
| Description: | Unspecified vulnerability in libpng before 1.6.20, as used in Android 4.x before 4.4.4, 5.0.x before 5.0.2, 5.1.x before 5.1.1, and 6.x before 2016-07-01, allows attackers to gain privileges via a crafted application, as demonstrated by obtaining Signature or SignatureOrSystem access, aka internal bug 23265085. |
| Affected Component: | libpng (Id: 2033) |
| Version Scope: | All Current Versions |
| Select Version(s): | 1.0.6 × |
| Select Reason: | False-Positive |
| Suppression Remarks: | Validated as false-positive |

4. Complete all editable fields on the window to define the vulnerability suppression. For a description of these fields, see Suppress Vulnerability Window.

5. Click **Suppress**. Then click **OK** in the pop-up to acknowledge that the vulnerability has been successfully suppressed for the specified component versions.

   If other vulnerabilities exist for the inventory item, you are returned to the **Security Vulnerabilities** window. The suppressed vulnerability should no longer be visible in the window (if you suppressed the vulnerability for the component version of the current inventory item).

   If this vulnerability is the only one for the inventory item, you are returned to the previous context from which you opened the **Security Vulnerabilities** window (for example, the **Inventory Details** tab or the **Lookup Component** window). The **Vulnerabilities** bar graph count should be reduced in the specific severity segment associated with the suppressed vulnerability (if you suppressed the vulnerability for the component version of the current inventory item).

# Viewing Suppressed Security Vulnerabilities

The following procedure describes how to obtain a view all currently suppressed security vulnerabilities in your Code Insight instance. Only a Code Insight System Administrator can access this view.

***Task*** ***To obtain a view of all currently suppressed security vulnerabilities in Code Insight, do the following:***

1. Click the **Open Menu** icon in the upper right of any Code Insight page:

   ☰

2. Select **Data Library** from the menu.

3. Click the **Suppressed Vulnerabilities** tab to view a list of the currently suppressed security vulnerabilities in Code Insight. (This tab is visible to only Code Insight System Administrators.) From this tab, you can do the following:

    - For each suppressed vulnerability, easily review the OSS or third-party component with which the vulnerability is associated and the specific versions of that component for which the vulnerability is currently suppressed. For a description of this tab, see Suppressed Vulnerabilities Tab.
    - View a pop-up that shows details about a given vulnerability by clicking the **Information** icon next to the ID of the suppressed vulnerability in the **Vulnerability Id** column.
    - View a pop-up window that shows the vulnerability-suppression details of every component version for which a given vulnerability is suppressed. To open the pop-up, click the **Information** icon next to the versions listed for the vulnerability in the **Affected Versions** column. The details shown for each of the component versions listed on the pop-up include the user who suppressed the vulnerability for the version, the date and time of the suppression, the reason for the suppression, and additional remarks.
    - To unsuppress a given vulnerability, click its associated **Unsuppress** button. For further details, see Unsuppressing Security Vulnerabilities

# Unsuppressing Security Vulnerabilities

The System Administrator can unsuppress a security vulnerability for one, some, or all of the component versions for which it was previously suppressed. The following sections provide more details:

- Effects of Unsuppressing a Security Vulnerability
- Unsuppressing a Security Vulnerability

## Effects of Unsuppressing a Security Vulnerability

When you unsuppress a security vulnerability, the effects of the vulnerability's previous suppression are reversed. That is, once a vulnerability is unsuppressed for a specific component version, it is now counted in vulnerability totals or is visibly listed at the project, inventory, and component-version levels. The count increase is evident on the project dashboards and on the **Vulnerabilities** bar graphs in the Web UI, as well as in subsequently generated API responses and reports (Project and Audit). Likewise, the actual vulnerability is now visible in the list of vulnerabilities on the **Security Vulnerabilities** window (which is opened when you click a **Vulnerabilities** graph) or in API responses or reports.

***Note** ▪ The **Vulnerabilities** graph in the UI is shown on the **Inventory** view, in the **Lookup Component** window for a specific component version, and in **Inventory Details** for a given inventory item (both on the **Project Inventory** tab and in the **Analysis Workbench**).*

The following describes the impact that unsuppressing a security vulnerability has on other features of Code Insight:

- **Advanced Search on the Project Inventory tab and Inventory View**—When an inventory search is based the vulnerability name or severity, the results now list any inventory items that are associated the unsuppressed vulnerability.

- **Alerts**—Any alerts that were automatically closed due to the previous vulnerability suppression are automatically reopened. Open and closed alert counts are adjusted to reflect the changes on the **Project Inventory** tab, in the **Analysis Workbench**, and on the **Inventory** view.
- **Policies**—Once a security vulnerability is unsuppressed, no changes are initially propagated to those review policies that are based on vulnerabilities. However, each time one of these policies is triggered thereafter (that is, when an inventory item is published), the policy will now consider the vulnerability when determining whether to automatically approve or reject the published inventory item.

  Additionally, a change in policy due to the unsuppression of a vulnerability does not change the existing approval/rejection status of a published inventory item unless the item is manually recalled and then republished.
- **Subsequent scans and rescans**—Once a vulnerability is unsuppressed, it is reflected in the results of subsequent rescans and initial scans, whether incremental or full.

## Unsuppressing a Security Vulnerability

The following procedure is used to unsuppress a security vulnerability for one, some, or all of component versions for which it was previously suppressed.

Only a Code Insight System Administrator can perform this operation.

***Note*** *▪ A System Administrator can also unsuppress a security vulnerability using the* ***UnSuppress vulnerability*** *REST API. For more information about this API, see the Code Insight Swagger documentation, available from the* ***Help*** *>* ***REST API Guide*** *option on the Code Insight menu.*

***Task*** ***To unsuppress a security vulnerability, do the following:***

1. Open the **Suppressed Vulnerabilities** tab using the steps described in Viewing Suppressed Security Vulnerabilities.
2. In the list of suppressed vulnerabilities (in grid format), locate the vulnerability that you want to unsuppress, and click its associated **Unsuppress** button.

   The **Unsuppress Vulnerability** window is displayed.
3. Complete all editable fields on the window to define the unsuppression process for the vulnerability. For a description of these fields, see Unsuppress Vulnerability Window.
4. Click **Unsuppress**. Then click **OK** in the pop-up to acknowledge that the vulnerability has been successfully unsuppressed for the specified component versions.

   You are returned to the **Suppressed Vulnerabilities** tab. The list of suppressed vulnerabilities on the tab is now modified in one of two ways:

   - If the vulnerability was unsuppressed for one or some of the component versions for which it was previously suppressed, those versions are no longer listed for the vulnerability. The remaining suppressed versions are still listed for the vulnerability.
   - If the vulnerability was unsuppressed for all of the component versions for which it was previously suppressed, the vulnerability is longer shown in the list.

# Managing Security Vulnerability Alerts

Code Insight provides the ability to view and clear security vulnerability alerts. When the Electronic Update process is run, it will generate these alerts for any new security vulnerabilities that impact inventory. The alerts allow you to investigate the most recent vulnerabilities and their effect on your project code, if any. Once you have addressed vulnerability impact, either by determining no impact exists or through remediation, you can close the alert.

***Note*** ▪ *An alert can be automatically closed when its associated security vulnerability is manually suppressed by a Code Insight System Administrator. See Suppressing/Unsuppressing Security Vulnerabilities for more information.*

When the Electronic Update generates security vulnerability alerts, an email notification is sent to the Project Contact of each project containing inventory impacted by the alerts. Additionally, users can view the alerts for a given project from the **Inventory Details** pane in the **Analysis Workbench** or from the **Project Inventory** tab (and from the **Inventory** view).

Refer to these topics for more information:

- Accessing Security Vulnerability Alerts
- Using the Alerts Dialog to Manage Alerts

# Accessing Security Vulnerability Alerts

The following methods provide access to the **Alerts** dialog, which allows you to view and manage the security vulnerability alerts impacting inventory in a given project:

- From Email Notifications
- From the Analysis Workbench
- From Project Inventory

## From Email Notifications

Project Contacts can be alerted via email to any projects and inventory items that contain new security vulnerabilities so that they can be reviewed and acted upon if necessary. For email alerts to be sent, the email server must be enabled and configured. For more information, see “Configuring an Email Server” in the *Installation & Configuration Guide*.

Vulnerability alert emails are sent as part of the Electronic Update. A vulnerability alert is generated for each new security vulnerability mapped to a published inventory item. While viewing the alert email, click any of the hyperlinked text in the email to open Code Insight or an advisory web site to view additional information about the alert.

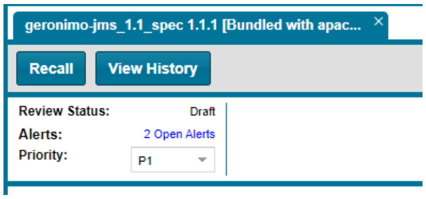
## From the Analysis Workbench

This procedure describes how to access security vulnerability alerts from the **Analysis Workbench**.

*Task* ***To view security vulnerability alerts from Project Inventory, do the following:***

1. Open the **Analysis Workbench** for the desired project. (For instructions, see Opening the Analysis Workbench.)
2. From the **Inventory Item** pane on the right, click the inventory item for which you want to check for alerts. The **Inventory Details** tab for the selected item is opened.

   If open alerts exist, the **Alerts** field provides a link to view them. If no alerts exist, the field shows **None**.

   Review Status: Draft
   Alerts: 2 Open Alerts
   Priority: P1

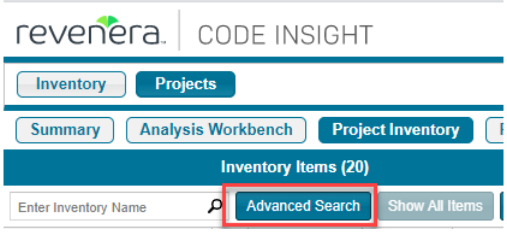
3. Click the link to open the **Alerts** dialog, where you can view the open (and closed) alerts for the inventory item.

## From Project Inventory

This procedure describes how to access the security vulnerability alerts for a specific inventory item from the **Project Inventory** tab for a project.

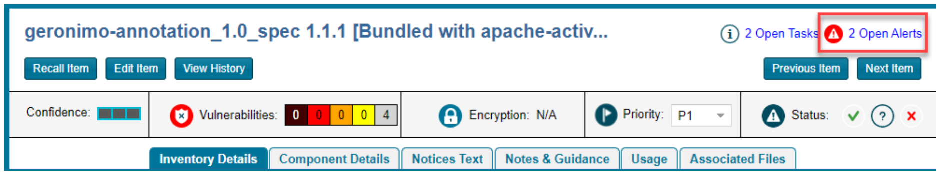
*Task* ***To view security vulnerability alerts from Project Inventory, do the following:***

1. Open the **Project Inventory** tab for the desired project (see Displaying Project Inventory). The **Inventory Items** list is displayed in the left pane.
2. (Optional) To filter the **Inventory Items** list to show only inventory that have alerts, click **Advanced Search**, select the **Inventory with Open Alerts** option, and click **Apply**.

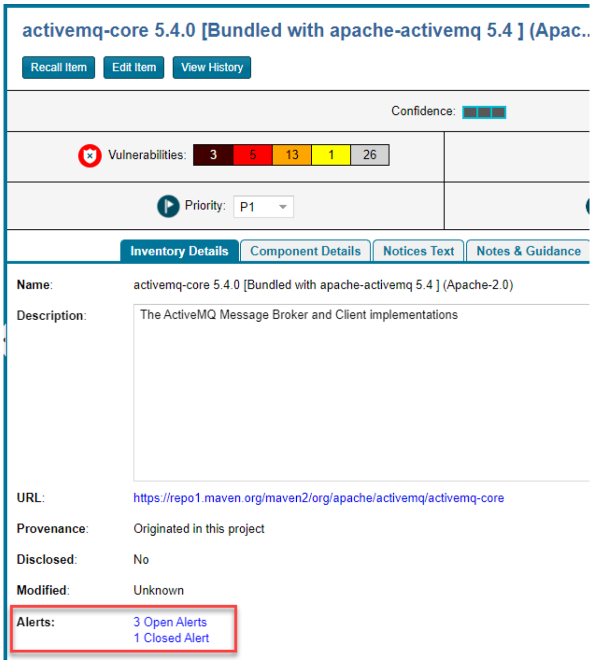
   

3. From the **Inventory Items** list, click the inventory item whose alerts you want to check. Information about the selected inventory item is displayed in the right pane.

   - For a quick check on any *open* alerts, locate for the ⚠ icon in the header of this page.



- To view open or closed alerts, open the **Inventory Details** tab in the right pane. If alerts exist, the **Alerts** field shows separate links to view open or closed alerts, as appropriate.



**Name**: activemq-core 5.4.0 [Bundled with apache-activemq 5.4 ] (Apache-2.0)
**Description**: The ActiveMQ Message Broker and Client implementations
**URL**: https://repo1.maven.org/maven2/org/apache/activemq/activemq-core
**Provenance**: Originated in this project
**Disclosed**: No
**Modified**: Unknown
**Alerts:** 3 Open Alerts
1 Closed Alert

4. Click the associated link to open the **Alerts** dialog, where you can view details about the alerts.

# Using the Alerts Dialog to Manage Alerts

The **Alerts** dialog shows the list of current alerts for a given inventory item in a project. The following describes how to use this dialog to manage security vulnerability alerts for an inventory item:

- Alert Details
- Changing the Priority of an Alert
- Changing the Status of an Alert

# Alert Details

The following columns describe each alert entry listed in the **Alert** dialog. To filter the list of alerts, see Filtering Alerts.

**Table 2-10 ▪** Alert Details

| Column | Description |
|---|---|
| **Type** | The alert type. Currently, only **New Vulnerability** alerts are available. |
| **Date** | The date that the alert was generated. |
| **Priority** | The priority of the alert, which, by default, is based on the official severity level of the security vulnerability associated with the alert, as described here:<br>● **High**—The default when the vulnerability severity level is Critical, High, or None in the CVSS v3.x scoring system or, in the CVSS v2.0 scoring system, High or Unknown.<br>● **Medium**—The default when the severity level is Medium in either scoring system.<br>● **Low**—The default when the severity level is Low in either scoring system.<br>You can change this value as needed. See Changing the Priority of an Alert.<br>For more information about the severity levels for security vulnerabilities, see Viewing Security Vulnerabilities for Project Inventory. |
| **Status** | The status of the alert in Code Insight:<br>● **Open**—The alert needs to be addressed.<br>● **Closed**—The alert has been addressed (usually because remediation was performed or it was determined to be a false positive for your code or the security vulnerability was suppressed). The ⓘ icon is shown to identify who closed the alert and when.<br>Change this value as needed. See Changing the Status of an Alert. |

**Table 2-10 ▪** Alert Details (cont.)

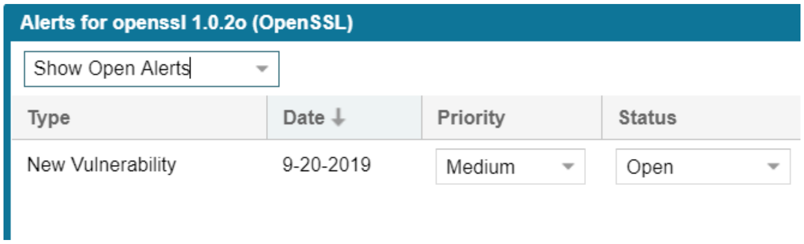
| Column | Description |
|---|---|
| **Details** | Details about the security vulnerability:<br>● **Source**—The advisory database in which the Code Insight located the vulnerability, such as **NVD** (National Vulnerability Database), **Secunia** (Secunia Advisories), **Debian Advisories**, or others.<br>● **ID**—The identification number of the vulnerability associated with the Common Vulnerabilities and Exposures (CVE). If this is a hyperlinked value, click it to go to the actual entry for the vulnerability on the advisory website.<br>● **CVSS Score**—The score of the vulnerability based on the Common Vulnerability Scoring System (CVSS). The values of the CVSS score range from 0.1 to 10, with 10 being the most serious. If the vulnerability has no score, the value is **N/A**.<br>● **Description**—The description of the vulnerability as found in the advisory database.<br>Also see Security Vulnerabilities Associated with Inventory. |

## Filtering Alerts

Use the following procedure to change the view of the list of alerts on the **Alerts** dialog.

***Task*** ***To filter the list of alerts, do the following:***

1. Locate the dropdown at the top of the **Alerts** dialog:



Alerts for openssl 1.0.2o (OpenSSL)

Show Open Alerts

| Type | Date | Priority | Status |
|---|---|---|---|
| New Vulnerability | 9-20-2019 | Medium | Open |

2. Select one of the following options from the dropdown:

    - **Show Open Alerts**—Display only open alerts.
    - **Show Closed Alerts**—Display only closed alerts.
    - **Show All Alerts**—Display both closed and open alerts. This option will only be available if more than one alert is available.

## Changing the Priority of an Alert

Use the following procedure to change the **High**, **Medium**, or **Low** priority of an alert. The priority indicates the urgency with which the security vulnerability associated with the alert needs to be addressed. The initial priority value defaults to the severity of the security vulnerability itself, but you can change this priority based on your site's needs. For more information about vulnerability priority, see Alert Details.

***Note*** ▪ *If the Code Insight System Administrator has switched Code Insight from CVSS v2.0 to CVSS v3.x scoring or vice versa, you might notice a change in the* ***Severity*** *and* ***CVSS Score*** *for the vulnerability associated with the alert. However, the alert* ***Priority*** *should not change from its value current at the time of the switch.*

***Task*** ***To change the priority of a security vulnerability alert, do the following:***

1. Open the **Alerts** dialog for a given inventory item in a project, as described in Accessing Security Vulnerability Alerts.
2. In the **Priority** column, select the new priority.

## Changing the Status of an Alert

Use the following procedure to change the **Open** or **Closed** status of a security vulnerability alert. Usually, you close an alert because the associated security vulnerability has been addressed in your product through remediation or the alert is a false positive. You might need to reopen an alert because further remediation is required.

For more information about the **Open** and **Closed** statuses, see Alert Details.

***Task*** ***To change the status of a security vulnerability alert, do the following:***

1. Open the **Alerts** dialog for a given inventory item in a project, as described in Accessing Security Vulnerability Alerts.
2. In the **Status** column for a given alert, select either option from the dropdown:
    - **Open** to reactivate the alert.
    - **Closed** to indicate that the alert has been addressed.

# Creating and Editing Custom Components

Code Insight enables you to create custom components that represent OSS or third-party software not found in the Code Insight data library or that represent commercial software that you want to track as part of your Bill of Materials. A custom component is currently created or edited within the context of the inventory item with which it is associated, but is saved to the data library and made available for global use.

Once the custom component is created, an inventory item can be associated with a registered instance of the component—that is, a unique component-version-license combination that you define. The custom component is also available for use by policies and is included in the Notices report.

The following topics describe how to create and edit custom components:

- Creating a Custom Component
- Editing a Custom Component
- Custom Component Properties
- Supported Forge-URL Domains for Custom-Component Creation

# Creating a Custom Component

Use the following steps to create a custom component:

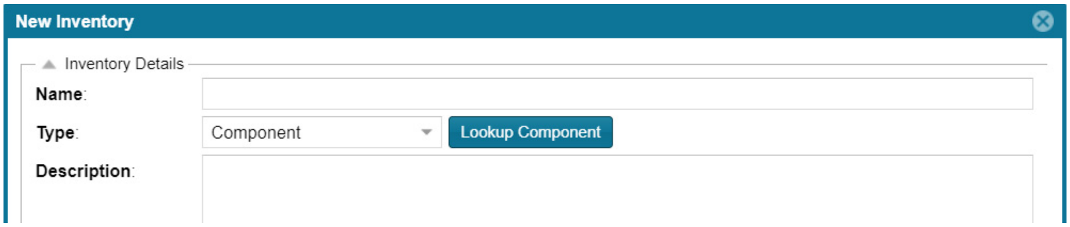
- Step 1: Access the Component Lookup Feature
- Step 2: Create the Custom Component
- Step 3: Associate an Instance of the Custom Component with the Inventory Item

## Step 1: Access the Component Lookup Feature

You create a custom component within the context of Component Lookup feature. Follow these steps to access that feature.

*Task* ***To access the Component Lookup feature, do the following:***

1. From the page on which you are creating or editing inventory in the **Analysis Workbench** or from the **Project Inventory** tab, select **Component** in the **Type** field for the inventory item.



New Inventory

Inventory Details

Name:

Type: Component

Description:

2. Click **Lookup Component** to open the **Lookup Component** window.

## Step 2: Create the Custom Component

Based on the information you have about the custom component you want to create, use one of the following methods to create the component.

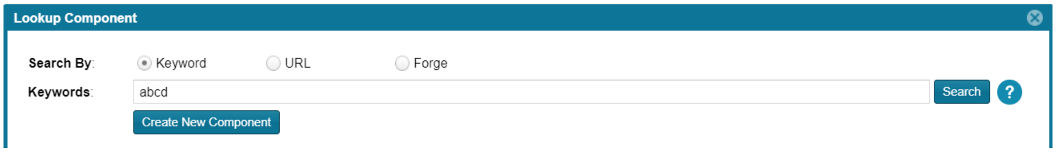
- Create the Custom Component Based on a Keyword in Its Name or Title
- Create the Custom Component Based on Its Project or Forge URL
- Create the Custom Component Based on Its Forge
- Create the Custom Component in Free Form

## Create the Custom Component Based on a Keyword in Its Name or Title

Use the following method to create the custom component based on a keyword in the name or title you intend to give the component.

***Task*** ***To create a custom component based on a keyword in its name or title:***

1. After you have performed Step 1: Access the Component Lookup Feature, select the **Keyword** option on the **Lookup Component** window.
2. In the **Keywords** field, enter a keyword used in the name of the component you are creating.

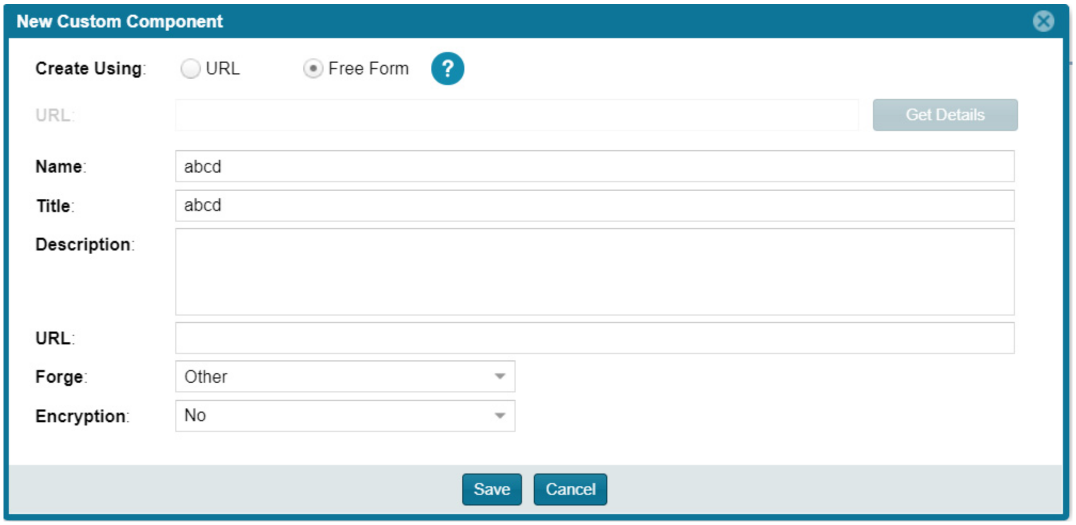


Search By: ● Keyword ○ URL ○ Forge
Keywords: abcd

3. (Optional) Click **Search** to see the list of existing components whose names contain the keyword. If the component you intend to create already exists, you can select an instance of the already existing component to associate with the inventory item, or you can continue to create a custom component with different name and title (continue with the next step).

   Keep in mind that the name and title of a component you create must be unique in the Code Insight data library.

4. Click **Create New Component** to open the **New Custom Component** window, showing the **Name** and **Title** fields automatically populated with the keyword you entered.



Create Using: ○ URL ● Free Form
URL:
Name: abcd
Title: abcd
Description:
URL:
Forge: Other
Encryption: No

   Note that window opens in the Free Form format, enabling you to add missing values and edit pre-populated values for the component as needed.

5. Update the component fields as necessary, noting that the **Name**, **Title**, and **URL** fields are required. (Click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by the **Forge** value you select.) If you do not know the URL, enter **NA**. For more information about these fields, see Custom Component Properties.

6. Click **Save**.

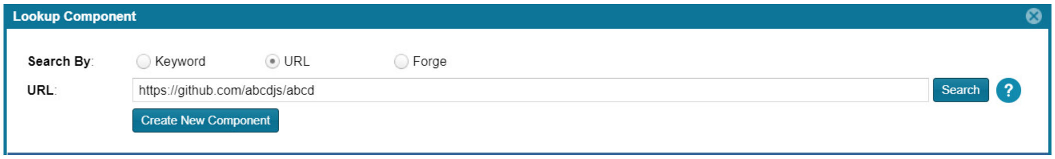
    - If the component information has been properly entered, the component is saved to the Code Insight data library and listed in the **Lookup Component** window. Continue with Step 3: Associate an Instance of the Custom Component with the Inventory Item.
    - If the component name and title combination already exists in the data library, an error message is displayed. You can either select the already-existing component from the **Lookup Component** window to associate it with the inventory item or edit the custom component details to provide a different name or title.

## Create the Custom Component Based on Its Project or Forge URL

Use the following method to create the custom component based on its known project or forge URL.

***Task*** ***To create a custom component based on its URL:***

1. On the **Lookup Component** window, select the **URL** option.
2. In the **URL** field, enter URL for the component. As you enter the URL, it is checked for an acceptable format (such as `http://example.com`), but not for validity.



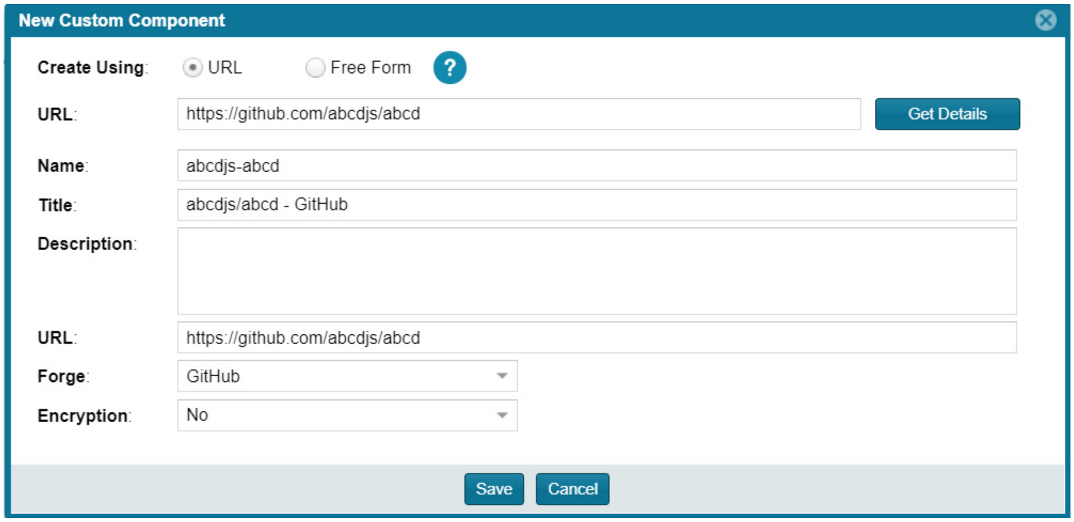
Lookup Component

Search By: ○ Keyword ● URL ○ Forge

URL: https://github.com/abcdjs/abcd

3. (Optional) Click **Search** to see the list of existing components that use the entered URL. If the component you intend to create already exists in the Code Insight data library, you can select an instance of the already-existing component to associate with the inventory item; or you can continue to create a custom component with a different name and title (continue with the next step)
4. Click **Create New Component**.

    - If Code Insight recognizes the URL you entered as belonging to one of the forge-URL domains currently supported for custom component creation, the **New Custom Component** window opens, showing component fields—including **Name**, **Title**, **URL**, and **Forge**—automatically populated with values based on domain conventions. (For the list of supported domains, see Supported Forge-URL Domains for Custom-Component Creation.)



Create Using: ● URL ○ Free Form

URL: https://github.com/abcdjs/abcd

Name: abcdjs-abcd

Title: abcdjs/abcd - GitHub

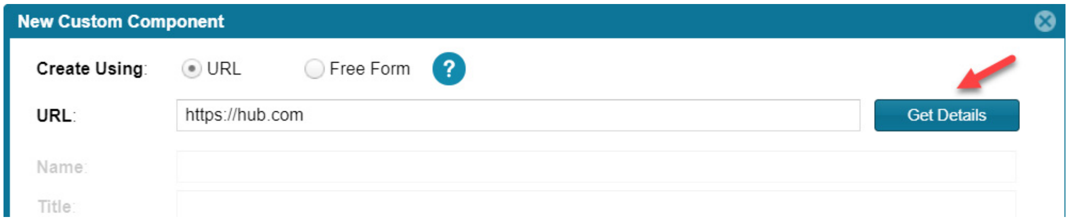
Description:

URL: https://github.com/abcdjs/abcd

Forge: GitHub

Encryption: No

- If Code Insight does not recognize the URL as belonging to a supported domain (that is, it is unable to parse the URL), the **New Custom Component** window opens showing the URL only. You must click **Get Details** to complete the component fields manually in **Free Form** mode. (Click **OK** on the "Unable to parse URL..." message box to proceed to Free Form mode.)



Create Using: ● URL ○ Free Form

URL: https://hub.com

Name:

Title:

5. Update the component fields as necessary, noting that the **Name**, **Title**, and **URL** fields are required. For more information about these fields, see Custom Component Properties.

    - If the URL you initially entered (at the top of the window) belongs to a supported domain and you edit that URL, click **Get Details** to update the remaining field values according to forge-URL domain conventions.

    - If you must manually provide field values because the URL you initially entered does not belong to a supported domain, click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by the **Forge** value you select.

6. Click **Save**.

    - If the required information has been properly entered, the component is saved to the Code Insight data library and listed in **Lookup Component** window. Continue with Step 3: Associate an Instance of the Custom Component with the Inventory Item.

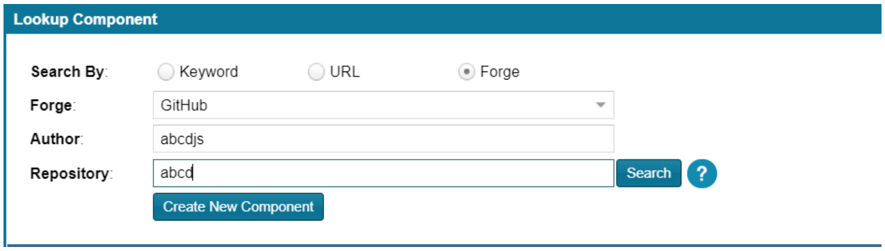
    - If the component name and title combination already exists in the data library, an error message is displayed. You can either select the already-existing component from the **Lookup Component** window to associate it with the inventory item or edit the custom component details to provide a different name or title.

## Create the Custom Component Based on Its Forge

Use the following method to create the custom component if you know the forge (that is, the third-party project repository) of the custom component you are creating.

**_Task_** **_To create a custom component based on its forge:_**

1. On the **Lookup Component** window, select the **Forge** option.
2. From the **Forge** field, select the forge of the custom component, and enter the information required to identify the forge. (This information varies by forge.)



Lookup Component

Search By: ○ Keyword ○ URL ● Forge

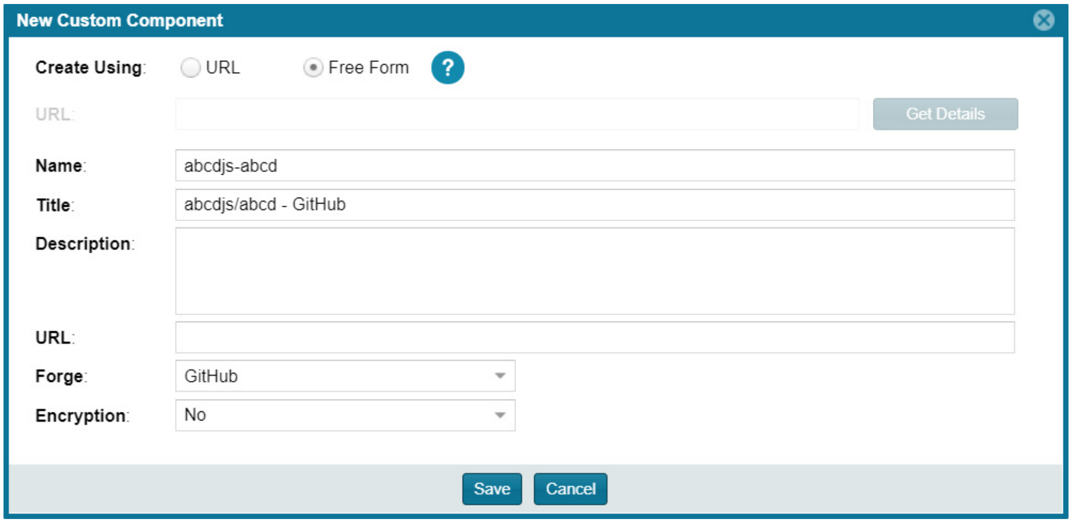
Forge: GitHub

Author: abcdjs

Repository: abcd

3. (Optional) Click **Search** to see the list of existing components that use the entered forge specifications. If the component you intend to create already exists in the Code Insight data library, you can select an instance of the already-existing component to associate with the inventory item; or you can continue to create an custom component with a different name and title (continue with the next step).
4. Click **Create New Component**.

- If the forge you entered is one of the forge-URL domains currently supported for custom component creation, the **New Custom Component** window opens, showing component fields—including **Name**, **Title**, and **Forge**—automatically populated with values based on domain conventions. (For the list of supported domains, see Supported Forge-URL Domains for Custom-Component Creation.)



New Custom Component

Create Using: ○ URL ● Free Form

URL:

Name: abcdjs-abcd

Title: abcdjs/abcd - GitHub

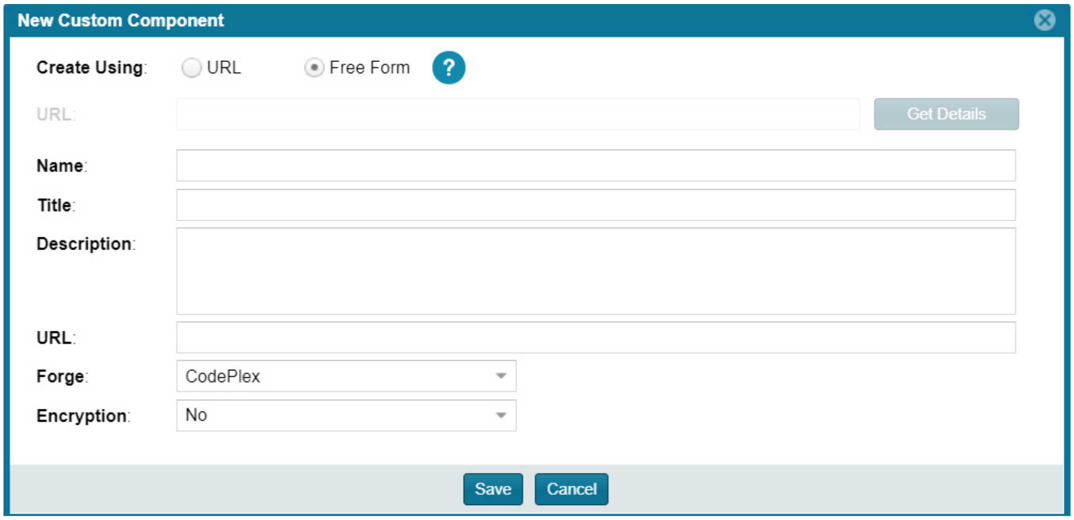
Description:

URL:

Forge: GitHub

Encryption: No

- If the forge you entered is not one of the supported domains, the **New Custom Component** window opens, showing the **Forge** field automatically populated with the forge type you selected.



Create Using: ○ URL ● Free Form ?

URL:

Name:

Title:

Description:

URL:

Forge: CodePlex

Encryption: No

Note that **New Custom Component** window opens in the **Free Form** mode, enabling you to add missing values and edit pre-populated values for the component as needed.

5. Update the component fields as necessary, noting that the **Name**, **Title**, and **URL** fields are required. (Click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by the **Forge** value you selected.) If you do not know the URL, enter **NA**. For more information about these fields, see Custom Component Properties.

6. Click **Save**.

    - If the component information has been properly entered, the component is saved to the Code Insight data library and listed in **Lookup Component** window. Continue with Step 3: Associate an Instance of the Custom Component with the Inventory Item.

    - If the component name and title combination already exists in the data library, an error message is displayed. You can either select the already-existing component from the **Lookup Component** window to associate it with the inventory item or edit the custom component details to provide a different name or title.

## Create the Custom Component in Free Form

Use following method to create the custom component when you do not know the component name (keyword), URL, or forge or when you simply want to provide your own values (for example, when creating a custom component for commercial software you want to track in Code Insight for inclusion in the Bill of Materials).

***Task*** ***To create a custom component in Free Form mode:***

1. On the **Lookup Component** window, click **Create New Component** to open the **New Custom Component** window.

    The window opens in the **Free Form** form, enabling you to provide your own values for the component fields. (No field values are automatically populated.)

Optionally, you can switch to the **URL** form on the **New Custom Component** window, enabling you to enter a project or forge URL by which to create the component. Once you enter the URL, you must click **Get Details** to continue:

- If the URL belongs to a supported forge-URL domain, component fields are automatically populated with values based on domain conventions, as described in Create the Custom Component Based on Its Project or Forge URL.
- If the URL does not belong to a supported domain (that is, Code Insight is unable to parse the URL), you must click **OK** on the resulting “Unable to parse URL…” message box to proceed to manually complete component fields in **Free Form** mode (continue with the next step 2).

2. Update the component fields, noting that the **Name**, **Title**, and **URL** fields are required. (Click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by the **Forge** value you select.) If you do not know the URL, enter **NA**. For more information about these fields, see Custom Component Properties.
3. Click **Save**.
   - If the component information has been properly entered, the component is saved to the Code Insight data library and listed in **Lookup Component** window. Continue with Step 3: Associate an Instance of the Custom Component with the Inventory Item.
   - If the component name and title combination already exists in the data library, an error message is displayed. You can either select the already-existing component from the **Lookup Component** window to associate it with the inventory item or edit the custom component details to provide a different name or title.

## Step 3: Associate an Instance of the Custom Component with the Inventory Item

A component instance is a unique component-version-license entity that you can associate with an inventory item. The following procedure creates an instance for the new custom component and associates it with the inventory item you are creating or editing.

You can create multiple instances of the custom component to associate with inventory items. Each instance is saved to the Code Insight data library and made available for global use.

*Task* ***To associate an instance of the new custom component with the inventory item you are creating or editing, follow these steps:***

1. On the **Lookup Component** window showing the custom component you just created, click **Show Instances**.
2. Click **Register New Instance** to create a component instance.
3. Complete the instance registration by selecting **Create Custom Version** to specify a version and then selecting the license to associate with the instance.
4. Click **Use This Instance** next to the new instance to associate it with the inventory item.

You are returned to the inventory item you are creating or updating. The **Name** and **Description** editable fields for the inventory item are automatically populated with information based on the registered instance you selected. Additionally, the **Component** and **License** fields are displayed, showing the component, its version, and the license for the instance.

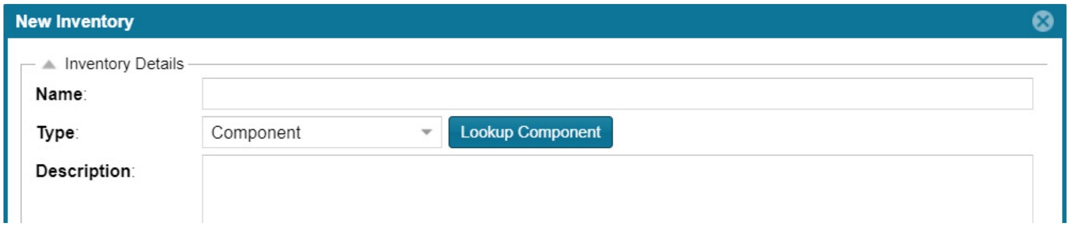
You can now proceed with completing the inventory creation or update.

# Editing a Custom Component

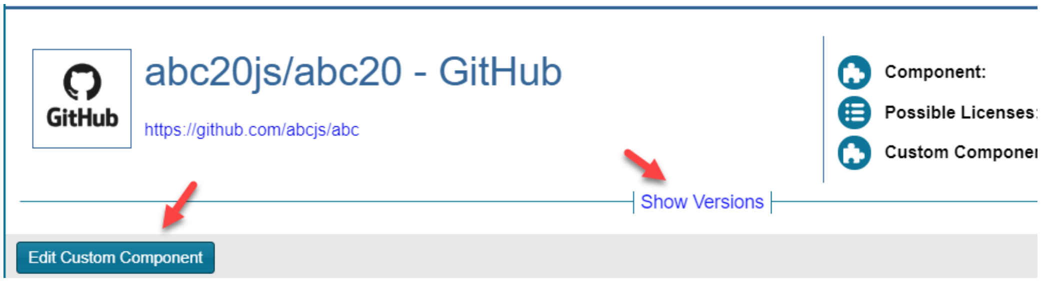
Use the following procedure to edit a custom component. Currently you must edit the component within the context of a inventory item to which it is associated. However, the changes you make are saved in the Code Insight data library and the updated component is available for global use.

***Task*** ***To edit a custom component, do the following:***

1. From the page on which you are creating or editing inventory, select **Component** in the **Type** field for the inventory item.



2. Click **Lookup Component** to open the **Lookup Component** window.
3. Search for the custom component by keyword, URL, or forge.
4. Locate the component in the search results, and update the component in any of these ways:
    - Click **Edit Custom Component** to update the component properties. See Custom Component Properties for field descriptions.
    - Click **Show Versions** to create one or more component-version-license instances for the component.



5. Click **Use This Instance** next to an instance to associate it with the inventory item or click **Cancel** to close the **Lookup Component** window.

You can now proceed with completing the inventory creation or update.

# Custom Component Properties

The following describes the fields on the **New Custom Component** window used to define a custom component. Certain fields might be automatically populated based on information entered on the Lookup Component window. However, any field can be edited.

**Table 2-11 ▪** Fields to Define a Custom Component

| Component Field | Description |
|---|---|
| **Create Using** | (Available during component creation) The "form" mode used to create the custom component. The information you entered on the **Lookup Component** window initially determines the mode used, but you can switch modes here.<br>● **URL**—Mode used when creating the component based on the URL for its project or forge (see Create the Custom Component Based on Its Project or Forge URL). This form includes the actual **URL** value, located beneath the **Create Using** section, which is either pre-populated from the **Component Lookup** window or manually entered here. If necessary, click **Get Details** to view component fields to complete or update them.<br>If the URL is recognized as belonging to a supported forge-URL domain supported for custom component creation, required component fields are automatically populated on the **Component Lookup** window. See the **URL** field description next in this table for more information.<br>● **Free Form**—Mode used to define or update component fields manually when creating the custom component using any other method (see Create the Custom Component Based on a Keyword in Its Name or Title, Create the Custom Component Based on Its Forge, Create the Custom Component in Free Form). Certain fields might be automatically populated based on the information entered previously on the **Lookup Component** window. |
| **URL** | (Available under **Create Using** during component creation when the **URL** form is selected) The project or forge URL for which you are creating the custom component.<br>If the URL is recognized as belonging to a supported forge-URL domain supported for custom component creation, required component fields are automatically populated according to domain conventions. Any change made to this **URL** is automatically updated to the other field values when you click **Get Details**.<br>If the URL does not belong to a supported domain, you must manually provide all remaining necessary component details. (Click **Get details** to display the fields.)<br>For information about supported forge-URL domains, see Supported Forge-URL Domains for Custom-Component Creation. |

**Table 2-11 ▪** Fields to Define a Custom Component (cont.)

| Component Field | Description |
|---|---|
| **Name** | (Required) The name of the custom component. During component creation, this field might be automatically populated based on information entered on the **Lookup Component** window.<br>To help you provide this value in an appropriate format, click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by different forge-URL domains. |
| **Title** | (Required) The component title. During component creation, this field might be automatically populated based on information entered on the **Lookup Component** window.<br>To help you provide this value in an appropriate format, click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by different forge-URL domains. |
| **Description** | A description of the custom component to provide any additional meaningful information about the component. |
| **URL** | (Required) The URL for the project or forge of the custom component. During component creation, this field is pre-populated with the same **URL** value provided in the URL form (see the **Create Using** field) when that URL is recognized as belonging to a supported forge-URL domain.<br>Otherwise, to help you provide this value in an appropriate format, click ? in the upper part of the window for examples of the standard name, title, and URL conventions used by different forge-URL domains.<br>If you do not know the URL for the component, enter **NA**. |
| **Forge** | The third-party project repository used by the custom component. During component creation, this field is automatically populated with an appropriate value when you are creating the component based on either of the following:<br>● Its forge (see Create the Custom Component Based on Its Forge).<br>● The URL for its project or forge *and* the URL belongs to a forge-URL domain supported by the custom-component creation process. For more information, see Create the Custom Component Based on Its Project or Forge URL and Supported Forge-URL Domains for Custom-Component Creation.<br>Otherwise, the default **Other** is displayed. However, you can select any other forge from the dropdown. |
| **Encryption** | **Yes** or **No** value depicting whether this component supports encryption. The default is **No**. |

## Supported Forge-URL Domains for Custom-Component Creation

When creating a custom component, you can select any forge supported by Code Insight and provide free-form component details. However, the forge might require that the URL and other component details be in a certain format. To help the user properly format component properties, the component-creation process supports certain forge-URL domains. When the user initiates the creation process by providing a URL or forge that the creation process recognizes as belonging to a supported domain, it automatically populates component fields with values formatted according to domain conventions.

The following are the forge-URL domains currently supported by the custom-component creation process:

- NuGet Gallery
- npm
- SourceForge
- RubyGems

For other forges that you might use to create custom components, you can click ? on the **Lookup Component** and **New Custom Component** windows for guidance on how to format the URL and the component name and title according to forge conventions.

# Creating and Editing Custom Licenses

Code Insight enables you to create custom licenses that represent licenses not found in the Code Insight data library or commercial EULAs that are typically not included in the data library. The opportunity to create or edit a custom license is made available during those processes prompting you to select a license to associate with an inventory item.

The custom licenses are saved to the data library so that they are available for use by other inventory items across the system. They are also available for use by policies.

The following topics describe how to create and edit custom licenses:

- Creating a Custom License
- Editing a Custom License
- Custom License Properties

## Creating a Custom License

Use the following steps to create a custom license:

- Step 1: Initiate the Creation of a Custom License
- Step 2: Create the Custom License

# Step 1: Initiate the Creation of a Custom License

You have the option to create a custom license during any of the following procedures that involve associating a license with an inventory item:

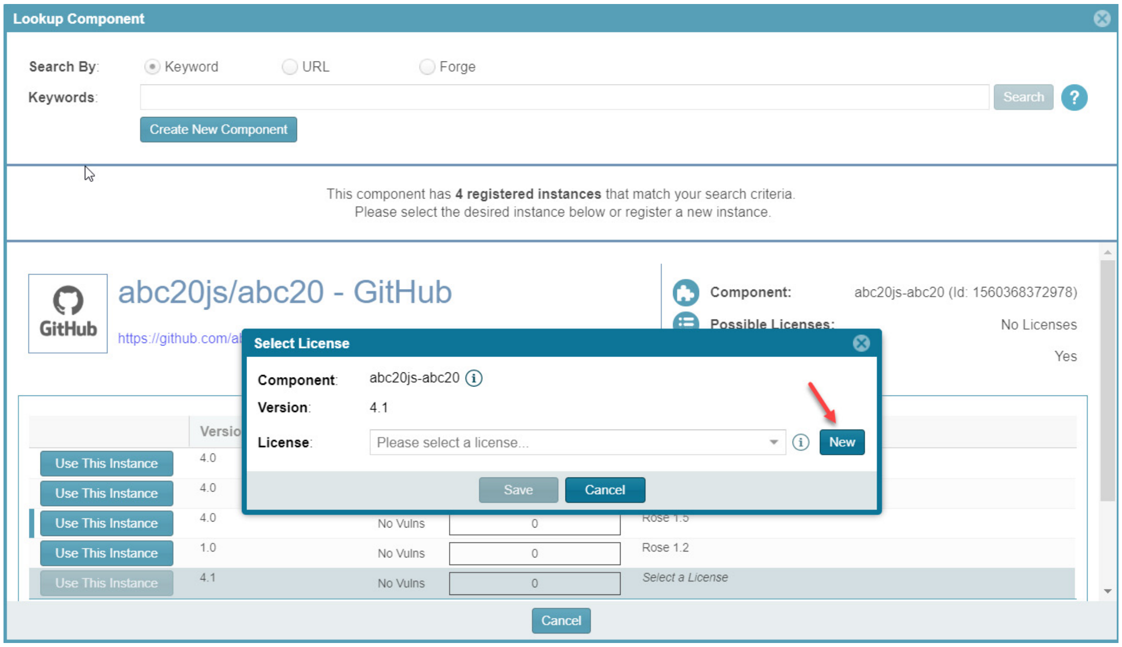
- When Using Component Lookup to Register a Component Instance for a “component” Inventory Item
- When Selecting a New License for an Existing Inventory Item
- When Creating or Editing a “License Only” Inventory Item
- When Creating or Updating a License Policy

## When Using Component Lookup to Register a Component Instance for a “component” Inventory Item

You can create a custom license when you access the Component Lookup feature to register a new component-version-license instance for a “component” inventory item.

*Task* ***To create a custom license when registering a component instance for a “component” inventory item, do the following:***

1. For an inventory item that you are creating or editing in the **Analysis Workbench** or from the **Project Inventory** tab, access the **Lookup Component** window (see Performing Component Lookup).
2. Locate the component for which you want to register a new component-version-license instance to associate with the inventory item, and click **Show Instances** (or **Versions**) to list its registered instances.
3. Click **Register New Instance** to set up a new instance with which to associate the custom license you are creating.
4. After selecting a component version in the new instance entry, click the **Select a License** dropdown and choose **Select Another License**. The **Select License** dialog is displayed.



Search By: ● Keyword ○ URL ○ Forge

Keywords:

This component has **4 registered instances** that match your search criteria.
Please select the desired instance below or register a new instance.

abc20js/abc20 - GitHub

https://github.com/al

Component: abc20js-abc20 (Id: 1560368372978)

Possible Licenses: No Licenses

Yes

Select License

Component: abc20js-abc20

Version: 4.1

License: Please select a license...

| Versio |  |  |  |
|---|---|---|---|
| 4.0 |  |  |  |
| 4.0 |  |  |  |
| 4.0 | No Vulns | 0 | Rose 1.5 |
| 1.0 | No Vulns | 0 | Rose 1.2 |
| 4.1 | No Vulns | 0 | *Select a License* |

5. Next to the **License** dropdown, click **New** to display the **Create Custom License** window.

6. Continue with Step 2: Create the Custom License.

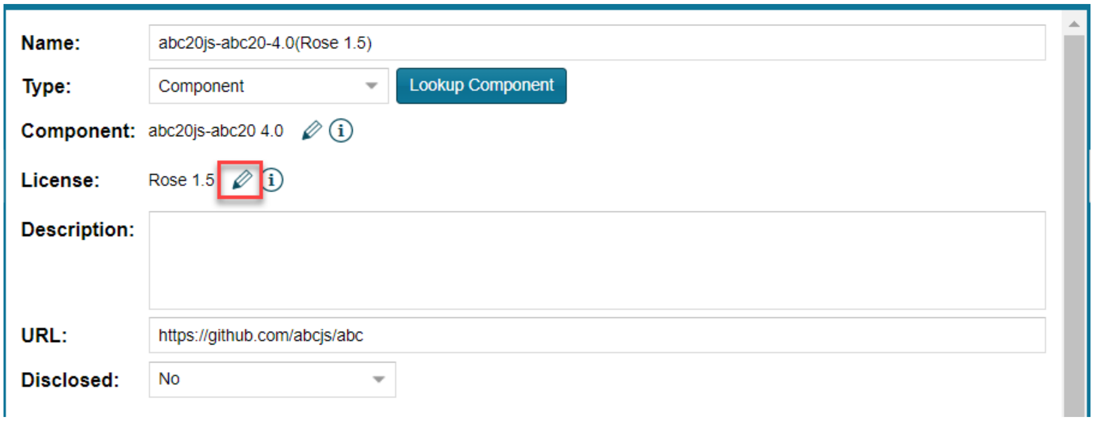
Once you save the custom license, it is automatically associated with the newly registered instance, which you can then associate with the inventory item.

## When Selecting a New License for an Existing Inventory Item

You can create a custom license when updating an existing inventory item of the type **Component** in the **Analysis Workbench** or from the Project Inventory tab.

***Task*** ***To create a custom license with which to associate an existing "Component" inventory item, do the following:***

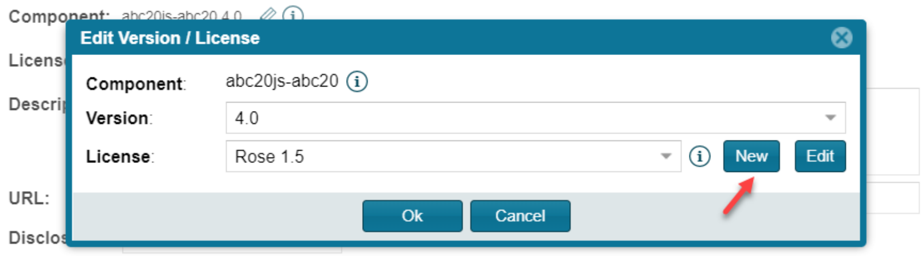
1. When editing an existing inventory item of the type **Component** in the **Analysis Workbench** or from the **Project Inventory** tab, click ✎ next to the **License** or **Component** field.



| | |
|---|---|
| **Name:** | abc20js-abc20-4.0(Rose 1.5) |
| **Type:** | Component |
| **Component:** | abc20js-abc20 4.0 |
| **License:** | Rose 1.5 |
| **Description:** | |
| **URL:** | https://github.com/abcjs/abc |
| **Disclosed:** | No |

The **Edit Version/License** dialog is displayed.

2. Next to the **License** dropdown, click **New** to display the **Create Custom License** window.



Component: abc20js-abc20 4.0

Edit Version / License

Component: abc20js-abc20

Version: 4.0

License: Rose 1.5

3. Continue with Step 2: Create the Custom License.

Once the custom license is saved, a new component-version-license instance using the custom license is automatically created and applied “behind the scenes” to the inventory item. You can view this new instance through the Component Lookup feature.

## When Creating or Editing a “License Only” Inventory Item

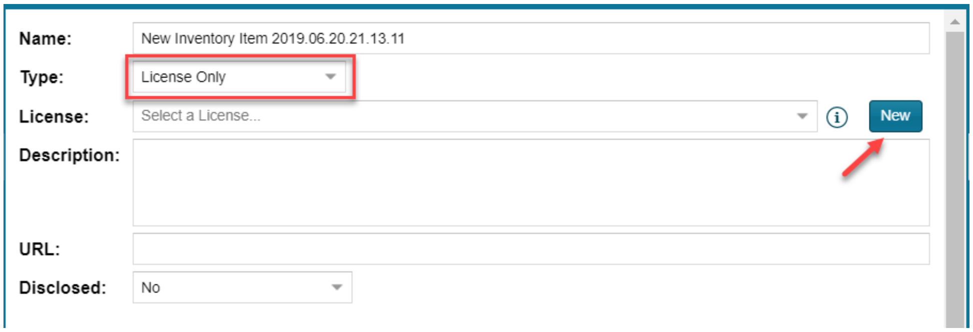
You can create a custom license to associate with an inventory item of the type **License Only** that you are creating or editing from the **Project Inventory** tab or in the **Analysis Workbench**.

***Note*** *▪ Generally you create a **License Only** inventory item if you know the license for the third-party code or artifact but do not know the component. (You can later edit this inventory item to convert it to one of the other inventory types.) This type of inventory is typically used for groups of files of unknown origin that are governed by a specific license. When you select the license, the inventory name is automatically generated as **Files under <LICENSE NAME> License**.*

***Task*** ***To create a custom license when creating or editing a “License Only” inventory item, do the following:***

1. When creating or editing inventory in the **Analysis Workbench** or from the **Project Inventory** tab, select **License Only** in the **Type** field for the inventory item.

2. Next to the **License** dropdown, click **New** to display the **Create Custom License** window.



Name: New Inventory Item 2019.06.20.21.13.11

Type: License Only

License: Select a License...

Description:

URL:

Disclosed: No

3. Continue with Step 2: Create the Custom License.

Once you save the custom license, it is automatically associated with the inventory item.

## When Creating or Updating a License Policy

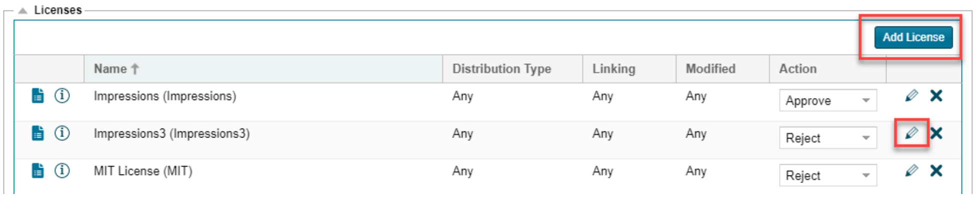
Policies are used by Code Insight to automate the inventory review process—that is, automatically mark published inventory items as approved or rejected—without the need for a manual review. A policy's criteria is based on OSS or third-party component versions, license attributes, or security vulnerability score and severities. For complete details, see Managing Policy Profiles.

As an alternative to selecting an existing license when you create or edit a policy based on a license, you can create a custom license to assign to the policy.

***Task*** ***To create a custom license to assign to a policy as you create or edit the policy, do the following:***

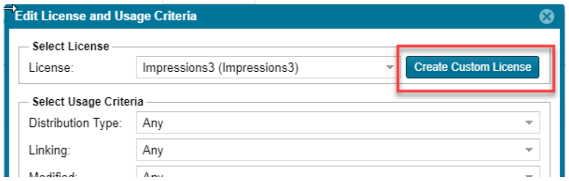
1. Open the policy profile for which you want to create or edit a license policy. See Adding or Editing a Policy Profile for details.
2. Navigate to the **Licenses** section in the policy profile and do either:

    - To edit an existing license policy, click the **Edit** icon ✎ at the end of that policy's row.
    - To create a new license policy, click **Add License**.

    | Name ↑ | Distribution Type | Linking | Modified | Action |
    |---|---|---|---|---|
    | Impressions (Impressions) | Any | Any | Any | Approve |
    | Impressions3 (Impressions3) | Any | Any | Any | Reject |
    | MIT License (MIT) | Any | Any | Any | Reject |

    The **Edit** (or **Add**) **License and Usage Criteria** window is displayed.

    License: Impressions3 (Impressions3)
    Distribution Type: Any
    Linking: Any
    Modified: Any

3. Click **Create Custom License** to open the **Create Custom License** window.
4. Continue with Step 2: Create the Custom License for details on how to create the license.

    Once you save the custom license, it is added to the **License** dropdown on the **Edit** (or **Add**) **License and Usage Criteria** window and is in focus for your immediate selection. For more details about fields on this window, see Maintaining License Policies.

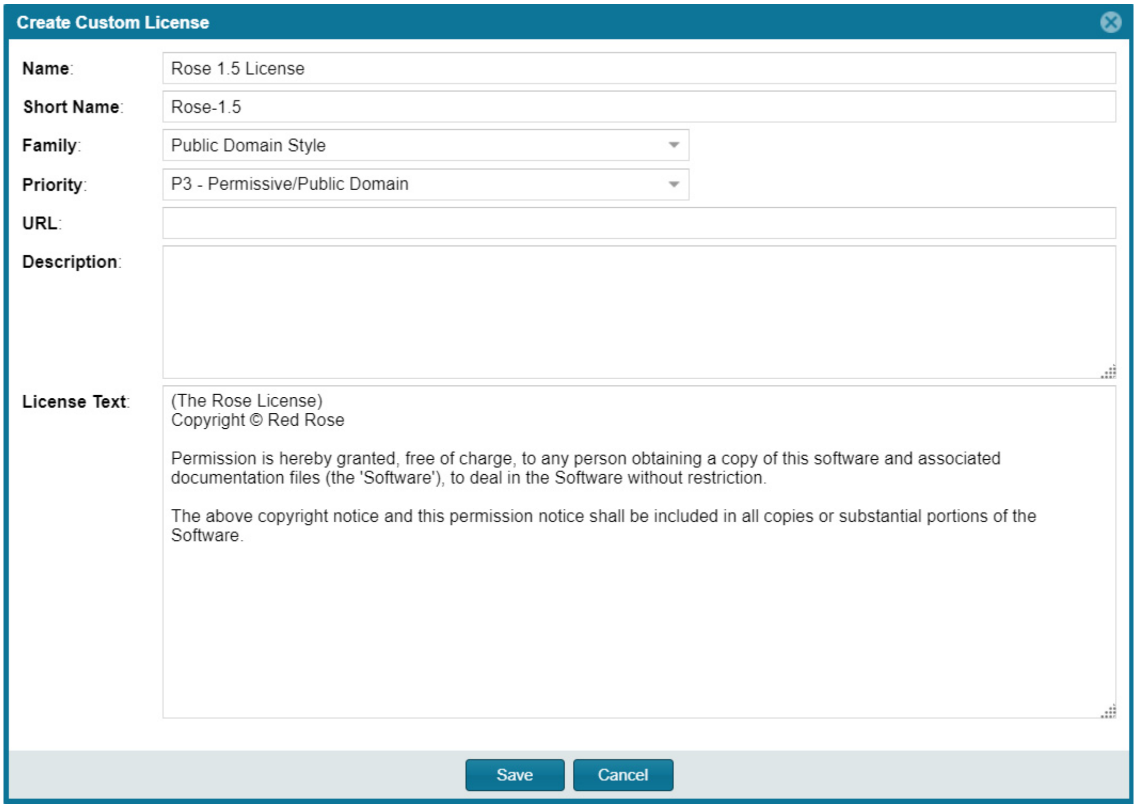
5. Click **Save** on the **Edit** (or **Add**) **License and Usage Criteria** window to return to the **Policy Details** window showing the policy profile. From here you can save the profile or continue to edit it.

## Step 2: Create the Custom License

Once you performed one of the procedures in Step 1: Initiate the Creation of a Custom License to open the **Create Custom License** window, use these steps to create the custom license.

***Task*** ***To create the custom license, follow these steps:***

1. From the **Create Custom License** window, provide the license properties. **Name**, **Short Name**, and **License Text** are required fields. For a description of the properties, see Custom License Properties.



Name: Rose 1.5 License
Short Name: Rose-1.5
Family: Public Domain Style
Priority: P3 - Permissive/Public Domain
URL:
Description:
License Text: (The Rose License)
Copyright © Red Rose

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

2. Click **Save**. The custom license is associated with the inventory item or the license policy as described in the previous sections in Step 1: Initiate the Creation of a Custom License.

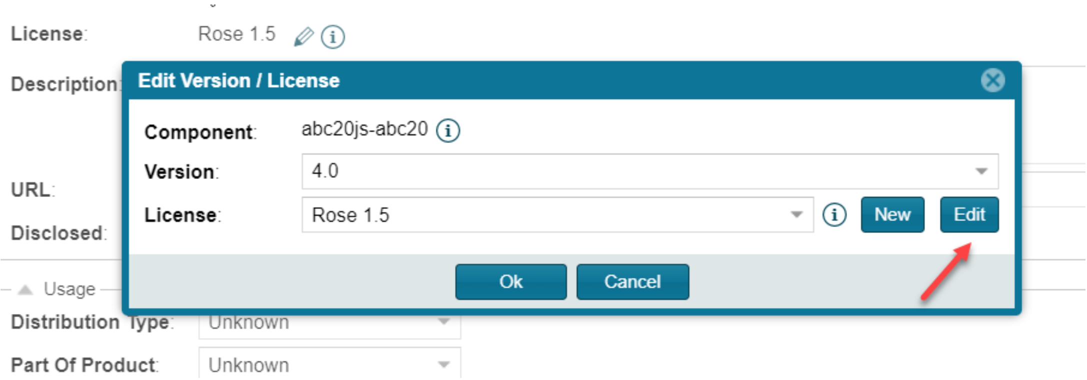
# Editing a Custom License

You can edit only custom licenses. When a custom license is selected in any **License** dropdown, the **Edit** button is displayed next to the **New** button, enabling you to update properties for the license. The updates you make to the license at the inventory-item level are saved to the license in the Code Insight data library.

***Task*** ***To edit a custom license, follow these steps:***

1. Using any of the procedures described in Step 1: Initiate the Creation of a Custom License, access the License dropdown used to associate a license with the given inventory item.

2. If the custom license you want to update is not already selected for the inventory item, select it from the **License** dropdown.



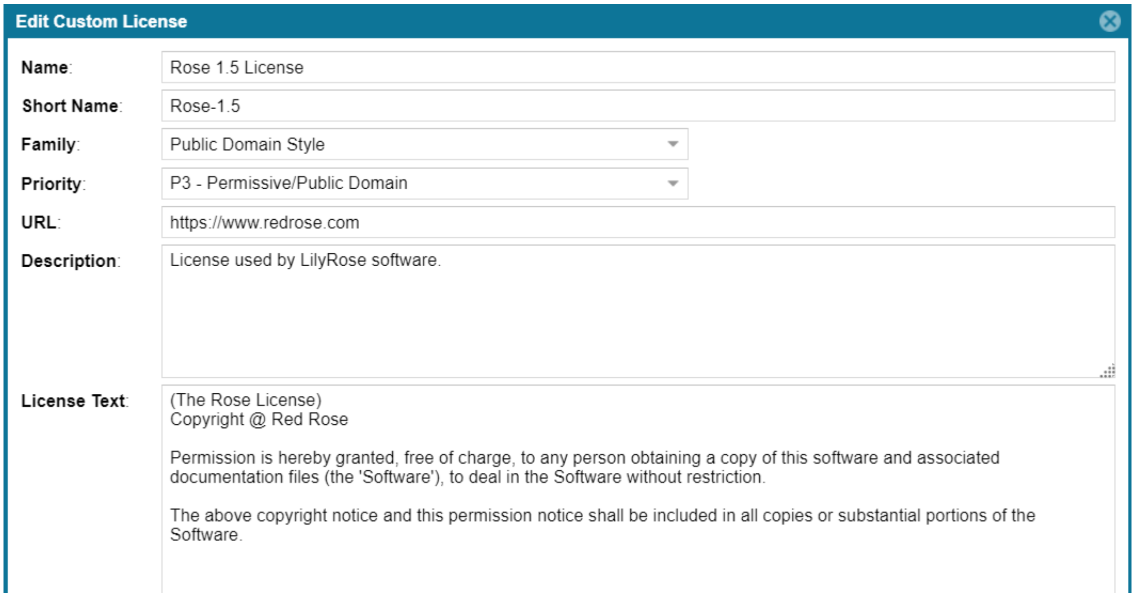
Edit Version / License

Component: abc20js-abc20

Version: 4.0

License: Rose 1.5

3. Click **Edit** next to the **License** dropdown to open the **Edit Custom License** window.



Edit Custom License

Name: Rose 1.5 License

Short Name: Rose-1.5

Family: Public Domain Style

Priority: P3 - Permissive/Public Domain

URL: https://www.redrose.com

Description: License used by LilyRose software.

License Text:
(The Rose License)
Copyright @ Red Rose

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

4. Update the license properties as needed, and click **Save**. For a description of the properties, see Custom License Properties.

The license is updated for the inventory item and saved to the Code Insight data library for global use.

## Custom License Properties

The following describes the fields on the **Create** (or **Edit**) **Custom License** window used to define a custom license.

**Table 2-12 ▪** Fields to Define a Custom License

| Component Field | Description |
|---|---|
| **Name** | The full name of the license (for example, `The Rose License 1.5`). |
| **Short Name** | A unique shorthand representation of the license. This is usually the license SPDX short identifier (for example, `Rose-1.5`). |
| **Family** | A license category that spans multiple license instances (for example, MIT, Public Domain, BSD-3 Clause, and others). The family designation is helpful to a legal reviewer to understand the “type” of license prior to investing the time to analyze the complete license text. |
| **URL** | The URL used to access the license on the Internet. This value should start with `http://` or `https://`. |
| **Priority** | The level of importance in investigating this license in terms of its possible business impact on your organization. The higher the level, the greater need to investigate the license:<br>● **P1**—Viral, strong copyleft license (requires immediate attention).<br>● **P2**—Weak copyleft or commercial or uncommon license (requires legal review).<br>● **P3**—Permissive or public domain license (generally allowed because of its minimal business impact). This is the default if no priority is specified.<br>For details about each priority level, see Auditing Scan Results in the Analysis Workbench. |
| **Description** | Meaningful information about the license for your reference. |
| **License Text** | The complete text of the license. Be sure to encode any HTML characters. |

# Managing Custom Detection Rules

During a Code Insight project scan, the Automated Analysis component of the Scan Server uses a set of internal detection rules, stored in the Code Insight data library, to automatically generate inventory items. However, in some cases, your manual analysis might find codebase files that are associated with a third-party component but not associated with inventory. Code Insight enables you to create custom detection rules that convert such findings into future automated inventory items. These rules are saved to the Code Insight data library for global use by Automated Analysis during scans on future projects (or rescans on current projects).

The following sections provide more information about managing custom detection rules:

- Creating a Custom Detection Rule
- Viewing All Current Custom Detection Rules
- Editing a Custom Detection Rule
- Deleting a Custom Detection Rule
- Rule-Processing Considerations

# Creating a Custom Detection Rule

The following sections describe the methods used to create a custom detection rule:

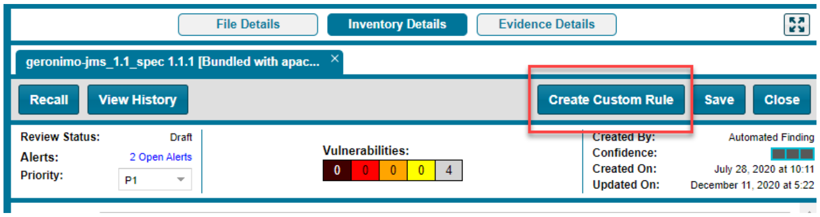
- Creating a Custom Detection Rule from Inventory of “Component” Type
- Creating a Custom Detection Rule from Scratch

## Creating a Custom Detection Rule from Inventory of “Component” Type

During codebase analysis in the **Analysis Workbench** for a project, you might find one or more codebase files that are associated with a specific third-party component but are not associated with current inventory. You can manually update the existing inventory item associated with the component to include the files or, if inventory does not exist, create an inventory item based on the component and associate the files. (The inventory item type must be defined as **Component**.) You then have the option to create a global custom detection rule based on the details of the updated or created inventory item, as described in the following procedure. This method pre-populates much of the information needed to create the rule, including the MD5 value for each file associated with the inventory item.

***Task*** ***To create a custom detection rule using the context of a manually created inventory item, do the following:***

1. In the **Analysis Workbench** for the desired project (see Opening the Analysis Workbench), navigate to the **Inventory Items** pane and select the inventory item with which you have manually associated the codebase files. The **Inventory Details** tab for inventory item is opened.



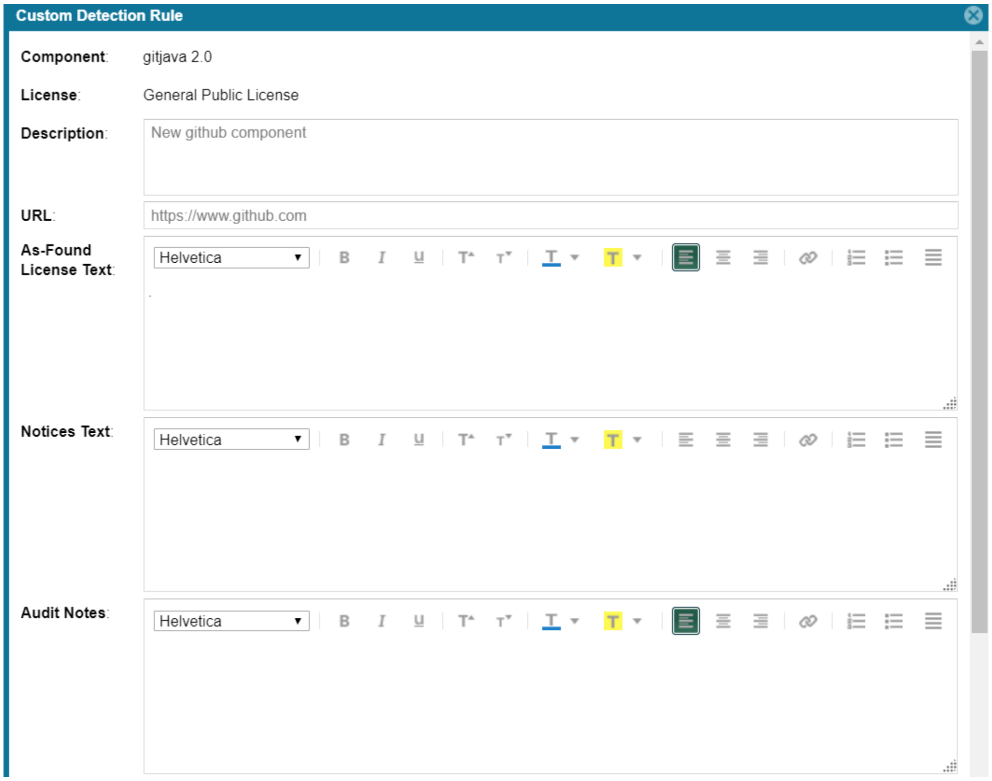
Review Status: Draft
Alerts: 2 Open Alerts
Priority: P1

Vulnerabilities: 0 0 0 0 4

Created By: Automated Finding
Confidence:
Created On: July 28, 2020 at 10:11
Updated On: December 11, 2020 at 5:22

2. Click the **Create Custom Rule** button to open the **Custom Detection Rule** dialog. For a description of the fields on this dialog, refer to Custom Detection Rule Dialog.

   Note that the **Component**, **License**, **Description**, and **URL** values are pre-populated from the inventory item on which you are basing the rule and are not editable.



Custom Detection Rule

Component: gitjava 2.0

License: General Public License

Description: New github component

URL: https://www.github.com

As-Found License Text: Helvetica

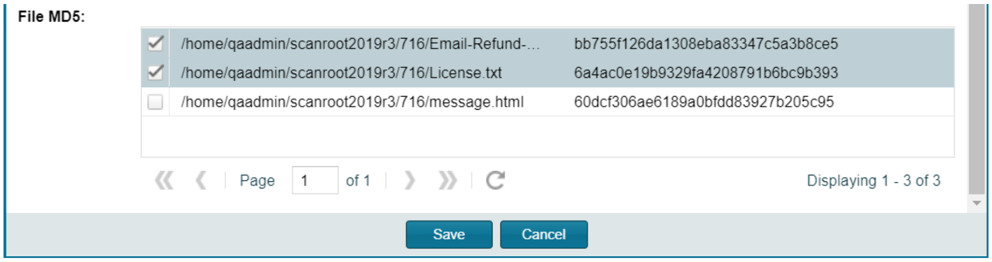
Notices Text: Helvetica

Audit Notes: Helvetica

3. Add or edit **As-Found License Text**, **Notices Text**, and **Audit Notes** content as needed for the rule. (These fields are pre-populated with any content currently defined in the inventory item on which you are basing the rule.) This information is displayed for inventory items automatically created or updated by the rule in the future.

4. Scroll down to the **File MD5** pane, which is pre-populated with the list of codebase files associated with the inventory item you created. The MD5 value for each file is provided.

5. Select one or more files to add to the rule.



File MD5:

| | | |
|---|---|---|
| ☑ | /home/qaadmin/scanroot2019r3/716/Email-Refund-... | bb755f126da1308eba83347c5a3b8ce5 |
| ☑ | /home/qaadmin/scanroot2019r3/716/License.txt | 6a4ac0e19b9329fa4208791b6bc9b393 |
| ☐ | /home/qaadmin/scanroot2019r3/716/message.html | 60dcf306ae6189a0bfdd83927b205c95 |

Page 1 of 1

Displaying 1 - 3 of 3

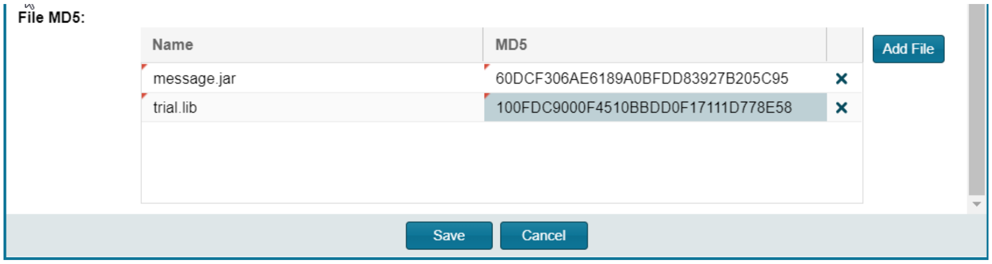
6. Click **Save** to create the rule and add it to the Code Insight data library. You will be asked for confirmation to proceed the creation.

## Creating a Custom Detection Rule from Scratch

You can create a custom detection rule from scratch—that is, without being in the context of an inventory item that you have manually updated or created to add codebase files, as described in Creating a Custom Detection Rule from Inventory of “Component” Type. The method enables you to create custom detection rules if you do not have access to the **Analysis Workbench** for a specific project. However, you will need to provide information manually, including the name and MD5 value for each codebase file that you want to associate with the rule.

***Task*** ***To create a custom detection rule from scratch, do the following:***

1. Open the **Custom Detection Rules** tab, using the procedure in Viewing All Current Custom Detection Rules.
2. Click **Create Custom Rule** to open the **Custom Detection Rule** dialog. For a description of the fields on this dialog, refer to Custom Detection Rule Dialog. Note the following:
   - Once you use **Lookup Component** to select or create a component for the rule, the component information is populated on the dialog. However, you can edit this information as needed.
   - You can add **As-Found License Text**, **Notices Text**, and **Audit Notes** content as needed for the rule. This information is displayed in the future inventory items created automatically by the rule.
3. Scroll down to the **File MD5** pane.
4. For each codebase file you want to add to the rule, click the **Add File** button and provide the file’s name and MD5 value.



File MD5:

| Name | MD5 | |
|---|---|---|
| message.jar | 60DCF306AE6189A0BFDD83927B205C95 | ✕ |
| trial.lib | 100FDC9000F4510BBDD0F17111D778E58 | ✕ |

5. Click **Save** to create the rule and add it to the Code Insight data library. You will be asked for confirmation to proceed with the creation.

# Viewing All Current Custom Detection Rules

Use the following the procedure to access the **Custom Detection Rules** tab, from which you can view all currently defined custom detection rules and act on them as needed.

***Task*** ***To view all current custom detection rules created for your Code Insight system, do the following:***

1. Click the **Open Menu** icon in the upper right of any Code Insight page:

   ☰

2. Select **Data Library** from the menu to open the **Custom Detection Rules** tab, showing the list of current custom detection rules. From this tab, you can do the following:

    - View the component information (name, version, license, and forge URL) on which each rule is based.
    - Create a custom detection from scratch (see Creating a Custom Detection Rule).
    - Edit a custom detection rule or remove it from the Code Insight system.

# Editing a Custom Detection Rule

Use the following procedure to edit a specific custom detection rule. The changed rule is applied to all future scans or rescans on projects in the Code Insight.

***Task*** ***To edit an existing custom detection rule, do the following:***

1. Open the **Custom Detection Rules** tab, following the procedure in Viewing All Current Custom Detection Rules.
2. In the **Actions** column for the entry you want to update, click the ✎ (**Edit**) icon. The **Edit Custom Rule** dialog opens.
3. Edit fields as needed. See Edit Custom Rule Dialog for a description of each field.
4. Manage the codebase files for the rules:
    - To add a file, click the **Add File** button and provide the file's name and MD5 value.
    - To remove a file from the rule, click the ✖ (**Delete File**) icon.
5. Click **Save** to update the detection rule changes to the Code Insight data library. You will be asked for confirmation to proceed with the updates.

# Deleting a Custom Detection Rule

Use the following procedure to remove a specific custom detection rule from the Code Insight data library. The rule will no longer be applied to any future scan in your Code Insight system.

***Task*** ***To remove a custom detection rule from the Code Insight data library, do the following:***

1. Open the **Custom Detection Rules** tab, following the procedure in Viewing All Current Custom Detection Rules.
2. In the **Actions** column for the entry you want to remove, click the ✖ (**Delete**) icon. You are asked to confirm the deletion.

# Rule-Processing Considerations

As you manage custom detection rules, consider how the rules are processed under certain circumstances:

- If the custom detection rule is associated with more than one file, the scan uses OR logic when processing the files against the target codebase. Consequently, only one file match between codebase and the rule is required to automatically create an inventory item.
- If two rules are created with identical details and codebase files, a single inventory item is generated during a scan when both rules are applied.
- If two rules are created using the same component, version, and license details and the same codebase files, but have different **Description**, **URL**, **Audit Notes**, **As-Found License Text**, or **Notices Text** content, a single inventory item is generated during a scan when both rules are applied. In the inventory item, values that differ between the rules for a given field are separated (shown on separate lines or with a separator) within the field.
- If two rules with are created with the same codebase files but use a different component, two inventory items are generated during the scan.

# Finalizing the Notices Text for the Notices Report

If you want to create or modify license text for a given inventory item, you can use the **Notices Text** field to provide the exact content to include in the Notices report. For example, you can edit any license text previously saved to this field or add your own license text, such as license information for rules that you developed during your manual research on the inventory item. You can also copy the **As-Found License Text** content to the **Notices Text** field and modify it as needed. (Content in the **As-Found License Text** pane is not editable but can be copied to the **Notices Text** field and modified.)

If the **Notices Text** field item field contains information when the Notices report is run, the content of this field alone is pulled into the report. If the **Notices Text** field is empty, the content of the **As-Found License Text** field is used in the report. If both fields are empty, the report uses the license content from Code Insight data library (see License Details from the Code Insight Data Library).

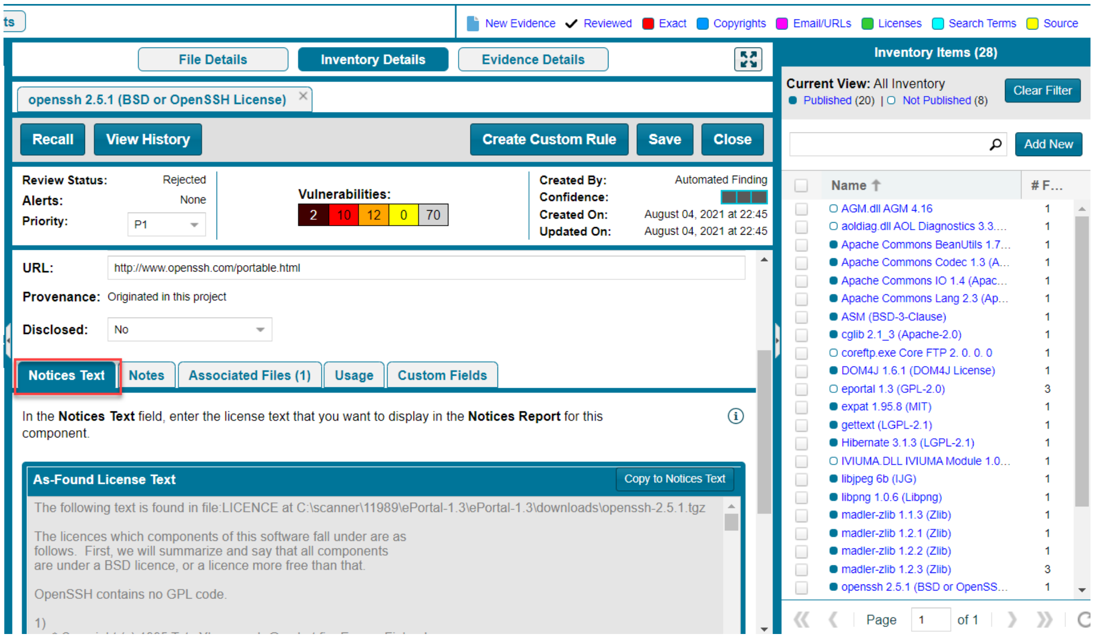
For more information about these fields, see Reporting of Detected License Text through the As-Found Text Inventory Field and Notices Text.

*Note* ▪ *If the **As-Found License Text** field contains content populated by the Scan Server, best practice is to leave the **Notices Text** field empty (as long as custom information or edits are not required) so that the report is forced to use the license information found in the **As-Found License Text** field.*

***Task*** ***To provide custom content for the Notices report, do the following:***

1. Navigate to one of these locations:

   In the **Analysis Workbench**, on the **Inventory Details** tab for a specific inventory item, open the **Notices Text** tab. If necessary, see Opening the Analysis Workbench.



Review Status: Rejected
Alerts: None
Priority: P1

Vulnerabilities: 2 | 10 | 12 | 0 | 70

Created By: Automated Finding
Confidence:
Created On: August 04, 2021 at 22:45
Updated On: August 04, 2021 at 22:45

URL: http://www.openssh.com/portable.html

Provenance: Originated in this project

Disclosed: No

In the **Notices Text** field, enter the license text that you want to display in the **Notices Report** for this component.

**As-Found License Text**

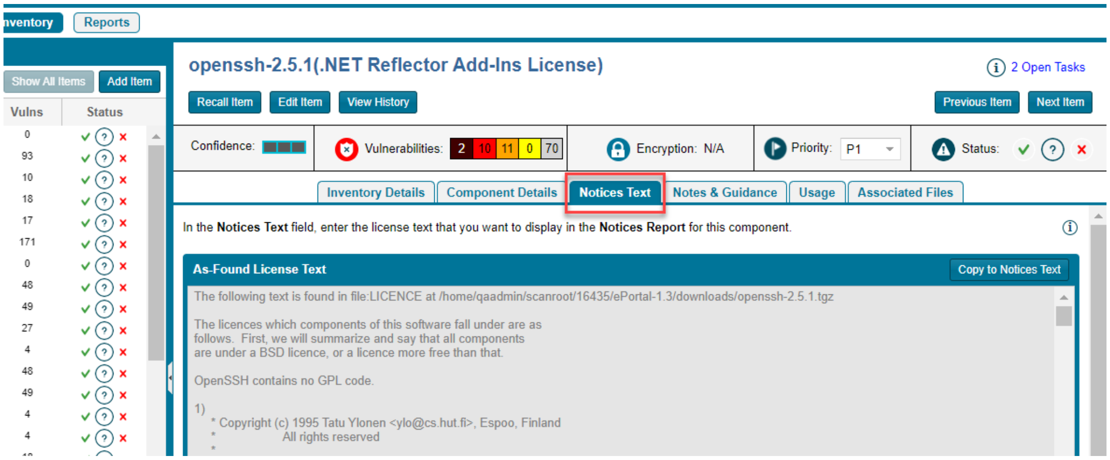
```
The following text is found in file:LICENCE at C:\scanner\11989\ePortal-1.3\ePortal-1.3\downloads\openssh-2.5.1.tgz

The licences which components of this software fall under are as
follows.  First, we will summarize and say that all components
are under a BSD licence, or a licence more free than that.

OpenSSH contains no GPL code.

1)
```

Current View: All Inventory
● Published (20) | ○ Not Published (8)

| | Name ↑ | # F... |
|---|---|---|
| □ | ○ AGM.dll AGM 4.16 | 1 |
| □ | ○ aoldiag.dll AOL Diagnostics 3.3.... | 1 |
| □ | ● Apache Commons BeanUtils 1.7... | 1 |
| □ | ● Apache Commons Codec 1.3 (A... | 1 |
| □ | ● Apache Commons IO 1.4 (Apac... | 1 |
| □ | ● Apache Commons Lang 2.3 (Ap... | 1 |
| □ | ● ASM (BSD-3-Clause) | 1 |
| □ | ● cglib 2.1_3 (Apache-2.0) | 1 |
| □ | ○ coreftp.exe Core FTP 2. 0. 0. 0 | 1 |
| □ | ● DOM4J 1.6.1 (DOM4J License) | 1 |
| □ | ○ eportal 1.3 (GPL-2.0) | 3 |
| □ | ● expat 1.95.8 (MIT) | 1 |
| □ | ● gettext (LGPL-2.1) | 1 |
| □ | ● Hibernate 3.1.3 (LGPL-2.1) | 1 |
| □ | ○ IVIUMA.DLL IVIUMA Module 1.0... | 1 |
| □ | ● libjpeg 6b (IJG) | 1 |
| □ | ● libpng 1.0.6 (Libpng) | 1 |
| □ | ● madler-zlib 1.1.3 (Zlib) | 1 |
| □ | ● madler-zlib 1.2.1 (Zlib) | 1 |
| □ | ● madler-zlib 1.2.2 (Zlib) | 1 |
| □ | ● madler-zlib 1.2.3 (Zlib) | 3 |
| □ | ● openssh 2.5.1 (BSD or OpenSS... | 1 |

Page 1 of 1

From **Project Inventory**, select an inventory item and open the **Notices Text** tab. If necessary, see Displaying Project Inventory.



openssh-2.5.1(.NET Reflector Add-Ins License)

2 Open Tasks

| Vulns | Status |
|---|---|
| 0 | ✓ ? × |
| 93 | ✓ ? × |
| 10 | ✓ ? × |
| 18 | ✓ ? × |
| 17 | ✓ ? × |
| 171 | ✓ ? × |
| 0 | ✓ ? × |
| 48 | ✓ ? × |
| 49 | ✓ ? × |
| 27 | ✓ ? × |
| 4 | ✓ ? × |
| 48 | ✓ ? × |
| 49 | ✓ ? × |
| 4 | ✓ ? × |
| 4 | ✓ ? × |

Confidence:
Vulnerabilities: 2 | 10 | 11 | 0 | 70
Encryption: N/A
Priority: P1
Status: ✓ ? ×

In the **Notices Text** field, enter the license text that you want to display in the **Notices Report** for this component.

**As-Found License Text**

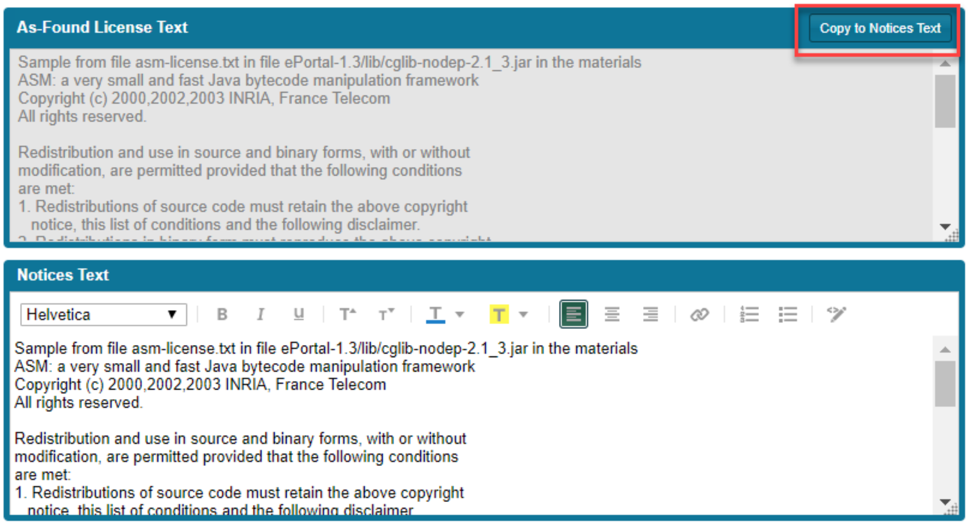
```
The following text is found in file:LICENCE at /home/qaadmin/scanroot/16435/ePortal-1.3/downloads/openssh-2.5.1.tgz

The licences which components of this software fall under are as
follows.  First, we will summarize and say that all components
are under a BSD licence, or a licence more free than that.

OpenSSH contains no GPL code.

1)
   * Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland
   *                    All rights reserved
   *
```

2. Do one or both of the following:

   - In the **Notices Text** field, enter new or modify existing license content for the inventory item. The text and its format should look exactly as you want it to appear in the Notices report.
   - Using the following steps, copy the **As-Found License Text** content to the **Notices Text** field and modify the content as needed:

     a. Click the **Copy to Notices Text** button in the top right corner of the **As-Found License Text** field.

        If the **Notices Text** field is empty, the **As-Found License Text** content is copied to the **Notices Text** field, with the formatting preserved.



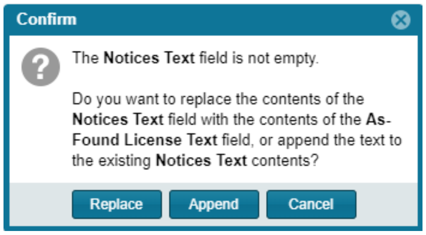
Sample from file asm-license.txt in file ePortal-1.3/lib/cglib-nodep-2.1_3.jar in the materials
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000,2002,2003 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright

Sample from file asm-license.txt in file ePortal-1.3/lib/cglib-nodep-2.1_3.jar in the materials
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000,2002,2003 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

If the **Notices Text** field is *not* empty, you are given the option to append the **As-Found License Text** content to the existing **Notices Text** content or to replace all the existing **Notices Text** content with the **As-Found License Text** content.



The **Notices Text** field is not empty.

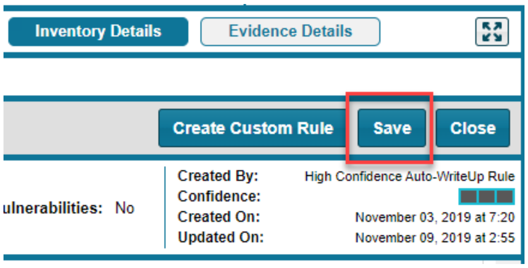
Do you want to replace the contents of the **Notices Text** field with the contents of the **As-Found License Text** field, or append the text to the existing **Notices Text** contents?

If you select **Append**, the appended content is added to the **Notices Text** field, starting on a new line after the existing content. If you select **Replace**, the existing **Notices Text** content is replaced.

b. Modify and format the **Notices Text** content as needed.

3. Save the **Notices Text** changes to the current inventory item:

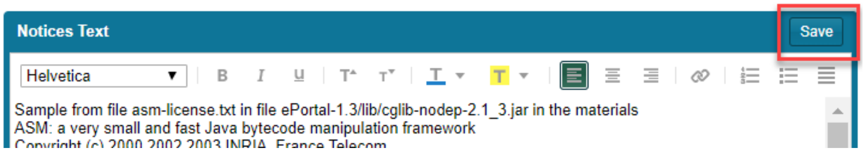
- If you are in the **Analysis Workbench**, click the **Save** button at the top of the **Inventory Details** tab. (Alternatively, click **Close** to shut down the tab for the current inventory item. You are prompted to save the inventory changes before the tab closes.)



ulnerabilities: No

Created By: High Confidence Auto-WriteUp Rule
Confidence:
Created On: November 03, 2019 at 7:20
Updated On: November 09, 2019 at 2:55

- If you are in **Project Inventory**, click the **Save** button at the top of the **Notices Text** field.



Sample from file asm-license.txt in file ePortal-1.3/lib/cglib-nodep-2.1_3.jar in the materials
ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000 2002 2003 INRIA, France Telecom

When the Notices report is run, the content from the **Notices Text** pane is used as the “notices” information for the inventory item in the report.

# Generating Reports for a Project

The **Reports** tab enables users to generate Code Insight reports that present different views of the data collected for a given project. The following sections describe the different reports available for any project and how to generate these reports:

- About the Standard Reports for Projects
- About Custom Reports for Projects
- Generating a Report for a Project

## About the Standard Reports for Projects

The following describes the reports that come standard with Code Insight and are available for any project:

- Project Report
- Audit Report
- Notices Report

### Project Report

The Project report summarizes the inventory, security vulnerabilities, remaining scan evidence, and review and remediation tasks for a selected project. It produces output in JSON and Excel format. This report is useful in understanding the existing project’s legal and security risks based on identified inventory items, as well as the additional potential risk based on the file-based scan results known as third-party indicators.

Note the following:

- The metrics and statistics in this report are based on the results of the most recent server scan and remote scan(s) associated with the project.
- Currently, Code Insight is able to report license evidence found in remote files scanned by a scan agent. This evidence is reflected (along with evidence detected by the Scan Server) in the charts and data in the following locations:
    - **Additional Evidence** section of the **Summary** sheet
    - **Files with License** sheet (with an **Alias** column to help you determine which files are remote)
    - **All Scanned Files** sheet

- When the report lists codebase files, an alias and file path can be included with each file name in the format <alias>:<filePath> (or as separate properties). The alias is a unique descriptive name representing the scan-root path for the Scan Server or remote scan agent, and the file path is relative to scan root. (The actual absolute scan-root path for each scanner associated with the project is available on the project's **Summary** sheet.)
- The security vulnerability information in the report is based on the CVSS version (v3.x or v2.0) currently used by your Code Insight system for reporting purposes. If CVSS 3.x is used, vulnerability counts and information in the report are based on data from all CVSS v3 systems supported by Code Insight, currently v3.1 and v3.0. (A given vulnerability can have only one v3 score—either a v3.1 or v3.0 score, not both.)
- Suppressed security vulnerabilities are not shown in this report, and total counts for vulnerabilities do not include suppressed vulnerabilities.

## Audit Report

Audit reports provide another way to distribute your research and findings to others in your organization. Only published inventory items appear in Audit reports.

Note the following:

- The metrics and statistics in this report are based on the results of the most recent server scan and remote scan(s) associated with the project.
- When the report lists codebase files, an alias and file path can be included with each file name in the format <alias>:<filePath>. The alias is a unique, descriptive name representing the scan-root path for the Scan Server or remote scan agent, and the file path is relative to scan root. (The actual scan root for each scanner associated with a project is available on the project's **Summary** sheet.)
- The total lines of code listed on the **Summary** sheet is based on the server-side codebase only; the total does not include lines of code in the remote codebase(s).
- The security vulnerability information in the report is based on the CVSS version (v3.x or v2.0) currently used by your Code Insight system for reporting purposes. If CVSS 3.x is used, vulnerability counts and information in the report are based on data from all CVSS v3 systems supported by Code Insight, currently v3.1 and v3.0. (A given vulnerability can have only one v3 score—either a v3.1 or v3.0 score, not both.)
- Suppressed security vulnerabilities are not shown in this report, and total counts for vulnerabilities do not include suppressed vulnerabilities.

## Notices Report

Code Insight provides the ability to produce a Notices report to satisfy the attribution requirements of most open source licenses. The report is created in text format.

After Engineering has completed the remediation plan, resolving all rejected inventory items, the codebase is rescanned until it is approved for release. When the codebase is approved for release, you need to generate a Notices report to accompany the software application. This report is a compilation of all the open source/third-party components contained in the product and their license content (notices).

The Notices report shows only published inventory. The inventory can be system-generated or custom and of any type—**Work in Progress**, **Component**, or **License**.

The following items can appear in the Notices report for each inventory item:

- **Inventory name**—The entry in this field is based on naming conventions, which is usually the component name, version, and governing license name.
- **Inventory URL**—If the inventory URL is not available, Code Insight uses the associated component URL. If both are unavailable, no URL will appear in the report.
- **Inventory Notices Text**— The final "notices" text associated with the inventory item. It is pulled from the **Notices Text** field on the **Notices Text** tab for a selected inventory item in the **Analysis Workbench** or in **Project Inventory**. If this field is empty, Code Insight uses the content in the **As-Found License Text** field (also on the **Notices Text** tab), which shows the verbatim text license text found in the codebase by the system. If no **As-Found License Text** or **Notices Text** information is available, the text pulled from the Code Insight data library for the selected license is used in the Notices report. For more information, see Finalizing the Notices Text for the Notices Report

# About Custom Reports for Projects

Code Insight provides a Custom Reports Framework that enables users to develop and register custom reports that show project data curtailed for the needs of one's site. For complete details on developing and registering a custom report, go to the following link:

https://community.flexera.com/t5/FlexNet-Code-Insight-Customer/Custom-Reports-Framework-in-FlexNet-Code-Insight/ta-p/132702

A custom report can be defined to output in one or more desired formats. Once a custom report is developed and registered through the Framework, it is available to any project as a selection along with the standard reports and any other custom reports on the **Reports** tab. Users can then generate the custom report using the same procedure used to generate a standard report.

Depending on how the custom report is defined, users might be prompted for additional information before they can generate the report. For example, they might be required to select a second project so that data from the current project and second project can be compared or combined into a single report. Or they might be prompted to enter other information such as values to filter report content.

Any user can set up custom reports for a project. (For private projects, any user assigned to a role for the private project can set up custom reports for the project.)

## Example Custom Reports

The following example custom reports (located in Revenera's public GitHub report repositories) are currently available to you:

- Project Inventory Report
- Evidence Report
- Project Comparison Report
- Claimed Evidence Report
- Vulnerabilities Report
- Project Compliance Report

You can download any of these reports, register them with Code Insight, and then generate the reports as needed. Additionally, these reports can be modified for your special use or serve as the basis for creating other custom reports. Note the following disclaimer for using these example reports.

### Disclaimer for Using the Example Custom Reports

These report scripts are being provided solely as examples. They are external to, and not an official part of, the Code Insight product.

THE REPORT SCRIPTS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SCRIPT OR THE USE OR OTHER DEALINGS IN THE REPORT SCRIPTS.

## Project Inventory Report

This report provides an easy, quick method for obtaining a high-level summary of the inventory items within a project.

If you have designated a parent-child hierarchy for your projects to better represent your company offerings, this report can be configured to pull in all child projects (recursively) for the current project and roll up the associated inventory information on a project, as well as an application, basis. Including child projects in the report is useful for keeping track of your software Bill of Materials (SBOM). The report can be further customized to report on other inventory attributes, such as third-party notices, which in turn would capture the notices for all the third-party components included in the report scope.

The report is available at this location:

> https://github.com/flexera/sca-codeinsight-reports-project-inventory

## Evidence Report

This report allows you to report on the following types evidence found in the project:

- Copyrights
- Licenses
- Emails and URLs
- Search terms
- Exact-file matches
- Source-code matches (snippets)

The report is available at this location:

> https://github.com/flexera/sca-codeinsight-reports-third-party-evidence

## Project Comparison Report

This report compares the inventory of two projects or two project versions, enabling you to identify inventory differences and commonalities.

The report is available at this location:

https://github.com/flexera/sca-codeinsight-reports-project-comparison

### Claimed Evidence Report

This report allows you to determine which files in a project contain only evidence that is claimable based on string comparisons to the follow evidence types:

- Copyrights
- Emails/URLs

Additionally, you can configure the report so that scanned files that contain only the evidence for the specified claimable values are marked as reviewed and associated with the appropriate inventory items.

The report is available at this location:

https://github.com/flexera/sca-codeinsight-reports-claim-files

### Vulnerabilities Report

This security-focused report calls out all vulnerable project inventory items and lists their associated security vulnerabilities. Use this report to easily collect and review security issues or to share data with your Security team. The report supports searches and enables you to click-through to the actual vulnerable inventory in Code Insight for additional information.

The report is available at this location:

https://github.com/flexera/sca-codeinsight-reports-project-vulnerabilities

### Project Compliance Report

This report visualizes inventory items in a project in terms of their various compliance issues. The compliance issues listed in this report include inventory with P1 licenses, rejected inventory items, unreviewed items, inventory with security vulnerabilities, and inventory with old component versions.

The report is available at this location:

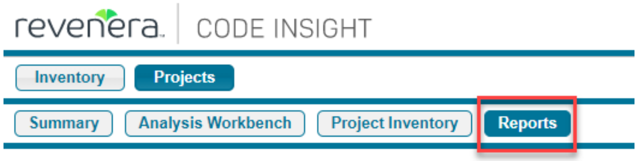
https://github.com/flexera/sca-codeinsight-reports-project-compliance

## Generating a Report for a Project

Use the following procedure for generating a report for a given project from its **Reports** tab. For more information about the details available on this tab, see Reports Tab. Any Code Insight user can generate reports.

***Task*** ***To generate a report for a project, do the following:***

1. Open a project in the **Projects** view. (For instructions, see Opening a Project.)
2. Click the **Reports** button at the top of the view to open the **Reports** tab.



The tab opens, showing the list of standard and custom reports available for the project.

3. Select a report from the report list.
4. Click **Generate Selected Report**.
    - If additional information is needed for the report, a pop-up window is displayed, prompting you for the information. See step 5. (Only custom reports can request additional information.)
    - If no additional information is needed, skip to step 6.
5. From the pop-up window requesting additional information to run the report, complete the fields:
    - If the **Include data from Second Project** field is displayed, enter the name of the second project whose data will be included along with the data from the current project for comparison purposes. As you type a string, project names containing that string are listed in a dropdown from which you can then select the desired project name. (This is a required field.)
    - If other fields are displayed, enter the requested values in those fields. Default values can be overwritten. Click the ? icon next to a field for more information about its purpose and possible values. The **Generate Report** button on the pop-up remains disabled until all required fields are completed. (Required fields left blank are outlined in red.)

    When these additional fields have been properly completed, click **Generate Report** on the pop-up window.
6. Click **OK** from the message box that is displayed, stating that the report will run in the background.

    The report generation starts.
7. (Optional) While a given report is generating, repeat steps 3 through 6 to generate another report. You can generate multiple different reports simultaneously. (While a report is generating, the **Generate Selected Report** button is disabled for that report, but *is* enabled for any other report not being generated.)
8. Once the generation of the report has successfully completed, links are displayed in the **View Report** and **Download Report** columns for the report. Select either or both options:
    - To view the report in your browser, click **View**.
    - To download the report, click **Download**. A `.zip` file is downloaded to your system's default location. The archive contains the report in one or more of these formats:
        - JSON—The report data can be processed programmatically to integrate with other applications.
        - XLSX—The report can be viewed in Microsoft Excel.
        - TXT—The report data is saved as text so that you can reformat the report data as desired.
9. Navigate to the folder where you saved the report `.zip` file, unzip the file, and open the report in the desired format.

### Post Report Generation

After the successful generation of a given report, its **View** and **Download** links continue to display, along with the date and time of the report generation, on the **Reports** tab until you regenerate the report.

### Report Generation Failure

If a report fails to generate, the message “Report generation failed, please refer to the logs for details” is displayed for the report on the **Reports** tab. A Code Insight System Administrator can review the contents of the `core.log` to determine the reason for the report failure and relay the information to the appropriate contacts to fix the issue. The message remains for the report on the **Reports** tab until another attempt to generate the report is made.

# Managing Scan Queues Across All Scan Servers

Code Insight provides a global Scan Queue that lets you manage the current scan queues for all active Scan Servers from a single location. You can monitor the queue for any Scan Server; and, if you have sufficient project permissions, you can stop the scan currently running on a given Scan Sever or remove any scans from that server’s scan queue.

- Monitoring Scan Queues
- Stopping a Scan Currently Running on a Scan Server
- Removing a Scan from the Scan Queue for a Scan Server

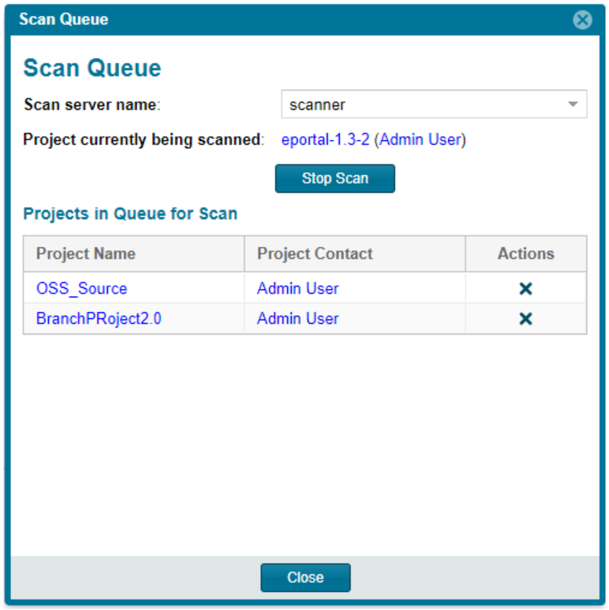
This feature does not monitor the scan queues for Code Insight remote scan agents.

## Monitoring Scan Queues

Use the following procedure to open the scan queue for a specific Scan Server to monitor its queue progress. You can then toggle between other Scan Servers to monitor their queues.

*Task* ***To access a global view of scan queues for all active Scan Servers, do the following:***

1. Click the icon ☰ in the upper right corner of the Code Insight Web UI to open the Code Insight main menu.
2. Select **SCAN QUEUE** from the menu.
3. From the **Scan Queue** dialog, use the **Scan server name** dropdown to select the Scan Server whose queue you want to view. (By default, this field initially displays the Scan Server selected by the System Administrator as the global project default.)



Scan Queue

Scan server name: scanner

Project currently being scanned: eportal-1.3-2 (Admin User)

Projects in Queue for Scan

| Project Name | Project Contact | Actions |
|---|---|---|
| OSS_Source | Admin User | ✕ |
| BranchPRoject2.0 | Admin User | ✕ |

The **Scan Queue** dialog shows the following information about the scan queue for the selected Scan Server:

| Field | Description |
|---|---|
| **Project currently being scanned** | A hyperlinked value in **<projectName> (<projectContact>)** format, showing the project currently being scanned by the selected Scan Server.<br>Optionally, click the project name to open to the project's **Summary** tab; or click the name of the Project Contact to create and send an email to this contact. These options can be useful for checking the scan's progress or notifying the Project Contact of an issue, especially if the scan if taking an excessive amount of time to execute.<br>If no project is being scanned, this field shows **N/A**.<br>If you have appropriate project permissions, a **Stop Scan** button is enabled to stop the currently running project. See Stopping a Scan Currently Running on a Scan Server for details. |
| **Projects in Queue for Scan** | The list of projects (in queue order) waiting to be scanned. Each project entry shows a hyperlinked project name and contact.<br>Optionally, click the name of a given project to open to the project's **Summary** tab; or click the name of a Project Contact to create and send an mail to this contact. These options can be useful if an issue exists with a project in the queue and notifications need to be sent.<br>If you have appropriate project permissions for a given project in the scan queue, an **X** icon is enabled in the **Actions** column to let you remove that project scan from the queue. See Removing a Scan from the Scan Queue for a Scan Server. |

4. Use the **Scan server name** dropdown to toggle between other Scan Server queues, or click **Close**.

# Stopping a Scan Currently Running on a Scan Server

Use this procedure to stop the scan currently running on a given Scan Server.

You can stop a scan only if you are an Analyst or a Project Administrator for the project associated with the scan.

***Task*** ***To stop the scan currently running on a given Scan Server, do the following:***

1. Click the icon ≡ in the upper right corner of the Code Insight Web UI to open the Code Insight main menu.
2. Select **SCAN QUEUE** from the menu.
3. From the **Scan Queue** dialog, use the **Scan server name** dropdown to select the Scan Server on which the scan you want to stop is running.

   Once you select a server, the **Stop Scan** button required to stop the scan will be enabled only if a scan is currently in progress on the Scan Server *and* you are an Analyst or a Project Administrator for the project on which the scan is running.
4. Ensure that the scan you are stopping is the desired scan; and, if so, click **Stop Scan**.

   You are asked to confirm that you want to stop the scan.
5. Click **Yes**.

   Once the scan stops, the scan for the next project in the queue begins.

# Removing a Scan from the Scan Queue for a Scan Server

Use this procedure to remove a scan from the scan queue for a given Scan Server.

You can remove a queued scan only if you are an Analyst or a Project Administrator for the project associated with the scan.

***Task*** ***To remove a scan from the scan queue for a given Scan Server, do the following:***

1. Click the icon ≡ in the upper right corner of the Code Insight Web UI to open the Code Insight main menu.
2. Select **SCAN QUEUE** from the menu.
3. From the **Scan Queue** dialog, use the **Scan server name** dropdown to select the Scan Server from whose scan queue you want to remove one or more scans.
4. In the scan queue, locate the project associated with the scan you want to remove, and click the **X** in the **Actions** column for that scan.

   ***Note*** ▪ *The* ***X*** *icon is enabled only if you are an Analyst or a Project Administrator for the project associated with the scan.*

   You are asked to confirm that you want to remove the scan from the scan queue.

5. Click **Yes** to remove the scan from the queue.
6. Repeat steps 4 and 5 to remove any additional scans in queue.

# Viewing Inventory Across All Projects

Code Insight enables you to view published inventory of open-source software (OSS) and other third-party components found across the projects in your Code Insight system. This inventory, displayed in a single scrollable window called the **Inventory** view, provides the means to make overall assessments of the OSS or third-party code used in your company's software.

The **Inventory** view can be filtered and refined as needed to focus on the inventory details and trends that most concern you and that are needed to make sound business decisions. For example, you might need to determine which open-source or third-party components are putting your software deliverables at security risk due to security vulnerabilities above a certain severity or CVSS score, or which are posing threats to your intellectual property due to non-compliant licensing per your corporate policies. You might want to filter to inventory where security, legal, and development resources are most needed to complete the review or remediation work required to ensure that OSS or third-party code is properly and safely integrated in all your software projects.

The **Inventory** view also provides direct links to inventory items and the projects associated with these items so that you can investigate or manage the inventory and projects as needed.

The following topics provide the procedures needed to access and use the **Inventory** view:

- Opening the Inventory View
- Switching the Context of the Inventory View
- Including the Inventory of Child Projects on the Inventory View
- Refining the Inventory View
- Viewing Inventory Properties and Linking to Additional Information
- Opening the Project Associated with Inventory from the Inventory View

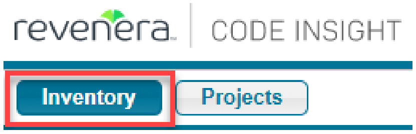
## Opening the Inventory View

Use the following procedure to access the **Inventory** view.

***Task*** ***To access the Inventory view, do the following:***

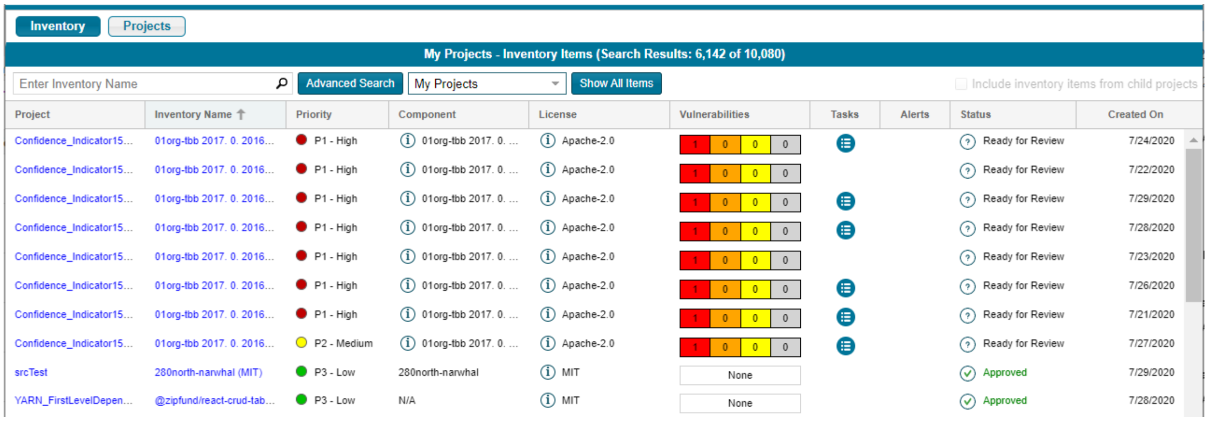
Open the **Inventory** view using one of these methods:

- From the Code Insight dashboard (which is displayed when you start Code Insight), click **view inventory**. See Opening Code Insight for details on accessing this dashboard.
- From any location in the Code Insight Web UI, click the **Inventory** button under the Code Insight logo.



- Click the ☰ icon in the upper right corner of the Code Insight Web UI to open the Code Insight main menu. Select **INVENTORY** from the menu.

The **Inventory** view is displayed. For a description of all the columns, fields, and button available on the view, see Inventory View.



My Projects - Inventory Items (Search Results: 6,142 of 10,080)

| Project | Inventory Name | Priority | Component | License | Vulnerabilities | Tasks | Alerts | Status | Created On |
|---|---|---|---|---|---|---|---|---|---|
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/24/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | | | ? Ready for Review | 7/22/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/29/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/28/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | | | ? Ready for Review | 7/23/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/26/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/21/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016... | ● P2 - Medium | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/27/2020 |
| srcTest | 280north-narwhal (MIT) | ● P3 - Low | 280north-narwhal | ⓘ MIT | None | | | ✓ Approved | 7/29/2020 |
| YARN_FirstLevelDepen... | @zipfund/react-crud-tab... | ● P3 - Low | N/A | ⓘ MIT | None | | | ✓ Approved | 7/28/2020 |

# Switching the Context of the Inventory View

When you open the **Inventory** view, you see the inventory associated with only your Code Insight projects. However, you can switch the context of inventory items as described in these sections:

- About the Available Contexts for the View
- Switching the Context of the View

## About the Available Contexts for the View

The following describe the three major contexts that are available for the **Inventory** view. You can switch between contexts as needed:

**Table 2-13 ▪** Contexts of the Inventory View

| Context | Description |
|---|---|
| **My Projects** | Shows all published inventory across those Code Insight projects in which you are assigned a role. You might use this context to show areas where you need to provide review or remedial work, or you might want to review the overall state of inventory found in your projects.<br>This is the context enabled by default when you open the **Inventory** view. |

**Table 2-13 ▪** Contexts of the Inventory View (cont.)

| Context | Description |
|---|---|
| **All Projects** | Shows all published inventory across all projects in your Code Insight system. This context is helpful in visualizing trends in your company's use of open- source and third-party code in its software projects. |
| **Selected Project** | Shows all published inventory for a selected Code Insight project. Since projects represent versions of a particular software product, this view allows you to see all inventory items for that product. Furthermore, you can also opt to list the inventory for all child projects of the selected project. These child projects represent modules used by your top-level product. You can directly link to the inventory item of the child project or to the child project itself. You can also view the parent hierarchy of the child project to understand the provenance of the inventory items.<br>See Including the Inventory of Child Projects on the Inventory View for details. |

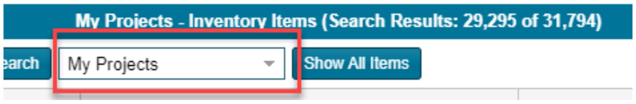
When you switch to a different context, any filtering criteria that you have selected on the **Advanced Inventory Search** dialog remains in effect. See Filtering by Inventory Details for more information about setting this criteria.

## Switching the Context of the View

Use the following procedure to switch the major context of the **Inventory** view.

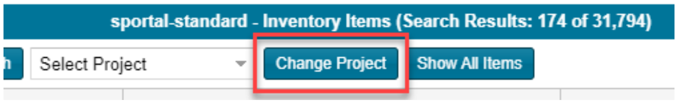
***Task*** ***To switch the context of the Inventory view, do the following:***

1. Open the **Inventory** view. See Opening the Inventory View.
2. In the dropdown list at the top of the view, select the context to which you want to switch the view.


My Projects - Inventory Items (Search Results: 29,295 of 31,794)
earch
My Projects
Show All Items

- If you select **All Projects** or **My Projects**, the list is refreshed with the inventory for the appropriate Code Insight projects.
- If you choose **Select Project**, a dialog is displayed from which to select a project. Once you choose the project, the inventory list is refreshed with the inventory for that project only.

  To switch to a different project, click the **Change Project** button that is displayed next to the dropdown, and choose another project.


sportal-standard - Inventory Items (Search Results: 174 of 31,794)
h
Select Project
Change Project
Show All Items

When select to focus the **Inventory** view on a specific project, you have the option to include the inventory of that project's child projects, if any exist. See Including the Inventory of Child Projects on the Inventory View.

# Including the Inventory of Child Projects on the Inventory View

When you have selected to focus on a specific Code Insight project in the **Inventory** view, you have the option to include the inventory for the child projects of the selected project.

A *child project* is a project dependent on another project. For example, the project for an application that your company is developing might be associated with modules for which you have created separate projects to track the specific OSS and third-party components found in their codebases. To help you keep track of the project relationships, Code Insight lets you identify these dependent projects, such as those for the modules, as child projects to the main application project (called a *parent project*). For more information about creating a project hierarchy, see Identifying Child Projects for a Project.

If you have identified child projects for the project currently selected for the **Inventory** view, you can include the inventory for the child projects in the view. In this way, you can examine the inventory found across the project codebases for all parts of your software project, including its dependencies and sub-modules.

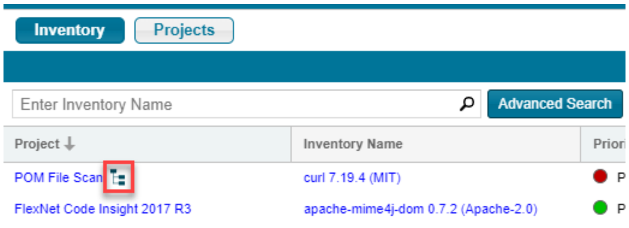
Note that by selecting to include inventory from child projects, all child projects associated with the current top-level project will be recursively included in your inventory items list.

***Task*** ***To include inventory for child projects of the project selected for the Inventory view, do the following:***

1. Open the **Inventory** view. See Opening the Inventory View.
2. In the dropdown list at the top of the view, choose **Select Project** and then choose the project on whose inventory you want to focus the view.
3. Select **Include Inventory items for child projects** in the top right of the **Inventory** view.

   The view is refreshed to include the inventory of all child projects (recursively) of the project in focus.

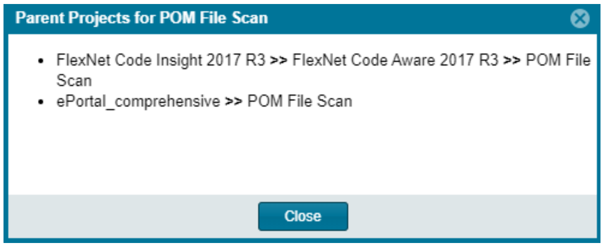
   In the entry for each inventory item of a child project, the hierarchy icon is displayed next to the project link in the **Project** column.



| Project ↓ | Inventory Name | Priori |
|---|---|---|
| POM File Scan | curl 7.19.4 (MIT) | ● P |
| FlexNet Code Insight 2017 R3 | apache-mime4j-dom 0.7.2 (Apache-2.0) | ● P |

4. To view a hierarchy description of each parent of the child project, click .

   The following shows an example of the hierarchy description for child project with two parents.



Parent Projects for POM File Scan

- FlexNet Code Insight 2017 R3 >> FlexNet Code Aware 2017 R3 >> POM File Scan
- ePortal_comprehensive >> POM File Scan

# Refining the Inventory View

You can use the following methods to refine the inventory on the **Inventory** view to pinpoint the data you want to examine:

- Filtering by Inventory Name
- Filtering by Inventory Details
- Focusing Column Content
- Removing All Filters

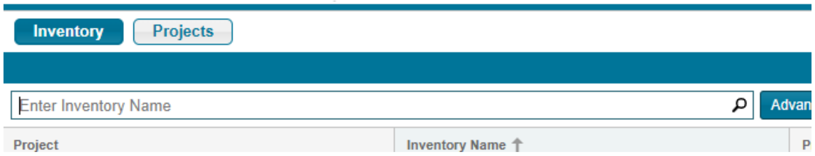
For instructions on accessing the **Inventory** view, see Opening the Inventory View.

## Filtering by Inventory Name

You can filter the inventory on the **Inventory** view by the name of the inventory item.

***Task*** ***To filter inventory by the inventory name, do the following:***

In the **Enter Inventory Name** field at the top of the **Inventory** view, enter a string by which to search inventory names.



If necessary, click the search icon next to the field to initiate search.

To remove the string and restore the full list of inventory items, click the **X** in the field.

## Filtering by Inventory Details

The **Inventory** view enables you to filter inventory by details that include:

- Inventory attributes such as inventory name, priority, age, review status, confidence level, and notifications
- Attributes of the tasks, security vulnerabilities, and licenses associated with inventory
- Attributes of security vulnerabilities associated with inventory

The filter criteria you set persist if you change the major context of the **Inventory** view (see Switching the Context of the Inventory View).

**Task** ***To filter the inventory by inventory details, do the following:***

1. Click the **Advanced Search** button at the top of the **Inventory** view. The **Advanced Inventory Search** dialog is displayed.
2. Select criteria by which to filter the inventory on the view. For details about the criteria fields, see Advanced Inventory Search Dialog.

   Once you have set the criteria, the **Inventory** view is refreshed to apply the filters.

*Note ▪ If you select to filter by license or security-vulnerability criteria on the* ***Advanced Inventory Search*** *dialog, the filtering process might take longer than usual.*

## Focusing Column Content

Focus the column content as needed on the **Inventory** view to see the values most pertinent to you.

**Task** ***To focus column content, do the following:***

1. Hover over the right side of any column header, and click its dropdown menu.
2. From the menu, do any of the following:
   - Re-sort the values in the column in ascending or descending order.

     *Note ▪ Currently you can re-sort the values in the* ***Project***, ***Inventory Name***, ***Priority***, ***#Files*** *(hidden by default),* ***Status***, *and* ***Created On*** *columns.*
   - Select **Columns** to display or hide any columns in the view.

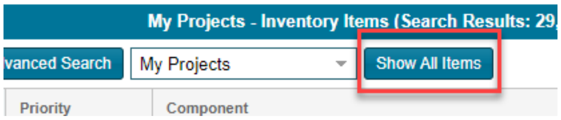
Your column configuration persists when you change the major context of the **Inventory** view (see Switching the Context of the Inventory View).

## Removing All Filters

The following procedure removes all current criteria configured on the **Advanced Inventory Search** dialog and switches the context of the **Inventory** view to show all projects.

**Task** ***To remove all filters and switch to the All Projects context, do the following:***

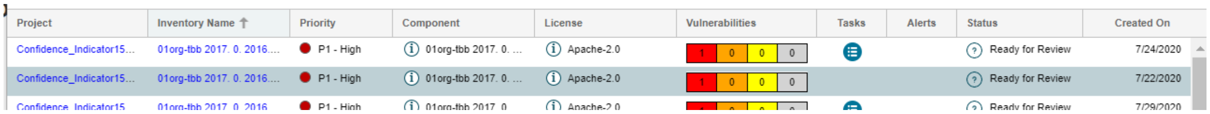
Click the **Show All Items** button at the top of the **Inventory** view.



***Note*** ▪ *This button does not display if the* ***Inventory*** *view is already using the* ***All Projects*** *context.*

# Viewing Inventory Properties and Linking to Additional Information

The **Inventory** view highlights the important properties for the inventory—the project to which the inventory is associated, the number of known security vulnerabilities, inventory review status, whether the inventory has open task or alerts, and more—enabling you to quickly review the listed inventory items for areas concern or interest. Some properties include links to additional information, enabling you to explore an inventory item in more depth.



| Project | Inventory Name | Priority | Component | License | Vulnerabilities | Tasks | Alerts | Status | Created On |
|---|---|---|---|---|---|---|---|---|---|
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016.... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | ☰ | | ? Ready for Review | 7/24/2020 |
| Confidence_Indicator15... | 01org-tbb 2017. 0. 2016.... | ● P1 - High | ⓘ 01org-tbb 2017. 0. ... | ⓘ Apache-2.0 | 1 0 0 0 | | | ? Ready for Review | 7/22/2020 |
| Confidence_Indicator15 | 01org-tbb 2017. 0. 2016 | ● P1 - High | ⓘ 01org-tbb 2017. 0 | ⓘ Apache-2.0 | | ☰ | | ? Ready for Review | 7/29/2020 |

For example, the **Tasks** property provides a link to view and edit any open tasks for an inventory item. The **Component** and **License** properties can link to more details about the component or license as found in Code Insight's data library of third-party and OSS component information. The **Vulnerabilities** and **Alerts** properties can link directly to the CVSS information pertaining to any security issues associated with an inventory item.

Additionally, for a more comprehensive information of a given inventory item, you can click within the row for a given inventory item on the **Inventory** view to open a read-only slide-out panel of the item's details. Alternatively, links are provided to directly access the project with which an inventory item is associated. Then, from within the project, you can explore information gathered for the specific inventory item (and for all inventory in the project) and edit this information as necessary according to your permissions. For more information, see the following topics:

- Opening a Read-Only Version of Inventory Details on the Inventory View
- Opening the Project Associated with Inventory from the Inventory View

For a complete description of the inventory properties and associated links available on the **Inventory** view, see Inventory View.

# Opening a Read-Only Version of Inventory Details on the Inventory View

If you want to examine a read-only version of the details for a given inventory item on the **Inventory** view, use the procedure described in this section. The inventory details are displayed on a slide-out panel within the **Inventory** view, providing an easy means of obtaining information about the inventory item without having to open the project link on the view (although links are available on the slide-out, enabling you to access the project if necessary).

Alternatively, you can open the links to the project directly from the **Inventory** view to view and edit inventory, as described in Opening the Project Associated with Inventory from the Inventory View.

***Task*** ***To open a read-only view of the details for a given inventory item, do the following:***

1. Open the **Inventory** view. For instructions, see Opening the Inventory View.
2. Click within the row of the inventory item whose details you want to view. (Click anywhere within the row except on linked text or a linked icon.)

   A slide-out panel is displayed, showing the most of the inventory tabs and details that are also available for the inventory item on its **Project Inventory Details** pane in the actual project. (Note that the **Component Details** tab is not available on the slide-out.) Unlike the **Project Inventory Details** pane, you cannot edit detail values on the slide-out.

   While the slide-out details are read-only, certain values are hyperlinked, enabling you to still explore and maintain the inventory item (as your permissions allow). For example, you can click the **Project** name link to open the **Project Inventory** tab in the actual project, where you can access and edit any inventory in the project. Or you can click the inventory item's **Name** link to open the item's **Project Inventory Details** pane in project, where you can actually edit the inventory item. Links are also available to open and maintain existing tasks and alerts for the inventory item and to access the item's **Provenance**, **Workflow**, and component **URL** sites.

   For a description of all available inventory details, see Project Inventory Details Pane.
3. Once you have completed examining the details, click the gray area behind the slide-out (or ⊗) to close the slide-out.

# Opening the Project Associated with Inventory from the Inventory View

From the **Inventory** view, you can open the project to which a given inventory item on the **Inventory** view is associated. Within the project, you can then view and edit details for the selected inventory item, edit other inventory, and perform project-related functions as your permissions allow. To open a project, use either procedure:

- Open the Associated Project Directly to the Details for a Given Inventory Item
- Open the Associated Project to the List of All Inventory in the Project

The project opens to the appropriate location on its **Project Inventory** tab. However, once on the **Project Inventory** tab, you can navigate anywhere in the project and perform any function for which you have user permissions.

Alternatively, if you simply want to examine the details of a given inventory item on the **Inventory** view, you can open a read-only version of the details within the view instead of opening the associated project. See Opening a Read-Only Version of Inventory Details on the Inventory View.

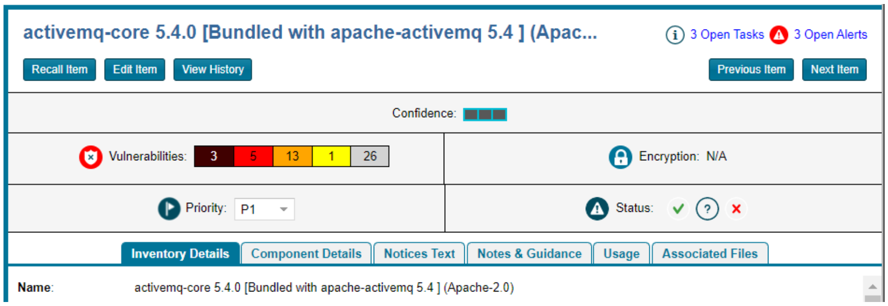
## Open the Associated Project Directly to the Details for a Given Inventory Item

For a given inventory item on the **Inventory** view, you can open its associated project directly to the **Project Inventory Details** pane for that inventory item, where you can then examine and edit the item's details.

***Task*** ***To open the project directly to the details for a given inventory item, do the following:***

1. Open the **Inventory** view. For instructions, see Opening the Inventory View.
2. For the given inventory item on the **Inventory** view, click its hyperlinked inventory name (in the **Inventory Name** column).

   The project opens to the **Project Inventory Details** pane for the specific inventory item (see the following example) on the **Project Inventory** tab, providing access to all information for the item and enabling you update this information according to your user permissions. For more information, see Project Inventory Details Pane.



activemq-core 5.4.0 [Bundled with apache-activemq 5.4 ] (Apac...

Confidence: ▆▆▆

Vulnerabilities: 3 | 5 | 13 | 1 | 26

Encryption: N/A

Priority: P1

Status: ✓ ? ✗

Name: activemq-core 5.4.0 [Bundled with apache-activemq 5.4 ] (Apache-2.0)

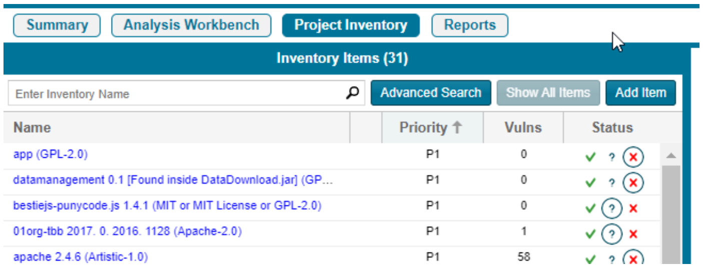
3. To return to the **Inventory** view, click the **Inventory** button under the Code Insight logo on the screen.

## Open the Associated Project to the List of All Inventory in the Project

For a given inventory item on the **Inventory** view, you can open its associated project directly to the inventory list on the **Project Inventory** tab, where you have access to all published inventory in the project, including the given inventory item, and to the project and its functionality. From here, you can edit any inventory and the project as your permissions allow.

***Task*** ***To open the project associated with a given inventory item to access information for all published inventory in the project, do the following:***

1. Open the **Inventory** view. For instructions, see Opening the Inventory View.
2. For the inventory item on the **Inventory** view, click its hyperlinked project name (in the **Project** column). The project opens directly to the **Inventory Items** list on the **Project Inventory** tab (see the following example), enabling you to access and update details for any published inventory item in the project according to your user permissions. You also have access to the entire project to perform any project-related function as needed. For more information, see Project Inventory Tab.



Inventory Items (31)

| Name | Priority ↑ | Vulns | Status |
|---|---|---|---|
| app (GPL-2.0) | P1 | 0 | ✓ ? ✗ |
| datamanagement 0.1 [Found inside DataDownload.jar] (GP... | P1 | 0 | ✓ ? ✗ |
| bestiejs-punycode.js 1.4.1 (MIT or MIT License or GPL-2.0) | P1 | 0 | ✓ ? ✗ |
| 01org-tbb 2017. 0. 2016. 1128 (Apache-2.0) | P1 | 1 | ✓ ? ✗ |
| apache 2.4.6 (Artistic-1.0) | P1 | 58 | ✓ ? ✗ |

3. To return to the **Inventory** view, click the **Inventory** button under the Code Insight logo on the screen.

# Accessing Projects in Code Insight

This section describes the various ways you can access projects in your Code Insight system in order to view their details and manage them:

- Opening the Projects View
- Showing Only Your Projects
- Searching Across All Projects the System
- Using the Project Dashboard
- Opening a Project
- Managing Items in the Projects Display

## Opening the Projects View

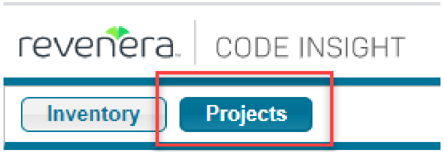
All Code Insight projects are created, accessed, and managed in the **Projects** view. This view shows a manageable list of the projects currently available in the system. From this list you open individual projects to assess scan their results, edit their details, and finalize their inventory of open-source and third-party software.

Use the following procedure to open the **Projects** view. The procedure assumes that you have logged into Code Insight.

***Task*** ***To open the Projects view, do the following:***

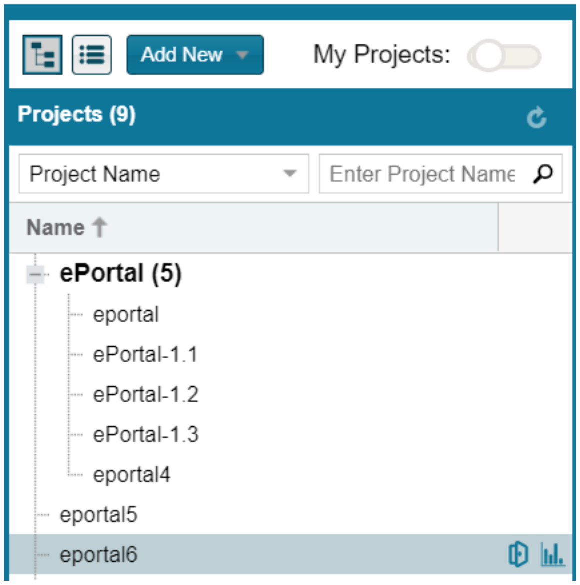
Open the **Projects** view using one of these methods:

- From the Code Insight dashboard (which is displayed when you start Code Insight), click **go to project**. See Opening Code Insight for details on accessing this dashboard.
- From any location in the Code Insight Web UI, click the **Projects** button under the Code Insight logo:



- Click the ☰ icon in the upper right corner of the Code Insight Web UI to open the Code Insight main menu. Select **PROJECTS** from the menu.

The **Projects** pane on the left side of the **Projects** view lists the projects in Code Insight. (The projects are listed in a tree or list format, depending on your configuration.)



From this view you can do the following (depending on your project role in some cases):

- Filter the projects by those with which you are associated as Project Contact or through a project role (see Showing Only Your Projects).
- Filter the projects by project name, name of associated inventory, or security vulnerability, as described in Searching Across All Projects the System.
- View the Code Insight scan statistics for a specific project (see Using the Project Dashboard).
- Create or delete projects (Creating a Project or Deleting a Project).
- Toggle the **Projects** pane between tree view and list view (see Managing Items in the Projects Display).
- Move projects to different folders (Managing Items in the Projects Display).
- Create, delete, and move folders (Managing Items in the Projects Display).
- Open a project to manage it, assess scan results, and finalize its inventory (see Opening a Project).

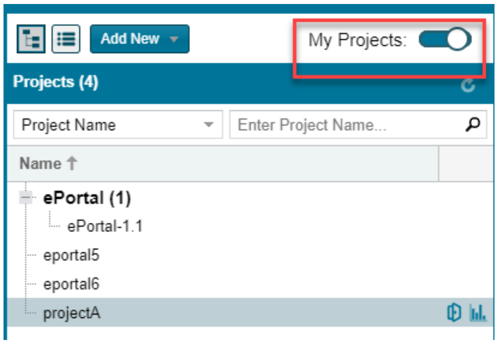
# Showing Only Your Projects

Code Insight provides the option to filter the list of projects to show only those projects with which the current user is associated as either Project Contact, Project Administrator, Analyst, Reviewer, or Observer. (For a description of these roles, see Assigning and Removing Project Users.)

Additionally, this filter can work in conjunction with the system filters described in Searching Across All Projects the System to display only your projects that have specific project or inventory attributes.

***Task*** ***To show your projects only, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. At the top of the **Projects** pane, click the **My Projects** toggle.



3. Select a project from the filtered list of projects to open its dashboard. See Showing Only Your Projects for details.

   Alternatively, open the project to view its inventory on the **Project Inventory** tab. See Opening a Project for details.

4. (Optional) To turn off this filter, click the **My Projects** toggle again.

# Searching Across All Projects the System

Using the search filters available in the **Projects** pane, you can search projects across the Code Insight system by name, inventory, components and versions, licenses, and security vulnerabilities. You can perform multiple types of these global searches, each one filtering deeper into the current search results. Then, at the project level, you can continue to narrow your search results to specific inventory items for the given project.

The following topics describe these search methods:

- Available Filters for Searching Across Projects
- Searching for Projects by Name
- Searching All Projects for Inventory Based on a Specific Component and Version
- Searching All Projects for Inventory Associated with a Specific License
- Searching All Projects for a Security Vulnerability Advisory
- Restoring the Full Project Tree or List

## Available Filters for Searching Across Projects

The following filters are available from the **Projects** pane to perform searches across all projects in your Code Insight system.

**Table 2-14 •** Available Filters for Searching Across Projects

| To search across all projects for... | ...use this filter | Filter criterion | Criterion example | Refer to... |
|---|---|---|---|---|
| **Projects with a name containing a specific string** | Project Name | Project name | `MyProject` | Searching for Projects by Name |

**Table 2-14 ▪** Available Filters for Searching Across Projects (cont.)

| To search across all projects for... | ...use this filter | Filter criterion | Criterion example | Refer to... |
|---|---|---|---|---|
| **Inventory based on a specific component and version** | Project Inventory | Component and version as it appears in the **Inventory Name** value | `Apache Struts 2.3.14.3` | Searching All Projects for Inventory Based on a Specific Component and Version |
| | | Component name as it appears on the **Component Details** tab for the inventory item | `struts2-core` | |
| **Inventory associated with a specific license** | Project Inventory | The **Selected License** value as it appears on the **Component Details** tab for the inventory item | `GNU General Public License v2.0` | Searching All Projects for Inventory Associated with a Specific License |
| | | The license name as it appears in the **Inventory Name** value | `GNU General Public License` or `GPL-2.0+` | |
| | | The SPDX short identifier for the license | `GPL-2.0+` | |
| **Inventory impacted by a specific security vulnerability** | Security Vulnerability | The complete ID of the security vulnerability | `CVE-2018-11776` for an NVD vulnerability<br>`SA40575` for a Secunia advisory<br>`DSA-4315` for a Debian advisory | Searching All Projects for a Security Vulnerability Advisory |

# Searching for Projects by Name

You can use the **Project Name** search filter available in the **Projects** pane to search for a projects by a full or partial name.

## Search Rules

When you search for projects by project name, the following rules apply:

- The name string value you enter is case-insensitive.
- All characters in the search string must be consecutive.
- A full or partial string value is supported as a search criterion.
- The string can contain any characters (letters, numbers, and special characters).

### Searching for Projects by Name

This procedure shows how to search for projects by a full or partial name string.

***Task*** ***To search projects by name, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. At the top of the **Projects** pane, select **Project Name** from search dropdown on the left.
3. Enter the project name (or a partial string in the name) in the **Enter Project Name** field. The list of projects changes to reflect the search results, and a filtered count (for example, "(19 of 123)") is provided in the **Projects** pane header to show the number of projects returned by the search.

   If no inventory items meet the specified criterion, the **Projects** pane shows "No Projects".
4. Select a project from the filtered list to open its dashboard. See Showing Only Your Projects for details.

   Alternatively, open the project to view its inventory displayed on its **Project Inventory** tab. See Opening a Project for details.

## Searching All Projects for Inventory Based on a Specific Component and Version

You can use the **Project Inventory** filter to search for those projects whose inventory contains items based on a specific component and version. The search returns a filtered list of projects; and, when a project in the list is opened, its inventory is also filtered by the search criterion.

This search method saves you time in locating specific inventory items that require attention across all projects. For example, you might also use this search method to pinpoint those projects containing inventory items affected by a recent component upgrade.

You can also use this search method to easily locate projects that contain inventory items impacted by a security vulnerability whose exact ID you do not know; you can search instead for projects with inventory based on a component and version known to be affected by the vulnerability. (This search method is an alternative to using the **Security Vulnerability** filter, which requires the exact vulnerability ID as the search criterion. See Searching All Projects for a Security Vulnerability Advisory.)

### Search Rules

When you search projects for inventory based on a specific component and version, the following rules apply.

- A full or partial string value is supported as search criterion.
- All characters in the search string must be consecutive.
- The string value is case-insensitive and can contain letters, numbers, hyphens, and special characters. Spaces are also allowed.
- Only inventory items on the **Project Inventory** tab are supported in the search.

### Searching All Projects by Component and Version

Use this procedure to locate projects with inventory based on a specific component and version.

***Task*** ***To search all projects for inventory based on a specific component and version, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. At the top of the **Projects** pane, select **Project Inventory** from search dropdown on the left.
3. In the **Enter Search Criteria** field, specify the complete name or a partial string for one of the following:
    - The name of the component as it appears on the **Component Details** tab (for example, `struts2-core`) for an inventory item
    - The name of the component and version as it appears in the **Inventory Name** value (for example, `Apache Struts 2.3.14.3`)

    The list of projects changes to reflect the search results, and a filtered count (for example, “(19 of 123)”) is also provided in the **Projects** pane header to show the number of projects returned by the search.
4. Open one of the projects to see a filtered list of inventory items that contain the component or component and version. (See Opening a Project for details.)

## Searching All Projects for Inventory Associated with a Specific License

You can use the **Project Inventory** filter to search for those projects containing inventory items associated with a specific license. The search returns a filtered list of projects; and, when a project in the list is opened, its inventory is also filtered by the search criterion.

This search method saves you time in locating inventory items with license-related issues, such as those items that are associated with a high-risk license, across all projects.

### Search Rules

When you search projects for inventory associated with a specific license, the following rules apply:

- A full or partial string value is supported as search criterion.
- All characters in the search string must be consecutive.
- The string value is case-insensitive and can contain letters, numbers, hyphens, and special characters. Spaces are also allowed.
- Only inventory items on the **Project Inventory** tab are supported in the search.

### Searching All Projects by License

Use this procedure to locate projects with inventory associated with a specific license.

*Task* ***To search all projects for inventory associated with specific license, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. At the top of the **Projects** pane, select **Project Inventory** from search dropdown on the left.
3. In the **Enter Search Criteria** field, specify the complete name or a partial string for one of the following:
    - The name of the **Selected License** as it appears on the **Component Details** tab (for example, `GNU General Public License v2.0`) for an inventory item
    - The license SPDX short identifier (for example, `GPL-2.0+`)
    - The license name as it appears in the **Inventory Name** value (for example, `GNU General Public License` or `GPL-2.0+`)

    The list of projects changes to reflect the search results, and a filtered count (for example, "(19 of 123)") is also provided in the header on the **Projects** pane to show the number of projects returned by the search.
4. Open one of the projects to see a filtered list of inventory items that contain the license. (See Opening a Project for details.)

# Searching All Projects for a Security Vulnerability Advisory

You might find it sometimes necessary to quickly see how a specific security vulnerability impacts your organization. You can search the system for a security vulnerability or advisory in one of the following ways:

- **If you know the exact ID of the security vulnerability or advisory**—Use the **Security Vulnerability** search filter with the *exact* security vulnerability ID as the search criterion, as described in this section.
- **If you do not know the ID of the security vulnerability or advisory**—Use the **Project Inventory** search filter to provide the name of the vulnerable component as the search criterion. See Searching All Projects for Inventory Based on a Specific Component and Version for details.

    ***Note*** *▪ A vulnerability or advisory might not have an ID, for example, in the case of a zero-day vulnerability for which an ID has not been published.*

## Search Rules

When you use the **Security Vulnerability** search filter to search projects associated with a specific security vulnerability, the following rules apply:

- Only one vulnerability ID can be specified as a search criterion.
- Only exact matches of the full vulnerability ID string are supported. Partial strings are not supported.
- The string you enter does not support spaces.
- Only published inventory items are searched.
- The search ignores inventory associated with a component version for which the vulnerability has been suppressed.

- The search does not validate the vulnerability ID you enter. If you enter an invalid ID, no results are returned in the **Projects** pane.

### Searching for a Security Vulnerability

Use this procedure to locate projects by the *exact* security vulnerability ID you specify.

***Task*** ***To search for projects affected by a specific security vulnerability, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. At the top of the **Projects** pane, select **Security Vulnerability** from search dropdown on the left.
3. In the **Enter Vulnerability ID** field, specify the complete ID of the vulnerability (for example, `CVE-2018-11776` for an NVD vulnerability).
4. Press Enter. The list of projects changes to reflect the search results, and a filtered count (for example, "(19 of 123)") is also provided in the header on the Project pane to show the number of projects returned by the search.

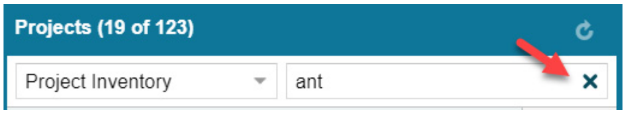
   If no inventory items meet the specified criterion, the **Projects** pane shows "No Projects".
5. Open one of the projects to see a filtered list of inventory items that are impacted by the security vulnerability. (See Opening a Project for details.)

## Restoring the Full Project Tree or List

Use this step to remove the current filter in effect in **Projects** pane to restore all projects in the project tree or list.

***Task*** ***To restore the full list of projects, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. Click the ✕ icon in the criterion field to remove the current **Projects** pane filter:

   Projects (19 of 123)

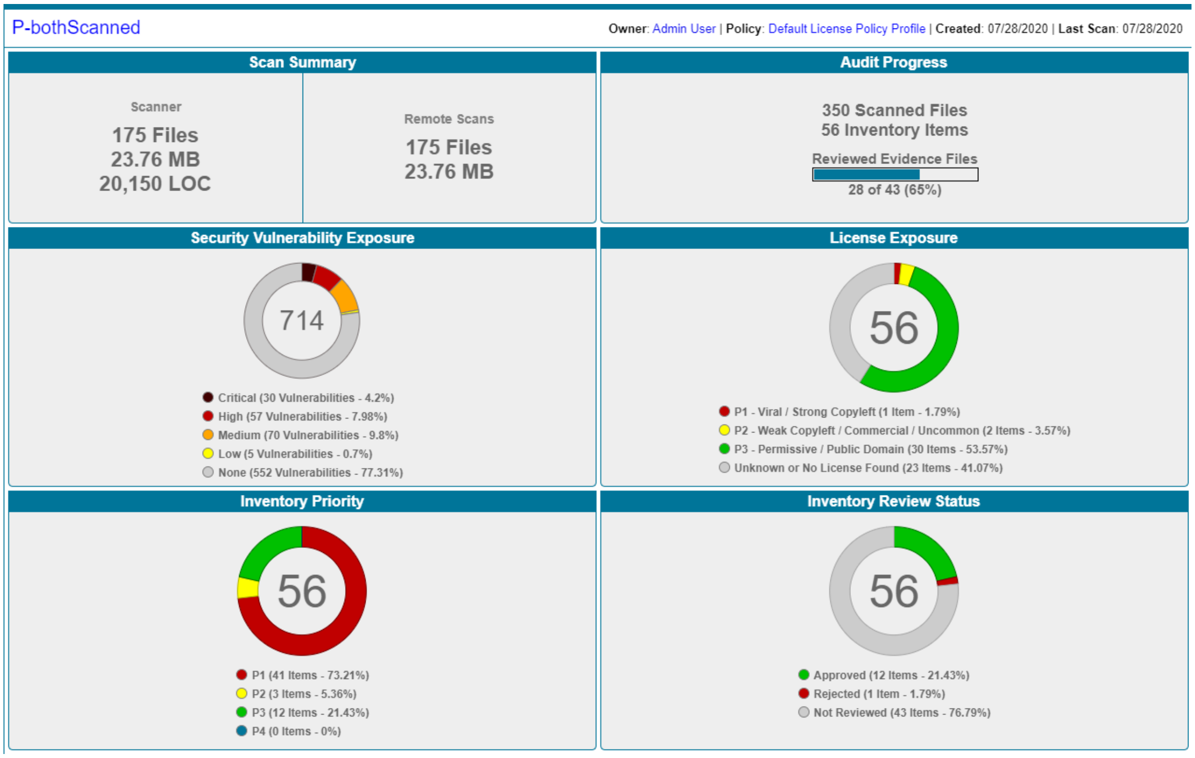
   Project Inventory ▾ | ant ✕

   The project tree or list is restored to show all projects.

# Using the Project Dashboard

When you select a project in the **Projects** pane, the project's dashboard is displayed, providing you with an interactive view of your project, including security-vulnerability and license exposure, codebase and inventory review statistics, and other information.

The procedure that follows this image describes how to use project dashboard features.



P-bothScanned

**Owner**: Admin User | **Policy**: Default License Policy Profile | **Created**: 07/28/2020 | **Last Scan**: 07/28/2020

**Scan Summary**

Scanner
175 Files
23.76 MB
20,150 LOC

Remote Scans
175 Files
23.76 MB

**Audit Progress**

350 Scanned Files
56 Inventory Items
Reviewed Evidence Files
28 of 43 (65%)

**Security Vulnerability Exposure**

714

- Critical (30 Vulnerabilities - 4.2%)
- High (57 Vulnerabilities - 7.98%)
- Medium (70 Vulnerabilities - 9.8%)
- Low (5 Vulnerabilities - 0.7%)
- None (552 Vulnerabilities - 77.31%)

**License Exposure**

56

- P1 - Viral / Strong Copyleft (1 Item - 1.79%)
- P2 - Weak Copyleft / Commercial / Uncommon (2 Items - 3.57%)
- P3 - Permissive / Public Domain (30 Items - 53.57%)
- Unknown or No License Found (23 Items - 41.07%)

**Inventory Priority**

56

- P1 (41 Items - 73.21%)
- P2 (3 Items - 5.36%)
- P3 (12 Items - 21.43%)
- P4 (0 Items - 0%)

**Inventory Review Status**

56

- Approved (12 Items - 21.43%)
- Rejected (1 Item - 1.79%)
- Not Reviewed (43 Items - 76.79%)

***Task*** ***To use the project dashboard, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)

2. In the **Projects** pane on the left, click a project in the list of projects. The dashboard for the selected project is displayed in the right panel. (Alternatively, hover over the project entry in the list, and click the **Load Project Dashboard** icon in the entry to display the dashboard.)

   The project dashboard contains the following charts to provide an overview of the project's most recent scan and the resulting inventory:

   - **Scan Summary**—A summary of your most recent **Scanner** scan (that is, server scan) and most recent **Remote Scans**. If multiple scan-agent plugins are used for remote scanning, the **Remote Scans** summary shows totals from the most recent scans of all the agents.

     If only a server scan has occurred, the tile shows only **Scanner** totals. Likewise, if only remote scans have occurred, the tile shows only **Remote Scans** totals.

   - **Audit Progress**—For a **Scanner** (server) scan, a snapshot of the audit progress that users have made on those files containing OSS or third-party evidence. No audit progress is shown for remote scans.

     The tile also shows the total number of scanned files and resulting inventory items for the project.

   - **Security Vulnerability Exposure**—An interactive color-coded chart and legend that provides an visual of security vulnerability percentages and totals by severity across all project inventory. The number in the center of the chart is the total number of security vulnerabilities found in the inventory. (The colors in this chart can vary depending on the CVSS version Code Insight is using.)

Additionally, the counts in this chart do not include vulnerabilities are currently suppressed. See Working with Security Vulnerabilities for details.)

- **License Exposure**—An interactive color-coded chart and legend that provide an overview of the licenses by priority across project inventory. The number in the center of the chart is the total number of inventory items identified for the current project.
- **Inventory Priority**—An interactive color-coded chart and legend that provide an overview of the percentages and totals of inventory priorities across all inventory in the current project. For more information about inventory priority, see Inventory Priority.
- **Inventory Review Status**—An interactive color-coded chart and legend that provide a visual of the how many inventory items have been approved, rejected, or not reviewed in the project.

3. (Optional) Hover your mouse cursor over the color-coded segments in the charts to view details related to a given segment. If you want, click a color-coded segment to open the project to view the inventory items associated with the segment. See Filtering Inventory for a Project from the Project Dashboard for details.

   Alternatively, you can open the project to view all its inventory (see Opening a Project), or select another project from the list of projects in the **Projects** pane to view the dashboard for that project.

## Filtering Inventory for a Project from the Project Dashboard

After you create a project, and upload and scan a codebase, you can quickly filter the project's inventory to view potential problems and take steps to eliminate issues, such as high exposure items (shown in red), from your project inventory.

***Task*** ***To quickly filter inventory on the project dashboard, do the following:***

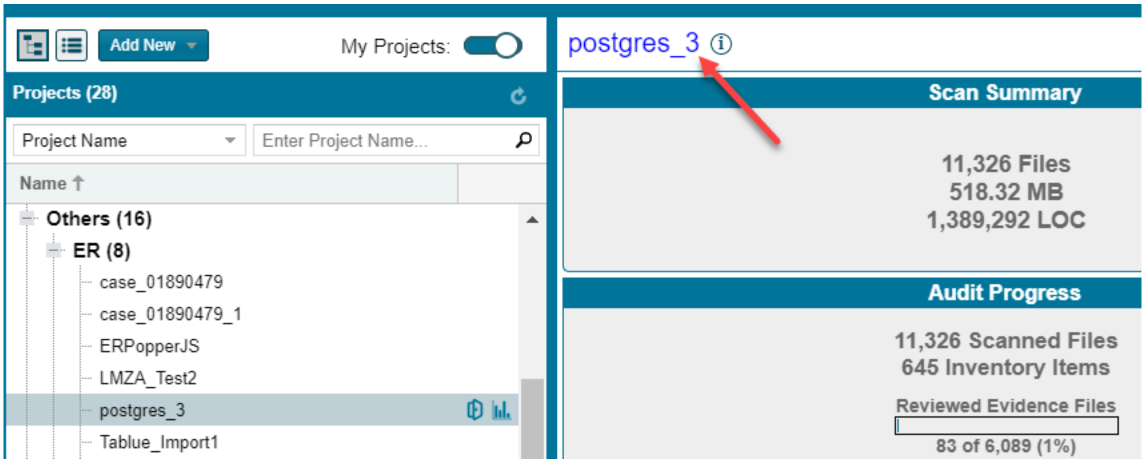
1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. In the **Projects** pane on the left, click a project in the list of projects. The dashboard for the selected project is displayed in the right panel.
3. In the project dashboard, navigate to the desired chart, and click a color-coded area in the chart. The project is opened to its **Project Inventory** tab, displaying the project inventory items associated with the information represented by color-code area. You can also click the corresponding colors in the legends below the charts to filter your inventory.
4. Click on a project inventory item listed to see more detail in the right pane of the **Project Inventory** tab. See Reviewing Published Inventory for a Project for details about this tab.

# Opening a Project

Use this basic procedure to open a project.

*Task* ***To open a project, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. In the **Projects** pane on the left, click a project in the projects display. The dashboard for the selected project is displayed in the right panel.
3. To open the project, either click the **Open Project** icon () next to the project in the projects display, or click the project's name link in the upper left corner of the project dashboard (shown below).



The project is opened on either of the following tabs for the project:

- If the project contains published inventory items, the **Project Inventory** tab. For more information about the **Project Inventory** tab, see Reviewing Published Inventory for a Project.
- If the project does not contain published inventory items, the project's **Summary** tab. For more information about the **Summary** tab, see Managing Projects. For information about publishing inventory, see Publishing or Recalling Inventory from the Analysis Workbench.

Note that the **Analysis Workbench** tab is also available for users with the proper permissions (although you have to navigate to open it). The Analysis Workbench enables a user to perform a deep analysis of the scan results. For more information, see The Analysis Workbench Layout.

# Managing Items in the Projects Display

The following procedures describe how to manage the display of available projects in Code Insight:

- Access the Display of Projects
- Select the Projects Display Format
- Manage Items in the Project Tree Format
- Manage Items in the Plain List Format

## Access the Display of Projects

This procedure describes how to access and to manage the display of available Code Insight projects.

***Task*** ***To access the projects display, do the following:***

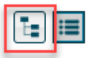
Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.) The display of projects is in the **Projects** pane on the left.

## Select the Projects Display Format

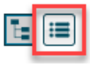
The following describes how to toggle between project tree or plain list format for the project list.

***Task*** ***To select a format for the projects display, do either:***

To display the projects in a project tree format, click this icon at the top of the **Projects** pane:



To display the projects in a plain list format, click this icon:



## Manage Items in the Project Tree Format

The following describes ways to manage items (projects and folders) in the project tree format. The right-click menu options described in this section are enabled or disabled depending on the permissions granted the current user.

***Task*** ***To manage the items in the project tree, do the following:***

Perform any of these tasks on an item in the project tree:

- **Rename a project** —(Project Administrator only) Double-click the project name and overwrite the current name with the new name.
- **Move a project to a different folder**—(Project Administrator only) Drag and drop the project to the desired folder.
- **Delete a project**—(Project Administrator only) Right-click the project and select **Delete Project**. See Deleting a Project for further details.
- **Create a project**—(User with Create Project permission) To create a project under a specific folder, right-click that folder or any project directly under that folder, and select **Create New Project | At This Level**. To create a project at the root level of the project tree, right-click anywhere in the tree, and select **Create New Project | At Root Level**.

  See Creating a Project for further details.

- **Create a folder**—(User with Create Project permission) To create a folder under a specific folder, right-click that folder or any project directly under that folder, and select **Create New Folder | At This Level**. To create a folder at the root level of the project tree, right-click anywhere in the tree, and select **Create New Folder | At Root Level**.
- **Delete a folder**—(User with Create Project permission) Right-click the folder and select **Delete Folder** (or click ✕ to the right of the folder). Any sub-folders under this folder are also deleted. Any projects under the deleted folder are moved to the parent folder or to the root of the project tree.

## Manage Items in the Plain List Format

The following describes ways to manage items (projects and folders) in the plain list format.

*Task* ***To manage a project list in plain list format, do the following:***

Perform any of these tasks on an item in the project list:

- **Rename a project** —(Project Administrator only) Double-click the project name and overwrite the current name with the new name.
- **Delete a project**—(Project Administrator only) Right-click the project and select **Delete Project**. See Deleting a Project for further details.
- **Create a project**—(User with Create Project permission) Right-click any project in the list, and select **Create a New Project**. See Creating a Project for further details.

# Managing Projects

The following topics describe how to use features on the **Summary** tab to manage the currently opened project. (Refer to the Code Insight User Roles and Permissions appendix for the various user roles required manage projects.)

- Opening the Project Summary Tab
- Assigning and Removing Project Users
- Editing the Project Definition and General Settings
- Setting Policies to Publish Inventory Automatically
- Updating Scan Settings for a Project
- Updating Inventory Review and Remediation Settings for a Project
- Connecting the Project to Remote Data Sources
- Identifying Child Projects for a Project
- Changing the Project Contact
- Rescanning Your Codebase (Server Scans Only)
- Exporting Project Data
- Importing Project Data

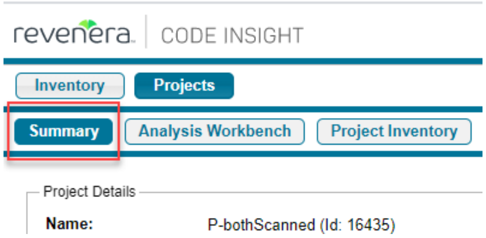
- Branching a Project
- Deleting a Project
- Creating a Private Project

# Opening the Project Summary Tab

The **Summary** tab for a given project displays important information about the project and provides access to the functionality used to manage the project.

***Task*** ***To open the Summary tab for a given project, do the following:***

1. Open a project in the **Projects** view. (For instructions, see Opening a Project.)
2. Click the **Summary** button at the top of the window to open the **Summary** tab for the project.



Project Details

Name: P-bothScanned (Id: 16435)

For a description of the fields and functionality available on the **Summary** tab, see Summary Tab.

# Assigning and Removing Project Users

The Project Administrator assigns roles to users, enabling them analyze the codebase and manage and publish inventory, review published inventory, or view private projects. The Project Administrator can also create other Project Administrators. The following are the available roles that users can have in a project:

- **Project Administrators** manage project users, manage project settings, upload and scan codebases, and rename, branch, and delete projects. They also manage Manage Source Control Management (SCM) and Application Lifecycle (ALM) instances.
- **Analysts** manage the codebase and inventory using the Analysis Workbench. They can upload and scan codebases, review files and add files to inventory, create new inventory, edit existing inventory, and publish and recall inventory.
- **Reviewers** use the Project Inventory tab to approve and reject inventory, recall inventory, set inventory priority, and edit third-party notices and audit/guidance notes for the inventory.
- **Observers** can view inventory in a private project. They have read-only access to project inventory and can run reports. Development managers and executives are usually assigned this role. The Observer role is available for private projects only.

***Note** ▪ Private projects are hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. For additional information about private projects, see Creating a Private Project.*

For a reference to the various user roles and their permissions, refer to the Code Insight User Roles and Permissions appendix.

The following procedure describes how to assign users to project roles and remove users from these roles.

***Task*** ***To assign users to or remove them from project roles, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project Users** from the dropdown menu. The **Edit project users** page appears.

   Note that all users assigned to a given role are shown on this page, whether the user was manually assigned the role or had inherited the role (for example, through project migration, designation as Project Contact, or Project Defaults set up by the System Administrator).
3. Do either of the following:
   - To assign users to a given role, drag and drop one or more user names from the **Select Users** list to the desired “role” pane (**Project Administrators**, **Analysts**, **Reviewers**, or **Observers**). Repeat this step a necessary. (A user can be assigned to multiple roles.)
   - To remove a user from a role, click ✕ next to the user’s name in the appropriate “role” pane. You can remove any user from a role, even those who inherited the role.

   ***Note** ▪ The **Observers** pane is visible for only private projects.*
4. Click **Close** when you have finished managing the project users.

# Editing the Project Definition and General Settings

The Project Administrator can edit the project’s definition and general settings.

***Task*** ***To update project settings, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **General** tab.
4. Update the fields as needed. Refer Edit Project: General Tab to for field descriptions.
5. Click **Save** to save the changes.

# Updating Scan Settings for a Project

You can update the scan configuration by switching the project to a different scan profile, update which subfolders to scan, and change settings for automatically publishing inventory during the scan (see Setting Policies to Publish Inventory Automatically).

See also the Edit Project: General Tab to configure the project setting that determines whether the scan retains inventory that has no files associations.

*Task* ***To update scan settings for the project, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **Scan Settings** tab.
4. Update the fields as needed. Refer Edit Project: Scan Settings Tab for field descriptions.
5. Click **Save** to save the changes.

# Setting Policies to Publish Inventory Automatically

Code Insight provides the ability to automatically publish inventory without the need for an analyst to be involved. This feature supports a fully automated end-to-end process where there is no human analyst involvement. (For example, the auto-publish feature works in conjunction with workflow policies that automatically review inventory items as they are published, as described in Managing Policy Profiles.) If there is a human analyst involved, the auto-publish feature can be turned off, allowing the analyst to publish the inventory manually after analysis.

When the auto-publish feature is enabled, additional options are made available to do the following:

- Set the minimum inventory confidence level for publishing inventory.
- Determine whether to automatically mark files associated with an auto-published inventory as “reviewed”.
- Determine whether to publish inventories with undetermined licenses (that is, their selected **License** value is **I don’t know**).

*Task* ***To set the auto-publish feature, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **Scan Settings** tab.
4. Enable or disable **Automatically publish system-created inventory items**. When you enable this option, additional auto-publish options are made available for configuration. See Edit Project: Scan Settings Tab for a description these options.
5. When you have set the auto-publish feature, click **Save**. The **Summary** tab is opened.

# Updating Inventory Review and Remediation Settings for a Project

You can overwrite default settings that configure the automation of the review, remediation, and status notification processes for published inventory in your project. These settings, which work in conjunction with the set of policies in the project's policy profile, are used to set up the following in your project:

- The policy profile to associate with the project. The policies in the selected profile work in conjunction with the review, remediation, and notification configuration defined on this tab.
- Automatic creation of manual review tasks for inventory items not reviewed by policy during publication performed as part of a scan. The tasks are automatically assigned to the default legal or security contact that you specify.
- Automatic creation of remediation tasks and associated external work items for inventory that is rejected either automatically by policy or during manual publication by an analyst. The tasks are automatically assigned to the default engineering contact that you specify.
- Automatic rejection of published inventory impacted by new vulnerabilities detected in the latest scan or Electronic Update.
- The automatic generation of email notifications only (instead of assigned tasks), which are sent to the Project Contact as alerts concerning the rejected or non-reviewed published inventory items.

*Task* ***To update settings that automate review, remediation, and status notification processes for published inventory, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **Review and Remediation Settings** tab.
4. Update the fields as needed. Refer Edit Project: Review and Remediation Settings Tab to for field descriptions.
5. Click **Save** to save the changes.

# Connecting the Project to Remote Data Sources

If your system is configured to connect to a remote data source, you will have access to update the following:

- Version Control Settings
- ALM Settings

## Version Control Settings

Use the **Version Control Settings** tab on the **Edit Project** page to synchronize one or more Source Code Management (SCM) repositories to the Scan Server for your project so you can scan and audit code without manually moving that data to the server. For information about connecting to a remote data sources, see the Configuring Source Code Management chapter.

# ALM Settings

Code Insight integrates with application lifecycle management (ALM) systems, enabling Code Insight users to create and manage external work items associated with inventory directly from Code Insight. In this way, inventory requiring further review and remediation can be tracked externally as part of the user's existing issue-tracking system.

For example, a Code Insight scan might uncover security vulnerabilities or "copyleft" licenses requiring further review by the Security and Legal teams. With an ALM integration, these issues can be quickly converted into work items that point to corresponding issues in the ALM instance.

Integration with a specific ALM system is enabled through a corresponding Code Insight connector that supports pre-populated data (in the form of one or more *instances*) used to connect to the ALM system and to set up work items. Additionally, a given ALM instance controls the synchronization of data between Code Insight and the server based on a configured synchronization frequency.

To configure an ALM connector, the Project Administrator defines one or more of these instances in Code Insight, a process described in the "Integrating with Application Life Cycle Management" chapter in the *Code Insight Installation and Configuration Guide*.

Then, in order to create and manage work items for a given project, you must associate the project with a specific ALM instance. The following sections describe how to associate (and unassociate) a project with an ALM instance. Currently, Code Insight is installed with a Jira connector. Future releases will provide additional support for other ALM systems.

- Associating a Jira Instance to a Project
- Using Code Insight Variables
- Unassociating an ALM Instance from a Project

## Associating a Jira Instance to a Project

Use the following instructions to associate a Code Insight project with a Jira instance.

**Task** ***To associate a Jira instance to a project, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **ALM Settings** tab.
4. From the **ALM Instance** dropdown, select the Jira instance to associate to this project. The current settings for the Jira instance are displayed on **ALM Settings** tab.

   If no instances are available in the dropdown, ensure that at least one instance is configured at the application level. Instructions for configuring a Jira instance are found in the *Code Insight Installation and Configuration Guide*.

5. Complete the fields on the **ALM Settings** tab. See the inline help for explanations of the fields.
    - Certain fields might already contain a value based on the global application defaults set when the Jira instance was created (as described in the *Code Insight Installation and Configuration Guide*.) However, you can override any global defaults with the information you enter here. For example, if you change the **Default Issue Type** from **Task** to **Bug**, the value **Bug** becomes the new default for this project. See ALM Tab for field details.
    - You can include (or override) Code Insight variables in the **Default Summary** and **Default Description** fields. These variables will be replaced by actual values in descriptive text that displays for a newly created Jira issue and work item. For more information, see the next section, Using Code Insight Variables.
6. When you have completed the settings, click **Save** to associate the Jira instance to the project.

    Validation for these field values takes place during work item creation. If the information entered here is invalid (for example, the **Assignee** value does not exist in the Jira system), the information will still be saved, but users will not be able to create the work item in the future.

    Once you have associated the Jira instance with the project, all work items created in this project will have a corresponding Jira issue on the provided instance.

## Using Code Insight Variables

The **Default Summary Text** and **Default Description Text** fields support Code Insight variables that communicate details about the Code Insight project, inventory item, and other relevant information in the work item and associated Jira issue.

### Supported Variables

The following table lists the available variables for use in the text entered in the **Default Summary Text** and **Default Description Text** fields:

**Table 2-15 ▪** Supported Code Insight Variables for Use in Work Item Summary and Description Text

| | |
|---|---|
| $PROJECT_NAME | Name of the Code Insight project containing the issue |
| $INVENTORY_ITEM_NAME | Name of the inventory item containing the issue |
| $COMPONENT_NAME | Name of the component associated with the inventory item |
| $VERSION_NAME | Version of the component associated with the inventory item |
| $LICENSE_NAME | Name of the selected license for the inventory item |
| $NUMBER_VULNERABILITIES | Total number of security vulnerabilities associated with the inventory item |
| $NUMBER_FILES | Total number of files associated with the inventory item |

**Table 2-15 ▪** Supported Code Insight Variables for Use in Work Item Summary and Description Text (cont.)

| | |
|---|---|
| $INVENTORY_URL | Link to the inventory item |

When the work item is created, the included variables are replaced by their respective values.

### Example Use of Variables

The following is example text you might enter in the **Default Summary Text** field. The text includes some of the available variables.

```
The $INVENTORY_ITEM_NAME inventory item in the project $PROJECT_NAME contains
$NUMBER_VULNERABILITIES vulnerabilities that require review. Go to $INVENTORY_URL to see the
vulnerable inventory item.
```

If your Code Insight project name is “MySampleProject” and the name of the inventory item name for which you create a work item is “Apache Commons BeanUtils”, the work item and Jira issue will display the following summary:

```
The Apache Commons BeanUtils 1.7.0 (Apache 2.0) inventory item in the project MySampleProject
contains 18 vulnerabilities that require review. Go to https://my.sample.server:8888/codeinsight to
see the vulnerable inventory item.
```

## Unassociating an ALM Instance from a Project

The Project Administrator can unassociate an ALM instance from a project at any time. If the association is removed, any existing work items will remain with the project, but the **Create Work Item** option becomes disabled.

***Task*** ***To unassociate an ALM instance from a project, do the following:***

1. As the Project Administrator, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **ALM Settings** tab.
4. In the **ALM Instance** dropdown, change the selection to **None**.

# Identifying Child Projects for a Project

Code Insight enables you to create and manage project hierarchies as a means to keep track of projects related each other. A project hierarchy is created by simply identifying one or more projects as *child projects* of another project (called the *parent project*). Once the hierarchy is created, links are established in the Code Insight Web UI between the parent project and the associated child projects so that you can easily move between projects to assess scan results and review inventory.

A project hierarchy is useful when your product application contains one or more modules, each with a codebase for which you want to set up a separate project to track and assess its open-source or third-party software. By setting up a project hierarchy, you can easily switch between the main project for your application (the parent project) and the projects for the modules (the child projects) to complete the work needed to build a composite Bill of Materials.

Note that a child project, in turn, can be a parent project to other projects. Likewise, a given parent project can be identified as a child project to other projects. Since hierarchies are created as needed, projects might have no association with a hierarchy.

Once the hierarchy for a given project is established either as a parent or a child, you can do the following:

- From the **Summary** page for the project, view and link to any of its child and parent projects (see Summary Tab).
- From the global **Inventory** view, examine the inventory of its child projects as well as link to any these projects (see Inventory View).

Continue with these next topics for descriptions on how manage project hierarchies:

- Identifying Child Projects for a Given Project
- Disassociating a Child Project from a Parent Project

## Identifying Child Projects for a Given Project

Use this procedure to associate one or more projects as child projects of a given project.

***Note*** ▪ *Ensure that each project that you want to identify as a child project has already been created.*

***Task*** ***To identify one or more projects as child projects of another project, do the following:***

1. As Project Administrator, navigate to the **Summary** tab of the project for which you are identifying one or more child projects. (For navigation instructions, see Opening the Project Summary Tab.)
2. Click **Manage Project** and select **Edit Project** from the dropdown menu. The **Edit Project** page opens.
3. Select the **Project Hierarchy** tab.
4. On the **Project Hierarchy** tab, click **Add Child Project**.
5. From the **Add Child Project** dialog that is displayed, select the project you that want to identify as a child project of the current project. (For a description of this dialog and the fields available on the **Project Hierarchy** tab, see Edit Project: Project Hierarchy Tab.)

   Once you select the project, you are returned to the **Project Hierarchy** tab, which now lists the new child project.
6. Repeat steps 4 and 5 to add other child projects to the current project.
7. Click **Save**.

## Disassociating a Child Project from a Parent Project

Use these steps to disassociate a child project from its parent project. Once you disassociate the child project, it is removed from the parent project's hierarchy.

For a description of the fields available on the **Project Hierarchy** tab used to perform this procedure, see Edit Project: Project Hierarchy Tab.

***Note*** ▪ *Disassociating a child project simply means that project is no longer identified as a child of the current project. This procedure does not delete or in any way change the project that you disassociate.*

***Task*** ***To disassociate a child project from a parent project, do the following:***

1. As Project Administrator, navigate to the **Summary** tab of the project for which you are disassociating one or more child projects. (For navigation instructions, see Opening the Project Summary Tab.)
2. Click **Manage Project** to open a dropdown menu, and select **Edit Project**. The **Edit Project** page opens.
3. Select the **Project Hierarchy** tab.
4. In the list of child projects on the tab, click **X** in the **Actions** column for the child project you want to disassociate from the parent project. A message box appears, prompting you to confirm the disassociation.

   Once you confirm to disassociate the project, child project is removed from the hierarchy. The links associated with this parent-child relationship are also removed from the **Summary** pages for the parent project and the project that was disassociated. The links are also removed from **Inventory** view.
5. Repeat the previous step for each child project you want to disassociate from the current project.
6. Click **Save**.

# Changing the Project Contact

Code Insight provides the ability to change the Project Contact for a given project. The Project Contact, initially the project creator, is the default contact for all task-workflow notifications generated during the inventory review process. That is, if a Legal, Security, or Development contact has not been explicitly assigned to the project through system Project Defaults or at the project-settings level, that contact defaults to the Project Contact. Additionally, the Project Contact is the default contact for any "miscellaneous" tasks created during an inventory review.

The current Project Contact, a Project Administrator, or a System Administrator can transfer the Project Contact role to different user. That user automatically inherits the roles the previous Project Contact user held.

***Note*** ▪ *Changing a Project Contact is a silent transaction. No email notifications will be sent as part of this operation.*

***Task*** ***To change the Project Contact, do the following:***

1. Log into Code Insight as the current Project Contact, a Project Administrator, or System Administrator.
2. Navigate to the **Summary** tab (see Opening the Project Summary Tab).
3. Click **Manage Project** and select **Change Project Contact** from the dropdown menu, or click the **Change Project Contact** button, whichever is available. The **Select New Project Contact** dialog appears.

   ***Note ▪*** *If you have not logged with the appropriate permissions, neither the menu option nor the button will be visible.*

4. Select a name in the list and click **Apply**. The **Summary** tab shows the selected name displayed in the **Project Contact** field.

# Rescanning Your Codebase (Server Scans Only)

During a server scan, Code Insight uses a combination of Automated Analysis and Advanced Analysis techniques to identify open-source and third-party content in your codebase (see About Code Insight Scans). Automated Analysis is always performed during a scan. Advanced Analysis is performed only if the Compliance Library (CL) has been installed in your Code Insight system and the scan settings for your project have enabled this type of analysis.

When you run the initial scan on your codebase, a *full* scan (that is, a scan on all codebase files) is automatically performed for both analysis techniques. For any subsequent codebase rescans that you initiate, you can manage the scan as follows:

- Force a full rescan (which can take considerable time since all codebase files will be rescanned).
- Allow default scan behavior for all regular rescans (that is, ones that are not forced full rescans). Default rescan behavior typically scans only those files that have changed since the last scan. However, certain Code Insight events that have occurred since the last scan can result in a full rescan.
- Configure the scan profile associated with the project so that all rescans (except forced full rescans) scan only changed files and skip unchanged files, despite any events that might have occurred.

The following topics provide information you should know about the rescan process and includes instructions on initiating a rescan:

- Default Rescan Behavior
- Configuring Rescans to Always Skip Unchanged Files
- Effects of Scan-Setting Changes on Rescans
- Handling of Edited Inventory During Rescans
- Initiating a Codebase Rescan
- Forcing a Full Codebase Rescan

## Default Rescan Behavior

When a user initiates a regular rescan (that is, does not a *force* a full rescan), typically only codebase files that have changed since the last scan are rescanned. However, by default, certain Code Insight events that might have occurred since the last scan will determine whether the rescan performs a scan on all files (full rescan) or on only those codebase files that have changed since the last scan.

The following table lists these events and the type of default scan performed by each analysis technique—Automated Analysis and Advanced Analysis (if this technique is configured)—during the rescan.

Note that a System Administrator can configure the scan profile associated with a project to override the default rescan behavior dictated by these listed events, so that each rescan skips unchanged files and scans only those files that have changed, despite any events that might have occurred. For more information, see Configuring Rescans to Always Skip Unchanged Files.

**Table 2-16 ▪** Default Rescan Behavior for Events

| Event | Automated Analysis | Advanced Analysis | Notes |
|---|---|---|---|
| **Change to codebase files** | Only changed files scanned | Only changed files scanned | Changes in codebase files are determined by the MD5 hash digest of the files. |
| **Change to Automated Analysis rule set** | Full rescan | See Notes | Changes in Automated Analysis rules result in Automated Analysis performing a full rescan to reapply the changes to all files. (The rule changes are automatically pushed to your Code Insight server through an internal process and the weekly Electronic Update.)<br>Additionally, Advanced Analysis performs a full rescan only if rule changes have occurred *and* either the CL version has changed or an NG-bridge update has occurred. (See the *Code Insight Installation and Configuration Guide* for information about NG-bridge updates.)<br>**Note ▪** *If an override of default rescan behavior is in effect (see Configuring Rescans to Always Skip Unchanged Files), the rules are applied to only changed files; unchanged files are skipped.*<br>**Note ▪** *Custom detection rules are applied only during the initial codebase scan and during a forced full rescan. They are not applied during a regular rescan initiated by the user.* |

**Table 2-16 ▪** Default Rescan Behavior for Events (cont.)

| Event | Automated Analysis | Advanced Analysis | Notes |
|---|---|---|---|
| **Code Insight version change** | See Notes | See Notes | Automated Analysis performs a full rescan only if the new Code Insight version includes changes to the Automated Analysis framework.<br>Advanced Analysis performs a full rescan only if the Code Insight version has changed *and* either the CL version has changed or an NG-bridge update has occurred. (See the *Code Insight Installation and Configuration Guide* for information about NG-bridge updates.)<br>***Note ▪*** *If an override of default rescan behavior is in effect (see Configuring Rescans to Always Skip Unchanged Files), the rescan skips unchanged files and scans only changed files, even if conditions are present to perform a full rescan for one or both analysis techniques.* |
| **Scan profile setting change** | Full rescan | See Notes | A full rescan for Automated Analysis is required if one or more specific scan profile settings have changed. See Effects of Scan-Setting Changes on Rescans for more information.<br>Advanced Analysis also performs a full rescan only if one or more of the specific profile settings have changed *and* either the CL version has changed or an NG-bridge update has occurred. Keep in mind that changes to settings related to source-code matching result in an *expensive* full scan for Advanced Analysis. (See the *Code Insight Installation and Configuration Guide* for information about NG-bridge updates.)<br>Scan profiles are configured by the Code Insight System Administrator as described in the *Code Insight Installation and Configuration Guide*.<br>***Note ▪*** *If an override of default rescan behavior is in effect (see Configuring Rescans to Always Skip Unchanged Files), the rescan skips unchanged files and scans only changed files, even if scan profile settings have changed.* |

## Configuring Rescans to Always Skip Unchanged Files

As described in Default Rescan Behavior, by default certain Code Insight events that might have occurred since the last scan can determine whether the rescan performs a scan on all files (full rescan) or on only those codebase files that have changed since the last scan.

However, the System Administrator can configure the scan profile associated with a project to override this default behavior. The configuration allows rescans to always skip unchanged files and scan only changed files, even if events that typically call for a full rescan have occurred. It also delineates *which* unchanged files are skipped: all unchanged files, only unchanged files that have been reviewed, only unchanged files that are associated with inventory, or only unchanged files that are both reviewed and associated with inventory.

See your System Administrator about the rescan configuration options.

***Note*** ▪ *The current rescan configuration options are ignored if the user initiates a forced full rescan. All files—both changed and unchanged—are completely scanned.*

## Effects of Scan-Setting Changes on Rescans

One type of change event that, by default, does result in a full rescan by either Automated Analysis or Advanced Analysis (or both) is an update to settings in the scan profile associated with the rescan. Depending on which settings have changed, the full rescan could be more expensive (requiring more time and resources) than other full rescans.

Note the following:

- If you have applied a new scan profile to your project, only those profile settings that are different from the settings in the previously associated profile will impact the rescan.
- If an override of the default rescan behavior is in effect (see Configuring Rescans to Always Skip Unchanged Files), no full rescan is performed even if any of the scan profile settings listed below have changed. The rescan skips unchanged files and scans only those files that have changed.
- If the following table shows that a change to a specific setting results in a full rescan for Advanced Analysis, note that the full rescan is performed only if, in addition to the setting change, either the CL version has changed or an NG-bridge update has occurred. Otherwise, the setting change results in a scan of only changed files.

The following table provides a list of the scan profile settings and the type of full rescan to expect should any of the settings be updated prior to a codebase rescan.

**Table 2-17 ▪** Types of Full Rescan to Expect Should Scan Profile Settings Change

| Scan Profile Settings | Automated Analysis | Advanced Analysis |
|---|---|---|
| A change to any of these settings:<br>● **Perform Package/License Discovery in Archive**<br>● **Dependency Support**<br>● **Automatically Add Related Files to Inventory** | Full rescan | — |
| A change to any of these settings:<br>● **Source Code Matches**<br>Related fields:<br>● **Include System Identified Files**<br>● **Include Files with Exact Matches**<br>● **Minimum Source Code Matches** | — | Full rescan (expensive) |
| A change to any of these settings:<br>● **Exact Matches**<br>● **Search Terms**<br>● **Scan Inclusions** | — | Full rescan (expensive but less expensive than that performed when **Source Code Matches** or related fields change) |

## Handling of Edited Inventory During Rescans

Code Insight enables you to make changes to inventory both in the **Analysis Workbench** and on the **Project Inventory** tab. You create inventory items as well as edit both user-created and system-generated inventory. Edits to existing inventory can include changes to the following elements in an inventory item:

- The component version string or the associated license
- Codebase-file associations (only in the **Analysis Workbench**)
- Inventory properties, Notices text, and notes

However, normal Code Insight rescan behavior can result in actions that impact your inventory changes. For example, an updated Automated Analysis rule set might associate codebase files to an inventory item different from the one to which you have *manually* associated these files. Logically, the rescan should remove the associations you defined and re-apply them to the inventory item identified in the rule set. However, losing the manual changes might not be desirable.

The following topics describe how the rescan process handles edited inventory:

- Rescan Rules to Preserve Inventory Data
- Rescan Scenario: Handling of Files Manually Disassociated from Inventory Before Rescan

## Rescan Rules to Preserve Inventory Data

In general, a rescan (full or on only changed files) can add or disassociate files and overwrite properties for any existing system-generated inventory that has not been manually updated. However, the rescan *does* retain the existing status and priority for such inventory items, as well as any existing notes or Notices text (although the scan can append new notes or Notices text).

For inventory that you have manually edited or created, the rescan applies the following rules:

- All the user-created inventory, its file associations, and edits are considered not system-updatable and therefore are preserved.
- Any manual change to a system-generated inventory item (including updates to the associated component) results in the inventory item being classified as user-created and therefore not system-updatable (see the previous rule.) However, the rescan can add additional files to the inventory item if the component, version, and license match.
- If one or more files were manually disassociated from a system-generated inventory item before the rescan, rescan logic assumes that these files were erroneously associated with the component initially. Therefore, the rescan does not attempt to re-associate these files to the inventory item; nor does it associate the files with another inventory item that uses the same component name (with a different version or license). The following example scenario illustrates this rule.

## Rescan Scenario: Handling of Files Manually Disassociated from Inventory Before Rescan

The following scenario demonstrates how the rescan process handles files that you manually disassociated from a system-generated inventory item before the rescan.

In this scenario, the initial scan on your codebase generated the inventory item **log4j 2.6** and associated the files **file1.jar** and **file2.jar** with the item.

However, after analyzing the inventory, you realize that **file2.jar** should be associated instead with **log4j 2.11**, an inventory item that does not exist in your current inventory. To remedy this, you perform the following steps:

1. Create an inventory item named **log4j 2.11**.
2. Disassociate the **file2.jar** from **log4j 2.6**.
3. Associate **file2.jar** with the inventory item **log4j 2.11** that you just you created.

On rescan, your edits remain intact:

- The file **file1.jar** remains associated with the inventory item **log4j 2.6**.
- The inventory item **log4j 2.11** that you created is preserved along with its association with the file **file2.jar**.

The rescan also results in the creation of a new system-generated inventory item, **log4j 2.10**. However, the rescan does not associate the file **file2.jar** with the new inventory item.

# Initiating a Codebase Rescan

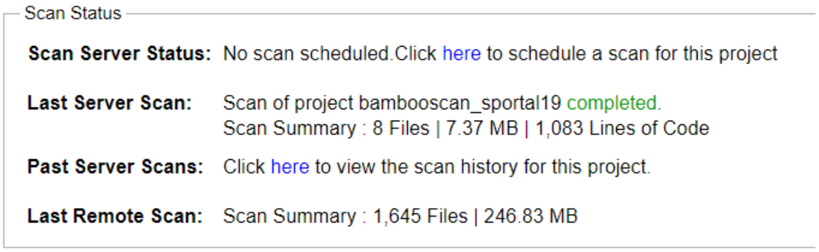
Use the following procedure to rescan your codebase.

Refer to the Code Insight User Roles and Permissions appendix for role requirements to scan a codebase.

***Task*** ***To start the rescan, do the following:***

1. Navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Perform either step:
    - Click the **Start Scan** button. If a scan is currently running or the Scan Server is currently not active, this button is disabled. You can use the second option (see next bullet) to schedule a scan.
    - Click the "here" link in **Scan Server Status** to schedule a scan. If other scans are running, the scan is queued and will automatically run based on queue order. If the Scan Server is inactive, the scan will automatically start based on queue order once the server is running again. (Click the link in **Past Scans** to view details about the scheduled scan.)

    Information about the server scan's progress is shown in the **Scan Status** section on the **Summary** tab.



Scan Status

**Scan Server Status:** No scan scheduled.Click here to schedule a scan for this project

**Last Server Scan:** Scan of project bambooscan_sportal19 completed.
Scan Summary : 8 Files | 7.37 MB | 1,083 Lines of Code

**Past Server Scans:** Click here to view the scan history for this project.

**Last Remote Scan:** Scan Summary : 1,645 Files | 246.83 MB

When the scan completes, **Last Server Scan** will display one of the following messages:

- **Completed**—The scan succeeded with no warnings during scan or analysis. This message appears on the screen in green.
- **Completed with warnings**—The scan succeeded but the analysis produced warnings. For more information, check the **scanEngineDetail** log for the Scan Server.
- **Failed**—The scan failed. This message appears on the screen in red. For more information, see Scan Failure Reasons and Troubleshooting Measures.

For an overall understanding of the scan results, see Overview of Scan Results.

# Forcing a Full Codebase Rescan

A forced full-codebase rescan enables you to scan your entire codebase at any time even if no change has occurred in your codebase, in your scan settings, or with the Code Insight or Compliance Library (CL) version. Such a rescan might be required, for example, to view the latest changes to inventory or to apply any new custom detection rules. See the following topics for more information:

- Forcing a Full Rescan
- Custom-rule Application During a Forced Full Rescan

Keep in mind that a full rescan can take considerable time.

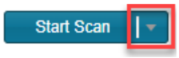
For general information about how any rescan handles existing system-generated inventory and manually created or updated inventory, see Handling of Edited Inventory During Rescans. For specific information about how the forced full rescan applies custom rules to existing system-generated inventory, see Custom-rule Application During a Forced Full Rescan.

## Forcing a Full Rescan

Use the following procedure to initiate a full codebase rescan.

***Task*** ***To force a full project rescan, do the following:***

1. As Project Administrator or Analyst, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click the drop-down arrow next to the **Start Scan** button.



3. Select **Full Rescan**.

   A confirmation message box is displayed asking you to confirm that you want to continue with the rescan.

4. Select **Yes**.

   Information about the scan's progress and its completion status is shown in the **Scan Status** section. For details, see the last step in Initiating a Codebase Rescan.

## Custom-rule Application During a Forced Full Rescan

Custom rules are applied only during the initial codebase scan and during a forced full rescan. During the forced full rescan, if a previously scanned file now matches a new custom rule, the existing system-generated inventory item for that file is overwritten with the custom-rule data. The following two scenarios describe this over-write process.

***Note*** ▪ *Custom rules affect only system-generated inventory items that have not been manually updated. The rules have no impact on manually-created (custom) inventory items and on system-generated inventory items that users have updated.*

### Scenario 1: The custom-rule data identifies the same component version and license as the existing inventory item

In this case, the scan applies the custom rule by updating the existing inventory item as follows:

- Appends new detection notes to reflect the custom rule.
- Updates the **Created by** field value for the inventory item to **High Confidence Custom Auto-WriteUp Rule** in the **Analysis Workbench**.

Note that, in this scenario, the scan retains the **Audit Notes**, **As-Found License Text**, or **Notices Text** content defined in the existing inventory item. It does not append custom-rule data to these fields.

### Scenario 2: The custom-rule data identifies a component version and license different from the existing inventory item

In this case, the scan applies the custom rule as follows:

- Creates a new inventory item based on the custom-rule data.
- Retains the existing inventory item, but disassociates its files and adds them to the new inventory item.
- Applies the status and priority of the existing inventory to the new inventory item.
- Appends any **Audit Notes**, **As-Found License Text**, or **Notices Text** content defined in the custom rule.

# Exporting Project Data

Code Insight allows you to export your project data to a JSON data file for use elsewhere. The following procedure describes how to export project data using the Code Insight Web UI.

For complete information about the export feature (including how to export project data using the public REST API), see the Exporting and Importing Project Data chapter.

*Task* ***To export project data, do the following:***

1. As Project Administrator or Analyst, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Export Project Data** from the dropdown menu.
3. When prompted, select a location to store the exported data. Code Insight creates a JSON data file, archives it in a `.zip` file and saves it in to a location specified in your browser settings.

# Importing Project Data

Code Insight allows you to import data from one Code Insight project into another project. The data to be imported must be in a properly formatted and archived JSON file, such the archive resulting from an export described in Exporting Project Data. The following procedure describes how to import project data using the Code Insight Web UI.

For complete information about the import feature (including how to import project data using the public REST API), see the Exporting and Importing Project Data chapter.

*Task* ***To import project data, do the following:***

1. As Project Administrator or Analyst, navigate to the **Summary** tab (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Import Project Data** from the dropdown menu.
3. Complete the fields as described in Import Project Data Dialog.
4. Click **OK** to perform the import.
    - If the import fails for some reason, as error dialog is displayed. Click **OK** and attempt the import again.
    - If the import completes successfully, a message dialog is displayed, stating as such. Click **OK**.

# Branching a Project

Code Insight provides the **Branch Project** wizard to automate the process of branching one Code Insight project to another, enabling the branch project to preserve any file-audit data, inventory, and inventory-review data that was created in the project from which you are branching.

The following sections describe how to use the **Branch Project** wizard to set up and start the branching process and what happens during the branching operation.

- Project-Branching Terminology
- Setting Up and Starting the Project-Branching Process
- Other Considerations About the Project-Branching Operation

## Project-Branching Terminology

The following terminology is used in the descriptions that follow:

- **Source project**—The Code Insight project whose data you are branching to another project.
- **Branch project**—The new project to which you are branching data from the source project.

## Overview of the Branching Operation

The following provides a basic overview of what happens during the project-branching operation.

**Table 2-18 ▪** Phases of the Project-Branching Operation

| Branching Phase | | Description |
|---|---|---|
| **Phase 1** | **Launch of Branch Project wizard** | The user opens the Branch Project wizard to set up the project-branching operation. Details about the entire setup process is described in Setting Up and Starting the Project-Branching Process. |
| **Phase 2** | **Branch project creation during setup** | As part of the setup, the branch project is created once the project properties on the **Project Information** page of the wizard have been validated (upon clicking **Next**), as described in Step 1: Creating the Branch Project. |

**Table 2-18 ▪** Phases of the Project-Branching Operation (cont.)

| Branching Phase | | Description |
|---|---|---|
| **Phase 3** | **Codebase uploads and SCM synchronization configuration during setup** | As part of the setup, one or more codebases can be uploaded directly from the **Upload Codebase** page in the wizard, as described in Step 2: Uploading a Codebase (Optional).<br>Additionally, as part of setup, the user can configure one or more Source Control Management (SCM) instances from the **Version Control Settings** page, enabling the branch process to synchronize the project with remote codebase repositories in your site's SCM applications. Unlike codebase uploads, synchronization takes place once the automated part of the branching process begins (Phase 5).<br>Codebase uploads and synchronization are optional, as the user might want to simply perform an inventory copy (Phase 7). |
| **Phase 4** | **Initiation of project-branching operation** | After all appropriate setup information is provided in the wizard, the user initiates the branching operation by clicking **Finish** on the **Summary** page of the wizard, as described in Step 5: Initiating the Branching Operation. |
| **Phase 5** | **Synchronization with remote codebases through SCM instances** | During the branching operation, the branch project is synchronized with one or more remote Source Code Management (SCM) repositories to obtain codebase files. This synchronization occurs only if SCM instances were configured during setup on the **Version Control Settings** page of the wizard (as described in Step 3: Configuring Synchronization with a Source Code Management Instance (Optional)). |
| **Phase 6** | **Scan of branch project codebase** | The branching operation then scans the codebase of the branch project. The codebase includes any of the uploaded codebase files as well as any codebase files obtained through synchronization with SCM instances. Note that the size of the codebase can impact the length of time needed for the scan. |
| **Phase 7** | **Copy of project information** | The branching operation copies (that is, imports) file-audit data, inventory, and inventory-review information from the source project to the branch project.<br>If no codebases have been uploaded or obtained through synchronization with SCM repositories, only inventory and inventory-review information is imported; no information about files associated with the inventory is included in the import. |
| **Phase 8** | **Operation completion** | Once the branching operation successfully completes, users can open the branch project and begin auditing files and reviewing inventory. |

# Setting Up and Starting the Project-Branching Process

The following procedures highlight important information about the various steps that the Branch Project wizard guides you through in setting up a project-branching operation.

- Opening the Branch Project Wizard
- Step 1: Creating the Branch Project
- Step 2: Uploading a Codebase (Optional)
- Step 3: Configuring Synchronization with a Source Code Management Instance (Optional)
- Step 4: Configuring a Project Copy
- Step 5: Initiating the Branching Operation

You can cancel the setup process at any time, as described in Canceling the Branching Setup Process. Also review Other Considerations About the Project-Branching Operation for special notes pertaining to the project-branching process.

## Opening the Branch Project Wizard

Begin the setup for the project-branching by opening the **Branch Project** wizard from the source project.

***Task*** ***To open the Branch Project wizard, do the following:***

1. Navigate to the **Summary** tab of the source project (see Opening the Project Summary Tab).
2. Click **Manage Project** and select **Branch Project** from the dropdown menu.

   The **Branch Project** wizard opens to the **Introduction** page.
3. Click **Next** to navigate to the **Project Settings** page.

***Note*** ▪ *The* ***Manage Project | Branch Project*** *option is disabled if the source project has not yet been successfully scanned or if a codebase upload, scan, or report generation is currently in progress on the project.*

## Step 1: Creating the Branch Project

The **Project Settings** page in the **Branch Project** wizard enables you to define the properties for the branch project and then creates the project once you click **Next** to move to the next wizard page. By default, property values are pre-populated with values from the source project. However, you can edit these properties as needed.

*Task* ***To create the branch project, do the following:***

1. On the **Project Settings** page in the **Branch Project** wizard, identify the properties for the new branch project. Initially, the properties from the source project pre-populate this page, but you can edit these properties as needed. See Branch Project: Project Information for a description of each property.

   The option you choose for **Source Code Options** determines how the branching process obtains the codebase files for the branch project. Initially, the **Upload Codebase** option is selected by default. The **Merge from Source Control** option is also selected if the source project obtains codebase files from synchronization with a Source Control Management instance. However, you can edit these options as needed for the branch project. If you select neither option, only inventory and inventory-review information is copied to the branch project; no information about the files associated with inventory is branched.

2. Click **Next** to create the project and proceed to the next appropriate page in the wizard.

   - If selected **Upload Codebase** in the **Source Code Options** section, the **Update Codebase** page opens. See Step 2: Uploading a Codebase (Optional).
   - If you selected only **Sync from Source Control** in the **Source Code Options** section, the **Version Control Settings** page opens. See Step 3: Configuring Synchronization with a Source Code Management Instance (Optional).
   - If you selected neither option, the **Project Copy Settings** page opens. See Step 4: Configuring a Project Copy.

   Once the project is created, you cannot edit the project **Name** should you return to this page to update project properties. Additionally, once a codebase is uploaded for the branch project, the **Scan Server** and **Upload Codebase** options on the **Project Information** page will be disabled.

## Step 2: Uploading a Codebase (Optional)

The **Upload Codebase** page in the **Branch Project** wizard identifies and uploads one or more codebase files for the branch project.

This page is enabled only if you selected the **Upload Codebase** option from the previous **Project Information** page.

The uploaded codebases can be used in conjunction with codebases obtained through synchronization with your site’s SCM applications to provide the complete set of codebase files for the branch project. See Step 3: Configuring Synchronization with a Source Code Management Instance (Optional).

If you decide not to upload a codebase, clicking **Next** moves you to the next appropriate wizard page. (When this page is enabled but no codebase is uploaded, you can always return to this page to perform a codebase upload during setup if you want.)

*Task* ***To upload a codebase for the branch project, do the following:***

1. On the **Upload Codebase** page in the **Branch Project** wizard, select the archive containing the codebase to upload.
2. Define the properties for the upload. See Branch Project: Upload Codebase for a description of the properties.

3. Click **Upload Project Codebase** to verify the properties and proceed with uploading the codebase. A message displays when the upload has completed successfully.
4. Repeat the above steps for each codebase you want to upload.
5. When the codebases have been uploaded, click **Next** to proceed to the next appropriate page in the wizard:
    - If you selected **Sync from Source Control** on the **Project Information** page, the **Version Control Settings** page in the wizard opens. See Step 3: Configuring Synchronization with a Source Code Management Instance (Optional).
    - If you did not select **Sync from Source Control**, the **Project Copy Settings** page opens. See Step 4: Configuring a Project Copy.

## Step 3: Configuring Synchronization with a Source Code Management Instance (Optional)

The **Version Control Settings** page in the **Branch Project** wizard configures one or more Source Control Management (SCM) instances, enabling the branching operation to synchronize the branch project with remote codebase repositories in your site's SCM applications. This synchronization takes place once the automated part of the branching process begins (see Step 5: Initiating the Branching Operation). For more information about how to set up for SCM synchronization and how the synchronization process works, refer to Configuring Source Code Management.

This page is enabled only if the **Sync from Source Control** option is selected on the previous **Project Information** page.

By default, any SCM instances used by the source project are automatically copied to this page, each instance defined on a separate tab. However, you can edit or remove any of these instances or add new ones as needed for the branch project. Alternatively, you can choose not to include any SCM instances on the **Version Control Settings** page, but can always return to this page to add instances later during setup if you want.

If you also uploaded codebases from the **Upload Codebase** page, the codebase files obtained through SCM synchronization, as defined on this page, will be added to the uploaded codebase files to provide the complete codebase for the branch product.

***Task*** ***To configure synchronization with SCM instances, do the following:***

1. On the **Version Control Settings** page in the **Branch Project** wizard, update or delete any currently defined SCM instances or click **Add Instance** to create a new instance as needed. See Branch Project: Version Control Settings for more information about managing the SCM instances and the properties used to define each instance.
2. For any given SCM instance, click the **Test Connection** to ensure that branching operation can successfully connect to the remote repository specified in the instance.
3. When SCM instance configuration is complete, click **Next** to perform a final connection test on all the SCM instances defined and, and if the connections are successful, proceed to the **Project Copy Settings** page in the wizard. See Step 4: Configuring a Project Copy.

## Step 4: Configuring a Project Copy

The **Project Copy Settings** page in the **Branch Project** wizard defines the parameters used by the branching process to import file-audit data, inventory, and inventory-review information from the source project to the branch project. By default, this page shows the properties used by the source project. However, you can edit these properties as needed for the branch project.

***Note*** *▪ During the import, the branching process ignores the setting **On rescan or import, delete inventory without any associated files** for the branch project and always creates inventory. (This setting, found on the **Manage Project | Edit Project | General** tab on the project's **Summary** page, can be accessed once the branching process is completed.)*

If neither **Upload Codebase** nor **Sync from Control Version** was selected on the **Project Information** page, this import process copies only inventory and inventory-review information from the source project to the branch project. With this scenario, no file information will be associated with the inventory copied to the branch project.

***Task*** ***To define project copy settings for the branching operation, do the following:***

1. On the **Project Copy Settings** page in the **Branch Project** wizard, edit the properties that the branching operation will use to import file-audit and inventory information from the source project to the branch project. See Branch Project: Project Copy Settings for more information about these properties.
2. Click **Next** to verify the properties and, if no errors exist, move to the **Summary** page in the wizard. See Step 5: Initiating the Branching Operation.

## Step 5: Initiating the Branching Operation

The **Summary** page in the **Branch Project** wizard provides an overview of the parameters that you defined for the project-branching process. From this page, you can start the branching process. Alternatively, you can navigate back to other pages in the wizard to make changes before starting the branching process, or you cancel the entire branching setup.

***Task*** ***To initiate the branching operation, do the following:***

1. On the **Summary** page in the **Branch Project** wizard, review the list of properties defined for the branching operation.
2. If you need to make any changes to the current configuration for the branching operation, click **Back** to navigate backwards through the wizard pages. Alternatively, you can click any of the enabled tabs for wizard pages to move directly to a page.
3. To start the branching operation, click **Finish** on the **Summary** page. The project dashboard for the branch project is displayed, showing the current status of the branching process.

When the branching operation is finished, you can open the branch project and navigate to its **Project Inventory** tab or **Analysis Workbench** to proceed with auditing codebase files or reviewing inventory.

For an overview of the branching-operation phases, see Overview of the Branching Operation.

### Canceling the Branching Setup Process

During the project-branching setup process, you can press the **Cancel** button on the **Branch Project** wizard window (or close the window) at any time to cancel the setup. A pop-up message will ask you to confirm the cancellation. If you select **Yes**, the branch project and its uploaded codebases will be removed from the Code Insight system.

## Other Considerations About the Project-Branching Operation

The following are additional notes about the branching operation:

- Users can run parallel branching operations on the same source project.
- If a user initiates the branching process in one browser tab, opens the same Code Insight instance in another browser tab, and then navigates to the branch project in the second tab, the branching status information for the branch project should be the same as in the first browser tab.
- The project-branching session might time out (for example, due to a delayed scan phase because the scan is queued). Should the session time out, the user must use the `DELETE projects/{projectId}` REST API to remove the branch project from Code Insight. Once the project is deleted, the user can then rerun the **Branch Project** wizard to recreate the branch project and initiate the branching operation.
- Should an SCM synchronization or the scan fail during the automated part of the branching operation, the branch process is terminated. The user can navigate to the **Summary** tab of the branch project to view any captured scan details.
- During the branching process, the branch codebase is scanned before the import phase begins. Therefore, any file association for a source inventory item most likely also exists in the branch inventory item by the time the import starts. By design, when the import detects the same file association for both the source and branch inventory item, it adds no new file association to the branch inventory item. However, when the import processes this association, a “duplicate entry” exception similar to the following might be logged:

  ```
  Duplicate entry ‘entry_id’ for key 'pse_inventory_group_files.UNIQ_FILE_GROUP
  ```

  This exception is benign and has no impact on the regular file-processing behavior—that is, the existing file association is retained; no new file association is added.

# Deleting a Project

Project Administrators can use this procedure to delete any of their projects. When a project is deleted, the following components are deleted:

- **From the Code Insight database**—The project record and all scan results, inventory, alerts, tasks, and user audit work associated with the project.
- **From the Scan Server**—The project’s codebase files.

***Note*** ▪ *The option to delete a project is not enabled if a scan is being performed on the project.*

*Task* ***To delete a project, follow these steps:***

1. As Project Administrator, perform either step:
    - Navigate to the **Summary** tab (see Opening the Project Summary Tab), click **Manage Project**, and select **Delete Project** from the dropdown menu.
    - Navigate to the **Projects** view (see Opening the Projects View), right-click the project in the project display, and select **Delete Project**.
2. When prompted, select **Yes** to proceed with the deletion. If the Scan Server associated with the project is temporarily inactive or is disabled, a pop-up is displayed to inform you that the server is down.

# Creating a Private Project

When a project is created, the default visibility for the project is **Public**, which means that any Code Insight user has read-only access to the project. To what degree a user can interact with the project depends on whether the user has a project role and what the role is.

Security-conscious project creators can control access to their projects within the enterprise by setting a project's visibility to **Private**. This feature gives project creator the ability to hide sensitive information from general view and select specific users who can view the project. Private projects are hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. Additionally, project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see a given private project.

When a private project is created, the creator automatically becomes the Project Contact and is assigned to the Project Administrator, Analyst, Reviewer, and Observer roles. These roles enable the creator to initially manage the project and its users, analyze the project codebase, and review project inventory. However, creators can remove themselves from any of these roles to let others handle project responsibilities.

***Note*** *▪ Users who have System Administrator privileges but are not part of a Private project can see the project in the list of projects in the* ***Projects*** *view, access the* ***Summary*** *tab for the project, and change the project contact.*

For information roles, see Assigning and Removing Project Users.

*Task* ***To create a private project, do the following:***

1. Navigate to the **Projects** view. (See Opening the Projects View if additional instructions are needed.)
2. In the **Projects** pane on the left, click **Add New**. The **Add Project** dialog appears with default values appearing in all the fields but **Name**.
3. In the **Name** field, enter a name for the new private project.
4. From the **Project Visibility** dropdown, select **Private**.
5. Complete the other fields as described in About Code Insight Projects.
6. Click **Save** to save the new private project.

This project is visible in the list of projects to only the Project Contact and any Project Administrator, Analyst, Reviewer, or Observer of the project. Additionally, the project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see these projects.

7. (Optional) Assign roles to users who will interact with the private project. For more information, see Assigning and Removing Project Users.

# Managing Policy Profiles

This section describes the purpose of policies in Code Insight and provides procedures for adding, editing, viewing, and copying polices. The following topics are included:

- Understanding Policy Profiles
- How Policy Profiles Work in the Automated Inventory-Review Process
- Opening the Policy Page
- Adding or Editing a Policy Profile
- Copying a Policy Profile

## Understanding Policy Profiles

Policy profiles are used by Code Insight to automate the inventory review process—that is, automatically mark published inventory items as approved or rejected—without the need for a manual review. (Inventory that are neither approved or rejected by policy will require a manual review.) Policy profiles can be defined up-front or revised during the manual inventory review process. The Code Insight Administrator grants the **Manage Policy** role to users who have rights to manage policy profiles. Typically, these would be legal or security experts.

Code Insight provides a default policy profile (called *Default License Policy Profile*) that can be used as is, modified, or copied to fit your need. This policy profile contains typical settings for a team who is distributing software. You can also create policies from scratch.

The topics are covered in this section:

- How Policy Profiles Work in the Automated Inventory-Review Process
- Adding or Editing a Policy Profile
- Viewing an Existing Policy
- Copying a Policy Profile
- Associating a Policy Profile with a Project

## How Policy Profiles Work in the Automated Inventory-Review Process

A policy profile is defined with a set of policies whose criteria is based on OSS or third-party component versions, licenses, or security vulnerability score and severities. Any conflicting criteria are resolved in favor of an automated rejection of the inventory item. In other words, rejections will always take precedence over approvals.

The policy criteria are evaluated when an inventory item is published. If none of the criteria in the profile applies to a given inventory item, the system leaves the inventory item in a **Not Reviewed** state, thus requiring the inventory to be manually reviewed.

You can further automate the inventory review process by setting additional review policies that define what action Code Insight takes once an item is rejected or is assigned a **Not Reviewed** status. See Updating Inventory Review and Remediation Settings for a Project for more information about these additional policies.

# Opening the Policy Page

Use the following procedure to open the **Policy** page, which provides access to the functionality needed to manage policy profiles. The procedure assumes that you have logged into Code Insight.

***Task*** ***To open the Projects view, do the following:***

Open the **Policy** page using either of these methods:

- From the Code Insight dashboard (which is displayed when you start Code Insight), click **view policy**. See Opening Code Insight for details on accessing the dashboard.
- Click the ☰ icon in the upper right corner of the Code Insight Web UI to open the Code Insight main menu. Select **POLICY** from the menu.

The **Policy** page is displayed, showing a list of all available policy profiles. For more information about this page, see Policy Page.

***Note*** ▪ *If the Code Insight System Administrator has changed Code Insight's CVSS version configuration (for example, from CVSS v2.0 to 3.x), policies defined for these profiles might have automatically changed based on the new version's scoring system. For more information, see Impact on Policies When Code Insight's CVSS Configuration Changes for details.*

# Adding or Editing a Policy Profile

The following procedure describes how to add a new policy profile or edit an existing one. Only users who have Policy Manager permissions can create or edit a policy.

***Task*** ***To add a new policy profile or edit an existing one, do the following:***

1. Open the **Policy** page (see Opening the Policy Page).
2. To edit an existing policy profile, select it from the list, and click the **Edit** icon ✎.

   *or*

   To add a new policy profile, click **Add Policy**.

   The **Policy Details** window for the policy profile is displayed.

3. Refer to the associated help (or to Policy Details Window) for details about the fields used to define the policy profile.
4. Click **Save** to save the updates or to add the new policy profile.

# Viewing an Existing Policy

The following procedure describes how to open a read-only version of a policy profile defined on your Code Insight instance. This version provides an quick to scan of the policy profile details without having to scroll through all the controls and space available for updating a profile in the editable version.

This function is available to all users, whether or not they have Policy Manager permissions.

***Note*** ▪ *In addition to the procedure described here, users can also open a read-only version of the policy profile associated with a given project by clicking the **View Policy Details** link on the project's **Summary** page.*

***Task*** ***To view an existing policy profile, do the following:***

1. Open the **Policy** page (see Opening the Policy Page).
2. From the policy list, select the policy profile you want to view, and click **View** icon .

   A read-only version of the **Policy Details** window for the profile is displayed.
3. Refer to the associated help (or to Policy Details Window) for details about the fields used to define the policy profile.
4. When you have finished reviewing the policy profile, click **Close**.

# Copying a Policy Profile

The following procedure describes how to create a copy of an existing policy profile. This can be useful for creating a template that be used to create other policy profiles or for backing up an existing profile.

Only users with Policy Manager permissions can perform this function.

***Task*** ***To copy an existing policy, do the following:***

1. Open the **Policy** page (see Opening the Policy Page).
2. From the policy list, select the policy profile to copy, and click **Copy** icon .

   The **Policy Details** window opens, showing a new policy profile called **Copy of** `policy name` and having the same policies has the original profile. You can then edit the new profile to change its name or update its policies. See Adding or Editing a Policy Profile.

## Associating a Policy Profile with a Project

The Project Administrator can associate a project with a policy profile by editing the project (see Updating Inventory Review and Remediation Settings for a Project). If no policy is explicitly selected for a project, the *Default License Policy Profile* is used.

# Managing Authorization Tokens

Code Insight uses a JSON Web Token (JWT) to authorize user access to the Code Insight public REST interface. You might be required to explicitly enter an authorization token for certain functionality that uses this REST interface directly (that is, not through Code Insight web UI), such as the following:

- Project import and export processes (see the Exporting and Importing Project Data chapter)
- The execution of remote scan agents (see Performing a Remote Scan)

Code Insight enables you to generate and manage one or more of these authorization tokens.

An authorization token is for use by the Code Insight user account that creates it. Thus, an authorization token that your user account generates will give you REST access to only the Code Insight functionality for which your account has permissions. Additionally, you can view and manage only those authorization tokens for the user account under which you are logged in.

Authorization tokens are created and managed from **Preferences** page, as described in the following procedures:

- Accessing the Preferences Page
- Generating an Authorization Token
- Copying the Authorization Token to the Clipboard
- Editing the Token Name
- Deleting an Authorization Token

## Accessing the Preferences Page

Use these steps to open the **Preferences** page.

***Task*** ***To open the Preference page, use these steps:***

1. Click the **Open Menu** icon in the upper right of any Code Insight page:

   ☰

2. Select **Preferences** to open the **Preferences** page.

# Generating an Authorization Token

Use the following procedure to generate an authorization token.

***Task*** ***To generate an authorization token, do the following:***

1. Access the **Preferences** page (see Accessing the Preferences Page).
2. From the **AUTH Tokens** pane, click **Add Token**.
3. Enter a name for the new token and specify an expiration date (or choose **Never Expires**).
4. Click **Save**.

# Copying the Authorization Token to the Clipboard

Use the following procedure to copy an authorization token to the clipboard so that you can paste it in your REST API interface.

***Task*** ***To copy an authorization token to the Clipboard, do the following:***

1. Access the **Preferences** page (see Accessing the Preferences Page).
2. From the **AUTH Tokens** pane, locate the token you want to copy, and click the **Copy to clipboard** icon (📋) in the **Actions** column.
3. Click **Select Token Text** to select the entire token value, and then press Ctrl + C to copy it.
4. Paste token in the appropriate location for use by the REST interface.

# Editing the Token Name

You can edit only the name of an authorization token, not its expiration date or value.

***Task*** ***To edit the token name, do the following:***

1. Access the **Preferences** page (see Accessing the Preferences Page).
2. From the **AUTH Tokens** pane, locate the token you want to edit, and click the Edit icon (✎).
3. Update the token name as needed.
4. (Optional) To copy the token value to the Clipboard for pasting into the REST interface, click **Select Token Text** to select the entire token value, and then press Ctrl + C to copy it.

## Deleting an Authorization Token

Use the following procedure to delete an authorization token.

***Task*** ***To delete an authorization token, do the following:***

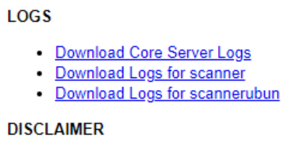
1. Access the **Preferences** page (see Accessing the Preferences Page).
2. From the **AUTH Tokens** pane, locate the token you want to edit, and click the Edit icon (✖).

# Downloading Code Insight Log Files

Code Insight allows Code Insight System Administrators to download Code Insight log files that have been generated for the Core Server and each Scan Server. The downloads are in `.zip` format, enabling you to easily distribute log files as needed for analysis or troubleshooting purposes.

***Task*** ***To download log files, do the following:***

1. Log into Code Insight as a Code Insight System Administrator.
2. Click the icon ☰ in the upper right corner of the Code Insight Web UI, The Code Insight main menu opens.
3. Select **HELP** from the menu to open the **Help** page.
4. Navigate to the **Logs** section, and click the link for the type of logs you want to download:



LOGS

- Download Core Server Logs
- Download Logs for scanner
- Download Logs for scannerubun

DISCLAIMER

The following logs are available:

- **Download Core Server Logs**—The logs generated by the Code Insight Core Server. These include `core.log` and `core.update.log`.
- **Download Logs for {scannerName}**—The logs for the Scan Server whose name is specified in the link label. (A separate download link is generated for each Scan Server configured in your Code Insight system.) Scan Server logs include Tomcat logs, as well as `codeaware.log`, `codeware.update.log`, `scanEngineDetail.log`, and possible archived versions of these logs.

3

# Performing Advanced Searches

This chapter discusses Code Insight's advanced inventory searching capabilities:

- Advanced Searches
- Dependencies in Advanced Searches

## Advanced Searches

Although you could use the simple search on the **Project Inventory** tab to find inventory items that match text strings, Code Insight provides the ability to use additional criteria to display only the items that are of interest. (Simply click the **Advanced Search** button on this tab to access the **Advanced Search** dialog.) Many combinations of search criteria are available, depending upon the type of inventory you want to find. The following table, which is arranged by persona (job function or department), presents a number of advanced searches and their typical results.

For details about using the Advanced Search dialog and all its available search fields, see Searching Published Inventory and Advanced Inventory Search Dialog

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Any** | By inventory, component, or license keyword | Inventory of interest based on full or partial inventory, component or license name.<br>Useful when you want a quick search for a specific component or license across all of your inventory items. | Inventory Name = *zlib 1.2.8 (zlib/ libpng License)*<br>Inventory Name = *zlib*<br>License Name = *EPL* |

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Any** | By criticality (priority) | Most critical inventory that requires security or legal review based on presence of high-severity vulnerabilities or P1 licenses.<br>Useful when you want to prioritize your inventory review by most important findings. | **Option 1:**<br>Inventory Priority = *P1*<br>**Option 2:**<br>Security Vulnerability Severity = *High* or *License Priority = P1 - Viral/Strong Copyleft* |
| **Any** | By review status | Inventory whose status is **Approved**, **Rejected**, or **Not Reviewed** (requires further review.<br>Useful to identify items that are yet to be reviewed. Also when you are further qualifying other search criteria with an additional expression based on review status. | Inventory Review Status = *Approved* |
| **Any** | By dependencies | Only dependency inventory items (both first-level and transitive dependencies), only top-level inventory items (excluding all dependency inventory items), or all inventory items.<br>Useful for focusing on or filtering out dependency inventory items. | Dependency Options = *All Inventory Items*<br>Dependency Options = *Only Top-Level Inventory Items*<br>Only Dependency Inventory Items = *Only Dependency Inventory Items* |
| **Any** | By inventory age | Inventory created within the specified time range.<br>Useful to filter to recent inventory items, which is especially valuable when a user logs into Code Insight at a regular interval (daily, weekly, etc.). | Inventory Age = *Last 7 Days* |

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Any** | By notification | Published inventory items that have new security vulnerability alerts or that have been rejected due to new non-compliant security vulnerabilities. You can select one or both options.<br>Useful for filtering to published inventory items that have important new security information or that have been rejected due to new security issues that are non-compliant with policy. | Inventory with Open Alerts = *checked*<br>Inventory Rejected Due to New Non-Compliant Security Vulnerabilities = *checked* |
| **Any** | By task status | Inventory tasks by their Open or Closed status.<br>Useful for determining the work required before the inventory review process can be completed. Also useful for locating inventory whose closed tasks might need to be reopened for extra work. | Task Status = *Open*<br>Task Status = *Closed* |
| **Any** | By task type | Inventory tasks by their type.<br>Useful for filtering to inventory that requires a manual legal or security review (**Manual Inventory Review**), source-code changes to make it compliant or secure (**Remediate Inventory**), or another type of effort (**Miscellaneous**). | Task Type = *Manual Inventory Review*<br>Task Type = *Remediate Inventory*<br>Task Type = *Miscellaneous*<br>Task Type = *Any* |
| **Any** | By inventory task age | Inventory tasks created within the specified date range.<br>Useful for keeping track of new work to be performed on inventory and old work still needs to be addressed. | Inventory Tasks Age = *Last 7 days*<br>Inventory Tasks Age = *Custom Date Range From: 09/05/2018 To: 10/31/2018* |
| **Any** | By inventory task owner | Inventory tasks owned by a specific user.<br>Useful for determining the workload of a specific user. | Inventory Tasks Owner = *Any*<br>Inventory Tasks Owner = *Only mine* (current user)<br>Inventory Tasks Owner = *<Username>* (selected user) |

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Analyst, Reviewer** | By value in custom inventory fields | Inventory whose custom inventory field values contain a specific string. | <customInventoryFieldLabel> Contains <Search Text> |
| **Analyst, Reviewer** | By Confidence Level | Inventory generated with a specific level of confidence (High, Medium, or Low). The level is based on the measure of the strength of the discovery technique used to generate the item. (See Inventory Confidence in the "Using Code Insight" chapter.)<br>Useful for determining whether the item should be triaged or reviewed to validate or further refine the finding. | Inventory Confidence Level = *High* (or *Medium* or *Low*) |
| **Security Analyst** | By vulnerability ID | Inventory with a specific vulnerability (NVD CVE or Secunia Advisory).<br>Useful when you are looking for inventory exposing you to a specific security issue, typically a newsworthy event. | Security Vulnerability ID = *SA71946* |
| **Security Analyst** | By security risk exposure | Inventory containing security vulnerabilities of a specified severity.<br>Useful to filter to inventory items that require immediate attention based on your corporate security policy. For example, we must address all high-severity security issues in the current release. | Security Vulnerability Severity = *High* |
| **Security Analyst** | By security vulnerability age | Inventory with new security vulnerabilities since a specified date.<br>Useful to see which inventory items have new security vulnerabilities reported against them based on the specified date range. | Security Vulnerability Age = *Last day* |

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Security Analyst** | By security risk exposure and vulnerability age | Inventory with new security vulnerabilities of a specified severity since a specified date.<br>Useful to see which inventory items have new security vulnerabilities reported against them based on the specified date range and a certain severity. | Security Vulnerability Age = *Last day* and Security Vulnerability Severity = *High* |
| **Security Analyst** | By inventory alert | Published inventory items that have new security vulnerability alerts or that have been rejected due to new non-compliant security vulnerabilities. You can select one or both options.<br>Useful for filtering to inventory items that have important new security information or that have been rejected due to new security issues that are non-compliant with policy. | Inventory with Open Alerts = *checked*<br>Inventory Rejected Due to New Non-Compliant Security Vulnerabilities = *checked* |
| **Security Analyst** | By new vulnerabilities (requires re-review) | Inventory that has gained a new security vulnerability since a specified date.<br>Useful to determine which inventory items require another look from a security analyst due to new associated vulnerabilities. | Review Status = *Approved* and *Security Vulnerability Age = Last 7 days* |
| **Legal** | By license risk exposure | Most critical inventory that requires legal review (contains a P1 license - Viral/Strong Copyleft).<br>Useful to prioritize legal work based on license classification. | License Priority = *P1 - Viral/Strong Copyleft* |
| **Analyst** | Requires re-review based on missing license | Approved inventory with a missing license.<br>Useful to catch scenarios where items were approved without an associated license. This should be a rare event. | Inventory Review Status = *Approved* and License Priority = *No License Found* |

**Table 3-1 ▪** Sample Advanced Searches

| Persona | Search Type | Finds... | Example |
|---|---|---|---|
| **Eng. Mgr./ Final Reviewer** | Stop shipment! | Approved inventory that may require a stop shipment due to high severity vulnerability or P1 license.<br>Useful to identify cases that would break the build. These are items that were approved at the time of review, but since then have a different license or high-severity vulnerability. | Inventory Review Status = *Approved* or License Priority = *P1 - Viral/Strong Copyleft* or Security Vulnerability Severity = *High* |

# Dependencies in Advanced Searches

Code Insight is able to scan archived and multi-layer codebases. When inventory items from these codebases are published, dependencies can be published as well. However, when performing searches on published inventory, the amount of data returned can be immense. So it is important to consider whether to include or exclude them in your inventory searches.

4

# Exporting and Importing Project Data

The following sections describe the project export and import functionality in Code Insight:

## About Exporting and Importing

Code Insight provides export and import functionality for project data. The export-import process can be performed on the same server or across servers. Project data can be quickly imported into an empty project to create inventory or imported into a scanned project to create inventory with file associations. The imported inventory in both cases is "live"—that is, ready to be reviewed and edited.

Export and import functionality is useful in any of these following scenarios:

- **Backup of project and audit data**—Use export to create a full backup of a Code Insight project. The backup data file includes project and scan details, all inventory (with inventory details, field values, file associations, and inventory review status), the review status of files, and any custom data. The project data can be restored to a new project for an archived view or for ongoing scanning and auditing.
- **Copying or branching a project**—Use the export-import process to create an exact copy of a project for future scanning and audit work. To do this, export the data from the source project, scan the target project (pointed to the same codebase), and import the data into the target project. The target project can be used for continued scanning and analysis work while the source project remains unchanged.
- **Versioning a project**—Use the export-import process to apply analysis work performed on one product version to the next product version. For example, you can apply the analysis work performed on project "foo-v1" to project "foo-v2". To do this, export the data from "foo- v1", scan "foo-v2", and import the data into "foo-v2".

- **Audit work reuse**—As in the versioning example above, you can use the export-import process to apply analysis work performed on one project to another project containing a subset of similar files. Export data from "project1", scan "project2" (pointed to "project2" codebase), and import the data into "project2". Likewise, this process can be used to apply analysis work from several different projects to the current project.
- **Sharing of live audit results between teams**—Use the export-import process to share live audit results between teams or with Revenera Professional Services. For example, you can export data from "project 1" on "instance 1" and import the data into "project 2" on "instance 2". Results are imported either into an empty project for a live view of inventory or into a scanned project for a live view of inventory with file associations and access to the codebase file tree.

  **Note ▪** *An empty project is one to which no codebase has been uploaded or synchronized or that has not yet been scanned.*

- **Migrating audited projects from Code Insight v6 to v7**—Use the import process to create live project inventory in Code Insight v7 from an exported v6 project or workspace data. After exporting data from a v6 project, run the Audit Data Migration Tool (available for download in the Product and License Center) to map inventory fields from Code Insight v6 to v7 and to convert the exported data to the proper JSON data format required for import into v7. Then perform an import into a Code Insight v7 project. Import into an empty project (see note for previous bullet) for a live view of inventory only or into a scanned project for a live view of inventory with file associations.
- **Creating inventory from an external system**—Use import to create project inventory in Code Insight from a data file containing legacy or external data. This type of import requires the conversion of the legacy data to the required JSON format prior to importing into the new project.

# Export/Import Processes with Legacy Projects

Previous to Code Insight 2020 R3, projects were one of two types—*standard* or *inventory-only*. A standard project managed the results of a server scan—that is, a scan performed by the Scan Server on a codebase that was uploaded or synchronized to the Scan Server. The inventory-only project managed the results of a remote scan, performed by a scan agent on a remote codebase. (The agent sent the scan results to the project on the Code Insight server.) The scan results in both project types included an inventory of open-source and third-party software. Additionally, scan results in the standard project included information about the codebase files, enabling users to perform file operations. The inventory-only project, however, included no information about the remote codebase files.

Beginning with Code Insight 2020 R3, all scanning—server and remote—is accomplished in a single "unified" project, simply called "project" in this documentation. In addition to an inventory of open-source and third-party software, the new project type can include information about codebase files from both server and remote scans, enabling file operations on all project files.

## Migration to the New Project Type

During an upgrade from a pre-2020 R3 Code Insight release to the current release, standard projects are automatically migrated to the new project type.

Inventory-only projects, now called *legacy projects*, continue to be supported with limitations. For example, these projects support only 2020 R2 or earlier scan-agent plugins and allow imports only from other legacy projects. Note that legacy projects will be deprecated in the future.

If you want to migrate an legacy project to the new project type, see the following Knowledge Base article in the Revenera Community:

https://community.flexera.com/t5/FlexNet-Code-Insight-Customer/Code-Insight-2020-R3-Changes-to-Projects/ta-p/160059

### New Export JSON Elements

Starting in Code Insight 2020 R3, the JSON output for an export process run contains the following elements to support remote codebase-file information that can be exported from a Code Insight project created in 2020 R3 or later:

- **remoteAlias**— The unique name for a remote scan agent used by the project
- **remoteScanFolders**—A list containing the scan root for each alias
- **remoteFilePaths** (for inventories)—The list of all remote file paths associated with inventory
- **remoteReviewedPaths**—The list of all reviewed remote files

# Prerequisites When Using the REST Interface

The following are prerequisites when using the Code Insight REST interface to execute an export or import:

- REST Client or Command-Line Tool Supporting cURL
- Authorization Token
- Project ID

These prerequisites are not needed if you are performing an export or import using the Code Insight Web UI.

## REST Client or Command-Line Tool Supporting cURL

A REST client or command-line interface with cURL support is required to execute the cURL commands that call the REST API to export or import project data.

To download cURL, go to https://curl.haxx.se/download.html. Once cURL is installed, ensure that its path is added to your PATH environment variable so that you can use it with batch or PowerShell scripts and call it from the command prompt of any working directory.

## Authorization Token

The export and import REST interface requires a valid JSON Web Token (JWT) for the owner of the project from which data is to be exported or to which data is to be imported, depending on the function being performed. For instructions on obtaining the JWT, see Managing Authorization Tokens in the “Using Code Insight” chapter. The token is not required when using the Code Insight Web UI to export or import project data.

# Project ID

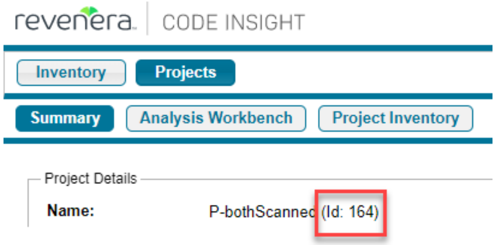
The export and import REST interface requires the ID of the project from which you are exporting data or to which you are importing data, depending on the function being performed. The following procedures describe methods for locating the project ID.

## Locating the Project ID in the Code Insight Web UI

The following are some ways to locate the project ID in the Code Insight Web UI.

***Task*** ***To obtain the project ID through the Web UI, do the following:***

Locate the **Name** value on the **Project Summary** page for the project. The ID is displayed in parentheses next to the name:



**Name:** P-bothScanned (Id: 164)

## Retrieving the Product ID Using the REST Interface

Retrieve the project ID by issuing a cURL command that calls the **Get Project Id** REST API.

***Important*** ▪ *If want to copy and paste the cURL command directly from these instructions, copy it to a text editor first to remove formatting and any line breaks or extra spaces.*

***Task*** ***To obtain the project ID by calling the Get Project Id REST API, do the following:***

Execute the following cURL command to invoke the **Get Project Id** REST API. Replace the highlighted variables with your server host ID (hostname or IP address plus the port), project name, and authorization token.

```
curl -X GET "HOST:PORT/codeinsight/api/project/id?projectName=PROJECT_NAME" -H "accept:
application/json" -H "Authorization: Bearer JWT_TOKEN"
```

The following is an example:

```
curl -X GET "http://localhost:8888/codeinsight/api/project/id?projectName=AllTypes" -H "accept:
application/json" -H "Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsInVzZXJJZCI6MSwiaWF0IjoxNTY2ODU5NTg2fQ.qV2j8ZLgNGNJsT8OdPR
wvE0-0y1x7w-0zr5h7Jz2d9uqY8tvACsV68posEUO9tD-YXlgXznX-IGnrnopDU7G3w"
```

***Note* ▪ If the project name contains a space or special character, replace the character with its encoded version. For example, for the project `project foo`, you would provide the name `project%20foo`, where the space is replaced with the encoded character `%20`.**

The response contains the project ID (in this example, 164):

```
{"Content: ":164}
```

# Exporting Project Data

The following sections provide the details about exporting project data in Code Insight:

- About an Export
- Types of Data Exported
- Prerequisites for Exporting Data
- Exporting Project Data Using the Web UI
- Exporting Project Data Using the REST Interface
- Verifying the Export Results

***Important* ▪ The instructions in this section assume that you are exporting project data using the Code Insight version for which this documentation was published. If you are exporting project data using another version of Code Insight, refer to the documentation for that version for export instructions.**

## About an Export

The Code Insight project-data export feature is available through these interfaces:

- Code Insight Web UI, as described in Exporting Project Data Using the Web UI.
- Code Insight REST interface, as described in Exporting Project Data Using the REST Interface.

During the export process, project data is exported to a JSON data file and then compressed in a `.zip` archive. The archive file can be stored for backup purposes or imported into a project (described in Importing Project Data).

The export runs as a background process that does not interfere with scanning or analysis work.

## Types of Data Exported

The export always processes project data in full—that is, there is no way to limit the exported data. Thus, the data exported includes the following:

- **Export information**—The project contact, the version of Code Insight in which the export was run, the Compliance Library and Electronic Update versions currently used by the project, the date and time of the export, and more.

- **Project details and scan settings**—The name, description, scan profile, policy profile, and scanroot path for the project.
- **Inventory**—All data for the project's inventory, including inventory fields, publication and review statuses, associated codebase files, and associated repository items.
- **Reviewed files**—The absolute file path and MD5 for each project codebase file marked as reviewed.
- **Custom data**—Custom inventory and any custom components, versions, and licenses to which this inventory is mapped.

***Note** ▪ Some data exported is for informational purposes only and is not necessarily processed during an import.*

# Prerequisites for Exporting Data

To export data, ensure that the following prerequisites are met:

- A Code Insight v7 instance currently running.
- An existing, non-empty project (containing at least one inventory item) on that instance.
- Items listed in Prerequisites When Using the REST Interface (if performing the export using REST API).

# Exporting Project Data Using the Web UI

Use the following instructions to export project data using the Code Insight Web UI. The exported data is written a JSON data file, which is then compressed in a `.zip` archive.

***Task*** ***To export project data using the Code Insight Web UI, do the following:***

1. Ensure that all requirements in Prerequisites for Exporting Data are met.
2. Log into Code Insight as the owner of the project you want to export.
3. Navigate to the **Project Summary** tab (see Opening the Project Summary Tab).
4. Open the **Manage Project** dropdown and select **Export Project Data**.

   The project data is exported to a JSON data file, which is then compressed in a `.zip` archive and saved to the download location configured for your browser. Both the archive and the data file it contains use the same default name, which includes the project name and the export timestamp. (Note that, depending on your browser configuration, you might be prompted to provide an archive name and download location before the export runs.)

   The following shows an example archive generated by a Code Insight export on a Windows system:

   ```
   C:\Users\kdr\Downloads\55-export-02-20-2021_10-42.zip
   ```

   The archive would include the following data file:

   ```
   55-export-02-20-2021_10-42.json
   ```

***Note*** ▪ *If an archive with the same name already exists in the download location, the new archive replaces the existing one.*

5. Verify that the export process completed successfully. See Verifying the Export Results.

# Exporting Project Data Using the REST Interface

Use the following information to export project data by issuing a cURL command that calls the **Export Project Data** REST API. During the export process, the project data is written to a JSON data file, which is then redirected to a `.zip` archive.

***Important*** ▪ *If want to copy and paste the cURL command directly from these instructions, copy it to a text editor first to remove formatting and any line breaks or extra spaces.*

***Task*** ***To export project data by calling the exportProjectData REST API, do the following:***

1. Ensure that the requirements listed in Prerequisites for Exporting Data and Prerequisites When Using the REST Interface are met.

2. To initiate the export process, execute the following cURL command to invoke the **Export Project Data** REST API using the GET method.

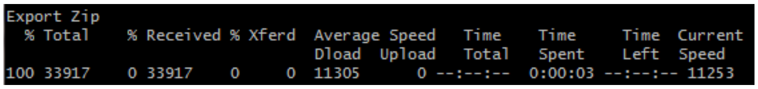
```
curl -X GET "HOSTNAME:PORT/codeinsight/api/project/exportProjectData?projectId=PROJECT_ID" -H
"accept: application/json" -H "Authorization: Bearer JWT_TOKEN" > PROJECT_DATA_FILE.zip
```

In the command syntax, replace the highlighted variables with your server host name (machine name or IP address) along with the port, the ID of the project you are exporting, and your authorization token. Also, replace `PROJECT_DATA_FILE` with the name that you want to use for the `.zip` archive to which the data file containing the exported data will be redirected. (The data file will use the same name, but with a `.json` extension.)

The following is an example:

```
curl -X GET "http://localhost:8888/codeinsight/api/project/exportProjectData?projectId=164" -H
"accept: application/json" -H "Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsInVzZXJJZCI6MSwiaWF0IjoxNTY2ODU5NTg2fQ.qV2j8ZLgNGNJsT8
OdPRwvE0-0y1x7w-0zr5h7Jz2d9uqY8tvACsV68posEUO9tD-YXlgXznX-IGnrnopDU7G3w" > ProjectKDR.zip
```

The status of the export process appears in the command prompt window:



```
Export Zip
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 33917    0 33917    0     0  11305      0 --:--:--  0:00:03 --:--:-- 11253
```

When the export completes, the `.zip` archive containing the exported data is found in the directory from which the export command was executed. For example, if the export command was executed from the `C:/fnci/project_export` directory and the output redirect value is `ProjectKDR-export-02-20-2021_10-42.zip`, the following is the archive location:

```
C:/fnci/project_export/ProjectKDR-export-02-20-2021_10-42.zip
```

The name of the data file in the archive is `ProjectKDR-export-02-20-2021_10-42.json`.

***Note*** ▪ *If an archive with the same name already exists in the download location, the new archive replaces the existing one.*

3. Verify that the export process completed successfully. See Verifying the Export Results.

## Verifying the Export Results

Use this procedure to verify that the export process completed successfully or that the exported data is without errors.

***Task*** ***To verify whether an export process was successful, do the following:***

1. Unzip the archive.

   If the export process did not complete successfully, the `.zip` archive will include a file that contains an appropriate status code and error message. Continue with step 2.

   Otherwise, you can go ahead and import the contents of the `.zip` file into a target Code Insight project or use the`.zip` file for backup purposes. However, you can always proceed with step 2 to ensure that the exported data is without errors.

2. To troubleshoot, open the JSON data file with a utility that supports JSON, such as Textpad or Notepad++, to determine noticeable errors.

3. If errors exist, resolve them (for example, modifying the project you exported and rescanning it), delete the invalid archive file, and rerun the export.

# Importing Project Data

The following sections provide the details about importing project data:

- About an Import
- Prerequisites for Importing Data
- Import Behavior and Configuration
- Importing Project Data Using the Web UI
- Importing Project Data Using the REST API
- Verifying the Import Results

***Important*** ▪ *The instructions in this section assume that you are importing project data using the Code Insight version for which this documentation was published. If you are importing project data using another version of Code Insight, refer to the documentation for that version for import instructions.*

# About an Import

The Code Insight project-data import feature is available through these interfaces:

- Code Insight Web UI, as described in Importing Project Data Using the Web UI
- Code Insight REST interface, as described in Importing Project Data Using the REST API

The following sections provide overview information about the import process.

## Input Used in the Import Process

The input for a project data import is an archived JSON data file containing project data. During the import process, this JSON data file (called the *import data file* in this documentation) is extracted from the archive and imported to a specific Code Insight project, called the *target project*.

You can create this input archive in a couple of ways:

- Use the Code Insight project-data export feature (see Exporting Project Data) to export data from a Code Insight v7 project.
- Export data from a Code Insight v6 project (see the Code Insight 6.14.x documentation for instructions). Then use the Audit Migration Tool to convert the exported data into the format required for importing in a v7 project.
- Manually create the import data file using data exported from an external system and formatting it as JSON code. Then compress the data file as a `.zip` archive.

In all cases, the input for the import process must be a data file whose contents are in the expected JSON format and that has been archived as a `.zip` file.

## Target Projects

The project data import is performed on a project—called the *target project*—that has been created in Code Insight. This target can contain the results of a server or remote scan (or both types of scans) performed on the target codebase prior to the import. (See Scan Types for a description of server and remote scans.) The import process creates inventory with file associations and provides access to codebase files through the **Analysis Workbench** in the target project. The inventory is live—that is, ready to be edited and reviewed.

## About Importing Data to Current and pre-2020 R3 Projects

The following are the allowed and disallowed import scenarios involving current projects and projects created before Code Insight 2020 R3. (Also see Legacy Projects.)

- You can import data exported from a project created in 2020 R3 or later into another 2020 R3 or later project.
- You can import data exported from a project created in 2020 R3 or later to a standard project (created in 2020 R2 or earlier) that has been migrated to 2020 R3 or later.
- You cannot import data exported from a project created in 2020 R3 or later to a legacy inventory-only project that has been migrated to 2020 R3 or later.

# Prerequisites for Importing Data

To import data, ensure that the following prerequisites are met:

- A Code Insight v7 instance currently running.
- An existing Code Insight project to which data will be imported (that is, a target project).

  **Note ▪** *Ensure that a scan is not running on this project during the import.*

- A `.zip` archive containing the JSON data file with the project data to be imported (see Input Used in the Import Process).
- A completed Electronic Update for the Code Insight system to which the target project belongs.
- Items listed in Prerequisites When Using the REST Interface (if performing the import using REST API).

# Import Behavior and Configuration

The following sections provide information you should know about the Code Insight import behavior and ways to configure this behavior.

- About File Processing During an Import
- Default Criteria for Handling File and Inventory Comparisons During the Import
- Available Import Options to Configure Import Behavior
- Other Import Considerations

## About File Processing During an Import

Depending on the import configuration, the import process might need to match files paths in the import data file with file paths in the target project codebase to do the following:

- Create new file associations in inventory in the target project.
- Mark reviewed files in the import data file as reviewed in the target project.
- Determine empty files.

The file path and the file MD5 value are the key criteria used to locate target codebase files that match files in the import data file. When the MD5 value is used as a criterion, the MD5 for a file in the import data file must have an exact MD5 match in the target codebase. However, when the file path is used as a criterion, the file-matching process can apply various rules.

### File-Path Processing

When the file path is used as a criterion for matching, the import process internally subtracts the root path from the absolute path of codebase files in the import data file and in the target project. The result is the *complete file* path for a given file, as illustrated in these examples:

- **Absolute file path**—`/home/fnci/scanRoot/1/ePortal-1.3/src/gettext.c`

- **Root path**—`/home/fnci/scanRoot/1/`
- **Complete File path**—`/ePortal-1.3/src/gettext.c`

Then, based on the file-path criterion selected by the user, the import locates matching files by searching complete file paths, partial paths, or simply file names. The following examples illustrate a complete path in comparison with a partial path or file name:

- **Complete path**—`/ePortal-1.3/copy1/src/gettext.c`
- **Partial path**—`/copy1/src/gettext.c` **or** `/src/gettext.c`)
- **File name**—`gettext.c`

When users select the partial path as a criterion for matching files, they must also provide a desired directory depth that defines the partial path.

## Default Criteria for Handling File and Inventory Comparisons During the Import

The following information describes the *default* criteria used for handling file and inventory comparisons during an import. You can modify this criteria as needed by reconfiguring import options, as described in the next section, Available Import Options to Configure Import Behavior.

- Only those files whose complete files paths match in both the import data file and the scanned target codebase can be associated with inventory in the target project. See Option for Creating New File Associations in Target Inventory for more information.
- Only those files whose MD5s and complete paths match in both the import data file and the scanned target codebase are processed when marking files as reviewed in the target project. See Option for Marking Target Codebase Files as Reviewed for more information.
- Only those inventory items in the import data file whose associated files match file paths in the target codebase are created in the target project (if these inventory items do not already exist in the target project). See Option to Create Empty Inventory for more information. (The import REST interface uses this behavior as the default; the Web UI might use different default behavior based on project settings.)
- When an inventory item in the import data file is an exact match to an inventory item in the target project, any notes content defined for the inventory item in the import data file overwrites the notes content defined in the target inventory item. See Options for Handling Inventory Notes for more information.

The default import logic is configurable, as described in the next section.

## Available Import Options to Configure Import Behavior

The following options can be specified to override the default import behavior described in the previous section.

- Option for Creating New File Associations in Target Inventory
- Option for Marking Target Codebase Files as Reviewed
- Option to Create Empty Inventory
- Options for Handling Inventory Notes

- Options for Handling Inventory Usage Values

Additionally, the section Specifying File-Matching Criteria in the Import REST Interface provides more details about the file-matching criteria used when creating new file associations in target inventory or when marking target codebase files as reviewed.

The information in this section directly applies to the procedures on how to execute an import using the Web UI or the REST interface described later in this chapter. However, because the configuration of import behavior is a thoughtful process and configurations must be set up before initiating an import, the configuration options in detail are described here.

## Option for Creating New File Associations in Target Inventory

The import Web UI and REST interface provide options to set the file-matching criteria needed by the import to create new file associations in target project inventory when that inventory is identical to inventory in the data import file. (For a description of *identical* inventory items, see Identical Inventory.) The import process will create a new file association in the identical target inventory item only if the defined file-matching criteria is met. If a file association in the import data file already exists in the target inventory item, no new file association is added to the target inventory item.

By default, a file associated with an inventory item in the import data file can be added to an identical inventory item in the target project codebase only if the file's complete path matches in both the import data file and in the target project codebase. For example, a file with a complete path of `/ePortal-1.3/src/gettext.c`, listed in the import data file as belonging to "Inventory Item 1", will be considered for association with this same inventory item in the target project only if the target project codebase contains a file with the same complete path, `/ePortal-1.3/src/gettext.c`.

However, you can set different criteria for adding associated files to target inventory, such as requiring that only partial paths or MD5 values match or requiring that both MD5 values and paths match. For more details, see the following topics:

- "Add Files to Inventory" Option in the Web UI
- "addFilesToInventory" Attribute in the REST Interface

***Note* ▪ *The same file-matching criteria defined for creating new file associations in target inventory is also used in determining empty inventory. See Option to Create Empty Inventory.***

### "Add Files to Inventory" Option in the Web UI

The field **Add Files to Inventory** on the **Import Project Data** dialog is used to set the file-matching criteria for creating new file associations in target inventory. For a description of the criteria options available with this field, see Import Project Data Dialog. For complete instructions on using the Web UI to import project data, see Importing Project Data Using the Web UI.

### "addFilesToInventory" Attribute in the REST Interface

The `addFilesToInventory` attribute is used in the **Import Project Data** REST API to set the file-matching criteria for creating new file associations in inventory. The following sections provide attribute details:

- About the "addFilesToInventory" Attribute

- Example “addFilesToInventory” Syntax in Import cURL Command

For instructions on executing **Import Project Data** API, see Importing Project Data Using the REST API.

### About the “addFilesToInventory” Attribute

The `addFilesToInventory` attribute must be set to `true` to enable the import process to add files to inventory in the target project. Along with this attribute, a second attribute, `inventoryFileMatchingCritieria`, must be included to set the file-matching criteria used to determine whether a given file can be added to target inventory. A third attribute is required to set the directory depth if you specify partial-path criteria. For more information about setting file-matching criteria, see Specifying File-Matching Criteria in the Import REST Interface.

If the user explicitly sets the `addFilesToInventory` attribute to `false` (or omits this attribute entirely), the import does not associate any additional files to inventory in the target project.

### Example “addFilesToInventory” Syntax in Import cURL Command

The following shows an example of the `addFilesToInventory` attribute in a cURL command that calls the REST `import` endpoint:

```
curl -H "Authorization:Bearer  %jwt%" -F importFile=@"FileToImport.zip" -F projectImportModel={
"createEmptyInventory” : true, “overwriteInventoryNotes” : true, “addFilesToInventory” : true,
“inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”, “inventoryDirectoryDepth” : 2,
“markFilesAsReviewed” : true, “reviewFileMatchingCriteria” : “PARTIAL_FILEPATH”,
“reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/json  http://
hostname:8888/codeinsight/api/projects/{projectId}/import
```

For complete instructions about using the cURL command to execute an import, see Importing Project Data Using the REST API.

For details about the implementation of the **Import Project Data** REST API, access the Code Insight Swagger documentation. To do so, click the ☰ icon in the upper right corner of the Code Insight Web UI, select **Help**, and click the **REST API Guide** link.

## Option for Marking Target Codebase Files as Reviewed

The import Web UI and REST interface provide options to set the criteria needed to mark file target codebase files as reviewed.

***Note** ▪ For this option, the import compares only those files in the import data file that are marked as reviewed with files in the target codebase.*

By default, the import can mark an unreviewed file in the target codebase as reviewed only if a file in the import data file has the same MD5 and complete file path as the target file. For example, suppose a file marked as “unreviewed” in the target project codebase has a complete path of `/ePortal-1.3/src/gettext.c`. The import can mark this file as reviewed only if the import data file contains a file whose complete path is `/ePortal-1.3/src/gettext.c` and whose MD5 is the same as the MD5 of the file in the target project codebase.

However, you can set different import criteria for marking files in the target project codebase as reviewed, such as requiring that only partial paths or only MD5 values match or requiring that both MD5 values and partial paths match. For more details, see the following topics:

- “Add Files to Inventory” Option in the Web UI

- markFilesAsReviewed Attribute in the REST Interface

## “Mark Files as Reviewed” Option in the Web UI

The field **Mark Files as Reviewed** on the **Import Project Data** dialog is used to set the file-matching criteria for marking target codebase files as reviewed during the import process. For a description of the criteria options available with this field, see Import Project Data Dialog. For complete instructions on using the Web UI to import project data, see Importing Project Data Using the Web UI.

## markFilesAsReviewed Attribute in the REST Interface

The markFilesAsReviewed attribute is used in the **Import Project Data** REST API to set the file-matching criteria for marking target codebase files as reviewed during the import process. The following sections provide attribute details:

- About the “markFilesAsReviewed” Attribute
- Example “markFilesAsReviewed” Syntax in Import cURL Command

For instructions on executing **Import Project Data** API, see Importing Project Data Using the REST API.

### About the “markFilesAsReviewed” Attribute

The markFilesAsReviewed attribute must be set to true to enable the import process to mark reviewed files in the import data file as reviewed in the target project. Along with this attribute, a second attribute, reviewFileMatchingCritieria, must be included to set the file-matching criteria used to determine whether a given unreviewed file in the target codebase can be flagged as “Reviewed”. A third attribute is required to set the directory depth if you specify partial-path criteria. For more information about setting file-matching criteria, see Specifying File-Matching Criteria in the Import REST Interface.

If the user explicitly sets the markFilesAsReviewed attribute to false (or omits this attribute entirely), the import does not change the “Reviewed” or “Not Reviewed” status of files in the target project codebase. The statuses remain as they were before the import process was performed.

### Example “markFilesAsReviewed” Syntax in Import cURL Command

The following shows an example of the explicit use of the markFilesAsReviewed attribute in a cURL command that calls the REST import endpoint:

```
curl -H "Authorization:Bearer  %jwt%" -F importFile=@"FileToImport.zip" -F projectImportModel={
"createEmptyInventory” : true, “overwriteInventoryNotes” : true, “addFilesToInventory” : true,
“inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”, “inventoryDirectoryDepth” : 2,
“markFilesAsReviewed” : true, “reviewFileMatchingCriteria” : “PARTIAL_FILEPATH”,
“reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/json  http://
hostname:8888/codeinsight/api/projects/{projectId}/import
```

For complete instructions about using the cURL command to execute an import, see Importing Project Data Using the REST API.

For more details about the implementation of the **Import Project Data** REST API, access the Code Insight Swagger documentation. To do so, click the ☰ icon in the upper right corner of the Code Insight Web UI, select **Help**, and click the **REST API Guide** link.

## Option to Create Empty Inventory

The import Web UI and REST interface provide an option to specify whether “empty” system-generated inventory items are still processed in the target project during the import. Empty inventory items either have no file associations in the data file to be imported or *do* have associated files in the import data file but no matching paths or MD5s for these files in the target project codebase (also see Complete vs Partial Paths in the Import Path-Matching Process).

When this option is enabled, all inventory items in the import data file—with or without matching associated files in the target codebase—are created in the target project during the import.

When the option is disabled, the import process does not create empty inventory items in the target project. It creates only inventory items whose associated files are found in the target codebase. To define what constitutes “matching files” between the import data file and the target project codebase, this option depends on the criteria set for the Option for Creating New File Associations in Target Inventory.

***Note* ▪ *If you are importing from a scanned project into an inventory-only project, which has no codebase, ensure this option is enabled so that inventory is generated in the new project.***

For more details about this option, see the following topics:

- Option in the Web UI
- “createEmptyInventory” Attribute in the REST Interface

### Option in the Web UI

The option to create empty inventory is available in the Web UI as the project setting, **On the data import or rescan, delete inventory with no associated files**, located on the Edit Project: General Tab. Ensure that this field is properly set for the import you are about to perform. (You can always reset this value for the project once the import is complete.) See Editing the Project Definition and General Settings for details.

The default for this setting is defined at a global-project level by the Code Insight System Administrator. (The initial global setting disables the creation of empty inventory, but obviously this can be changed at the administrator’s discretion to affect all projects.)

### “createEmptyInventory” Attribute in the REST Interface

The option to create empty inventory is available as the `createEmptyInventory` attribute when invoking the **Import Project Data** REST API. The following sections provide attribute details:

- About the “createEmptyInventory” Attribute
- Example “createEmptyInventory” Syntax in Import cURL Command

#### About the “createEmptyInventory” Attribute

To enable the creation of empty inventory items in the target project, explicitly include the `createEmptyInventory` attribute and set it to `true` in the cURL command that calls the `import` REST endpoint. All inventory items in the import data file—with or without matching associated files in the target codebase—are created in the target project during the import.

To disable the creation of empty inventory items in the target project, explicitly include the `createEmptyInventory` attribute and set it to `false` in the cURL command. Only those inventory items with associated files in the import data file that match files in the target codebase are created in the target project.

If you omit the `createEmpyInventory` attribute entirely from cURL command, the import process uses the value of the `deleteEmptyInventory` attribute defined for the project (the same setting as the **On the data import or rescan, delete inventory with no associated files** value on the Edit Project: General Tab.) To override this project setting for the current import process only, explicitly include the `createEmptyInventory` attribute with the appropriate setting when invoking the import endpoint.

### Example “createEmptyInventory” Syntax in Import cURL Command

The following example shows the attribute explicitly included in the cURL command that calls the `import` REST endpoint:

```
curl -H "Authorization:Bearer  %jwt%" -F importFile=@"FileToImport.zip" -F projectImportModel={
"createEmptyInventory” : true, “overwriteInventoryNotes” : true, “addFilesToInventory” : true,
“inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”, “inventoryDirectoryDepth” : 2,
“markFilesAsReviewed” : true, “reviewFileMatchingCriteria” : “PARTIAL_FILEPATH”,
“reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/json  http://
hostname:8888/codeinsight/api/projects/{projectId}/import
```

For complete instructions about using the cURL command to execute an import, see Importing Project Data Using the REST API.

For more details about the implementation of the **Import Project Data** REST API, access the Code Insight Swagger documentation. To do so, click the ☰ icon in the upper right corner of the Code Insight Web UI, select **Help**, and click the **REST API Guide** link.

## Options for Handling Inventory Notes

The import Web UI and REST interface enable you to specify whether the content of notes fields in an inventory item in the target project should be overwritten with the notes content for an identical inventory item in the import data file. As an alternative to overwriting notes, the notes content for a given inventory item in the import data file can be *appended* to existing notes content in the identical target inventory. (For a description of *identical* inventory items, see Identical Inventory.)

This configuration applies to the following inventory notes fields:

- **Notices Text**
- **Audit Notes**
- **Usage Guidance**
- **Remediation Notes**

For more details about this option to overwrite target inventory notes, see the following topics:

- “Inventory Notes Handling” Options in the Web UI
- “overwriteInventoryNotes” Attribute in the REST Interface

## “Inventory Notes Handling” Options in the Web UI

Select the appropriate **Inventory Notes Handling** option on the **Import Project** dialog to configure whether the contents of inventory notes fields are overwritten in target inventory during the import. For a description of this option, see Import Project Data Dialog. For complete instructions on using the Web UI to import project data, see Importing Project Data Using the Web UI.

## “overwriteInventoryNotes” Attribute in the REST Interface

The option to overwrite target inventory notes is available as the `overwriteInventoryNotes` attribute in the **Import Project Data** REST API. The following sections provide attribute details:

- About the “overwriteInventoryNotes” Attribute
- Example “overwriteInventoryNotes” Syntax in the Import cURL Command

### About the “overwriteInventoryNotes” Attribute

To overwrite the content of the notes fields in the target inventory with notes content for identical inventory in the import data file, explicitly include the `overwriteInventoryNotes` attribute and set it to **`true`** in the cURL command that calls the `import` REST endpoint. Note that, if the data for a given notes field is blank in the import data file, no overwrite occurs; any existing content for the field in the target inventory is retained.

To append notes from the source inventory item to the end of the existing notes in the identical target inventory item, explicitly include the `overwriteInventoryNotes` attribute and set it to **`false`** (or omit the attribute entirely). When content is appended in a given notes field in the target inventory item, it is separated from the existing content with a line break and the following heading:

```
Copied during import from <ProjectName>:<InventoryName> (TimeStamp)
```

However, if the note content for a given notes field is the same in both the import data file and the target inventory item, no content is appended in that field in the target inventory item.

### Example “overwriteInventoryNotes” Syntax in the Import cURL Command

The following example shows the attribute explicitly included in the cURL command that calls the `import` REST endpoint:

```
curl -H "Authorization:Bearer  %jwt%" -F importFile=@"FileToImport.zip" -F projectImportModel={
"createEmptyInventory” : true, “overwriteInventoryNotes” : false, “addFilesToInventory” : true,
“inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”, “inventoryDirectoryDepth” : 2,
“markFilesAsReviewed” : true, “reviewFileMatchingCriteria” : “PARTIAL_FILEPATH”,
“reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/json  http://
hostname:8888/codeinsight/api/projects/{projectId}/import
```

For complete instructions about using the cURL command to execute an import, see Importing Project Data Using the REST API.

For more details about the implementation of the **Import Project Data** REST API, access the Code Insight Swagger documentation. To do so, click the ☰ icon in the upper right corner of the Code Insight Web UI, select **Help**, and click the **REST API Guide** link.

## Options for Handling Inventory Usage Values

The import Web UI and REST interface enable you to specify whether the import process should copy the current inventory Usage values to the target project or reset all values for imported inventory to the system default value, `Unknown`. For details, see the following topics:

- "Inventory Usage Handling" Options in the Web UI
- "resetInventoryUsage" Attribute in the REST Interface

The default import behavior is to reset Usage values for imported inventory to the default value, `Unknown`.

### "Inventory Usage Handling" Options in the Web UI

Select the appropriate **Inventory Usage Handling** option (**Reset usage field values to system default** or **Copy existing usage field values**) on the **Import Project** dialog to configure how inventory usage values should be handled. For a description of these two options, see Import Project Data Dialog. For complete instructions on using the Web UI to import project data, see Importing Project Data Using the Web UI. The default is to reset all values to the default **Unknown**.

### "resetInventoryUsage" Attribute in the REST Interface

The attribute `resetInventoryUsage` in the **Import Project Data** REST API is used to determine how the import process should handle Usage values for imported inventory. The following sections provide attribute details:

- About the "resetInventoryUsage" Attribute
- Example "resetInventoryUsage" Syntax in the Import cURL Command

#### About the "resetInventoryUsage" Attribute

If you set the `resetInventoryUsage` attribute to **`true`** (or omit the attribute entirely) in the cURL command that calls the `import` REST endpoint, the import process resets all Usage values for imported inventory to the system default value, `Unknown`. (This is the default behavior.)

Conversely, to configure the import process to copy existing Usage values to the imported inventory, you must explicitly include the `resetInventoryUsage` attribute in the cURL command and set it to **`false`**.

#### Example "resetInventoryUsage" Syntax in the Import cURL Command

The following example shows the attribute explicitly included in the cURL command that calls the `import` REST endpoint:

```
curl -H "Authorization:Bearer  %jwt%" -F importFile=@"FileToImport.zip" -F projectImportModel={
"createEmptyInventory” : true, “overwriteInventoryNotes” : false, “addFilesToInventory” : true,
“inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”, “inventoryDirectoryDepth” : 2,
“markFilesAsReviewed” : true, “reviewFileMatchingCriteria” : “PARTIAL_FILEPATH”,
“reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/json  http://
hostname:8888/codeinsight/api/projects/{projectId}/import
```

For complete instructions about using the cURL command to execute an import, see Importing Project Data Using the REST API.

For more details about the implementation of the **Import Project Data** REST API, access the Code Insight Swagger documentation. To do so, click the ☰ icon in the upper right corner of the Code Insight Web UI, select **Help**, and click the **REST API Guide** link.

## Specifying File-Matching Criteria in the Import REST Interface

If the addFilesToInventory attribute (see Option for Creating New File Associations in Target Inventory) or the markFilesAsReviewed attribute (see Option for Marking Target Codebase Files as Reviewed) is set to **true**, you must include an additional attribute that defines the file-matching criteria needed to compare files in the import data file with the target codebase files:

- For addFilesToInventory, include the inventoryFileMatchingCritieria attribute.
- For markFilesAsReviewed, include the reviewFileMatchingCritieria attribute.

***Note*** *▪ This ...FileMatchingCriteria attribute is required whenever addFilesToInventory or markFilesAsReviewed is set to* **true***. If this attribute is omitted, an error occurs when you attempt to execute the import command.*

If you are using the **Import Project Data** REST API to execute the import process, refer the information in this section for details about the criteria.

If you are performing the import through the Code Insight Web UI, refer to the Import Project Data Dialog for criteria descriptions.

### Available File-Matching Criteria in the Import REST Interface

The following describes the available criteria for the inventoryFileMatchingCritieria attribute or reviewFileMatchingCritieria attribute used to locate file matches between the import data file and the target project codebase.

**Table 4-1 ▪** Possible Values for File-Matching Criteria

| Attribute Value | Description |
|---|---|
| MD5_AND_COMPLETE_FILEPATH | A file's MD5 value and complete path (including the file name) in the import data file must match the MD5 and complete path of a file in the target project codebase. |
| MD5_AND_PARTIAL_FILEPATH | A file's MD5 value and partial path (including the file name) in the import data file must match the MD5 and partial path of a file in the target project codebase.<br>For this criterion, you must include an additional attribute to define the depth of the partial path. See Specifying Directory Depth for Partial-Path Criteria in the REST Interface for more information. |
| MD5_AND_FILENAME | The MD5 and name of a file in the import data file must match the MD5 and name of a file in the target project codebase. |
| MD5 | A file's MD5 value in the import data file must match an MD5 in the target project codebase. |
| COMPLETE_FILEPATH | The complete path of a file (including the file name) in the import data file must match a complete file path in the target project codebase. |

**Table 4-1** ▪ Possible Values for File-Matching Criteria (cont.)

| Attribute Value | Description |
|---|---|
| PARTIAL_PATH | A file's partial path (including of the file name) in the import data file must match the partial path of a file in the target project codebase.<br>For this criterion, you must include an additional attribute to define the depth of the partial path. See Specifying Directory Depth for Partial-Path Criteria in the REST Interface for more information. |
| FILENAME | The name of the file in the import data file must match a file name in the target project codebase. (No path is compared in the file-matching process.) |

### Specifying Directory Depth for Partial-Path Criteria in the REST Interface

If you specify `MD5_AND_PARTIAL_PATH` or `PARTIAL_PATH` as the value for the `inventoryMatchingCriteria` or the `reviewFileMatchingCriteria` attribute, you must include another attribute that defines the directory depth by which to match the partial paths:

- For the `inventoryMatchingCriteria`, include the `inventoryDirectoryDepth` attribute.
- For `reviewFileMatchingCriteria`, include the `reviewDirectoryDepth` attribute.

***Note*** ▪ *This `...DirectoryDepth` attribute is required whenever a partial-path criterion is specified for file matching. If this attribute is omitted, an error occurs when you attempt to execute the import command.*

Provide a value 1 through 20 to designate the number of directories above the file name that must be the same when matching file paths in the import data file with file paths in the target project codebase.

For example, suppose a file has a complete path `/ePortal-1.3/copy1/src/gettext.c`. If a partial path criterion is set with a directory depth of 2 (that is, 2 directories above the file name), the partial path `copy1/src/gettext.c` in the import data file must match the same path in the target project codebase to meet the criterion.

The partial-path depth enables the import to match files when the codebase location is different between the target project codebase and the project codebase whose scanned results are included in the import data file. See also Complete vs Partial Paths in the Import Path-Matching Process.

## Other Import Considerations

Consider the import behavior when it processes the following:

- Complete vs Partial Paths in the Import Path-Matching Process
- Identical Inventory
- Unreviewed Files

## Complete vs Partial Paths in the Import Path-Matching Process

When the import process considers whether to add a file to inventory or mark it as reviewed, the file-matching criteria can mandate that the path of the file match between the import data file and the target project. For example, if the file-matching criteria requires that the *complete* paths of files match, the file `/ePortal-1.3/src/gettext.c`—the only file belonging to "InventoryItem 1.0 (License1)" in the import data file—is considered to be a different file from `/ePortal-2.0/src/gettext.c` in the target project. As a result, `ePortal-2.0/src/gettext.c` cannot be associated with "InventoryItem 1.0 (License1)"; and, if the `createEmptyInventory` value is **`false`**, "InventoryItem 1.0 (License1)" is not created in the target project since it has no associated file. Additionally, `ePortal-2.0/src/gettext.c` will not be marked as reviewed in the target project.

In order for a file in the target project to be treated as identical to a file in the data file, the paths for the two files must match in both locations. To accomplish this, you can use "partial path" and its directory depth as the file-matching criteria. In the case above, you would ensure that the file paths match by selecting "partial path" and setting its directory depth to 1 (one directory above the file name). In this way, the import is matching only the partial path /src/gettect.c for both files. Alternatively, you can manually manipulate paths in the import data file to match those in the target project, but you are strongly recommended to apply the "partial path" and directory depth criteria instead.

## Identical Inventory

An inventory item in the source project is considered identical to an inventory item in the target project if both items are associated with the same unique combination of component-version-license (CVL). By default, identical inventory is reconciled during the import by overriding all fields in the target inventory item with information from the import data file. In cases where the source inventory item has empty fields, the data in the target inventory item will be left as is (that is, will not be removed).

However, Code Insight does provide the option to *append* the contents of notes fields in the import data file to target inventory. See Options for Handling Inventory Notes for details.

## Unreviewed Files

A codebase file that was flagged as "unreviewed" in the source Code Insight project (that is, the project from which the import data file was created) does not retain its unreviewed status in the target project if the target codebase scan has marked the corresponding target file as reviewed. This occurs because the import data file stores information for only those codebase files that have been marked as reviewed in the source project. Hence, no information about the unreviewed file exists in the import data to overwrite information for this same file in the target project.

If, during a standard import, you want the target project codebase to retain the reviewed and unreviewed status of codebase files in the source project, manually flag all of the codebase files in the target project as "unreviewed" before importing the data file. In this way, any unreviewed files in the source project remain unreviewed in the target project; files marked as reviewed in the source project are included in the import data file and will be marked as reviewed in the target project if they meet the import file-matching criteria.

# Importing Project Data Using the Web UI

Use the following instructions to import project data using the Code Insight Web UI.

***Task*** ***To import project data using the Code Insight Web UI, do the following:***

1. Ensure that all requirements in Prerequisites for Importing Data are met.
2. Log into Code Insight as the owner of the project you want to which you are importing data.
3. Navigate to the **Project Summary** tab (see Opening the Project Summary Tab).
4. Open the **Manage Project** dropdown and select **Import Project Data**.
5. Click **Browse** next to the **Choose File to Import** field to search for and select the `.zip` file containing the JSON data file you are importing.
6. For a description of the remaining fields, refer to Import Project Data Dialog.
7. Click **OK** to perform the import.
   - If the import fails for some reason, as error dialog is displayed. Click **OK** and attempt the import again.
   - If the import completes successfully, a message dialog is displayed, stating as such. Click **OK**.
8. Verify that the import results are what you expected. See Verifying the Import Results.

# Importing Project Data Using the REST API

Use the following instructions to import project data by explicitly executing a cURL command that calls the **Import Project Data** REST API. (Alternatively, you can invoke this API using an API client such as Postman. See Importing Project Data Using the Postman API Client.)

***Note* ▪ *If copying the cURL command directly from the following instructions for your own use, copy it to a text editor first to remove formatting and any line breaks or extra spaces.***

***Task*** ***To run an import, do the following:***

1. Ensure that all prerequisites in Prerequisites for Importing Data and Prerequisites When Using the REST Interface are met.
2. Set up a cURL command to invoke the **Import Project Data** REST API (`import` endpoint). You choose one of these methods in which to execute the command:
   - Explicitly Provide the Import Attributes in the cURL Command
   - Point the cURL Command to a File Containing the Import Attributes
3. Execute the command.

   When the import is complete, a status message with **OK** will appear in the command prompt window. If the import is not successful, a status code and error message is displayed.

4. Verify that the import results are what you expected. See Verifying the Import Results.

# Explicitly Provide the Import Attributes in the cURL Command

One option for setting up the cURL command that invokes the **Import Project Data** REST API (`import` endpoint) is to explicitly include the import attributes in the command. For the complete instructions on running the import using this command, return to Importing Project Data Using the REST API.

The following shows the cURL command syntax with the available import attributes:

```
curl -H "Authorization:Bearer  JWT_TOKEN" -F importFile=@"FILE_TO_IMPORT.zip" -F
projectImportModel={ \"createEmptyInventory\” : true/false, \“overwriteInventoryNotes\” : true/
false, \“addFilesToInventory\” : true/false, \“inventoryFileMatchingCriteria\” :
\“FileMatchingCriteria\”, \“inventoryDirectoryDepth\” : 1-20, \“markFilesAsReviewed\” : true/false,
\“reviewFileMatchingCriteria\” : \“FileMatchingCriteria\”, \“reviewDirectoryDepth\” : 1-20,
\“resetInventoryUsage\” : true/false };type=application/json  http://HOSTNAME:PORT/codeinsight/api/
projects/PROJECT_ID/import
```

This next code excerpt is an example of the cURL command. Refer to Available Import Options to Configure Import Behavior for details about the available import attributes.

```
curl -H "Authorization:Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsInVzZXJJZCI6MSwiaWF0IjoxNTY2ODU5NTg2fQ.qV2j8ZLgNGNJsT8OdPR
wvE0-0y1x7w-0zr5h7Jz2d9uqY8tvACsV68posEUO9tD-YXlgXznX-IGnrnopDU7G3w" -F importFile=@"1184-export-4-
02-20-2021-40.zip" -F projectImportModel={ "createEmptyInventory” : true, “overwriteInventoryNotes”
: false, “addFilesToInventory” : true, “inventoryFileMatchingCriteria” : “MD5_AND_PARTIAL_FILEPATH”,
“inventoryDirectoryDepth” : 2, “markFilesAsReviewed” : true, “reviewFileMatchingCriteria” :
“PARTIAL_FILEPATH”, “reviewDirectoryDepth” : 2, "resetInventoryUsage": false" };type=application/
json  http://localhost:8888/codeinsight/api/projects/217/import
```

Note the following about the example command. Users must explicitly provide their own values in the command based on their environment.

- `JWT_TOKEN` has been replaced with an example user authorization token.
- `HOSTNAME:PORT` has been replaced with an example machine name and port—in this case, `localhost:8888`. (The IP address for the machine can be used instead of the machine name.)
- `FILE_TO_IMPORT` has been replaced with the name of an import data file (that is, the file name of the zip file to be imported) in the example. In this case, the import data file is `1184-export-02-20-2021_09-40.zip`. The entire file name must be enclosed in quotes.
- `PROJECT_ID` has been replaced with the ID of the project to which data is being imported (in this case, `217`). See Project ID for help on obtaining the project ID.
- The value for the `inventoryFileMatchingCriteria` and `reviewFileMatching` attributes must be enclosed in quotes.

# Point the cURL Command to a File Containing the Import Attributes

Another option for setting up the cURL command that invokes the **Import Project Data** REST API (`import` endpoint) is to save the import attributes to a `.json` file and point to that file in the cURL command. For the complete instructions on running the import using this command, return to Importing Project Data Using the REST API.

***Task*** ***To set up a cURL command that points to a file containing the import attributes, do the following:***

1. Create and save a `.json` file containing the import attributes. Refer to Available Import Options to Configure Import Behavior for details about the available import attributes.

   The following shows sample file contents:

   ```
   {
         "createEmptyInventory": false,
         "overwriteInventoryNotes": true,
         "addFilesToInventory": true,
         "inventoryFileMatchingCriteria": "COMPLETE_FILEPATH",
         "inventoryDirectoryDepth" : 2
         "markFilesAsReviewed": true,
         "reviewFileMatchingCriteria": "MD5_AND_COMPLETE_FILEPATH",
         "reviewDirectoryDepth": 2,
         "resetInventoryUsage": false,
   }
   ```

   Note that the value for the `inventoryFileMatchingCriteria` attribute (not shown) and the `reviewFileMatching` attribute must be enclosed in quotes.

   For purposes of example, this file is saved as `Import217Settings.json`, but you can provide any name with the `json` extension. Alternatively, you can save the json-formatted contents as a simple text file. However, when you point to the file in the cURL command, provide the file name only, not extension. For example, for `Import217Settings.txt`, provide only **`Import217Settings`** as the file name in the cURL command; do not include the `.txt` extension.

2. Set up the cURL command, pointing to the file containing the import attributes.

   The following shows the cURL command syntax:

   ```
   curl -H "Authorization:Bearer  JWT_TOKEN" -F importFile=@"FILE_TO_IMPORT.zip" -F
   projectImportModel=@"IMPORT_SETTINGS.json;type=application/json" http://HOSTNAME:PORT/
   codeinsight/api/projects/PROJECT_ID/import
   ```

   This is an example of the cURL command:

   ```
   curl -H "Authorization:Bearer
   eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsInVzZXJJZCI6MSwiaWF0IjoxNTY2ODU5NTg2fQ.qV2j8ZLgNGNJsT8
   OdPRwvE0-0y1x7w-0zr5h7Jz2d9uqY8tvACsV68posEUO9tD-YXlgXznX-IGnrnopDU7G3w" -F importFile=@"1184-
   export-02-20-2021_09-40.zip" -F projectImportModel=@"Import217Settings.json;type=application/
   json" http://localhost:8888/codeinsight/api/projects/217/import
   ```

   Note the following about the example command. Users must explicitly provide their own values in the command based on their environment.

   - `JWT_TOKEN` has been replaced with an example user authorization token.
   - `HOSTNAME:PORT` has been replaced with an example machine name and port—in this case, **`localhost:8888`**. (The IP address for the machine can be used instead of the machine name.)

- FILE_TO_IMPORT has been replaced with the name of an import data file (that is, the file name of the zip file to be imported) in the example. In this case, the import data file is **1184-export-02-20-2021_09-40**.zip. The entire file name must be enclosed in quotes.
- ImportSettings has been replaced with the name of the example file containing the import attributes. In this case, the file is **Import217Settings.json**. The entire file name and type must be enclosed in quotes.
- PROJECT_ID has been replaced with the ID of the project to which data is being imported (in this case, **217**). See Project ID for help on obtaining the project ID.

## Importing Project Data Using the Postman API Client

Instead of explicitly executing a cURL command that calls the **Import Project Data** REST API, you can invoke the API using an API client such as Postman. For instructions on running an import using Postman, see the following Knowledge Base article in the Revenera Software Community:

https://community.flexera.com/t5/FlexNet-Code-Insight-Customer/Using-Postman-to-Execute-a-Project-Data-Import-in-FlexNet-Code-Insight/ta-p/145532

# Verifying the Import Results

Use this procedure to verify that the import process completed as expected.

*Task* ***To verify that the import results are as expected, do the following:***

1. Open the target project in Code Insight and navigate to the **Project Inventory** page.
2. Confirm that the total number of inventory items includes the newly imported items. (Keep in mind that, by default, only inventory with matching associated files in the target codebase are imported.)
3. Confirm that the inventory items contain accurate inventory details and file path associations.
4. If the import results are not what you expect, adjust the import configuration (see Import Behavior and Configuration), and run the import again.

### Possible Benign Error When Processing Existing File Associations

During the import process, if a file association in the import data file already exists in the target inventory item, no new file association is added to the target inventory item. However, when the import processes this association, a “duplicate entry” exception similar to the following might be logged:

```
Duplicate entry ‘entry_id’ for key 'pse_inventory_group_files.UNIQ_FILE_GROUP
```

This exception is benign and has no impact on the regular file-processing behavior—that is, the existing file association is retained; no new file association is added.

5

# Automated Analysis

This chapter covers the following topics to describe the Automated Analysis of features in Code Insight:

- What is Automated Analysis?
- Supported Development Ecosystems
- Supported Archive Formats
- Additional Rule-based Detection Capabilities

## What is Automated Analysis?

Code Insight provides Automated Analysis capability to automatically inventory various package formats without the need for manual analysis. New automated detection rules are delivered to Code Insight as part of the Electronic Update process and through internal processes.

Automated Analysis is used in both scanning scenarios outlined below:

- Local scanning where the codebase is uploaded to the Scan Server or synchronized to the server from a Source Control Management system like Git or Perforce.
- Remote scanning, where a scan-agent plugin performs a scan remotely on built artifacts or source code on an Engineering build server and sends results back to Code Insight.

## Supported Development Ecosystems

Code Insight provides native support for operating in many development ecosystems (each encompassing a language, package type, and public registry). See the following topics for more information:

- Supported Ecosystems
- Notes About Ecosystem Support
- Notes about Dependencies Support

# Supported Ecosystems

The table below provides the following information about each ecosystem that Code Insight supports in the Automated Analysis process:

- **Language/File Type**—The code language or file type supported by the ecosystem.
- **Package**—The name of a package type in the ecosystem.
- **Registry**—The URL for the public registry or repository that hosts the package type.
- **Manifest File**—The file for which the Code Insight scan searches to locate a package of this type.
- **Top-level Inv.**—The indicator ✓ for “yes” or a dash (—) for “no”, showing whether the Code Insight scan supports the detection of third-party software in the package (displayed as top-level inventory).
- **Direct Dep., Trans. Dep.**—If top-level inventory is supported, the discovery of this component’s direct (first-level) dependencies and transitive dependencies (that is, dependencies of dependencies).
- **Notes**—Link to notes (if available) pertaining to Code Insight’s support of the specific ecosystem.

**Table 5-1 ▪** Supported Ecosystems

| Language/ File Type | Package | Registry | Manifest File | Top-level Inv. | Direct Dep. | Trans. Dep. | Notes |
|---|---|---|---|---|---|---|---|
| **BitBake, BitBake recipe** | Yocto | N/A | `.bb` | ✓ | N/A | N/A | See Yocto Ecosystems. |
| **C++, FORTRAN, Java, JavaScript, Lua, Python, R, Ruby, Scala** | Conda | https://anaconda.org/ | `index.json` | ✓ | ✓ | — | See Conda Ecosystems. |
| **DLL/EXE** | PE Header | N/A | `.dll, .exe` | ✓ | N/A | N/A | — |
| **Go** | glide | `https://go-search.org` | `glide.yaml` | ✓ | — | — | See Go Ecosystems. |
| | godep | | `godeps.json` | ✓ | — | — | |
| | govendor | | `vendor.json` | ✓ | — | — | |
| | module | | `go.mod`<br>`_go.mod` | ✓ | ✓ | — | |

**Table 5-1 ▪** Supported Ecosystems (cont.)

| Language/ File Type | Package | Registry | Manifest File | Top-level Inv. | Direct Dep. | Trans. Dep. | Notes |
|---|---|---|---|---|---|---|---|
| **Java** | Gradle | `http://search.maven.org/` | `build.gradle` | ✓ | ✓ | ✓ | — |
| | Maven | | `pom.xml` | ✓ | ✓ | ✓ | — |
| **JavaScript** | Bower | `https://registry.bower.io/packages/` | `bower.json` | ✓ | ✓ | — | — |
| | | | `.bower.json` | ✓ | ✓ | — | — |
| | | | `package.json` | ✓ | ✓ | — | — |
| **.NET** | NuGet | `https://api.nuget.org/v3-flatcontainer/` | `.csproj` | ✓ | ✓ | ✓ | See .NET Ecosystems. |
| | | | `.nupkg` | ✓ | ✓ | ✓ | |
| | | | `.nuspec` | ✓ | ✓ | ✓ | |
| **NodeJS** | NPM | `https://registry.npmjs.org/` | `package.json`<br>`package-lock.json`<br>`OR`<br>`npm-shrinkwrap.json` | ✓ | ✓ | ✓ | See NPM Ecosystems. |
| | Yarn | `https://registry.npmjs.org/` | `package.json`<br>`yarn.lock` | ✓ | ✓ | — | See Yarn Ecosystems. |
| **PHP** | Composer | `https://packagist.org/` | `composer.json` | ✓ | ✓ | — | — |
| | | | `composer.lock` | ✓ | ✓ | — | — |
| **Python** | PyPI | `https://pypi.org/` | `PKG-INFO` | ✓ | — | — | See PyPI Ecosystems. |
| | | | `requirements.txt` | N/A | ✓ | — | |
| | | | `setup.py` | ✓ | ✓ | — | |
| | | | `.whl` | ✓ | ✓ | — | |
| **RPM** | RPM Header | N/A | `.rpm` | ✓ | N/A | N/A | — |

**Table 5-1 ▪** Supported Ecosystems (cont.)

| Language/ File Type | Package | Registry | Manifest File | Top-level Inv. | Direct Dep. | Trans. Dep. | Notes |
|---|---|---|---|---|---|---|---|
| **Ruby** | Gem | `https://rubygems.org/api/v1/` | `.gem` | ✓ | ✓ | — | See Ruby Ecosystems. |
| | | | `Gemfile` | ✓ | ✓ | — | |
| | | | `.gemspec` | ✓ | ✓ | — | |
| **Swift, Obj-C** | CocoaPods | N/A | `Podfile.lock` | ✓ | — | — | — |
| | | | `.podspec` | ✓ | ✓ | — | — |
| **Various** | Git Repo | `https://github.com` | `config` | ✓ | — | — | See Git Ecosystems. |

# Notes About Ecosystem Support

The following sections provide additional information (such as limitations, requirements, and clarifications) to consider for the various ecosystems supported in the Code Insight Automated Analysis process:

- Conda Ecosystems
- Git Ecosystems
- Go Ecosystems
- .NET Ecosystems
- NPM Ecosystems
- PyPI Ecosystems
- Ruby Ecosystems
- Yarn Ecosystems
- Yocto Ecosystems

## Conda Ecosystems

First level dependencies are supported for `index.json`, but the semver resolution of version is not yet supported.

## Git Ecosystems

Code Insight scans the configuration file (`config` or `gitconfig`) inside a `.git` folder in a project codebase to identify OSS and third-party components and evidence and then uses this information to create inventory items.

***Note ▪*** *To support the detection of components in a Git repository, the configuration file in a* `.git` *folder will always be included in scans even if this folder has been added to the* ***Scan Exclusions*** *list in the scan profile.*

## Go Ecosystems

Note the following for Go ecosystems:

- A golang project configured with a supported package manager must include a license file to enable Code Insight to discover it as top-level inventory.
- Currently, Code Insight supports the discovery of top-level inventory only in scans of pre-build Artifact source code.
- If the codebase is uploaded from the release section of the VCS repository, Code Insight must use the version in the name of the project's parent folder as the version in the top-level inventory name. Any changes to the version in the parent folder name can result in the wrong version being reported in the inventory.

### More About Direct Dependencies in Go Mod Pre-Built Artifacts

Currently Code Insight supports the detection of direct dependencies for Go modules (in `go.mod` and `_go.mod` files) from only the Github forge, provided these dependencies are not *indirect* dependencies.

## .NET Ecosystems

When .NET projects are created using .NET Core, Code Insight is not able to report inventory in `.csproj` files due to missing information about top-level items. Additionally, with no information about top-level items, no dependencies are reported.

## NPM Ecosystems

Note the following for NPM ecosystems:

- Code Insight provides scan support for `package.json` alone or for `package.json` with either `package-lock.json` or `npm-shrinkwrap.json`.
- The `package-lock.json` or `npm-shrinkwrap.json` file is scanned only if it co-exists with `package.json`. (The `package.json` file contains the component and dependency data. The `package-lock.json` or `npm-shrinkwrap.json` file is used to identify the exact dependency versions for components that Code Insight should pull from `package.json`.)
- If both `package-lock.json` or `npm-shrinkwrap.json` are present with `package.json`, Code Insight scans `npm-shrinkwrap` (along with `package.json`) and ignores `package-lock.json`.

## PyPI Ecosystems

Code Insight supports the discovery of top-level inventory and direct dependencies for both pre-build and post-build artifacts of a Python project. Pre-build artifacts include source packages, such as `tar.gz`, `.zip`, and other such files. Post-build artifacts are binary packages such as `.whl` files.

### More About Direct Dependencies in Pre-Built Artifacts

Direct dependencies for the pre-build artifacts are retrieved from the `requirements.txt` file if it exists. (In the absence of `requirements.txt`, direct dependencies are reported from the `install_requires` section in the `setup.py` file.)

When direct dependencies are retrieved from `requirements.txt`, the top-level inventory item to which these dependencies are mapped is determined as follows:

- If `PKG-INFO` or `setup.py` resides in the same directory as `requirements.txt`, the top-level inventory item is determined by information in either `PKG-INFO` or `setup.py`.
- If `PKG-INFO` or `setup.py` does not reside in the same directory as `requirements.txt`, the top-level inventory item is determined in one of two ways:
    - If the Code Insight obtains the codebase through a `git sync` or `git clone` operation, the top-level inventory item to which direct dependencies are mapped is created from the configuration information found in the `.git` file.
    - If the codebase has been directly downloaded from a GitHub or PYPI repository and then uploaded to Code Insight for the scan, the top-level inventory item is created using the name of the directory under which `requirements.txt` resides. Direct dependencies identified in `requirements.txt` are then mapped to this inventory item.

      Note that, upon creation, such an inventory item is considered a “place holder” item because it is created from a directory name, which might or might not be a valid component name. The item is published during the automated analysis only if its name matches a valid component in the Code Insight data library, its forge is PyPI or GitHub, and it meets your site’s inventory publication policies. Otherwise, the item remains unpublished for further review.

      The inventory type for the item is determined as follows:

        - If the component name matches a component name in the Code Insight data library, the inventory type is **Component**.
        - If the component is not found in the data library but the inventory’s license matches a license in the data library, the inventory type is **License Only**.
        - If neither the component nor license has a match in the data library, the inventory type is **Work In Progress**.

## Ruby Ecosystems

Note the following for Ruby ecosystems:

- For RubyGem projects, Code Insight shows all platform-related dependencies and those dependencies that are not part of a “test” or “dev” group as inventory. Any gems identified as “dev” or “test” are not considered for inventory.
- Only SemVer expressions in the *major.minor.patch* format are supported to resolve dependencies listed in the manifest file.

## Yarn Ecosystems

Note the following for Yarn ecosystems:

- Code Insight provides scan support for `package.json` alone or for `package.json` with the `yarn.lock` file.
- The scan `yarn.lock` file is scanned only if it co-exists with `package.json`. (The `package.json` file contains the component and dependency data. The `yarn.lock` file is used to identify the exact dependency versions for components that Code Insight should pull from `package.json`.)

### Yocto Ecosystems

Code Insight parses a `.bb` file only if it contains an `SRC_URI` property value that starts with **`git://`** or **`https://`**. If the `SRC_URI` property contains more than one URI, only the first supported URI is considered.

# Notes about Dependencies Support

Code Insight supports scanning for top-level inventory items, direct dependencies, and transitive dependencies. The scan profile, managed by the Code Insight System Administrator, is used to configure of the desired depth of scan with respect to dependencies. See About Scan Profiles for information about scan profiles.

Note the following additional information about dependency scanning:

- Dependencies represent open-source packages that are referenced by the scanned codebase, but not necessarily present in the codebase.
- Dependency scanning is designed to be used when scanning pre-build artifacts, typically found in source-code bundles. Since this scenario relies on package-management configuration files, it is not 100% precise in the resolution of the declared dependencies. In many cases, dependencies will be resolved to the latest available version within the declared range. However, this version can differ from the actual package version pulled down as part of the build.
- Dependency scanning is not designed for scanning post-build artifacts when using the scan-agent plugins to scan on the build servers as part of the build process. In such scenarios, all dependencies have already been resolved by the build system and are present in the scanned codebase.

# Supported Archive Formats

Note the following about the archives supported by Automated Analysis:

- Automated Analysis uses 7-Zip to read archive files. Use the following link to view the archive formats currently supported by 7-Zip:

  https://sevenzip.osdn.jp/chm/general/formats.htm

- Automated Analysis discovers inventories and evidences associated with `.apk` archives that are uploaded as part of the codebase (but are not part of an ecosystem, as listed in the Supported Ecosystems table). Note that Automated Analysis does not fetch any dependency information from these archives. The names of inventories generated from `.apk` archives are suffixed with “found inside <archive_name>.apk”.

# Additional Rule-based Detection Capabilities

Automated detection used in the Automated Analysis process can also generate findings based on other rule-based techniques that include the following:

- Search term analysis
- File name analysis
- CDN analysis

6

# Performing a Remote Scan

This section discusses Code Insight remote scanning. The following topics are covered in this section:

- About Remote Scans
- Creating a Project Without Uploading a Codebase
- Code Insight Plugins
- Important: Plugin Upgrades in Code Insight

## About Remote Scans

Code Insight has the ability to scan files on a remote system and manage the inventory items created from this remote location. This remote scan allows you to integrate automatic package-level scanning into your build process using a Code Insight scan-agent plugin. This integration includes automated package discovery (see Automated Analysis) and targeted components.

Note the following:

- For files scanned by a Code Insight scan-agent plugin on a remote system, currently only license evidence found in these files is currently reported in Code Insight.
- Code Insight does not generate email notifications for remote scan events.

## Creating a Project Without Uploading a Codebase

Some organizations might be interested in reviewing the inventory that results from a scan of their product's post-build artifacts on the build server. Other organizations might want to review the inventory resulting from a codebase scan but are reluctant to upload their product codebase (or synchronize a Source Control Management repository) to Code Insight. Instead, they want to keep their codebase in its existing development system due to security, consistency, or other concerns.

To address these requirements, Code Insight provides scan-agent plugins that scan codebase files or built artifacts wherever they reside and send the results as inventory to the Code Insight Core Server for review and remediation by users. This process requires a Code Insight project on the Core Server for handling the returned results, but requires no codebase upload or synchronization to Code Insight.

Organizations might still want to upload a their product codebase to Code Insight to perform a server scan, but then use a scan plugin to remotely scan post-build artifacts directly on the build server. They can use the same Code Insight project to handle the results of both scans, enabling them to compare the resulting inventories, resolve discrepancies, and determine a final inventory list.

## Overview of How to Set Up for Remote Scanning

The following is an overview of setting up for remote scanning:

**Phase 1**—Create a project in Code Insight. See About Code Insight Projects.

**Phase 2**—Create a valid JSON Web Token (JWT) for the user whose account will be used to connect to Code Insight. For instructions on generating the JWT, see Managing Authorization Tokens in the “Using Code Insight” chapter.

**Phase 3**—Install and configure the appropriate scan-agent plugin. (For information how to install and configure the plugin, see the *Code Insight Plugins Guide*.) As part of the configuration process, you will need to provide the name of the project that you created, the URL of the Code Insight core server, and the JWT.

When the scan-agent plugin is invoked (for example, during a build in Jenkins), the remote codebase will be scanned and any identified inventory items will be created in the existing project on the Code Insight server for further review and remediation.

# Code Insight Plugins

Code Insight offers the following scan-agent plugins for remote scanning.(Refer to the *Code Insight Plugins Guide* for a list of requirements for each scan-agent plugin.)

**Table 6-1 ▪** Overview of the Standard Plugins

| Build Environment | Code Insight Plugin | Performs automated scanning of... |
|---|---|---|
| **IDEs** | Eclipse | An Eclipse workspace in the Eclipse IDE environment. |
| | Visual Studio | A Visual Studio solution. |

**Table 6-1 ▪** Overview of the Standard Plugins (cont.)

| Build Environment | Code Insight Plugin | Performs automated scanning of... |
|---|---|---|
| **CI Tools** | Azure DevOps | An Azure DevOps workspace as part of the build process. |
| | Bamboo | A Bamboo workspace as part of the build process (on Local Agents only) |
| | GitLab | GitLab projects as part of the build process. |
| | Jenkins | A Jenkins workspace as part of the build process.<br>A separate plugin is available (called the Scan Schedule Plugin) that enables you to simply schedule the scan of a codebase residing on the Code Insight Scan Server via the Jenkins scheduler. |
| | TeamCity | TeamCity projects as part of the build process. |
| **Package Manager and Build Tools** | Ant | Apache Ant as part of the build process. |
| | Gradle | Gradle projects as part of the build process. |
| | Maven | Maven projects as part of the build process. |
| **Binary Repositories** | JFrog Artifactory | Artifactory repositories to identify non-compliant artifacts. |
| **Container Platforms** | Docker Images | Docker images on a Docker server. |

Additionally, a generic scan-agent plugin is available with Code Insight that enables you to scan arbitrary file systems of your choice. It also easily integrates with certain Engineering systems, such as TeamCity and GitLab, to perform scans as part of a build process and can serve as an example for developing your own scan-agent plugin (as described in the *Code Insight Plugins Guide*).

# Important: Plugin Upgrades in Code Insight

In Code Insight 2020 R2 and earlier, scan-agent plugins required an inventory-only project on Code Insight to which to send the results of the remote scans. Starting in Code Insight 2020 R3, the scan-agent plugins were upgraded to send the results of remote scans to a project type (introduced in 2020 R3) that can manage results from both remote scans and scans performed by the Scan Server.

Currently, Code Insight continues to support existing inventory-only projects, enabling users to scan these projects using only the older plugins installed from previous Code Insight releases. However, inventory-only projects will be deprecated in a future release. For more information about the plugins upgrade, see the *Code Insight Plugins Guide*. Also see Legacy Projects.

7

# Configuring Source Code Management

Code Insight provides a connector that allows you to use Source Code Management systems (SCM) as a source for codebase data. This section discusses the following topic:

- Managing Source Code Management (SCM) Instances
- Configuring a Git SCM Instance
- Configuring a Perforce SCM Instance
- Configuring a Subversion SCM Instance
- Configuring a TFS SCM Instance

## Managing Source Code Management (SCM) Instances

Code Insight provides the ability to scan data obtained from synchronization with a remote data source. The following sections provide information on adding and managing SCM instances.

- Prerequisites
- Adding an SCM Instance to the Code Insight Project
- Testing an SCM Instance
- Synchronizing an SCM Instance
- Deleting an SCM Instance

# Prerequisites

Before performing the procedures in this section, ensure that an SCM command-line client is properly installed on the Code Insight Scan Server and that connectivity between the SCM client and the SCM server is properly configured. Refer to the “Integrating with Source Code Management” chapter in the *Code Insight Installation and Configuration Guide* for details.

If Code Insight is running as a service, make sure that the user context under which the service runs has appropriate permissions to run the SCM client.

# Adding an SCM Instance to the Code Insight Project

You can specify configuration information about your remote data source when you edit your Code Insight project.

***Task*** ***To add an SCM instance, do the following, do the following:***

1. Navigate to the **Summary** tab for the project to which you are synchronizing codebase files.
2. Open the **Manage Project** menu and select **Edit Project**. The **Edit Project** page opens.
3. Select the **Version Control Settings** tab.
4. Select the desired connector (remote data source) from the **Application** dropdown menu.
5. Click **Add Instance**. The available fields for the selected application will appear on a new **Instance** tab. See the inline help for explanations of the fields on this tab.
6. After editing the fields for your specific instance, click **Save**. You should now test and synchronize the instance.

# Testing an SCM Instance

If you add an instance or edit any of the fields associated with your SCM instance, you should test the connection to ensure the repository is responsive.

***Task*** ***To test your connection, do the following:***

1. If you are not already on the **Version Control Settings** tab on the **Edit Project** page, navigate to it.
2. Select the **Instance** tab for the connection you want to test.
3. Click **Test Connection** to confirm that the repository is reachable. After a moment, Code Insight displays a success message dialog if the connection is successful. If the connection is not successful, ensure that your entries on the **Instance** tab are correct and click **Test Connection** again.

## Synchronizing an SCM Instance

After testing your SCM connection, you can synchronize the instance to get the codebase files from the selected repository.

Note that the scan process also automatically synchronizes the data before initiating the actual scan to ensure that the latest codebase is in place.

*Task* ***To sync an SCM instance, do the following:***

1. If you are not already on the **Version Control Settings** tab on the **Edit Project** page, navigate to it.
2. Click **Sync Now** to synchronize the codebase repositories from all SCM instances to your project's directory (identified by the project ID) on the Code Insight Scan Server. (Project directories are located under the scan root directory on the Scan Server.) Your project's directory will contain subdirectories, with names such as `git.0` or `git.1`, for each SCM instance created for your project.

Note the following:

- If multiple instances have been added, clicking **Sync Now** will synchronize all instances to your project. If the sync fails for one instance, the overall sync fails.
- If the Scan Server assigned to the project to which you are synchronizing codebase files is disabled, the **Sync Now** button is also disabled. Consider reassigning the project to an enabled Scan Server. (If necessary, see your Code Insight System Administrator for information about which servers are enabled).

## Deleting an SCM Instance

This section shows you how to delete an SCM instance if it is no longer needed.

*Task* ***To delete an SCM instance, do the following:***

1. If you are not already on the **Version Control Settings** tab, navigate to it.
2. Select the **Instance** tab for the instance you want to delete.
3. Click **Delete Instance**. The selected instance is deleted from the system.

# Configuring a Git SCM Instance

Code Insight provides an SCM connector that enables you to scan a codebase hosted on a Git server. To perform the scan, you must first configure a Git SCM instance for the Code Insight project. Refer to the following topics for more information:

- Adding a Git SCM Instance to the Code Insight Project
- Fields Used to Configure a Git SCM Instance

# Adding a Git SCM Instance to the Code Insight Project

The following procedure describes how to add a Git SCM instance to the Code Insight project.

*Task* ***To configure a Git SCM instance, do the following:***

1. Use the instructions in Adding an SCM Instance to the Code Insight Project to navigate to the **Version Control Settings** tab and add a Git SCM instance, selecting **Git** from the **Application** dropdown.
2. See Fields Used to Configure a Git SCM Instance for a description of the settings used to define a Git SCM instance, or use the inline help provided for each setting on the tab.
3. Once you save the Git SCM instance, test the Code Insight connection to the instance, as described in Testing an SCM Instance.

# Fields Used to Configure a Git SCM Instance

The following settings are used to configure a Git SCM instance.

**Table 7-1 ▪** Setting Used to Configure a Git SCM Instance

| Git SCM Instance Setting | Description |
|---|---|
| **Git Repository URL** | Provide the repository URL in either format:<br>● `http(s)://<host.xz>/<path>/to/repo.git`<br>● `<user>@<host>:<path>/repo.git`<br>The contents of the repository will be cloned to the following directory on the Scan Server, based on the specified branch, tag, or commit ID:<br>`<scanroot>/<projectID>/<instanceID>` |
| **Git Username** | Provide the user name for Authenticated access to the repository.<br>Leave this field blank for “anonymous” or SSH access (the system automatically looks for an SSH keypair on the server). See the *Code Insight Installation and Configuration Guide* for instructions on configuring Git over SSH. |
| **Git Password** | Enter the password associated with the user name provided. |
| **Git Branch, Git Tag, or Git Commit ID** | Specify either the Git branch, tag, or commit ID to identify the source code version to which to synchronize.<br>Alternatively, leave these fields blank to synchronize to the mainline branch. |

# Configuring a Perforce SCM Instance

Code Insight provides an SCM connector that enables you to scan a repository (codebase) hosted on a Perforce server. To perform the scan, you must first configure a Perforce SCM instance to identify this repository to the Code Insight project. Refer to the following topics for more information:

- Adding a Perforce SCM Instance to the Code Insight Project
- Fields Used to Configure a Perforce SCM Instance

## Adding a Perforce SCM Instance to the Code Insight Project

The following procedure describes how to add a Perforce SCM instance to the Code Insight project.

*Task* ***To configure a Perforce SCM instance, do the following:***

1. Use the instructions in Adding an SCM Instance to the Code Insight Project to navigate to the **Version Control Settings** tab and add a Perforce SCM instance, selecting **Perforce** from the **Application** dropdown.
2. See Fields Used to Configure a Perforce SCM Instance for a description of the settings used to define a Perforce SCM instance, or use the inline help provided for each setting on the tab.
3. Once you save the Perforce SCM instance, test the Code Insight connection to the instance, as described in Testing an SCM Instance.

## Fields Used to Configure a Perforce SCM Instance

The following settings are used to configure a Perforce SCM instance on Code Insight.

Keep the following in mind as you set up the instance, especially when providing the **Username** and **Password** credentials:

- The repository identified by the instance must reside on a Perforce server that is configured with Security Level 1, 2, or 3. The Code Insight Perforce connector does not support instances created for a Perforce server configured with Security Level 0, in which users are created without passwords.
- The Code Insight Perforce connector supports LDAP authentication on Perforce. If Perforce is configured with LDAP, you must provide the appropriate LDAP credentials for the **Username** and **Password** fields to access the Perforce repository identified by the instance.

**Table 7-2 ▪** Setting Used to Configure a Perforce SCM Instance

| Perforce SCM Instance Setting | Description |
|---|---|
| **URL (P4PORT)** | Provide the URL of the Perforce instance with which to synchronize. Note the following example URL formats:<br>**For a TCP connection**<br>`tcp:<p4ServerHostID>:<p4Port>`<br>**For an SSL connection**<br>`ssl:<p4ServerHostID>:<p4Port>`<br>`p4ServerHostID` and `p4Port` identify the hostID (hostname or IP address) and port of the Perforce server. |
| **Username (P4USER)** | Provide the user name that has access to the Perforce depot to which this instance is synchronizing. If Perforce is configured with LDAP authentication, provide the LDAP user name. |
| **Password (P4PASSWD)** | Provide the password associated with the user name (see the previous field). If Perforce is configured with LDAP authentication, provide the LDAP password associated with the LDAP user name.<br>If you are using a P4 ticket provided by the Perforce administrator, this field is optional. |
| **Branch Spec (P4CLIENT)** | Provide the path (in the following format) to the Perforce branch to which this instance is synchronizing:<br>`{depot}/{projectPath}`<br>The synchronization process will create a workspace for this branch on the Scan Server at the following location:<br>`{scanRoot}/{projectID}/{instanceID}`<br>Once the synchronization process copies the branch contents to this workspace, the workspace is scanned. |
| **Changelist No** | (Optional) Provide a changelist number only if this instance is synchronizing to a particular changelist. Otherwise, this value defaults to the latest revision. |
| **Label** | (Optional) Provide a label for the perforce branch. |

# Configuring a Subversion SCM Instance

Code Insight provides an SCM connector that enables you to scan a codebase hosted on a Subversion instance. To perform the scan, you must first configure a Subversion SCM instance for the Code Insight project. Refer to the following topics for more information:

- Adding a Subversion Instance to the Code Insight Project
- Fields Used to Configure the Subversion Instance

## Adding a Subversion Instance to the Code Insight Project

The following procedure describes how to add a Subversion SCM instance to the Code Insight project.

***Task*** ***To configure a Subversion SCM instance, do the following:***

1. Use the instructions in Adding an SCM Instance to the Code Insight Project to navigate to the **Version Control Settings** tab and add a Subversion SCM instance, selecting **Subversion** from the **Application** dropdown.
2. See Fields Used to Configure the Subversion Instance for a description of the settings used to define a Subversion SCM instance, or use the inline help provided for each setting on the tab.
3. Once you save the Subversion SCM instance, test the Code Insight connection to the instance, as described in Testing an SCM Instance.

## Fields Used to Configure the Subversion Instance

The following settings are used to configure a Subversion SCM instance for the Code Insight project.

**Table 7-3 ▪** Settings Used to Configure a Subversion SCM Instance

| Subversion SCM Instance Setting | Description |
|---|---|
| **Subversion URL** | Enter the URL of the Subversion repository containing the revision that you want to synchronize to your project. Use the following format:<br><protocol>://<host>/<subversionRoot>/<repository> |
| **Username** | Provide the user name needed to access the repository. Leave this field blank to make an anonymous connection. |
| **Password** | Provide the password needed to access the repository. Leave this field blank to make an anonymous connection. |
| **Revision** | (Optional) Enter the Subversion revision that you want to synchronize to your project. Leave this field blank to default to the latest revision. |

# Configuring a TFS SCM Instance

Code Insight provides an SCM connector that enables you to scan a codebase hosted on a Team Foundation Server (TFS) instance. To perform the scan, you must first configure a TFS SCM instance for the Code Insight project. Refer to the following topics for more information:

- Adding a TFS SCM Instance to the Code Insight Project
- Fields Used to Configure a TFS SCM Instance

## Adding a TFS SCM Instance to the Code Insight Project

The following procedure describes how to add a TFS SCM instance to the Code Insight project.

*Task* ***To configure a TFS SCM instance, do the following:***

1. Use the instructions in Adding an SCM Instance to the Code Insight Project to navigate to the **Version Control Settings** tab and add a TFS SCM instance, selecting **TFS** from the **Application** dropdown.
2. See Fields Used to Configure a TFS SCM Instance for a description of the settings used to define a TFS SCM instance, or use the inline help provided for each setting on the tab.
3. Once you save the TFS SCM instance, test the Code Insight connection to the instance, as described in Testing an SCM Instance.

## Fields Used to Configure a TFS SCM Instance

The following settings are used to configure a TFS SCM instance for the Code Insight project.

**Table 7-4 ▪** Settings Used to Configure a TFS SCM Instance

| Perforce SCM Instance Setting | Description |
|---|---|
| **TFS URL** | Provide the URL of the TFS with which to synchronize. Note the following example URL formats.<br>For the latest version of TFS:<br>`<protocol>:<tfs_host>:<port>/<collection>/<project>`<br>For earlier versions of TFS:<br>`<protocol>:<tfs_host>:<port>/<collection>/<tfsroot>/<project>` |

**Table 7-4 ▪** Settings Used to Configure a TFS SCM Instance (cont.)

| Perforce SCM Instance Setting | Description |
|---|---|
| **Username** | Provide the user name that has access to the TFS collection to which this instance is synchronizing.<br>If you are synchronizing with a VSTS project in TFS, enter the user name from the alternate authentication credentials enabled in VSTS. For details about enabling alternate credentials, refer to “Special Requirement for a VSTS Project in TFS” in the “Integrating with Source Code Management” chapter in the *Code Insight Installation and Configuration Guide*. |
| **Password** | Provide the password associated with the user name provided.<br>If you are synchronizing with a VSTS project in TFS, enter the password from the alternate authentication credentials enabled in VSTS. |
| **Changeset** | (Optional) Provide a changeset number to which the TFS SCM instance is synchronizing. Otherwise, this value defaults to the latest revision.<br>If a changeset and label are both specified (see the **Label** description next), the label is ignored, and the instance synchronizes to the changeset. |
| **Label** | (Optional) Provide a specific label to which the TFS SCM instance is synchronizing.<br>If a label and changeset (see the previous **Changeset** description) are both specified, the label is ignored, and the instance synchronizes to the changeset instead. |

# 8

# Pages and Panels

This chapter contains reference information for the following elements in the Code Insight Web UI:

- Add Project Dialog
- Add Token Dialog
- Add User Dialog
- Advanced File Search Add Dialog
- Advanced File Search Dialog
- Advanced Inventory Search Dialog
- ALM Tab
- Analysis Workbench
- Branch Project: Project Copy Settings
- Branch Project: Project Information
- Branch Project: Summary
- Branch Project: Upload Codebase
- Branch Project: Version Control Settings
- Branch Project Wizard
- Code Insight Dashboard
- Component Details Window
- Create/Edit Scan Profile Dialog
- Custom Detection Rule Dialog
- Custom Detection Rules Tab
- Edit (Default) Project Users Page

- Edit Custom Rule Dialog
- Edit Project: General Tab
- Edit Project: Project Hierarchy Tab
- Edit Project: Review and Remediation Settings Tab
- Edit Project: Scan Settings Tab
- Edit Token Dialog
- Edit User Dialog
- Electronic Updates Tab
- Email Server Tab
- Evidence Details Tab in the Analysis Workbench
- File Search Results Pane
- Import Project Data Dialog
- Inventory Details Tab in the Analysis Workbench
- Inventory History Window
- Inventory View
- LDAP Tab
- License Details Window
- Lookup Component Window
- Policy Details Window
- Policy Page
- Preferences Page
- Project Defaults Tab
- Project Inventory Details Pane
- Project Inventory Tab
- Projects Pane and Associated Dashboard
- Reports Tab
- Scan History Dialog
- Scan Profiles Tab
- Scan Server Dialog
- Scan Servers Tab
- Select a New Project Contact Page
- Summary Tab
- Suppress Vulnerability Window

- Suppressed Versions of <component> for <vulnerability> Window
- Suppressed Vulnerabilities Tab
- System Settings Tab
- Unsuppress Vulnerability Window
- Users/Permissions Tab

# Add Project Dialog

The **Add Project** dialog displayed when you select to create a project from the **Project** pane (see Creating a Project). From this dialog, you define the basic properties for the new project using the following fields:

**Table 8-1 ▪** Add Project Dialog

| Column/Field | Description |
|---|---|
| **Name** | Enter a name for the new project. |
| **Project Visibility** | Select the default for visibility status—**Public** or **Private**—for the project.<br>Any user in the system read-only access to a public project. To what degree a user can interact with the project depends on whether the user has a project role and what the role is—Project Administrator, Analyst, or Reviewer.<br>However, private projects are hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. Additionally, project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see a given private project. |
| **Scan Server** | Select the local Scan Server for this project. Once this project is scanned, the Scan Server cannot be changed. (Select a Scan Server even if you are using a remote scan agent plugin to scan the project.) |

# Add Token Dialog

The **Add Token** dialog appears when you click the **Add Token** button on the **Preferences** page. It lets you create an authorization token (that is, a JSON Web Token known as a JWT) to be used to authenticate calls to Code Insight REST APIs. The token is associated with the user account under which you are logged in or whose password you have changed in the **Change Password** fields on the **Preferences** page. The dialog has the following fields:

**Table 8-2 ▪** Add Token Dialog

| Column/Field | Description |
|---|---|
| **Name** | Enter a name for the token you are creating. |

**Table 8-2 ▪** Add Token Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Token Validity** | Select one of the validity periods:<br>● **Never Expires**: The authorization token never expires.<br>● **Expires On**: The authorization token is valid until the date you pick on the Validity Calendar. |
| **Validity Calendar** | If you check the **Expires On** option, the validity calendar becomes active. Type an expiration date (for example, 10/10/10), or click the calendar icon and select a date. |
| **Save** | Click this button to save the token. |
| **Cancel** | Click this button to exit the **Add Token** dialog without saving the token. |

**See Also**
Preferences Page
Edit Token Dialog

# Add User Dialog

The **Add User** dialog on the **Administration** page allows you to add new users to the Code Insight system. The dialog contains the following columns and fields:

**Table 8-3 ▪** Add User Dialog

| Column/Field | Description |
|---|---|
| **Login** | Enter the login of the new user. |
| **First Name** | Enter the first name of new user. |
| **Last Name** | Enter the last name of new user. |
| **Email** | Displays the email address of the user associated with the login. To change the user’s email address, type over the existing email address. |
| **Password** | The current password is not displayed in this field. A user with Code Insight System Administrator permissions can type a new password in this field. |
| **Password Confirm** | The current password is not displayed in this field. A user with Code Insight System Administrator permissions can type a new password in this field. |
| **Question** | The prompt that the user must answer to retrieve a forgotten password. |
| **Answer** | The answer to the question in the previous field. |

**Table 8-3 ▪** Add User Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Submit** | Select **Submit** to have the system save your user edits. A prompt appears to notify you that your edits have been saved. |
| **Cancel** | Select Cancel to return to the Administration Users tab without saving your changes. |

**See Also**
Users/Permissions Tab

# Advanced File Search Add Dialog

The **Advanced File Search Add** dialog allow you to select a standard search, create your own search, or delete an existing search. The dialog has the following fields:

**Table 8-4 ▪** Advanced File Search Add Dialog

| Column/Field | Description | |
|---|---|---|
| **Name** | The name of the search. For example, *Files not in inventory*. | |
| **Description** | A short description of the search. For example, *Files not associated with inventory items*. | |
| **Criteria** | Use these fields to build the new search. | |
| | **Add Criteria** | Click the dropdown menu and select search criteria. To add more criteria, click **Add Criteria** and select another item from the dropdown menu. When you select search criteria from the dropdown menu, a boolean operator appears in the center dropdown, and a new dropdown appears from which you must select a criteria value to search the selected field for.<br>***Note ▪*** *Files scanned by a remote scan agent can be searched by only the following criterion:* ***File Size****,* ***File Path****,* ***File Digest****,* ***Review Status****,* ***Inventory Status, Evidence status****,* ***Has license matches****,* ***Does not have license matches****, and* ***License****.* |
| | **Add Criteria Group** | Click to add a group of criteria. |
| **Save** | Click to save the new search. | |
| **Save and Search** | Click to execute the new search without saving the search for future use. | |

**Table 8-4 ▪** Advanced File Search Add Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Search without saving** | Click to execute the new search without saving the search for future use. |
| **Cancel** | Click to close the **Search Files** dialog without searching. |

# Advanced File Search Dialog

The **Advanced File Search** dialog allows you to select a standard search, create your own search, or delete an existing search. The dialog has the following fields:

**Table 8-5 ▪** Advanced File Search Dialog

| Column/Field | Description |
|---|---|
| **Add New** | Click this button to access the **Advanced File Search Add** dialog. |
| **Name** | The name of the search. For example, *Files not in inventory*. |
| **Description** | A short description of the search. For example, *Files not associated with inventory items*. |
| ✕ | Click to delete a search. |
| **Search** | Click to execute the selected search. |
| **Close** | Click to close the Search Files dialog without searching. |

# Advanced Inventory Search Dialog

The **Advanced Inventory Search** dialog is opened when you click the **Advanced Search** button on the **Project Inventory** tab or the **Inventory** view. This dialog provides the following options that enable you to search project inventory in a variety of ways:

**Table 8-6 ▪** Advanced Inventory Search Dialog

| | Column/Field | Description |
|---|---|---|
| **Inventory Items** | The following options enable you to filter inventory by inventory attributes. | |
| | **Inventory Name** | Enter the whole or partial inventory name by which to filter the inventory display. For example, if you enter apache in this field, Code Insight will find all inventory items that have the *apache* string in their names. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| Column/Field | Description |
|---|---|
| **Inventory Priority** | Select one or more of checkboxes (**P1**, **P2**, **P3**, or **P4**) to search the inventory by inventory priority.<br>For more information about inventory priority, see Inventory Priority in the “Using Code Insight” chapter. |
| **Inventory Review Status** | Select one or more of the following checkboxes to filter the inventory display based on the review status of inventory items:<br>● **Approved**—Show only inventory that has been reviewed and approved, either manually by a reviewer or automatically during the auto-publish process.<br>● **Rejected**—Show only inventory that has been reviewed and rejected, either manually by a reviewer or automatically during the auto-publish process.<br>● **Not Reviewed**—Show only inventory that has not yet been reviewed.<br>For more information about the review status, see Review Status of Inventory in the “Using Code Insight” chapter. |
| **Dependency Options** | Select one of the following options to filter the inventory display based on dependency level:<br>● **All Inventory Items**—Show all inventory—that is, all top-level inventory items, along with their first-level and transitive dependencies.<br>● **Only Top-Level Inventory Items**—Show all top-level inventory items only. No first-level or transitive dependencies are displayed.<br>● **Only Dependency Inventory Items**—Show only first-level and transitive dependencies. No top-level inventory is displayed. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| | Column/Field | Description |
|---|---|---|
| | **Inventory Age** | Select one of the following to filter the inventory display by the time frame in which the inventory items were published:<br>● **Last 1 day**—Show inventory published in the last day. For example, if today is Feb 6th, search from Feb 5th 12 AM.<br>● **Last 7 days**—Show inventory published in the last week. For example, if today is Feb 6th, search from Jan 30th 12 AM.<br>● **Last month**—Show inventory published in the last month. For example, if today is Feb 6th, search from Jan 7th 12 AM (30 days).<br>● **Custom Date Range**—Show inventory published within the specified time frame. Select a beginning (**From**) and ending (**To**) date from the popup calendar.<br>● **Any**—Show all published inventory. |
| | **Inventory Notifications** | Select one or more of the following checkboxes to filter the inventory display based on security vulnerability alerts:<br>● **Inventory with Open Alerts**—Show only inventory items that have open vulnerability alerts (that is, alerts for vulnerabilities that were discovered post-publication and have not been closed).<br>● **Inventory Rejected Due to New Non-Compliant Security Vulnerabilities**—Show inventory items that have been rejected due to new security alerts that are non-compliant with policy. |
| | **Inventory Confidence Level** | Select one or more Confidence levels—**High**, **Medium**, or **Low**—by which to filter system-generated inventory items in the inventory display.<br>The Confidence level is the measure of the strength of the discovery technique used by Code Insight to generate an inventory item. For a description of the Confidence levels and how they are used, see Inventory Confidence in the "Using Code Insight" chapter. |
| **Inventory Tasks** | The following options filter inventory to show only those inventory items that have tasks. Refine the search using one or more task attributes—for example, task status, type, age, or owner. | |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| Column/Field | Description |
|---|---|
| **Task Status** | Select one of the following to filter the inventory display by the current status of the tasks associated with inventory:<br>● **Open Tasks**—Show inventory associated with at least open task.<br>● **Closed Tasks**—Show inventory associated with at least one closed task.<br>● **All Tasks**—Show all inventory associated with tasks, open or closed. |
| **Tasks Type** | Select one of the following to filter the inventory display by the type of task associated with inventory:<br>● **Manual inventory review**—Show inventory associated with a least one task requesting that a manual legal or security review be performed. (This review is needed to flag the inventory as accepted or rejected.)<br>● **Remediate Inventory**—Show inventory (currently or previously rejected) associated with at least one task requesting that software development take some action to make rejected inventory acceptable.<br>● **Miscellaneous**—Show inventory associated with at least one task requesting that additional attention of some sort be given to the inventory.<br>● **Any**—Show all inventory associated with tasks of any type. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| Column/Field | Description |
|---|---|
| **Inventory Tasks Age** | Select one of the following to filter the inventory display by the time frame in which tasks associated with inventory items have been created:<br>● **Last day**—Show inventory associated with at least one task created within the last day. For example, if today is Feb 6th, search from Feb 5th 12 AM.<br>● **Last 7 days**—Show inventory associated with at least one task created within the last week. If today is Feb 6th, search from Jan 30th 12 AM.<br>● **Last month**—Show inventory associated with at least one task created within the last month. If today is Feb 6th, search from Jan 7th 12 AM (30 days).<br>● **Custom Date Range**—Show inventory associated with at least one task created in the specified time frame. Select a beginning (**From**) and ending (**To**) date from the popup calendar.<br>● **Any**—Show all inventory associated with tasks, no matter when the tasks were created. |
| **Inventory Task Owner** | Select one of the following to filter the inventory display by the user who is assigned to tasks associated with inventory items:<br>● **Only Mine**—Show inventory associated with at least one task assigned to you (the current user).<br>● **Specific User**—Show inventory associated with at least one task assigned to the specified user. (A **Select user** pop-up enables you to select the user.)<br>● **Any**—Show all inventory associated with tasks, no matter to whom the tasks are assigned. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

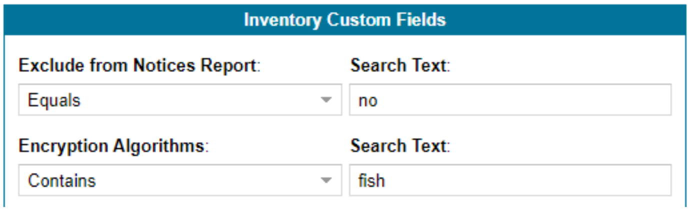
| Column/Field | Description |
|---|---|
| **Inventory Custom Fields** | The section is displayed only if one or more custom inventory fields have been defined for your site. If such fields have been defined, each field is listed, enabling you to set up a criterion for a given field that filters inventory by the field's value.<br>For each field whose value you want to use as a criterion for filtering inventory, do the following:<br>1. Under the field name, select the operation (**Contains** or **Equals**) in the field on the left.<br>2. In the **Search Text** field on the right, enter the partial or full field value by which to search inventory.<br><br>Inventory Custom Fields<br>Exclude from Notices Report: Equals — Search Text: no<br>Encryption Algorithms: Contains — Search Text: fish<br>If you have set up multiple custom fields as criteria, the **And** or **Or** operator selected for all criteria in the dialog is applicable across the custom-field criteria.<br>● To appear in search results when **Or** is selected, an inventory item must contain at least one of the custom-field criteria you defined.<br>● To be a candidate in the search results when **And** is selected, an inventory item must meet *all* the custom-field criteria you defined. |
| **Security Vulnerabilities** | The following options enable you to filter inventory by the attributes of the security vulnerabilities associated with inventory items.<br>If you accessed this dialog from the Inventory View, setting any of the following security-vulnerability criteria might increase the inventory search time significantly.<br>***Note ▪*** *When you search by the ID or severity of a suppressed vulnerability, the results do not include inventory items associated with component versions for which the vulnerability was suppressed.* |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| Column/Field | Description |
|---|---|
| **Security Vulnerability ID** | Enter the complete valid ID for the security vulnerability by which to filter the inventory display to show only those inventory items associated with the specified vulnerability. |
| **Security Vulnerability Severity** | Select one or more vulnerability severity levels by which to filter the inventory display to show only those inventory items associated with at least one vulnerability that has one of the selected severities.<br>The severity-level options differ depending on the CVSS version used by Code Insight.<br>If CVSS v3.x (3.0 and 3.1) is used, the following severity options are available:<br>● Critical (CVSS score 9.0 - 10.0)<br>● High (CVSS score 7.0 - 8.9)<br>● Medium (CVSS score 4.0 - 6.9)<br>● Low (CVSS score 0.1 - 3.9)<br>● None (CVSS score = 0)<br>If CVSS v2.0 is used, these severity options are available:<br>● High (CVSS score 7.0 - 10.0)<br>● Medium (CVSS score 4.0 - 6.9)<br>● Low (CVSS score 0.1 - 3.9)<br>● Unknown (N/A)<br>For more information about vulnerability severities, see Security Vulnerabilities Associated with Inventory in the “Using Code Insight” chapter. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| | Column/Field | Description |
|---|---|---|
| | **Security Vulnerability Age** | Select one of the following options to filter the inventory display by the time frame in which security vulnerabilities associated with inventory items were detected.<br>***Note ▪*** *The detection date is either the inventory creation date (if a vulnerability was reported when the inventory was created) or the date that a new vulnerability applicable to this inventory was delivered by the update service.*<br>● **Last day**—Show inventory associated with at least one vulnerability detected within the last day. For example, if today is Feb 6th, search from Feb 5th 12 AM.<br>● **Last 7 days**—Show inventory associated with at least one vulnerability detected within the last week. For example, if today is Feb 6th, search from Jan 30th 12 AM.<br>● **Last 30 days**—Show inventory associated with at least one vulnerability detected within the last month. For example, if today is Feb 6th, search from Jan 7th 12 AM.<br>● **Custom Date Range**—Show inventory associated with at least one vulnerability detected within a specific time frame. Select a beginning (**From**) and ending (**To**) date from the popup calendar.<br>● **Any**—Show all inventory associated with security vulnerabilities, no matter when the vulnerabilities were detected. |
| **Licenses and Versions** | The following options enable you to filter inventory by attributes of the selected license for inventory items.<br>If you accessed this dialog from the Inventory View, setting any of the following license criteria might increase the inventory search time significantly. | |
| | **License Name** | Enter the full or partial license name by which to filter the inventory display. For example, if you enter bsd in this field, Code Insight will find all inventory items whose **Selected License** value has the *bsd* string in its name. |

**Table 8-6 ▪** Advanced Inventory Search Dialog

| | Column/Field | Description |
|---|---|---|
| | **License Priority** | Select one or more license priorities by which to filter the inventory display. The display will show only those inventory items whose **Selected License** has one of the priorities you select:<br>● **P1**— Viral/Strong Copyleft<br>● **P2**—Weak Copyleft/Commercial/Uncommon<br>● **P3**—Permissive/Public Domain<br>● No License Found<br>For more information about license priority, see Auditing Scan Results in the Analysis Workbench in the “Using Code Insight” chapter. |
| | **No Associated Version** | Select this option filter to those licenses to which no version has been associated. |
| **Actions** | The following are actions you can take to define criteria logic and apply the filters. | |
| | **Apply *And* \|*Or* Criteria** | Select the boolean operator to apply to the search criteria:<br>● **Or**—To appear in the search results, an inventory item must contain at least one of the criteria you selected on this dialog. This is the default operator.<br>● **And**—To appear in the search results, an inventory item must meet *all* the criteria selected on this dialog. |
| | **Apply** | Click this button to apply the selected search criteria and return to the **Inventory Items** page to view the results. The title bar of the **Inventory Items** page denotes that the display shows only filtered results. |
| | **Clear Form** | Click this button to return the search criteria configuration to its default state. |
| | **Close** | Click this button to close this dialog and return to the **Inventory Items** page without applying your search criteria. |

# ALM Tab

The **ALM** tab on the **Administration** page allows you to configure Jira and other ALM (Application Lifecycle Management) instances for integration with Code Insight for the purpose of creating work items in external workflow systems. The tab contains the following columns and fields:

**Table 8-7 ▪** ALM Tab

| Column/Field | Description |
|---|---|
| **Application** | Name of the ALM application for which to add an instance. |
| **Add Instance** | Click to open a new **Instance** tab to configure an instance to point to a server in the ALM system. |
| **Existing Issues Sync Frequency** | Click the ✎ to the right of this field, and select the synchronization frequency that will apply to *all* the configured ALM instances. (The default value is **Hourly** repeated every **1** hour.)<br>● **Never**<br>● **Hourly** (enter number of hours)<br>● **Daily** (enter time of day)<br>● **Weekly** (enter day of the week and time of day)<br>Click ✓ to accept the updated synchronization frequency or ✖ to restore the previous frequency. |
| **Test Connection** | Click to validate that Code Insight can connect to the current instance based on the supplied ***ALM_type*** **Instance Name**, ***ALM_type*** **Server URL**, ***ALM_type*** **Username**, and ***ALM_type*** **Password**. |
| **Delete Instance** | Click to delete the current ALM instance after verifying that no project references to this instance exist. |
| ***ALM_type*** **Instance Name** | Unique name of the ALM instance. |
| ***ALM_type*** **Server URL** | URL of the ALM server to which to connect in the format `http(s):<`*`server_name_or_ip`*`>`. |
| ***ALM_type*** **Username**<br>***ALM_type*** **Password** | Credentials of ALM instance user for authentication on the ALM server. This user is also the designated reporter on work items (issues) created for the instance. |
| **Default Project Key** | Key for the project for which issues will be created on the ALM server. |
| **Default Issue Type** | The default issue type created on the ALM server. |
| **Default Priority** | Default priority of the issued created on the ALM server. |

**Table 8-7 ▪** ALM Tab (cont.)

| Column/Field | Description |
|---|---|
| **Default Assignee** | The default user to whom to assign work items created for this instance. |
| **Default Summary Text** | Default text to display as a summary for the issue on the ALM server. The text supports the following Code Insight variables: `$PROJECT_NAME`, `$INVENTORY_ITEM_NAME`, `$COMPONENT_NAME`, `$VERSION_NAME`, `$LICENSE_NAME`, `$NUMBER_VULNERABILITIES`, `$NUMBER_FILES`, or `$INVENTORY_URL`. |
| **Default Description Text** | Default text to display as a description for the issue on the ALM server. The text supports the following Code Insight variables: `$PROJECT_NAME`, `$INVENTORY_ITEM_NAME`, `$COMPONENT_NAME`, `$VERSION_NAME`, `$LICENSE_NAME`, `$NUMBER_VULNERABILITIES`, `$NUMBER_FILES`, or `$INVENTORY_URL`. |

# Analysis Workbench

The **Analysis Workbench** is a facility that lets you examine the evidence in a project's scanned codebase in your project and interact with the inventory resulting from the scan.

The **Analysis Workbench** has the following fields.

***Note ▪*** *For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported in Code Insight. The* ***Analysis Workbench*** *indicates which remotely scanned files contain license evidence (a green icon is displayed next to the files under a remote scan-agent node in the* ***Codebase Files*** *and* ***File Search Results*** *pane) and lets you view this evidence on the* ***Evidence Details*** *pane and a file's* ***Evidence Summary*** *pane.*

***Note ▪*** *Some panes do not contain data until you choose a file in another pane.*

**Table 8-8 ▪** Analysis Workbench

| Column/Field | Description |
|---|---|
| **Legend** | A color-coded and hyperlinked guide to the files and inventory in your scanned codebase:<br>● **New Evidence**: Click this link to filter the search results to display only files that are new since the last scan. If only a single scan took place, all files with evidence are displayed in the **Files Search Results** pane.<br>● **Reviewed**: Click this link to display files in the **File Search Results** pane that have been reviewed.<br>● **Exact**: Click this link to display files in the **File Search Results** pane that are exact matches.<br>● **Copyrights**: Click this link to display files in the **File Search Results** pane that contain copyright text.<br>● **Email/URLS**: Click this link to display files in the **File Search Results** pane that contain email addresses and URLs.<br>● **Licenses**: Click this link to display files in the **File Search Results** pane that contain licenses.<br>● **Search Terms**: Click this link to display files in the **File Search Results** pane that match default search terms.<br>● **Source**: Click this link to display files in the **File Search Results** pane that match |
| **Codebase Files pane** | A tree listing the files in the project codebases. The tree can include one or more nodes, each node identifying a specific Scan Server or remote scan agent and listing the files scanned by that scanner. |
| **File Search Results** | A tree listing the files resulting from a search. The tree is organized by the nodes and directories containing the files. Drill down into the nodes and directories to view the files. |
| **File Details Tab** | Click a codebase file to open the **Files Details** tab (in the middle pane). This tab includes a expandable header that lists metadata about the selected codebase file, as well as the three sub-tabs—**Evidence**, **Exact Matches**, and **Partial Matches**—available to examine the file's open-source or third-party evidence. For more information, see Examining and Managing Open-Source Evidence for a Given File. |
| **Inventory Items (*x*) Pane** | |
| **Current View** | Lists what portion of the project inventory that is being displayed. |

**Table 8-8 ▪** Analysis Workbench (cont.)

| Column/Field | Description |
|---|---|
| **Quick Filters** | Provides options to quickly filter the inventory items listed:<br>● Published (*x*)<br>● Not Published (*x*) |
| **Clear Filter** | Clears any search terms that have been entered. |
| **Search** | Enter terms to search for in the inventory. |
| **Add New** | Click to create a new inventory item on the **New Inventory Item** tab. |
| **Publish** | Highlight an inventory item from the list and click **Publish** to publish the item. |
| **Recall** | Click to recall (remove) a published inventory item from **Inventory Items** list if it does not fit the criteria for inclusion. |
| **Delete** | Highlight an inventory item from the list and click **Delete** to delete the item from inventory. |
| **Inventory Details tab** | Click an inventory item in the list to open the Inventory Details tab (in the middle pane) showing information about the inventory item. See Inventory Details Tab in the Analysis Workbench for details. |
| **Evidence Details** | Click **Evidence Details** (in the middle pane header) to open the **Evidence Details** tab in the middle tab. From here, you can view a summary of OSS and third-party evidence found across the codebase during the last scan. You can also filter the evidence based on files selected in the **Codebase Files** pane; or filter the files in the **Codebase Files** pane by selected evidence. |

**See Also**
Opening the Analysis Workbench
The Analysis Workbench Layout
Reviewing Published Inventory for a Project

# Branch Project: Project Copy Settings

The **Project Copy Settings** page in the **Branch Project** wizard defines the parameters used by the branching process to import file-audit data, inventory, and inventory-review information from the source project to the branch project.

If neither **Upload Codebase** nor **Sync from Control Version** was selected on the **Project Information** page, this import process copies only inventory and inventory-review information from the source project to the branch project. In this scenario, no file information will be associated with the inventory copied to the branch project.

For a description of the procedures related to the **Project Copy Settings** page, see the following:

- Branching a Project
- Step 4: Configuring a Project Copy

The following describes the properties and actions available on the **Project Copy Settings** page:

**Table 8-9 ▪** Branch Project: Project Copy Settings Page

| Field | Description |
|---|---|
| **Add Files to Inventory** | Refer to Import Project Data Dialog for a description of this field and its related **File matching criteria** fields. The branch project is the same as the "target project" in this description.<br>If the branching process is performing an inventory-only copy to the branch project (that is, if neither **Upload Codebase** nor **Sync from Control Version** was selected on the **Project Information** page), this criterion is ignored during the branching process. |
| **Mark Files as Reviewed** | Refer to Import Project Data Dialog for a description of this field and its related **File matching criteria** fields. The branch project is the same as the "target project" in this description.<br>If the branching process is performing an inventory-only copy to the branch project (that is, if both **Upload Codebase** and the **Sync from Control Version** are not selected on the **Project Information** page), this criterion is ignored during the branching process. |
| **Inventory Notes Handling** | Refer to Import Project Data Dialog for a description of this field. The branch project is the same as the "target project" in this referenced description. |
| **Inventory Usage Handling** | Select one of these options to define how the branch process should handle the Usage attributes for inventory items:<br>● **Copy existing usage field values**—Copy the existing Usage values for the inventory items from the source project to the branch project. (Default)<br>● **Reset usage field values to system default**—Do not copy existing Usage values for inventory items from the source project to the branch project. Instead, in the branch project, reset all Usage fields for these inventory items to the system default value: **Unknown**.<br>For a description of the inventory usage fields, refer to Usage tab in the **Project Inventory Details Pane** topic. |
| **Next** | Click this button to validate the information on this page. If errors are found, you must correct them before moving to the next page. If no errors exist, the **Summary** page in the wizard open. |

**Table 8-9** ▪ Branch Project: Project Copy Settings Page (cont.)

| Field | Description |
|---|---|
| **Back** | Click this button to move to the previous wizard page.<br>Note that if you navigate back to the **Project Information** page, you cannot edit the **Name** field for project. Additionally, if you uploaded a codebase from the **Upload Codebase** page, the **Scan Server** and **Upload Codebase** options on the **Project Information** page are disabled. |
| **Cancel** | Click this button to cancel the project-branching setup. The branch project and any uploaded codebases for the project are deleted from Code Insight. |

# Branch Project: Project Information

The **Project Information** page in the **Branch Project** wizard identifies the essential information needed to create the branch project. Certain fields are pre-populated with values from the current project but can be edited as needed for the new project. After you complete the fields and click **Next**, the branch project is created if the information you provided is valid.

For a description of the procedures related to the **Project Information** page, see the following:

- Branching a Project
- Step 1: Creating the Branch Project

The following describes the properties and actions available on the **Project Information** page:

**Table 8-10** ▪ Branch Project: Project Information Page

| Field | Description |
|---|---|
| **Name** | Enter a name for the branch project. The name must be unique in your Code Insight system. |
| **Description** | If necessary, edit the description for the branch project. This field is initially populated with the description of the source project. |
| **Policy Profile** | If necessary, select a different policy profile for the branch project to automate its inventory review process. This field is initially populated with the policy profile used by the source project.<br>A given policy profile uses a combination of policies to automatically mark published inventory items as approved or rejected without the need of a manual review. (Inventory items that are neither approved or rejected by policy are marked as **Not Reviewed** and will require a manual review.) For more information about policy profiles, see Policy Details Window and Managing Policy Profiles. |

**Table 8-10 ▪** Branch Project: Project Information Page (cont.)

| Field | Description |
|---|---|
| **Scan Server** | If necessary, select a different Scan Server that will scan the codebase for the branch project.<br>This field is initially populated with the Scan Server used by the source project. |
| **Scan Profile** | If necessary, select a different scan profile that defines the settings applied whenever the branch project is scanned.<br>This field is initially populated with the scan profile used by the source project. |
| **Project Visibility** | If necessary, change the visibility attribute—**Public** or **Private**—of the branch project.<br>● **Public**—A project that provides read-only access to any user in the system. To what degree a user can interact with the project depends on whether the user has a project role and what the role is—Project Administrator, Analyst, or Reviewer.<br>● **Private**—A project that is hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. Additionally, project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see a given private project.<br>This field is initially populated with the attribute of the source project. |
| **Project Risk** | If necessary, change the vulnerability risk value (**Low**, **Medium**, or **High**) of the branch project. This field is initially populated with the risk value of the source project. |
| **Project Status** | If necessary, change the status of the branch project. (The meaning of these statuses might be adjusted for your site.)<br>● **Not Started**—Indicates that the project scan results are not yet available for manual analysis.<br>● **Analysis in Progress**—Indicates that the project scan results are available for manual analysis, or a manual analysis is underway.<br>● **Analysis Completed**—Indicates that manual analysis is finished; and the published inventory is ready for legal, security, or software-engineering review.<br>● **Project Complete**—Indicates that the project analysis and review processes are complete, and notices content for all OSS and third-party software is finalized.<br>This field is initially populated with the status of the source project. |

**Table 8-10** ▪ Branch Project: Project Information Page (cont.)

| Field | Description |
|---|---|
| **Source Code Options** | Select one or both options (or neither option) defining the method for obtaining source code for branch project:<br>● **Upload Codebase**—The source codebase is uploaded from an archive accessible from current instance. By default, this option is always selected. (You can upload multiple codebases.)<br>● **Sync from Source Control**—The source code base is obtained through a synchronization process with one or more instances of your site's Source Control Management (SCM) system. For more information about the synchronization process with SCMs, see Configuring Source Code Management. This option is selected by default only if SCM instances were configured in the source project.<br>Alternatively, to copy only inventory and inventory-review information from the source project to the branch project, do not select either of these options. No file information will be associated with the inventory copied to the branch project. |
| **Copy Project Users** | Select the option that determines whether project roles of the source project are copied to the branch project:<br>● **Yes**—All current project-role assignments—including Project Administrators, Analysts, Reviewers, Observers, and the Legal, Developer and Security contacts—are copied from the source project to the branch project once the branching process completes. (This is the default for the branching process.)<br>The user who performs the branching process is assigned the Project Contact role.<br>● **No**—The Project Administrator user who is creating the branch project is assigned to each of the project roles in the branch project. Additionally, those users assigned to roles by the System Administrator as global defaults for all new projects are added to the branch project (see "Setting Project Defaults" in the *Code Insight Installation & Configuration Guide*). |

**Table 8-10 ▪** Branch Project: Project Information Page (cont.)

| Field | Description |
|---|---|
| **Retain Child Project Links** | Select the option that determines whether the entire child hierarchy of the source project is copied to the branch project. This hierarchy includes the child projects directly associated with the source project, any child projects of those projects, and so on. (See Identifying Child Projects for a Project for more information about project hierarchy.) When this hierarchy is copied, links to the child projects are also copied.<br>This option does not copy the parent hierarchy that might be associated with the source project.<br>● **Yes**—The entire child hierarchy of the source project is copied to the branch project, along with links to the projects in the hierarchy.<br>● **No**—The child hierarchy of the source project is *not* copied to the branch project. |
| **Next** | Click this button to create the project and move to the next wizard page:<br>● If selected **Upload Codebase** in the **Source Code Options** section, the Branch Project: Upload Codebase page opens.<br>● If you selected only **Sync from Source Control** in the **Source Code Options** section, the Branch Project: Version Control Settings page opens.<br>● If you selected neither option, the Branch Project: Project Copy Settings page opens.<br>However, if errors are found on this page, you must correct them before the project can be created and you can move to the next page. |
| **Cancel** | Click this button to cancel the project-branching setup. |

# Branch Project: Summary

The **Summary** page in the **Branch Project** wizard provides an overview of the parameters that you defined for the project-branching process. From this page, you can start the branching process. Alternatively, you can navigate back to other pages in the wizard to make changes before starting the branching process, or you cancel the entire branching setup.

Once the branching process starts, any SCM synchronization is performed first. Then the branching process scans the branch-project codebase and finally performs an import to copy file-audit data, inventory, and inventory-review information from the source project to the branch project. For an overview of the branching-operation phases, see Overview of the Branching Operation.

For a description of the procedures related to the **Summary** page, see the following:

- Branching a Project
- Step 5: Initiating the Branching Operation

The following describes the actions available on the **Summary** page:

**Table 8-11 ▪** Branch Project: Summary Page

| Field | Description |
|---|---|
| **Finish** | Click this button to initiate the project-branching process. The dashboard for the new branch project is displayed, showing the current status of the branching process. Once the branching is finished, you can navigate to the **Project Inventory** tab and the **Analysis Workbench** for the branch project to proceed with the file-audit and inventory-review processes. |
| **Back** | Click this button to move to the previous wizard page. Alternatively, you can click any of the enabled tabs for wizard pages to move directly to a page. This backward navigation allows you to make changes to the parameters you set for the project-branching process.<br>Note that if you navigate back to the **Project Information** page, you cannot edit the **Name** field for project. Additionally, if you uploaded a codebase from the **Upload Codebase** page, the **Scan Server** and **Upload Codebase** options on the **Project Information** page are disabled. |
| **Cancel** | Click this button to cancel the project-branching setup. The branch project and any uploaded codebases for the project are deleted from Code Insight. |

# Branch Project: Upload Codebase

The **Upload Codebase** page in the **Branch Project** wizard identifies and uploads one or more codebase files for the branch project. For each codebase you want to upload, repeat the process of selecting the codebase archive, specifying the upload options, and then uploading the codebase.

This page is enabled only if you selected the **Upload Codebase** option from the previous **Project Information** page.

If you also selected the **Sync with Source Control** option on the **Project Information** page, the codebases obtained through synchronization with your Source Control Management system (described in Branch Project: Version Control Settings) during the branching operation will be added to the codebases uploaded from this page. For a description of the procedures related to the **Upload Codebase** page, see the following:

- Branching a Project
- Step 2: Uploading a Codebase (Optional)

The following describes the properties and actions available on the **Upload Codebase** page:

**Table 8-12 ▪** Branch Project: Upload Codebase Page

| Field | Description |
|---|---|
| **Select Archive File** | Click this button to select the archive file containing the codebase you want to upload. The selected archive is displayed in the text box next to this button.<br>Only a .zip, .tar, .tar.gz, or .7z archive is accepted. The maximum archive file size is 10 GB. |
| **Delete existing project codebase files** | Select this option to delete all codebase files that you might have already uploaded from this page and replace them with files from the codebase currently selected for upload. If you leave this option unchecked, files from the selected codebase are added to the already uploaded files. |

**Table 8-12 ▪** Branch Project: Upload Codebase Page (cont.)

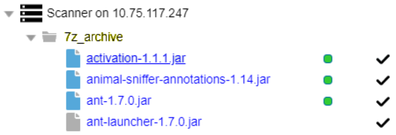
| Field | Description |
|---|---|
| **Archive Expansion Options** | Configure the behavior of archive expansion during the upload:<br>● **Uploaded file only**—Extract the files from the uploaded archive. Any extracted archives are not expanded.<br>● **Uploaded file and first-level archives only**—Extract the files from the uploaded archive and expand all first-level archives in the codebase. Note that the expanded archive itself is retained along with its extracted contents in the parent folder.<br>● **Uploaded file and all contained archives**—Extract the files from the uploaded archive and expand archives at all levels (that is, archives with archives within archives and so forth) in the codebase. Note that each expanded archive is retained along with its extracted contents in the parent folder.<br>Configure the of the upload process once archives are expanded. These settings are optional and are enabled only if the **Uploaded file and first-level archives only** or **Uploaded file and all contained archives** option has been selected.<br>● **Delete archive files after expansion**—Remove those archives that have been expanded during an upload. (The archive is removed from the uploaded codebase after the upload is finished.) If you leave this option unselected, the archive is retained as an additional file directly under its parent folder.<br>● **Append value to expanded archive directory name**—(Optional) Define a string to append to the name of any folder automatically created during the upload to store an archive's contents. After the codebase is scanned, this appended string helps you to identify those folders in the codebase tree whose contents were extracted from archives, especially if the original archives were removed from the codebase during the upload (see the previous option).<br>For example, suppose the appended value is `_archive`, and the upload process extracts an archive called `7z.zip`. After the upload process expands the archive, the name of the folder containing the archive contents becomes `7z_archive`, as shown in this example. Note that the example also shows that the 7z.zip archive has been removed due to the selection of **Delete archives after expansion**.<br><br>Scanner on 10.75.117.247<br>7z_archive<br>activation-1.1.1.jar ● ✓<br>animal-sniffer-annotations-1.14.jar ● ✓<br>ant-1.7.0.jar ● ✓<br>ant-launcher-1.7.0.jar ✓<br>This appended value has a maximum of 20 characters and does not support certain special characters. (Hover over the ⓘ icon for a list of unsupported characters.) For more information about archive expansion during a codebase upload, see More About Archive Expansion Behavior During Codebase Uploads. |

**Table 8-12 ▪** Branch Project: Upload Codebase Page (cont.)

| Field | Description |
|---|---|
| **Upload Project Codebase** | Click this button to upload the selected codebase.<br>Once the upload completes, you can upload another codebase by selecting its archive, specifying the appropriate expansion option for the upload, and clicking this button again. You can upload as many codebases as needed. |
| **Next** | Click this button to move to the next available wizard page.<br>If you selected **Sync from Source Control** on the **Project Information** page, the **Version Control Settings** page in the wizard opens. If you did not select this option, the **Project Copy Settings** page opens. |
| **Back** | Click this button to navigate back to the **Project Information** page. Note that you cannot edit the **Name** field for project. Additionally, if you uploaded a codebase, the **Scan Server** and **Upload Codebase** options on the **Project Information** page are disabled. |
| **Cancel** | Click this button to cancel the project-branching setup. The branch project and any uploaded codebases for the project are deleted from Code Insight. |

# Branch Project: Version Control Settings

The **Version Control Settings** page in the **Branch Project** wizard configures one or more Source Control Management (SCM) instances, enabling the branching process to synchronize the branch project with remote codebase repositories in your site's SCM applications. The synchronization takes place once the branching process begins. For more information about how to set up for SCM synchronization and how the synchronization process works, refer to Configuring Source Code Management.

By default, any SCM instances used by the source project are automatically copied to this page, each instance defined on a separate tab. However, you can edit or remove any of these instances or add new ones for the branch project. Alternatively, you can choose not to include any SCM instances on the **Version Control Settings** page, but can always return to this page to add instances later during setup if you want.

This page is enabled only if you selected the **Sync from Source Control** option from the previous **Project Information** page.

If you also uploaded codebases from the **Upload Codebase** page, the codebase files obtained through SCM synchronization, as defined on this page, will be added to the uploaded codebase files to provide the complete codebase for the branch product.

For a description of the procedures related to the **Version Control Settings** page, see the following:

- Branching a Project
- Step 3: Configuring Synchronization with a Source Code Management Instance (Optional)

The following describes the properties and actions available on the **Version Control Settings** page:

**Table 8-13 ▪** Branch Project: Version Control Settings

<table>
<tr><th>Field</th><th colspan="2">Description</th></tr>
<tr><td><b>Add Instance</b></td><td colspan="2">Click this button to create a new instance to connect to and synchronize with a remote SCM repository.</td></tr>
<tr><td><b>Application</b></td><td colspan="2">Select the type of SCM application to which the remote repository belongs: <b>Git</b>, <b>Perforce</b>, or <b>TFC</b> (Team Foundation Server).</td></tr>
<tr><td rowspan="4"><b>Instance tab</b></td><td colspan="2">If the source project contains SCM instances, these are copied to the branch project, each on a separate tab. Additionally, a new tab is created for each new SCM instance you decide to set up. The following provides information about the properties required for each instance and the actions you can perform on the instance.</td></tr>
<tr><td><b>Instance properties</b></td><td>Provide or edit the properties for the new or existing SCM connection instance. Click the appropriate link below for a description of the properties used to configure an instance based on the SCM application type.<br>● <b>Git instance</b>—Refer to Configuring a Git SCM Instance.<br>● <b>Perforce instance</b>—Refer to Configuring a Perforce SCM Instance.<br>● <b>TFS instance</b>—Refer Configuring a TFS SCM Instance.</td></tr>
<tr><td><b>Test Connection</b></td><td>To ensure that Code Insight is able to connect to the remote SCM repository specified by the instance, click this button. After a moment, Code Insight displays a success message if the connection is successful. If the connection is not successful, ensure that your entries on the <b>Instance</b> tab are correct, and click <b>Test Connection</b> again.</td></tr>
<tr><td><b>Delete Connection</b></td><td>Click this button to permanently remove this instance from the branch project (and thus from the Code Insight system). Keep in mind that, if this instance was copied from the source project and you delete it here, the branch might no longer contain the same codebase files as the source project.</td></tr>
<tr><td><b>Next</b></td><td colspan="2">Click this button to perform a final “test connection” on each connection instance incrementally.<br>If all connections are successful, the <b>Project Copy Settings</b> page in the wizard is opened. If a connection to a given repository fails, an error message is displayed, specifying which connection instance has failed. You must correct all connection issues before proceeding with the project-branching setup.</td></tr>
<tr><td><b>Back</b></td><td colspan="2">Click this button to move to the previous wizard page.<br>Note that if you navigate back to the <b>Project Information</b> page, you cannot edit the <b>Name</b> field for project. Additionally, if you uploaded a codebase from the <b>Upload Codebase</b> page, the <b>Scan Server</b> and <b>Upload Codebase</b> options on the <b>Project Information</b> page are disabled.</td></tr>
</table>

**Table 8-13 ▪** Branch Project: Version Control Settings (cont.)

| Field | Description |
|---|---|
| **Cancel** | Click this button to cancel the project-branching setup. The branch project and any uploaded codebases for the project are deleted from Code Insight. |

# Branch Project Wizard

The **Branch Project** wizard automates the process of branching from one Code Insight project to another, enabling the target branch project to preserve any file-audit data, inventory, and inventory-review information that was created in the source project (the project from which you are branching). The wizard is accessed from the **Manage Project | Branch Project** option on the **Summary** page of the source project. For complete information about using the wizard, refer to Branching a Project.

The **Branch Project** wizard opens to the **Introduction** page. From this page, you click **Next** to begin the steps necessary to set up your project-branching process. The wizard navigates you through the following pages to complete the setup and initiate the branching process.

For a complete description of the procedures related to the Branch Project wizard and the project-branching process in general, see Branching a Project.

**Table 8-14 ▪** Branch Project Wizard Pages

| Wizard Page | Description |
|---|---|
| **Branch Project: Project Information** | The **Project Information** page in the **Branch Project** wizard enables you to define the properties needed to create the branch project. Once you navigate from this page, the project is created. |
| **Branch Project: Upload Codebase** | The **Upload Codebase** page enables you to upload one or more codebases to the branch project. If necessary, you can also synchronize with one or more Source Control Management instances (see Branch Project: Version Control Settings) to obtain the complete codebase for the project.<br>Uploading a codebase is optional. |
| **Branch Project: Version Control Settings** | The **Version Control Settings** page sets up the properties needed to synchronize the branch project with one or more remote codebase repositories from your site's Source Control Management applications. The connection to each SCM repository is defined on this page as an SCM instance, with each instance on a separate tab. The synchronization is performed once the project-branching process begins.<br>If you uploaded codebase files from the **Upload Codebase** page (see Branch Project: Upload Codebase), the codebase files obtained through SCM synchronization, as defined on this page, will be added to the uploaded codebase files to provide the complete codebase for the branch product.<br>Synchronizing with Source Code Management instances is optional. |

**Table 8-14 ▪** Branch Project Wizard Pages (cont.)

| Wizard Page | Description |
|---|---|
| Branch Project: Project Copy Settings | The **Project Copy Settings** page identifies the parameters used by the branching process to import inventory as well as file-audit and inventory-review information from the source project to the branch project.<br>If you have selected to not include codebase files in the branching process (that is, you have neither uploaded codebases nor enabled synchronization with remote codebases through SCM instances), only inventory and inventory-review information is imported to the branch project from the source project. No file-related information is imported. |
| Branch Project: Summary | The **Summary** page provides an overview of the parameters that you defined for the project-branching process. From this page, you can start the branching process or navigate back to other pages in the wizard to make changes before starting the branching process. You can also cancel the entire branching setup. |

# Code Insight Dashboard

The Code Insight dashboard is displayed when you access Code Insight. The dashboard contains the following options:

**Table 8-15 ▪** Code Insight Dashboard

| Column/Field | Description |
|---|---|
| **analyzed** | The number of lines of code that have been analyzed since Code Insight was installed. |
| **scanned** | The number of lines of code that have been scanned since Code Insight was installed. |
| **identified** | The number of open-source or third-party components that were identified in your codebase. |
| **view inventory** | Select this option to open the **Inventory** view, which provides a compilation of inventory across all current Code Insight projects. The list can be filtered at a basic level to show inventory for all projects, only your projects, or for a specific project. Filtering can be further refined by vulnerability severity, review status, and many other criteria. For more information, see Viewing Inventory Across All Projects. |
| **go to project** | Select this option to open the **Projects** view, which provides access to all current projects in Code Insight. For more information, see Accessing Projects in Code Insight. |

**Table 8-15 ▪** Code Insight Dashboard (cont.)

| Column/Field | Description |
|---|---|
| **view policy** | Select this option to open the **Policy** page, where you have access to all policies that automate the review process of project inventory when it is published. For more information, see Managing Policy Profiles. Access to this page requires Manage Policy permissions. |
| **administration** | Select this option to open the **Administration** page, where you have access to Code Insight administrative functionality. See the *Code Insight Installation and Configuration Guide* for a description of administrative tasks. Access to this page requires Code Insight System Administrator permissions. |

***Note ▪** If this is the first time Code Insight has been accessed or if no codebase has been analyzed, the **analyzed**, **scanned**, and **identified** fields will be empty.*

**See Also**
Projects Pane and Associated Dashboard
Policy Page
Users/Permissions Tab
Electronic Updates Tab
Email Server Tab
LDAP Tab
ALM Tab
Scan Servers Tab
Scan Profiles Tab

# Component Details Window

The following window is displayed when you click ⓘ next to the **Component** field on the **Inventory Details** tab in the **Analysis Workbench**. The window shows publicly available information about the component associated with the currently selected inventory item, including the following properties.

**Table 8-16 ▪** Component Details Window

| Column/Field | Description |
|---|---|
| **Component** | The name of the OSS or third-party component and its internal ID, as identified in the Code Insight data library. You can associate the inventory item with a different component from the **Inventory Details** tab (see Editing Inventory from the Analysis Workbench). |

**Table 8-16 ▪** Component Details Window

| Column/Field | Description |
|---|---|
| **Version** | The component version and its internal ID, as identified in the Code Insight data library. You can associate the inventory item with a different version of the component from the **Inventory Details** tab (see Editing Inventory from the Analysis Workbench). |
| **Forge** | The external repository associated with the component. You can click the forge link to open the forge website. |
| **Possible Licenses** | License candidates associated with this component. Click the ⓘ icon next to a given license to view information about the license on the License Details Window. |
| **Custom Component** | The **Yes** or **No** value indicating whether the component is custom (created by a user) or provided as part of the Code Insight data library |
| **Vulnerabilities** | A bar graph showing the count of known vulnerabilities by severity color for the component. Click the graph to view the list of vulnerabilities and their details. For details about the graph and vulnerabilities in general, see Security Vulnerabilities Associated with Inventory.<br>If no vulnerabilities have been found for the inventory item, the value **No** is displayed in place of the graph. |
| **Encryption** | The **Yes**, **No**, or **N/A** value indicating whether the component provides the encryption capabilities used in your product or whether these capabilities are not applicable. Encryption can affect export controls. |
| **CPE** | The list of CPE names—from the National Vulnerability Database—that are mapped to the component. CPE (Common Platform Enumeration) is a structured naming scheme that includes the component's vendor and product names in the following format:<br>**cpe://<part>:<vendor>:<product>**<br>where <part> is either **a** (applications), **h** (hardware platforms), or **o** (operating systems).<br>***Note ▪*** *The data provided represents only the part, vendor, and product; the version information is truncated from the CPE string.* |

# Create/Edit Scan Profile Dialog

Both the **Create Scan Profile** dialog and the **Edit Scan Profile** dialog contain the fields described in this table to define or update a scan profile. Code Insight System Administrators access either dialog from the **Scan Profiles** tab on the **Administration** page.

In addition to letting you create custom scan profiles, Code Insight ships with the following pre-defined scan profiles, which you can modify, assign to projects, or use as templates for creating your own profiles:

- **Basic Scan Profile (without CL)**—Defines a scan that uses Automated Analysis to detect evidence of open-source software (OSS) and third-party code in your codebase and generate an inventory of the findings. This scan does not perform exact-file or source-code matching and therefore does not use the Compliance Library (CL).
- **Standard Scan Profile**—Defines a scan that includes the basic scan features but also performs exact-file matching (that is, identifies codebase files that have an exact MD5 match in the CL). This scan requires the CL. This is the scan profile used as a template when you create a new profile. It cannot be modified.
- **Comprehensive Scan Profile**—Defines a scan that includes the basic scan features but also performs exact-file and source-code matching. (Source-code matches are strings in the codebase files that have an exact match to content in files in the CL). This scan requires the CL.

For your reference, the table below indicates which scan settings are enabled for each pre-defined profile.

**Table 8-17 ▪** Scan Profile Dialog

| Field | Description | Basic | Standard | Comprehensive |
|---|---|---|---|---|
| **Name** | Enter or edit the profile name. | X | X | X |
| **Perform Package/ License Discovery in Archives** | Select this option to have the Scan Server recursively perform package discovery and license detection within all archive files encountered in the project codebase. By default, this option is selected. | X | X | X |
| **Dependency Support** | Determine the level of dependency scanning to be performed by the Scan Server. The available options include:<br>● **No Dependencies**: Only top-level inventory items are reported without any dependencies. (Default)<br>● **Only First Level Dependencies**: Only first-level (or direct) dependencies are reported along with top-level inventory items.<br>● **All Transitive Dependencies**: All first-level and transitive dependencies are reported along with top-level inventory items. The Scan Server calls out to the relevant package management repository to obtain transitive dependency information.<br>For a description of Code Insight dependency support for supported ecosystems, see the “Automated Analysis” chapter in the *Code Insight User Guide*. | X | X | X |
| **Automatically Add Related Files to Inventory** | Select this option to have the system associate additional files to existing inventory items based on the data available in automatic detection rules. | X | X | X |

**Table 8-17 ▪** Scan Profile Dialog (cont.)

| Field | Description | Basic | Standard | Comprehensive |
|---|---|---|---|---|
| **Rescan Options** | By default, when a user initiates a regular rescan (that is, not a forced full rescan), only those files that have changed since the last scan are scanned. However, certain Code Insight events that have occurred since the previous scan can result in a rescan of all files (a full rescan). For a description of these events, see “Default Scan Behavior” in the *Code Insight User Guide*.<br>These options are used to override this default rescan behavior so that, even if any of the events that would normally call for a full rescan have occurred, all rescans will skip unchanged files and scan changed files only. | | | |
| | **Do not rescan files that have not changed since previous scan**<br>Select this option so that rescans always skip unchanged files and scan only those files that have changed since the last scan (even if events have occurred since the last scan that call for a full rescan). | | | |
| | **Apply this option to:**<br>If the **Do not rescan files...** option is selected, further clarify *which* unchanged files to skip during the rescan:<br>● All unchanged files<br>● Only unchanged files marked as reviewed<br>● Only unchanged files associated with inventory<br>● Only unchanged files that are both marked as reviewed and associated with inventory | | | |
| **Exact Matches** | Select this option to enable the detection and recording of scanned files that exactly match entire-file data in the Compliance Library (CL). | | X | X |

**Table 8-17 ▪** Scan Profile Dialog (cont.)

| Field | Description | Basic | Standard | Comprehensive |
|---|---|---|---|---|
| **Source Code Matches** | Select this option to enable the detection and recording of any source-code snippets in the scanned files that match data in the Compliance Library (CL).<br>If you enable this source-code matching, specify any of the following additional parameters for the matching process. | | | X |
| | **Include System-Identified Files**<br>Select this option if you want the Scan Server to perform source-code matching for files that have already been associated with one or more inventory items during automated analysis. | | | X |
| | **Include Files with Exact Matches**<br>Select this option if you want the Scan Server to perform source-code matching for files that have already been identified as having exact-file matches in the CL. | | | X |
| | **Minimum Source Code Matches**<br>Enter the minimum number of source-code matches that the scan needs to detect in a given codebase file before reporting the file as having such matches. (A *source-code match* is a snippet of code in a codebase file that matches an open-source code snippet found in the CL data.)<br>Enter a new minimum value from 1 to 32767. (The default is 3.)<br>For example, if this value is increased to `10`, ten code snippets in a given codebase file must match data in the CL before the scan reports the file as having source-code matches.<br>In general, the higher this value, the fewer source-code matches an analyzer has to review. | | | X |
| **Search Terms** | Provide a list of search terms to be used in the scan. Use the **+** button to add a term and the **-** button to remove a term. | X | X | X |
| **Scan Exclusions** | Provide a list of file extensions to be excluded from the scan. Use the **+** button to add an exclusion term and the **-** button to remove an exclusion. See "Creating Exclusion Patterns for Scan Profiles" in the *Code Insight Installation & Configuration Guide* for further instructions. | X | X | X |

**See Also**

Scan Profiles Tab
About Code Insight Scans
Applying a Scan Profile to the Project
Edit Project: Scan Settings Tab

# Custom Detection Rule Dialog

The **Custom Detection Rule** dialog enables you to create a custom detection rule. You can define custom rules as needed to supplement the internal detection rules used by Automated Analysis to automatically create inventory during a scan. The custom detection rules are saved to the Code Insight data library for global use across projects. For complete details about custom detection rules, see Managing Custom Detection Rules.

This dialog is accessed from two locations:

- From the **Inventory Details** tab in the **Analysis Workbench** for an inventory item of the “component” type—whether system-generated or manually created—to which codebase files have been manually associated (as described in Creating a Custom Detection Rule from Inventory of “Component” Type).
- From **Custom Detection Rules** tab accessed from the **Data Library** page on the Code Insight main menu (as described in Creating a Custom Detection Rule from Scratch).

The ability to edit certain fields depends on how you accessed the dialog. To help explain these differences, the following table designates the two access locations as “**Inventory Details** tab” and “**Custom Detection Rules** tab”.

The following describes the columns and actions you can perform from the **Custom Detection Rule** dialog.

**Table 8-18 ▪** Custom Detection Rule Dialog

| Category | Column/Field | Description |
|---|---|---|
| **Component selection** | The following fields describe the component on which the custom detection rule is based. If you have accessed this dialog from the **Inventory Details** tab for an inventory item in the **Analysis Workbench**, these fields are auto-populated with component information from the inventory item and are *not* editable. If you have accessed this dialog from the **Custom Detection Rules** tab, these fields are populated once you select the component and *are* editable as described below. | |
| | **Component** | The name of the component on which this detection rule is based.<br>If you accessed this dialog from the **Custom Detection Rules** tab, click **Lookup Component** to select the component and its version, license, and forge URL. The **License** and **URL** fields are populated accordingly.<br>If you accessed this dialog from the **Inventory Details** tab, this field is not editable. |
| | **License** | The license associated with the component.<br>If you accessed this dialog from the **Custom Detection Rules** tab, you cannot edit the field directly once it is populated from the component selection, but you can select a different license. To do so, click ✎ to switch to another license and, optionally, change the component version. Additionally, click ⓘ to view the details and text of the selected license as stored in the Code Insight data library.<br>If you accessed this dialog from the **Inventory Details** tab, this field is not editable. |
| | **Description** | A description of the component.<br>If you accessed this dialog from the **Custom Detection Rules** tab, this field is editable. It is not editable if you accessed the dialog from the **Inventory Details** tab. |
| | **URL** | The forge URL for the component.<br>If you accessed this dialog from the **Custom Detection Rules** tab, this field is editable. It is not editable if you accessed the dialog from the **Inventory Details** tab. |

**Table 8-18 ▪** Custom Detection Rule Dialog (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **License, notices, and note content** | The following fields are used to provide license or notice content and any audit notes for the inventory item generated from this rule. These field are editable.<br>If you accessed this dialog from the **Inventory Details** tab, these fields might be pre-populated with information from the manually created inventory. However, you can edit this information as needed. | |
| | **As-Found License Text** | The license content you want to associate with the inventory item. If no **Notices Text** content is provided, the Notices report uses the information in this field as the license text for the third-party component. For more information, see Finalizing the Notices Text for the Notices Report. |
| | **Notices Text** | The exact content to include in the Notices report. This is usually a modification of the content in **As-Found License Text**. (You can copy the **As-Found License Text** content to the **Notices Text** pane and edit it.)<br>If content exists in this field, the Notices report uses it as the license text for the third-party component and ignores any information in the **As-Found License Text** pane. For more information, see Finalizing the Notices Text for the Notices Report. |
| | **Audit Notes** | Any notes you want to add to the inventory item based on your findings during the analysis. |

**Table 8-18 ▪** Custom Detection Rule Dialog (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Associated codebase files** | This pane identifies the codebase files (by file name and MD5 value) on which to base the rule. You must identify at least one file.<br>If you have accessed this dialog from the **Inventory Details** tab, the files associated with the inventory item are automatically listed and available for selection. If you have accessed this dialog from the **Custom Detection Rules** tab, you must manually provide file name and MD5 value for each file.<br>Keep in mind that, if the custom detection rule is associated with multiple files, the scan uses OR logic when processing the files against the target codebase. Consequently, only one file match between codebase and the rule is required to automatically create an inventory item. | |
| | **File MD5** | If you accessed this dialog from the **Inventory Details** tab:<br>● To add one or more files, click the check-box next to each desired file.<br>● To remove a from the rule, click its check-box to deselect it.<br>If you accessed this dialog from the **Custom Detection Rules** tab:<br>● To add a file, click **Add File** and enter the exact file name and MD5 value for the file.<br>● To remove a file from the rule, click ✕ in the file entry. |
| **Actions** | The following are actions conclude the rule-creation session. | |
| | **Save** | Click **Save** to save the new custom detection rule to the Code Insight data library. You will be asked for confirmation to proceed with the creation. |
| | **Cancel** | Click **Cancel** to cancel the rule creation process. You will be asked for confirmation to proceed with the cancellation. |

**See Also**
Managing Custom Detection Rules
Creating a Custom Detection Rule from Inventory of “Component” Type
Creating a Custom Detection Rule from Scratch
Finalizing the Notices Text for the Notices Report

# Custom Detection Rules Tab

The **Custom Detection Rules** tab on the **Data Library** page lists the custom detection rules currently available for use in codebase scans in your Code Insight system. Custom detection rules are user-defined when needed to supplement the internal detection rules used by Automated Analysis to automatically create inventory during a scan. These custom rules are saved to the Code Insight data library for global use across projects.

From this page, you can also select to edit a rule or remove a rule from your Code Insight system.

For complete details about custom detection rules, see Managing Custom Detection Rules.

The following describes the columns and actions you can perform from the **Custom Detection Rules** tab.

**Table 8-19 ▪** Custom Detection Rules Tab

| Category | Column/Field | Description |
|---|---|---|
| **Actions** | **Enter Component Name search string** | To locate specific custom detection rules, enter a component-name string by which to filter the list of detection rules. The search results show only those rules whose component name contains the search string you provided. (This filter applies to only those custom rules visible in the UI; no call is made to the data library.) |
| | **Create Custom Rule** | Click to open the **Custom Detection Rule** dialog to create a new custom detection rule. |
| **Custom rule entry** | The following columns provide details about each custom detection rule and give you access to actions you can take on the rule. | |
| | **Component** | The name of the component on which the custom detection rule is based. |
| | **Version** | The component version. |
| | **License** | The license found in the Code Insight data library and associated with the component. |
| | **URL** | The forge URL for the component. |
| | **Actions for custom rule entry** | Actions you can perform on the currently selected rule:<br>● Click ✎ to edit the custom detection rule. The **Edit Custom Rule** dialog is opened.<br>● Click ✖ to delete the custom detection rule from your Code Insight system. The rule will no longer be applied during project scans. |

**See Also**
Managing Custom Detection Rules
Custom Detection Rule Dialog
Edit Custom Rule Dialog

# Edit (Default) Project Users Page

The **Edit Project Users** page, accessed from the **Manage Project** menu on the **Summary** tab for a specific project, is used to assign Code Insight users to various roles for the project.

The **Edit Default Project Users** page, available from the **Project Defaults** tab on the **Administration** page, is used to set project role assignments that default for any new project created (but which can then be edited at the project level on the **Edit Project Users** page).

For a description of the project roles and more information about the procedures used to manage them on the **Edit Project Users** page, see Assigning and Removing Project Users. (These same procedures basically apply to the **Edit Default Project Users** page.) Additionally, for a description of the permissions enabled for each project role, see the appendix Code Insight User Roles and Permissions.

The following describes the fields on the **Edit Project Users** and **Edit Default Project Users** page:

**Table 8-20 ▪** Edit (Default) Project Users Page

| Column/Field | Description |
|---|---|
| **Select Users** | The list of all users defined for your Code Insight system. From this list, you select the users to which you want to assign project roles. |
| **Add User** | Select one or more users in the **Select Users** pane, and then select the appropriate option—**Add to Analysts**, **Add to Reviewers**, or **Add to Observers**— from the **Add User** dropdown to add the users to the desired "role" pane. This procedure is an alternative to dragging and dropping users to the appropriate "role" pane. |
| **Enter Search Criteria** | Enter a full or partial user name to search for a user in the system and click 🔍 . |
| **Project Administrators** | The pane listing users who are currently assigned to the Project Administrator role for any project when it is created. To assign additional users to this role, drag and drop one or more users from the **Select Users** list to this pane. (Alternatively, select **Add to Project Administrators** from the **Add User** dropdown to add the selected users to the pane.) |
| **Analysts** | The pane listing users who are currently assigned to the Analyst role for any project when it is created. To assign additional users to this role, drag and drop one or more users from the **Select Users** list to this pane. (Alternatively, select **Add to Analysts** from the **Add User** dropdown to add the selected users to the pane.)<br>To remove a user from this role, click ✕ next to the user name in the pane. |
| **Reviewers** | The pane listing users who are currently assigned to the Reviewer role for any project when it is created. To assign additional users to this role, drag and drop one or more users from the **Select Users** list to this pane. (Alternatively, select **Add to Reviewers** from the **Add User** dropdown to add the selected users to the pane.)<br>To remove a user from this role, click ✕ next to the user name in the pane. |

**Table 8-20 ▪** Edit (Default) Project Users Page (cont.)

| Column/Field | Description |
|---|---|
| **Observers** | The pane listing users who are currently assigned to the Observer role. To assign additional users to this role, drag and drop one or more users from the **Select Users** list to this pane. (Alternatively, select **Add to Observers** from the **Add User** dropdown to add the selected users to the pane.)<br>To remove a user from this role, click × next to the user name in the pane.<br>***Note ▪*** *On the* ***Edit Project Users*** *page, the* ***Observers*** *pane is visible only for private projects. On the* ***Edit Default Project Users*** *page, this pane is always visible, enabling you to assign observers that will default for any private project that might be created. For more information, see Creating a Private Project.* |
| **Close** | Click this button to save your changes. |

**See Also**
Assigning and Removing Project Users

# Edit Custom Rule Dialog

The **Edit Custom Rule** dialog enables you to edit an existing custom detection rule. Custom detection rules are user-defined when needed to supplement the internal detection rules used by Automated Analysis to automatically create inventory during a scan. These custom rules are saved to the Code Insight library for global use across projects. For complete details about custom detection rules, see Managing Custom Detection Rules.

The following table describes the fields, buttons, and icons on the **Edit Custom Rule** dialog. You can edit any of the fields, using the methods described in the table.

**Table 8-21 ▪** Edit Custom Rule Dialog

| Category | Column/Field | Description |
|---|---|---|
| **Component selection** | The following fields describe the component on which the custom detection rule is based. These fields are editable. | |
| | **Component** | The name of the component on which this detection rule is based. You cannot edit this value directly, but you can switch to another component. To do so, click **Lookup Component** to select another component, along with its version, license, and forge URL. |
| | **License** | The license associated with the component. You cannot edit this value directly, but you can select a different license. Click ✎ to switch to another license and, optionally, change the component version.<br>Additionally, you can click ⓘ to view the details and text of the selected license as stored in the Code Insight data library. |
| | **Description** | A description of the component. |
| | **URL** | The forge URL for the component. |
| **License, notices, and note content** | The following fields are used to provide license or notice content and any audit notes for the inventory item generated from this rule. These fields are editable. | |
| | **As-Found License Text** | The license content you want to associate with the inventory item. If no **Notices Text** content is provided, the Notices report uses the information in this field as the license text for the third-party component. For more information, see Finalizing the Notices Text for the Notices Report. |
| | **Notices Text** | The content to include in the Notices report. This is usually a modification of the content in **As-Found License Text** pane. (You can copy the **As-Found License Text** content to the **Notices Text** pane and edit it.)<br>If content exists in this field, the Notices report uses it as the license text for the third-party component and ignores any information in the **As-Found License Text** pane. For more information, see Finalizing the Notices Text for the Notices Report. |
| | **Audit Notes** | Any notes you want to add to the inventory item based on your findings during the analysis. |

**Table 8-21 ▪** Edit Custom Rule Dialog (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Associated codebase files** | | This pane lists the codebase files (by file name and MD5 value) on which the rule is based. You can add or delete files as needed within this list. When adding a file, you are required to manually enter the its MD5 value.<br>At least one file must be associated with the rule.<br>Keep in mind that, if the custom detection rule is associated with multiple files, the scan uses "OR" logic when processing the files against the target codebase. Consequently, only one file match between codebase and the rule is required to automatically create an inventory item. |
| | **File MD5** | To add a file, click **Add File** and enter the exact file name and MD5 value for the file.<br>To remove a file from the rule, click ✖ in the file entry. |
| **Actions** | | The following are actions conclude your update session. |
| | **Save** | Click **Save** to save the rule updates to the Code Insight data library. You will be asked for confirmation to proceed with the creation. |
| | **Cancel** | Click **Cancel** to cancel your updates. You will be asked for confirmation to proceed with the cancellation. |

**See Also**

Managing Custom Detection Rules
Finalizing the Notices Text for the Notices Report

# Edit Project: General Tab

The **General** tab on the **Edit Project** dialog displays information about the selected project that you can edit. The tab contains the following fields:

**Table 8-22 ▪** Edit Project: General Tab

| Column/Field | Description |
|---|---|
| **Project Name** | The name of the selected project. You can change the name by typing over the current project name. |
| **Description** | A freeform text field in which you can enter a description for the project. This field provides enough space to add as much detail about the project as necessary. |

**Table 8-22 ▪** Edit Project: General Tab (cont.)

| Column/Field | Description |
|---|---|
| **Project Visibility** | The visibility status—**Public** or **Private**—of the project.<br>● **Public**—A project that provides read-only access to any user in the system. To what degree a user can interact with the project depends on whether the user has a project role and what the role is—Project Administrator, Analyst, or Reviewer.<br>● **Private**—A project that is hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. Additionally, project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see a given private project. |
| **Project Risk** | The current vulnerability risk value (**Low**, **Medium**, or **High**) for the project. To edit, select another value from the dropdown. |
| **Project Status** | The current status of the project. The following statuses are available and their suggested definitions are provided here. However, you can apply these statuses as appropriate for your site:<br>● **Not Started**—Indicates that the project scan results are not available for manual analysis. This value automatically defaults when the initial project scan has not been run or when the initial scan fails. However, you can manually change this status.<br>● **Analysis in Progress**—Indicates that the project scan results are available for manual analysis, or a manual analysis is underway. This value is automatically set when the initial project scan is complete. (For a project import, however, the status of the target project is retained.) You can manually change this status.<br>● **Analysis Completed**—Indicates that manual analysis is finished; and the published inventory is ready for legal, security, or software-engineering review. (You must manually set this value.)<br>● **Project Complete**—Indicates that the project analysis and review processes are complete, and notices content for all OSS and third-party software is finalized. (You must manually set this value.) |

**Table 8-22 ▪** Edit Project: General Tab (cont.)

| Column/Field | Description |
|---|---|
| **On the data import or rescan, delete inventory with no associated files** | This option determines whether "empty" system-generated inventory items are deleted in the target project during project imports and rescans. Empty inventory items have no associated files.<br>● **Selected**—Deletes empty inventory items from the target project during project imports and rescans. Only inventory items with associated files are retained/created.<br>● **Unselected**—Retains/creates all inventory items—with or without matching associated files in the target codebase—in the target project during imports and rescans. For example, you might want to retain inventory items to save their analysis details. (You will need to manually delete inventory that is not applicable to the current project.)<br>This configuration (unselected) is required when importing a scanned codebase into a project for which no codebase has been uploaded or obtained through synchronization. This option ensures that inventory is generated in the target project. |
| **Project Folder** | The folder in the list of projects under which the project is currently grouped. To edit the project location in the list, select one of the following:<br>● **Clear Project Folder button**—Click this button to remove the project from the current folder in the project list and place it in the root folder.<br>● **Select a New Folder dropdown**—Click the down arrow to locate and select an available folder to which to move the project. |

**See Also**
Editing the Project Definition and General Settings

# Edit Project: Project Hierarchy Tab

The **Project Hierarchy** tab on the **Edit Project** dialog for a given Code Insight project enables you to manage project's hierarchy. A project hierarchy provides a means to keep track of projects related each other. It is created by simply identifying one or more projects as *child projects* of the current project on which the **Project Hierarchy** tab is opened (called the *parent project*). Once the hierarchy is created, links are established in Code Insight between the parent project and the associated child projects so that you can easily move between projects to assess scan results and review inventory.

A project hierarchy is useful when your product application contains one or more modules, each with a codebase for which you want to set up a separate Code Insight project to track and assess the open-source or third-party software. By setting up a project hierarchy, you can easily switch between the main project for your application (the parent project) and the projects for the modules (the child projects) to complete the work needed to build a composite Bill of Materials.

Note that a child project, in turn, can be identified as a parent project to other child projects. Likewise, a given parent project can be identified as a child project to another parent project. Since hierarchies are created as needed, projects might have no association with a hierarchy.

For complete information about creating and managing project hierarchies, see Identifying Child Projects for a Project.

Once a project hierarchy is established for a given project, you can do the following:

- From the **Summary** page for the project, view and link to any of its child and parent projects (see Summary Tab).
- From the **Inventory** view showing inventory across all projects, examine the inventory of its child projects as well as link to any these projects (see Inventory View).

The following table describes the fields and button available on the **Project Hierarchy** tab.

**Table 8-23 ▪** Project Hierarchy Tab

| Category | Column/Field | Description |
|---|---|---|
| **Child project entry** | The following columns show the properties of each child project in the hierarchy for the project currently open and describe the actions available for the child project. | |
| | **Project Name** | The name of the project identified as child project of the project currently open (parent project). |
| | **Project Contact** | The main contact of the child project (initially the project creator). |
| | **Action** | Click **X** to disassociate the child project from the current parent project. Once you confirm to disassociate the project, child project is removed from the hierarchy. The links associated with this parent-child relationship are also removed from the **Summary** pages for the parent project and the project that was disassociated. The links are also removed from the **Inventory** view. |
| **Add Child Project** | Click this button to add a new child project to the current project. The **Add Child Project** dialog is opened, enabling you to select the new child project. (If necessary, navigate the project list more quickly by using the page navigation tools at the bottom of the dialog; or search the project list by entering a project name string in the search box.)<br>After you select a project, click **Add Project** to return to the **Project Hierarchy** tab, which now lists the new child project.<br>***Note ▪** To avoid cyclical parent-child relationships, the **Add Child Project** dialog does not list projects that are parents, parents of parents, children, or children of children of the current project.* | |

**Table 8-23 ▪** Project Hierarchy Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Actions for overall project settings** | These buttons control whether changes to project settings are saved across all **Edit Project** tabs. | |
| | **Save** | Click this button to save all your project edits and return to the **Summary** tab. |
| | **Cancel** | Click this button to return to the **Summary** tab without saving your project edits on other tabs. |

# Edit Project: Review and Remediation Settings Tab

The **Review and Remediation Settings** tab on the **Edit Project** dialog enables you to overwrite default settings that configure the automation of the review, remediation, and status notification processes for published inventory in your project. These settings, which work in conjunction with the set of policies in the project's policy profile, are used to set up the following in your project:

- The policy profile to associate with the project. The policies in the selected profile work in conjunction with the review, remediation, and notification configuration defined on this tab.
- Automatic creation of manual review tasks for inventory items not reviewed by policy during publication performed as part of a scan.
- Automatic assignment of review tasks to the default legal or security contact that you specify.
- Automatic creation of remediation tasks and associated external work items for inventory that is rejected either automatically by policy or during manual publication by an analyst.
- Automatic assignment of remediation tasks to the default engineering contact that you specify.
- Automatic rejection of published inventory impacted by new vulnerabilities detected in the latest scan or Electronic Update.
- The automatic generation of email notifications that alert the Project Contact of rejected or non-reviewed published inventory items that need attention.

See the following field descriptions for more information.

**Table 8-24 ▪** Edit Project: Review and Remediation Settings Tab

| Category | Section/Field | Description |
|---|---|---|
| **Automated Review Options** | **Policy Profile** | Select policy profile you want to associate with your project.<br>The policy profile contains a set of policies that use vulnerability scores and severities, license types, and component versions as criteria to automatically reject or approve inventory items during a codebase scan (or post-scan).<br>For more information about policy profiles in general, see Managing Policy Profiles in the “Using Code Insight” chapter. |
| | **Automatically reject inventory items impacted by a new vulnerability that violates your policy** | Determine what action the system should take for published inventory affected by a new security vulnerability discovered during a post-publication scan or an Electronic Update. The selected action applies to both non-reviewed and previously approved inventory items on the **Project Inventory** tab.<br>● Select this checkbox to automatically reject those project inventory items impacted by a new security vulnerability only if this vulnerability has a CVSS score or severity *greater than* the thresholds configured as policy for the Code Insight project. For each inventory item rejected due to a new security vulnerability, the 🚩 icon and a tip are added to indicate the status change and its reason.<br>If a new vulnerability does not exceed policy thresholds, the current status of the inventory item is not affected.<br>● Leave the checkbox unselected to retain the current status of inventory items impacted by the new vulnerability.<br>For information about setting policies that define CVSS-score and severity thresholds used to reject or approve inventory items automatically, see Policy Page and Policy Details Window. For information about associating these policies with a project, see Managing Policy Profiles. |

**Table 8-24 ▪** Edit Project: Review and Remediation Settings Tab (cont.)

| Category | Section/Field | Description |
|---|---|---|
| **Manual Review Options** | **What should happen if inventory items are not reviewed by policy?** | Determine what action should be triggered for those inventory items that are *not* affected by policy (and therefore have a **Not Reviewed** status) during the publication of inventory either as part of a scan or manually by a user:<br>● **do nothing**—Simply show the status of the inventory item as **Not Reviewed** on the **Project Inventory** tab.<br>● **send an email notification to the project contact**—Automatically send an email to the Project Contact, stating the need for a manual review of the item. The value for **Select the minimum priority...** (described in the next table entry) affects this option.<br>● **automatically create a manual review task**—Automatically create a manual review task assigned to the default security or legal contact, and send an email, notifying the contact about the assigned task. (Information about managing such a task to track the progress of a manual review is found in Creating Inventory from the Project Inventory Tab in the “Using Code Insight” chapter.) The value for **Select the minimum priority...** (described in the next table entry) affects this option. |
| | **Select the minimum priority to perform the action selected above** | (Enabled when an option other than **do nothing** is selected for the previous field.) Select the minimum inventory priority (**P1**, **P2**, **P3**, or **P4**) to which the value for the previous field applies.<br>For example, if the previous field is set to **send an email notification to the project contact** and minimum priority is set to **P3**, then the email notification will be sent for only those non-reviewed inventory items with a P1, P2, or P3 priority. No email notification will be sent for P4 inventory items.<br>***Note ▪*** *This option has no effect on the* ***do nothing*** *value.* |

**Table 8-24 ▪** Edit Project: Review and Remediation Settings Tab (cont.)

| Category | Section/Field | Description |
|---|---|---|
| | **What type of manual reviews will be performed on this project?** | Determine the type of manual review tasks to be generated:<br>● **Legal Only**—Review tasks are generated for those non-reviewed inventory items that meet no policy criteria. The tasks are automatically assigned to the default Legal reviewer.<br>● **Security Only**—Review tasks are generated for only those non-reviewed inventory items that have security vulnerabilities. The tasks are automatically assigned to the default Security reviewer.<br>● **both Legal and Security**—Review tasks are generated for all non-reviewed inventory items meeting no policy criteria and are assigned to the default Legal reviewer. Additionally, review tasks are generated for those non-reviewed inventory tasks associated with security vulnerabilities and are assigned to the default Security reviewer.<br>With this value, a single inventory item might have both a legal review task and security review task generated. However, if the default reviewers are the same user, a single task is created, describing the requirement for both a legal and security manual review. |
| | **Select reviewers for this project** | If desired, designate a new default reviewer to which to assign manual review tasks. (The available reviewer types—Legal or Security or both—depend on the type of manual reviews your site performs, as defined for the previous option.)<br>If your site generates both Legal and Security review tasks, Code Insight determines which reviewer—Legal or Security—is assigned the task and then notified of the task by email. (See the previous option description for “both Legal and Security” for more information about how this determination is made.)<br>The reviewer can then manage the task accordingly, possibly reassigning it to another user. For details about managing and reassigning tasks, see Creating and Managing Tasks for Project Inventory in the “Using Code Insight” chapter.<br>To select a new reviewer, click **Change User** next to the name of the current **Legal reviewer** or **Security reviewer** assignee, select a user from the **Select new...contact** dialog, and click **Apply**.<br>When a new default reviewer is selected, that user is automatically given the role of project “reviewer” if the user does not currently have this role. However, if a specific task is reassigned to another user, that user is not automatically given the “reviewer” role and must be given that role manually (if the user is not already have it). |

**Table 8-24 ▪** Edit Project: Review and Remediation Settings Tab (cont.)

| Category | Section/Field | Description |
|---|---|---|
| | **What should happen if inventory items are rejected?** | Determine what action should be triggered for those inventory items that are automatically rejected by policy during the publication of inventory either as part of a scan or manually by a user:<br>● **do nothing**—Simply show the status of the inventory item as **Reject** on the Project Inventory tab.<br>● **send an email notification to the project contact**—Automatically send an email to the Project Contact stating the need for remediation work on the inventory item.<br>● **automatically create a remediation task**—Automatically create a remediation task assigned to the default development contact (see the **Assignee for remediation work** option) and send an email, notifying the contact about the assigned task. (Information about managing such a task to track the remediation progress is found in Creating and Managing Tasks for Project Inventory in the “Using Code Insight” chapter.)<br>● **automatically create a remediation task and an external work item**—Automatically do the following:<br>    ● Create a remediation task assigned to the default development contact (see the **Assignee for remediation work** option) and send an email, notifying the contact about the assigned task. (Information about managing such a task to track the remediation progress is found in Creating and Managing Tasks for Project Inventory in the “Using Code Insight” chapter.)<br>    ● Associate a work item with the task, creating the work item in an Application Lifecycle Management (ALM) system (such as an issue in Jira). The work item is created and assigned using the settings defined for the ALM instance to which the Code Insight project is associated. For more information about configuring an ALM instance for the project, see ALM Settings in the “Using Code Insight” chapter. |
| | **Assignee for remediation work** | If desired, designate a new default development contact to which to assign remediation tasks. This contact can then manage the task accordingly—for example, reassigning it to another user or manually creating an external work item and assigning it to someone on the development team. For details about managing and reassigning tasks, see Creating and Managing Tasks for Project Inventory in the “Using Code Insight” chapter.<br>To select a new contact, click **Change User** next to the name of the current assignee, select a user from the **Select new...contact** dialog, and click **Apply**. |

**Table 8-24 ▪** Edit Project: Review and Remediation Settings Tab (cont.)

| Category | Section/Field | Description |
|---|---|---|
| **Actions** | These buttons control whether changes to project settings are saved across all **Edit Project** tabs. | |
| | **Save** | Click this button to save your project edits and return to the **Summary** tab. |
| | **Cancel** | Click this button to return to the **Summary** tab without saving your project edits. |

**See Also**

Editing the Project Definition and General Settings
Policy Page
Policy Details Window
Project Defaults Tab
Managing Policy Profiles
Creating Inventory from the Project Inventory Tab
Creating and Viewing External Work Items for a Project Inventory Task
Updating Inventory Review and Remediation Settings for a Project
ALM Settings

# Edit Project: Scan Settings Tab

The **Edit Project: Scan Settings** tab on the **Edit Project** dialog displays information about the scan settings defined for the selected project. You can edit the following information on this tab. (See also the Edit Project: General Tab to configure the project setting that determines whether the scan retains inventory that has no files associations.)

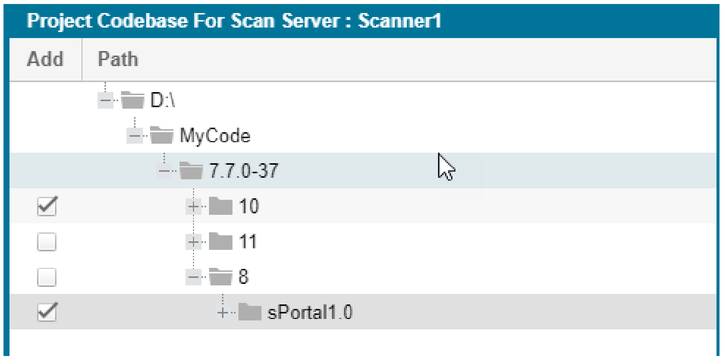
**Table 8-25 ▪** Edit Project: Scan Settings Tab

| Category | Column/Field | Description |
|---|---|---|
| **Scan Server Options** | These fields identify the Scan Server and profile used to run a project scan. | |
| | **Scan Profile** | The scan profile associated with the project. You can select a different scan profile from the dropdown list. Click ⓘ to view the properties of the currently selected scan profile. |
| | **Scan Server** | The Scan Server assigned to this project. This field is not editable.<br>Click ⓘ to view the properties of the currently selected Scan Server. |

**Table 8-25 ▪** Edit Project: Scan Settings Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Auto-Publish** | These options enable and configure the automatic publication of project inventory as part of the project scan process.<br>If the **Auto-publish system-created inventory items meeting this minimum Confidence Level** is selected to enable auto-publication, the other auto-publish options are made available. | |
| | **Auto-publish system-created inventory items meeting this minimum Confidence Level** | Select this option to enable the auto-publication of system-generated inventory items. (By default, the option is selected.) Then select the minimum Inventory Confidence level required to determine which items to auto-publish:<br>● **Low**—Auto-publish all system-generated inventory.<br>● **Medium**—Auto-publish only those system-generated inventory items with Medium and High confidence levels. (This is the default value.)<br>● **High**—Auto-publish only those system-generated inventory items with a High confidence level.<br>For a description of the Confidence levels and how they are used, see Inventory Confidence in the “Using Code Insight” chapter. |
| | **Do not auto-publish inventory items with an undetermined license** | Select this option to *not* auto-publish any system-generated inventory item with an undetermined license (that is, an inventory item whose **License** value is **I don’t know**). An undetermined license can occur under the following conditions:<br>● The scan was not able to identify a license for the given component during the scan and therefore set the **I don’t know** license value.<br>● The inventory item has multiple possible disjunctive licenses (for example, “GPLV2 or MIT”). However, the scan could find no evidence of the desired selected license and therefore set the **I don’t know** license value.<br>● The inventory item has multiple possible conjunctive licenses (for example, “GPLv2 and MIT”). However, since Code Insight currently supports only a single selected license, the scan automatically set the **I don’t know** value for the inventory item.<br>By default, this option is *not* selected, allowing the auto-publication of inventory with undetermined licenses.<br>The option is available only if **Auto-publish system-created inventory items meeting this minimum Confidence level** is selected. |
| | **Mark associated files as reviewed** | Select this option to automatically mark the files associated with each auto-published inventory item as “reviewed”. (By default, the option is selected.)<br>This option is available only if **Auto-publish system-created inventory items meeting this minimum Confidence level** is selected. |

**Table 8-25 ▪** Edit Project: Scan Settings Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Project Codebase for Scan Server** | These settings enable you to limit which directories are scanned in your codebase. | |
| | **Path** | From the interactive directory tree representing the project's codebase on the Scan Server, select the checkbox next to one or more top-level directories that you want to scan. To scan only specific subdirectories in a top-level directory, drill down in that directory and select the desired subdirectories.<br><br>Project Codebase For Scan Server : Scanner1<br>Add Path<br>D:\<br>MyCode<br>7.7.0-37<br>☑ 10<br>□ 11<br>□ 8<br>☑ sPortal1.0<br>If the Scan Server is down, no project tree is displayed. |
| | **Selected Paths** | The pane showing the path for each directory currently selected for the scan. As a quick method for removing a given directory from the scan without having to drill down in the tree to locate it, simply click the **X** next to the directory in this pane. If the Scan Server is down, this pane is blank. |
| **Actions** | These buttons control whether changes to project settings are saved across all **Edit Project** tabs. | |
| | **Save** | Click this button to save your project edits and return to the **Summary** tab. |
| | **Cancel** | Click this button to return to the **Summary** tab without saving your project edits. |

**See Also**

Editing the Project Definition and General Settings
Updating Scan Settings for a Project
Setting Policies to Publish Inventory Automatically

# Edit Token Dialog

The **Edit Token** dialog appears when you click the **Edit Token** icon on the **Preferences** page. It lets you edit an authorization token (that is, a JSON Web Token known as a JWT) used to authenticate calls to Code Insight REST APIs. The token is associated with the user account under which you are logged in or whose password you have changed in the **Change Password** fields on the **Preferences** page.

This dialog also allows you to copy the token value to the Clipboard so that you paste it whenever needed for Code Insight REST access (for example, when configuring the Code Insight plugins, importing or exporting project data, or running REST API directly).

The dialog has the following fields:

**Table 8-26 ▪** Edit Token Dialog

| Column/Field | Description |
|---|---|
| **Name** | Enter a name for the token you are creating. |
| **Token** | Displays the actual characters of the system-generated token. |
| **Select Token Text** | Click this button to highlight the token characters displayed in the **Token** field. To copy the toke to the clipboard, press **CTRL-C**. |
| **Expiration** | A read-only field that displays the expiration date of the token, or the text “Token has no expiration date.” |
| **Save** | Click this button to save your edits. |
| **Cancel** | Click this button to exit the **Edit Token** dialog without saving your edits. |

**See Also**
Preferences Page
Add Token Dialog
Exporting and Importing Project Data
Performing a Remote Scan

# Edit User Dialog

The **Edit Users** dialog is where you can edit users who are already in the Code Insight system. The dialog contains the following columns and fields:

**Table 8-27 ▪** Edit User Dialog

| Column/Field | Description |
|---|---|
| **Login** | Displays the login of the selected user. This field is read-only and cannot be changed. |
| **First Name** | Displays the first name of selected user. To change the user’s first name, type over the existing name. |
| **Last Name** | Displays the last name of selected user. To change the user’s last name, type over the existing name. |
| **Email** | Displays the email address of the user associated with the login. To change the user’s email address, type over the existing email address. |

**Table 8-27 ▪** Edit User Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Password** | The current password is not displayed in this field. A user with Code Insight System Administrator permissions can type a new password in this field. |
| **Password Confirm** | The current password is not displayed in this field. A user with Code Insight System Administrator permissions can type a new password in this field. |
| **Question** | The prompt that the user must answer to retrieve a forgotten password. |
| **Answer** | The answer to the question in the previous field. |
| **Submit** | Select **Submit** to have the system save your user edits. A prompt appears to notify you that your edits have been saved. |
| **Cancel** | Select **Cancel** to return to the Administration Users tab without saving your changes. |

**See Also**
Users/Permissions Tab

# Electronic Updates Tab

An initial full Electronic Update is run automatically after your initial startup of Code Insight. It provides the basis of a local data library used by Code Insight to identify OSS and third-party code in your codebase. The **Electronic Updates** tab on the **Administration** page enables you to configure how and when subsequent Electronic Updates are run to keep this library up to date. Refer to the following topics for more information:

- Overview of Electronic Update Setup
- Field Descriptions

For detailed instructions on how to schedule and run Electronic Updates, see “Configuring Code Insight” in the *Code Insight Installation & Configuration Guide*.

## Overview of Electronic Update Setup

The following describes the basics for configuring electronic updates:

- Specifying an Update as Server or Local
- Scheduling Electronic Updates
- Notification of an In-Progress Electronic Update

## Specifying an Update as Server or Local

The **Electronic Update Type** field (see Field Descriptions below) on the **Electronics Update** tab enables you to configure the Electronic Update to run as either a server or local update. The difference between the two methods is the means by which the Code Insight server obtains the files required to run the update:

- During a **server** Electronic Update, the most recent Electronic Update files are automatically downloaded from Revenera to the Code Insight server prior to processing the update.
- For a **local** Electronic Update, you must manually download the Electronic Update files from Revenera to a location that is locally accessible to the Code Insight server, such as a shared drive or a local USB drive. Then, when an update is triggered, the Code Insight server automatically uploads the files and proceeds with the update. This type of Electronic Update is useful when the Code Insight server has no external Internet access or when a specific Electronic Update version is needed for testing or demo purposes.

By default, the file download from Revenera is performed using HTTPS. However, you can configure the download process to use SFTP instead. For more information, see “Configuring an SFTP Connection for Downloading Update Files” in the *Code Insight Installation & Configuration Guide*. This configuration must be performed before running updates.

## Scheduling Electronic Updates

The remaining fields (see Field Descriptions below) on the **Electronic Updates** tab allow you either to schedule an Electronic Update to run automatically at regular intervals or to manually request an update as needed. By default, an update is *incremental* (that is, the update applies on changes from the previous update). However, you have the option force a *full* Electronic Update, which replaces all data from the previous update. (A full update might be necessary, for example, if the most recent update did not complete properly.)

In summary, you can schedule the following:

- An incremental Electronic Update (server type only) that runs automatically at a regular frequency that you define.
- An incremental Electronic Update (server or local type) that you manually run as needed.
- A full Electronic Update (server or local type) that you manually run when necessary. Use this option with caution as forcing a full update to run will take several hours to complete, similar to the initial update run when Code Insight was first installed.

***Note** ▪ Codebase scans cannot be performed during the Electronic Update process, but a scan that is already underway will not be interrupted when an update is scheduled to begin. The Electronic Update will be queued and automatically run based on queue order.*

## Notification of an In-Progress Electronic Update

Whenever an Electronic Update is in progress, a banner is displayed at the top of the Code Insight UI, indicating that an update is running and that scheduled scans will resume once the update completes. The banner is shown for any Electronic Update—whether server or local, forced or automatically run by schedule—and persists across all Code Insight pages.

The banner is automatically closed once the Electronic Update completes. (Users cannot manually close the banner.)

# Field Descriptions

The tab contains the following columns and fields:

**Table 8-28 ▪** Electronic Updates Tab

| Category | Column/Field | Description |
|---|---|---|
| **General** | The initial step in running an Electronic Update is to determine whether you are running it as a local or server update. For more information about these two types of updates, see Specifying an Update as Server or Local. | |
| | **Electronic Update Type** | Select the type of Electronic Update to run based how the Code Insight server obtains the Update Manifest and Update Data files required to perform the update:<br>● **Local**—This type of Electronic Update requires that you have manually downloaded these files from Revenera to a locally accessible location prior to the update. During the update, the Code Insight server uploads each of these files from this location, which you identify in the **Update Manifest File** and **Update Data File** fields.<br>● **Server**—The most recent Electronic Update files are automatically downloaded from Revenera to the Code Insight server as part of the Electronic Update process.<br>***Note ▪*** *By default, downloading files from Revenera is performed using HTTPS. However, you can configure the download process to use SFTP instead. For more information, see “Configuring an SFTP Connection for Downloading Update Files” in the Code Insight Installation & Configuration Guide. This configuration must be performed prior to running updates.* |

**Table 8-28 ▪** Electronic Updates Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Local Electronic Update configuration** | If you intend to run a *local* Electronic Update, you must specify the location of the two required “update” files that you manually downloaded from Revenera. The location of each file must be locally accessible. To run the update, use the **Run Update Now** option. | |
| | **Update Manifest File** | Click **Select File** to search for and select the Update Manifest file (`update_manifest.txt`) to upload to the Code Insight server. The manifest file contains the following:<br>● Information that Code Insight uses to determine whether to perform the update.<br>● The expected hash value for each data file stored in the `update.zip` file (see the **Update Data File** field description). This information will be compared with the hash values of actual files in the archive to ensure that the files have not changed or been tampered with. |
| | **Update Data File** | Click **Select File** to search for and select the data archive (`update.zip`) file to upload to the Code Insight server. This archive contains data files that provide the CVSS information used by Code Insight to perform update.<br>Code Insight uses the hash information in the manifest file (see the **Update Manifest File** field description) to ensure that the data files are the expected ones and have not changed or been tampered with. |

**Table 8-28 ▪** Electronic Updates Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Configuration for automatic Electronic Updates (server update only)** | Code Insight enables you to configure *server* incremental updates to run automatically at a frequency you define, as described below.<br>Note that you can always manually force an incremental or full update between the scheduled updates, or you can disable scheduled automatic updates altogether and manually run updates as needed. In either case, you would need to use the **Run Update Now** option to run an Electronic Update. | |
| | **Update Frequency** | Select from one of the available frequencies for running an incremental Electronic Update automatically:<br>● **Never**—If you select **Never**, Electronic Updates will not be run automatically. (Selection of this option hides any further dropdowns.)<br>You can always manually schedule an incremental or full update as needed using the **Run Update Now** option.<br>● **Daily**—If you select **Daily**, a second dropdown is displayed to choose the time of day when you want the Electronic Update to occur.<br>● **Weekly**—If you select **Weekly**, both the “time of day” dropdown and the **Select a day...** dropdown are displayed. Select both the time of day and the day of the week when you want the Electronic Update to occur. |
| | **Save Schedule** | Click this button to save the schedule. Your future incremental updates will run automatically according to the frequency you defined. |

**Table 8-28 ▪** Electronic Updates Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Configuration for manually running an Electronic Update** | You can manually run an Electronic Update at any time. The update is run immediately or placed in queue and initiated once all pending scans have completed. Currently, this is the only way to schedule a local update. If automatic server updates are also configured, a manually-run update is in addition to the automatic updates. | |
| | **Run Update Now** | Select the scope of the manually-run update:<br>● **Incremental Update**—Schedule an incremental Electronic Update. However, if no changes have occurred in the Electronic Update data since it was last run, the update is not executed.<br>● **Full Update**—Force a full Electronic Update to run whether or not changes have occurred in the Electronic Update data since it was last run. Use this option judiciously as a full Electronic Update takes several hours to complete, similar to the time required to run the initial update when Code Insight was installed. |
| | **Update** | Click this button to initiate the Electronic Update immediately or once pending scans are completed. |

**See Also**
"Configuring Code Insight" chapter in the *Code Insight Installation & Configuration Guide*

# Email Server Tab

The **Email Server** tab on the **Administration** page allows you to enable email notifications and set email options. The tab contains the following columns and fields:

**Table 8-29 ▪** Email Server Tab

| Column/Field | Description |
|---|---|
| **Enable Email Server** | Select **Yes** to enable Code Insight to use the email server or **No** to leave it disabled. The default is **No**. The rest of the fields on this page are not available until you select **Yes**. |
| **Sender's Email Address** | Enter the email address of the sender. |
| **SMTP Host Name** | Enter the Simple Mail Transfer Protocol (SMTP) host name. |
| **SMTP Host Port** | Enter the port number of the SMTP host. |
| **SMTP User Name** | Enter the SMTP user name. This field is optional. Leave it blank if you are using anonymous SMTP. |

**Table 8-29 ▪** Email Server Tab (cont.)

| Column/Field | Description |
|---|---|
| **SMTP User Password** | Enter the SMTP user password. This field is optional. Leave it blank if you are using anonymous SMTP. |
| **Enable SMTP over TLS** | Select **Yes** to use Transport Layer Security (TLS) to secure email over SMTP or select **No** to leave this option disabled. |

# Evidence Details Tab in the Analysis Workbench

The **Evidence Details** tab provides details about the inventory, component, and the files in the inventory.

***Note ▪*** *For files scanned by a Code Insight scan-agent plugin on a remote system, only license evidence is currently reported in Code Insight.*

The tab has the following fields:

**Table 8-30 ▪** Evident Details Tab

| Column/Field | Description |
|---|---|
| **Expand All/Collapse All** | Click to toggle between expanded and collapsed display. |
| **Search field** | Enter search criteria. |
| **Tree view** | Click to change the display to a tree view. |
| **List view** | Click to change the display to a list view. |
| **Select Evidence Types** | Click to select evidence types to display. |

# File Search Results Pane

The **File Search Results** pane displays the results of your file search. The **File Search Results** pane has the following fields.

***Note*** *▪ Some panes do not contain data until you choose a file in another pane.*

**Table 8-31** ▪ File Search Results Pane

| Column/Field | Description |
|---|---|
| ⟳ | Click to refresh the search. |
| **Advanced Search** | Click to open the **Advanced File Search** dialog on which you can choose a standard search or add a new one. |
| **Clear Search Results** | Click to clear the results of the search. |
| **Current Search** | Displays the criteria for the current search. |
| **Results Tree** | The results of the current search. |

# Import Project Data Dialog

The **Import Project Data** dialog is displayed when you select the **Import Project Data** option from the **Manage Project** dropdown menu on the **Summary** tab for a project. This dialog enables you to import data from another Code Insight project or source into the current Code Insight project (called the *target project*) from which you are invoking the import. The project data to be imported must be in a properly formatted and archived JSON file, called the *import data file* (such as the output of a Code Insight project data export).

Complete details on import process—how to prepare for it, how to run the import, and what results to expect—are provided in the Exporting and Importing Project Data chapter.

An additional setting, not shown on the **Import Project Data** dialog but used during the import process, is defined at the project level. The setting (**On data import or scan, delete inventory...**) determines whether the import should create “empty” inventory on the target—that is, inventory in the import data file that either has *no* associated files or *has* associated files that have no match in the target project codebase. For a description of this setting and how to change its value, if necessary, before running the import, see Edit Project: General Tab.

The following table describes the import options available on the **Import Project Data** dialog.

**Table 8-32** ▪ Import Project Data Dialog

| Column/Field | Description |
|---|---|
| **Choose File to Import** | Click **Browse** next to the **Choose File to Import** field to search for and select the `.zip` file containing the JSON project data you are importing. |

**Table 8-32 ▪** Import Project Data Dialog (cont.)

| Column/Field | Description | |
|---|---|---|
| **Add Files to Inventory** | Select **Yes** to enable the import to create new file associations with target inventory. The **File Matching Criteria** field is displayed (if it is not already).<br>Select **No** to disable the creation of any new file associations with target inventory during the import process. | |
| | **File Matching Criteria** | Select the option that identifies the criteria (file path, file MD5, or both) used to determine whether a file associated with an inventory item in the import data file has a match in the target project codebase. Only those files in the import data file that have matches in the target project codebase can be associated with target inventory.<br>For a description of the criteria options, see About the File-Matching Criteria for the Import. |
| | **...to *n* directories above the file** | If you have selected **Check only the file partial path** or **Check the file MD5 and the partial path** in the **File Matching Criteria** field, provide the directory depth that defines the partial path. That is, specify the number of directories *above the file name* that must be the same before two paths are considered as matches.<br>For example, a value of 2 indicates that file paths must match 2 directories above the file name. Suppose a file in the import data file has the path `/ePortal-1.3/copy1/src/gettext.c` and a file in the target codebase has the path `/ePortal-2.0/copy1/src/gettext.c`. With a directory depth of 2, only `/copy1/src/gettext.c` needs to match in the two file paths to meet the partial-file criterion.<br>You can enter a directory-depth value between 1 and 20, inclusively.<br>For more information, see About the File-Matching Criteria for the Import. |

Table 8-32 ▪ Import Project Data Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Mark Files as Reviewed** | Select **Yes** to enable the import to flag files with the “Reviewed” status in the target project codebase if they do not already have this status. The **File Matching Criteria** field is displayed (if it is not already).<br>Select **No** to disable the import’s ability to mark files as reviewed during the import process.<br>***Note*** *▪ For this option, the import compares only those files in the import data file that are marked as reviewed with files in the target codebase.* |
| **File Matching Criteria** | Select the option that identifies the criteria (file path, file MD5, or both) used to determine whether a file in the import data file has a match in the target project codebase. Only those files in the import data file that have matches in the target project codebase can be marked as reviewed.<br>For a description of the criteria options, see About the File-Matching Criteria for the Import. |
| **...to *n* directories above the file** | If you have selected **Check only the file partial path** or **Check the file MD5 and the partial path** in the **File Matching Criteria** field, provide the directory depth that defines the partial path. That is, specify the number of directories *above the file name* that must be the same before two paths are considered as matches.<br>For example, a value of 2 indicates that file paths must match 2 directories above the file name. Suppose a file in the import data file has the path `/ePortal-1.3/copy1/src/gettext.c` and a file in the target codebase has the path `/ePortal-2.0/copy1/src/gettext.c`. With a directory depth of 2, only `/copy1/src/gettext.c` needs to match in the two file paths to meet the partial-file criterion.<br>You can enter a directory-depth value between 1 and 20, inclusively.<br>For more information, see About the File-Matching Criteria for the Import. |

**Table 8-32 ▪** Import Project Data Dialog (cont.)

<table>
<tr><th>Column/Field</th><th colspan="2">Description</th></tr>
<tr><td rowspan="3"><b>Inventory Notes Handling</b></td><td colspan="2">Select the appropriate option to determine how the import process handles content in the notes fields between identical inventory items in the import data file and the target project. The inventory notes fields to which this option applies include the following:<br>● <b>Notices Text</b><br>● <b>Audit Notes</b><br>● <b>Usage Guidance</b><br>● <b>Remediation Notes</b><br><i><b>Note ▪</b> Inventory items are considered identical if they are associated with the same component-version-license combination, called CVL, as stored in the Code Insight data library.</i></td></tr>
<tr><td><b>Overwrite existing notes with imported notes</b></td><td>(Default) Select this option to overwrite the notes fields in the target inventory item with content from the identical inventory item in the import data file. However, if a given notes field is blank in the import data file, any existing content for that same field in the target inventory is retained.</td></tr>
<tr><td><b>Append imported notes to existing notes</b></td><td>Select this option to append data from the notes fields for a given inventory item in the import data file to those same fields in the identical target inventory item. When content is appended in a given notes field in the target inventory item, it is separated from the existing content with a line break and this heading:<br>`Copied during import from <ProjectName>:<InventoryName> (TimeStamp)`<br>However, if the content in a given notes field is the same for the inventory item in both the import data file and the target inventory, no content is appended in this field in the target inventory item.</td></tr>
<tr><td rowspan="3"><b>Inventory Usage Handling</b></td><td colspan="2">Select either of these options to define how the import should handle the Usage attributes for imported inventory items. For a description of the inventory Usage fields, refer to Usage tab in the <b>Project Inventory Details Pane</b> topic.</td></tr>
<tr><td><b>Reset usage field values to system default</b></td><td>Do not copy existing Usage values for inventory items from the source project to the target project. Instead, in the target project, reset all Usage fields for imported inventory to the system default value: <b>Unknown</b>. (Default)</td></tr>
<tr><td><b>Copy existing usage field values</b></td><td>Copy the existing Usage values for inventory items from the source project to the target project.</td></tr>
</table>

# About the File-Matching Criteria for the Import

When configuring the import to create new file associations in target inventory or to mark files in the target project codebase as reviewed, you must define the file-matching criteria needed by the import to compare files in the import data file with the target codebase files. Target files that match files in the import data file are eligible for either of these import functions.

The file path and the file MD5 value are the key criteria used to locate target codebase files that match files in the import data file. When the MD5 value is used as a criterion, the MD5 for a file in the import data file must have an exact MD5 match in the target codebase. However, when the file path is used as a criterion, the file-matching process can apply various rules.

For more information, see the following sections:

- About File-Path Processing During the Import
- Available File-Matching Criteria

## About File-Path Processing During the Import

When the file path is used as a criterion for locating a matching file, the import process internally subtracts the scan root path from the absolute path of codebase files in the import data file and in the target project. The result is the *complete file* path for a given file, as illustrated in these examples:

- **Absolute file path**—`/home/fnci/scanRoot/1/ePortal-1.3/src/gettext.c`
- **Root path**—`/home/fnci/scanRoot/1/`
- **Complete File path**—`/ePortal-1.3/src/gettext.c`

Then, based on the file-path criterion selected by the user, the import locates matching files by searching complete file paths, partial paths, or simply file names. The following examples illustrate a complete path in comparison with a partial path or file name:

- **Complete path**—`/ePortal-1.3/copy1/src/gettext.c`
- **Partial path**—`/copy1/src/gettext.c` **or** `/src/gettext.c`
- **File name**—`gettext.c`

## Available File-Matching Criteria

The following describes the available options for the **File Matching Criteria** field used by the import to locate file matches between the import data file and the target project codebase.

**Table 8-33 ▪** Options to Define File-Matching Criteria

| Option | Description |
|---|---|
| **Check only the file MD5** | A file’s MD5 value in the import data file must match an MD5 in the target project codebase. |
| **Check only the file name** | The name of the file in the import data file must match a file name in the target project codebase. (No path is compared in the file-matching process.) |

**Table 8-33 ▪** Options to Define File-Matching Criteria (cont.)

| Option | Description |
|---|---|
| **Check only the complete path** | The complete path of a file (including the file name) in the import data file must match a complete file path in the target project codebase. |
| **Check only the partial path** | A file’s partial path (including of the file name) in the import data file must match the partial path of a file in the target project codebase.<br>For this criterion, you must also specify the directory depth of the partial path. See the **...to *n* directories above the file** field description in the previous table, Import Project Data Dialog.<br>The partial-path depth enables the import to match files when the codebase location is different between the target project codebase and the project codebase whose scanned results are included in the import data file. |
| **Check the file MD5 and file name** | The MD5 and name of a file in the import data file must match the MD5 and name of a file in the target project codebase. |
| **Check the file MD5 and the complete file path** | A file’s MD5 value and complete path (including the file name) in the import data file must match the MD5 and complete path of a file in the target project codebase. |
| **Check the file MD5 and the partial file path** | A file’s MD5 value and partial path (including the file name) in the import data file must match the MD5 and partial path of a file in the target project codebase.<br>For this criterion, you must also specify the directory depth of the partial path. See the **...to *n* directories above the file** field description in the previous table, Import Project Data Dialog.<br>The partial-path depth enables the import to match files when the codebase location is different between the target project codebase and the project codebase whose scanned results are included in the import data file. |

# Inventory Details Tab in the Analysis Workbench

The **Inventory Details** tab in the **Analysis Workbench** contains a sub-tab for each inventory item you have opened from the **Inventory Items** pane. Each sub-tab contains the following fields describing a given inventory item:

**Table 8-34 ▪** Inventory Details Tab

| Category | Column/Field | Description |
|---|---|---|
| **Header information** | The **Inventory Details** tab header shows buttons that enable you take actions on the inventory item and lists attributes about the item and its associated component. | |
| | **Recall** | Click to recall (remove) a published inventory item from **Inventory Items** list if it does not fit the criteria for inclusion. The selected items are removed from the **Project Inventory** view and are only visible in the **Analysis Workbench**. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **View History** | Click open the Inventory History Window, which shows a list of all updates made to the inventory item up to the current date and provides details for each update. |
| | **Create Custom Rule** | (Available when inventory **Type** is **Component**) Click to open the **Custom Detection Rule** dialog to define an new detection rule for codebase files that are associated with a third-party component but not associated with inventory. For details, see Managing Custom Detection Rules. |
| | **Save** | Click to save any changes you have made to the inventory details. |
| | **Close** | Click to close the **Inventory Details** pane without saving changes. You are asked to save changes before the actual closure. |
| | **Review Status** | The status of the inventory item:<br>● **Approved**—The item is approved for use in the software project.<br>● **Not Reviewed**—The item has not been automatically reviewed by policy (therefore requires a manual review).<br>● **Draft**—This item is in the process of being reviewed.<br>● **Rejected**—The item is not approved for use in the software project. Instead, the item needs further review and remediation before being used in the software project. |
| | **Alerts** | Notifies you whether or not security alerts exist for this item. If alerts exist, click the ***x*** **Open Alerts** or ***x*** **Closed Alerts** link to view their details. If no alerts exist, **None** is displayed. You can access the **Alerts** dialog from this pane to change the status or priority of an alert. For more information, see Managing Security Vulnerability Alerts. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Priority** | A dropdown list showing the priority level given to this inventory item by the system, with P1 as the highest priority and P4 as the lowest.<br>You can change the priority for this inventory item by selecting a different priority from the dropdown list and clicking **Save**. For more information about priorities, see Inventory Priority. |
| | **Vulnerabilities** | A bar graph showing the count of known vulnerabilities by severity color for the inventory item. Click the graph to view the list of vulnerabilities and their details. For details about the graph and vulnerabilities in general, see Security Vulnerabilities Associated with Inventory.<br>The counts in this graph do not include vulnerabilities that are currently suppressed. If no vulnerabilities have been found for the inventory item, the value **No** is displayed in place of the graph. Additionally, if the **Type** value for the inventory item is **Work in Progress** or **License Only**, the value **N/A** is displayed. |
| | **Created By** | The name of the person or process that created the inventory item. |
| | **Confidence** | A simple three-segment graph representing the Confidence level (High, Medium, or Low) of the inventory item. The Confidence level is the measure of the strength of the discovery technique used to generate the inventory item. The graph shows three shaded segments for High confidence, two for Medium, and one for Low.<br>For more information about the Confidence levels, see Inventory Confidence in the “Using Code Insight” chapter. |
| | **Created On** | The date that the inventory item was created. |
| | **Updated On** | The date that the inventory item was updated. If the item has not been updated since the creation date, the date shown here will be the same as the Created On date. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Inventory details** | The following attributes describe the inventory item. You can update these attributes as needed from this pane. For details, see Editing Inventory from the Analysis Workbench or Creating an Inventory Item from the Analysis Workbench. | |
| | **Name** | The name of the inventory item. |
| | **Type** | The type of finding of this item:<br>● **Work in Progress**—A set of files with something in common. The work in progress will become a component or license only via manual audit work.<br>● **Component**—Files from a specific component version with known or unknown license. If this type is selected, the **Lookup Component** button becomes active, enabling you to select a new component instance for the inventory item.<br>● **License Only**—Files under a specific license without a known component. |
| | **Component** | The name of the component. Click ⓘ to view publicly available information about the component. See Component Details Window.<br>Click ✎ to select a new version (or license) for the component. |
| | **License** | The name of the license associated with this component. Click ⓘ to view additional information about the license. See License Details Window.<br>Click ✎ to select a new license (or version) for the component. |
| | **Description** | A description of the inventory item. You can update the description as needed. |
| | **URL** | The URL of the license for this inventory item. You can update the URL as needed. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

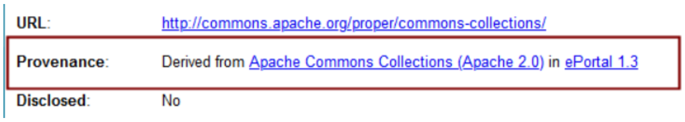
| Category | Column/Field | Description |
|---|---|---|
| | **Provenance** | The source project from which the current inventory item was derived.<br>***Note ▪*** *You cannot update this property from the Code Insight Web UI in general, but you can edit it when creating or updating inventory using the Inventory REST API.*<br>If the inventory item is not derived from another project, the value **Originated in this project** is displayed.<br>However, if the inventory item is derived from another project (for example, the inventory item was imported to the current project), the origin of the inventory is displayed with the inventory name and project name:<br><br>URL: http://commons.apache.org/proper/commons-collections/<br>Provenance: Derived from Apache Commons Collections (Apache 2.0) in ePortal 1.3<br>Disclosed: No<br>If the source project and inventory item still exist, this value is hyperlinked so that you can open the source project directly to the **Project Inventory** tab, with focus on the **Inventory Details** page for the original inventory item. This direct link enables you to explore the auditing and review details of the original inventory item to determine inventory history—for example, the reason the item was previously approved or rejected. If the source inventory item or its project no longer exists, no link to the original inventory item is provided. |
| | **Disclosed** | The **Yes** or **No** option indicating whether the third-party component or artifact represented by the inventory item known third-party dependency in your code before it was discovered by the scan or you.<br>This field is used most often by analysts to denote information about the state of the inventory item. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Workflow URL** | The URL (or a text reference such as a Jira issue number) that points to the request data pertaining to this inventory item as found in your site's external workflow system.<br>When you view this value on the **Inventory Details** tab in **Project Inventory**, the URL displays as a link (labeled as **View Associated Request**), enabling the reviewer to easily access to the workflow data that tracks the status of open tasks for the inventory item.<br>A text reference entered here is not converted to a link on the **Inventory Details** tab, but it still provides direction in locating the appropriate data in the workflow system.<br>The value is **None** if you enter no URL or reference.<br>Additionally, when you view the **Inventory Details** tab in **Project Inventory**, an ⓘ icon will be displayed next to the URL if additional request-related details are available for the inventory item. The reviewer can then click the icon for a quick review of pertinent details about the request without having to access the workflow system. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Notices Text tab** | The **Notices Text** tab is used to finalize the exact content to include in the Notices report. You can edit the notices content as needed from this pane when editing an existing inventory item or creating a new one. For more information, see Finalizing the Notices Text for the Notices Report. | |
| | **As-Found License Text** | The **As-Found License** `Text` field shows the license text or license references found in the scanned codebase. You cannot edit this field, but you can click **Copy to Notices Text** to copy the text to the **Notices Text** field. If content already exists in the **Notices Text** field, you can choose either to append the **As-Found License Text** content to the existing notices content or to replace the existing notices content. |
| | **Notices Text** | The exact content to include in the Notices report. You can edit any license text previously saved to this field or add your own license text, such as license information for rules that you developed during your manual research on the inventory item. You can also copy the **As-Found License Text** content to the **Notices Text** field and modify it as needed. Or you can leave this field empty.<br>If you provide information in this field, the Notices report pulls the content of only this field into the report. If this field is empty, the content of the **As-Found License Text** field is used in the report. If both fields are empty, the report uses the license content from Code Insight data library (see License Details from the Code Insight Data Library).<br>For more information, see Finalizing the Notices Text for the Notices Report. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Notes tab** | The **Notes** tab provides information about the automated and manual analysis of codebase as it relates to an inventory item. | |
| | **Detection Notes** | System notes that can specify the following:<br>● The automated detection technique that was used to locate the component<br>● License information in the case that the license has changed from one version to another or if the component has multiple licenses<br>● Attributes extracted from a POM or manifest file containing project and configuration details |
| | **Audit Notes** | Any notes added to the inventory item by the auditor or reviewer, based on findings during the analysis. You can edit these notes as needed from this pane when editing an existing inventory item or creating a new one. See Viewing and Updating Detection and Auditing Notes in the Analysis Workbench. |

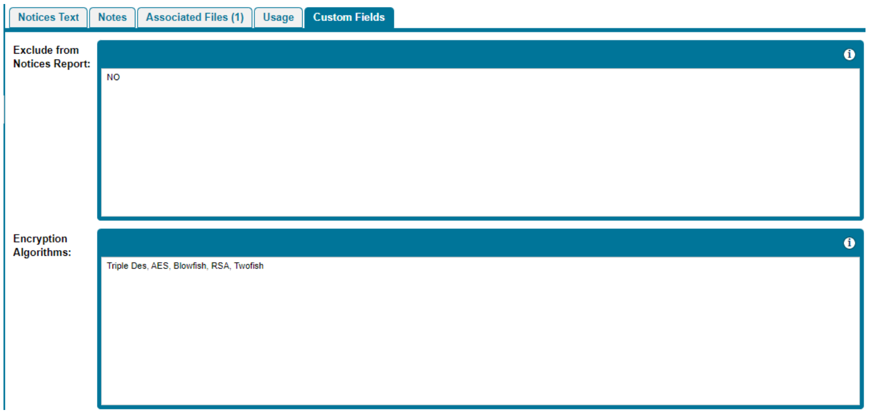
**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Associated Files tab** | Click this tab to view a list of the files that are part of the inventory for this project. Each file entry shows the following:<br>● **Action**—Icons that you can click to perform certain actions on the file. Currently, only the ✕ icon shows, enabling you to disassociate the file from the inventory item.<br>● **Alias**—The unique user-defined alias that was defined for the scanner (Scan Server or remote scan agent) to represent its scan-root path containing the codebase in which the file is located. The alias provides a name that is more meaningful than the scan-root path. (The actual absolute scan-root path for each scanner associated with the project is available on the project's **Summary** tab.)<br>● **File Path**—The file's path relative to the scan-root path on instance hosting the scanner. You can click the file-path link to open the **File Details** tab for that file.<br>● **Evidence**—The color-coded icons representing the types of open-source or third-party evidence found in the file (see Using the Filter Legend Options to Filter the Codebase for a description of the icons). A check mark indicates that the file has been reviewed.<br>***Note ▪** You cannot sort the file list.*<br>Optionally, you can right-click a file entry for options that enable you to perform additional operations on the file, such as marking it as reviewed, reverting its reviewed status to unreviewed, and other operations. See Managing the Codebase Files for details about these same options that are also available from the **Codebase Files** and **File Search Results** panes in the **Analysis Workbench**.<br>To add associated files to this list, see Adding Files to Inventory From the Codebase List. | |
| **Usage tab** | The **Usage** tab provides details on how your product uses the OSS or third-party software. You can update this information as needed from this pane when editing an existing inventory item or creating a new one. See Viewing or Editing Inventory Usage Information from the Analysis Workbench. | |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Distribution Type** | The option indicating how you are distributing the OSS or third-party component associated with the inventory item. The distribution type can affect license priority and obligations:<br>● **Internal**—The component is distributed internally only (for example, as an internal test framework included in the codebase but not distributed publicly with the software package).<br>● **External**—The component is a separate entity from your software package. It might be shipped as a separate component along with the software package or deployed through some method, such as a private cloud at the customer site.<br>● **Hosted**—The component is hosted in your company's data center (for example, as a SAAS application)<br>● **Unknown**—The distribution type is unknown. |
| | **Part of Product** | The option indicating whether the item is part of the core product or an infrastructure piece such as a build or test tool. This can affect whether third-party notices are required for this item. The value can be **Yes**, **No**, or **Unknown**. |
| | **Linking** | The option identifying how your software package links to the OSS or third-party component libraries. This method can affect license priority and obligations.<br>● **Not linked**—The software package uses no links to the component libraries.<br>● **Statically linked**—The component libraries are included in the software materials and thus linked statically.<br>● **Dynamically linked**—The component libraries are brought in at runtime.<br>● **Unknown**—The type of linking is unknown. |
| | **Modified** | The option indicating whether code from the OSS or third-party package has been modified for use by your organization. The value can be **Yes**, **No**, or **Unknown**. |
| | **Encryption** | The option indicating whether the component provides the encryption capabilities used in the product. Encryption can affect export controls. The value can be **Yes**, **No**, or **Unknown**. |

**Table 8-34 ▪** Inventory Details Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Custom Fields tab** | | The **Custom Fields** tab displays fields that were defined specifically for your site to provide information that standard Code Insight fields on the **Inventory Details** tab do not capture about the inventory.<br><br> Notices Text<br>Notes<br>Associated Files (1)<br>Usage<br>Custom Fields<br>Exclude from Notices Report: NO<br>Encryption Algorithms: Triple Des, AES, Blowfish, RSA, Twofish<br>If no custom fields have been defined, the tab displays the message “There are no custom fields configured”.<br>Use the following guidelines for entering (or editing) a value in a custom inventory field:<br>- If available, click the ⓘ icon in the upper right corner of a field to obtain help on completing the field.<br>- You can enter a value up to 64k (64000 characters) in size.<br>- To save the value, click the **Save** button next to **Create Custom Rule** button (in the **Inventory Details** tab header. |

# Inventory History Window

Users can view the history of updates made to a specific inventory item in project by clicking the **View History** button on either the Inventory Details Tab in the Analysis Workbench or the Project Inventory Details Pane. The **Inventory History** window is opened, listing the updates in a grid, each row representing a specific update made to the inventory item. The updates are grouped by revision IDs, each representing a *revision session*—that is, group of updates that were saved together at a specific point in time.

Refer to the following topics for procedures related to this window:

- Viewing the Update History for an Inventory Item in the Analysis Workbench
- Viewing the Update History for an Inventory Item in Project Inventory

The following describes the information recorded for each update.

**Table 8-35 ▪** Attributes of Each Update Listed in Inventory History

| Field/Column | Description | |
|---|---|---|
| **Revision header** | Inventory updates are grouped chronologically by the revision session in which they were made, each session with its own header. | |
| | **Revision ID** | Identifies a single revision session consisting of one or more updates that were saved together at a specific point in time. |
| | **Revision details** | The date and time at which the revision was saved and the number of updates made during the revision session. |
| **Details of update** | The following details describe each update made to the inventory item. | |
| | **Date** | The date and time at which the update was saved.<br>You can sort on this column. By default, the updates are listed in descending order by date so that users view the most recent updates first. |
| | **Event** | The type of update: **Inventory Created**, **Inventory Updated**, **Inventory Published**, **Inventory Recalled**, or **Inventory Reviewed**. |
| | **Action** | The user action that resulted in the update:<br>● **Scan**—A Code Insight project scan.<br>● **Project Import**—A Code Insight project import. This can be either a standard project import or the import performed during a project-branching process.<br>● **Manual Analysis**—A manual update made to the inventory item through the Code Insight Web UI or a REST API.<br>● **Policy**—An update to the policy profile that impacted the status of the inventory item.<br>● **hyphen (-)**—The value that defaults when the inventory was migrated from a pre-2021 R3 version of Code Insight. |
| | **User** | The name of the user who performed the update. The name is hyperlinked to open an email draft addressed to this user.<br>If the system performed the update, the value **System** is displayed with no hyperlinked text. |
| | **Field** | The name of the inventory field that was updated. |

**Table 8-35 ▪** Attributes of Each Update Listed in Inventory History (cont.)

| Field/Column | | Description |
|---|---|---|
| | **Old Value** | The previous value of the inventory field. Note the following:<br>● The value for the **URL** field is hyperlinked and, when clicked, opens to the linked site on a new browser tab.<br>● If the value for the **Description** field or a notes field (such as **Audit Notes**, **Notices Text**, and others) exceeds 150 characters, a **Show more...** link is displayed to expand the row to show the entire text. You can then click **Show less...** to collapse the text.<br>● When you click the ⓘ icon next a **Component** or **License** value, a pop-up dialog shows read-only details about that item. |
| | **New Value** | The value of the inventory field after the change. Note the following:<br>● The value for the **URL** field is hyperlinked and, when clicked, opens to the linked site on a new browser tab.<br>● If the value for the **Description** field or a notes field (such as **Audit Notes**, **Notices Text**, and others) exceeds 150 characters, a **Show more...** link is displayed to expand the row to show the entire text. You can then click **Show less...** to collapse the text.<br>When you click the ⓘ icon next a **Component** or **License** value, a pop-up dialog shows read-only details about that item. |
| **Actions** | The following buttons and icons enable you to navigate and manage the history view. | |
| | ↻ | Refresh the history view. |
| | **Page controls** | Move to the next or previous page or to the first or last page in the view; or enter a specific page number in the **Page** field.<br>Note that the default page size is 25 revision records. (Page size is based on the number of revisions, not updates.) |
| | **Close** | Exit the **Inventory History** window. |

# Inventory View

Code Insight enables you to view published inventory of open-source (OSS) or third-party components found across the projects in your Code Insight system. This inventory, displayed in a single scrollable window called the **Inventory** view, provides the means to make overall assessments of the OSS or third-party code used in your company's software deliverables.

**Table 8-36 ▪** Inventory View

| Category | Column/Field | Description |
|---|---|---|
| **Search and filter fields and buttons** | Use these fields and buttons (which display at the top of the **Inventory** view) to filter and modify the inventory list in the view. For your reference, the total number of filtered inventory items currently displayed compared to the total number of items in the full **Inventory** view is tracked in the **Inventory** view header:<br> <br>Inventory Items (Search Results: 47 of 31,794) | |
| | **Enter Inventory Name** | Use this field to filter the inventory list by inventory name. Enter a string by which to filter the inventory names.<br>If necessary, click the search icon next to the field to initiate search.<br>To remove the string and restore the full list of inventory items, click the **X** in the field. |
| | **Advanced Search button** | Click this button to open the **Advanced Inventory Search** dialog. From this dialog, you can set search criteria (based on inventory attributes) by which to filter the inventory list. For details about the criteria available on this dialog, see Advanced Inventory Search Dialog. |

**Table 8-36 ▪** Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Context for the view (dropdown)** | From this dropdown, select the major context for the **Inventory** view:<br>● **My Projects**— Show all published inventory across those Code Insight projects in which you are assigned a role. You might use this context to show areas where you need to provide review or remedial work, or you might want to review the overall state of inventory found in your projects. (This is the context enabled by default when you open the **Inventory** view.)<br>● **All Projects**—Show all published inventory across all projects in your Code Insight system. This context is helpful in visualizing trends in your company's use of open- source and third-party code in its software projects.<br>● **Select Project**—Show all published inventory for a selected Code Insight project. Since projects represent versions of a particular software product, this view allows you to see all inventory items for that product. Furthermore, you can also opt to list the inventory for all child projects of the selected project. These child projects represent modules used by your top-level product. You can directly link to the inventory item of the child project or to the child project itself. You can also view the parent hierarchy of the child project to understand the provenance of the inventory items. (See Including the Inventory of Child Projects on the Inventory View for details.)<br>If you choose this option, a dialog is displayed from which to select a project. Once you choose the project, the inventory list is refreshed with the inventory for that project only. |
| | **Change Project** | (Displayed once a specific project is selected for **Select Project** in the previous field) Click this button to select a different project whose inventory you want to display in the **Inventory** view. |

**Table 8-36 ▪** Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Show All Items button** | Click this button to remove all current criteria configured on the **Advanced Inventory Search** dialog and switch the focus of the **Inventory** view to show all projects.<br>***Note ▪*** *This button does not display if the **Inventory** view is already using the **All Projects** focus.* |
| | **Include inventory items from child projects** | If child projects have been identified for project currently in context in the **Inventory** view, select this option to refresh the view to include the inventory for these child projects. In this way, you can examine the inventory found across the project codebases for all parts of your software project, including its dependencies and sub-modules For more information, see Including the Inventory of Child Projects on the Inventory View.<br>Note that by selecting to include inventory from child projects, all child-projects associated recursively to the current top-level project will be included in your inventory items list. Each child project is identified by the icon next to its name in the list. |

**Table 8-36 ▪** Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Inventory columns** | The following columns identify and provide information about each inventory item listed in the **Inventory** view.<br>To manage column content, hover over the right side of a specific column header, and click its dropdown menu. From this menu, you can re-sort column values in ascending or descending order, as well as display or hide any column in the **Inventory** view. (By default, the **#Files** column is hidden.)<br>***Note ▪** Currently you can re-sort the values in the **Project**, **Inventory Name**, **Priority**, **#Files**, **Status**, and **Created On** columns.*<br>To open a read-only version of the details for the given inventory item, click anywhere in the row for the item (except on linked text or a linked icon). A slide-out is displayed, showing most of the details that are also available for the item on its **Project Inventory Details** pane in the actual project. However, unlike the **Project Inventory Details** pane, the values on the slide-out are not editable. (While these values are read-only, certain ones are hyperlinked, enabling you to still explore and maintain the inventory item if you want.) For more information, see Opening a Read-Only Version of Inventory Details on the Inventory View. For a description of the inventory details available on the slide-out, see Project Inventory Details Pane.<br>Otherwise, you can use links directly on the **Inventory** view to open an inventory item's associated project to examine the item within the context of its actual project and to edit its details as your permissions allow. See the Project and Inventory Name column descriptions. | |

**Table 8-36 ▪** Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Project** | The name of the Code Insight project to which the given inventory item belongs.<br>If a project is a child project of the current project, the icon displays next to the child project name. Click this icon to view the recursive hierarchy of the child project's parents.<br>To open the project to its **Project Inventory** tab, click the hyperlinked project name. From here, you can explore and edit all published inventory (including the given inventory item) as your permissions allow. For more information, see Open the Associated Project to the List of All Inventory in the Project. |
| | **Inventory Name** | The name of the inventory item in *component version (license)* format.<br>To open the project to which the given inventory item belongs, click the hyperlinked inventory name. The project opens to the **Project Inventory Details** pane on the **Project Inventory** tab, providing access to all information available for the given inventory item within the project. From here you can explore and edit this inventory item as your permissions allow. For more information, see Open the Associated Project Directly to the Details for a Given Inventory Item.<br>Alternatively, instead of opening the project, you can view a read-only version of the inventory details within the **Inventory** view. See the Inventory columns description. |
| | **Priority** | The inventory priority of the item (**P1**, **P2**, **P3**, or **P4**). For more information about this attribute, see Inventory Priority |
| | **Component** | The name and version of the open-source or third-party component on which the inventory item is based. For more information about the component, click ⓘ to open the **Component Details** window. This window shows publicly available details for the component as found in Code Insight's data library of third-party and OSS component information.<br>When a component is not known, **N/A** is displayed. |

**Table 8-36 ▪** Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **License** | The license associated with the open-source or third-party component. For more information about the license, click ⓘ to open the **License Details** window. See License Details Window for a description of the available details.<br>When a license is not known, the value **I don’t know** is displayed. |
| | **Vulnerabilities** | A bar graph showing the count of known security vulnerabilities by severity color for the inventory item. Click the graph to view a list of these vulnerabilities and their CVSS details. For more information about security vulnerabilities, see Working with Security Vulnerabilities.<br>The counts in this graph do not include vulnerabilities that are currently suppressed. If the inventory item has no known vulnerabilities, **None** is displayed. |
| | **Tasks** | Access to the open tasks for the inventory item:<br>● If open tasks exist for the inventory item, the ☰ icon is displayed. Click this icon to open a **Tasks** window, listing the open tasks specific to the inventory item. From here, you can view or edit details for each open task, close the task, or create new tasks for the inventory item if needed.<br>● If no open tasks exist for the inventory item, no icon is displayed. |
| | **Alerts** | Access to any security alerts for the inventory item. An alert are is generated if the Electronic Update detects a new security vulnerability for the inventory item since the last scan.<br>● If alerts exist for the inventory item, the ⚠ icon is displayed. Click this icon to open an **Alerts** window, listing the new security vulnerabilities and their CVSS information. From here, you can change the priority or status of the alert. See Managing Security Vulnerability Alerts for details.<br>● If no alerts exist for the inventory item, no icon is displayed. |
| | **# Files** | The number of codebase associated with the inventory item. |

**Table 8-36** ▪ Inventory View (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Status** | The status of the inventory item:<br>● ✓ **Approved**—Approved for inclusion in the final notices of open-source and third-party components (such as a Bill of Materials or a similar document).<br>● ⊗ **Rejected**—Rejected for inclusion in the final notices.<br>● ? **Ready for Review**—Not yet reviewed. |
| | **Created On** | The date on which the inventory item was created. |

# LDAP Tab

Code Insight supports user authentication and authorization through LDAP (Lightweight Directory Access Protocol). The **LDAP** tab on the **Administration** page configures the synchronization of user identification data from LDAP to Code Insight, thus enabling LDAP user authentication for Code Insight. For detailed information about the fields on this tab and about the configuration in general, see “Configuring Code Insight for LDAP” in the *Code Installation and Configuration Guide*.

The tab contains the following columns and fields:

**Table 8-37** ▪ LDAP Tab

| Category | Column/Field | Description |
|---|---|---|
| **LDAP enablement** | This option enables the use of LDAP for your Code Insight system. When LDAP is enabled, the settings used to configure Code Insight for LDAP are made available for editing on this tab. You can use this option to turn off LDAP whenever necessary. | |
| | **Enable LDAP** | Select **Yes** or **No** to determine if LDAP will be used for user authentication. The default is **No**. |

**Table 8-37 ▪** LDAP Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **LDAP Connection Details** | These settings configure the Code Insight connection to the LDAP server. This connection is required for each synchronization process of LDAP user information to Code Insight and for authentication each time a user logs into Code Insight. | |
| | **LDAP URL** | Specify the URL of the LDAP server in the following format:<br>`ldap://<ldap_server_host>:<ldap_port>`<br>where `<ldap_server_host>` is either the hostname or IP address of the LDAP server; and `<ldap_port>` is the port on which the server listens for requests.<br>The following is an example URL, which uses the standard LDAP server port 389:<br>`ldap://acme.com:389`<br>If using SSL to provide data encryption security for user information passed over the network, specify the `ldaps://` protocol with the port 636, which is the default dedicated port for SSL:<br>`ldaps://acme.com:636`<br>***Note ▪*** *When the LDAP directory service is Active Directory, requests from users located outside the global catalog's base domain will fail to authenticate if you use the port specified above. This occurs because requests sent to the default LDAP port 389 (or 636 if SSL is used) search for objects only within the global catalog's base domain. To authenticate users from outside the base domain, change the LDAP port to 3268 (or 3269 if SSL is used). Requests sent to this port search for objects in the entire forest.* |

**Table 8-37 ▪** LDAP Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Authentication Type** | Select the type of LDAP authentication used to establish a connection with the LDAP server:<br>● **Anonymous**—Code Insight will establish a connection with the LDAP server without the use of user credentials. (When this option is selected, the **LDAP Username** and **LDAP Password** fields in this section are disabled.) This authentication type is generally used for testing purposes.<br>● **Authenticated**—Code Insight requires the user credentials provided in the **LDAP Username** and **LDAP Password** fields to authenticate and establish a connection with the LDAP server. |
| | **LDAP Username** | Depending on your LDAP setup, enter either of the following to identify the user used connect to the LDAP server:<br>● The user's login ID, such as `mburns`<br>● The user's Distinguished Name (DN), such as:<br>`CN=Monty Burns,OU=usa,DC=acme,DC=com`<br>For more information about providing the DN, see "Distinguished Name for an Object" in the *Code Installation and Configuration Guide*.<br>This identification, along with the associated password (see the next field), is used to authenticate the connection to the LDAP server. Note that the user must have READ permissions to query the LDAP server (and therefore does not need to be an administrator).<br>This field is disabled if **Anonymous** is selected for **Authentication Type**. |
| | **LDAP Password** | Enter the password associated with the user specified for **LDAP Username**. This field is disabled if **Anonymous** is selected for **Authentication Typ**e. |

**Table 8-37 ▪** LDAP Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **LDAP Query Details** | The following fields define the query that identifies the subset of users on the LDAP server to be synchronized to Code Insight. This query is used for the initial synchronization process and for each subsequent synchronization performed per the **LDAP User Sync Frequency** value. | |
| | **LDAP Base** | Specify the Distinguished Name (DN) of the LDAP base domain in the Directory Information Tree (DIT) on your LDAP server. This domain is the top-level directory to which all other objects in the directory structure belong; it typically represents your organization. The base domain is identified by domain controller objects (DCs), which make up its DN. For example, the base domain in the example DIT in Figure 2-1 is the following:<br>`DC=acme,DC=com`<br>In some cases, a sub-domain can be a part of the base domain:<br>`DC=software,DC=acme,DC=com`<br>For more information, see “LDAP Base” in the *Code Installation and Configuration Guide*. |
| | **LDAP Search Base** | Specify the DIT directory, relative to the LDAP base directory, under which you store all Code Insight objects on the LDAP server and from which you search for Code Insight users.<br>In reference to the example DIT in Figure 2-1, if you enter `OU=usa` for the search base, all searches for user information will be performed below the directory “usa”. (LDAP internally identifies the DN for this directory as the **LDAP Base** + **LDAP Search Base** value.)<br>If you leave this field blank, the search is performed at the LDAP base level.<br>For more information, see “Setting Up a User Search” in the *Code Installation and Configuration Guide*. |
| | **LDAP Search Query** | Specify the search query used to retrieve the users from **LDAP Search Base** directory to synchronize to Code Insight. Each attribute in a query is listed in parenthesis in the format (*attribute*=*value*), such as in the following, which searches for only those users belonging to the “engineering” group under the “usa” node:<br>`(&(objectClass=person)(memberOf=CN=engg,OU=usa,DC=acme,DC=com))`<br>For other search query examples, see “Setting Up a User Search” in the *Code Installation and Configuration Guide*. |

**Table 8-37 ▪** LDAP Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Use Paging** | Select **Yes** if the LDAP server has paging enabled for synchronization results. If you select **Yes**, the **LDAP Page Size** field is enabled, enabling you to customize the page size.<br>Select **No** if the server does not have paging enabled. If you select **No**, the server sends 1000 elements per page by default unless this behavior is changed at the organization level on the LDAP server. |
| | **LDAP Page Size** | Indicate the page size you want for the synchronization results. The default page size is 1000 elements. |
| | **LDAP User Sync Frequency** | Specify the frequency at which Code Insight will synchronize user data with the LDAP server:<br>● **Never**—Select this option to disable the automatic user synchronization. A synchronization occurs only if the user clicks the **Sync Now** button. For all other values, automatic user synchronization is enabled per the configured frequency. (This is the default value.)<br>● **Hourly**—Enter an integer value representing the number of hours between user synchronizations.<br>● **Daily**— Select a time at which the user synchronization will run every day.<br>● **Weekly**—Select a day of the week and a time of the day when the user synchronization will run each week. |
| | **Search Sub-tree** | Select this checkbox to enable deep searches through the subtree of the path defined by **LDAP Base** + **LDAP Search Base**. Note that, while helpful in locating users in certain cases, a deep search can negatively affect performance (and therefore, by default, is not enabled). For more information, see “Setting Up a User Search” in the *Code Installation and Configuration Guide*. |

**Table 8-37 ▪** LDAP Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **LDAP User Property Mappings** | The following information maps LDAP attribute labels to their corresponding labels in Code Insight (the field names shown below). These mappings are used for LDAP synchronization to Code Insight and for user authentication each time a user logs into Code Insight. | |
| | **Login** | Enter the user attribute label on your LDAP server corresponding to the user **Login** field in Code Insight. This is the same attribute that the user will use to log into Code Insight. |
| | **First Name** | Enter the user attribute label on your LDAP server corresponding to the user **First Name** field in Code Insight. |
| | **Last Name** | Enter the user attribute label on your LDAP server corresponding to the user **Last Name** field in Code Insight. |
| | **Email** | Enter the user attribute label on your LDAP server corresponding to the user **Email** field in Code Insight.<br>***Note ▪*** *Only those users with a valid email address specified as a user attribute on the LDAP server will be synchronized. Therefore, ensure that you have entered the correct label here for the email attribute on your LDAP server and that each user has valid email for this attribute on the server. See “Setting Up a User Search” in the “Code Installation and Configuration Guide” for more information.* |
| | **Login Filter** | Specify a filter for the user-login search performed in the LDAP search base location. For example, the value `(sAMAccountName={0})`, when used against the **LDAP Search Query** results, searches for each entry where the `sAMAccountName` is equal to the user login name. |

# License Details Window

The **License Details** window lets you view general information and license text for a selected license. The window has the following fields:

**Table 8-38 ▪** License Details Window

| Tab | | Description |
|---|---|---|
| **General Information tab** | The attributes of the current license. | |
| | **Id** | The identification number of the license in the data library. |
| | **Name** | The name of the license (for example, *Academic Free License v1.1*). |

**Table 8-38 ▪** License Details Window (cont.)

| Tab | | Description |
|---|---|---|
| | **Priority** | The priority ranking of the license as determined by Code Insight. For more information, see Auditing Scan Results in the Analysis Workbench. |
| | **URL** | The URL where the license is available on the Internet. |
| | **Description** | A short description of the license. |
| | **Family** | The immediate classification to which the license belongs (for example, the MIT license belongs to the family of MIT Style licenses). The family designation is helpful to a legal reviewer to understand the “type” of license prior to investing the time to analyze the complete license text. |
| | **Category** | The broader license category to which the license belongs. |
| | **Custom License** | The Yes or No indicator for whether this license is a custom license. |
| | **Commercial** | The Yes or No indicator for whether the license is classified as commercial. |
| | **Copyleft** | The Yes or No indicator for whether the license is considered a copyleft license. This type of license requires that all modified or extended portions that you have added to the free open-source code in your software also be free for others to use. |
| | **Free Software License** | The Yes or No indicator for whether the license is a free software license. |
| | **GPL V2 Compatible** | The Yes or No indicator for whether a license is compatible with GPL-2.0. |
| **License Text tab** | The complete text of the selected license as stored in the Code Insight data library. This text represents the external forge license text. | |

# Lookup Component Window

The **Lookup Component** window is displayed when you click **Lookup Component** within the context a inventory item, with the purpose of letting you search for a new component-version-license instance to associate with the inventory item. The search is performed against the Code Insight data library to locate components that meet your criteria. The search results in a list of components, each component displayed with a set of details and a list of its available version-license instances.

Once you locate the desired component, you can select the appropriate version-license combination to associate with your inventory item. Alternatively, you can create your own instance. (Any custom version-license instances created for a component are made available at the system level for association with inventory in other projects.) If no component meets your criteria for the inventory item, the **Lookup Component** window provides access to a feature that lets you create a custom component.

**Table 8-39 ▪** Lookup Component Window

| Category | | Column/Field | Description |
|---|---|---|---|
| **Search controls** | Use one of these fields to enter the criterion by which to search for a component or by which to create a custom component. | | |
| | **Search by** | Select the method by which to search component or to create a new component. | |
| | | **Keyword** | Select this option to search by a string in the component name or title. In the **Keywords** field, enter the string.<br>If you are creating a new component, the string is used to pre-populate certain fields in the **New Custom Component** window. See the **Create New Component** description. |
| | | **URL** | Select this option to search by the project or forge URL of the component. In the **URL** field, enter the URL.<br>If you are creating a new component, the URL is used to pre-populate certain fields in the **New Custom Component** window. See the **Create New Component** description. |
| | | **Forge** | Select this option, and then select the forge (project repository) by which to search components.<br>If you are creating a new component, the selected forge is used to pre-populate certain fields in the **New Custom Component** window. See the **Create New Component** description. |
| | | **Search** | Click this button obtain the search results. |
| | **Create New Component** | Click this button to open the **New Custom Component** window. Certain fields in this window are pre-populated with values based on the criterion you entered on the **Lookup Component** window. For information on creating a custom component, see Creating and Editing Custom Components. | |

**Table 8-39 ▪** Lookup Component Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Search results** | | The results of the search is a list of components, each component with a set of details (see **Component details**) and a list of available version-license instances to which you can associate with the current inventory item (see **Version-license instances**). The following describes the information shown for each component listed. |
| | **Component details** | The details for a given component can include the component's product logo, vendor content describing the component, and a link to the actual OSS or third-party product. It also includes the following component details from the Code Insight data library. |
| | **Component** | The name of the OSS or third-party component and its internal ID, as identified in the Code Insight data library. |
| | **Possible Licenses** | License candidates that can be associated with this component. |
| | **Custom Component** | The **Yes** or **No** value, indicating whether the component is custom (created by a user) or provided as part of the Code Insight data library. |
| | **CPE** | The list of CPE names—from the National Vulnerability Database—that are mapped to the component. CPE (Common Platform Enumeration) is a structured naming scheme that includes the component's vendor and product names in the following format:<br>**cpe://<part>:<vendor>:<product>**<br>where <part> is either **a** (applications), **h** (hardware platforms), or **o** (operating systems).<br>***Note ▪*** *The data provided represents only the part, vendor, and product; the version information is truncated from the CPE string.* |

Table 8-39 ▪ Lookup Component Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Version-license instances** | The information for each component includes a list of its available version-license instances. (To toggle between showing or hiding the list, click **Show Versions/ Instances** or **Hide Instances**.)<br>From this list, you can do any of the following:<br>● Select a given version-license instance to associate with the current inventory item.<br>● Register a new version-license instance for the component.<br>● If the component is custom, edit the component as needed.<br>A bar graph is included with each instance to show its current security-vulnerability counts by severity level (if any). See Security Vulnerabilities Associated with Inventory for details. |
| | **Use This Instance** | Click this button to associate the version-license instance with the inventory item you are currently creating or editing. You are directed back to the inventory item, now showing the new component-version-license association. |
| | **Register New Instance** | Click this button to add a new version-license instance to the component.<br>From the **Version** dropdown, select an existing version associated with this component (as stored in the Code Insight data library), or create your own version.<br>From the **License** dropdown, select an existing license associated with this component, or choose **Select Your Own License** to select or create a different license.<br>New instances are made available at the global level for use by inventory in other projects. |
| | **Edit Custom Component** | (Available if the component is custom) Click this button to open the **Edit Custom Component** window to update the component properties. For information on editing a custom component, see Creating and Editing Custom Components. |

# Policy Details Window

The **Policy Details** window is displayed when you select to create, edit, or view a policy profile from the Policy Page. A policy profile contains a set of policies used to perform an automatic review of inventory items upon their publication. Each policy defines criteria based on OSS or third-party component versions, licenses, or security vulnerabilities. Inventory items that meet any of the profile's policy criteria can be automatically approved or rejected (or flagged for a manual review). Any one policy that results in a rejection causes the inventory item to be rejected despite any approvals.

The following topics explain how to use the **Policy Details** window to define policies within a policy profile:

- Policy Fields
- Maintaining License Policies
- Adding Reviewer Content to Policies
- Impact on Policies When Code Insight’s CVSS Configuration Changes

Only users who have Policy Manager permissions can create, edit, or copy a policy profile. All other users can view the policy profile only.

**See Also**
Policy Page
License Details Window
Lookup Component Window
Managing Policy Profiles

# Policy Fields

The Policy Details Window provides the following fields to define policies that automatically approve or reject an inventory item when it is published. If no policy applies to an inventory item, the item’s status is **Not Reviewed**, requiring the item to be reviewed manually. Only users with Policy Manager permissions can edit these fields.

When you select to *view* a policy profile from the Policy Page, the following fields are read-only. Any user can view a profile, including those users who do not have Policy Manager permissions.

**Table 8-40 ▪** Policy Details Window

| Category | Column/Field | Description |
|---|---|---|
| **General** | These fields identify the policy profile you are creating or editing. | |
| | **Name** | The name of the policy profile that you are editing or copying.<br>If you are copying a profile, the name of the copy will be `Copy of` *`selected_policyProfile`*, where *`selected_policyProfile`* is the name of the original profile. To change the name of the profile copy, type over the generated name with the new name in this field. |
| | **Description** | The policy profile description, if it exists. You can edit or add a description. |
| | **Created** | (Available in the Edit and View versions of the profile) The name of the user who created the policy profile, and the date and time the profile was created. You can click the hyperlinked name to send an email to the user who created the profile. |
| | **Updated** | (Available in the Edit and View versions of the profile) The name of the user who last updated the policy profile, and the date and time the profile was updated. You can click the hyperlinked name to send an email to the user who updated the profile. |

**Table 8-40 ▪** Policy Details Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Vulnerabilities** | The following define policies that automatically approve or reject inventory items with security vulnerabilities.<br>***Note ▪*** *These policies ignore suppressed vulnerabilities when making decisions whether to automatically approve or reject published inventory items. (A change in policy due to the suppression of a vulnerability does not change the existing approval/rejection status of an inventory item unless the item is manually recalled and then republished.)* | |
| | 📄 | Click this icon next to a vulnerability policy to provide (or edit or view) meaningful content intended for inventory reviewers about the impact of the given policy. For example, this content might include reasons why the specific security vulnerabilities identified in the policy pose a risk to your intellectual property.<br>This information is propagated to those project inventory items that are actually rejected by the policy, providing reviewers with context about the inventory's status. For more information, see Adding Reviewer Content to Policies. |
| | **Only auto-approve inventory items if there are no associated security vulnerabilities** | Select this checkbox to have Code Insight skip any matching license-based or component policies if the inventory item has any associated security vulnerabilities. |
| | **Reject inventory items if any associated security vulnerabilities have a CVSS score above <score>** | Select this checkbox to have Code Insight automatically reject any inventory items with any associated security vulnerabilities that have a CVSS score above the value you enter. (The scores available for this field are based on the CVSS version currently used by Code Insight. For information, see Security Vulnerabilities Associated with Inventory.)<br>This policy takes precedence over any other automated approval policy.<br>***Note ▪*** *If the Code Insight System Administrator changes the CVSS version used by Code Insight, the value you selected for this field might change. See Impact on Policies When Code Insight's CVSS Configuration Changes for details.* |

Table 8-40 ▪ Policy Details Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Reject inventory items if any associated security vulnerabilities have a severity equal to or higher than <severity level>** | Select this checkbox to have Code Insight automatically reject any inventory items with any associated security vulnerabilities that have a severity equal to or higher than severity you select. (The severities available for this field are based on the CVSS version currently used by Code Insight. For information, see Security Vulnerabilities Associated with Inventory.)<br>This policy takes precedence over any other automated approval policy.<br>***Note*** *▪ If the Code Insight System Administrator changes the CVSS version used by Code Insight, the value you selected for this field might change. See Impact on Policies When Code Insight's CVSS Configuration Changes for details.* |
| **Licenses** | The following fields describe and manage the policies that automatically approve or reject inventory associated with a given license. | |
| | **Add License** | Click this button to add a new license policy based on a selected license and inventory usage criteria. (The **Edit** [or **Add**] **License and Usage Criteria** window is opened to enable you to do this.) See Maintaining License Policies for further details.<br>Once you create the license policy, its entry is added to the **Licenses** list. For the entry, you can then select the review status (under **Action**) that this policy automatically assigns an inventory item if the policy's criteria are met. |
| | 📄 | Click this icon to the left of each license policy to provide (or edit or view) meaningful content intended for inventory reviewers about this policy. For example, the content might list requirements for using the licenses identified in the policy's criteria or reasons why these licenses pose a legal risk.<br>This information is then propagated to those project inventory items that are actually approved or rejected by the policy, providing reviewers with context about the inventory's status. For more information, see Adding Reviewer Content to Policies. |
| | ⓘ | Click this icon to the left of each license policy to open the License Details Window, enabling you to view information about the license, including its attributes and license text. |

**Table 8-40 ▪** Policy Details Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Licenses (list)** | The list of license policies (in a grid format) currently used by this profile for automatically reviewing inventory items. Each license policy entry contains the license name, inventory usage criteria that can impact the obligations incurred by the use of the license, and actions you can perform on the policy.<br>● **Name**—The name of the license on which the policy is based.<br>The following read-only criteria are currently defined for the given license policy and describe how a software package developed in your organization uses the OSS or third-party component associated with an inventory item. (This usage can have an impact on your license obligations and conditions of use.) To define or edit these criteria for a license policy, see for Maintaining License Policies.<br>● **Distribution type**—The criterion specifying how the OSS or third-party component associated with the inventory item is distributed with your software package.<br>● **Linking**—The method that your software package uses to link to libraries in the OSS or third-party component associated with the inventory item.<br>● **Modified**—The criterion specifying whether code from the OSS or third-party package has been modified for use by your organization.<br>The following field specifies the review status automatically assigned to inventory items based on their meeting the criteria for this license policy:<br>● **Action**—Select one of the following to indicate what review status to automatically assign an inventory item that meets the criteria in this license policy:<br>○ **Approve**<br>○ **Reject**<br>○ **No Action** (same as the **Not Reviewed** status, thus requiring a manual review)<br>The following icons at the right of each license policy are used to manage the policy:<br>● ✕ **(delete)**—Click this icon to delete the license policy.<br>● ✎ **(edit)**—Click this icon to edit the license policy criteria. See Maintaining License Policies. |

Table 8-40 ▪ Policy Details Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Components** | The following fields define and manage policies that automatically approve or reject inventory based the component version associated with the inventory. | |
| | **Add Component** | Click this button to select a component on which to create the policy, or create a new component from the **Lookup Component** window. (See Lookup Component Window for information about how to use this window.) Once you select a component, its entry is added to the **Components** policy list. |
| | 📄 | Click this icon to the left of each component policy to provide (or edit or view) meaningful content intended for inventory reviewers about this component. For example, this content might include “need to know” information about why the component versions identified in the policy pose a risk.<br>This information is then propagated to those project inventory items that are actually approved or rejected by the policy, providing reviewers with context about the inventory’s status. For more information, see Adding Reviewer Content to Policies. |
| | ⓘ | Click this icon to the left of each component policy to open the Component Details Window window, enabling you to view relevant information about the component, including its forge, possible licenses, CPE names, and more. |

**Table 8-40 ▪** Policy Details Window (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Components (list)** | The list of current components and versions (in a grid format) currently used as criteria for automatically reviewing inventory items.<br>● **Name**—The name of the component.<br>● **Versions**—Select a specific version or a range of versions for the given component. (The **Versions from** and **to** drop-down lists are populated with available versions for the component.) Here are some example ways to specify a version or version range:<br>○ To enter a specific version, select the same version in the **Versions from** and **to** fields.<br>○ To enter an explicit range, select a minimum version in the **Versions from** field and the maximum version in the **to** field.<br>○ To specify any version for the given component, select the wild card ***** in both **Versions from** and **to** fields.<br>○ To specify any version up to a specific version, enter the wild card ***** in the **Version from** field and the maximum version in the **to** field.<br>○ To specify any version after a specific version, select the specific version in the **Versions from** field and the wild card ***** in the **to** field.<br>The **unknown** option applies to certain components that were collected without a version value. To specifically handle unknown versions, set both **Versions from** and **to** fields to **unknown**.<br>● **Action**—Select one of the following to indicate what status to automatically assign to inventory items associated with this component-version:<br>○ **Approve**<br>○ **Reject**<br>○ **No Action** (same as the **Not Reviewed** inventory status, thus requiring a manual review)<br>Click ✕ to the right of each component to delete the component from the policy. |
| **Actions** | These actions manage the entire policy profile. | |
| | **Save** | Click to save the changes you have made to this policy profile. |
| | **Close** | Click to close the **Policy Details** window. If you have made changes the profile, be sure that you have clicked **Save** before closing the page; otherwise, changes are lost. |

# Maintaining License Policies

On the Policy Details Window window, when you click **Add License** to create a license policy or click the ✎ icon (at the end of a license policy entry) to edit a license policy, the **Add** (or **Edit**) **License and Usage Criteria** window is displayed. From here you can add or modify the following criteria used to define the license policy.

Note that a license policy must be unique. That is, you cannot have two license policies for the same license when all their usage criteria are the same (such as all are **Any**). However, you can have two license policies for the same license if they define different usage criteria.

**Table 8-41 ▪** License and Usage Criteria Used in a License Policy

| Category | Column/Field | Description |
|---|---|---|
| **Select License** | The license on which the policy is based. | |
| | **License** | Do either:<br>● Select the license from the dropdown list of available licenses.<br>● Click **Create Custom License** to create a license to assign to this policy. See the next description. |
| | **Create Custom License** | Click this button to open the **Create Custom License** window and create your own license. See Step 2: Create the Custom License for instructions. Once you save the custom license, you are returned to the current **Add** (or **Edit**) **License and Usage Criteria** window, where the custom license has now been added to the **License** dropdown and is in focus for your immediate selection. |
| **Select Usage Criteria** | These properties are defined for an inventory item to describe how a software package developed in your organization *uses* the OSS or third-party component associated with the inventory item. Because usage can have an impact on your license obligations and conditions of use, the following usage properties are available as criteria in a given license policy. | |
| | **Distribution Type** | Specify the criterion that identifies how the OSS or third-party component associated with the inventory item is distributed with your software package. Distribution type can affect license priority and your license obligations.<br>● **Internal**—The component is distributed internally only (for example, as an internal test framework included in the codebase but not distributed publicly with the software package).<br>● **External**—The component is a separate entity from your software package. It might be shipped as a separate component along with the software package or deployed through some method, such as a private cloud at the customer site.<br>● **Hosted**—The component is hosted in your company’s data center (for example, as a SAAS application)<br>● **Any**—The policy ignores this criterion. |

**Table 8-41 ▪** License and Usage Criteria Used in a License Policy (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Linking** | Specify the criterion that identifies how your software package links to the OSS or third-party component libraries. This method can affect license priority and obligations.<br>● **Not linked**—The software package uses no links to the component libraries.<br>● **Statically linked**—The component libraries are included in the software materials and thus linked statically.<br>● **Dynamically linked**—The component libraries are brought in at runtime.<br>● **Any**—The policy ignores this criterion. |
| | **Modified** | Specify the criterion that identifies whether code from the OSS or third-party package has been modified for use by your organization.<br>● **No**—The component software has not been modified.<br>● **Yes**—The component software has been modified.<br>● **Any**—The policy ignores this criterion. |
| **Actions** | These actions manage the license policy creation or update. | |
| | **Add** | Click this button to add the license policy to the **Licenses** list on the **Policy Details** window. Once added, you can select the review status that the policy automatically assigns when an inventory item meets the policy criteria. |
| | **Save (for updates)** | Click this button to save the policy changes and return the to the **Policy Details** window. |
| | **Cancel** | Click this button to discard any policy changes and return to the **Policy Details** window. |

# Adding Reviewer Content to Policies

When reviewers (and other users) examine published inventory that has been approved or rejected automatically by the Code Insight policy, they likely do not know or have access to the policy that resulted in the approved or rejected inventory. Without this information, they might not know what factors were involved in the rejection of inventory, what issues need to be addressed for rejected inventory, or what guidelines or special notes are available for approved inventory.

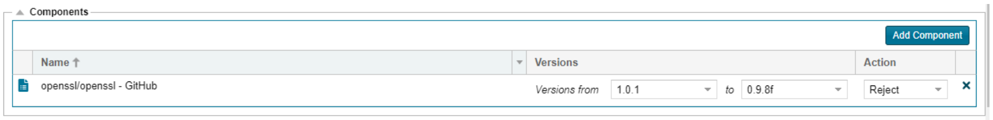
From the Policy Details Window, users with Policy Manager permissions can provide such information for reviewers by adding guidance content for any given policy in the policy profile currently open in the window. (To add this information, simply click 📄 at the left of the policy row, and enter the content in the **Usage Guidance** pop-up.) Then, if an inventory item is automatically approved or rejected by the policy, this content is propagated to the **Usage Guidance** pane for the item on the **Project Inventory** tab, providing reviewers with context about the inventory's status. (While users can generally edit content in the **Usage Guidance** pane for project inventory, they cannot edit the specific content propagated from policies to this pane.)

Refer to the following topics for more information:

- Usage Guidance Scenario
- Samples of Usage Guidance Content

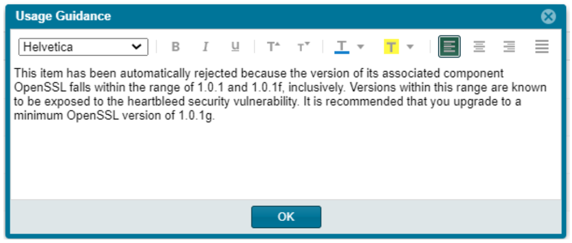
## Usage Guidance Scenario

Suppose you define the following policy for a component version on the **Policy Details** window.



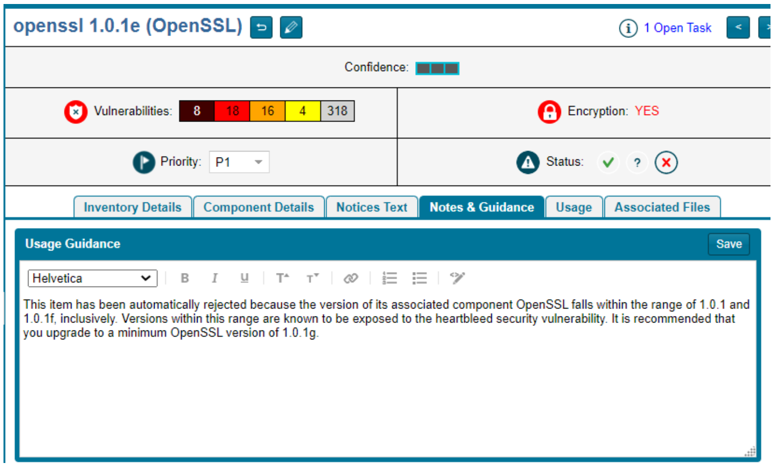
| Name | Versions | | | | Action |
|---|---|---|---|---|---|
| openssl/openssl - GitHub | Versions from | 1.0.1 | to | 0.9.8f | Reject |

You also provide the following content in the **Usage Guidance** popup for the policy. (Click 📄 to the left of the policy to open the pop-up.)



This item has been automatically rejected because the version of its associated component OpenSSL falls within the range of 1.0.1 and 1.0.1f, inclusively. Versions within this range are known to be exposed to the heartbleed security vulnerability. It is recommended that you upgrade to a minimum OpenSSL version of 1.0.1g.

When an inventory item is rejected because its component and version meet the criteria of this policy, the **Usage Guidance** pane on the **Notes & Guidance** tab for this inventory item shows your explanation.



This item has been automatically rejected because the version of its associated component OpenSSL falls within the range of 1.0.1 and 1.0.1f, inclusively. Versions within this range are known to be exposed to the heartbleed security vulnerability. It is recommended that you upgrade to a minimum OpenSSL version of 1.0.1g.

If one or more policies approve an inventory item, the **Usage Guidance** content from each of the policies is listed in the **Usage Guidance** pane for the inventory.

If one or more policies reject an inventory item, the **Usage Guidance** content from the only the first policy that rejected the inventory item is displayed in the **Usage Guidance** pane for that item.

User cannot edit the specific content propagated from policies to the **Usage Guidance** pane. However, they can edit and add other information in this pane.

## Samples of Usage Guidance Content

The following shows examples of **Usage Guidance** content that can be provided for policies:

- For Vulnerability Policies
- For License Policies
- For Component Policies

### For Vulnerability Policies

These are samples of **Usage Guidance** content that can be provided for policies listed in the **Vulnerabilities** section.

**Rejection based on CVSS score:**

> *This item has been automatically rejected due to one or more associated security vulnerabilities with a CVSS score greater than 7.0. Please consult with your security team for further guidance.*

**Rejection based on severity:**

> *This item has been automatically rejected due to one or more associated security vulnerabilities with a high severity. Please consult with your security team for further guidance.*

### For License Policies

These are samples of **Usage Guidance** content that can be provided for policies listed in the **Licenses** section.

**Approval based on license:**

> *License Name: Apache License 2.0 (Apache-2.0)*
> *License Priority: 3 - Permissive / Public Domain*
>
> *Usage Guidance:*
> *- Registration required before use*
> *- Include in third-party notices if shipped*
> *- Retain copyright notices*

**Rejection based on license:**

> *This item has been automatically rejected based on a combination of the associated weak-copyleft license (LGPL-2.1) and the fact that the items has been modified and is being distributed.*

### For Component Policies

This is a sample of **Usage Guidance** content that can be provided for a policy listed in the **Components** section.

> **Rejection based on a component with version range:**
>
> *This item has been automatically rejected due to the component being OpenSSL versions 1.0.1 through 1.0.1f. These are known to be exposed to the heartbleed security vulnerability. We recommend you upgrade to a minimum OpenSSL version of 1.0.1g.*

## Impact on Policies When Code Insight's CVSS Configuration Changes

If the Code Insight System Administrator changes the CVSS version for Code Insight, the following describes the impact on policies related to vulnerabilities on the Policy Details Window.

### When CVSS v2.0 is switched to CVSS v3.x

Code Insight makes the following changes:

- If the severity level for the **Reject inventory items if any associated securities vulnerabilities have a severity level equal to or higher than...** field was **Unknown** previously, it is now **None**.
- An additional severity, **Critical**, is available for this same field.

### When CVSS v3.x is switched to CVSS v2.0

Code Insight makes the following changes:

- If the severity level for the **Reject inventory items if any associated securities vulnerabilities have a severity level equal to or higher than...** field was previously **None**, it is now **Unknown**.
- If the severity level for this same field was previously **Critical**, note that this severity is no longer available. To handle the conversion, Code Insight checks to see if a score was previously entered in the **Reject inventory items if any associated security vulnerabilities have a CVSS score above**... field. If a score less than **9** was entered, that value is retained in the field (since the previous **Critical** severity started with the score **9**). If a value greater than **9** or no value was entered, the value for this field is now **9**.

# Policy Page

The **Policy** page lists all current policy profiles available for use by Code Insight projects. A policy profile contains a set of policies used to perform an automatic review of a project's inventory items upon publication of the inventory. The policies within a given policy profile are defined and maintained on the associated Policy Details Window for the profile.

From the **Policy** page, users who have Policy Manager permissions can access functionality to create new policy profiles, as well as update or copy existing profiles, as described in Managing Policy Profiles. Users without Policy Manager permissions can view policies only.

To open the **Policy** page, use either of these methods:

- From the Code Insight dashboard (which is displayed when you start Code Insight), click **view policy**. See Opening Code Insight for details on accessing the dashboard.
- Click the ☰ icon in the upper right corner of the Code Insight Web UI to open the Code Insight main menu. Select **POLICY** from the menu.

The **Policy** page contains the following information and functionality:

**Table 8-42 ▪** Policy Page

| Column/Field | Description |
|---|---|
| **Policy** | The list of current policy profiles in a grid format. Each entry shows the profile name and its description, the user who last updated the profile, and date of the last update for the profile.<br>Select a policy profile to edit, copy, or view.<br>● **View icon**—Click 🗎 to view the selected policy profile in a read-only format. The **Policy Details** window is opened, listing all the policy details without providing editing capabilities. (This is the only option available to users who do not have Policy Manager permissions.)<br>● **Edit icon**—Click ✎ to edit the selected policy profile. The **Policy Details** window is opened, showing the profile details.<br>● **Copy icon**—Click 🗐 to copy the selected policy profile. The **Policy Details** window is opened, showing the new copy. (The selected profile is always saved first.) This new copy is named `Copy of` *`selected_policyProfile_name`*. |
| **Add Policy** | Click the **Add Policy** button to create a new policy profile. The **Policy Details** window is opened. |

**See Also**
Policy Details Window
License Details Window
Managing Policy Profiles

# Preferences Page

The **Preferences** page appears when you select **Preferences** from the main menu. From this page, you can change your Code Insight user account password. In addition, you can view and add authorization tokens—that is, JSON Web Tokens known as JWTs—for use with Code Insight REST APIs. (The authorization token are associated with the current user account.) The page has the following fields:

**Table 8-43 ▪** Preferences Page

| Column/Field | Description |
|---|---|
| **Change Password** | |
| **New Password** | Enter a new password for the selected authorization token. The password must be a minimum of 8 characters, one of which must be numeric and one of which must in upper case. No spaces are allowed in the password. |
| **New Password Confirm** | Reenter the password you entered in the **New Password** field. |
| **Update Password** | After entering the password in both fields, click **Update Password** to save your changes. |
| **AUTH Tokens** | |
| **Add Token** | Click this button to display the **Add Token** dialog. |
| **Name** | A list of the names of previously created tokens. |
| **Token** | The system-generated token associated with the name. |
| **Create Date** | The date on which the token was created. |
| **Actions** | A group of icons that indicate actions you can take on each token:<br>● **Edit (✎)**: Click to open the **Edit Token** dialog.<br>● **Delete (✕)**: Click to delete the selected token. The token is deleted immediately.<br>● **Copy to clipboard (📋)**: Click to copy the selected token to the clipboard. You can use this option to copy tokens so that you paste them whenever needed for Code Insight REST access (for example, when configuring the Code Insight plugins, importing or exporting project data, or running REST API directly). |

**See Also**

Add Token Dialog
Edit Token Dialog
Exporting and Importing Project Data
Performing a Remote Scan

# Project Defaults Tab

The settings on **Project Defaults** tab on the **Administration** page work provide a convenient way to pre-populate fields used to configure new projects to ensure consistency and enable an easier project creation experience for users. Although the settings you define here are global across all projects, they can be overridden at the project level as needed. See the following field descriptions for more information.

**Table 8-44 ▪** Project Defaults Tab

| Category | Field | |
|---|---|---|
| **General Options** | These options set defaults for project creation and assign default users to project roles. Users can change these defaults when creating a project or when editing a project or its users using **Manage Project \| Edit Project \| General** or **Manage Project \| Edit Project \| Edit Project Users** on the project **Summary** tab. | |
| | **Project Visibility** | Select the default for visibility status—**Public** or **Private**—for projects. (The initial system default is **Public**.)<br>Any user in the system read-only access to a public project. To what degree a user can interact with the project depends on whether the user has a project role and what the role is—Project Administrator, Analyst, or Reviewer.<br>However, private projects are hidden from all users except the Project Contact and those users assigned as Project Administrators, Analysts, Reviewers, or Observers of the project. Additionally, project and vulnerability ID searches will not return private projects unless the user performing the search has the permissions to see a given private project. |
| | **Project Risk** | Select the default risk value (**Low**, **Medium**, or **High**) for projects. To edit, select another value from the dropdown. The initial system default is **Medium**. |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| | **Project Users** | Click the **Edit Project Users** link to open the **Edit Default Project Users** page. From here you assign project roles—Analysts, Reviewers, and Observers—that will default for any new project created (but which can then be edited at the project level). See Edit (Default) Project Users Page for details. |
| | **On the data import or rescan, delete inventory with no associated files** | This option determines whether “empty” system-generated inventory items are deleted in the target project during project imports and rescans. Empty inventory items have no associated files.<br>● **Selected**—Deletes empty inventory items from the target project during project imports and rescans. Only inventory items with associated files are retained/created.<br>● **Unselected**—Retains/creates all inventory items—with or without matching associated files in the target codebase—in the target project during imports and rescans. For example, you might want to retain inventory items to save their analysis details. (Users will need to manually delete inventory that is not applicable to the current project.)<br>This configuration (unselected) is required when importing a scanned codebase into a project for which no codebase has been uploaded or obtained through synchronization. This option ensures that inventory is generated in the target project. |
| **Scan Settings** | These options identify the default Scan Server and scan profile for projects. Users can change these settings at the project level by navigating to **Manage Project \| Edit Project \| Scan Settings** from the project **Summary** tab. | |
| | **Scan Profile** | Select the scan profile to default for projects. Click ⓘ to view the details of the scan profile. |
| | **Scan Server** | Select the Scan Server to default for projects. Note that only those Scan Servers in an “enabled” state are available for selection. If only one Scan Server has been identified to the system, this server is automatically selected as the default. |
| **Automated Inventory Publish Options** | These options enable and configure the automatic publication of project inventory as part of the project scan process. Users can change these settings at the project level by navigating to the project **Summary** tab and selecting **Manage Project \| Edit Project \| Scan Settings**.<br>If the **Auto-publish system-created inventory items meeting this minimum Confidence Level** is selected to enable auto-publication, the other auto-publish options are made available. | |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| | **Auto-publish system-created inventory items meeting this minimum Confidence Level** | Select this option to enable the auto-publication of system-generated inventory items. (By default, the option is selected.)<br>Then select the minimum Inventory Confidence level required to determine which items to auto-publish:<br>● **Low**—Automatically publish all system-generated inventory.<br>● **Medium**—Automatically publish only those system-generated inventory items with Medium and High confidence levels.<br>● **High**—Automatically publish only those system-generated inventory items with a High confidence level.<br>For a description of the Confidence levels and how they are used, see Inventory Confidence. |
| | **Do not auto-publish inventory items with an undetermined license** | Select this option to *not* auto-publish any system-generated inventory item with an undetermined license (that is, an inventory item whose **License** value is **I don’t know**). An undetermined license can occur under the following conditions:<br>● The scan was not able to identify a license for the given component during the scan and therefore set the **I don’t know** license value.<br>● The inventory item has multiple possible disjunctive licenses (for example, “GPLV2 or MIT”). However, the scan could find no evidence of the desired selected license and therefore set the **I don’t know** license value.<br>● The inventory item has multiple possible conjunctive licenses (for example, “GPLv2 and MIT”). However, since Code Insight currently supports only a single selected license, the scan automatically set the **I don’t know** value for the inventory item.<br>This option is available only if **Auto-publish system-created inventory items meeting this minimum Confidence level** is selected. By default, when you first open Code Insight instance after it has been installed or migrated, this option is unselected, allowing the auto-publication of inventory with undetermined licenses. |
| | **Mark associated file as reviewed** | Select this option if you want Code Insight to automatically mark the files associated with each automatically published inventory item as “reviewed”.<br>This option is available only if **Auto-publish system-created inventory items meeting this minimum Confidence level** is selected. |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| **Automated Review Options** | These options configure defaults for enabling policies that automatically accept or reject inventory when it is published. Users can change these settings at the project level by navigating to **Manage Project \| Edit Project \| Review and Remediation Settings** from the project **Summary** tab. | |
| | **Policy Profile** | Select the default policy profile to associate with all new projects. (The system default is **Default License Policy Profile**.)<br>The policy profile contains a set of policies that use components, versions, licenses, and vulnerability scores and severities as criteria to automatically reject or approve inventory items during a codebase scan (or post-scan).<br>For more information about policy profiles in general, see Managing Policy Profiles. |
| | **Automatically reject inventory items impacted by a new vulnerability that violates your policy** | Indicate the default action to take for published inventory affected by a new security vulnerability downloaded as part of an Electronic Update. The selected action applies to both non-reviewed and previously approved inventory items on the **Project Inventory** tab.<br>● Select this checkbox to automatically reject those project inventory items impacted by a new security vulnerability only if this vulnerability has a CVSS score or severity *greater than* the thresholds configured as policy for the Code Insight project. For each inventory item rejected due to a new security vulnerability, the ⚑ icon and a tip are added to indicate the status change and its reason.<br>If a new vulnerability does not exceed policy thresholds, the current status of the inventory item is not affected.<br>● Leave the checkbox unselected to retain the current status of inventory items impacted by the new vulnerability.<br>For information about setting policies that define CVSS-score and severity thresholds used to reject or approve inventory items automatically, see Policy Page and Policy Details Window. For information about associating these policies with a project, see Managing Policy Profiles. |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| **Manual Review Options** | These options configure defaults for project inventory not automatically reviewed by policy. Users can change these settings at the project level by navigating to **Manage Project \| Edit Project \| Review and Remediation Settings** from the project **Summary** tab. | |
| | **What should happen if inventory items are not reviewed by policy?** | Indicate the default action to trigger for those inventory items that are *not* affected by policy (and therefore have a **Not Reviewed** status) during the publication of inventory either as part of a scan or manually by a user:<br>● **do nothing**—Simply show the status of the inventory item as **Not Reviewed** on the **Project Inventory** tab.<br>● **send an email notification to the contact**—Automatically send an email to the Project Contact, stating the need for a manual review of the item. The value for **Select the minimum priority...** (described in the next table entry) affects this option.<br>● **automatically create a manual review task**—Automatically create a manual review task assigned to the default legal or security reviewer (or both reviewers), and send an email, notifying the reviewer(s) about assigned task.<br>Information about managing such a task to track the progress of a manual review is found in Creating and Managing Tasks for Project Inventory.<br>The value for **Select the minimum priority...** (described in the next table entry) affects this option. |
| | **Select the minimum priority to perform the action selected above** | (Enabled when an option other than **do nothing** is selected for the previous field.) Select the default minimum inventory priority (**P1**, **P2**, **P3**, or **P4**) to which the value for the previous field applies.<br>For example, if the previous field is set to **send an email notification to the project contact** and minimum priority is set to **P3**, then the email notification will be sent for only those non-reviewed inventory items with a P1, P2, or P3 priority. No email notification will be sent for P4 inventory items.<br>***Note ▪*** *This option has no effect when the* ***do nothing*** *value is selected.* |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| | **What type of manual reviews will be performed on this project?** | Set the default type of manual review tasks to be generated:<br>● **Legal Only**—Review tasks are generated for those non-reviewed inventory items that so not meet legal policy criteria. The tasks are automatically assigned to the default Legal reviewer.<br>● **Security Only**—Review tasks are generated for only those non-reviewed inventory items that have security vulnerabilities. The tasks are automatically assigned to the default Security reviewer.<br>● **both Legal and Security**—Review tasks are generated for all non-reviewed inventory items that do *not* meet legal policy criteria; these are assigned to the default Legal reviewer. Additionally, review tasks are generated for those non-reviewed inventory tasks associated with security vulnerabilities and are assigned to the default Security reviewer.<br>With this value, a single inventory item might have both a legal review task and security review task generated. However, if the default reviewers are the same user, a single task is created, describing the requirement for both a legal and security manual review. |
| | **Select reviewers for this project** | If desired, designate a new default reviewer to which to assign manual review tasks. (The available reviewer types—Legal or Security or both—depend on the type of manual reviews your site performs, as defined for the previous option.)<br>If your site generates both Legal and Security review tasks, Code Insight determines which reviewer—Legal or Security—is assigned the task and then notified of the task by email. See the previous option description for “both Legal and Security” for more information on how this determination is made.<br>For details about managing and reassigning tasks, see Creating and Managing Tasks for Project Inventory.<br>To select a new default reviewer, click **Change User** next to the name of the current **Legal reviewer** or **Security reviewer** assignee, then select a user from the **Select new...contact** dialog, and click **Apply**. (To reset the reviewer to the Project Contact, click **Reset**.)<br>When a new default reviewer is selected, that user is automatically given the role of project “reviewer” should the user not currently have this role.<br>If “Project Contact” is specified as a default reviewer, the Project Contact’s actual user name is displayed for the reviewer in the project. |

**Table 8-44 ▪** Project Defaults Tab (cont.)

| Category | Field | |
|---|---|---|
| **Remediation Options** | These options configure defaults for rejected project inventory. Users can change these settings at the project level by navigating to **Manage Project \| Edit Project \| Review and Remediation Settings** from the project **Summary** tab. | |
| | **What should happen if inventory items are rejected?** | Indicate the default action to trigger for those inventory items that are automatically rejected by policy during the publication of inventory either as part of a scan or manually by a user:<br>● **do nothing**—Simply show the status of the inventory item as Reject on the Project Inventory tab.<br>● **send an email notification to the project contact**—Automatically send an email to the Project Contact, stating the need for remediation work on the inventory item.<br>● **automatically create a remediation task**—Automatically create a remediation task assigned to the default development contact (see the **Assignee for remediation work** option) and send an email, notifying the contact about the assigned task.<br>● **automatically create a remediation task and an external work item**—Automatically do the following:<br>  ● Create a remediation task assigned to the default development contact (see the **Assignee for remediation work** option) and send an email, notifying the contact about the assigned task. (See the previous bulleted item for more information.)<br>  ● Associate a work item with the task, creating the work item in an Application Lifecycle Management (ALM) system (such as an issue in Jira). The work item is created and assigned using the settings defined for the ALM instance to which the Code Insight project is associated. For more information about configuring an ALM instance for the project, see ALM Settings. |
| | **Assignee for remediation work** | If desired, designate a new default development contact—for example, an engineering manager—to which to assign remediation tasks. (The Project Contact is the initial system default.) This contact can then manage the task accordingly—for example, reassigning it to another user or manually creating an external work item and assigning it to someone on the development team. For details about managing and reassigning tasks, see Creating and Managing Tasks for Project Inventory.<br>To select a new contact, click **Change User** next to the name of the current assignee, select a user from the **Select new...contact** dialog, and click **Apply**. (To reset the reviewer to the Project Contact, click **Reset**.)<br>If “Project Contact” is specified as the default, the Project Contact’s actual user name is displayed as the remediation assignee in the project. |

**See Also**

Policy Page
Policy Details Window
Edit Project: General Tab
Edit Project: Scan Settings Tab
Edit Project: Review and Remediation Settings Tab
Edit (Default) Project Users Page
Managing Policy Profiles
Assigning and Removing Project Users
Creating Inventory from the Project Inventory Tab
Creating and Viewing External Work Items for a Project Inventory Task
ALM Settings

# Project Inventory Details Pane

The **Project Inventory Details** pan is located on the **Project Inventory** tab for the current project. It is populated with details about the currently selected inventory item on the **Project Inventory** tab. (For a description of the **Project Inventory** tab and how to access it, see Project Inventory Tab.)

The **Project Inventory Details** pane enables legal and security experts to review the published inventory as needed and either approve items for inclusion in the Bill of Materials or reject them until further review or remediation efforts are performed. The reviewers can create tasks for the additional reviews or for the remediation work required by software engineering to fix security or legal risks in the code. They can also finalize the third-party Notices content that can be used in the Bill of Materials.

The following table describes the **Project Inventory Details** pane.

**Table 8-45 ▪** Project Inventory Details Pane

| Category | Column/Field | Description |
|---|---|---|
| **Header information** | The header on the **Project Inventory Details** pane provides buttons that enable you take actions on the inventory item. It also lists attributes about the item and its associated component. | |
| | **Recall Item** | Click to recall (remove) a published inventory item from **Inventory Items** list if it does not fit the criteria for inclusion. The selected items are removed from the **Project Inventory** view and are only visible in the **Analysis Workbench**. |
| | **Edit Item** | Click to open the Edit Inventory dialog where you can update inventory attributes. See for Editing Inventory from the Project Inventory Tab details. |
| | **View History** | Click open the Inventory History Window, which shows a list of all updates made to the inventory item up to the current date and provides details for each update. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Previous Item/ Next Item** | Show the details for the previous or next inventory item in the **Inventory Items** list. |
| | **Confidence** | A simple three-segment graph representing the Confidence level (High, Medium, or Low) of the inventory item. The Confidence level is the measure of the strength of the discovery technique used to generate the inventory item. The graph shows three shaded segments for High confidence, two for Medium, and one for Low.<br>For more information about the Confidence levels, see Inventory Confidence in the “Using Code Insight” chapter. |
| | **Encryption** | The **Yes**, **No**, or **N/A** value indicating whether the component associated with the inventory item provides the encryption capabilities used in your product. Encryption can affect export controls.<br>This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Vulnerabilities** | A bar graph showing the count of known vulnerabilities by severity color for the component associated with the inventory item. Click the graph to view the list of vulnerabilities and their details. For details about the graph and vulnerabilities in general, see Working with Security Vulnerabilities.<br>The counts in this graph do not include vulnerabilities that are currently suppressed. If no vulnerabilities have been found for the inventory item, the value **No** is displayed in place of the graph. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Priority** | A dropdown list showing the priority level given to this inventory item by the system, with P1 as the highest priority and P4 as the lowest.<br>You can change the priority for this inventory item by selecting a different priority from the dropdown list and clicking **Save**. For more information about priorities, see Inventory Priority. |
| | **Status** | The status of the inventory item:<br>● **Approved**—The item is approved for use in the software project.<br>● **Not Reviewed**—The item has not been automatically reviewed by policy (and therefore requires a manual review).<br>● **Draft**—This item is in the process of being reviewed.<br>● **Rejected**—The item is not approved for use in the software project. Instead, the item needs further review and remediation before being used in the software project. |
| **Inventory Details tab** | The **Inventory Details** tab lists attributes of the inventory item. | |
| | **Name** | The name of the inventory item. This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Description** | A description of the inventory item. This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **URL** | The URL of the license for this inventory item. You can click the URL link to open the component website. This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

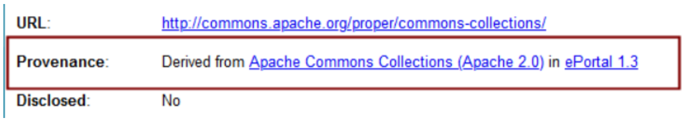
| Category | Column/Field | Description |
|---|---|---|
| | **Provenance** | The source project from which the current inventory item was derived.<br>***Note ▪*** *You cannot update this property from the Code Insight Web UI in general, but you can edit it when creating or updating inventory using the Inventory REST API.*<br>If the inventory item is not derived from another project, the value **Originated in this project** is displayed.<br>However, if the inventory item is derived from another project (for example, the current inventory item was imported to the current project), the origin of the inventory is displayed with the inventory name and project name:<br><br>**URL**: http://commons.apache.org/proper/commons-collections/<br>**Provenance**: Derived from Apache Commons Collections (Apache 2.0) in ePortal 1.3<br>**Disclosed**: No<br>If the source project and inventory item still exist, this value is hyperlinked so that you can open the source project directly to the **Project Inventory** tab, with focus on the **Inventory Details** page for the original inventory item. This direct link enables you to explore the auditing and review details of the original inventory item to determine inventory history—for example, the reason the item was previously approved or rejected. If the source inventory item or project no longer exists, no link to the original inventory item is provided. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Disclosed** | The property indicating whether the third-party component or artifact represented by the inventory item known third-party dependency in your code before it was discovered by the scan or you. The value is either **Yes** or **No**.<br>This field is used most often by analysts to denote information about the state of the inventory item.<br>This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Modified** | The property indicating whether code from the OSS or third-party package has been modified for use by your organization. The value is either **Yes, No,** or **Unknown**.<br>This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Alerts** | The property notifying you whether or not security vulnerability alerts exist for this item. If alerts exist, click the ***x* Open Alerts** or ***x* Closed Alerts** link to view their details. If no alerts exist, **None** is displayed. You can open the **Alerts** dialog from this pane to change the status or priority of an alert. For more information, see Managing Security Vulnerability Alerts. |
| | **Tasks** | The number of open or closed tasks for this inventory item. Click the **x Closed Tasks** or **x Open Tasks** link to view and update the tasks. If no tasks are associated with this inventory item, **None** is displayed. You can access the Tasks dialogs from this pane to create, edit, and close tasks. See for Creating and Managing Tasks for Project Inventory details. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Workflow URL** | The URL link or a plain text reference (such as a Jira issue number) to request data pertaining to this inventory item in your site's external workflow system. The link enables the reviewer to easily access the workflow data that tracks the status of open tasks for the inventory item. (The plain text reference still helps the reviewer locate the appropriate data in the workflow system.)<br>You can define this attribute when you edit or manually create an inventory item from the **Analysis Workbench** or the **Project Inventory** tab.<br>If no URL or reference has been defined, the value is **None**.<br>If additional request-related details are available for this inventory item, the ⓘ icon is displayed next to the URL. Click the icon to open the **Workflow Request Details** window for a quick review of pertinent details about the request without having to access the workflow system.<br>***Note ▪*** *These details come from the specific external workflow system associated with your site. The details can vary based on your workflow system.* |

Table 8-45 • Project Inventory Details Pane (cont.)

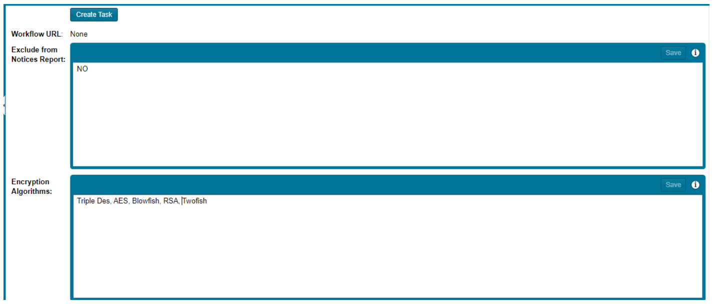
| Category | Column/Field | Description |
|---|---|---|
| | **Custom fields** | Any custom fields that were defined specifically for your site display at the bottom of the **Inventory Details** tab (after the **Workflow URL** field). These fields provide information that standard Code Insight fields on the **Inventory Details** tab do not capture about the inventory.<br> <br>Workflow URL: None<br>Exclude from Notices Report: NO<br>Encryption Algorithms: Triple Des, AES, Blowfish, RSA, Twofish<br>If no custom fields have been defined, nothing is displayed after the **Workflow URL** field.<br>Use the following guidelines for entering (or editing) a value in a custom inventory field:<br>● If available, click the ⓘ icon in the upper right corner of a field to obtain help on completing the field.<br>● You can enter a value up to 64k (64000 characters) in size.<br>● To save the value, click the **Save** button in the upper right corner of the field. (This button is activated when you begin to type in the field.) |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Component Details tab** | The **Component Details** tab lists attributes of the OSS or third-party component associated with the inventory item. | |
| | **Component** | The name of the OSS or third-party component and internal ID, as identified in the Code Insight data library. You can associate the inventory item with a different component using the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Version** | The component version and its internal ID, as identified in the Code Insight data library. You can associate the inventory item to a different version of the component using the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Forge** | The external repository associated with the component. You can click the forge link to open the forge website. |
| | **Selected License** | The name of the license selected for this component. Click ⓘ to view additional information about the license. See License Details Window.<br>You can switch to a different license from the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |
| | **Possible Licenses** | Other licenses that can be associated with the component. |
| | **Custom Component** | The **Yes** or **No** value indicating whether the component is custom (created by a user) or provided as part of the Code Insight data library. |
| | **Vulnerabilities** | A bar graph showing the count of known vulnerabilities by severity color for the component. Click the graph to view the list of vulnerabilities and their details. For details about the graph and vulnerabilities in general, see Security Vulnerabilities Associated with Inventory.<br>If no vulnerabilities have been found for the inventory item, the value **No** is displayed in place of the graph. |
| | **Encryption** | The **Yes**, **No**, or **N/A** value indicating whether the component provides the encryption capabilities used in your product. Encryption can affect export controls.<br>This attribute can be updated on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **CPE** | The list of CPE names—from the National Vulnerability Database—that are mapped to the component. CPE (Common Platform Enumeration) is a structured naming scheme that includes the component's vendor and product names in the following format:<br>**cpe://<part>:<vendor>:<product>**<br>where <part> is either **a** (applications), **h** (hardware platforms), or **o** (operating systems).<br>Note that the data provided only represents the part, vendor and product; the version information is truncated from the CPE string. |
| **Notices Text tab** | The **Notices Text** tab is used to finalize the exact content to include in the Notices report. For more information, see Finalizing the Notices Text for the Notices Report. | |
| | **As-Found License Text** | The **As-Found License** `Text` field shows the license text or license references found in the scanned codebase. You cannot edit this field, but you can click **Copy to Notices Text** to copy the text to the **Notices Text** field. If content already exists in the **Notices Text** field, you can choose either to append the **As-Found License Text** content to the existing notices content or to replace the existing notices content. |
| | **Notices Text** | The exact content to include in the Notices report. You can edit any license text previously saved to this field or add your own license text, such as license information for rules that you developed during your manual research on the inventory item. You can also copy the **As-Found License Text** content to the **Notices Text** field and modify it as needed. Or you can leave this field empty. Click **Save** at the top of the field if you make any changes to this field.<br>If you provide information in this field, the Notices report pulls the content of only this field into the report. If this field is empty, the content of the **As-Found License Text** field is used in the report. If both fields are empty, the report uses the license content from Code Insight data library (see License Details from the Code Insight Data Library).<br>For more information, see Finalizing the Notices Text for the Notices Report. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Notes & Guidance tab** | The **Notes & Guidance** tab provides information about the automated and manual analysis of codebase as it relates to an inventory item. | |
| | **Detection Notes** | System notes that can specify the following:<br>● The automated detection technique that was used to locate the component<br>● License information in the case that the license has changed from one version to another or if the component has multiple licenses<br>● Attributes extracted from a POM or manifest file containing project and configuration details |
| | **Audit Notes** | Any notes added to the inventory item by the auditor or reviewer, based on findings during the analysis. |
| | **Usage Guidance** | Notes helpful provided by a reviewer to assist other reviewers or to provide guidance to software engineers assigned tasks to fix or modify the use of the OSS or third-party software in the product code. |
| **Usage tab** | The **Usage** tab provides details on how your product uses the OSS or third-party software. You cannot update these items on the **Usage** tab, but you can update them on the **Edit Inventory** dialog (see Editing Inventory from the Project Inventory Tab). | |
| | **Distribution Type** | The option indicating how the inventory item is distributed:<br>● **Internal**—Internally only (such as test framework that might be included in the codebase but is not distributed with the product).<br>● **External**—Externally with the product, shipped to customers (outside of your organization, including a private cloud deployment at the customer's site)<br>● **Hosted**—Hosted in your company's data center (such as a SAAS application).<br>● **Unknown**—Unknown distribution type. |
| | **Part of Product** | The option indicating whether the item is part of the core product or an infrastructure piece such as a build or test tool. This can affect whether third-party notices are required for this item. The value can be **Yes**, **No**, or **Unknown**. |

**Table 8-45 ▪** Project Inventory Details Pane (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Linking** | The option identifying how your software package links to the OSS or third-party component libraries. This method can affect license priority and obligations.<br>● **Not linked**—The software package uses no links to the component libraries.<br>● **Statically linked**—The component libraries are included in the software materials and thus linked statically.<br>● **Dynamically linked**—The component libraries are brought in at runtime.<br>● **Unknown**—The type of linking is unknown. |
| | **Modified** | The option indicating whether code from the OSS or third-party package has been modified for use by your organization. The value can be **Yes**, **No**, or **Unknown**. |
| | **Encryption** | The option indicating whether the component provides the encryption capabilities used in the product. Encryption can affect export controls. The value can be **Yes**, **No**, or **Unknown**. |
| **Associated Files tab** | Click this tab to view a list of the files that are part of the inventory for this project. Each file entry shows the following:<br>● **Alias**—The unique user-defined alias that was defined for the scanner (Scan Server or remote scan agent) to represent its scan-root path containing the codebase in which the file is located. The alias provides a name that is more meaningful than the scan-root path. (The actual absolute scan-root path for each scanner associated with the project is available on the project's **Summary** tab.)<br>● **File Path**—The file's path relative to the scan-root path on instance hosting the scanner. You can click the file-path link to open the **File Details** tab for that file.<br>If you have Analyst permissions, the path is hyperlinked to open to the file's **File Details** tab in the **Analysis Workbench**, where you can view file evidence. If necessary, while in the **Analysis Workbench**, you can also add or remove files associated with the inventory. If you do not have Analyst permissions, the path remains in plain text. | |

# Project Inventory Tab

The **Project Inventory** tab lets you search and review the published inventory of open-source and third-party components found in the source code and artifacts scanned in a Code Insight project. To access the **Project Inventory** tab, see Displaying Project Inventory.

## About Published Project Inventory

An inventory item can be placed in a *published* state automatically by a scan (based on policy criteria) or manually by an analyst if enough evidence exists to ensure that this component is actually used by your code. At publication, those inventory items that are not automatically approved (based on policy criteria) will need to be reviewed to validate that they should be included in the Bill of Materials for your product.

The **Project Inventory** tab provides the means of finalizing the inventory to include in your Bill of Materials. The tab enables legal and security experts to review the published inventory as needed and either approve items for inclusion in the Bill of Materials or reject them until further review or remediation efforts are performed. The reviewers can create tasks for the additional reviews or for the remediation work required by software engineering to fix security or legal risks in the code.

## Field Descriptions

The **Project Inventory** tab consists of two panes: the left pane showing the list of published inventory for the project and the **Project Inventory Details** pane on the right showing the details of the inventory item currently selected in the list. The following table describes the information on this tab.

**Table 8-46 ▪** Project Inventory Tab

| | | Field/Panes | Description |
|---|---|---|---|
| **Inventory Items (x) pane** | This pane shows the list of inventory items that have been published in the project. The pane title includes the number of inventory items currently displayed in the list. | | |
| | **Inventory item fields** | The following properties describe each inventory item in the list. | |
| | | **Name** | The inventory item name in *componentName version (license)* format, as in the following:<br>apache-activemq 5.4 (Apache-2.0)<br>If the inventory item is a dependency of another inventory item, the relationship is shown within brackets, as in this example:<br>activemq-optional 5.4.0 [Bundled with apache-aapache-activemq 5.4] (Apache-2.0)<br>You can sort the inventory list on this column in ascending or descending alphabetical order. |
| | | **Priority** | The inventory priority, indicating how the item ranks in importance in the review process, with P1 as the highest priority and P4 as the lowest. For more information about inventory priority, see Inventory Priority in the “Using Code Insight” chapter.<br>You can sort the inventory list on this column in ascending (P1 to P4) or descending numeric order (P4 to P1). |

**Table 8-46 ▪** Project Inventory Tab (cont.)

| | | Field/Panes | Description |
|---|---|---|---|
| | | **Vulns** | The total number of security vulnerabilities associated with the inventory item. Vulnerability details for the inventory item are available in the Project Inventory Details Pane when you select the item. |
| | | **Status** | The review status of the inventory item: **Not Reviewed**, **Approved**, or **Rejected**. For more information about the inventory status, see Review Status of Inventory in the “Using Code Insight” chapter.<br>You can use this column to sort the inventory list by review status in ascending (**Not Reviewed**, **Approved**, **Rejected**) or descending (**Rejected**, **Approved**, **Not Reviewed**) order. |
| | **Inventory search options** | If you need to filter the inventory list to locate the inventory items to review, use either or both of these options. | |
| | | **Enter Inventory Name** | To filter the inventory list by a string contained in the item name, enter this string in the **Enter inventory name** box, and click 🔍 (or press Enter) to view the results. |
| | | **Advanced Search** | To filter the inventory list by one or more inventory properties, click the **Advanced Search** button. For complete details about performing an advanced inventory search, see Advanced Inventory Search Dialog. |
| | **Add Item** | This button is visible only if you have Analyst permissions.<br>To add a new inventory item to the list, click this button to open the **New Inventory** dialog. (The new item is automatically placed in the published state.) For more information creating an inventory item, see Creating Inventory from the Project Inventory Tab in the “Using Code Insight” chapter. | |
| **Project Inventory Details pane** | When you select an inventory item, the **Project Inventory Details** pane on the right is populated with details about the item. From this pane, you can edit the item’s properties, set up review and remediation tasks for the item, provide audit, usage-guidance, and remediation notes, edit the item’s third-party Notices content, and ultimately approve or reject the item for its inclusion in the Bill of Materials. For complete information, see Project Inventory Details Pane. | | |

**See Also**

Displaying Project Inventory
Advanced Inventory Search Dialog
Project Inventory Details Pane
Editing Inventory from the Project Inventory Tab
Reviewing Published Inventory for a Project

# Projects Pane and Associated Dashboard

The **Projects** pane in the **Projects** view displays the projects available in your Code Insight instance. You can configure this display as needed to help you locate projects and open them. You can also create, delete, and rename projects from this pane. When you select a project from this pane, its associated dashboard is displayed. The **Projects** pane and associated project dashboard contain the following fields:

**Table 8-47 ▪** Projects Pane Page

| Column/Field | Description |
|---|---|
| **Tree view** | Click to toggle the project display to tree view. |
| **List view** | Click to toggle the project display to a plain list view. |
| **Add New** | Click to add a new folder or project to the list. (This button is displayed only if you have permission to create projects.) |
| **My Projects** | Click to show only those projects with which you are associated, either as Project Contact or with a project role (Analyst, Reviewer, or Observer). |
| **Projects (x)** | The number of projects in the system. If the list is filtered, the filtered count is shown in relation to the full count (for example, “(19 of 123)”). |

**Table 8-47 ▪** Projects Pane Page (cont.)

| Column/Field | Description |
|---|---|
| **Project Search fields** | From the search filters on the left, select the filter based on the type of search you want to perform (**Project Name**, **Project Inventory**, or **Security Vulnerability**).<br>In the field on the right, enter the string criterion for the search:<br>● When searching for a project name, enter a partial string or full project name.<br>● When searching for project inventory, enter the inventory name or the inventory’s component name, license name, or license SPDX short identifier. The characters must be consecutive in the search string. A partial string is supported.<br>● When searching for a security vulnerability, enter the exact vulnerability ID.<br>Press Enter to view the filtered projects display. If no inventory items meet the specified criterion, the projects display shows “No Projects”.<br>To clear the search filter and display of all projects, click ✖ in the criterion field.<br>See for Searching Across All Projects the System for full details. |
| **Project tree or plain list** | The display format (tree or plain list) of listed projects based on the current filter. You can perform the following from this list (depending on your permissions):<br>● To open a project, either click the **Open Project** icon () next to the project entry in the project tree or list, or click the project’s name link in the upper left corner of the project’s dashboard (to the right of the **Projects** pane). The project opens to its **Project Inventory** tab.<br>● To display the dashboard for a project, either click the project in the project tree or list, or click the **Load Project Dashboard** icon 📊 in the project entry.<br>● To rename a project, double-click the project name and overwrite the current name with the new name.<br>● To move a project to a different location in the project tree, drag and drop the project to the desired folder (or to the root location).<br>● To create or delete a project or folder from the project tree or list, use the right-click menu. See Managing Items in the Projects Display. |
| **Selected Project Name** | Select a project from the project tree or list to refresh the dashboard on the right with information about the selected project. You can click the project name displayed in the top left of the dashboard to open the project. |

**Table 8-47 ▪** Projects Pane Page (cont.)

| Column/Field | Description |
|---|---|
| **Project Contact** | In the project dashboard header, the hyperlinked name of the user designated as the Project Contact—the main point of contact for the project. Click the name to open your default email program to send an email to the Project Contact. |
| **Profile** | In the project dashboard header, the name of the profile attached to the selected project. If no profile is attached to the project, “No profile selected” appears. |
| **Created** | In the project dashboard header, the date on which the project was created. |
| **Last Scan** | In the project dashboard header, the date on which the codebase was last scanned. |
| **Project Summary Graphs** | On the project dashboard, graphs providing overview statistics about the inventory associated with the selected project. The graphs are interactive; when a section of a graph is clicked, the **Project Inventory** tab is displayed, showing a filtered view of the inventory that applies to the section of the graph you clicked. See also Using the Project Dashboard. |

**See Also**

About Code Insight Projects
Summary Tab
Showing Only Your Projects
Searching Across All Projects the System
Using the Project Dashboard

# Reports Tab

The **Reports** tab is displayed within the context of a given project and enables you to generate any standard and custom Code Insight reports currently available for the project. (The same reports are available to all projects.) You can generate a single report or multiple different reports simultaneously for the project. Once the generation of a report has completed, links are provided to view an HTML version of the report and to download the report in all its available formats.

For more information about the procedure used to generate a report and for a description of standard and custom reports, see Generating Reports for a Project.

The following table describes the components of the **Reports** tab.

**Table 8-48 ▪** Reports Tab

| Category | Column/Field | Description |
|---|---|---|
| **Report list** | The list of reports on the **Reports** tab shows the following information for each available Code Insight report—standard or custom. | |
| | **Name** | The name of the report.<br>To generate a report, select its name and click **Generate Selected Report**. |
| | **View Report** | The **View** link to open the last generated version of the report in HTML format. When you click the link, the report is displayed in your browser. Anytime that you regenerate the report, the link is updated to open the new report version.<br>No link is displayed until the report is generated for the first time. |
| | **Download Report** | The **Download** link to download an archive containing all available formats of the last generated version of the report. When you click the link, the archive is downloaded to your system's default download location, where you can then open (and save) the report in any of its formats. Anytime you regenerate the report, the link is updated to download the new report version.<br>No link is displayed until the report is generated for the first time. |
| | **Generated On** | The date and time of the last successful generation of the report. No date and time is displayed until the report is generated for the first time. |

**Table 8-48 ▪** Reports Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Generate Selected Report** | | Click this button generate the currently selected report. The following happens:<br>● A message prompt appears explaining that the report will be generated in the background, enabling you to continue to work in Code Insight as the report generates. Click **OK** to proceed with the report generation.<br>Or<br>● If the **Include data from Second Project** field is displayed, enter the name of the second project whose data will be included along with the data from the current project for comparison purposes. As you type a string, project names containing that string are listed in a dropdown from which you can then select the desired project name. (This is a required field.)<br>● If other fields are displayed, enter the requested values in those fields. Default values can be overwritten. Click the ? icon next to a field for more information about its purpose and possible values. The **Generate Report** button on the pop-up remains disabled until all required fields are completed. (Required fields left blank are outlined in red.)<br>When these additional fields have been properly completed, click **Generate Report** on the pop-up window.<br>Note that the **Generate Selected Report** button is disabled for each report that is currently generating, but *is* enabled for any other report not being generated. While one report is generating, you can select and generate another report, enabling you to generate multiple different reports for the project simultaneously.<br>Once a report is successfully generated, the **View** and **Download** links and the date/time of the generation are provided for that report.<br>If a report generation fails, a message is displayed (in the area where the links ad day/time information for the report would have displayed), stating that the report has failed and to check the logs for further information. (A Code Insight System Administrator can review the contents of the `core.log` to determine the reason for the report failure and relay the information to the appropriate contacts to fix the issue.) |

**See Also**

Opening a Project
Generating Reports for a Project

# Scan History Dialog

The **Scan History** dialog displays a list of previous scans that have been performed on the selected project. The dialog contains the following fields:

**Table 8-49 ▪** Scan History Dialog

| Column/Field | Description |
|---|---|
| + | Click to view messages about the scan. If no messages were generated during the scan, the message field will be blank. |
| **Scheduled On** | The date and time that the scan was scheduled. |
| **Started On** | The date and time that the scan was started. |
| **Completed On** | The date and time that the scan completed. |
| **Duration** | The amount of time the scan took. |
| **Scheduled By** | The user name of the person who scheduled the scan. |
| **Status** | The status of the scan: *Completed* or *Failed*. |
| **Ok** | Click **Ok** to exit the **Scan History** dialog and return to the **Scan Summary** page. |

**See Also**
Auditing Scan Results in the Analysis Workbench

# Scan Profiles Tab

The **Scan Profiles** tab on the **Administration** page allows you to add a scan profile and edit information about an existing scan profile. The tab contains the following columns and fields:

**Table 8-50 ▪** Scan Profiles Tab

| Column/Field | Description |
|---|---|
| **Scan Profiles** list | A list (in grid format) of available scan profiles. The following are the predefined scan profiles:<br>● Standard Scan Profile<br>● Basic Scan Profile (without CL)<br>● Comprehensive Scan Profile<br>The list will contain additional profiles if you have added them.<br>The following are key attributes shown for each scan profile in the list. These attributes are described in detail in Create/Edit Scan Profile Dialog.<br>● **Scan Archives**—Whether the Scan Server will perform package discovery and license detection within all archive files in the project codebase.<br>● **Dependencies**—The level of component-dependency scanning to be performed by the Scan Server.<br>● **Exact Matches**—Whether Scan Server is enabled to identify those codebase files that exactly match file data in the CL (Compliance Library).<br>● **Source Code Matches**—Whether the Scan Server is enabled to identify source-code strings (snippets) in the scanned codebase files that match exact strings in the CL. |
| **Edit** icon ✎ | To edit a scan profile, click this icon in the Actions column for the profile. The **Edit Scan Profile** dialog is opened, enabling you to edit profile settings. See Create/Edit Scan Profile Dialog for setting descriptions. |
| **Add Scan Profile** button | Select this button to create a new scan profile. The **Create Scan Profile** dialog is opened. See Create/Edit Scan Profile Dialog for setting descriptions. |

**See Also**

Create/Edit Scan Profile Dialog
About Code Insight Scans
Applying a Scan Profile to the Project

# Scan Server Dialog

Before a user can assign project codebases to a Scan Server in order to scan them, the Scan Server must first be installed either on the same instance as the Code Insight Core Server or on a separate instance, as described in the *Code Insight Installation and Configuration Guide*. (The Scan Server must have the same version as the Core Server.) As Code Insight System Administrator, you must then use the **Scan Server** dialog to “add”—that is, identify—the server to the Code Insight system to make it available for scanning purposes.

In addition to adding a new Scan Server, you use the **Scan Server** dialog to edit an existing Scan Server’s properties. For detailed instructions on adding or editing a Scan Server, see “Adding or Editing Scan Servers” in the *Code Insight Installation and Configuration Guide*.

## Multiple Scan Servers

If multiple Scan Servers have been installed, you can identify more than one of these servers to the system, thus enabling users to distribute codebase scans. Keep in mind that, when multiple Scan Servers are installed, each should be installed on a different instance with a unique host ID and port. The codebase for a given project can be assigned to only one of the Scan Servers (but multiple project codebases can be assigned to a single Scan Server). All codebases assigned to a given Scan Server are stored on that server in a location that you specify.

## Prerequisite for Adding or Editing a Scan Server

Ensure that the Scan Server that you are adding or editing is currently running and that the Scan Server you are adding has the same version as the Core Server.

## Dialog Fields

The **Scan Server** dialog contains the following fields:

**Table 8-51 ▪** Scan Server Dialog

| Column/Field | Description |
|---|---|
| **Alias** | Enter a common name for the Scan Server. |
| **Host** | Provide the hostname (such as `krl.eng.companyA.com`) or IP address of the instance hosting the Scan Server. If the Scan Server is on the same instance as the Core Server, enter `localhost`.<br>The same host-and-port combination must be unique among the *enabled* Scan Servers. (See **Status** is this table for a description of enabled Scan Servers.) |
| **Port** | Specify the port used by the Scan Server on the host instance. By default, the port is `8888`.<br>The same host-and-port combination must be unique among the *enabled* Scan Servers. (See **Status** is this table for a description of enabled Scan Servers.) |

**Table 8-51 ▪** Scan Server Dialog (cont.)

| Column/Field | Description |
|---|---|
| **CL Path** | Provide the path for the Code Insight Compliance Library (CL), downloaded from the Product and License Center. The CL is a database used by the Scan Server to perform exact-file and source-code fingerprint (snippet) matching. Code Insight compares elements of scanned codebase files with information contained in the CL to generate file-level evidence on which you can take action.<br>The validity of the entered path is checked when you click **Save**.<br>Alternatively, leave this field blank to scan your codebase without using the CL. (Code Insight provides the scan profile “Basic Scan Profile (without CL)” to perform the scan.) This type of scan generates inventory from Code Insight’s Automated Analysis feature but has limitations, as described in “About Scanning without the Compliance Library” in the *Code Insight Installation and Configuration Guide*. Keep in mind that, when you run a scan using the CL (that is, by specifying a valid CL path), you obtain a deeper, more comprehensive scan on your codebase.<br>For additional information, see the following:<br>● “Managing Scan Profiles” in the *Code Insight Installation and Configuration Guide* for more information about the “Basic Scan Profile (without CL)” and about creating and managing scan profiles in general.<br>● Applying a Scan Profile in the “Using Code Insight” chapter in this book for instructions on associating a scan profile with a project.<br>● About Code Insight Scans in the “Using Code Insight” chapter in this book for information about Code Insight scans in general. |
| **Codebase Path** | Provide the path on the Scan Server where Code Insight will store and manage all uploaded code for projects that use this Scan Server. Ensure you have adequate disk space to store the codebases. The recommended starting size for this directory is 500GB.<br>The directory must already exist. The validity of the entered path is checked when you click **Save**.<br>Once the Scan Server to added to the Code Insight system, this field cannot be edited. |

**Table 8-51 ▪** Scan Server Dialog (cont.)

| Column/Field | Description |
|---|---|
| **Status** | By default, the Scan Server is enabled for scanning.<br>However, if necessary for an existing Scan Server, select **Disabled** to make the Scan Server unavailable for further scans. Once disabled, the server is no longer displayed in the **Scan Server** dropdown during project creation or when setting global project defaults. Additionally, this field becomes read-only on the **Edit Project** dialog.<br>Note the following about disabling a Scan Server:<br>● If this Scan Server is the system default Scan Server (as defined on the **Project Defaults** tab), you must change this default to another server before you can disable the current server. See Project Defaults Tab for instructions on updating the default Scan Server.<br>● If this Scan Server is associated with one or more projects, a warning is displayed before you can disable the server. Once you click **Yes**, the **Start Scan** and **Upload Project Codebase** options are disabled on the **Summary** page for each project associated with the server.<br>If you attempt to re-enable a disabled Scan Server when another currently *enabled* Scan Server has the same host-and-port combination or alias, you receive an error when you click **Save**. |
| **Save** | Click this button to save any changes you made to the Scan Server properties. Errors are generated when the following conditions exist:<br>● The Scan Server you are adding or editing is not running.<br>● The version of the Scan Server you are adding is different from the Core Server version.<br>● The codebase path or CL path is invalid. |
| **Cancel** | Click this button to cancel any changes you made to the fields on the Scan Server dialog. |

**See Also**
Project Defaults Tab
Edit Project: Scan Settings Tab
Scan Servers Tab
About Code Insight Scans

# Scan Servers Tab

The **Scan Servers** tab on the **Administration** page lists the Scan Servers that you have identified to your Code Insight system. Each entry in the list shows basic information about the given Scan Server including its status. From a given entry, you can access a separate dialog to edit server properties as well as refresh the entry itself to see the latest server status. The tab also lets you define a new Scan Server. The tab contains the following columns and buttons to identify and manage Scan Servers:

**Table 8-52 ▪** Scan Servers Tab

| Category | Field | | |
|---|---|---|---|
| **Scan Server entry** | The following columns for each entry in the Scan Server list provide information about the given Scan Server, as well as a means to refresh the Scan Server status and edit server properties. | | |
| | **Alias** | The user-defined name for the Scan Server, as well as the server’s current status. The following icons represent the server status: | |
| | | 🟩 | The green icon indicates that the Scan Server is “enabled” for scanning and is currently running (turned on). Scans are run in queue order. |
| | | 🟥 | The red icon indicates that the Scan Server is “enabled” for scanning but is currently not running (that is, it is turned off). Any attempts to associate a project with the Scan Server or upload a codebase to the server generates an error. Additionally, any attempt to initiate a scan will result in the scan’s being queued. However, once the server is active, the scan will start based on queue order. (Users can click the **Past Scans** link on the project **Summary** page to view details about the scheduled scan.) |
| | | ◼ | The gray icon indicates that the Scan Server is “disabled” (that is, cannot be used for scanning). Whether or not the server is running has no effect on this status. If an enabled server is needed for scans on a project assigned to a disabled Scan Server, you must create a new project. |
| | **Host** | The `localhost` value is used if the Scan Server is on the same instance as the Core Server. | |
| | **Port** | The port used by the Scan Server on the host instance. By default, the port is `8888`. | |
| | **CL Path** | The path for the Code Insight Compliance Library (CL). If the path is specified, the CL is accessed as part of the scan to perform exact-file and source-code fingerprint (snippet) matching. Elements of scanned codebase files are compared with information contained in the CL to generate file-level evidence on which you can take action.<br>If the path is not specified, the codebase is scanned without using the CL. This type of scan generates inventory from Code Insight’s Automated Analysis feature but has limitations. For more information about the Compliance Library, see About Code Insight Scans. | |

**Table 8-52 ▪** Scan Servers Tab (cont.)

| Category | Field | |
|---|---|---|
| | **Codebase Path** | The path on the Scan Server where Code Insight will store and manage all uploaded code for projects that use this Scan Server. Once the Scan Server is added to the Code Insight system, this field cannot be edited. |
| | **Actions for scan server entry** | Select the appropriate icon for the desired action:<br>● The ✎ (**Edit**) icon to edit the configuration for the given Scan Server. The **Scan Server** dialog is opened, enabling you to make edits.<br>● The ↻ Refresh icon (visible when you hover over the Scan Server entry) to refresh the Scan Server entry, including its status. |
| **Action** | Use the following button to add a new Scan Server to the Code Insight system—that is, that is, identify the server to the Code Insight Core Server to make it available for scanning purposes. To add a Scan Server, ensure that it has been installed and is running. See the *Code Insight Installation and Configuration Guide* for instructions for installing and starting a Scan Server. | |
| | **Add Scan Server** | Click this button to add a new Scan Server. The **Scan Server** dialog is displayed. |

**See Also**
Scan Server Dialog

# Select a New Project Contact Page

The **Select a new project contact** page lets you change the Project Contact of the current project. The Project Contact, initially the project creator, is the default contact for all task-workflow notifications generated during the inventory review process. That is, if a Legal, Security, or Development contact has not been explicitly assigned to the project through system Project Defaults or at the project-settings level, that contact defaults to the Project Contact. Additionally, the Project Contact is the default contact for any “miscellaneous” tasks created during an inventory review.

The project creator is the initial Project Contact. The Project Contact user is initially assigned to all project roles (but can be removed from these roles as needed). This user can also reassign the Project Contact to a different user. When Project Contact is reassigned to another user, that user is assigned to the same roles as the previous Project Contact to ensure a continuation of the same permissions.

The following fields are used to reassign the Project Contact:

**Table 8-53 ▪** Select a New Project Contact Page

| Column/Field | Description |
|---|---|
| **List of Users** | The names of all the users in the system are listed in this field. Select a name and click **Apply** to change the Project Contact. |

**Table 8-53 ▪** Select a New Project Contact Page (cont.)

| Column/Field | Description |
|---|---|
| **Apply** | Click this button to assign the selected user as Project Contact. |
| **Cancel** | Click this button to cancel changes without saving. |

# Summary Tab

The **Summary** tab for a project allows you to add and edit users who can work in Code Insight, view scan settings and status, generate reports, and manage projects. The page contains the following fields:

**Table 8-54 ▪** Project Summary Tab

| Category | Column/Field | Description |
|---|---|---|
| **Project Details** | These field describe the project attributes. You can edit these details using the **Manage Project \| Edit Project** and **Manage Project \| Edit Project Users** options available on this **Summary** tab.<br>***Note ▪*** *For inventory-only projects migrated from Code Insight 2020 R2 or earlier, a legacy attribute,* ***Project Type****, will also display. However, for migrated standard projects, this attribute is no longer required and therefore does not display. See also Legacy Projects.* | |
| | **Name** | The name and ID of the selected project. |
| | **Project Contact** | The hyperlinked name of the user who is the main point of contact for the project. Initially, the Project Contact is the project creator but can be assigned to another user (see Changing the Project Contact). You can click the user name to open your default email program to send an email to the Project Contact.<br>By default, all Miscellaneous tasks created for project inventory are assigned to the Project Contact. |

**Table 8-54 ▪** Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Legal Contact** | The hyperlinked name of the legal contact assigned to tasks created to review legal issues in the project inventory (for example, inventory that do not meet your site's legal policies). Click the name to open your default email program to send an email to the contact.<br>For details on changing the legal contact for the project, see Updating Inventory Review and Remediation Settings for a Project. If no changes are made to the initial system default for the legal contact (which is set to the Project Contact) or if the automated legal-review process is disabled, this value shows the Project Contact name. |
| | **Security Contact** | The hyperlinked name of the default security contact assigned to tasks created to review security issues in the project inventory. Click the name to open your default email program to send an email to the contact.<br>For details on changing the security contact for the project, see Updating Inventory Review and Remediation Settings for a Project. If no changes are made to the initial system default for the security contact (which is set to the Project Contact) or if the automated security-review process is disabled, this value shows the Project Contact name. |
| | **Developer Contact** | The hyperlinked name of the default development contact assigned to remediation tasks created to take action on code-related issues in the project inventory. Click the name to open your default email program to send an email to the contact.<br>For details on changing the default development contact, see Updating Inventory Review and Remediation Settings for a Project. If no changes are made to the initial system default for the developer contact (which is set to the Project Contact) or if the automated remediation process is disabled, this value shows the Project Contact name. |
| | **Description** | A description of the project, if provided in the project definition, appears in this field. |
| | **Project Visibility** | The visibility of the project:<br>● **Public**—A project visible to all users in the system. Users assigned to roles for the project can perform various maintenance functions based on their project roles.<br>● **Private**—A project visible to and accessible by the Project Contact and those users assigned to roles for the project. |
| | **Project Risk** | The project vulnerability risk value (**Low**, **Medium**, or **High**). |

Table 8-54 ▪ Project Summary Tab (cont.)

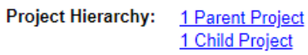
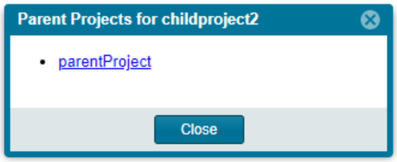
| Category | Column/Field | Description |
|---|---|---|
| | **Project Hierarchy** | Links to the projects that have been defined as parent and child projects of the current project. These links provide a means to easily navigate to projects directly related to the current project. (Relationships between projects are established by the creation of project hierarchies, as described in Identifying Child Projects for a Project.)<br><br>Project Hierarchy: 1 Parent Project 1 Child Project<br>When you click a **Project Hierarchy** link, a dialog is displayed listing the direct links to the parent or child projects.<br><br>Parent Projects for childproject2 • parentProject Close<br>Click a link on the dialog to open the given child or parent project on its **Project Inventory** tab. From here you can navigate the project as needed. |

**Table 8-54 ▪** Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Project Status** | The current status of the project that can be manually updated through the **Manage Project \| Edit Project** menu option available on this tab. Available status types include:<br>● **Not Started**—Indicates that the project scan results are not available for manual analysis. This value automatically defaults when the initial project scan has not been run or when the initial scan fails. However, you can manually change this status.<br>● **Analysis in Progress**—Indicates that the project scan results are available for manual analysis, or a manual analysis is underway. This value is automatically set when the initial project scan is complete. (For a project import, however, the status of the target project is retained.) You can manually change this status.<br>● **Analysis Completed**—Indicates that manual analysis is finished; and the published inventory is ready for legal, security, or software-engineering review. (You must manually set this value.)<br>● **Project Complete**—Indicates that the project analysis and review processes are complete, and notices content for all OSS and third-party software is finalized. (You must manually set this value.)<br>For more information, see Editing the Project Definition and General Settings and Edit Project: General Tab. |
| | **Policy Profile** | The name of the policy profile associated with this project. Click **View Policy Details** to open a read-only version of the Policy Details Window for the policy profile.<br>A policy profile contains a set of policies used to perform an automatic review of inventory items upon their publication during the scan. Each policy defines criteria based on OSS or third-party component versions, licenses, or security vulnerabilities. Inventory items that meet any of the profile’s policy criteria can be automatically approved or rejected (or flagged for a manual review). For more information, see Managing Policy Profiles. |

**Table 8-54 ▪** Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Scan Settings** | The following fields show scan configuration details. You can edit these details on the **Scan Settings** tab accessed using the **Manage Project \| Edit Project** option available on this tab. | |
| | **Scan Profile** | The name of the scan profile associated with this project. Click ⓘ to view the details of the scan profile. |
| | **Scan Paths** | The absolute path for each scan folder for the project. A given scan folder contains files for either:<br>● A codebase scanned by the Scan Server.<br>● A codebase scanned by a Code Insight scan-agent plugin on a remote Engineering system. (The results of the remote scan are sent to the project.) For more information, see Performing a Remote Scan.<br>Click ⓘ to view the details about Scan Server or the scan-agent plugin. |

**Table 8-54** ▪ Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Scan Status** | The following fields provide information about current and historical scans for this project. For more information about scans, see About Code Insight Scans. | |
| | **Scan Server Status** | The server scan status for this project:<br>● If you have started a server scan for this project, but the scan has been placed in queue, this field shows “Scan Scheduled”. The **Scan Progress** field provides a link to view the scan queue and other details (see the “Scan Progress” description below).<br>● If the server scan you scheduled for this project is running, this field shows “Project being scanned”. The **Scan Progress** field keeps track of the number of files that have been scanned.<br>● When the server scan for this project completes, this field shows “No scan scheduled” and provides link to schedule another scan. (If the Scan Server is disabled, the link is also disabled.) The **Scan Progress** field is not available when this status is in effect. |
| | **Scan Progress** | (Available only when a server scan for the project is scheduled or is running) The progress of the scan as follows:<br>● If the scan has been placed in queue, this field shows “In Scan Queue” and provides a **Show Details** link to open the **Scan Server Status** window. This window identifies the scan server, shows the project currently being scanned by the server, lists the other project scans (if any) currently waiting in queue order (up to 25 scans), and provides an email link for the Project Contact of each project listed. You cannot sort or reorganize the queue list.<br>● If the scan you scheduled is running, this field keeps track of the number of files that have been scanned against the total number of files to scan in the project. |

**Table 8-54** ▪ Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Last Server Scan** | The final status of the last server scan for the project and a statistical summary of files, disk space, and lines of code scanned. The following are available scan statuses:<br>- **Completed**—The scan succeeded with no warnings during the scan or the analysis phase. This message appears on the screen in green.<br>- **Completed with warnings**—The scan succeeded but the analysis phase produced warnings. For more information, check **scanEngineDetail** log for the Scan Server.<br>- **Failed**—The scan failed. This message appears on the screen in red. For more information, see Scan Failure Reasons and Troubleshooting Measures. |
| | **Past Server Scans** | Click the hyperlinked term **here** to view a history of the server scans performed for the project. If a server scan has not yet been performed for the project, the list will be empty. |
| | **Last Remote Scan** | Statistics for remote scans:<br>- The total number of scanned codebase files across all scan agents associated with the project.<br>- The total number of inventory items created across all scan agents associated with the project.<br>- The combined size of all agent-scanned codebases.<br>For information about remote scans, see Performing a Remote Scan.<br>**Note** ▪ *If the project is a newly migrated inventory-only project from Code Insight 2020 R2 or earlier, only the statistics for the scan agent that last performed a scan (on the previous Code Insight version) are shown.* |

**Table 8-54 ▪** Project Summary Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Upload Project Codebase** | Click to upload a codebase on which you will perform a server scan for the selected project. To upload multiple codebases, repeat the upload process by clicking this button for each codebase. You can also use this option to overwrite the current codebase with a more recent version.<br>If the current project is not associated with a Scan Server or if the Scan Server is disabled, this button is disabled.<br>***Note ▪*** *Alternatively, you can synchronize the project with one or more remote Source Control Management (SCM) codebase repositories (see Configuring Source Code Management). You can also both upload codebase files and synchronize with SCM repositories to provide the complete project codebase on which to run the server scan.* |
| | **Manage Project** | A dropdown menu that allows you to edit project settings, assign user roles for the project, export and import project data, delete a project, or change the Project Contact. The options available depend on project roles to which you are assigned. |

**See Also**
About Code Insight Projects
Editing the Project Definition and General Settings
Edit Project: General Tab
Projects Pane and Associated Dashboard
Uploading a Project Codebase (for Server Scans)
Exporting and Importing Project Data chapter

# Suppress Vulnerability Window

The **Suppress Vulnerability** window is displayed when you click the **Suppress** button for a given vulnerability on the **Security Vulnerabilities** window. (For more information about accessing this window and about suppressing vulnerabilities, see Suppressing a Security Vulnerability.)

The **Suppress Vulnerabilities** window enables you to suppress the given security vulnerability for one or more (or all) versions of the OSS or third-party component with which the vulnerability is associated. You might want to suppress a vulnerability, for example, if the vulnerability has proven to be a “false positive” (that is, is associated with an incorrect component version) or if remedial steps have been taken to protect your code against the vulnerability.

Vulnerability suppression takes place at the system level in Code Insight. Once suppressed, the vulnerability is no longer published in reports, counted in vulnerability totals at the project, inventory, and component levels, or automatically associated with inventory during future project scans in your Code Insight instance. For a complete description of the impact of suppressing a vulnerability, see Effects of Suppressing a Security Vulnerability.

Vulnerability suppression is performed by a Code Insight System Administrator only, who can also monitor a list of suppressed vulnerabilities and unsuppress vulnerabilities as needed.

The follow describes the fields and features on the **Suppress Vulnerability** window that enable you to suppress a given vulnerability.

**Table 8-55 ▪** Suppress Vulnerability Window

| Category | Description |
|---|---|
| **Vulnerability Id** | (Not editable) The ID assigned to the vulnerability by the source that reported it (see the next field).<br>Optionally, you can click the hyper-linked CVE ID in an entry to view the vulnerability details found on the NVD or other website:<br> <br>**Source:** NVD<br>**ID:** CVE-2017-1000372 |
| **Source** | (Not editable) The advisory system that reported the vulnerability (for example, NVD or Secunia). |
| **Severity** | (Not editable) The level of security risk that this vulnerability can have on your software. The advisory system uses the vulnerability's CVSS score to set the severity. See Understanding Severity Levels for Security Vulnerabilities. |
| **CVSS v3.x (or v2.0) Score** | (Not editable) The vulnerability's CVSS score as determined by the advisory system. Depending on your Code Insight configuration, this score is in either CVSS 3.x or CVSS 2.0 format. For more information, see Understanding Severity Levels for Security Vulnerabilities.<br>For a vulnerability found in the NVD, the UI also provides access to a CVSS calculator (provided by NVD). Using this calendar, you can tweak the factors that determined the NVD-based score to calculate another score that is more realistic for your product. This score can then be used internally to direct your review and remediation processes. For information about accessing the CVSS calculator, see the CVSS <version> Score description in Examining Security Vulnerability Details. |
| **Description** | (Not editable) The vulnerability description, as captured from the advisory system. |
| **Affected Component** | (Not editable) The OSS or third-party component that is impacted by this security vulnerability. |

**Table 8-55 ▪** Suppress Vulnerability Window (cont.)

<table>
<tr><th>Category</th><th colspan="2">Description</th></tr>
<tr><td><b>Version Scope</b></td><td colspan="2">(Required) Select the scope of component versions to which the vulnerability suppression will apply.<br>● <b>Specific Version(s)</b>—One or more component versions that you choose from the <b>Select Version</b> dropdown (which is enabled only when this option is selected). Note that the dropdown will show only those versions for which the vulnerability is currently unsuppressed.<br>By default, this option is initially selected, and the <b>Select Version</b> field shows the component version for the current inventory item.<br>● <b>All Current Versions</b>—All component versions for which the vulnerability is currently unsuppressed.</td></tr>
<tr><td><b>Select Version(s)</b></td><td colspan="2">(Enabled and required when <b>Version Scope</b> is <b>Specific Version(s)</b>) From the dropdown (listing all <i>unsuppressed</i> versions currently affected by the vulnerability), select each version for which you want the vulnerability to be suppressed.<br>By default, the component version for the current inventory item is initially specified.<br>If necessary, you can remove any of your version selections by clicking the small ✕ icon to the right of the version.</td></tr>
<tr><td><b>Select Reason</b></td><td colspan="2">(Required) Select the reason for suppressing the vulnerability for this component version:<br>● <b>False-positive</b>—The vulnerability was incorrectly associated with the component version and hence does not apply to the version.<br>● <b>Remediated</b>—The risk posed by the vulnerability on the component version has been addressed or fixed.<br>● <b>Other</b>—Another reason.</td></tr>
<tr><td><b>Suppression Remarks</b></td><td colspan="2">(Required) Enter all additional information pertinent to the suppression of the vulnerability for this component version.</td></tr>
<tr><td><b>Actions</b></td><td colspan="2">The following buttons enact or discontinue the vulnerability suppression process.</td></tr>
<tr><td></td><td><b>Suppress</b></td><td>(Enabled when all required fields have been completed) Click to suppress the security vulnerability for the given component version. Then click <b>OK</b> in the pop-up to acknowledge that vulnerability has been suppressed.</td></tr>
<tr><td></td><td><b>Close</b></td><td>Close window without saving your input.</td></tr>
</table>

# Suppressed Versions of <component> for <vulnerability> Window

This pop-up window is displayed when you click ⓘ next the **Affected Versions** value for a given suppressed security vulnerability listed on the Suppressed Vulnerabilities Tab. The window lists the component versions for which the vulnerability was suppressed and, for each version, provides details about the suppression (who suppressed the vulnerability and when, why the vulnerability was suppressed for this version, and more).

The following table describes the details provided for each component version, as well as describes the navigation features of the window.

**Table 8-56 ▪** Vulnerability Suppression Details for Each Impacted Component Version

| Category | Column/Field | Description |
|---|---|---|
| **Suppression details per component version** | The following describes details related to the suppression of the security vulnerability for the given component version. These details are not editable. | |
| | **Version Name** | A version of the given component for which the vulnerability was suppressed. |
| | **Reason** | The reason why the vulnerability was suppressed for this component version. (This information was entered by the System Administrator at the time of suppression.)<br>● **False-positive**—The vulnerability was incorrectly associated with the component version and hence does not apply to the version.<br>● **Remediated**—The risk posed by the vulnerability on the component version has been addressed or fixed.<br>● **Other**—Another reason. (Check the **Remarks** field for a possible reason description.) |
| | **Remarks** | Any additional notes about the suppression of the vulnerability for this component version, as entered by the System Administrator at the time of suppression. |
| | **Suppressed By** | The user ID of the System Administrator who suppressed the vulnerability for this component version. |
| | **Suppressed Date** | The date and time that the vulnerability was suppressed for this component version. |

**Table 8-56 ▪** Vulnerability Suppression Details for Each Impacted Component Version (cont.)

| Category | Column/Field | Description |
|---|---|---|
| **Actions** | The following buttons and icons enable you to navigate and manage the list of component versions on the window. | |
| | ↻ | Refresh the data in the window. |
| | **Page controls** | Move to the next or previous page or to the first or last page of the window; or enter a specific page number in the **Page** field.<br>Note that the default page size is 100 component version records. |
| | **Close** | Exit the **Suppressed Vulnerabilities** tab. |

# Suppressed Vulnerabilities Tab

The **Suppressed Vulnerabilities** tab on the **Data Library** page lists the security vulnerabilities currently suppressed in your Code Insight instance. (The data is listed in a grid format.) This tab is visible to only Code Insight System Administrators. For more information about accessing this tab, see Viewing Suppressed Security Vulnerabilities.

For a newly installed Code Insight instance or an pre-2021 R3 instance migrated to the current instance, this page initially shows no suppressed security vulnerabilities. (However, the tab will list any vulnerability you subsequently suppress.)

The **Suppressed Vulnerabilities** tab provides the following information and features:

**Table 8-57 ▪** Suppressed Vulnerabilities Tab

| Category | Column/Field | Description |
|---|---|---|
| **Filter by** | These fields enable you to filter the list of suppressed vulnerabilities. Select the filter type, either **Vulnerability Id** or **Component Name**, from the dropdown; and then enter the string by which to filter the list. For example, if you select **Component Name** and enter the string open, the list will filter to those suppressed vulnerabilities associated with a component whose name contains “open”. | |
| **Details for each suppressed vulnerability** | The following describes the details of each suppressed vulnerability listed in the grid. These details are not editable. The **Action** column includes the button used to unsuppress the given vulnerability. | |

**Table 8-57 ▪** Suppressed Vulnerabilities Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Vulnerability ID** | The ID assigned to the vulnerability by the advisory system that reported it.<br>Click ⓘ next to the ID to display a pop-up containing details about the vulnerability. The details include:<br>● **Vulnerability ID**—The ID assigned to the vulnerability by the source that reported it (see the next field).<br>● **Source**—The advisory system that reported the vulnerability (for example, NVD or Secunia).<br>● **Severity**—The level of security risk that this vulnerability can have on your software. The advisory system uses the vulnerability's CVSS score to set the severity. See Understanding Severity Levels for Security Vulnerabilities.<br>● **Score**—The vulnerability's CVSS score as determined by the advisory system. Depending on your Code Insight configuration, this score is in either CVSS 3.x or CVSS 2.0 format. For more information, see Understanding Severity Levels for Security Vulnerabilities.<br>For a vulnerability found in the NVD, the UI also provides access to a CVSS calculator (provided by NVD). Using this calculator, you can tweak the factors that determined the NVD-based score to calculate another score that is more realistic for your product. This score can then be used internally to direct your review and remediation processes. For information about accessing the CVSS calculator, see the CVSS <version> Score description in Examining Security Vulnerability Details.<br>● **Description**—A description of the vulnerability captured from the advisory system.<br>You can sort on this column alphabetically in ascending or descending order. By default, the IDs are listed in ascending order. |

**Table 8-57 ▪** Suppressed Vulnerabilities Tab (cont.)

| Category | Column/Field | Description |
|---|---|---|
| | **Affected component** | The OSS or third-party component that is impacted by the vulnerability. |
| | **Affected versions** | The one or more component versions for which the vulnerability is currently suppressed. If the versions are too numerous list in the grid, the value ends with “...”. However, you can always mouse-over the value to see the entire list of versions for which the vulnerability is suppressed.<br>Click ⓘ next to the value to display a pop-up window that shows suppression details for each listed version. For more information, see Suppressed Versions of <component> for <vulnerability> Window. |
| | **Action** | Click **Unsuppress** to unsuppress the vulnerability for one or more of the component versions for which it is suppressed. The Unsuppress Vulnerability Window is displayed to walk you through the process.<br>When you return to this window, the component versions for which you unsuppressed the vulnerability are no longer displayed. (If all the component versions for which the vulnerability was previously suppressed are now unsuppressed, the vulnerability is no longer listed on this window.) |
| **Actions** | The following buttons and icons enable you to navigate and manage the **Suppressed Vulnerabilities** tab. | |
| | ↻ | Refresh the vulnerability data on the tab. |
| | **Page controls** | Move to the next or previous page or to the first or last page on the tab; or enter a specific page number in the **Page** field.<br>Note that the default page size is 100 vulnerability records. |
| | **Close** | Exit the **Suppressed Vulnerabilities** tab. |

# System Settings Tab

The **System Settings** tab is used to define settings that configure your Code Insight system. The tab provides the following configuration settings:

**Table 8-58 ▪** System Settings Tab

| Column/Field | Description |
|---|---|
| **Security Vulnerability Options** | Select the CVSS (Common Vulnerability Scoring System) version—**CVSS v3.x** (3.0 and 3.1) or **CVSS v2.0** in which to display security vulnerability scores and severities in the Code Insight Web UI. Initially, **CVSS v 2.0** is the default.<br>If you switch versions, the CVSS scores and severity values displayed for vulnerabilities will be impacted, as will policies based on these values. For more information, see Security Vulnerabilities Associated with Inventory and Managing Policy Profiles. |

**Table 8-58 ▪** System Settings Tab (cont.)

<table>
<tr><th>Column/Field</th><th colspan="2">Description</th></tr>
<tr><td rowspan="2">**Custom Fields for Inventory**</td><td colspan="2">This section shows the list of custom inventory fields that you (the System Administrator) have created using Code Insight’s Custom Fields feature. From **Custom Fields for Inventory** list, you can create new custom inventory fields and edit existing ones.<br>Depending on its configuration, a given field can be made available in the Code Insight Web UI and through the REST interface (or through the REST interface alone). From either interface, users (analysts and reviewers) can access the field’s value for any inventory item. Users can also filter inventory on custom field values at the project and global levels.<br>Currently, a maximum of five custom inventory fields can be created.<br>***Note ▪*** *Once a custom field is created, it cannot be deleted. However, it can be disabled and re-enabled as needed.*<br>You can do the following from the **Custom Fields for Inventory** list:<br>● To create a new field, click **Add Field** to open the **Add Custom Field** window. Once the field attributes are saved, the new field is added to the list.<br>● To edit a field, click the ✎ icon in the **Action** column for the selected field. The **Edit Custom Field** dialog is opened, enabling you to change the field’s attributes. Once the changes are saved, they are reflected in the list.<br>The following describes the attributes used to define a custom inventory field. For more information about managing custom inventory fields, see “Creating and Managing Custom Inventory Fields” in the *Code Insight Installation & Configuration Guide*.</td></tr>
<tr><td>**Enabled**</td><td>The attribute determining whether the custom field will be activated in Code Insight. Use this option to control when you want the custom field to be made available.<br>● **Yes**—The field will be activated and made available in Code Insight. Use the **Visible in UI** attribute (see next) to determine *where* the field will be made available.<br>● **No**—The field will not be available in Code Insight. All other attributes defined here are ignored (until the field is enabled).</td></tr>
</table>

**Table 8-58 ▪** System Settings Tab (cont.)

| Column/Field | Description |
|---|---|
| **Visible in UI** | The attribute determining whether the custom field is displayed for inventory items in the Code Insight Web UI (that is, in the **Inventory Details** tab on the **Project Inventory** pane and the **Inventory Details** tab in the **Analysis Workbench**).<br>● **Yes**—The field displays in the Code Insight Web UI, enabling users (analysts and reviewers) to use the UI or the REST interface to view and update field's value for individual inventory items. (Default)<br>● **No**—The field does not display in the Web UI. However, users can view the field and update its value for individual inventory items through the REST interface. |
| **Field Label** | (Required) The name of the custom field. The maximum length is 30 characters. |
| **Help Text** | Information that is displayed when a user selects the ⓘ icon for the field in the Web UI. Provide content that helps users enter an appropriate value for the field. For example, you might describe the purpose the field and the type of value it requires. If you specify text with **http://** or **https://**, the value will be hyperlinked.<br>The maximum length is 150 characters.<br>If this attribute is left blank, the ⓘ icon will not be available for the custom field in the Web UI. |

**See Also**

Security Vulnerabilities Associated with Inventory
Managing Policy Profiles
Policy Details Window
"Setting the Common Vulnerability Scoring System" in the *Code Insight Installation and Configuration Guide*
"Creating and Managing Custom Inventory Fields" in the *Code Insight Installation and Configuration Guide*

# Unsuppress Vulnerability Window

The **Unsuppress Vulnerability** window is displayed when you click the **Unsuppress** button for a given security vulnerability on the Suppressed Vulnerabilities Tab. (For information about accessing this window, see Viewing Suppressed Security Vulnerabilities.)

The **Unsuppress Vulnerabilities** window enables you to unsuppress the security vulnerability for one, some, or all of component versions for which it was previously suppressed. The unsuppression process takes place at the system level of Code Insight, affecting all projects in your Code Insight instance.

Basically this process reverses the effects of the vulnerability’s previous suppression so that the vulnerability is once again available for processing. It can be published in reports, counted in vulnerability totals at the project, inventory, and component levels, and automatically associated with inventory during future project scans and rescans in your Code Insight instance. For a complete description of the impact on Code Insight when you unsuppress a vulnerability, see Effects of Unsuppressing a Security Vulnerability.

Only a Code Insight System Administrator only can unsuppress a vulnerability.

The follow describes the fields and features on the **Unsuppress Vulnerability** window that enable you to unsuppress a given vulnerability.

**Table 8-59 ▪** Unsuppress Vulnerability Window

| Category | Description |
|---|---|
| **Vulnerability Id** | (Not editable) The ID assigned to the vulnerability by the source that reported it (see the next field).<br>Optionally, you can click the hyper-linked CVE ID in an entry to view the vulnerability details found on the NVD or other website:<br> <br>**Source:** NVD<br>**ID:** CVE-2017-1000372 |
| **Source** | (Not editable) The research system or organization that reported the security vulnerability (for example, **NVD**, **Secunia**, or another advisory entity). |
| **Severity** | (Not editable) The level of security risk that this vulnerability can have on your software. The advisory system uses the vulnerability’s CVSS score to set the severity. See Understanding Severity Levels for Security Vulnerabilities. |
| **CVSS v3.x (or v2.0) Score** | (Not editable) The vulnerability’s CVSS score as determined by the advisory system. Depending on your Code Insight configuration, this score is in either CVSS 3.x or CVSS 2.0 format. For more information, see Understanding Severity Levels for Security Vulnerabilities.<br>For a vulnerability found in the NVD, the UI also provides access to a CVSS calculator (provided by NVD). Using this calculator, you can tweak the factors that determined the NVD-based score to calculate another score that is more realistic for your product. This score can then be used internally to direct your review and remediation processes. For information about accessing the CVSS calculator, see the CVSS <version> Score description in Examining Security Vulnerability Details. |
| **Description** | (Not editable) The vulnerability description, as captured from the advisory system. |
| **Affected Component** | (Not editable) The OSS or third-party component that is impacted by this security vulnerability. |

**Table 8-59 ▪** Unsuppress Vulnerability Window (cont.)

| Category | Description | |
|---|---|---|
| **Version Scope** | (Required) Select the scope of component versions for which you want to unsuppress the vulnerability.<br>● **Specific Suppressed Version(s)**—The one or more component versions that you choose from the **Select Version(s)** dropdown (which is enabled only when this option is selected). Note that the dropdown will show only those versions for which the vulnerability is currently suppressed.<br>By default, this option is initially selected.<br>● **All Suppressed Versions**—All component versions for which the vulnerability is currently suppressed. | |
| **Select Version(s)** | (Enabled and required when **Version Scope** is **Specific Suppressed Version(s)**) From the dropdown, select each version for which the vulnerability should be unsuppressed. The dropdown shows only those versions for which the vulnerability is currently suppressed.<br>If necessary, you can remove any of your version selections by clicking the small × icon to the right of the version. | |
| **Unsuppression Remarks** | (Required) Enter all additional information pertinent to the unsuppression of the vulnerability for the component version(s). | |
| **Actions** | The following buttons enact or discontinue the process of unsuppressing the vulnerability. | |
| | **Unsuppress** | (Enabled when all required fields have been completed) Click to unsuppress the security vulnerability for the specified component version(s). Then click **OK** in the pop-up to acknowledge that vulnerability has been unsuppressed. |
| | **Close** | Close window without saving your input. |

# Users/Permissions Tab

The **Users/Permissions** tab on the **Administration** page allows you to add and edit users who can work in Code Insight. The tab contains the following columns and fields:

**Table 8-60 ▪** Users/Permissions Tab

| Column/Field | Description |
|---|---|
| **Add User** | Click to display the **Add User** dialog. |

**Table 8-60 ▪** Users/Permissions Tab (cont.)

| Column/Field | Description |
|---|---|
| **Manage Permissions** | Click to display the **Manage Permissions** dialog used to assign the following permissions to users: System Administrator, Manage Policy, and Create Projects. |
| **Login** | Displays the login of each user that has been added. |
| **First Name** | Displays the first name of each defined user. |
| **Last Name** | Displays the last name of each defined user. |
| **Email** | Displays the email address of the user associated with the login. |
| **Actions** | This column contains the pencil icon (✎). Click it to open the **Edit User** dialog, where you can edit information about the selected user. |
| **Enter Search Criteria** | Enter a string by which to filter the list of users. A full or partial match to any of the user details is allowed. Click ✖ to remove the filter. |

**See Also**
Add User Dialog
Edit User Dialog
“Configuring Code Insight” in the *Code Insight Installation & Configuration Guide*

9

# Code Insight User Roles and Permissions

This appendix serves as a reference to the various user roles and permissions that Code Insight offers as a means to control user access to its functionality at your site:

- System Roles and Permissions
- Project Roles and Permissions
- Roles and Permissions to Manage the Review Task Flow

## System Roles and Permissions

The following table lists the roles and associated permissions used to manage Code Insight at the system level. The initial Code Insight System Administrator (and any subsequent System Administrators) manages user accounts and assigns system-level roles to any of these users as needed. For more information, see “Managing Users” in the “Configuring Code Insight” chapter in the *Code Insight Installation and Configuration Guide*.

One user can be assigned to multiple system roles.

**Table 9-1 ▪** System Roles and Permissions

| | | | Roles | | |
|---|---|---|---|---|---|
| | | | System Admin | Policy Manager | Project Creator |
| Responsibility | Permissions | Notes | | | |
| **Administer Code Insight** | Manage user accounts and permissions, create other system administrators, create policy managers, and allow all/or specified users to create projects | | ✓ | — | — |
| | Schedule or force Electronic Updates | | ✓ | — | — |
| | Configure an email server workflow notifications | | ✓ | — | — |
| | Configure LDAP users | | ✓ | — | — |
| | Configure Application Lifecycle (ALM) instances to manage inventory review tasks | | ✓ | — | — |
| | Configure Scan Servers and scan profiles | | ✓ | — | — |
| | Define global project defaults | | ✓ | — | — |
| | Determine the CVSS version used for security vulnerability reporting | | ✓ | — | — |
| | Create and manage custom fields | | ✓ | | |
| | View Code Insight logs | | ✓ | — | — |
| | Suppress security vulnerabilities | | ✓ | | |

**Table 9-1 ▪** System Roles and Permissions (cont.)

| | | | Roles | | |
|---|---|---|---|---|---|
| | | | **System Admin** | **Policy Manager** | **Project Creator** |
| **Manage polices for automating inventory review processes** | | | — | ✓ | — |
| **Create projects** | Create public and private projects | The user who creates a project automatically becomes the Project Contact for that project. (See Project Roles and Permissions for additional Project Contacts permissions.) | — | — | ✓ |
| | Manage project folders (in **Projects** pane) | | — | — | ✓ |

# Project Roles and Permissions

The following table lists the various roles and associated permissions used to manage a given project in Code Insight. The project creator automatically becomes the initial Project Contact and Project Administrator. In turn, a Project Administrator can assign Analyst, Reviewer, and Observer roles to Code Insight users, as well as create other Project Administrators. The Project Administrator can also remove users from any of these roles. For more information, see Assigning and Removing Project Users in this guide.

Users can be assigned multiple project roles.

**Table 9-2 ▪** Project Roles and Permissions

| | | | Roles | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | Analyst | Reviewer | Observer* | Proj. Contact | Proj. Admin | Sys. Admin |
| **Responsibility** | **Permissions** | **Notes** | | | | | | |
| **Manage project** | Reassign the project contact | | — | — | — | ✓ | ✓ | ✓ |
| | Manage project users | | — | — | — | — | ✓ | — |
| | Rename the project | | — | — | — | — | ✓ | — |
| | Move projects in **Projects** pane | | — | — | — | — | ✓ | — |
| | Manage scan settings | | — | — | — | — | ✓ | — |
| | Manage review/ remediation settings | | — | — | — | — | ✓ | — |
| | Manage Source Control Management (SCM) and Application Lifecycle (ALM) instances | | — | — | — | — | ✓ | — |
| | Delete the project | | — | — | — | — | ✓ | — |
| | Branch the project | | — | — | — | — | ✓ | — |
| **Invoke/stop scans** | | | ✓ | — | — | — | ✓ | — |
| **Upload codebases** | | | ✓ | — | — | — | ✓ | — |
| **Import/export project data** | | | ✓ | — | — | — | ✓ | — |
| **View project inventory** | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** |

**Table 9-2 ▪** Project Roles and Permissions (cont.)

| | | | **Roles** | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | **Analyst** | **Reviewer** | **Observer*** | **Proj. Contact** | **Proj. Admin** | **Sys. Admin** |
| **Review project inventory** | Recall inventory | | ✓ | ✓ | — | — | — | — |
| | Approve/reject inventory | | — | ✓ | — | — | — | — |
| | Set inventory priority | | — | ✓ | — | — | — | — |
| | Edit/create inventory | Only Analysts have access to the **Add Item** and **Edit Item** buttons to create/edit project inventory properties. | ✓ | — | — | — | — | — |
| | Update Notices text and notes | This permission refers to inventory's **Notices Text** field (on the **Notices Text** tab) and the information on the **Notes & Guidance** tab (except **Detection Notes**). | ✓ | ✓ | — | — | — | — |
| | Edit custom fields on the **Inventory Details** tab | | ✓ | ✓ | | | | |
| | View evidence found in files listed on the **Associated Files** tab and manage the inventory's file associations | For Analysts only, the file path for an associated file is hyperlinked, enabling them to open to the file's **File Details** tab in **Analysis Workbench** to view evidence. In **Analysis Workbench**, Analysts can also add/remove files associated with inventory. | ✓ | — | — | — | — | — |
| **Use Analysis Workbench** | View/analyze codebase files | | ✓ | — | — | — | — | — |

**Table 9-2 ▪** Project Roles and Permissions (cont.)

| | | | Roles | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | Analyst | Reviewer | Observer* | Proj. Contact | Proj. Admin | Sys. Admin |
| | Edit alerts | | ✓ | — | — | — | — | — |
| | Create, edit, and recall inventory and manage custom detection rules | | ✓ | — | — | — | — | — |
| | Edit **Notices Tex**t field on **Notices Text** tab | | ✓ | — | — | — | — | — |
| | Edit **Audit Notes** field on the **Notes** tab | | ✓ | — | — | — | — | — |
| | Edit custom fields on the **Custom Fields** tab | | ✓ | | | | | |
| **Generate reports** | | Any user (not just one with a project role) can generate reports. For a “private” project, the Observer is considered an “any user”, restricted to viewing project inventory and generating reports. | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

* The Observer role is available for only projects defined as “Private”. Private projects are hidden from all users except the Project Contact, the System Administrator (restricted to **Summary** tab only), and those users assigned as Project Administrators, Analysts, Reviewers, and Observers of the project. An Observer is limited to viewing project inventory and generating reports for the “Private Project”.

** In general, a System Administrator has permission to access both public and private projects. However, the **Project Inventory** tab for a private project is visible to a System Administrator only if the user assigned to the System Administrator role is also assigned to a role in the project (Project Administrator, Project Contact, Observer, Analyst, or Reviewer).

# Roles and Permissions to Manage the Review Task Flow

The following table lists the project roles and permissions used to manage the tasks to review or remediate inventory items in a project.

**Table 9-3 ▪** Project Task-Flow Roles and Permissions

| | | **Roles** | | | | | |
|---|---|---|---|---|---|---|---|
| | | **Analyst** | **Reviewer** | **Observer** | **Project Contact** | **Task Assignee** | **Project Admin** |
| **Permissions** | **Notes** | | | | | | |
| **Create/edit tasks** | Any user assigned to a project role can create and edit tasks. | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| **Reassign tasks** | | — | — | — | — | ✓ | ✓ |
| **Close manual review tasks** | | — | ✓ | — | — | — | — |
| **Close remediation tasks** | | — | — | — | — | ✓ | ✓ |
| **Close miscellaneous tasks** | Any user assigned to a project role can close a miscellaneous task. | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |