

Code Insight 2023 R3 SP1 Release Notes

September 2023

Introduction	2
About Code Insight	2
Revenera Resources	2
Changes in Code Insight 2023 R3 SP1	3
New Features and Enhancements.....	3
Advanced Inventory Searches.....	4
Components and Licenses	4
Custom Detection Rules	7
Data Library, Library Refreshes, and Electronic Updates	7
Jobs Queue	8
Installation, Upgrades, and Configuration.....	9
Project Inventory	10
Project Management	11
Scan Agent Plugins.....	11
Scanning and Automated Discovery.....	12
User Experience	14
REST API Enhancements.....	14
New APIs.....	14
Updates to Existing APIs	15
Resolved Issues.....	18
Issues Resolved in 2023 R3 SP1	19
Issues Resolved in 2023 R3	19
Known Issues	21
All-Project Inventory View	21
Automated Workflow for Inventory Review/Publication.....	22
Custom Detection Rules	22
Data Library, Library Refreshes, and Electronic Updates	22
Export and Import.....	24
Installation, Upgrades, and Configuration.....	25
Inventory History	26
Jobs Queue	26
Manual Codebase Analysis	27
Performance	28
Project Inventory	28
Project Management	29

Project Reporting	30
REST APIs	30
Scan Agent Plugins	31
Scanning and Automated Discovery	34
Source Code Management (SCM) Support	37
Vulnerability Suppression/Unsuppression	39
Web UI	40
Legal Information	41

Introduction

These Release Notes provide the following information about the Code Insight 2023 R3 release:

- [About Code Insight](#)
- [Reverera Resources](#)
- [Changes in Code Insight 2023 R3 SP1](#)
- [New Features and Enhancements](#)
- [Resolved Issues](#)
- [Known Issues](#)
- [Legal Information](#)

About Code Insight

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

Reverera Resources

The following resources can help you stay up to date with Code Insight news and product knowledge:

- In addition to providing case management, the [Reverera Community](#) site can help you 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 Reverera's product solutions, you can access forums, blog posts, and knowledge base articles.
- You can find documentation for Code Insight and all other Reverera products on the [Reverera Product Documentation](#) site.
- The [Reverera Learning Center](#) offers free, self-guided online videos to help you quickly get the most out of your Reverera products. You can find a complete list of these training videos in the Learning Center.

- For customers who have purchased a maintenance contract for their product(s), you can submit a support case or check the status of an existing case by making selections on the **Get Support** menu of the [Reverera Community](#).

Changes in Code Insight 2023 R3 SP1

The Code Insight 2023 R3 Service Pack (SP) 1 release contains the following enhancements and updates:

- The following new features/enhancements:
 - [New Library Refresh Service](#)
 - [Job Type Added for the Library Refresh](#)
 - [Job Type Filtering by Updated Names for Report Jobs](#)
 - [All Top-Level Inventory Reported from *.versions.toml in Gradle Version Catalog](#)
- Resolved issues described in [Issues Resolved in 2023 R3 SP1](#)

The remaining information in this document pertains to the previous Code Insight 2023 R3 release and contains no additional updates to address the service pack.

New Features and Enhancements

The Code Insight 2023 R3 release and its subsequent service packs offer new features and enhancements in the following areas:

- [Advanced Inventory Searches](#)
- [Components and Licenses](#)
- [Custom Detection Rules](#)
- [Jobs Queue](#)
- [Installation, Upgrades, and Configuration](#)
- [Project Inventory](#)
- [Project Management](#)
- [Scan Agent Plugins](#)
- [Scanning and Automated Discovery](#)
- [User Experience](#)
- [REST API Enhancements](#)

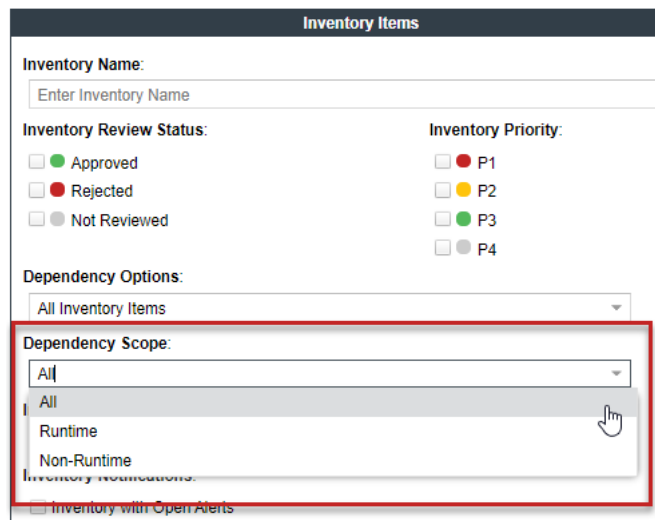
Advanced Inventory Searches

The following enhancements to the Advanced Inventory Search feature is now available. (For complete information refer to “Advanced Inventory Search Dialog” in the *Code Insight User Guide*.)

New Inventory Filters on Dependency Scope

Users can now filter inventory by its **Dependency Scope**, a new property being reported for inventory (as described in [Dependency Scope Now Reported for Inventory](#)).

Users can choose to filter to only those dependencies whose scope is **Runtime** or to those whose scope is **Non-Runtime** or choose to simply show all inventory despite their scope.

The screenshot shows the 'Inventory Items' dialog box. It has a title bar 'Inventory Items'. Below it is a text field for 'Inventory Name' with the placeholder 'Enter Inventory Name'. There are two sections of filters: 'Inventory Review Status' with radio buttons for 'Approved' (green), 'Rejected' (red), and 'Not Reviewed' (grey); and 'Inventory Priority' with radio buttons for 'P1' (red), 'P2' (yellow), 'P3' (green), and 'P4' (grey). Below these is a 'Dependency Options' dropdown menu currently set to 'All Inventory Items'. The 'Dependency Scope' dropdown menu is open, showing options: 'All', 'Runtime', and 'Non-Runtime'. A red rectangle highlights the 'Dependency Scope' dropdown and its options. A mouse cursor is pointing at the 'Runtime' option.

Also see these descriptions for other enhancements related to dependency scope:

- [Reporting of Scope for Dependencies Detected in Gradle and Maven Packages](#)
- [New Scan Profile Option to Report Non-Runtime Dependencies](#)
- [REST API Enhancements](#) for **Inventory** APIs

Components and Licenses

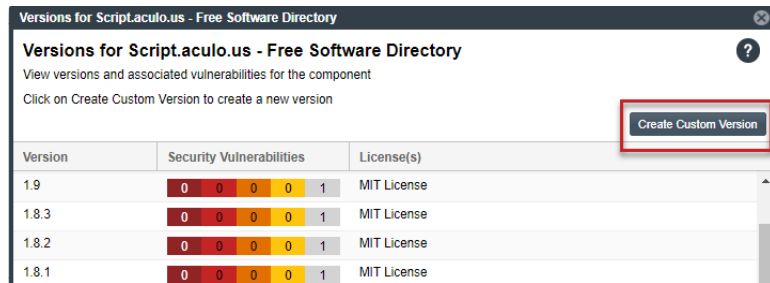
The following enhancements to component and license information in Code Insight Data Library and database are now available.

Ability to Create Custom Component Versions

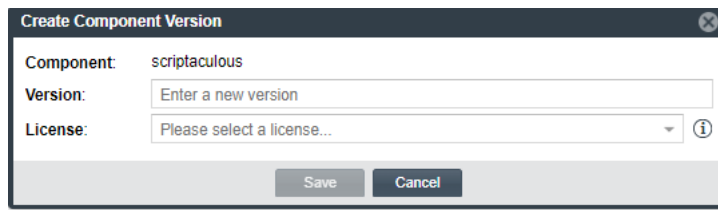
The **Versions for <component>** window now includes a **Create Custom Version** button that enables users to create a custom version for the current component.



Note - The **Versions for <component>** window is accessed from the **Components** tab in **Global Component & License Lookup** or within the context of an inventory item being edited in the **Analysis Workbench** or on the **Project Inventory** tab.



Once the button is clicked, the **Create Component Version** window is displayed, enabling the user to enter the new version and select the license with which to associate the version.



Depending on the license selected, the user can save the license as a user-preferred license so that is mapped to all future inventory created by the system across projects for the new component version.

Also see these descriptions for other enhancements related to the new feature to create custom versions:

- [Support for “user-preferred” License Mappings for Future Inventory](#)
- [REST API Enhancements](#) for **Component** APIs

For complete information about creating custom versions, refer to “Creating Custom Component Versions” in the Code Insight User Guide.

Support for “user-preferred” License Mappings for Future Inventory

During a review of the inventory after a scan, users might find that the license to which Code Insight has mapped a component version for a new inventory item is incorrect or unknown or that the item is associated with multiple licenses with no single license selected. In such cases, the reviewer must manually edit the inventory item to select the correct license. Because this same license-mapping issue can reoccur for inventory generated for the same component version during upcoming scans, users might be faced with future repeated edits system-wide.

In this release, Code Insight has introduced “user-preferred license” mapping, enabling users to select a single license to be mapped to all future inventory that the system generates for a given component version across projects. In this way, users do not have to repeat the manual mapping process for new inventory items system-wide every time scans are run.

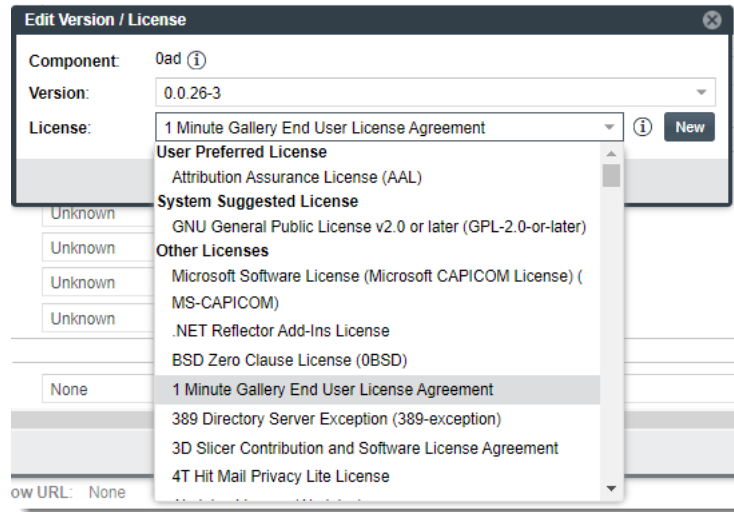
Users can have the option to save a license as the user-preferred license for a component version when they map a new license to any of the following:

- An inventory item they editing
- A component version they are editing or adding in Lookup Component
- A component version they are creating

The following enhancements in the UI have been added to support user-preferred licenses.

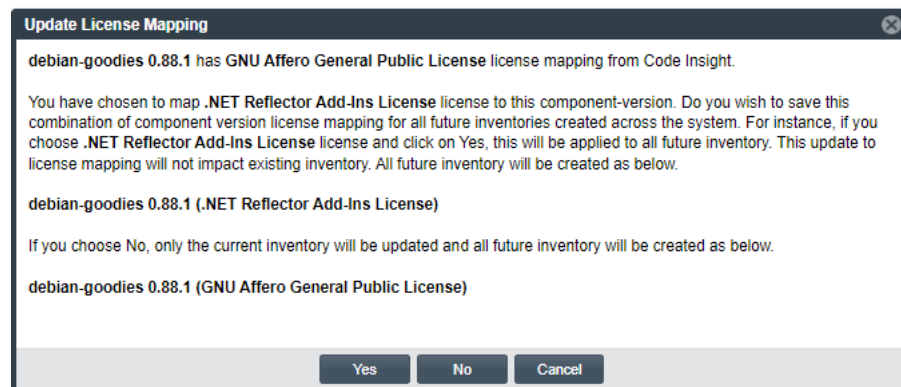
License Categories

Available licenses for a component version are now classified under certain categories on the **License** dropdown. A given dropdown can include one or more of these categories: **User Preferred License**, **System Suggested License**, **User-Preferred and System Suggested License**, and **Other Licenses**.



New “Update License Mapping” window

Depending on the license a user maps to given component version or inventory item, a version of the **Update License Mapping** window is displayed (see an example below), offering the option to save the license as a user-preferred license.



Additional Information

Also see these descriptions for other enhancements related to user-preferred licenses:

- [Ability to Create Custom Component Versions](#)
- [REST API Enhancements](#) for **Component** APIs

For more information about this new feature, refer to the “Specifying a User-Preferred License Mapping” section in the *Code Insight User Guide*.

Custom Detection Rules

The following enhancements to the custom detection rules are now available. (For complete information, see “Managing Custom Detection Rules” in the *Code Insight User Guide*.)

Support for Custom Inventory Names in Custom Detection Rules

The **Custom Detection Rule** dialog for creating or editing detection rules now provides an **Inventory Name** field to provide a custom name for the inventory item created by this rule. This name overwrites the default *component version (license)* name that the rule normally assigns to the inventory item (based on the **Component** and **License** attributes specified in the dialog).

Also see the description for API updates related to this feature in the [REST API Enhancements](#) (for **Rules APIs**).

Data Library, Library Refreshes, and Electronic Updates

The following are enhancements to the Code Insight data library, the daily Library Refresh (which generates alerts for new vulnerabilities associated with inventory), or the Electronic Update (which keeps Code Insight systems up to date with the latest data-library information).

New Library Refresh Service

(Introduced in 2023 R3 SP1) Code Insight now includes a Library Refresh service, which runs daily to keep a Code Insight instance up to date with new vulnerabilities associated with inventory across projects. This new service notifies users of new vulnerabilities associated with their projects on a daily basis so that they do not have to wait for the next regularly scheduled Electronic Update to be informed of new vulnerability threats. Users can immediately begin vulnerability investigation and remediation work as needed, thus reducing the window of possible exploitation.

About the Library Refresh

Basically, the Library Refresh is like a partial Electronic Update. While an Electronic Update performs an overall update of the Code Insight Data Library, the Library Refresh focuses on only these operations:

- Updating library tables with new vulnerability data

- Updating library tables with new mapping information defining which component versions are associated with each new vulnerability
- Generating vulnerability alerts, email notifications, and remediation tasks for project inventory affected by the new vulnerabilities (see [User Notifications of New Vulnerabilities](#)).

Additionally, because the Library Refresh is run daily, it processes only those new vulnerabilities discovered since the previous day.

User Notifications of New Vulnerabilities

Depending on Code Insight and project configuration, a Library Refresh can also notify users in the following ways when new vulnerabilities are discovered:

- Generate alerts in the UI for inventory items associated with the newly discovered vulnerabilities. (For more information about alerts, refer to the “Managing Security Vulnerability Alerts” section in the *Code Insight User Guide*.)
- Issue emails to the project owners whose inventory is affected by any of the vulnerabilities.
- Create remediation tasks for those inventory items that are rejected due to associated vulnerabilities that exceed policy thresholds.

Refresh Schedule

The Library Refresh runs daily at 12 am.

More Information

For complete information about the Library Refresh service, refer to “Managing the Daily Check for New Security Vulnerabilities” in the *Code Insight Installation & Configuration Guide*.

Jobs Queue

The following enhancements to the Code Insight **Jobs** queue are now available. (For complete information, refer to “Monitoring the Code Insight Jobs Queue” in the *Code Insight User Guide*.)

Job Type Added for the Library Refresh

(Available in 2023 R3 SP1) The **Jobs** queue has been updated with a **Job Type** called **Library Refresh** to support the new Library Refresh service. (For more information, see [New Library Refresh Service](#).)

Job Type Filtering by Updated Names for Report Jobs

(Available in 2023 R3 SP1) **Job Type** filters corresponding to the job names updated in 2023 R3 for the generation of custom and standard reports are now available (see also [Changes to Job Types for Reports](#)). These filters include **Report - Project Report**, **Report - Notices Report**, and **Report - Audit Report** for standard reports and the filter format **Report - <customReportName>** for custom reports.

Similarly, the **Get jobs details based on filters** API has been updated to support **jobType** filters based on the updated job names for report generation. See the description for this enhancement under **Jobs** in [REST API Enhancements](#).

Support for Project Imports and Remote Scans

The **Jobs** queue has been updated to support project imports and the processing of remote-scan results in Code Insight as jobs. For more information, see [Project Imports Now Running as Background Jobs](#) and [Processing of Remote-Scan Results in Code Insight Now a Background Job](#), respectively.

Changes to Job Types for Reports

The **Job Type** for the custom reports has been updated from the general term **Custom Report** to the actual name of the custom report. Additionally, the **Job Type** for the various standard and custom reports are now prefaced with the term “Report -”, as in **Report - Project Report**, **Report - Audit Report**, **Report - <customReportName>**, and so on.

Jobs			
Jobs			
Job ID ↓	Job Type	Project Name	Status
451	Remote Scan	microsoft_transitive_scan_b54	Complete
450	Report - Project Report	microsoft_transitive_scan_b54	Complete
449	Remote Scan	microsoft_transitive_scan_b54	Complete
448	Remote Scan	microsoft_transitive_scan_b54	Complete
447	Remote Scan	microsoft_transitive_scan_b54	Complete
446	Report - Notices Report	microsoft_transitive_scan_b54	Complete
445	Report - Audit Report	microsoft_transitive_scan_b54	Complete
444	Report - Project Report	microsoft_transitive_scan_b54	Complete

Also see the description of this enhancement for the **Jobs** API in [REST API Enhancements](#).

New Column for Scan Server Name

The **Jobs** queue now includes a **Scan Server** column, which lists the Scan Server alias for each **Project Scan** and **Project Re-Scan** job type. For all other jobs (that is, jobs performed on the Code Insight Core Server), the value is **N/A**.

Jobs					
Jobs					
Job ID ↓	Job Type	Project Name	Status	Scan Server	Trig
105	Project Scan	NPM	Completed	scanner	adm
104	Project Scan	awsk_1	Completed	scanner	adm
103	Project Copy	assss	Completed	N/A	trup
102	Remote Scan	testbranch	Completed	N/A	trup

Also see the description of the API update related to this **Jobs** queue enhancement in [REST API Enhancements](#) (for **Jobs** APIs).

Installation, Upgrades, and Configuration

The following enhancements have been added to the Code Insight installation, upgrade, or configuration experience.

Support for a Reverse Proxy

User can specify a reverse proxy to mask the actual Code Insight Core server host and port in SSL communications and in all communications with outside servers. For complete information, refer to “Using a Reverse Proxy for Code Insight” in the *Code Insight Installation & Configuration Guide*.

Project Inventory

The following enhancements to the project inventory are now available.

Dependency Scope Now Reported for Inventory

The scope of a reported dependency is now shown in the new **Dependency Scope** field listed in the inventory details on the **Project Inventory** tab and in the **Analysis Workbench**. Its value is reported as follows:

- Inventory items that are dependencies required at runtime have **Runtime** value.
- Items that are dependencies not required at runtime have a **Non-Runtime** value.
- All other items have an **N/A** value. These items include top-level inventory, dependencies for which Code Insight does not currently support the reporting of scope, and migrated inventory for which a scan has not been run.

Currently, Code Insights reports the scope for those dependencies detected in Gradle and Maven packages only.

jmock 1.1.0 [Dependency of android 4.1.1.4] (BSD)

Recall ItemEdit ItemView History

Confidence: ■ ■ ■

Vulnerabilities: No

Inventory Details

Component Details

Name: jmock 1.1.0 [Dependency of android 4.1.1.4] (BSD)

Description: jMock is a library for testing Java code using mock objects
Homepage: <http://www.jmock.org>
Sources: <http://cvs.jmock.codehaus.org/browse/jmock/jmock>
License history:
No information - for versions: 20031129.200437, 20031129.200113, 20031129.200113
BSD - for versions: 1.2.0, 1.1.0
License links:
No information - for versions: 20031129.200437, 20031129.200113, 20031129.200113

URL: <https://repo1.maven.org/maven2/jmock/jmock>

Provenance: Originated in this project

Dependency Scope: Non-Runtime

Disclosed: No

Also see these descriptions for other enhancements related to dependency scope:

- [Reporting of Scope for Dependencies Detected in Gradle and Maven Packages](#)
- [New Scan Profile Option to Report Non-Runtime Dependencies](#)
- [New Inventory Filters on Dependency Scope](#)
- [REST API Enhancements](#) for **Inventory** APIs

Project Management

The following enhancements to Code Insight project management are now available.

Project Imports Now Running as Background Jobs

A project import is now queued as a **Project Import** job in the **Jobs** queue so that it runs in the background without contention with other operations. Users can monitor the status of the job and, when needed, filter the queue on the **Project Import** job type to see all other project imports in the queue.

Jobs				
Jobs				
Job ID ↓	Job Type	Project Name	Status	Scan Serv
471	Project Branching	as	Completed	Scanner
470	Project Import	trpTest	Completed	N/A
469	Remote Scan	DockerPluginTest2023B3	Completed	N/A

For more information about this feature, refer to the following in topics in the *Code Insight User Guide*:

- “Monitoring Code Insight Jobs Queue”
- “Exporting and Importing Project Data”

Also see the description of the API update related to this **Jobs** queue enhancement in [REST API Enhancements](#) (for **Jobs** APIs).

Scan Agent Plugins

The following enhancements to Code Insight remote scanning are now available.

Processing of Remote-Scan Results in Code Insight Now a Background Job

For scans performed by the generic and Docker Images scan-agent plugins only, the phase in which the scan results are processed in Code Insight for a given project is now run as a background job that users can track in the **Jobs** queue. During such scans, the remote plugin sends the scan results in JSON format to Code Insight as a .txt file to be stored temporarily in Code Insight. Once the results are successfully sent, the job ID for processing the results in Code Insight is created and the job is added to the job queue. (The ID is also returned to the plugin to indicate that the results were successfully sent.) Users can track the progress of the results-processing as a **Remote Scan** job in the **Jobs** queue. (Progress is shown as **Active**, **Scheduled**, **Completed**, or **Failed**.) When Code Insight completes the processing, the temporary file is removed.

The **Queued On** value for a **Remote Scan** job is the time when the scan results from the scan-agent plugin were successfully sent to Code Insight for processing.



Note • As soon as Code Insight receives results from a scan-agent plugin other than the generic plugin or the Docker Images plugin, the results are processed in the foreground (just as they have been processed in all previous releases).

Jobs						
Jobs						
Show jobs for: 15 Days ?						
Job ID ↓	Job Type	Project Name	Status	Scan Server	Triggered By	Queued On
102	Remote Scan	testbranch	Completed	N/A	trupti	08/09/2023 at 09:38 AM
101	Remote - Native Remote	testbranch	Completed	N/A	trupti	08/09/2023 at 09:37 AM

For more information, refer to the following topics in the *Code Insight User Guide*.

- “Monitoring Code Insight Jobs Queue”
- “Support for Processing Remote Scan Results in the Background”

Also see the description of the API update related to this **Jobs** queue enhancement in [REST API Enhancements](#) (for **Jobs** APIs).

Support for Running Docker Images Plugin Scan with Command-Line Options

In previous releases, users were required to run a Docker Images agent scan with a properties file that they manually updated before the executing scan.

Users now have the option to run the scan using command-line options instead of a properties file. For more information, refer to “Running a Docker Images Plugin Scan with Command-Line Options” in the *Code Insight Plugins Guide*.

Scanning and Automated Discovery

This release includes the following enhancements to Code Insight scans and the Automated Analysis techniques used to discover and report inventory during scans.

All Top-Level Inventory Reported from *.versions.toml in Gradle Version Catalog

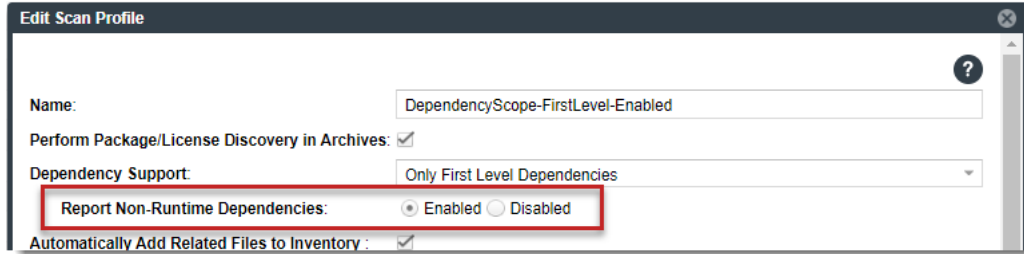
(Available in 2023 R3 SP1) During the scan of a Gradle project with a version catalog in a *.versions.toml file, top-level inventory items are now created for all items in the libraries section of the file (with the exception of those items also referenced in build.gradle or build.gradle.kts). These top-level inventories are created with their **Part of Product** usage property set to **No**.



Note • These scan results occur when the scan profile is configured to detect **Only First Level Dependencies** or **All Transitive Dependencies**.

New Scan Profile Option to Report Non-Runtime Dependencies

A new setting in the scan profile specifies whether the scan should report, along with top-level inventory, runtime dependencies only or both runtime and non-runtime dependencies. (Runtime dependencies are required during application runtime; non-runtime dependencies are not.)



This setting is impacted by the existing **Dependency Support** profile setting, which determines whether dependencies are reported and, if so, the type of dependencies that are reported. For more information about scan profiles, refer to “Managing Scan Profiles” in the *Code Insight Installation & Configuration Guide*.

Also see these descriptions for other enhancements related to dependency scope:

- [Reporting of Scope for Dependencies Detected in Gradle and Maven Packages](#)
- [Dependency Scope Now Reported for Inventory](#)
- [New Inventory Filters on Dependency Scope](#)
- [REST API Enhancements](#) for **Inventory** APIs

Reporting of Scope for Dependencies Detected in Gradle and Maven Packages

The scope of a dependency is now reported in the new **Dependency Scope** field listed in the inventory details on the **Project Inventory** tab and in the **Analysis Workbench** (see also [Dependency Scope Now Reported for Inventory](#)). This field indicates whether an inventory item that is a dependency is a **Runtime** or **Non-Runtime** dependency. (Inventory items to which scope does not apply have an **N/A** scope value.)

Currently, Code Insight reports scopes for dependencies found in only the following manifest files during scans. (Dependencies not found in these files show **N/A** for **Dependency Scope**.)

- build.gradle and build.gradle.kts files in Gradle development ecosystems
- .jar, pom.xml, and .pom files in Maven development ecosystems

For more information about the scope-reporting for dependencies, refer to “Automated Analysis” in the *Code Insight User Guide*.

Also see these descriptions for other enhancements related to dependency scope:

- [New Inventory Filters on Dependency Scope](#)
- [New Scan Profile Option to Report Non-Runtime Dependencies](#)
- [REST API Enhancements](#) for **Inventory** APIs

Reporting of Top-Level Inventory in Debian Packages

Project scans now report top-level inventory found in Debian (.deb) packages. For more information, refer to “Automated Analysis” in the *Code Insight User Guide*.



Reporting of Transitive Dependencies in cocoapod Packages

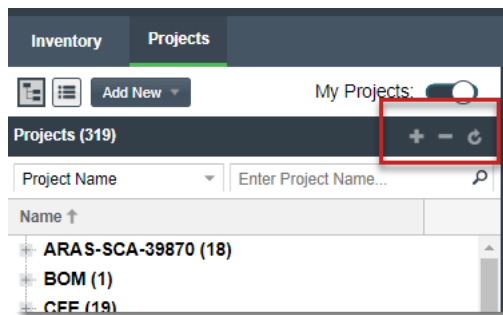
Project scans now report transitive dependencies found in cocoapod packages. For more information, refer to “Automated Analysis” in the *Code Insight User Guide*.

User Experience

This release includes the following enhancement to overall user experience.

Ability to Expand or Collapse All Folders in the Project Tree

Buttons are now available that expand  or collapse  all folders in the project tree. By default, all folders are collapsed. These buttons enable users to more easily navigate project trees that contain a large number of folders and files within folders.



REST API Enhancements

This release includes the following changes to the Code Insight REST interface.

- [New APIs](#)
- [Updates to Existing APIs](#)

New APIs

No new APIs have been added in this release.

Updates to Existing APIs

The following section describes updates that have occurred to existing APIs in this release.

Table 1 • Updates to Existing APIs

Resource	API Name/Endpoint	Method	Function Change Description
Component	Get Component version details /components/versions/{Id}	GET	The property userPreferredLicense has been added to details for each license associated with a given component version in the response. The true or false response for the property indicates whether the license is <i>user-preferred</i> —that is, selected by a user to be mapped to all future inventory generated by the system for the version. For more information, see Support for “user-preferred” License Mappings for Future Inventory .
	Get Component /components/{componentId}		

The following shows this property in an example **Get Component version details** response:

```
{
  "data": {
    "id": 1122344,
    "name": "2.5.1",
    "componentId": 3243434,
    "licenses": [
      {
        "id": "1313",
        "name": "BSD-Style License",
        "userPreferredLicense": true
      }...
    ]
  }
}
```

The following shows this property in an example **Get Component** response:

```
"versionList": [
  {
    "id": 16016865,
    "name": "3.6.1p2",
    "custom": false,
    "versionLicenses": [
      {
        "id": 1916,
        "name": "389 Directory Server Exception",
        "shortName": "389-exception",
        "priority": "P3",
        "url": "https://spdx.org/licenses/389-exception.html",
        "description": "",
        "familyName": null,
        "custom": false,
        "externalId": "",
        "userPreferredLicense": true
      }...
    ]
  }
]
```

Table 1 ▪ Updates to Existing APIs (cont.)

Resource	API Name/Endpoint	Method	Function Change Description
Inventory	Get the inventories of file/folder Ids /inventories/search	GET	The inventory details returned in response now list the property dependencyScope , indicating whether the inventory item is a dependency required at runtime (Runtime) or not required at runtime (Non Runtime).
	Get details of an inventory /inventories/{inventoryId}		The N/A value indicates that the inventory item cannot be classified as a runtime or non-runtime dependency. Such items include top-level inventory, dependencies for which Code Insight does not currently support the reporting of scope, and migrated inventory for which a scan has not been run.

Table 1 ▪ Updates to Existing APIs (cont.)

Resource	API Name/Endpoint	Method	Function Change Description
Jobs	Get jobs details based on filters /jobs	GET	<p>Job details in the response now include a new server property that provides the internal ID and alias of the server that performed the job.</p> <p>Note that for scans, the scan server's alias is listed. For jobs performed on the Code Insight Core Server, the alias is Core Server.</p> <pre> "server": { "id": 2, "aliasName": "scanner36" }, ... "server": { "id": 1, "aliasName": "Core Server" }, ... </pre>
	Get job details based on the jobId provided /jobs/{jobId}		<p>The jobType property for a custom-report job in the response now shows the actual name of the custom report. Additionally, the jobType value for every report job—custom or standard—is now prefaced with the term “Report -” (for example, “Report - Project Report”).</p> <p>(Available in 2023 R3 SP1) The Get jobs details based on filters API supports jobType filters that correspond to the updated names for the standard and custom report jobs. Note that, for custom report filters, users must enter the jobType parameter as the string Report - <customReportName>, providing the actual report name and using a space before and after the hyphen.</p>
			<p>Remote Scan has been included as a new jobType. A job of this type is available in the Jobs queue as soon as the scan results from a scan agent plugin have been successfully sent to Code Insight for processing. The processing is then run in the background and tracked in the queue.</p> <p>Project Import has been included is included as a new jobType so that project imports can run in the background and be tracked on the Jobs queue.</p>

Table 1 ▪ Updates to Existing APIs (cont.)

Resource	API Name/Endpoint	Method	Function Change Description
Project API	Get Project Inventory /project/inventory/{projectId}	GET	<p>The inventory details returned in response now list the property dependencyScope, indicating whether the inventory item is a dependency required at runtime (Runtime) or not required at runtime (Non Runtime).</p> <p>The N/A value indicates that the inventory item cannot be classified as a runtime or non-runtime dependency. Such items include top-level inventory, dependencies for which Code Insight does not currently support the reporting of scope, and migrated inventory for which a scan has not been run.</p>
	Get Rules /rules	GET	<p>The response includes the new inventoryName property that has been added to the custom rule definition. This property shows the custom name that is given to the inventory items created by the rule. The custom name overwrites the default <i>component version (license)</i> name that the rule normally assigns to the inventory item.</p> <p>If no inventoryName value has been defined, the property shows N/A. Inventory items created by the rule are assigned the default name when no custom name is specified.</p>
Rules	Get Rule By Id /rules/{ruleId}		
	Create Rule /rules	POST	
	Update Rule /rules	PUT	
			<p>The request includes the new inventoryName property that has been added to the custom rule definition. This property enables you to specify a custom name (or update an existing one) that will be given to any new inventory items created by the rule. The custom name overwrites the default <i>component version (license)</i> name that the rule normally assigns to the inventory items it creates.</p> <p>If no inventoryName value is defined for the rule, the property shows N/A whenever the rule details are retrieved. Inventory items created by a rule are assigned the default name when no custom name is specified.</p>

Resolved Issues

The following issues have been resolved in Code Insight 2023 R3 and its subsequent service packs:

- [Issues Resolved in 2023 R3 SP1](#)
- [Issues Resolved in 2023 R3](#)

Issues Resolved in 2023 R3 SP1

The following issues were resolved in the Code Insight 2023 R3 SP1 release.

Issue	Resolution Notes
SCA-49480	The issue with a Project Copy creating duplicate inventory and causing subsequent scans to fail has been resolved.
SCA-49795	In some cases, multiple Project Copy operations were being allowed to perform simultaneously on the same Code Insight instance. This issue has been resolved so that only one Project Copy can run in Code Insight at any given time.
SCA-49844, SCA-49845	Previously, file names that included a backslash (\) were not handled properly if the file name was being displayed with a forward slash instead of a backslash. The reversed slash caused a mismatch in files and file counts between the source and target projects during a Project Copy. Ultimately the scan on the target project would fail. This issue has been resolved.
SCA-49988	Previously, Job Type filters had not been updated to match the job names revised in 2023 R3 for the generation of custom and standard reports. In this release, Job Type filters corresponding to the revised job names for all reports are now available. (See also Job Type Filtering by Updated Names for Report Jobs and the Jobs resource in REST API Enhancements .)

Issues Resolved in 2023 R3

The following issues were resolved in the Code Insight 2023 R3 release.

Issue	Resolution Notes
SCA-42458	New buttons to expand or collapse all folders in a project tree have been added to address usability issues with navigating a large number of folders and projects. See Ability to Expand or Collapse All Folders in the Project Tree .
SCA-43664	Issues when attempting to use a CNAME to mask the Code Insight Core Server hostname have been resolved with Code Insight's new support for reverse proxies. The reverse proxy enables users to successfully mask the actual Core Server host and port in SSL communications and in all communications with outside servers. See Support for a Reverse Proxy .
SCA-43874	Scan results now correctly report the MIT license for inventory associated with NPM files.
SCA-46028	Source Code Management support for GitLab and GitHub repositories on Linux and Windows has been fully verified. The <i>Code Insight Installation & Configuration Guide</i> reflects support for both platforms.

Issue	Resolution Notes
SCA-46489, SCA-44039	<p>The <i>Code Insight Installation & Configuration Guide</i> now contains instructions for configuring the new skip.post.pdl.vul.processing variable in the PAS_GLOBAL_PROPERTIES table to disable or enable the post-Update phase in Electronic Updates. (The variable was introduced in Code Insight 2023 R2 SP1.)</p> <p>The post-Update phase processes new vulnerabilities against inventory to create alerts and tasks for impacted inventory and trigger emails to project owners. This phase can take anywhere from under 24 hours to 5 days or more to complete depending on the amount of inventory data processed and the impact of new security vulnerabilities on the inventory. The variable provides users the option to skip this phase.</p>
SCA-48051	<p>Previously, when a folder in the project tree contained a large number of projects, creating a project in that folder project could take considerable time. Even though the creation eventually succeeded, a blank Error pop-up could continue to display. Performance improvements to the project-creation process now result in faster creation times and no blank pop-up.</p>
SCA-43387	<p>The following Code Insight component has been upgraded to remediate security vulnerabilities associated with the component:</p> <p>Apache tika parser 1.26 upgraded to 1.28.5</p>
SCA-48450	<p>The Docker Images scan-agent plugin failed with the following error when it attempted to scan a deb file:</p> <p>java.lang.ClassNotFoundException: org.tukaani.xz.XZInputStream</p> <p>The plugin now successfully scans such a file.</p>
SCA-48469, SCA-48346, SCA-44715	<p>The Jenkins, Docker Images, and generic scan-agent plugins are now properly verifying a user's project permissions before starting a scan. An appropriate error message is logged if the user does not have proper permissions.</p>
SCA-48470	<p>Previously, a scan performed by the generic or Docker Images scan-agent plugin could log a failed status but still show results in the Code Insight project. This issue occurred because the scan agent timed-out while waiting for a response from Code Insight indicating that the scan results had been successfully processed. This issue has been resolved by adding the results-processing as a job on the Jobs queue in Code Insight and running the job in the background. See Processing of Remote-Scan Results in Code Insight Now a Background Job.</p>
SCA-48497	<p>During the migration of a Code Insight project from v6 to v7, the Audit Data Migration tool could fail to import data from the usage fields in the project. This issue has been resolved.</p>
SCA-48706	<p>In Code Insight 2023 R2, the execution of Python scripts to call Code Insight API (or the execution of equivalent calls through Swagger) returned a status code of 500. This has been resolved with an increase of the maxHttpHeaderSize to 16384. This increase has been updated in the <i>Code Insight Installation & Configuration Guide</i>.</p>
SCA-48906	<p>After the migration of a TFS server to a new instance, the TFS Source Code Management plugin was not cleaning up old workspaces due to a command that was not properly verifying TFS workspace existence. This issue has been resolved.</p>

Issue	Resolution Notes
SCA-49176	When users attempted to synchronize Code Insight with a Git Source Code Management instance identifying a repository URL whose branch and tag had the same name, the message “You are not currently on a branch” was displayed. This issue has been resolved so that a URL containing a branch and tag of the same name no longer throws an error during synchronization.

Known Issues

The following are current known issues in Code Insight. The issues are organized as follows:

- [All-Project Inventory View](#)
- [Automated Workflow for Inventory Review/Publication](#)
- [Custom Detection Rules](#)
- [Data Library, Library Refreshes, and Electronic Updates](#)
- [Export and Import](#)
- [Installation, Upgrades, and Configuration](#)
- [Inventory History](#)
- [Jobs Queue](#)
- [Manual Codebase Analysis](#)
- [Performance](#)
- [Project Inventory](#)
- [Project Management](#)
- [Project Reporting](#)
- [REST APIs](#)
- [Scan Agent Plugins](#)
- [Scanning and Automated Discovery](#)
- [Source Code Management \(SCM\) Support](#)
- [Vulnerability Suppression/Unsuppression](#)
- [Web UI](#)

All-Project Inventory View

The following are known issues in the **Inventory** view, which shows inventory across all Code Insight projects.

SCA-34403: Inventory details slide-out panel opening twice

When a user clicks an inventory item in the **Inventory** view, the panel showing the inventory's read-only details appears briefly on the right side of the view and then properly slides out from the right.

Workaround: None exists.

Automated Workflow for Inventory Review/Publication

The following are known issues with the automated workflow for inventory review and publication.

SCA-11193: Incorrect URL(s) in email notifications

In cases where Code Insight is running on a server that uses multiple IP addresses (for example, a server that has both a wired and wireless active network connection), the Core Server address cannot be accurately resolved. As a consequence, users can encounter an incorrect URL in the email notification received from Code Insight. This issue is most often seen if the Code Insight core server is configured as "localhost" instead of a full IP address.

Workaround: None exists.

Custom Detection Rules

The following are known issues when managing custom detection rules.

SCA-48572: No error message indicating "Inventory Name" value in rule exceeds limit

When you attempt to save a custom detection rule with an **Inventory Name** value that exceeds its 255-character limit, the rule is not saved, and no error message is displayed to describe the problem.

Workaround: If you enter a value for **Inventory Name**, ensure that has no more than 255 characters.

SCA-48564: Changes to "Inventory Name" in custom detection rules not applied to inventory during rescans

When you update an existing value in the **Inventory Name** field for a custom detection rule, a rescan or forced full rescan does not apply the change to the inventory name.

Workaround: Create a new project and perform a fresh scan so that the name change is correctly applied to the inventory item.

Data Library, Library Refreshes, and Electronic Updates

The following are known issues related to the Code Insight data library, the daily Library Refresh, which reports new vulnerabilities associated with inventory, or the Electronic Update, which keeps Code Insight systems up to date with the latest data-library information.

SCA-43625: Server shutdowns when simultaneous custom-component indexing and searching occur

As of 2022 R4, Code Insight starts the background process of indexing a custom component in the Code Insight data library as soon as the component is created or updated. Even though indexing takes only a half minute or less per component, a server shutdown can occur if you attempt to perform a component search against the data library while multiple components are being simultaneously indexed.

Workaround: Wait until the indexing process is finished for all components before you use the Global Component & License Lookup feature or the **Component Search** REST API to search components in the data library.

SCA-43568: Sequential creation of multiple custom components with similar names resulting in incorrect component search counts and pagination

As of 2022 R4, Code Insight starts the background process of indexing a custom component in the Code Insight data library as soon as the component is created or updated. If multiple custom components with similar names are sequentially created/updated and indexed in the background, the search results for these components might show incorrect search counts and pagination.

Workaround: After the custom-component updates are indexed, run an Electronic Update to fix the indexes.

SCA-40194: Duplicate inventory issues for MIT-related components

The Code Insight MIT data-library update does not fix inventory items with names that include multiple licenses separated by commas (instead of ORs), as shown in this example:

```
jquery (MIT, MIT License)
```

On a rescan, duplicates can be created for such inventory items:

```
jquery (MIT, MIT License)
```

```
jquery (MIT)
```

Two possible workarounds are available.

Workaround 1: Before starting the rescan, select the option **On data import or rescan, delete inventory with no associated files** on the **Manage Project > Edit Project > General** tab accessed from the project's **Summary** tab. This option deletes the original inventory item as long as it is system-generated.

Workaround 2: Manually delete the original inventory item in the **Analysis Workbench** by right clicking the item and selecting **Delete inventory**. You must repeat this step for each such inventory item.

SCA-31562: Component license remapping issues from MIT-Style to MIT for inventories

Remapping Issues have occurred once the latest Electronic Update (available from Code Insight 2021 R4 and later) has been run. These issues involve the remapping of licenses from MIT-Style to MIT for inventories. The next two sections combined illustrate a typical remapping issue.

Before running the Electronic Update available at the release of Code Insight 2021 R4 and later:

The following inventory mapping existed in inventory:

concurrent-ruby 1.1.9 (MIT License)

This component was mapped with License id 744.

After running the Electronic Update available at the release of Code Insight 2021 R4 and later:

The inventory item was remapped as follows:

concurrent-ruby 1.1.9 (MIT-Style)

The license short name had been changed (in this example, from MIT License to MIT-Style). However, the mapped license ID remained 744.

Ideally this component should be remapped to MIT, which is License id 7.

Workaround: Follow these steps:

1. Click **Inventory** on the main Code Insight window to open the **Inventory** view, showing inventory across projects.
2. Switch from **My Projects** to **All Projects**.
3. Search for the inventories containing the string *(MIT-Style)*.
4. Locate the **Possible Licenses** value for a given inventory. If this value is **MIT** (Id 7) *and* the term *MIT-Style* is in the inventory name or is the value of **Selected License**, then an incorrect license remapping has been performed for this specific inventory item. One incorrect license remapping is a possible indicator of other incorrect remappings.
5. Run the Code Insight cleanup SQL script to correct the license mappings for the inventory in your Code Insight system. (To obtain this script, download the codeinsight-MITCleanupPackage archive from the Product and Licensing Center, and extract the script and its instructions.)

Export and Import

The following are known issues with the Code Insight project export and import functionality.

SCA-3222: Import overrides inventory details

Importing the same inventory into a project that already contains inventory can cause some details to be overwritten or blanked out. If duplicate inventory (by associated repository item ID) is encountered during the import process, inventory details are overwritten with data from the export data file.

Recommended: Perform an export of the project prior to importing into the project in case you need to return to the original project state.

SCA-21295: Import of Detection Notes over 16 MB generates an error

When Code Insight uses a MySQL database, an error can occur during a project import if the source project's **Detection Notes** content exceeds 16 MB. The import process generates an error message and continues processing the inventory but does not import the notes.

Workaround: Ensure that only a single network interface controller is enabled on the core server running Code Insight. As an added measure, configure the core server using a numerical IP address instead of a “localhost”.

Installation, Upgrades, and Configuration

The following are known issues with a Code Insight installation or upgrade and configuration.

SCA-48759: Longer Tomcat startup during migration to 2023 R3 due to new column creation

In Code Insight 2023 R3, a new column was added to the `pse_inventory_groups` table to support the reporting of dependency scope. During a migration to Code Insight 2023 R3 or later, the creation of this column (performed through Liquibase) can slow down the initial Tomcat startup if the number of inventory items is large. (For example, if the inventory count is 15,000, the initial Tomcat startup can take 30 minutes to complete.)

Workaround: None exists.

SCA-35918: Upgrades to Code Insight possibly more time-consuming than previous upgrades

Upgrading to Code Insight might take longer than previous upgrades, especially if the number of inventory items in your Code Insight system has increased since the last upgrade. For example, an upgrade for a system with about 1 million inventory items can now take around 15-20 minutes, which might be longer than your previous upgrades. The extra time needed for the upgrade is due the **Inventory History** feature (introduced in 2021 R3), which requires that the inventory items for all projects be processed for inclusion in the history.

Note, however, that once an inventory item is included in the history, it does not need to go through this initialization process in subsequent upgrades.

Workaround: None exists. If you have any concerns about the time taken for this process, contact Revenera Support for assistance.

SCA-15952: Installer unable to install embedded JRE on some Windows 10 instances

Running the installer on some (but not all) Windows 10 systems results in an “Installation: Successful null” message and does not completely populate the `<INSTALL_ROOT>\jre` directory.

Workaround: Should you encounter the above error, install the JRE manually. Download JRE 8u192. Configure the `JAVA_HOME` and `JRE_HOME` variables in `catalina.*` to point to the newly installed JRE.

SCA-1652 / SCA-5812: Deleted or disabled users still visible in the Web UI

Users who are deleted from the LDAP server or disabled in LDAP still appear on the **Users** page in the Code Insight Web UI and in some selection lists, such as for projects.

Workaround: None exists. However, deleted or disabled users are blocked from logging into the application and attempting to add one of these users results in an error.

If this blocking is not sufficient or doable, contact Reverera Support for information about executing a database SQL script that can help to complete the index process within the expected time. The script must be run *before* the Electronic Update is started. (To contact Reverera Support, access the **Get Support** menu in the Reverera Community at <https://community.reverera.com>.)

Inventory History

The following are known issues with the Inventory History feature.

SCA-36420: Inventory URL and Description attributes shown as updates in Inventory History without their being modified

After an initial transaction is performed against an inventory item (such as editing or viewing the item), entries for the **URL** and **Description** properties are displaying in the **Inventory History** window even though these properties were never modified as part of the transaction. These two initial entries will remain in the history. However, any future transactions against the inventory item will not create an update entry for the **URL** or **Description** property unless the value for either property has actually changed.

Workaround: None exists.

Jobs Queue

SCA-50496: Possible deadlock between a scheduled Electronic Update or Library Refresh and another scheduled job

If an Electronic Update or Library Refresh is added to the **Jobs** queue when a job that normally waits for an Electronic Update or Library Refresh (such as a **Project Import** or **Update Notices** job) is currently running, the **Electronic Update** or **Library Refresh** job is placed in a **Scheduled** status. If, at the same time, *another such job (that is, one that normally waits for an Electronic Update or Library Refresh) is already scheduled*, the **Electronic Update** or **Library Refresh** job should take precedence over this job and run next once the current job finishes. Instead, the Electronic Update or Library Refresh incorrectly waits for this other job, which, in turn, is correctly waiting on the Electronic Update or Library Refresh to run, thus causing a deadlock.

Workaround: Fail the scheduled Electronic Update or Library Refresh by running the following database query:

```
Update PSE_SCHEDULER_TASKS set STATE_='FAILED' where ID_=<PDL/LR JOB ID>
```

where <PDL/LR JOB ID> is the **Job ID** of the Electronic Update or Library Refresh in the queue.

Once this job fails, the other scheduled job becomes active. Once it completes, you can add the Electronic Update or Library Refresh to the **Jobs** queue.

Manual Codebase Analysis

The following are known issues with manual codebase analysis in the **Analysis Workbench**.

SCA-46104: Not able to retrieve Advanced File Search results when using same criteria but with distinct values and AND logic

The Advanced File Search feature does not retrieve the expected results when you define a filter using multiple criteria that are the same (but with a distinct value for each criterion) and apply AND logic to the criteria. Files known to meet all the specified criteria are not listed in the **File Search Results** pane.

SCA-44366: Error thrown when navigating file search results

When you use the **Enter search string...** field at the top of the **Codebase Files** pane to search for files by name, you can use the Next or Previous button adjacent to the field to navigate to the search results highlighted in the codebase tree. However, if you click these buttons a rapid pace, you can generate an error (although the UI does not hang).

Workaround: Click the buttons at a slower pace.

SCA-41440: “Show File Evidence” right-click option on “File Search Results” pane not working at node, folder, and sub-folder levels

When you right-click an alias node, codebase node, folder, or sub-folder in the **File Search Results** pane in the **Analysis Workbench**, and then select **Show File Evidence**, the **Evidence Details** tab on the right displays the message “No Evidences found”.

However, when you select **Show File Evidence** at the file level in the **File Search Results** pane, the evidences properly are listed on the **Evidence Details** tab as expected.

This behavior occurs whether the files were scanned by a Scan Server or a scan-agent plugin.

Workaround: None exists.

SCA-41964: Empty results when Advanced Search with “File Path” criterion attempts to fetch 2000 or more results

An **Advanced Search** using the **File Path** criterion can produce empty results in the **Analysis Workbench** if the search attempts to retrieve 2000 or more results. This issue can occur whether searching a file system scanned by a remote scan agent or a codebase scanned by a Scan Server.

This issue does not occur when the search fetches less than 2000 results.

Workaround: None exists.

SCA-27011: Advanced Search based on low confidence inventory not working

In the **Analysis Workbench**, an **Advanced Search** for files associated with inventory that has a low confidence level is returning incorrect or no results.

Workaround: None exists.

SCA-22398: Licenses not highlighted even though evidence exists

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 in the file.

Workaround: None exists.

SCA-22308: “Email/URLs” evidence truncated

In some cases after running a scan, the **Email/URLs** evidence on the **Evidence Details** tab in **Analysis Workbench** is truncated.

Workaround: None exists.

SCA-10414: Associated files not displayed when user adds more than 37K files to inventory

When more than 37K files are added to an inventory item, the associated files are not displayed on the **Associated Files** tab.

Workaround: Right-click the inventory item and select **Show Inventory Files**. The content on the **File Search Results** pane in **Analysis Workbench** is filtered to the associated files for the inventory item.

Performance

The following are known issues with Code Insight performance.

Performance slower with MySQL 8 than with MySQL 7

Codebase scans and updates to the Code Insight data library are slower when Code Insight uses the MySQL 8 (5.8) database compared to when it uses MySQL 7 (5.7).

Project Inventory

The following are known issues with the review process for Code Insight project inventory.

SCA-44107: Unable to delete an inventory item with a large number of associated files

Attempts to delete an inventory item associated with a large number files (50KB or more) can fail.

Workaround: None exists.

SCA-44077: Deletion of a top-level inventory item causing deletion of dependency inventory

When a user deletes a top-level inventory item, all of its dependent inventory items are also deleted.

Workaround: None exists.

SCA-41263: License text shown twice in As-Found License Text field in Analysis Workbench

In the **Analysis Workbench**, the text for a license can be repeated twice for some components (such as the component glob) when the license file contains more than one license.

Workaround: None exists.

SCA-11520: Policies not applied on rescan of a project

The triggering event for applying policy to project inventory is “Publish” (not scan). Policies are applied during the initial scan if the default setting **Automatically publish system-created inventory items** is selected, but policies are not applied during a *rescan* because inventory is not re-published. This behavior is in place to avoid inadvertent overriding of inventory status due to a change in policy by another user.

Workaround: To apply policy, first recall all inventory and rescan with **Automatically publish system-created inventory items** enabled.

Project Management

The following are known issues with project management in Code Insight.

SCA-41957: Project Copy performance slower when the Code Insight database resides on a separate machine

Processing time for Project Copy increases when the Code Insight database resides on a machine different from the machine where the Core Server resides. Project Copy processing is most efficient when the Core Server, Scan Server, and database reside on the same machine.

Workaround: None exists.

SCA-41862: Increased time for Project Copy and other operations when Project Copy runs in parallel

If a Project Copy is triggered when any other operation—such as an import, export, scan, or report generation—is also running in your Code Insight system, the processing time for the Project Copy as well as for the other operation (especially an import, export, or scan) will be relatively greater than if these operations were run at separate times.

Workaround: In general, perform the listed operations at separate times for better performance. Ensure that Project Copy does not run in parallel with any of these operations.

SCA-41682: Project dashboard of copied project shows both Scanner and Remote Scans sections even though source project was only remotely scanned

The project dashboard of the copied project shows both Scanner and Remote Scans sections info even though the source project was scanned by a scan agent only. Only the Remote Scan section should be displayed.

Workaround: None exists.

SCA-20012: File filters in Chrome and Edge browsers not showing supported upload archive types correctly

When selecting a codebase archive to upload from File Upload dialog, the file filter on the browser you are using might list the supported archive types properly:

- On the Chrome browser, the file filter list incorrectly shows “Custom Files” instead of “Supported Files” and does not allow you to filter on the individual supported archive types.
- On the Edge browser, the file filter list shows unsupported archive types.

Project Reporting

The following are known issues with Code Insight reporting.

SCA-22054: Project Report not showing URLs for custom vulnerabilities

The Project report is not showing the NVD (National Vulnerability Database) URLs for custom vulnerabilities until they are updated to the Data Library.

Workaround: Use the Web UI to view all vulnerabilities associated with inventory.

SCA-11263: Project Report hyperlink on tasks worksheet for inventory does not work

Clicking on an inventory link in the Project Report takes the user to the login page even if user is currently logged in. This is a bug in Excel.

Workaround: Log into the application. Go back to the Excel report output and click on the hyperlink again. This is an issue only for inactive sessions.

REST APIs

The following are known issues with the Code Insight REST interface.

SCA-16508: Swagger page hangs when required API parameters are missing

Instead of producing an appropriate error message, a Swagger page can hang when you attempt to execute an API without providing required parameters.

Workaround: None exists.

SCA-7950: Page and size parameters are not working with some REST APIs

Limiting the result set returned by some REST APIs is not currently supported. Using the page and size parameters with the Component Lookup and Get Project Inventory APIs (and possibly others) returns the full result set.

Workaround: None exists.

Scan Agent Plugins

The following are known issues with Code Insight scan-agent plugins.

SCA-48543: Unable to install Jenkins scan-agent plugin on Jenkins Server

The Code Insight Jenkins scan-agent plugin requires certain Jenkins dependency plugins that Jenkins automatically installs before the scan agent is installed. Jenkins will download only those dependency plugin versions that are compatible with the baseline-support version of the Jenkins Server (currently, 2.332.1). For example, Jenkins will download the **Pipeline: Groovy** dependency version that has been updated to support Jenkins Server 2.332.1 or later.

Consequently, if you are running a pre-2.332.1 Jenkins Server, some of the downloaded dependencies might be incompatible your server version, causing the Jenkins scan-agent plugin installation to fail. In this situation, consider migrating the server to version 2.332.1 or later. If migration is not feasible, you must manually install an older version of the dependency plugins that is compatible with your server version. For the list of required dependency plugins for the Jenkins Server, refer to the [Plugins Index](#) on the Jenkins site.

SCA-46097: Docker Images name with “/” causing scan to fail

A Docker Images plugin scan on a Docker image fails if the image name contains a forward slash (/), but the command that runs the scan does not include a valid tag for the name.

Workaround: If the Docker image name contains a forward slash, be sure that the command that runs the scan includes a valid tag for the image name. The following example command illustrates the correct <name>:<tag> format required in the command:

```
./code-insight-docker-plugin.sh -image alpinelinux/darkhttpd:latest
```

In the example, **alpinelinux/darkhttpd** is the image name containing a forward slash, and **latest** is the added tag (preceded by a colon).

SCA-44239: Delta file calculation during rescan not synchronized with scan

The Docker Images plugin can sometimes acknowledge files that have not changed since the previous scan as changed in the rescan. This error can impact scan time.

Workaround: None exists.

SCA-44209: Associated files not available in Syft findings for Docker Images plugin scans on Centos

File associations are not available for inventories reported by Syft during a Docker Images plugin scan on a Centos agent machine. This issue does not occur for scans performed by the same plugin on RedHat Enterprise Linux and Ubuntu machines.

Workaround: None exists.

SCA-44073: Invalid file association for transitive dependencies generated from go.sum

During a transitive scan, inventory generated from the go.sum file can have an invalid association to go.mod.

Workaround: None exists.

SCA-43034: No valid error message for scan failure when using current plugin with older Code Insight release

A current scan-agent plugin is not compatible with an earlier Code Insight release. Therefore, any attempt to run a scan-agent plugin with a Code Insight release previous to the plugin release results in failure. However, no appropriate message for this type of failure is provided.

Workaround: None exists.

SCA-42606: Seemingly “Successful” completion of Docker plugin scan despite errors

A Docker plugin scan can fail on a codebase/artifact system containing large archive files but a small /tmp partition. However, the scan status can still show “SUCCESS” (although the agent log might record the error that caused the failure).

Workaround: None exists.

SCA-41197: SHA-1 calculated for only files scanned during agent rescans subsequent to re-enablement of SHA-1

When SHA-1 is disabled and then re-enabled, any subsequent rescan by a scan agent calculates a SHA-1 value for only those files that are scanned (that is, updated or new files). SHA-1 is not calculated for those files that are skipped by the scan because they remained unchanged since previous scan.

Workaround: None exists.

SCA-41154: No scan agent support for full rescans

Prior to Code Insight 2022 R2, scan agents plugins performed only full scans. Starting 2022 R2, scan agents now support *only* incremental rescans. After the scan agent’s initial full scan of a file system, any subsequent rescans are incremental only; no forced full rescans are supported. However, a full rescan should automatically occur whenever Automated Analysis rules change, a new Code Insight version introduces new rules or data library changes, or the scan-profile settings change. Currently, no logic exists to support such an automatic full rescan when these conditions exist.

Workaround: None exists.

SCA-40626: I/O exception during Jenkins plugin scan after deletion of “.codeinsight” folder from Jenkins agent

Users can delete the .codeinsight folder from the Jenkins agent if needed. However, once the folder is deleted, scans scheduled for the Jenkins plugin might fail with an I/O exception.

For your reference, this folder is identified as \$user_dir.codeinsight, where \$user_dir is as follows:

- /home/<user>/ on Linux
- C:/Users/<user>/ on Windows

Workaround: Restart the Jenkins server.

SCA-38346: NVD calls are not going through proxy for plugin scans

When a proxy is enabled for the generic scan-agent plugin or the Jenkins plugin, NVD calls bypass the proxy during scans.

Workaround: None exists.

SCA-33465: Scan agent inventory results impacted when CODEINSIGHT_ROOT variable set to wrong path

A scan agent can produce different inventory count results when the CODEINSIGHT_ROOT variable is set as environment variable and defined with an incorrect path compared to when the variable is set to the correct path or simply not used as an environment variable. (The scan agent does not require CODEINSIGHT_ROOT to be set as an environment variable.)

Workaround: If you are running the scan agent on the same machine as Code Insight Core Server, determine whether CODEINSIGHT_ROOT has been set as environment variable. If it has, ensure that it points to the correct path. Otherwise, do not set CODEINSIGHT_ROOT as an environment variable.

SCA-28141: Maven, Ant, and Gradle scan-agent rescans might fail in dynamic host environments

Rescans performed by Maven, Ant, and Gradle scan-agent plugins v2.0 (introduced in Code Insight 2020 R3) might fail in dynamic host environments. This is due to a v2.0 requirement that rescans use the same scan-agent alias and hostname used in the previous scan. This will be addressed in a future release.

Workaround: Use the Jenkins scan-agent or the scan-agent for another CI tool that supports the “host” property. This property enables you to provide a user-defined hostname that does not change between scans.

SCA-27678: Possible deadlocks with parallel agent scans on same project

Deadlocks might occur when at least one scan-agent scan and one or more other scans (agent or server) run simultaneously on the same project.

Workaround: Scans can be scheduled in sequence to avoid deadlock exceptions.

SCA-27431: Dependencies currently not reported for Maven and Gradle scan agents

Previous versions (1.x) of the Maven and Gradle scan-agent plugins scanned both the dependencies section *and* the project build directory of the Maven or Gradle application project. However, version 2 of the plugins, introduced in Code Insight 2020 R3, scans the project build directory, but not the dependencies section. Thus, dependencies are currently not reported for scans performed by the two plugins.

Workaround for Maven: Refer to the Maven documentation for instructions on how to include dependencies as a part of build directory. An example install command for including dependencies might be:

```
maven-dependency-plugin install copy-dependencies ${project.build.directory}/project-dependencies
```

Workaround for Gradle: Refer to the Gradle documentation for instructions on how to include dependencies as a part of build directory. An example install command for including dependencies might be:

```
task copyToLib(type: Copy) { into "$buildDir/output/lib" from configurations.runtime }
```

You would then use the following command to run the scan agent from the Gradle application project:

```
gradle build copyToLib code-insight-scan
```

SCA-3378: Jenkins scan-agent plugin – downgrade not supported

After an upgrade to a Jenkins scan-agent plugin, a downgrade button option is available in the Web UI. Clicking on the option results in a 404 error.

Workaround: None exists.

SCA-3000: Scan agent plugins might generate published inventory with no selected license

For scan agent plugins not updated from 1.x (supports only legacy inventory-only projects) to 2.x, the scan results might show published inventory items that have no associated licenses. This occurs when the scan agent finds no license evidence in the codebase files or when Code Insight is able to map to the component, but multiple licenses are associated with it. In this case, the inventory item is created using Compliance Library data. It might show one or more *possible* licenses but most likely no selected license. Since the **Analysis Workbench** is not available for the legacy “inventory only” plugins, the user cannot not resolve the license issue.

Workaround: Recall the inventory item to prevent it from showing up in the published inventory items list.

Scanning and Automated Discovery

The following are known issues with Code Insight codebase scans and the detection techniques used by scans.

SCA-50489: Generic plugin scan on Gradle codebase reporting duplicate inventory and file associations

A Gradle-codebase scan performed by the generic plugin using an **Only First Level Dependencies** or **All Transitive Dependencies** scan profile with **Report Non-Runtime Dependencies** enabled or disabled can incorrectly report the following from the `build.gradle` and `libs.versions.toml` files.

- Duplicate top-level inventory
- Duplicate dependency inventory
- Duplicate associated files

Workaround: None exists.

SCA-50459: Dependencies with case-insensitive scopes in build.gradle not reported

Gradle-codebase scans using an **Only First Level Dependencies** or **All Transitive Dependencies** scan profile with **Report Non-Runtime Dependencies** enabled or disabled are not reporting those dependencies in the build.gradle file that are defined with a scope supported by Code Insight but entered in a combination of upper and lower case (or all upper case) characters.

Workaround: Ensure that the scopes added in the build.gradle file are identical in spelling and case to the scopes listed in “Gradle Dependency Scopes Supported by Code Insight” section in the *Code Insight User Guide*.

SCA-50448: Invalid duplicate transitive dependencies reported for Gradle codebases

A Gradle-codebase scan using an **Only First Level Dependencies** or **All Transitive Dependencies** scan profile with **Report Non-Runtime Dependencies** enabled or disabled can sometimes report invalid duplicate transitive dependencies for a given dependency.

Workaround: None exists.

SCA-49499: Scan not identifying correct podspec file in inventory name when multiple podspec files exist

When multiple podsec files exist in the root directory of a cocoapod project, the scan can fail to identify the correct file for “project_name” in the “[Dependency of project_name]” suffix used in those inventory names that identify dependencies.

Scan results are random as to when the correct or incorrect file is identified.

Workaround: None exists.

SCA-49181: Migrated project scan showing incorrect detection notes for inventory though mapping to Debian forge and URL is successful

After a scan on a migrated project, an inventory item whose component is found in the Debian forge is showing incorrect detection notes even though the component is successfully mapping to the Debian forge and URL. As a result, the inventory is not getting published.

Workaround: Create a new project and do a fresh scan of the codebase.

SCA-48341: Scans on Windows Server platform hang when codebase contains linux.tar files

When a Scan Server that runs on a Windows Server platform scans a codebase containing linux.tar files, the scan can hang indefinitely unless you stop and restart Tomcat.

Workaround: Perform *one* of these options before scanning the codebase:

- Untar the linux.tar file and archive the resulting folder in a zip file. Then replace the linux.tar file with the zip file in the codebase and upload the codebase to the Scan Server.

- In the scan profile, use a pattern to exclude the impacted files, `aux.c` and `aux.h`, from the scan, as shown in this example:

```
**/i2c/aux.c  
**/i2c/aux.h
```

See the “Creating Exclusion Patterns for Scan Profiles” section in the *Code Insight Installation and Configuration Guide* for more information about setting up file exclusions.

SCA-44154: Transitive dependencies not reported for `golang.org/x/tools` module

During a transitive scan of the `tools` module `golang.org/x/tools`, the Go Analyzer reports no inventory.

Workaround: The next Electronic Update will resolve this issue.

SCA-43792: Issue with Go module inventory names when associated component URL has a version suffix

When a discovered component in a Go module has a `/v<digit>` suffix in its URL, the inventory name is displayed as simply **v<digit>** in the Code Insight UI and API responses. For example, if the URL for the `blackfriday` component is `github.com/russruss/blackfriday/v2`, its inventory name is displayed as **v2**, instead of **blackfriday**.

Workaround: None exists.

SCA-43659: Security vulnerabilities not reported for Go components

Scans on Go packages are not reporting security vulnerabilities for Go components.

Workaround: None exists.

SCA-43103: Files with path change but same MD5 still being rescanned

Files whose path has changed but whose MD5 remains the same are still being rescanned even those the project’s scan profile is configured *not* to rescan unchanged files.

Workaround: None exists.

SCA-34070: Scan status not immediately in effect after “Stop Scan” issued

Currently, when a user forces a currently running scan to stop (for example, by clicking **Stop Scan** from the project **Summary** tab or the global **Scan Queue** dialog), the stopped status for the scan might not take effect immediately, even after a screen refresh.

Workaround: None exists.

SCA-30756: Increased scan times for some codebases when NG-bridge data update facility is enabled

In cases where the instance on which the Code Insight Scan Server is running has the NG-bridge data update facility enabled, the scan is able to identify more exact-file matches. However, increased matching can also cause the scan and rescan times to increase for certain codebases. This increased time can be a problem for some sites.

Workaround: Disable the NG-bridge data update facility. (Note that this facility is initially disabled by default.)

SCA-30423: Scans with large number of source-code matches resulting in longer scan times

When project is scanned with the Comprehensive scan profile or a custom scan profile, either of which has source-code matching enabled, the scan takes longer than usual if it encounters a large number of matches.

Workaround: None exists.

Inventory automatically published during previous scan now unpublished after rescan

To address issues, Code Insight now assigns a confidence level of **Low** to those inventory items that are identified by a file-name analyzer technique (a part of automated analysis) during a scan. If your project is configured to publish inventory with **Medium** or **High** confidence, inventory detected by this technique will now have an automatic unpublished status. This change is applicable only for new scans.

Workaround: The previously published inventory items are still available. In the **Analysis Workbench**, simply filter inventory by **Not Published** to view the unpublished inventory, and then publish inventory as needed.

SCA-26486: Conda first-level dependencies with Semantic versions not resolved

Semantic versions for Conda first-level dependencies are not being resolved.

Workaround: None exists.

SCA-7820: Some NPM version patterns are not supported

When scanning an NPM project, certain versions might not be detected through automated analysis. The following are not supported: URLs as dependencies, versions containing a hyphen (for example, "crypto-js": "3.1.9-1"), and versions of the format X.X.X (for example, "through": "X.X.X").

Workaround: None exists.

Source Code Management (SCM) Support

The following are known issues with Code Insight SCM support.

SCA-48045: scmInstances GET API not returning all URL details when any Git URL has multiple delimiters

Currently, if the scope for the **scmInstance** GET API includes at least one Git SCM instance with a repository URL that is defined with more than one delimiter (branch, tag, or commit ID), an error message similar to the following is returned. The message (with the status code 400) lists those Git URLs defined with multiple delimiters in the instance.

```
{
  "errors": [
    {
      "param": "[scmType: GIT; instanceId: 0; URL: https://github.com/sbnl/
TestPublicRepo.git~master>>commit123, scmType: GIT; instanceId: 0; URL: https://
github.com/test/newtest^^tag123~branchname]",
      "message": "Branch, Tag, Commit ID are mutually exclusive."
    }
  ]
}
```

When this message is returned, details are not returned for repositories of the other URLs in the same Git SCM instance, nor are they returned for the repositories identified in the other SCM instances of any type in the project.

The missing details for the other URLs in the Git SCM instance as well as for the other SCM instances makes it difficult to obtain the information needed to update a given instance. (The request body for the PUT method requires the complete updated definition of the SCM instance.)

Workaround: When creating or updating a Git SCM instance, provide only one delimiter per URL in a Git SCM instance. Additionally, use the Code Insight UI to update SCM instance and retrieve their details.

SCA-47353: Unhelpful message when testing or synchronizing with invalid or missing credentials

When you attempt to test the connections for a specific SCM instance (or run a synchronization across all SCM instances in a project) *and* the connection credentials are invalid or missing for any instance, a message with an unhelpful error description is displayed.

Workaround: Refer to the core or catalina logs for an accurate description of the error.

SCA-46441: TFS repository failing to synchronize

The synchronization between a TFS codebase repository and a Code Insight project can fail even though the TFS instance connection is configured correctly (with a valid URL, user name, password, and other properties) in the project.

Workaround: On the **Version Control Settings** tab for the project, provide a personal access token (PAT) in the **Password** field instead of a password to enable successful synchronizations.

SCA-40067: SCM instance numbering systems used in REST API output and Web UI not in sync

The instance Ids shown in the **GET SCM Instance** API response are not in sync with SCM instance numbers generated in the Web UI.

Workaround: None exists

SCA-27751: Failure of Perforce SCM instance to synchronize Unicode files to Scan Server

Perforce SCM instances can fail to synchronize Unicode-formatted files to the Scan Server if the instance is running in Windows and configured for SSL.

Workaround: None exists.

SCA-27674: Synchronization with Team Foundation Server failing (Linux only)

Codebase synchronization with a Team Foundation Server (TFS) instance on Linux fails when character spaces or certain special characters exist in attributes used to set up the synchronization on the **Version Control Settings** tab for a project.

The following issue has been logged with TFS:

<https://github.com/microsoft/team-explorer-everywhere/issues/321>

Workaround: None exists.

Vulnerability Suppression/Unsuppression

The following are known issues with the Vulnerability Suppression/Unsuppression functionality.

SCA-37089: Unable to suppress/unsuppress a vulnerability for more than 2097 versions of a component all at once (in a SQL Server environment)

When a user attempts to suppress or unsuppress a security vulnerability for more than 2097 versions of a component all at once (using the **All Current Versions** scope or the **Specified Versions** scope with more than 2097 entries), the operation fails with an appropriate error message. This same problem occurs when running the **Suppress vulnerability** or **Unsuppress vulnerability** REST APIs.

This issue occurs only when the Code Insight database is SQL Server.

Workaround: Suppress or unsuppress the vulnerability using the **Specified Versions** scope with fewer entries. Repeat this operation until the vulnerability has been suppressed or unsuppressed for all desired versions.

SCA-36973: Open alert counts not automatically refreshed after vulnerability suppression

After a security vulnerability is suppressed for a component version with open an open alert associated with the vulnerability, the open alert count is not automatically refreshed to show the reduced count in the Code Insight Web UI.

Workaround: Manually refresh the browser screen.

SCA-36768: “Vulnerabilities” bar graph not automatically refreshed after vulnerability suppression

After a security vulnerability is suppressed for a component version, the count in the appropriate “severity” segment of the **Vulnerabilities** bar graph for the component version is not automatically reduced.



Note ▪ The issue has been fixed for the bar graph on the **Inventory** view and **Project Inventory** tab. However, the issue has not been fixed for the bar graph displayed in other locations.

Workaround: Manually refresh the browser screen.

Web UI

The following are known issues with the Code Insight Web UI.

SCA-27892: Project Dashboard showing incorrect format after report generation for agent scans

If only a scan agent has performed a remote scan for a project and a report is subsequently generated for the project, the project Dashboard is showing a split **Scan Summary** pane with scan statistics for both the scan agent and the scanner (which shows statistics of 0 since it has run no scan). The **Scan Summary** should not be split; the full pane should list statistics for the remote scan only.

SCA-20683: Project details not automatically updating after scan

Project details are not automatically updating after a scan in the Web UI.

Workaround: Refresh the screen.

Legal Information

Copyright Notice

Copyright © 2023 Flexera Software

This publication contains proprietary and confidential information and creative works owned by Flexera Software and its licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such publication in whole or in part in any form or by any means without the prior express written permission of Flexera Software is strictly prohibited. Except where expressly provided by Flexera Software in writing, possession of this publication shall not be construed to confer any license or rights under any Flexera Software intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera Software, must display this notice of copyright and ownership in full.

Code Insight incorporates software developed by others and redistributed according to license agreements. Copyright notices and licenses for these external libraries are provided in a supplementary document that accompanies this one.

Intellectual Property

For a list of trademarks and patents that are owned by Flexera Software, see <https://www.revenera.com/legal/intellectual-property.html>. All other brand and product names mentioned in Flexera Software products, product documentation, and marketing materials are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The Software is commercial computer software. If the user or licensee of the Software is an agency, department, or other entity of the United States Government, the use, duplication, reproduction, release, modification, disclosure, or transfer of the Software, or any related documentation of any kind, including technical data and manuals, is restricted by a license agreement or by the terms of this Agreement in accordance with Federal Acquisition Regulation 12.212 for civilian purposes and Defense Federal Acquisition Regulation Supplement 227.7202 for military purposes. The Software was developed fully at private expense. All other use is prohibited.