Nexus IQ Server for Developers

In this guide, we'll go over what the IQ Server is and how it helps you select better components and build better software, faster. We'll give you some great tips to get started integrating the IQ Server into your environment, helping you add component intelligence to your everyday workflow.

Build Better Applications with the IQ Server **1**

Nexus Integrations for Developers **2**

Nexus Intelligence in your IDE 2

View Evaluation Results in GitHub & GitLab 3

Automatically Create tickets with the Jira Plugin 4

Block Bad Components with Firewall 5

Evaluation Scan Results in Jenkins 6

Inspect Packages with the Chrome Extension 7

Recap **8**

Resources 9

Build Better Applications with the IQ Server

The Nexus IQ Server acts as the brain for an organization implementing component lifecycle management. In IQ, you'll find a platform that provides functionality for managing policy, reviewing component and application information, and using our integrations to evaluate applications and repositories.

The Nexus IQ provides a platform that helps you make informed decisions when selecting components for your projects. By making smart dependency choices up-front, you can focus on your own innovation and let Nexus IQ Server ensure that the elements of your software come from well maintained, appropriately licensed, and security-conscience projects.

Nexus Integrations for Developers

Integrating with the Nexus IQ Server provides an easy way to add component intelligence to your development process and build better applications. Whether it's viewing component information

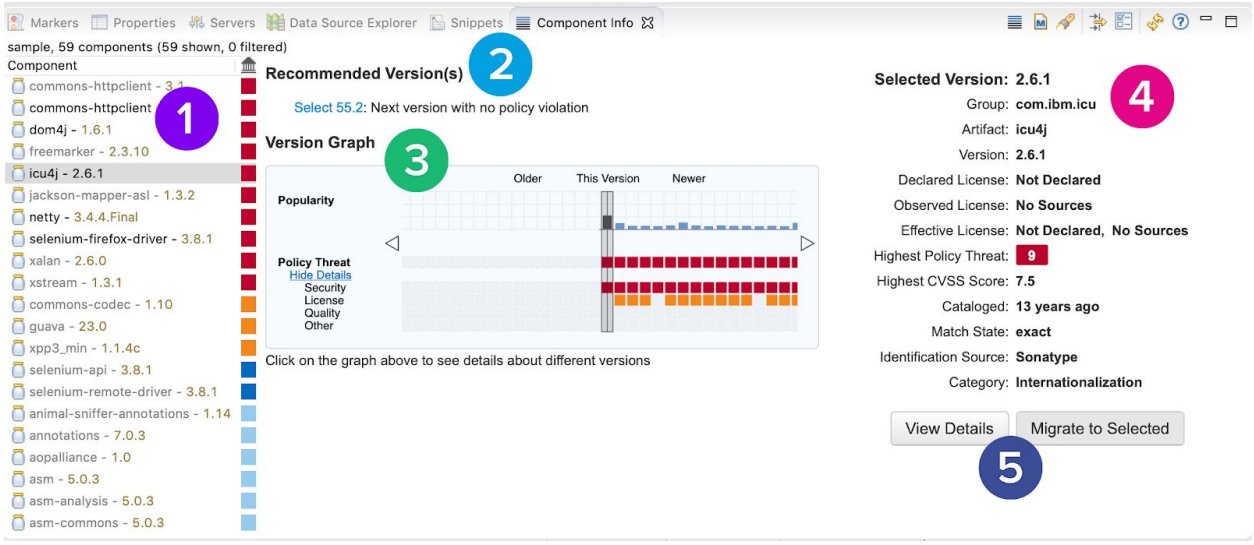
in your IDE, or adding evaluation results to your Jenkins builds, developers can use IQ Server data to be more efficient at their jobs — without sacrificing speed and reliability.

The Sonatype Nexus Integrations team works hard to make sure developers have a great experience with the IQ Server. They want to make your job easier, and they've come up with some great integrations and plugins to help you do just that.

Nexus Intelligence in your IDE

For developers, Nexus IQ Server IDE integrations are designed to work in an environment you're familiar with. Immediate feedback on component quality, including architectural, licensing, and security information, is available right in your IDE, letting you make informed decisions about component selection.

This means you can proactively make changes and choose better components before any build warnings or failures. Our IDE integrations let you quickly vet components used in an application against your organization's open source policies, greatly reducing time wasted with complicated and exhaustive research. The graphic and information below provide an example of the data you'll have access to with an IDE and IQ integration:



1	**Component List.** This is where you will see a list of components found in your project and identified by their artifact identifier and version number. The color indicator signals potential violations (red=severe, orange=medium, yellow=low, blue=none). Components with a darker font indicate that they are direct dependencies included in your application. Components brought in via a transitive dependency are displayed with a lighter font.

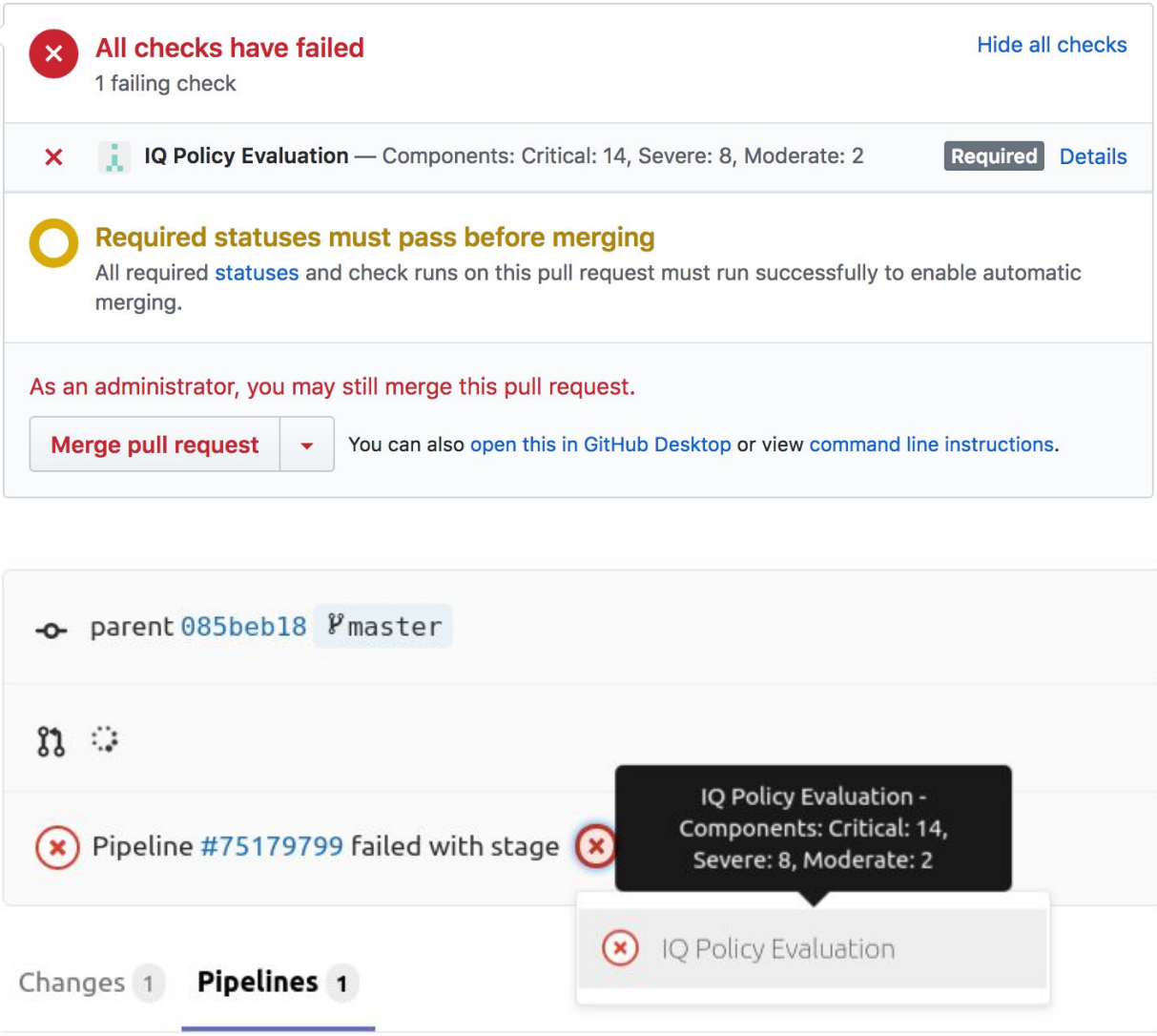
2	**Recommended Versions.** The recommended version is based on the availability of a newer version of the same component that does not violate any configured policies for the application. If such a version exists, a hyperlink is displayed with the suggested version. Clicking on the link will select the recommended version in the version graph and populate the version details with information about this version. For more information, see our help docs on IDE Recommended Versions.
3	**Version Graph**. Shows various properties for different available versions of the selected component. Older versions are displayed on the left and newer versions on the right. Arrows to the left and right of the graph let you view the full range of available versions. Click on any section in the graph, and all information for that particular version is displayed. For more information, see our help docs on the IDE Component Info View.
4	**Version Details**. Displays details of the selected component and version. Details include: component identifiers (differs depending on the language), version, overridden license, declared license, observed license, highest policy threat, highest security threat, age, identification source, and link to the project website (if available). For more information, see our help docs on the IDE Version Details.
5	**View Details and Migrate buttons**. The *View Details* button opens a dialog showing you a list of all the policies that have been violated by the component; the threat levels posed by the licenses declared for each component, as well as those that have been observed in the source code; and a list of security issues found. When you select a different, non-vulnerable version than the one currently used, the *Migrate* button becomes active. Pressing the button opens a dialog that assists you in the migration to the newer component.

Sonatype currently provides IDE integration with IntelliJ IDEA, Eclipse, and Visual Studio.

View Evaluation Results in GitHub & GitLab

Nexus IQ for GitHub and GitLab show you the information you need to begin remediating vulnerabilities in software solutions by pushing policy evaluation information into commits and pull requests. As a developer, integrating with GitHub and GitLab means you can view IQ Server evaluation results where you're working.

When you request an evaluation against a Git commit, the evaluation violation counts for components affected are summarized on the commit in GitHub or GitLab. This can be seen on pull requests or on individual commits:

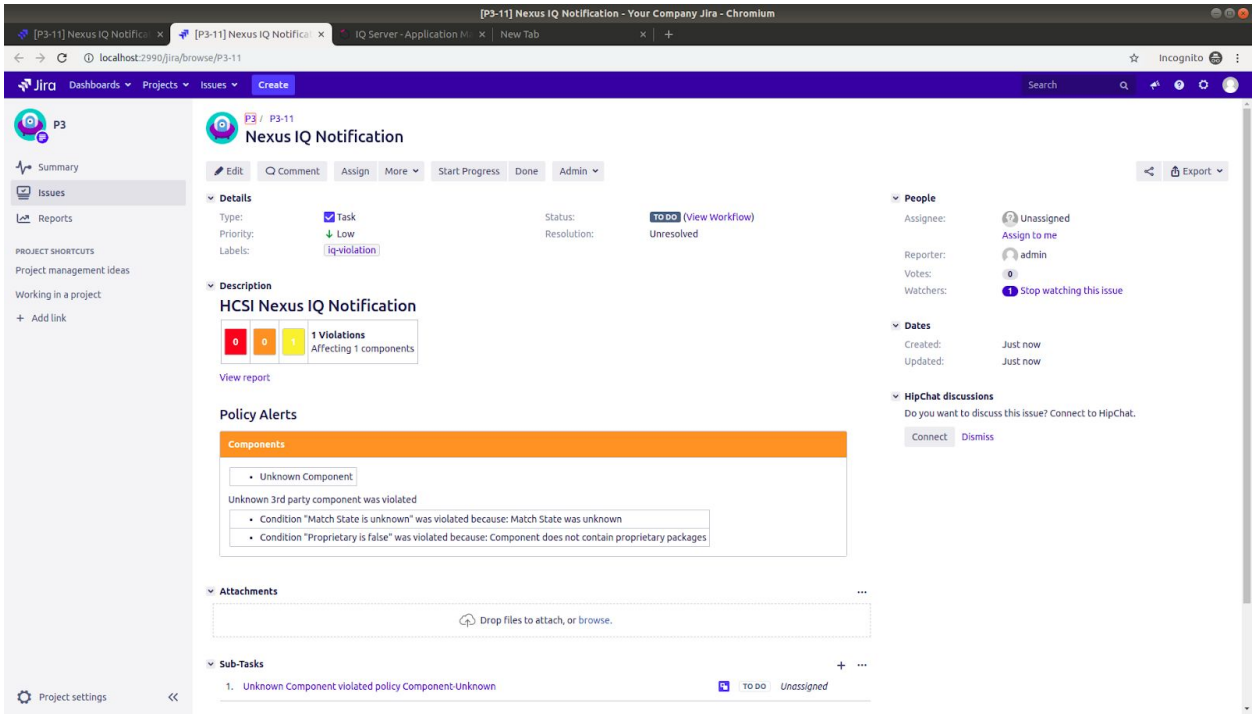


Clicking the *Details* link, or status, opens the IQ Application Evaluation report. There, you'll see the current version used, and other vulnerable and non-vulnerable versions, of that component.

Automatically Create tickets with the Jira Plugin

The Nexus IQ Jira Plugin lets you automate the creation of Jira tickets for policy violations, allowing development teams to focus on application security. The plugin uses a new IQ Server webhook violation event to trigger the creation of tickets whenever new violations occur. When an issue is found, a Jira ticket is created in the linked application, and automatically creates a ticket per component.

For programmers, this means that you can easily find and triage policy violations with a tool that you're already using for story tracking and bug fixes.



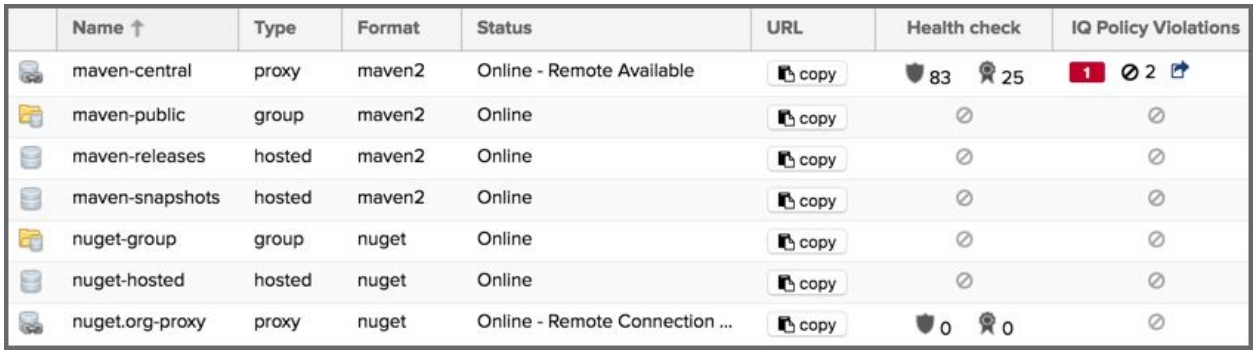
For more information, see our help docs on Nexus IQ for Jira and our guide on Nexus IQ for Jira Integration.

Block Bad Components with Firewall

Nexus Firewall automatically quarantines components that violate policy, preventing quality issues from entering the software you're developing. This process immediately reduces risk and avoids wasteful rework down the line.

Firewall works by providing Audit and Quarantine features that give you a way to protect your development environment from risky or undesirable components. When Audit is enabled, adding and deleting components to a proxy repository causes your Repository Manager to contact IQ Server and evaluate the components within the proxy repository. If violations are found, they're summarized in your Repository Manager and then detailed in IQ Server.

For example, in Nexus Repository Manager 3.x, the results of an audit are summarized in the *IQ Policy Violations* column of the *Repositories* view as shown in the image below.



	Name	Type	Format	Status	URL	Health check	IQ Policy Violations
	maven-central	proxy	maven2	Online - Remote Available	copy	83 25	1 2
	maven-public	group	maven2	Online	copy		
	maven-releases	hosted	maven2	Online	copy		
	maven-snapshots	hosted	maven2	Online	copy		
	nuget-group	group	nuget	Online	copy		
	nuget-hosted	hosted	nuget	Online	copy		
	nuget.org-proxy	proxy	nuget	Online - Remote Connection ...	copy	0 0	

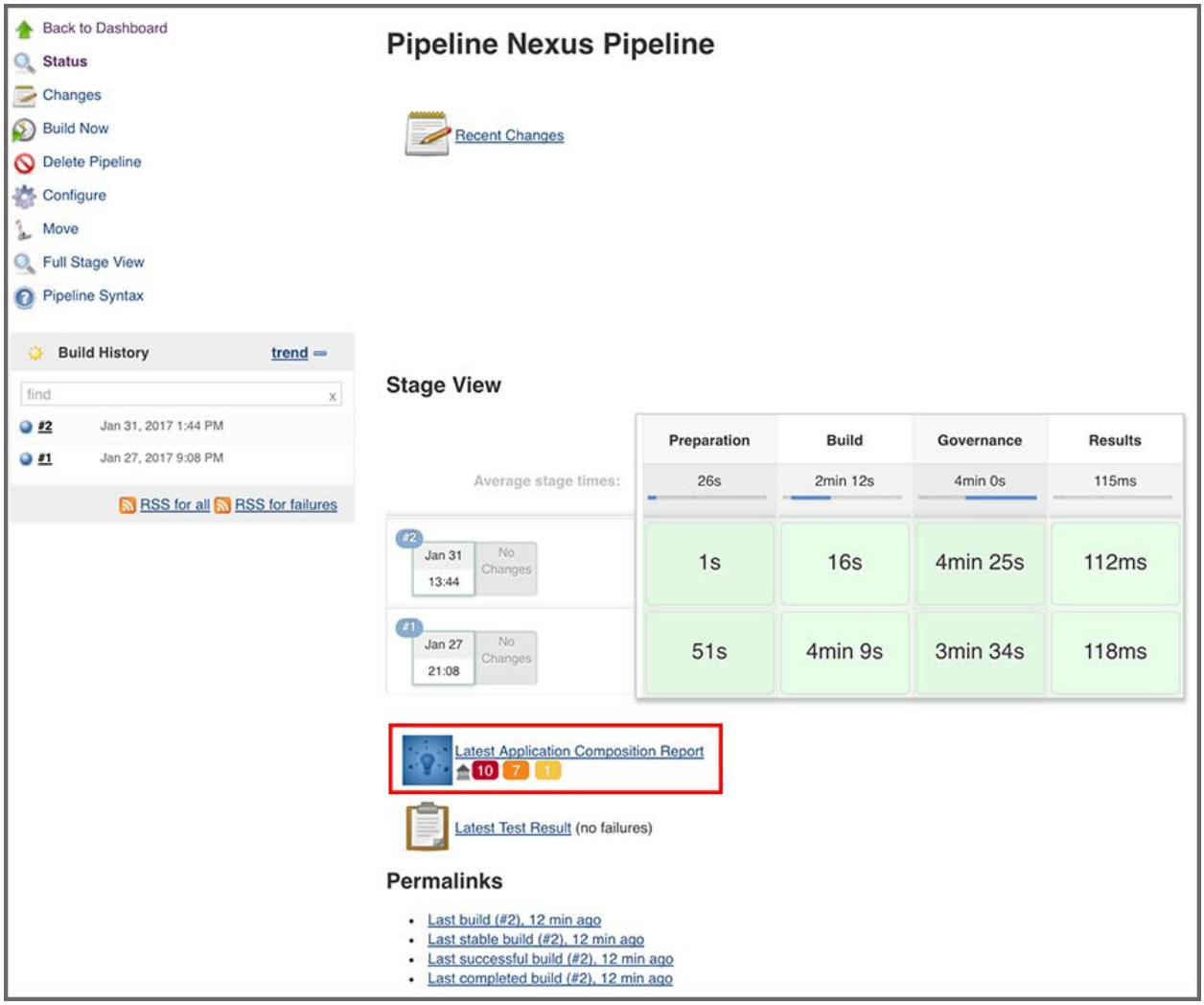
Here, you'll see (1) a count of components by their highest violation level, (2) a count of quarantined components, and (3) a link to Repository Results on IQ Server.

For more information, see our help docs on IQ Server and Repository Management.

Evaluation Scan Results in Jenkins

Nexus IQ Server can analyze the components used in your software development for security and license characteristics. When integrated with a continuous integration server, it becomes a dynamic analysis performed on a regular basis, occurring potentially with each build running on the server.

Nexus Platform Plugin for Jenkins scans a build workspace for components, creates a summary file about all the components found, and then submits that file to IQ Server for a detailed policy evaluation. A report is generated containing detailed analysis of security and license information, and a summary of that report is sent back to the Jenkins server to be included in the build results. The link to the detailed evaluation report can be followed from the Jenkins UI.



Sonatype also has integrations with other CI servers, like Bamboo and GitLab CI. All of our CI tools allow you to perform a full security and license analysis of the artifacts produced by the configured build backed by your Nexus IQ Server. It will provide you access to the analysis report.

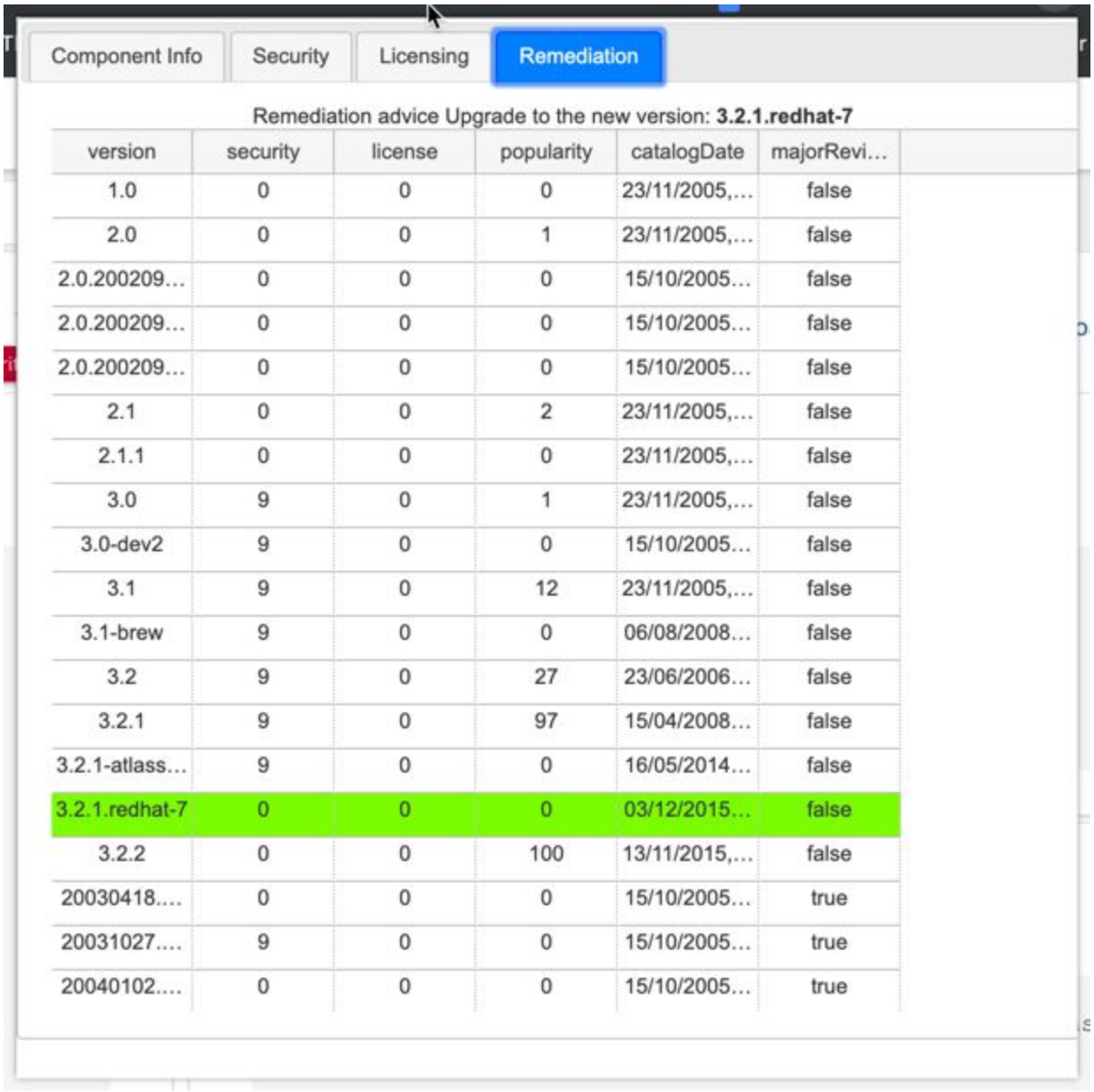
For more information, please see our help documentation on Nexus and Continuous Integration.

Inspect Packages with the Chrome Extension

NOTE: The Chrome plugin is not officially supported by Sonatype. It is a community contribution as part of the Nexus Exchange. For support, ask a question in the Sonatype Community.

The Nexus IQ Chrome Extension lets you inspect a package before you download it. The plugin requires a valid Sonatype Nexus Lifecycle license. Once the plugin is installed on your Chrome

browser, you can scan packages from several repositories like Maven, npm, Nuget, and PyPi, just to name a few.



version	security	license	popularity	catalogDate	majorRevi…
1.0	0	0	0	23/11/2005,…	false
2.0	0	0	1	23/11/2005,…	false
2.0.200209…	0	0	0	15/10/2005…	false
2.0.200209…	0	0	0	15/10/2005…	false
2.0.200209…	0	0	0	15/10/2005…	false
2.1	0	0	2	23/11/2005,…	false
2.1.1	0	0	0	23/11/2005,…	false
3.0	9	0	1	23/11/2005,…	false
3.0-dev2	9	0	0	15/10/2005…	false
3.1	9	0	12	23/11/2005,…	false
3.1-brew	9	0	0	06/08/2008…	false
3.2	9	0	27	23/06/2006…	false
3.2.1	9	0	97	15/04/2008…	false
3.2.1-atlass…	9	0	0	16/05/2014…	false
3.2.1.redhat-7	0	0	0	03/12/2015…	false
3.2.2	0	0	100	13/11/2015,…	false
20030418.…	0	0	0	15/10/2005…	true
20031027.…	9	0	0	15/10/2005…	true
20040102.…	0	0	0	15/10/2005…	true

With the Chrome Extension, you'll have access to IQ Server data like component info (format, package, version), security (severity, source, threat category, reference details), licensing (declared and observed), and most importantly, remediation (version history, recommended version).

For more information, please see the Nexus IQ Chrome Extension project on GitHub.

Recap

As you can see, Sonatype provides many ways that you can add component intelligence to your development workflow. As a first step, we recommend setting up your IDE integration. This will let you view component information, recommended versions, and even migrate and remediate fixes, all in the environment you are already using.

We have IDE integrations with IDEA, Eclipse, and Visual Studio. Please check out our IDE integration help docs to get started.

Resources

Need more help? We have you covered:

- My.sonatype.com for all things Sontaype.
- Help.sonatype.com for step-by-step instructions.
- Community.sonatype.com for asking questions and connecting with the Nexus Community.