Sonatype

Sonatype Policy Guide:

GUIDE TO UNDERSTANDING ANY POLICY VIOLATIONS IN YOUR SONATYPE OPEN SOURCE REPORT

GUIDE

INTRODUCTION

HP Fortify on Demand® and Sonatype® have come together to provide a quick and easy way to assess risk in the open source and third party components used in your software. This data for this risk assessment is powered by Sonatype's Nexus Lifecycle solution (formerly Component Lifecycle Management).

This report is a small part of the full Sonatype (Nexus Lifecycle) solution. Users of the full Nexus Lifecycle product are able to define their own policies, however, for Fortify on Demand a pre-set standard set of policies were used to trigger any potential policy violations.

The goal of this document is to help you understand the meaning of policy violations which may appear in your Fortify on Demand report by better understanding the policies that were used.

If you have any questions please contact us at www.sonatype.com/fortify/contact.

Best regards,
The Sonatype Nexus Lifecycle Team

TABLE OF CONTENTS

THE SONATYPE OPEN SOURCE REPORT

The Sonatype Open Source Report is constructed using several approaches to assessing component risk. Each approach is given its own section in the report and is identified as follows:

- Summary - A summary of what is contained in the report.
- Policy Violations - A list of components in your application, organized by their policy violations.
- Security Issues - A list of components in your application, organized by any security vulnerabilities.
- License Analysis - A list of components in your application, organized by any license issues. Implementing a project with Nexus staging
- Components - An alphabetized list of every component found in your application, as well as information combining the four previous areas.

Note: More information on how to read specific results of the report can be found at www.sonatype.com/fortify/report.

From the list above, Sonatype isolates three main ways to provide data:

- Policy Violations
- Security Vulnerabilities
- License Issues

We will discuss each of these in the following sections.

What is a policy?

Policies are simply a set of rules. The policies are organized by threat level, which can range from severe (e.g. items like high-risk security violations), to low (e.g. quality elements such as the age of an open source component).

When a policy violation occurs, it means that a component within your application has been matched to a known component in The Central Repository and it has not met the established criteria for that policy.

For example, a policy may say that a violation should occur if the application uses a component that has a GPL (Gnu Public License) license. If that component is included in your application, you will see a violation of the GPL Policy. Possible fixes to that violation could include upgrading to the latest version of the component or the removal of that component from your application.

Policies themselves are provided in four categories:

- Security Policies - rules based on specific security vulnerabilities
- License Policies - rules based on indicated license risk.
- Architecture Policies - rules based on aspects of a component that could degrade architecture quality.
- Component Policies - rules based on the identification of components in your applications to those contained in central.

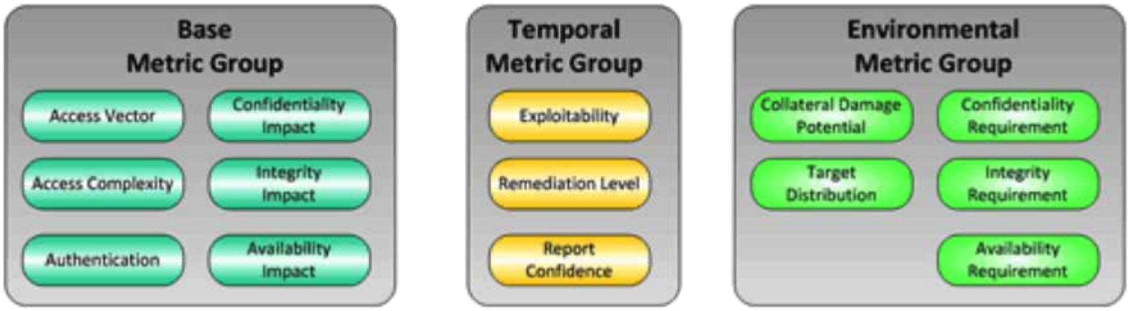
In the list above, there is some overlap with the Security Issues and License Analysis sections of the report. However, it's important to remember that these areas are separate because although they share data, they each provide different levels of information. At the end of this guide, details for all policies used in the Sonatype Open Source Report are provided.

What are security vulnerabilities?

Security vulnerabilities provide entry points for exploitation or attack. The security threat level rankings are based on the CVSS values assigned within each finding from the CVE. If you aren't familiar with terms such as CVSS or CVE, you may want to learn more here: Common Vulnerability Scoring System (CVSS) or Common Vulnerability Enumeration (CVE).

CVSS is comprised of three elements:

- Base: represents the intrinsic and fundamental characteristics of a vulnerability that are constant over time and user environments
- Temporal: represents the characteristics of a vulnerability that change over time but not among user environments
- Environmental: represents the characteristics of a vulnerability that are relevant and unique to a particular user's environment.



Reference: Information is provided by http://www.first.org/cvss/cvss-guide.html.

Given the above information, the Sonatype Open Source Report will provide security related information in two ways:

- Security-focused Policies
- Raw Security Information

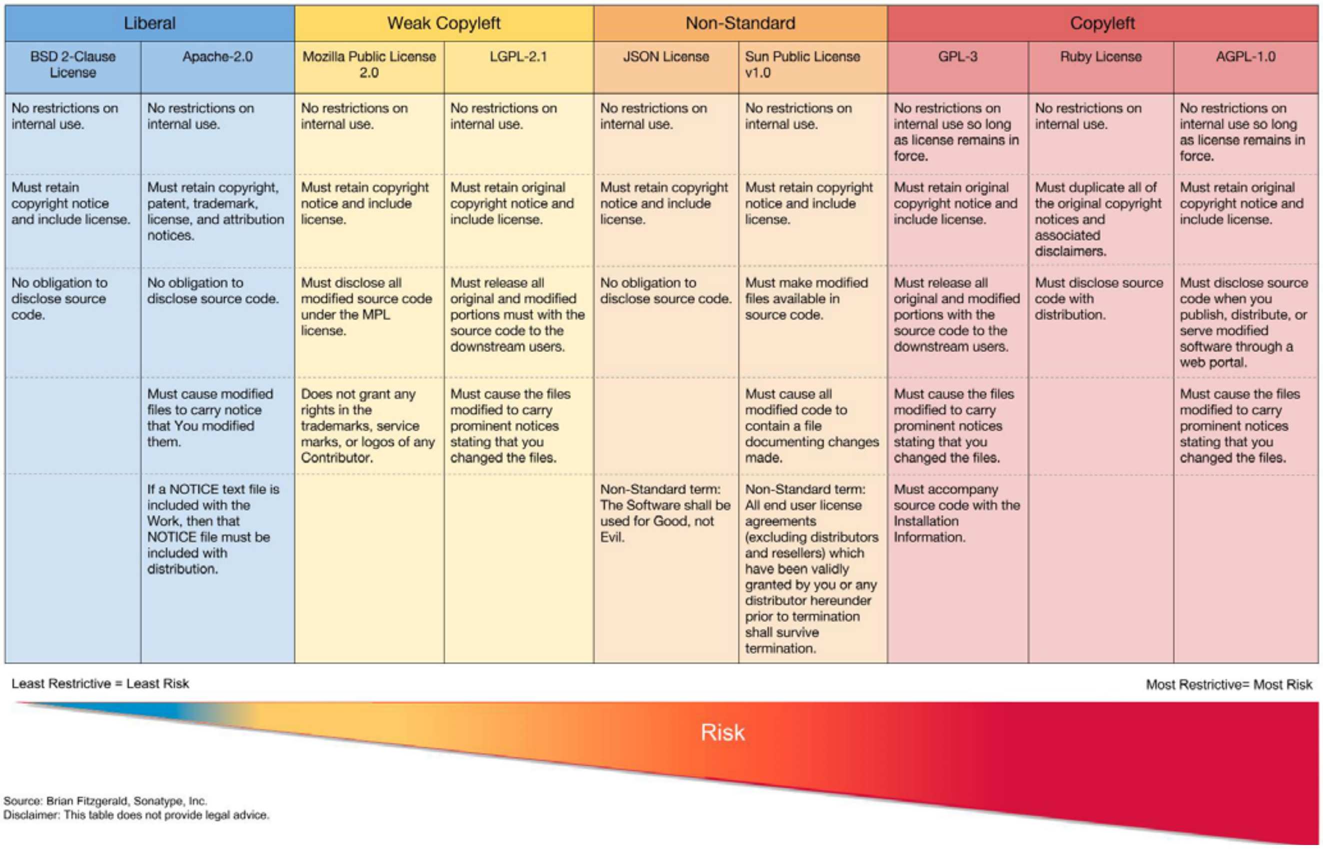
Later in our document, we'll describe all policies, including those that take into consideration the various security risk levels and CVSS scores. In contrast, the Raw Security data will provide a list of all known security vulnerabilities, and the associated score. This will all be included in the Security Issues section of the report.

What are license issues?

License related issues are often ignored because a typical approach is what we don't know won't hurt us. The opposite is true. Licenses represent real legal consequence to those that choose to ignore the often not-so-clear-legalese. We can't stress this enough, but license related issues should always be vetted against sound legal counsel. What we provide is a way to better understand the dynamics of license, but should not be substituted for true legal advice.

License threat level rankings are based on generalized information in the image below. It comes from a variety of sources, including the Sonatype categorization of license traits. Risk increases from left to right.

Free Open Source Software (FOSS) Licenses



Liberal		Weak Copyleft		Non-Standard		Copyleft		
BSD 2-Clause License	Apache-2.0	Mozilla Public License 2.0	LGPL-2.1	JSON License	Sun Public License v1.0	GPL-3	Ruby License	AGPL-1.0
No restrictions on internal use.	No restrictions on internal use.	No restrictions on internal use.	No restrictions on internal use.	No restrictions on internal use.	No restrictions on internal use.	No restrictions on internal use so long as license remains in force.	No restrictions on internal use.	No restrictions on internal use so long as license remains in force.
Must retain copyright notice and include license.	Must retain copyright, patent, trademark, license, and attribution notices.	Must retain copyright notice and include license.	Must retain original copyright notice and include license.	Must retain copyright notice and include license.	Must retain copyright notice and include license.	Must retain original copyright notice and include license.	Must duplicate all of the original copyright notices and associated disclaimers.	Must retain original copyright notice and include license.
No obligation to disclose source code.	No obligation to disclose source code.	Must disclose all modified source code under the MPL license.	Must release all original and modified portions must with the source code to the downstream users.	No obligation to disclose source code.	Must make modified files available in source code.	Must release all original and modified portions with the source code to the downstream users.	Must disclose source code with distribution.	Must disclose source code when you publish, distribute, or serve modified software through a web portal.
	Must cause modified files to carry notice that You modified them.	Does not grant any rights in the trademarks, service marks, or logos of any Contributor.	Must cause the files modified to carry prominent notices stating that you changed the files.		Must cause all modified code to contain a file documenting changes made.	Must cause the files modified to carry prominent notices stating that you changed the files.		Must cause the files modified to carry prominent notices stating that you changed the files.
	If a NOTICE text file is included with the Work, then that NOTICE file must be included with distribution.			Non-Standard term: The Software shall be used for Good, not Evil.	Non-Standard term: All end user license agreements (excluding distributors and resellers) which have been validly granted by you or any distributor hereunder prior to termination shall survive termination.	Must accompany source code with the Installation Information.		

Source: Brian Fitzgerald, Sonatype, Inc.
Disclaimer: This table does not provide legal advice.

Note: References to open source ("OSS") license agreements included in license threat groups, or Nexus Lifecycle policy configurations, do not constitute legal advice or guidance. In the end you are responsible for seeking appropriate legal advice regarding your rights and obligations set forth in any such OSS license agreement.

WHAT DO I DO NEXT?

By now, hopefully you have a basic understanding of the data provided in the Sonatype Open Source Report. The next step is to determine what to do with the results. Each organization will be different, and you will need processes to define your thresholds for which vulnerabilities you will and will not accept based on the issues raised in your report.

In some cases, there will be a zero-tolerance approach, meaning an application must have a report with no issues in order to reach production. While this is possible, it's more likely you will need to consider exceptions and various solutions based on each application's included components and exposure.

Please keep in mind that the full Sonatype Nexus Lifecycle is designed to automate the remediation of issues and keep your software safe over time.

While security and license related information will always be provided, we suggest you focus on the policy data we've provided. In the next section, we've described each policy, what the policy detects and ways to deal with possible exceptions. This is simply a guide, and over time it will be best for you to develop your own standards and best practices. To help you work through this reference section, we've also included an table that should serve a quick reference when looking at each policy.

This report is a small part of the full Sonatype (Nexus Lifecycle) solution. Users of the full Nexus Lifecycle product are able to define their own policies, however, for Fortify on Demand a pre-set standard set of policies were used to trigger any potential policy violations.

SONATYPE POLICY DEFINITIONS

Security - Critical

Policy Details:

Policy Category	Threat Level	Application Types
Security	10	All

Detection: Detects the most egregious security vulnerabilities. This policy checks for components that have a CVSS score of 10, which is the highest possible CVSS score. If using the full Nexus Lifecycle suite, the policy will also check for a status value of "Not Applicable.

Exception Handling: Review the CVE and CVE details for an explanation of this issue. Security issues in this category are almost always issues that require immediate attention and resolution. Issues in this category are generally exploitable simply by being on the classpath of the running executable. An exception could be requested if:

An external mitigating control exists and has been verified and documented by your security team. An update to the vulnerability status or a waiver can be made via the Application Composition Report.

- The exploit has been internally documented as not applicable in the context of this application. In the full Nexus Lifecycle suite, the vulnerability may also be marked as not applicable, and would then be excluded from this particular violation.
- The security policy does not apply or the business is willing to accept the risk.
- The full Nexus Lifecycle suite provides the ability to create a waiver for the policy, muting the policy violation, as well as setting the status to "Not Applicable."

Security – High

Policy Details:

Policy Category	Threat Level	Application Types
Security	9	All

Detection: Detects high-level security vulnerabilities posing a serious risk to your application. This will include any components with a CVSS score greater than or equal to 7 and less than 10. If using the full Nexus Lifecycle suite, the policy will also check for a status value of "Not Applicable.

Exception Handling: Review the CVE and CVE details for an explanation of this issue. Security issues in this category are almost always issues that require immediate attention and resolution. Issues in this category are generally exploitable simply by being on the classpath of the running executable, or through the use of a specific set of component functionality. An exception can be requested if:

- An external mitigating control exists and has been verified and documented by your security team.
- The exploit has been documented as not applicable in the context of this application.
- The security policy does not apply or the business is willing to accept the risk.
- The full Nexus Lifecycle suite provides the ability to create a waiver for the policy, muting the policy violation, as well as setting the status to "Not Applicable."

Security – Medium

Policy Details:

Policy Category	Threat Level	Application Types
Security	7	All

Detection: Detects medium-level security vulnerabilities which pose an elevated risk to your applications. This includes components with a CVSS score greater than or equal to 4, and less than 7. If using the full Nexus Lifecycle suite, the policy will also check for a status value of "Not Applicable.

Exception Handling: Review the CVE and CVE details for an explanation of this issue. Security issues in this category should be addressed after the critical and high issues. Issues in this category are generally exploitable simply by the use of a specific set of component functionality or a misconfiguration in the component usage. An exception can be requested if:

- An external mitigating control exists and has been verified and documented by your security team.
- The exploit has been documented as not applicable in the context of this application.
- The security policy does not apply or the business is willing to accept the risk.
- The full Nexus Lifecycle suite provides the ability to create a waiver for the policy, muting the policy violation, as well as setting the status to "Not Applicable."

Security – Low

Policy Details:

Policy Category	Threat Level	Application Types
Security	3	All

Detection: Detects low-level security vulnerabilities, which pose a moderate risk to your applications. This includes components with a CVSS score less than 4. If using the full Nexus Lifecycle suite, the policy will also check for a status value of "Not Applicable.

Exception Handling: Review the CVE and CVE details for an explanation of this issue. Security issues in this category should be addressed after the medium issues. Issues in this category are generally exploitable simply by the use of a specific set of component functionality or a misconfiguration in the component usage. An exception can be requested if:

- An external mitigating control exists and has been verified and documented by your security team.
- The exploit has been documented as not applicable in the context of this application.
- The security policy does not apply or the business is willing to accept the risk.
- The full Nexus Lifecycle suite provides the ability to create a waiver for the policy, muting the policy violation, as well as setting the status to "Not Applicable."

Security – Unscored

Policy Details:

Policy Category	Threat Level	Application Types
Security	5	All

Detection: Detects unscored security vulnerabilities which pose an unknown risk to your applications. Unscored vulnerabilities have not been assigned a CVSS score and the risk is unknown. If using the full Nexus Lifecycle suite, the policy will also check for a status value of "Not Applicable.

Exception Handling: Review the CVE and CVE details for an explanation of this issue. Security issues in this category should be addressed after the medium issues. Issues in this category are generally exploitable simply by the use of a specific set of component functionality or a misconfiguration in the component usage. An exception can be requested if:

- An external mitigating control exists and has been verified and documented by your security team.
- The exploit has been documented as not applicable in the context of this application.
- The security policy does not apply or the business is willing to accept the risk.
- The full Nexus Lifecycle suite provides the ability to create a waiver for the policy, muting the policy violation, as well as setting the status to "Not Applicable."

License – AGPL

Policy Details:

Policy Category	Threat Level	Application Types
License	10	Distributed

Detection: Detects components with a declared or observed license among the following licenses:

- (AGPL-3.0) GNU Affero General Public License v3, (AGPL) AGPL-Style License Not Identifiable by Sonatype
- In the full Nexus Lifecycle suite, users can change and update the licenses included in this group.

Exception Handling: Generally, a banned license is a license that has been determined by your organization as not allowed in any situation. If the component is a similar match, the license information is derived from the closest match and may not accurately reflect the actual component licensing (see Component-Similar). An exception to this policy can be applied if:

- The component is dual licensed and the chosen license is not the banned license.
- A license can be incorrect due to a commercial license or the license information is inaccurate (defect).
- The license policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status as well as an overridden license can be made via the Application Composition Report.

License – None

Policy Details:

Policy Category	Threat Level	Application Types
License	9	Distributed

Detection: Detects components that have no declared and no observed license information. Declared license information is extracted from the pom.xml file or the project website when known. Observed license information is analyzed from the source files contributed with the component. This means that the developer of the application has not provided a license, and further Sonatype research has been unable produce a license for the component. Missing observed licenses are enumerated by "No Sources" and "No Source License" which denotes the source code was not available or the source code was available but did not contain license information.

Exception Handling: Generally no license information is indicative of an older component or a very unpopular one that is not adequately maintained by the project contributors. Manual research is often required to track down any license information. If the component is a similar match, the license information is derived from the closest match and may not accurately reflect the actual component licensing (see Component-Similar). An exception to this policy can be applied if:

- Research determines the license for the component. Be careful that the license information is correct for version of the component being used.
- The license policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status as well as an overridden license can be made via the Application Composition Report.

License – Copyleft

Policy Details:

Policy Category	Threat Level	Application Types
License	8	Distributed

Detection: Detects components with a declared or observed license among the following licenses:

(GPL) *GPL-Style License Not Identifiable by Sonatype*, (GPL-1.0) *GNU General Public License v1.0 only,* (GPL-2.0) *GNU General Public License v2.0 only*, (GPL-3.0) *GNU General Public License v3.0 only, (*OSL) OSL *Style License Not Identifiable by Sonatype*, (OSL-1.0) *Open Software License 1.0*, (OSL-2.0) *Open Software License 2.0*, (OSL-2.1) *Open Software License 2.1,* (OSL-3.0) *Open Software License 3.0,* (Ruby) *Ruby License*

Copyleft licenses often have strict implications (e.g. any customized code or derivative work linked to the usage of the component must also be exposed to the community at no charge) when a component is utilized in commercial instances. In the full Nexus Lifecycle suite users can change and update the licenses included in this group.

Exception Handling: Research should be conducted to verify the specific details of the license, including but not limited to, review by a legal team. If the component is a similar match, the license information is derived from the closest match and may not accurately reflect the actual component licensing (see Component-Similar). An exception to this policy can be applied if:

- The component is dual licensed and the chosen license is not the Copyleft license. An update to the license status and selected license can be made via the Application Composition Report.
- A license can be incorrect due to a commercial license or the license information is inaccurate (defect).
- The license policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status as well as an overridden license can be made via the Application Composition Report.

License – Non Standard

Policy Details:

Policy Category	Threat Level	Application Types
License	6	Distributed

Detection: Detects components containing a license within the Non-Standard license threat group. This includes:

(Adobe), Adobe-AFM), (Adobe-EULA), (ATT), (Beerware), (Boost), (DOCBOOK), (Dyade), (HP-DEC), (IETF), (IETF-style), (ImageMagick), (InfoSeek), (IPTC), (ISO-8879), (Java-Multi-Corp), (Java-WSDL-Policy), (Java-WSDL-Schema), (JPEG), (MS-IP), (Non-Standard) *Raw License String Could Not Be Mapped to a Standardized SPDX License* (OSD), (RedHat), (RSA-Security), (Sun), (Sun-BCLA), (Sun-EULA), (Sun-IP), (Sun-Non-commercial), (Sun-Restricted), (Sun-TM), (Unicode), and (Xerox)

In this list above, a few detailed examples include the Beerware and Sun licenses. The Beerware license specifies the user is encouraged to buy the author a beer should they meet. The Sun license is a click through license for acceptance. These types of licenses may be a concern depending on your circumstances. In the full Nexus Lifecycle suite users can change and update the licenses included in this group.

Exception Handling: Generally, Non Standard license have an unspecified risk. If the component is a similar match, the license information is derived from the closest match and may not accurately reflect the actual component licensing (see Component-Similar). An exception to this policy can be applied if:

- The component license restrictions are understood and after legal review the license policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status - including an override - can be made via the Application Composition Report.

License – Modified Weak Copyleft

Policy Details:

Policy Category	Threat Level	Application Types
License	5	Distributed

Detection: Detects components containing a license among the following licenses:

(AFL) *AFL-Style License Not Identifiable by Sonatype,* (AFL-1.2) *Academic Free License v1.2,* (AFL-2.0) *Academic Free License v2.0,* (AFL-2.1) *Academic Free License v2.1,* (AFL-3.0) *Academic Free License v3.0,* (Artistic) *AGPL-Style License Not Identifiable by Sonatype,* (Artistic-1.0) *Artistic License 1.0,* (Artistic-2.0) *Artistic License 2.0,* (CC-BY) *CC-BY-Style License Not Identifiable by Sonatype,* (CC-BY-SA) *CC-BY-SA-Style License Not Identifiable by Sonatype,* (CC-BY-SA-1.0) *Creative Commons Attribution Share Alike 1.0,* (CC-BY-SA-2.0) *Creative Commons Attribution Share Alike 2.0,* (CC-BY-SA-2.5) *Creative Commons Attribution Share Alike 2.5,* (CC-BY-SA-3.0) *Creative Commons Attribution Share Alike 3.0,* (CDDL) *CDDL-Style License Not Identifiable by Sonatype,* (CDDL-1.0) *Common Development and Distribution License 1.0,* (CDDL-1.1) *Common Development and Distribution License 1.1,* (CECILL) *CeCILL Free Software Style License Not Identifiable by Sonatype,* (CECILL-1.0) *CeCILL Free Software License Agreement v1.0,* (CECILL-1.1) *CeCILL Free Software License Agreement v1.1,* (CECILL-2.0) *CeCILL Free Software License Agreement v2.0,* (CECILL-B) *CeCILL-B Free Software License Agreement,* (CECILL-C) *CeCILL-C Free Software License Agreement,* (CPAL-1.0) *Common Public Attribution License 1.0,* (CPL-1.0) *Common Public License 1.0*, (EPL) *EPL-Style License Not Identifiable by Sonatype,* (EPL-1.0) *Eclipse Public License 1.0,* (GPL-2.0-with-autoconf-exception) *GNU General Public License v2.0 w/Autoconf exception,* (GPL-2.0-with-bison-exception) *GNU General Public License v2.0 w/Bison exception,* (GPL-2.0-with-classpath-exception) *GNU General Public License v2.0 w/Classpath exception,* (GPL-2.0-with-font-exception) *GNU General Public License v2.0 w/Font exception,* (GPL-2.0-with-GCC-exception) *GNU General Public License v2.0 w/GCC Runtime Library exception,* (GPL-3.0-with-autoconf-exception) *GNU General Public License v3.0 w/Autoconf exception,* (GPL-3.0-with-classpath-exception) *GNU General Public License v3.0 w/Classpath exception,* (GPL-3.0-with-GCC-exception) *GNU General Public License v3.0 w/GCC Runtime Library exception,* (LGPL) *LGPL-Style License Not Identifiable by Sonatype,* (LGPL-2.0) *GNU Library General Public License v2* *only,* (LGPL-2.1) *GNU Lesser General Public License v2.1 only,* (LGPL-3.0) *GNU Lesser General Public License v3.0 only,* (MPL) *MPL-Style License Not Identifiable by Sonatype,* (MPL-1.0) *Mozilla Public License 1.0,* (MPL-1.1) *Mozilla Public License 1.1,* (MPL-2.0) *Mozilla Public License 2.0,* (NPOSL-3.0) *Non-Profit Open Software License 3.0*

In some instances a Weak Copyleft license may have implications if the component has been modified and is used in a commercial capacity. When using a modified weak copyleft component, the source code of the modification must be supplied in the distribution. In the full Nexus Lifecycle suite users can change and update the licenses included in this group.

Exception Handling: Generally Modified Weak Copyleft licenses are Weak Copyleft license components that are a similar match. Sometimes this can occur due to the component not being the authentic version and can be resolved by downloading the authentic version. An exception to this policy can be applied if:

- The source code of the modified component is distributed with the application.
- The license policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status, as well as an overridden license can be made via the Application Composition Report.

Architecture – None

Policy Details:

Policy Category	Threat Level	Application Types
Architecture	1	All

Detection: Detects components that have potential quality issues like being too old, too new, or unpopular. The threat level for this policy is 1 which denotes an informational issue.

This policy uses the age check to identify components older than 5 years or newer than 3 months and the relative popularity check to identify components with a 10% or less popularity score. Labels could also be used as a means to quantify other quality metrics such as rate of fix, outstanding defects, and active project forum.

Exception Handling: To determine which criteria is causing the component to fail the quality criteria, check the policy condition value that has caused the violation. In general, this component should be removed from the deployed applications or upgraded to a better version of the component if one is available. An exception can be requested if:

- The policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status - as well as an override - can be made via the Application Composition Report.

Component – Similar

Policy Details:

Policy Category	Threat Level	Application Types
Component	2	All

Detection: Detects components that are similar to a known component. The threat level of this policy is a 2 which denotes a low severity issue. Similar components are components that have been downloaded from a non official source or have been modified from the original. The exact contents of the component have been modified from the original. The recommended threat level for this policy should be a medium value. This policy checks for components that have a match state of similar and are not proprietary. In the full Nexus Lifecycle suite users can specific which component should be marked proprietary.

Exception Handling: Similar matched components are components that have been modified from the authentic version stored in the repository of record. Usually this is indicative of a component being downloaded from an unofficial repository or location and the integrity of the contents is unknown. The authentic version of this component should be used in place of this version. An exception can be requested if:

- A Modified Weak Copyleft component will be a similar match component. If the source code is supplied with the distribution and this issue is being reported due to a Weak Copyleft licensed component being modified, and exception can be considered.
- The component is included as part of another package component archive. After researching why a modified version of the component has been included, an exception can be made.
- The policy does not apply or the business is willing to accept the risk.
- In the full Nexus Lifecycle suite an update to the license status - including an override - can be made via the Application Composition Report.

Sonatype helps organizations build better software, even faster. Like a traditional supply chain, software applications are built by assembling open source and third party components streaming in from a wide variety of public and internal sources. While re-use is far faster than custom code, the flow of components into and through an organization remains complex and inefficient. Sonatype's Nexus platform applies proven supply chain principles to increase speed, efficiency and quality by optimizing the component supply chain. Sonatype has been on the forefront of creating tools to to improve developer efficiency and quality since the inception of the Central Repository and Apache Maven in 2001, and the company continues to serve as the steward of the Central Repository serving 17.2 Billion component download requests in 2014 alone. Sonatype is privately held with investments from New Enterprise Associates (NEA), Accel Partners, Bay Partners, Hummer Winblad Venture Partners and Morgenthaler Ventures. Visit: www.sonatype.com