



User Guide

2024.7.3

Copyright ©2024 by Black Duck.

All rights reserved. All use of this documentation is subject to the license agreement between Black Duck Software, Inc. and the licensee. No part of the contents of this document may be reproduced or transmitted in any form or by any means without the prior written permission of Black Duck Software, Inc.

Black Duck, Know Your Code, and the Black Duck logo are registered trademarks of Black Duck Software, Inc. in the United States and other jurisdictions. Black Duck Code Center, Black Duck Code Sight, Black Duck Hub, Black Duck Protex, and Black Duck Suite are trademarks of Black Duck Software, Inc. All other trademarks or registered trademarks are the sole property of their respective owners.

12-11-2024

Contents

Preface	10
Black Duck documentation.....	10
Customer support.....	10
Black Duck Community.....	11
Training.....	11
Black Duck Statement on Inclusivity and Diversity.....	11
Black Duck Security Commitments.....	12
1. Getting started with Black Duck	13
2. About Black Duck - Binary Analysis	17
3. Understanding Component Scanning	18
Using Black Duck Detect (Desktop).....	20
About Rapid Scanning.....	34
Using the Signature Scanner.....	38
Signature Scanner client requirements.....	38
Downloading and installing the Signature Scanner CLI.....	39
Defining your version of JRE for Signature Scanner.....	39
Running a component scan using the Signature Scanner command line.....	40
Minimum Scan Interval.....	50
Accessing the Black Duck server via a proxy.....	50
Running an offline component scan using Signature Scanner.....	52
Using certificate-based authentication with Signature Scanner.....	53
Defining the scan name.....	54
Approved signature lists.....	54
Resolving memory issues.....	55
About Reduced Persistence Signature Scanning.....	56
About Stateless Scanning.....	57
About duplicate BOM detection.....	59
About component dependency duplication.....	60
Viewing component dependency duplication.....	60
Changing how duplicate dependencies are displayed.....	60
Maximum limit for component matches.....	61
About snippet matching.....	61
Snippet extensions.....	65
Resolving proxy errors.....	70
About custom scan signatures.....	70
Hosting location for Black Duck Detect.....	73
4. Managing scans in the Black Duck UI	76
Filtering scans.....	76
Filtering by text.....	76
Filtering by Scan Status.....	76

Filtering by Created Date.....	76
Removing filters.....	77
Uploading a scan file in Black Duck.....	77
Browsing scans.....	80
Mapping a scan to a project.....	83
Removing a scan from a project.....	84
Deleting a scan.....	85
Exporting a scan file or scan results.....	86
Viewing an audit log for a BOM file.....	87
5. Understanding projects in Black Duck.....	91
Creating a project.....	92
Deleting a project.....	95
Watching projects.....	96
Viewing a list of your watched projects.....	96
Decreasing the number of watched projects.....	99
Watching projects.....	99
Cloning projects.....	100
Enabling cloning.....	101
Updating project settings.....	102
Managing project team membership.....	103
Managing tags.....	107
About project versions.....	108
Creating a new version of a project.....	109
Updating project version information.....	112
Cloning project versions.....	113
Deleting a project version.....	115
About project version phases.....	115
6. Viewing a project version's BOM.....	117
Understanding the information in a project version's BOM.....	117
Reviewing the contents of a BOM.....	118
Managing comments in a BOM.....	119
Adding a comment in the Components view.....	120
Adding a comment in the Source view.....	122
Managing files associated with BOM components.....	123
Comparing BOMs.....	127
Printing a BOM.....	130
Viewing issues in a project.....	131
Viewing component versions with encryption.....	132
About Linux distributions in Black Duck.....	133
Viewing Linux distributions in Black Duck.....	134
7. Editing a BOM.....	135
Applying edits to all versions of a project.....	135
Persistent edit examples.....	136
Enabling or disabling persistent edits for a project.....	136
Manually adding a component to a BOM.....	138
Excluding a component from a BOM.....	140
Deleting a component from a BOM.....	141

Removing components from a BOM.....	142
Ignoring a component in a BOM.....	143
Adjusting the component and/or component version in a BOM.....	145
Editing an origin or origin ID.....	146
Editing the component usage type.....	149
Modifying licenses in a BOM.....	150
Selecting the license term fulfillment status.....	151
Editing license text in the BOM.....	153
Managing subprojects.....	155
Reviewing snippet matches.....	157
8. Managing components.....	159
About custom components.....	160
Managing custom components.....	161
Managing custom component versions.....	164
About Black Duck KnowledgeBase components.....	167
Understanding the component information available from the Black Duck KB.....	167
Understanding the component version information available from the Black Duck KB.....	169
Modifying KB components.....	172
Resetting a Black Duck KB component's values.....	174
About the KnowledgeBase Feedback Service.....	175
Setting or modify a component's status.....	176
Changing the status of components and/or versions.....	176
9. Viewing risk in Black Duck.....	179
Dashboards.....	179
Project version pages.....	181
Viewing the health of your projects.....	182
Viewing your dashboards.....	188
Viewing overall risk for all projects.....	199
Understanding the types of project risk.....	200
Viewing overall risk at the project version level.....	200
Understanding the types of component risk.....	202
10. About security risk.....	204
Security risk levels.....	204
Estimated Security Risk.....	204
Suggested work flow.....	205
Defining the default security risk calculation.....	206
Viewing the security vulnerabilities of your projects, project versions, and component versions.....	207
Related vulnerabilities.....	207
Viewing project version vulnerabilities.....	207
Analyzing the impact of a vulnerability.....	211
Vulnerability impact analysis process.....	211
Viewing reachable vulnerabilities.....	212
Viewing vulnerability details.....	214
Black Duck Security Advisories.....	214
CVE record.....	222
Remediating security vulnerabilities.....	226
Remediating a vulnerability.....	227

Getting remediation guidance for components with security vulnerabilities.....	228
Managing global remediation for a vulnerability.....	231
11. Managing policies.....	234
About the policy process.....	234
Viewing policy rules.....	234
Viewing policy rule violations.....	235
Overriding violations.....	236
Removing policy overrides.....	236
Default policy rules.....	236
Creating a policy rule.....	237
Policy conditions.....	238
Creating policy rules for approved or barred items.....	246
Editing a policy rule.....	248
Copying a policy rule.....	248
Deleting a policy rule.....	249
Disabling or enabling a policy rule.....	249
Overriding policy violations.....	249
Removing policy overrides.....	250
12. Managing open source licenses.....	252
Suggested work flow.....	253
About license families.....	253
Managing license families.....	255
About custom license families.....	257
Creating custom license families.....	257
Editing custom license families.....	259
Deleting custom license families.....	260
Viewing licenses.....	261
Viewing license text.....	264
Viewing license use.....	265
Determining license risk.....	267
License risk - by usage.....	268
License risk by license family.....	273
Managing deep license data.....	276
Detecting embedded licenses.....	282
Supported file extensions/ file names.....	284
License detection process.....	285
Reviewing embedded licenses.....	285
Modifying licenses for a component.....	289
Reverting BOM-level license edits.....	292
About custom licenses.....	293
Creating custom licenses.....	294
Editing a custom license.....	295
Deleting custom licenses.....	297
About license terms.....	298
Suggested work flow.....	299
License terms process.....	299
Viewing license terms.....	301
About license term fulfillment.....	302
Creating license terms.....	305

Managing license term categories.....	309
Associating a license term to a license.....	311
Editing a custom license term.....	315
Deleting a license term.....	317
Deprecating or removing the deprecation status of a custom license term.....	317
Removing a license term.....	319
Deactivating a KnowledgeBase term.....	323
Restoring a KnowledgeBase license term.....	328
KnowledgeBase licenses.....	333
Editing a KnowledgeBase license.....	333
Restoring the original text and family of a KnowledgeBase license.....	335
Viewing detected copyright statements.....	336
Supported file extensions/ file names.....	337
Copyright detection process.....	338
Reviewing copyright data.....	338
Managing copyrights.....	342
Viewing and managing copyright statements.....	342
Creating custom copyright statements.....	344
Editing custom copyright statements.....	345
Deactivating copyright statements.....	345
Activating copyright statements.....	346
Editing KnowledgeBase copyright statements.....	346
Reverting KnowledgeBase copyright statements.....	346
Updating KnowledgeBase copyright statements.....	347
Managing attribution statements.....	347
About license conflicts.....	349
Defining conflicts for customer license terms.....	349
Enabling management of license term conflicts.....	353
Managing project license conflicts.....	356
13. Running a report.....	359
Notices File report.....	359
Excluding a component or subproject from the Notices File report.....	361
Project version reports.....	363
Vulnerability Remediation report.....	363
Vulnerability Status report.....	365
Vulnerability Update report.....	366
Deleting reports.....	369
14. Managing Black Duck user accounts.....	370
Configuring password requirements.....	370
Creating a user account.....	372
Disabling a user.....	375
Converting a user account.....	376
Viewing a user's groups.....	378
Viewing a user's projects.....	379
Changing your Black Duck password.....	380
Changing user account information.....	381
Changing a user's password.....	382
Understanding roles.....	383
Managing user roles.....	387

Viewing your roles.....	388
Black Duck user role matrix.....	389
Managing the Project Manager and Project Group Manager roles.....	396
Authenticating users with LDAP.....	397
Configuring secure LDAP.....	399
About locked out user accounts.....	401
15. Managing groups in Black Duck.....	402
Viewing your groups.....	402
Creating groups.....	403
Managing group information.....	404
Managing group projects.....	406
Managing group roles.....	408
Adding members to a project group.....	410
Adding members to a project group by managing a project group.....	411
Adding a member to a project group by managing a user.....	412
Adding a member to a project group by managing a project.....	413
Removing a member from a group.....	413
Deleting groups.....	415
16. About Project Groups.....	417
Project Group Hierarchy.....	417
Ancestor, Parent, and Child Projects and Project Groups.....	418
Members and User Groups.....	418
Creating a Project Group.....	419
Editing a Project Group.....	420
Editing the name or description.....	421
Adding project sub-groups.....	421
Removing project sub-groups.....	421
Moving a project group to a different project group.....	421
Moving another project group into the selected group.....	422
Adding a member to a project group.....	422
Removing a member from a project group.....	422
Editing a member's roles in a project group.....	422
Adding a user group to a project group.....	422
Removing a user group from a project group.....	423
Enabling or disabling SBOM report validation.....	423
18. Other administrative tasks.....	424
Creating system announcements.....	424
Markdown language.....	425
Viewing project and project version audit information.....	426
Viewing product registration information.....	428
Updating your product registration.....	431
Managing your code size limits.....	431
Managing user access tokens.....	432
Enabling license term fulfillment.....	435
Customizing the logo.....	436
Project version auto-deletion.....	436
Changing the data retention period for unmapped scans.....	439

Retain Unmatched File Data.....	440
BDSA Auto Remediation.....	442
Changing the BDSA Auto Remediation setting.....	442
Access Tokens.....	444
Manual access token deletion.....	444
Setting up the automated access token purge job.....	445
On-demand access token purge.....	445
Downloading log files and heatmap data.....	445
Viewing heatmap data.....	446
Creating a Heatmap in Excel.....	447
Creating a heatmap in Excel on Mac.....	447
Creating a heatmap in Excel on Windows.....	454
Viewing jobs.....	460
Setting up SCM providers.....	464
Managing SCM providers.....	467
19. Understanding how to search in Black Duck.....	469
Searching for projects.....	469
Searching for components.....	473
Searching for vulnerabilities.....	478
Saving and managing search results.....	481
Saving search results.....	482
Editing saved searches.....	483
Renaming saved searches.....	483
Deleting saved searches.....	483
Filtering the data shown in tables.....	483
Risk Graphs.....	484
Advanced Filters.....	484
20. Working with notifications.....	486
Viewing notifications.....	486
Viewing more information.....	486
Hiding notifications.....	487
21. About the Tools page.....	488
22. Black Duck C/CPP Tool.....	490
23. Integrating Protex with Black Duck.....	491
Understanding the Protex BOM integration process.....	492
Requirements.....	492
Downloading the Protex BOM tool.....	493
Exporting a Protex BOM.....	493
Exit Statuses.....	495
Viewing multiple versions of a Protex BOM in Black Duck.....	495
Examples.....	495
Importing the Protex BOM file.....	497
Mapping or unmapping a Protex BOM.....	497

Preface

Black Duck documentation

The documentation for Black Duck consists of online help and these documents:

Title	File	Description
Release Notes	release_notes.pdf	Contains information about the new and improved features, resolved issues, and known issues in the current and previous releases.
Installing Black Duck using Docker Swarm	install_swarm.pdf	Contains information about installing and upgrading Black Duck using Docker Swarm.
Installing Black Duck using Kubernetes	install_kubernetes.pdf	Contains information about installing and upgrading Black Duck using Kubernetes.
Installing Black Duck using OpenShift	install_openshift.pdf	Contains information about installing and upgrading Black Duck using OpenShift.
Getting Started	getting_started.pdf	Provides first-time users with information on using Black Duck.
Scanning Best Practices	scanning_best_practices.pdf	Provides best practices for scanning.
Getting Started with the SDK	getting_started_sdk.pdf	Contains overview information and a sample use case.
Report Database	report_db.pdf	Contains information on using the report database.
User Guide	user_guide.pdf	Contains information on using Black Duck's UI.

The installation methods for installing Black Duck software in a Kubernetes or OpenShift environment are Helm. Click the following links to view the documentation.

- [Helm](#) is a package manager for Kubernetes that you can use to install Black Duck. Black Duck supports Helm3 and the minimum version of Kubernetes is 1.13.

Black Duck integration documentation is available on:

- <https://sig-product-docs.blackduck.com/bundle/integrations-detect/page/integrations/integrations.html>
- https://documentation.blackduck.com/category/cicd_integrations

Customer support

If you have any problems with the software or the documentation, please contact Black Duck Customer Support:

- Online: <https://community.blackduck.com/s/contactsupport>
- To open a support case, please log in to the Black Duck Community site at <https://community.blackduck.com/s/contactsupport>.
- Another convenient resource available at all times is the [online Community portal](#).

Black Duck Community

The Black Duck Community is our primary online resource for customer support, solutions, and information. The Community allows users to quickly and easily open support cases and monitor progress, learn important product information, search a knowledgebase, and gain insights from other Black Duck customers. The many features included in the Community center around the following collaborative actions:

- **Connect** – Open support cases and monitor their progress, as well as, monitor issues that require Engineering or Product Management assistance
- **Learn** – Insights and best practices from other Black Duck product users to allow you to learn valuable lessons from a diverse group of industry leading companies. In addition, the Customer Hub puts all the latest product news and updates from Black Duck at your fingertips, helping you to better utilize our products and services to maximize the value of open source within your organization.
- **Solve** – Quickly and easily get the answers you're seeking with the access to rich content and product knowledge from Black Duck experts and our Knowledgebase.
- **Share** – Collaborate and connect with Black Duck staff and other customers to crowdsource solutions and share your thoughts on product direction.

[Access the Customer Success Community](#). If you do not have an account or have trouble accessing the system, click [here](#) to get started, or send an email to community.manager@blackduck.com.

Training


Black Duck Customer Education is a one-stop resource for all your Black Duck education needs. It provides you with 24x7 access to online training courses and how-to videos.

New videos and courses are added monthly.

In Black Duck Education, you can:

- Learn at your own pace.
- Review courses as often as you wish.
- Take assessments to test your skills.
- Print certificates of completion to showcase your accomplishments.

Learn more at <https://blackduck.skilljar.com/page/black-duck> or for help with Black Duck, select **Black Duck**

Tutorials from the Help menu () in the Black Duck UI.

Black Duck Statement on Inclusivity and Diversity

Black Duck is committed to creating an inclusive environment where every employee, customer, and partner feels welcomed. We are reviewing and removing exclusionary language from our products and supporting customer-facing collateral. Our effort also includes internal initiatives to remove biased language from our

engineering and working environment, including terms that are embedded in our software and IPs. At the same time, we are working to ensure that our web content and software applications are usable to people of varying abilities. You may still find examples of non-inclusive language in our software or documentation as our IPs implement industry-standard specifications that are currently under review to remove exclusionary language.

Black Duck Security Commitments

As an organization dedicated to protecting and securing our customers' applications, Black Duck is equally committed to our customers' data security and privacy. This statement is meant to provide Black Duck customers and prospects with the latest information about our systems, compliance certifications, processes, and other security-related activities.

This statement is available at: [Security Commitments | Black Duck](#)

1. Getting started with Black Duck

Black Duck offers a comprehensive suite of services and tools that support customers on their security journey. From customers just starting with security, to customers strengthening an established program, Black Duck has the expertise, skills, and products necessary for success.


Black Duck, a Software Composition Analysis (SCA) tool, helps with managing the supply chain of software, understanding the third-party components in use and minimizing risks from known vulnerabilities and licensing. Black Duck is a comprehensive solution for supply chain management, based primarily on source analysis.

Using Black Duck, you can:

- Scan your code and identify open source software that exists in your code base.
- View the generated Bill of Materials (BOM) for your software projects.
- View vulnerabilities that have been identified in open source components.
- Assess your security, license, and operational risk.


Protex users can use Black Duck to view and manage security vulnerabilities in their existing BOMs.

Logging in to Black Duck

 **Note:** You must have a username and password to access Black Duck. Contact your system administrator if you do not have a username. If Black Duck is configured to use LDAP, you may be able to log in to Black Duck using those credentials.

To log in to Black Duck:

1. Using a browser, navigate to the Black Duck URL supplied by your system administrator. Typically, the URL is in the format `https://<server hostname>`.
2. Enter the username and password provided by your Black Duck administrator. Your password is case sensitive.

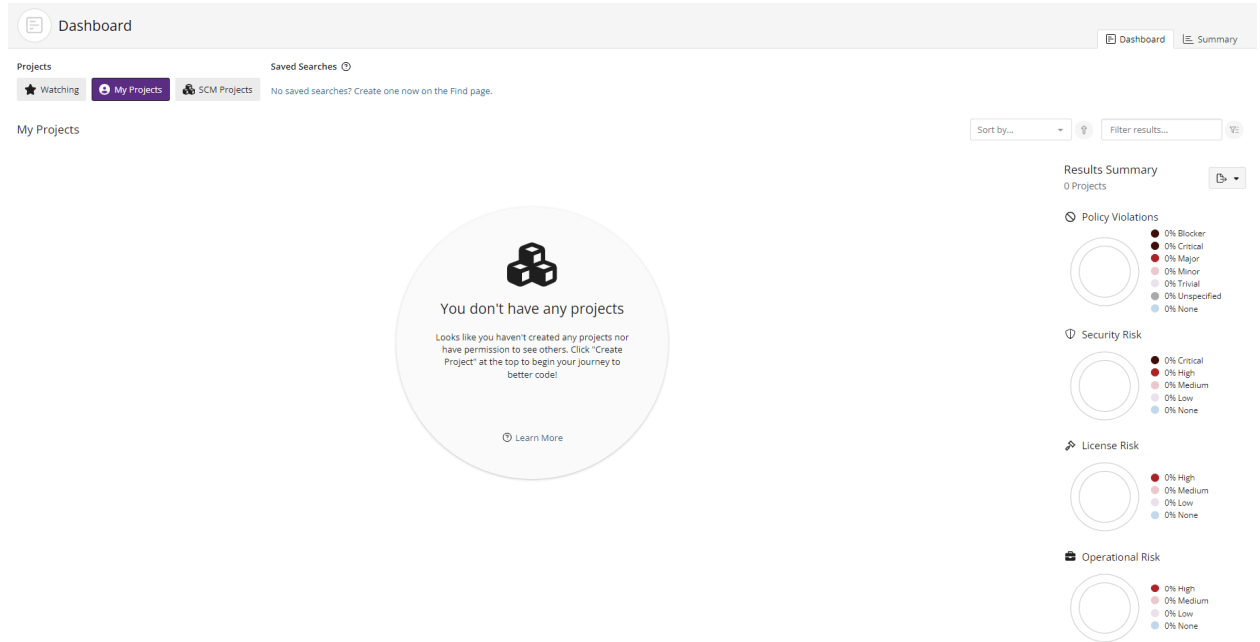
 **Note:** If your administrator has enabled [password requirements](#) and your password does not meet the requirements, a dialog box appears notifying you that you must change your password. When updating your password, make sure that it meets the requirements, as listed in the dialog box. You will not be able to log in to Black Duck unless the password meets *all* requirements.

3. Click **Login**.

When you log in, Black Duck displays your dashboard page.

- For new installations of Black Duck, when you first log in after installing Black Duck, an empty Dashboard appears.

1. Getting started with Black Duck •




For information to appear in Black Duck, you need to:

- Scan your code and map it to a project.
and/or
- Import and map a Protex BOM.

Once these tasks are complete, you can [view the discovered components in the BOM](#) and manage your security vulnerabilities.

- For existing installations of Black Duck, if this is not the first time you are logging in to Black Duck, the dashboard page that appears depends on the last main dashboard (specific [Dashboard](#) page or [Summary](#)) you viewed previously.

The Dashboard page has two default dashboards: the [Watching and My Projects](#) dashboards. You can also create custom dashboards so that you can quickly view the project versions, component versions, or security vulnerabilities that are important to you: [search for projects](#), [components](#), and/or [security vulnerabilities](#) then [save the searches](#). Your saved searches appear on the Dashboard page.

 **Note:** You will be locked out of your account for 10 minutes if you fail to enter the correct password after 10 attempts. After the 10th failed attempt, a message appears on the login page notifying you that your account is locked. Note that there is no defined time period in which the 10 failed attempts must occur – any failed attempt will be included in the count of failed password attempts. The count resets to 0 after you successfully log in to Black Duck.

The permissions assigned to your Black Duck user account by your system administrator determine which:

- navigation elements are visible to you on each page
- projects and project data you can view on each page
- actions you can perform in Black Duck

Seeing What's New in Black Duck

Discover the latest features and enhancements introduced in the current Black Duck release with the new What's New window.

The What's New window will appear automatically after login, highlighting the most impactful updates in this version. Users can choose to disable this window for future logins, but it will reappear with the next Black Duck server upgrade. Even after being dismissed, the What's New content can be accessed under the Help menu.

From the What's New window, you can also select the desired Black Duck version to see its highlights.



Black Duck Version

Scanning your code and mapping scans to projects

Use these methods to scan your code:

- [Black Duck Detect Desktop](#) which you can download from Black Duck's Tools page
- [Plugins](#).
- Black Duck Detect. Use Black Duck Detect for package management level analysis combined with signature scanning

After running a scan, [browse the available component scan results](#) in Black Duck to view the results of a component scan and the status of a scan that is in progress.

Mapping scans to projects

After scanning your code, [use Black Duck's UI to map your component scan](#) if you did not map the scan while scanning. Mapping connects your scan results to a Black Duck project.

A *project* is the base unit in Black Duck. A project can be both a stand-alone development project and part of another project. For example, Apache Tomcat is a project in its own right but it may also be part of other, larger projects. Projects can have [multiple versions](#).

Administrative tasks

Other tasks for administrators include:

- [Managing users](#). Administrators need to create and manage users in Black Duck and assign [roles](#).
- [Managing groups](#). In addition to managing role assignments and project team membership at the individual user account level, administrators can manage these for multiple Black Duck users at the same time by creating a user group.

Importing a Protex BOM

Use the Protex BOM tool to import a Protex BOM. Click [here](#) for an overview of the process and [here](#) for more information on using the Protex BOM tool.

Scanning C and C++ projects

C and C++ projects lack a standardized package manager or method for handling dependencies. As a result, generating an accurate bill of materials (or [BOM](#)) for these projects is more challenging. Use the [C/CPP Tool](#) to generate a BOM report for projects written in C/C++ by building the project, capturing the source and binary files involved, and then delivering a BDIO and signatures to Black Duck.

2. About Black Duck - Binary Analysis

Black Duck - Binary Analysis(BDBA) identifies the open source security, compliance, and quality risks in the software libraries, executables, and vendor-supplied binaries in use within your codebase. BDBA supports expanded file type support including various firmware formats, filesystems/disk images, installation formats, and various compression and archive formats. With Black Duck - Binary Analysis, you can:

- Analyze virtually any compiled software, firmware, mobile applications, or multiple installer formats, without needing to access the source.
- Identify embedded open source usage and risks within binary executables and libraries.
- Manage code decay and improve software quality within binary dependencies.
- Monitor new vulnerabilities in previously scanned binaries.

After installing Black Duck - Binary Analysis:

1. Use Black Duck Detect to scan your software or firmware.
2. View the results of your scan in a comprehensive [project version BOM](#).
For you to easily identify these files, the BOM displays the match type as Binary.
3. Use the BOM to identify known vulnerabilities and licensing obligations within software components.

Refer to the installation guides for more information on installing Black Duck with Black Duck - Binary Analysis.

3. Understanding Component Scanning

Black Duck Component Scanning is scanning functionality that provides an automated way to determine the set of open source software components that make up a software project. Component Scanning helps organizations manage their use of open source by identifying and cataloging components in order to provide additional metadata such as license, vulnerability, and project health for those components. Component Scanning lets users use the scanner to scan software artifacts on their local computers, which automatically generates a BOM that can be linked to a specific project in Black Duck.

Black Duck Component Scanning can extract the following archive types:

- AR
- ARJ
- CPIO
- DUMP
- TAR
- RPM
- ZIP
- 7z

Archives may optionally be compressed using any of the following compression algorithms:

- Bzip2
- Gzip
- Pack200
- XZ
- LZMA
- Snappy
- Z (compress)
- DEFLATE

During the component scan, Component Scanning examines similarities and differences between large clusters of files and can find:

- Exact matches to unmodified archives and directories of open source.
- Fuzzy matches to modified archives and directories of open source.

It scans an arbitrary file system directory or archive and matches to known components in the Black Duck KnowledgeBase (KB).

The core concept behind component scanning and discovery is the ability to compare the signatures of artifacts in the repository with the signatures of all OSS components in the Black Duck KB and quickly recognize a match. The recognition can be fuzzy—it does not need to be an exact match to be recognized. When there are multiple possible matches, Component Scanning determines the preferred match.

Component Scanning can discover and identify code that is:

- **Unmodified:** A collection of files that have not changed since they were released by the open source project.
- **Renamed:** A collection of files that have been renamed without other modification.
- **Compressed and/or recompiled:** Jars that have been compressed and/or recompiled after they were released by the open source project.
- **Modified or rebundled:** For example, with a jar:
 - Class files from more than one component jar combined into a single jar
 - Class files added to or deleted from a component jar
 - Nested component jars with jar files added or deleted

Component Scanning classifies each match based on how it was made:

- **Exact:** Component Scanning identified the set of files as an exact match to a component in Black Duck KB.
- **File Dependency.** Component Scanning identified a match via a file dependency.
- **Files Modified:** Component Scanning identified a fuzzy match to a component in Black Duck KB, where some of the files were modified. Sometimes this is a match to a previous or subsequent version of the component, which may have been missing from Black Duck KB at the time that the match was made.
- **Files Added/Deleted & Modified:** The component scan identified a fuzzy match to a component in Black Duck KB This can happen when:
 - An OSS component is matched, but some of the files associated with the component have been added, deleted, or modified. This can be a match to a previous or a subsequent version of the component, which may have been missing from Black Duck KB at the time of the match.
 - A component is only matched against a common directory structure (structure-only), but because a significant number of components share this structure, Black Duck KB may propose a match that has very little similarity to the scanned component.
 - A component is only matched against a common directory structure, but because proprietary or third-party code can share a common directory structure with components, Black Duck KB may propose a match that has very little similarity to the scanned code.

The Black Duck KB contains a 'blacklist' of very common, non-unique, directory tree structures. For example, many components include a directory that contains three subdirectories: 'css', 'img', and 'js'. This structure has been blacklisted, so that Black Duck KB will not propose irrelevant matches.

Supported languages

For the current list of supported languages, refer to the list of supported languages shown in the [Black Duck Detect documentation](#).

Individual file matching

Individual file matching is the identification of a component based purely upon the checksum information of a single file. Prior to Black Duck 2020.2.0, for a small set of file extensions (.js, .apklib, .bin, .dll, .exe, .o, and .so), regular signature scanning matched files to components based upon a checksum match to the one file. Unfortunately, this matching was not always accurate and produced a fair amount of false positives that required you to spend additional effort reviewing and adjusting the BOM. Therefore, individual file matching is no longer the default behavior and instead is an optional capability as of the Black Duck 2020.2.0 release.

This may cause some components to drop off your BOM, which may or may not be desired. Therefore, in the Black Duck 2020.2.0 release, Black Duck provides parameters in the scanning tools so that you can re-enable individual file matching. Refer to the command line parameters for the [Signature Scanner CLI](#) and Black Duck Detect documentation for more information.

ISO files

The Signature Scanner cannot scan an ISO file: you must first mount the file to your local file system and then scan the file system.


Supported package managers

Refer to the [Black Duck Detect documentation](#) for a list of supported package managers.

Scanning tools

Download, install, and scan using one of the following tools:

- Black Duck Detect. [Black Duck Detect](#) is the recommended scanning tool for Black Duck.
- [Black Duck Detect Desktop](#)
- [Command line](#) (CLI) version of Signature Scanner.

 **Tip:** Review the *Scanning Best Practices Guide* for information on the best practices for scanning.

Using Black Duck Detect (Desktop)


Black Duck Detect (Desktop) provides a new interface to make it easier to scan code.

With Black Duck Detect (Desktop), you can:

- [Scan](#) source directories, binaries and executables, and docker images and distributions.
- [Create a scan file](#) to be uploaded at a later time.
- [Manage scan files](#).
- [Upload scan files](#) directly to Black Duck.
- [View uploaded scans](#).

To use Black Duck Detect (Desktop):

1. Download and install Black Duck Detect (Desktop).
2. Configure Black Duck Detect (Desktop) with your Black Duck server settings and complete the installation process.
3. Use Black Duck Detect (Desktop) to scan and/or upload your files.

 **Note:** An error message appears if you exceed the scan size limit, which is 5 GB (6 GB for Black Duck - Binary Analysis). Contact Customer Support if you receive this message.


Be sure that your system meets the system requirements of Black Duck Detect.

- Click [here](#) for the system requirements for the latest version of Black Duck Detect.
- Click [here](#) for the documentation for previous versions of Black Duck Detect. Use this page to find the Black Duck Detect version and view the system requirements.

Downloading and installing Black Duck Detect (Desktop)

1. Log in to Black Duck.
2. Navigate to the drop-down menu under your username and select **Tools**.
3. Select the operating system you wish to use in the **Downloads Black Duck Detect (Desktop)** section to download the executable from Google Cloud Storage.
4. Run the executable to install Black Duck Detect (Desktop).

If you are upgrading from a previous version of Black Duck Detect (Desktop), an option appears to migrate data from the previous version.

-  **Note:** As the application installs into a directory related to its name, Black Duck Detect (Desktop) will not uninstall previous versions of Black Duck Detect Desktop. It also will not uninstall versions of Black Duck Detect (Desktop) that were installed in a non-default directory. You must manually uninstall all previous versions of Black Duck Detect Desktop, versions of Black Duck Detect (Desktop) installed in the non-default directory, and fix or delete any shortcuts.

If the Black Duck Detect (Desktop) does not open after installation and the following error message appears:

```
The SUID sandbox helper binary was found, but is not configured correctly. Rather than
run without sandboxing I'm aborting now. You need to make sure that /opt/Black Duck
Detect/chrome-sandbox is owned by root and has mode 4755.
```

your operating system does not support the Sandbox at the kernel layer. To run Black Duck Detect (Desktop) with the Sandbox disabled, enter the following at the command line:

```
blackduck-detect --no-sandbox
```

Command line options for Windows

- Unattended (silent) install for Black Duck Detect
`./blackduck-detect-latest.exe /S`
- Installing to a specific directory
`./blackduck-detect-latest.exe /D=C:\directory`

Installing the Linux version of Black Duck Detect (Desktop)

1. Download the executable from your Black Duck server, as described in the previous section.
2. Install Black Duck Detect (Desktop):

```
cd Downloads
```

To install on CentOS/RedHat:

```
sudo yum localinstall blackduck-detect-latest.rpm
```

To install on Ubuntu/Debian:

```
sudo apt install ./blackduck-detect-latest.deb
```

3. Change the permission of chrome-sandbox:

```
cd "/opt/Black Duck Detect"
sudo chmod 4755 chrome-sandbox
```

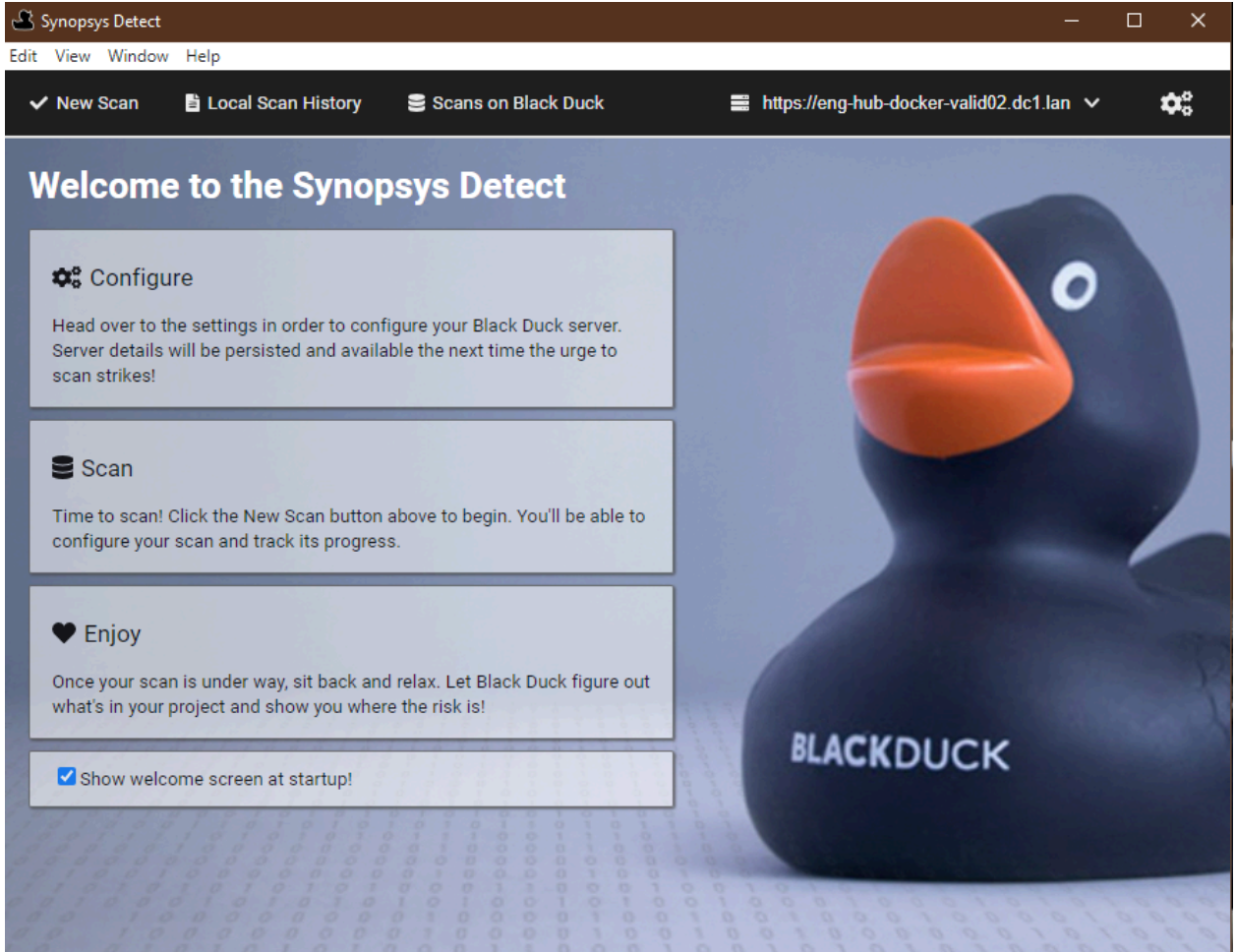
4. Run Black Duck Detect (Desktop):

```
./blackduck-detect --no-sandbox
```

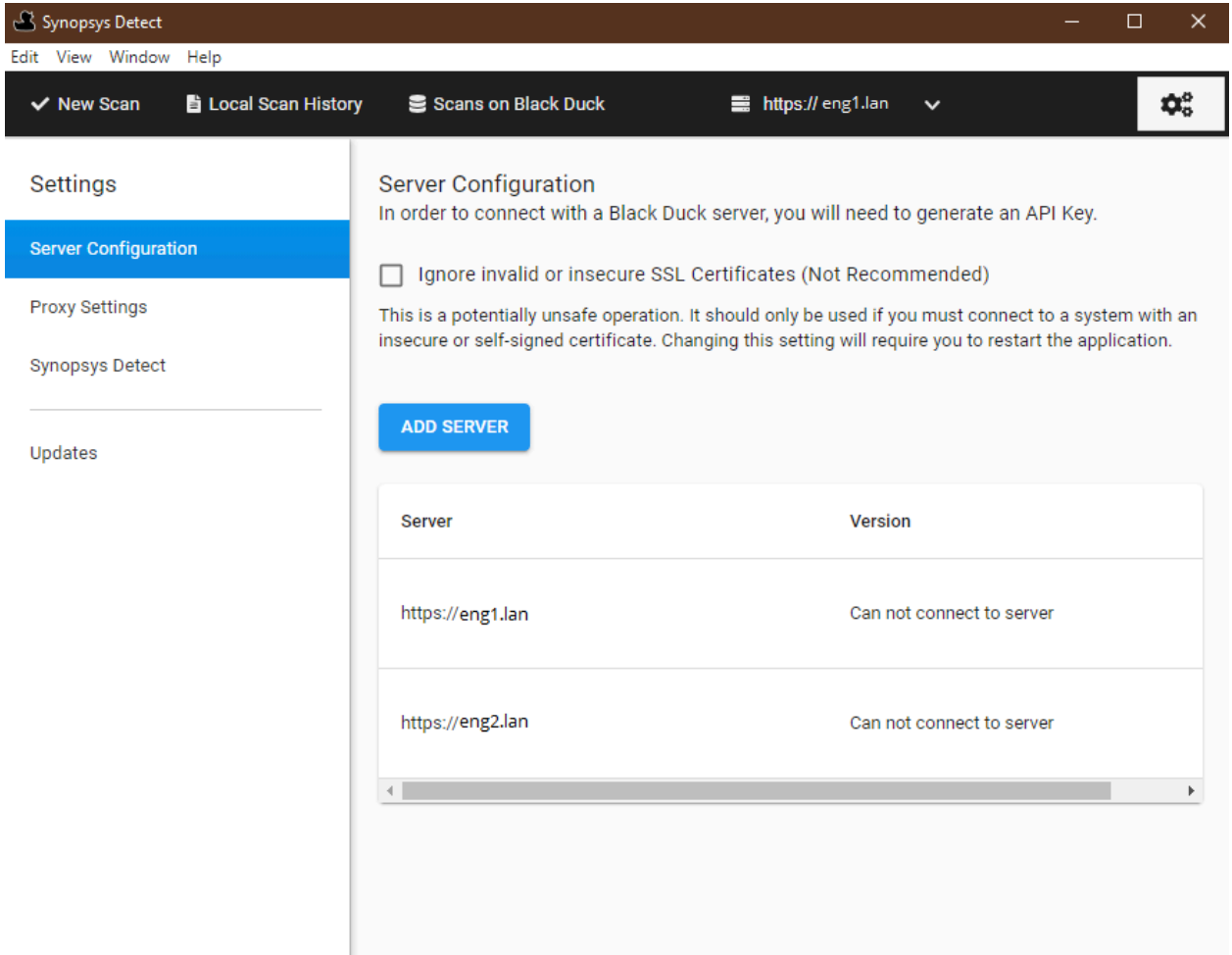
Configuring Black Duck Detect (Desktop)

After installing Black Duck Detect (Desktop), continue the installation process by configuring your Black Duck settings.

1. After installing or upgrading to Black Duck Detect (Desktop), the Welcome page appears.



2. Click , located in the upper right corner display the Settings page.



3. As described below, select one of the following tabs and complete the installation and configuration process:
- Server Configuration
 - Proxy Settings
 - Black Duck Detect
 - Updates

Black Duck server configuration

To add a server:

1. Select the **Server Configuration** tab and click **Add Server**.

The Add Server dialog box appears.

Add Server

Black Duck Server URL

Generate New API Key

[Already have a key?](#)

To generate a new API key, enter your username and password for your Black Duck server. The API key name is used to identify the key and must be unique.

API Key Name

Username *

Password *

CANCEL


CREATE

2. Specify the Black Duck Server URL. Enter the URL to the Black Duck server as you would type it in the browser, for example `https://servername:8443/`
If required, enter context information, for example, if the X-Forwarded-Prefix header is being specified in a proxy server/load balancer configuration.
3. Generate or enter an API key (user access token).
 - To generate a new API key:
 - a. Enter a key name, your username, and password.
 - b. Click **Create**.
 - To enter an API key:
 - a. Select **Already have a key?**
 - b. Enter the API key in the field.
 - c. Click **Create**.
4. Click **Save**. Black Duck Detect (Desktop) connects to the Black Duck server and displays the version of Black Duck you are connected to.

To remove an API key:

Removing the API key does not delete the key in Black Duck. It only removes it locally.


1. Select the **Server Configuration** tab.

2. Click  in the row of the server and select **Remove API Key**.

The Remove API Key dialog box appears.

3. Click **OK** to confirm.

To delete a configuration

1. Click  in the row of the server and select **Delete Configuration**.

The Delete Server Configuration dialog box appears.

2. Click **OK** to confirm.

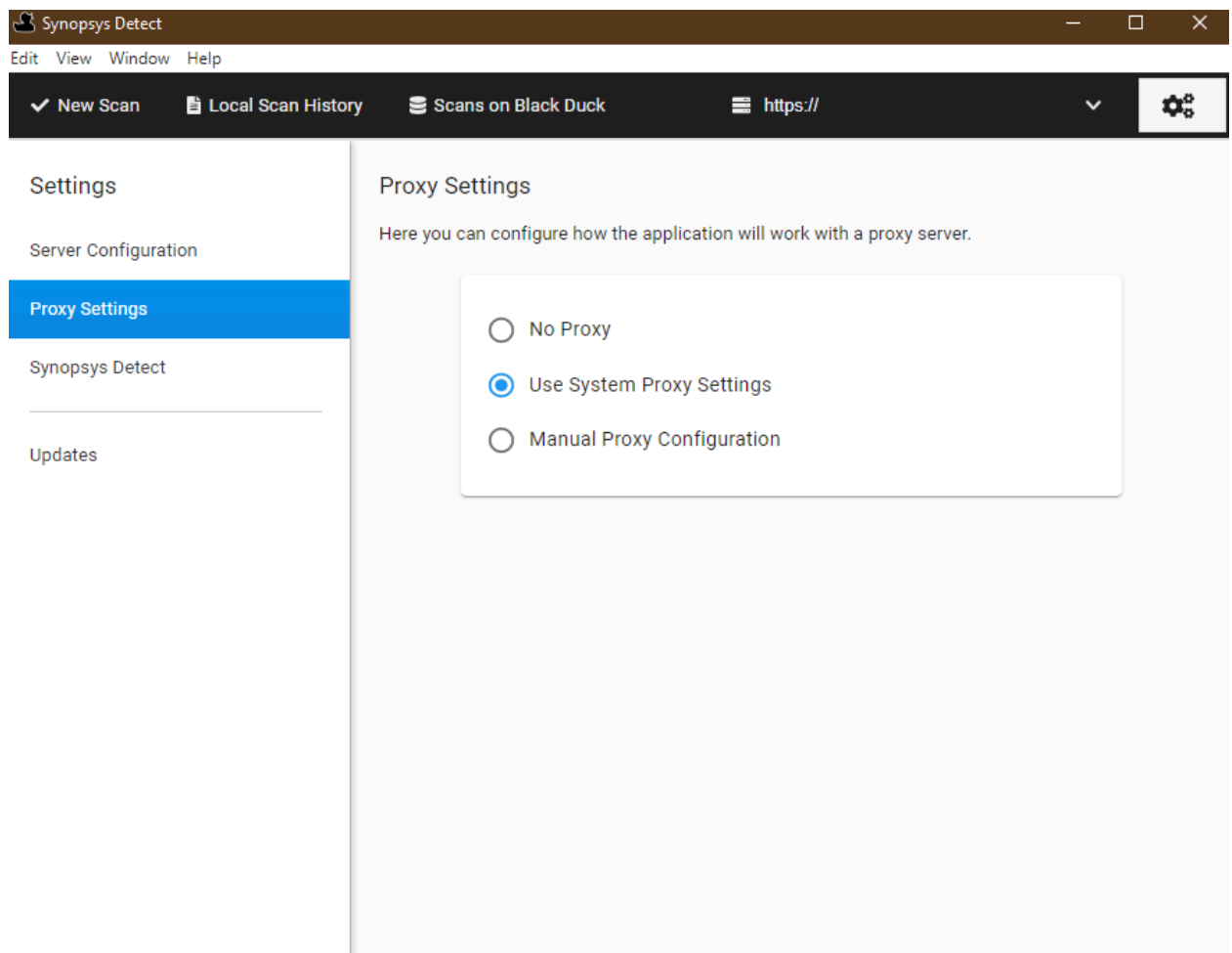
Proxy settings

Accessing Black Duck Detect (Desktop) through a proxy is supported. Black Duck Detect (Desktop) automatically uses your local system proxy setup.

If you are required to manually enter your proxy settings or you do not require a proxy, you can modify these default settings.

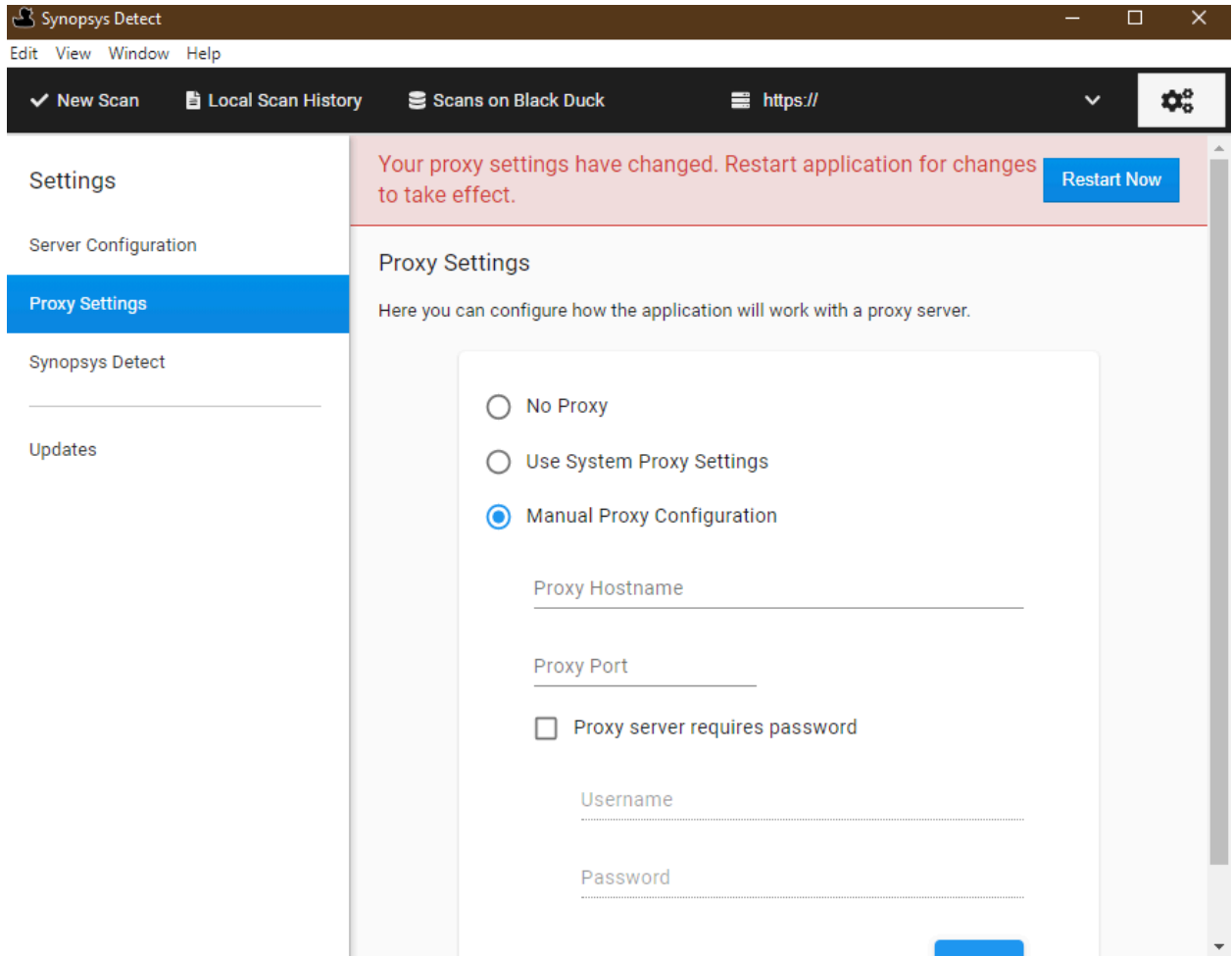
To modify the default proxy settings:

1. Select the **Proxy Settings** tab.



3. Understanding Component Scanning • Using Black Duck Detect (Desktop)

2. Select either **No Proxy** or **Manual Proxy Configuration**.
3. If you select a manual proxy configuration:



- a. Enter the following information:
 - Your proxy host name.
 - Port number.
 - Whether authentication is required.
 - Your username and password.

If a proxy is enabled and authentication is required, you may have to re-enter your username and password.

- b. Click **Save**.
4. Restart the application.

Configuring Black Duck Detect settings


Optionally, select **Synopsys Detect** and if necessary, define any Black Duck Detect settings, clear any build tools you do not want to use, or manually configure the path to the build tools.


Checking for updates

You can check to see if there are updates to the Black Duck Detect (Desktop) by selecting the **Updates** tab. The page lists the last time you checked for updates. Click **Check for updates** to view if there are newer versions available. This option is only available for Windows and MacOS systems.

Certificates


When connecting to Black Duck, you can ignore invalid or insecure SSL certificates.

1. Click , located in the upper right corner display the Settings page.
2. Select the **Server Configuration** tab.
3. Check the **Ignore invalid or insecure SSL Certificates** checkbox.
4. Restart the application.

 **CAUTION:** This is a potentially unsafe operation. It should only be used if you must connect to a system with an insecure or self-signed certificate.

Alternatively, if you want to imported a self-signed certificate, this can be done following the standard keytool import process for your JRE.

Identify the location of the JRE being used by Black Duck Detect:


1. Click , located in the upper right corner display the Settings page.
2. Select the **Black Duck Detect** tab.
3. Select **paths** from the Properties menu. Alternatively, type **paths** in the Search Properties search field to narrow the options displayed.
4. If the **Java Executable** field has no value, Black Duck Detect will use the JRE installed under \$JAVA_HOME set in your system environment variables.

Now that the location of the JRE that Black Duck Detect is using is known, the certificate should be imported to the relevant `cacerts` file (typically found in the `lib\security` folder).

1. Within a terminal session, run the following command (changing the paths to suit):

```
keytool -import -trustcacerts -keystore <path_to_keystore> -file <path_to_certificate> -alias <alias_for_cert>
```

2. You will be prompted for a password. Provide it and press enter.
3. You will be prompted whether or not to trust the certificate. Inspect the contents and accept as appropriate.

 **Note:** It may be necessary to also import any intermediary certificates associated with a chain. If you encounter any issues with the import process, please contact your IT department.

Scanning options

The Black Duck Detect (Desktop) makes it easier to scan:

- Source directories
- Binaries or executables
- Docker images or distributions

3. Understanding Component Scanning • Using Black Duck Detect (Desktop)


By default, all scans are uploaded to the Black Duck server and mapped to a project version. However, you can create a scan file as described [here](#), to output the scan to a file which you can later upload to Black Duck.

To specify project and/or version names:

1. Click **ADD** located next to **Project Settings**.
2. Select **Project Name** and/or **Version Name**. The fields appear in the UI.
3. Specify the values for the field(s).

Scanning Source Directory

To scan a source directory:

1. Click **New Scan**.
2. From the **What type of scan?** list, select **Source Directory**,
3. Click  to select the directory you would like to scan.
4. Optionally, modify or configure any project or scan settings by clicking **ADD** and selecting the setting.

If you have purchased a snippet scanning license and want to enable snippet scanning, select **Snippet Matching** from the **Scan Settings** options and enable it.


5. Click **Scan**.

The status of the scan appears along with an option to cancel the scan.

6. When the scan is complete, select the **Local Scan History** tab to view information on the completed scan. From this tab, you can [manage your scan](#). You can also view the uploaded scan using the **Scans** tab.

Scanning binary/executable

To scan a single binary or executable:

1. Click **New Scan**.
2. From the **What type of scan?** list, select **Binary/Executable**,
3. Click  to select the binary or executable you would like to scan.
4. Optionally, modify or configure any project settings by clicking **ADD** and selecting the setting.
5. Click **Scan**.


The status of the scan appears along with an option to cancel the scan.

6. When the scan is complete, select the **Local Scan History** tab to view information on the completed scan. From this tab, you can [manage your scan](#). You can also view the uploaded scan using the **Scans** tab.

Scanning a Docker image or distribution

To scan a Docker image or distribution (.tar file):


1. Click **New Scan**.
2. From the **What type of scan?** list, select **Docker**,
3. Do one of the following:

- Enter the Docker image name.
 - Select **Choose Docker archive (.tar)** and click  to select the directory you would like to scan.
4. Optionally, modify or configure any project settings by clicking **ADD** and selecting the setting.
 5. Click **Scan**.

The status of the scan appears along with an option to cancel the scan.
 6. When the scan is complete, select the **Local Scan History** tab to view information on the completed scan. From this tab, you can [manage your scan](#). You can also view the uploaded scan using the **Scans** tab.

Creating a scan file

You can use Black Duck Detect (Desktop) to output the scan to a file which you can later upload to Black Duck by using Black Duck Detect (Desktop), as described below, the [command line](#), or by [using the Black Duck UI](#).

 **Note:** Snippet scanning cannot be completed offline as it requires communication with the Black Duck server.

To create a scan file:

1. Click **New Scan**.
2. Select the type of scan (**Source Directory**, **Binary/Executable**, or **Docker**).
3. Optionally, modify or configure any project or, for source directory scanning, scan settings by clicking **ADD** and selecting the setting.
4. Select **Offline Mode**.
5. Click **Scan**.

The status of the scan appears along with an option to cancel the scan.
6. When the scan is complete, select the **Local Scan History** tab to view information on the completed scan.

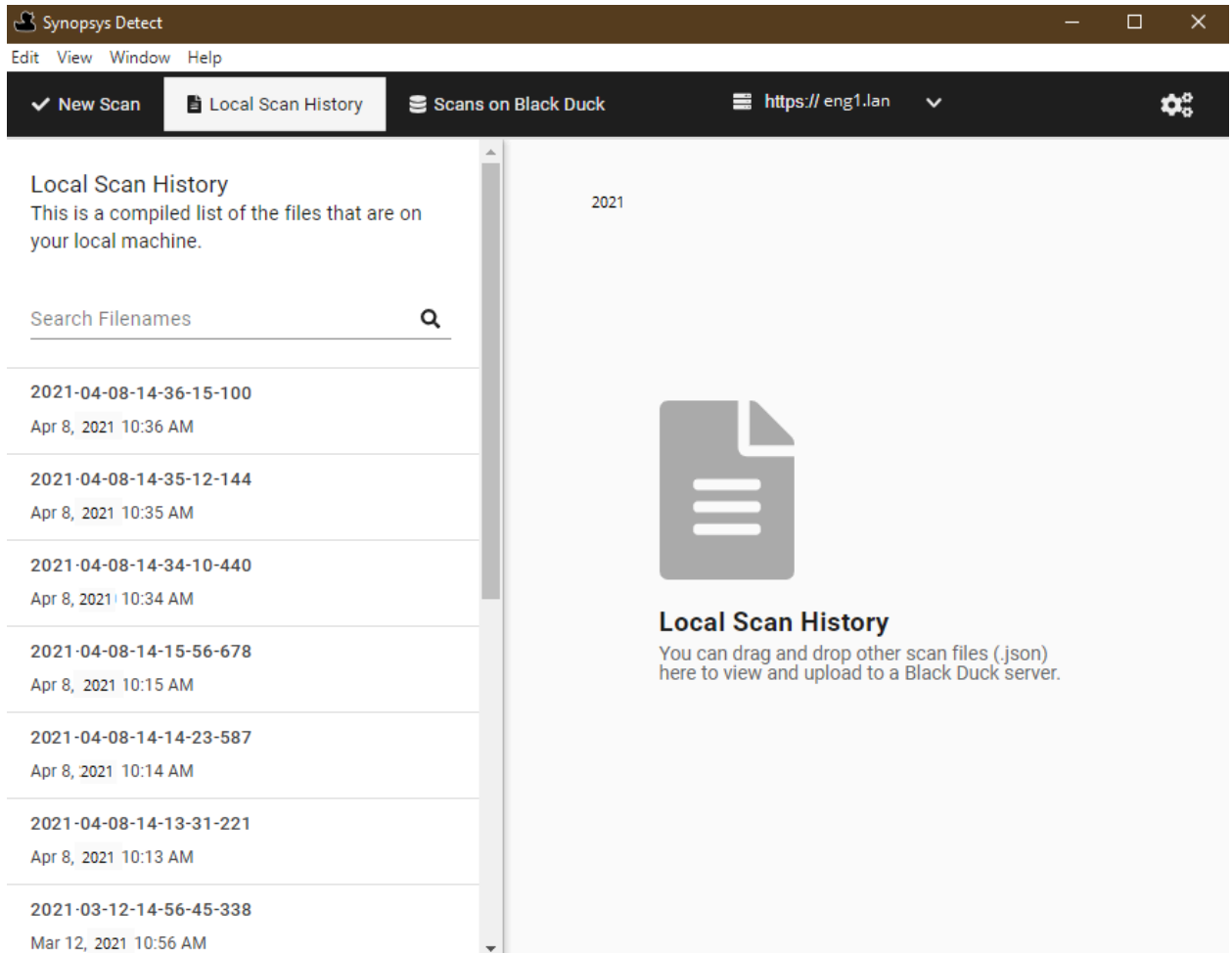
Managing scans

Use the **Local Scan History** tab to manage your scans.

1. Click **Local Scan History**.

A list of scans on your local system appears in the left column of the tab.

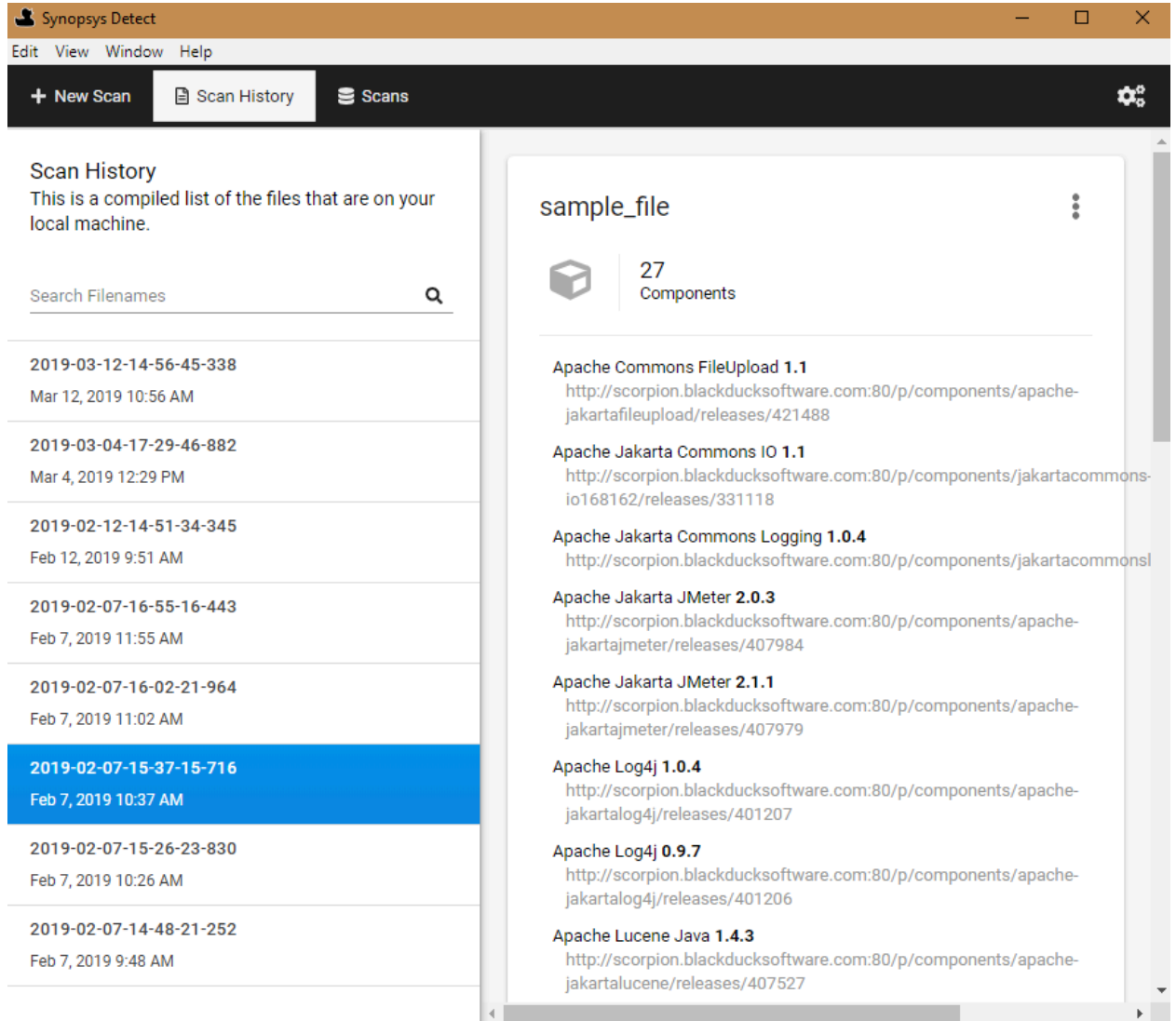
3. Understanding Component Scanning • Using Black Duck Detect (Desktop)




Drag and drop scans from your local machine to this tab to manage them.

From this tab, select a scan and:

- View information on the contents of the scan:



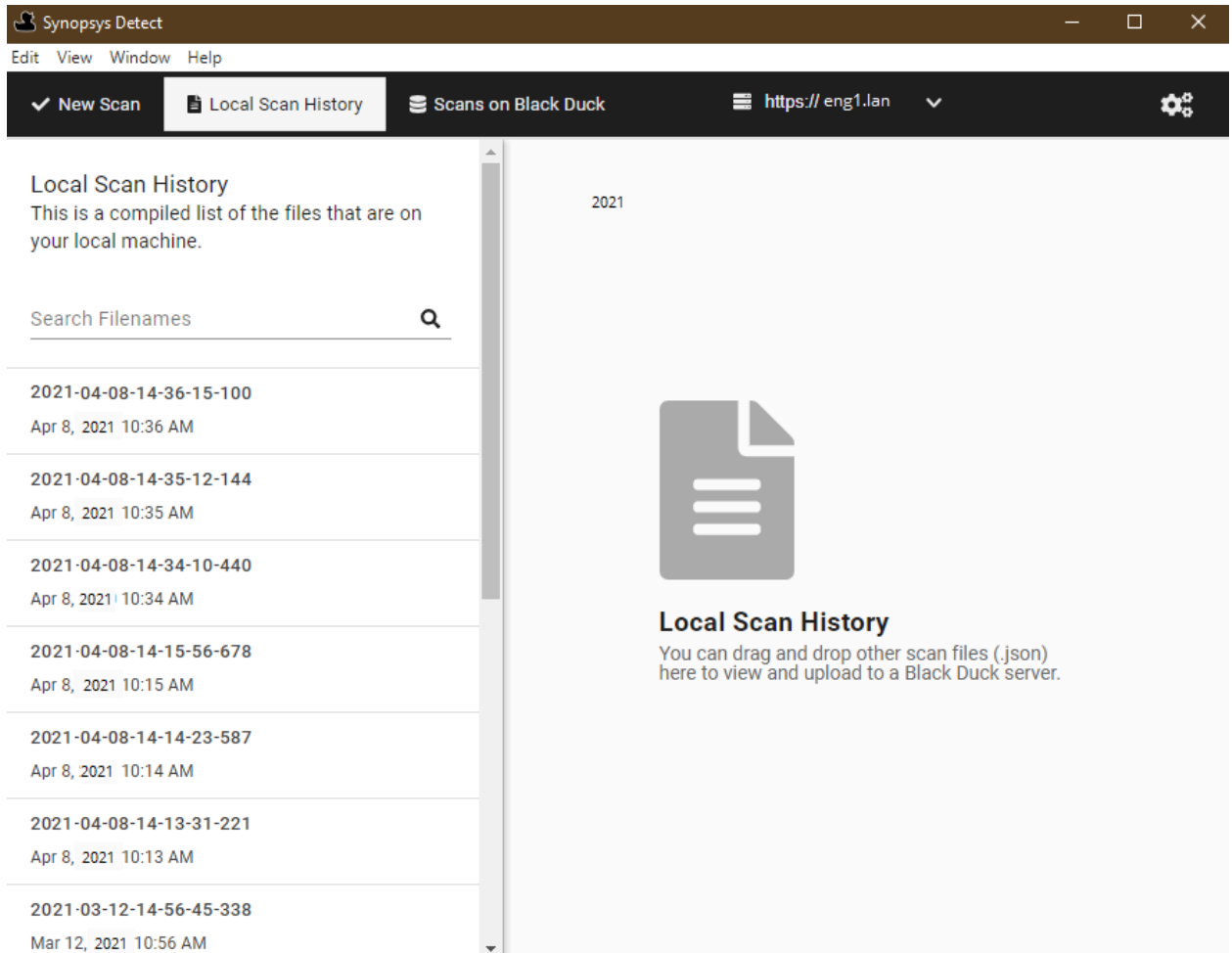
- View the location of the file on your system by clicking  and selecting **Show File**.
- Upload the file, as described in the next section.
- Delete the scan by hovering over the scan name in the left column and clicking **Delete**. Click **Yes** to confirm.


Uploading scan files to Black Duck

You can use Black Duck Detect (Desktop) to upload scan files to Black Duck.

1. Click **Local Scan History**.

3. Understanding Component Scanning • Using Black Duck Detect (Desktop)



2. If the file is on your local system, you can drag and drop the scan file from your local machine to the **Scan History** tab.
3. Select the file to upload and click  in the upper right corner to display the file options.
4. Click **Upload Scan File to Black Duck**. The Upload Progress window appears showing you the status of the upload. Close the window when the process is complete.
You can confirm that the scan has been uploaded by clicking **Scans** and viewing the uploaded file.

Viewing uploaded scans


You can view the scans that have been uploaded to Black Duck's UI by clicking **Scans on Black Duck**:

The screenshot shows the Synopsys Detect application window. The title bar reads 'Synopsys Detect'. The menu bar includes 'Edit', 'View', 'Window', and 'Help'. The main navigation bar has three tabs: '+ New Scan', 'Scan History', and 'Scans' (which is selected). On the left, under the 'Scans' tab, there is a heading 'Scans' with the text 'These are the list of scans on the Black Duck Server.' Below this is a search field labeled 'Search Scans'. Under the heading 'By Status', there is a donut chart showing a total of 27 scans, with a small red segment at the top. The main content area on the right displays a list of seven scans, each in a white card with a green border. Each card contains the scan name, a version number, and the date 'May 3, 2019'. The scan names include 'blackduck-alert/Black Ducky Alert/6.6.6 scan', 'blackduck-alert/blackduck-alert/4.0.0 npm/bom', 'blackduck-alert/alert-common/blackduck-alert/alert-common/4.2.0-SNAPSHOT gradle/bom', 'blackduck-alert/alert-database/blackduck-alert/alert-database/4.2.0-SNAPSHOT gradle/bom', 'blackduck-alert/com.blackducksoftware.integration/blackduck-alert/4.2.0-SNAPSHOT gradle/bom', 'hub-alert/blackduck-alert/4.2.0-SNAPSHOT scan', and 'hub-alert/blackduck-alert/4.0.0 npm/bom'. Each card also features a small icon of a black duck and the text 'Black Ducky Alert' followed by the version number.

This tab displays the following information:

- The left side of the tab shows uploaded scans by status (in progress, completed, or error). Use the search field to find a scan or limit the scans shown.
- The right side of the page lists the scans and shows the following information for each scan:
 - Name
 - Project and project version scan is mapped to or indicates that the scan is not mapped to a project.
 - Date the scan was uploaded to Black Duck.

Select a scan to open the *Scan Name* page in Black Duck for the selected scan.

 **Note:** The number of scanned bytes displayed in Black Duck Detect (Desktop) may differ from the number of scanned bytes shown in Black Duck. This is because of how Black Duck calculates and counts the number of bytes used. This is normal and is expected to occur in some scans.

About Rapid Scanning

Black Duck's Rapid Scanning provides a way for developers to quickly determine if the versions of open source components included in a project violate corporate policies surrounding the use of open source. Using Black Duck Detect, Rapid Scanning quickly returns results as it only employs package manager scanning and does not interact with the Black Duck server database. Use Rapid Scanning when you need quick feedback and when persisting the data in Black Duck is not necessary.

Using Rapid Scanning enables you to run thousands of scans while eliminating the need to deploy additional instances of Black Duck. It provides you with actionable results (such as failing the build) that can be used without a project version or without access to Black Duck's user interface.

Results are printed as part of Black Duck Detect's normal log-style output. If there are policy violations, the output from Rapid Scanning lists the components with at least one policy violation, providing information such as the component name, version, component identifier, and policy name so that you can easily determine the component that is violating a policy.

Note that you can use a Black Duck Detect property to save the results as a `.json` file, as described below.

When a policy violation is found, the developer can:

- replace the triggering component with a component that does not violate policies.
- obtain an exception for the violation with the policy being modified to exclude the violating component.

For Rapid Scanning:

- Ensure that the user running Rapid Scanning has the ability to invoke Black Duck Detect.
- Ensure that your Black Duck system meets the system requirements needed to run Rapid Scanning.
- Create policy rules that will be triggered when a scan violates the rule.

Policy rules can apply to all projects or a subset of projects.

To use Rapid Scan to fetch all vulnerabilities regardless of policies, simply create a single policy, setting the condition severity ≥ 0 .

Supported policy conditions are:

Component properties:

- Component (all versions or a specific version)
- Component Approval Status
- Component Modified
- Component Modification
- Component Purpose
- Component Usage
- Component Version Approval Status
- Match Type
- Newer Version Count
- Review Status
- Unknown Component Version

Licenses:

- Licenses (Declared)
- License Expiration Date (Declared)
- License Family (Declared)
- License Risk
- License Status (Declared)
- Unfulfilled License Terms

Operational:

- Component Release Date
- Commits in the past year
- Contributors in the past year

Vulnerabilities

- Critical Severity Vulnerability Count
- High Severity Vulnerability Count
- Medium Severity Vulnerability Count
- Low Severity Vulnerability Count
- Highest Vulnerability Score
- Published Date
- Vendor Fix Date
- Workaround
- Solution
- CWE IDs

Custom Fields:

- Component custom fields
- Component version custom fields
- Project custom fields
- Project version custom fields

Supported vulnerability conditions are:

- Overall Score
- CWE IDs
- Solution Available
- Workaround Available
- Exploit Available
- Reachable from Source
- Remediation Status
- RCE (Remote Code Execution)

All other policy conditions are silently ignored.

[Upgrade guidance information](#) is not used as policy conditions, however it is returned as part of Rapid Scan results separately to the policy violations output.

Note that the severity of the policy rule determines how Black Duck Detect reacts to violations. Black Duck Detect treats violations of Blocker and Critical policies as errors; if any such violations are found, Black Duck Detect prints error messages and terminates with a non-zero exit code (so that builds can be failed if desired). For all other policy severities, Black Duck Detect treats violations as warnings; Black Duck Detect will print warning messages, but they alone will not cause Black Duck Detect to terminate with a non-zero exit code.

- Use Black Duck Detect 8.0 and later.
- Enable Rapid Scanning in Black Duck Detect by including these properties:
 - **--detect.blackduck.scan.mode="RAPID"**

Refer to the [Black Duck Detect](#) documentation for more information about these properties.

- If **--detect.cleanup=false** is set, the raw JSON response from the Black Duck server will be saved as `<DETECT_OUPUT_PATH>/runs/<timestamp>/scan/<project_name>_<project_version>_BlackDuck_DeveloperMode_Result.json`.
- Any other Black Duck Detect properties that affect package manager scans are also applicable to Rapid Scanning.
- If a project and/or version is specified, but does not exist, policies will still be evaluated.
- Rapid Scanning will not create projects if they do not already exist, which is the opposite of what occurs when running other types of scans.
- For projects with complex builds, and especially when such projects use Gradle, the runtime of Rapid Scanning is frequently dominated by the runtime of the package manager when it is invoked to report the project's dependencies.
- A new job, `CollectScanStatsJob`, collects scan statistics which are shown on the **usage: rapid scan completion** section on the System Information page.
- Caching is used extensively. If any of the following values are changed, it may be up to 15 minutes for those changes to be reflected in Rapid Scanning results:
 - Component Approval Status
 - Component Custom Fields
 - Component Modified
 - Component Modification
 - Component Purpose
 - Component Usage
 - Component Version Approval Status
 - Component Version Custom Fields
 - Match Type
 - Project Custom Fields
 - Project Name
 - Project Version Name
 - Project Tags

- Project Version Custom Fields
- Review Status
- Unfulfilled License Terms

You can override the Rapid Scanning caching interval by setting the value in the `blackduck.rapidscan.policy.cache.interval.mins` environment variable in the `blackduck-config.env` file.

- If a code location was previously scanned with a traditional scan where Black Duck Detect created a project for it, a subsequent rapid scan of the same code location will be associated with that project even if no project was explicitly provided.
- The following default values are used in evaluations if the project version is not known or the component is not found in the BOM.
 - Dynamically Linked for Component Usage
 - Not Reviewed for Review Status
 - File Dependency for Match Type
 - False for Component Purpose, Component Modified, Component Modification, and Component Terms Unfulfilled

Scan Custom Config File

The policy overrides config will be ingested by Black Duck through a custom config YAML file that a developer could check-in to SCM alongside other build config files. Detect or CodeSight would then archive that file and scan header file and upload through the "Initialize Scan With Custom Config" endpoint when creating a scan.

```
version: 1.0
policy:
  overrides:
    - policyName: policyA
      components:
        - name: component1
          version: version1
        - name: component2
    - policyName: policyB
      components:
        - name: component3
          version: version3
```

3. Understanding Component Scanning • Using the Signature Scanner

Each policy override must apply to a list of specific components, on a specific version (e.g. component1 + version1) or on all versions (e.g. component2).

The endpoint to initialize a rapid scan with a custom config is:

```
POST /api/developer-scans
```

```
Consumes: application/vnd.blackducksoftware.developer-scan-1-ld-2-yaml-1+zip
```

This endpoint's request body is a zipped archive of the custom config YAML file and the bdiio header file.

BOM Comparison Modes

By using the `X-BD-RAPID-SCAN-MODE` header or `detect` option `detect.blackduck.rapid.compare.mode` and providing it one of the values below, you can change the Rapid Scan mode.

Set the compare mode of rapid scan:

- `ALL`: The default operation. It will evaluate policy rules that are `RAPID` or (`RAPID` and `FULL`). When the header is absent, this is the default behaviour.
- `BOM_COMPARE`: Will evaluate policy rules like the `ALL` option, but will now evaluate differently based on the type of policy rule modes. When the policy rule is (`RAPID` and `FULL`) it will behave like `BOM_COMPARE_STRICT` but if the policy rule is only (`RAPID`) it will evaluate the result of the rapid scan against the policy ignoring results in the BOM.
- `BOM_COMPARE_STRICT`: Will only evaluate policy rules that are (`RAPID` and `FULL`). Policy violations are compared to the existing project version BOM. If the policy violation was already known and visible in the BOM (active or overridden) it is not part of the rapid scan positive result, it will still be part of the full result following existing restrictions.

In order to run either of the `BOM_COMPARE` modes there must be an existing project version in HUB.

Enabling full results data for BOM support

You can configure Rapid Scan to provide a full results format to include data points for BOM support. To do so, set the following environment variable: `BLACKDUCK_RAPID_SCAN_EXTENDED_DATA=true`. The additional data points include:

-
- | | | |
|-------------------------------|------------------------------|------------------------|
| • componentDescription | • Release date | • SPDX license ID |
| • Homepage link | • Meta links to component id | • Source (NVD or BDSA) |
| • OpenHub link | component version id and | • Match type |
| • Declared License definition | component version origin | |
| | URIs | |
| | • External namespace | |
| | • Package URL | |

Using the Signature Scanner

The Signature Scanner is the default method for scanning your code. However, you may also use Black Duck Detect or Black Duck Detect (Desktop) to scan your code.

Signature Scanner client requirements

A Windows 7 or later, Mac OS X 10.9 or later, or Linux 64-bit system is required to run Signature Scanner. Client systems must have a minimum of 6 GB of RAM.


Downloading and installing the Signature Scanner CLI

Downloading the Signature Scanner CLI

The Signature Scanner CLI is packaged as a .zip file. Download it from the Black Duck application.

Before downloading the Signature Scanner CLI, be sure that:

- Your Black Duck license is enabled for Component Scanning.
- Your Black Duck account has the Global or Project Code Scanner role.

 **Note:** Java Runtime Environment (JRE) is included with the download of Signature Scanner. However, there may be situations that require you to use your version of JRE, for example you have self-signed certificates stored in a preferred version of Java or your company policy only allows you to run a specific version of JAVA or JRE. In these instances, you need to [set the BDS_JAVA_HOME environment variable](#) prior to running Signature Scanner.

To download the Signature Scanner CLI from the Black Duck user interface:

1. Log in to Black Duck.
2. Navigate to the drop-down menu under your username, and select **Tools**.
3. On the Tools page under **Legacy Downloads**, click the expand arrow to view and select the download link for the Linux, Mac OS X, or Windows CLI of the Signature Scanner.

Installing the Signature Scanner CLI

Install the scanner on the computer that contains the archives to be scanned. You cannot scan archives on a remote server.

To install the Signature Scanner CLI:


1. Unzip the Signature Scanner CLI.
The following is the directory structure for Windows:

<input type="checkbox"/> Name	Type
 bin	File folder
 jre	File folder
 lib	File folder

Defining your version of JRE for Signature Scanner

The Java Runtime Environment (JRE) is included with the download of the Signature Scanner. As a result, you do not need to configure the JRE or the JAVA_HOME environment variable.

However, there may be situations that require you to use your version of JRE, for example you have self-signed certificates stored in a preferred version of Java or your company policy prohibits using the version of the JRE included with Signature Scanner. In these instances, you can set the BDS_JAVA_HOME environment variable to define the installed version of JRE Signature Scanner should use. The Signature Scanner will then use this version when scanning components.

 **Note:** If you do not configure BDS_JAVA_HOME, Signature Scanner uses the version of JRE packaged with the download of the Scanner.

To configure the BDS_JAVA_HOME environment variable on Windows:

3. Understanding Component Scanning • Using the Signature Scanner

1. Access the System Properties dialog box. For example, from the Control Panel, click **System > Advanced System Settings**.
2. Select the **Advanced** tab and click **Environment Variables**.
3. In the Environment Variables dialog box, under **System Variables**, click **New**.
4. Enter the following information:
Variable name: BDS_JAVA_HOME
Variable value: <path to JRE>
5. Click **OK**.


To configure the BDS_JAVA_HOME environment variable on Linux or Mac OS X:

1. Start a terminal session.
2. At the command line, type

```
export BDS_JAVA_HOME=<path to JRE>
```
3. Close the terminal session.

Running a component scan using the Signature Scanner command line

You run a component scan to identify the components contained in an archive or a directory of files.

 **Note:** An error message appears if you exceed the scan size limit, which is 5 GB. Contact Customer Support if you receive this message.

The usage is:

```
scan.cli.bat [parameter1]...[parameterN]...<scan_path>
```

Parameter	Description
-?, --help	Shows help for this tool.
<scan_path>	Path to the file directory location or archive that you want to scan.
--bdio2	Use BDIO 2's JSON-LD format for storing and transmitting scan evidence to the Black Duck server.
--binaryAllowedList <file extensions> --sourceAllowedList <file extensions>	Use to create approved signature lists . <ul style="list-style-type: none">• --binaryAllowedList <i>x, y, z</i> where <i>x, y, z</i> are the approved file extensions for SHA-1 (binary) files.• --sourceAllowedList <i>a, b, c</i> where <i>a, b, c</i>, are the approved file extensions for clean SHA-1 (source code) files.
--cloneFrom <version>	Specifies the name of an existing project version to use as a clone . To clone a project version, use the: <ul style="list-style-type: none">• --project parameter to specify the project you wish to clone from.• --release parameter to specify the new project version.• --cloneFrom parameter to specify the project version to use as a clone. For example, to clone version 1.0 of project SampleProject to a new version called 2.0, you would include these parameters:

Parameter	Description
	<code>--project SampleProject --release 2.0 --cloneFrom 1.0</code>
<code>--context <context></code>	Additional URL context. Use this parameter, for example, if the X-Forwarded-Prefix header is being specified in a proxy server/load balancer configuration.
<code>--copyright-search</code>	Enables copyright text detection .
<code>--correlationId <value></code>	Used by the Black Duck system to provide the ability to correlate results from different types of scan (package manager scan and signature scan) for the same code location to improve the reliability of results.
<code>--dryRunReadFile <data directory></code>	Specifies the directory, including the file name, from a dryRun scan and posts the scan to the Black Duck server.
<code>--dryRunWriteDir <data directory></code>	Specifies the directory to which the scanner outputs a BDIO file with the original file metadata used for scanning. The scanner does not connect to or post the scan to the Black Duck server. Note that the <code>data</code> directory is created inside the specified directory.
<code>--exclude <pattern></code> <code>--exclude-from <Filename></code>	<p>Excludes a directory or several directories from scanning. The scanner automatically excludes these directories and the contents of these directories:</p> <ul style="list-style-type: none"> • CVS • .svn • .git • .hg • .bzz • __MACOSX <p>The scanner automatically excludes files named:</p> <ul style="list-style-type: none"> • .cvsignore • .git • .gitignore • .gitattributes • .gitmodules • .hgignore • .hgsub • .hgsubstate • .hgtags • .bzrignore • vssver.scc • .DS_Store <p>To exclude other directories, use --exclude to exclude a single directory; --exclude-from to specify a file that lists directories that should be excluded.</p> <p>Exclusion guidelines:</p> <ul style="list-style-type: none"> • Leading and trailing forward slashes are required. For example, if you enter <code>exclude /directory</code>, a warning message will appear and the directory will not be excluded. If you enter <code>/directory</code> in the file, the directory will not be excluded.

Parameter	Description
	<ul style="list-style-type: none"> • Directory names cannot contain double asterisks (**). • Specify one directory per line in the file. Include the complete direct path. The path must be a relative path rather than an absolute path. • You cannot exclude archives or contents within archives. <p>There are two additional methods you can use to exclude directories from scanning:</p> <ul style="list-style-type: none"> • Create an <code>ignore</code> file located in the <code>\$HOME/config/blackduck</code> directory. Use this file to list excluded directories, relative to root. This option provides you with the ability to use one location to list all directories that need to be excluded. Lines in the <code>ignore</code> file must have the path from source root to the ignored directory, may have multiple subdirectories, and must have leading and trailing forward slashes (/). Create one of the following environment variables, as shown here, configured for Linux or Mac OS X: <ul style="list-style-type: none"> • <pre>export JAVA_TOOL_OPTIONS=" -Duser.home=<path>"</pre> <p>The <code>.ignore</code> file must be located here: <user.home>/ .config/blackduck/ignore</p> • <pre>export XDG_CONFIG_HOME=<path></pre> <p>The <code>.ignore</code> file must be located here: <code>\$XDG_CONFIG_HOME/blackduck/ignore</code></p> • <pre>export JAVA_TOOL_OPTIONS="-Dblackduck.scan.excludesFile=<path>/<file>"</pre> <p>The <code>.ignore</code> file can be in any location and have any name.</p> <p>For Windows systems, use the Control Panel to access the Advanced System Settings dialog box to create the environment variables.</p> • Create individual <code>.bdignore</code> files which can be located in any directory. Use this file to list the excluded subdirectories in the directory where the <code>.bdignore</code> file is located. You must create a <code>.bdignore</code> file in each directory that has subdirectories you want to exclude. <p>You must also follow the exclusion guidelines as described above when using either of these methods.</p> <p>Tip: Use the <code>debug</code> parameter when excluding directories to ensure that the scanner visited and excluded the directory.</p>
<p>--fs-wait-time <number of minutes></p>	<p>Number of minutes the scan client waits for the File System scan (signature scan) to be in either the completed or error status before starting the snippet or string search scan. Must be a positive integer number.</p>
<p>--host <host></p>	<p>Server hosting the Black Duck installation.</p>

Parameter	Description
--individualFileMatching <option>	<p>Individual file matching is the identification of a component based purely upon the checksum information of a single file. By default, individual file matching is disabled. To enable individual file matching, select one of the following options:</p> <ul style="list-style-type: none"> • source. Performs individual file matching only on files with this extension: <code>.js</code>, <code>c</code>, <code>h</code>, <code>c++</code>, <code>cc</code>, <code>cpp</code>, <code>cxx</code>, <code>hh</code>, <code>hpp</code>, <code>hxx</code>, <code>h+</code>, <code>cs</code>, <code>idl</code>, <code>rc</code> • binary. Performs individual file matching on files with these extensions: <code>.apklib</code>, <code>.bin</code>, <code>.dll</code>, <code>.exe</code>, <code>.o</code>, and <code>.so</code>. • all. Performs individual file matching on all files with extensions matching <code>"*"</code>. • both. Performs individual file matching on file extensions only using both approved signature lists. <p>Any other value will be ignored and individual file matching will remain disabled. Click here for more information on using this parameter with approved signature lists.</p>
--insecure	<p>Ignores TLS validation errors, allowing the scanner to connect to the Black Duck server.</p>
--license-search	<p>Enables searching for embedded licenses.</p>
--logDir <log directory>	<p>Location of the <code>log</code> directory which contains all scanner log files. You must specify the --logDir parameter for log files to be saved.</p>
--matchConfidenceThreshold <value>	<p>Specify the match confidence threshold as a percentage value between 1 - 100. A low value returns more matches; whereas a high value returns fewer matches.</p>
--max-request-body-size <size> --max-update-size <size>	<p>Controls how scan data is streamed (buffered) from the Signature Scanner to Black Duck.</p> <p>In rare cases, you may need to modify these values to better suit your network, for example, decreasing the values if there are issues with your network or increasing the default values if your network is highly stable.</p> <ul style="list-style-type: none"> • --max-request-body-size. Size of the main request that uploads the scan data for scanned paths. Specify a value, in bytes. The default is 20000000 bytes. The recommended minimum value is 2000000 bytes; the recommended maximum value is 200000000 bytes. • --max-update-size. Buffers an update request to inform Black Duck when Signature Scanner has completed uploading the data of individual URIs (scanned paths). Specify a value, in bytes. The default value of 10000 bytes. The recommended minimum value is 1000 bytes; the recommended maximum value is 1000000 bytes.
--min-scan-interval =<time in hours>	<p>Minimum scan interval setting (in hours), which may be used to limit daily rate of the signature scan for given code location. If set to greater than 0, signature scans will not be processed if they occur before the set scan interval.</p>

Parameter	Description
--name <scan name>	<p>Unique name identifying this scan. This name is displayed on the Scans page. Click here for more information.</p> <p>Note: The --name parameter is not supported when specifying multiple scan paths in a single command line.</p>
--no-prompt	<p>Non-interactive mode.</p> <p>Instead of the --no-prompt parameter, you can set the <code>BD_HUB_NO_PROMPT</code> environment variable to enable non-interactive mode.</p>
--no-signature-generation	<p>Use the traditional Signature Scanner instead of the Enhanced Signature Generation.</p>
--password <password>	<p>Forces the scanner to prompt you for the password for the user account with the code scanner role:</p> <ul style="list-style-type: none"> • Specifying the --password parameter without the <i>password</i> value results in the scanner prompting you for the password. • Specifying the <i>password</i> value displays a warning message notifying you that specifying the password on the command line will not be supported in future versions of Black Duck; the scan then runs. <p>Set the <code>BD_HUB_PASSWORD</code> environment variable with the Black Duck server password instead of passing an argument to the --password parameter:</p> <ul style="list-style-type: none"> • If you set this environment variable <i>and</i> specify the --password parameter, the scanner prompts you for the password; it does not check the password value against the value specified in the environment variable. • If you set this environment variable <i>and do not</i> specify the password parameter, the scanner does <i>not</i> prompt you for the password. <p>Important: Set the <code>BD_HUB_PASSWORD</code> environment variable with the Black Duck server password. If you supply the password parameter, an error message appears and the scan will not complete.</p> <p>If this environment variable is <i>not</i> set, the scanner prompts you for the password whether you include or omit the --password parameter.</p> <p>Note: If the password parameter is the parameter immediately before <scan_path> use <code>--</code> to indicate you are finished passing parameters, for example <code>--password -- <scan_path></code>. Otherwise, the scanner will try to use the <scan_path> value as the password.</p> <p>Instead of specifying a username and password, use the BD_HUB_TOKEN environment variable to specify a Black Duck API token.</p>
--port <port>	<p>Port on which the Black Duck server instance is listening.</p>
--project <project>	<p>Name of the project to which you want to map the scan results. If you specify a project, you must specify a version.</p>

Parameter	Description
	<ul style="list-style-type: none"> If the project and project version exist, the scanner maps or remaps the scan results. If the project exists, but the version does not, the scanner creates the version and maps the scan results.
--project-group <project group name>	Assigns the project to the designated project group. If the project does not already exist, it is created in the corresponding project group. This parameter has no effect if the project already exists or if the specified project group does not exist.
--release <release>	Name of the project version to which you want to map the scan results. If you specify a version, you must specify a project. <ul style="list-style-type: none"> If the project and project version exist, the scanner maps or remaps the scan results. If the project exists, but the version does not, the scanner creates the version and maps the scan results.
--retain-unmatched-files --discard-unmatched-files	Used to retain or discard, respectively, any unmatched files discovered by this scan and this scan only. If either option is supplied, project and global retention settings are ignored; otherwise, retention is determined by project or global settings as described in "Settings". Specifying both options with a single scan is an error.
--scheme <scheme>	Protocol to use to connect to the server hosting the Black Duck installation. Possible values are http or https; https is the default value. You must include <code>-scheme https</code> to specify the https protocol.
--statusWriteDir <directory>	Specifies the directory to which the scanner outputs a JSON file which contains the complete scan status information.
--selfTest	Performs a self-test; will not connect to or post the scan to the Black Duck server.
--snippet-matching --snippet-matching-only --snippet-matching-all-source --full-snippet-scan	Select one of the following for snippet matching: <ul style="list-style-type: none"> --snippet-matching. Selecting this parameter enables a two-phase approach to scanning. First, a component scan is completed whereby only files that have changed since the previous scan are scanned. Once that component scan is completed, a snippet scan runs on those newly scanned files only: if a previously scanned file has not changed, it will not be rescanned for snippets. Black Duck Software recommends using this parameter for snippet scanning. --snippet-matching-only. Selecting this parameter runs a snippet scan only on files that have changed; a component scan is not performed. You must have successfully completed a full file scan prior to selecting this parameter. --snippet-matching-all-source. Selecting this parameter runs a snippet scan for all files with supported extensions (whether they belong to unmatched directories/archives or not).

Parameter	Description
	<ul style="list-style-type: none"> • --full-snippet-scan. Selecting this parameter forces the snippet scan to search the KnowledgeBase regardless of locally cached matches from previous snippet scans. This parameter must be used with the --snippet-matching, --snippet-matching-only, or the --snippet-matching-all-source parameter: <ul style="list-style-type: none"> • With the --snippet-matching parameter: First, a signature scan is completed. Once that scan is completed, a snippet scan is performed on unmatched files or files belonging to unmatched directories/archives • With the --snippet-matching-only parameter: A snippet scan is performed on unmatched files or files belonging to unmatched directories/archives; a signature scan is not executed, but it must already exist. • With the --snippet-matching-all-source. First, a signature scan is completed. Once that scan is completed, a snippet scan is performed for all files with supported extensions (whether they belong to unmatched directories/archives or not). <p>To upload source files, you must use the --upload-source parameter, as described below.</p>
--tlscertpass	<p>Forces the scanner to prompt you for the password for the client certificate.</p> <p>You can specify the --tlscertpass parameter and/or set the <code>BD_HUB_CLIENTCERT_PASS</code> environment variable which specifies the private key password for the client certificate, for example, when --tlscert points to an encrypted PKCS #12 key store. The result of specifying the --tlscertpass parameter depends on whether the key is encrypted.</p> <ul style="list-style-type: none"> • If the key <i>is</i> encrypted, the scan will fail if you do not set the <code>BD_HUB_CLIENTCERT_PASS</code> environment variable <i>or</i> specify the --tlscertpass parameter. <ul style="list-style-type: none"> • If you set the environment variable <i>and</i> specify the --tlscertpass parameter, the scanner prompts you for the password; it does not check the password value against the value specified in the environment variable. • If the key <i>is not</i> encrypted, regardless of whether the <code>BD_HUB_CLIENTCERT_PASS</code> environment variable is set: <ul style="list-style-type: none"> • Specifying the --tlscertpass parameter forces the scanner to prompt you for the password for the client certificate. The scan will fail unless the password is empty. • If you do not specify the --tlscertpass parameter, the scan will succeed.
--tlskey <keyFile>	<p>Black Duck client certificate private key file. Automatically sets --scheme to <code>https</code>.</p> <p>Note: This parameter is optional as the key and certificate can be included in the key store file specified with --tlscert.</p>

Parameter	Description
<code>--tlscert <certFile></code>	Black Duck client certificate chain file or key store file. Automatically sets <code>--scheme</code> to https. Click here for more information on using certificate-based authentication.
<code>--upload-source</code>	Uploads the source file, an optional feature for snippet matching, embedded license search, and copyright text search. This parameter is optional with the <code>--snippet-matching</code> or <code>--snippet-matching-only</code> parameters or with the <code>--license-search</code> and/or <code>--copyright-search</code> parameters.
<code>--username <username></code>	Black Duck user account with the code scanner role . Instead of specifying a username and password, use the BD_HUB_TOKEN environment variable to specify a Black Duck API token.
<code>-V, --version</code>	Shows the version information of this tool.
<code>-v, --verbose</code>	Sets the logging level to verbose.
<code>--debug</code>	Shows debug output.
Other environment variable: • BD_HUB_TOKEN	Used to specify the Black Duck API token which is the preferred authentication method over username and password. Use the Profile page in the Black Duck UI or the <code>api-token-rest-server</code> API to manage API tokens.

Specifying the password

Set the `BD_HUB_PASSWORD` environment variable with the Black Duck server password. If you supply the `password` parameter, the scan will not complete.

About package management files

By default, the scanner does not include components declared in supported package management files. Use Black Duck Detect to discover declared dependencies.

Exit Statuses

The possible exit statuses are:

- **0:** SUCCESS. The export completed successfully.
- **1:** FAILURE. Generic failure.
- **2:** NOT_EXECUTED. Returned by the scan client when the configured minimum scan interval is not exceeded and the scan was not executed. See the `--min-scan-interval` command line argument above for additional details.
- **64:** USAGE. The command to run the tool was used incorrectly, for example, with the wrong number of arguments or a bad syntax.
- **65:** DATA_ERROR. The input data was incorrect.
- **66:** NO_INPUT. An input file (not a system file) did not exist or was not readable.
- **67:** NO_USER. The specified user does not exist.
- **68:** NO_HOST. The specified host does not exist.
- **69:** UNAVAILABLE. A service is unavailable.

3. Understanding Component Scanning • Using the Signature Scanner

- **70: SOFTWARE.** An internal software error has been detected.
- **71: OS_ERROR.** An operating system error has been detected.
- **72: OS_FILE.** A system file does not exist, cannot be opened, or has some sort of error, for example a syntax error.
- **73: CANNOT_CREATE.** An output file cannot be created.
- **74: IO_ERROR.** An error occurred while doing input/output on a file.
- **75: TEMPORARY FAILURE.** Temporary failure,
- **76: PROTOCOL.** The remote system returned something that was "not possible" during a protocol exchange.
- **77: NO_PERMISSION.** You did not have sufficient permission to perform the operation.
- **78: CONFIGURATION.** Something was found in an unconfigured or misconfigured state.
- **79: NO_REGISTRATION.** Registration to Black Duck or Protex was not valid.

You can also find more information about these exit codes [here](#).

Examples

The following are examples of using the command line to run the Signature Scanner CLI.

- Scanning and sending scan data to Black Duck
- Scanning and mapping the scan data

Note that:

- In all examples, the user has a code scanner role. Contact your Black Duck administrator for more information.
- The examples show only required parameters.

To scan and send the scan data to Black Duck:

1. Open a command prompt.
2. Go to the directory where the Signature Scanner is installed.

For example:

Linux/MAC OSX:

```
/opt/blackduck/hub/scan.cli-2024.7.3/scan.cli-2024.7.3/bin
```

Windows:

```
C:\scan.cli-2024.7.3\scan.cli-2024.7.3\bin
```

3. Run the following command to configure and initiate the scan.

For example:

Linux/Mac OSX:

```
./scan.cli.sh --username <username> --host <host> --port <port> <scan_path>
```

Windows:

```
scan.cli.bat --username <username> --host <host> --port <port> <scan_path>
```

The Signature Scanner sends the scan data to Black Duck's server. Log in to Black Duck to , which adds the identified components to the project BOM.

To scan over HTTPS, sending the scan data to Black Duck, and automatically mapping scan to a project:

1. Open a command prompt.
2. Go to the directory to which the scanner is installed.

Linux/MAC OSX:

```
/opt/blackduck/hub/scan.cli-2024.7.3/scan.cli-2024.7.3/bin
```

Windows:

```
C:\scan.cli-2024.7.3\scan.cli-2024.7.3\bin
```

3. Run the following command to configure and initiate the scan.

Linux/Mac OSX:

```
./scan.cli.sh --username <username> --host <host> --port <port> --scheme HTTPS --project <project> --release <release> <scan_path>
```

Windows:

```
scan.cli.bat --username <username> --host <host> --port <port> --scheme HTTPS --project <project> --release <release> <scan_path>
```

The Signature Scanner sends the scan data to the Black Duck server and automatically maps the scan to the version of the project you specified.

Reducing the number of parameters entered on the command line for Signature Scanner

You may need to scan numerous times using the same values for some or all of the parameters. To make this procedure easier, use the alias command in Linux and Mac OS X or the DOSKEY utility in Windows to reduce the number of parameters you must enter on the command line.

To reduce the number of parameters in Linux and Mac OS X:

Create an alias that runs Signature Scanner and specifies those parameters that will not change.

1. Open a terminal window and optionally, go to the directory where Signature Scanner is installed.
2. Create an alias. The alias command has the following syntax:

```
alias <AliasName>="<PathToCommand> --<Parameter1> <Value1> --<Parameter2> <Value2>...--<ParameterN> <ValueN>"
```

The following example contains all required parameter excluding the **<scan path>** value and password:

```
alias HubScan="./scan.cli.sh --host hostName --port 80 --username sysadmin --project projectName --release releaseNumber"
```

3. Run the alias command.

```
AliasName --<RemainingParameter1> <Value 1>... --<RemainingParameterN> <ValueN>
```

The following example runs the alias command with the password and path to the file directory specified:

```
HubScan /path/to/file/to/scan --password passwordValue
```

To reduce the number of parameters in Windows:

Use the DOSKEY utility to create a macro that executes Signature Scanner.


3. Understanding Component Scanning • Using the Signature Scanner

1. Open a command prompt and optionally go to the directory where Signature Scanner is installed.
2. Create the macro. DOSKEY has the following syntax:

```
DOSKEY <Macro_Name>=<path to command> -<Parameter1> <Value1> -<Parameter2>  
<Value2>...-<ParameterN> <ValueN> $*
```

The following example contains all required parameter excluding the **<scan path>** value and password:

```
DOSKEY HubScan=scan.cli.bat -host hostName -port 80 -username sysadmin -project  
projectName -release releaseNumber $*
```

 **Note:** DOSKEY must have \$* at the end in order to specify additional parameters when the macro is called.

3. Run the DOSKEY command.

```
DOSKEYName -<RemainingParameter1> <Value1>... -<RemainingParameterN> <ValueN>
```


The following example runs the DOSKEY command with the password and path to the file directory specified:

```
HubScan /path/to/file/to/scan -password passwordValue
```

Minimum Scan Interval

This setting allows users to change the minimum hourly frequency of which signature scans can be performed for a given code location when using the enhanced signature scanning. This will allow customers to reduce the load on their servers, thus making scans running faster and with less errors which result from overloading the server.

The default setting is set to 0, or no minimum scan interval, meaning scans are not prevented from occurring regardless of frequency. If set to greater than 0, signature scans will not be processed if they occur before the set scan interval. For example, a setting of 4 will not allow signature rescans before 4 hours of time have elapsed.

 **Note:** For users of Detect 8 and 9, Detect will only finish with a success message in this scenario if the `detect.force.success.on.skip` value has been changed to true. (Default is false). Please see [Detect's Configuration Property Details](#) page for more information.

Changing the minimum scan interval

Users with the system administrator role can change this setting by:

1. Log in to Black Duck with the System Administrator role.



Click **Admin**.

3. Select **System Settings**.
4. Click **Scan**.
5. Under **Minimum Scan Interval**, enter an integer for the number of hours between subsequent signature scans.
6. Click **Save**. To indicate that the default value has changed, the button changes to **Saved**.

Accessing the Black Duck server via a proxy

If the client running the component scans communicates with Black Duck via a proxy server, for example, the Black Duck instance is located outside of your company and your company policy requires a proxy

server, you must set a `SCAN_CLI_OPTS` environment variable prior to running the client. If this environment variable is not configured, scans will fail.

The Black Duck scan client supports Digest, Basic, and NTLM authentication.

For an HTTP proxy server:

```
SCAN_CLI_OPTS=-Dhttp.proxyHost=<ProxyHostName> -Dhttp.proxyPort=<ProxyPort>
-Dhttp.nonProxyHosts=<NonProxyHostName> -Dhttp.proxyUser=<Username> -
Dhttp.proxyPassword=<Password>
```

For an HTTPS proxy server:

```
SCAN_CLI_OPTS=-Dhttps.proxyHost=<ProxyHostName> -Dhttps.proxyPort=<ProxyPort>
-Dhttp.nonProxyHosts=<NonProxyHostName> -Dhttp.proxyUser=<Username> -
Dhttp.proxyPassword=<Password>
```

For NTLM authentication:

```
SCAN_CLI_OPTS=-Dhttp.proxyHost=<ProxyHostName> -Dhttp.proxyPort=<ProxyPort>
-Dhttp.proxyUser=<Username> -Dhttp.proxyPassword=<Password> -
Dhttp.auth.ntlm.domain=<ntlmDomain> -
Dblackduck.http.auth.ntlm.workstation=<ntlmWorkstation>
```

where

- (required) **<ProxyHostName>** The name of the proxy server host.
- (required) **<ProxyPort>** The port on which the proxy server host is listening.
- (optional) **<NonProxyHostName>** The name of any non-proxy hosts. These are servers that are trusted and do not need to go through the proxy server.
- (optional) **<Username>** Username to access the proxy server.
- (optional) **<Password>** Password to access the proxy server.
- (if required by proxy server for NTLM authentication) **<ntlmDomain>** The domain to authenticate within.
- (if required by proxy server for NTLM authentication) **<ntlmWorkstation>** The workstation the authentication request is originating from. Essentially, the computer name for this machine.

To configure the `SCAN_CLI_OPTS` environment variable in Linux or Mac OS X:

1. Start a terminal session.
2. At the command line, type


```
export SCAN_CLI_OPTS="<variable values>"
```
3. Close the terminal session.

To configure the `SCAN_CLI_OPTS` environment variable in Windows:

1. Access the System Properties dialog box. For example, from the Control Panel, click **System > Advanced System Settings**.
2. Select the **Advanced** tab and click **Environment Variables**.
3. In the Environment Variables dialog box, under **System Variables**, click **New**.
4. Enter the following information:

Variable Name: `SCAN_CLI_OPTS`


Variable value: `<Variable Values>`
5. Click **OK**.

For information on resolving proxy errors in ,Black Duck refer to [Resolving Proxy Errors](#).

Running an offline component scan using Signature Scanner

If a client does not have access to Black Duck, you can [use the command line for Signature Scanner](#) to run an offline scan to identify the open source software (OSS) components contained in an archive or a directory of files. Running an offline scan lets you:

- Use the Signature Scanner to run a scan and save the results to a data file.
- Upload the data file from a client that does have access to Black Duck to create a BOM.

 **Note:** An error message appears if you exceed the scan size limit, which is 5 GB. Contact Customer Support if you receive this message.

To run an offline component scan:

1. Be sure that you have a code scanner [role](#).
2. Using a client that has access to Black Duck, [download the Signature Scanner CLI](#) for the platform where the offline scan will occur.
3. Move the zip file to the client that does not have access to Black Duck and extract the files.
4. From the client that does not have access to Black Duck, go to the directory where the Signature Scanner is installed and enter the command to run the scan.

For example:

Linux/Mac OS X:

```
./scan.cli.sh --dryRunWriteDir <data_directory> <scan_path>
```

Windows:

```
scan.cli.bat --dryRunWriteDir <data_directory> <scan_path>
```

5. Move the `data` directory that contains the JSON file to a client that has access to Black Duck.
6. From the client that has access to Black Duck, send the scan data to Black Duck using the user interface or the Signature Scanner.

To send the data using the user interface:

1. Log in to Black Duck.
- 2.



3. In the Scans page, click **+Add** and select **Scan File**.
4. Use the Upload Scan File dialog box to locate the JSON file, and click **Close**.

To send the data using the Signature Scanner CLI:

1. Open a command prompt.
2. Go to the directory to which the Signature Scanner is installed and run the following command:
For example:

Linux/Mac OS X:

```
./scan.cli.sh --dryRunReadFile <data_directory> --username <username> --host <host> --port <port>
```

Windows:

```
scan.cli.bat --dryRunReadFile <data_directory> --username <username> --host <host> --port <port>
```


Using certificate-based authentication with Signature Scanner

You can use a client certificate, also known as a signed key pair, to authenticate to a TLS-enabled server.

From the command line, enter the `--tlscert <certFile>` and optionally the `--tlskey <keyFile>` parameters. These two parameters represent both the signed public key and the private key, respectively, used to authenticate to the TLS-enabled server.

Optionally you can specify the `--tlscertpass` parameter to force a password prompt for the client certificate or use the `BD_HUB_CLIENTCERT_PASS` environment variable to specify the password for the private key. Click [here](#) for more information.

Examples

The following are examples of using certificate-based authentication with a certificate that does and does not include a separate private key file.

Note that:

- The examples show only required parameters.
- The key is encrypted and the `BD_HUB_CLIENTCERT_PASS` environment variable has been set. Therefore, the `--tlscertpass` parameter is not included.

To use a certificate that does not include the private key (that is, a key store):

1. Open a command prompt.
2. Go to the directory where Signature Scanner is installed.

Linux/MAC OS X:

```
/opt/blackduck/hub/scan.cli-2024.7.3/scan.cli-2024.7.3/bin
```

Windows:

```
C:\scan.cli-2024.7.3\scan.cli-2024.7.3\bin
```

3. Run the following command to configure and initiate the scan.

Linux/Mac OS X:

```
./scan.cli.sh --host <host> --port <port> --tlskey <keyFile> --tlscert <certFile> <scan_path>
```

Windows:

```
scan.cli.bat --host <host> --port <port> --tlskey <keyFile> --tlscert <certFile> <scan_path>
```

To use a certificate that includes the private key (that is, a key store):

1. Open a command prompt.
2. Go to the directory where Signature Scanner is installed.

Linux/MAC OS X:

```
/opt/blackduck/hub/scan.cli-2024.7.3/scan.cli-2024.7.3/bin
```

Windows:

```
C:\scan.cli-2024.7.3\scan.cli-2024.7.3\bin
```

3. Run the following command to configure and initiate the scan.

Linux/Mac OS X:

```
./scan.cli.sh --host <host> --port <port> --tlscert <certFile> <scan_path>
```

Windows:

3. Understanding Component Scanning • Using the Signature Scanner

```
scan.cli.bat --host <host> --port <port> --tlscert <certFile> <scan_path>
```

Signature Scanner sends the scan data to the Black Duck server. Log in to Black Duck to [map the component scan to a project](#), which adds the identified components to the project BOM.

Defining the scan name

By default, the name of a scan, as shown on the Scans page, is a combination of the host name of the server that ran the scan and the path to the code. This name is created when you run the scan. You may want to specify a different name.

Some examples of why you may want to specify a scan name are:

- You are using a continuous integration build system and have multiple slave/client servers running a scan. Each slave/client server has a different host name. Depending on which slave/client server completes the scan, there can be duplicate scan files for the same scan. Your BOM may also be inaccurate as old scans are included although the code has been rescanned.

By entering a unique scan name, duplicate scan files are eliminated. Your BOM no longer contains old scans as multiple slaves/clients can now run the same scan: the newest scan replaces the existing scan as the most current scan for given code.

- You have many different build system work spaces that you scan and you want to reuse the same workspace for multiple projects. By using a different name for the scans, you can use the same workspace and have the code point to different projects.

To specify a name, use the **--name** parameter when using the [command line](#) and provide a unique name for a scan. This name appears on the Scans page.

Note the following:

- Scan names are case insensitive. Scan1, scan1, and SCAN1 are considered the same name.
- Scans with the same host and path but different names are considered different scan files.
- The host name of the server that ran the scan and the path to the code are shown in the **Scan Details** table in the *Scan Name* page.

Specifying names for BOM or JSON files

You can change the default scan name specified in BOM files (such as from Maven, Gradle, or from the Protex BOM tool) and in JSON files (such as the file that is output when using the **--dryrunWrite** parameter).

To change the existing name, open the file using an application such as Notepad and enter a new value for the **spdx:name** parameter:

```
spdx:name : "Scan Name"
```

Approved signature lists

As the Signature Scanner examines files, it generates “signatures” of the files and sends SHA-1 and clean SHA-1 signatures to Black Duck’s web application. Black Duck filters these signatures based on the individual file matching parameters (if selected) and/or allowed signature lists, which you can create. Black Duck then sends the signatures to the Black Duck KnowledgeBase (KB) web service to identify the open source software contained in the your scanned code.

You can create an allowed signature list for SHA-1 and/or clean SHA-1 file extensions. Each list is optional and works independently of the other list.

To create a list of approved signatures::

- Use one or both of the following parameters in the Signature Scanner:
 - **--binaryAllowedList** *x, y, z* where *x, y, z* are the approved file extensions for SHA-1 (binary) files.
 - **--sourceAllowedList** *a, b, c* where *a, b, c*, are the approved file extensions for clean SHA-1 (source code) files.
- Create an environment variable. The following example is for SHA-1 and clean SHA-1 signatures for Linux or Mac OS X.

```
export JAVA_TOOL_OPTIONS="-Dblackduck.scan.cli.BinaryAllowedList=x,y,z -
Dblackduck.scan.cli.SourceAllowedList=a,b,c"
```

For Windows systems, use the Control Panel to access the Advanced System Settings dialog box to create the environment variable.

If you enable individual file matching (using the **--individualFileMatching** parameter) in Signature Scanner *and* create list(s) of allowed signatures, the outcome depends on the option you select:


- **source** option
 - *Replaces* the existing file extension for the **source** option with the list of file extensions from your clean SHA-1 signature list (**sourceAllowedList**).
 - Does *not* use the list of file extensions from your SHA-1 signature list **binaryAllowedList**.
- **binary** option
 - Replaces the existing list of file extensions used for the **binary** option with the list of file extensions from your SHA-1 signature list (**binaryAllowedList**).
 - Does *not* use the list of file extensions from your clean SHA-1 signature list (**sourceAllowedList**).
- **all** option
 - Matches with all file extensions.
- **both** option
 - *Only* uses the file extensions from your SHA-1 and clean SHA-1 signature list (**binaryAllowedList** and **sourceAllowedList**).

Resolving memory issues

You may receive the following error when trying to run Signature Scanner:

```
ERROR: Insufficient memory <Value>
```

To resolve this error, increase the memory that is available for use by Signature Scanner. You can accomplish this by using the SCAN_CLI_OPTS environment variable to increase the values for the initial and maximum heap size.

 **Note:** The value you specify for the maximum heap size must be larger than that value shown in the error message.

The instructions shown below describe how to use the command line to configure the environment variable. These instructions can be adapted so you can create an alias definition in Linux or Mac OS X or use the Control Panel in Windows.

To configure the SCAN_CLI_OPTS environment variable in Linux or Mac OS X:

1. Start a terminal session.
2. At the command line, type:

```
export SCAN_CLI_OPTS="-Xms<Initial heap size> -Xmx<Maximum heap size>"
```

3. Understanding Component Scanning • Using the Signature Scanner

For example, to set the minimum size to 1 GB and the maximum to 6 GB:

```
export SCAN_CLI_OPTS="-Xms1g -Xmx6g"
```

3. Close the terminal session.


To configure the SCAN_CLI_OPTS environment variable in Windows :

1. At the command line, type:

```
set SCAN_CLI_OPTS=-Xms<Initial heap size> -Xmx<Maximum heap size>
```

For example, to set the minimum size to 1 GB and the maximum to 6 GB:

```
set SCAN_CLI_OPTS=-Xms1g -Xmx6g
```

 **Note:** There are limits in the maximum scan size when scanning with a 32-bit system as the increase in addressable memory is restricted by the limitations of the 32-bit system.

About Reduced Persistence Signature Scanning

In the source view, many users rarely look at unmatched files from signature scanning, but all of that data is by far the biggest consumer of space in the database. Performance testing also shows that inserting the file data into the database is a significant fraction of scan time.

Reduced Persistence Signature Scanning aims to decrease database size and growth and also increase scan performance by no longer saving data for files that are not matched by signature scanning. However, if your use case requires that unmatched file data be saved, there are settings to retain that data available at the global, project, and/or scan levels.


There is also the option to purge existing unmatched files to further reduce database disk usage.

Settings

Unmatched files are no longer retained by default. However, retention policy can be changed.

- **Global setting.** In System Settings under Data Retention, there is an option to [enable retention of unmatched files](#). When it is not enabled (the default), there is an option to purge all existing unmatched files from all projects. The option to purge is not presented if retention is enabled. Note that the global setting only applies to projects and scans that do not explicitly specify their own setting; similarly, changing the global setting does not affect projects or scans that do specify their own setting.
- **Project setting.** The Project Settings tab has similar options to enable retention and purge existing unmatched files, except that they only apply to files and scans done under the project. Other than scope, the primary difference from the global retention setting is that the project setting has three possibilities: (1) disabled, (2) enabled, and (3) use the global setting (the default). The global setting only affects scans under a project when option (3) is selected.
- **Per-scan setting.** Each individual signature scan can specify that unmatched files discovered by that scan be retained or not retained; see "Scan Options" below. If no retention option is provided by a scan, retention is determined by the project or global setting as described above.

Note that the determination of whether unmatched files resulting from a scan need to be retained is made at the beginning of a scan and cannot be changed afterwards.

 **Warning:** Once unmatched files are purged, they cannot be recovered except by restoring from backup.

Custom signature matching

When custom signatures are enabled for a project, unmatched files in that project must be retained for the feature to work so that scans of other projects can match against them. In such cases, retention is enabled by default and cannot be disabled unless custom signatures are disabled first.

Note that if a project's custom signature setting is changed from disabled to enabled, unmatched file retention will automatically be enabled as well. However, if retention was previously disabled, unmatched files within that project will be missing, and custom signature matching will not work as expected. In such cases, all versions in that project will need to be rescanned so that all files can be retained.

Scan options

The scan CLI has two new options, `--retain-unmatched-files` and `--discard-unmatched-files`, which will retain or discard, respectively, any unmatched files discovered by this scan and this scan only. If either option is supplied, project and global retention settings are ignored; otherwise, retention is determined by project or global settings as described in "Settings". Specifying both options with a single scan is an error.

If scanning with Detect, use one of the following arguments as appropriate:


```
--detect.blackduck.signature.scanner.arguments='--retain-unmatched-files'
```

```
--detect.blackduck.signature.scanner.arguments='--discard-unmatched-files'
```

Use cases requiring retention

Retention must be manually enabled to support two potential use cases:

1. Snippet-only scans operating on files previously discovered by a signature scan. If unmatched files from that signature scan were not retained, the subsequent snippet-only scan will be unable to scan them.

 **Note:** Snippet-only scans require unmatched file retention to be enabled.

2. Workflows requiring unmatched files be examined.

Temporary retention

Some signature scan options such as snippet matching, license search, and copyright search will require unmatched files to be retained so that those features can operate. However, if unmatched file retention was not enabled for such a scan, the unmatched files will be purged within a short time after the scan is complete.

Note that these temporarily retained unmatched files may be briefly visible in the source tree view until they are purged.

About Stateless Scanning

Stateless Scan is a scan mode that does not create or use any permanent storage within Black Duck, thus there is no bill of material (BOM) stored. It is used to quickly find policy violations within the designated scan target. In order to use the Stateless Signature Scan, you must have the following:

- Black Duck Detect 8.2.0 or later
- Black Duck 2023.1.0 or later
- Hosted KnowledgeBase
- Match as a Service must be enabled

Enabling Stateless Scan mode

Enable this feature by adding `--detect.blackduck.scan.mode=STATELESS` to a run of Detect.

Restrictions and Limitations

Stateless Scan Mode has a unique set of restrictions, mode of configuration and set of results. It is similar to Rapid Scan Mode however it differs in that it supports usage of the SIGNATURE_SCAN tool:

1. A limited subset of Tools can be run.
 - The currently supported tools are: DETECTOR, BAZEL, SIGNATURE_SCAN and DOCKER.
 - The Stateless Scan will not persist on Black Duck.
 - All other tools are disabled when running in Stateless Scan mode.
2. Stateless Scan and non-persistent SIGNATURE_SCAN
 - To perform a non-persistent Signature Scan in Stateless mode, SIGNATURE_SCAN must be included within `--detect.tools`.
 - Permitted tools omitted from the `detect.tools` list will not be run.
3. Stateless Scan requires [Black Duck policies](#).
 - Stateless Scan only reports components that violate policies.
 - If no policies are violated or there are no defined policies, then no components are returned.
4. Stateless Scan does not support `detect.policy.check.fail.on.severities`
 - Black Duck Detect will fail with `FAILURE_POLICY_VIOLATION` if any component violates Black Duck policies with a CRITICAL or BLOCKER severity.
 - Stateless Scan supports the same policy conditions as Rapid Scan. Click [here](#) for a list of policy conditions that are supported by Stateless Scan.
5. Stateless Scan does not support `detect.policy.check.fail.on.names`
6. Stateless Scan will not create a [Project](#) or [Version](#) in Black Duck.
7. Stateless Scan when running SIGNATURE_SCAN requires communication with Black Duck.

How to invoke a stateless scan

To invoke Stateless scan only:

- `--detect.tools=SIGNATURE_SCAN --detect.blackduck.scan.mode=STATELESS`

To invoke Stateless package manager scans:

- `--detect.tools=DETECTOR --detect.blackduck.scan.mode=STATELESS`
- `--detect.tools=BAZEL --detect.blackduck.scan.mode=STATELESS`
- `--detect.tools=DOCKER --detect.blackduck.scan.mode=STATELESS`
- `--detect.target.type=IMAGE --detect.blackduck.scan.mode=STATELESS`

Stateless scan results

Unlike persistent scans, no data is stored on Black Duck and all scans are done transiently. These scans are primarily intended to be fast, although the SIGNATURE_SCAN can take some time as communication with Black Duck is a requirement.

The results are saved to a json file named 'name_version_BlackDuck_DeveloperMode_Result.json' in the Scan Output directory, where name and version are the project's name and version.

```

2021-07-20 13:25:18 EDT INFO [main] --- Stateless Scan Result: (for more detail look in the log
for Stateless Scan Result Details)
2021-07-20 13:25:18 EDT INFO [main] ---
2021-07-20 13:25:18 EDT INFO [main] --- Critical and blocking policy violations for
2021-07-20 13:25:18 EDT INFO [main] --- * Components: 0
2021-07-20 13:25:18 EDT INFO [main] --- * Security: 5
2021-07-20 13:25:18 EDT INFO [main] --- * License: 0
2021-07-20 13:25:18 EDT INFO [main] ---
2021-07-20 13:25:18 EDT INFO [main] --- Other policy violations
2021-07-20 13:25:18 EDT INFO [main] --- * Components: 101
2021-07-20 13:25:18 EDT INFO [main] --- * Security: 0
2021-07-20 13:25:18 EDT INFO [main] --- * License: 0
2021-07-20 13:25:18 EDT INFO [main] ---
2021-07-20 13:25:18 EDT INFO [main] --- Policies Violated:
2021-07-20 13:25:18 EDT INFO [main] --- Security Vulnerabilities Great Than Or Equal to High
2021-07-20 13:25:18 EDT INFO [main] --- Warn on Low Security Vulnerabilities
2021-07-20 13:25:18 EDT INFO [main] --- Warn on Medium Security Vulnerabilities
2021-07-20 13:25:18 EDT INFO [main] ---
2021-07-20 13:25:18 EDT INFO [main] --- Components with Policy Violations:
2021-07-20 13:25:18 EDT INFO [main] --- Apache PDFBox 2.0.12
(maven:org.apache.pdfbox:pdfbox:2.0.12)
2021-07-20 13:25:18 EDT INFO [main] --- Handlebars.js 4.0.11 (npmjs:handlebars/4.0.11)
2021-07-20 13:25:18 EDT INFO [main] ---
2021-07-20 13:25:18 EDT INFO [main] --- Components with Policy Violation Warnings:
2021-07-20 13:25:18 EDT INFO [main] --- Acorn 5.5.3 (npmjs:acorn/5.5.3)

```

About duplicate BOM detection

Duplicate BOM detection determines if a new package manager scan duplicates the existing BOM, and if so, stops processing the scan and denotes it as complete. For high-frequency scans that generate redundant (identical) data, Black Duck's duplicate BOM detection can provide significant performance improvements.

The only indication in the Black Duck UI as to whether a scan is a duplicate is on the *Scan Name* page: for duplicate scans, the scan status is "Complete" and the number of matches is "Unchanged":

Scans

Duplicate BOM detection

Scan Details - for the last completed scan

Path	/	Match Count	6
Host	<unknown host>	Folders	0
Created on	Tue, Mar 23, 2021 11:26 AM	Files	0
Scan Size	0 B		

Delete Scan

Mapped to Project Version

Duplicate BOM detection ▾ 1.0.1

Unmap from Project

Scan History

Status	Matches	Host	Path	Scan Size	Last Updated	Scan Initiated By	
Complete	Unchanged	<unknown host>	/	0 B	Tue, Mar 23, 2021 11:28 AM	sysadmin	View BOM Import Log
Complete	Unchanged	<unknown host>	/	0 B	Tue, Mar 23, 2021 11:27 AM	sysadmin	View BOM Import Log
Complete	6 Matches	<unknown host>	/	0 B	Tue, Mar 23, 2021 11:26 AM	sysadmin	View BOM Import Log

Displaying 1-3 of 3

Note the following:

- Duplicate BOM Detection is currently for *package manager scans* only and works with any version of Black Duck Detect. No additional Black Duck Detect properties are required.

3. Understanding Component Scanning • About component dependency duplication

- This feature is automatically enabled, however, you can disable this feature. Refer to the *Installing Black Duck using Docker Swarm* guide for more information.
- Black Duck only compares a scan to recent BOMs: Black Duck will not compare a package manager scan to a BOM that is older than 7 days.
- If results were requested when configuring the scan, those results are still returned from the existing data.
- If Black Duck does not detect a duplicate BOM, scan processing proceeds as usual.
- Duplicate BOM information, such as the number of unique and total BOMs, is shown in the **usage: scan completion** section of the System Information page.

About component dependency duplication

When scanning a project, several different types of matching processes can happen. The signature match looks at the structure of directories and files and try to match the “signatures” to what’s stored in the KnowledgeBase. The snippet match looks at code snippets and looks for matches in what’s stored in the KnowledgeBase. The package manager match uses external tools to examine build configuration files to find declared dependencies and then find matching components in the KB.

After scan, match and BOM computation are completed, the Components tab will display all the components detected with the above matching processes. The Match Type column will display “Transitive Dependency” or “Direct Dependency” alone or together with other types. These components are detected by the package manager matching process. The Source column may show multiple matches representing the number of different paths in the dependency tree.

Viewing component dependency duplication

Clicking the matches link in the Sources column will direct you to the Source tab. This view has a left pane that shows the dependency tree (in addition to source code tree for signature match), and a right pane that shows components (possibly filtered) under a tree node.

With default settings, all duplicate matches of a particular component will be agglomerated into a single entry. This means that if a project has multiple paths that lead to a specific component, only one entry will be displayed in the right-hand pane of the Source tab.

Changing how duplicate dependencies are displayed

Users with the system administrator role can define the depth of displayed component duplication where 1 is no duplication and 10 will display all components up to a maximum 10 levels of relation. Please note that setting this level too high will result in reduced product performance. The default level is 1.

1. Log in to Black Duck with the System Administrator role.



3. Select **System Settings**.

4. Click **Scan**.

5. Under **Component Dependency Duplication Sensitivity**, enter an integer (1 to 10) for the number of levels to display more component dependency entries.

6. Click **Save**. To indicate that the default value has changed, the button changes to **Saved**.

Maximum limit for component matches

Black Duck uses a system property to control the maximum number of nodes (matches) per component added to resulting dependency tree in package manager scan:

```
blackduck.match.limit.per.component
```

This system property needs to be set for match engine container, therefore the following must be added to the `MATCHENGINE_SERVICE_OPTS` environment variable:

```
-Dblackduck.match.limit.per.component=<value>
```

The default value of this system property is 10, thus the number of duplicated components in the tree can not exceed the `blackduck.match.limit.per.component` value (match limit per component). The allowable range of values for this property is 1 to 100 inclusively. If the setting falls outside of that range, it will automatically be set to default value (with corresponding warning in the log).

Component Dependency Duplication Sensitivity still applies: match limit per component restricts number of duplicates above Component Dependency Duplication Sensitivity level, but if the node level is below Component Dependency Duplication Sensitivity, the match is dropped regardless (under condition that it is already added to the tree). In other words, `blackduck.match.limit.per.component` sets maximum number of duplicates that can be added to the dependency tree above Component Dependency Duplication Sensitivity level (below that level, duplicates are dropped).

For example, say a component comes as transitive dependency from many other components, so that there are 100 of these components in the dependency tree.

Black Duck parameters are set to:

- `blackduck.match.limit.per.component = 10`, and;
- Component Dependency Duplication Sensitivity = 5.

In this case, if there are 20 of said components on the level above 5, component dependency tree will have 10 of these components.

If there are 7 components on the level above 5, resulting tree will have 7 components (since remaining 93 components below level 5 will be dropped even when match limit per component is not reached).

About snippet matching

Snippets are small reusable pieces of computer code. A snippet of open source software can easily find its way into your proprietary files. For example, a developer may find a useful function from an open source program and cut and paste that code into their program.

Snippet matching is beneficial to managing legal risk and detecting possible license infringement. A snippet match occurs when a portion of code in your file matches code in one or more KnowledgeBase files.

As the use of open source software is managed through licenses that allow you to use, modify, and/or share the software under defined terms and conditions, it is important to identify the open source software used in your proprietary code so that you can manage the legal risk and detect possible license infringement. Although your proprietary code may include only a portion of open source software code, you still must comply with the license associated with that open source software.

Snippet matching finds these fragments of open source code used in your proprietary files or open source files moved into your proprietary directories and matches that code with open source code found in one or more Black Duck KnowledgeBase files. Though many of the details are proprietary, the mechanism to find snippets is through creating “codeprints” over the contents of scanned source files with a sliding window algorithm. Then, a statistically relevant sampling of those codeprints is sent to the Black Duck KnowledgeBase for matching and the results are presented to the user for review in the Black Duck UI.

Codeprints are a type of one-way cryptographic hash which cannot be decomposed back into the original source code. Codeprints are analogous to fingerprints used for crime scene investigation: a fingerprint can be used to identify a person, but a fingerprint cannot be turned into that person. Codeprints allow the Black Duck application to securely and accurately scan code for snippet reuse.

Typically, five to seven lines of average source code can generate a match depending upon the density of non-ignored characters in the line of code. The scanner will ignore white space, tabs, and other non-relevant characters (for example, lines of `*****`). One line of code can generate a match if it has enough words/characters in it (see certain javascript libraries). Very short lines of code may require more than 20 lines for a match. So, the density of the information content over those lines plays a factor into generating matches. However, other factors can also come into effect and Black Duck has a variety of rules and exclusions to optimize the scanning process and reduce false positives, but which can also impact what gets scanned and what is detected.

Click [here](#) for the list of file extensions supported for snippet scanning.

Snippet scanning process

All scanning methods have an option to enable snippet scanning. Enabling the snippet scanning option scans files not identified as open source (proprietary files). The methods to scan your code for snippets are by using:

- Signature Scanner command line
- Black Duck Detect (Desktop)
- Black Duck Detect

The process for snippet scanning is:

1. Run component analysis.

The component scan is completed first. This identifies the open source components using directory/file-level signatures.

2. Generate snippet codeprints.

If enabled, a second-pass snippet scan is performed. This scan analyzes the unmatched files in the initial component scan. For example, individually matched files or files in directories which are matched to open source components do not get scanned for snippets, as they have already been identified and to further scan them for snippets is unnecessary for the typical scanning process. The unmatched files are those which, under the component scan, did not show indications of being open source and have a file extension which indicates they are a source file. These files are the candidates for further analysis.

Note that Black Duck only analyzes the first 1MB of data for codeprints.

3. Perform snippet matching.

Snippet codeprints for the file candidates are generated and sent to Black Duck which then sends them to the Black Duck KB Snippet Matching Service. Depending upon the scan parameters selected (see below), Black Duck will send the codeprints for all files or only changed files (delta scans) to the Black Duck KB for matching. The matching service first looks for an exact file match before looking for a snippet match. If any matches are found for the file, a list of matches is produced and a heuristic is run to select the best match as a likely source. Due to the nature of using codeprints over a sliding window, fuzzy matches (inexact or modified textual areas from the original) can be detected. All the matches are then consolidated and available for review in the Black Duck UI.

4. The user reviews match details.

Unlike components detected via signature or package management scans, components detected via snippet scans are not automatically added to the BOM. This is because the source of a snippet match

can often be in many places. Black Duck attempts to choose the best match and show alternative options, but ultimately it is up to the individual users to review these matches and confirm them before they are added to their BOM. While reviewing, a user can look at the matched open source code and (optionally) compare their scanned code with the matched open source code. Please note however, that when viewing the matched area to an open source file, due to the nature of hashed-based scanning using a sliding window algorithm, the highlighted text is only an approximation of the matched area for references purposes. Parts of the match may exceed, and unmatched matched parts may be displayed, in this highlighted area.

Each snippet scanning option is discussed as follows.

Using the Signature Scanner command line


The command line has three parameters you can select for snippet matching:

- **--snippet-matching.** Using this parameter enables a two-phase approach to scanning. First, a signature scan is completed. Once the signature scan is completed, a snippet scan is performed on unmatched files or files belonging to unmatched directories/archives.

Black Duck recommends using this parameter for snippet scanning.

- **--snippet-matching-only.** When using this parameter, a snippet scan is performed on unmatched files or files belonging to unmatched directories/archives; a signature scan is not executed, but it must already exist. Its purpose is to add a snippet scan to an already existing component scan.

You must have successfully completed a full file scan prior to selecting this parameter, otherwise the scan will error.

 **Note:** Snippet-only scans require unmatched file retention to be enabled.


- **--snippet-matching-all-source.** First, a signature scan is completed. Once that scan is completed, a snippet scan is performed for all files with supported extensions (whether they belong to unmatched directories/archives or not).
- **--full-snippet-scan.** Selecting this parameter performs a snippet scan on *all* files, regardless of if they have changed or not. It effectively overrides the delta scanning capability at the cost of scan performance. On a first time scan, as all snippet candidates are analyzed for matching, this parameter will have no impact.

This parameter must be used with the **--snippet-matching** or **--snippet-matching-only** parameter:

- With the **--snippet-matching** parameter: First, a component scan is completed whereby only files that have changed since the previous scan are scanned. Once that scan is completed, a snippet scan is performed on *all* snippet candidate files.
- With the **--snippet-matching-only** parameter: A snippet scan is performed on *all* snippet candidate files; a component scan is *not* completed.

Please note that this option is not available by default for all customers. Using this option may cause significant performance and scalability issues, and should only be used in extreme situations.. If you are interested in enabling this feature on your registration key, please contact Black Duck Support for assistance.

Click [here](#) for more information on using the command line.

 **Note:** Snippet scanning cannot be completed offline as it requires communication with the Black Duck server.

Using Black Duck Detect (Desktop)

To enable scanning for snippets, select the select **Snippet Scanning** from the **Settings** options and enable it. Selecting this option runs the scanner using the command line **--snippet-matching** parameter, as described above.

Using Black Duck Detect

Use the **--detect.blackduck.signature.scanner.snippet.matching** property to enable snippet scanning in Black Duck Detect. With this property enabled, Black Duck Detect uses the command line **--snippet-matching** parameter, as described above.

Uploading source files for snippet matching

Black Duck provides the ability for you to upload your source files so that BOM reviewers can see the file contents for reviewing snippet matches from within the Black Duck UI. When source files are uploaded, Black Duck provides a side-by-side comparison of the source file to the match which can help BOM reviewers in the evaluation and review of the snippet match.

After your administrator has enabled source uploads, as described in the installation guides, use the Signature Scanner and include the **--upload-source** parameter when using the **--snippet-matching** or **--snippet-matching-only** parameter.

Reviewing snippet matches

It can be difficult to determine where a snippet of code originated; in other words, which open source supplied the snippet of code. The matching process attempts to select the best match for a snippet of code by selecting a component and version in the following order:

1. Highest KB ranked component/version.
2. Highest license risk component/version.
3. Earliest version of component by release date.
4. Component with the most versions for which a match appears.

As snippet matching is an imprecise technique, snippet matches must be reviewed prior to including these matches in your BOM. Use the **Source** tab, as described [here](#), to determine if the snippet match is relevant; in other words, does this snippet belong in your BOM? If so, determine if the snippet match is correct.

After reviewing the snippet match, add it to your BOM. The component is shown with:

- Match type = Snippet
- Usage = Source Code

Any policies you have created execute.

Retaining partial snippet identifications

By default, identifications you made to partial snippet matches are not retained in subsequent snippet rescans.

You can change this default setting so that you can minimize the number of snippet matches you need to re-identify: in the project's **Settings** tab, in the **Snippet Adjustments** section, select **Apply IDs from partial snippet matches to new exact file matches**.

Snippet matches and Vulnerabilities


Black Duck does not include any vulnerabilities related to components/versions that are identified through snippet matching *only*: vulnerabilities are not counted when showing the total number of vulnerabilities for a project/project version and are also excluded from vulnerability reports. Black Duck will add vulnerabilities/security risk identified by a snippet match if another type match type (for example, exact) identified the same component/version.

Modifying the default maximum snippet file size

By default, Black Duck only analyzes the first 1MB of data for snippet codeprints.

You can modify this default value and select a value from 1MB to 4MB.

To modify the default maximum snippet file size:

1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **System Settings**.
4. Click **Scan** in the left-hand menu.
5. In the **Snippet Max File Size** section, enter a value from 1 to 4 to set the maximum file size in MB for snippet scanning.
6. Click **Save**.

Snippet extensions

These are the file types supported for a snippet scan.

Extension	Language
.4th	Forth
.actionscrip	ActionScript
.ada	Ada
.adb	Ada
.ads	Ada
.aidl	Interface Definition Language (IDL)
.as	ActionScript
.as8	Assembly
.asm	Assembly
.asp	ASP.NET (C#)
.aspx	ASP.NET (VB)
.aug	Augeas
.awk	Awk
.bas	Classic BASIC

3. Understanding Component Scanning • About snippet matching

Extension	Language
.bash	Shell
.bat	Windows batch
.bf	Brainf*ck
.bfpp	Brainf*ck++
.bi	Structured BASIC
.bms	Text
.bmx	BlitzMax
.boo	Boo
.c	C, C++
.c#	C#
.c++	C++
.cbl	COBOL
.cc	C++
.cfc	ColdFusion
.cfm	ColdFusion
.cgi	Text
.chai	ChaiScript
.clj	Clojure
.cljc	Clojure
.cljs	Clojure
.cls	Visual Basic
.cmd	Rexx
.com	DCL
.cpp	C++
.cpy	Text
.cs	Text
.cu	CUDA
.cuh	CUDA
.cxx	C++
.d	D
.dpr	Delphi
.dylan	Dylan
.e	Eiffel

Extension	Language
.ec	eC
.eh	eC
.el	Emacs Lisp
.erl	Erlang
.es	ECMAScript
.exec	Rexx
.exheres-0	Exheres
.exlib	Exheres
.f	Text
.f77	Text
.f90	Text
.factor	Factor
.for	Text
.fpp	Text
.fr	Forth
.frag	OpenGL Shading language (GLSL)
.frm	Visual Basic
.frx	Visual Basic
.fs	F#
.g77	Fortran (free-format)
.g90	Fortran (free-format)
.glsl	Fortran (free-format)
.go	Go
.groovy	Groovy
.gs	Genie
.h	C, C++
.h++	C++
.haml	Ruby
.hh	C++
.hpp	C++
.hrl	Erlang
.hs	Haskell
.hx	Haxe

3. Understanding Component Scanning • About snippet matching

Extension	Language
.hxx	C++
.i	Fortran (fixed-format)
.i3	Modula-3
.idl	Interface Definition Language (IDL)
.inc	Text
.jar	Java archive
.java	Java
.js	JavaScript
.jsp	Java, JavaScript
.jws	Java
.l	C
.lhs	Haskell
.lisp	Lisp
.lsp	Lisp
.lua	Lua
.m	Text
.m2	Modula-2
.m3	Modula-3
.m4	Text
.ml	OCaml
.mli	OCaml
.mm	Java
.mod	Modula-2
.nb	Mathematica
.nbs	Mathematica
.octave	Octave
.pas	Pascal
.php	PHP
.php3	PHP
.php4	PHP
.php5	PHP
.phps	PHP
.phtml	PHP

Extension	Language
.pl	Prolog
.pm	Perl
.pp	Puppet
.py	Python
.r	R
.r3	Rebol
.rb	Ruby
.rc	Text
.reb	Rebol
.rebol	Rebol
.rexx	Rexx
.ru	Ruby
.s	Assembly
.sc	Scala
.scala	Scala
.scm	Scheme
.sh	Shell
.sqb	SQL
.sql	SQL
.ss	Scheme
.st	Smalltalk
.swift	Swift
.tcl	Tcl
.tk	Text
.ts	Typescript
.v	Coq
.vb	Visual Basic
.vba	Visual Basic
.vbe	VBScript
.vbs	VBScript
.vert	OpenGL Shading Language (GLSL)
.vhd	VHDL
.vhdl	VHDL

Extension	Language
.vim	Vimscript
.y	Text
.z80	Assembly
.zip	ZIP archive

Resolving proxy errors

Black Duck version 4.5.0 introduced a larger HTTP header size. The larger header size may cause problems with the load balancer. If this occurs, the larger header size may cause authentication errors in Black Duck environments running a proxy server. To prevent possible authentication errors and to support HTTP responses from Black Duck, Black Duck Software recommends increasing the allowed maximum HTTP header size in Black Duck versions 4.5.0 and higher to 8192.

About custom scan signatures

Your software projects may contain a mix of open source, third-party, and proprietary software components. While Black Duck KnowledgeBase can identify your open source components, it cannot identify third-party or proprietary software components. As such, your BOM may not include all the software components used in your code.

To ensure that your BOM tracks all your code, you can enable custom scan signatures which you can use to identify third-party and proprietary software used in your code. Once identified, and displayed in your BOM, you can track the use of proprietary code within your organization and ensure that you meet the license obligations required by your third-party software,

Understanding the custom scan signature process

Custom code signatures is an optional feature. Once enabled, the match service uses these signatures to identify where internal components are used in your applications. The internal component would need to be scanned by Black Duck with custom scan signatures enabled with Retain Unmatched File Data enabled.

Unlike Black Duck KnowledgeBase, custom scan signatures reside on your local Black Duck instance (whether the server is on premises or hosted by Black Duck).

Please note, there may be performance issues when using this feature.

Identifying custom code signatures in your code

As the scan client scans the code, it generates “signatures” of the files and directories it is scanning. After the scan completes, these signatures are initially sent to the Black Duck KnowledgeBase (KB) web service where the match service uses the signatures to identify the open source components/versions that are contained in the code being scanned. After identifying the open source components, these signatures are then sent to your local Black Duck instance where the match service compares the signatures to the custom scan signatures. After identifying the custom code signatures that are in the scanned code, the BOM is then created.

By default, custom scan signatures have been limited to the top five levels in the directory structure. System Administrators can modify the global default value. Global Project Administrators and Project Managers can modify the setting for a specific project.

Defining default scanning levels

Users with the system administrator role can define the depth of the scan, as measured by number of levels in the directory structure, from root, to perform custom signature scanning. The default level is 5.

To configure the default custom signature scanning level:

1. Log in to Black Duck with the System Administrator role.



3. Select **System Settings**.
4. Select **Scan** from the lefthand menu.
5. In the **Custom Scan Signature Level** section, enter an integer for the number of levels to perform custom signature scanning. You cannot enter 0.
6. Click **Save**. To indicate that the default value has changed, the button changes to **Saved**.

Custom scan signatures on the project level

To enable custom scan signatures for a project:


1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the **Settings** tab.
4. Check the **Custom Scan Signature** checkbox. You can also set the custom scan signature depth.

- Custom Scan Signature

Custom Scan Signature can identify third-party and proprietary software used in your code. There may be performance issues seen when using this feature.

Custom Scan Signature Depth


Depth, as measured in the number of levels in the directory structure, from root, to perform custom signature scanning for this project. The initial value is the default value defined by the System Administrator.

-  **Note:** If you remove a project owner, the user remains a member of the project. If you add a project owner who is not already a project member, Black Duck adds the user as a member.

5. Click **Save**.

Creating custom scan signatures

Custom code signatures are managed as projects and after identifying the code the custom code signatures are pulled into the BOM as a [subproject](#).

-  **Note:** A project needs to be created before and custom signatures must be enabled for that project before taking advantage of this feature. Also, ensure [purging unmatched data](#) is disabled **before** running a scan.

To create custom scan signatures:

3. Understanding Component Scanning • About custom scan signatures

1. Scan the third-party or proprietary code you wish to identify as a custom scan signature.
Use the `--blackduck.signature.scanner.individual.file.matching` property set to **ALL** in Black Duck Detect.
2. Map the scan to a project version.
3. Identify this project as a custom scan signature in the project's **Settings** tab:
 - a. Enable the feature
 - b. Optionally, select the depth, as measured by the number of levels in the directory structure, from root, to perform custom signature scanning. The value shown here is the default value, as defined by your system administrator.
 - c. Click **Save**.
4. Scan your code. The custom scan signature appears in your BOM as a subproject:

Component	Source	Match Type	Usage	License	Security Risk	Operational Risk
AAFedora 123	26 Components	Exact Directory, Exact File	Dynamically Linked	Unknown License	High	High
Abstract Rendering 0.5.1	3 Matches	Exact Directory, Exact File	Dynamically Linked	BSD-3-Clause	Medium	High
affine 1.1.0	1 Match	Exact Directory	Dynamically Linked	BSD-3-Clause	Medium	Medium
alabaster 0.7.3	1 Match	Exact Directory	Dynamically Linked	BSD 2.0 Digital License	Medium	Medium
almond 0.2.9	1 Match	Exact File	Dynamically Linked	BSD-3-Clause <small>and 1 more... </small>	High	High
anaconda-build 0.10.7	1 Match	Exact Directory	Dynamically Linked	BSD-3-Clause	Medium	Medium
anaconda-client 1.0.2	1 Match	Exact Directory	Dynamically Linked	BSD-3-Clause	Medium	Medium
anaconda-client 2.3.0	1 Match	Exact Directory	Dynamically Linked	BSD-3-Clause	Medium	Medium
ansi2html 1.1.0	1 Match	Exact Directory	Dynamically Linked	GPL-3.0+	High	High
appscript - appscript 1.0.1	2 Matches	Exact Directory, Exact File	Dynamically Linked	Public Domain	Medium	Medium
argcomplete 0.9.0	1 Match	Exact Directory	Dynamically Linked	Apache-2.0	High	High
Astropy 1.0.4	14 Matches	Exact Directory, Exact File	Dynamically Linked	BSD-3-Clause	Medium	Medium
atom 0.3.9	1 Match	Exact Directory	Dynamically Linked	BSD-3-Clause	High	High

The **Source** column displays the number of components in the subproject.

Note the following:

- If a project contains several versions of a custom scan signature project, the BOM will display only one match to one version of the custom code signature project.
- If the custom scan signature project contains open source components, values for security and operational risk may also appear in the BOM.
- Although you may have selected only one custom code signature project, if you have scanned several projects, you will experience performance issues.
- Policy violations within the subproject will not appear in the BOM. However, a policy violation will appear in the BOM for the subproject if a policy rule is violated at the project level.
- Users who do not have permission to the subproject will not be able to drill down to view additional data about that project version.
- A Custom Scan Signature filter has been added to the Project dashboard and the BOM page to help you find custom scan signature projects.

Associating custom components to custom scan signatures

1. [Create the custom component.](#)

Users with the Component Manager role can create custom components.

2. Create a custom scan signature, as described above:
 - a. Scan the code for the custom component and map the scan(s) to a project version.
 - b. In the project's **Settings** tab, select the option to enable custom scan signatures.
 - c. Define the number of levels to scan. The value shown here on the Settings tab
 - d. Click **Save**.
3. Select to view the project version created in step 2.
4. From the BOM page, select the **Source** tab and select the top node.
5. Modify the match for the custom component:
 - a. Click **Edit** to open the Edit Component dialog box.
 - b. Select the custom component created previously and click **Update**.

Click [here](#) for more information on using the **Source** tab.

Disabling custom scan signatures

If you experience significant performance degradation in scanning, you can disable this feature.

To disable custom scan signatures:

1. Clear the custom scan signature option for *all* projects.
2. Rescan your code.

Hosting location for Black Duck Detect

Managing and updating the versions of Detect across various pipeline jobs in Black Duck can be a challenge. When incompatible versions of Detect and Black Duck are used, it can take a lot of time and effort to update all jobs. Additionally, it's not always clear which version of Detect is being used or which versions are available for a given Black Duck version.

Black Duck offers two means to connect with Black Duck Detect to better suit your needs; Internally Hosted and Black Duck Hosted.

How does it work?

When Detect is invoked to scan source files, it first determines the configuration set in Black Duck (see below) and then validates the version set in Black Duck. This information is then communicated with Detect on client side.

If there is a difference between the client Detect version and the configuration set in Black Duck, Detect will pull the proper Detect version as configured in Black Duck and scan the source with the newly pulled "blackduck-detect-x.x.x.jar" instead.

- Enabling the **Internally Hosted** setting provides the option to host the Detect Binary JAR file directly on your own Artifactory to be pulled with the specific version specified for all users. If set to Internally Hosted, Detect will pull and use the version dictated in the Detect URL field.

Using this option allows integration with Code Sight and Detect, but internally hosting the Detect JAR file does not provide a complete Detect installation; it will not include any inspector scripts or inspector tools

like in a full [air-gap mode](#) installation and is not meant as an alternative to deploying Detect via air-gap mode.

Additionally, scan host machines still require access to the Internet for full functionality.

Warning: Black Duck does not validate the JAR file obtained from the provided internally hosted URL. Ensure that a valid version of the Detect JAR is available for downloaded in the hosting location.

- Using the **Black Duck Hosted** setting allows the option to use our Black Duck "sig-repo" to download the Detect version set based on the system setting configured.. If set to Black Duck Hosted, it will pull from our repository directly from client side.

Internally hosted Black Duck Detect

Users with limited external connectivity can define the internal hosting location of Black Duck Detect. Using this information, these users can leverage Code Sight for deployment across their developer base to run on-demand Software Composition Analysis (SCA) scans.

To specify the hosting location of Black Duck Detect:

1. Log in to Black Duck with the System Administrator role.



3. Select **System Settings**.
4. Click **Black Duck Detect** in the left-hand menu.
5. Click the **Internally Hosted** box.
6. In the **Hosting location for Black Duck Detect** section, enter the valid URI for your internal instance of Black Duck Detect.
7. Click **Save**.

Black Duck hosted Detect

Non-airgapped users who want Black Duck to manage the version of Detect to use can select the Black Duck Hosted option:

1. Log in to Black Duck with the System Administrator role.



3. Select **System Settings**.
4. Click **Black Duck Detect** in the left-hand menu.
5. Click the **Black Duck Hosted** box.
6. Select the desired version of Black Duck Detect from the **Black Duck Detect Version** dropdown menu.
7. Optionally, check the **Force newer versions of Detect to downgrade** box if you want to ensure users cannot perform scans with newer versions of Detect. If enabled, Black Duck Detect will downgrade to the selected version.

Note: Black Duck Detect does not support downgrading itself to versions prior to 8.9.0 because such a downgrade will lose the ability to self-update again.

8. Click **Save**.

4. Managing scans in the Black Duck UI

Use Black Duck's UI to manage scans:

- [Uploading a scan file using the Black Duck UI.](#)
- [Browsing component scans.](#)
- [Mapping a scan to a project.](#)
- [Removing a scan from a project.](#)
- [Deleting a scan.](#)
- [Viewing an audit log for a BOM file.](#)

Filtering scans

You can filter the scans on the *Scans* page by scan name, *Scan Status*, and/or by *Created Date*.

1. Log in to Black Duck.

2.



Click

3. Enter the desired text in the text field, and/or;

4. Click the **Add Filter** dropdown button and select an option in the dropdown menu.



Filtering by text

This filter allows you to view scans that contains specific text in its name.

Filtering by Scan Status

This filter allows you to view scans that match selected scan statuses. Selecting this filter opens a menu where you can select any number of statuses from the list below:

- Skipped: Scan that have been skipped.
- Complete: Scans and matching processes that are complete and a BOM is available for review.
- Not Started: Scans that have not been started.
- In Progress: Scans or the building of BOMs that are currently in progress.
- Error: Scans where an error has occurred.

Clicking the **OK** button in the dropdown menu will apply the filter to the Scans list.

Filtering by Created Date

This filter allows you to view scans that were created during the desired time frame. Selecting this filter opens a calendar selector allowing you to choose between two dates.



Removing filters

You can remove an active filter by clicking the  button to the right of the filter.

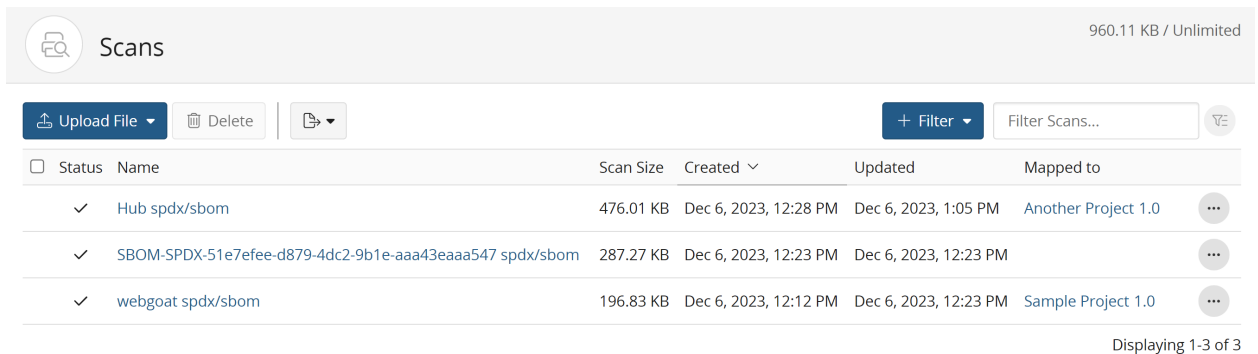
Uploading a scan file in Black Duck

If you output the scan to a file, you can import the file into Black Duck using the UI.

To upload a file:

1. Log in to Black Duck.
2. Do one of the following:

-  Click .



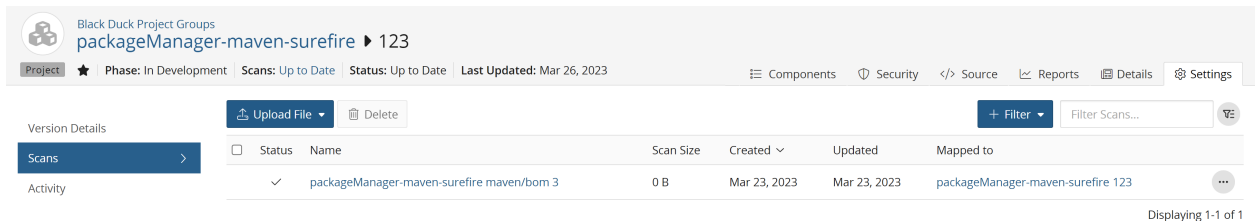
Scans 960.11 KB / Unlimited

Upload File Delete Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

- From the **Settings** tab for a project version, select **Scans**.



Black Duck Project Groups
packageManager-maven-surefire 123

Project Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: Mar 26, 2023

Components Security Source Reports Details Settings

Version Details

Upload File Delete Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	packageManager-maven-surefire maven/bom 3	0 B	Mar 23, 2023	Mar 23, 2023	packageManager-maven-surefire 123

Displaying 1-1 of 1


3. Click **Upload File** and select a file format:

- BDIO Scan: Supported file types: .json, .bdio, .bdmu.
- SBOM-SPDX: Supported file types and formats: .json, .yaml, .rdf, .spdx
- SBOM-CycloneDX: Supported file types and formats: .json

4. Upload the desired file(s) in **Upload file format** dialog box:

- Click **Browse Computer...** or anywhere inside the dotted line box and navigate to the desired report file.
- Drag the report file into the dialog box.


You can multiple files by repeating the step above. The selected report files will be listed as Queued, ready to be uploaded.


You can remove unwanted report files from the list by clicking the .

5. Optionally, you can enable the **Unmatched Component Auto-Creation** checkbox to [automatically create custom components](#) from SBOM unmatched origin IDs.
6. Click **Scan** in the Upload dialog box after uploading the file.

The **Upload file format** dialog box will remain open after the scan(s) have completed, should you want to add additional report files to scan.

7. Click **Close** to dismiss the **Upload file format** dialog box.

 **Warning:** When uploading a BDIO or SBOM file, it will override an existing BDIO or SBOM file if they share the same name. This will update the BOM of the project version to which it is [mapped](#).

 **Note:** The scan will not appear on the project version's **Settings** tab unless you mapped the scan to this project version during the scan; view the scan on the Scans page.

After uploading the file, if the scan is unmapped, use Black Duck to [map the file to a project](#).

Error code references

If the scan upload fails, it will display an error code and a message.

Table 1: External communication errors

Error code	Message
ERR01_1001	ScanCLI REST communication error
ERR01_1002	Host name could not be resolved
ERR01_1003	ScanCLI Host connection error

Table 2: Internal communication errors

Error code	Message
ERR02_1001	Unknown connection error
ERR02_1002	Scan processing timed out

Table 3: Resource allocation errors

Error code	Message
ERR03_1001	Failed to establish the validity of the server's certificate
ERR03_1002	Failed to establish a secure connection to the server
ERR03_1003	Failed to verify the server certificate for host
ERR03_1004	Unable to secure the connection to the host
ERR03_1005	File not found

Table 4: Registration errors

Error code	Message
ERR04_1001	Registration is invalid

Table 5: Internal error codes

Error code	Message
ERR05_1001	Unable to update scan status
ERR05_1002	Unable to persist results to scan database
ERR05_1017	Unrecognized error
ERR05_1018	Failed to run the scan
ERR05_1019	An error occurred while ingesting chunk
ERR05_1020	BDIO archive upload processing failed
ERR05_1021	Unable to process KnowledgeBase component matching
ERR05_1022	Unable to process KnowledgeBase signature matching
ERR05_1023	Unable to process KnowledgeBase signature polling request
ERR05_1024	kbMatchData is required for IP scans
ERR05_1025	Scan processing failed in Match Engine
ERR05_1026	Knowledge Base lookup failed in Match Engine
ERR05_1027	Signature lookup failed in Match Engine
ERR05_1028	Failed to create or update code location
ERR05_1029	Scan match creation failed
ERR05_1030	Failed to run scan auto BOM job
ERR05_1031	Snippet scan auto BOM job failed
ERR05_1032	Failed to run BOM job
ERR05_1033	Failed to transfer BDIO data
ERR05_1034	An error occurred while ingesting chunk
ERR05_1035	FS scan waiting period was surpassed
ERR05_1036	Exception thrown when trying to poll match results
ERR05_1037	Scan data was not present in the database
ERR05_1038	Error processing KB request message
ERR05_1039	Error processing KB request message
ERR05_1040	Failed saving document data for document
ERR05_1041	Failed to complete post work for scan
ERR05_1042	Scan failed while adding chunk data

Table 6: External error codes

Error code	Message
ERR06_1003	Cannot create output directory

Error code	Message
ERR06_1004	Username missing
ERR06_1005	Authorization failure
ERR06_1006	Bad exclude file
ERR06_1007	Provided bdio archive has wrong format
ERR06_1008	Failed exclude file
ERR06_1009	Failed global exclude file
ERR06_1010	Invalid key store
ERR06_1011	Failed key store
ERR06_1012	Certificate password missing
ERR06_1013	Invalid certificate password
ERR06_1014	Failed key pair
ERR06_1015	Invalid dry run file
ERR06_1016	Server scan was not found

Table 7: Stopped by user codes

Error code	Message
ERR06_1002	Scan Stopped by user

Browsing scans

You can view the results of a scan and the status of a scan that is in progress on the *Scan Name* page.

To browse component scans:

1. Log in to Black Duck.

2.  Click .

Scans page

The Scans page displays a list of all scans available in Black Duck. These can be created through Detect (as BDIO files) or by importing [SBOM files](#).

Scans 960.11 KB / Unlimited

Upload File Delete Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

The Scans page is composed of the following elements:

- The header bar contains the options to [upload new scans](#), [delete scans](#), export the scan list to CSV, and [filter the scan list](#).
- The table contains the list of scans available in Black Duck:
 - The **Status** column displays whether or not the scan was a success with a ✓ or a failure with a ▲.
 - The **Name** column displays the [scan's name](#). If the scan or upload failed, an error will be displayed in this column under the name of the scan.
 - The **Scan Size** column displays the file size of the scan.
 - The **Created** column displays when the scan was added to Black Duck. Note that this timestamp may not necessarily reflect when the scan was created. To find the creation date of the scan, click the scan and see the **Created On** timestamp in the **Scan Details** section.
 - The **Updated** column displays the date when the scan was last modified.
 - The **Mapped To** column displays the project name to which this scan is [mapped](#).

Scan name page

To view the details of a particular scan:

- Click the name of the scan in the **Name** column to open the *Scan Name* page.

Scans Demo Small 2024 - Deep License/2.0_latest bdio

Scan Details - for the last completed scan

Path	/	Match Count	44
Host	<unknown host>	Folders	0
Created on	Jul 4, 2024, 12:56 PM	Files	0
Scan Size	0 B		

Delete Scan

Mapped to Project Version
Demo Small 2024 - Deep License > 2.0_latest
Unmap from Project

Scan History

Status	Matches	Host	Path	Scan Size	Last Updated	Scan Initiated By
Complete	44 Matches	<unknown host>	/	0 B	Jul 4, 2024	sysadmin

View Import Log

Displaying 1-1 of 1

The **Scan Details** section provides the following information:

4. Managing scans in the Black Duck UI • Browsing scans

- Path: Path to the code.
- Host: Name of the machine where the latest scan was performed.
- Created on: When the scan was created. This is the specific timestamp when the scan was completed. This may not necessarily reflect the time displayed in the Scan table.
- Scan Size: File size of the scan.
- Match count: The total number of folders and files matched.
- Folders: Number of folders found in the scanned code.
- Files: Number of files in the scanned code.

The **Mapped to Project Version** section displays the project and project versions to which the scan is currently mapped. If this scan is unmapped, use the **Map Scan to Project Version** section to map this scan to a project or create a project and/or version.

The **Scan History** section displays the following information about each of the scans:

- State of a scan. Possible values are:
 - **PROCESSING**: Scanning is in progress. This is a running state. A reason will also be added to further explain the current state. These include:
 - **TOOL_SUBMISSION**
 - **USER_UPLOAD**
 - **SCAN_INGESTED**
 - **COMPLETE**: The scanner has completed the scan successfully. This is a terminal state. It will be accompanied with a transition reason explaining further how the scan was successfully completed. These include:
 - **COMPLETE**: The scan and matching process is complete and that BOM computation may proceed. Note that this status also appears if Black Duck has determined that the scan was a [duplicate](#).
 - **CLONED**: Black Duck is cloning the project version.
 - **SKIPPED**: The scan has been skipped.
 - **ERROR**: The scanner was not able to complete the scan successfully. This is a terminal state, meaning that it will be accompanied with a transition reason explaining further how the scan failed. These include:
 - **CANCELLED**: A user cancelled the scan before it was completed.
 - **ERROR_TOOL**: "Scan Submitted and Errored". The scan was submitted but an error or timeout occurred in the tool that submitted the scan and the tool is failing the scan.
 - **ERROR_SCANNING**: "Scan Error". The scan could not be completed by scanner.
 - **ERROR_SAVING_SCAN_DATA**: "Saving Scan Data Error". An error occurred when attempting to save scan data.
 - **ERROR_MATCHING**: "Matching Error". An error occurred during the matching process.
 - **ERROR_BUILDING_BOM**: "Building BOM Error". An error occurred when attempting to build the BOM. This is for migration and backward compatibility only.
 - **ERROR**: A schema error has occurred.
- Host name of the machine where the latest scan was performed.

- Path to the code.
- Scan size.
- Time the scan was created.
- User who initiated the component scan.
- View Import Log: The import log is a collection of audit records that detail information on KB component matching successes and failures for external namespaces and identifiers. An example of use would be to help identify what **components were "not found"** during the scan and subsequently not added to the BOM report as it may not be immediately obvious from looking at the BOM.

This only applies for the following scan types:

- Package manager scans
- Binary analysis scans
- Docker inspector scans
- Protex BOM import scans

Signature/snippet scans do not have this functionality and that is intended. The Source tab should be used when reviewing signature/snippet matches.

Mapping a scan to a project

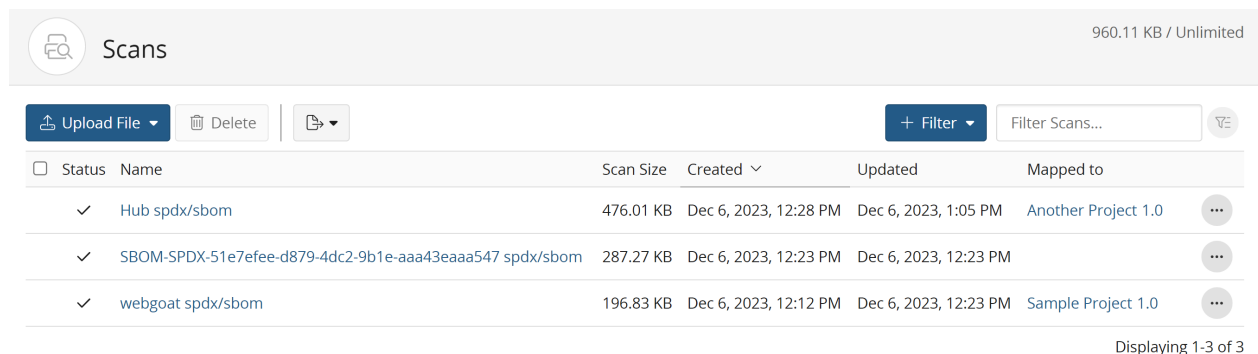
Mapping a scan adds the scan data to the BOM of a project version.

- 📌 **Note:** You can scan a Docker image or file directory location or archive more than once, but you only have to map it to a project version once. The host and path may be changed, but as long as code location name is the same, Black Duck automatically updates the BOM of the project with any new information discovered during subsequent scans.

To map a scan to a project:

1. Log in to Black Duck.
- 2.


Click .



Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

3. Do one of the following:

- Click  and select **Map to Project** in the row of the scan that you want to map.
- Select the path of the scan you want to map to open the *Scan Name* page.

4. Managing scans in the Black Duck UI • Removing a scan from a project

The screenshot shows the 'Scans' section for 'Hub spdx/sbom'. It displays 'Scan Details - for the last completed scan' with the following information:

Path	/	Match Count	390
Host	<unknown host>	Folders	0
Created on	Apr 7, 2022, 8:11 AM	Files	0
Scan Size	476.01 KB		

There is a 'Delete Scan' button below the details. To the right, the 'Map Scan to Project Version' section shows that the scan is not mapped to any versions. It includes a '+ Create Project' button, a 'Project' dropdown menu (currently showing 'Start typing to select a project...'), and a 'Version' section with the instruction 'Select a project to list its versions.' and a 'Save' button.

Below the details is a 'Scan History' table:

Status	Matches	Host	Path	Scan Size	Last Updated	Scan Initiated By	
Complete	390 Matches	<unknown host>	/	476.01 KB	7:52 AM	sysadmin	View Import Log

At the bottom right, it says 'Displaying 1-1 of 1'.

4. Start typing the name of a project to progressively display matches in the **Project** field.


If necessary, select **Create Project** to create a new project and version.

5. Select the project version to which you want to map the component scan.

If necessary, select **Create Version** to create a new version for a project.

6. Click **Save**.

Black Duck displays the name and version of the project to which you mapped the component scan. Select the link to open the [BOM page](#).

 **Note:** Black Duck displays an aggregate project version BOM. If a component version appears more than once in an archive, it is only displayed in the BOM once.

Removing a scan from a project

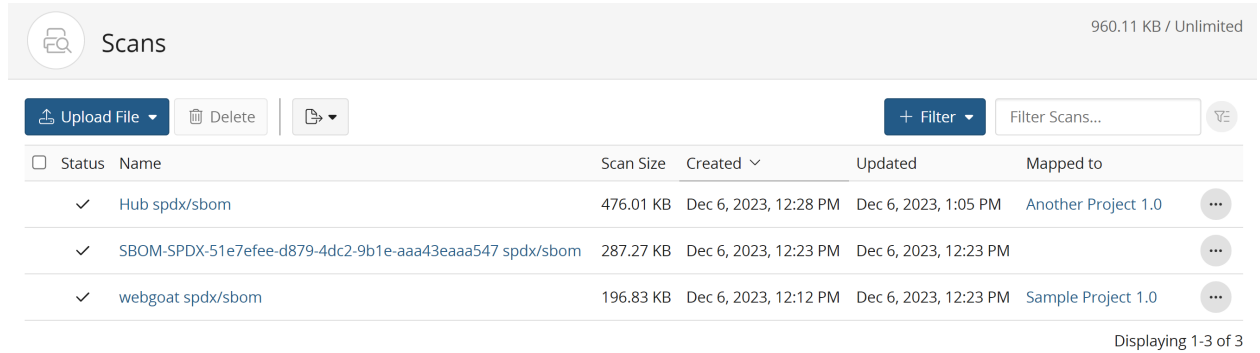
Removing the mapping of a scan removes the scan data from the BOM.

To remove a mapping:

1. Log in to Black Duck.
2. Do one of the following:

-  Click **Scans**.

4. Managing scans in the Black Duck UI • Deleting a scan



Scans 960.11 KB / Unlimited

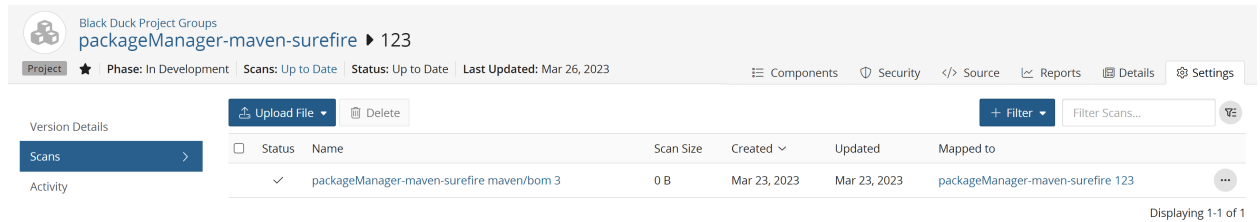
Upload File Delete

+ Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

- From the **Settings** tab for a project version, select **Scans**.



Black Duck Project Groups
packageManager-maven-surefire 123

Project Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: Mar 26, 2023

Components Security Source Reports Details Settings

Version Details

Scans


Upload File Delete

+ Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	packageManager-maven-surefire maven/bom 3	0 B	Mar 23, 2023	Mar 23, 2023	packageManager-maven-surefire 123

Activity

Displaying 1-1 of 1

- Click  and select **Unmap from Project** in the row of the scan that you want to remove the mapping.
- Click **Remove** to confirm.

Deleting a scan

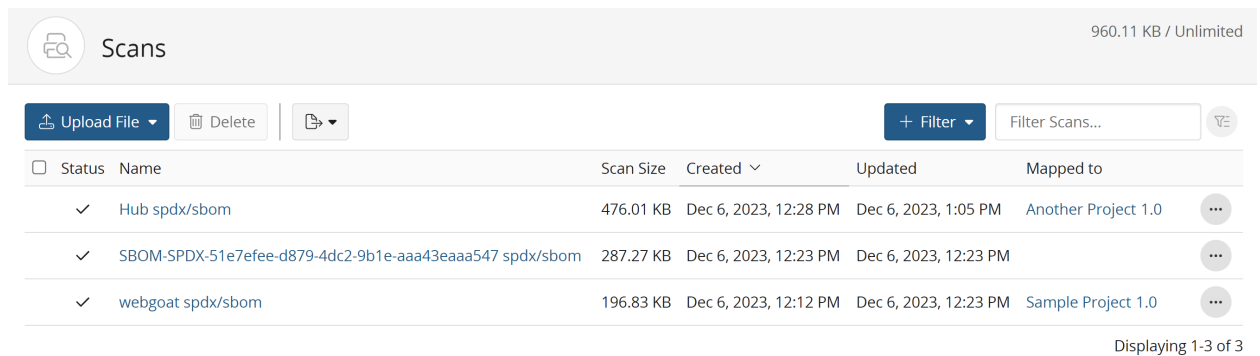
If you have scanned an incorrect path or Docker image, no longer require the scan, or want to free up space, you can delete the scan(s).

- Users with the global code scanner role can delete any scan.

To delete a scan:

- Log in to Black Duck.
- Do one of the following:

-  Click **Scans**.



Scans 960.11 KB / Unlimited

Upload File Delete

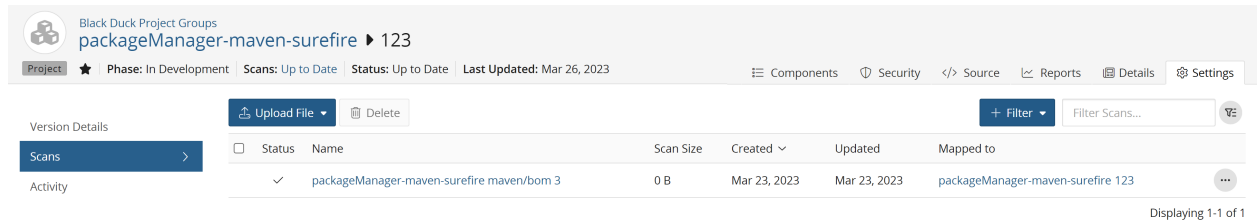
+ Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0


Displaying 1-3 of 3

4. Managing scans in the Black Duck UI • Exporting a scan file or scan results

- If the scan is mapped to a project version, from the **Settings** tab for a project version, select **Scans**.



3. Select the scan(s) you want to delete by using the checkbox(es) and click **Delete**.

You can also click  and select **Delete** in the row of the scan that you want to delete.

4. In the Delete Scan dialog box, confirm that you have selected the correct scan(s), and click **Delete**.

Black Duck removes the scan.



Exporting a scan file or scan results

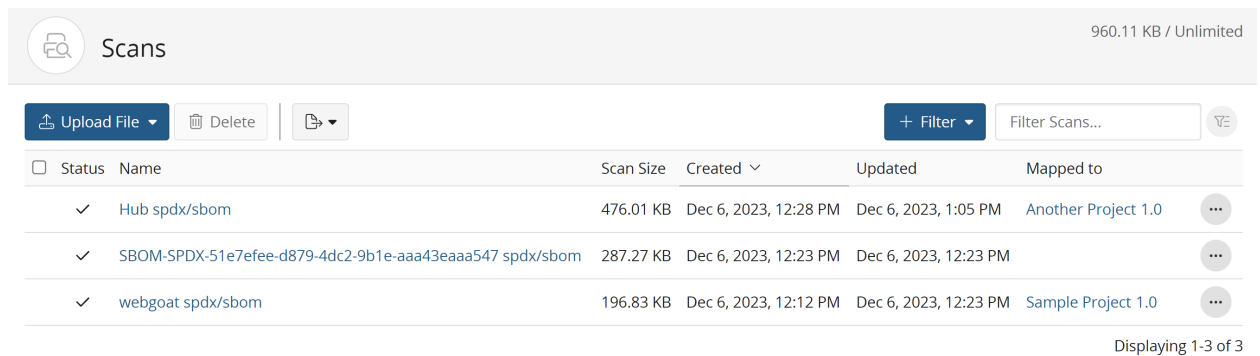
You may need a scan file, which is a file of a scan that has been imported to Black Duck, similar to a dry run file. For example, you may need to provide Customer Support with the scan file if you are experiencing scanning issues, as this file may help them investigate the issue.

- **Note:** This feature is not available if you initially scanned using Black Duck version 5.x or earlier. If the option does not appear, delete the code location and re-scan.

To export a file:

1. Log in to Black Duck.
2. Do one of the following:

-  For unmapped scans, click .



- For scans mapped to a project version, from the **Settings** tab for a project version, select **Scans**.

4. Managing scans in the Black Duck UI • Viewing an audit log for a BOM file

Black Duck Project Groups
packageManager-maven-surefire ▶ 123

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: Mar 26, 2023


Components Security Source Reports Details Settings

Version Details
Scans
Activity


Upload File Delete Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	packageManager-maven-surefire maven/bom 3	0 B	Mar 23, 2023	Mar 23, 2023	packageManager-maven-surefire 123

Displaying 1-1 of 1

3. Click  and select **Download Scan Archive** in the row of the scan that you want to obtain a scan file. The file is downloaded with a `.bdio` extension, and is a zip file.

Exporting to CSV

You can export your search results to CSV which converts the individual rows to tabular data. To do so, click the  button and select CSV.

Viewing an audit log for a BOM file

Use the View Import Log to view your results of importing a BOM file. This log lists the components and licenses that were mapped to Black Duck. It also provides details for any items that were unable to be mapped.

To view an audit log:

1. Log in to Black Duck.
- 2.



Click

The Scans page appears.

Scans 960.11 KB / Unlimited

Upload File Delete Filter Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

3. Select the scan you wish to view.


The *Scan Name* page appears. The host, path, and scan details (status of the scan, date, or time (if the date is today) the scan completed, and the username of the user that ran the scan) appear at the top of the page.

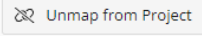
4. Managing scans in the Black Duck UI • Viewing an audit log for a BOM file

Scans
Demo Small 2024 - Deep License/2.0_latest bdio


Scan Details - for the last completed scan

Path	/	Match Count	44
Host	<unknown host>	Folders	0
Created on	Jul 4, 2024, 12:56 PM	Files	0
Scan Size	0 B		

 Delete Scan

Mapped to Project Version
Demo Small 2024 - Deep License > 2.0_latest
 Unmap from Project


Scan History

Status	Matches	Host	Path	Scan Size	Last Updated	Scan Initiated By	
Complete	44 Matches	<unknown host>	/	0 B	Jul 4, 2024	sysadmin	 View Import Log

Displaying 1-1 of 1

4. Select **View Import Log** to view the log. If this code location has been scanned multiple times, only the latest BOM Import Log will be available to view.

The BOM Import Log displays, if applicable, the number of components mapped and components not found above the table.

 **Note:** The BOM Import Log will not display licenses missing, mapped, or not found for scans imported after 2024.7.0. However, they will continue to be displayed for scans imported prior to 2024.7.0.

BOM Import Log ✕

The following is a list of components and licenses that were mapped to Black Duck. Items that failed to map will contain a description of the failure.

44 Components Mapped + Filter ▾

Import Name	Black Duck Name	Status
aopalliance 1.0 aopalliance:aopalliance:1.0	AOP Alliance (Java/J2EE AOP standard) 1.0	Component Mapped
reload4j 1.2.22 ch.qos.reload4j:reload4j:1.2.22	reload4j 1.2.22	Component Mapped
protobuf-java 3.21.9 com.google.protobuf:protobuf-java:3.21.9	protobuf-java 3.21.9	Component Mapped
mysql-connector-j 8.0.32 com.mysql:mysql-connector-j:8.0.32	MySQL Connector/J 8.0.32	Component Mapped
grizzly-framework 1.9.65 com.sun.grizzly:grizzly-framework:1.9.65	grizzly-framework 1.9.65	Component Mapped
grizzly-lzma 1.9.65 com.sun.grizzly:grizzly-lzma:1.9.65	grizzly-lzma 1.9.65	Component Mapped
grizzly-utils 1.9.65 com.sun.grizzly:grizzly-utils:1.9.65	grizzly-framework 1.9.65	Component Mapped
commons-fileupload 1.4 commons-fileupload:commons-fileupload:1.4	Apache Commons FileUpload 1.4	Component Mapped
commons-io 2.6 commons-io:commons-io:2.6	Apache Commons IO 2.6	Component Mapped

Close

Component/License Import Events & Statuses

The Status column will display whether or not the component or license was successfully imported. If a component or license was not successfully imported, an error message will explain the reason for the failure.

Tip: The Import BOM Log table can also be filtered by these statuses.

The possible statuses are as follows:

- **Component mapped.** This status indicates that the component version was successfully mapped to the Black Duck KnowledgeBase.

- **Component not found.** This status indicates that the scanned component version was not successfully mapped to the Black Duck project version because no mapping is present for the given external identifier.
- **License mapped.** This status indicates that the license was successfully mapped to the Black Duck KnowledgeBase.
- **License not found.** This status indicates that the scanned license was not successfully mapped because no mapping is present for the given external identifier in the Black Duck KnowledgeBase .
- **License missing.** This status indicates that the scanned license was not found.

Using the `GET /api/bom-import/<graphId>/component-import-events` API request

When using the `GET /api/bom-import/<graphId>/component-import-events` API request, the statuses will be displayed in the following responses. If the import was unsuccessful, the `failureReason` value will provide more information. See below for response examples.

A successful component import:

```
{
  "event": "COMPONENT_MAPPING_SUCCEEDED",
  "importComponentName": "imported component name",
  "importComponentVersionName": "imported component version",
  "componentName": "full component name",
  "componentVersionName": "full component version",
  "externalId": "external identification"
},
```

An unsuccessful component import:

```
{
  "event": "COMPONENT_MAPPING_FAILED",
  "importComponentName": "imported component name",
  "importComponentVersionName": "imported component version",
  "externalId": "external identification",
  "failureReason": "Unable to map scanned component version to Black Duck project version because no mapping is present for the given external identifier"
}
```

A successful license import:

```
{
  "event" : "LICENSE_MAPPING_SUCCEEDED",
  "protexLicenseName" : "license name",
  "externalId" : "external identification",
}
```

An unsuccessful license import may fail in one of the following reasons:

- Unable to map scanned license to Black Duck license because no external identifier is present
- Unable to map scanned license to Black Duck license because no mapping is present for the given external identifier
- Unsupported license mapping

Regardless of the failure, the API response will be formed as follows:

```
{
  "event": "LICENSE_MAPPING_FAILED",
  "protexLicenseId": "license ID",
  "protexLicenseName": "license name",
  "externalId": "external identification",
  "failureReason": "see reasons above"
}
```

5. Understanding projects in Black Duck

Black Duck helps project teams manage project information and the OSS components that are being used in each of the versions of a project.

At the project level, team members can:

- [Update the project](#) and [project version](#) information.
This information is searchable in Black Duck.
- [Manage tags associated with the project](#).
This information is searchable in Black Duck.
- [Create a new version of the project](#).
- [Manage project team membership](#).
- [Delete a project](#) or [project version](#).

The [My Projects dashboard](#) lists all projects where you are a member or where you have project-group privileges. Select the name of the project to go to the *Project Name* page which displays the **Overview** tab by default.

Black Duck Project Groups
Sample Project

Project ★ Watching Project Versions: 2 Owner: System Administrator Overview Settings

Description
No description

Created
Nov 28, 2022 by sysadmin

Tags
No Tags

Custom Fields
Reason for adding
Critical for release.

Updated
Nov 28, 2022 by sysadmin

+ Create Version + Filter Filter versions...

Version	Phase	Last Updated	Last Scanned	License	Security Risk	License Risk	Operational Risk
1.0	In Planning	11:00 AM	Never	Unknown License	————	————	————
1.1	In Planning	11:00 AM	Never	Unknown License	————	————	————

Displaying 1-2 of 2

This tab provides the following information for each version in this project:

Column	Description
N/A	<p>Icons shown to the left of the version name:</p> <ul style="list-style-type: none"> • Policy violation. Select the icon to view information on the policy violation. • Policy violation has been overridden. <p>Select the icon to view information on the policy violation.</p>
Version	Name of the project version.
Phase	<p>The development phase of this version. The possible values are:</p> <ul style="list-style-type: none"> • In Planning • In Development • Pre-release • Released • Deprecated

Column	Description
	<ul style="list-style-type: none"> Archived <p>The value in this field is used to calculate risk for the project. Archived versions are not included in project risk calculations. Click here for more information about project version phases.</p>
Last Updated	<p>When this project version was last updated. Hover over the value to see:</p> <ul style="list-style-type: none"> When the scan mapped to this version of the project was last scanned. If there are multiple scans mapped to this version of the project, this is when any of those scans was most recently scanned. When the BOM was last updated. There are several ways that the BOM could have been updated, including manual adjustments, new scans of existing code or Docker images, and newly-mapped scans.
Last Scanned	Date of the last scan for this project version. Hover over the value to see the date and time.
License	Name of the license for this project version.
Security Risk	<p>Bars show the critical, high, medium, and low security risk levels for the OSS components in this version of the project.</p> <p>Select the bar to view the number of affected components.</p>
License Risk	<p>Bars show the high (100% red), medium (50% red), and low (100% gray) license risk levels for the OSS components in this version of the project.</p> <p>Select the bar to view the number of affected components.</p>
Operational Risk	<p>Bars show the high (100% red), medium (50% red), and low (100% gray) operational risk levels for the OSS components in this version of the project.</p> <p>Select the bar to view the number of affected components.</p>

Above the table, the following information is shown:

- Description.** Description of this project. Select the **Settings** tab to create or revise the description.
- Created.** The user who created this project and the date it was created.
- Updated.** The user who last updated this project (by modifying any project information or by adding a member) and the date it was last updated.


Updates do not include adding or modifying a project version.

- Tags.** Any [tags](#) for this project.
- Additional Fields.** Project custom field information.

Creating a project


A project is the base unit in Black Duck. A project can be both a stand-alone development project and part of another project. For example, Apache Tomcat is a project in its own right but it may also be part of other, larger projects. You must create the projects that you want to make available for search by other developers in your organization.

Projects or applications are limited to 10GB of Managed Code base.

-  **Note:** If [SCM Integration](#) is enabled in your environment and you want to create a SCM project, see [Creating a SCM project](#).

To create a project:

1. Log in to Black Duck.
2. Click **+ Create Project** at the top of any page. If [SCM Integration](#) is enabled in your environment, select **Standard Project** from the menu. The **Project Details** page will display.

 **Create Project**

Project Details

Project Group

Project Name *

SCM Repository

Description

Version Details

Version Name *

SCM Branch

License

Phase *

Distribution *

3. Enter a project name. This name must be unique among projects in Black Duck, although it can have the same name as a project in Black Duck KB.

Tip: As a best practice, you should think about how other users will search for your projects when creating project names. For example, if your project is related to 3D graphics, naming it "3DGraphics" means that the user must type the entire project name in order to find your project. If you use a space or an underscore in the name, for example, "3D Graphics" or "3D_Graphics", the additional separator characters will allow users to locate the project using the search term "3D".

4. Optionally, enter additional information such as:
 - **SCM Repository:** The URL of the source code management (SCM) repository where your code resides. This field is visible only if this feature is enabled in your environment. It can be manually edited or automatically populated by Detect after completing a package manager scan. Manually changing the SCM repository URL could break an existing scan if the URLs don't match. Note that this feature is available with Detect 8.x only.
 - **Description:** As a best practice, you should think about how other users will search for your projects when creating project descriptions. The description should be specific about what the project does and how it is unique, so that it is easily distinguishable from other similar projects.
5. Type the version for this project in the **Version Name** field.
6. By default, edits to a version of this project [apply to all versions of this project](#), excluding archived versions and manually added components. Clear this option if you want edits to apply to specific versions only.
7. Click **Save**.
Black Duck displays the *Project Name* page.

Black Duck Project Groups
Sample Project

Project ★ Watching Project Versions: 2 Owner: System Administrator Overview Settings

Description
No description

Created
Nov 28, 2022 by sysadmin

Tags
No Tags

Custom Fields
No custom fields

Updated
Nov 28, 2022 by sysadmin

+ Create Version + Filter Filter versions...

Version	Phase	Last Updated	Last Scanned	License	Security Risk	License Risk	Operational Risk
1.0	In Planning	12:32 PM	Never	Unknown License	—	—	—
1.1	In Planning	12:32 PM	Never	Unknown License	—	—	—



Displaying 1-2 of 2

Deleting a project

CAUTION: Once you delete a project, you cannot restore it. You can create another project with the same name, but the new project will not have any of the version or BOM information associated with the deleted project.

To delete a project:

1. Log in to Black Duck.


2. Locate the project by using the **Watching** or **My Projects** dashboard.
3.  Click  in the project you want to delete and select **Delete**.
4. In the Delete Project dialog box, confirm that you have selected the correct project, and click **Delete**. The project is deleted.

Watching projects

If you created a project, became a project member, or became a group member of a project, you are automatically "watching" that project.

- You will receive [notifications](#) for all projects (and the components in the projects) you are watching.
- The **Watching** dashboard, one of the default dashboards, displays all your watched projects.
- Your watched projects is also a filter available on the Find page for project searches.



You can remove projects you are watching and add projects you previously stopped watching.

 **Note:** The **My Projects** dashboard lists all your projects, including those you are no longer watching.

Viewing a list of your watched projects

The **Watching** dashboard lists your watched projects. The list of your watched projects also appears on the My Profile page.

To view a list of your projects using the Watching dashboard:

1.  Click  **Dashboard**.
2. If not selected, click **Watching**.
The dashboard of your watched projects appears.

5. Understanding projects in Black Duck • Watching projects

The screenshot displays the Black Duck dashboard interface. At the top, there is a 'Dashboard' header with a menu icon and a 'Summary' link. Below this, the 'Projects' section includes tabs for 'Watching', 'My Projects', 'SCM Projects', and 'Critical security risk projects'. The 'Watching' tab is active, showing a 'Sample Project' card. This card displays four risk categories: 'No Policy Violations', '2 Critical Security Risks', '2 High License Risks', and '2 High Operational Risks'. It also shows 'Project Versions: 5 Active | 0 LTS' and 'Group: Black Duck Project Groups'. To the right of the project card is a 'Results Summary' section for '1 Project', which includes four donut charts: 'Policy Violations', 'Security Risk', 'License Risk', and 'Operational Risk'. Each chart has a legend with risk levels and their percentages. The 'Policy Violations' chart shows 100% None. The 'Security Risk' chart shows 100% Critical. The 'License Risk' chart shows 100% High. The 'Operational Risk' chart shows 100% High. A search bar with the text 'sample' is visible at the top right of the project list.

To view a list of your watched projects from the My Profile page:

1. From the user menu located on the top navigation bar, select **My Profile**.

The My Profile page appears.

2. Select **Watched Projects**.

The table with your watched projects appears.

<input type="checkbox"/>	Name	Notifications	Created	
<input type="checkbox"/>	HubTestDataProject	Disabled	Jan 2, 2024	☆
<input type="checkbox"/>	HubLongNameComponentsProject	Disabled	Jan 2, 2024	☆
<input type="checkbox"/>	QaAutoTestDataProjectWithoutScan	Disabled	Jan 2, 2024	☆
<input type="checkbox"/>	ComponentIntelligence_CapabilityAnalysis	Disabled	2:44 AM	☆
<input type="checkbox"/>	QaAutolacProject-240103-0521-r2immm	Disabled	5:21 AM	☆
<input type="checkbox"/>	QAAutoProjectPolicyOverride-240103-0327-72cddk	Disabled	3:27 AM	☆
<input type="checkbox"/>	VerifyProjectAndProjectVersionCustomFieldsAreCloned2-240102-2243-a04ffb	Disabled	3:40 AM	☆

Displaying 1-7 of 7

The table lists each project name; select the project name to view the *Project Name Overview* tab.

The **Notifications** column displays whether or not you will receive [notifications](#) for this project. You can enable or disable notifications for any project listed in this table by checking the checkbox next to the desired project or the topmost checkbox to select all projects, and then clicking either the **Enable Notifications** or **Disable Notifications**. You can also click the **Disable All Notifications** to stop receiving notifications from all projects in the list.




The **Created** column displays the date or time (if today) you become a watcher for this project. This could be date or time the project was created, you became a project member, or when you selected to watch the project.

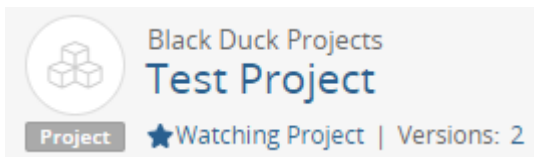
Decreasing the number of watched projects

When you stop watching a project, you will no longer receive notifications for the project and its versions and the project is removed from the **Watching** dashboard.

To stop watching a project:

Do one of the following:

- Click  in the **Watching** or **My Projects** dashboards.
The project no longer appears on the **Watching** dashboard.
The Not Watching icon () appears for the project on the **My Projects** dashboard.
- In the **Watched Projects** tab of the My Profile page:
 - Click  in the row of the project you no longer wish to watch. The project is removed from the table.
 - To easily unwatch one or more projects, click to the left of the project name and click **Stop Watching Projects**.
Click **Confirm** in the Stop Watching Project dialog box. The project is removed from the table.
- From the *Project Name* **Overview** or **Settings** tab, click **Watching Project** in the project banner.



The heading now indicates that you are no longer watching this project:





Watching projects

If you selected to stop watching a project, you can select to watch it again. You will receive notifications for the project again and the project appears on the **Watching** dashboard.

To watch a project:

Do one of the following:

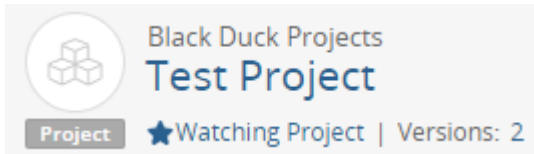
- Click  in the **My Projects** dashboard of the project you wish to watch.
The icon  now indicates that you are watching the project.

The project now appears on your **Watching** dashboard.

- From the *Project Name Overview* or **Settings** tab, click **Not Watching Project** in the project banner to watch the project.



The heading now indicates you are watching the project:



Cloning projects

Use project cloning to fork an existing project to a new project. Cloning helps reduce your workload by using the data, analysis, and resolutions you defined in an existing project as a baseline for a new project.

Users who can create projects can clone projects. For each project, select the versions you wish to clone and the project's attributes, such as the project's settings or project members and groups. Note that the attributes cloned for each project version depend on the [project version cloning](#) settings you selected, as shown in the **Cloning** section in the **Project Details** section of the **Settings** tab:

Cloning	Select the attributes you'd like to clone for any new versions of this project.
<input checked="" type="checkbox"/>	Additional Fields
<input checked="" type="checkbox"/>	Component Edits
<input checked="" type="checkbox"/>	Deep License Data
<input checked="" type="checkbox"/>	License Fulfillment Status
<input checked="" type="checkbox"/>	Remediation Details
<input checked="" type="checkbox"/>	Version Settings

Note that unlike persistent edits which synchronizes edits made in one version to all other versions of that project, edits made to the baseline project do not propagate to the cloned project or its cloned versions. This gives you the ability to experiment with the cloned project while keeping the original version intact.

To successfully use cloning:

1. Enable cloning, as described below.
2. Run a scan to the new project.

Cloned information appears in the cloned project versions for components that are the *same* as in the original project version for this project. A scan will need to be performed to replicate the components of the base project to the cloned project. If a component is not included in the newly scanned files, then that component *will not* be included in the new cloned project version. Cloned information *will* appear in the cloned project version for components that were manually added in the original project version.

Enabling cloning

To clone a project:

1. Open the *Project Name* **Settings** tab for the project you wish to clone.
2. Click **Clone Project** in the **Clone Project** section.

The Clone Project dialog box appears.

The screenshot shows a dialog box titled "Clone Project" with a close button (X) in the top right corner. The dialog contains the following elements:

- Project Name ***: A text input field.
- Version to Clone ***: A text input field with the placeholder text "Start typing to select versions to clone..."
- Copy Project**: A section header followed by the instruction "Select the project's attributes that you would like to clone".
- Four checkboxes:
 - Additional Fields
 - Application ID
 - Members and Groups
 - Settings
- At the bottom right, there are two buttons: "Cancel" (light gray) and "Clone" (dark blue).

3. Do the following:
 - Enter a name for this clone.
 - Select the versions you wish cloned.
 - Select what you would like to clone:
 - Additional Fields.
 - Application ID.
 - Members and Groups.
 - Settings. This option is selected by default. This includes the values of all attributes, excluding the project name, shown in the **Settings** section in the **Project Details** tab for this project.
4. Click **Clone**.

Updating project settings

Project team members can view and update project settings, such as:

- [Project Details](#)
- [SCM Repository](#). This field is visible only if this feature is enabled in your environment.
- [Users](#)
- [Groups](#)
- [Custom Fields](#)
- [SBOM Fields](#)
- [Activity](#)

To configure a project's settings:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the **Settings** tab.

Updating project details

From the Project Details tab, you can perform the following tasks:

- Update the Project Name, Description, Owner, or Tier.
If you remove a project owner, the user remains a member of the project. If you add a project owner who is not already a project member, Black Duck adds the user as a member.
- Configure the [ability to apply edits to all versions of a project](#).
- Configure [snippet adjustments](#).
- Update project version [cloning settings](#).
- Configure [custom scan signatures](#).
- Configure your project's [unmatched files data retention](#).
- Set how [deep license data](#) applies to your BOMs or snippet component matches.
- Set an Application ID. This field is used to store an external mapping ID for the project to an external system, such as an asset management system or application catalog.
- [Clone](#) this project.
- [Delete](#) this project

Setting a project's SCM Repository

Projects can be associated with a single SCM Repository. Once mapped, then each version can be associated with a branch in the repository. Select an SCM repository, or manually enter a repository URL.

To update the project's SCM repository, click  and select from the following options:

- Select New Repository. Choose from the list of [configured SCM providers and repositories](#).

- **Manually Enter Repository.** Enter the desired repository URL.
- **Clear Repository.** Remove the project's linked SCM repository.

Updating project members

[Manage the users and groups](#) associated to this project.

Updating a project's custom fields

If there are [custom fields created for projects](#), you can provide the requested details here.

Updating a project's SBOM fields

These are additional fields that can be included in the SBOM report. These field values will propagate when this project is used as subproject, you can override them at the BOM level. See [SBOM Project fields](#) for more information.

Viewing a project's activity audit trail

The [Activity Audit Trail](#) retains the activity audit records of user actions and key events, such as project version component and vulnerability records, in the application affecting a project and/or project version.

Managing project team membership

Once you have been added to a project team, you can add and remove other users as team members in one of two ways:

- As users:
 - [Add users to the project team](#)
 - [Remove users from the project team](#)
- As groups, which contain several users:
 - [Add groups to the project team](#)
 - [Remove groups from the project team](#)

To add users to the project team:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the **Settings** tab and then select **Members** to view the list of members for this project.

Black Duck Projects
Sample Project
Project Versions: 1

Overview Settings

Project Details

Members

Groups

+ Add Member

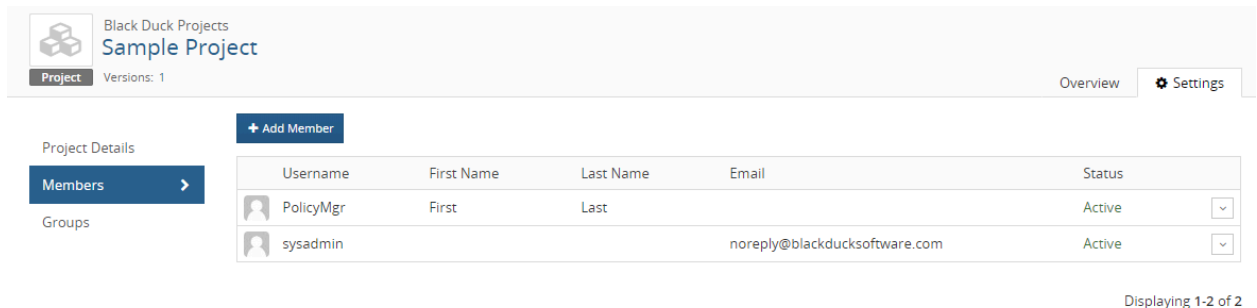
Username	First Name	Last Name	Email	Status
PolicyMgr	First	Last		Active
sysadmin			noreply@blackducksoftware.com	Active

Displaying 1-2 of 2

4. Click **+ Add Member**.
5. In the Add Member dialog box, type the name of the user that you want to add. The list is type-ahead enabled, so you can see a list of available usernames that contain the text you have typed and whether those users are active.
6. Select the username to add this user to the project team.
7. Optionally, to add multiple users, either:
 - Type and select the name of additional users.
 - Click the **Users** dropdown box to see a list of users and select the desired users.
8. Select the **roles** for this user for this project.
9. Click **Add**.
The user(s) are added to the project team.

To remove a member from the project team:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the **Settings** tab and then select **Members** to view the list of members for this project.



4. Click in the row of the user you want to remove from the project team and select **Remove**.
5. In the Remove Member dialog box, click **Remove**.
The user is removed from the project team.

To add a group to the project team:

You can manage project membership from the *Project Name* page or from the *Group Name* page.

From the *Project Name* page:

1. Log in to Black Duck.
2. Locate the project using the **Projects** tab on the Dashboard.
3. Select the name of the project to go to the *Project Name* page.
4. Select the **Settings** tab and select **Groups** to view the list of groups for this project.



5. Click **+ Add Group**.
6. Type the name of the group that you want to add. The list is type-ahead enabled, so you can see a list of available groups that contain the text you have typed and whether the group is active.
7. Select the **roles** for this group for this project.
8. Click **Add**.
The group is added to the project team.

From the *Group Name* page:

1. Log in to Black Duck.



3. Select **Users & Groups** to display the **User & Group Management** page.
4. Click the **Groups** tab.



5. Select the name of the group you want to add.

5. Understanding projects in Black Duck • Managing project team membership

Administration / Groups
Sample Group

Group Details

Overall Roles

Project Groups

Projects

Users

Group Name *

Sample Group

External User Group Name

The name of this group in the external authentication system (LDAP or SSO). Black Duck uses the external name to synchronize the group and its users with integrated authentication/authorization systems. Generally, the two group names are the same when created automatically by synchronization. However, if the group names changes on the external system, you can edit them to keep the Black Duck group name in sync with the external authentication system group name.

Active Group

Default Group

All new users will be added to this group. Users will have the roles and access to projects configured for this group.

Reset Update

Delete Group

You can delete this user group, but deleting is permanent. Users in this group will lose access to the projects and project groups assigned to this user group.

Delete Group

6. Click **Add Project** in the **Group Projects** section to display the Add Project dialog box.
7. Enter the name of the project.
8. Select the project **roles** for this group and click **Add**.

To remove a group from the project team:

You can manage project membership from the *Project Name* page or from the *Group Name* page.

From the *Project Name* page:

1. Log in to Black Duck.
2. Locate the project using the **Projects** tab on the Dashboard.
3. Select the name of the project to go to the *Project Name* page.
4. Select the **Settings** tab and then select **Groups** to view the list of groups for this project.

Black Duck Projects
Sample Project 1


Project Versions: 3

Overview Settings


+ Add Group

Group Name	Status
Htest	Active

Displaying 1-1 of 1

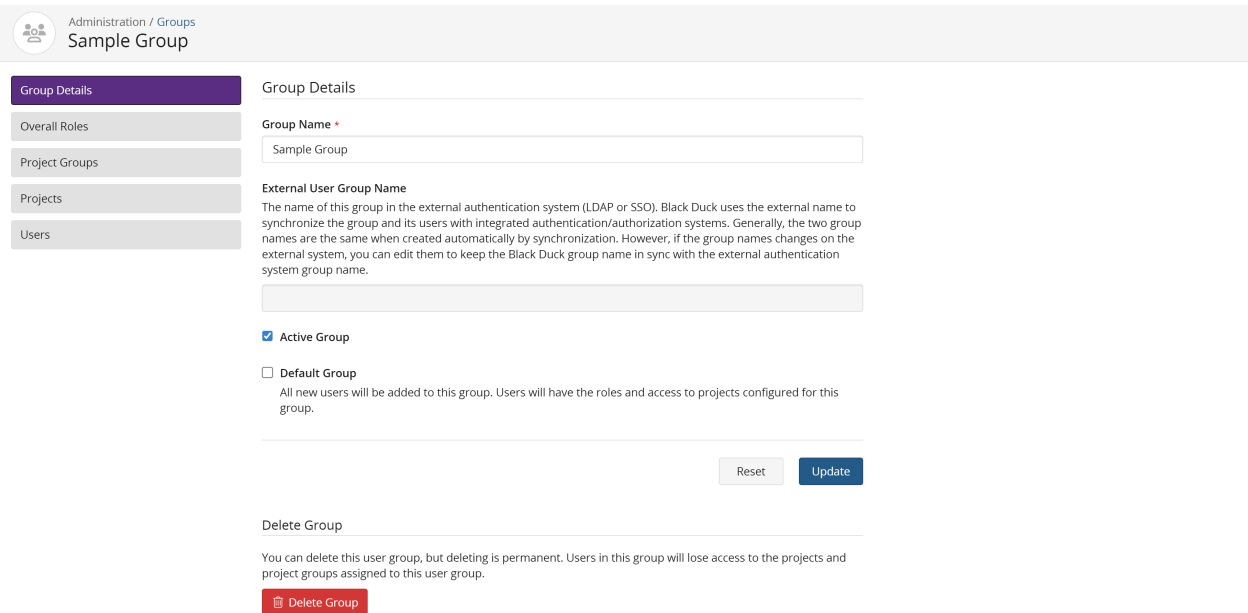
5. Click  in the row of the group that you want to remove from the project team and select **Remove**.
6. In the Remove Group dialog box, click **Remove** to confirm.
The group is removed from the project team.


From the *Group Name* page:

1. Log in to Black Duck.
2.  Click **Admin**.
3. Select **Users & Groups** to display the **User & Group Management** page.
4. Click the **Groups** tab.



5. Select the name of the group you want to remove.



6. In the **Group Projects** section, click  in the row of the group you want to remove and select **Remove**.
7. Click **Remove** to confirm.

Managing tags

You can add tags to projects and custom components to describe them and provide additional metadata, such as the programming language, frameworks, operating systems, purpose, and any other information that you think might help other users find it. Tags act as keywords when searching and filtering.

- Tags for components in Black Duck KB have been created by the users at [The Open Hub](#).
- Tags for projects are created by project team members.
- Tags for custom components are created by users with the Component Manager [role](#).

Best practices for tagging projects and custom components:

- Use a few, specific tags rather than many tags. Tags are limited to 20 for each project or custom component.
- Tags must be at least one character long (nulls not allowed) and are limited to 50 characters in length. You can use letters and numbers to create tags.
- The only special characters supported in tags are the underscore (`_`), the plus sign (`+`), and parentheses (`()`). You cannot use spaces in tags.
- Do not use punctuation unless it is necessary for the tag, for example, `C` vs. `C#` vs. `C++`.
- Use singular nouns, for example, “server” instead of “servers”.

To add tags to a project:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Type the tag in the Tags field and press **Enter**.
The tag is added to the project.

To add tags to a custom component:

1. Log in to Black Duck with the Component Manager role.

2.



Click **Manage** > **Component Management**.

The Component Management page appears.

3. Select the name of the custom component to go to the *Custom Component Name* page.
4. Type the tag in the Tags and press **Enter**.
The tag is added to the custom component.

To edit a tag:

1. Click the **Tags** field.
2. Select **X** next to the tag you wish to edit.
3. Type the revised text in the field and press **Enter**.

To remove a tag:

1. Click the **Tags** field.
2. Select **X** next to the tag.

About project versions

Use the **Details** tab to obtain information about a project version.

This tab provides the following information:

- The **Where Used** table lists the project name, project version, tier, release date, distribution, and phase for all projects where this project version is a [subproject](#).
- To the right of the table, the following information is shown:

- **Description.** Description of this project. Select the **Settings** tab for the project to create or revise the description.
- **Created.** The user who created this project version and the date it was created.
- **Updated.** The user who last updated this project version settings and the date it was last updated.
- **Last Scan.** Date and time the latest scan(s) mapped to this project version completed.
- **Last KnowledgeBase Update.** Date and time of the last KnowledgeBase update.
- **Tags.** Any [tags](#) for this project version.

To view the project version **Details** tab:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name which opens the **Components** tab.
3. Select the **Details** tab.


Creating a new version of a project

When you create a project, it has one version. You can create more project versions as needed.

To add a new project version:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.



Tip: If you wish to clone an existing version, click  in the row of the version of the project you want to clone and select **Clone**. The Clone Version dialog box appears with the information in the **Version to Clone** field completed.

3. Click **+ Create Version**.

The Create a New Version dialog box appears.

Create Version ✕

Version *

License

Notes

Nickname


Release Date


Phase *


Distribution

Retain Project Version regardless of data retention policies
Prevent this Project Version from ever being deleted by the automated data deletion policies set up by your administrator.

4. Type a name for this version of the project. This name can be a numerical release number, a text description of the version, or any combination of both.
5. From the drop-down list in the **License** field, select the license for this project version. This value is used, for example, for the license of this project version when it is a subproject.

6. In the **Notes** field, type any information about this version of the project that distinguishes it from other project versions, or that will be useful to other developers working on the version or searching for it.
7. If appropriate, in the **Nickname** field type a nickname for the project version. This might be a development code name or a shortened name by which this version of the project is commonly called.
8. If known, in the **Release Date** field, click  to select the anticipated release date for the project version or the actual date on which the project version was released.
9. From the drop-down list in the **Phase** field, select the development phase that this version of the project is currently in. The available options are:
 - In Planning (Default)
 - In Development
 - Pre-release
 - Released
 - Deprecated
 - Archived

 **Note:** The value in this field is used to calculate risk for the project. Archived versions are not included in project risk calculations. Click [here](#) for more information about project version phases.
10. From the drop-down list in the **Distribution** field, select the method by which this version of the project is being released. The available options are:
 - External (Default): An application/product that is sold to customer's externally for use. The source of the application/product is not available to the users but the product is provided externally.
 - SaaS (Software as a Service): An application/product that is shipped as a SaaS service (hosted only).
 - Internal: An application/product not sold and stays within the company. Essentially never leaves the company and gets into customer's hands.
 - Open Source: An application/product shared, essentially full open source (shared on github, etc. with full source code).

 **Note:** The value in this field is used to calculate risk for the project. Project versions that are internally distributed are not included in the risk calculations for the project.
11. Enable or disable **Data Retention**. Enabling this checkbox will prevent this Project Version from ever being deleted by the [automated data deletion](#) policies setup by your administrator.

Once the project has been created, the project version on the project's page will display a lock (🔒) icon at the end of its row to indicate that this project version is protected from automated data deletion.

5. Understanding projects in Black Duck • About project versions

The screenshot shows the 'Sample Project' page in Black Duck. At the top, there's a header with the project name and 'Watching Project' status. Below this, there are sections for 'Description' (No description), 'Custom Fields' (No custom fields), 'Created' (Nov 28, 2022 by sysadmin), and 'Updated' (Nov 28, 2022 by sysadmin). A 'Tags' section shows 'No Tags'. A '+ Create Version' button is visible. Below the buttons is a table with columns: Version, Phase, Last Updated, Last Scanned, License, Security Risk, License Risk, and Operational Risk. The table contains two rows for versions 1.0 and 1.1, both in the 'In Planning' phase, last updated at 12:32 PM, with 'Unknown License'. A '+ Filter' button and a search box are also present. At the bottom right, it says 'Displaying 1-2 of 2'.

Version	Phase	Last Updated	Last Scanned	License	Security Risk	License Risk	Operational Risk
1.0	In Planning	12:32 PM	Never	Unknown License			
1.1	In Planning	12:32 PM	Never	Unknown License			


12. Click **Save**.

Black Duck saves the project version.


Updating project version information

You can rename a project version and update the following information:

- Version
- License
- SCM Repository Branch

 **Note:** The SCM Repository Branch field is visible only if this feature is enabled in your environment. Manually changing the SCM branch name could break existing scans.

- Notes
- Nickname
- Release Date
- Phase
- Distribution
- Data Retention

 **Note:** This option is displayed only if the respective flag is set in your environment.

To update project version information:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name of the project that you want to manage. The **Components** tab for the version opens.

Black Duck Project Groups
Sample Project 1.0.0

Project: ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components

Critical 4
High 3
Medium 4
Low 0
None 17

License Risk
Number of Components

High 6
Medium 2
Low 0
None 20

Operational Risk
Number of Components

High 24
Medium 3
Low 0
None 1

Snippets
78 Unconfirmed
Unmatched Components
1 Unmatched

Add Bulk Actions Compare to...

Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter

Print Filter Components...

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4. Select the **Settings** tab and select **Version Details** to update the version information.

Note: The ability to delete a version is also available in the **Version Details** section, if there is more than one version of a project. You cannot delete a project version if that version is a [subproject in a BOM](#): you must remove the project version from all BOMs before you can delete it. Select the **Details** tab to view where this project version is used as a subproject.

5. Click **Save**.
Black Duck saves the project version information.

Cloning project versions

When creating a new project version, Black Duck now lets you select an existing project version and clone its component edits, remediation details, and/or license term fulfillment status to the new project version. Use cloning to help reduce your workload by using the analysis and resolutions you defined in an existing project version as a baseline for a new version.

Unlike persistent edits which synchronizes edits made in one version to all other versions of that project, edits made to the baseline version do not propagate to the cloned version. This gives you the ability to experiment with the cloned version while keeping the original version intact. Note that if you have enabled persistent edits, then edits made to the baseline version *will be* propagated to the cloned version.

To successfully use cloning:

1. Enable cloning, as described below.
2. Run a scan to the new version.


Cloned information appears in the cloned project version for components that are the *same* as in the original project version: if a component is not included in the newly scanned files, then that component *will not* be included in the new cloned project version. Cloned information *will* appear in the cloned project version for components that were manually added in the original project version.

By default, all options are cloned:

- Additional Fields: Custom field information.
- Component Edits:
 - Component and/or version information

- Review flag
- License
- Usage
- Ignored components
- Comments
- Manually added components
- Confirmed snippet adjustments
- Policy violation overrides and comments
- Deep License Data
- Remediation Details:
 - Remediation status
 - Target date
 - Actual date
 - Remediation comments
- License Fulfillment Status. For license terms requiring fulfillment:
 - Fulfillment status (fulfilled or unfulfilled)
 - For fulfilled license terms, the user who fulfilled the term and the date it was fulfilled
- Version Settings:
 - License
 - Notes
 - Nickname
 - Release Date
 - Phase
 - Distribution

You can modify these settings by using the **Cloning** section in the **Project Details** section of the **Settings** tab of a project, as described [here](#).

 **Note:** You cannot clone individual component or remediation values.


Enabling cloning

You enable cloning:


- Select **Clone** for the version you wish to clone from the *Project Name* page. The version of the Clone Version dialog box that appears depends on whether **Version Settings** was selected as a cloning attribute for the project:
 - If **Version Settings** was selected, specify the version name in the Clone Version dialog box and click **Clone**.
 - If **Version Settings** was not selected, specify the version name in the Clone Version dialog box, enter any of the other project version settings, and click **Clone**.
- Use the `--cloneFrom` parameter when using [the command line](#) to scan and create a project version.

Deleting a project version

You can delete a version from a project.

 **Note:** You cannot delete a project version if that version is a [subproject in a BOM](#): you must remove the project version from all BOMs before you can delete it. Select the **Details** tab to view where this project version is used as a subproject.


To delete a project version:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Click  in the row of the version of the project you want to delete and select **Delete**.
4. Click **Delete** to confirm.
Black Duck removes the project version.

About project version phases

Projects versions include a phase which you can use to manage your development projects in Black Duck. Possible phase values are:


- **In Planning:** The *In Planning* phase is where the project solution is further developed in as much detail as possible and the steps necessary to meet the project's objective are planned.
- **In Development:** The *In Development* phase is when the project plan is put into motion and the work of the project is performed.
- **Pre-release:** The *Pre-release* phase is when the project development has been completed and testing is underway.
- **Released:** The *Released* phase is when the project has officially progressed to general availability and the software product is ready to be delivered or has been delivered.
- **Deprecated:** The *Deprecated* phase is when the software has reached end-of-life and is no longer sold or supported. In this phase, the software might still be used by customers.
- **Archived:** The *Archived* phase is similar to the Deprecated phase, however the software is no longer available and customers are no longer utilizing it, having moved on to more recent releases.

 **Note:** The definitions above serve only as examples. You can use the phases in any way to suit your organization's software development life cycle steps.

You can select the phase when [creating](#) or [editing](#) a project version. By default, a project version is in the "In Planning" phase.

Black Duck treats In Planning, In Development, Pre-release, Released, and Deprecated project versions the same. Black Duck does not differentiate between these phases: these phases are to help you manage your projects. Project versions with these phases are included in project risk calculations.

Archived project versions are treated differently than the other project version phases.

 **Note:** You can "lock" a project version BOM against any component and license changes from Black Duck KnowledgeBase by select the archived phase, as described below.

About archived project versions

You can modify archived project versions, as you would a project version in any other phase, for example, modifying component usage and licenses.

However, archived project versions are treated differently than all other project version phases.

- Archived project versions are excluded from project [risk calculations](#).

Project versions with any other phase are included in project risk calculations.

- If you enabled [persistent edits](#):

- Your edits made to a project version *are not* propagated to archived project versions.
- Your edits made to an archived project version *are* propagated to all other non-archived project versions.

Those edits *are not* applied to any other archived project version.

- Updates from Black Duck KnowledgeBase regarding security vulnerabilities *are* applied to archived project versions.

Other updates from Black Duck KB, such as updates to license information, *are not* applied to archived project versions.

- New policy rules and updated expressions are not evaluated in archived project versions.
- Disabled and deleted policy rules violations will be removed from archived project versions.

6. Viewing a project version's BOM

Once you have mapped a component scan or a Protex BOM to a project version, the results automatically create the [project version's BOM](#).

To view a project version's BOM:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version that you want to view.
The **Components** tab displays the BOM. The example below is what appears for a user with the BOM Manager [role](#) using the List view:

The screenshot shows the Black Duck Project Groups interface for 'Sample Project' 1.0.0. The top navigation bar includes 'Project', 'Phase: In Development', 'Scans: Up to Date', 'Status: Up to Date', 'Last Updated: 10:21 AM', and tabs for 'Components', 'Security', 'Source', 'Reports', 'Details', 'Legal', and 'Settings'. Below the navigation are three risk graphs: Security Risk (4 Critical, 3 High, 4 Medium, 0 Low, 17 None), License Risk (6 High, 2 Medium, 0 Low, 20 None), and Operational Risk (24 High, 3 Medium, 0 Low, 1 None). A 'Snippets' section shows 78 Unconfirmed and 1 Unmatched. Below the graphs are buttons for 'Add', 'Bulk Actions', 'Compare to...', 'Ignore', 'Not Ignored', 'Snippet Match Status', 'Confirmed', 'Match Ignore', 'Not Ignored', and 'Filter'. A 'Print' button is also present. The main table lists components with columns for Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk.

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

Tip: Refer to Black Duck online help system for information on how users with the BOM Manager and Project Manager role can modify the project version's BOM to reflect how you are actually using the OSS components in the project.

Understanding the information in a project version's BOM

On a project version page (from the Dashboard, select *Project Name* > *Project Version*), the **Components** tab displays the BOM. The page displays a header, risk graphs and a data table.

6. Viewing a project version's BOM • Reviewing the contents of a BOM

The screenshot displays the Black Duck Project Groups interface for 'Sample Project' 1.0.0. It features three risk charts: Security Risk (4 Critical, 3 High, 4 Medium, 0 Low, 17 None), License Risk (6 High, 2 Medium, 0 Low, 20 None), and Operational Risk (24 High, 3 Medium, 0 Low, 1 None). Below these are filters for Snippets (78 Unconfirmed, 1 Unmatched) and a table of components. The table has columns for Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk. The components listed are AOP Alliance (java/JEE AOP standard) 1.0, Apache Commons DBCP 1.2.2, Apache Commons FileUpload 1.3.3, and Apache Commons IO 2.2.

The above example is the List view of the BOM. You can also view a Tree version of the BOM.



Reviewing the contents of a BOM

Any user that can edit a BOM can review the contents and indicate that a component version or subproject is correctly included in that BOM.

 **Note:** Project members with no roles assigned to them cannot flag BOM contents as reviewed.

In the component list view of the BOM, next to each component or subproject name is an icon which indicates whether this item has been reviewed:

The screenshot shows a component list view with a header 'Component ^'. Below it are two entries: 'Push Notifications Android' 0.9.0 and '51Degrees core 3.1.3.2'. Each entry has a toggle icon (a circle with a checkmark) to its left, indicating its review status.

-  - Not reviewed
-  - Reviewed

Use this icon to flag component versions and subprojects as reviewed: the icon is a toggle – select it to change its status.

To review multiple component versions or subprojects:

Use the bulk review feature to indicate that all component versions and/or subprojects that appear on a *single* page are reviewed or unreviewed.

1. Optionally, filter the BOM so that the component versions and subprojects you wish to review/unreview appear on the page.

2. Select **Select all**.

All components and/or subprojects on this page are selected.


You can select individual rows so that they are not included.

3. From the **Bulk Actions** menu, select one of the following:

- **Mark as reviewed** to indicate the component/subproject has been reviewed.
- **Mark as unreviewed** to indicate the component/subproject has not been reviewed.

4. Click **Review** or **Unreview** in the confirmation dialog box.

5. Refresh the page to view your changes. It may take some time for the review status to appear.

 **Tip:** To review or unreview multiple pages, repeat steps 2-5 for each additional page in the BOM.

 **Note:**

- Hover over the Reviewed icon (✓) to view the username of the user who reviewed this component version/subproject and the date and time when it was reviewed.
- If you selected to [apply edits to all versions](#) of a project, the review status will persist if you rescan the same code into a new project version.
- Use the filters on the BOM page to view the BOM page by review status.
- The `components_date_time.csv` and the `bom_component_custom_fields_date_time.csv` files in the [Project Version report](#) include the review status, the username of reviewers, and the review date.
- Changing the review status does not cause the [Information icon](#) (i) to appear.
- The review status cannot be changed in the Tree View of the BOM.

Managing comments in a BOM

Comments apply to a specific component version or subproject in a BOM. For example, you can use comments to explain why a component version was ignored or why a policy violation was overridden.


Note:

- Comments are applied to a component version or [subproject](#):
 - If the component version or subproject is deleted in a BOM, the comment is deleted. If the component version or subproject is then added back to the BOM, the comment(s) will reappear.
 - If the version of a component or subproject is changed in a BOM, the comment no longer appears.
- Comments do not [persist to all versions of a project](#).
- Comments by users who become inactive still appear in the BOM.
- A component version or subproject can have multiple comments.
- The search feature is not available for comments.
- Comments cannot be added to the Tree View of the BOM.

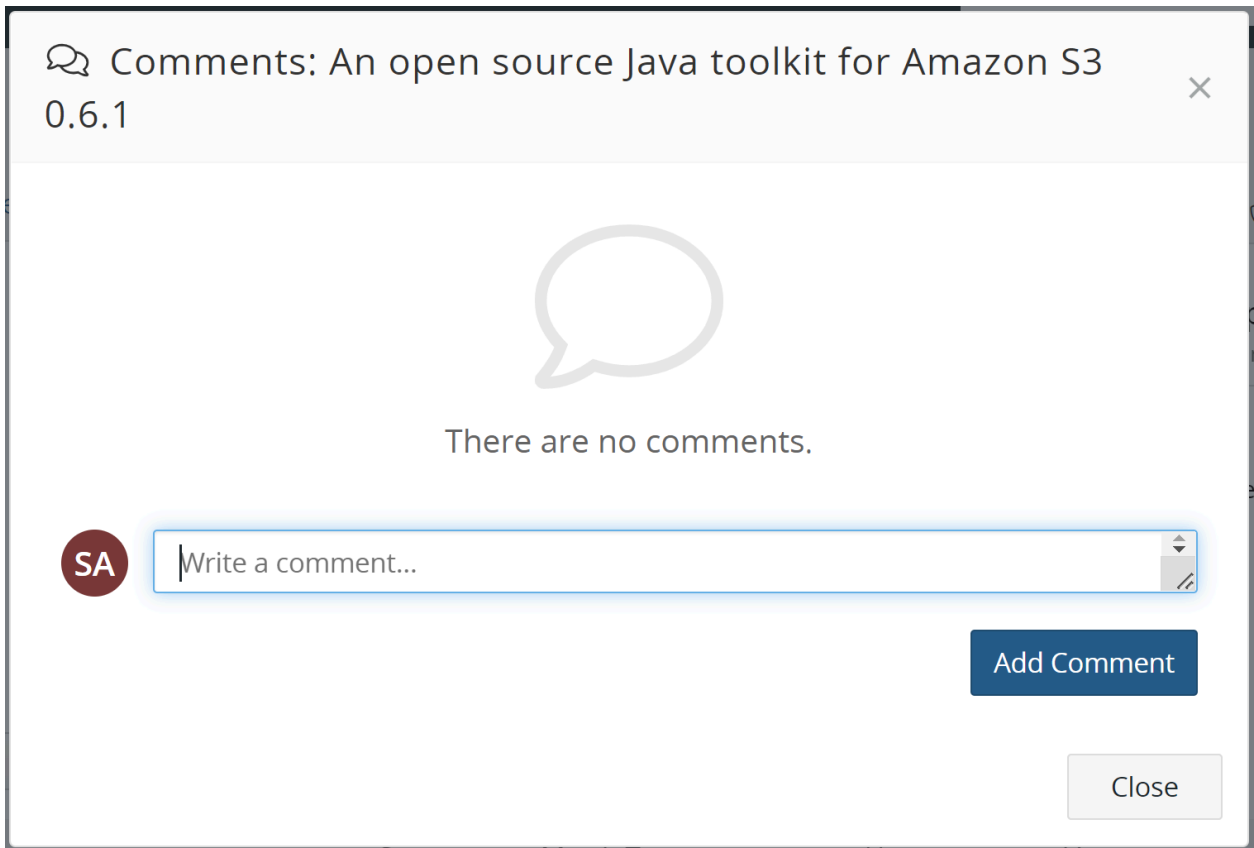
Adding a comment in the Components view

Adding comments to components makes your feedback more clear, giving everyone the ability to discuss any actions to be taken on a particular component. You can add a comment to a single comment, or you can comment on a number components by using the Bulk Actions button.

To comment on a single component:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Click  in the row where you want to add a comment and select **Comments**.

The *Component/Subproject Name Version* Comment dialog box appears.

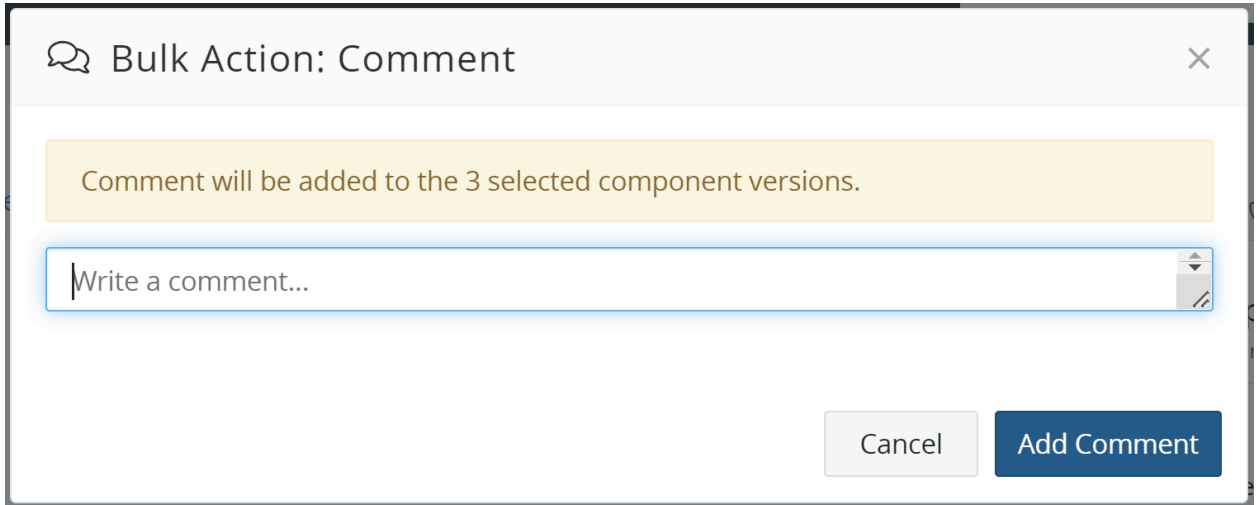


3. Enter the comment and click **Add Comment**.

To comment on multiple components simultaneously:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Check the box next to any number of components.
3. Click the **Bulk Actions** button.
4. Select **Comment**.

The **Bulk Action: Comment** dialog box appears.




5. Enter the comment and click **Add Comment**.

After either of the actions above, a comment icon () appears in the component version or subproject row indicating a comment was added. The number shown in the icon indicates the number of comments for this component version or subproject.




Component ^	Source	Match Type	Usage	License	Security Risk	Operational Risk
Apache Struts 2.0.4		Manually Added	Dynamically Linked	Apache-2.0	15 16 1	High

Viewing a comment

Click  in the row where you want to view a comment.




Editing a comment

Only the original writer can edit their comment.



1. Click  or  (if there are already comments) in the row where you want to edit a comment and select **Comment**.
2. Click  next to the comment you want to edit and select **Edit**.
3. Edit the comment, click **Update**, and then select **Close**.

Deleting a comment

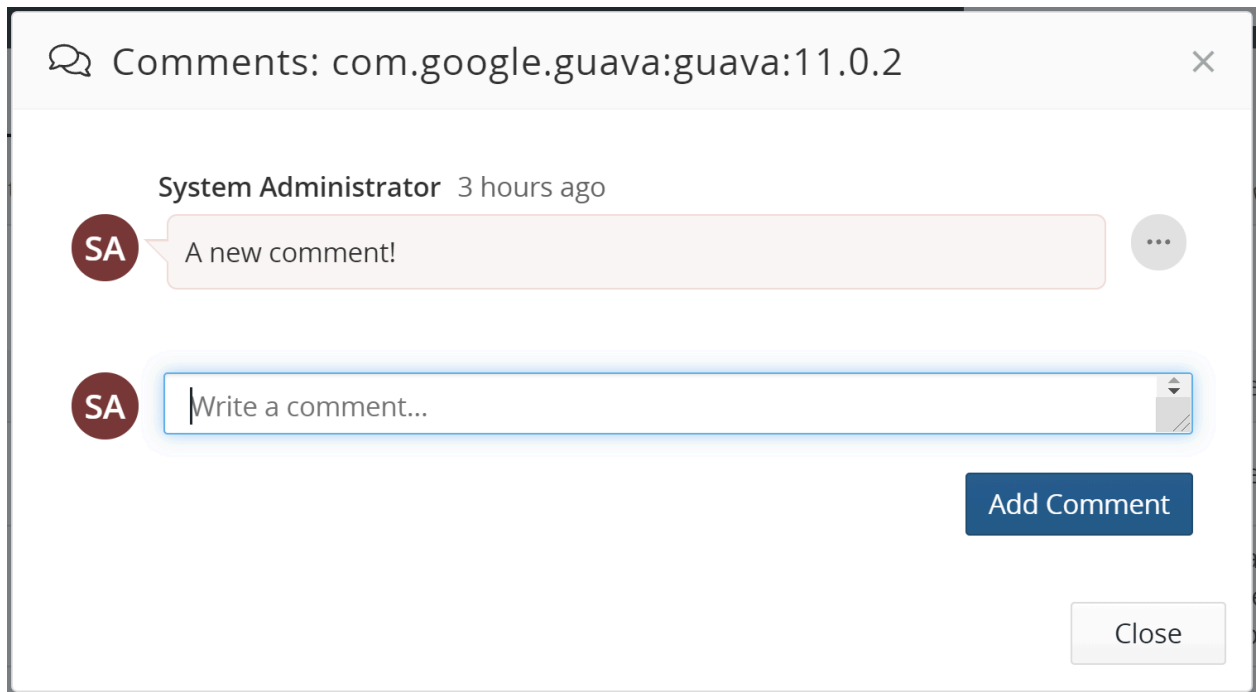
Only the original writer of the comment or Project Administrator can delete a comment.

1. Click  or  (if there are already comments) in the row where you want to edit a comment and select **Comment**.
2. Click  next to the comment you want to delete and select **Delete**.


Adding a comment in the Source view

1. [Display the project version BOM](#). Ensure you are in the Source view.
2. Click the desired item in the file tree.
3. Click  in the row where you want to add a comment and select **Comments**, or click  if there are already comments present.

The Comments dialog box appears.



4. Enter the comment and click **Add Comment**.

A comment icon () appears in the entry row indicating a comment was added. The number shown in the icon indicates the number of comments for this component version or subproject.



<input type="checkbox"/>	Name	Component	Match Type	License	Usage	Discovery Types
	com.google.guava:guava:11.0.2	Guava: Google Core Libraries for Java 11.0.2	Direct Dependency	Apache-2.0	Dynamically Linked	


Viewing a comment

Click  in the row where you want to view a comment and select **Comments**, or click  if there are already comments present.

Editing a comment




Only the original writer can edit their comment.

1. Click  or  (if there are already comments) in the row where you want to edit a comment and select **Comments**.

2. Click  next to the comment you want to edit and select **Edit**.
3. Edit the comment, click **Save**, and then select **Close**.

Deleting a comment

Only the original writer of the comment or Project Administrator can delete a comment.

1. Click  or  (if there are already comments) in the row where you want to edit a comment and select **Comments**.
2. Click  next to the comment you want to delete and select **Delete**.

Managing files associated with BOM components

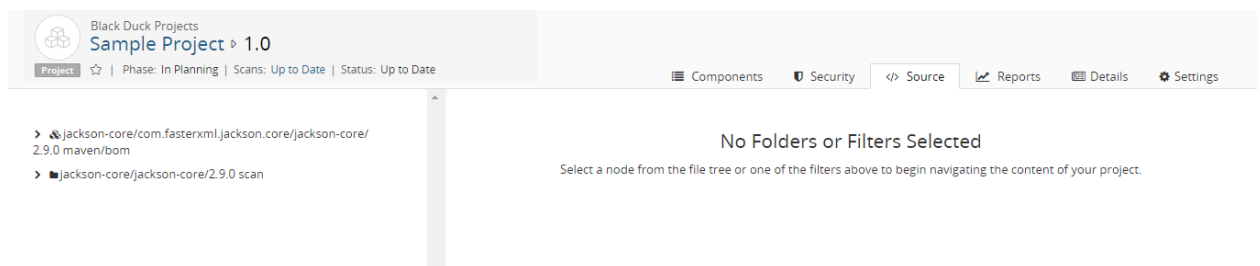
Use the **Source** tab to manage the files associated with BOM components. Common cases include:

- Analyzing and identifying unmatched files. Unmatched files can be related to a component, a proprietary component, or a third-party component. Review these files to determine if they must be matched to a component version or if they can be excluded.
- Validating files that were matched to a component. Review these files to determine if they were matched to the correct component version or if they were incorrectly matched. Incorrectly matched files can be associated with the correct component version or excluded.
- [Reviewing snippet matches](#).
- [Reviewing detected embedded licenses](#).

Accessing the Source tab

You can access the **Source** tab to view all files in a project or automatically filtered to view specific matches.

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to open the **Components** tab and view the BOM.
3. Do one of the following:
 - Select the **Source** tab to view all files in this BOM.



Select an item in the left pane to see information in the table.

- Select a value in the **Match Count** column to view the **Source** tab filtered to that component.

6. Viewing a project version's BOM • Managing files associated with BOM components

Black Duck Projects
Sample Project > 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Legal Settings

Component Apache Struts - 2.3.7

Name	Component	Match Type	License	Usage	Discovery Types
org.apache.struts.xwork:xwork-core:2.3.7	Apache Struts 2.3.7	Transitive Dependency	Apache-2.0		
org.apache.struts.struts2-core:2.3.7	Apache Struts 2.3.7	Direct Dependency	Apache-2.0		

Displaying 1-2 of 2

About the Source tab

The **Source** tab consists of:

- A left pane which shows the tree structure of the files. Use this pane to navigate and select the information shown in the table.

Select an item in the left pane to display the information in the table for the selected item.

Selecting to view an archive:

Black Duck Projects
Sample Project > 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Legal Settings

com.sun.grizzly:grizzly-framework:1.9.14

Match Type	Component	License	Usage
Direct Dependency	grizzly-framework 1.9.14	CDDL-1.0	Dynamically Linked

Edit Reset File Adjustments

Name	Component	Match Type	License	Usage	Discovery Types
com.sun.grizzly:grizzly-utils:1.9.14	grizzly-framework 1.9.14	Transitive Dependency	CDDL-1.0		

Displaying 1-1 of 1

Selecting to view a folder:

Black Duck Projects
Sample Project > 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Legal Settings

/# > bds_jenkins_test


Files Discoveries All Subfolders

Edit Reset File Adjustments

Name	Component	Match Type	License	Usage	Discovery Types
.travis.yml					
AcmeAuthor.java					
AcmeConfig.java					
AuthorsConfig.java					
README.md					
UserService.java					
WEB-INF					
acme					
azure-pipelines.yml					
bootstrap.js	Bootstrap (Twitter) 3.3.2	Exact File	MIT	Dynamically Linked	











The table displays the files/directories directly under the selected item in the left pane.

Information about the selected item, such as the component name and version, path, and scan size appear above the table.

Click  and select **Copy path** to copy the path to your clipboard.


- A right pane which displays the following information:
 - A header banner containing relevant information for the selected component such as the file name or namespace for the component. In the case of imported SBOM files, the banner will contain information such as the SBOM type (SPDX or CycloneDX), when the SBOM was imported, who supplied the SBOM, and the version of the tool used to create the SBOM.
 - A table which provides the following information on the item selected in the pane:
 - **Name.**
Select the name to filter the information shown in the table. The item you selected is also highlighted in the tree shown in the left pane.
 - **Component.** Name and version of the OSS component in use in this version of your project.
Select the component name or version to open [Black Duck KB component version page](#) which displays more information of the component version, such as a list of the projects and project versions in which this version of the component is used.
 - **Match type.** Indicates how the match between the component in use in this version of your project and a specific version of a project in Black Duck KB was made.
 - **License.** Declared license of the component in use in this version of your project.
 - **Usage.** Indicates how this file is intended to be included in the project when this version is released. Click [here](#) for more information on usage.
 - **Discovery Types.** Indicates the type of discovery. Possible values of License and License Reference are for [embedded licenses detected during the scan](#).
- Filters located above the table, to filter the information shown on the tab.
- Check box located above the table, to view subfolder information. Select **All Subfolders** to include information on all subfolders and files.
- **Files/Discoveries** tab to view files or discoveries. Select **Discoveries** to view [embedded license information](#) detected in the scan.

The tab uses the following icons:

-  Package manager scan/archive
-  or  Signature scan/directory
-  Empty dependency tree - which contains no OSS matches
-  or  File
-    Snippet information. Click [here](#) for more information.
-  Source file. Used when [reviewing snippet matches](#) and [detected embedded licenses](#).


Modifying matches


To modify a match:

1. Open the **Source** tab as described above.
2. Select one or more items in the table and click  located above the table.
3. In the Edit Component (if you selected one item) or Bulk edit (if you selected multiple items) dialog box, modify the component, version, origin ID, and/or usage.
Click [here](#) for more information about modifying snippet matches.
4. Click **Update**.

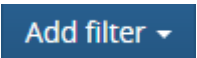

Identifying unmatched components


An unmatched component means that it was not possible to match the stated external identifier to a component in the Black Duck KnowledgeBase. The external identifiers in the KnowledgeBase are taken from the public Forges, like Maven Central, etc. Unmatched components found in the BOM Import log can also be seen in the Source view of a scan. You can triage these components and either identify them to a [KnowledgeBase component](#) or [create a custom component](#) and associate it with the components.

1. Open the **Source** tab as described above.
2. Click the root folder of the scan in the left-hand panel of the Source view.
3. Select one or more entries and click . The Edit Component dialog box (if you selected one item) or Bulk Edit dialog box (if you selected multiple items) appears.
 - If the component already exists, enter the name in the **Component** field and specify a version.
 - If the component does not exist, create a custom component first before completing this step.
4. Click **Update**.

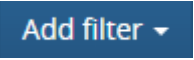

A  appears in the BOM in the row of the component you selected to indicate that a manual adjustment was made to this file. The match type changes to **Manually Identified Package**.


Identifying unmatched files

1. Open the **Source** tab as described above.
2. Click  and select **Match type > Unmatched** and click **OK**.
3. Select one or more entries and click . The Edit Component dialog box (if you selected one item) or Bulk Edit dialog box (if you selected multiple items) appears.
 - If the file is part of a component that is in use, enter the name in the **Component** field and specify a version.
 - If the file should not be included in the project, select **Dev. Tool / Excluded** from the **Usage** list.
4. Click **Update**.

A  appears in the BOM in the row of the component you selected to indicate that a manual adjustment was made to this file. The match type changes to **Manually Identified**.

Validating matched files

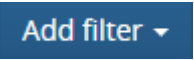
1. Open the **Source** tab as described above.
2. Click  and select **Match Type > Type of match(es)** and click **OK**.
3. Select one or more entries and click . The Edit Component dialog box (if you selected one item) or Bulk Edit dialog box (if you selected multiple items) appears.
 - If the file was incorrectly matched to a component during the scan, enter the new name in the **Component** field and specify a version in the **Version** field.
 - If the file was incorrectly matched to an origin or origin ID, specify a different value using the **Origin** and **Origin ID** fields.
 - If the file should not be included in the project, select **Dev. Tool / Excluded** from the **Usage** list.
4. Click **Update**.

A  appears in the BOM for this component to indicate that a manual adjustment was made to this file.

Resetting files and components

You can revert manually adjusted files and components to their original match type.

This option is not available for unmatched files and is not enabled if the file cannot be reset.

1. Open the **Source** tab as described above.
2. Click  and select **Adjusted**.
3. Select one or more files or components and click **Reset Adjustments**.

If you select multiple files or components, only those files that can be reverted are reset.
4. Click **Save**.

Deleting files from a BOM

You cannot delete files that were automatically added to a component. You can [ignore a component](#) in the BOM that contains the file so that it is not included when calculating the security, license, and operational risks for this version of your project.

To remove an automatically-added scanned component from a project version's BOM, you must remove it from your source code or Docker image and then rescan that code or Docker image. This will automatically update the project version's BOM to reflect only those component's that were automatically discovered in the mapped scans and manually added to the BOM.

To remove an automatically-added component from a Protex BOM, you must remove it in Protex and then use the Protex BOM tool to re-import the Protex BOM. This will automatically update the project version's BOM to reflect the changes in the Protex BOM.


Comparing BOMs

Use the Project Comparison window to view the differences between two project version BOMs. You can view the differences between two versions of the same project or between two versions of different projects.

 **Note:** You can only compare projects which you have permission to view.

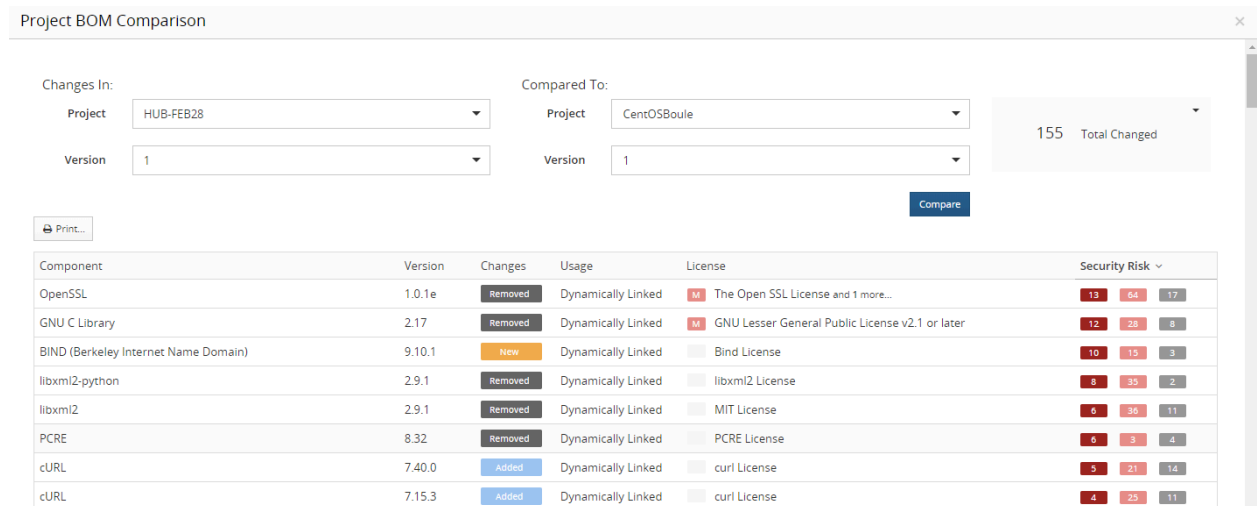
6. Viewing a project version's BOM • Comparing BOMs

To view a comparison of two project version BOMs:

 **Note:** While you can compare any two versions of a BOM for the same or different projects, this page uses the terms "current" and "compared to" to differentiate the versions.

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to open the **Components** tab and view the BOM. This is the "current" version of the BOM.
3. Select **Compare to** and then select a different version of this BOM or select **Other project** to select a different project and version.

The Project BOM Comparison window appears.



Component	Version	Changes	Usage	License	Security Risk
OpenSSL	1.0.1e	Removed	Dynamically Linked	The Open SSL License and 1 more...	13 64 17
GNU C Library	2.17	Removed	Dynamically Linked	GNU Lesser General Public License v2.1 or later	12 28 8
BIND (Berkeley Internet Name Domain)	9.10.1	New	Dynamically Linked	Bind License	10 15 3
libxml2-python	2.9.1	Removed	Dynamically Linked	libxml2 License	8 35 2
libxml2	2.9.1	Removed	Dynamically Linked	MIT License	6 36 11
PCRE	8.32	Removed	Dynamically Linked	PCRE License	6 3 4
cURL	7.40.0	Added	Dynamically Linked	curl License	5 21 14
cURL	7.15.3	Added	Dynamically Linked	curl License	4 25 11

At the top of the page are the projects and versions being compared. The "current" project and version of the BOM appears in the **Changes In** column.

- If you selected to compare a different version of the same project, that project name and version appears in the **Compared To** column and the table shows the comparison of the two BOMs.
- If you selected **Other project**, the table is empty; use the **Project** and **Version** fields to select the BOM to be compared and click **Compare**.

This is the "compared to" version of the BOM.

This window shows the adjustments to components or subprojects that occurred in the BOM and the associated change to the security risk. Adjustments to components consist of:

- New components/subprojects. Components or subprojects in the "current" version of the BOM that were not in the "compared to" version of the BOM.
- Updated components/subprojects. While the components or subprojects were in the "compared to" version of the BOM, one or more of the following changed:
 - Component/Subproject version
 - Usage
 - License
- Removed components/subprojects. The components or subprojects that were in the "compared to" version of the BOM that are not in the "current" version of the BOM.

Note the following:

- There is only a top-level comparison of subprojects: the components in subprojects are not compared.
- If you selected to [maintain component adjustments to all versions of a project](#), the Project Comparison window may show little to no changes between versions of the same project.
- Only confirmed snippets are compared.

To view and work with the information that is important to you:

- Filter the information shown by the type of adjustment.

Select the **# New Components**, **# Removed Components**, or **# Updated Components** filters located at the top right section of the window to filter the information shown in the table.

Select **# Total Changed** to view all information. This is the default view.

- Print the information shown in the window.

1.

Click . A print dialog box appears.

2. Configure the print settings and print the comparison.

Column	Description
Component	Component or subproject name.
Version	Component or subproject version.
Changes	<p>Possible values are:</p> <ul style="list-style-type: none"> • Added. The component or subproject is in the "current" and "compared to" version of the BOM, however, it had a different version in the "compared to" version of the BOM. The version shown here is the version in the "current" version of the BOM. • Modified. The usage or license for this component/subproject version has changed. • New. The component or subproject is new – it was not in the "compared to" version of the BOM. • Removed. The component/subproject was in the "compared to" version of the BOM, however, it is not in the "current" version of the BOM. • Replaced. The component/subproject is in the "current" and "compared to" version of the BOM, however, there is a different version in the "current" version of the BOM. The version shown here is the version in the "compared to" version of the BOM.

For modifications to ignored components:

- Components ignored in both versions are not compared.
- Components ignored in the "compared to" version but not ignored in the "current" version have a value of **Added**.
- Components ignored in the "current" version but not ignored in the "compared to" version have a value of **Removed**.

Note that for a modification to the version:

- The component/subproject and original version are shown with **Replaced** as the value in the **Changes** column.
- The component/subproject and new version are shown with **Added** as the value in the **Changes** column.

In the following example, the component Lucene had version 1.4.3 in the "compared to" version of the BOM and version 4.5 in the "current" version of the BOM:

Column	Description								
	<table border="1"> <tr> <td>Lucene</td> <td>4.5</td> <td>Added</td> <td>Dynamically Linked</td> </tr> <tr> <td>Lucene</td> <td>1.4.3</td> <td>Replaced</td> <td>Dynamically Linked</td> </tr> </table>	Lucene	4.5	Added	Dynamically Linked	Lucene	1.4.3	Replaced	Dynamically Linked
Lucene	4.5	Added	Dynamically Linked						
Lucene	1.4.3	Replaced	Dynamically Linked						
Usage	<p>Usage of the component or subproject version in the "current" version of the BOM. Strikeout usage text shows the usage for this component version from the "compared to" version of the BOM.</p>								
License	<p>Declared license of the component or subproject in use in the "current" version of the project. Strikeout license text shows the license for this component version from the "compared to" version of the BOM.</p>								
Security Risk	<p>Number of high risk (100% red), medium risk (50% red), and low risk (100% gray) vulnerabilities associated with this version of the component or with the subproject. The value in the Security Risk column indicates an increase or decrease in security risk depending on the value in the Changes column. If the value in the Changes column is:</p> <ul style="list-style-type: none"> • Removed or Replaced. The value indicates a decrease in security risk from the "compared to" version of the BOM. • New, Modified, or Added. The value indicates an increase in security risk from the "compared to" version of the BOM. 								

Printing a BOM

You can print a BOM.

The printout displays the BOM similar to what is shown in the UI: security, license, and operational risk graphs appear at the top of the page; component and subproject information is listed in a table.

You can filter the BOM prior to printing so that it only includes the data you wish to view. Any filters applied to the BOM are listed above the table.

To print a BOM:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version that you want to view. The **Components** tab displays the BOM.

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components

Critical	4
High	3
Medium	4
Low	0
None	17

License Risk
Number of Components

High	6
Medium	2
Low	0
None	20

Operational Risk
Number of Components


High	24
Medium	3
Low	0
None	1

Snippets
78 Unconfirmed

Unmatched Components
1 Unmatched

Print

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

- Optionally, filter the BOM so that the printout only shows the information you want to see.
- Click . A print dialog box appears.
- Configure the print settings and print the BOM.

Viewing issues in a project

Black Duck provides information on the issues associated with a project version as monitored by an issue tracking system. Currently, this feature is supported using Black Duck Alert 6.2.0 and later.

Black Duck displays an **Issues** link in the project version header for a project version if an issue tracking system was configured to the Black Duck project version using Black Duck Alert. Once Black Duck Alert creates issues for this project version, the link appears. No additional configuration is needed.

Black Duck Projects
gk_alert_test ▶ issue_tracker_test

Project ☆ | Owner: Gavin Gavin | Phase: In Planning | Issues: 4 | Scans: Up to Date | Status: Up to Date

Security Risk
Number of Components

Critical	0
High	3
Medium	1
Low	0
None	0

License Risk
Number of Components

High	1
Medium	0
Low	0
None	3

Note that the **Issues** link does not appear if there are no issues or all issues have been deleted.

Users with the [Global Project Administrator](#), [Global Project Manager](#) roles and all project members (users assigned to the project) can select the issues value to display the Issue Management table.

6. Viewing a project version's BOM • Viewing component versions with encryption

Issue Management						
Component	ID	Summary	Assignee	Status	Updated	
Apache Struts 1.3.5	ALERT-3	Alert - Provider: Black Duck, Project: gk_alert_test, Project Version: issue_tracker_test, Component: Apache Struts, Component Version: 1.3.5, Vulnerability	Alert User	Created by Alert	Never	
Apache Struts 1.3.8	ALERT-4	Alert - Provider: Black Duck, Project: gk_alert_test, Project Version: issue_tracker_test, Component: Apache Struts, Component Version: 1.3.8, Vulnerability	Alert User	Created by Alert	Jul 16, 2020	
Apache Struts 1.3.5	GALERT-689	Alert - Provider: Black Duck, Project: gk_alert_test, Project Version: issue_tracker_test, Component: Apache Struts, Component Version: 1.3.5, Vulnerability	Alert User	Created by Alert	Never	
Apache Struts 1.3.8	GALERT-690	Alert - Provider: Black Duck, Project: gk_alert_test, Project Version: issue_tracker_test, Component: Apache Struts, Component Version: 1.3.8, Vulnerability	Alert User	Created by Alert	Jul 16, 2020	

Displaying 1-4 of 4

Close

This table lists the issues created in the external issue tracking systems. You can then use this table to see the status of the issue in your workflow.

The table provides the following information for each issue:

Column	Description
Component	Component name and version affected by this ticket.
ID	Issue identifier.
Summary	Summary of the external issue.
Assignee	User assigned to this ticket.
Status	Status of the ticket.
Updated	Time when this ticket was last updated.

Note that this table does not display all changes from the external issue tracker system. Changes to the issue, such as manual changes to the description, will not be reflected in the Black Duck issue table other than those changes made automatically to the issue by Black Duck Alert.

Viewing component versions with encryption

Open source software can use or implement cryptographic algorithms which can impact your organization from security and compliance perspective.

On the compliance side, whenever you send software out of the country – for example, on a computer, as source code, or compiled into an application that is for sale – depending upon where you live, you may be required to adhere to certain governmental regulations regarding the export of cryptography. This is especially true of strong cryptographic algorithms which may require licenses to export, however the regulations have eased in recent years.

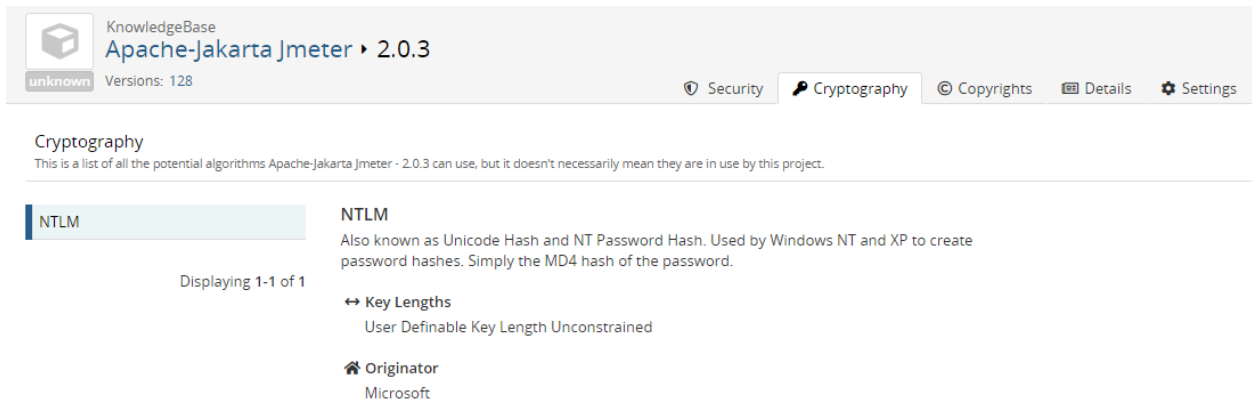
On the security side, companies may be interested in understanding if open source is using weak cryptography or obsolete hashing mechanisms. Using a cracked (or insecure) cryptographic algorithm can add unnecessary risk to your organization, especially if well-known techniques exist to break the algorithm. Understanding algorithms in use can help companies comply with security standards.

Black Duck helps you identify the component versions that have encryption algorithms.

- A cryptography filter in the component version [BOM page](#) identifies those component versions with encryption.

- A cryptography icon (🔒) appears in the BOM page for any component version with encryption algorithms.

Select the component version to open the *Component Version* page and then select the **Cryptography** tab:



KnowledgeBase
Apache-Jakarta Jmeter ▸ 2.0.3
unknown Versions: 128

Security Cryptography Copyrights Details Settings

Cryptography

This is a list of all the potential algorithms Apache-Jakarta Jmeter - 2.0.3 can use, but it doesn't necessarily mean they are in use by this project.

NTLM
<p>NTLM</p> <p>Also known as Unicode Hash and NT Password Hash. Used by Windows NT and XP to create password hashes. Simply the MD4 hash of the password.</p> <p>↔ Key Lengths</p> <p>User Definable Key Length Unconstrained</p> <p>🏠 Originator</p> <p>Microsoft</p>


Displaying 1-1 of 1

- The table lists the encryption algorithms found in this component version.
- The warning symbol (⚠️) indicates that this algorithm has a known weakness.
- Select an algorithm from the table to view more information, such as a description, key lengths, originator, licensing, and patent information.

Possible values for key lengths, with key length values where applicable, are:

- Single Fixed Key Length
- Multiple Fixed Key Lengths
- User Definable Key Length within a Closed Range
- User Definable Key Length Unconstrained
- No Encryption or No Key Used

Note that the **Cryptography** tab does not appear if a component version does not have encryption algorithms.

 **Note:** While components added manually to existing BOMs will display cryptography information, legacy BOMs may require a rescan for cryptography data to appear.

For more information on federal regulations, visit the Bureau of Industry and Security's (BIS) website: <https://www.bis.doc.gov>

About Linux distributions in Black Duck

Linux distributions combine the Linux kernel with other software, mostly open source software, to create a complete package. Black Duck reports on the vulnerabilities associated with the OSS components in these packages. However, this may lead to false positives as Linux distribution packages can be patched and these patches are not tracked by NVD.

Black Duck displays these vulnerabilities with a [remediation status](#) of "Needs Review", "Patched", or "New" (if Black Duck has verified that the vulnerability affects that version of the OSS component).

If you determine that the version of your package has been patched, you can change the remediation status to "Patched." A remediation status of "Patched" removes the CVE from the security risk calculation.

Viewing Linux distributions in Black Duck

Black Duck shows the origin and origin ID:

- In the **Component** column when viewing details for a component on the Project Version page/**Components** tab
- In the list of components shown in the Project Version page/**Security** tab
- In the **Component** column when viewing details in the Project Version page/**Source** tab.

You can [add or edit the origin and origin ID](#) shown for a component.

7. Editing a BOM


Users with the appropriate [role](#) can:

- [Apply edits to all versions of a project.](#)
- [Manually add a component to a BOM.](#)
- [Exclude a component from a BOM.](#)
- [Delete a component from a BOM.](#)
- [Remove components from a BOM.](#)
- [Adjust the component and/or component version in a BOM.](#)
- [Edit an origin or origin ID.](#)
- [Ignore a component in a BOM.](#)
- [Select a different license for a component in a BOM.](#)
- [Edit license text in the BOM.](#)
- [Manage subprojects.](#)
- [Triaging snippets.](#)


Applying edits to all versions of a project


You can select whether edits to a component apply to a specific version of a project or if edits are persistent – they apply to all versions of a project. If you select to make edits persistent then edits apply to all existing versions of a project, excluding [archived versions of projects](#) and manually added components, and will also be carried forward as additional scans are completed at the same code or Docker image.







For example, if you edit a matched component to a different component, then all other versions of the project that have that same matched component will have the match adjusted and all versions going forward will also have this match adjusted.

 **Note:** There are instances when edits may not propagate to all versions. See [Persistent edit examples](#) below.

Persistent edits are enabled by default when you [create a project](#).

 **CAUTION:** Please exercise caution when changing this option in order to avoid the [Persistent edit examples](#) below. Projects created prior to release 3.1.0 will have this feature disabled by default. See the examples described below as those results will apply if you enable this feature to those projects.

When you edit a component (using the BOM or Files page), a  appears in the table row to indicate that a manual adjustment was made to this component:

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
 Apache Commons Collections ?	 1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	 1  3	 

Displaying 1-1 of 1

 **Note:** A  appears on the BOM page for any edits that you make to a BOM.

There is also the option of [cloning project versions](#) which enables you to baseline a project version.

Persistent edit examples

Edits may appear to work differently than expected depending on the status of persistent edits and when the edits are made.

In the examples below, a project has several versions, none of which are archived.

Example	Final Result
<ol style="list-style-type: none"> 1. Persistent edits are enabled. 2. An edit is made to an item in a component in one version of the project. For example, the license for Component A is changed in Version 1 of the project. The edit is propagated to all versions of the project. 3. Persistent edits are then disabled. 4. An edit is made to the <i>same item</i> in Component A in a version of the project. For example, the license for Component A is changed in Version 1 (or Version 2) of the project. 	<p>Although persistent edits are disabled, the edit is propagated to <i>all versions</i> of the project as the original edit was made when persistent edits were enabled.</p>
<ol style="list-style-type: none"> 1. Persistent edits are disabled. 2. An edit is made to an item in a component in one version of the project. For example, the license for Component A is changed in Version 1 of the project. The edit appears in only Version 1 of the project. 3. Persistent edits are then enabled. 4. An edit is made to the same item in the same component in the same version. For example, the license for Component A is changed again in Version 1 of the project. 	<p>The edit is applied to only that version of the project (Version 1 in our example). The edit does not propagate to other versions of the project as the original edit was made when persistent edits were disabled.</p>
<ol style="list-style-type: none"> 1. Persistent edits are disabled. 2. An adjustment is made to an item in a component in one version of the project. For example, the license for Component A is changed in Version 1 of the project. The edit appears in only Version 1 of the project. 3. Persistent edits are then enabled. 4. An adjustment is made to the same item in a component in a different version of the project. For example, the license for Component A is changed in Version 2 of the project. 	<p>The edit is propagated to all versions <i>except</i> Version 1.</p>

Enabling or disabling persistent edits for a project

1. Log in to Black Duck.

2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the **Settings** tab.

Black Duck Project Groups
Sample Project

Project: ★ Watching Project | Versions: 1 | Owner: System Administra... Overview Settings

Project Details

SCM Repository

Users

Groups

Custom Fields

SBOM Fields

Activity

Settings

Project Name

Sample Project

Description

Owner

System Administrator (no-reply@synopsys.com) X

Tier

Select...

Component Adjustments
Always maintain component adjustments to all versions of this project. Archived project versions and manually added components are excluded.

Snippet Adjustments
Apply the following snippet adjustments upon snippet rescans. Apply IDs from partial snippet matches to new exact file matches.

Cloning
Select the attributes you'd like to clone for any new versions of this project.

Custom Fields

Component Edits

Deep License Data

Remediation Details

License Fulfillment Status

Version Settings

Custom Scan Signature
Custom Scan Signature can identify third-party and proprietary software used in your code. There may be performance issues seen when using this feature.

Custom Scan Signature Depth
Depth, as measured in the number of levels in the directory structure, from root, to perform custom signature scanning for this project. The initial value is the default value defined by the System Administrator.

5

Retain Unmatched File Data
If enabled, unmatched file data for scans will always be retained. When disabled (default), unmatched file data will be purged.

System Default (Don't Retain Data)

Purge ONLY Archived Project Version Unmatched File Data

Purge ALL Unmatched File Data

Apply Deep License Data to Bill of Materials
Enabling this checkbox will apply deep license data to your components and allow visibility to embedded licenses which may exist in your components beyond declared licenses. Please note, this can affect the license risk and policy violation for components. It can also impact the Bill of Materials calculation time depending upon the number of components and amount of deep licenses.

Apply Deep License Data to Snippet Component Matches
If enabled, component snippet matches are included in the deep license data calculation.

License Conflicts
Enabling this checkbox will apply license conflicts data to your components

Reset Save

Application ID

Application ID
A field that can be used to store an external mapping id for the project to an external system, like an asset management system or application catalog.

Reset Save

Clone Project

Clone selected versions of this project as well as existing project settings, users, groups, custom fields, component edits and application ID.

Clone Project

Delete Project

Once you delete a project, you cannot restore it and you lose all information and versions related to the project. Scans will be unmapped from all versions and not deleted.

Delete Project

4. Do one of the following in the **Component Adjustments** area of the **Project Details** section:


7. Editing a BOM • Manually adding a component to a BOM

- Select **Always maintain component adjustments to all versions of this project** to enable persistent edits.
 - Clear **Always maintain component adjustments to all versions of this project** to disable persistent edits.
5. Click **Save**.

Manually adding a component to a BOM

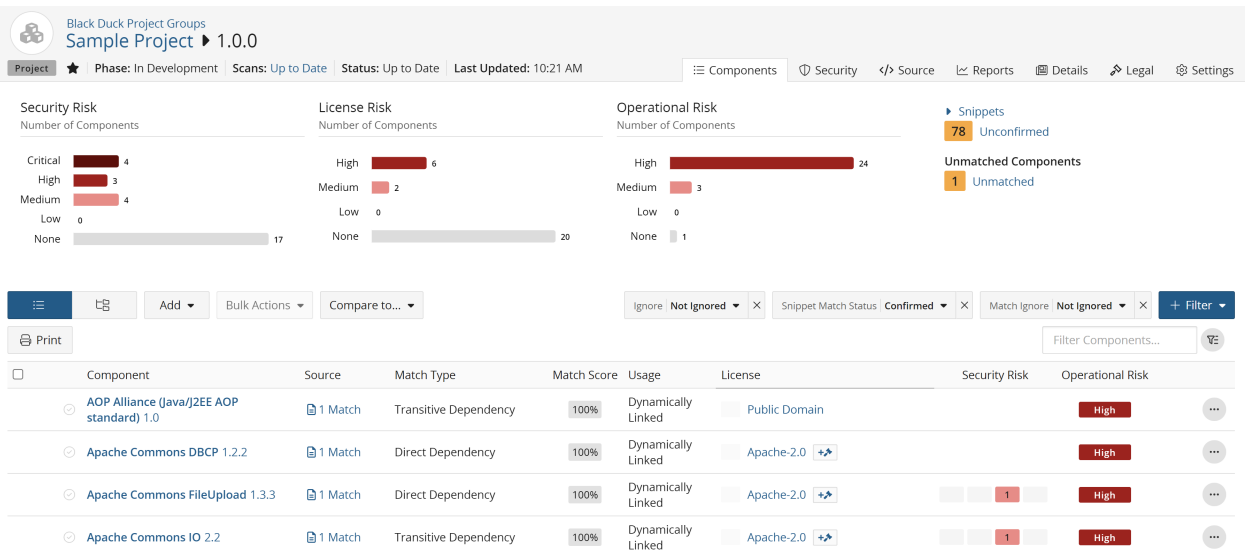
Once you have mapped a component scan to a project version, the scan results automatically populate the project version's BOM with the discovered components. Although the BOM contains all the components discovered in the mapped scan, there may be other components that you are using in that version of your project that either were not discovered in one of the mapped scans or were not scanned.

You can manually add components to the project version's BOM so that they are included in all project version information and risk calculations. You must manually add the component to the BOM of each version of the project in which you use it. You cannot manually add a component to the BOMs of multiple versions of a project at once.

 **Note:** If a subsequent component scan automatically updates the project version's BOM to reflect the discovered OSS components that are included in the BOM, any OSS components that you have manually added to the BOM will be unaffected by that update. Components that are added to the BOM manually can only be [deleted from the BOM](#) manually.

To manually add a component to a BOM:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab.



The screenshot shows the Black Duck Project Groups interface for 'Sample Project 1.0.0'. The 'Components' tab is active, displaying risk metrics and a table of components.

Risk Metrics:

- Security Risk:** Critical (4), High (3), Medium (4), Low (0), None (17)
- License Risk:** High (6), Medium (2), Low (0), None (20)
- Operational Risk:** High (24), Medium (3), Low (0), None (1)

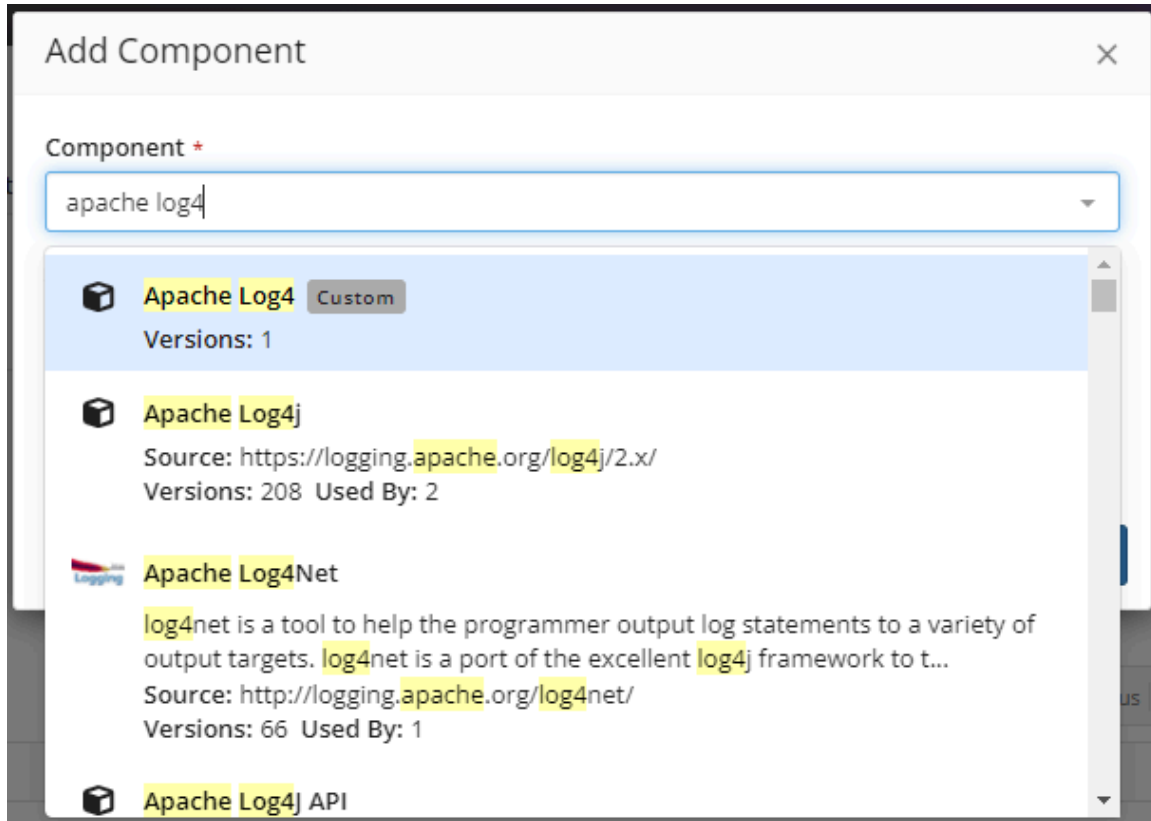
Unmatched Components: 78 Unconfirmed, 1 Unmatched


Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain		High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0		High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4. Click **Add** and select **Component** to open the Add Component dialog box.
5. Enter the name of the component that you want to add. The search results will attempt to return possible matches based on this priority ranking:
 - Exact string matches regardless of component origin i.e. "apache log4j"

- Custom components with fuzzy match
- Fuzzy KnowledgeBase component match used in this or other projects
- Other fuzzy match KnowledgeBase components

Note below, when searching for "apache log4", the "Apache Log4" component appears first as this is a exact string match and is a custom component. It is then followed by Apache Log4j, the KnowledgeBase component match.



6. Optionally, enter or select a version and an origin ID.
7. Optionally, select **Advanced Attributes** and do the following:
 - Enter the purpose for adding this component.
 - Select **Modification** if you modified this component and optionally, enter information regarding the modification.
8. Click **Save**.
 - Black Duck adds the component to the project version's BOM. An  icon appears in the row of the manually added component. If you entered a purpose or you specified that you modified the component and entered information regarding the modification.
 - The **Match Type** column indicates that the component was added to the project version's BOM manually (**Manually Added**).
 - All vulnerability data, license information, version age information, and project development activity information for the component that you added to the BOM is pulled from Black Duck KB and used to update the security, license, and operational risks for this version of your project.

Excluding a component from a BOM


A component's usage indicates how it is intended to be included in the released version of the project.


The usage statuses are:

- Dynamically Linked
- Statically Linked
- Source Code
- Separate Work
- Implementation of Standard
- Merely Aggregated
- Prerequisite
- Dev. Tool / Excluded
- Unspecified

Click [here](#) for more information on usage.

You can change a component's usage to indicate that it is not included in the project version's BOM because it is not actually being distributed with the released project version. For example, if scanning identified development tools in scanned code or a Docker image mapped to the project version, but they will not actually be included in the released version of the project, you should change their usage to exclude them from the project version's BOM.

 **Note:** If you choose to exclude an automatically-added component from a project version's BOM, it will continue to be excluded even if the code or Docker image where it was discovered is rescanned and the BOM is updated.

 **Important:** When you exclude a component from a project version's BOM, the license associated with that component *is not considered* when [calculating the project version's license risk](#). The security and operational risks associated with an excluded component **are still considered** when calculating the project version's security and operational risk.

To exclude a component from a project version's BOM:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components
Critical 4
High 3
Medium 4
Low 0
None 17


License Risk
Number of Components
High 6
Medium 2
Low 0
None 20

Operational Risk
Number of Components
High 24
Medium 3
Low 0
None 1

Snippets
78 Unconfirmed
Unmatched Components
1 Unmatched

Print Add Bulk Actions Compare to... Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain		High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0		High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

- In the component list view of the BOM, click  and select **Edit** to open the Edit Component dialog box.
- Select **Dev. Tool / Excluded** from the **Usage** list,
- Optionally, enter a purpose for this change and/or select the **Modification** checkbox and enter information regarding this modification in the field.
- Click **Save**.


 **Tip:** You can change the [matched component and version](#) and [license](#) at the same time as you change the OSS component's usage.

Deleting a component from a BOM

If you added a component manually to a project version BOM, you can delete it so that it is no longer included in the project version information and risk calculations.

Common reasons to delete a component that was added manually include:

- The same component was discovered in a later component scan and automatically added to the BOM.
- The component version that you selected when you added it was not the correct version.
- You are no longer using component in that project version.

 **CAUTION:** You cannot manually delete components that were automatically added to a project version's BOM. You can [ignore an automatically-added component in the BOM](#) so that it is not included when calculating the security, license, and operational risks for this version of your project. If you want to completely remove an automatically-added component from a project version's BOM, you must remove it from your source code or Docker image and then rescan. This will automatically update the project version's BOM to reflect only those component's that were automatically discovered in the mapped scans and manually-added to the BOM.

To delete a component that was added manually:


- Log in to Black Duck.

7. Editing a BOM • Removing components from a BOM

2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab.


The screenshot shows the 'Components' tab for a project named 'Sample Project 1.0.0'. At the top, there are navigation tabs for 'Project', 'Phase: In Development', 'Scans: Up to Date', 'Status: Up to Date', and 'Last Updated: 10:21 AM'. Below this, there are three risk summary cards: Security Risk (4 Critical, 3 High, 4 Medium, 0 Low, 17 None), License Risk (6 High, 2 Medium, 0 Low, 20 None), and Operational Risk (24 High, 3 Medium, 0 Low, 1 None). To the right, there are 'Snippets' (78 Unconfirmed) and 'Unmatched Components' (1 Unmatched). Below the risk cards is a toolbar with 'Add', 'Bulk Actions', 'Compare to...', 'Ignore', 'Not Ignored', 'Snippet Match Status', 'Confirmed', 'Match Ignore', 'Not Ignored', and a '+ Filter' button. A 'Print' button is also visible. The main area is a table with columns: Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk. The table lists four components: AOP Alliance (Java/J2EE AOP standard) 1.0, Apache Commons DBCP 1.2.2, Apache Commons FileUpload 1.3.3, and Apache Commons IO 2.2. Each component has a '1 Match' icon, a '100%' match score, and is 'Dynamically Linked'. The license for all is 'Apache-2.0'. The Security Risk for the first three is 'None' and for the last is '1'. The Operational Risk for all is 'High'.

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	None	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	None	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4. In the List view of the BOM, click  and select **Delete** to open the Delete Component dialog box.
5. Click **Delete**.
The BOM is updated and the risk is recalculated.

Removing components from a BOM

The best way to remove components that were automatically added to a component version BOM is to remove the link between the component version and the scan that discovered those components.

 **Note:** If you manually remove automatically-added components from a project version BOM, those components will be automatically added to the project version BOM again if the code or Docker image is rescanned.

To remove a scan from a project version to update the BOM:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security </> Source Reports Details Legal Settings

Security Risk
Number of Components

Critical	4
High	3
Medium	4
Low	0
None	17

License Risk
Number of Components

High	6
Medium	2
Low	0
None	20

Operational Risk
Number of Components

High	24
Medium	3
Low	0
None	1

Snippets
78 Unconfirmed

Unmatched Components
1 Unmatched

Ignore Not Ignored × Snippet Match Status Confirmed × Match Ignore Not Ignored × + Filter

Print Filter Components...

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4. Select the **Settings** tab and then select **Scans**.

Select the name of the scan to display the *Scan Name* page which provides information such as the projects and versions mapped to this scan.

Black Duck Projects
Sample Project ▶ 1.0

Project Versions: 1 | Phase: In Planning | Distribution: External | Scan Status: Up to Date

Components Security </> Source Reports Details Legal Settings

Version Details

Scans >

Activity

Scans
Your project version includes 1 scan with 1.19 MB of code scanned.

Status	Name	Scan Size	Last Updated
✓	ComplexBomMainProject_2015-12-04 10:28:23	1.19 MB	May 29, 2019

Displaying 1-1 of 1


5. Click in the row of the scan you want to remove the link (unmap) and then select **Unmap from Project**.


Black Duck removes the mapping between the scan and the project version. This removes all OSS components discovered in that scan from the BOM.

Ignoring a component in a BOM

You ignore an OSS component in the BOM of a project version so that any associated risks are excluded from the risk calculations.

Ignoring a component is considered a component adjustment. Therefore, if you selected to apply **persistent edits**, ignoring a component applies to all versions of the project.

 **Note:** If you ignore an automatically-added OSS component from a project version BOM, it will continue to be ignored even if the code where it was discovered is rescanned to update the BOM.

 **Note:** You cannot ignore manually added components.

7. Editing a BOM • Ignoring a component in a BOM

To ignore a component in a project version BOM:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components
Critical 4
High 3
Medium 4
Low 0
None 17


License Risk
Number of Components
High 6
Medium 2
Low 0
None 20

Operational Risk
Number of Components
High 24
Medium 3
Low 0
None 1

Snippets
78 Unconfirmed
Unmatched Components
1 Unmatched

Print Add Bulk Actions Compare to... Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter Components...

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
<input type="checkbox"/> AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
<input type="checkbox"/> Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
<input type="checkbox"/> Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
<input type="checkbox"/> Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4. In the List view of the BOM, click  and select **Ignore** to open the Ignore Component dialog box.
5. Click **Ignore**.
The component is ignored when calculating project version risk and is not displayed in the BOM.

To ignore multiple components in a project version BOM:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Check the box next to any number of components.
3. Click the **Bulk Actions** button.
4. Select **Ignore**.

The **Bulk Action: Ignore** dialog box appears.

Bulk Action: Ignore

This bulk action will apply to all versions of SampleHierBomProject1 - excluding archived versions.

Are you sure you want to ignore the 4 selected components?

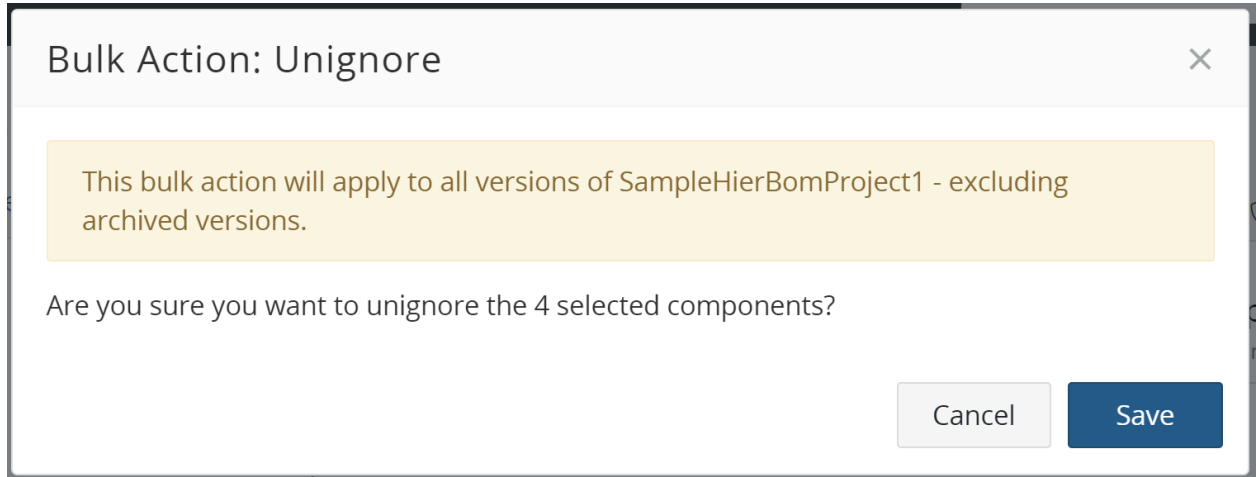
Cancel Save

5. Click the **Save** button to perform the action or the **Cancel** button to exit the dialog box.

To unignore multiple components in a project version BOM:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Check the box next to any number of components.
3. Click the **Bulk Actions** button.
4. Select **Unignore**.

The **Bulk Action: Unignore** dialog box appears.



To view ignored components:

1. While viewing the BOM using the component list view, select **Ignore** from the **Add filter** list.
A list of filters appears.
2. Select **Ignored** and click **OK**.
The table displays all ignored components.

Adjusting the component and/or component version in a BOM

Once you have mapped a component scan to a project version, the scan results automatically create the project version's BOM. Although component scanning automatically discovers the open source component and component version from most archive files by comparing them to components in Black Duck KB, you may be using a version of the component that is not available in Black Duck KB, or you may be using a modified version of a component. You can adjust the component and version for a component in a BOM.

- If the component/version is available in Black Duck KB, users with the appropriate [role](#) can adjust the component or component version, as described below.
- If the component version of a component is not available in Black Duck KB, users with the [Component Manager role](#) can create a custom version and add it to the BOM.

To select an alternate component and/or version match for a component in a BOM:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.

3. Select the version name to open the **Components** tab and view the BOM.

The screenshot shows the Black Duck Project Groups interface for 'Sample Project 1.0.0'. It features three risk charts: Security Risk (Critical: 4, High: 3, Medium: 4, Low: 0, None: 17), License Risk (High: 6, Medium: 2, Low: 0, None: 20), and Operational Risk (High: 24, Medium: 3, Low: 0, None: 1). Below the charts is a table of components with columns for Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk. The table lists four components: AOP Alliance (Java/J2EE AOP standard) 1.0, Apache Commons DBCP 1.2.2, Apache Commons FileUpload 1.3.3, and Apache Commons IO 2.2.

4. In the component list view of the BOM, click and select **Edit** to open the Edit component dialog box.
5. Type the name of the OSS component in the **Component** field and select the alternate match.
6. Select the version of the component from the **Version** list. The list contains all versions of the component that are available in Black Duck KB.
7. Optionally, enter a purpose for this adjustment and/or select the **Modification** checkbox and optionally, enter information regarding this modification in the field.
8. Click **Save**.

The component and version for the BOM entry are updated. The Information indicator () appears in the table row to indicate that the component and/or version were changed from the one automatically discovered in the component scan:

The screenshot shows the Black Duck Project Groups interface for 'Sample Project 1.0.0' after editing. The table now shows one component: Apache Commons Collections ? with an information icon in the Operational Risk column. The Security Risk is 1 and Operational Risk is 3.

Editing an origin or origin ID

You can select a different origin or origin ID shown for a Linux distribution and used in a project version's BOM.

To select a different origin or origin ID:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to display the **Components** tab and view the BOM.

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM Components Security </> Source Reports Details Legal Settings

Security Risk
Number of Components

Critical 4
High 3
Medium 4
Low 0
None 17

License Risk
Number of Components

High 6
Medium 2
Low 0
None 20

Operational Risk
Number of Components

High 24
Medium 3
Low 0
None 1


Snippets
78 Unconfirmed

Unmatched Components
1 Unmatched

Print Add Bulk Actions Compare to... Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

4.

In the component list view of the BOM, click  and select **Edit** to open the Edit component dialog box.

Edit Component ✕

This adjustment will apply to all versions of project2 - excluding archived versions.

Component *

| 📦 jackson-databind ✕ ▾

Version

2.13.0 ✕ ▾

Origin ID

maven ▾

com.fasterxml.jackson.core:jackson-databind:2.13.0 ✕ ▾

Usage

Dynamically Linked ▾

Purpose

Modification

Cancel
Save

5. If the component you selected does not have a distribution, the **Origin ID** lists do not appear. If necessary, select a different component and version to display the **Origin ID** lists.
6. Select the name of the distribution and then the version from the **Origin ID** lists.
 - 📘 **Tip:** You can edit the [matched component and version](#), [license](#), and [usage](#) at the same time as you change the origin and origin ID.
7. Optionally, enter a purpose for this adjustment and/or select the **Modification** checkbox and enter information regarding this modification in the field.
8. Click **Save**.
The origin and/or origin ID is updated. If the new values carry a different type of risk than the previous one, the security risk calculations for the OSS component and for the project version are updated.


Editing the component usage type

A component's usage indicates how it is intended to be included in the released version of the project.

The usage statuses are:

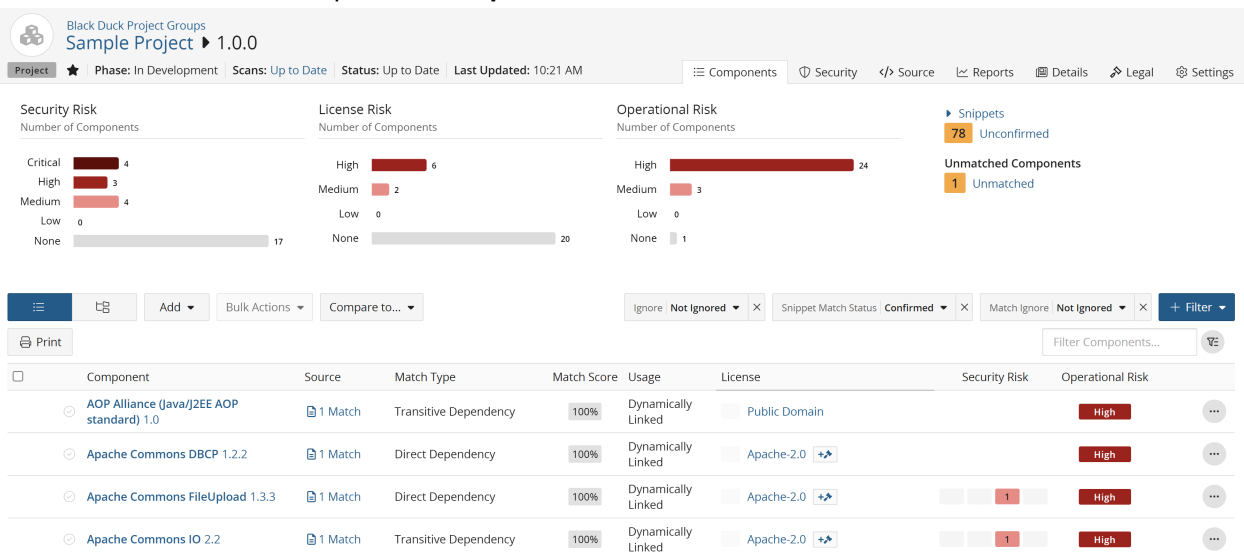
- Dynamically Linked
- Statically Linked
- Source Code
- Separate Work
- Implementation of Standard
- Merely Aggregated
- Prerequisite
- Dev. Tool / Excluded
- Unspecified

Click [here](#) for more information on usage.

 **Note:** It is not possible to edit the usage type for binaries and snippets through the Source view. Usage can only be edited via the BOM component.

To change a component's usage type:

1. Log into Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.



Black Duck Project Groups
Sample Project 1.0.0

Project Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Security Risk: Critical 4, High 3, Medium 4, Low 0, None 17


License Risk: High 6, Medium 2, Low 0, None 20

Operational Risk: High 24, Medium 3, Low 0, None 1

Snippets: 78 Unconfirmed

Unmatched Components: 1 Unmatched

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain		High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0		High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

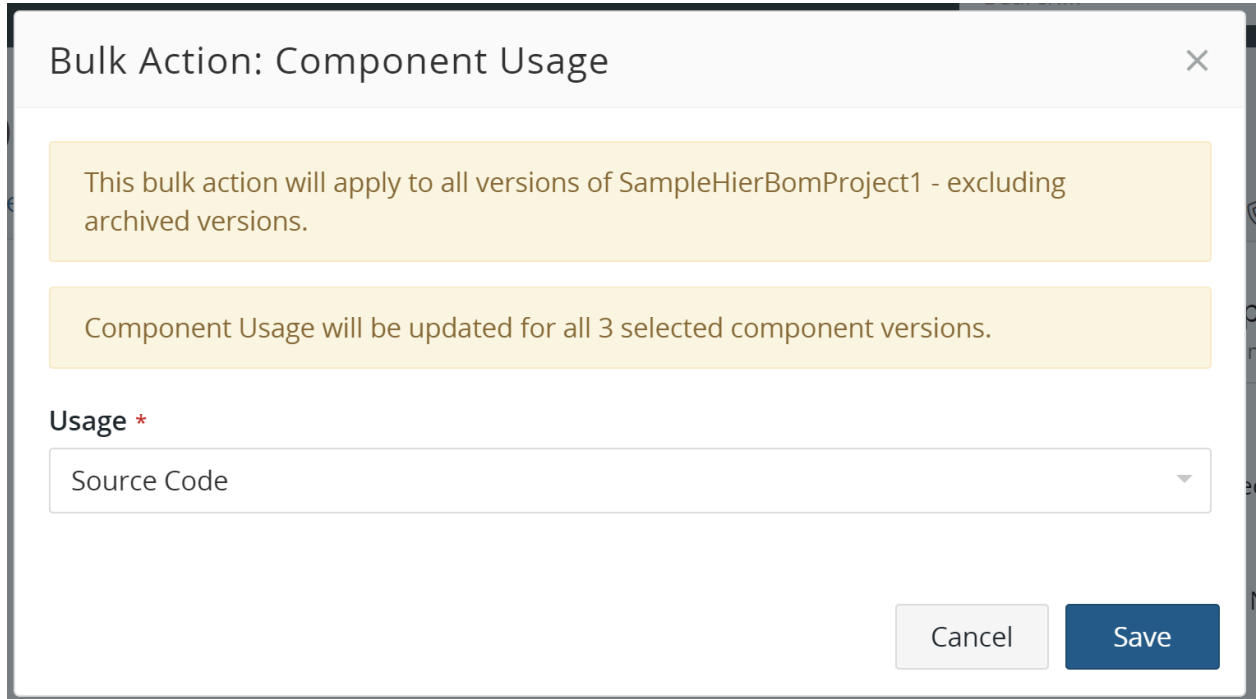
4. In the component list view of the BOM, click  and select **Edit** to open the Edit Component dialog box.
5. Select any of the options from the **Usage** list.

6. Click **Save**.

To change multiple component's usage type:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Check the box next to any number of components.
3. Click the **Bulk Actions** button.
4. Select **Component Usage**.

The **Bulk Action: Component Usage** dialog box appears.




5. Select any of the options from the **Usage** list.
6. Click **Save**.

Modifying licenses in a BOM

So that you can successfully manage license risk, you may need to edit the license(s) for a component version used in a BOM so that it is different from the component's declared license identified in Black Duck KB.

You can modify a single license or include multi-license scenarios, such as "License A AND License B" or "License A OR License B". This lets you accurately represent the licenses in Black Duck for the components in your projects

If you have modified a license, you can select to revert it back to the license as defined by Black Duck KnowledgeBase.

 **Note:** Edits made to a license in the BOM are *local* edits. These edits apply to this version of the component in this BOM only.

To modify licenses:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.
4. Select the single license or multi-license to open the *Component Name Version* Component License dialog box.

Component Name Version Component License

Attribution Statement ▾

License Edit

Select a license to view. Toggle Edit Mode on to edit.

(Apache License 2.0)

Include in Notices File Report

Edit Mode off

Close Save Changes

5. Select the Edit Mode option to enable editing.

Component Name Version Component License

Attribution Statement ▾

License Edit

Modify the existing license or add a new license/group. ?

(Apache License 2.0 ▾)

[+ Add License](#) [+ Add Group](#) [Reset Changes](#)

Include in Notices File Report

Edit Mode on

Close Save Changes

6. Edit the license as described [here](#).

Selecting the license term fulfillment status

After a License Manager has defined the license terms that must be fulfilled and the system administrator [has enabled the License Fulfillment tab on the Legal tab](#), BOM Managers, and other users with the appropriate [role](#), can denote the fulfillment status for a license term by using the *Project Version Legal* tab.

By default, the fulfillment status of a license term is unfulfilled.

To change the fulfillment status for a license term::

1. From a project version BOM, select the **Legal** tab, and if necessary, the **Term Fulfillment** tab, to view a list of license terms that require fulfillment.

7. Editing a BOM • Selecting the license term fulfillment status

The screenshot shows the Black Duck Projects interface for 'Sample Project 1.0'. The 'Legal' tab is active, displaying a table of license terms. The table has four columns: Fulfillment, Term, Responsibility, and Category. There are three rows of data, all with 'Permitted' responsibility and 'KnowledgeBase' category. The 'Fulfillment' column shows icons: a red 'X' for 'Private Use', a red 'X' for 'Place Warranty', and a green checkmark for 'Modify'. Above the table, there are filters for 'Fulfillment' (set to 'Not Fulfilled') and a search box for 'Filter Terms...'. A 'Term Fulfillment' dialog box is also visible at the top left.

Fulfillment	Term	Responsibility	Category
	Private Use	Permitted	KnowledgeBase
	Place Warranty	Permitted	KnowledgeBase
	Modify	Permitted	KnowledgeBase

Displaying 1-3 of 3

By default, the **Legal** tab is filtered to show all license terms that are not fulfilled.

The tab displays the following information:

Column	Description
Fulfillment	Indicates fulfillment status: <ul style="list-style-type: none"> indicates this license term is not fulfilled. indicates this license term is fulfilled.
Term Name	License term name. Select the term to display the Term Fulfillment dialog box from which you can manage the fulfillment status for all licenses that have this term.
Responsibility	Indicates the responsibility for this term. Possible values are Required, Forbidden, or Permitted.
Category	Category for this license term.

2. Select a license term to view all licenses with this license term in this BOM which require fulfillment.

The Term Fulfillment dialog box appears.

Term Fulfillment
✕

Term	Source	Responsibility	Category
Private Use	KnowledgeBase	Permitted	KnowledgeBase

▶ Description

Mark as fulfilled
Mark as unfulfilled
Fulfillment | Not Fulfilled ▼ ✕

Add Filter

<input type="checkbox"/>	Fulfillment	Component ▼	License	Last Updated
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Commons IO 1.1	Apache License 2.0	-
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Apache-Jakarta Jmeter 2.1.1	Apache License 2.0	-
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Apache-Jakarta Jmeter 2.0.3	Apache License 2.0	-
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Apache Lucene 1.4.3	Apache License 2.0	-
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Apache Commons FileUpload 1.3.3	Apache License 2.0	-

Displaying 1-5 of 5

Close

This dialog box lists the component name and version, license that includes this term, and the username and date that this license term was last updated.

- indicates this license term is not fulfilled.
 - indicates this license term is fulfilled.
3. Select one or more checkboxes to denote the fulfillment status.
To select all terms on a page, select located at the top of the table.
 4. Select **Mark as fulfilled** to indicate this license term is fulfilled or **Mark as unfulfilled** to indicate this license term is unfulfilled.
 5. Click **Close**.

Editing license text in the BOM


You may notice that the license text for some components is incomplete as the Black Duck KB may not have the full license text for some components. Since most attribution clauses in licenses usually require at a minimum that the license text be provided in any redistributions, you may need to edit the existing license text.

Note the following:

- Edits to license text only apply to the license text for that component version: edits do not apply to other components with the same license.
- If you selected to [make edits persistent](#) then edits to license text apply to all existing versions of a project and will also be carried forward as additional scans are completed for the same code or Docker image.

7. Editing a BOM • Editing license text in the BOM

- There is an option to revert to the original license text.
- The dialog box displays the first and last name and date or time the license text was edited above the license text.

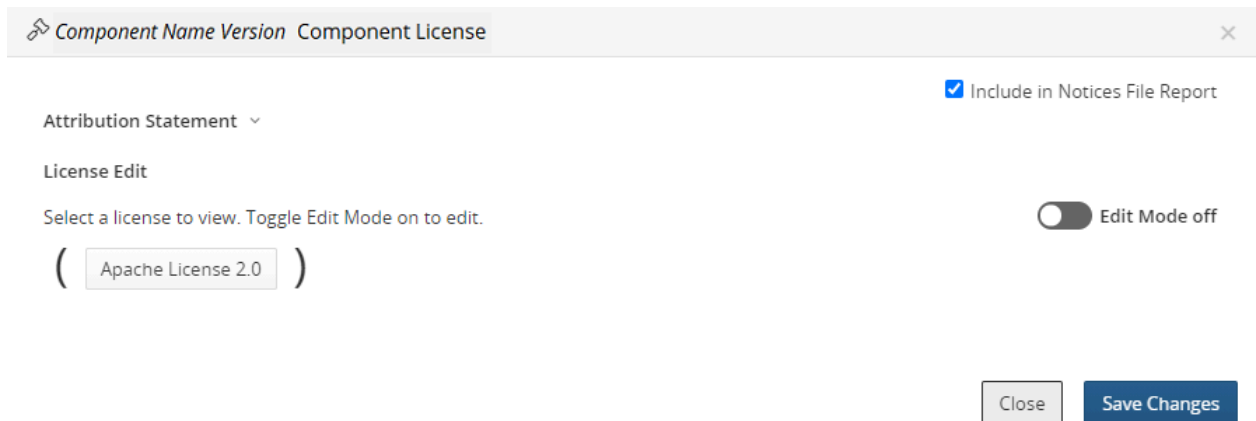
Updated by System Administrator - 11:16 AM 

This message appears for local or [global edits](#) (made by the License Manager).

- If you edited the original license, saved the changes, selected a different license, and then select the original license, your edited version of the license will appear.
- [Edits made globally to licenses](#) by the License Manager will propagate to the version used in the BOM unless the BOM Manager or Project Manager has edited the license,

To edit license text:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.
4. Select the license name to open the *Component Name Version* Component License dialog box.




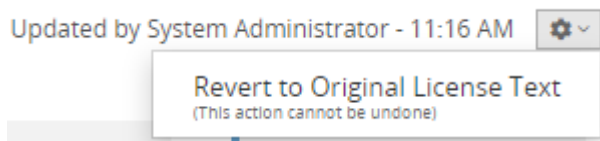
5. Select the license you wish to edit.

The dialog box expands to show the obligations and license text for the selected license.

6. Edit the text directly in the field.
7. Click **Save Changes**.

To revert to the original license text:

1. Open the *Component Name Version* Component License dialog box as described above.
2. Click  located above the license text and select **Revert to Original License Text**.



Managing subprojects

You may have applications that include code from other projects, for example, a user management module that is included in several other applications. You can see risk information about the user management module as a project with its own BOM but may also want to see the same information in the BOM for every application that uses that module without having to re-scan the code.

Adding projects to your application's BOM gives you a complete view of this application and all associated risks, including vulnerabilities, license, and operational risk.

For these subprojects:

- You must have permission to the project to add it to the BOM.
- Users who do not have permission to the subproject will not be able to drill down to view additional data about that project version.
- Modifications made to a project outside of the BOM will propagate to the subproject in the BOM. For example, if additional scans are completed for scans mapped to this project, those changes will propagate to the subproject.

An exception to this is the subproject version license: edits made to the project version license may or may not propagate to the license shown for the subproject in the BOM:

- If you modify the project version license outside of the BOM and have *not* edited the subproject license from within the BOM, the edited license will appear in the BOM for the subproject.
- If you modify the project version license outside of the BOM and have edited the subproject license from within the BOM, the license edit will *not* appear in the BOM for the subproject.

If you modify the subproject version license from within the BOM, that change is *not* propagated outside of the BOM.

- Policy violations within the subproject will not appear in the BOM. However, a policy violation will appear in the BOM for the subproject if a policy rule is violated at the project level. For example, if you specified a policy rule that triggers a violation for unknown licenses and the project is added to the BOM with an unknown license, a policy violation will be triggered for that subproject.
- Subprojects and their associated licenses are included in the Notices File report. You can [exclude](#) the subproject from the Notices File report.
- Subproject security risks are added to the sum in the parent project.

To add a subproject:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to open the **Components** tab.

7. Editing a BOM • Managing subprojects

Black Duck Project Groups
Sample Project 1.0.0

Project Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components

Critical	4
High	3
Medium	4
Low	0
None	17

License Risk
Number of Components

High	6
Medium	2
Low	0
None	20

Operational Risk
Number of Components

High	24
Medium	3
Low	0
None	1

Snippets: 78 Unconfirmed
Unmatched Components: 1 Unmatched

Print Add Bulk Actions Compare to... Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain	High	High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	High	High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1 High	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1 High	High

3. Click **Add** and select **Project** to open the Add Project dialog box.

4. Enter the name and version of the project.

Note: You must have permission to the project to add it to the BOM.

5. Optionally add a license for this project or modify the existing license. If you do not enter a license, "Unknown License" appears in the BOM for the license for this project. The license selected for the subproject will determine its [license risk](#).

6. Click **Save**.

Black Duck adds the selected project to the BOM.

To edit a project:

1. Select the BOM as described in the previous section.

2. Click and select **Edit** to open the Edit Component dialog box.

3. Select one or more different values and click **Update**.

To delete a subproject from a BOM:

1. Select the BOM as described in the previous section.

2. Click and select **Delete** to open the Delete Component dialog box.

3. Click **Delete**.

The BOM is updated and the risk is recalculated.

To view where projects are included as subprojects:

The **Where Used** table lists the projects where this project version is included in the BOM.

1. Locate the project using the **Projects** tab on the Dashboard by selecting the name of the project to go to the *Project Name* page.

2. Select the version name which opens the **Components** tab.

3. Select the **Details** tab to view where this project version is included as subprojects.

Black Duck Projects
Sample Project ▾ 1.0

Project ☆ | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

Where Used

Project	Version	Tier	Released	Distribution	Phase
Sample Projects 4	1.0		Never	External	In Planning

Displaying 1-1 of 1

Description
No description.

Created
Sep 4, 2018 by sysadmin

Updated
Sep 4, 2018 by username

Last Scan
Tue, Sep 4, 2018 12:55 PM

Last KnowledgeBase Update
Wed, Sep 5, 2018 3:37 PM

Tags
No Tags

The **Where Used** table lists the project name, project version, tier, release date, distribution, and phase for all projects where this project version is a subproject.

Reviewing snippet matches

Use the **Source** tab to determine if the snippet belongs in your BOM and if so, if the snippet match is correct.

Click [here](#) for more information on using the **Source** tab.

Snippets in the BOM

If a snippet scan has been run and snippet matches were found, a snippet badge appears next to the risk charts in the BOM indicating the number of snippets that need confirmation.

▶ Snippets
35 Unconfirmed

By default, the BOM does not display your unconfirmed snippet matches. Unlike reviewing a component in the BOM (which marks all instances of that component as reviewed) snippet matches are confirmed on the match level. Only after a snippet match has been confirmed will it appear unfiltered in the BOM.

You can filter the BOM to view unconfirmed snippet matches by selecting the **Unconfirmed** option for the **Snippet Match Status** filter and the not ignored snippet matches by selecting the **Not Ignored** option for the **Ignore** filter.

7. Editing a BOM • Reviewing snippet matches

The screenshot displays the Black Duck Project Groups interface for a project named "Sample Project" (version 1.0.0). The interface includes a navigation bar with tabs for Project, Security, Source, Reports, Details, Legal, and Settings. Below the navigation bar, there are three risk assessment charts: Security Risk, License Risk, and Operational Risk. The Security Risk chart shows 5 Critical, 3 High, 4 Medium, 0 Low, and 16 None components. The License Risk chart shows 6 High, 2 Medium, 0 Low, and 20 None components. The Operational Risk chart shows 23 High, 3 Medium, 0 Low, and 2 None components. A "Snippets" section on the right indicates 78 Unconfirmed and 1 Unmatched components. Below the charts, there is a table of snippet matches with columns for Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk. The table lists five components: AdNauseam v0.8.4a, Apache Tapestry 5.4-alpha-15, Apache Wicket 0.9.3, Apache Wicket 0.9.6, and Bokeh 0.11.0. Each component row includes a "2 Matches" or "1 Match" indicator, a "Snippet" match type, a "--" match score, and a license type (GPL-3.0+, Apache-2.0, or BSD-3-Clause) with a "+/-" icon. A "Print" button and a "Filter Components..." search box are also visible.

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AdNauseam v0.8.4a	2 Matches	Snippet	--	Source Code	GPL-3.0+		
Apache Tapestry 5.4-alpha-15	1 Match	Snippet	--	Source Code	Apache-2.0		
Apache Wicket 0.9.3	1 Match	Snippet	--	Source Code	Apache-2.0		
Apache Wicket 0.9.6	3 Matches	Snippet	--	Source Code	Apache-2.0		
Bokeh 0.11.0	1 Match	Snippet	--	Source Code	BSD-3-Clause		

Retaining partial snippet identifications

By default, identifications you made to partial snippet matches are not retained in subsequent snippet rescans.

You can change this default setting so that you can minimize the number of snippet matches you need to re-identify: in the project's **Settings** tab, in the **Snippet Adjustments** section, select **Apply IDs from partial snippet matches to new exact file matches**.

8. Managing components

Users with the [Component Manager role](#) can:

- Create and manage [custom components](#).
- Modify Black Duck [KnowledgeBase components](#).
- Manage components with [unmatched origins](#).

Use the Component Management table to manage components.

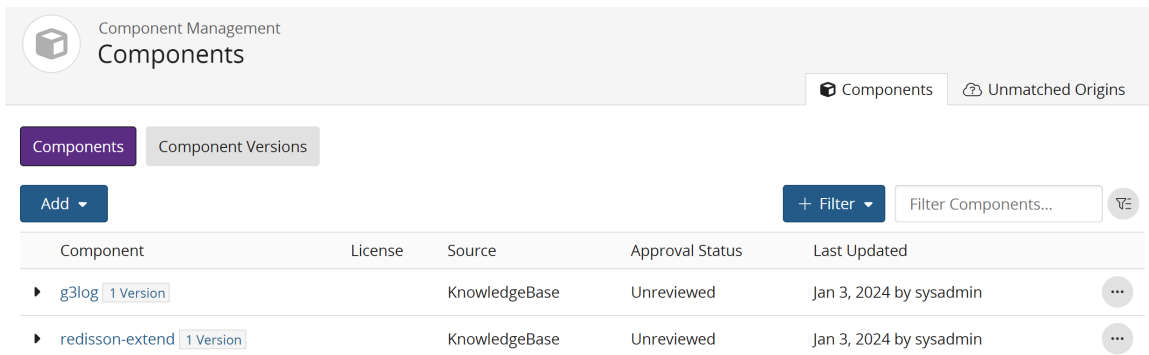
To view managed components:

1. Log in to Black Duck with the Component Manager role.

2.

Click  > **Components**.

The **Components** tab appears.



Component Management
Components

Components Unmatched Origins


Components Component Versions

Add Filter Filter Components...

Component	License	Source	Approval Status	Last Updated
▶ g3log <small>1 Version</small>		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin
▶ redisson-extend <small>1 Version</small>		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin

Displaying 1-2 of 2

The table in **Components** tab contains the following information:

Column	Description
Component	<p>Name of the component. Select the component name to open the Overview tab of the <i>Component Name</i> page. If there are multiple versions for this component, select > to display the versions. Select a version to open the Details tab of the <i>Component Name > Version</i> page.</p> <p> indicates that there is a note for this component or component version. Hover over the icon to view the information.</p>
License	License for this component.
Source	<p>Source for this component. Possible values are:</p> <ul style="list-style-type: none"> • Custom. A custom component. • KnowledgeBase. An unmodified Black Duck KnowledgeBase component.


Column	Description
	<ul style="list-style-type: none"> Modified KnowledgeBase. A modified Black Duck KnowledgeBase component.
Approval Status	<p>Approval status of this component. Possible values are:</p> <ul style="list-style-type: none"> Unreviewed In Review Reviewed Approved Limited Approval Rejected Deprecated

Select the **Component Versions** tab to view information for each component version.

Column	Description
Component Version	<p>Name of the component version.</p> <p>Select the component name to open the Overview tab of the <i>Component Name</i> page.</p> <p>Select the version to open the Details tab of the <i>Component Name > Version</i> page.</p>
License	License for this component version.
Source	<p>Source for this component version. Possible values are:</p> <ul style="list-style-type: none"> Custom. A custom component version. KnowledgeBase. An unmodified Black Duck KnowledgeBase component version. Modified KnowledgeBase. A modified Black Duck KnowledgeBase component version.
Approval Status	<p>Approval status of this component version. Possible values are:</p> <ul style="list-style-type: none"> Unreviewed In Review Reviewed Approved Limited Approval Rejected Deprecated
Last Updated	Date this component version was last modified and the user who last modified it.

About custom components

You may want to use a component in your BOM that is not available from Black Duck KnowledgeBase; for example, your project uses an open source component that is not tracked by the Black Duck KB or there is a commercial component you want to add to your BOM. So that your BOM accurately reflects your project, users with the [Component Manager role](#) can create and manage custom components which can then be added to a BOM.

 **Note:** Contact Black Duck Customer Support for missing versions of open source components that are managed by Black Duck KnowledgeBase.

Black Duck provides the information Component Managers need to successfully manage their custom components. They can use the:

- *Custom Component Name* **Overview** tab to [view the versions for a component](#), including the status, description, and tags for this custom component.
You can also use this tab to [create tags](#) for the custom component.
- *Custom Component Name* **Settings** tab to view and/or [edit the details of a component](#).
Use this tab to delete a custom component.
- *Custom Component Name > Version* **Details** tab which provides details of this component version and lists the [projects used by a custom component version](#).
Component Managers must have permission to view the projects for them to appear on this page.
- *Custom Component Name > Version* **Settings** tab to view and/or edit the [details of a component version](#).
Use this tab to delete a custom component version.

Note the following:

- In the BOM:
 - The match type for a custom component added to a BOM is **Manually Added**.
 - Custom components display license risk. (Note that the license risk shown is [determined by the license](#) selected for this component.) No security risk values are shown as no security vulnerabilities are associated with the custom component. Also, no operational risk is shown.
- Policy Managers can create policy rules for custom components.
- You can [use the search feature](#) for custom components. A component filter, **Component Source**, has the value **Black Duck Custom Component** for custom components.
- In the `components_date_time.csv` and `bom_component_custom_fields_date_time.csv` files in the [Project Version report](#), a new column, labeled **Source/Type** denotes whether the component is a custom component (value of CUSTOM_COMPONENT) or a component that is managed by Black Duck KnowledgeBase (value of KB_COMPONENT).
- Custom components do not have origins.


Managing custom components

Users with the Component Manager [role](#) can create, edit, and delete custom components.

You can:


- Create custom components.
- View custom component information.
- Edit custom components.
- Delete a component.
- [Create additional versions for a custom component](#).
- [Add a status](#).

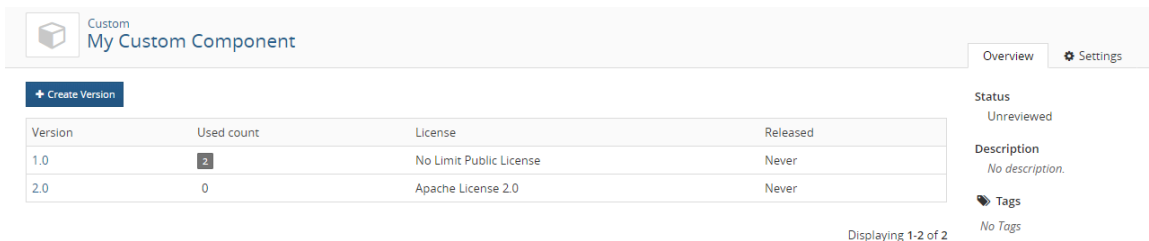
Creating custom components

1. Log in to Black Duck with the Component Manager [role](#).
2.  Click **Manage** > **Components**.
The **Components** tab appears.
3. Click **Add** > **Create a component**.
The Create a Component dialog box appears.
4. Enter the component name, version, and license, which are required fields, and optionally, values for the origin ID, description, URL, and release date. If you enter any data in the origin ID fields, the Namespace, Package ID, and Origin Version fields become mandatory.
5. Click **Create**.

The **Components** tab appears with the new component listed in the table; the **Component Versions** tab lists the new component and version.

Viewing custom component information

1. Log in to Black Duck with the Component Manager [role](#).
2.  Click **Manage** > **Components**.
The **Components** tab appears.
3. Select the component name you wish to view information.
The **Overview** tab of the *Component Name* page appears.



Version	Used count	License	Released
1.0	2	No Limit Public License	Never
2.0	0	Apache License 2.0	Never

Displaying 1-2 of 2

If provided, additional information, such as the status, description, and tags for the custom component is shown along with the following information.

Column	Description
Version	Version(s) for this component. Select a version to open the Details tab of the <i>Component Name</i> > <i>Version</i> which lists the projects that use this component version.
Used Count	Number of projects that use this component version.
License	License for this component version.

Column	Description
Release	Release date for this component version. Never is listed if a value was not entered.

Editing custom components

1. Log in to Black Duck with the Component Manager [role](#).

2.



Click **Manage** > **Components**.

The **Components** tab appears.

3. Select the component you wish to modify.

The *Component Name* page appears listing the versions for this component.

4. Select the **Settings** tab to add or edit the information for this component, such as a description, URL, notes, or to define a [status](#) for this component.

Custom Component

Overview Settings

Component Details

Custom Fields

SBOM Fields

Component Name *

Custom Component

Description

URL

Notes

Approval Status *

Unreviewed

Save

Delete Component

Once you delete a component, you will lose all information and versions related to the component.

Delete Component

5. Click **Save**.

Deleting custom components

You cannot delete a custom component that is in use.

To delete a custom component:

1. Log in to Black Duck with the Component Manager [role](#).



2.



Click **Manage** > **Components**.

The **Components** tab appears.

3. Do one of the following:

-  Click  in the row of the component that you want to delete and select **Delete**.
- Click the custom component you wish to delete to view the **Overview** tab of the *Component Name* page.
Select the **Settings** tab and click **Delete Component**.

4. Click **Delete** to confirm in the Delete Custom Component dialog box.

Managing custom component versions

Users with the Component Manager [role](#) can:

- Create additional versions for a custom component.
- View where a custom component version is used.
- Edit version information.
- Delete a version.

Creating additional versions for a custom component

1. Log in to Black Duck with the Component Manager [role](#).

2.



Click  > **Components**.

The **Components** tab appears.

3. Select the component to which you want to add versions. Note that you can also select the component from the **Component Versions** tab.

The **Overview** tab of the *Component Name* page appears listing the versions for this component.

4. Click **Create Version**.

The Create a New Version dialog box appears.

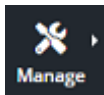
5. Enter the version and license, and optionally, select a release date for this version and click **Create**.

The **Details** tab of the *Component Name > Version* page appears for the new version.

Viewing the projects where a version is used

1. Log in to Black Duck with the Component Manager [role](#).

2.



Click  > **Components**.

The **Components** tab appears.

3. Select the **Component Versions** tab.

Component Management
Components

Components Unmatched Origins

Components Component Versions

Add + Filter Filter Components...

Component Version	License	Source	Approval Status	Last Updated
g3log - v1.1	Public Domain	KnowledgeBase	Approved	Jan 3, 2024 by System Administrator
redisson-extend - 2.15.0	3dsview License	Modified KnowledgeBase	Deprecated	Jan 3, 2024 by System Administrator

Displaying 1-2 of 2

4. Select the version to open the **Details** tab of the *Component Name* > *Version* page.

Black Duck Custom Components
New Custom Component • 1.0

Custom Versions: 0 | Phase: In Development | Distribution: External

Details Settings

Where Used

Project	Version	Tier	Released	Distribution	Phase
Sample Project	1.0		Never	External	In Planning

Description
New component needed for Sample Project
Released on May 9, 2018

Licenses
[template] Basic MIT-Style License
Permissive

Displaying 1-1 of 1

The **Where Used** table lists the projects that use this version.

Note: You must have permission for a project for you to view it on this page.

From this table:

- Select the project name to view the [Project Name](#) page.
- Select the project versions to view the BOM.
- Select the license to view the license text.

Editing custom component versions

1. Log in to Black Duck with the Component Manager [role](#).

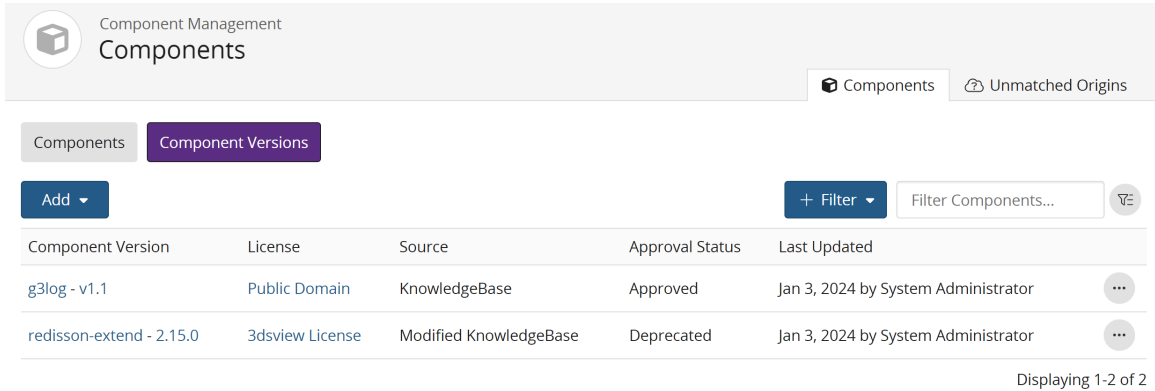
2.



Click **Manage** > **Components**.

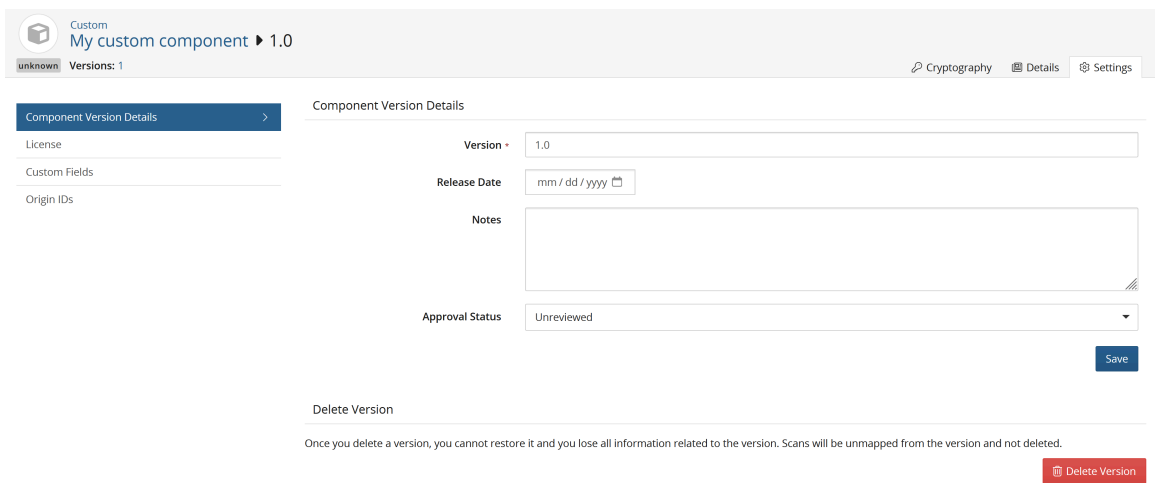
3. Select the **Component Versions** tab.

8. Managing components • About custom components



The screenshot shows the 'Component Management' interface. At the top, there's a header with 'Component Management' and 'Components'. Below this, there are tabs for 'Components' and 'Unmatched Origins'. A secondary set of tabs shows 'Components' and 'Component Versions', with 'Component Versions' being the active tab. There's an 'Add' button and a '+ Filter' button with a search input 'Filter Components...'. Below this is a table with columns: 'Component Version', 'License', 'Source', 'Approval Status', and 'Last Updated'. The table contains two rows: 'g3log - v1.1' with license 'Public Domain', source 'KnowledgeBase', status 'Approved', and 'Jan 3, 2024 by System Administrator'; and 'redisson-extend - 2.15.0' with license '3dsview License', source 'Modified KnowledgeBase', status 'Deprecated', and 'Jan 3, 2024 by System Administrator'. At the bottom right, it says 'Displaying 1-2 of 2'.

4. Select the version to open the **Details** tab of the *Component Name > Version* page.
5. Select the **Settings** tab to edit the information.



The screenshot shows the 'Component Version Details' page for a custom component named 'My custom component' with version '1.0'. The page has tabs for 'unknown', 'Versions: 1', 'Cryptography', 'Details', and 'Settings'. The 'Details' tab is active. On the left, there are sub-tabs: 'Component Version Details', 'License', 'Custom Fields', and 'Origin IDs'. The main content area has fields for 'Version' (1.0), 'Release Date' (mm/dd/yyyy), 'Notes' (a text area), and 'Approval Status' (Unreviewed). There is a 'Save' button at the bottom right. Below the form, there is a 'Delete Version' link and a warning: 'Once you delete a version, you cannot restore it and you lose all information related to the version. Scans will be unmapped from the version and not deleted.' A red 'Delete Version' button is also present.

- Select **Component Version Details** to edit the version, release date, notes or approval status.
- Select **License** to [modify the existing license or add a new license or group](#).
- Select **Custom Fields** to populate any [custom fields](#) enabled in your environment.
- Select **Origin IDs** to manage origin IDs mapped to this custom component. You can create new origin IDs so that when it is found in a scan, it will automatically be matched to this component. You can also delete any origin IDs associated to this custom component.

6. Click **Save**.

Deleting a custom component version

There must be at least one version for a custom component.

You cannot delete a version that is being used in a project.

To delete a custom component version:

1. Log in to Black Duck with the Component Manager [role](#).
- 2.

Click  **Manage** > **Components**.

The **Components** tab appears.

3. Select the **Component Versions** tab.

Component Management
Components

Components Unmatched Origins

Components Component Versions

Add + Filter Filter Components...

Component Version	License	Source	Approval Status	Last Updated
g3log - v1.1	Public Domain	KnowledgeBase	Approved	Jan 3, 2024 by System Administrator
redisson-extend - 2.15.0	3dsview License	Modified KnowledgeBase	Deprecated	Jan 3, 2024 by System Administrator

Displaying 1-2 of 2

4. Click in the row of the custom component version you wish to remove and select **Delete**.
The Delete Custom Component dialog box appears.

5. Click **Delete**.

You can also delete a version using the **Settings** tab, as described in the previous section.

About Black Duck KnowledgeBase components

The Black Duck® KnowledgeBase™ (Black Duck KB) is the industry's most comprehensive database of open source component information. Since 2003, Black Duck has searched the Internet for information on open source software (OSS) components and downloadable source code. The complete version of Black Duck KB includes more than 2 million unique components from more than 10,000 sites and contains detailed data on more than 79,000 actively traced vulnerabilities across more than 530 billion lines of code. The Black Duck KB includes detailed data for more than 2,500 unique licenses, including the full license text and dozens of encoded attributes and obligations for each license. Black Duck connects to a version of Black Duck KB hosted in the cloud.

New OSS component versions and meta data, such as vulnerabilities, are continually added and updated to the version of Black Duck KB that supports Black Duck.

The Black Duck KB provides information about OSS components at the [component level](#) and at the [component version level](#).

So that your BOM accurately reflects your project, users with the [Component Manager role](#) can:

- [Modify](#) Black Duck KB components and/or Black Duck KB component versions.
- Undo these modifications and [reset the KB data](#) back to its original values.
- [Define an approval status](#) for a Black Duck KB component and/or component version to ensure that only approved components/version are included in your BOM.

Understanding the component information available from the Black Duck KB

The Black Duck KB *Component Name* page displays information about the open source software (OSS) component.

8. Managing components • About Black Duck KnowledgeBase components

The screenshot displays the Apache Tomcat component overview page. At the top, it shows the component name 'Apache Tomcat' with 'Versions: 911'. Below this is a 'Vulnerability History' section with a line graph showing vulnerability counts for various versions from 2002 to 2022. The graph has a legend for Critical, High, Medium, Low, and Unknown vulnerabilities. Below the graph is a table with columns for Version, Used count, License, Released, and Vulnerability Count. To the right, there are sections for Approval Status (Unreviewed), Description (The Apache Tomcat software is an open source implementation of the Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket technologies.), Component Links (http://tomcat.apache.org/), Tags (apache, http, http_server, java, jsp, servlet, web, web_services), and Custom Fields (No custom fields).

Version	Used count	License	Released	Vulnerability Count
-atlassian-hosted	0	Apache License 2.0	Sep 25, 2017	
0.0.1	0	Apache License 2.0	Jul 7, 2014	
0.1	0	Apache License 2.0	Nov 10, 2010	

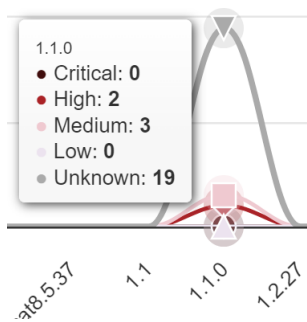
This page is comprised of two tabs: an **Overview** tab and a **Settings** tab.

About the Overview tab

The **Overview** tab displays information such as a status, description, component links, and tags, and information about each of the component versions that are available in Black Duck KB.

A graph at the top of the page shows a history of high, medium, and low vulnerabilities for each version of this component. Use this graph to quickly view vulnerability information for component versions.

- Select **Previous** or **Next** to view older or newer versions.
- Hover over a data point in the graph to view the version, release date, and number of vulnerabilities for this version:



To view information on versions that interest you, use the filter, located above the table, to filter the versions shown in the vulnerability graph and in the table below.

The following information is available for each version:

Column	Description
Version	Release number of this version of the component. Select the version number to display the Component Name > Version page .
Used Count	Number of project version BOMs in which this version of the OSS project is used.

Column	Description
	Tip: Select the number to go to the Details tab for this version of the OSS component. That tab lists each project and project version in which this version of the OSS component is used.
License	Declared license of this version of the OSS component. Other license types include: <ul style="list-style-type: none"> • "Unknown" indicates that the OSS component version's license is not known. • "License Not Found" indicates that although researched by Black Duck, no declared license was found for the component. • "No License" indicates that Black Duck has found a declaration of 'No License' for the component. <p>For known licenses, select the license name to view license details and license text.</p>
Released	The date this version of the OSS component was released.
Security Risk	A graph which shows the number of high risk, medium risk, low risk, and unknown vulnerabilities associated with this version of the OSS component.

About the Settings tab

The **Settings** tab shows details on this component. Information shown here appears on the **Overview** tab.

The screenshot shows the 'Settings' tab for the 'Apache Tomcat' component. At the top, there is a header with the component name, version '778', and tabs for 'Overview' and 'Settings'. Below the header, there are sections for 'Component Details', 'Additional Fields', and 'SBOM Fields'. The 'Additional Fields' section is active and shows a 'Date added to project' field with a calendar icon and a 'Save' button.

Users with the Component Manager [role](#) can use the **Settings** tab to edit the description, URL, notes, and status for this KB component. Click [here](#) for information on editing component information and [here](#) for information on modifying a component's status.

Users with the System Administrator role can use the **Settings** tab to edit the component [custom field information](#), as shown in the **Additional Fields** section.

Understanding the component version information available from the Black Duck KB

On the *Component Name Version* page, the **Details** tab provides the following information:

- Description.
- Count of known security vulnerabilities.
- Associated licenses.
- Component links, if available.
- Tags, if available.
- Date this version was released.
- Number of newer versions.

8. Managing components • About Black Duck KnowledgeBase components

- [Approval Status](#) of this version.
- Date this component was last updated.
- Commit activity and the trend for the component over the last 12 months.
- Number of contributors for the component for the past 12 months.
- A Where Used table which lists the projects and the respective versions in which this version of the component is used.

http://struts.apache.org/
Apache Struts • 2.3.7
 java Versions: 173

Security Cryptography Copyrights Details Settings

40 Vulnerabilities

Description
 The goal of the Apache Struts project is to encourage application architectures based on the "Model 2" approach, a variation of the classic Model-View-Controller (MVC) design paradigm. Under Model 2, a servlet (or equivalent) manages business logic execution, and presentation logic resides mainly in server pages.

Released	Newer Versions	Approval Status	Updated
Nov 6, 2012	77	Unreviewed	Oct 19, 2019

Activity
 Last 12 Months: 353 commits ↓ decreasing
 Last commit: Oct 19, 2019

Community
 Last 12 Months: 18 contributors

Where Used

Project	Version	Released	Phase
bds_jenkins_test	2020_04_03--15_49_49	Never	In Development

Displaying 1-1 of 1

Licenses
 Apache License 2.0

Open Hub
<https://www.openhub.net/p/3569>

Component Links
<http://struts.apache.org/>

Tags
 apache development framework java model-view-controller mvc mvcfamework programming servlet web webapplication

The Where Used table contains the following information:

Column	Description
Project	Name of the project that uses this version of the OSS component from Black Duck KB. Select the project name to display the Overview tab of the Project Name page which provides information on this project.
Version	Version of the project that uses this version of the OSS component from Black Duck KB. Select the version to display the BOM filtered to display that component version.
Released	Date this version was released.
Phase	Development phase that this version of the project is currently in.

On Black Duck KB *Component Name Version* page, the **Security** tab displays the list of vulnerabilities associated with this version of the OSS component from Black Duck KB.

Identifier	Published	Overall Score
> NVD CVE-2016-3081	Apr 26, 2016	9.3 High
> BDSA BDSA-2018-2905 (CVE-2018-11776)	Aug 22, 2018	8.3 High
> BDSA BDSA-2013-0027 (CVE-2013-4316)	Oct 8, 2018	7.4 High
> BDSA BDSA-2014-0067 (CVE-2013-2251)	May 24, 2018	6.5 Medium
> BDSA BDSA-2017-0903 (CVE-2017-9805)	Sep 6, 2017	6.2 Medium
> BDSA BDSA-2014-0117 (CVE-2014-0094)	Feb 19, 2019	6.2 Medium
> BDSA BDSA-2013-0028 (CVE-2013-1966)	Oct 9, 2018	6.2 Medium
> BDSA BDSA-2017-0367 (CVE-2017-9791)	Aug 9, 2017	6.2 Medium
> BDSA BDSA-2017-0031 (CVE-2017-5638)	Mar 10, 2017	6.2 Medium
> BDSA BDSA-2017-0954 (CVE-2017-12611)	Sep 11, 2017	6.2 Medium
> BDSA BDSA-2013-0052 (CVE-2013-2115)	Aug 14, 2019	6.2 Medium
> BDSA BDSA-2020-2097 (CVE-2019-0230)	Aug 18, 2020	5.9 Medium

This tab contains the following information:

Column	Description
Identifier	The identifier and value associated with this vulnerability. Select > in the table next to the vulnerability to view a brief description. Depending on the identifier, select to view the BDSA record or the CVE record .
Published	Date on which the vulnerability was published.
Overall Score	<p>Shows the Temporal score (for BDSA), or Base score (for NVD) and associated risk level. Hover over the Overall Score value to see the individual values.</p> <ul style="list-style-type: none"> For BDSA, the Temporal, Base, Exploitability, and Impact scores are shown. For NVD, the Base, Exploitability, and Impact scores are shown. <p>The Temporal score represents time-dependent qualities of a vulnerability, taking into account the confirmation of the technical details of a vulnerability, the existence of any patches or workarounds, and the availability of exploit code or techniques.</p> <p>The Base score reflects the overall basic characteristics of a vulnerability that are constant over time and user environments:</p> <ul style="list-style-type: none"> Access Vector (AV) - CVSS v2 / Attack Vector (AV) - CVSS v3.x Access Complexity (AC) - CVSS v2 / Attack Complexity (AC) - CVSS v3.x Authentication (Au) Integrity (I) Availability (A) Confidentiality (C) <p>Note: The Authentication value is not available for CVSS v3.x scores.</p> <p>The Exploitability score measures how the vulnerability is accessed and if extra conditions are required to exploit it, taking into account access vector, complexity, and authentication.</p>

Column	Description
	The Impact score reflects the possible impact of successfully exploiting the vulnerability, considering the integrity, availability, and confidentiality impacts.

The **Cryptography** tab shows information on component versions that have encryption algorithms. Click [here](#) for more information.

The **Copyrights** tab shows the copyright statements for this component version. Click [here](#) for more information.

The **Settings** tab shows details on this component version. Information shown here also appears on the **Details** tab.

lucene.apache.org
Apache Lucene > 0.01
 java Versions: 2500 Security Cryptography Copyrights Details **Settings**

Component Version Details
 License
 Custom Fields

Settings

Version *
 0.01

Release Date
 09/16/2021

Notes

Approval Status
 Unreviewed

Save

Users with the Component Manager [role](#) can use the **Settings** tab to edit information for this KB component version.

- Select **Component Details** to edit the release date, notes, and status for this KB component version.
- Select **License** to [modify the existing license or add a new license or group](#).
- Select **Custom Fields** to edit any [custom](#) values or properties set by the Custom Fields Administrator.

Click [here](#) for information on editing component information and [here](#) for information on modifying a component version's status.

Users with the System Administrator role can use the **Settings** tab to edit the component version [custom field information](#), as shown in the **Additional Fields** section.


Modifying KB components

Users with the Component Manager [role](#) can modify the information shown for a Black Duck KB component or component version.

The revised information will appear in your current BOMs and in any future BOMs that contain this component/component version. Note that local edits to a component in a BOM made by a user, such as the BOM Manager, to a BOM supersede the edits to the component/component version made by the Component Manager.

To modify a KB component or component version:

1. Add the component and/or component version to Component Management.
2. Modify the KB component or component version.

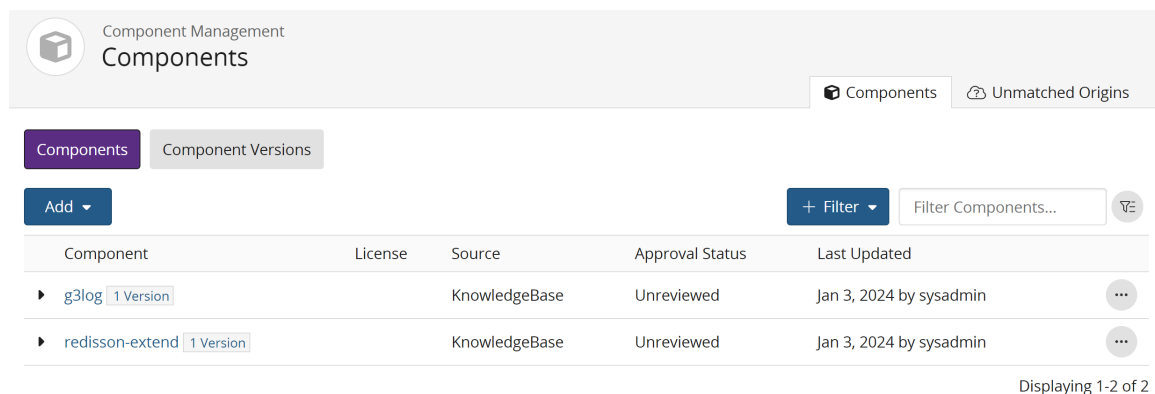
 **Note:** Setting the [status](#) of a KB component and all versions listed in the Component Management table to **Unreviewed** removes the KB component and its versions from the Component Management table. Note that this does not apply to those KB components and versions shown with a source of **Modified KnowledgeBase**.

To add a KB component or component version to the Component Management table:

1. Log in to Black Duck with the Component Manager [role](#).
- 2.

Click  > **Components**.

The **Components** tab appears.



The screenshot shows the 'Component Management' interface with the 'Components' tab selected. The table below lists the components:

Component	License	Source	Approval Status	Last Updated
▶ g3log 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin
▶ redisson-extend 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin

Displaying 1-2 of 2

3. Select **Add > Add a KnowledgeBase component** to open the Add Component dialog box.
4. Select the KB component and if adding a component version, select a version.
5. Select a status for this component.

The unreviewed status is not available when adding a KnowledgeBase component.

6. Click **Save**.

The component appears in the **Components** tab with **KnowledgeBase** as the Source.

To add additional versions, repeat this process, selecting the component and versions from the Add Component dialog box.

To modify a KB component:


1. Log in to Black Duck with the Component Manager [role](#).
- 2.

Click  > **Components**.

The **Components** tab appears.

3. Select the KB component you wish to modify.

The **Overview** tab for the *Component Name* page appears.

 **Note:** You can also display the **Overview** tab by searching for the component and selecting to view it from the search results.

4. Select the **Settings** tab.
5. Modify the information and click **Save**.

The Source for this component is now **Modified KnowledgeBase**.

To modify a KB component version:

1. Log in to Black Duck with the Component Manager [role](#).

- 2.



Click **Manage** > **Components**.

The **Components** tab appears.

3. Select the KB component version you wish to modify. Select the version from the **Component Versions** tab or in the **Components** tab, select > next to the KB component name to display the versions.

The **Details** tab for the *Component Name > Version* page appears.

4. Select the **Settings** tab.
 - Select **Component Details** to edit the release date, notes, and status for this KB component version.
 - Select **License** to [modify the existing license or add a new license or group](#).
5. Modify the information and click **Save**.

If you modified the license or release date, the Source for this component version is now **Modified KnowledgeBase**.

Resetting a Black Duck KB component's values

If you have modified the values of a Black Duck KB component or component version, you can undo those changes and reset the KB data back to its original values.

Resetting a KB component to its original values does not change the status of the component.

 **Note:** Resetting the component or component version removes all modifications.

To reset a KB component:

1. Log in to Black Duck with the Component Manager [role](#).

- 2.




Click **Manage** > **Components**.

The **Components** tab appears.

3. Do one of the following:
 - Use the *Component Name* page to reset the component:
 - a. Select the KB component you wish to reset.

The *Component Name* page appears.

- b. Select the **Settings** tab to view the component details.
- c. In the **Reset Component** section, click **Reset Component** to open the Reset Component dialog box.
- d. Click **Reset** to confirm.
- Use the **Components** tab in Component Management to reset the component:
 - a. Click  in the row of the KB component you wish to reset.
 - b. Select **Restore** to open the Reset Component dialog box.
 - c. Click **Reset** to confirm.

In the table on the **Components** tab, the source for this component reverts from **Modified KnowledgeBase** back to **KnowledgeBase**.

To reset a KB component version:

1. Log in to Black Duck with the Component Manager [role](#).

2.




Click **Manage** > **Components**.

The **Components** tab appears.

3. Select the **Component Versions** tab.

4. Do one of the following:

- Use the *Component Name > Version* page to reset the component version:
 - a. Select the KB component version you wish to reset.
The *Component Name > Version* page appears.
 - b. Select the **Settings** tab to view the component version details.
 - c. In the **Reset Component Version** section, click **Reset Version** to open the Reset Component Version dialog box.
 - d. Click **Reset** to confirm.
- Use the **Component Versions** tab in Component Management to reset the component:
 - a. Click  in the row of the KB component version you wish to reset.
 - b. Select **Restore** to open the Reset Component dialog box.
 - c. Click **Reset** to confirm.

In the table on the **Component Versions** tab, the source for this component version reverts from **Modified KnowledgeBase** back to **KnowledgeBase**.


About the KnowledgeBase Feedback Service

To improve and refine Black Duck KnowledgeBase (KB) capabilities, a feedback service has been instituted.

If you are discovering that the KB has incorrectly matched or missed matches, this service provides you with a way to send this information back to Black Duck KB. Feedback is sent when you make BOM adjustments to the component, version, origin, origin ID, or license of a match made by the KB. Feedback is also sent if

you identify unmatched files to a component; it is not sent on manually added components that do not have files associated with them.

The Black Duck KB will use the feedback to improve the accuracy of future matches. This information also helps us to prioritize our resources so that we take a closer look at the components that are important to our customers.

 **Note:** No customer-identifiable information is transmitted to the KB.

Setting or modify a component's status

You may want to approve versions or restrict usage in your BOM to approved Black Duck KB or custom components and/or component versions.

Users with the Component Manager [role](#) can set a review/approval status on the component or component version at the global level and then use that status in policy rules.

For example, to ensure that only approved components are included in your BOM:

1. Determine the components (from Black Duck KB and custom components) that are approved for your BOMs.
2. Set the status for each of these components and/or component versions to "Approved".
3. [Create policy rules](#) such that any component or component version that does not have an "Approved" status triggers a policy violation.

Policy violations appear in your BOM for all components that do not have an approved status.

Changing the status of components and/or versions

- For KB components, you set the initial status of a KB component and/or component version when you added it to Component Management.

The unreviewed status is not available for KB components.

- By default, a custom component/custom component version has a status of "Unreviewed".

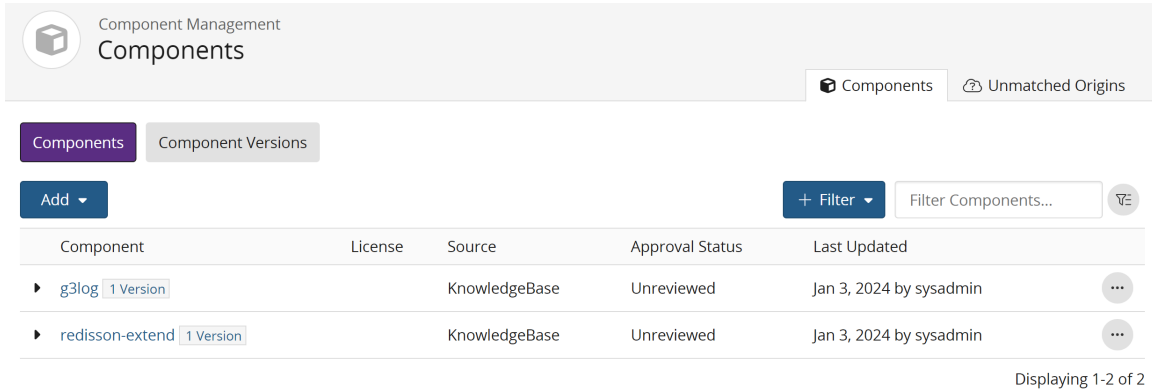
Note that the status of a component is independent of the status of its versions.

To modify the status for a component:

1. Log in to Black Duck with the Component Manager role.
- 2.

Click  > **Components**.

The **Components** tab appears.



Component Management
Components

Components Unmatched Origins


Components Component Versions

Add + Filter Filter Components...

Component	License	Source	Approval Status	Last Updated
▶ g3log 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin
▶ redisson-extend 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin

Displaying 1-2 of 2

3. Do one of the following:

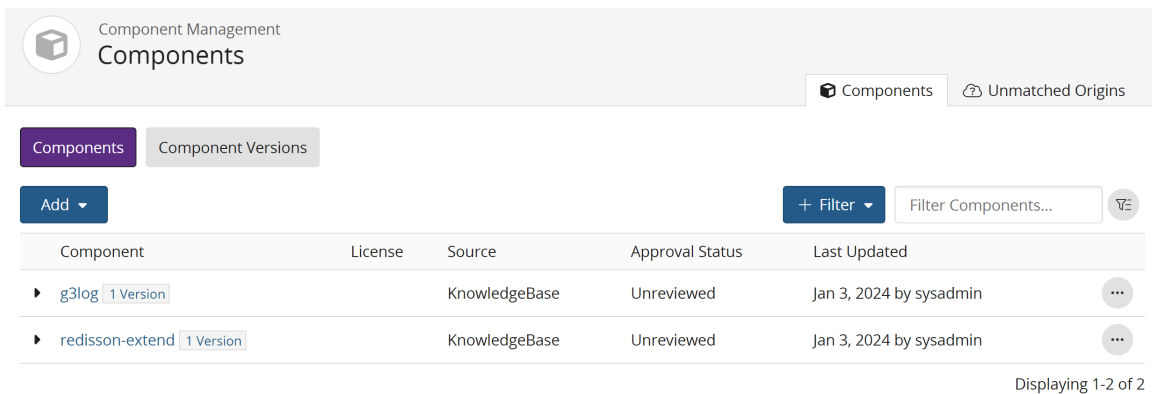
- Click  in the row of the component that you want to change the status and select a status from the list.
- Modify the status using the **Settings** tab in the *Component Name* page:
 - Select the component you wish to modify from the **Components** tab.
The **Overview** tab of the *Component Name* page appears.
 - Select the **Settings** tab.
 - Select a status from the **Approval Status** list and click **Save**.

To modify the status for a component version:

1. Log in to Black Duck with the Component Manager role.
- 2.

Click  > **Components**.

The **Components** tab appears.



Component Management
Components

Components Unmatched Origins

Components Component Versions

Add + Filter Filter Components...

Component	License	Source	Approval Status	Last Updated
▶ g3log 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin
▶ redisson-extend 1 Version		KnowledgeBase	Unreviewed	Jan 3, 2024 by sysadmin

Displaying 1-2 of 2


3. Select the **Component Versions** tab.

8. Managing components • Setting or modify a component's status

The screenshot shows the 'Component Management' interface. At the top, there is a header with a cube icon, the text 'Component Management', and 'Components'. Below this, there are two tabs: 'Components' and 'Unmatched Origins'. Underneath, there are two sub-tabs: 'Components' and 'Component Versions', with 'Component Versions' being the active tab. On the left, there is an 'Add' button with a dropdown arrow. On the right, there is a '+ Filter' button, a search box labeled 'Filter Components...', and a search icon. Below these elements is a table with the following columns: 'Component Version', 'License', 'Source', 'Approval Status', and 'Last Updated'. The table contains two rows of data. The first row is for 'g3log - v1.1' with a 'Public Domain' license, 'KnowledgeBase' source, 'Approved' status, and 'Jan 3, 2024 by System Administrator' last updated. The second row is for 'redisson-extend - 2.15.0' with a '3dsview License', 'Modified KnowledgeBase' source, 'Deprecated' status, and 'Jan 3, 2024 by System Administrator' last updated. Each row has a three-dot menu icon on the right. At the bottom right of the table area, it says 'Displaying 1-2 of 2'.

Component Version	License	Source	Approval Status	Last Updated
g3log - v1.1	Public Domain	KnowledgeBase	Approved	Jan 3, 2024 by System Administrator
redisson-extend - 2.15.0	3dsview License	Modified KnowledgeBase	Deprecated	Jan 3, 2024 by System Administrator

4. Do one of the following:

- Click  in the row of the component version that you want to change the status and select a status from the list.
- Modify the status using the **Settings** tab in the *Component Name > Version* page:
 - a. Select the component version you wish to modify from the **Component Versions** tab. The **Overview** tab of the *Component Name > Version* page appears.
 - b. Select the **Settings** tab.
 - c. Select a status from the **Approval Status** list and click **Save**.

9. Viewing risk in Black Duck

Black Duck helps you understand the type and severity of risks, at several levels of detail, across your projects. The data used to calculate risk is provided by Black Duck KB.


Use the following pages to identify and manage risk in projects:

- Dashboard pages
- Project version page/**Components** tab
- Project version page/**Security** tab

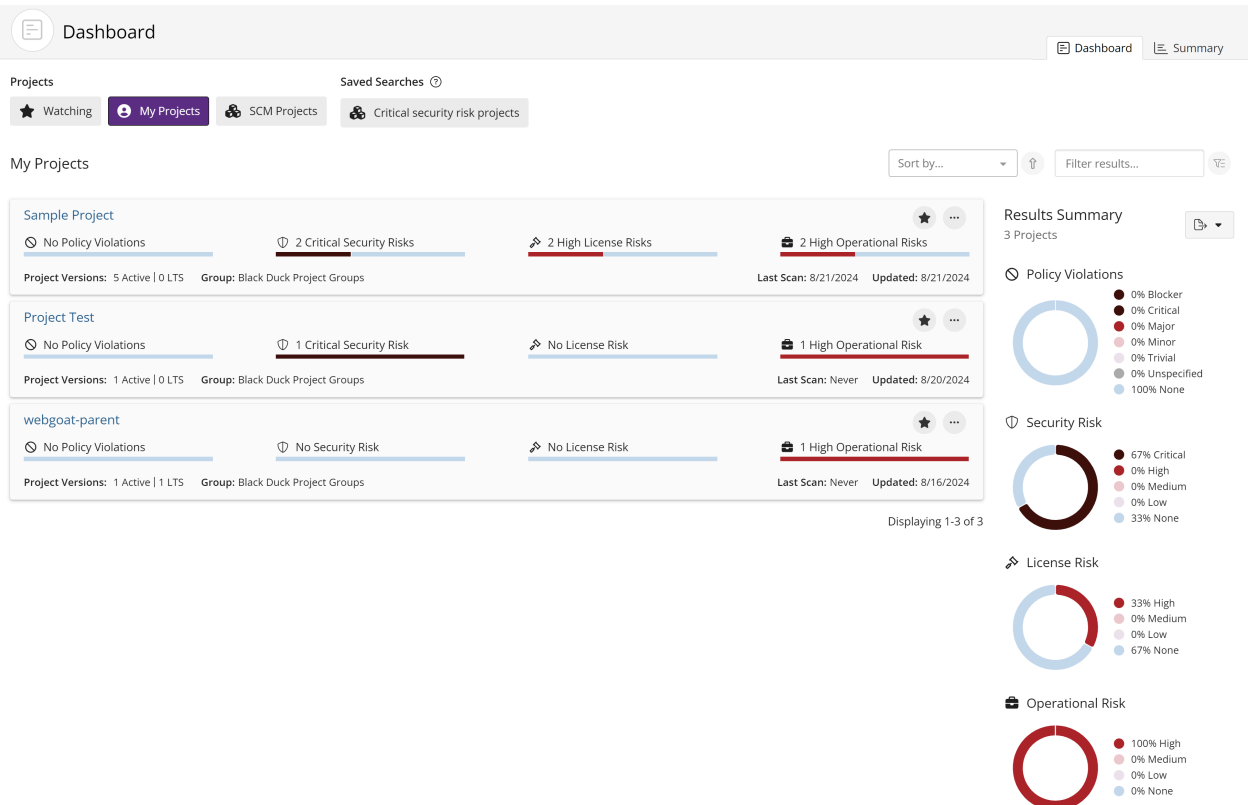
Note that the security risk values shown use CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default, CVSS v3.x scores are shown. Note that the security risk graph displays a Critical risk category with a value of 0, if you selected CVSS v2.

Dashboards

Dashboards provide a high-level overview of risk from different perspectives.

 **Note:** Dashboards will not contain any project or component information until you [create projects](#) and then [map scans](#) to these projects or [manually add components](#) to BOMs. The risk information for the components in your project versions' BOMs will then appear on the Dashboard pages.

- You can view the projects that interest you by using the **Watching** or **My Projects** dashboard or create a custom dashboard by [saving your project search results](#).



The screenshot displays the Black Duck Dashboard interface. At the top, there's a navigation bar with 'Dashboard' and 'Summary' tabs. Below this, there are sections for 'Projects' and 'Saved Searches'. The 'My Projects' section is active, showing a list of three projects: 'Sample Project', 'Project Test', and 'webgoat-parent'. Each project card displays risk metrics for Policy Violations, Security Risks, License Risks, and Operational Risks. For example, 'Sample Project' has 2 Critical Security Risks and 2 High Operational Risks. To the right, a 'Results Summary' section provides a high-level overview of risk across 3 projects, with four donut charts for Policy Violations, Security Risk, License Risk, and Operational Risk. Each chart includes a legend with risk levels and their percentages.

Project	Policy Violations	Security Risks	License Risks	Operational Risks
Sample Project	No Policy Violations	2 Critical Security Risks	2 High License Risks	2 High Operational Risks
Project Test	No Policy Violations	1 Critical Security Risk	No License Risk	1 High Operational Risk
webgoat-parent	No Policy Violations	No Security Risk	No License Risk	1 High Operational Risk

Results Summary (3 Projects)

- Policy Violations:** 0% Blocker, 0% Critical, 0% Major, 0% Minor, 0% Trivial, 0% Unspecified, 100% None
- Security Risk:** 67% Critical, 0% High, 0% Medium, 0% Low, 33% None
- License Risk:** 33% High, 0% Medium, 0% Low, 67% None
- Operational Risk:** 100% High, 0% Medium, 0% Low, 0% None

9. Viewing risk in Black Duck • Dashboards

Create a saved [component search](#) to view the components that interest you that are used in one or more projects.

Component - Apache Commons

Component	Used By	License Risk	High	Medium	Low	None
Apache Commons BeanUtils > 1.9.4	1 Project Version	No License Risk	0	0	0	0
Apache Commons Codec > 1.13	1 Project Version	No License Risk	0	0	0	0
Apache Commons Codec > 1.15	2 Project Versions	No License Risk	0	0	0	0
Apache Commons Collections > 3.2.2	2 Project Versions	No License Risk	0	0	0	0
Apache Commons Collections > 4.1	2 Project Versions	No License Risk	0	0	0	0
Apache Commons Collections > 4.4	1 Project Version	No License Risk	0	0	0	0

Results Summary
42 Components
Results updated as Jun 12, 2024, 7:42 AM

Security Risk

Risk Level	Percentage
Critical	1%
High	3%
Medium	4%
Low	0%
None	92%

License Risk

Risk Level	Percentage
High	17%
Medium	12%
Low	0%
None	71%

Operational Risk

Risk Level	Percentage
High	29%
Medium	21%
Low	15%
None	35%

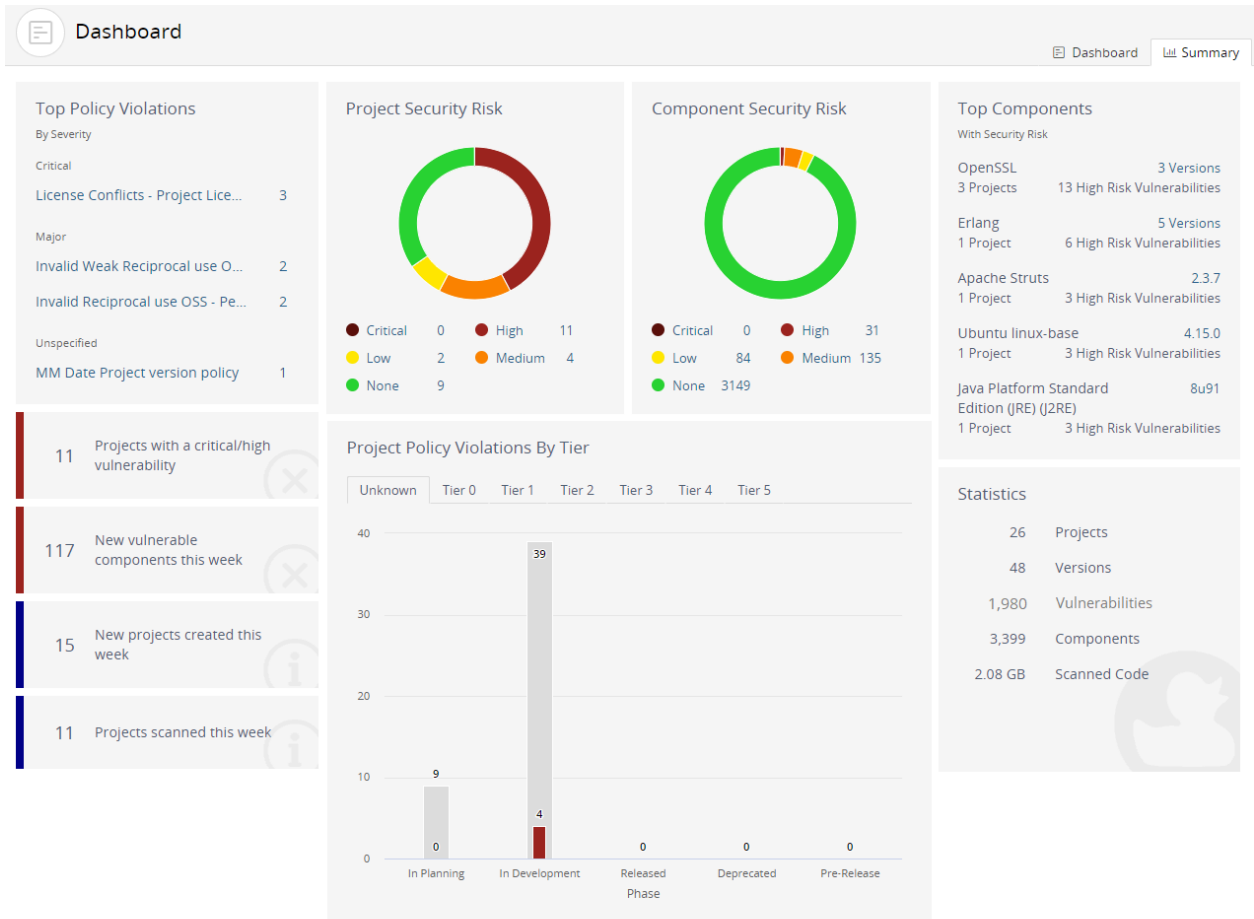
- Create a saved [vulnerability search](#) to view the vulnerabilities that interest you.

Vulnerability - Critical

Vulnerability ID	Overall Risk	Solution	Workaround	Exploit
BDSA-2015-0068	9.2 Critical	✓ Solution	No Workaround	Exploit CWE-79
BDSA-2015-0080	9.2 Critical	✓ Solution	No Workaround	Exploit CWE-79
BDSA-2010-0003	9.1 Critical	No Solution	No Workaround	No Exploit CWE-20, CWE-712
BDSA-2011-0017	9.2 Critical	✓ Solution	No Workaround	Exploit CWE-80

Results Summary
24,562 Vulnerabilities
Results updated as Jun 12, 2024, 7:42 AM


- Use the [Summary Dashboard](#) to view the overall health of the projects you have permission to view and identify areas of concern.



Note:

- The Dashboard page that appears when you log in depends on the last main dashboard (Dashboard or Summary) you viewed prior to previously logging out.

-

Click  or the logo in the upper left corner of the navigation bar to view the last dashboard (Dashboard or Summary) you viewed.

Project version pages

- Use the [project version page/Components tab](#), also known as the project version BOM, to view the components, specific to that project version, that have security, license, and operational risk.

9. Viewing risk in Black Duck • Viewing the health of your projects

Black Duck Projects
Sample Project ▾ 1.0
Project ☆ | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security </> Source Reports Details Legal Settings

Security Risk
Number of Components

Critical 2
High 2
Medium 0
Low 0
None 10

License Risk
Number of Components

High 8
Medium 1
Low 0
None 7

Operational Risk
Number of Components

High 10
Medium 0
Low 0
None 4

Add Compare to... Print... Select all Bulk Actions Filter components... Add Filter

Component	Source	Match Type	Usage	License	Security Risk	Operational Risk
Apache Commons FileUpload 1.1	1 Match	Direct Dependency	Dynamically Linked	Apache-2.0	1 1 3	High
Apache log4j 0.9.7	1 Match	Direct Dependency	Dynamically Linked	Apache-2.0		High
Apache log4j 1.0.4	1 Match	Direct Dependency	Dynamically Linked	Apache-1.1	1	High
Apache Lucene 1.4.3	1 Match	Direct Dependency	Dynamically Linked	Apache-2.0		High
Apache-jakarta jmeter 2.0.3	1 Match	Direct Dependency	Dynamically Linked	Apache-2.0	3	High
Apache-jakarta jmeter 2.1.1	1 Match	Direct Dependency	Dynamically Linked	Apache-2.0	3	High
Bit5Blog Bit 5 Blog 6.0	1 Match	Direct Dependency	Dynamically Linked	GPL-2.0+		High

- Use the [project version page/ Security tab](#) to view the security vulnerabilities of each severity associated with the components used in a project version.

Black Duck Projects
Sample Project ▾ 1.0
Project ☆ | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security </> Source Reports Details Settings

Security Risk
Number of Unique Component Origins

Critical 0 High 2 Medium 15 Low 8

Filter components... Add Filter

Hibernate Validator 5.2.4.Final
maven: org.hibernate:hibernate-validator:5.2.4.Final

Short Term Upgrade Recommendation
5.4.3.Final
2 Known Vulnerability
Vulnerabilities

Long Term Upgrade Recommendation
6.1.2.Final
Has no known vulnerabilities

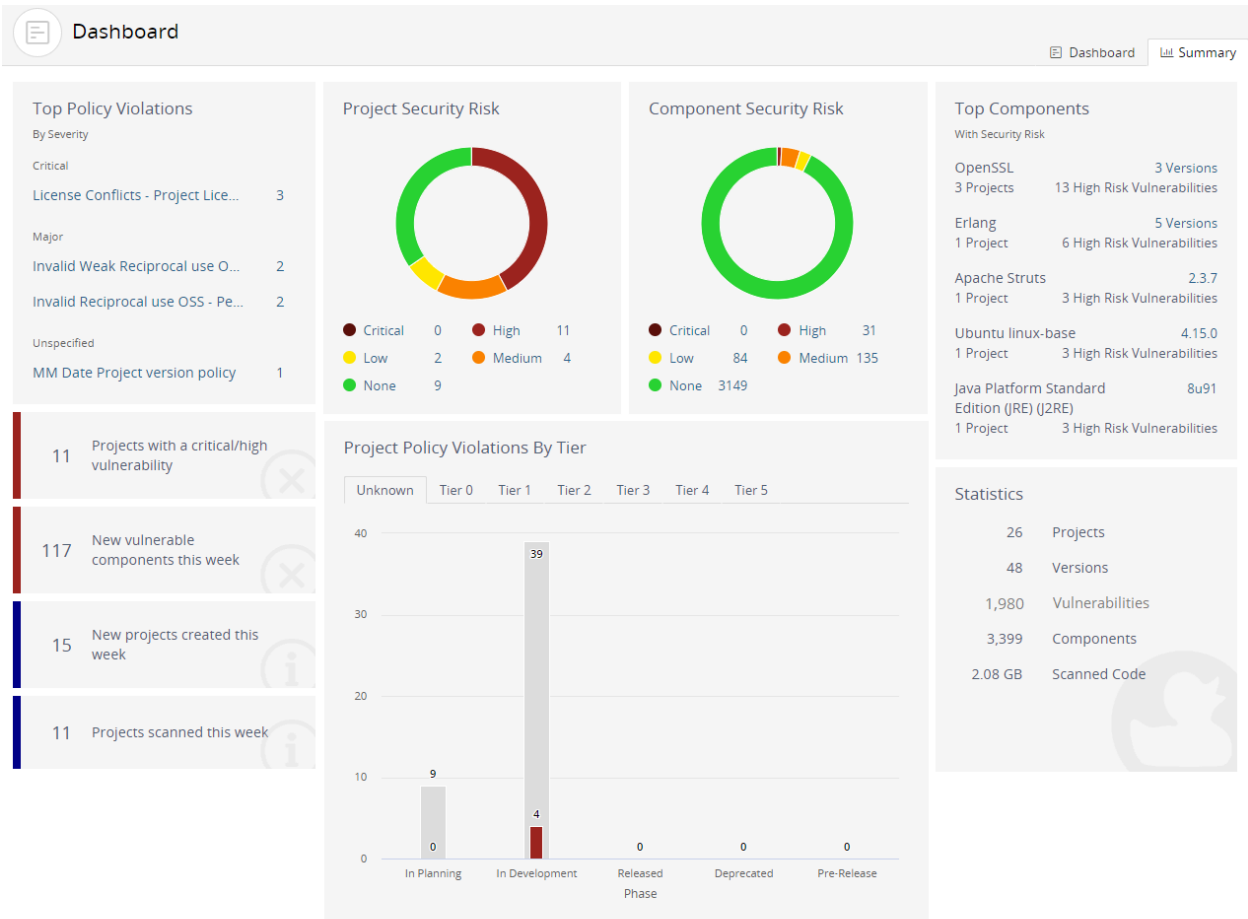
Identifier	Published	Overall Score	Status	CWE	Exploit	Workaround	Solution
BDSA BDSA-2019-3481	Nov 13, 2019	3.2 Low	New	CWE-79	-	✓	✓
NVD CVE-2017-7536	Jan 10, 2018	4.4 Medium	New	CWE-470	-	-	-

Displaying 1-2 of 2

Capture screenshot

Viewing the health of your projects

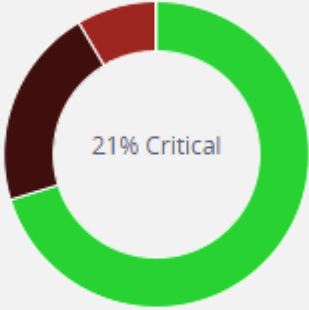
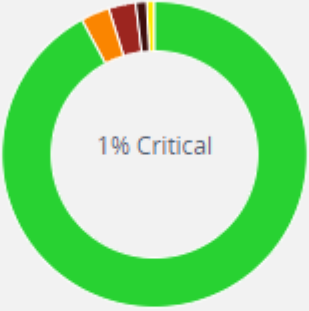
Use the **Summary** tab to view the overall health of your projects and identify areas of concern. The page consists of widgets that provide business critical information which you can use to quickly assess areas where you need to focus your attention.



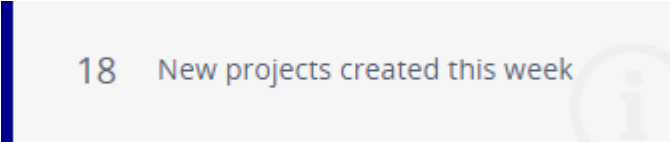
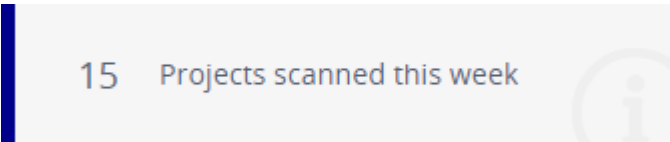
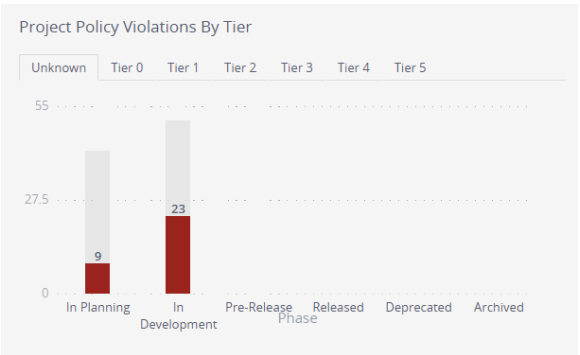
Note: The **Summary** tab only displays information for the projects you have permission to view.

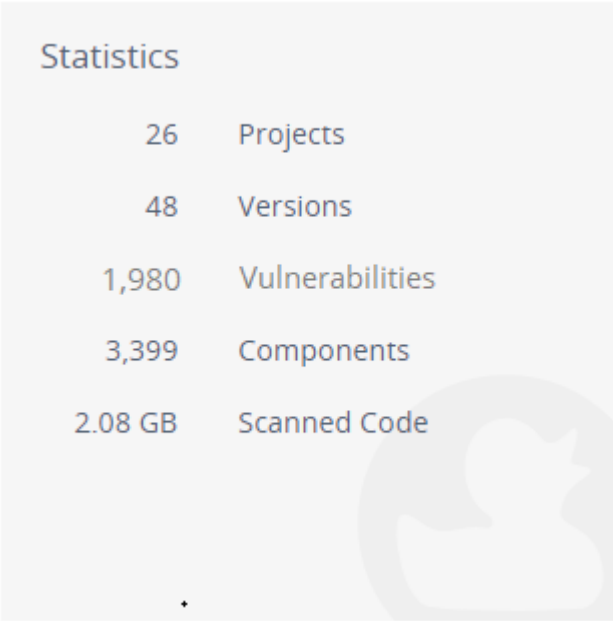
The following table describes each widget shown on the **Summary** tab and, where available, how to view additional information. Note that the security risk values shown use CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default CVSS v3.x scores are shown. Note that the graphs display a Critical risk category with a value of 0, if you selected CVSS v2.

Widget	Description	More Information
<p>Top Policy Violations</p> <p>By Severity</p> <p>Major</p> <p>High Severity 10</p> <p>Medium Severity 10</p> <p>Major Rule 10</p> <p>Vulnerability Count 10</p> <p>Sample Rule 10</p>	<p>The Top Policy Violations widget displays up to the top five policy violations across all projects that you have permission to view.</p> <p>Policy rules are listed by severity level and then by the number of policy violations, in descending order. If policy rules do not have severity levels assigned to them, the widget displays the top five policy violations, in descending order by the number of violations.</p>	<p>Select a policy rule to view the My Projects tab filtered to display the projects with a version that violates that policy rule.</p>
	<ul style="list-style-type: none"> If you do not have the Policy Management module, this widget will not appear on the page. A message appears if you have the Policy Management module but do not have any policy rules configured or have any policy violations. 	

Widget	Description	More Information												
<p data-bbox="250 289 570 327">Project Security Risk</p>  <table border="1" data-bbox="250 737 818 884"> <tr> <td>● Critical</td> <td>10</td> <td>● High</td> <td>4</td> </tr> <tr> <td>● Low</td> <td>0</td> <td>● Medium</td> <td>0</td> </tr> <tr> <td>● None</td> <td>33</td> <td></td> <td></td> </tr> </table>	● Critical	10	● High	4	● Low	0	● Medium	0	● None	33			<p>The Project Security Risk widget displays the number of projects you have permission to view for each level of security risk. Note that this widget counts the highest security risk level for a project, not all security levels affecting a project. For example, if a project has medium and low security risks, it is counted as a project with medium security risk; it is not included as a project with low security risks.</p>	<p>Hover over the graph to view the number of projects with that level of security risk.</p>
● Critical	10	● High	4											
● Low	0	● Medium	0											
● None	33													
<p data-bbox="250 978 646 1016">Component Security Risk</p>  <table border="1" data-bbox="250 1423 818 1570"> <tr> <td>● Critical</td> <td>29</td> <td>● High</td> <td>73</td> </tr> <tr> <td>● Low</td> <td>21</td> <td>● Medium</td> <td>77</td> </tr> <tr> <td>● None</td> <td>2331</td> <td></td> <td></td> </tr> </table>	● Critical	29	● High	73	● Low	21	● Medium	77	● None	2331			<p>The Component Security Risk widget displays the number of components in projects you have permission to view for each security risk level. Note that the widget counts only the highest security risk for a component. For example, if a component has medium and low security risks, it is counted as one component with a medium security risk.</p>	<p>Hover over the graph to view the number of components with that level of security risk.</p>
● Critical	29	● High	73											
● Low	21	● Medium	77											
● None	2331													

Widget	Description	More Information																				
<p>Top Components With Security Risk</p> <table border="0"> <tr> <td>GNU C Library</td> <td>2.23</td> </tr> <tr> <td>1 Project</td> <td>5 Critical Risk Vulnerabilities</td> </tr> <tr> <td>Apache Struts</td> <td>2 Versions</td> </tr> <tr> <td>2 Projects</td> <td>3 Critical Risk Vulnerabilities</td> </tr> <tr> <td>Apache Camel</td> <td>2.15.1</td> </tr> <tr> <td>1 Project</td> <td>4 Critical Risk Vulnerabilities</td> </tr> <tr> <td>GnuTLS</td> <td>3.5.16</td> </tr> <tr> <td>1 Project</td> <td>1 High Risk Vulnerabilities</td> </tr> <tr> <td>zlib</td> <td>1.1.3</td> </tr> <tr> <td>1 Project</td> <td>1 High Risk Vulnerabilities</td> </tr> </table>	GNU C Library	2.23	1 Project	5 Critical Risk Vulnerabilities	Apache Struts	2 Versions	2 Projects	3 Critical Risk Vulnerabilities	Apache Camel	2.15.1	1 Project	4 Critical Risk Vulnerabilities	GnuTLS	3.5.16	1 Project	1 High Risk Vulnerabilities	zlib	1.1.3	1 Project	1 High Risk Vulnerabilities	<p>The Top Components with Security Risk widget displays up to the top five components used in the projects you have permission to view. The information shown for each component is:</p> <ul style="list-style-type: none"> • Component name and number of versions used in your projects. If only one version is used, the specific version is listed here. • Number of your projects that have this component. • Number of security risks in this component, with the highest security risk listed here. <p>Components are organized by security risk, with those components with the highest risk listed first.</p>	<p>Select the specific version or number of versions to view the Component Version Details page.</p>
GNU C Library	2.23																					
1 Project	5 Critical Risk Vulnerabilities																					
Apache Struts	2 Versions																					
2 Projects	3 Critical Risk Vulnerabilities																					
Apache Camel	2.15.1																					
1 Project	4 Critical Risk Vulnerabilities																					
GnuTLS	3.5.16																					
1 Project	1 High Risk Vulnerabilities																					
zlib	1.1.3																					
1 Project	1 High Risk Vulnerabilities																					
<p>30 Projects with a critical/high vulnerability</p>	<p>The Projects have a critical/high vulnerability widget displays the number of projects with versions that contain components with a critical and/or high security risk.</p>	<p>N/A.</p>																				
<p>80 New vulnerable components this week</p>	<p>The New vulnerable components this week widget displays the number of components the Black Duck KB mapped a vulnerability to in the past seven days, including today.</p>	<p>N/A.</p>																				

Widget	Description	More Information
 <p>18 New projects created this week</p>	<p>The New projects created this week widget displays the number of projects that you have permission to view that have been created in the past seven days, including today.</p>	<p>N/A.</p>
 <p>15 Projects scanned this week</p>	<p>The Projects scanned this week widget displays the number of projects with scans from the past seven days, including today.</p>	<p>N/A.</p>
 <p>Project Policy Violations By Tier</p> <p>Unknown Tier 0 Tier 1 Tier 2 Tier 3 Tier 4 Tier 5</p> <p>9 23</p> <p>In Planning In Development Pre-Release Phase Released Deprecated Archived</p>	<p>The Project Policy Violations by Tier widget displays the total number of projects by phase that have a policy violation, grouped by tiers.</p> <ul style="list-style-type: none"> If you do not use tiers for your projects, projects are grouped in a single category called Unknown. If you do not have the Policy Management module, this widget displays Projects by Tier. 	<p>For each tier, hover over a bar to see the number of projects in this phase and the number of projects in this phase with a policy violation.</p>

Widget	Description	More Information
 <p>The screenshot shows a 'Statistics' widget with a light blue background. It lists five categories: Projects (26), Versions (48), Vulnerabilities (1,980), Components (3,399), and Scanned Code (2.08 GB). A small '+' icon is visible at the bottom left of the widget area.</p>	<p>The Statistics widget displays the following information:</p> <ul style="list-style-type: none"> • Projects lists the number of your projects. • Versions lists the number of project versions for your projects. • Vulnerabilities lists the number of vulnerabilities in your projects. • Components lists the number of components used in your projects, <i>including</i> ignored components. • Scanned Code lists the number of GBs scanned for all scans. 	N/A.

Viewing your dashboards

Use dashboards to view the types and severity of risk and policy violations that are associated with the components that are in one or more versions of your projects. Dashboards provide an overall view across your projects, components, and vulnerabilities.

So that you can view the projects and project versions that are important to you, Black Duck's provides two default dashboards and the ability for you to create an unlimited number of custom dashboards.

Black Duck displays these two default dashboards:

- **Watching.** Your [watched projects](#).
- **My Projects.** All of your projects, including projects that you are not watching.

These dashboards display information on the Dashboard page at the project level.

In addition, you can create custom dashboards so that you can quickly view the project versions, component versions, and vulnerabilities that are important to you: [search for projects](#), [components](#), and/or [vulnerabilities](#) and then [save the searches](#); use the Dashboard page to view the information from those saved searches.

Viewing dashboards

To view the dashboards, click

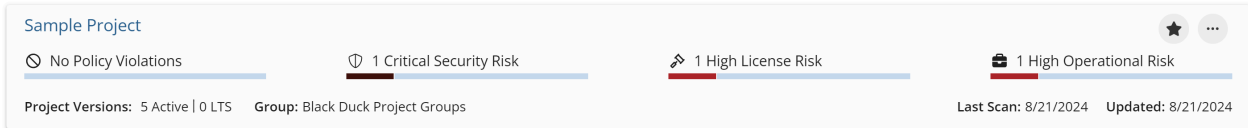


The dashboard page that appears depends on the last dashboard (a specific Dashboard page or [Summary Dashboard](#)) you viewed previously. If not displayed, select **Dashboard** to display your dashboards.

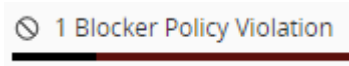
About the Watching and My Projects dashboards

Use the **Watching** or **My Projects** dashboards to view risk and policy violation information at the *project* level.

The following information is shown for each project:



- To view policy violation information for a specific project:
 - Use the bar to view the number of project versions with the highest policy severity level.



Note: The text states the number of project versions with this highest policy severity level, not all policy severity levels affecting this project.

- Hover over the bar to see the number of project versions with their highest severity level of policy violations:

Policy Violations by Project Version

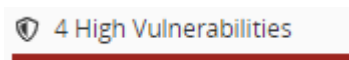
1	Blocker	0	Minor
3	Critical	0	Trivial
0	Major	0	Unspecified

* Each project version is counted once by its highest severity risk

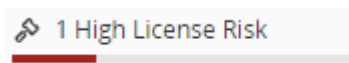
In the above example, there are four project versions which have policy violations; one version has a policy violation which has Blocker as the highest severity level, the other three versions have Critical as the highest severity level. Note that this does not indicate the number of policy violations in these versions, just the highest severity level for each version.

- To view risk information:
 - Use the risk bar to view the number of project versions with the highest risk level:


Security risk:




License risk:



Operational risk:

 4 High Operational Risks

 **Note:** The text states the number of project versions with this highest risk level, not all risk levels affecting the versions.

- Hover over a risk bar to see the number of versions of this project with their highest level of risk.

Security Risk


by Project Version

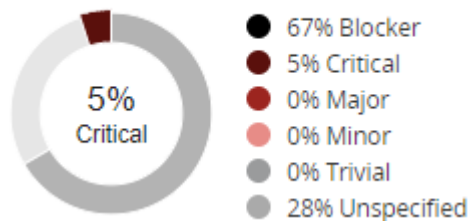


* Each project version is counted once by its highest severity risk

If a project version has risk, the version is only counted once and only its highest risk level is shown.

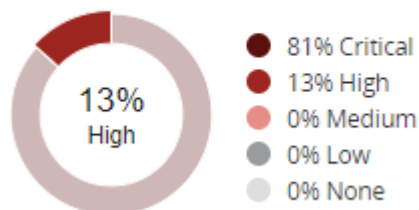
- Use the graphs to see overview information for all projects in this dashboard.
 - The risk graph shows the percentage of projects in this dashboard that have policy violations by severity level. You can also hover over an area in the graph to view this information:

 Policy Violations





- The risk graphs show the percentage of projects in this dashboard that have this level of security, license, or operational risk. You can also hover over an area in the graph to view this information:

 Security Risk



- Hover over a value in the legend to highlight the value in the graph.
- View additional information for each project, including:
 - Number of versions.
 - Last scan date.

- Date when this project was last updated, such as when a scan that was mapped to any project version was last run or when the BOM for any project version was last updated, either manually or by a new scan.
- Select a project name to view the *Project Name* page which lists all versions of this project.
- Manage how the projects are shown in these dashboards:
 - Use the **Sort by** field to select an attribute to sort by and click an arrow to select the sort order  (ascending) or  (descending).
 - Use the **Filter projects** field to filter the projects shown in either dashboard.
- Use the icons to [manage your watched projects](#) or [delete a project](#).

About saved searches dashboards

Use a saved search to view the project versions, component versions, and vulnerabilities that are important to you.

For each saved search, Black Duck lists the date and time this search was last updated.

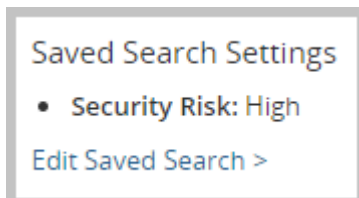
Results Summary

9 Components

Results updated at Feb 8, 2021 10:03 AM

 [Saved Search Settings](#)

Select **Saved Search Settings** to view the filters for this saved search.



Select **Edit Saved Search** to open the Find page displaying your saved search. Use the page to edit and save this revised saved search.

Project version saved searches

The following information is shown for each project version:

- located in front of the saved search name indicates that this is a project saved search.
- To view policy violation information for a specific project version:
 - Use the bar to see the number of components with the highest policy severity level for this project version.

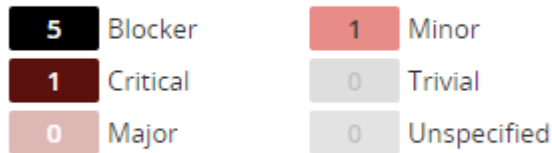
For example, the following shows that while there are components with lower severity levels, the highest policy severity level for this project version is Blocker and there are five components that have Blocker as their highest policy severity level.

Note: The text states the number of components with the highest policy severity level for this project version, not all policy severity levels affecting this project version.

- Hover over the bar to see the number of components with policy violations by the highest policy severity level:

Policy Violations

by Component

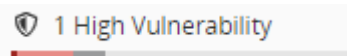


* Each component is counted once by its highest severity risk

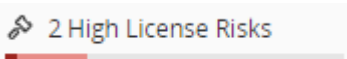
If a component has a policy violation, the component is only counted once and only its highest policy severity level is shown.

- To view risk information:
 - Use the risk bars to quickly view the number of components with the highest level of security, license, or operational risk.

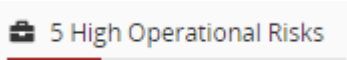
Security risk:



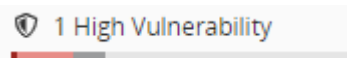
License risk:



Operational risk:



For example, the following shows that while there are components with lower risk, the highest security risk for this project version is High and that one component in this project version has a high level of security risk as their highest risk level:



- Hover over the bar to see the number of components for each risk category.


Security Risk

by Component



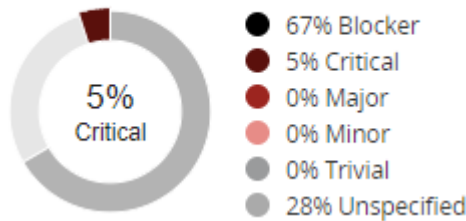
* Each component is counted once by its highest severity risk

In this example, there is one component that has a high risk level as its highest risk, 10 components that have medium risk as their highest risk level, and six components that have low risk as their highest risk level.

 **Note:** Each component is only counted once and is shown with its highest risk level.

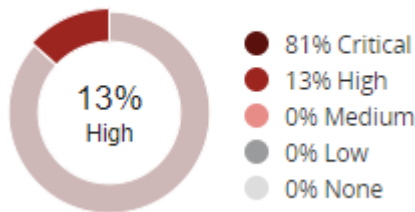
- Use the graphs to view overview information for all project versions in this dashboard categorized by policy severity and risk levels. The graphs lists the percentages for each level. You can also:
 - Hover over the graph to view the percentage of project versions with policy violations for each policy severity level.



🚫 Policy Violations



- Hover over the graph to view the percentage of project versions in this dashboard for each risk level.

🛡️ Security Risk




- Hover over a value in the legend to highlight the value in the graph.
- For each project version, the dashboard also shows:
 - Number of components in this project version.
 - Last scan date.
 - Date when this project version was last updated, such as when a scan that was mapped to this project version was last run or when the BOM for this project version was last updated, either manually or by a new scan.
 - License of this project version.
 - Phase for this project version.
 - Distribution of this project version.
- Select the project or version name to view the BOM.
- Manage how the projects are shown in these dashboards:
 - Use the **Sort by** field to select an attribute to sort by and click an arrow to select the sort order  (ascending) or  (descending).
 - Use the **Filter projects** field to filter the projects shown in the dashboard.

Component saved searches

The screenshot shows a dashboard titled 'Component - Apache Commons'. It lists several components with their respective risk levels and usage statistics. On the right side, there are three donut charts: 'Security Risk' (1% Critical, 3% High, 4% Medium, 0% Low, 92% None), 'License Risk' (17% High, 12% Medium, 0% Low, 71% None), and 'Operational Risk' (29% High, 21% Medium, 15% Low, 35% None).

The following information is shown for each component.

This detailed view for 'Apache Struts 2.3.7' shows it is used by 9 project versions and has 4 critical policy violations. The risk level is 'High'. Summary statistics indicate 0 critical, 3 high, 28 medium, and 11 low risk items. Other details include 'Approval Status: Unreviewed', 'First Detected: Never', 'Released Date: 11/6/2012', 'Newer Versions: 80', and 'Last Vuln: 10/9/2020'.

-  located in front of the saved search name indicates that this is a component saved search.
- Select the component name/version to display the [Component Name Version](#) page.
- View the number of project versions that use this component version as shown by the value next to **Used By**.

Used By | **2** Project Versions

Select **Project Versions** to open the Where Used dialog box.

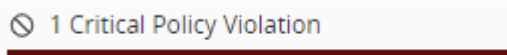
Project Name	Phase	License	Review Status	Security Risk
Sample Project - 4.0	In Planning	Apache License 2.0	Not Reviewed	0 3 6 0

Close

This dialog box shows the project versions that use this version of the component.

Column	Description
Project Name	Name of project and version that uses this component version. Select the project name to display the project version's Components tab.
Phase	Project Phase .
License	License for this component version.
Review Status	Whether this component has been reviewed in this project version.
Security Risk	Lists the vulnerabilities for each severity level, from left to right: Critical, High, Medium, and Low. <div style="display: flex; justify-content: space-around; margin: 5px 0;"> 0 3 28 11 </div> Select a value to display the Security tab of the Black Duck <i>KBComponent Name Version</i> page, which lists the vulnerabilities associated with this version of this component.

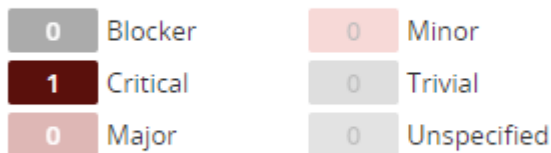
- Use the bar to quickly see the number of components with the highest policy severity level.




Select the bar to see the number of components with policy violations by severity level:

Policy Violations

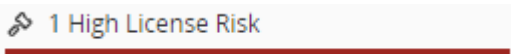
by Component



* Each component is counted once by its highest severity risk

 **Note:** A component is only counted once with the highest policy severity level, not all policy severity levels affecting this component.

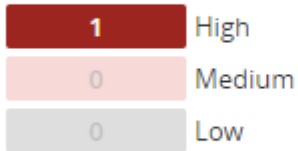
- Use the bar to quickly view the number of components with the highest level of license risk.



Select the bar to view the number of components in each risk category.

License Risk

by Component



* Each component is counted once by its highest severity risk

- View the operational risk for this component version:



- View the number of vulnerabilities by severity associated with this component version for each severity level, from left to right: Critical, High, Medium, and Low.

The **Last Vuln** date is the date when a vulnerability for this component was last updated in Black Duck (by the Black Duck KnowledgeBase or a user).



Select a value to display the **Security** tab of the Black Duck KB *Component Name Version* page, which lists the vulnerabilities associated with this version of this component.

commons.apache.org
Apache Commons Collections › 3.2.1
java Versions: 62

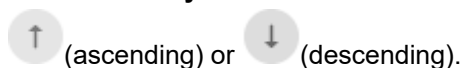
Security Cryptography Copyrights Details Settings

Filter Vulnerabilities...

Identifier	Published	Overall Score
> BDSA BDSA-2015-0001 RCE	Apr 3, 2017	8.9 High
> BDSA BDSA-2015-0753 (CVE-2015-6420) RCE	May 3, 2019	8.3 High
> BDSA BDSA-2017-2285 (CVE-2017-15708) RCE	Dec 14, 2017	5.5 Medium
> BDSA BDSA-2015-0766 RCE	Aug 6, 2019	5.5 Medium

Displaying 1-4 of 4

- For each component version, the search results also show:
 - Approval status. Status indicates whether this component version has been reviewed.
 - First detected date.
 - Date this component version was released.
 - Number of newer versions.
 - Date when a vulnerability for the component was last updated in Black Duck (by updates from Black Duck KnowledgeBase or a user manually changing the associated vulnerability and so on).
- Manage how the components are shown in these dashboards:
 - Use the **Sort by** field to select an attribute to sort by and click an arrow to select the sort order



9. Viewing risk in Black Duck • Viewing your dashboards

- Use the filter field to filter the components shown in the dashboard.

Vulnerability saved searches

The screenshot shows a dashboard with a search bar and navigation tabs. The 'Vulnerability - Critical' filter is active. A table lists four vulnerabilities:

BDSA ID	Overall Risk	Criticality	Solution	Workaround	Exploit
BDSA-2015-0068	9.2	Critical	✓	No Workaround	Exploit (CWE-79)
BDSA-2015-0080	9.2	Critical	✓	No Workaround	Exploit (CWE-79)
BDSA-2010-0003	9.1	Critical	No Solution	No Workaround	No Exploit (CWE-20, CWE-712)
BDSA-2011-0017	9.2	Critical	✓	No Workaround	Exploit (CWE-80)

The following information is shown for each vulnerability:

Details for BDSA-2020-1234 (CVE-2020-13430):

- Used By: 2 Project Versions
- Overall Risk: 8.1 High
- Solution: ✓
- Workaround: ✓
- No Exploit: No Exploit (CWE-79)

- Select the vulnerability ID to view more information about the vulnerability, such as additional score values. You can view National Vulnerability Database (NVD) information by selecting the [CVE number](#) or view Black Duck Security Advisory (BDSA) information by selecting the [BDSA number](#).
- View the number of project versions that affected by this vulnerability next to **Used By**.

Used By | **2** Project Versions

Select **Project Versions** to open the **Affected Projects** tab for the vulnerability which lists the project versions affected by this vulnerability.

The screenshot shows the 'Affected Projects' tab for BDSA-2014-0126. A table lists affected projects:


Project	Component	Component Origin	Status	Target date	Actual date
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 3.1	maven/commons-httpclient:commons-httpclient:3.1	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 4.3.3	maven/org.apache.httpcomponents:httpclient:4.3.3	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.2	maven/org.apache.httpcomponents:httpcore:4.3.2	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.3	maven/org.apache.httpcomponents:httpcore:4.3.3	New	Never	Never

- View the overall risk score. The search results show the Temporal Score for BDSA vulnerabilities, or the Base Score for NVD vulnerabilities and the associated risk level. Note that the score shown and risk level depends on the [selected security rankings](#).

Select the score to view individual scores: temporal, base, exploitability, and impact for BDSA; base, exploitability, and impact for NVD.

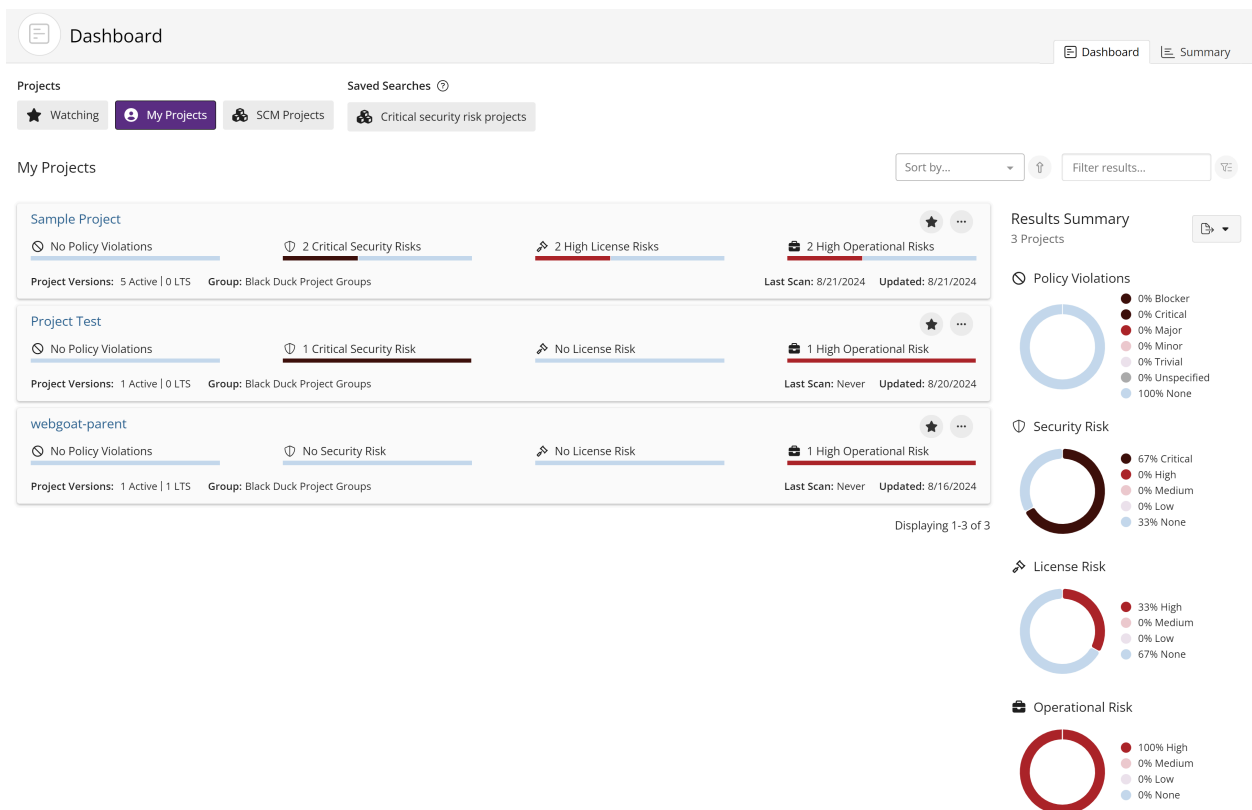
- View whether a solution, workaround, or exploit is available:
 - ✓ indicates that there is a solution or workaround available for this vulnerability.
 - ⚠ indicates there is an exploit for this vulnerability.
- For each vulnerability, the search results also show:
 - First Detected.
 - Published date.
 - Last modified date.
 - Common Weakness Enumeration (CWE) number for this security vulnerability.

Exporting to CSV

You can export your Dashboard to CSV which converts the individual rows to tabular data. To do so, click the  button and select CSV.

Viewing overall risk for all projects

The **Watching**, **My Projects**, and saved search dashboards show the overall risk across all projects where you are a project team member.



The screenshot shows a dashboard interface for viewing overall risk across all projects. The main section is titled 'My Projects' and displays three project cards:

- Sample Project:** No Policy Violations, 2 Critical Security Risks, 2 High License Risks, 2 High Operational Risks. Project Versions: 5 Active | 0 LTS. Group: Black Duck Project Groups. Last Scan: 8/21/2024, Updated: 8/21/2024.
- Project Test:** No Policy Violations, 1 Critical Security Risk, No License Risk, 1 High Operational Risk. Project Versions: 1 Active | 0 LTS. Group: Black Duck Project Groups. Last Scan: Never, Updated: 8/20/2024.
- webgoat-parent:** No Policy Violations, No Security Risk, No License Risk, 1 High Operational Risk. Project Versions: 1 Active | 1 LTS. Group: Black Duck Project Groups. Last Scan: Never, Updated: 8/16/2024.

On the right side, there are four donut charts representing risk distributions:

- Results Summary (3 Projects):** 0% Blocker, 0% Critical, 0% Major, 0% Minor, 0% Trivial, 0% Unspecified, 100% None.
- Policy Violations:** 0% Blocker, 0% Critical, 0% Major, 0% Minor, 0% Trivial, 0% Unspecified, 100% None.
- Security Risk:** 67% Critical, 0% High, 0% Medium, 0% Low, 33% None.
- License Risk:** 33% High, 0% Medium, 0% Low, 67% None.
- Operational Risk:** 100% High, 0% Medium, 0% Low, 0% None.

Click [here](#) for more information about using this page to understand security vulnerabilities associated with your projects.

Understanding the types of project risk

There are three types of risk being assessed across all projects:

- **Security Risk.** Projects can have one of four categories of security risk, based on the vulnerabilities associated with the components that comprise the project.

Vulnerabilities are linked to components by the CVE numbers, as reported in the National Vulnerabilities Database (NVD) maintained by NIST or by Black Duck Security Advisories (BDSA) numbers.

Note that the security risk values shown use CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default, CVSS v3.x scores are shown. Note that the graph displays a Critical risk category with a value of 0, if you selected CVSS v2.

Possible risk categories are:

- Critical. The project has critical severity vulnerabilities.
 - High. The project has high severity vulnerabilities.
 - Medium. The project has at least one component with at least one medium severity vulnerability.
 - Low. The project has at least one component with at least one low severity vulnerability.
 - None. All components in this project have no vulnerabilities.
- **License Risk.** Projects are assigned one of four categories of overall license risk:
 - High. The project has at least one component with a high risk license.
 - Medium. The project has at least one component with a medium risk license.
 - Low. The project has at least one component with a low risk license.
 - None. All components in this project do not have license risk.

Click [here](#) for more information on how license risk for a component is determined.

- **Operational Risk.** Operational risk is based on a combination of factors: (1) the strength of the component community, including the number of contributors and the level of commit activity; and (2) the number of newer versions of the component that are available than the one that is currently in use.

There are four categories of operational risk:

- High. The project has a version that has at least one component with high combined operational risk.
- Medium. The project has a version that has at least one component with medium combined operational risk.
- Low. The project has at least one component with low combined operational risk.
- None. All components in this project do not have operational risk.

Viewing overall risk at the project version level

Risk information for a specific project version is shown on the project version's **Components** tab.

9. Viewing risk in Black Duck • Viewing overall risk at the project version level

Black Duck Project Groups
Sample Project ▶ 1.0.0

Project ★ Phase: In Development Scans: Up to Date Status: Up to Date Last Updated: 10:21 AM

Components Security Source Reports Details Legal Settings

Security Risk
Number of Components

Critical	4
High	3
Medium	4
Low	0
None	17

License Risk
Number of Components

High	6
Medium	2
Low	0
None	20

Operational Risk
Number of Components

High	24
Medium	3
Low	0
None	1

Snippets
78 Unconfirmed
Unmatched Components
1 Unmatched

Print Add Bulk Actions Compare to... Ignore Not Ignored Snippet Match Status Confirmed Match Ignore Not Ignored Filter Filter Components...

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain		High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0		High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

Also known as the [BOM page](#), this tab shows all type of risks associated with each component in the project version's BOM.

There are three types of risk being assessed across all projects. As shown in the graphs at the top of the page:

- **Security Risk.** Risk values for project versions are based on the vulnerabilities associated with the components that comprise the project version's BOM.

Vulnerabilities are linked to components by the CVE numbers, as reported in the National Vulnerabilities Database (NVD) maintained by NIST or by Black Duck Security Advisories (BDSA) numbers.

Note that the security risk values shown use CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default, CVSS v3.x scores are shown. Note that the graph displays a Critical risk category with a value of 0, if you selected CVSS v2.

Possible risk categories are:

- Critical. The number of components in this project version with critical severity vulnerabilities.
- High. The number of components in this project version with high severity vulnerabilities.
- Medium. The number of components in this project version's BOM with medium severity vulnerabilities.
- Low. The number of components in this project version's BOM with low severity vulnerabilities.
- None. The number of components in this project version's BOM with no vulnerabilities.
- **License Risk.** Project versions can have four levels of overall license risk:
 - High. The number of components in this project version's BOM high risk license.
 - Medium. The number of components in this project version's BOM with medium risk license.
 - Low. The number of components in this project version's BOM with low risk license.
 - None. The number of components in this project version's BOM with no license risk.

Click [here](#) for more information on how license risk for a component is determined.

- **Operational Risk.** Project versions can have four levels operational risk. Operational risk is based on a combination of factors: (1) the strength of the component community, including the number of

contributors and the level of commit activity; and (2) the number of newer versions of the component that are available than the one that is currently in use.

Project versions can have four categories of operational risk:

- High. The number of components in this project version's BOM with high combined operational risk.
- Medium. The number of components in this project version's BOM with medium combined operational risk.
- Low. The number of components in this project version's BOM with low combined operational risk.
- None. The number of components in this project version's BOM with no operational risk.

You can [use the risk graphs and table filters to filter](#) the BOM to show only components that have the selected severity and type of risk.

Click [here](#) for more information about understanding the BOM page.

Understanding the types of component risk

There are three types of risk being assessed for components used in projects:

Security Risk

Components are assigned one of four categories of security risk, based on the vulnerabilities associated with the versions in use in projects.

Vulnerabilities are linked to components by the CVE numbers, as reported in the National Vulnerabilities Database (NVD) maintained by NIST or by Black Duck Security Advisories (BDSA) numbers.

Note that the security risk values shown use CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default, CVSS v3.x scores are shown.

Possible risk categories are:

- Critical. The component has critical high severity vulnerabilities.
- High. The component has high severity vulnerabilities.
- Medium. The component has medium severity vulnerabilities.
- Low. The component has low severity vulnerabilities.
- None. The component has no vulnerabilities.

License Risk

Projects are assigned one of four categories of overall license risk:

- High. At least one version of the component in use in a project has a declared license that is high risk.
- Medium. At least one version of the component in use in a project has a declared license that is medium risk.
- Low. At least one version of the component in use in a project has a declared license that is low risk.
- None. At least one version of the component in use in a project has a declared license that is no risk.

See [Determining license risk](#) for more information on how license risk for a component is determined.


Operational Risk

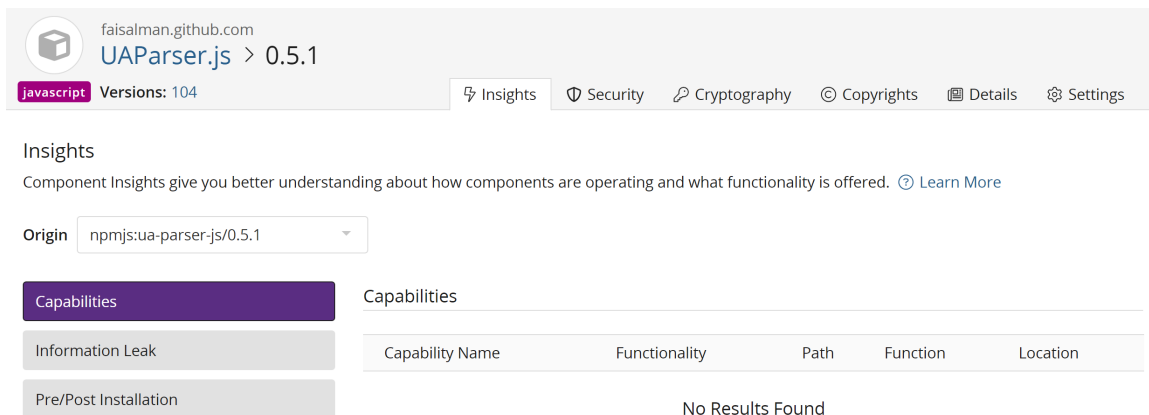
Operational risk is based on a combination of factors: (1) the strength of the component community, including the number of contributors and the level of commit activity; and (2) the number of newer versions of the component that are available, other than the one that is currently in use.

Project versions can have four categories of operational risk.

- High. The component has high combined operational risk and a version is used in at least one project.
- Medium. The component has medium combined operational risk and a version is used in at least one project.
- Low. The component has low combined operational risk and a version is used in at least one project.
- None. The component has no operational risk and a version is used in at least one project.

Additionally, if a component version's origin has been flagged with further Component Intelligence risks,

you can click the menu button found at the end of the component's row  and select Insights to view the component version's Insights page.



faismalman.github.com
UAParser.js > 0.5.1

javascript Versions: 104

Insights Security Cryptography Copyrights Details Settings

Insights

Component Insights give you better understanding about how components are operating and what functionality is offered. [Learn More](#)

Origin npmjs:ua-parser-js/0.5.1

Capabilities

Capability Name	Functionality	Path	Function	Location
No Results Found				

Information Leak

Pre/Post Installation

On this page, you can view a component version's insights:

Capabilities: Native functionality offered by the component. Capabilities currently detectable:

- Network Communications: The software component possesses the ability to communicate over the network.
- System Operation: The software component possesses the ability to execute system level commands.
- Cryptography: The software component possesses the ability to secure communications.
- Serialization: The software component possesses the ability to convert data structures to byte streams.
- File System Access: The software component possesses the ability to interact with the local file system.
- Compression: The software component possesses the ability to compress data.
- Security operations (Sanitizers & Validators): The software component possesses the ability to sanitize and/or validate input data.

Information Leak: Detected valid IPv4/6 addresses including domains identified within the codebase.

Pre/Post Installation: Various component configuration findings and observations. Observations may signify security and/or operational concerns.

10. About security risk

Black Duck helps security and development teams identify security risks across their applications.

By mapping vulnerabilities to your open source software, Black Duck can provide you with high-level overview information on security risk of your projects, along with detailed information on security vulnerabilities which you can use to investigate and remediate your security vulnerabilities.

Vulnerabilities are linked to the open source components by the Common Vulnerabilities and Exposures numbers (CVEs), as reported in the National Vulnerabilities Database (NVD) maintained by the National Institutes of Standards and Technology (NIST) and/or by (BDSA) numbers. If you have licensed Black Duck Security Advisories. Note that Black Duck displays the numbers together in reports and in the UI because they represent the same vulnerability from different sources.

Security risk levels

NVD and BDSA use the Common Vulnerability Scoring System (CVSS) which provides a numerical score reflecting the severity of a vulnerability. The numerical score is then translated into a risk level to help you assess and prioritize security vulnerabilities.

Black Duck provides you with the option of viewing CVSS v2 or CVSS v3.x scores. By default, Black Duck displays CVSS v3.x scores.

- CVSS v2 scores have the following values:
 - Low risk: 0.0 - 3.9
 - Medium risk: 4.0 - 6.9
 - High risk: 7.0-10.0

Note that Black Duck shows vulnerabilities with a 0.0 score as no risk.

Although CVSS v2 does not have a Critical risk category, the security graphs in the Black Duck UI display a Critical risk category. This category will display a value of 0 for CVSS v2.

- CVSS v3.x scores have the following values:
 - None: 0.0
 - Low risk: 0.1 - 3.9
 - Medium risk: 4.0 - 6.9
 - High risk: 7.0 - 8.9
 - Critical risk: 9.0 - 10.0


Note that the scores shown for CVSS v3.x can be v3.0 or v3.1 scores.

Estimated Security Risk

This estimated risk statistic is formulated by looking at all the versions of a component sorted by security vulnerability severity category and calculating the maximum vulnerability count for each severity category for each component version. The maximum vulnerability count for each severity category is shown in the "Estimated Security Risk by Severity Category" on the Bill of Material for Security risk. The highest severity category counts may reference different component versions. For example:

- Version 1.1 has 2 Critical, 3 High, 15 Medium, 4 Low
- Version 1.2 has 2 Critical, 4 High, 12 Medium, 1 Low
- Estimated Security Risk by severity category for components with unknown versions would return as 2 Critical, 4 High, 15 Medium, 4 Low on the BoM.

Users should choose the exact version used in the application to view the accurate risk instead of the estimated risk. This estimated risk information is provided to help prioritize what components to review first. Users are encouraged to use estimated risk information in conjunction with BD Policy Management to further prioritize what components to triage first based on their company's security policies.

 **Note:** The information presented is only a statistical data estimation. As a result, the estimated security risks will not have CVE data.

Suggested work flow

To manage security risk using Black Duck:

1. With the assistance of your security team, determine your security risk policies.
2. If necessary, users with the system administrator role can [define the default security ranking](#).

Note that the security ranking also defines how vulnerabilities appear in reports. Depending on the data available, the vulnerability will be presented as either: BDSA (NVD) or NVD (BDSA). For example, if the security ranking is NVD2, BDSA2, BDSA3, NVD3 then:

- Vulnerability A has data for just NVD3. The vulnerability is listed as NVD-1234-5678 in the report.
 - Vulnerability B has data for NVD3 and BDSA3. The report lists it as BDSA (NVD).
 - Vulnerability C has data for everything. The report lists it as NVD (BDSA).
3. [Create policies](#) that trigger violations when components do not comply with your security policies.
 4. Depending on your interests:
 - Use the [Summary Dashboard](#) to view the overall health of your projects and identify areas of concern. Use this page to quickly assess areas where you need to focus your attention.
 - Use these Dashboard pages for a high-level overview of risk:
 - [Use the Watched or My Project dashboards](#) to view the security risk across all your projects.
 - [Create saved searches](#) to customize the information shown on the Dashboard page to view the projects, components, and vulnerabilities that interest you.
 - Use these pages for project version-level information:
 - [project version page/Components tab](#), also known as the project version BOM, to view the components specific to that project version, that have security risk.
 - [project version page/ Security tab](#) to view the security vulnerabilities of each severity associated with the components used in a project version.
 5. Investigate vulnerabilities and policy violations. For detailed information on security vulnerabilities, view the:
 - [CVE page](#)
 - [BDSA page](#) if you have licensed Black Duck Security Advisories (BDSA)
 6. After reviewing the severity of the vulnerability, users with the appropriate [role](#) can [change the remediation status](#) of the security vulnerability.

7. [Monitor notifications](#) for any new security vulnerabilities.

You will receive notification alerts if security vulnerabilities are published or updated against components that are included in one or more of your projects.

Defining the default security risk calculation

Users with the system administrator role can redefine the order of security ranking that Black Duck uses to define the risk score and risk categories of security vulnerabilities. Black Duck uses the following order to calculate risk:

- If you have not licensed BDSA, the default order is:
 1. NVD v3.x
 2. NVD v2
- If you have BDSA licensed, the default order is:
 1. BDSA (CVSS v3.x)
 2. NVD (CVSS v3.x)
 3. BDSA (CVSS v2)
 4. NVD (CVSS v2)

As shown above, by default Black Duck defines security risk initially using CVSS v3.x scores. You can modify the order by which Black Duck determines security risk so that CVSS v2 scores are used.

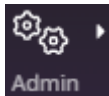
Note the following:

- Changing the order of the security risk configuration will result in revised security risk calculations for all project version BOMs and may result in new policy violations. These calculations may take a *considerable amount of time* to complete.
- The ability to change the security risk ranking is disabled if the security risk configuration has been reconfigured and jobs are running to recalculate security risk. Once the jobs are completed, the security risk ranking can be reconfigured.
- If a CVE record has a related BDSA record (or vice versa), it cannot be remediated unless that vulnerability record type is prioritized in the Security Risk Ranking below. This is due to the fact that the non-prioritized vulnerability record is not being used as a determinant and is not used to calculate security risk.

To configure the default security risk calculation:

1. Log in to Black Duck with the System Administrator role.

2.



Click **Admin**.

3. Select **System Settings**.
4. Click **Security Risk Ranking**.
5. In the **Security Risk Ranking** section, drag and drop the tiles so that the ranking is in the correct order.
6. Click **Save**.

A confirmation dialog box appears. Do one of the following:

- Click **Confirm**.

The VulnerabilitySummaryFetchJob starts once you click **Confirm**.

Refresh the page to update the status of these jobs on this page. You can also view the status on the [Jobs page](#).

Once these jobs complete, the new security rankings appear in the Black Duck UI.

- Click **Cancel**.

The security risk configuration ranking returns to its previous order.

Viewing the security vulnerabilities of your projects, project versions, and component versions

Use your [dashboards](#) to view the types and severity of risk that are associated with the components that are in one or more versions of your projects. Dashboards provide an overall view of risk across the components in your projects and project versions. Use the project version **Security** tab to view a list of vulnerabilities for each component version origin.

Related vulnerabilities

Note that BDSA-1234-6789 or CVE-1234-5678 is the ID for a single vulnerability from BDSA or NVD: there is one vulnerability, but there are two databases and each has its own set of IDs to distinguish the same vulnerability.

There can be instances when the Black Duck UI shows vulnerabilities as related and in other instances (for example a different component version origin) when the same vulnerabilities are not shown as related. This may occur as sometimes NVD or BDSA does not evaluate certain origins, components, or component versions.

For example, suppose vulnerability X is found; NVD identifies it as CVE1 and BDSA identifies it as BDSA1. NVD has also found vulnerability X in component version origin A and component version origin B but BDSA has only found it in component version origin B (BDSA has either decided NVD is incorrect or not evaluated it). If your BOM has component version origin A, the project version's **Security** tab displays just the NVD identifier (CVE1) for that component version origin. BDSA does not apply in this context because it is not linked to this component version origin. If your BOM has component version origin B, both NVD and BDSA have found that Vulnerability X applies. You will see either BDSA 1 (CVE 1) or CVE 1 (BDSA 1) depending on your security priority system settings.

If NVD does not find the exploit at all then Black Duck only lists the BDSA ID, whether it be for specific component version origins or just generally looking up the BDSA identifier in the Black Duck application.

Viewing project version vulnerabilities

Use the project version page's **Security** tab to view the security vulnerabilities associated with the components used in a project version.

The information shown uses CVSS v2 or CVSS v3.x scores, depending on which [security risk calculation you selected](#); by default CVSS v3.x scores are shown. Note that the graph displays a Critical risk category with a value of 0, if you selected CVSS v2.

10. About security risk • Viewing project version vulnerabilities

The screenshot shows the Black Duck Projects interface for a project named 'Sample Project' with the version '2021_06_15--23_27_39'. The 'Security' tab is active, displaying a 'Security Risk' section with a graph showing 0 Critical, 2 High, 19 Medium, and 10 Low vulnerabilities. Below this is a 'Components' list on the left, showing 'Apache Commons FileUpload 1.2.2' selected. The main area displays details for this component, including a 'Short Term Upgrade Recommendation' to 'commons-fileupload-1.4' and a 'Long Term Upgrade Recommendation' to 'commons-fileupload-1.4'. A table of 5 known vulnerabilities is shown, with columns for Identifier, Overall Score, Status, CWE, Exploit, Workaround, and Solution.

Identifier	Overall Score	Status	CWE	Exploit	Workaround	Solution	
> BDSA BDSA-2013-0013 RCE	6.5	Medium	New	CWE-626	✓	✓	✓
> BDSA BDSA-2016-1636 (CVE-2016-3092)	6.1	Medium	New	CWE-400	✓	✓	✓
> BDSA BDSA-2016-1573 (CVE-2016-100031) RCE	5.9	Medium	New	CWE-502	✓	-	✓
> BDSA BDSA-2014-0102 (CVE-2014-0050)	3.9	Low	New	CWE-20, CWE-400	✓	-	✓
> BDSA BDSA-2013-0001 (CVE-2013-0248)	2.4	Low	New	CWE-367	-	-	✓


This page has these sections:


- Security Risk graph.
- Components list.
- Filters.
- Remediation guidance section, shown above the vulnerabilities table. Click [here](#) for more information about this feature.
- Vulnerabilities table.

Security Risk graph

The Security Risk graph shows how many vulnerabilities of each severity for each component version and subproject used in this version of the project.

The Security Risk graph shows the number of components with vulnerabilities for each severity level.

 **Note:** This graph lists the number of components which have this level of security risk as their *highest* risk level – it is not the total number of components which have this risk level. For example, if you select to view components with a medium risk level, only those components that have medium as the highest risk level appear in the table; components that have both high *and* medium vulnerabilities are not shown.

 **Note:** The number of components with vulnerabilities shown here may not be the same value as shown in your project version BOM (**Components** tab). In the BOM, the security graph aggregates similar components with different origins. On this page, the graph displays security risk by unique component origins, as a vulnerability may be origin-specific.

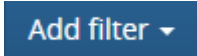
Select a severity level in the Security Risks graph to view all components that share the same level of risk.

Components list

This section lists each component with vulnerabilities. For each component, the component name, component version, and origin are shown along with risk bars that list how many vulnerabilities of each severity exist in this component version or subproject.

Select the component to display its vulnerabilities in the vulnerabilities table. To view vulnerabilities for a subproject, if you have permission to view this project, select the subproject name in the component list, then select the link shown on the page which displays the vulnerabilities for the subproject.

Filters

Use the **Filter components** field to view specific components. Click  to view other available filters.

- Some filter options apply to the values shown in the vulnerabilities table. If you select those filter options, components that have at least one vulnerability with the specified filter value will appear on the page.
- Filters filter the list of components shown on the left side of the page. However, the data shown in the vulnerability table for those components is not filtered.

For example, if you select to view those components that have vulnerabilities with an overall score greater than 9.0, the page displays the list of components that have at least one vulnerability with an overall score greater than 9.0. The information shown in the vulnerability table for those components is not filtered: it still shows all vulnerabilities for the filtered components, including those vulnerabilities with an overall score less than 9.0.

Vulnerabilities table

Initially, the vulnerabilities table shows the vulnerabilities of the first component in the Components list. Select a component to display its vulnerabilities.

The vulnerabilities table lists the following information for each vulnerability:

Column	Description
Identifier	<p>The identifier, value associated with this vulnerability, and any vulnerability tags (if applicable).</p> <p>Select > in the table next to the vulnerability to view a brief description. Depending on the identifier, select to view the BDSA record and/or the CVE record.</p> <p>Users with the appropriate role can also use this section to remediate the vulnerability.</p>
Overall Score	<p>Shows the Temporal score (for BDSA), or Base score (for NVD) and associated risk level. Hover over the Overall Score value to see the individual values.</p> <ul style="list-style-type: none"> • For BDSA, the Temporal, Base, Exploitability, and Impact scores are shown. • For NVD, the Base, Exploitability, and Impact scores are shown. <p>The Temporal score represents time-dependent qualities of a vulnerability taking into account the confirmation of the technical details of a vulnerability, the existence of any patches or workarounds, and the availability of exploit code or techniques.</p> <p>The Base score reflects the overall basic characteristics of a vulnerability that are constant over time and user environments:</p> <ul style="list-style-type: none"> • Access Vector (AV) - CVSS v2 / Attack Vector (AV) - CVSS v3.x • Access Complexity (AC) - CVSS v2 / Attack Complexity (AC) - CVSS v3.x • Authentication (Au) • Integrity (I) • Availability (A) • Confidentiality (C) <p>Note: The Authentication value is not available for CVSS v3.x scores.</p>

Column	Description
	The Exploitability score measures how the vulnerability is accessed and if extra conditions are required to exploit it, taking into account access vector, complexity, and authentication. The Impact score reflects the possible impact of successfully exploiting the vulnerability, considering the integrity, availability, and confidentiality impacts.
Status	Remediation status of this vulnerability. Possible values are: Duplicate, Ignored, Needs Review, New, Mitigated, Patched, Remediation Complete, or Remediation Required.
CWE	Common Weakness Enumeration (CWE) number for this security vulnerability. – indicates a CWE number is not available.
Exploit	Indicates whether an exploit for this vulnerability is available: <ul style="list-style-type: none"> – No exploit available ✓ Exploit available
Workaround	Indicates whether a workaround for this vulnerability is available: <ul style="list-style-type: none"> – No workaround available ✓ Workaround available
Solution	Indicates whether a solution for this vulnerability is available: <ul style="list-style-type: none"> – No solution available ✓ Solution available

Direct match upgrade recommendations

The simplest way to minimize or resolve security risk is to upgrade the version of the used component with fewer vulnerabilities. It is easier to do for components used as direct match.

The screenshot displays the 'Upgrade Recommendation' section for the component 'JUnit - 4.11'. It shows that there is 1 known vulnerability and 1 direct dependency match. Two upgrade paths are offered: a 'Short-Term' recommendation to upgrade to JUnit 4.13.2, and a 'Long-Term' recommendation to upgrade to JUnit 4.13.2. Both recommended versions are noted as having no known vulnerabilities.

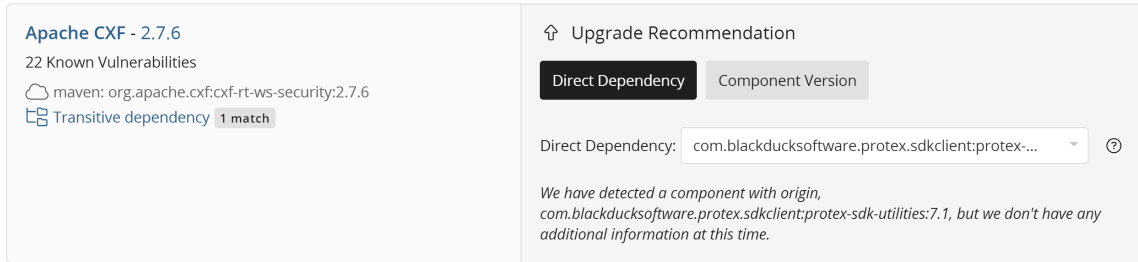
If your project version contains any component versions which have known vulnerabilities or are simply out of date, the Upgrade Recommendation section will display options you can explore to mitigate risk:

Short-Term recommendations provides a short-term upgrade path as it is typically the same major version as the version currently used in your BOM.

Unlike the short term upgrade recommendation, **Long-Term** recommendations usually requires a major version upgrade. This may require more planning and/or engineering work to implement.

Transitive match upgrade recommendations

It is more difficult to mitigate or remove component vulnerabilities brought in as transitive dependencies without understanding what root direct dependency brought in that component. Transitive Upgrade Guidance is calculated for top level parent of the component (transitive dependency match type) that has vulnerabilities and has known dependency tree.



In the Upgrade Recommendation section, you can see what the **Direct Dependency** is for the selected transitive component and the suggested upgrade for that component. By clicking the **Component Version**, you will see the upgrade guidance suggestions for the transitive component. Please see [Getting remediation guidance for components with security vulnerabilities](#) for more information on Risk Guidance and mitigation.

Analyzing the impact of a vulnerability

If a project version has several vulnerabilities, how can you decide which vulnerability you should focus on first?

To help you to prioritize which vulnerabilities you should address first, Black Duck can determine if any external public methods called by your Java applications are potentially involved in a known vulnerability. Black Duck can identify the called fully qualified public functional names in your source code and match them to the known function names being exploited by a vulnerability. By knowing whether any external public methods called by your Java applications are potentially involved in a known vulnerability, you can prioritize what vulnerabilities you need to concentrate on.

Vulnerability impact analysis works with Black Duck Security Advisories (BDSAs), a Black Duck-exclusive vulnerability data feed. Vulnerability metadata has been added to BDSAs that includes the fully qualified public function names that expose the vulnerability. Using this data, Black Duck can determine if vulnerable code is more likely to be invoked, and flags those vulnerabilities as reachable, indicating to you that these vulnerabilities are a higher priority for remediation.

! Important: Vulnerability impact analysis can help you triage vulnerabilities in open source components. It is *not* a definitive list of the vulnerabilities that do or do not affect your code.

Although Black Duck may indicate that a vulnerability is reachable, it does not indicate that the vulnerability definitely affects your code, as your code may not trigger the vulnerability.

Likewise, Black Duck may not denote that a vulnerability is reachable. However, that does not indicate that your code is safe from the vulnerability as Black Duck may not have data for a specific vulnerability.

Vulnerability impact analysis process

The process to display the possible impact of a vulnerability in Black Duck is:

1. Black Duck Detect analyzes the code

When the `--detect.impact.analysis.enabled` property in Black Duck Detect is set to **true**, Black Duck Detect creates a call graph (a list of calls made by your code) to understand the public methods your code is using in your application. The call graph shows the fully qualified public method names as well as the line number where the function was called.

Along with creating BOM files, Black Duck Detect creates a file which will be used by Black Duck KnowledgeBase for matching call graph signatures against BDSA-provided function signatures. This file is encrypted with SHA1 hashes. Hashing of the call graph signatures is completed at the client system.

The data is packaged into a single file and Black Duck Detect sends the file over HTTPS to the Black Duck server.

2. Black Duck sends data to the KnowledgeBase

Black Duck sends the hashed call graph function signatures to Black Duck KnowledgeBase via a Black Duck KnowledgeBase API.

3. Black Duck KnowledgeBase identifies vulnerable methods

The Black Duck KnowledgeBase uses the Function Signature Match Service to compare the hashed call graph signatures to KnowledgeBase hashed data for BDSA vulnerabilities with associated fully qualified public method names metadata. Fully qualified public method name matching is similar to signature matching: a discovered set of fully qualified public method names is compared against those known to the Black Duck KnowledgeBase.

The Black Duck KnowledgeBase sends the vulnerability metadata (vulnerable methods) to the Black Duck server via HTTPS.

4. Black Duck identifies vulnerabilities

Black Duck creates the data that needs to persist in the Black Duck PostgreSQL database. The data is stored as a new scan type (Vulnerability Impact) in Black Duck.

Black Duck cross references the identified methods with the vulnerable methods from BDSA to identify which vulnerabilities the user is calling in their code.

Black Duck displays the vulnerabilities for a component that have a method that is being called in the project version's **Security** tab.

Viewing reachable vulnerabilities

To view reachable vulnerabilities:

1. Set the **--detect.impact.analysis.enabled** property in Black Duck Detect to **true** to enable vulnerability impact analysis.
2. Once scanning completes and Black Duck has built the BOM, open the project version's **Security** tab which lists the security vulnerabilities for this project version.
3. Select a component from the **Component** list on the left side of the page to view a table which lists the vulnerabilities for this component.

Black Duck Projects
vulnImpactProject > 123

Project | Phase: In Development | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

Security Risk
Number of Unique Component Origins

Critical 2 High 3 Medium 3 Low 0

Filter Components... Add Filter

Components

- Apache Ant 1.8.2
maven: org.apache.ant:ant:1.8.2
Vulnerabilities 1 2
- Apache Ant 1.8.2
maven: org.apache.ant:ant-launcher:1.8.2
Vulnerabilities 1 2
- Apache Commons BeanUtils 1.9.3
maven: commons-beanutils:commons-beanutils:1.9.3
Vulnerabilities 2
- Apache Commons Email 1.2
Vulnerabilities 2
- Apache Commons Email 1.2
maven: org.apache.commons:commons-email:1.2

Apache Commons Email 1.2

2 Known Vulnerabilities

Short Term Upgrade Recommendation
1.18
Has no known vulnerabilities

Long Term Upgrade Recommendation
6.2
Has no known vulnerabilities

Identifier	Overall Score	Status	CWE	Exploit	Workaround	Solution
> BDSA BDSA-2017-0721 (CVE-2017-9801) Reachable	6.5	Medium	New	CWE-20, CWE-93	-	✓
> BDSA BDSA-2018-0558 (CVE-2018-1294) Reachable	5.7	Medium	New	CWE-144	-	✓

Displaying 1-2 of 2

Reachable located next to the BDSA record number indicates that there is a function call in your code that makes this vulnerability reachable. Use the **Reachable** filter to view all such BDSA records.

Click **Reachable** to view more information for an individual BDSA record.

Reachable Vulnerability

Vulnerable function calls in your code that make this vulnerability reachable.

BDSA-2017-0721 (CVE-2017-9801)

Vulnerable Functions

org.apache.commons.mail.Email.setSubject

Calls

We found 1 function call

Called by: com.nickavv.strutstest.App.main
Line: 17

Displaying 1-1 of 1

Close

The Reachable Vulnerability dialog box lists:

- Component name and version
- BDSA record
- A list of all vulnerable function calls in your code.

Select a function name to view the method name and line number in your code.

Note the following:

- This feature is available in Black Duck Detect version 6.5 or later (and Black Duck Detect (Desktop) that uses Black Duck Detect 6.5 and later). Set the `--detect.impact.analysis.enabled` property in Black Duck Detect to `true` to enable vulnerability impact analysis.
- This feature is for Java applications only. Black Duck Detect looks at Java `.class` files only.
- Black Duck Detect only discovers vulnerabilities in public methods that call potentially vulnerable functions.
- This feature displays reachable functions for BDSAs only.
- There is a [project version report](#), `vulnerability_matches_date_time.csv`, that lists the component, vulnerability data, and vulnerability impact analysis data for each component potentially reached by a vulnerability.
- A vulnerability condition, **Reachable from Source**, is available enabling you to create policy rules for vulnerabilities which has been identified as reachable. Use this condition to create policy rules that prioritize those vulnerabilities.

Viewing vulnerability details

Black Duck provides detailed information on a security vulnerability depending on whether you are viewing:

- [BDSA record](#)
- [CVE record](#)

Black Duck Security Advisories

Black Duck Security Advisories (BDSAs) are a Black Duck-exclusive vulnerability data feed sourced and curated by our Cybersecurity Research Center (CyRC). BDSAs offer deeper coverage for a wide set of vulnerabilities than is available through the National Vulnerability Database (NVD). While providing more timely and detailed vulnerability insights, including severity, impact and exploitability metrics. BDSAs also provide actionable remediation guidance to save time by providing details on fixed versions, patch information, exploits, and workarounds where available. Validated additional vulnerability references are also provided under the Technical page on BDSA records.

The CyRC team provide detailed vulnerability guidance over beyond what the NVD typically provide in CVE records. BDSA are also cross-checked and validated against possibly affected component versions this often results in additional and more accurate mappings for components and versions affected by a given vulnerability.

Where a BDSA has not been mapped to a component version which is mapped to a CVE record this indicates that the COSRI's team additional research has determined that this component version is not affected by the vulnerability. BDSAs are frequently reviewed and updated often on an hourly basis in the event of a new zero day vulnerability.

The NVD CVE records are typically not cross-checked nor does the NVD verify vulnerability data published or provided from 3rd parties. The NVD are typically slower to update their records when new vulnerabilities or data becomes available.

BDSA records should not be considered separate vulnerabilities from CVE records or other publicly available vulnerability data sources but instead viewed as additional research and insights which users can use to make better decisions, faster with regard to open source security vulnerabilities.

Black Duck Security Advisory
jQuery Vulnerable to Cross-Site Scripting Attack via location.hash Property
 BDSA-2013-0010 CVE-2011-4969 Published Feb 8, 2018 Updated Feb 8, 2018

HIGH 8.9 BDSA **Fix Available** Aug 22, 2011 **Exploit Available** Jan 29, 2013 **3,990 Days** Vulnerability Age

The jQuery library is vulnerable to cross-site scripting (XSS) attack caused by a lack of user input sanitization. It is possible for an attacker to craft a malicious <tag>, containing arbitrary JavaScript code and execute it by tricking a user into selecting it with the `location.hash` property. This would allow an attacker to steal an administrator's session tokens or execute arbitrary code on their behalf. It is also possible for an attacker to craft a malicious URI to exploit.

- How to fix it**
- Solution - Fix Available**
Fixed in version **1.6.3**.
A commit patch is available [here](#).
- No Workaround**

Scores and Metrics
 Scores for the related BDSA and NVD records, based on the Common Vulnerability Scoring System (CVSS).

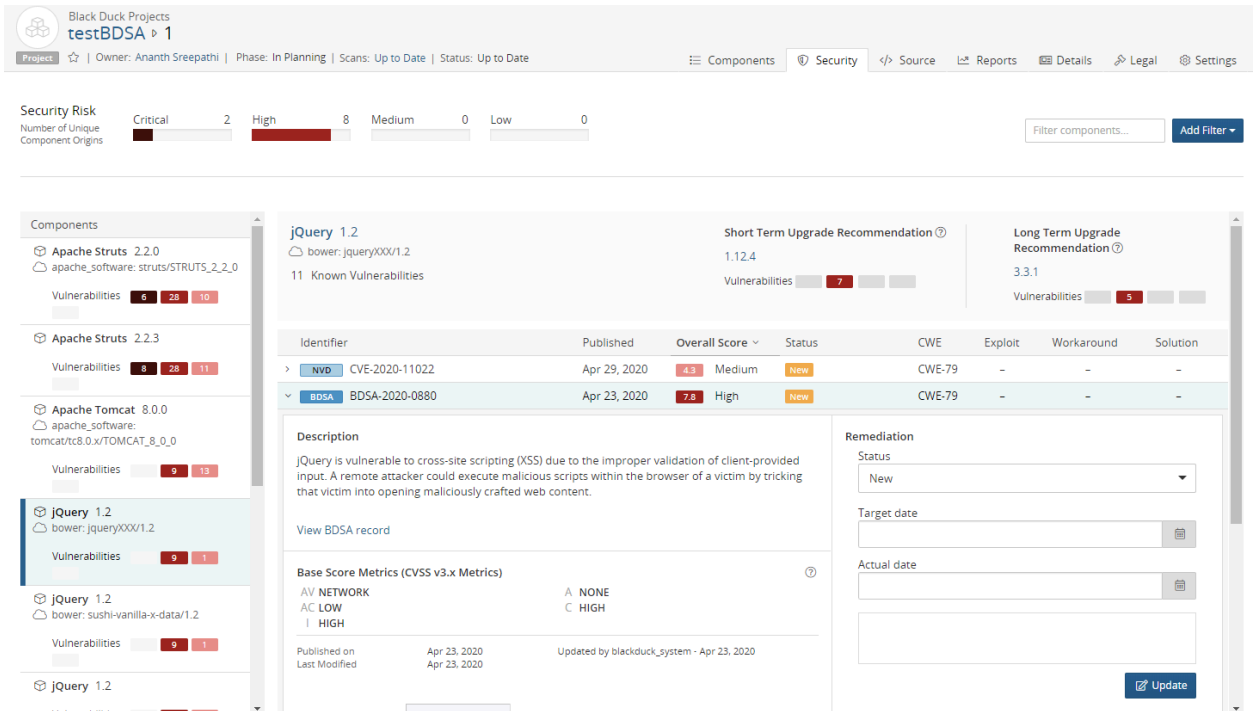


Common Weakness Enumeration (CWE)
 CWE-80 - Improper Neutralization of Script-Related HTML Tags in a Web Page (Basic XSS)
 The software receives input from an upstream component, but it does not neutralize or incorrectly neutralizes special characters such as "<", ">", and "&" that could be interpreted as web-scripting elements when they are sent to a downstream component that processes web pages.

Viewing a BDSA record

To view a BDSA record:

- Use the [Search feature](#) to locate BDSAs.
For example, search for BDSA-2017 to see the list of Black Duck Security Advisories from 2017.
Select a BDSA to view the record.
- Use the **Security** tab for a project version to view the vulnerabilities for a project version BOM.



The BDSA identifier (**BDSA**) indicates those vulnerabilities with a BDSA record.

Click > to view a description of the vulnerability and select **View BDSA record**.

Tip: Use your browser print feature to print the information shown in a tab.

Overview tab

By default, the **Overview** tab appears and displays the following information:

- The title bar displays the name of the vulnerability, BDSA number, CVE number (if there is a related CVE vulnerability), a published date (also known as the disclosure date), and an updated date (the last time the record was updated by NVD or BDSA).
- At the top of the page, the following information appears:



Apache Tomcat is vulnerable to reflected cross-site scripting (XSS) due to improper validation of user-supplied input in server-side includes (SSI) commands. This could allow an attacker to inject arbitrary web scripts and steal sensitive information such as authentication tokens or user cookies.

Shown here are the:

- BDSA score. Score based on analysis by Black Duck Software security analysts, who further investigated the vulnerability and provided a more detailed and accurate score. This includes the temporal score.
- Date of an available fix (if there is a fix available).
- Whether there is an exploit for this vulnerability.
- Vulnerability Age. Today's date - Disclosure date.
- A brief description of the vulnerability.
- The Vulnerability Tags section displays any specific vulnerabilities for this particular record, if applicable:
 - Zero-click Remote Code Execution (RCE). This vulnerability can result in the execution of code on the system, triggered by a remote attacker without requiring or relying on any third party action.



Zero-click Remote Code Execution

This vulnerability can result in the execution of code on the system, triggered by a remote attacker without requiring or relying on any third party action.

- Malicious Code Identified. This software contains code with malicious intent and is designed to have harmful or destructive consequences if executed within your system.



Malicious Code Identified

This software contains code with malicious intent and is designed to have harmful or destructive consequences if executed within your system.

- Embargoed Vulnerability Details. Technical details of this vulnerability are currently under embargo and the details are not published by the vendor at this time. An embargo remains in place for a fixed period. The BDSA record will be reviewed and updated with further details where possible once the embargo has been lifted.



Embargoed Vulnerability Details

Technical details of this vulnerability are currently under embargo and the details are not published by the vendor at this time. An embargo remains in place for a fixed period. The BDSA record will be reviewed and updated with further details where possible once the embargo has been lifted.

- Unconfirmed Vulnerability. This vulnerability does not have a code-based fix because the vendor has decided that the behavior of the component is intended and does not believe there is a vulnerability. The vendor may have resolved this issue by providing clarification in their documentation.



Unconfirmed Vulnerability

This vulnerability does not have a code-based fix because the vendor has decided that the behavior of the component is intended and does not believe there is a vulnerability. The vendor may have resolved this issue by providing clarification in their software documentation.

- Automated Security Advisory (ASA). Automated Security Advisories are automatically created by Black Duck's Cyber Security Research Center using automated AI tools. ASAs are created from various trusted security feeds such as the GitHub Security Advisories (GHSA) feeds along with automated vetting using AI tooling. These advisories are designed to supplement the BDSAs identified and verified by our [Cyber Security Research Center](#).



Automated Security Advisory

This is an automated vulnerability advisory. The details have not been independently verified by the Synopsys Cybersecurity Research Center.

- CISA Known Exploited Vulnerability. This vulnerability is listed in the Cybersecurity & Infrastructure Security Agency's (CISA) catalog. All federal civilian executive branch agencies are required to

remediate vulnerabilities in the KEV catalog within prescribed timeframes. Please visit [CISA's Known Exploited Vulnerability Catalog](#) page for more information.



CISA Known Exploited Vulnerability

All federal civilian executive branch agencies are required to remediate vulnerabilities in the KEV catalog within prescribed timeframes.

HTTP/2 Rapid Reset Attack Vulnerability

Added: Oct 10, 2023 Due Date: Oct 31, 2023

Action: Apply mitigations per vendor instructions or discontinue use of the product if mitigations are unavailable.

- **AI Assisted.** AI Assisted Security Advisories are automatically created by the Black Duck Cybersecurity Research Center using automated AI tools. These BDSAs have not been independently verified by the BDSA team but are created using automated processes and generative AI assistance. These advisories are designed to supplement the BDSAs identified and verified by the Cybersecurity Research Center.

AI Assisted

AI Assisted Security Advisories are automatically created by the Synopsys Cybersecurity Research Center using automated AI tools. These BDSAs have not been independently verified by the BDSA team but are created using automated processes and generative AI assistance. These advisories are designed to supplement the BDSAs identified and verified by the Cybersecurity Research Center.

- The **How to fix it** section describes a solution, if one is available, and a workaround.
- The **Scores and Metrics** section displays the scores for the related BDSA and NVD records (if applicable), based on the Common Vulnerability Scoring System (CVSS). Select a value above the graph to view the information in the graph and details below.


This section may also display a comparative, side-by-side graph if the vulnerability also has a NVD record.

Scores and Metrics

Scores for the related BDSA and NVD records, based on the Common Vulnerability Scoring System (CVSS).

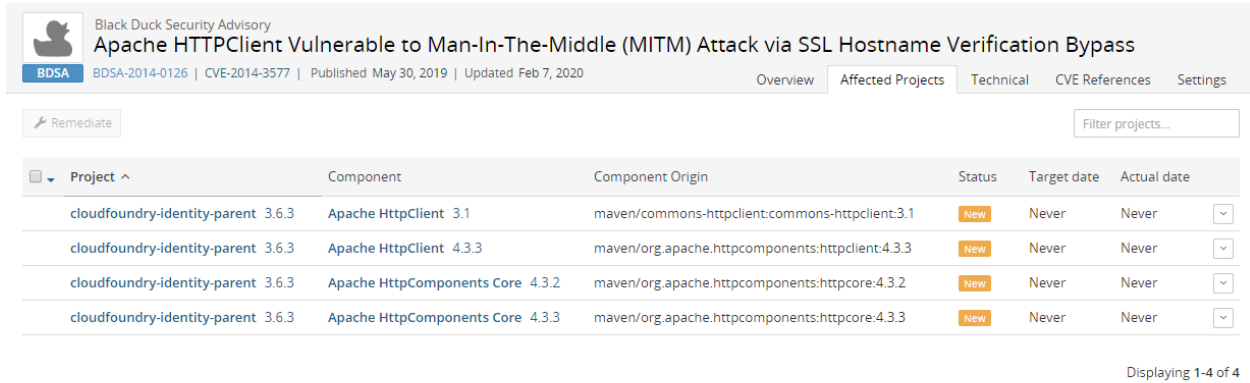
CVSS v2
CVSS v3.x

This Record BDSA-2020-0964 CVSS v3.x	Related Record CVE-2020-11023 CVSS v3.x
<div style="text-align: center; margin-bottom: 10px;"> Overall 8.6 High </div> <p>CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:L/E:P/RL:O/RC:C</p> <div style="text-align: center; margin-bottom: 10px;"> Temporal 8.6 High </div> <p>Exploitability</p> <p>Not Defined Unproven Proof of Concept Functional High</p> <p>Remediation Level</p> <p>Not Defined Official Fix Temporary Fix Workaround Unavailable</p> <p>Report Confidence</p> <p>Not Defined Unknown Reasonable Confirmed</p>	<div style="text-align: center; margin-bottom: 10px;"> Overall 6.1 Medium </div> <p>CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/L/I:L/A:N</p> <div style="text-align: center; margin-bottom: 10px;"> Temporal N/A </div> <p><i>NVD does not provide Temporal metrics.</i></p>
<div style="text-align: center; margin-bottom: 10px;"> Exploitability 2.8 Low </div> <p>Attack Vector</p> <p>Network Adjacent Network Local Physical</p> <p>Attack Complexity</p> <p>Low High</p> <p>Privileges Required</p> <p>None Low High</p> <p>User Interaction</p> <p>None Required</p> <p>Scope</p> <p>Unchanged Changed</p>	<div style="text-align: center; margin-bottom: 10px;"> Exploitability 2.8 Low </div> <p>Attack Vector</p> <p>Network Adjacent Network Local Physical</p> <p>Attack Complexity</p> <p>Low High</p> <p>Privileges Required</p> <p>None Low High</p> <p>User Interaction</p> <p>None Required</p> <p>Scope</p> <p>Unchanged Changed</p>
<div style="text-align: center; margin-bottom: 10px;"> Impact 6 Medium </div> <p>Confidentiality Impact</p> <p>None Low High</p> <p>Integrity Impact</p> <p>None Low High</p> <p>Availability Impact</p> <p>None Low High</p>	<div style="text-align: center; margin-bottom: 10px;"> Impact 2.7 Low </div> <p>Confidentiality Impact</p> <p>None Low High</p> <p>Integrity Impact</p> <p>None Low High</p> <p>Availability Impact</p> <p>None Low High</p>

 **Note:** For more information on vulnerability metrics, visit the NVD web site: <https://nvd.nist.gov/vuln-metrics>

Affected Projects tab

Select this tab to see a list of your projects that are affected by this vulnerability.




Project	Component	Component Origin	Status	Target date	Actual date
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 3.1	maven/commons-httpclient:commons-httpclient:3.1	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 4.3.3	maven/org.apache.httpcomponents:httpclient:4.3.3	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.2	maven/org.apache.httpcomponents:httcore:4.3.2	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.3	maven/org.apache.httpcomponents:httcore:4.3.3	New	Never	Never

Displaying 1-4 of 4


This tab lists all projects affected by this vulnerability:

- Project name and version affected by this vulnerability.
- Component name and version that contains this vulnerability.
- Component origin that contains this vulnerability.
- Remediation status of this vulnerability. Possible values are: New, Needs review, Mitigated, Patched, Duplicate, Remediation Required, Remediation Complete, or Ignored.
- Target date for remediating this vulnerability.
- Actual date this vulnerability was remediated.

Select  in the row of a project and select:

- **View all vulnerabilities** to view all vulnerabilities affecting this project version.
- **View related files to view** to display the **Source** tab filtered to display the affected files.

Use this tab to remediate the vulnerability for one or more projects by origin:

- In the row of the single project you want to remediate, do one of the following:
 - Select , select **Update Remediation Plan**, enter the remediation details, and click **Update**.
 - Select and click **Remediate**. Enter the remediation details, and click **Update**.
- For multiple projects that need the same remediation status, select in each row and click **Remediate**. In the Bulk Remediation dialog box, enter the remediation details, and click **Update**

Technical tab

Select the **Technical** tab to view a technical description and a list of references and related links.

Black Duck Security Advisory

Apache Tomcat Vulnerable to Reflected Cross-Site Scripting (XSS) via SSI 'printenv' Debugging Command

BDSA-2019-1661 | CVE-2019-0221 | Published May 30, 2019 | Updated May 30, 2019

Overview
Affected Projects
Technical
CVE References
Settings

Technical Description

Tomcat does not sanitize user-supplied variables in the SSI `printenv` debugging command within the file `java/org/apache/catalina/ssi/SSIPrintenv.java`. An attacker could exploit this on a Tomcat instance which has enabled SSI (disabled by default), and enabled the `printenv` directive for debugging purposes within `ssi/printenv.shtml`. For such an instance, the attacker could craft a malicious URL to be supplied to a victim, which if accessed will result in included web scripts being executed on their system.

References and Related Links



Advisories

<https://lists.apache.org/thread.html/6e6e9eac7b28fd63d249711e9d3ccd4e0a83f556e324aee37be5a8c@%3Cannounce.tomcat.apache.org%3E>



<https://www.nightwatchcybersecurity.com/2019/05/27/xss-in-ssi-printenv-command-apache-tomcat-cve-2019-0221/>



Vendor Upgrade

<http://tomcat.apache.org/>

<https://github.com/apache/tomcat/releases/tag/7.0.94>

<https://github.com/apache/tomcat/releases/tag/8.5.40>

<https://github.com/apache/tomcat/releases/tag/9.0.18>

Included in the **References and Related Links** section is a list of Key Events:

- **Discovered.** Date that the vulnerability was discovered.
- **Vendor Notified.** Date the official vendor was notified of this vulnerability.
- **Vendor Fix.** Date that the official vendor released a patch or upgrade to fix this vulnerability.
- **Disclosure.** Date the vulnerability was first publicly disclosed, whether as a bug or as a security vulnerability.
- **Vulnerability Age.** Today's date - Disclosure date.
- **Exploit Available.** Date an exploit became publicly available for this vulnerability.

Components tab

Select the **Components** tab to view a list of all known component versions affected by this particular BDSA vulnerability record.

10. About security risk • Viewing vulnerability details

Black Duck Security Advisory
Apache Log4j Vulnerable to Remote Code Execution (RCE) via Non-Default Pattern Layout

BDSA-2021-3779 | CVE-2021-45046 | Published Dec 15, 2021 | Updated Feb 18, 2022

Overview Affected Projects Technical Components CVE References Settings

Component	Vendor	Product	Versions
Log4j Compatibility API	apache	log4j_1-2_api	2.15.0
Apache Log4j JUL Adapter	apache	log4j_jul	2.15.0
Apache Log4j Web Adapters	apache	log4j_web	2.15.0
Apache Log4j	maven_package	org.apache.logging.log4j:log4j-core	2.15.0
Apache Log4j Layout for JSON template	maven_package	org.apache.logging.log4j:log4j-layout-template-json	2.15.0
Apache Log4j App Server Support	org_apache_logging_log4j	log4j_appserver	2.15.0
Log4j Flume NG	org_apache_logging_log4j	log4j_flume_ng	2.15.0
Apache Log4j JPA	org_apache_logging_log4j	log4j_jpa	2.15.0
logging [deprecated]	apache	log4j	2.15.0-rc1 → 2.15.1-rc1
apache:log4j	apache	log4j	2.15.0-rc1 → 2.15.1-rc1

CVE References tab

Select the **CVE References** tab to view links for additional information.

Black Duck Security Advisory
Apache Tomcat Vulnerable to Reflected Cross-Site Scripting (XSS) via SSI 'printenv' Debugging Command

BDSA-2019-1661 | CVE-2019-0221 | Published May 30, 2019 | Updated May 30, 2019

Overview Affected Projects Technical CVE References Settings

All 27 >

- BID 1
- BUGTRAQ 1
- CONFIRM 3
- DEBIAN 1
- FEDORA 2
- FULLDISC 1
- GENTOO 1
- MISC 2
- MLIST 8
- N/A 1
- REDHAT 2
- SUSE 2
- UBUNTU 2

BID

<http://www.securityfocus.com/bid/108545>

BUGTRAQ

<https://seclists.org/bugtraq/2019/Dec/43>

CONFIRM

<https://lists.apache.org/thread.html/6e6e9eac7b28fd63d249711e9d3ccd4e0a83f556e324aee37be5a8c@%3Cannounce.tomcat.apache.org>

<https://security.netapp.com/advisory/ntap-20190606-0001/>

https://support.f5.com/csp/article/K13184144?utm_source=f5support&utm_medium=RSS

DEBIAN

<https://www.debian.org/security/2019/dsa-4596>

Settings tab

Use this tab to manage the global remediation for this vulnerability. Click [here](#) for more information.

CVE record

Vulnerabilities are linked to components by the Common Vulnerabilities and Exposures numbers (CVEs), as reported in the National Vulnerabilities Database (NVD) maintained by the National Institutes of Standards and Technology (NIST).

The CVE record provides overview information on a vulnerability, a list of affected projects, and links to references.

Overview tab

By default, the **Overview** tab appears and displays the following information:

- The title bar displays the CVE number, a published date (date that NVD published the CVE), and an updated date (last modified date by NVD).
- A description of the vulnerability.

If there is a BDSA record, select the link to view this information.

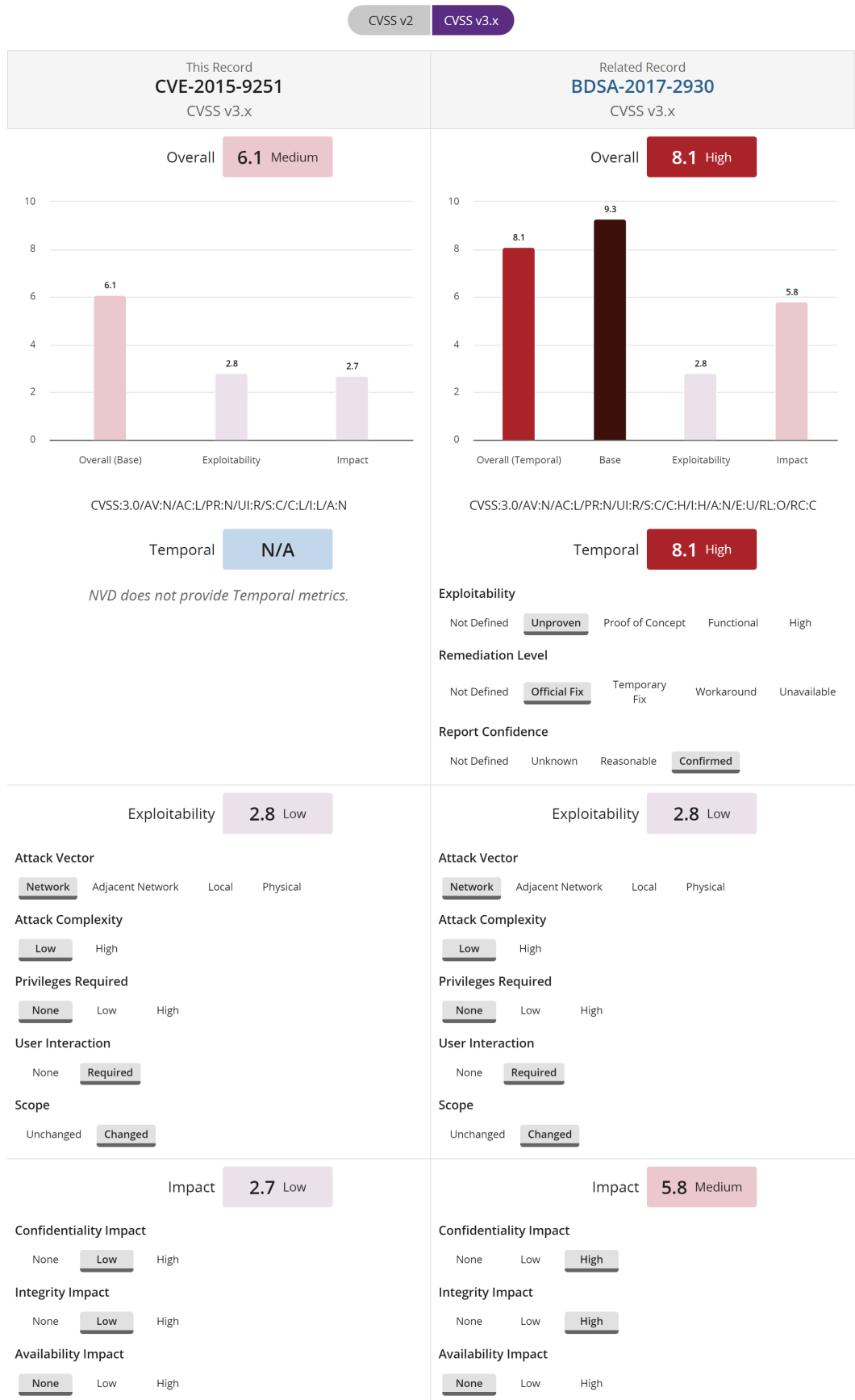
- The **Scores and Metrics** section displays the scores for the related BDSA and NVD records (if applicable), based on the Common Vulnerability Scoring System (CVSS). Select a value above the graph to view the information in the graph and details below.


This section may also display a comparative, side-by-side graph if the vulnerability also has a BDSA record.

10. About security risk • Viewing vulnerability details

Scores and Metrics

Scores for the related BDSA and NVD records, based on the Common Vulnerability Scoring System (CVSS).



 **Note:** For more information on vulnerability metrics, visit the NVD web site: <https://nvd.nist.gov/vuln-metrics>

Affected Projects tab

Select this tab to see a list of your projects that are affected by this vulnerability.



National Vulnerability Database
CVE-2014-3577
 NVD Published Aug 21, 2014 | Updated Jul 18, 2018 | <https://nvd.nist.gov/vuln/detail/CVE-2014-3577>

Overview **Affected Projects** References Settings

Remediate

Project ^	Component	Component Origin	Status	Target date	Actual date
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 3.1	maven/commons-httpclient:commons-httpclient:3.1	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 4.3.3	maven/org.apache.httpcomponents:httpclient:4.3.3	New	Never	Never

Displaying 1-2 of 2

This tab lists all projects affected by this vulnerability:

- Project name and version affected by this vulnerability.
- Component name and version that contains this vulnerability.
- Remediation status of this vulnerability. Possible values are: New, Needs review, Mitigated, Patched, Duplicate, Remediation Required, Remediation Complete, or Ignored.
- Target date for remediating this vulnerability.
- Actual date this vulnerability was remediated.

Select in the row of a project and select:

- **View all vulnerabilities** to view all vulnerabilities affecting this project version.
- **View related files to view** to display the **Source** tab filtered to display the affected files.

Use this tab to remediate the vulnerability for one or more projects by origin:

- In the row of the single project you want to remediate, do one of the following:
 - Select , select **Update Remediation Plan**, enter the remediation details, and click **Update**.
 - Select and click **Remediate**. Enter the remediation details, and click **Update**.
- For multiple projects that need the same remediation status, select in each row and click **Remediate**. In the Bulk Remediation dialog box, enter the remediation details, and click **Update**

References tab

Select the **References** tab to view links for additional information.

10. About security risk • Remediating security vulnerabilities

National Vulnerability Database
CVE-2016-8735
 NVD Published Apr 6, 2017 | Updated Apr 23, 2019 | <https://nvd.nist.gov/vuln/detail/CVE-2016-8735>

Overview Affected Projects References Settings

All 32 >

BID 1

CONFIRM 13

DEBIAN 1

MISC 2

MLIST 11

REDHAT 3

SECTRACK 1

BID

<http://www.securityfocus.com/bid/94463>

CONFIRM

<http://seclists.org/oss-sec/2016/q4/502>

<http://svn.apache.org/viewvc?view=revision&revision=1767644>

<http://svn.apache.org/viewvc?view=revision&revision=1767656>

<http://svn.apache.org/viewvc?view=revision&revision=1767676>

<http://svn.apache.org/viewvc?view=revision&revision=1767684>

<http://tomcat.apache.org/security-6.html>

<http://tomcat.apache.org/security-7.html>

<http://tomcat.apache.org/security-8.html>

<http://tomcat.apache.org/security-9.html>

<http://www.oracle.com/technetwork/security-advisory/cpujan2018-3236628.html>

<http://www.oracle.com/technetwork/security-advisory/cpujul2018-4258247.html>

<http://www.oracle.com/technetwork/security-advisory/cpuoct2017-3236626.html>

<https://security.netapp.com/advisory/ntap-20180607-0001/>

DEBIAN

<http://www.debian.org/security/2016/dsa-3738>

Settings tab

Use this tab to manage the global remediation for this vulnerability. Click [here](#) for more information.

Remediating security vulnerabilities

Vulnerabilities have a remediation status assigned to them. A new vulnerability can have a status of **New**, **Needs Review**, **Patched**, or **Duplicate**.


The following table describes each remediation status and whether a vulnerability with this status is included in the security risk calculations:


Remediation Status	Included in Security Risk Calculation?	Definition
New	Yes	Black Duck has determined that a vulnerability affects this component version.
Needs Review	Yes	Black Duck cannot determine if a vulnerability definitely affects this component version. This can occur when a component version is known to contain a vulnerability, but it cannot be determined whether the patch or sub-version being used is affected by this vulnerability.
Remediation Required	Yes	Remediation is required for the component version.
Remediation Complete	No	Remediation for the vulnerability is complete.
Duplicate	No	This vulnerability is a duplicate.

Remediation Status	Included in Security Risk Calculation?	Definition
Mitigated	No	The vulnerability has been mitigated.
Patched	No	The vulnerability in this version of a Linux distribution package has been patched. Although a vulnerability has been reported on the overall component version, the vulnerability does not affect this specific matched version as the version has been patched from the source from where it came.
Ignored	No	The vulnerability has been ignored.

Remediating a vulnerability

You may wish to change the remediation status after reviewing the severity of the vulnerability. Black Duck can help you [determine which version you should use](#) when a component has a vulnerability. Black Duck helps you to understand your options when a component has a security vulnerability.

 **Important:** If a CVE record has a related BDSA record (or vice versa), it cannot be remediated unless that vulnerability record type is prioritized in the [Security Risk Ranking](#). This is due to the fact that the non-prioritized vulnerability record is not being used as a determinant and is not used to calculate security risk.

 **Note:** You can select any value for the remediation status. Selecting **Remediation Complete**, **Mitigated**, **Patched**, or **Ignored** removes the vulnerability from the security risk calculations.

You can remediate a vulnerability for current projects or set a [global remediation status](#) that applies to new instances of that vulnerability when that component appears on new BOMs.

Only users with the appropriate [role](#) can remediate vulnerabilities.

To remediate vulnerabilities for current projects:

Use this method to identify and remediate the vulnerabilities affecting a specific project version.

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.
4. Select the **Security** tab which lists all components and subprojects with associated security vulnerabilities for this project version.

10. About security risk • Remediating security vulnerabilities

The screenshot shows the Black Duck Security Risk interface. At the top, it displays 'Black Duck Projects Sample Project v 1.0' and navigation options like 'Components', 'Security', 'Source', 'Reports', 'Details', and 'Settings'. Below this, a 'Security Risk' summary shows a bar chart with categories: Critical (0), High (2), Medium (15), and Low (8). A 'Filter components...' button is also present.

The main content area is divided into two parts. On the left, a sidebar lists components with their vulnerability counts: Hibernate Validator (1), iText (1), Jersey (1), and Jetty (1). On the right, a detailed view for 'Hibernate Validator 5.2.4.Final' is shown. It includes a 'Short Term Upgrade Recommendation' (5.4.3.Final) and a 'Long Term Upgrade Recommendation' (6.1.2.Final). Below this is a table of vulnerabilities:

Identifier	Published	Overall Score	Status	CWE	Exploit	Workaround	Solution
> BDSA BDSA-2019-3481	Nov 13, 2019	3.2 Low	New	CWE-79	-	✓	✓
> NVD CVE-2017-7536	Jan 10, 2018	4.4 Medium	New	CWE-470	-	-	-

At the bottom of the table, it says 'Displaying 1-2 of 2'. There is also a 'Capture screenshot...' button.


5. Select a component in the table on the left to view the associated vulnerabilities.

Black Duck provides [remediation guidance](#) for components with security vulnerabilities.

6. Select > in the table next to the vulnerability to expand the row. A brief description, additional information, and fields to remediate the vulnerability appear. You can also hover over the **Overall Score** value to see a breakdown of the score.

- To remediate the vulnerability: enter a different status, additional remediation details, such as the target and actual date, and click **Update**.
- To view additional information, select to view [the BDSA record](#) or [the CVE record](#).

View the projects affected by this vulnerability by selecting the **Affected Projects** tab.

 **Note:** When viewing projects on this tab, you may see rows that are hashed out and cannot be selected for remediation. This is due to the project having a linked type of security vulnerability record, either BDSA or CVE. If that vulnerability record is not prioritized in the [Security Risk Ranking](#), a Remediation Plan cannot be undertaken for that project. Consider viewing the prioritized security vulnerability record to update the Remediation Plan.

Use this tab to remediate the vulnerability for this and/or additional project versions by origin:

- In the row of the single project you want to remediate, do one of the following:
 - Select , select **Update Remediation Plan**, enter the remediation details, and click **Update**.
 - Select and click **Remediate**. Enter the remediation details, and click **Update**.
- For multiple projects that need the same remediation status, select in each row and click **Remediate**. In the Bulk Remediation dialog box, enter the remediation details, and click **Update**.

Getting remediation guidance for components with security vulnerabilities

Black Duck informs you of the vulnerabilities that impact the components in your BOMs. Detailed information is provided for each vulnerability, including a description and vulnerability scores.

After reviewing this information, you may need guidance as to what other component versions are available and whether there is a version that fixes the security vulnerability that affects the component version used in your BOM.

Black Duck provides this information: for a security vulnerability in your BOM, Black Duck displays the possible versions of the component that are available to you:

- The version used in your BOM with the number of vulnerabilities.
- Dependency information. When direct or transitive dependencies are found in a Black Duck Detect scan, Black Duck lists the number of matches for each type of dependency.

Apache Tomcat 8.5.32
 maven: org.apache.tomcat:tomcat-juli:8.5.32
 Direct dependency 39 matches
 Transitive dependency 2 matches
 16 Known Vulnerabilities

Select **Transitive dependency** to view the dependency tree for that component.

Dependency Tree for Apache Tomcat 8.5.32

maven: org.apache.tomcat:tomcat-juli:8.5.32 1 Critical 10 High 5 Medium 0 Low

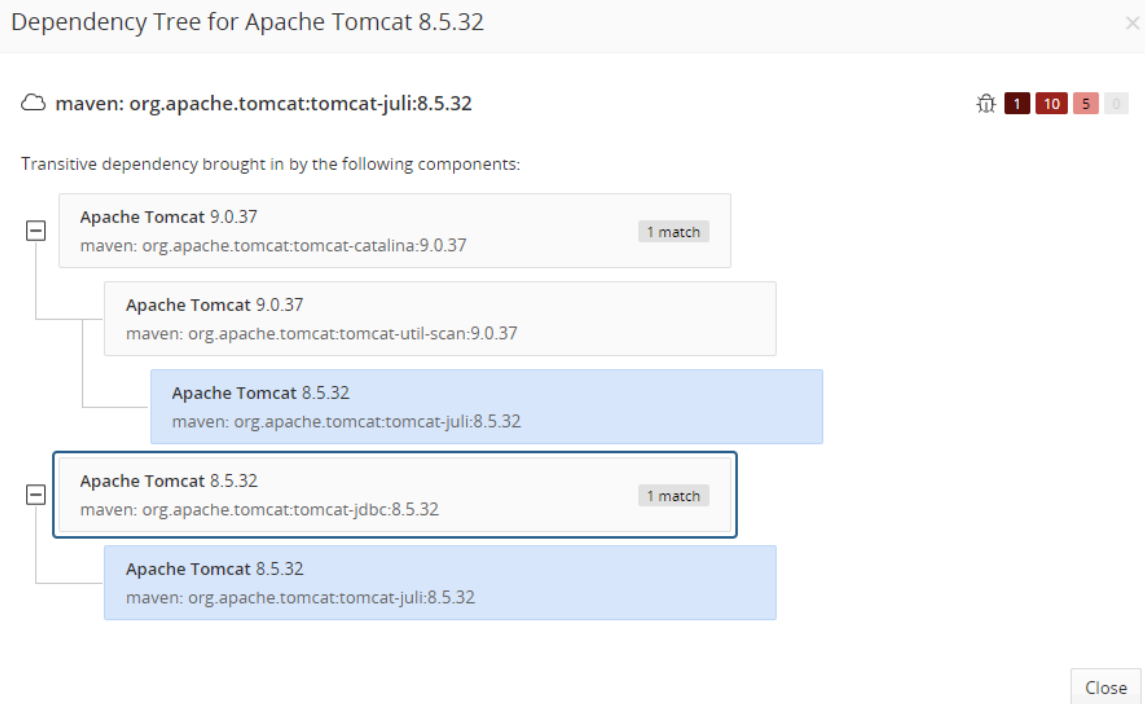
Transitive dependency brought in by the following components:

- Apache Tomcat 9.0.37 (1 match)
maven: org.apache.tomcat:tomcat-catalina:9.0.37
- Apache Tomcat 8.5.32 (1 match)
maven: org.apache.tomcat:tomcat-jdbc:8.5.32

Close

The dependency tree shows the components that brought in this dependency. In the upper right corner is a list of the vulnerabilities by severity level, from left to right: Critical, High, Medium, and Low for this origin of this component version. The match count is the number of times the component was brought in with that dependency path.

Click + to open the tree.



- **Recommendations.** If available, Black Duck provides a short term and long term upgrade recommendation. In both instances, the recommended version has fewer reported vulnerabilities than the version you are currently using in your BOM. The recommended version is also from the same origin as the version you are currently using in your BOM.

The guidance is based on a combination of factors including overall vulnerability posture being no worse or better than the original version and tie breakers on the versions that are newer. The algorithm does not seek to mitigate any specific vulnerability, it only seeks to improve posture. If however, a particular version had an additional vulnerability, or the number of vulnerabilities was the same but it's severity was "Critical" then no short-term recommendation would be made (would suggest staying on the current version).

- **Short Term Upgrade Recommendation.** This recommendation provides a short-term upgrade path as it is typically the same major version as the version currently used in your BOM. Components using non-semantic versioning will not have a short-term recommendation.
- **Long Term Upgrade Recommendation.** Unlike the short term upgrade recommendation, this recommendation usually requires a major version upgrade. This may require more planning and/or engineering work to implement.

For each suggestion, select the version number to open the [Component Name Version](#) page.

Use this information to guide you in determining [how to remediate](#) a security vulnerability.

To view guidance information:

1. a. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to open the **Components** tab and view the BOM.
3. Select the **Security** tab which lists all components and subprojects with associated security vulnerabilities for this project version.

- Select a component from the **Component** table on the left side of the page to view a table which lists the vulnerabilities for this component and provides more information on each vulnerability. Above the table are the suggestions of versions you can use to replace the selected component.

The screenshot shows the Black Duck Projects interface for a 'Sample Project' in the 'In Planning' phase. The 'Security Risk' section shows a bar chart with 0 Critical, 2 High, 15 Medium, and 8 Low risks. The left sidebar lists components like Hibernate Validator, iText, Jersey, and Jetty. The main area displays details for 'Hibernate Validator 5.2.4.Final', including a table of vulnerabilities:

Identifier	Published	Overall Score	Status	CWE	Exploit	Workaround	Solution
> BDSA BDSA-2019-3481	Nov 13, 2019	3.2 Low	New	CWE-79	--	✓	✓
> NVD CVE-2017-7536	Jan 10, 2018	4.4 Medium	New	CWE-470	--	--	--

Managing global remediation for a vulnerability

There may be a vulnerability that appears frequently in your BOMs. Instead of repeatedly reviewing and remediating that vulnerability, set a global default remediation status for it.

After you set a global remediation status, when that vulnerability appears in new BOMs, it will automatically get the global remediation status you defined.

Note: Any component in your existing BOMs that has that vulnerability retains its current remediation status. The global remediation status only applies to *new* instances of the vulnerability.

You must have the Global Security Manager [role](#) to set or remove a global remediation status for a vulnerability.

Setting a global remediation for a vulnerability

Use this process to set a global remediation status or edit an existing status.

- Find the vulnerability you want to remediate globally. For example, you can:
 - Use the [Security tab for a specific project version](#).
 - Use the [Search feature](#).
- View the security record.
 - When viewing a list of vulnerabilities in a table, select > in the table next to the vulnerability to view a brief description. Then select either **View BDSA record** or **View CVE record**.
 - When using the Search feature, select the BDSA or CVE record in the search results.
- Select the **Settings** tab.

10. About security risk • Remediating security vulnerabilities

The screenshot shows the 'Default Remediation Status' dialog box. At the top, it displays the Black Duck Security Advisory logo and the title 'Mozilla Thunderbird Vulnerable to Information Disclosure via Out-of-Bounds Read in 'nsParseMailbox.cpp' File'. Below the title, there are navigation tabs: 'Overview', 'Affected Projects', 'Technical', 'CVE References', and 'Settings'. The main content area contains a dropdown menu labeled 'Status' with the placeholder text 'Select Default Remediation Status...'. Below the dropdown is a text area labeled 'Comments'. At the bottom right, there are two buttons: 'Clear' and 'Save'.

4. Select a default status and optionally, enter a comment. This comment appears when viewing the description, as described below.

5. Click **Save**.

The Default Remediation Status Confirmation dialog box appears.

6. Click **Confirm**.

Clearing a global default remediation status

You can remove a global default remediation status. Clearing a status only affects future vulnerabilities: components with the existing global vulnerability status will retain that status. To modify the status of the existing vulnerabilities, modify the remediation status manually either individually or by using bulk remediation.

1. Find and display the vulnerability record as described in the previous section.

2. Select the **Settings** tab.

The screenshot shows the 'Default Remediation Status' dialog box with the 'Settings' tab selected. The 'Status' dropdown menu is set to 'Remediation Required'. The 'Comments' text area contains the text: 'Although the security risk is low, this vulnerability must be remediated.' The 'Clear' and 'Save' buttons are visible at the bottom right.

3. Click **Clear**.

The Default Remediation Status confirmation dialog box appears.

4. Click **Confirm**.

Viewing all vulnerabilities with global remediation

You can view all vulnerabilities with global remediation by selecting the **Default Remediation** filter when [searching for vulnerabilities](#). Select the BDSA or CVE record number in the search results and then select the **Settings** tab, as described previously to view the remediation status.

11. Managing policies


The Policy Management feature enables you to create rules to govern your use of open source components. With policy rules, open source usage can be managed on an exception basis – as long as open source components meet the policy requirements their usage is allowed. Any open source components/versions that fail to meet your policy rules are flagged, enabling you to review and determine if the use of the component should be allowed in the particular application.

About the policy process

To use the policy management feature:

1. **Create rules** that enforce your policies; a user with the Policy Manager **role** can create and manage policy rules. When creating policy rules determine:

- Whether to enable the rule. BOMs will not be evaluated until the rule is enabled.
- Whether the rule can be manually overridden.
- The conditions for this rule.

 **Note:** Rules can have multiple conditions; *all* conditions must be true for a component to be in violation of the rule.

2. View the violations and determine what to do with components that are in violation of a rule.

If you enabled the option, violations can be **manually overridden**.

3. Optionally,

- Create additional policies and/or **edit**, **delete**, or **disable or enable** your existing policies.
- Select a category for your rule. Black Duck provides these categories for a policy rule: component, security, license, operational, and uncategorized (default).

By using categories and filters, you can easily find policies (on the Policy Management page) or policy violations (on the BOM page) by category.

- **View the Project Version report**. This report includes policy violation information:
 - The `components_date_time.csv`, `bom_component_custom_fields_date_time.csv`, and `source_date_time.csv` files list the policy status and override information.
 - The `version_date_time.csv` file indicates whether this version of the project has a policy violation.

To assist you, Black Duck provides five **default policy rules** that you can view, modify, enable, or delete. These policy rules are disabled by default.

Viewing policy rules

The Policy Management page lists all your policy rules and indicates whether the rule allows manual

overrides. View this page by clicking  and selecting **Policies**:

Policy Management				
+ Create Policy Rule		Delete		Enabled Enabled ▾ × Add Filter ▾
Policy Rule	Description	Severity	Category	
>	No External Tier 1 or Tier 2 Projects With More Than 1 High Vulnerability	Disallow External Projects With More Than 1 High Vulnerability at Tier 1 or 2	Critical	Security
>	No External Tier 1 or Tier 2 Projects With More Than 3 Medium Vulnerabilities	Disallow External Projects With More Than 3 Medium Vulnerabilities at Tier 1 or 2	Major	Security
>	No Modified Components Without Description	Disallow components that have the modification flag on but do not have a description.	Unspecified	Uncategorized

Displaying 1-3 of 3

- The page is filtered to display enabled rules. Modify or clear the filter to view disabled rules.
- All rules can be overridden unless noted.
- Click > to view the conditions of this rule and who created and last updated it.

From this page, you can view, [create](#), [edit](#), or [delete](#) policy rules.

Viewing policy rule violations

When a component is in violation of a policy rule, the Policy Violation icon (⚠) appears in the UI on the following pages:

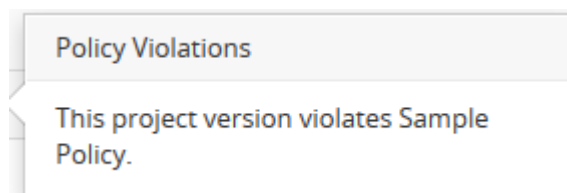
- Source page. Icon appears next to the file name to indicate that a file in a component is in violation.
- BOM page. Icon appears next to components in violation.

In the Tree View of the BOM, ⚠ next to the parent component indicates that a child has a policy violation.

- Custom dashboards. Icon appears next to the project name to indicate that this project has a version which has a policy violation.
- Project Version page. Icon appears next to the version to indicate that it has a policy violation.

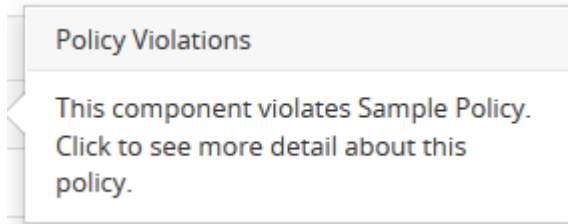
Hover over the icon to view to view more information:

- On the project level, information such as the following appears:



This information also appears at the component/file level for users who are members of projects or have project-group privileges.

- On the component/file level, the following information appears for users with the BOM Manager, Global Project Administrator, Global Project Manager, Project Manager, and Policy Violation Reviewer roles:

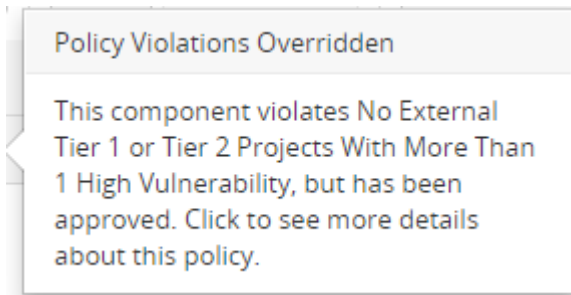


Clicking the icon (when viewing the BOM using the List view) displays the Policy Violations dialog box from which you can override the policy violation.

Overriding violations

If a rule was configured to allow manual overrides of violations, then you can [override a disapproved component](#) or file in that project.

When all component violations have been overridden, the Policy Violation Override icon (Ⓞ) appears in the UI. In the Tree View, Ⓞ indicates that a child's policy violation has been overridden; it appears at the parent level. Click the icon to view more information.



Removing policy overrides

If a violation of a policy should not have been overridden, you can [remove](#) the override.

Default policy rules

Black Duck provides five default policy rules which are disabled by default. Users with the Policy Manager role can [enable](#), [edit](#), or [delete](#) these rules.

View these policy rules on the Policy Management page by clicking



> **Policies** and selecting to view disabled rules:

Policy Management				
+ Create Policy Rule		Delete		Add Filter
Policy Rule	Description	Severity	Category	
>	No External Tier 1 or Tier 2 Projects With More Than 1 High Vulnerability	Disallow External Projects With More Than 1 High Vulnerability at Tier 1 or 2.	Unspecified	Uncategorized
>	No External Tier 1 or Tier 2 Projects With More Than 3 Medium Vulnerabilities	Disallow External Projects With More Than 3 Medium Vulnerabilities at Tier 1 or 2.	Unspecified	Uncategorized
>	No External Projects With Reciprocal Licenses	Disallow External Projects With Reciprocal Licenses	Unspecified	Uncategorized
>	No Components Marked for Modification	Disallow components that have the modification flag on in the edit window.	Unspecified	Uncategorized
>	No Modified Components Without Description	Disallow components that have the modification flag on but do not have a description.	Unspecified	Uncategorized

Displaying 1-5 of 5

Click > to view a description and the conditions for these rules.

These policy rules are in the **Uncategorized** category.

The default rules are:

- No External Projects With Reciprocal Licenses
- No External Tier 0, Tier 1 or Tier 2 Projects With More Than 1 High Vulnerability
- No External Tier 0, Tier 1 or Tier 2 Projects With More Than 3 Medium Vulnerabilities
- No Components Marked for Modification
- No Modified Components Without Description

Creating a policy rule

Create rules to ensure that your projects do not have an open source component, component version, or vulnerability that violates your policies.

You can create multiple rules and rules can have multiple conditions giving you the flexibility to create generic or highly specific global policy rules.

 **Note:** Only users with the Policy Manager [role](#) can create policy rules.

To create a policy:

1. Log in to Black Duck as a user with the Policy Manager role.

2.



Click **Manage** and select **Policies**.

3. Click **Create Policy Rule** to display the Create Policy Rule dialog box.

4. Complete the following:

- **Name.** Required. Name of this policy.
- **Category.** Optional. Assign one of the following categories for this policy:
 - Uncategorized. This is the default value.
 - Component.
 - License.

- Operational.
- Security.

Black Duck provides a filter in the project version BOM (to view policy violations by category) and the Policy Management pages (to view policies by category).

- **Description.** Optional. This description appears when you select > on the Policy Management page.
- **Severity.** Optional. The severity level of this policy. You can use this option with build integrations to indicate what should happen when a policy violation occurs. For example, all policy violations with a severity of Blocker should fail the build.

Select one of the following values: **Blocker**, **Critical**, **Major**, **Minor**, or **Trivial**.

- **Scan Modes.** Select whether this policy rule applies to Full Scans (default value), Rapid Scans, or both.
- **Enabled.** Clearing this option disables this rule. BOMs will not be evaluated until the rule is enabled. Clear the option if you want to create draft policy rules.

You can [enable or disable](#) the rule after it is created.

- Select whether to allow manual overrides for this rule.

Users with the Policy Manager role can [override a disapproved component](#) in projects in which they are a member or have project-group privileges.

- Select whether this policy rule applies to all projects or a subset of filtered projects – projects with specific properties.

Selecting filtered projects displays the policy filters, as described above, that you can select for this policy rule. Select a project filter, an operator, and specify a value.

5. For a project condition: ensure you have the **A Subset of Projects, filtered by...** option is enabled, select an attribute from the Project Conditions list, select an operator, and specify a value.


Click **+ Project Condition** to specify additional project conditions.

6. For a component condition: select an attribute from the **Component Conditions** list, select an operator, and specify a value.

Click **+ Component Condition** to specify additional component conditions.

7. For a vulnerability condition: select an attribute from the **Vulnerability Conditions** list, select an operator, and specify a value.

Click **+ Vulnerability Condition** to specify additional vulnerability conditions.

8. To remove a condition, click  in the row of the condition you wish to remove.

9. Click **Create**.

If the rule is enabled, existing BOMs are evaluated to determine if they are in violation of this rule. For any components that are in violation of component or vulnerability conditions, the Policy Violation icon (Ⓢ) appears next to component name.

Policy conditions

Creating the condition(s) for a policy rule consists of selecting the projects that this rule applies to (all or specific project attributes) and then selecting:

1. A component and/or vulnerability attribute


You can create a policy rule for a component, a vulnerability, or for a component and vulnerability combination.

2. An operator (such as equals or greater than)
3. A value (depending on the option you selected)

Components that meet the conditions will violate the policy rule. For vulnerability conditions, components that have vulnerabilities that meet the vulnerability conditions violate the policy rule.

You can create multiple conditions for a policy rule: *all* conditions must be true for a component to be in violation.

When evaluating components with multiple licenses for policy rules created using one or more of these license conditions: license, license status, license family and/or license expiration date, each license is evaluated and *all* license conditions must be true for a policy violation. If license risk is included as a policy condition, license risk is evaluated independently: all licenses for the component are evaluated, not just the license that met the other license policy conditions. Therefore, a policy violation can be triggered if one license meets the policy rule for multiple conditions while another license for that component meets the license risk condition.

 **Note:** All attributes appear for you to select, including attributes for those modules for which you are not licensed.

The table below shows the project filters you can select and the values you can specify.

Project Conditions filters

The Project Conditions filter is displayed when the **A Subset of Projects, filtered by...** option is enabled.



Project Condition filters are divided into these categories:

- Properties
- Project Custom Fields
- Project Version Custom Fields

Table 8: Properties

Project Filters	Value
Project Name	Begin typing to view possible values.
Project Group Name	Begin typing to view possible values.
Project Tags	Enter the tag name.
Project Tier	Enter one of the following values: 0 - 5.
Project Phase	Select one of the following values: <ul style="list-style-type: none"> • Deprecated • In Development • In Planning • Pre-Release • Released
Project Distribution Type	Select one of the following values:

Project Filters	Value
	<ul style="list-style-type: none"> • External • Internal • Open Source • SaaS

Table 9: Project Custom Fields

Project Filters	Value
Project Custom Field Name	Available for Boolean, Date, Drop Down, Multiple Selections, Single Selection, and Text field types. Select a value.

Table 10: Project Version Custom Fields

Project Filters	Value
Project Version Custom Field Name	Available for Boolean, Date, Drop Down, Multiple Selections, Single Selection, and Text field types. Select a value.

Component conditions

Component conditions are divided into these categories:

- Properties
- Operational
- Vulnerabilities
- Licenses
- Custom Fields: BOM Component, Component, and Component Version

Table 11: Properties

Component Condition	Value
Component	Begin typing to view possible component values. After selecting a component, the version field appears whereby you can enter a version number. Any Version is the default value if you do not enter a specific version.
Component Usage	Select one of the following values: <ul style="list-style-type: none"> • Dev. Tool / Excluded • Dynamically Linked • Source Code • Statically Linked • Separate Work • Implementation of Standard • Prerequisite • Merely Aggregated • Unspecified

Component Condition	Value
Review Status	Select one of the following values: <ul style="list-style-type: none"> • Not Reviewed • Reviewed
Match Type	Select one of the following values: <ul style="list-style-type: none"> • Files Added/Deleted • File Dependency • Direct Dependency • Transitive Dependency • Exact Directory • Exact File • Files Modified • Manually Added • Manually Identified • Partial • Snippet • Binary • Direct Dependency Binary • Transitive Dependency Binary
Component Purpose	Select either Yes or No. Indicates whether information was added in the Purpose field when manually adding or editing a component.
Component Modified	Select either Yes or No. Indicates whether the Modification option was selected when manually adding or editing a component.
Component Modification	Select either Yes or No. Indicates whether information was added to the Modification field when manually adding or editing a component.
Component Approval Status	Select one of the following values: <ul style="list-style-type: none"> • Unreviewed • In Review • Reviewed • Approved • Limited Approval • Rejected • Deprecated
Component Version Approval Status	Select one of the following values: <ul style="list-style-type: none"> • Unreviewed • In Review • Reviewed • Approved • Limited Approval • Rejected • Deprecated
Unknown Component Version	Select True or False .

Component Condition	Value
	If you select True , any component that has a ? as the version will trigger a policy violation.
Unconfirmed Snippets	Select True or False . If you select True , any snippet that has not been reviewed will trigger a policy violation.

Table 12: Operational

Component Condition	Value
Component Release Date	Select a date.
Newer Versions Count	Enter a number.
Commits in the past year	Enter a number.
Contributors in the past year	Enter a number.

Table 13: Vulnerabilities

Component Condition	Value
Critical Severity Vulnerability Count	Enter a number.
High Severity Vulnerability Count	Enter a number.
Medium Severity Vulnerability Count	Enter a number.
Low Severity Vulnerability Count	Enter a number.
Highest Vulnerability Score	Enter a number between 0 and 10, including decimal numbers.

Table 14: Licenses

Component Condition	Value
Unfulfilled License Terms	Select True or False . If you select True , any component that has unfulfilled license terms will trigger a policy violation. Note: The Legal tab must be enabled for a user to indicate that a term is fulfilled. If the Legal tab is disabled, a user will be unable to indicate that a term is fulfilled, and policy violations cannot be cleared.
License Conflict with Project Version	Select True or False . If you select True , a policy violation is triggered when a component's license conflicts with the license for a project version.
License Risk	Select one of the following values: <ul style="list-style-type: none"> • None • Low • Medium

Component Condition	Value
	<ul style="list-style-type: none"> High
License (Declared)	Begin typing to view possible declared license values.
License Family (Declared)	Select one of the following KB license families (Permissive, Reciprocal, Weak Reciprocal, AGPL, or Unknown) or a custom license family for the declared license.
License Status (Declared)	<p>Select one of the following values:</p> <ul style="list-style-type: none"> Unreviewed In Review Reviewed Approved Limited Approval Rejected Deprecated
License Expiration Date (Declared)	Select an expiration date for the declared license.
License Expiration Date Comparison (Declared)	<p>Use this condition to compare the declared license expiration date of a component to the project version release date. Specify a number which equals the number of days to <i>add to</i> the project version release date for the comparison. Black Duck triggers a policy violation if the date is less than or greater than that date.</p> <ul style="list-style-type: none"> Use 'before' to trigger a policy violation when the license expiration date is more than X number of days before (or less than), the project version release date. Use 'after' to trigger a policy violation when the license expiration date is more than X number of day after (or greater than), the project version release date. <p>The following are examples using 'before.' The project version release date is the 10th.</p> <ul style="list-style-type: none"> Number of days = 0: triggers a policy violation when the license expiration date is the day before the project version release date (the 9th) or earlier. Number of days = 1: triggers a policy violation when the license expiration date is the same day as the project version release date (the 10th) or earlier. Number of days = 2: triggers a policy violation when the license expiration date is the 11th or earlier. Number of days = -2: triggers a policy violation when the license expiration date is the 7th and earlier. <p>The following are examples using 'after.' The project version release date is the 10th.</p> <ul style="list-style-type: none"> Number of days = 0: triggers a policy violation when the license expiration date is the day after the project version release date (the 11th) or later.

Component Condition	Value
	<ul style="list-style-type: none"> Number of days = -1: triggers a policy violation when the license expiration date is the same as the project version release date (the 10th) and later. Number of days = 2: triggers a policy violation when the license expiration date is the 13th and later. Number of days is -2: triggers a policy violation when the license expiration date is the 9th and later.
License (Deep License)	Begin typing to view possible deep (embedded) license values.
License Family (Deep License)	Select one of the following KB license families (Permissive, Reciprocal, Weak Reciprocal, AGPL, or Unknown) or a custom license family for the deep (embedded) license .
License Status (Deep License)	Select one of the following values for the deep (embedded) license : <ul style="list-style-type: none"> Unreviewed In Review Reviewed Approved Limited Approval Rejected Deprecated
License Expiration Date (Deep License)	Select an expiration date for the deep (embedded) license .
License Expiration Date Comparison (Deep License)	<p>Use this condition to compare the deep license expiration date of a component to the project version release date. Specify a number which equals the number of days to <i>add to</i> the project version release date for the comparison. Black Duck triggers a policy violation if the date is less than or greater than that date.</p> <ul style="list-style-type: none"> Use 'before' to trigger a policy violation when the license expiration date is more than X number of days before, or less than, the project version release date. Use 'after' to trigger a policy violation when the license expiration date is more than X number of day after, or greater than, the project version release date. <p>The following are examples using 'before.' The project version release date is the 10th.</p> <ul style="list-style-type: none"> Number of days = 0: triggers a policy violation when the license expiration date is the day before the project version release date (the 9th) or earlier. Number of days = 1: triggers a policy violation when the license expiration date is the same day as the project version release date (the 10th) or earlier. Number of days = 2: triggers a policy violation when the license expiration date is the 11th or earlier. Number of days = -2: triggers a policy violation when the license expiration date is the 7th and earlier. <p>The following are examples using 'after.' The project version release date is the 10th.</p>

Component Condition	Value
	<ul style="list-style-type: none"> Number of days = 0: triggers a policy violation when the license expiration date is the day after the project version release date (the 11th) or later. Number of days = -1: triggers a policy violation when the license expiration date is the same as the project version release date (the 10th) and later. Number of days = 2: triggers a policy violation when the license expiration date is the 13th and later. Number of days is -2: triggers a policy violation when the license expiration date is the 9th and later.

Table 15: BOM Component Custom Fields

Component Condition	Value
BOM Component Custom Field Name	Available for Boolean, Date, Drop Down, Multiple Selections, Single Selection, and Text field types. Select a value.

Table 16: Component Custom Fields

Component Condition	Value
Component Custom Field Name	Available for Boolean, Date, Drop Down, Multiple Selections, Single Selection, and Text field types. Select a value.

Table 17: Component Version Custom Fields

Component Condition	Value
Component Version Custom Field Name	Available for Boolean, Date, Drop Down, Multiple Selections, Single Selection, and Text field types. Select a value.

The table below shows the vulnerability attributes that you can select and the values you can specify.

Vulnerability conditions

Vulnerability condition	Value
Overall Score	Enter a number from 0 to 10.
Vulnerability IDs	Enter a specific vulnerability ID (CVE or BDSA).
CWE IDs	Enter a Common Weakness Enumeration (CWE) number.
Solution Available	Select either Yes or No. Indicates whether there is a solution for the vulnerability.
Workaround Available	Select either Yes or No. Indicates whether there is a workaround available for the vulnerability.
Exploit Available	Select either Yes or No.

Vulnerability condition	Value
	Indicates whether there is an exploit for the vulnerability.
Reachable from Source	Select either Yes or No. Indicates whether the vulnerability is reachable from the source code .
Remediation Status	Select one or more of the following values: <ul style="list-style-type: none"> • Duplicate • Ignored • Mitigated • Needs Review • New • Patched • Remediation Complete • Remediation Required
Vulnerability Tags	Select one or more of the following values: <ul style="list-style-type: none"> • AI Assisted • Zero-click Remote Code Execution • Malicious Code Identified • Embargoed Vulnerability Details • Unconfirmed Vulnerability • Automated Security Advisory • CISA Known Exploited Vulnerability

Creating policy rules for approved or barred items

You can create policy rules that enforce your company's policy of approved or barred items. For example, you can create a policy rule to:

- pre-approve a component version in your BOM: any component version that does not match your approval list triggers a policy violation.
- bar a component version from your BOM: a policy violation is automatically triggered for any component version that matches your list of barred components.

Pre-approved policy rule examples

Suppose you want to create a policy rule whereby externally distributed projects with permissive licenses are pre-approved: any component versions that have non-permissive licenses will trigger a policy violation.

To create this policy rule, follow the instructions for [creating a policy rule](#), and set these conditions:

Project Distribution Type ▼

equals ▼

External ▼

🗑️

[+ Add Filter](#)

Policy Rules

License Family ▼

not equal to ▼

Permissive ▼

🗑️

[+ Add Rule](#)

Suppose you want to create a policy rule whereby only a specific version of a component is approved: all other component versions trigger a policy violation.

To create this policy rule, follow the instructions for [creating a policy rule](#), and set these conditions:

Policy Rules

Component	equals	Apache Tomcat	Any Version	
Component	not in	Apache Tomcat	8.0.1	

[+ Add Rule](#)

In this example, a policy violation is triggered when the Apache Tomcat version is not 8.0.1.

Suppose you want to create a policy rule whereby multiple versions of a component are approved: all other component versions trigger a policy violation.

To create this policy rule, follow the instructions for [creating a policy rule](#), and set these conditions:

Component	equals	Apache Tomcat	Any Version	
Component	not in	Apache Tomcat	8.0.3	

Apache Tomcat 8.0.1

 Apache Tomcat 8.0.3

[+ Add Rule](#)

In this example, a policy violation is triggered when the Apache Tomcat version is not 8.0.1 or 8.0.3.

To create this condition:



1. Select the component, the equals operator, and the component.
2. For the second condition: select the component, the 'not in' operator, and the approved versions.
To select multiple versions, select the version and click **Set selected component**, Repeat selecting approved versions and clicking **Set selected component** until all approved versions are selected.

Barred policy rule example

Suppose you want to create a policy rule whereby any component versions in SaaS distributed projects in the development or planning phase with licenses in the AGPL license family trigger a policy violation.


To create this policy rule, follow the instructions for [creating a policy rule](#), and set these conditions:

Projects All Filtered

Project Distribution Type	equals	SaaS	
Project Phase	equals	In Development In Planning	

[+ Add Filter](#)

Policy Rules

License Family	equals	AGPL	
----------------	--------	------	---




[+ Add Rule](#)

Editing a policy rule

Users with the Policy Manager [role](#) can edit policy rules.

After you edit a policy rule, BOMs are evaluated to determine if they are in violation of the edited rule.




To edit a policy:

1.  Click **Manage** > **Policies**.
The Policy Management page appears.
2.  Click  in the row of the policy you want to edit and select **Edit** to display the Edit Policy Rule dialog box.
3. Edit the policy and click **Update**.

Copying a policy rule

Users with the Policy Manager [role](#) can copy policy rules.

To copy a policy:

1.  Click **Manage** > **Policies**.
The Policy Management page appears.
2.  Click  in the row of the policy you want to copy and select **Copy**.
3. Add the information for this policy and click **Create**.
The policy name is the only required field.

Deleting a policy rule

Users with the Policy Manager [role](#) can delete policy rules.

Violations are removed for any component that was in violation of the deleted policy rule.

To delete a policy:

1. Log in to Black Duck as a user with the Policy Manager or Sysadmin role.

- 2.



Click **Manage** > **Policies**.

The Policy Management page appears.

- 3.

Click in the row of the policy you want to delete and select **Delete**.

4. When prompted, click **Delete** to confirm.

Disabling or enabling a policy rule

Users with the Policy Manager [role](#) can disable or enable policy rules.

- When a rule is disabled, violations are removed for any component that was in violation of the policy rule (if the rule was previously enabled).
- When a rule is enabled, existing BOMs are immediately evaluated to determine if they are in violation of this rule.

To disable or enable a policy:

- 1.



Click **Manage** > **Policies**.

The Policy Management page appears.

- 2.

Click in the row of the policy rule that you want to enable or disable and select **Edit**.


3. Do one of the following:

- Clear the **Enabled** option to disable the rule.
- Select the **Enabled** option to enable the rule.

4. Click **Update**.

Overriding policy violations

If a rule was configured to allow manual overrides of violations, then users with the appropriate [role](#) can override a disapproved component or file in that project.

 **Note:** If you override a file, the component will still be in violation if at least one file in the component is in violation of a policy.

To override a violation:

1. On the BOM page using the List view, click the Policy Violation icon (Ⓢ) of the component you wish to override. The Policy Violations dialog box appears.

The screenshot shows a dialog box titled "Policy Violations" for the component "jackson-databind 2.9.6". The dialog contains the following information:

- Title:** Policy Violations
- Component:** jackson-databind 2.9.6
- Policy Rule:** No External Tier 0, Tier 1 or Tier 2 Projects With More Than 1 High Vulnerability
- Severity:** Unspecified
- Category:** Uncategorized
- Scan Modes:** Full
- Found:** Sep 29, 2022
- Comment:** (Text input field)
- Expiration:** Never (Date picker)
- Description:** Disallow External Projects With More Than 1 High Vulnerability at Tier 0, Tier 1 or Tier 2
- Condition:**
 - Project Distribution Type EQUALS External
 - Project Tier LESS THAN 3
 - High Severity Vulnerability Count GREATER THAN 1
- Buttons:** Override, Override All, Close

2. Depending on whether there is one or more policy violation:

- For one policy violation, click **Override**. Optionally, enter a comment and click **Confirm**.

If you entered a comment, it appears, along with the username of the user who overrode the policy violation, in the Policy Violations dialog box.

If you enter a date, the override will expire at that date. When it expires it will return to a violation state.

- For multiple policy violations:

- Click **Override All** to override all policy violations. The Policy Violations dialog box displays the username of the user who overrode the policy rule.


You cannot enter a comment or date when using the **Override All** feature.

- Click **Override** for each policy violation you want to override. Optionally, enter a comment and click **Confirm**.

If you entered a comment, it appears, along with the username of the user who overrode the policy violation, in the Policy Violations dialog box.

3. Click **Close**.

The Policy Violation Override icon (Ⓢ) appears next to the component that you overrode if all policy violations were overridden. If a component has multiple policy violations and not all are overridden, then the Policy Violation icon (Ⓢ) will still appear.

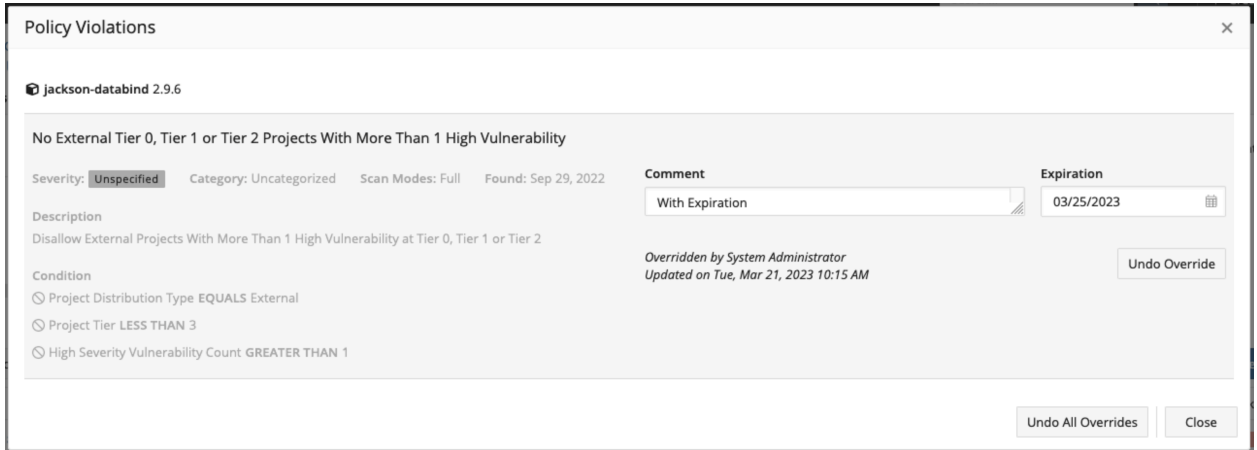
 **Note:** Overrides can be [removed](#).

Removing policy overrides

You can remove an override of a component or file that was in violation of a policy rule. Only users with the appropriate [role](#) can override a disapproved component or file in that project.

To remove an override:

1. On the BOM page using the List view, click the Policy Violation Override icon (Ⓢ) located next to the component. The Policy Violations dialog box appears.



2. Depending on whether there is one or more policy override to remove:

- To remove one policy override, click **Undo Override**. Optionally, enter a comment and click **Confirm**.

If you entered a comment, it appears, along with the username of the user who removed the override, in the Policy Violations dialog box.

- For multiple policy violations:

- Click **Undo All Overrides** to remove all policy overrides.

You cannot enter a comment when using the **Undo All Overrides** feature.

- Click **Undo Override** for each policy violation you want to override. Optionally, enter a comment and click **Confirm**.

If you entered a comment, it appears, along with the username of the user who removed the override, in the Policy Violations dialog box.

3. Click **Close**. The BOM appears and the Policy Violation icon (🚫) reappears.

12. Managing open source licenses

The use of open source software (OSS) is managed through licenses that allow you to use, modify, and/or share the software under defined terms and conditions. The conditions regarding the reuse of open source software can vary from things you can do (rights), things you cannot do (restrictions) and things you must do (obligations) in order to comply with the license.

Best practices for the redistribution of open source software include identifying all OSS content in the distribution and ensuring compliance to licensing obligations. Virtually all open source licenses contain an attribution clause as part of the licensing obligation. The attribution clause requires that the source of the software, and generally the copyright holder, be identified. Compliance with the attribution clause of these licenses generally takes the form of an attribution document, sometimes called a Notices File, which lists all OSS and the appropriate copyright and license information.

With Black Duck, you can create accurate and compliant open source notice file reports at a project/release level. Black Duck provides the actual license text for the MIT, variants of the BSD, and the ISC licenses, which are the top components in our KnowledgeBase, based upon customer usage.

For example, the following is an HTML version of the Notices File report from Black Duck:

Sample Project A - 4.0 Notices File

Phase: In Planning | Distribution: External

Notices Report Content

- License Data
- License Text
- Origin Copyright Text

Components

Component	License	Component Link
Apache Log4j API 2.17.1	Apache License 2.0	http://logging.apache.org/log4j/2.x/log4j-api/
Apache Tomcat 10.0.20	Apache License 2.0	http://tomcat.apache.org/

Copyright Data

Apache Log4j API 2.17.1 - maven:org.apache.logging.log4j:log4j-api:2.17.1
<http://logging.apache.org/log4j/2.x/log4j-api/>

- Copyright 1969-1999 The Apache Software Foundation

Apache Tomcat 10.0.20 - maven:org.apache.tomcat:tomcat-jasper-el:10.0.20
<http://tomcat.apache.org/>

- Copyright 1999-2022 The Apache Software Foundation This product includes software developed at

Licenses

Apache License 2.0

Apache Log4j API 2.17.1, Apache Tomcat 10.0.20

```

Apache License
Version 2.0, January 2004
*****

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licenser" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.

```

You can edit and maintain the data needed to create this report. The notice files can then be included with the distribution or incorporated into documentation to satisfy the attribution obligation.

Suggested work flow

To manage component licenses using Black Duck:

1. With the assistance of your legal counsel, determine the best combination of licenses for your company's work. This planning work can help you determine whether you need to make changes to a BOM to bring a project into compliance.
2. Use the [License Management page](#) to view licenses currently used by your company and existing [license families](#).
 - If a component uses a license that is not available from Black Duck KnowledgeBase, users with the License Manager role can [create custom licenses](#) or [edit KnowledgeBase licenses](#).
 - If a license family does not accurately reflect your license risk, users with the License Manager role can [create custom license families](#).
 - If a license term does not accurately reflect a license obligation, users with the License Manager role can [manage license terms](#) of their custom or KnowledgeBase licenses.
3. [Create policy rules](#) that trigger violations when components do not comply with your license policies.
4. Review the BOM for any license policy violations and determine what to do with components that are in violation of a rule.
5. Determine whether you want to enable [deep license data](#).
6. Review the BOM for license accuracy:
 - Research components that have Unknown License or License Not Found values.
 - Review components that have [license risk](#). Confirm that the [usage](#) of the component is correct as the combination of project distribution, usage, and license determines the license risk.
 - For components that have disjunction (OR) licenses, investigate and decide which license you plan to use.
7. [Review copyright statements](#) and/or review [detected copyright statements](#). Optionally edit the Black Duck KnowledgeBase copyright statements and/or create custom copyright statements.
8. Create the [Notices File report](#). Optionally, make these modifications to the report contents:
 - Determine if any components or subprojects should be [excluded from the report](#).
 - [Add attribution statements](#).
 - [Edit the license text](#) if necessary.

About license families

The use of open source software is managed through licenses that allow the software to be utilized, modified, and/or shared under defined terms and conditions. The conditions regarding the reuse of open source software can vary from things you can do (rights), things you cannot do (restrictions) and things you must do (obligations) in order to comply with the license. Black Duck tracks over 2000 open sources licenses that can range from those with few restrictions and obligations to those with many restrictions and obligations.

Depending upon the nature of these restrictions and obligations, some licenses are deemed to be riskier than others, as they require more management and care to ensure compliance with the license terms. Typically, the riskiest licenses are those that are reciprocal in nature. Reciprocal licenses, often pejoratively

called “Viral Licenses”, are those in which the license terms can extend beyond the open source code itself and can try to apply to other code as well. The other code could be modifications to the open source, or even simply code that uses the open source code in a way that triggers the reciprocal nature of the licenses. Once triggered, it is possible that in order to be in compliance to the license, developers who create software applications may need to treat the entire application as under the open source license and comply with all these obligations for the entire application. This could include the obligation to provide all the source code for the application (not just the open source) and allowing people who receive the application to modify and redistribute it without restrictions. This may be in conflict with a proprietary license model.

Please note, the legal aspects of managing open source licenses can be complicated and often it is best to seek legal counsel when making decisions about open source licenses and creating policies regarding their use. Legal counsel can best help determine if the license rights, restrictions, and obligations apply in a particular scenario. However, in order to help customers manage these risks in a simple and effective way, Black Duck categorizes open source licenses into license families for purposes of risk calculations and the definition of open source policy rules. These families range from those that are highly reciprocal to those with few obligations and restrictions. These license families, called KnowledgeBase licenses are:

- **Affero General Public License (AGPL)**

Licenses in the AGPL family tend to be highly reciprocal. The reciprocity can be easily triggered depending upon how the component is incorporated into the overall body of work and how much the original work is based upon the open source code. In addition, the obligations can apply when software is exposed over a network (for example, the internet). Companies who distribute software applications (either on a device or as media/downloads) or create software as a service (SaaS) applications need to pay particular attention to software under these licenses in order to ensure compliance.

- **Reciprocal**

Reciprocal licenses are those in which the license terms can easily apply to the overall body of work (like the AGPL) depending upon how it is used. However, typically the reciprocal nature of the license is triggered by distribution. Therefore, companies who distribute software in some fashion are generally concerned with highly managing software under these types of licenses.

- **Weak Reciprocal**

Licenses in this family can be reciprocal, but they are intended for open source software that is expected to be combined with other software under other licenses and therefore they tend to have a smaller reach. In this case, depending upon how the software is used, the reciprocal nature may simply cover modifications to the OSS and do not necessarily apply to the whole body of work. Companies who distribute software generally need to be keenly aware of these licenses, but tend to allow usage of components under these licenses with guidelines as to how they can be used. Staying in compliance and not triggering the reciprocity of the license tends to be easier.

- **Restrictive Third Party Proprietary**

Licenses in the Restrictive Third Party Proprietary family are for the licenses which cover other company’s commercial proprietary code. Typically Restrictive Third Party Proprietary licenses have restrictions on the use of the code and can be risky.

- **Permissive**

Permissive Licenses tend to not place restrictions on the use of the open source code and generally have few obligations. Companies, for the most part, view these licenses as easy to manage and non-risky.

- **Internal Proprietary**

Licenses in the Internal Proprietary family are typically for your licenses which are used to cover your company-owned proprietary software. Licenses in this family tend to not place restrictions on your use of the code and are generally not very risky when you use code with licenses in this family.

- Unknown

In this case, Black Duck was unable to determine the license for a component. Additional review should be done to determine the license for this component.

The following table shows the license family for the top 20 open source licenses used in open source projects:

License Family	Examples
Affero General Public License (AGPL)	<ul style="list-style-type: none"> • GNU Affero General Public License v3 or later
Reciprocal	<ul style="list-style-type: none"> • GNU General Public License (GPL) 2.0 or 3.0 • Sun GPL with Classpath Exception v2.0
Weak Reciprocal	<ul style="list-style-type: none"> • Code Project Open License 1.02 • Common Development and Distribution License (CDDL) 1.0 or 1.1 • Eclipse Public License • GNU Lesser General Public License (LGPL) 2.1 or 3.0 • Microsoft Reciprocal License • Mozilla Public License
Permissive	<ul style="list-style-type: none"> • Apache 2.0 • Artistic License • BSD License 2.0 (2-clause Simplified, 3-clause, New, or Revised) • Do What The F*ck You Want To Public License • ISC License • Microsoft Public License • MIT License • Zlib-Libpng License
Unknown	N/A

Managing license families

Users with the License Manager [role](#) can use the License Management page to manage their license families.

From this page you can view the KnowledgeBase license families or create [custom license families](#).

To view the License Management page:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.

3. Select the **License Families** tab to view a table which lists all license families.

12. Managing open source licenses • About license families

License Management

Licenses | License Families | License Terms

+ Create License Family

Filter license families...

Add Filter ▾

License Family	Source ▾	Last Updated
Internal Proprietary	KnowledgeBase	Never
Permissive	KnowledgeBase	Never
Reciprocal	KnowledgeBase	Never
AGPL	KnowledgeBase	Never
Restricted Third Party Proprietary	KnowledgeBase	Never
test	Custom	Aug 5, 2019 by System Administrator ▾
Unknown	KnowledgeBase	Never
Weak Reciprocal	KnowledgeBase	Never

Displaying 1-8 of 8

The table provides the following information:

Column	Description
--------	-------------

License Family The license family for this license.
 Select a license family to view a definition and risk profile for that license family:

License Family
×

Reciprocal

Reciprocal licenses are those in which the license terms can easily and broadly apply to the overall body of work (including code covered under different licenses) upon distribution, depending upon how it is used within the overall body of work.

[Learn More](#)

Risk Profile

License risk is determined by the component usage and its distribution.

Component Usage	Distribution			
	External	SaaS	Internal	Open Source
Source Code	High	Low	None	None
Statically Linked	High	Low	None	None
Dynamically Linked	High	Low	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	Medium	None	None	None
Dev. Tool / Excluded	None	None	None	None
Unspecified	High	Low	None	None

Displaying 1-9 of 9

Close

Source Source for this license. Possible values are:

- KnowledgeBase. From Black Duck KnowledgeBase.
- Custom. [Custom license family](#).

Column	Description
Last Updated	Date that the license family was created or last updated and the username of the user who created or last updated this license family.

Use the filter to limit the information shown on this page. You can filter by:

- License Family Source: Custom or KnowledgeBase.


About custom license families

If you discover that a KnowledgeBase license family does not accurately reflect your license risk, License Managers – users with the License Manager [role](#) – can create and manage custom license families. These custom license families can then be selected for a custom license which can then be assigned to custom components. This ensures that BOMs accurately show your license risk.

Custom license families:

- Consist of a name, a risk profile and optionally, a description.
- Can be assigned to a [custom license](#).
- Can be used to [create policy rules](#).
- Use a combination of component usage and distribution to determine [license risk](#).

License Managers can use the **License Families** tab in [License Management](#) to [create](#), [edit](#), and [delete](#) custom license families.

 **Note:** If your License Manager created a custom license family labeled "Restrictive Third Party Proprietary" or "Internal Proprietary" before the 2019.10.0 release, the number "(1)" is appended to those custom license family names.

Creating custom license families

Only users with the License Manager role can create [custom license families](#).

To create a custom license family:

1. Log in to Black Duck with the License Manager [role](#).

2.



Click **Manage** > **Licenses**.

The License Management page appears.

Select the **License Families** tab to display all license families.

12. Managing open source licenses • About license families

The screenshot shows the 'License Management' interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there is a '+ Create License Family' button and a search filter 'Filter license families...' with an 'Add Filter' button. The main content is a table with the following data:

License Family	Source	Last Updated
Internal Proprietary	KnowledgeBase	Never
Permissive	KnowledgeBase	Never
Reciprocal	KnowledgeBase	Never
AGPL	KnowledgeBase	Never
Restricted Third Party Proprietary	KnowledgeBase	Never
test	Custom	Aug 5, 2019 by System Administrator
Unknown	KnowledgeBase	Never
Weak Reciprocal	KnowledgeBase	Never

At the bottom right of the table area, it says 'Displaying 1-8 of 8'.

3. Click **Create License Family** to open the Create a Custom License Family dialog box.

The screenshot shows the 'Create a Custom License Family' dialog box. It has a close button (X) in the top right corner. The form contains the following fields:

- Name ***: A text input field.
- Description**: A larger text area.
- Risk Profile**: A section with the text 'Configure the risk based on usage and distribution. You can copy the profile from an existing family and change it.' Below this is a 'Copy from' dropdown menu.

Below the 'Copy from' dropdown is a table for configuring risk values:


Component Usage	External	SaaS	Internal	Open Source
Source Code	None	None	None	None
Statically Linked	None	None	None	None
Dynamically Linked	None	None	None	None
Separate Work	None	None	None	None
Implementation of Standard	None	None	None	None
Dev. Tool / Excluded	None	None	None	None

At the bottom right of the dialog box, there is a 'Save' button.

4. Enter a name for this license family.
5. Optionally, enter a description.
6. Optionally, modify the license risk values. By default, the **license risk** is None for all usages and distributions. You can select a license family to use as a baseline for the license risk by selecting one from the **Copy from** list. You can then use these license risk values for the custom license family or modify the values by using the drop downs to modify the license risk by usage and distribution. Possible license risk values are: none, low, medium, and high.
7. Click **Save**.

Editing custom license families

Custom license families can be edited by users with the License Manager [role](#).

 **Note:** Adjusting the risk profile for a license family will not change the calculated license risk for components on existing BOMs. Changes will only be reflected on project versions when the BOM is recalculated, such as during rescans or when assigning a scan to a project version.

To edit a custom license family:

1. Log in to Black Duck with the License Manager role.

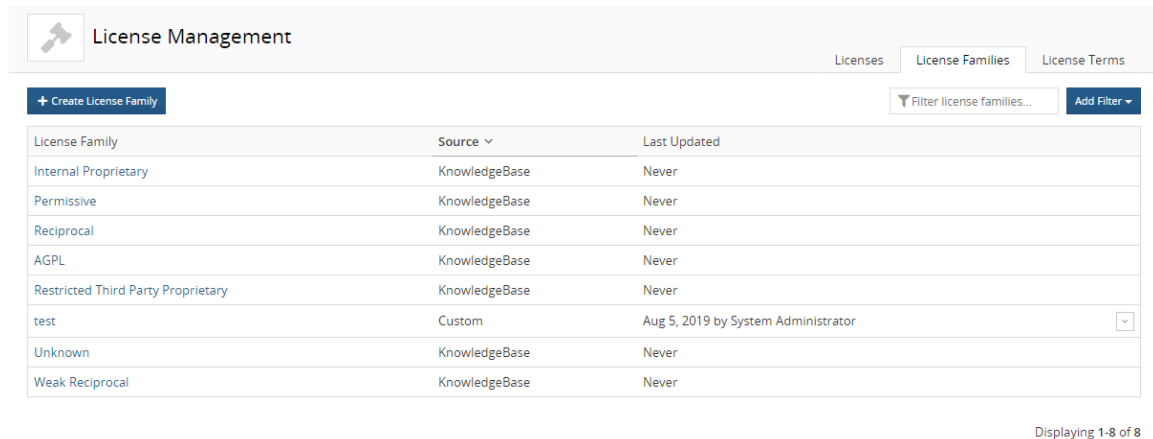
2.



Click **Manage** > **Licenses**.

The License Management page appears.

3. Select the **License family** tab to display all license families.




The screenshot shows the License Management page with the 'License Families' tab selected. The page includes a '+ Create License Family' button, a search filter, and a table of license families.

License Family	Source	Last Updated
Internal Proprietary	KnowledgeBase	Never
Permissive	KnowledgeBase	Never
Reciprocal	KnowledgeBase	Never
AGPL	KnowledgeBase	Never
Restricted Third Party Proprietary	KnowledgeBase	Never
test	Custom	Aug 5, 2019 by System Administrator
Unknown	KnowledgeBase	Never
Weak Reciprocal	KnowledgeBase	Never

Displaying 1-8 of 8

4.

Select the custom license family name or click  and select **Edit** in the row of the custom license family that you want to edit to display the Edit Custom License Family dialog box.

Edit Custom License Family
✕

Name *

Description

Component Usage	External	SaaS	Internal	Open Source
Source Code	High ▼	Medium ▼	None ▼	None ▼
Statically Linked	High ▼	Medium ▼	None ▼	None ▼
Dynamically Linked	High ▼	Medium ▼	None ▼	None ▼
Separate Work	None ▼	None ▼	None ▼	None ▼
Implementation of Standard	None ▼	None ▼	None ▼	None ▼
Dev. Tool / Excluded	None ▼	None ▼	None ▼	None ▼

5. Modify the information shown for this custom license family.
6. Click **Save** in the Edit Custom License dialog box. The username of the user who edited this license family and the date appears in the **Last Updated** column.


Deleting custom license families

You cannot delete a license family that is being used by a license in a BOM.

You also cannot delete licenses provided by Black Duck KnowledgeBase.

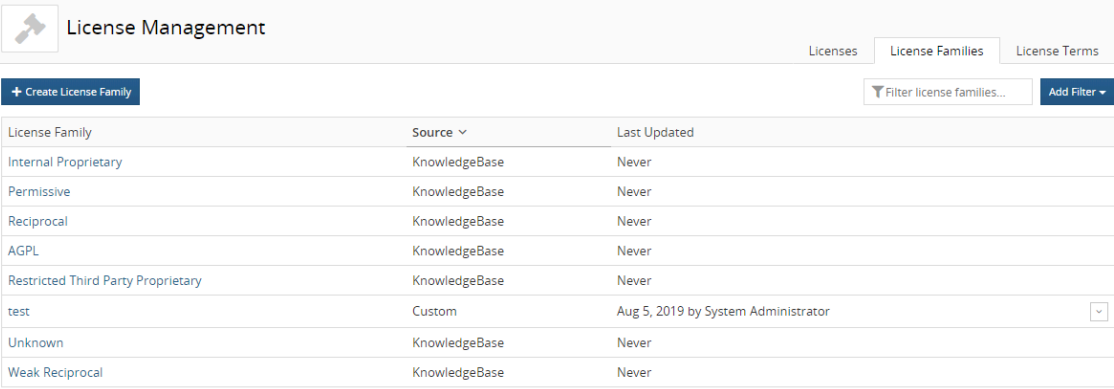
1. Log in to Black Duck with the License Manager [role](#).
- 2.



Click  > **Licenses**.

The License Management page appears.


3. Select the **License Family** tab to display all license families.



The screenshot shows the 'License Management' page with the 'License Families' tab selected. A table lists various license families with columns for 'License Family', 'Source', and 'Last Updated'. The 'test' row is highlighted, indicating it is a custom license family.

License Family	Source	Last Updated
Internal Proprietary	KnowledgeBase	Never
Permissive	KnowledgeBase	Never
Reciprocal	KnowledgeBase	Never
AGPL	KnowledgeBase	Never
Restricted Third Party Proprietary	KnowledgeBase	Never
test	Custom	Aug 5, 2019 by System Administrator
Unknown	KnowledgeBase	Never
Weak Reciprocal	KnowledgeBase	Never


Displaying 1-8 of 8

- Click  and select **Delete** in the row of the custom license family that you want to delete to display a confirmation dialog box.
An error message appears if you try to delete a custom license family that is currently being used by a license.
- Click **Delete** to confirm.

Viewing licenses


The **Licenses** tab in the License Management page displays custom licenses you have created and the licenses from Black Duck KnowledgeBase that are used in all projects in your organization.

Users with the License Manager [role](#) can use the License Management page to manage licenses.

 **Note:** The License Manager role is intended to be a cross-project, enterprise role. Typically, attorneys or privileged users that have broad access to information would have this role. Therefore, License Managers can view the licenses for *all* projects, including projects in which they are not project members.

From this page, you can:

- [Create](#), [edit](#), or [delete](#) custom licenses.
- [Edit KnowledgeBase](#) licenses.
- [View the full text](#) of custom and Black Duck KnowledgeBase licenses.
- [View the number of components](#) in your projects that use a specific license.

 **Note:** Edits made locally by a BOM manager or Project Manager to the license text of a custom or KnowledgeBase license will not appear on this page.

To view the License Management page:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **License Management**.

The License Management page appears.

License Management

Licenses | License Families | License Terms

[+ Create License](#)

In Use x
Filter licenses... ▼
Add Filter ▼

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

Select the **Licenses** tab to view a table with the following information.

Note: Newly added Black Duck KnowledgeBase licenses or modifications made to Black Duck KnowledgeBase licenses may not be visible here for up to 30 minutes.

C Description

License name.

Select the name to display the *License Name* page. Use the:

- **Settings** tab to view information for this license, such as the [license family](#) and [license text](#).
- **License Terms** tab to view the terms for this license.
- **Where Used** tab to view the component and subproject versions where this license is used.

Components

Note: The value shown here does *not* include projects assigned with this custom license.

Select the component value to display a page which lists the component versions or subprojects [where this license is used](#).

License family

Select a license family to view a definition and risk profile for that license family:

C Description

License Family ×

Reciprocal
 Reciprocal licenses are those in which the license terms can easily and broadly apply to the overall body of work (including code covered under different licenses) upon distribution, depending upon how it is used within the overall body of work.
[Learn More](#)

Risk Profile
 License risk is determined by the component usage and its distribution.

Component Usage	Distribution			
	External	SaaS	Internal	Open Source
Source Code	High	Low	None	None
Statically Linked	High	Low	None	None
Dynamically Linked	High	Low	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	Medium	None	None	None
Dev. Tool / Excluded	None	None	None	None
Unspecified	High	Low	None	None

Displaying 1-9 of 9

Last Updated Date, if updated today, or date that the license was last updated.

User Username of the user who created or last updated the license.
 This field is empty for licenses from Black Duck KnowledgeBase that have not been edited.

Source Source for this license. Possible values are:

- KnowledgeBase. From Black Duck KnowledgeBase.
- Modified KnowledgeBase. An [edited Black Duck KnowledgeBase license](#).
- Custom. Custom license.

Status Review status for the license. Possible values are:

- Unreviewed
- In Review
- Reviewed
- Approved
- Limited Approval
- Rejected
- Deprecated


Use the filters to limit the information shown on this page. You can filter by:

- License Source: KnowledgeBase, Modified KnowledgeBase, or Custom.
- License Family: a KnowledgeBase license family or a custom license family.
- License Status.

- In Use. Only displays those licenses associated with a component version or subproject. This filter is selected by default.

Viewing license text

You can view the text of custom and KnowledgeBase licenses.

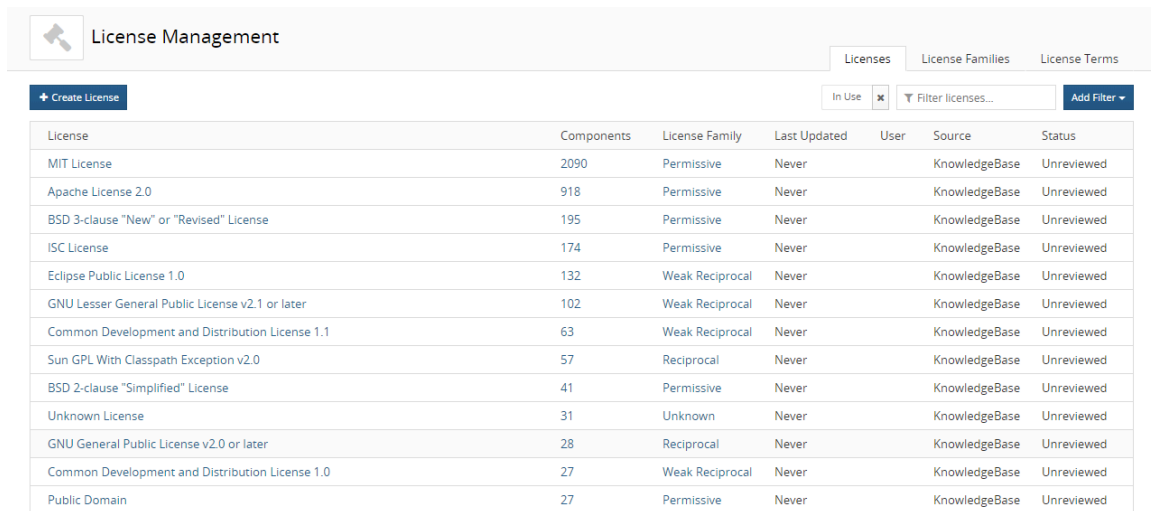
-  **Note:** For KnowledgeBase licenses, if the license is one that is modified for individual components (like the BSD or MIT license), then the template license text is shown here. However, when viewing the license text in the context of a component (such as viewing the component's license in a BOM), the actual license text for that component is shown.

To view license text:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.



License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. In the **Licenses** tab, select the license name to display the **License Name Settings** tab which displays the license text:

License Management
Apache License 2.0
 Family: Permissive | Status: Unreviewed

Settings

License Terms

Where Used

Name: Apache License 2.0

License Family: Permissive

Status: Unreviewed

Notes

Expiration Date

License Text: Apache License
Version 2.0, January 2004


Created: never
Updated: never

Save

With the appropriate [role](#), you can also [view the license text in a BOM](#).

Viewing license use

You can view the component and subproject versions where a specific license is used.

 **Note:** The information shown here lists the components and subprojects that use a license. It does not include licenses assigned to project versions.

To view where a license is used:

1. Log in to Black Duck with the License Manager role.
- 2.

Click  > **Licenses**.

The License Management page appears.

License Management

Licenses | License Families | License Terms

+ Create License

In Use x Filter licenses... Add Filter

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Select the license name to display the *License Name Settings* tab.

12. Managing open source licenses • Viewing licenses

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings
License Terms
Where Used

Settings

Name: Apache License 2.0

License Family: Permissive

Status: Unreviewed

Notes

Expiration Date

License Text: Apache License
Version 2.0, January 2004

Created never
Updated never

Save

4. Select the **Where Used** tab.

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings
License Terms
Where Used

Component	Used Count
@gulp-sourcemaps/map-sources 1.0.0	1
A Swiss Army Knife for OSGi (bndlib) 2.2.0	1
Abdera Bundle Jar 1.1.2	1
Abdera Client 1.1	1
Abdera Core 1.1	1
Abdera Extensions - GData 1.1.2	1
Abdera Extensions - Geo 1.1	1
Abdera Extensions - HTML 1.1	1
Abdera Extensions - JSON 1.1.3	1
Abdera Extensions - Main 1.1	1
Abdera Extensions - Media 1.1	1
Abdera Extensions - OpenSearch 1.1	1
Abdera Extensions - Serializer 1.1	1
Abdera Extensions - Sharing 1.1	1
Abdera Security 1.1.1	1
Abdera Server 1.1.3	1
Abdera Spring Integration 1.1	1
Acegi Security System for Spring 1.0.7	1
ActiveMQ Protocol Buffers Implementation and Compiler 1.1	2


- Select the component name to display the [Black Duck KB component page](#) which displays information about the component, such as a description, component links, and tags, and information about each of the component versions that are available in Black Duck KB.
- Select the component version to display the **Details** tab of the [Component Name Version page](#), which displays a list of the projects and project versions in which this version of the component is used.
- Select the subproject name to display the **Overview** tab of the [Project Name page](#) which project more information about this project.

- Select the subproject version to display the **Details** tab of the *Project Version* page to view [more information about this project version](#)

Determining license risk

License Risk is determined by the license risk of the components and subprojects in the project version's BOM.

There are four levels of overall license risk (high, medium, low, and none), based on the [license family](#) declared by the component, the type of distribution for the project (external, internal, SaaS, or open source) and the [usage](#) (statically linked, dynamically linked, source code, dev. tool/excluded, implementation of standard, merely aggregated, prerequisite, separate work, and unspecified).

 **Note:** Other licenses include "Unknown" which indicates that the OSS component version's license is not known; "License Not Found" which indicates that although researched by Black Duck, no declared license was found for the component; and "No License" which indicates that Black Duck found a declaration of 'No License' for the component.

These licenses are included in the Unknown license family in the tables below.


For components with multiple licenses:


- "AND" licenses: license risk is determined by the license with the highest risk.
- "OR" licenses: license risk is determined by the license with the lowest risk.

Risk calculations assume that your project is being distributed under a proprietary license.

Subproject license risk

If your project contains subprojects, the license risk is determined the subproject's license risk. A subproject's license is determined when it is [added to the project](#).

 **Notice:** Black Duck 2023.10.0 introduces Enhanced license risk aggregation as a Limited Customer Availability Feature which improves the way subproject risk is determined. When enabled, the License Risk displayed for a subproject in your project's BOM will be determined by the subproject's license risk and the highest license risk of its components which reduces the possibility that license risk could be missed when using subproject hierarchies.

 **Important:** When modifying a subproject's [distribution type](#) after it has been added to a project, the license risk of the parent project may not necessarily change to reflect the modification. The parent project's distribution takes precedence when calculating license risk.

Estimated licenses

A default license may be assigned to components with an unknown version found during a scan. This is an estimated license based on greatest number of times it shows up across the top 1,000 versions of the component.

When viewing the BOM for a project, components with unknown versions will have a question mark next to the component name.

12. Managing open source licenses • Determining license risk

The screenshot shows the Black Duck Projects interface for a project named 'NG Manual Test'. It displays three risk categories: Security Risk, License Risk, and Operational Risk. Each category has a bar chart showing the number of components at different risk levels. Below the charts is a table of components with columns for Component, Source, Match Type, Usage, License, Security Risk, and Operational Risk. Two components are listed: 'Apache Struts 2.3.9' and 'Apache Tomcat', both with 'Apache-2.0' licenses. A warning banner is visible below the table.

Clicking the license in the License column will open the [Modify License](#) window which will display the following warning banner:

Black Duck wasn't able to identify the component version, therefore this is an estimated license. It was defined based on popularity across multiple versions. For a more accurate result, manually specify a version for this component.

[Edit Component](#) [Learn More](#)

It is recommended that you review these components and manually specify a version for more accurate results.

Default license risk

The following tables show the license risk for the default (KnowledgeBase) license families. Users with the License Manager [role](#) can [create custom license families](#) and define the license risk by usage and distribution for those custom license families.

Note: If your License Manager created a custom license family labeled "Restrictive Third Party Proprietary" or "Internal Proprietary" before the 2019.10.0 release, the number "(1)" is appended to those custom license family names.

License risk - by usage

Statically linked

The following table lists the license risk when the component's usage is **Statically Linked**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	High	High	None	None
Reciprocal	High	Low	None	None
Weak Reciprocal	High	Low	None	None
Permissive	None	None	None	None

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	High	High	High	High

Dynamically linked

The following table lists the license risk when the component's usage is **Dynamically Linked**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	High	High	None	None
Reciprocal	High	Low	None	None
Weak Reciprocal	Medium	Low	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	High	High	High	High

Source code

The following table lists the license risk when the component's usage is **Source Code**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	High	High	None	None
Reciprocal	High	Low	None	None
Weak Reciprocal	High	Low	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Internal Proprietary	None	None	None	Medium
Unknown	High	High	High	High

Dev. tool / excluded

The following table lists the license risk when the component is not distributed with your product. (Usage value is **Dev. Tool / Excluded**).

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	None	None	None	None
Reciprocal	None	None	None	None
Weak Reciprocal	None	None	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Low	Low	Low	Low
Internal Proprietary	None	None	None	None
Unknown	None	None	None	None

Implementation of Standard

The following table lists the license risk when the component usage is **Implementation of Standard**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	None	None	None	None
Reciprocal	None	None	None	None
Weak Reciprocal	None	None	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Low	Low	Low	Low
Internal Proprietary	None	None	None	None

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Unknown	None	None	None	None

Separate Work

The following table lists the license risk when the component usage is **Separate Work**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	None	None	None	None
Reciprocal	None	None	None	None
Weak Reciprocal	None	None	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	None	None	None	None

Merely aggregated

The following table lists the license risk when the component's usage is **Merely aggregated**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	None	None	None	None
Reciprocal	None	None	None	None
Weak Reciprocal	None	None	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	Medium	Medium	Low	Low

Prerequisite

The following table lists the license risk when the component's usage is **Prerequisite**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	Medium	None	None	None
Reciprocal	Medium	None	None	None
Weak Reciprocal	Low	None	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	Medium	Medium	Low	Low

Unspecified

The following table lists the license risk when the component's usage is **Unspecified**.

License Family	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Affero General Public License (AGPL)	High	High	None	None
Reciprocal	High	Low	None	None
Weak Reciprocal	High	Low	None	None
Permissive	None	None	None	None
Restrictive Third Party Proprietary	Medium	Medium	Medium	High
Internal Proprietary	None	None	None	Medium
Unknown	High	High	High	High

License risk by license family

Affero General Public License (AGPL)

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	High	High	None	None
Statically Linked	High	High	None	None
Dynamically Linked	High	High	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	Medium	None	None	None
Dev. Tool/ Excluded	None	None	None	None
Unspecified	High	High	None	None

Reciprocal

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	High	Low	None	None
Statically Linked	High	Low	None	None
Dynamically Linked	High	Low	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	Medium	None	None	None
Dev. Tool/ Excluded	None	None	None	None
Unspecified	High	Low	None	None

Weak Reciprocal

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	High	Low	None	None
Statically Linked	High	Low	None	None
Dynamically Linked	Medium	Low	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	Low	None	None	None
Dev. Tool/ Excluded	None	None	None	None
Unspecified	High	Low	None	None

Permissive

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	None	None	None	None
Statically Linked	None	None	None	None
Dynamically Linked	None	None	None	None
Separate Work	None	None	None	None
Merely Aggregated	None	None	None	None
Implementation of Standard	None	None	None	None
Prerequisite	None	None	None	None
Dev. Tool/ Excluded	None	None	None	None
Unspecified	None	None	None	None

Restrictive Third Party Proprietary

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	Medium	Medium	Medium	High

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Statically Linked	Medium	Medium	Medium	High
Dynamically Linked	Medium	Medium	Medium	High
Separate Work	Medium	Medium	Medium	High
Merely Aggregated	Medium	Medium	Medium	High
Implementation of Standard	Low	Low	Low	Low
Prerequisite	Medium	Medium	Medium	High
Dev. Tool/ Excluded	Low	Low	Low	Low
Unspecified	Medium	Medium	Medium	High

Internal Proprietary

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	None	None	None	Medium
Statically Linked	None	None	None	Medium
Dynamically Linked	None	None	None	Medium
Separate Work	None	None	None	Medium
Merely Aggregated	None	None	None	Medium
Implementation of Standard	None	None	None	None
Prerequisite	None	None	None	Medium
Dev. Tool/ Excluded	None	None	None	None
Unspecified	None	None	None	Medium

Unknown


Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Source Code	High	High	High	High
Statically Linked	High	High	High	High
Dynamically Linked	High	High	High	High

Usage	External Projects	SaaS Projects	Internal Projects	Open Source Projects
Separate Work	None	None	None	None
Merely Aggregated	Medium	Medium	Low	Low
Implementation of Standard	None	None	None	None
Prerequisite	Medium	Medium	Low	Low
Dev. Tool/ Excluded	None	None	None	None
Unspecified	High	High	High	High

Managing deep license data


Black Duck displays declared licenses for the components in your BOM. However, deep licenses (also known as sub-licenses or embedded licenses) may also exist in your open source components. Managing this deep license data reduces the risk of license infringement and makes it easier to understand and report on deep licenses and their risks in the open source being used.

Deep license data is not enabled by default; you must enable including deep license data to your BOM components. Once enabled, any deep licenses, as determined by Black Duck KnowledgeBase, are automatically active.

 **Note:** Depending upon the number of components and number of deep licenses, enabling the viewing of deep license data can impact the BOM calculation scan time. Adding deep license data to your BOM can affect your license risk and can trigger policy violations.

To manage your deep license data:

1. Enable deep level license data. As this feature is enabled at the project level, deep license data will be enabled and active for all project versions.

In your project version BOM, the deep license data icon () identifies the components with deep level licenses.

2. View the deep license data. You can:

- Review the evidence as determined by Black Duck KnowledgeBase.

Evidence consists of the list of files and file content which you can view to confirm the inclusion of deep license data.

- Activate or deactivate the license. By default, deep license data is activated for all origins. If there are multiple origins, deep license data is activated for all origins.
- Add licenses.
- Read the license text.

Enabling or disabling deep license data

Enabling this checkbox will apply deep license data to your non-snippet components matches and allow visibility to embedded licenses which may exist in your components beyond declared licenses. Deep license data is enabled at the project level.

Please note, this can affect the license risk and policy violation for components. It can also impact the Bill of Materials calculation time depending upon the number of components and amount of deep licenses.

To enable deep license data:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the **Settings** tab.
3. Click the **Project Details** tab.
4. Check the **Apply Deep License Data to Bill of Materials** checkbox or clear the checkbox to disable this feature.

Apply Deep License Data to Bill of Materials

Enabling this checkbox will apply deep license data to your components and allow visibility to embedded licenses which may exist in your components beyond declared licenses. Please note, this can affect the license risk and policy violation for components. It can also impact the Bill of Materials calculation time depending upon the number of components and amount of deep licenses.

5. Click **Save**.

Enabling or disabling deep license data to snippet component matches

If enabled, component snippet matches are included in the deep license data calculation.

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the **Settings** tab.
3. Click the **Project Details** tab.
4. Check the **Apply Deep License Data to Snippet Component Matches** checkbox or clear the checkbox to disable this feature.

Apply Deep License Data to Snippet Component Matches

If enabled, component snippet matches are included in the deep license data calculation.



5. Click **Save**.

Reviewing deep license data

1. Open the project version BOM to view the components which have deep license data.

12. Managing open source licenses • Managing deep license data

The screenshot shows the Black Duck Projects interface for a 'Sample Project 1.0'. It displays three risk categories: Security Risk, License Risk, and Operational Risk, each with a bar chart showing the distribution of components across risk levels. Below the charts is a table of components with columns for Component, Source, Match Type, Usage, License, Security Risk, and Operational Risk. The table lists several components, including 'abbrev 1.1.1', 'amdefine 1.0.1', 'ansi-regex 2.1.1', 'anymatch 1.3.2', 'Apache Commons FileUpload 1.1', 'Apache Commons Logging 1.0.4', 'Apache Lucene 1.4.3', and 'Apache Tomcat 6.0.26'. The 'Apache Tomcat 6.0.26' component is highlighted with a red 'High' risk label.

2. Components with  have deep license data. Click  to open the *Component Name Version Deep License* page.

The screenshot shows the 'Component Name Version Deep License' page for 'hh nodegoat demo 3.0-3 Apache Tomcat 6.0.26'. It displays the declared license as 'Apache License 2.0' and indicates '2 policy violations'. Below this is a table of licenses with columns for License, Active, License Family, Status, and Last Updated. The table lists several licenses, including 'Apache License 2.0', 'Bzip2 License', 'Common Public License 1.0', 'Eclipse Public License 1.0', 'libpng License', and 'zlib License'. The 'Apache License 2.0' license is highlighted with a red 'High' risk label.

This page displays the following information:

C Description

License

name.
Select the name to display the *License Name* page which displays the license text.
Click > to view the origins for this license.

Active

indicates whether this license is active.
Active licenses are included in the calculation of license risk and policy violations.

License Family

Last Updated

review status for the license. Possible values are:

- Unreviewed
- In Review

C Description

- Reviewed
- Approved
- Limited Approval
- Rejected
- Deprecated

3. From this page:

- View the evidence for the inclusion of this deep license.

The Black Duck KnowledgeBase determines deep license data at the origin level. Therefore, click > to display the origins for this license.

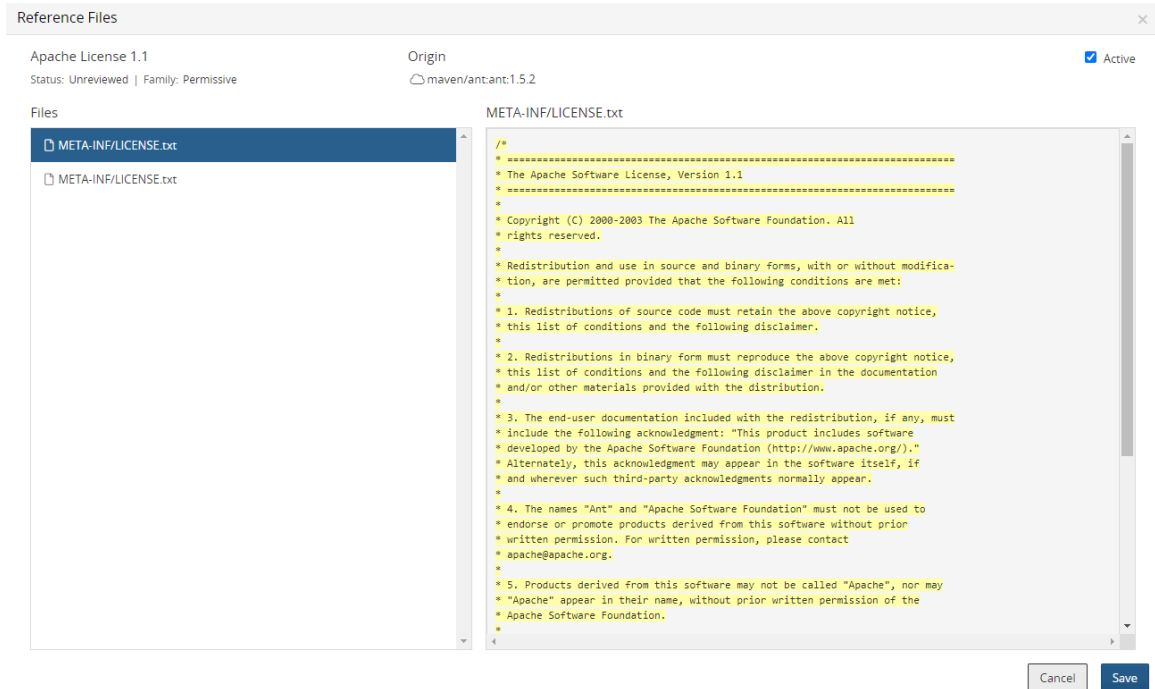
Select an origin to open the Reference Files dialog box which displays the files and corresponding evidence for inclusion of this license.

The screenshot shows a dialog box titled "Reference Files" with a close button (X) in the top right corner. The dialog is divided into three main sections:

- Header:** "Apache License 2.0" with status "Status: Approved | Family: Permissive" on the left, "Origin" with a location "maven/org.apache.tomcat:apache-tomcat:6.0.26" in the center, and an "Active" checkbox on the right.
- Files:** A list of files on the left, with "apache-tomcat-6.0.26/bin/jsvc.tar.gz/jsvc-src/native/debug.c" selected and highlighted in blue. Other files include "home.h", "arguments.c", "debug.h", "tomcat-native-1.1.20-src/jni/build.xml", "jsvc-unix.c", "buildconf.sh", "install.sh", "replace.h", "dso.h", "dso-dyld.c", "home.c", "dso-dlfcn.c", and "arguments.h".
- Content:** A text area on the right showing the content of the selected file. The content is a C header file for a debug utility, starting with a copyright notice for the Apache Software Foundation (2001-2004) and a license statement. It includes a URL to the Apache License (http://www.apache.org/licenses/LICENSE-2.0) and a block of C code for logging debug messages.

At the bottom of the dialog, there are three buttons: "Capture screenshot.", "Cancel", and "Save".



The **Files** section lists the files found containing deep license data. Select a file to view the contents of that file. Deep license data is highlighted.

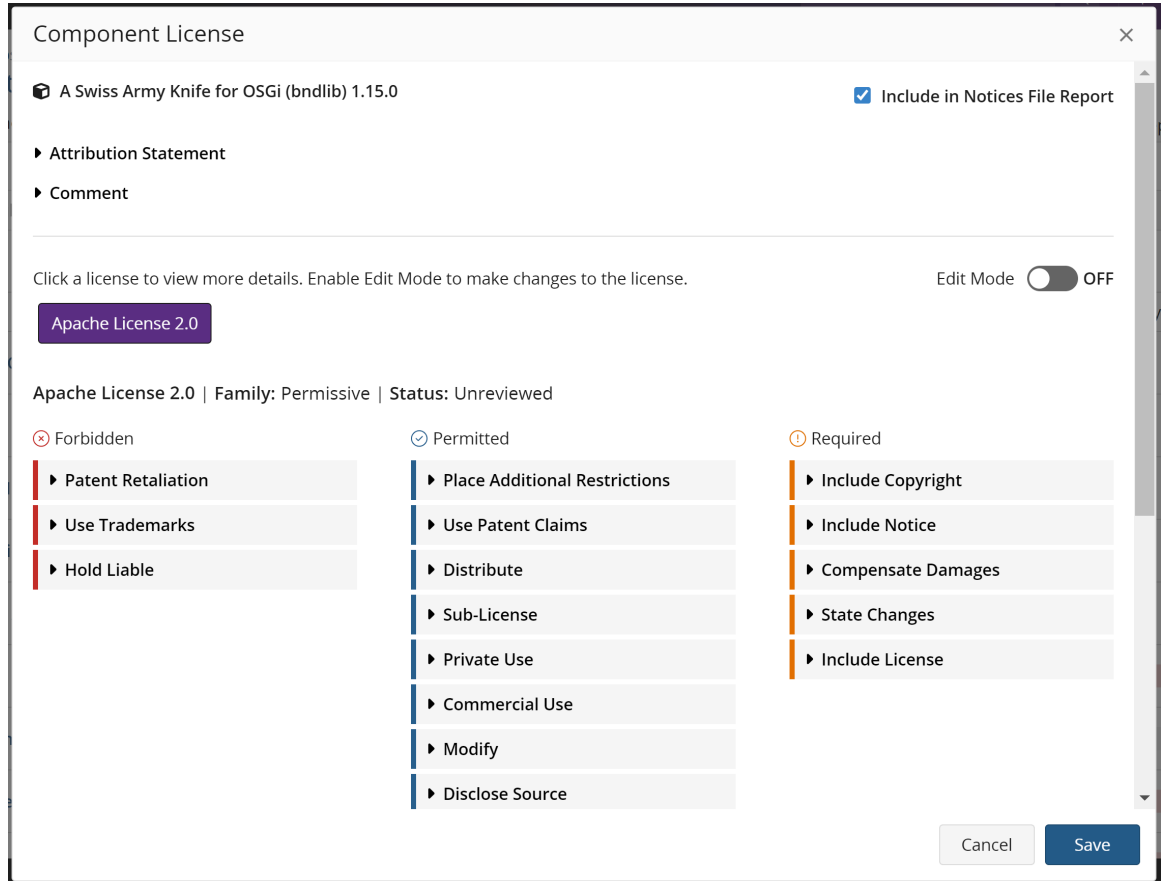


If the file cannot be determined, the file name and path display "Unknown."

- Activate or deactivate the deep license. By default, all deep licenses are active.

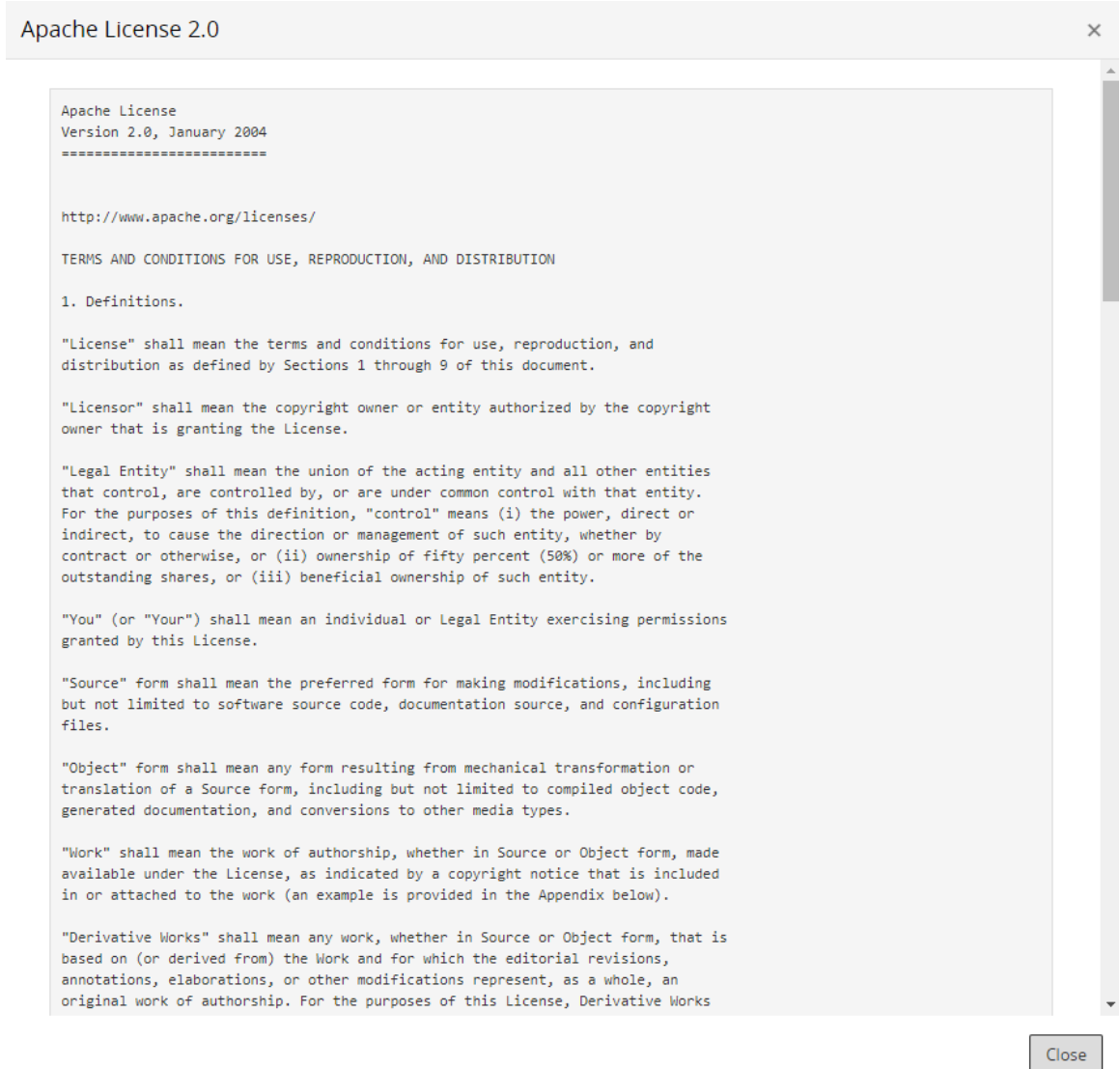
You can activate or deactivate a deep license by:

- Selecting a license in the *Component Name Version* Deep License page and clicking **Activate** or **Deactivate**.
- Selecting or clearing the **Active** option located in the upper right-corner of the Reference Files dialog box.
- Add a license or remove a manually added license.
 - To add a license, click **Add License**, select the license, and click . The new license appears in the table.
 - To remove a license, click  in the row of the manually added license you want to delete and select **Remove** in the confirmation dialog box.
- View license text:
 - View the declared license text and obligation information. Select the license name in the header to open the *Component Name Version* Component License dialog box.



Note that you can [modify the declared license](#).

- View deep license text by selecting the license name from the table.



Detecting embedded licenses

Black Duck can detect instances of embedded open source licenses not declared by Black Duck KnowledgeBase for a component.

By enabling detection of deep license data when scanning code, users focused on license compliance can view the licenses that were detected in their open source to ensure there are no problematic licenses and that all licenses are accounted for in their BOM.

With this feature, Black Duck performs a search for license string text and displays the licenses found in the **Source** tab.

By displaying this information in the **Source** tab, you can easily find the files and directories that interest you and determine if embedded licenses are located there.

The screenshot shows the Black Duck Projects interface for a project named "Sample Project 1.0". The left sidebar displays a file tree with folders like "external", "lib", "licenses", "new", "src", "src_joy", "src_surfaces", "src_pgsl", and "tools". The main content area is titled "src/# » src" and has tabs for "Files" and "Discoveries". Below the tabs, there are two sections: "License Searches" and "License References".

License Searches (221 Files):

License	Hits	Files
Apache License Version 2.0	224	219
Eclipse Public License Version 1.0	3	1
GNU General Public License Version 2	1	1

License References (6 Files):

License	Hits	Files
GNU General Public License	8	3
Apache License	5	2
BSD	5	1

Black Duck groups the detected licenses into one of two categories:

- **Licenses.** An exact match to a license and version.
- **License References.** A "fuzzy" match to a license; license version information was not found.

For each license statement found, Black Duck displays the number of:

- "Hits". The number of instances that license text was found in all files.
- Files where these "hits" were found.

In the example shown above, there were five instances of Apache License text found in two files, while there was 224 instances of Apache License Version 2.0 found in 219 files.

Black Duck also lists the total number of files affected for each category. Note that this value may not equal the total number of files shown for each license in that category as a file can have multiple different licenses, as shown above for the **Licenses** category.

Optionally, to help you review this information, upload your source files so that BOM reviewers can view discovered license text from within the **Source** tab. When source files are uploaded, Black Duck provides a list of embedded licenses and displays the highlighted license text in the file. This can help BOM reviewers evaluate the license text.

12. Managing open source licenses • Detecting embedded licenses


The screenshot shows a window titled "Discoveries" with a close button in the top right. The main text reads: "We found these discoveries in this file: 1 License, 2 License References, 2 Copyrights". Below this is a summary table:

Category	Hits
Licenses	
GNU General Public License v2.0 or later	2
License References	
GNU General Public License	3
GNU Lesser General Public License	2
Copyrights	
Copyright (C) yyyy name of author	1
Copyright (C) 1989, 1991 Free Software Foundation, Inc.	1

To the right of the table is a preview of the source file content, with line numbers 1 through 21. Line 1 is highlighted in yellow and contains the text "GNU GENERAL PUBLIC LICENSE". A "Close" button is located at the bottom right of the window.

If you do not upload the source files, the Black Duck UI only displays the location of the discovered license text in the file, by line number:

To include your source files, after your administrator has enabled source uploads, as described in the installation guide, include the upload source parameter when scanning.

 **Note:** Regardless whether you upload your source files or not, embedded license detection cannot be completed offline as it requires communication with the Black Duck server.

Supported file extensions/ file names

Embedded license search occurs in file extensions such as `.bat` or `.js` and for these file names, or file names that include the following text, regardless of case:

- bdsI
- copying
- copyright
- control
- dad
- gpl
- install
- legal
- lgpl
- license
- licence
- licenses
- licences
- notice
- readme

License detection process

The process to view embedded licenses is:

1. Enable detecting of deep license data when scanning and optionally, enable uploading source files for viewing embedded licenses within the file.
2. Review embedded licenses.

Enable detecting of deep license data

All scanning methods have an option to enable license string search:

- Signature Scanner command line
- Black Duck Detect (Desktop) Black Duck Detect
- Black Duck Detect

Using the Signature Scanner command line

Use the `--license-search` parameter to enable embedded licenses.

Click [here](#) for more information on using the command line.

Using Black Duck Detect (Desktop) or Black Duck Detect

Use the `--detect.blackduck.signature.scanner.license.search` property to enable deep license data detection. This property is available in Black Duck Detect version 6.2 and later.

Reviewing embedded licenses

Black Duck displays the location of these licenses in your code tree.

To review embedded licenses :

1. After enabling license search, select the **Source** tab from your project version BOM page.
2. Select a folder in the code tree that you want to determine if there are embedded licenses.

Optionally, select **All Subfolders** to view information for all subfolders.

The table displays information in the table for the selected location. By default the **Files** option is selected.

12. Managing open source licenses • Detecting embedded licenses

Black Duck Projects
Sample Project ▾ 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

src/# » src

Files Discoveries All Subfolders

Edit Reset File Adjustments Add Filter

Name	Component	Match Type	License	Usage	Discovery Types
.travis.yml	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	
Base64Decoder.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License
Base64Decoder.java	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License </>
Base64DecoderTestCase.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License
Base64DecoderTestCase.java	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License </>
CONTRIBUTING.md	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	
Closeable.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	
Closeable.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	
Closeable.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License
Closeable.html	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License
Closeable.java	Apache Commons FileUpload fileupload-1.4	commons-Exact Directory	Apache-2.0	Dynamically Linked	License </>

3. Select **Discoveries** to view the list of embedded licenses for this location.

Black Duck Projects
Sample Project ▾ 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

src/# » src

Files Discoveries All Subfolders

License Searches

Licenses 221 Files

Apache License Version 2.0 | 224 hits
219

Eclipse Public License Version 1.0 | 3 hits
| 1

GNU General Public License Version 2 | 1 hits
| 1

License References 6 Files

GNU General Public License | 8 hits
3

Apache License | 5 hits
2

BSD | 5 hits
1

4. Select a license to view the **Source** tab filtered to display the files that contain the selected embedded license text.

The screenshot shows the Black Duck Projects interface for a project named "Sample Project 1.0". The project status is "Phase: In Planning | Scans: Up to Date | Status: Up to Date".

The left sidebar displays a file tree structure:

- hh 2/6 TT 6 scan
 - blowfish.c
 - external
 - lib
 - lib - Copy
 - licenses
 - new
 - samplefile1.h
 - samplefile2.c
 - src (selected)
 - commons-fileupload-1.4-sr...
 - commons-fileupload-1....
 - .travis.yml
 - CONTRIBUTING.md
 - LICENSE.txt
 - NOTICE.txt
 - pom.xml
 - README.md
 - RELEASE-NOTES.txt
 - site-content
 - src
 - src_jo
 - src_ourfaces
 - src_pgsl
 - tools

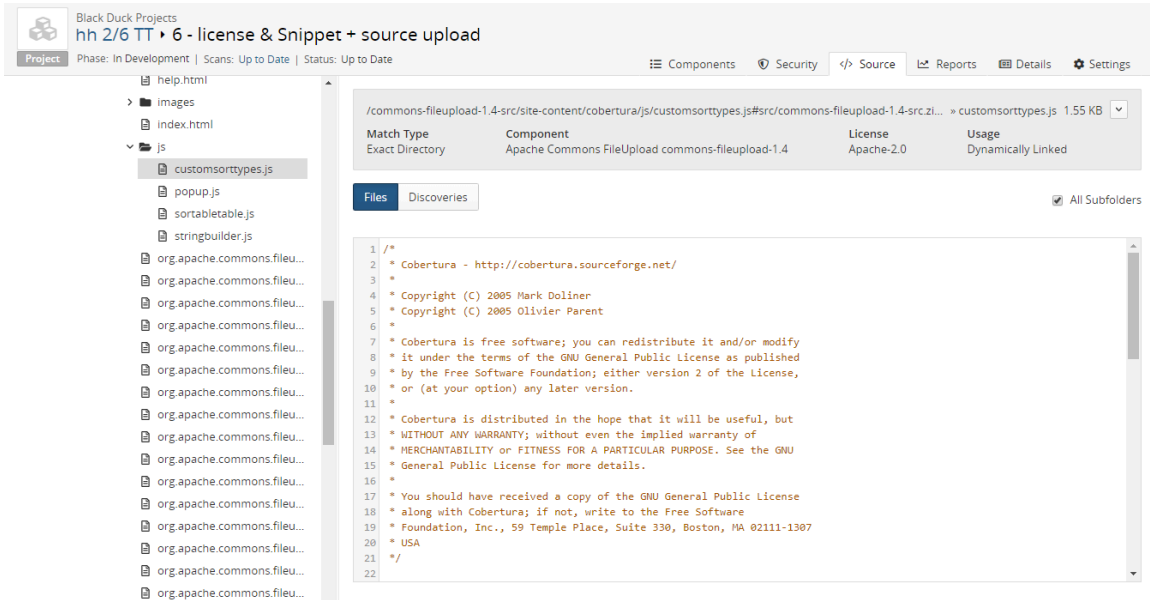
The right pane shows the selected file path: `src/# » src`. Below this, there are tabs for "Files" (selected) and "Discoveries". There are also buttons for "Edit" and "Reset File Adjustments".

A table below the buttons shows the file details:

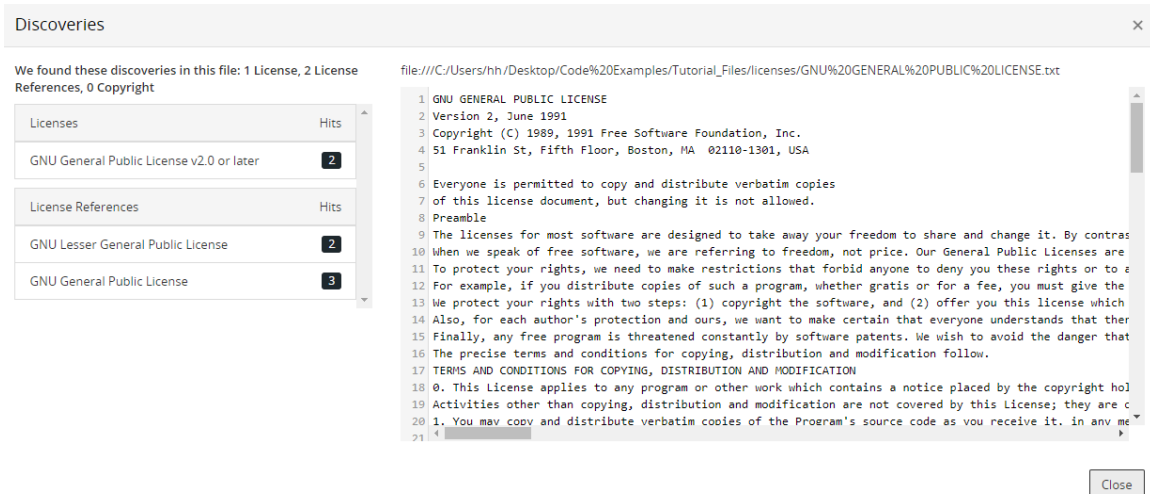
<input type="checkbox"/>	Name	Component
<input type="checkbox"/>	<code>customsorttypes.js</code>	

Optionally, select a file name to view the location of the file in the code tree. If you uploaded your source files, the file contents appears on the page.

12. Managing open source licenses • Detecting embedded licenses



5. Select a type of discovery (**License** or **License Reference**) from the **Discovery Type** column to open the Discoveries dialog box.

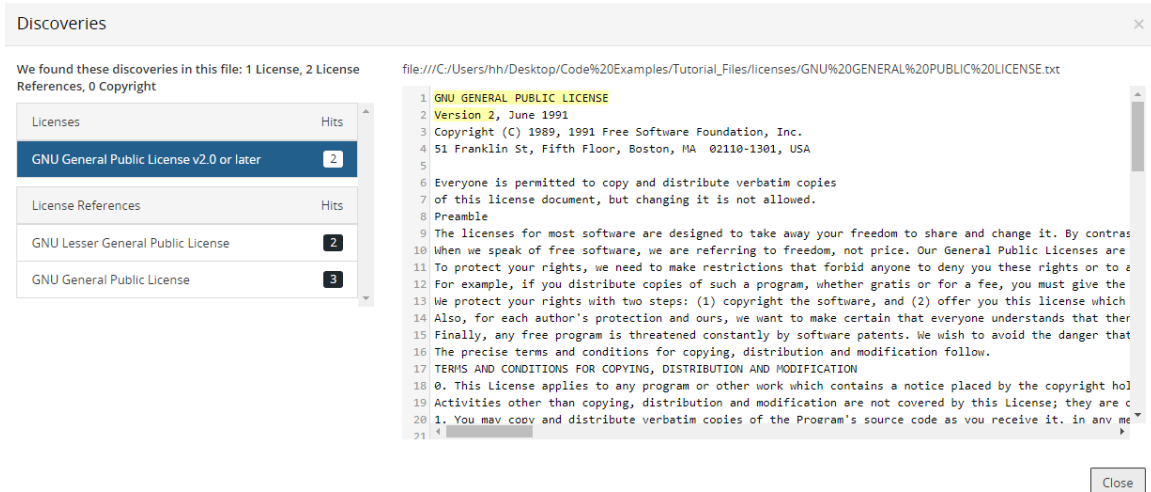


The Discoveries dialog box shows all licenses and license references found for the selected file.

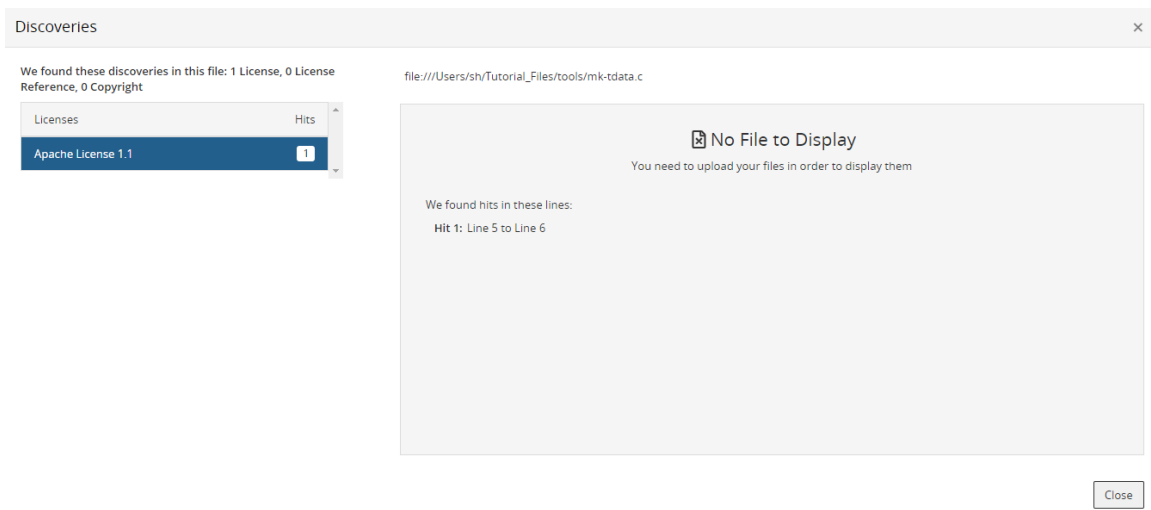
The information that appears here depends on whether you uploaded source files.

In the example shown above, source files were uploaded in the scan.

6. Select a license to view the highlighted license text indicating the embedded license text found.



If you did not upload source files, the Discoveries dialog box displays the location of the discovered license text in the file, by line number:



Modifying licenses for a component

So that you can successfully manage license risk, you may need to edit the license(s) for a component version so that it is different from the component version's declared license identified in Black Duck KB or the license originally selected for the version of the custom component.


You can modify a single license or include multi-license scenarios, such as "License A AND License B" or "License A OR License B". This lets you accurately represent the licenses in Black Duck for the components in your projects.

Note the following:



- Edits made to a license in the BOM are *local* edits. These edits apply to this version of the component for this BOM only.
- Edits made to a license from the [Black Duck KnowledgeBase component version page](#) or the [custom component version page](#) are *global* edits. These edits apply to all instances of this version of the component. However, edits made at the BOM level will override these edits.

To modify licenses:

1. To modify a single license:

a. Click  located next to the license name and select the license from the list of suggestions.

b. Do one of the following:

- Click  to confirm this selection.
- Click  to delete this license and the operand.

2. To add a license to the existing license(s):

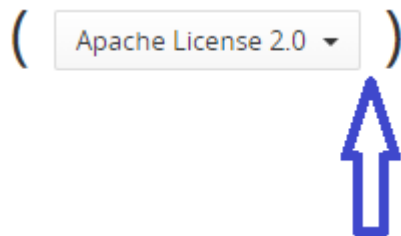
a. Click **Add License**. Black Duck adds the following at the root level:



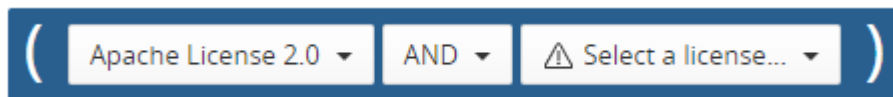
For example, when added to a single license, the following appears:



- To add a license at the original license level, select the license by placing the cursor within the parentheses of that license.



Click **Add License**. The licenses is added at the level of the original license:



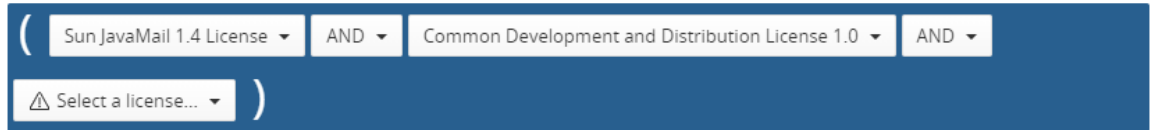
For example, when added to an existing multi-license scenario, the following appears:







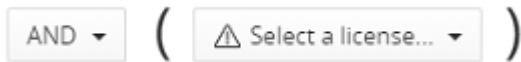
- To add a license at the same level as the existing multi-licenses, select the license by placing the cursor within the parentheses of the existing group.



Click **Add License**. The license is added at the same level as the existing licenses:



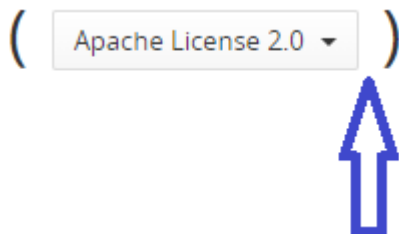
- b. Optionally, click  next to the operand to change it. Possible values are AND or OR.
 - c. Click  located next to the license name and select the license from the list of suggestions.
 - Click  to confirm this selection.
 - Click  to delete this license and the operand.
 - d. Repeat as necessary.
3. To add a multi-license scenario (for example, License A AND (License B OR License C)):
 - a. Click **Add Group**. Black Duck adds the following at the root level:



When added to a single license, the following appears:



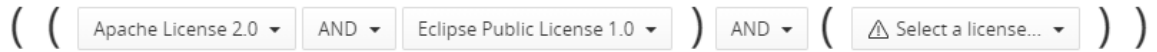
- To add a group at the original license level, select the license by placing the cursor within the parentheses.



Click **Add Group**. The following appears:



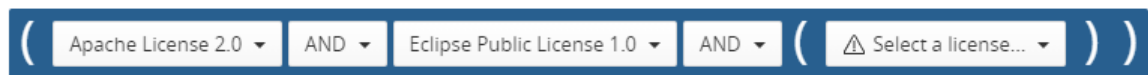
When added to an existing multi-license scenario, the following appears:






- To add a group at the original license level, select the license by placing the cursor within the parentheses:



Click **Add group**. The following appears:



- Optionally, add additional licenses as described in step 6a.
 - Optionally, click ▼ next to the operand to change it. Possible values are AND or OR.
 - Click ▼ located next to the license name and select the license from the list of suggestions.
 - Click  to confirm this selection.
 - Click  to delete this license and operand.
 - Repeat as necessary.
- Optionally:
 - Select **Reset Changes** to display the license(s) that appeared when you initially opened this dialog box.
 - Select a group and select **Delete Selected Group** to remove this group.
 - Click **Save Changes** if editing the license in the BOM or **Save**.
 The assigned license is updated. If the new license carries a different type of license risk than the previous one, the [license risk calculations](#) for the component and for the project version are updated in project version BOMs. A  appears in the table row in the BOM to indicate that a manual adjustment was made to this component.

When viewed in the BOM, the license obligations for the revised license(s) will appear when you re-open the *Component Name Version Component License* dialog box.

Reverting BOM-level license edits

If you selected a different license for a component when editing licenses in the BOM, you can revert the license to its original license as defined in Black Duck KnowledgeBase.

To revert to an original license:

1. Log in to Black Duck.

2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.
4. Select the to open the *Component Name Version* Component License dialog box.

The screenshot shows a dialog box titled "Component License" with a close button (X) in the top right corner. Below the title bar, there is a checked checkbox labeled "Include in Notices File Report". Underneath, there is a dropdown menu for "Attribution Statement" and a "License Edit" section. The "License Edit" section contains the text "Select a license to view. Toggle Edit Mode on to edit." and a toggle switch for "Edit Mode off". A dropdown menu below this text shows "Apache License 2.0" selected. At the bottom right, there are two buttons: "Close" and "Save Changes".


5. Select the Edit Mode option to enable editing.

The screenshot shows the same "Component License" dialog box, but now the "Edit Mode" toggle is turned "on". The text in the "License Edit" section has changed to "Modify the existing license or add a new license/group. ?". A "Revert to Original" button with a circular arrow icon is now visible. The "Apache License 2.0" dropdown menu now has a downward arrow. At the bottom left, there are three links: "+ Add License", "+ Add Group", and "Reset Changes". The "Close" and "Save Changes" buttons remain at the bottom right.

6. Select **Revert to Original** to revert the license
7. Click **Save Changes**.
The license is reverted. If the license carries a different type of license risk than the previous one, the [license risk calculations](#) for the component and for the project version are updated in the BOM.

About custom licenses

If you discover that a license that you use for a component in your BOM is not available from Black Duck KnowledgeBase, License Managers – users with the License Manager [role](#) – can create and manage custom licenses. These custom licenses can then be selected for a component version in a BOM to ensure that the BOMs are accurate.

 **Note:** If Black Duck KnowledgeBase is missing an open source license, instead of creating a custom license, you can contact [Black Duck Support](#) to request that this license be added to the KnowledgeBase.

Custom licenses:

- Consist of a name, a [license family](#), and license text.
- Can be used to [create policy rules](#).
- Use the same [rules to determine license risk](#) as licenses from Black Duck KnowledgeBase.
- Can be [modified locally in a BOM](#) by users with the appropriate [role](#). That user cannot edit the name or license family but can edit the license text. The edited license text only applies to the version of the license associated with the BOM.

License Managers can use the [License Management page](#) to manage custom licenses and Black Duck KnowledgeBase licenses used in all the projects in your organization.

Creating custom licenses

Only users with the License Manager role can create [custom licenses](#).

1. Log in to Black Duck with the License Manager [role](#).

2.



Click **Manage** > **Licenses**.

The License Management page appears.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Click **Create License** to open the Create a Custom License dialog box.

Create a Custom License

Name *

License Family * Nothing Selected

License Text *

Status Nothing Selected

Notes

Expiration Date

Cancel Create

4. Enter the name for this custom license.
5. Select the [license family](#) for this custom license. This license family, along with the component usage, determines the [license risk](#).
You can select a KnowledgeBase or custom license family.
6. Enter the license text.
7. Optionally, select a status, enter any notes, and select an expiration date for this license.
8. Click **Create**.

Editing a custom license

Custom licenses can be edited by users with the License Manager role and by users with the BOM Manager, or Project Manager role:

- License Managers can make *global* edits to custom licenses. The License Manager can edit any of the custom license settings.
These edits are propagated to BOMs with components using the custom license as described below.
- BOM Managers and Project Managers can only make *local* edits to the license text of a custom license used in a BOM.
These edits only apply to the version of the custom license used in the BOM.

When the License Manager edits a custom license:

- Edits to the license family and license name are always propagated to the custom licenses used in BOMs.
- Edits to the license text *may or may not* be propagated to the custom licenses used in BOMs:
 - If the BOM Manager or Project Manager *edited the license text*, the edits made by the License Manager *are not* propagated to the version of the custom license used in the BOM.

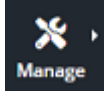
12. Managing open source licenses • About custom licenses

- If the BOM Manager or Project Manager *did not edit* the license text, the edits made by the License Manager *are* propagated to the custom license used in the BOM.

To edit a custom license:

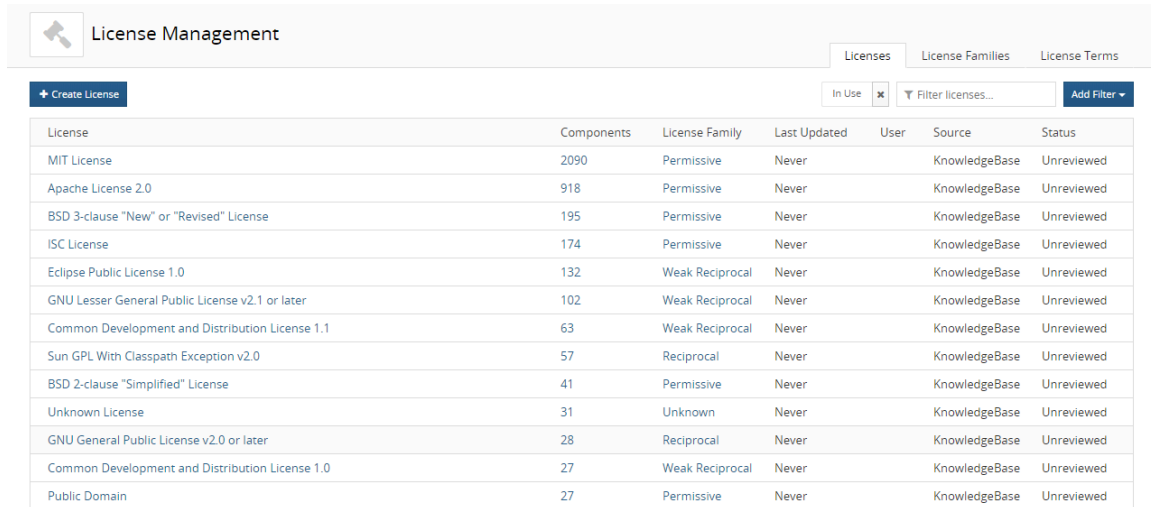
1. Log in to Black Duck with the License Manager role.

2.



Click **Manage** > **Licenses**.

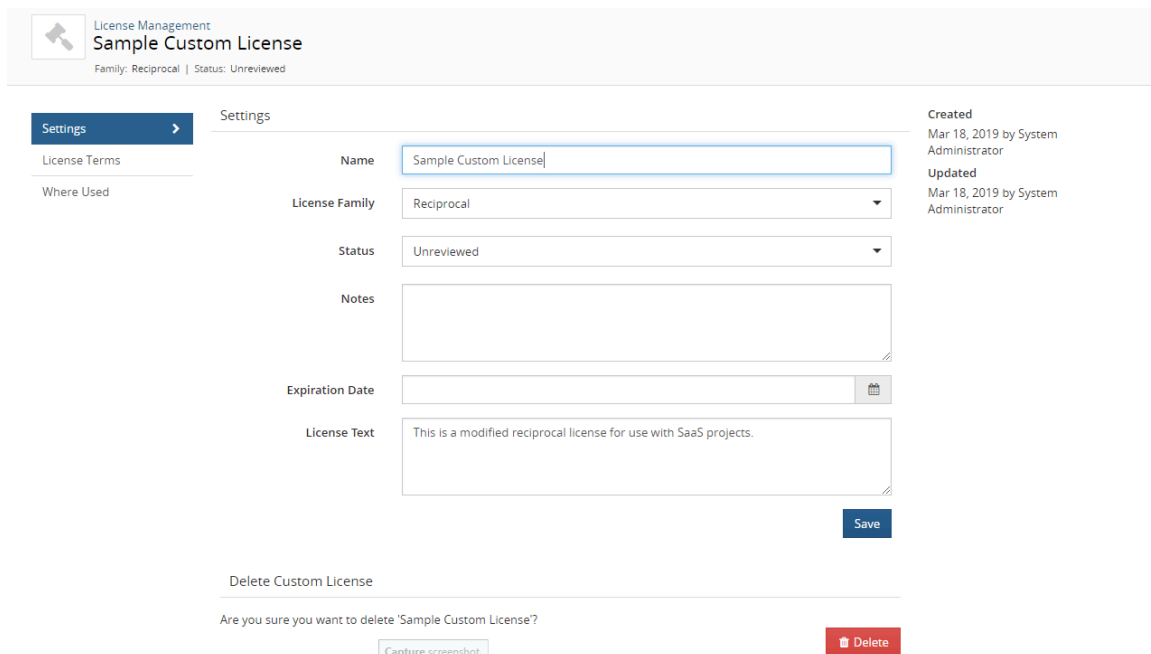
The License Management page appears.



The screenshot shows the 'License Management' page with a table of licenses. The table has columns for License, Components, License Family, Last Updated, User, Source, and Status. The licenses listed include MIT License, Apache License 2.0, BSD 3-clause 'New' or 'Revised' License, ISC License, Eclipse Public License 1.0, GNU Lesser General Public License v2.1 or later, Common Development and Distribution License 1.1, Sun GPL With Classpath Exception v2.0, BSD 2-clause 'Simplified' License, Unknown License, GNU General Public License v2.0 or later, Common Development and Distribution License 1.0, and Public Domain.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Select the license name to display the *License Name Settings* tab.



The screenshot shows the 'License Name Settings' page for a custom license named 'Sample Custom License'. The page includes a 'Settings' tab, a 'Where Used' section, and a 'Created' section. The 'Settings' section contains fields for Name, License Family, Status, Notes, Expiration Date, and License Text. The 'Created' section shows the creation date and user. A 'Save' button is at the bottom right, and a 'Delete' button is at the bottom center.

Settings

Name

License Family

Status

Notes

Expiration Date

License Text

Created
Mar 18, 2019 by System Administrator
Updated
Mar 18, 2019 by System Administrator

Save

Delete Custom License

Are you sure you want to delete 'Sample Custom License'?

Delete

4. Modify the information shown for this custom license.

- **Name:** License name. You can modify a custom license name.

- **License Family:** Use the drop-down selector to choose the license family.
 - **Status:** Use the drop-down selector to choose the license status.
 - **Notes:** You can type any text in this field. Use this for additional information or helpful notes.
 - **Expiration Date:** Use the calendar tool to set the expiration date.
 - **License Text:** The actual license as found in the component.
5. Click **Save**.
- The username of the user who edited this license appears in the **User** column and the time the license was modified appears in the **Last Updated** column in the License Management page.
 - Edit information also appears in the *Component/Subproject Name Version Component License* dialog box.

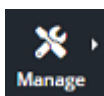
Deleting custom licenses

You cannot delete a license that is being used in a BOM.

You also cannot delete licenses provided by Black Duck KnowledgeBase.

1. Log in to Black Duck with the License Manager [role](#).


2.



Click **Manage** > **Licenses**.

The License Management page appears.

License Management						
Licenses License Families License Terms						
+ Create License						
In Use x Filter licenses... Add Filter						
License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Click  and select **Delete** in the row of the custom license that you want to delete to display a confirmation dialog box.

An error message appears if you try to delete a custom license that is currently being used in a BOM.

4. Click **Delete** to confirm.

About license terms

License terms are the provisions in the license which grant rights or impose restrictions on the use of the software under that license. They summarize the conditions regarding the reuse of software that is contained in the text of the license. They indicate the things you can do (permitted), the things you cannot do (forbidden) and the things you must do (required) to comply with the license. Please note that the license terms provide by the Black Duck application are just general summaries of the license and cannot be taken as legal advice.

You can create custom license terms and manage existing KnowledgeBase license terms to ensure that you meet the legal obligations associated with a license. Manage license terms to help your developers know the legal obligations associated with a license and to help you bring a project into compliance with licensing obligations.

Users with the License Manager [role](#) can:

- [Create](#), [edit](#), or [delete](#) custom license terms.
- [Associate](#) a custom or KnowledgeBase license term to one or more custom or KnowledgeBase licenses.
- [Remove](#) custom license terms from custom or KnowledgeBase licenses.
- [Remove](#) KnowledgeBase license terms from custom licenses or KnowledgeBase licenses that were not originally defined by Black Duck KnowledgeBase.
- [Deprecate](#) custom license terms.
- [Disable](#) or [restore](#) KnowledgeBase license terms for a KnowledgeBase license.
- Determine if the license term [requires fulfillment](#).

Suggested work flow

To manage custom and KnowledgeBase license terms:

1. With the assistance of your legal counsel, review the license terms associated with Black Duck KnowledgeBase licenses. However, please note that not all licenses will have pre-defined license terms and not every condition of use may be represented by Black Duck-provided license terms. The license terms provided by the Black Duck application are just general summaries of the license and cannot be taken as legal advice or replace a legal review.
2. Determine if there are any Black Duck KnowledgeBase terms that need to be modified to more accurately reflect your legal obligations.
 - You can disable KnowledgeBase terms associated with Black Duck KnowledgeBase licenses so that these terms are not shown to your end users.
 - Optionally, you can create new custom terms and then associate them to KnowledgeBase licenses either in addition or replacing an existing KnowledgeBase term.
3. If you created custom licenses, determine if you need to create new custom license terms or associate existing KnowledgeBase terms to the custom license.
4. Continue the review process, as you may wish to eventually deprecate a custom license term or remove a KnowledgeBase term.

License terms process

If, after reviewing the existing terms, you determine that you need to create new license terms, do the following:

1. [Create categories](#) to manage your license terms. Categories are used to manage your license terms. You can also create a category while creating a license term.
2. [Create a custom license term](#).
3. [Associate](#) the new term to one or more licenses.

The **License Terms** tab shows all license terms for custom and KnowledgeBase licenses.

To view the License Terms tab:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  **Manage** > **Licenses**.

The License Management page appears.

12. Managing open source licenses • About license terms

License Management

Licenses License Families License Terms

+ Create License In Use x Filter licenses... Add Filter

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Select the License Terms tab.

License Management

Licenses License Families License Terms

+ Create Term Categories Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source ⚠️ Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

The table provides the following information:

C Description

Term

Hover over the name to view the description for this term.

For custom license terms, select the name to open the Edit a License Term dialog box.

Source

- KnowledgeBase. This is a standard term from Black Duck KnowledgeBase.
- Custom. A license term you created.

Responsibility

- Permitted

C Description

- Forbidden
- Required

Category for this license term.

License terms from Black Duck KnowledgeBase have **KnowledgeBase** as the category. Custom license terms list the category defined when adding the term.


Last Updated that the license term was last updated and the username of the user who updated this term.

Updated column lists **Never** for KnowledgeBase license terms.

Viewing license terms

License terms are categorized into things you are permitted to do (rights), things you are forbidden to do (restrictions), and things you are required to do (obligations) to comply with the license.

You can view license terms using the License Management page and when viewing license information in the BOM.

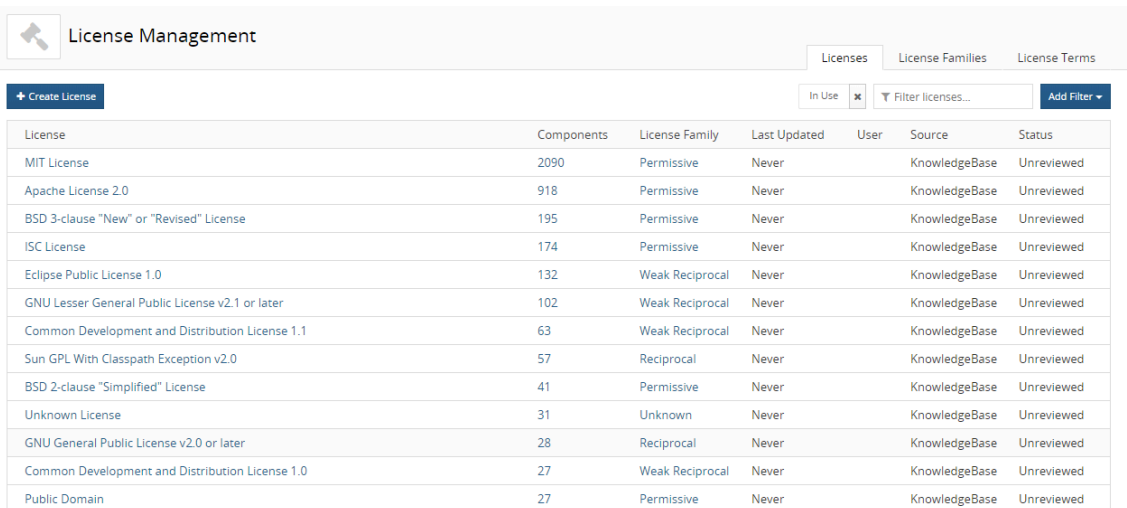
 **Note:** License obligation will not appear in the UI, if the information is unavailable from [OpenHub](#),

To view the license terms from the License Management page:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

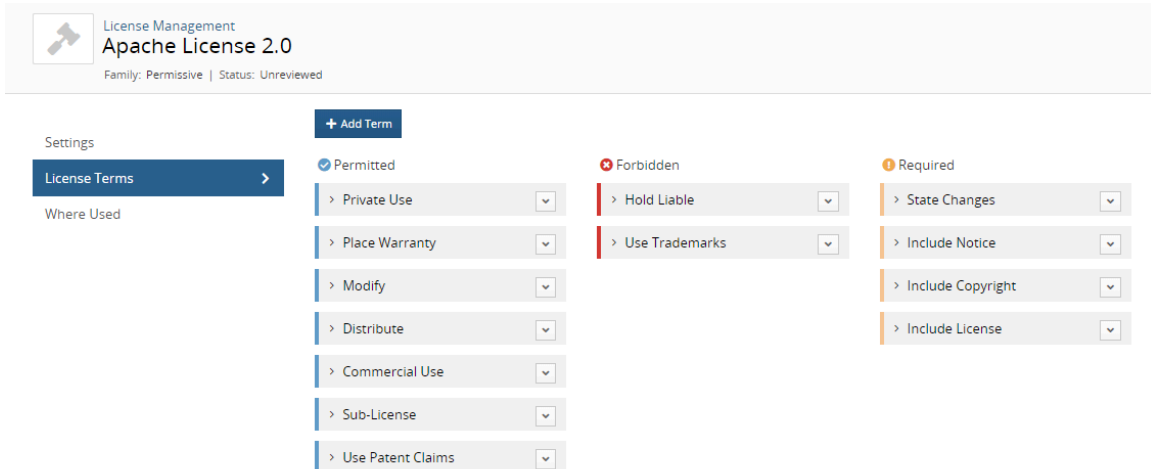
The License Management page appears.



The screenshot shows the License Management interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there is a '+ Create License' button and a search filter 'Filter licenses...'. The main table lists various license terms with columns for License, Components, License Family, Last Updated, User, Source, and Status.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Select a license from the **License** tab to display the *License Name* page.
4. Select the **License Terms** tab to view the obligations for this license.

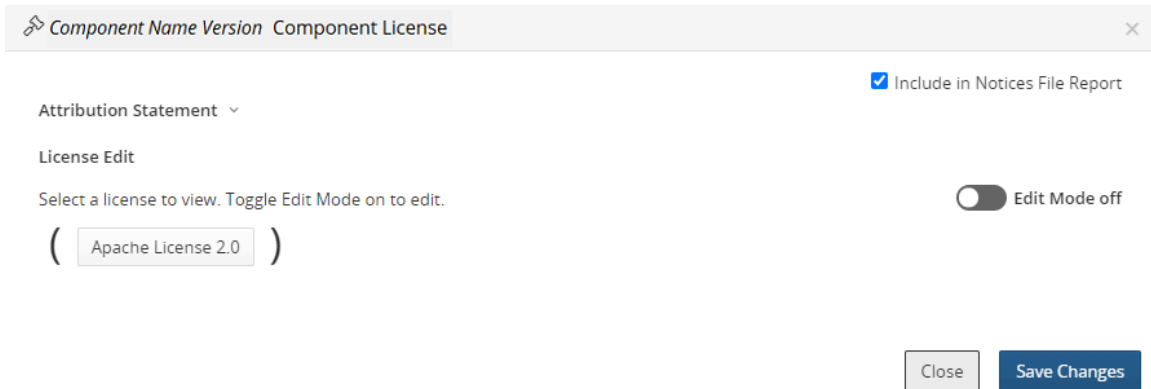


If available, select > to view additional information.

To view the license term information in a BOM:

Only users with the appropriate [role](#) can view this information in the BOM.

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to open the **Components** tab and view the BOM.
3. Select the license name to open the *Component Name Version Component License* dialog box.



4. Select the license you wish to view the license obligation information. The dialog box expands to show the obligations and license text for the selected license.

About license term fulfillment

License Managers can define which license terms require fulfillment.

The fulfillment status of a license term is defined for a term at the license level, as not all instances of a license term may require fulfillment. This allows you to easily define the fulfillment requirements for a license term,

The work flow for license term fulfillment is:

1. License Managers determine the license terms that require fulfillment. Fulfillment can be defined when:

- [Associating a license term.](#)
 - [Viewing all terms for a specific license.](#)
 - [Creating a new term or adding an existing term for a specific license](#) when using the *License Name License Terms* tab.
2. The System Administrator [enables the Project Version's Legal tab.](#)
 3. BOM Manager's use the **Term Fulfillment** tab on the *Project Version's Legal* tab to view all license terms that require fulfillment and [indicate which license terms are fulfilled.](#)

Note the following:

- It may take time for license term fulfillment requirements to appear on the **Legal** tab.
- Policy managers can [create a policy rule](#) that will trigger a violation when there are unfulfilled license terms.

Note that the **Term Fulfillment** tab on the **Legal** tab must be enabled so that a user can indicate that a term is fulfilled. If the **Legal** tab is disabled, which is the default setting, a user will be unable to indicate that a term is fulfilled, and policy violations cannot be cleared.

- License term fulfillment status can be [cloned](#).
- A new project version report, `license_term_fulfillment_date_time.csv` lists the license terms and fulfillment status for a project version.

Defining fulfillment when viewing terms for a license

License Managers can indicate a license term is required when using the *License Name License Terms* tab which shows all terms for a specific license.

To define the fulfillment requirement when viewing a license:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.

License Management							
Licenses License Families License Terms							
+ Create License							
In Use x Filter licenses... Add Filter							
License	Components	License Family	Last Updated	User	Source	Status	
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed	
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed	
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed	
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed	
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed	
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed	
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed	
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed	
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed	
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed	
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed	
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed	
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed	


3. In the **Licenses** tab, select the license name to display the *License Name Settings* tab.


12. Managing open source licenses • About license terms

The screenshot shows the 'Settings' page for the 'Apache License 2.0'. The page header includes 'License Management' and 'Apache License 2.0' with sub-headers 'Family: Permissive' and 'Status: Unreviewed'. On the left, there is a navigation menu with 'Settings' (selected), 'License Terms', and 'Where Used'. The main content area is titled 'Settings' and contains several fields: 'Name' (Apache License 2.0), 'License Family' (Permissive), 'Status' (Unreviewed), 'Notes' (empty text area), 'Expiration Date' (calendar icon), and 'License Text' (Apache License Version 2.0, January 2004). On the right, there are 'Created' and 'Updated' fields, both set to 'never'. A 'Save' button is located at the bottom right.

4. Select the **License Terms** tab to view the terms associated with this tab.

The screenshot shows the 'License Terms' page for the 'Apache License 2.0'. The page header is the same as the previous screenshot. The left navigation menu now has 'License Terms' selected. The main content area is titled '+ Add Term' and is divided into three columns: 'Permitted' (blue header), 'Forbidden' (red header), and 'Required' (orange header). Each column contains a list of license terms with a dropdown arrow next to each. The 'Permitted' column includes: Private Use, Place Warranty, Modify, Distribute, Commercial Use, Sub-License, and Use Patent Claims. The 'Forbidden' column includes: Hold Liable and Use Trademarks. The 'Required' column includes: State Changes, Include Notice, Include Copyright, and Include License.

5. Click  next to the KnowledgeBase license term you wish to indicate fulfillment is required and select **Fulfillment Required**.

The Fulfillment Required icon () appears to indicate this license term is required.

License Management
Apache License 2.0
Family: Permissive | Status: Limited Approval

Settings
License Terms >
Where Used

+ Add Term

Permitted

- > Private Use
- > Place Warranty
- > Modify
- > Distribute
- > Commercial Use
- > Sub-License
- > Use Patent Claims

Forbidden

- > Hold Liable
- > Use Trademarks

Required

- > State Changes
- > Include Notice
- > Include Copyright
- > Include License

Creating license terms

You can create a license term when viewing all available license terms or when viewing the terms that apply to a specific license.

Only users with the License Manager role can create license terms.

To create a license term:

1. Log in to Black Duck with the License Manager [role](#).

2.

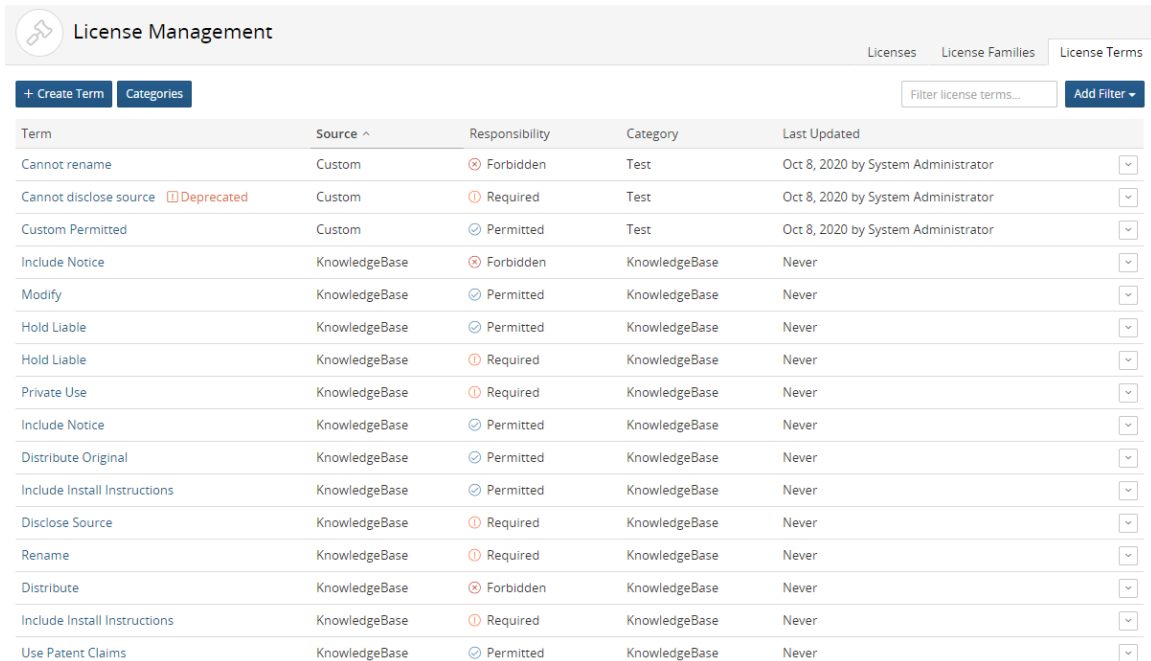


Click **Manage** > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.

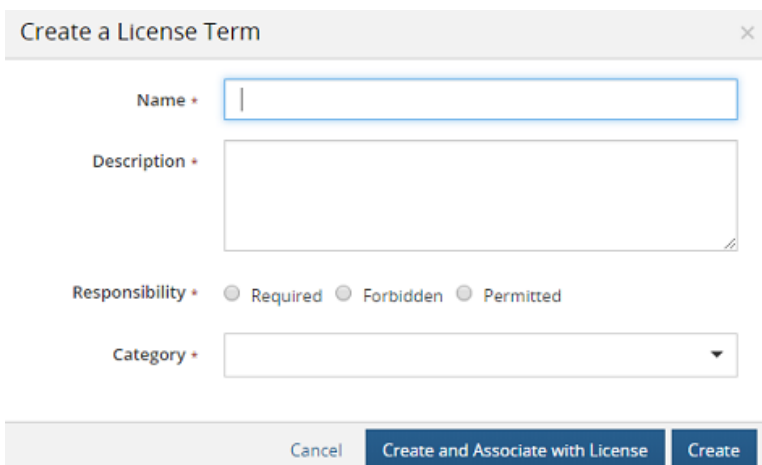
12. Managing open source licenses • About license terms



The screenshot shows the 'License Management' interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there are buttons for '+ Create Term' and 'Categories'. A search bar labeled 'Filter license terms...' and an 'Add Filter' button are also present. The main content is a table with the following columns: Term, Source, Responsibility, Category, and Last Updated. The table lists various license terms such as 'Cannot rename', 'Cannot disclose source', 'Custom Permitted', 'Include Notice', 'Modify', 'Hold Liable', 'Private Use', 'Distribute Original', 'Include Install Instructions', 'Disclose Source', 'Rename', 'Distribute', and 'Use Patent Claims'. Each row includes a 'Last Updated' date and a dropdown arrow.

Term	Source	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click **Create Terms** to open the Create a License Term dialog box.



The screenshot shows the 'Create a License Term' dialog box. It has a title bar with a close button. The form contains the following fields and controls:

- Name ***: A text input field.
- Description ***: A text area.
- Responsibility ***: Radio buttons for 'Required', 'Forbidden', and 'Permitted'.
- Category ***: A dropdown menu.

At the bottom, there are three buttons: 'Cancel', 'Create and Associate with License', and 'Create'.

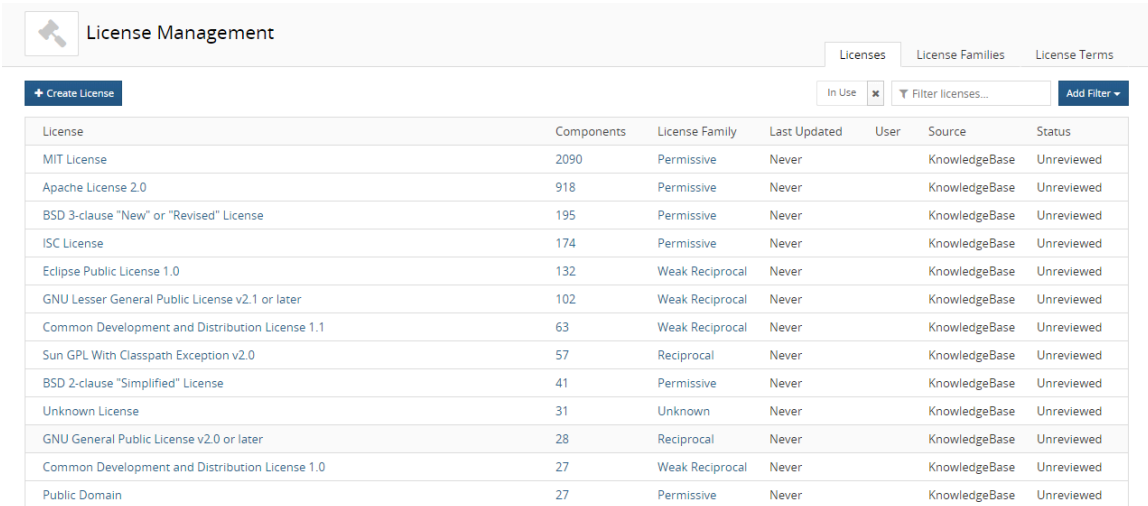
4. Complete the information in the dialog box:
 - **Name.**
 - **Description.**
 - **Responsibility.** Select whether this responsibility is required, forbidden, or permitted.
 - **Category.** Select a category for this license term. Optionally, create a new category by entering text in the field and selecting to add this new category. The new category will be automatically created.
5. Do one of the following:
 - Click **Create and Associate with License**. The License Association dialog box appears. Select the licenses to associate to this license term, optionally select whether this term requires fulfillment, and click **Add**. Click [here](#) for more information about associating a term to a license.
 - Click **Create**. The new license term appears in the table in the **License Terms** tab.

To create a license term for a specific license:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

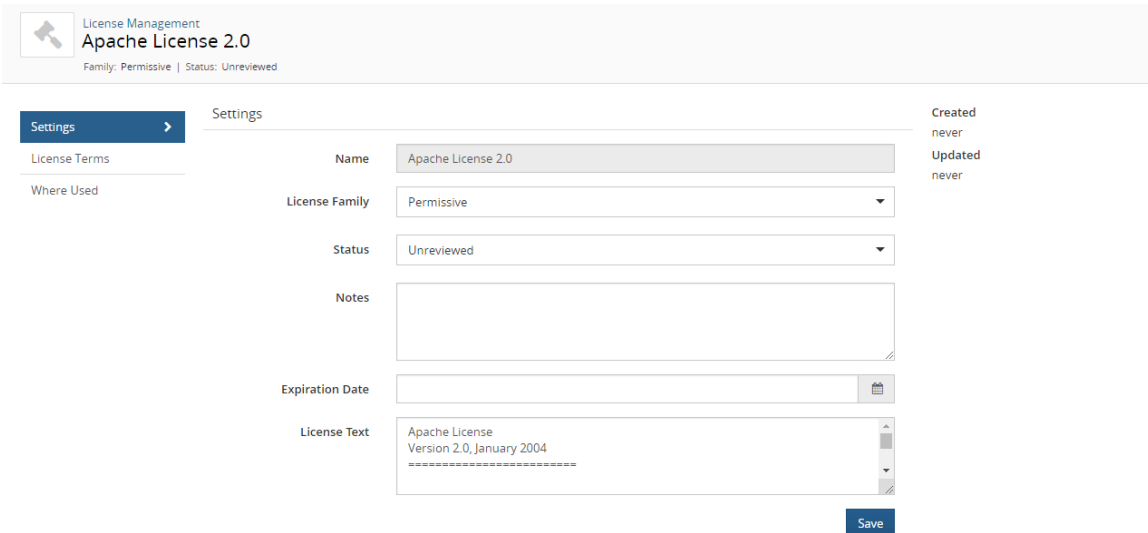
The License Management page appears.



The screenshot shows the 'License Management' page with a table of licenses. The table has columns for License, Components, License Family, Last Updated, User, Source, and Status. The following table represents the data shown in the screenshot:

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. In the **Licenses** tab, select the license name to display the *License Name Settings* tab.



The screenshot shows the 'License Name Settings' page for 'Apache License 2.0'. The page includes a sidebar with 'Settings' selected, and a main form area with the following fields:

- Name:** Apache License 2.0
- License Family:** Permissive
- Status:** Unreviewed
- Notes:** (Empty text area)
- Expiration Date:** (Empty date field)
- License Text:** Apache License
Version 2.0, January 2004
.....

On the right side, there are labels for 'Created never', 'Updated never', and 'never'. A 'Save' button is located at the bottom right of the form.

4. Select the **License Terms** tab to view the terms associated with this tab.

12. Managing open source licenses • About license terms

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings

License Terms >

Where Used

+ Add Term

Permitted

- > Private Use
- > Place Warranty
- > Modify
- > Distribute
- > Commercial Use
- > Sub-License
- > Use Patent Claims

Forbidden

- > Hold Liable
- > Use Trademarks

Required

- > State Changes
- > Include Notice
- > Include Copyright
- > Include License

5. Select **New** to create a new term. The Add Term dialog box displays the fields you need to complete to create a new term.

Add Term

Existing New

Name +

Description +

Responsibility + Required Forbidden Permitted

Category +

Fulfillment Required

Cancel Add

6. Complete the information in the dialog box:
 - **Name.**
 - **Description.**
 - **Responsibility.** Select whether this responsibility is required, forbidden, or permitted.
 - **Category.** Select a category for this license term. Optionally, create a new category by entering text in the field and selecting to add this new category. The new category will be automatically created.
 - **Fulfillment.** Indicate whether this term must be fulfilled.
7. Click **Add**. The new term is added to this license.

License Management
Sample Custom License
Family: Reciprocal | Status: Unreviewed

Settings
License Terms >

+ Add Term

Permitted
> New Permitted Term

Forbidden
No term associated

Required
No term associated

Where Used

The new license term is also listed in the **License Terms** table. You can then [associate this term](#) to other licenses and specify whether the term must be fulfilled for those licenses.

Managing license term categories

Categories help you manage and organize your license terms.

You must assign a license term to a category when you create the license term.

You can create or delete custom license term categories. License terms from Black Duck KnowledgeBase are in the KnowledgeBase category. You cannot delete this category or add custom licenses to it.

Only users with the License Manager [role](#) can create or delete categories.

To create a category:

You can also create a category when [creating a license term](#).

1. Log in to Black Duck with the License Manager role.
- 2.

Click  > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.

License Management

Licenses License Families License Terms

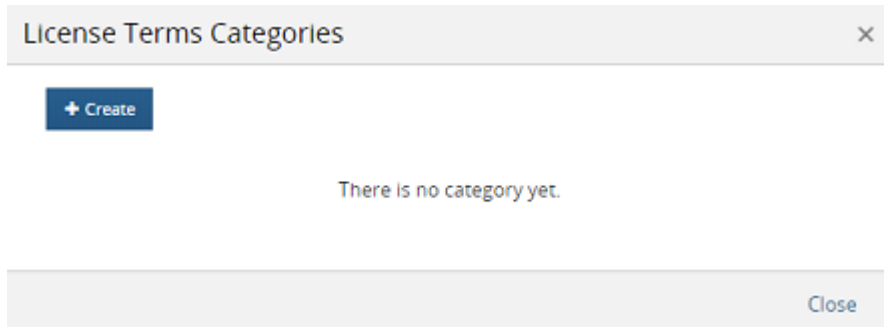
+ Create Term Categories

Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click **Categories**.

The License Terms Categories dialog box appears.



4. Click **Create** to display the field to enter the category name. Type the name of the new category in the field and select it (**Add Category Name**) located below the field. Click **Create** to create additional categories.

5. Click **Close** when you have finished creating categories.

To delete a category:

You cannot delete a category that is in use.

1. Log in to Black Duck with the License Manager role.

2.



Click **Manage** > **Licenses**.

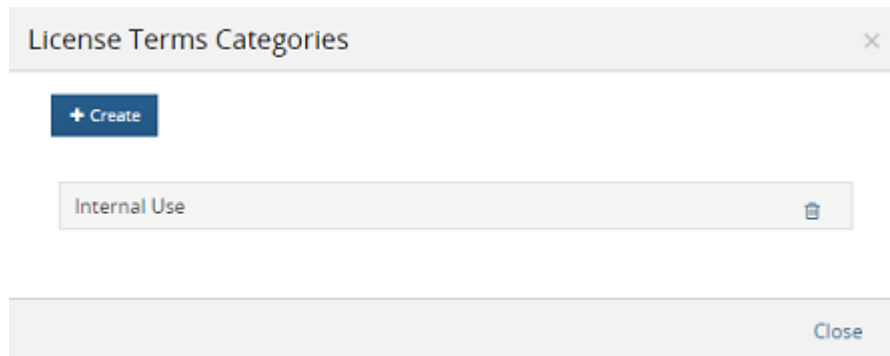
The License Management page appears.


Select the **Terms** tab to display all license terms.

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source ❌ Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click **Categories**.

The License Terms Categories dialog box appears.



4. Click  in the row of the category you want to delete.
5. Select **Delete** to confirm.

Associating a license term to a license

You can associate a new license term you created or an existing KnowledgeBase term to one or more custom or KnowledgeBase licenses.

When a license term is associated to a license, that term will appear to users when viewing licenses terms, for example, in the BOM.

Only users with the License Manager role can associate a license term to a license.

You can associate a term to a license when:

- Creating a license term. Click [here](#) for more information about creating a new term.
- Using the **License Terms** tab which lists all license terms:

License Management					
		Licenses		License Families	License Terms
+ Create Term		Categories		Filter license terms... Add Filter	
Term	Source ^	Responsibility	Category	Last Updated	
Cannot rename	Custom	⊘ Forbidden	Test	Oct 8, 2020 by System Administrator	
Cannot disclose source ⊘ Deprecated	Custom	⊘ Required	Test	Oct 8, 2020 by System Administrator	
Custom Permitted	Custom	⊙ Permitted	Test	Oct 8, 2020 by System Administrator	
Include Notice	KnowledgeBase	⊘ Forbidden	KnowledgeBase	Never	
Modify	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	
Hold Liable	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	
Hold Liable	KnowledgeBase	⊘ Required	KnowledgeBase	Never	
Private Use	KnowledgeBase	⊘ Required	KnowledgeBase	Never	
Include Notice	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	
Distribute Original	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	
Disclose Source	KnowledgeBase	⊘ Required	KnowledgeBase	Never	
Rename	KnowledgeBase	⊘ Required	KnowledgeBase	Never	
Distribute	KnowledgeBase	⊘ Forbidden	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	⊘ Required	KnowledgeBase	Never	
Use Patent Claims	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never	

- Using the **License Terms** tab for an individual license:

12. Managing open source licenses • About license terms

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings
License Terms >
Where Used

+ Add Term

- Permitted
 - > Private Use
 - > Place Warranty
 - > Modify
 - > Distribute
 - > Commercial Use
 - > Sub-License
 - > Use Patent Claims
- Forbidden
 - > Hold Liable
 - > Use Trademarks
- Required
 - > State Changes
 - > Include Notice
 - > Include Copyright
 - > Include License

To associate a license term to one or more licenses:

Use these procedures to associate a license term to one or more licenses.

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.


Select the **License Terms** tab to display all license terms.

License Management

Licenses License Families License Terms

+ Create Term Categories Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click  in the row of the license term and select **License Association**.

The License Association dialog box appears.

License Association

Term	Source	Responsibility	Category
Modify	KnowledgeBase	Required	KnowledgeBase

> Description

Select Licenses *

Add

Require Fulfillment Remove Fulfillment Requirement

No Results Found


Close

4. Use this dialog box to associate the term. To add a license: Begin typing the license name that you want to associate to this term. The list is type-ahead enabled, so you can see a list of available licenses that contain the text you have typed. Select the license and click **Add**.

Enter additional license names to associate the term with additional licenses.

5. Optionally, select the licenses for which this term requires fulfillment:

- a. Select the check box next to the license where fulfillment of this term is required.

- b. Click **Require Fulfillment**. The Fulfillment Required icon () appears in the table for the license where this term is required.

Click **Remove Fulfillment Requirement** to remove the requirement that this term must be fulfilled.

6. Click **Close**.

To associate an existing license term to a specific license:

1. Log in to Black Duck with the License Manager [role](#).

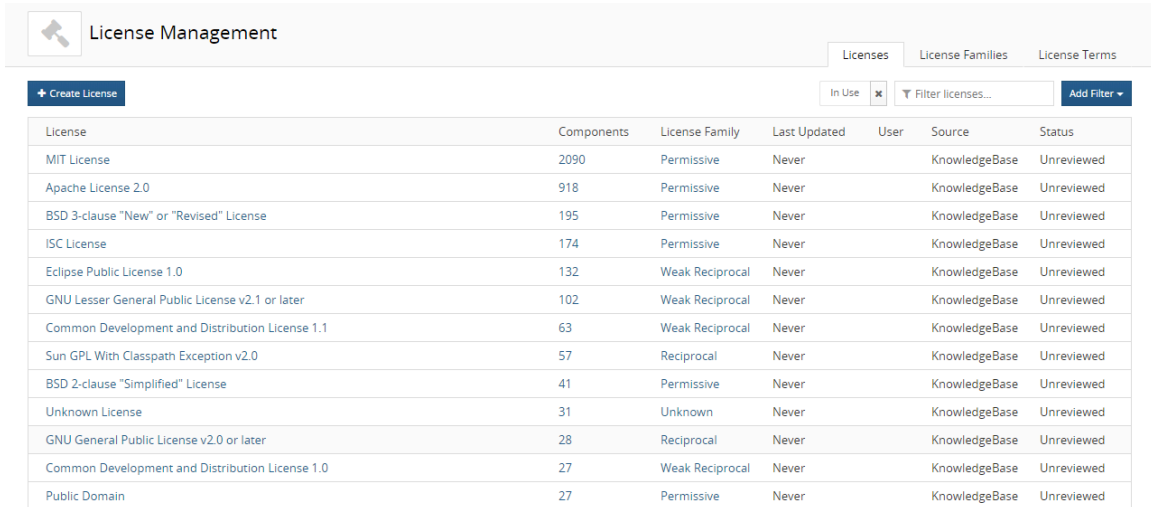
- 2.



Click **Manage** > **Licenses**.

The License Management page appears.

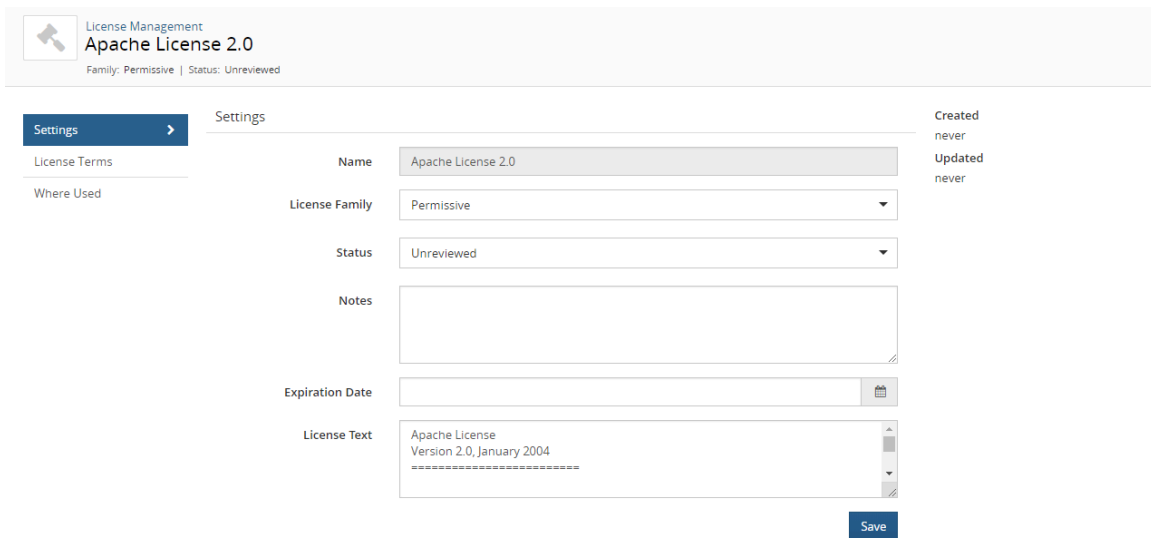
12. Managing open source licenses • About license terms



The screenshot shows the 'License Management' interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there is a '+ Create License' button and a search filter 'Filter licenses...' with an 'Add Filter' button. The main content is a table with the following columns: License, Components, License Family, Last Updated, User, Source, and Status.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. In the **Licenses** tab, select the license name to display the *License Name Settings* tab.

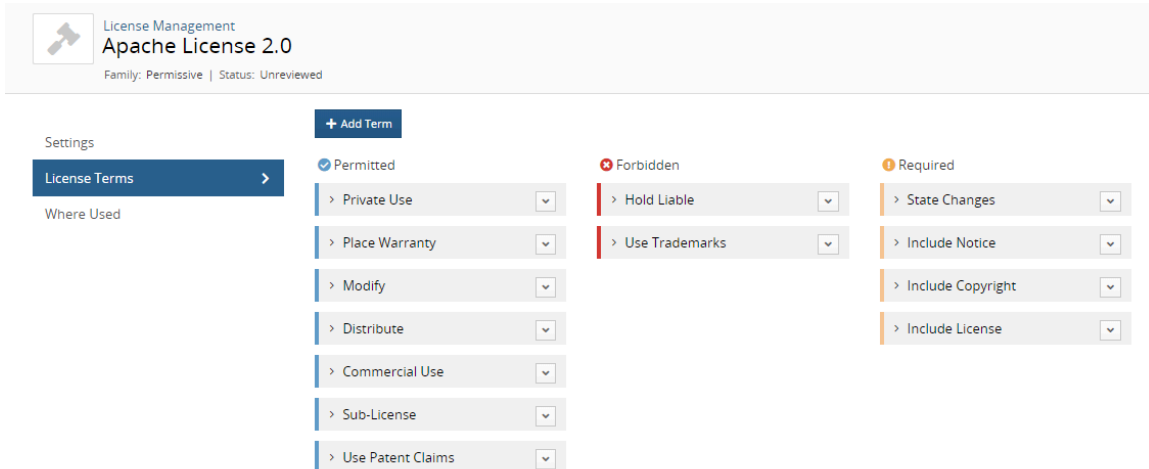


The screenshot shows the 'License Name Settings' interface for 'Apache License 2.0'. The page title is 'License Management Apache License 2.0' with subtext 'Family: Permissive | Status: Unreviewed'. On the left, there is a sidebar with 'Settings' (selected), 'License Terms', and 'Where Used'. The main content area has the following fields:


- Name:** Apache License 2.0
- License Family:** Permissive (dropdown menu)
- Status:** Unreviewed (dropdown menu)
- Notes:** Empty text area
- Expiration Date:** Empty date field with a calendar icon
- License Text:** Apache License
Version 2.0, January 2004
=====

On the right side, there are two status indicators: 'Created never' and 'Updated never'. A 'Save' button is located at the bottom right of the form.

4. Select the **License Terms** tab to view the terms associated with this tab.



5. Click **Add Term** to open the Add Term dialog box.
6. Select **Existing** to add an existing license term.

7. Begin typing the license name that you want to associate to this term. The list is type-ahead enabled, so you can see a list of available license terms that contain the text you have typed. This list displays all license terms – custom and KnowledgeBase terms.
8. Select the license term. The information for this term appears in the dialog box.
9. Optionally, select whether fulfillment is required for this term.
10. Click **Add**. The **License Terms** tab appears for this license with the new term added. The Fulfillment Required icon () will appear for any required terms.

Editing a custom license term

You can edit custom license terms.

12. Managing open source licenses • About license terms

Only users with the License Manager role can edit license terms.

To edit a license term:

1. Log in to Black Duck with the License Manager [role](#).

2.



Click **Manage** > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.

The screenshot shows the "License Management" page with the "License Terms" tab selected. At the top right, there are tabs for "Licenses", "License Families", and "License Terms". Below the tabs is a "+ Create Term" button and a "Categories" dropdown. A search bar labeled "Filter license terms..." and an "Add Filter" button are also present. The main content is a table with the following columns: Term, Source, Responsibility, Category, and Last Updated. The table lists 16 license terms with their respective sources, responsibilities, categories, and last update dates.

Term	Source	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Select the license term to open the Edit a License Term dialog box.

The screenshot shows the "Edit a License Term" dialog box. It has a title bar with a close button. The form contains the following fields and controls:

- Name ***: Text input field containing "Internal Use".
- Description ***: Text area containing "For internal use only".
- Responsibility ***: Radio button group with three options: "Required" (selected), "Forbidden", and "Permitted".
- Category ***: Dropdown menu showing "Internal".

At the bottom of the dialog are "Cancel" and "Save" buttons.

4. Edit the information in the dialog box and click **Save**.

Deleting a license term

You can only delete custom license terms.

You cannot delete a KnowledgeBase license terms. Instead you can [deactivate a KnowledgeBase license term](#) so that the term does not apply to a specific license.

Only users with the License Manager role can delete license terms.

You cannot delete a custom license term that is associated to a license.

To delete a license term:

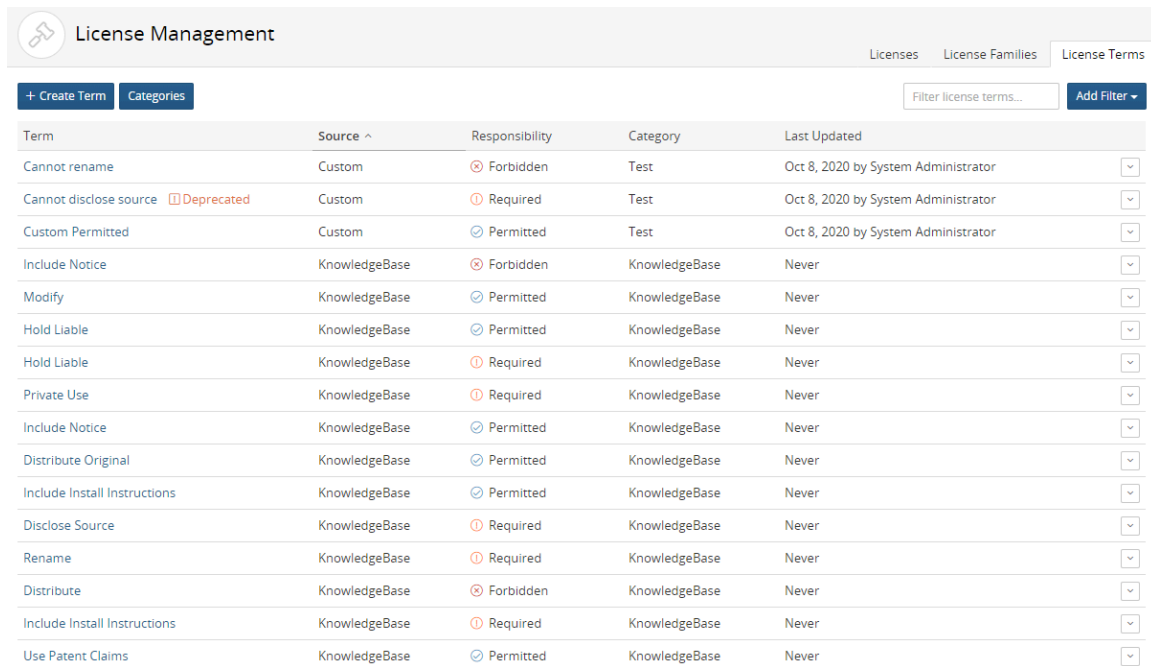
1. Log in to Black Duck with the License Manager [role](#).

- 2.

Click  > **Licenses**.


The License Management page appears.

Select the **License Terms** tab to display all license terms.



The screenshot shows the 'License Management' page with the 'License Terms' tab selected. The table lists various license terms with their sources, responsibilities, categories, and last updated dates. A 'Filter license terms...' search box and an 'Add Filter' button are visible at the top right of the table area.

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source ❌ Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click  in the row of the license term and select **Delete**.

The Delete a License Term dialog box appears.

4. Click **Delete** to confirm.

Deprecating or removing the deprecation status of a custom license term

You can deprecate a custom license term. Deprecating a custom license term is a global action – it applies to all licenses (custom and KnowledgeBase) that have this custom license term associated to it.

A deprecated custom license term is not available for new associations to licenses and cannot be edited. Existing licenses that have the deprecated term will still display the term to users in existing or new projects/components with no indication to these users that the term is deprecated.

12. Managing open source licenses • About license terms

Only users with the License Manager role can deprecate license terms.

To deprecate a custom license term:

Use these procedures to deprecate the term for *all* licenses that have this term associated to it.

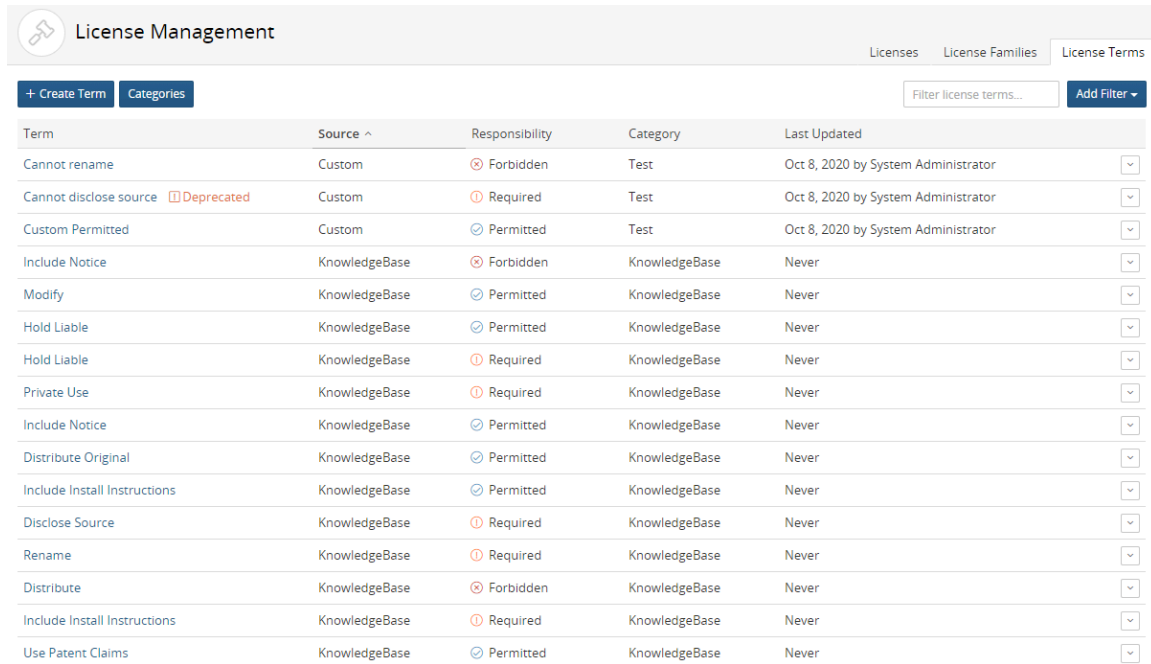
1. Log in to Black Duck with the License Manager [role](#).

2.

Click  > **Licenses**.

The License Management page appears.


Select the **License Terms** tab to display all license terms.



The screenshot shows the License Management interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there is a '+ Create Term' button and a 'Categories' dropdown. A search bar labeled 'Filter license terms...' and an 'Add Filter' button are also present. The main content is a table with the following columns: Term, Source, Responsibility, Category, and Last Updated. The table lists various license terms, including 'Cannot rename', 'Cannot disclose source' (marked as deprecated), 'Custom Permitted', 'Include Notice', 'Modify', 'Hold Liable', 'Private Use', 'Distribute Original', 'Distribute', and 'Use Patent Claims'. Each row includes a dropdown arrow in the 'Last Updated' column.

Term	Source	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3.

Click  in the row of the license term and select **Deprecate**.

The Deprecate a License Term dialog box appears.

4. Click **Deprecate** to confirm.

The date and username of the user who deprecated this term appears in the **Last Updated** column.

The Deprecated label appears next to the license term where the term appears in the **License Terms** tabs in License Management.

Note that the Deprecated label does not appear to the BOM manager for any licenses that have this term associated to it.

To undo the deprecation status of a custom license term:

1. Log in to Black Duck with the License Manager [role](#).

2.




Click **Manage** > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.


License Management					
		Licenses		License Families	License Terms
+ Create Term		Categories		Filter license terms...	Add Filter
Term	Source ^	Responsibility	Category	Last Updated	
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator	
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator	
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator	
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never	
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never	
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never	
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never	
Private Use	KnowledgeBase	Required	KnowledgeBase	Never	
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never	
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never	
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never	
Rename	KnowledgeBase	Required	KnowledgeBase	Never	
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never	
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never	

3.

Click  in the row of the license term and select **Remove Deprecated Status**.

The Deprecate a License Term dialog box appears.

4.

Click **Remove Deprecated Status** to confirm. The  label is removed from the license term.

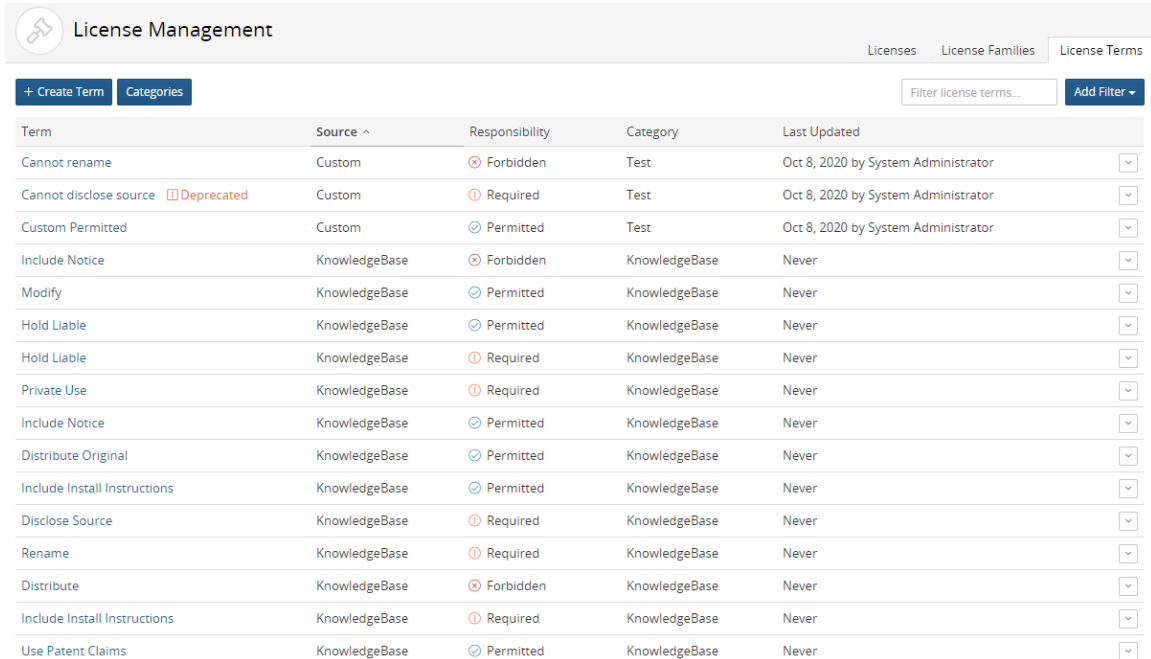
Removing a license term

Use these procedures to remove a license term that you associated to a custom license or a KnowledgeBase license. When you remove a license term from a license, the term no longer appears to users viewing license terms, for example when BOM Managers view license information in the BOM.

There are two methods you can use to remove a license term from a license:

- Using the **License Terms** tab which lists all license terms

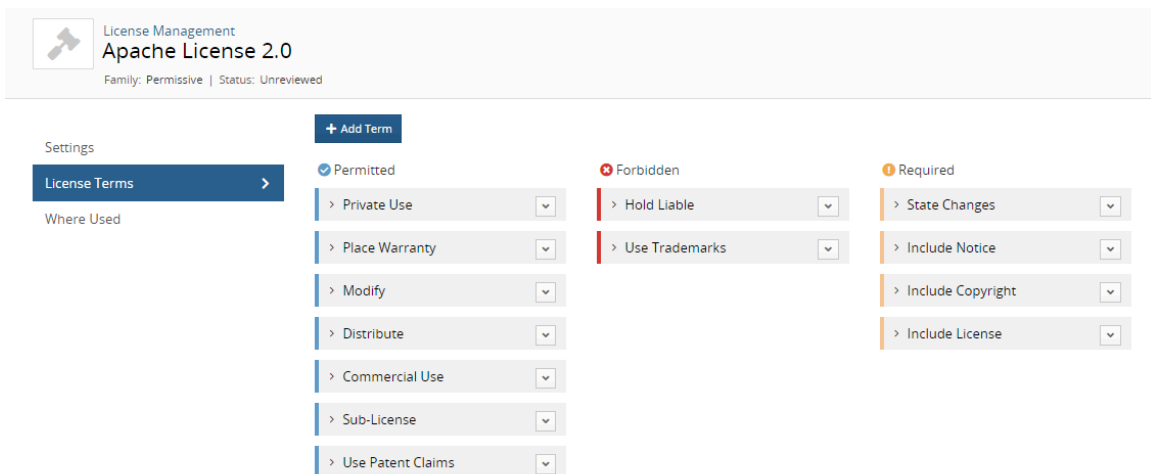
12. Managing open source licenses • About license terms



The screenshot shows the 'License Management' interface with the 'License Terms' tab selected. The table lists various license terms with their source, responsibility, category, and last updated date.

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

- Using the **License Terms** tab for an individual license



The screenshot shows the 'License Management' interface for an individual license, 'Apache License 2.0'. The 'License Terms' tab is selected, and the 'Where Used' section is visible. The interface shows a list of license terms and their associated responsibilities.

Settings
License Terms >
Where Used

+ Add Term

- Permitted
 - Private Use
 - Place Warranty
 - Modify
 - Distribute
 - Commercial Use
 - Sub-License
 - Use Patent Claims
- Forbidden
 - Hold Liable
 - Use Trademarks
- Required
 - State Changes
 - Include Notice
 - Include Copyright
 - Include License

To remove a license term from one or more licenses :

Use this method to remove a term from many licenses or if you want to view all the licenses to which this term is associated.

1. Log in to Black Duck with the License Manager [role](#).


2.

Click  > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.

License Management				
Licenses License Families License Terms				
+ Create Term Categories				
Filter license terms... Add Filter				
Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click  in the row of the license term and select **License Association**.

The License Association dialog box appears showing all licenses that have this license terms associated to it.

License Association ✕





Term	Source	Responsibility	Category
Private Use	KnowledgeBase	Permitted	KnowledgeBase

› Description

Select Licenses *

Add


Require Fulfillment Remove Fulfillment Requirement

<input type="checkbox"/> License	Fulfillment Required	
<input type="checkbox"/> MIT License	-	
<input type="checkbox"/> Artistic License 2.0	-	
<input type="checkbox"/> Apache License 2.0	-	
<input type="checkbox"/> Eclipse Public License 1.0	-	

Displaying 1-4 of 4

Close

12. Managing open source licenses • About license terms

4. Click  in the row of the license you want to disassociate from this license term.
5. Select **Delete** to confirm.

The term is removed from the license.

To remove a license term from a single license:

Use this method to remove a license term when viewing all terms for a single license.

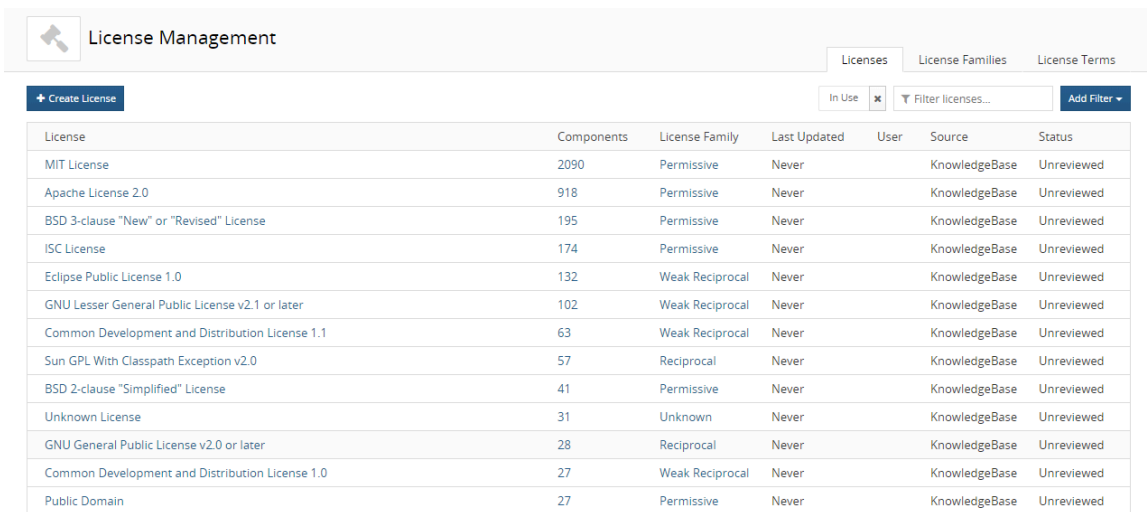
1. Log in to Black Duck with the License Manager [role](#).

2.



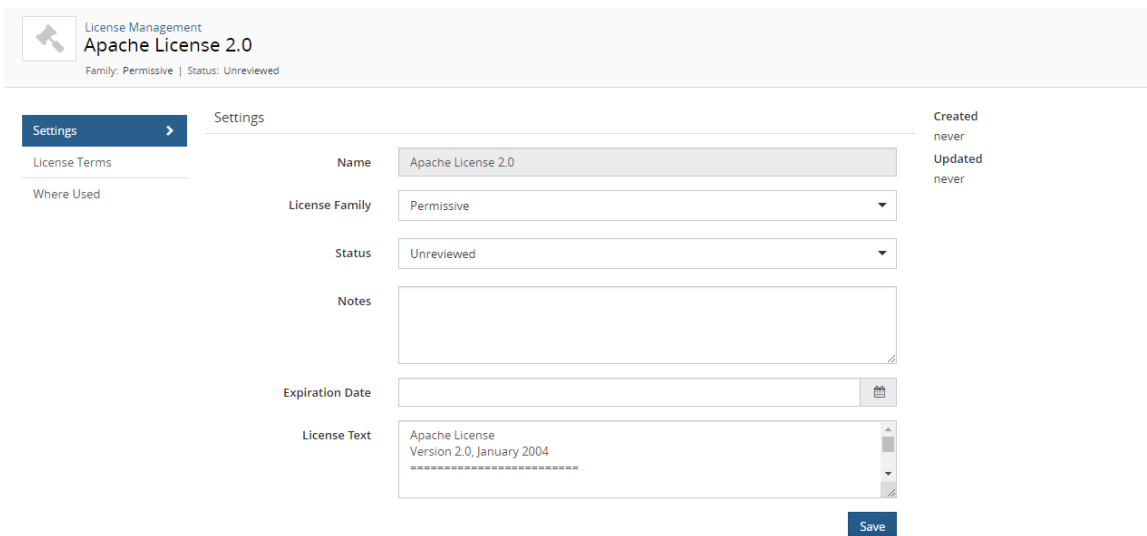
Click **Manage** > **Licenses**.

The License Management page appears.

A screenshot of the License Management page. At the top, there's a header "License Management" with a wrench icon. Below it are tabs for "Licenses", "License Families", and "License Terms". A "Create License" button is on the left. On the right, there are filters for "In Use" (checked), "Filter licenses...", and "Add Filter". The main content is a table with columns: License, Components, License Family, Last Updated, User, Source, and Status. The table lists various licenses like MIT License, Apache License 2.0, BSD 3-clause, etc., with their respective component counts and statuses.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. In the **Licenses** tab, select the license name to display the *License Name Settings* tab.

A screenshot of the License Name Settings page for "Apache License 2.0". The page shows fields for Name, License Family, Status, Notes, Expiration Date, and License Text. The "Name" field is "Apache License 2.0", "License Family" is "Permissive", and "Status" is "Unreviewed". The "License Text" field contains the text "Apache License Version 2.0, January 2004". There are "Created", "Updated", and "never" indicators on the right. A "Save" button is at the bottom right.

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings

License Terms

Where Used

Name: Apache License 2.0

License Family: Permissive

Status: Unreviewed

Notes

Expiration Date

License Text: Apache License Version 2.0, January 2004

Created: never
Updated: never

Save

4. Select the **License Terms** tab to view the terms associated with this tab.

License Management
Apache License 2.0
 Family: Permissive | Status: Unreviewed

+ Add Term

Settings
 License Terms >
 Where Used

Permitted


- > Private Use
- > Place Warranty
- > Modify
- > Distribute
- > Commercial Use
- > Sub-License
- > Use Patent Claims

Forbidden

- > Hold Liable
- > Use Trademarks

Required

- > State Changes
- > Include Notice
- > Include Copyright
- > Include License

5. Click  in the row of the license term of the term you wish to remove and select **Remove**.
The Remove Term dialog box appears.
6. Click **Remove** to confirm.
The **License Terms** tab displays the terms for this license with the term removed.

Deactivating a KnowledgeBase term

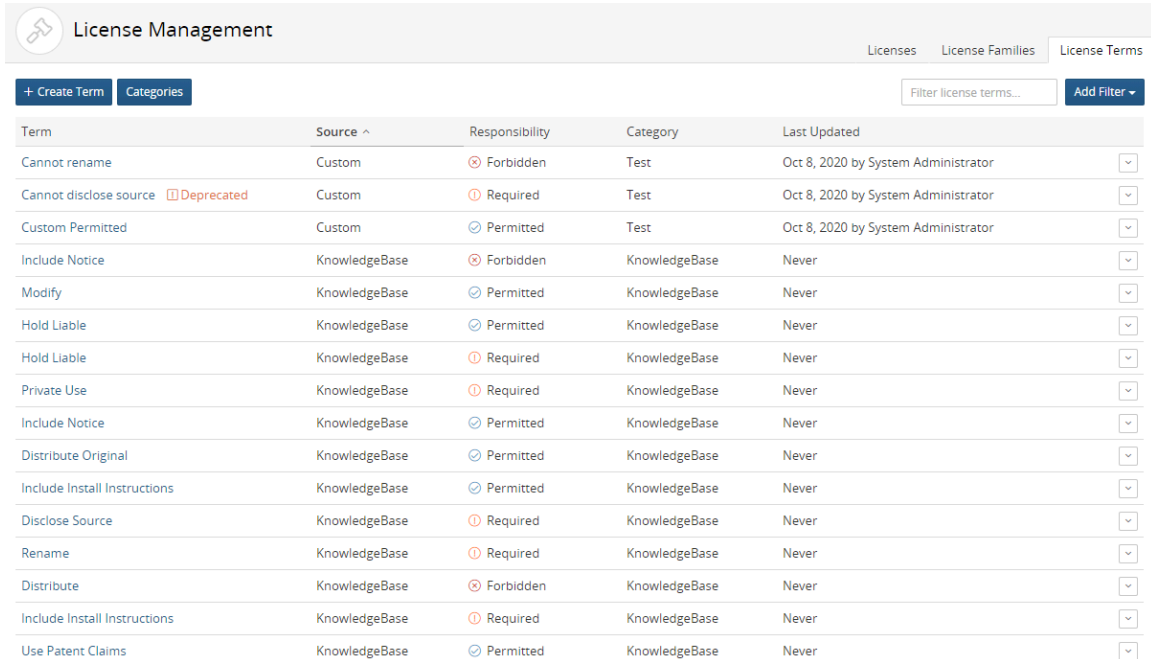
You may decide not to show your users specific license terms that are defined by Black Duck KnowledgeBase.

When a term is deactivated, it does not appear when users view the terms for a KnowledgeBase license; for example, when BOM Managers view the license terms in the BOM.

There are two methods you can use to deactivate a KnowledgeBase license term:

- Using the **License Terms** tab which lists all license terms

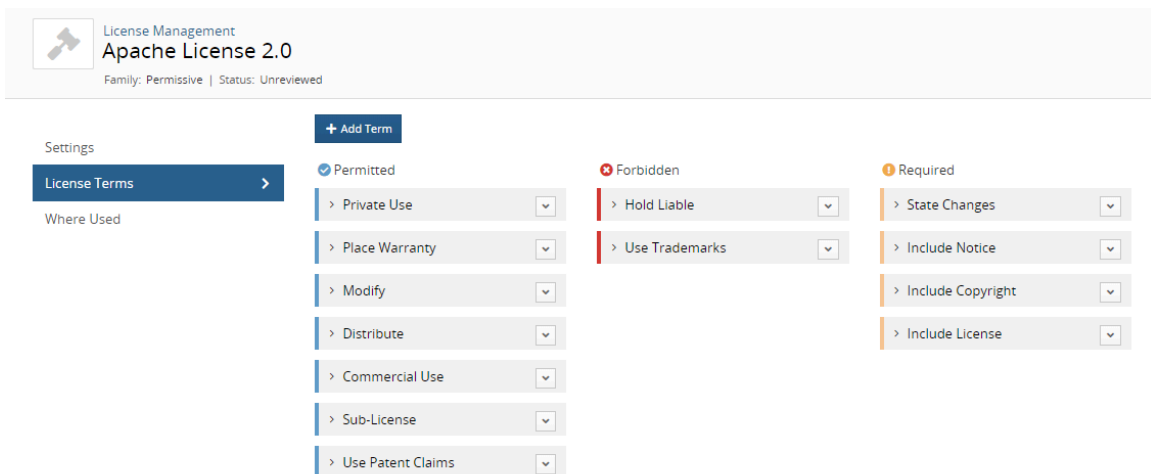
12. Managing open source licenses • About license terms



The screenshot shows the 'License Management' interface with the 'License Terms' tab selected. The table lists various license terms with their source, responsibility, category, and last updated date.

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

- Using the **License Terms** tab for an individual license



The screenshot shows the 'License Management' interface for an individual license, 'Apache License 2.0'. The 'License Terms' tab is selected, and the 'Where Used' section is expanded. The terms are categorized by responsibility: Permitted, Forbidden, and Required.

Settings
License Terms >
Where Used

+ Add Term

- Permitted
 - Private Use
 - Place Warranty
 - Modify
 - Distribute
 - Commercial Use
 - Sub-License
 - Use Patent Claims
- Forbidden
 - Hold Liable
 - Use Trademarks
- Required
 - State Changes
 - Include Notice
 - Include Copyright
 - Include License

Deactivated KnowledgeBase license terms can [be restored](#).

To deactivate a KnowledgeBase license term when viewing all terms :

1. Log in to Black Duck with the License Manager [role](#).

2.

Click  > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.

License Management				
Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	⊗ Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source ⊠ Deprecated	Custom	⊙ Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	⊙ Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	⊗ Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	⊙ Required	KnowledgeBase	Never
Private Use	KnowledgeBase	⊙ Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	⊙ Required	KnowledgeBase	Never
Rename	KnowledgeBase	⊙ Required	KnowledgeBase	Never
Distribute	KnowledgeBase	⊗ Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	⊙ Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	⊙ Permitted	KnowledgeBase	Never

3.

Click  in the row of the license term and select **License Association**.

The License Association dialog box appears showing all licenses that have this license terms associated to it.

×
License Association

Term	Source	Responsibility	Category
Private Use	KnowledgeBase	Permitted	KnowledgeBase

› Description

Select Licenses *

Add


Require Fulfillment Remove Fulfillment Requirement

License	Fulfillment Required
<input type="checkbox"/> MIT License	-
<input type="checkbox"/> Artistic License 2.0	-
<input type="checkbox"/> Apache License 2.0	-
<input type="checkbox"/> Eclipse Public License 1.0	-

Displaying 1-4 of 4

Close

12. Managing open source licenses • About license terms

- Click  in the row of the license you want to disassociate to this license term.
- Select **Deactivate** to confirm. The license term is no longer associated to that license.

✕
License Association





Term Private Use	Source KnowledgeBase	Responsibility Permitted	Category KnowledgeBase
----------------------------	--------------------------------	------------------------------------	----------------------------------

> Description

Select Licenses ▾

Add

Require Fulfillment Remove Fulfillment Requirement

<input type="checkbox"/> License	Fulfillment Required
<input type="checkbox"/> MIT License 	-
<input type="checkbox"/> Artistic License 2.0 	-
<input type="checkbox"/> Apache License 2.0  Restore	-
<input type="checkbox"/> Eclipse Public License 1.0 	-

Displaying 1-4 of 4

Close

To deactivate a KnowledgeBase license term when viewing a license:

- Log in to Black Duck with the License Manager [role](#).
-

Click  > **Licenses**.

The License Management page appears.

← License Management
 Licenses | License Families | License Terms

+ Create License
In Use ✕ Add Filter ▾

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

- In the **Licenses** tab, select the license name to display the *License Name Settings* tab.

The screenshot shows the 'License Management' interface for 'Apache License 2.0'. The page header includes 'License Management', 'Apache License 2.0', and 'Family: Permissive | Status: Unreviewed'. On the left, there is a sidebar with 'Settings' selected, and sub-tabs for 'License Terms' and 'Where Used'. The main content area is titled 'Settings' and contains the following fields:


- Name:** Apache License 2.0
- License Family:** Permissive (dropdown)
- Status:** Unreviewed (dropdown)
- Notes:** (empty text area)
- Expiration Date:** (empty date field with a calendar icon)
- License Text:** Apache License Version 2.0, January 2004 (text area with a scrollbar)

On the right side, there are two status indicators: 'Created never' and 'Updated never'. A 'Save' button is located at the bottom right of the settings area.

- Select the **License Terms** tab to view the terms associated with this tab.

The screenshot shows the 'License Management' interface for 'Apache License 2.0' with the 'License Terms' tab selected. The page header is the same as in the previous screenshot. The sidebar shows 'Settings' and 'License Terms' selected. The main content area is titled 'License Terms' and features an '+ Add Term' button at the top. Below it, there are three columns of terms categorized by status:

- Permitted (blue circle icon):**
 - > Private Use (dropdown)
 - > Place Warranty (dropdown)
 - > Modify (dropdown)
 - > Distribute (dropdown)
 - > Commercial Use (dropdown)
 - > Sub-License (dropdown)
 - > Use Patent Claims (dropdown)
- Forbidden (red circle icon):**
 - > Hold Liable (dropdown)
 - > Use Trademarks (dropdown)
- Required (orange circle icon):**
 - > State Changes (dropdown)
 - > Include Notice (dropdown)
 - > Include Copyright (dropdown)
 - > Include License (dropdown)

- Click  next to the KnowledgeBase license term you wish to deactivate and select **Deactivate**. The Deactivate Term dialog box appears.
- Click **Deactivate** to confirm. The **License Terms** tab displays the term as deactivated.

12. Managing open source licenses • About license terms

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

+ Add Term

Settings

License Terms >

Where Used

- Permitted
 - Private Use
 - Place Warranty
 - Modify
 - Distribute
 - Commercial Use
 - Sub-License
 - Use Patent Claims
- Forbidden
 - Hold Liabile
 - Use Trademarks
- Required
 - State Changes
 - Include Notice
 - Include Copyright
 - Include License

Restoring a KnowledgeBase license term

Use these procedures to restore a KnowledgeBase license term that you previously [deactivated](#).

There are two methods you can use to restore a KnowledgeBase license term:

- Using the **License Terms** tab which lists all license terms

License Management

Licenses License Families License Terms

+ Create Term Categories

Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liabile	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liabile	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

- Using the **License Terms** tab for an individual license

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings
License Terms >
Where Used

+ Add Term

Permitted

- > Private Use
- > Place Warranty
- > Modify
- > Distribute
- > Commercial Use
- > Sub-License
- > Use Patent Claims

Forbidden

- > Hold Liabile
- > Use Trademarks

Required

- > State Changes
- > Include Notice
- > Include Copyright
- > Include License

To restore a KnowledgeBase license term when viewing all terms:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.


Select the **License Terms** tab to display all license terms.

License Management

Licenses License Families License Terms

+ Create Term Categories Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source ❌ Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liabile	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liabile	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click  in the row of the KnowledgeBase license term and select **License Association**.

The License Association dialog box appears showing all licenses that have this license terms associated to it.

12. Managing open source licenses • About license terms

License Association ✕

Term	Source	Responsibility	Category
Private Use	KnowledgeBase	Permitted	KnowledgeBase

> Description

Select Licenses +

[Add](#)

[Require Fulfillment](#) [Remove Fulfillment Requirement](#)

<input type="checkbox"/> License	Fulfillment Required	
<input type="checkbox"/> MIT License	-	
<input type="checkbox"/> Artistic License 2.0	-	
<input type="checkbox"/> Apache License 2.0	-	Restore
<input type="checkbox"/> Eclipse Public License 1.0	-	

Displaying 1-4 of 4

[Close](#)

4. Click **Restore** in the row of the license(s) you want to restore.
The license term is enabled for this license.

License Association ✕

Term Private Use	Source KnowledgeBase	Responsibility Permitted	Category KnowledgeBase
----------------------------	--------------------------------	------------------------------------	----------------------------------

> Description

Select Licenses *
Add

Require Fulfillment Remove Fulfillment Requirement

<input type="checkbox"/>	License	Fulfillment Required	
<input type="checkbox"/>	MIT License	-	
<input type="checkbox"/>	Artistic License 2.0	-	
<input type="checkbox"/>	Apache License 2.0	-	
<input type="checkbox"/>	Eclipse Public License 1.0	-	

Displaying 1-4 of 4

Close

To restore a KnowledgeBase license term when viewing a license:

1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.

License Management ✕

Licenses License Families License Terms

+ Create License
In Use ✕
Filter licenses...
Add Filter ▾

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed


3. In the **Licenses** tab, select the license name to display the *License Name Settings* tab.

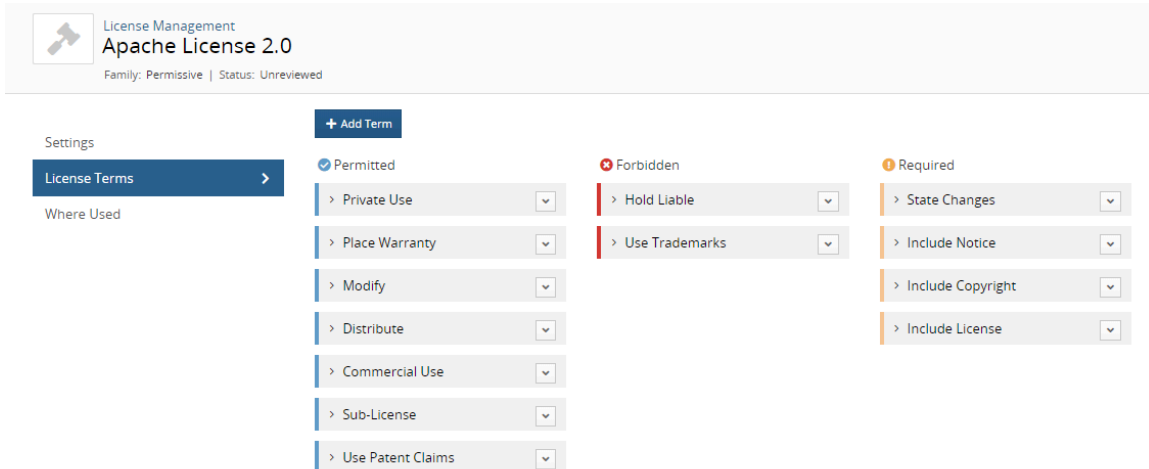
12. Managing open source licenses • About license terms

The screenshot shows the 'Settings' page for the 'Apache License 2.0' in a 'License Management' system. The page header includes the license name and its family/status: 'Family: Permissive | Status: Unreviewed'. On the left, there is a navigation menu with 'Settings' selected, and sub-items for 'License Terms' and 'Where Used'. The main content area is titled 'Settings' and contains several form fields: 'Name' (Apache License 2.0), 'License Family' (Permissive), 'Status' (Unreviewed), 'Notes' (empty text area), 'Expiration Date' (empty date field), and 'License Text' (containing the license text: 'Apache License Version 2.0, January 2004'). On the right side, there are two status indicators: 'Created never' and 'Updated never'. A 'Save' button is located at the bottom right of the form.

4. Select the **License Terms** tab to view the terms associated with this tab.

The screenshot shows the 'License Terms' page for the 'Apache License 2.0'. The page header is the same as the previous screenshot. The left navigation menu now has 'License Terms' selected. The main content area is titled '+ Add Term' and displays a list of license terms organized into three categories: 'Permitted' (with a blue checkmark icon), 'Forbidden' (with a red 'x' icon), and 'Required' (with an orange 'i' icon). Each category contains several terms, each with a dropdown arrow to its right. The 'Permitted' terms include: Private Use, Place Warranty, Modify, Distribute, Commercial Use, Sub-License, and Use Patent Claims. The 'Forbidden' terms include: Hold Liable and Use Trademarks. The 'Required' terms include: State Changes, Include Notice, Include Copyright, and Include License.

5. Click  next to the KnowledgeBase license term you wish to activate and select **Restore**.
The **License Terms** tab displays the terms for this license with the term restored.



KnowledgeBase licenses

Editing a KnowledgeBase license

KnowledgeBase licenses can be edited by users with the License Manager role and by users with the BOM Manager, or Project Manager role:

- License Managers can make *global* edits to KnowledgeBase licenses. The License Manager can edit the license family, license text, and other license settings. License Managers can also edit the [license terms](#). The license name *cannot* be changed.


These edits are propagated to BOMs with components using the KnowledgeBase license.

- BOM Managers and Project Managers can only make *local* edits to the license text of a KnowledgeBase license used in a BOM.

These edits only apply to the version of the KnowledgeBase license used in the BOM.

When the License Manager edits a KnowledgeBase license:

- Edits to the license family and license terms are always propagated to the KnowledgeBase licenses used in BOMs.
- Edits to the license text *may or may not* be propagated to the KnowledgeBase licenses used in BOMs:
 - If the BOM Manager/Project Manager *edited the license text*, the edits made by the License Manager *are not* propagated to the version of the KnowledgeBase license used in the BOM.
 - If the BOM Manager/Project Manager *did not edit* the license text, the edits made by the License Manager *are* propagated to the KnowledgeBase license used in the BOM.

 **Note:** KnowledgeBase updates may modify existing KnowledgeBase licenses. However, if a KnowledgeBase license has been edited by a License Manager or BOM Manager, then modifications to a KnowledgeBase license due to KnowledgeBase updates are not propagated globally (if the License Manager has edited this license) or to the edited local version (if the BOM Manager has modified this license).

1. Log in to Black Duck with the License Manager role.

2.



The License Management page appears.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed

3. Select the KnowledgeBase license name to display the *License Name Settings* tab.

License Management
Apache License 2.0
Family: Permissive | Status: Unreviewed

Settings

- Settings
- License Terms
- Where Used

Name: Apache License 2.0

License Family: Permissive

Status: Unreviewed

Notes: [Empty text area]

Expiration Date: [Calendar icon]

License Text: Apache License
Version 2.0, January 2004

Created: never
Updated: never

Save

4. Modify the information:

- **Name:** License name. Note that this field is read-only.
- **License Family:** Use the drop-down selector to choose the license family.
- **Status:** Use the drop-down selector to choose the license status.
- **Notes:** You can type any text in this field. Use this for additional information or helpful notes.
- **Expiration Date:** Use the calendar tool to set the expiration date.
- **License Text:** The actual license as found in the component.

5. Click **Save**.

In the License Management page, the source for this license changes to **Modified KnowledgeBase** with the username of the user who edited this license listed in the **User** column and the time the license was modified listed in the **Last Updated** column.

KnowledgeBase licenses can be [restored](#) to their original values.

Restoring the original text and family of a KnowledgeBase license

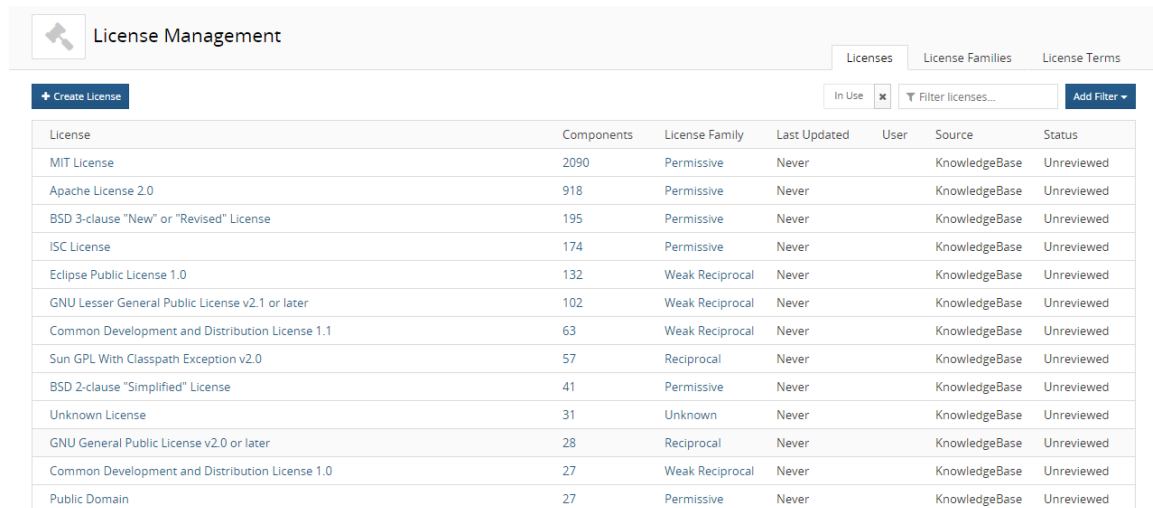
If a user with the License Manager role has modified the text or license family of a KnowledgeBase license, they can restore that license to its original values, as defined by the Black Duck KnowledgeBase.

To restore a KnowledgeBase license:

1. Log in to Black Duck with the License Manager role.
- 2.

Click  > **Licenses**.


The License Management page appears.



The screenshot shows the License Management page with a table of licenses. The table has columns for License, Components, License Family, Last Updated, User, Source, and Status. The table lists various licenses such as MIT License, Apache License 2.0, BSD 3-clause, etc.

License	Components	License Family	Last Updated	User	Source	Status
MIT License	2090	Permissive	Never		KnowledgeBase	Unreviewed
Apache License 2.0	918	Permissive	Never		KnowledgeBase	Unreviewed
BSD 3-clause "New" or "Revised" License	195	Permissive	Never		KnowledgeBase	Unreviewed
ISC License	174	Permissive	Never		KnowledgeBase	Unreviewed
Eclipse Public License 1.0	132	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
GNU Lesser General Public License v2.1 or later	102	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.1	63	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Sun GPL With Classpath Exception v2.0	57	Reciprocal	Never		KnowledgeBase	Unreviewed
BSD 2-clause "Simplified" License	41	Permissive	Never		KnowledgeBase	Unreviewed
Unknown License	31	Unknown	Never		KnowledgeBase	Unreviewed
GNU General Public License v2.0 or later	28	Reciprocal	Never		KnowledgeBase	Unreviewed
Common Development and Distribution License 1.0	27	Weak Reciprocal	Never		KnowledgeBase	Unreviewed
Public Domain	27	Permissive	Never		KnowledgeBase	Unreviewed


3. Do one of the following:

- Click  and select **Restore** in the row of the KnowledgeBase license that you want to restore to display the Restore KnowledgeBase License dialog box.
- Select the KnowledgeBase license name to display the *License Name Settings* tab. In the **Restore KnowledgeBase License** section, click **Restore original**.

4. Click **Restore** in the Restore KnowledgeBase License dialog box.

In the License Management page, the source for this license reverts to **KnowledgeBase**.

- If the license family or text were the only changes made to the license (as defined on the **Settings** tab), the values in the **Last Updated** and **User** columns are removed.
- If additional changes were made (as defined on the **Settings** tab), the values in the **Last Updated** and **User** columns displays the date and username when the last of these changes occurred.

 **Note:** This procedure does not restore the KnowledgeBase *license terms* to their original values. Click [here](#) for more information on restoring KnowledgeBase license terms.

Viewing detected copyright statements

Black Duck can detect instances of copyright statements for a component. By enabling detection of copyright data when scanning code, users focused on license compliance can reduce license compliance risks by detecting and managing open source software and proprietary copyrights statements.

With this feature, Black Duck performs a search for copyright string text and displays the text found in the **Source** tab.

By displaying this information in the **Source** tab, you can easily find the files and directories that interest you and determine if copyright text is located there.

The screenshot shows the Black Duck interface for a project named "Sample Project" version 1.0. The "Source" tab is active, displaying search results for the path "/TutorialFiles/tools/# » tools". The interface is divided into three main sections: License Searches, Copyright Searches, and a file tree on the left.

License Searches:

- Licenses:** 3 Files. Includes "GNU General Public License Version 2 | 3 hits" with a blue progress bar.
- License References:** 3 Files. Includes "GNU General Public License | 6 hits" with a blue progress bar.

Copyright Searches:

- Copyrights:** 7 Files. Includes:
 - "Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc. | 5 hits" with a blue progress bar.
 - "Copyright (C) 2001, 2002, 2003 Free Software Foundation, Inc. | 1 hits" with a blue progress bar.
 - "Copyright 1997 Werner Koch (dd9jn) ; break; | 1 hits" with a blue progress bar.

The left sidebar shows a file tree with folders like "lib", "licenses", "src_ourfaces", and "tools". The "tools" folder is selected.

Black Duck groups the detected copyright statements into the **Copyright Searches** section.

For the copyright text found, Black Duck displays the number of:

- "Hits". The number of instances that copyright text was found in all files.
- Files where these "hits" were found.

In the example shown above, there were three instances of copyright text found in seven files.

Black Duck also lists the total number of files. Note that this value may not equal the total number of files shown for the copyright text as a file can have multiple different copyright statements.

Optionally, to help you review this information, upload your source files so that reviewers can view discovered copyright text from within the **Source** tab. When source files are uploaded, Black Duck provides a list of copyright statements. Select a copyright statement to highlight the text in the file. This can help reviewers evaluate the copyright text.

The screenshot shows a 'Discoveries' window with the following content:

We found these discoveries in this file: 1 License, 1 License Reference, 1 Copyright

Licenses	Hits
GNU General Public License Version 2	1

License References	Hits
GNU General Public License	2

Copyrights	Hits
Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.	1

file:///Users/sh/Downloads/Tutorial_Files/tools/bftest.c

```

1 /* bftest.c - Blowfish test program
2  * Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.
3  *
4  * This file is part of GnuPG.
5  *
6  * GnuPG is free software; you can redistribute it and/or modify
7  * it under the terms of the GNU General Public License as published by
8  * the Free Software Foundation; either version 2 of the License, or
9  * (at your option) any later version.
10 *
11 * GnuPG is distributed in the hope that it will be useful,
12 * but WITHOUT ANY WARRANTY; without even the implied warranty of
13 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
14 * GNU General Public License for more details.
15 *
16 * You should have received a copy of the GNU General Public License
17 * along with this program; if not, write to the Free Software
18 * Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
19 */
20
21 #include <config.h>
22 #include <stdio.h>
23 #include <stdlib.h>
24 #include <string.h>
25 #ifdef HAVE_DOSISH_SYSTEM
26 #include <dos.h>

```

Close

If you do not upload the source files, the Black Duck UI only displays the location of the discovered text in the file, by line number:

The screenshot shows a 'Discoveries' window with the following content:

We found these discoveries in this file: 0 License, 0 License Reference, 1 Copyright

Copyrights	Hits
Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.	1


file:///Users/sh/Tutorial_Files/tools/mk-tdata.c

No File to Display
You need to upload your files in order to display them

We found hits in these lines:
Hit 1: Line 2

Close

To include your source files, after your administrator has enabled source uploads, as described in the installation guide, include the upload source parameter when scanning.

 **Note:** Regardless whether you upload your source files or not, copyright detection cannot be completed offline as it requires communication with the Black Duck server.

Supported file extensions/ file names

Copyright text search occurs in file extensions such as .bat or .js and for these file names, or file names that include the following text, regardless of case:

- bdsl

- copying
- copyright
- control
- dad
- gpl
- install
- legal
- lgpl
- license
- licence
- licenses
- licences
- notice
- readme

Copyright detection process

The process to view copyright text is:

1. Enable detecting of copyright data when scanning and optionally, enable uploading source files for viewing copyright text within the file.
2. Review the copyright text.

Enable detecting of copyright data

All scanning methods have an option to enable copyright string search:

- Signature Scanner command line
- Black Duck Detect (Desktop) Black Duck Detect
- Black Duck Detect

Using the Signature Scanner command line

Use the **-copyright-search** parameter to enable copyright text search.

Click [here](#) for more information on using the command line.

Using Black Duck Detect (Desktop) or Black Duck Detect

A property to enable copyright detection will be available in Black Duck Detect version 6.4 and later.

Reviewing copyright data

Black Duck displays the location of these copyright statements in your code tree.

To review copyright text:

1. After enabling copyright text search, select the **Source** tab from your project version BOM page.
2. Select a folder in the code tree that you want to determine if there is copyright text.

Optionally, select **All Subfolders** to view information for all subfolders.

The table displays information in the table for the selected location. By default the **Files** option is selected.

Black Duck Projects
Sample Project » 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

shamnani-mac#/Users/shamnani...
blowfish.c
lib
licenses
openvpn.exe
samplefile1.h
samplefile2.c
src_jo
src_jo's files
src_ourfaces
src_pgsi
tools
bftest.c
clean-sat.c
fileone.c
filetwo.c
gpgsplit.c
mk-tdata.c
mpicalc.c
shmttest.c
Tutorial_Files_Default_Detect...

tools/# » tools

Files Discoveries All Subfolders

Edit Reset File Adjustments

Name	Component	Match Type	License	Usage	Discovery Types
bftest.c	GnuPG 0.2.1	Manually Identified	Unknown License	Dynamically Linked	Copyright, License Reference, License
clean-sat.c	GnuPG 0.2.1	Manually Identified	Unknown License	Dynamically Linked	Copyright
fileone.c					
filetwo.c					
gpgsplit.c					Copyright, License Reference, License
mk-tdata.c	GnuPG 0.2.1	Manually Identified	Unknown License	Dynamically Linked	Copyright
mpicalc.c	GnuPG 0.2.1	Manually Identified	Unknown License	Dynamically Linked	Copyright, License Reference, License
shmttest.c	GnuPG 0.2.1	Manually Identified	Unknown License	Dynamically Linked	Copyright

Displaying 1-8 of 8

3. Select **Discoveries** to view the list of copyright text, shown in the **Copyright Searches** section.

Black Duck Projects
Sample Project » 1.0
Project | Phase: In Planning | Scans: Up to Date | Status: Up to Date

Components Security Source Reports Details Settings

shamnani-mac#/Users/shamnani...
blowfish.c
lib
licenses
openvpn.exe
samplefile1.h
samplefile2.c
src_jo
src_jo's files
src_ourfaces
src_pgsi
tools
Tutorial_Files_Default_Detect...

tools/# » tools

Files Discoveries All Subfolders

License Searches

Licenses 3 Files

GNU General Public License Version 2 | 3 hits
3

License References 3 Files

GNU General Public License | 6 hits
3

Copyright Searches

Copyrights 7 Files

Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc. | 5 hits
5

Copyright (C) 2001, 2002, 2003 Free Software Foundation, Inc. | 1 hits
1

Copyright 1997 Werner Koch (dd9jn) ; break; | 1 hits
1

4. Select a copyright statement to view the **Source** tab filtered to display the files that contain the selected copyright text.

12. Managing open source licenses • Viewing detected copyright statements

The screenshot shows the Black Duck Projects interface for a project named 'Sample Project' (version 1.0). The left sidebar displays a file tree with the 'tools' folder selected. The main area shows a table of discovered files:

Name	Component	Match Type	License	Usage	Discovery Types
bftest.c					Copyright, License Reference, License
clean-sat.c					Copyright
mk-tdata.c					Copyright
mpicalc.c					Copyright, License Reference, License
shmttest.c					Copyright

At the bottom right of the table area, it says 'Displaying 1-5 of 5'.

Optionally, select a file name to view the location of the file in the code tree. If you uploaded your source files, the file contents appears on the page.

The screenshot shows the Black Duck Projects interface with the 'clean-sat.c' file selected. The main area displays the file's details:

Match Type	Component	License	Usage
Manually Identified	GnuPG 0.2.1	Unknown License	Dynamically Linked

Below the table, the file's content is displayed in a code editor:

```
1 /* clean-sat.c
2 * Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.
3 *
4 * This file is free software; as a special exception the author gives
5 * unlimited permission to copy and/or distribute it, with or without
6 * modifications, as long as this notice is preserved.
7 *
8 * This program is distributed in the hope that it will be useful, but
9 * WITHOUT ANY WARRANTY, to the extent permitted by law; without even the
10 * implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
11 */
12
13 #include <stdio.h>
14
15 int
16 main(int argc, char **argv)
17 {
18     int c;
19
20     if( argc > 1 ) {
21         fprintf(stderr, "no arguments, please\n");
22         return 1;
23     }
```

5. Select **Copyright** from the **Discovery Type** column to open the Discoveries dialog box.

12. Managing open source licenses • Viewing detected copyright statements

The Discoveries dialog box displays the following information:

We found these discoveries in this file: 1 License, 1 License Reference, 1 Copyright

Category	Hits
Licenses	Hits
GNU General Public License Version 2	1
License References	Hits
GNU General Public License	2
Copyrights	Hits
Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.	1

file:///Users/sh/Downloads/Tutorial_Files/tools/bftest.c

```
1 /* bftest.c - Blowfish test program
2  * Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.
3  *
4  * This file is part of GnuPG.
5  *
6  * GnuPG is free software; you can redistribute it and/or modify
7  * it under the terms of the GNU General Public License as published by
8  * the Free Software Foundation; either version 2 of the License, or
9  * (at your option) any later version.
10 *
11 * GnuPG is distributed in the hope that it will be useful,
12 * but WITHOUT ANY WARRANTY; without even the implied warranty of
13 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
14 * GNU General Public License for more details.
15 *
16 * You should have received a copy of the GNU General Public License
17 * along with this program; if not, write to the Free Software
18 * Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
19 */
20
21 #include <config.h>
22 #include <stdio.h>
23 #include <stdlib.h>
24 #include <string.h>
25 #ifdef HAVE_DOSISH_SYSTEM
26 #include <dos.h>
```

The Discoveries dialog box shows all copyright text found for the selected file. If embedded licenses and license references were also found, that text is also shown.

The information that appears here depends on whether you uploaded source files.

In the example shown above, source files were uploaded in the scan.

6. Select the copyright text to view the highlighted text.

The Discoveries dialog box displays the following information:

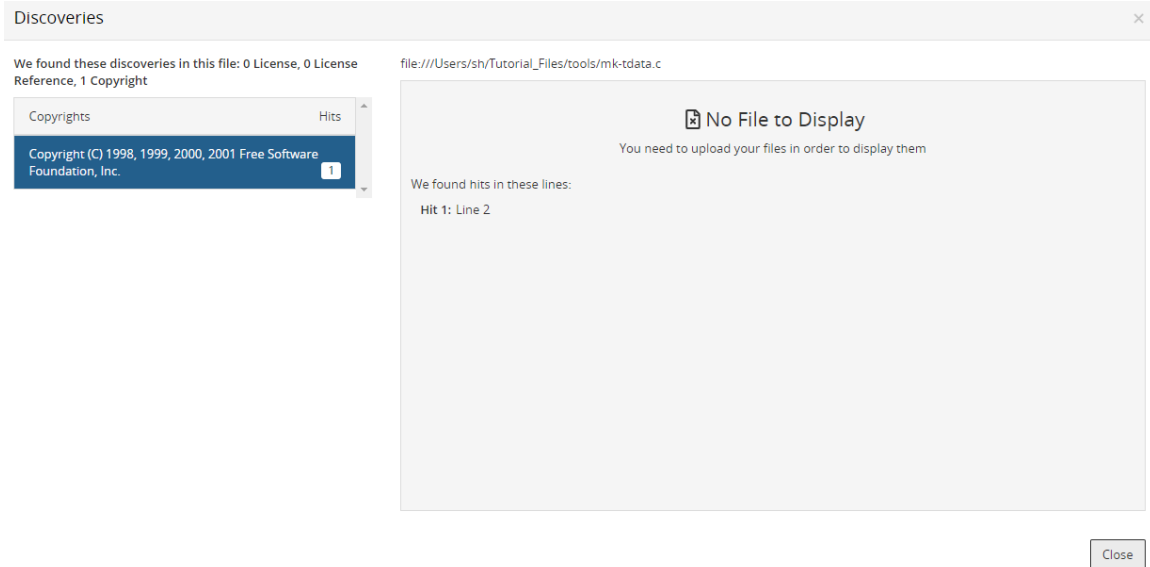
We found these discoveries in this file: 1 License, 1 License Reference, 1 Copyright

Category	Hits
Licenses	Hits
GNU General Public License Version 2	1
License References	Hits
GNU General Public License	2
Copyrights	Hits
Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.	1

file:///Users/sh/Downloads/Tutorial_Files/tools/bftest.c

```
1 /* bftest.c - Blowfish test program
2  * Copyright (C) 1998, 1999, 2000, 2001 Free Software Foundation, Inc.
3  *
4  * This file is part of GnuPG.
5  *
6  * GnuPG is free software; you can redistribute it and/or modify
7  * it under the terms of the GNU General Public License as published by
8  * the Free Software Foundation; either version 2 of the License, or
9  * (at your option) any later version.
10 *
11 * GnuPG is distributed in the hope that it will be useful,
12 * but WITHOUT ANY WARRANTY; without even the implied warranty of
13 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
14 * GNU General Public License for more details.
15 *
16 * You should have received a copy of the GNU General Public License
17 * along with this program; if not, write to the Free Software
18 * Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
19 */
20
21 #include <config.h>
22 #include <stdio.h>
23 #include <stdlib.h>
24 #include <string.h>
25 #ifdef HAVE_DOSISH_SYSTEM
26 #include <dos.h>
```

If you did not upload source files, the Discoveries dialog box displays the location of the discovered copyright text in the file, by line number:



Managing copyrights

Users with the Copyright Editor [role](#) can manage open source copyright statements for their organization. Using this feature makes it easier for you to include the full list of copyright holders for the open source components you use in your notices file report.

Users with the Copyright Editor role can:

- View all copyright statements for a component version.
- Create or edit custom copyright statements.
- Edit Black Duck KnowledgeBase copyright statements
- Revert an edited Black Duck KnowledgeBase copyright statement to its original text.
- Activate or deactivate copyright statements.


Black Duck manages copyright statements by the origin name/id for a component version. Therefore, edits made to copyright statements for an origin for a component version apply to all BOMs that use that component version origin. This enables you to reuse data across your organization and reduce your workload.

To manage copyright statements in Black Duck:

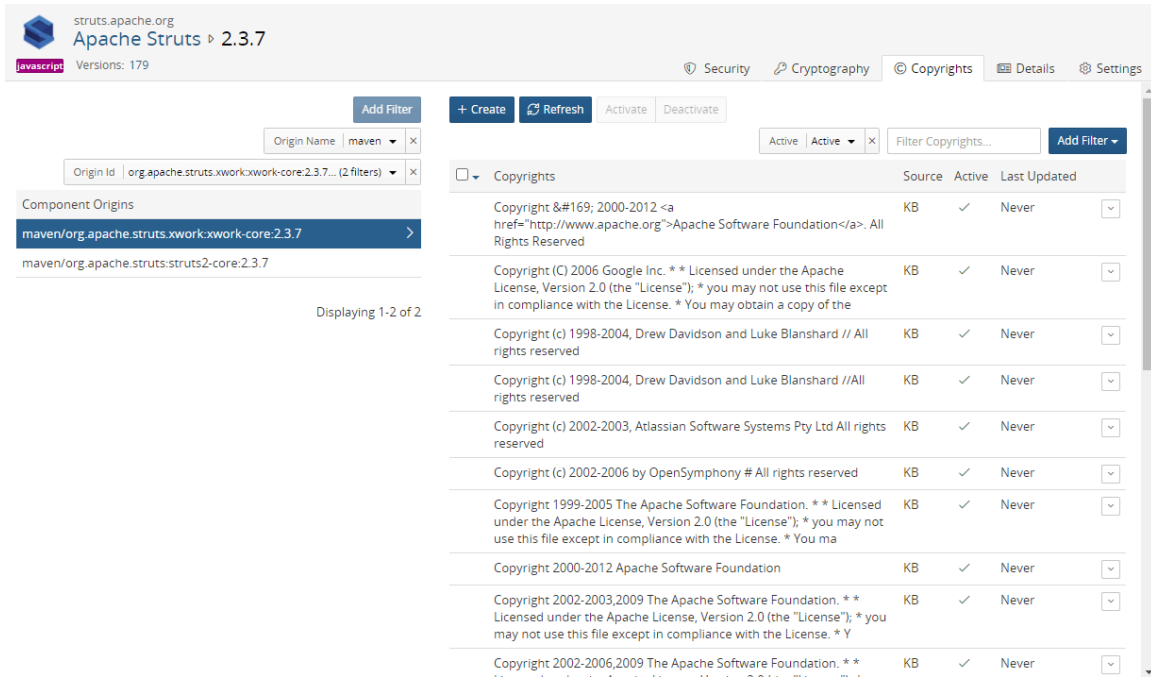
1. Review the existing Black Duck KnowledgeBase copyright statements.
2. If necessary, edit the existing KnowledgeBase copyright statements and/or create custom copyright statements.
3. Deactivate any copyright statements that do not apply.
4. Create the [Notices file report](#) and select the **Copyright Data** option to include copyright statements in your report.

Viewing and managing copyright statements

To view and manage the copyright statements, do one of the following:

In the project version BOM, click  in the row of the component version you wish to view copyright statements and select **Copyrights**.

The component version **Copyrights** tab appears filtered to display the copyright statements for the origin used in your BOM.

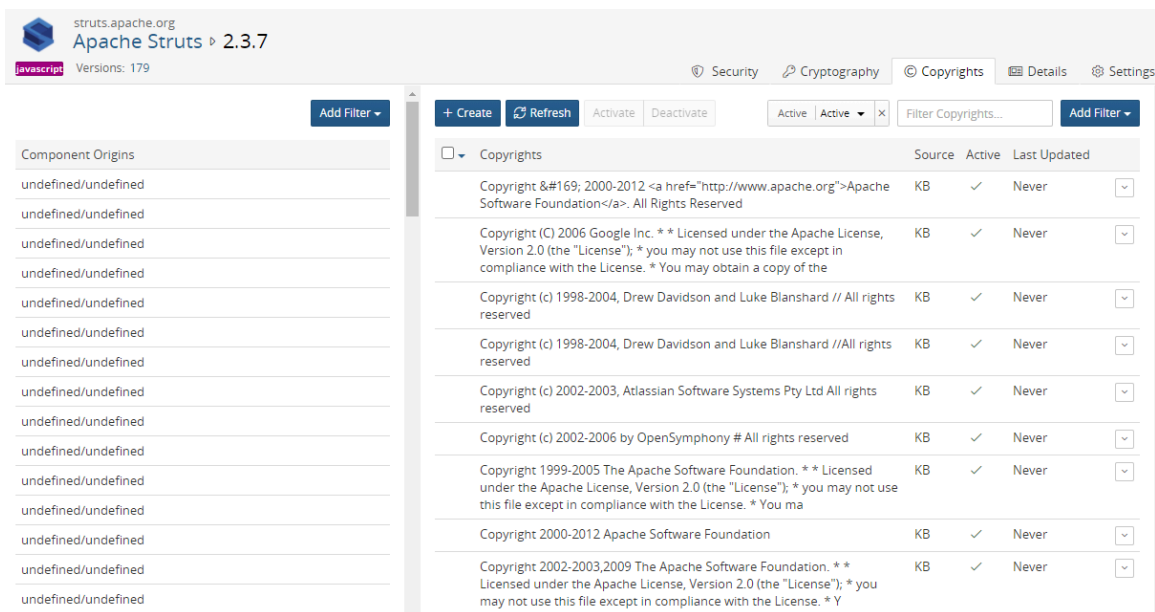


The screenshot shows the Apache Struts 2.3.7 component page. The 'Copyrights' tab is active, displaying a table of copyright statements. The table has columns for 'Source', 'Active', and 'Last Updated'. The statements are filtered to show only those for the selected origin.

Copyright	Source	Active	Last Updated
Copyright © 2000-2012 Apache Software Foundation. All Rights Reserved	KB	✓	Never
Copyright (C) 2006 Google Inc. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the	KB	✓	Never
Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard // All rights reserved	KB	✓	Never
Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard //All rights reserved	KB	✓	Never
Copyright (c) 2002-2003, Atlassian Software Systems Pty Ltd All rights reserved	KB	✓	Never
Copyright (c) 2002-2006 by OpenSymphony # All rights reserved	KB	✓	Never
Copyright 1999-2005 The Apache Software Foundation. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You ma	KB	✓	Never
Copyright 2000-2012 Apache Software Foundation	KB	✓	Never
Copyright 2002-2003,2009 The Apache Software Foundation. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * Y	KB	✓	Never
Copyright 2002-2006,2009 The Apache Software Foundation. **	KB	✓	Never

Note if a component version is not defined in the BOM (as shown by (?) for the version), the **Copyrights** option is not available.

- Select to view an open source component version and select the **Copyrights** tab.



The screenshot shows the Apache Struts 2.3.7 component page. The 'Copyrights' tab is active, displaying a list of copyright statements for all origins. The table has columns for 'Source', 'Active', and 'Last Updated'.



Copyright	Source	Active	Last Updated
Copyright © 2000-2012 Apache Software Foundation. All Rights Reserved	KB	✓	Never
Copyright (C) 2006 Google Inc. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You may obtain a copy of the	KB	✓	Never
Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard // All rights reserved	KB	✓	Never
Copyright (c) 1998-2004, Drew Davidson and Luke Blanshard //All rights reserved	KB	✓	Never
Copyright (c) 2002-2003, Atlassian Software Systems Pty Ltd All rights reserved	KB	✓	Never
Copyright (c) 2002-2006 by OpenSymphony # All rights reserved	KB	✓	Never
Copyright 1999-2005 The Apache Software Foundation. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * You ma	KB	✓	Never
Copyright 2000-2012 Apache Software Foundation	KB	✓	Never
Copyright 2002-2003,2009 The Apache Software Foundation. ** Licensed under the Apache License, Version 2.0 (the "License"); * you may not use this file except in compliance with the License. * Y	KB	✓	Never

The page is unfiltered and lists all origins for this component version.

Select an origin to view the copyright statements for that origin.

Use the component origin name and ID filters to limit the origins displayed on the page.

For each copyright statement, the following information appears:

Column	Description
Copyrights	Copyright text.
Source	Source for this copyright statement. Possible values are: <ul style="list-style-type: none"> • KB. An unmodified, active copyright statement from Black Duck KnowledgeBase. • KB Modified. A copyright statement from Black Duck KnowledgeBase that has been edited, deactivated, or reactivated. • Custom. Copyright statement created by a user with the Copyright Editor role.
Active	One of the following icons appears: <ul style="list-style-type: none"> •  Active copyright statement which will appear in your Notices File report. •  Inactive copyright statement which will not appear in your Notices File report.
Last Updated	One of the following appears: <ul style="list-style-type: none"> • Never indicates that the statement from Black Duck KnowledgeBase has never been modified. • Date and username. <ul style="list-style-type: none"> • For Black Duck KnowledgeBase copyright statements, the date when this copyright statement was modified and the responsible user. A date and username also appears for Black Duck KnowledgeBase copyright statements that have been deactivated or reverted back to their original text. • For custom copyright statements, the date when this statement was either created or last edited and the responsible user.

Creating custom copyright statements

To create a custom copyright statement:

1. As copyright statements are based by component origin, select the origin for this copyright statement from the **Component Origins** section.
2. Click **Create**. The Create Copyright dialog box appears.

3. Enter the copyright text and click **Save**.

Copyright statements are active by default. See below to deactivate this statement.

Editing custom copyright statements

To edit a custom copyright statement:

1. In the row of the copyright statement you want to edit, select and select **Edit**.
The Edit Copyright dialog box appears.

2. Edit the text and/or select or clear the **Active** option and click **Save**.

Deactivating copyright statements

By default, all copyright statements are active.

To deactivate a copyright statement:

1. Do one of the following:
 - Click in the row of the copyright statement you wish to deactivate and select **Deactivate**.
 - Select one or more checkboxes located to the left of the copyright statement and click **Deactivate**.

You can also deactivate a copyright statement when editing it.

Activating copyright statements

To activate copyright statements:

1. Do one of the following:

- Click in the row of the copyright statement you wish to activate and select **Activate**.
- Select one or more checkboxes located to the left of the copyright statement and click **Activate**.

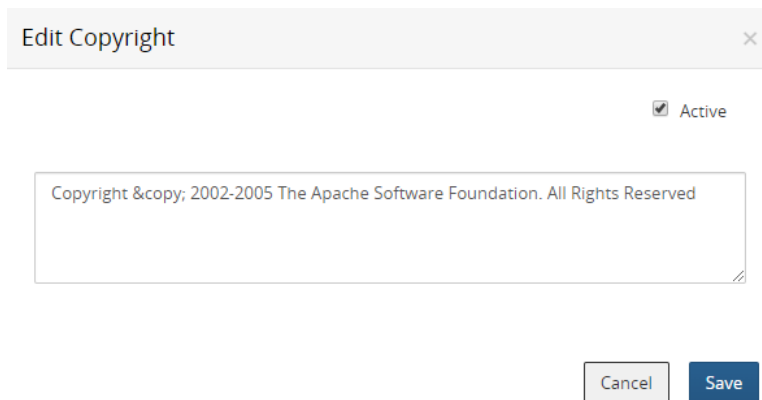
You can also activate a copyright statement when editing it.

Editing KnowledgeBase copyright statements

You can modify an existing Black Duck KnowledgeBase copyright statement.

To edit a KnowledgeBase copyright statement:

1. Click in the row of the copyright statement you wish to edit and select **Edit**.



If this is the initial attempt to edit a KnowledgeBase copyright statement, the option to revert to the original text is not available.

2. Edit the text and/or clear or select the **Active** option and click **Save**.


Reverting KnowledgeBase copyright statements

If you edited a KnowledgeBase copyright statement, you can revert to the original text of the KnowledgeBase copyright statement.

To revert a KnowledgeBase copyright statement:

1. Click in the row of the copyright statement you wish to edit and select **Edit**.

The dialog box displays the edited text and the original copyright text from the KnowledgeBase.

 **Note:** Reverted text may include poorly formatted and extraneous text not shown in the original copyright statement, which was edited to make it more readable.

2. Click **Revert to Original**.
3. Click **Save**.

Updating KnowledgeBase copyright statements

The Black Duck KnowledgeBase may have updated copyright information.

You can refresh the copyright statements for a component origin: if there is new or updated data, Black Duck updates the information shown while keeping any edits that you made.

To update KnowledgeBase copyright statements for an origin:

1. Open the **Copyrights** tab, as described previously.
2. Select a component origin.
3. Click **Refresh**.


Managing attribution statements

You may want to add an attribution statement to your Notices File report. An attribution statement is typically an acknowledgment to the copyright holder and is placed at the component version level.

To add an attribution statement:

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Select the version name to open the **Components** tab and view the BOM.
4. Select the license to open the *Component/Subproject Name Version* Component License dialog box.

12. Managing open source licenses • Managing attribution statements

 **Component Name** *Version* **Component License** ✕

Attribution Statement Include in Notices File Report

License Edit

Select a license to view. Edit Mode off

(Apache License 2.0)

Close Save Changes

5. Click > to open the **Attribution Statement** field and enter the text.

Delete the text in this field to remove an attribution statement.

6. Click **Save Changes**.

The attribution statement appears after the component name/version in the Components table in the [Notices File report](#). This example is from the HTML version of the report:

Sample Project A - 4.0 Notices File

Phase: In Planning | Distribution: External

Notices Report Content

- License Data
- License Text
- Origin Copyright Text

Components

Component	License	Component Link
Apache Log4j API 2.17.1	Apache License 2.0	http://logging.apache.org/log4j/2.x/log4j-api/
Apache Tomcat 10.0.20	Apache License 2.0	http://tomcat.apache.org/

Copyright Data

Apache Log4j API 2.17.1 - maven:org.apache.logging.log4j:log4j-api:2.17.1
<http://logging.apache.org/log4j/2.x/log4j-api/>

- Copyright 1969-1999 The Apache Software Foundation

Apache Tomcat 10.0.20 - maven:org.apache.tomcat:tomcat-jasper-el:10.0.20
<http://tomcat.apache.org/>

- Copyright 1999-2022 The Apache Software Foundation This product includes software developed at

Licenses

Apache License 2.0

Apache Log4j API 2.17.1, Apache Tomcat 10.0.20

```
Apache License
Version 2.0, January 2004
*****

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensors" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.
```


About license conflicts

License terms are the provisions in the license which grant rights or impose restrictions on the use of the software under that license. They indicate the things you can do (permitted), cannot do (forbidden), and must do (required) to comply with the license.

License terms can be in conflict with each other because different licenses can have contradictory requirements. Although license terms for permitted actions cannot be in conflict with other license terms, forbidden or required license terms can be in conflict with each other.

Black Duck has identified those KnowledgeBase license terms that are in conflict with other KnowledgeBase terms that have the same name but opposing responsibilities (a required license term that is incompatible with a forbidden license term).

You can now [manage conflicts](#) between open source component licenses that have terms which conflict with the project license. You can also [define incompatible terms](#) for your custom license terms.

Defining conflicts for customer license terms

Black Duck has identified those KnowledgeBase license terms that are in conflict with other KnowledgeBase terms that have the same name but opposing responsibilities.

You can define the custom license terms for forbidden or required actions that are in conflict with Black Duck KnowledgeBase terms or with your custom license terms.

Defining an incompatible term

You can define incompatible terms for your custom license terms with a forbidden or required responsibility, including deprecated custom license terms.

- A required license term can only be defined as incompatible to a forbidden license term.
- A forbidden license term can only be defined as incompatible to a required license term.

You cannot define incompatible terms for:

- Black Duck KnowledgeBase license terms
- Custom license terms with a permitted responsibility

To define an incompatible term:

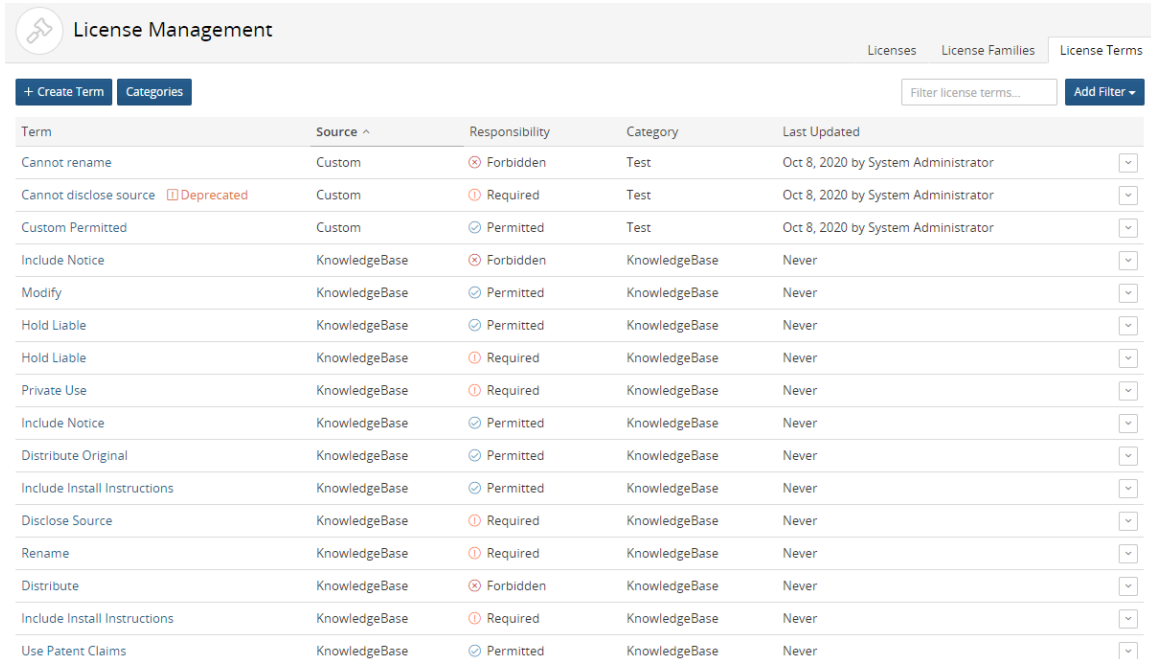
1. Log in to Black Duck with the License Manager [role](#).
- 2.

Click  > **Licenses**.

The License Management page appears.


Select the **License Terms** tab to display all license terms.

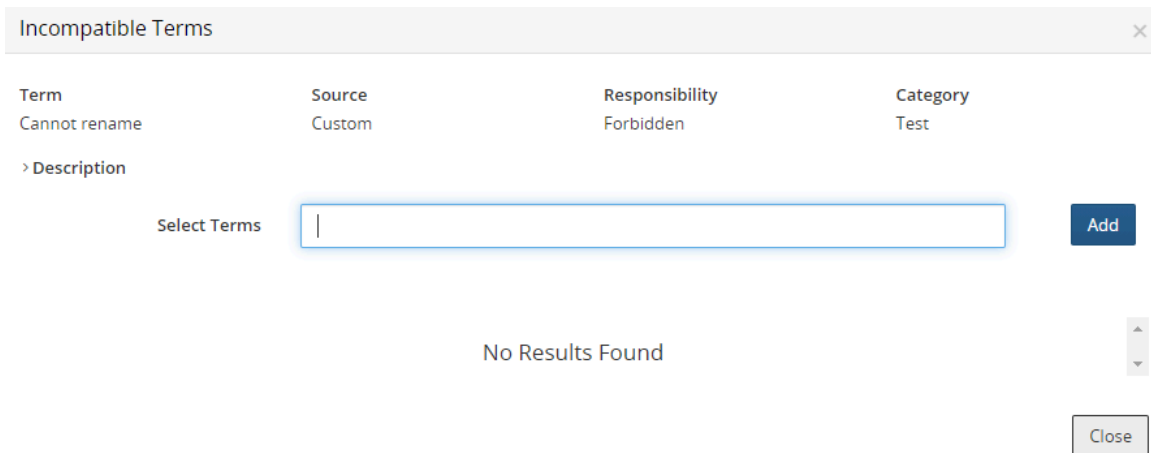
12. Managing open source licenses • About license conflicts



The screenshot shows the 'License Management' interface. At the top, there are tabs for 'Licenses', 'License Families', and 'License Terms'. Below the tabs, there are buttons for '+ Create Term' and 'Categories', along with a search box 'Filter license terms...' and an 'Add Filter' button. The main content is a table with the following columns: Term, Source, Responsibility, Category, and Last Updated. The table lists various license terms such as 'Cannot rename', 'Cannot disclose source', 'Custom Permitted', 'Include Notice', 'Modify', 'Hold Liable', 'Private Use', 'Distribute Original', 'Include Install Instructions', 'Disclose Source', 'Rename', 'Distribute', and 'Use Patent Claims'. Each row includes a dropdown arrow in the 'Last Updated' column.

Term	Source	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3. Click  in the row of the custom license term and select **Incompatible Terms** to open the Incompatible Terms dialog box.



The screenshot shows the 'Incompatible Terms' dialog box. It has a title bar with a close button. Below the title bar, there is a table with the following columns: Term, Source, Responsibility, and Category. The table contains one row: 'Cannot rename', 'Custom', 'Forbidden', 'Test'. Below the table, there is a section labeled '> Description'. Underneath, there is a 'Select Terms' label, a text input field, and an 'Add' button. Below the input field, there is a message 'No Results Found'. At the bottom right, there is a 'Close' button.

Term	Source	Responsibility	Category
Cannot rename	Custom	Forbidden	Test

> Description

Select Terms

Add

No Results Found

Close

4. Type the incompatible license term name in the **Select Terms** field.
Black Duck displays a list of terms that have the opposite responsibility as possible incompatible license terms; for example if you are defining conflicts for a forbidden license term, only required terms appear in the list.
Select a term and click **Add**.
5. Optionally, repeat step 4 to add additional incompatible license terms.
6. Click **Close**.

Viewing incompatible terms

1. Log in to Black Duck with the License Manager [role](#).

2.

Click  > **Licenses**.

The License Management page appears.

Select the **License Terms** tab to display all license terms.


License Management

Licenses License Families License Terms

+ Create Term Categories Filter license terms... Add Filter

Term	Source ^	Responsibility	Category	Last Updated
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator
Cannot disclose source Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Permitted	KnowledgeBase	Never
Hold Liable	KnowledgeBase	Required	KnowledgeBase	Never
Private Use	KnowledgeBase	Required	KnowledgeBase	Never
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never
Rename	KnowledgeBase	Required	KnowledgeBase	Never
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never

3.

Click  in the row of the license term and select **Incompatible Terms** to open the Incompatible Terms dialog box which lists the incompatible terms for this license term.

Incompatible Terms

Term	Source	Responsibility	Category
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase


> Description

Term	Source	Responsibility	Description
Include Notice	KnowledgeBase	Required	You are required to include a notice in product documentation with certain requirements as specified by the license

Displaying 1-1 of 1

Close

Note that if a Black Duck KnowledgeBase license term does not have any incompatible license terms, the **Incompatible Terms** option is not available.

 **Tip:** Use the **Has Incompatible Term(s)** filter to easily view all those license terms for which incompatible terms have been identified.

Deleting incompatible license terms

You cannot delete incompatible terms defined for Black Duck KnowledgeBase license terms. You can only delete incompatible terms that you have defined for your custom license terms.

1. Log in to Black Duck with the License Manager [role](#).

- 2.



Click **Manage** > **Licenses**.


The License Management page appears.

Select the **License Terms** tab to display all license terms.

License Management					
		Licenses		License Families	License Terms
+ Create Term		Categories		Filter license terms...	Add Filter
Term	Source ^	Responsibility	Category	Last Updated	
Cannot rename	Custom	Forbidden	Test	Oct 8, 2020 by System Administrator	
Cannot disclose source ⚠️ Deprecated	Custom	Required	Test	Oct 8, 2020 by System Administrator	
Custom Permitted	Custom	Permitted	Test	Oct 8, 2020 by System Administrator	
Include Notice	KnowledgeBase	Forbidden	KnowledgeBase	Never	
Modify	KnowledgeBase	Permitted	KnowledgeBase	Never	
Hold Liabile	KnowledgeBase	Permitted	KnowledgeBase	Never	
Hold Liabile	KnowledgeBase	Required	KnowledgeBase	Never	
Private Use	KnowledgeBase	Required	KnowledgeBase	Never	
Include Notice	KnowledgeBase	Permitted	KnowledgeBase	Never	
Distribute Original	KnowledgeBase	Permitted	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	Permitted	KnowledgeBase	Never	
Disclose Source	KnowledgeBase	Required	KnowledgeBase	Never	
Rename	KnowledgeBase	Required	KnowledgeBase	Never	
Distribute	KnowledgeBase	Forbidden	KnowledgeBase	Never	
Include Install Instructions	KnowledgeBase	Required	KnowledgeBase	Never	
Use Patent Claims	KnowledgeBase	Permitted	KnowledgeBase	Never	

- 3.



Click  in the row of the custom license term and select **Incompatible Terms** to open the Incompatible Terms dialog box.

Incompatible Terms
✕

Term	Source	Responsibility	Category
Cannot rename	Custom	Forbidden	Test

> Description


Select Terms

Add

Term	Source	Responsibility	Description
Rename	KnowledgeBase	ⓘ Required	If modified, you are required to rename the software to indicate it is not the original work 🗑

Displaying 1-1 of 1

Close

4. Click  in the row of the custom term that you want to remove.
5. Click **Remove** to confirm.

Enabling management of license term conflicts


BOM Managers, and other users with the appropriate [role](#), can view conflicts for a license term using the **License Conflicts** tab in the *Project Version's Legal* tab.

By default, these tabs are disabled. To enable this feature:

1. Use the System Setting page to enable the feature for all *future* projects.
2. Once future projects are enabled, use a project's **Setting** tab to enable the feature for a *current* project.

To enable license conflicts for future projects:

Use the System Settings page to enable the **Legal** and **License Conflicts** tabs for all future projects.

1. Log into Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **System Settings**.
4. Click **Legal**.
5. Set the toggle located in the **License Conflicts** section to *Setting is enabled* to display the **Legal** and **License Conflicts** tabs. Enabling the setting will not take effect immediately for existing projects.

License Conflicts

Enables License Conflicts on the Legal Tab for all future projects so that project users can determine the license and term that is causing the incompatibility. Note: license conflicts can also be enabled or disabled for individual projects.

 Setting is enabled

Set the toggle located in the **License Conflicts** section to *Setting is disabled* to remove the **Legal** and **License Conflicts** tabs. Note that if you select to enable license term fulfillment, the **Legal** tab will appear, but the **License Conflicts** tab will not appear.

License Conflicts

Enables License Conflicts on the Legal Tab for all future projects so that project users can determine the license and term that is causing the incompatibility. Note: license conflicts can also be enabled or disabled for individual projects.

Setting is disabled

Enabling or disabling license conflicts for a specific project

If the system setting is not enabled prior to the creation of the Black Duck project, then the setting must be set at the project level.

To enable or disable license term conflicts:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the **Settings** tab.

Black Duck Project Groups
Sample Project

Project: ★ Watching Project Versions: 1 Owner: System Administra... Overview Settings

Project Details

SCM Repository

Users

Groups

Custom Fields

SBOM Fields

Activity

Settings

Project Name *
Sample Project

Description

Owner
System Administrator (no-reply@synopsys.com) X -

Tier
Select...

Component Adjustments
Always maintain component adjustments to all versions of this project. Archived project versions and manually added components are excluded.

Snippet Adjustments
Apply the following snippet adjustments upon snippet rescans. Apply IDs from partial snippet matches to new exact file matches.

Cloning
Select the attributes you'd like to clone for any new versions of this project.

Custom Fields

Component Edits

Deep License Data

Remediation Details

License Fulfillment Status

Version Settings

Custom Scan Signature
Custom Scan Signature can identify third-party and proprietary software used in your code. There may be performance issues seen when using this feature.

Custom Scan Signature Depth
Depth, as measured in the number of levels in the directory structure, from root, to perform custom signature scanning for this project. The initial value is the default value defined by the System Administrator.

5

Retain Unmatched File Data
If enabled, unmatched file data for scans will always be retained. When disabled (default), unmatched file data will be purged.

System Default (Don't Retain Data)

Purge ONLY Archived Project Version Unmatched File Data

Purge ALL Unmatched File Data

Apply Deep License Data to Bill of Materials
Enabling this checkbox will apply deep license data to your components and allow visibility to embedded licenses which may exist in your components beyond declared licenses. Please note, this can affect the license risk and policy violation for components. It can also impact the Bill of Materials calculation time depending upon the number of components and amount of deep licenses.

Apply Deep License Data to Snippet Component Matches
If enabled, component snippet matches are included in the deep license data calculation.

License Conflicts
Enabling this checkbox will apply license conflicts data to your components

Reset Save

Application ID

Application ID
A field that can be used to store an external mapping id for the project to an external system, like an asset management system or application catalog.

Reset Save

Clone Project
Clone selected versions of this project as well as existing project settings, users, groups, custom fields, component edits and application ID.

Clone Project

Delete Project
Once you delete a project, you cannot restore it and you lose all information and versions related to the project. Scans will be unmapped from all versions and not deleted.

Delete Project

3. Do one of the following:

- Enable the **Apply License Conflicts Data to Bill of Materials** option in the **License Conflicts** section.
- Clear the option to disable this feature for this project.

4. Click **Save**.

Managing project license conflicts

As a BOM reviewer, you need to understand when a component or subproject in your BOM has a license with terms that are incompatible with the declared license of a project. Black Duck identifies the specific license and term that is causing the incompatibility, thereby letting you manage this conflict and reducing the risk of license infringement.

Black Duck identifies the Black Duck KnowledgeBase conflicts (license terms that have the same name but [opposing responsibilities](#)) and the custom license terms that [you defined as in conflict](#) with Black Duck KnowledgeBase terms for a project version.

Note the following:

- License conflict information is not automatically enabled. System Administrators must [enable the Legal and License Conflicts tab](#) for all *future* project versions. Use the project's **Settings** tabs to enable the feature for *current* projects.
- Note that Black Duck only determines license conflicts for subprojects and component versions with high license risk. For the Black Duck license risk model, "high risk" means that licenses in this family tend to have license conflicts under this business scenario (combination of distribution type and component usage) making them incompatible. Medium or low risks means it may have risks if the business scenario changes (or is defined incorrectly) or due to other, non-license conflicts factors.
- Black Duck calculates license risk during a scan or if you select to enable the feature for a current project.

Manual edits to a BOM, including changing the usage for a component or the license of the project version using the **License Conflict** or **Components** tab will trigger a recalculation of the license conflict.

- License conflicts for snippets are not shown until the [snippet is confirmed](#).
- You can [create a policy rule](#) that is triggered when a component's license conflicts with the license for a project version.

Viewing project license conflicts

From a project version BOM, select the **Legal** tab, and if necessary, select the **License Conflicts** tab to view a list of components that have a license that conflicts with the project license.

Black Duck Project Groups
Parent Project 1 > 1.0

Project License Conflicts

Project license conflicts occur when a component has a license with terms that are incompatible with the project version license. Select "Conflicts" to see details about term incompatibilities. [Learn More](#)

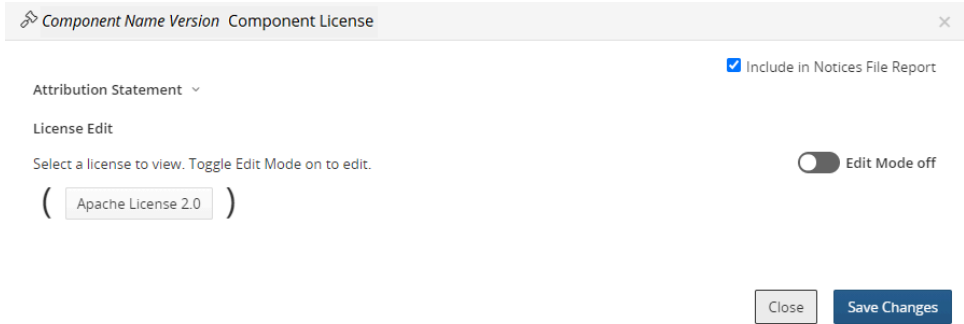
🔗 GNU General Public License v3.0 or later conflicts with:

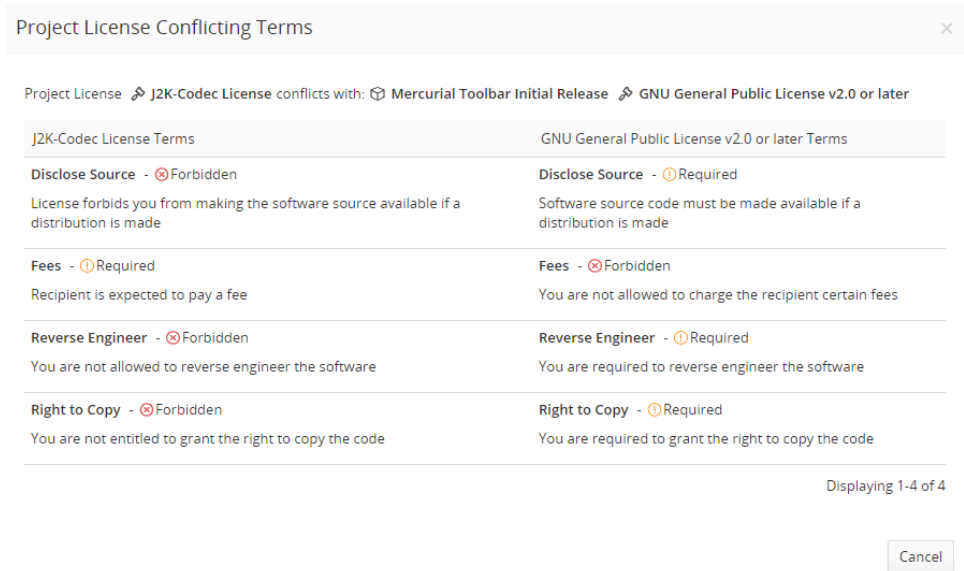
Component	Usage	License	
Sub project 1 1.0	Dynamically Linked	MySQL Commercial License	🔗 Conflict ⋮
jquery 1.11.3-1 (🔗 Sub project 1 1.0)	Dynamically Linked	MySQL Commercial License	🔗 Conflict ⋮

Displaying 1-2 of 2


The table displays the following information:

Column	Description
🔗	Policy violation . Hover over the icon to view the policy rule(s) this component violates. Select the icon to open the Policy Violations dialog box.

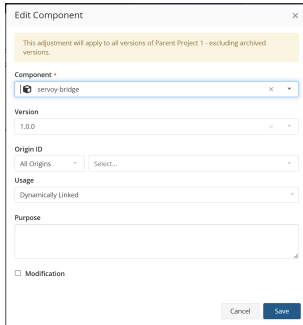
Column	Description
Component	Component or subproject name. If a license conflict is detected in a subproject of this project, the component name will be displayed along with the subproject in parentheses. <small>jquery 1.11.3-1 (Sub project 1.1.0)</small>
Usage	Indicates the usage of this component.
License	<p>The license for this component. Select the license name to open the <i>Component Name Version</i> Component License dialog box.</p>  <p>Use this dialog box to edit the existing license(s), view obligations, and view/edit the license text.</p>

Conflict	<p>Indicates there is a conflict. Select Conflict to open the Project License Conflicting Terms dialog box.</p>  <p>Use this dialog box to view the list of project license terms and conflicting component version license terms.</p>
-----------------	---

Editing a component or subproject with license conflicts


You can edit the component or subproject by clicking  in the row of the component and select **Edit** to open the Edit Component modal.

12. Managing open source licenses • About license conflicts




If the component or subproject is found in the parent project itself, a warning at the top of the Edit Component modal notifying you that any changes made to the Component details will be reflected in all versions of the project it is located in. Once you have made the desired updates, click the **Save** button to accept your changes.

If you are editing a component found in a subproject, a warning will appear at the top of the modal notifying you that you are editing a component belonging to another project along with the project's name and version. Clicking the project name will take you to that project version's Details page.

You are editing a component that belongs to another project:
 [Sub project 1 1.0](#)

Adding comments to a component or subproject with license conflicts


Optionally, to add a comment, click  in the row of the component and select **Comment** to open the *Component/Subproject Name Version Comment* dialog box.

Enter the comment and click **Add Comment**. The comment appears for this component in the BOM.

13. Running a report


Black Duck provides the following reports:

- [Notices File](#)
- [Project version](#). Detail and vulnerability reports.
- Global or project(s) reports:
 - [Vulnerability Remediation](#)
 - [Vulnerability Status](#)
 - [Vulnerability Update](#)

 **Tip:** Reporting schemas in the PostgreSQL database provide access to Black Duck data for reporting purposes. See the Report Database guide which contains information on using the report database.

These reports help you:

- View the list of components and associated license text for a project version.
- Identify the security vulnerabilities associated with all your projects.
- Track the remediation status of vulnerabilities in all your projects.
- Export and share the information of a single project version.

 **Note:** Reports include subproject information *if* you have permission to the [subproject](#).

Notices File report

The Notices File report provides a list of open source components, versions, the associated license text, and optionally, copyright statements. You can use this report to create an attribution report for your project release or to share BOM and license information.

This report is available as a text file or in HTML format. Each format provides the following information:

- Header information. Lists the project name, version, phase, and distribution.
- Components. Lists all components, component versions, subprojects, subproject versions, and associated licenses, including [deep license data](#).

You can [exclude a component or subproject](#) or add an [attribution statement](#),

- Licenses. Provides the license text for all licenses listed in the **Components** section.

You can [edit the license text](#) shown here.

- Other options:
 - [Deep License Data](#): Adds deep licenses discovered via component origin to the list of components. Only available if deep licenses are enabled for the project.
 - File Copyright Text: Provides a Copyright Text section that contains copyright statements discovered in file matches. Only available if file matches are present.
 - File License Data: Licenses discovered in file matches. Only available if file matches are present.

13. Running a report • Notices File report

- **License Data.** Choose to include the licenses of components in the project.
- **License Text.** Choose to include the text of licenses in the project.
- **Origin Copyright Text.** Provides a report section that contains the copyright statements obtained from the Black Duck KnowledgeBase, edited KnowledgeBase copyright statements, and/or custom copyright statements for the open source components you use.

User with the Copyright Editor role can [create](#) or [edit](#) copyright statements for an open source component version origin.

- **Unmatched File Discoveries:** Provides an Unmatched File Data section that includes file discoveries unassociated with components in the project. Only available if unmatched files are present in the project.

The following is an example of a portion of the HTML version of the report:

Sample Project A - 4.0 Notices File

Phase: In Planning | Distribution: External

Notices Report Content

- License Data
- License Text
- Origin Copyright Text

Components

Component	License	Component Link
Apache Log4j API 2.17.1	Apache License 2.0	http://logging.apache.org/log4j/2.x/log4j-api/
Apache Tomcat 10.0.20	Apache License 2.0	http://tomcat.apache.org/

Copyright Data

Apache Log4j API 2.17.1 - maven:org.apache.logging.log4j:log4j-api:2.17.1
<http://logging.apache.org/log4j/2.x/log4j-api/>

- Copyright 1969-1999 The Apache Software Foundation

Apache Tomcat 10.0.20 - maven:org.apache.tomcat:tomcat-jasper-el:10.0.20
<http://tomcat.apache.org/>

- Copyright 1999-2022 The Apache Software Foundation This product includes software developed at

Licenses

Apache License 2.0

Apache Log4j API 2.17.1, Apache Tomcat 10.0.20

```
Apache License
Version 2.0, January 2004
*****

http://www.apache.org/licenses/


TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright
owner that is granting the License.
```

Note that licenses from the [Unknown license family](#) are not included in the Notices File report, however the component with the unknown license is included in the report unless you select to remove it.

 **Note:** If you notice omissions or errors in the license text, contact Black Duck Support and provide the correct information so that the Black Duck KnowledgeBase can be updated.

To run a Notices File report:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version of the project for which you want to run the report.
3. Select the **Reports** tab.

4. Click **+ Create New Report**.
5. Select **Notices File** and then select the format for the report:
 - Text
 - HTML
6. Check or uncheck the **Include Subprojects** checkbox.
7. Optionally, select one or more of the following options:
 - Deep License Data
 - File License Data
 - File Copyright Text
 - License Data
 - License Text
 - Origin Copyright Text
 - Unmatched File Discoveries

The Notices File report may take more time to run if any of these options are selected.

8. Click **Create** to run the report.
9. A link that includes the project, version name, and date appears when the report completes. Any user who is a member of the project can access the link.
 - If you selected the text format, download the report and extract the zip file locally.
 - If you selected the HTML format, select the link to open the report in a new tab.

Excluding a component or subproject from the Notices File report

By default, all components and subprojects are included in the [Notices File report](#).

To exclude a component in the Notices File report:

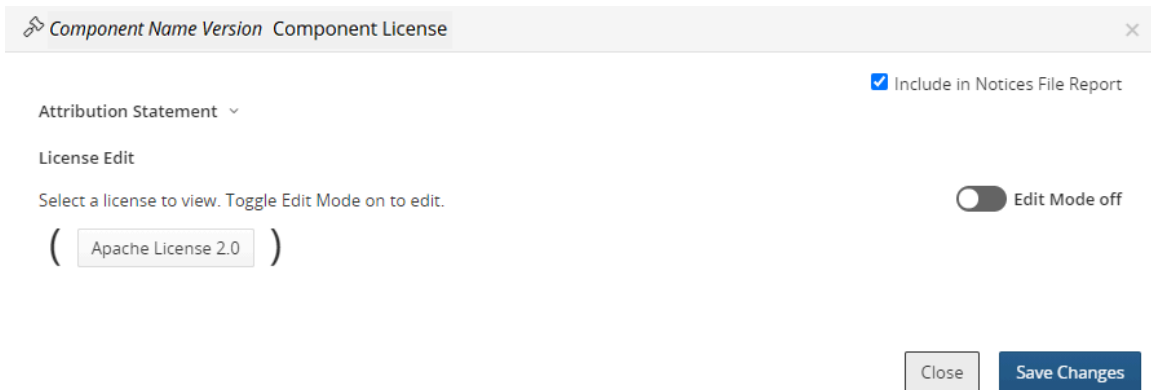
1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the version name to display the **Components** tab and view the BOM.

The screenshot shows the Black Duck Project Groups interface for 'Sample Project' version 1.0.0. The interface includes a navigation bar with tabs for Components, Security, Source, Reports, Details, Legal, and Settings. Below the navigation bar, there are three risk charts: Security Risk, License Risk, and Operational Risk. Each chart shows the number of components at different risk levels (Critical, High, Medium, Low, None). The Security Risk chart shows 4 Critical, 3 High, 4 Medium, 0 Low, and 17 None. The License Risk chart shows 6 High, 2 Medium, 0 Low, and 20 None. The Operational Risk chart shows 24 High, 3 Medium, 0 Low, and 1 None. Below the charts, there is a table of components with columns for Component, Source, Match Type, Match Score, Usage, License, Security Risk, and Operational Risk. The table lists four components: AOP Alliance (Java/J2EE AOP standard) 1.0, Apache Commons DBCP 1.2.2, Apache Commons FileUpload 1.3.3, and Apache Commons IO 2.2. Each component has a match score of 100% and is associated with a High Operational Risk.

Component	Source	Match Type	Match Score	Usage	License	Security Risk	Operational Risk
AOP Alliance (Java/J2EE AOP standard) 1.0	1 Match	Transitive Dependency	100%	Dynamically Linked	Public Domain		High
Apache Commons DBCP 1.2.2	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0		High
Apache Commons FileUpload 1.3.3	1 Match	Direct Dependency	100%	Dynamically Linked	Apache-2.0	1	High
Apache Commons IO 2.2	1 Match	Transitive Dependency	100%	Dynamically Linked	Apache-2.0	1	High

13. Running a report • Notices File report

3. Select the existing license from the **License** column to open the *Component/Subproject Name Version* Component License dialog box.



4. In the License window, clear **Include in Notices File Report** to exclude the component or subproject from the report.

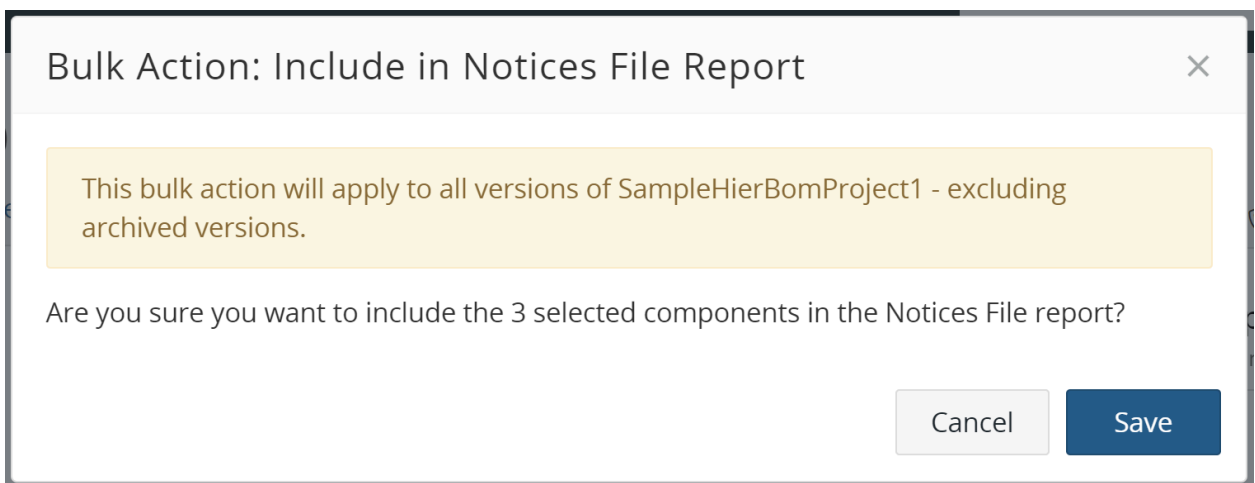
Select **Include in Notices File Report** to include the component or subproject in the report.

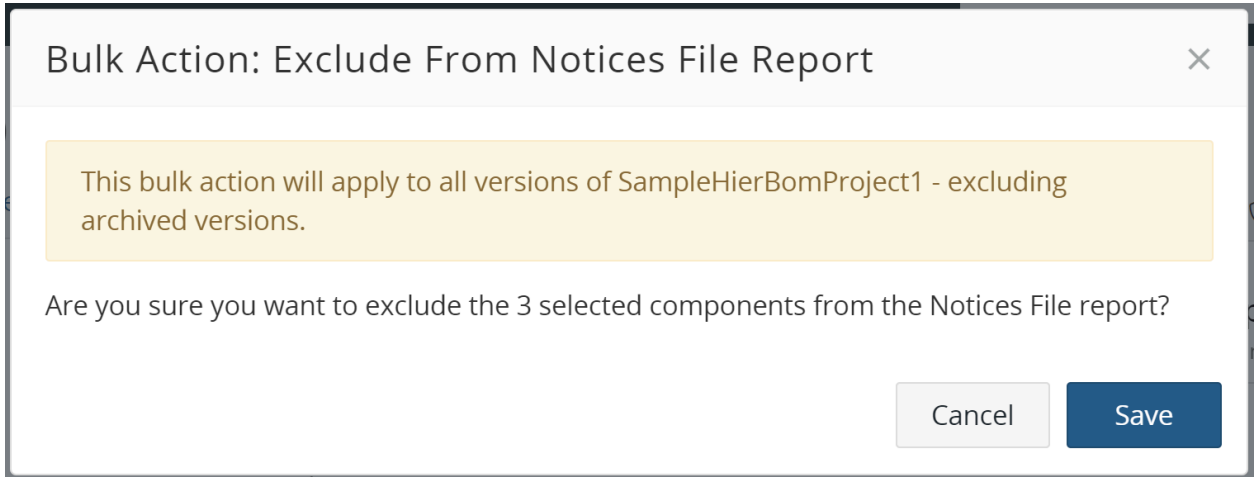
5. Click **Save Changes**.

To include/exclude multiple components in the Notices File report:

1. [Display the project version BOM](#). Ensure you are in the Components view.
2. Check the box next to any number of components.
3. Click the **Bulk Actions** button.
4. Select **Include in Notices File Report** or **Exclude from Notices File Report**.

The **Bulk Action: Include in Notices File Report** or **Bulk Action: Exclude From Notices File Report** dialog box appears.





5. Click **Save**.

Project version reports

Use project version reports to export and share the content of a single project version. You can run a Version Details report, Vulnerability report, Notices File report, or a Software Bill of Materials (SBOM) report.

Vulnerability Remediation report


Based on a specific date range, the Vulnerability Remediation report lists all the vulnerabilities that match a specific [remediation status](#).

For example, you can use this report to identify all of the vulnerabilities that require remediation or all the vulnerabilities that have been mitigated and ignored.

This report can be run at the global level (for all projects to which you have access) or for one or more projects to which you have access. It can also be run at the [project version level](#) to view this information for a specific project version.

Running a Vulnerability Remediation report at the global level

To run a Vulnerability Remediation report at the global level:

1. Log in to Black Duck.
2.  Click **Reports**.
3. Click **+ Create new report**. The Create New Report dialog box appears.
4. Select **Vulnerability Remediation Report** from the **Report Type** list.
5. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.
6. Select either **HTML** or **CSV** as the report format.

 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.


13. Running a report • Vulnerability Remediation report

7. Select the dates for this report. The date represents the day when the vulnerability was published. By default, the end date is the current date.
8. Optionally, select one or more remediation statuses.
9. Click **Confirm** to run the report.
One of the following links appears when the report completes:
 - vulnerability-remedation-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
 - vulnerability-remedation-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

10. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file.

 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Running a Vulnerability Remediation report at the project level

To run a Vulnerability Remediation report at the project level:

1. Log in to Black Duck.
2. Click the desired project name on the Dashboard.
3. Click the desired project version.
4. Click the **Reports** tab.
5. Click **+ Create new report**. The Create New Report dialog box appears.
6. Select **Vulnerability Remediation Report** from the **Report Type** list.
7. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.
8. Select either **HTML** or **CSV** as the report format.


 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.

9. Select the dates for this report. The date represents the day when the vulnerability was published. By default, the end date is the current date.
10. Optionally, select one or more remediation statuses.
11. Click **Confirm** to run the report.
One of the following links appears when the report completes:
 - vulnerability-remedation-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
 - vulnerability-remedation-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

12. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file.

 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Vulnerability Status report


The Vulnerability Status report includes all the vulnerabilities that are associated with the projects and project versions to which you have access.

For example, you can use this report to identify which projects are secure and which projects and project versions contain security risks.

This report can be run at the global level (for all projects to which you have access) or for one or more projects to which you have access. It can also be run at the [project version level](#) to view this information for a specific project version.

Running a Vulnerability Status report at the global level

To run a Vulnerability Status report at the global level:

1. Log in to Black Duck.
2.  Click **Reports**.
3. Click **+ Create new report**. The Create New Report dialog box appears.
4. Select **Vulnerability Status Report** from the **Report Type** list.
5. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.
6. Select either **HTML** or **CSV** as the report format.

 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.

7. Click **Confirm** to run the report.


One of the following links appear when the report completes:

- vulnerability-status-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
- vulnerability-status-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

8. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file.


 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Running a Vulnerability Status report at the project level

To run a Vulnerability Status report at the project level:

13. Running a report • Vulnerability Update report

1. Log in to Black Duck.
2. Click the desired project name on the Dashboard.
3. Click the desired project version.
4. Click the **Reports** tab.
5. Click the **+ Create New Report**.
6. Select **Vulnerability** from the report list.
7. Select **Vulnerability Status Report** from the **Report Type** list.
8. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.
9. Select either **HTML** or **CSV** as the report format.

 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.

10. Click **Confirm** to run the report.


One of the following links appear when the report completes:

- vulnerability-status-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
- vulnerability-status-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

11. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file.

 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Vulnerability Update report

Based on a specific date range, the Vulnerability Update report includes the following information for projects to which you have access:

- New vulnerabilities.
For example, you can use this report to identify new vulnerabilities after code or a Docker image has been rescanned.
- Updates to the [remediation status](#) of existing vulnerabilities.
For example, you can use this report to track the progress of a remediation effort.
- Updates to any of the data that is associated with vulnerabilities.
For example, you can use this report to identify if the risk scores associated with existing vulnerabilities have changed.

This report can be run at the global level (for all projects to which you have access) or for one or more projects to which you have access. It can also be run at the [project version level](#) to view this information for a specific project version.

Running a Vulnerability Update report at the global level

To run a Vulnerability Update report at the global level:

1. Log in to Black Duck.



3. Click **+ Create new report**. The Create New Report dialog box appears.

4. Select **Vulnerability Update Report** from the **Report Type** list.

5. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.

6. Select either **HTML** or **CSV** as the report format.

 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.

7. Select the dates for this report. The date represents the day on which the vulnerability was added to a project version or the information associated with the vulnerability was updated. By default, the end date is the current date.

8. Click **Confirm** to run the report.

One of the following links appear when the report completes:

- vulnerability-update-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
- vulnerability-update-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

9. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file. The CSV report will contain four reports:

- new-remediated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv

This report will contain all new remediations for the selected project(s) within the specified time frame. A vulnerability is newly remediated if the remediation was created after the vulnerability was created and it is in the specified time range.

- new-vulnerabilities_YYYY-MM-DD_HHMMSS.csv


This report will contain all new vulnerabilities for the selected project(s) within the specified time frame. A vulnerability was associated to a BOM component whether or not it is remediated and the bom association was in the specified time range.

- updated-remediated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv

This report will contain all updated remediations for vulnerabilities that occurred within the specified time range for the selected project(s). A vulnerability has an updated remediation if the remediation update happened after the remediation was created and is in the specified time range.

- updated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv


This report will contain all updated vulnerabilities for the selected project(s) within the specified time frame. A vulnerability is updated if the the updated dateTime is within the specified time range and this vulnerability risk was updated after the risk was created.

 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Running a Vulnerability Update report at the project level

To run a Vulnerability Update report at the project level:

1. Log in to Black Duck.
2. Click the desired project name on the Dashboard.
3. Click the desired project version.
4. Click the **Reports** tab.
5. Click **+ Create new report**. The Create New Report dialog box appears.
6. Select **Vulnerability Update Report** from the **Report Type** list.
7. To run the report for selected projects, enter one or more project names in the **Projects** field. Leave the field blank to create the report for all projects to which you have access.
8. Select either **HTML** or **CSV** as the report format.

 **Tip:** Use the CSV option when your data becomes too large to render and view in the browser.

9. Select the dates for this report. The date represents the day on which the vulnerability was added to a project version or the information associated with the vulnerability was updated. By default, the end date is the current date.

10. Click **Confirm** to run the report.

One of the following links appear when the report completes:

- vulnerability-update-report_all_assigned_projects_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for a global version of the report
- vulnerability-update-report_YYYY-MM-DD_HHMMSS (time stamp in system timezone) for one or more projects

Reports for a specific project can be accessed by any user who is a member of the project. However, if the report contains multiple projects, the user must be a member of all projects to access the report.

11. Select the link to view the report.

If you selected CSV as the report format, download the report and extract the zip file. The CSV report will contain four reports:

- new-remediated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv

This report will contain all new remediations for the selected project(s) within the specified time frame. A vulnerability is newly remediated if the remediation was created after the vulnerability was created and it is in the specified time range.

- new-vulnerabilities_YYYY-MM-DD_HHMMSS.csv


This report will contain all new vulnerabilities for the selected project(s) within the specified time frame. A vulnerability was associated to a BOM component whether or not it is remediated and the bom association was in the specified time range.

- updated-remediated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv

This report will contain all updated remediations for vulnerabilities that occurred within the specified time range for the selected project(s). A vulnerability has an updated remediation if the remediation update happened after the remediation was created and is in the specified time range.


- updated-vulnerabilities_YYYY-MM-DD_HHMMSS.csv

This report will contain all updated vulnerabilities for the selected project(s) within the specified time frame. A vulnerability is updated if the the updated dateTime is within the specified time range and this vulnerability risk was updated after the risk was created.

 **Note:** You can use the native print functionality of your web browser to print the HTML version of the report.

Deleting reports

To delete a report:

1. Click  in the row of the report you wish to delete.
2. Click **Delete** in the confirmation dialog box.

Note the following:


- Reports older than 30 days are automatically deleted.
- The system retains up to 20 reports, per user, across all project versions. If a user creates more than 20 reports, the system automatically deletes the oldest reports and retains the 20 newest reports.

14. Managing Black Duck user accounts

There are two ways to manage user accounts in Black Duck:

1. Managing user accounts manually. A user with the [User Administrator](#) role can:
 - [Add a new user account](#)
 - [Inactivate a user account](#)
 - [Change user account information](#)
 - [Change a user's password](#)
 - [View a user's groups](#)
 - [Manage user roles](#)
2. [Enabling and configuring LDAP to manage user accounts.](#)

After you configure LDAP to manage user accounts for Black Duck, new user accounts will be automatically created the first-time users attempt to log in. Your LDAP server will then manage passwords and account details for those user accounts in Black Duck.

 **Tip:** If you are using LDAP to manage most of your user accounts in Black Duck, you can still manually manage those user accounts that do not also exist in your LDAP directory, such as a default system administrator account.

Note that you can also create external user accounts.

Users with the System Administrator role can also configure the [password requirements](#) for user accounts.

Configuring password requirements

Users with the System Administrator role can set password requirements for *local* Black Duck accounts. If enabled, Black Duck ensures that the new password meets your requirements and also rejects passwords that are considered weak, such as "password", "blackduck", or a user's username or email address.

 **Note:** These requirements do not apply to external (LDAP or SAML) accounts.

System Administrators can:

- define the minimum password length (from 8 to 25 characters). The maximum length is 128 characters.
- define the minimum number of character types for the password (from one to four character types). Possible character types are lowercase letters, uppercase letters, numbers, or special characters.
- select whether to enforce the password requirements on current users when they log in to Black Duck.

If you select this option, current users who try to log in with a password that does not meet the requirements will be forced to create a new password before they can access the system.

Note that when using the Black Duck APIs, users with a password that does not meet your requirements will receive a 412 response code which will include the reason why the current password does not meet requirements.

If password requirements are enabled, all new passwords must satisfy the requirements. Password requirements are still enforced on current users when they attempt to change their password. Administrators

must also create passwords that meet these requirements when resetting a current user's password or when they make any changes to a user's detail information (such as their first name).

By default, password requirements are enabled and have these settings:

- The minimum password length is eight characters.
- Only one character type is required.
- Password requirements are not enforced on current users when logging in to Black Duck.

To manage password requirements:

1. Log in to Black Duck with the System Administrator role.

2.



Click **Admin**.

3. Select **System Settings**.

4. Click **User Authentication**.

5. Click the **Open Settings** button next to **Password Requirement Settings** in the **Local Authentication** section. A **Password Settings** modal will appear.

6. Select to enable or disable password settings.

- To disable, remove the check from the **Enable Password Settings** checkbox.

- To enable, add a check in the **Enable Password Settings** checkbox.

Password Settings [X]

Minimum Password Length +
Select a value representing the minimum number of characters a password may contain.

8

Minimum Character Requirements
Select the minimum number of character types. Character types are: uppercase, lowercase, number, and symbol.

1

Enforce Configuration
Prompt users to update their password upon login (if current password doesn't meet requirements)

Enable Password Settings

Cancel Save

7. If you enabled password settings:
 - a. Select the following:
 - **Minimum length.** Minimum number of characters in the password.
 - **Character requirements.** Select the minimum number of character types.
For example, if you select the value 2, passwords must include at least two of the following: lowercase letters, uppercase letters, numbers, or special characters.
 - **Enforce configuration.** Select this option to enforce the password requirements on your current users when logging in to Black Duck.
 - b. Click **Save**.

Creating a user account

You can create a Black Duck user account for a local user (an internal user account) for an external user (such as a user managed by an external source, such as LDAP).

If you have enabled LDAP, you can create users on your LDAP server instead of in Black Duck. Black Duck will authenticate user IDs against the LDAP server, and if the username and password are valid, will copy the user ID to Black Duck database.

Note that with external user accounts:

- You can create users and assign roles without the user logging in to Black Duck.
- User information, such as the first or last name, can be changed in Black Duck, however passwords are not managed by Black Duck.
- The first name, last name, and email address of the external user will be overridden with the information present on the external server, (such as an LDAP server), at the time of login.
- An external user is only created when an administrator configures either SAML or LDAP in Black Duck. If both SAML and LDAP are enabled, or *both* are disabled, the external user will not be created.

To create a user account:

1. Log in to Black Duck.

2.



Click **Admin**.

3. Select **Users & Groups** to display the **User & Group Management** page.

Administration
Users & Groups

Users Groups

+ Create User

User Status: Active x + Filter Filter user list...

User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active ...
Project Viewer	Project	Viewer		Project Viewer	✓ Active ...
Scanner	Global	Scanner		Global Code Scanner	✓ Active ...
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active ...
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active ...

Displaying 1-5 of 5

4. Click **+ Create User**. The Create a New User dialog box appears.

Create a New User ✕

Type
 Internal External (LDAP, SAML)

User Name *

First Name *

Last Name *

Email

Passwords must:
 Contain between 8 and 128 characters
 Be difficult to guess

Password *

Confirm Password *

Active user

5. Select whether this user is an internal (managed within Black Duck) or external (managed by LDAP, SAML) account.
6. Do one of the following:
 - For an internal user, enter the following information
 - Username.
 - First Name.
 - Last Name.
 - Email. This field is optional.
 - Password.

If there are [password requirements](#), those requirements are listed in this dialog box. Black Duck notes when each requirement is met. You will not be able to create the user account unless the password meets *all* requirements.

- Confirm password: This must match the password you entered.
- For an external user, enter the following information:
 - Username.
 - First Name.
 - Last Name.
 - Email. This field is optional.

Note that the passwords for external accounts are managed by the external source such as LDAP, not by Black Duck.


7. Select whether this user is active or inactive. Clearing this check box inactivates this user.
8. Click **Create**.
Black Duck creates the user account with the password you specified.

After creating a user, you can:

- [assign roles to this user](#)
- [assign groups to this user](#)
- [add this user to a project team](#)

If you [created default groups](#), this user is automatically added to the default group and is granted all roles and access to all projects configured for that group.

Disabling a user

 **Note:** If you have enabled LDAP, you should manage user records in the LDAP server. If you delete a record in Black Duck and do not delete the user from the LDAP server, the next time the user attempts to log in to Black Duck, their user record will be recreated with data from the LDAP server.

To disable a user account:

1. Log in to Black Duck.

2.  Click **Admin**.

3. Select **Users & Groups** to display the **User & Group Management** page.


User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

4. Find the user you want to inactivate:
 - Filter the users that appear on the page.
 - Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.
5. Select the user to display the *Username's* User Details page.
6. Clear the **Active user** check box in the **Internal** or **External User Details** section and click **Save**.

Converting a user account

You can convert an internal account to an external account or an external account to an internal account.

 **Note:** Converting an internal user account to an external user account requires that an administrator has configured *either* SAML or LDAP in Black Duck. If *both* SAML and LDAP are enabled, or both are disabled, you will be unable to convert the internal user account to an external user account.

To convert an account:

1. Log in to Black Duck.



3. Select **Users & Groups** to display the **Users & Groups** page.

User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

4. Select the username of the account you wish to convert. The *Username's* User Details page appears. Depending on whether the account you selected is an internal or external account, do one of the following:
 - To convert an existing external account to an internal account, click **Convert to Internal Account (Black Duck)**.
 - To convert an existing internal account to an external account, click **Convert to External Account (LDAP, SAML)**.
5. Do one of the following:
 - To convert from an external account to an internal account, enter the following information:
 - Username. Enter a username.
 - First Name. The existing first name is shown. You can retain the existing name or enter a new first name.
 - Last Name. The existing last name is shown. You can retain the existing name or enter a new last name
 - Email. This field is optional.
 - Password
 - Confirm password: This must match the password you entered. Black Duck validates this when you create the user account.
 - To convert an internal account to an external account, enter the following information:
 - Username. Enter a username.
 - First Name. The existing first name is shown. You can retain the existing name or enter a new first name.
 - Last Name. The existing first name is shown. You can retain the existing name or enter a new first name.
 - Email. This field is optional.

Note that the passwords for external accounts are managed by LDAP, not by Black Duck.
6. Select whether this user is active or inactive. Clearing this check box inactivates this user.
7. Click **Save**.

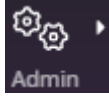
Viewing a user's groups

You can view the groups a user belongs to, and the source, status, and roles associated with that group.

To view a user's groups:

1. Log in to Black Duck.

- 2.



Click **Admin**.

3. Select **Users & Groups** to display the **User & Groups** page.

User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

4. Find the user you want to find:

- Select the **Display Inactive Users** check box to include inactive users. Clearing this check box hides all inactive users.
- Filter the users that appear on the page.
- Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.


5. Select the user to display the *Username* page.

6. Click **Project Groups** in the left-hand menu.

The **Projects Groups** page lists the project groups to which this user belongs. In this section, you can select a group name to view the *Group Name* page from which you can [manage group information](#), [group roles](#), and group membership by [adding](#) or [removing](#) this user from one or more groups.


7. Click **User Groups** in the left-hand menu.

The User Groups page lists the user groups to which this user belongs. In this section, you can also:

- Select a group name to view the *Group Name* page from which you can [manage group information](#), [group roles](#) and [group membership](#).
- [Add this user to one or more groups](#).
- [Remove this user from a group](#) by clicking  in the row of the group. Select **Remove** in the Remove User from Group dialog box to confirm.

Users can view the user groups that they belong to by using the My Profile page.

Exporting to CSV

You can export the list of users or groups to CSV which converts the individual rows to tabular data. To do so, click either the Users or Groups tab on the top right, click the  button, and select CSV.

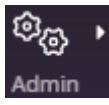
Viewing a user's projects

You can view the projects a user belongs to, and whether the user was added individually or as a member of a group.

To view a user's projects:

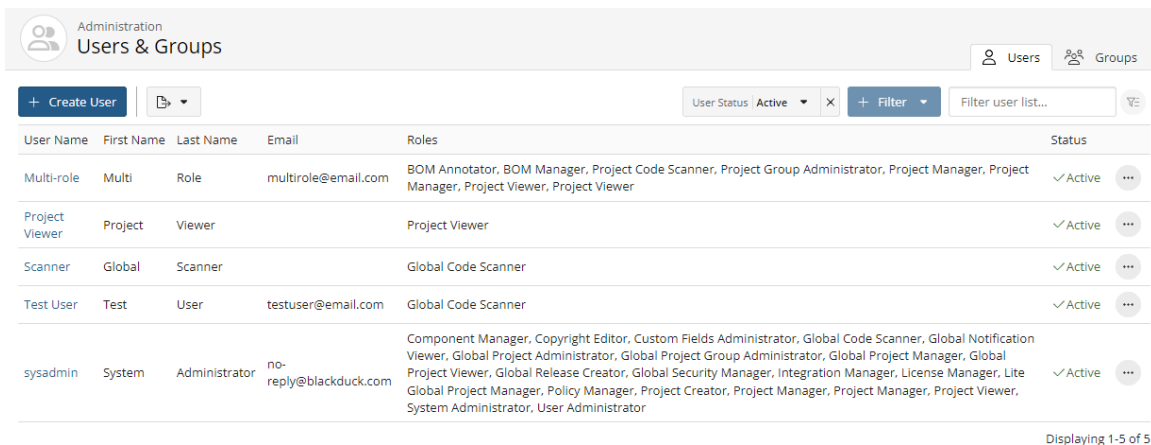
1. Log in to Black Duck.

2.



Click .

3. Select **Users & Groups** to display the **User & Group Management** page.


 A screenshot of the "Users & Groups" management page. The page has a header with "Administration Users & Groups" and tabs for "Users" and "Groups". Below the header is a toolbar with a "+ Create User" button, a dropdown menu, and filters for "User Status" (set to "Active") and "Filter" (set to "Filter user list..."). The main content is a table with columns: "User Name", "First Name", "Last Name", "Email", "Roles", and "Status". The table lists five users: "Multi-role", "Project Viewer", "Scanner", "Test User", and "sysadmin". Each user row has a checkmark and the word "Active" in the status column, followed by a three-dot menu icon. At the bottom right of the table, it says "Displaying 1-5 of 5".

User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

4. Find the user you want to find:

- Filter the users that appear on the page.
- Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.

5. Select the user to display the *Username* page.


6. Click **Projects** in the left-hand menu.

7. The **Projects** page lists the projects to which this user belongs. For each project, it lists whether the user is a direct member (the user was added individually and not as part of a group), and the groups the user is a member of that have access to the project. You can:


- Select a project name to view the *Project Name* page from which you can view project versions and [manage project details, members, and groups](#).

- Add this user to projects: click **Add project**, enter the name of one or more projects, select the roles for this user for this project, and click **Add**.
- Remove members that were directly added to a project: click **Remove** and then confirm removal of this user.

Changing your Black Duck password

 **Note:** If your system administrator has enabled LDAP on the Black Duck server, user account information and passwords are managed by LDAP. You cannot change your password in Black Duck.

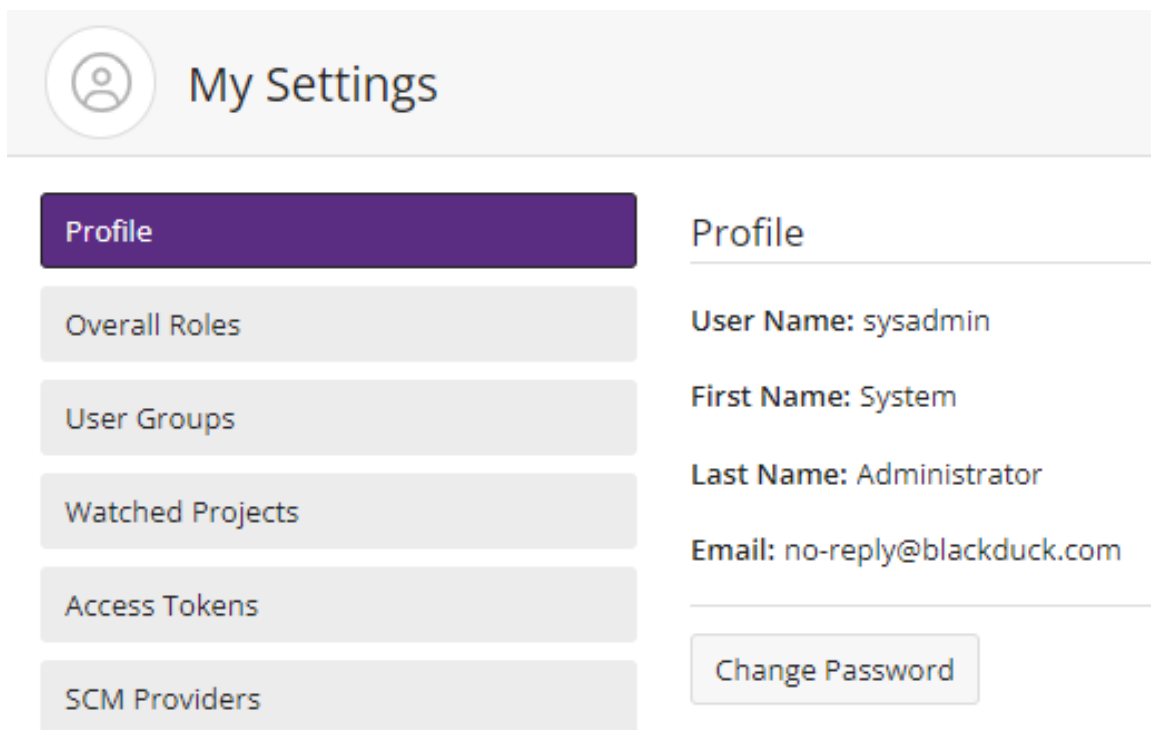
If your Black Duck server does not use LDAP to manage user accounts, your username and initial password were created by your Black Duck administrator. You can change your password on your profile page.

 **Tip:** If you forget your password, a user with the User Administrator role can change it for you.

To change your password:

1. Log in to Black Duck.
2. From the user menu located on the top navigation bar, select **My Profile**.

The My Profile page appears.



My Settings

Profile

Overall Roles

User Groups

Watched Projects

Access Tokens

SCM Providers

Profile

User Name: sysadmin

First Name: System

Last Name: Administrator

Email: no-reply@blackduck.com

Change Password

3. Click **Change Password**.

The Change Password dialog box appears.

Change Password
✕

Passwords must:

Contain between 8 and 128 characters

Be difficult to guess

Current Password *

New Password *

Confirm Password *

4. Type your current password in the **Current Password** field.

5. Type your new password in the **New Password** field.


If there are [password requirements](#), those requirements are listed in the dialog box. Black Duck notes when each requirement is met as you type your new password. You will not be able to save this password if it does not meet *all* requirements.

6. Type the same new password in the **Confirm Password** field.

7. Click **Save**.

Changing user account information

You can modify the information for internal or external user accounts.

 **Note:** If you have enabled LDAP, you can manage user account information on the LDAP server or, in Black Duck (for *external* Black Duck user accounts only). Note that any changes you make to user account information in Black Duck for *external* Black Duck user accounts will be overwritten the next time user information is synchronized with the data on the LDAP server. You can only update the information for an external user if an administrator has configured either SAML or LDAP in Black Duck. If *both* SAML and LDAP are enabled, or both are disabled, you cannot modify the information for an external user.


To change user account information:

1. Log in to Black Duck.


2.  Click **Admin**.

3. Select **Users** to display the **Users & Groups** page.

4. Find the user whose information you want to change:
 - Filter the users that appear on the page.
 - Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.
5. Select the username to open the *Username's* User Details page.
6. Select the **User Details** tab.
7. Enter the updated information in the **User Details** section.

 **Note:** If you are updating information for internal users in the **Internal User Details** section and [password requirements](#) have been defined, you will not be able to save the updated information if this user's password does not meet the password requirements; an error message appears notifying you of which password requirements are not met. Update the user's password to meet the password requirements and then update the information in this section.
8. Click **Update**.

Changing a user's password

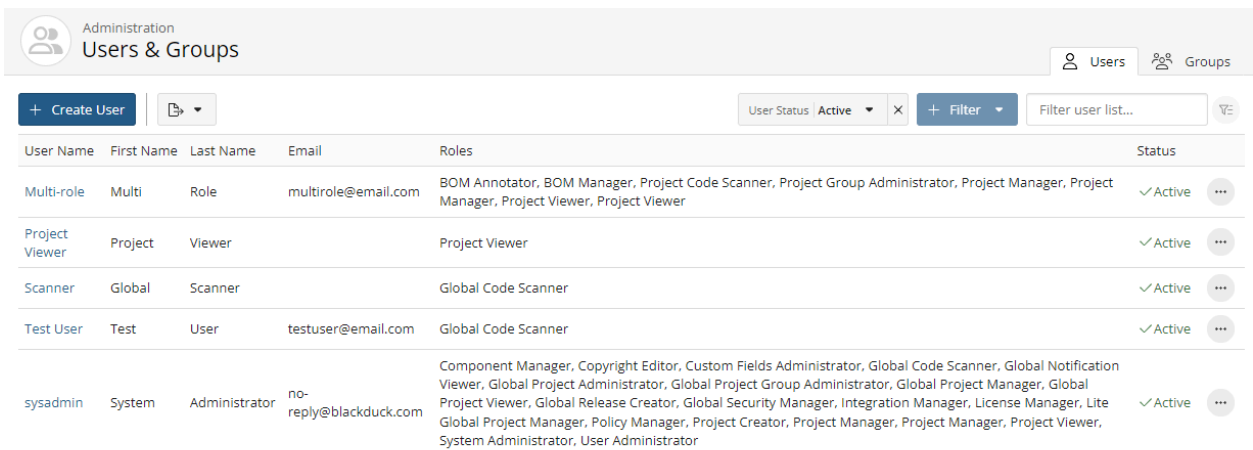
-  **Note:** If you have enabled LDAP authentication, user account passwords are managed by LDAP. You will not be able to change passwords in Black Duck.

To change a user's password:

1. Log in to Black Duck.



3. Select **Users & Groups** to display the User Management page.



User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

4. Find the name of the user whose password you want to reset:
 - Filter the users that appear on the page.

- Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.
5. Select the username to open the *Username* page and click **Reset Password for User**.
 6. In the Reset Password for User dialog box, type the new password in the **Password** field.
If there are [password requirements](#), those requirements are listed in this dialog box. Black Duck notes when each requirement is met. You will not be able to save this password if it does not meet *all* requirements.
 7. Type the same password in the **Confirm Password** field.
 8. Click **Save**.

Understanding roles

Black Duck provides global and project roles which helps you control access and capabilities without impeding productivity. Roles define the tasks users can perform and the information users can view. Project-level roles provide the flexibility so that you can assign users individual roles per project - the roles only apply to the projects a user is assigned.

- You can assign roles to either [individual user accounts](#) or to [groups](#).
- If you assign a role to a group, the entire group membership inherits the role and its permissions.
- If you do not assign a role, users have read-only access to Black Duck and read-only access to the BOM and projects that user is assigned

For more information on the tasks that can be performed for each role, refer to the [Black Duck user role matrix](#).

Global roles

The following global roles are available:

- **Component Manager**
The Component Manager is responsible for creating, editing, and/or deleting custom components and reviewing Black Duck KnowledgeBase components.
This role is often assigned to a centralized group responsible for the management of custom components. In smaller organizations, this role can be given to subject matter experts (SMEs) or development managers.
- **Copyright Editor**
The Copyright Editor is responsible for creating or editing [copyright statements](#) for components.
This role is often assigned to someone within the Legal department.
- **Custom Fields Administrator**
The Custom Fields Administrator is responsible for managing [custom fields](#) in projects.
- **Global Code Scanner**
The Global Code Scanner has access to all scans in Black Duck and can run, map, or delete scans for any existing project within the system.

This role is often assigned to a user account used for continuous integration (CI) builds and sometimes, in smaller organizations, given to a release/build engineer who manages all builds for a company.

- Global Notification Viewer

This role has read only access to all projects and receives all system notifications regardless of user preferences.

- Global Project Group Administrator

The Global Project Group Administrator has access to all project groups and can:

- view/create/modify/delete project groups, including adding users to any project group.

- Global Project Administrator

The Global Project Administrator has access to all Black Duck projects and can:

- Create/modify/delete projects and project versions.
- Add users with a defined role to projects as well as removing users from projects.
- Manage tags.
- Map or unmap scans to projects.
- Run/delete project vulnerability and project version reports.
- View BOMs.
- Add/edit/view comments.

- Global Project Manager

The Global Project Manager is similar to the Global Project Administrator in that they have access to all Black Duck projects. However, they also have the ability to manage BOMs. Global Project Managers can:

- Create/modify/delete projects and project versions.
- Add/remove users from projects but cannot define their roles. Users added to projects by a global project manager will have read only access to the projects and will not be able to edit or modify the BOM.
- Manage tags.
- Map or unmap scans to projects.
- Run/delete project vulnerability reports, project version reports (must be assigned to a project to view data).
- View/create/modify/delete BOMs.
- Add/edit/view comments.

- Global Project Viewer

The Global Project Viewer can view *all* projects. Users with this role can view all BOMs but cannot edit the BOM; they can only add or edit comments.

When you assign a user this role, they automatically have read-only access to all projects – you do not have to assign the users to the projects.

This role is often assigned to executives and users in the Legal department.

- Global Release Creator

The Global Release Creator can create releases or versions of projects.

This role is often assigned to a user account used for continuous integration (CI) builds and sometimes, in smaller organizations, given to a release/build engineer who manages all builds for a company.

- Global Security Manager

This Security Manager can create, edit, or delete global remediation statuses for vulnerabilities associated with components.

In smaller organizations this role is often assigned to the development manager while in larger enterprises this is commonly assigned to someone in the security group reporting to the CISO.

- Integration Manager

This role grants the ability to manage all [integrations](#).

- License Manager

The License Manager is responsible for approving and/or rejecting licenses and managing the licenses that can be used in applications. Users with this role can create, edit, and delete custom licenses, custom license terms, and custom license families. They can also manage BlackDuck KnowledgeBase licenses and license terms.

This role is often assigned to someone within the Legal department.

- Lite Global Project Manager

This role grants administration privileges to a Lightweight BOM.

- Policy Manager

The Policy Manager can create, edit, or delete global policy rules.

The Policy Manager role should be assigned to users who are responsible for defining and managing all your OSS company policies. Often, these users are from the Legal/Compliance department or the IT/Security department. This user can also be the CTO overseeing all technology/development or the CISO who is responsible for all security practices.

- Project Creator

The Project Creator can create projects and can edit project and settings.

The Project Creator role is often assigned to the Global Code Scanner or the Project Code scanner if that user needs to create new projects. The Global Code Scanner should almost always have the Project Creator role as well unless your organization has a centrally managed system for setting up new applications company wide.

- System Administrator

The System Administrator role can configure system settings.

The System Administrator role is geared primarily to the user that installs, sets up, and configures the Black Duck application. Most of the time, this will be an IT person responsible for registering the product, configuring LDAP and SSO, and so on.

- User Administrator

The User Administrator manages [users](#) and [groups](#), including [resetting passwords](#). They can also manage [access tokens](#).

This role should be assigned to users who manage people and teams working in your organization, such as a development managers or supervisors.

Project roles

The following project roles are available:

- BOM Annotator

The BOM Annotator can add or edit comments in a BOM for a specific project, but cannot edit the BOM. Users with this role can also update BOM component [custom fields](#).

- BOM Manager

The BOM Manager can modify the BOM for projects in which they are members or have project-group privileges, including modifying component identifications, ignoring components, updating the review status, adding comments, and running project version reports.

This role is often assigned to a lead developer or developer manager for a project.

- Project Code Scanner

The Project Code Scanner only has access to specific project scans in Black Duck and can map or delete scans for that project within the system. Unlike the Global Code Scanner, the Project Code Scanner only has code scanning capability for a set of projects – users are restricted from all other projects. The Project Code Scanner can create project versions of projects they have access to but cannot create projects.

This role is often used in larger enterprises where multiple groups are responsible for builds/releases. This role could be assigned to a release engineer for a specific business unit or for a CI account for that business unit.

- Project Manager

Similar to the Global Project Manager, the Project Manager has complete access to a specific Black Duck project. Project Managers can create/modify/delete versions for projects in which they are members or have project-group privileges but cannot create projects. Project Managers can run reports, modify BOM entries, and assign users to the project but cannot define their roles. Users added to projects by a project manager will have read only access to the projects and will not be able to edit or modify the BOM.

By default Project Managers can manage policy violations and remediate security vulnerabilities. However, the system administrator can [disable these capabilities](#).

In smaller organizations this role is often assigned to the development manager or team lead and in larger enterprises this role could be assigned to the Director of engineering.

- Project Group Administrator

The Project Group Administrator can view/create/modify/delete sub-project groups, including adding users to the project group in which they are a member.

- Project Administrator

Similar to the Project Manager, the Project Administrator has to a specific Black Duck project group. Project Administrators can create/modify/delete versions for projects in which they are members or have project-group privileges but cannot create projects. This also includes managing tags on project versions.

- Project Viewer

The Project Viewer role provides read-only access to individual projects. This is the lowest level of access and is often assigned to users who need to view information and access reports but should not be allowed to change anything. Project Viewers can add comments to a BOM.

This role is assigned to users by default if no other role is assigned to the user: a user without any project roles (no other project roles selected), will be a project viewer. This role is not shown as a selectable option.

- Policy Violation Reviewer

The Policy Violation Reviewer can override policies in projects in which they are members or have project-group privileges.

In smaller organizations this role is often assigned to a development manager, Director or VP of engineering, or even a program manager. In larger enterprises this role is often assigned to users who manage the OSS policies across the entire system. These users verify that what was needed to obtain approval for an override was completed as well as vet the validity of the override for each instance.

- Security Manager

This Security Manager can modify remediation for vulnerabilities associated with components.

In smaller organizations this role is often assigned to the development manager while in larger enterprises this is commonly assigned to someone in the security group reporting to the CISO.

Project Group roles

The following project group roles have the same permissions as their project-only counterparts, except they apply for every project in their assigned project group:


- BOM Annotator
- BOM Manager
- Project Group Administrator
- Project Code Scanner
- Project Manager
- Project Viewer
- Policy Violation Reviewer
- Security Manager

Direct Access vs Indirect Access

New concepts used in Project Groups are Direct Access and Indirect Access to a project. Direct Access refers to a user being directly linked to a project. This has been the normal behavior and remains unchanged with the advent of Project Groups. Introduced with Project Groups, Indirect Access means that a user is linked to a project as a result of being in a user group that is linked to a project group, or because the project is in a project group to which this user is associated.

Managing user roles

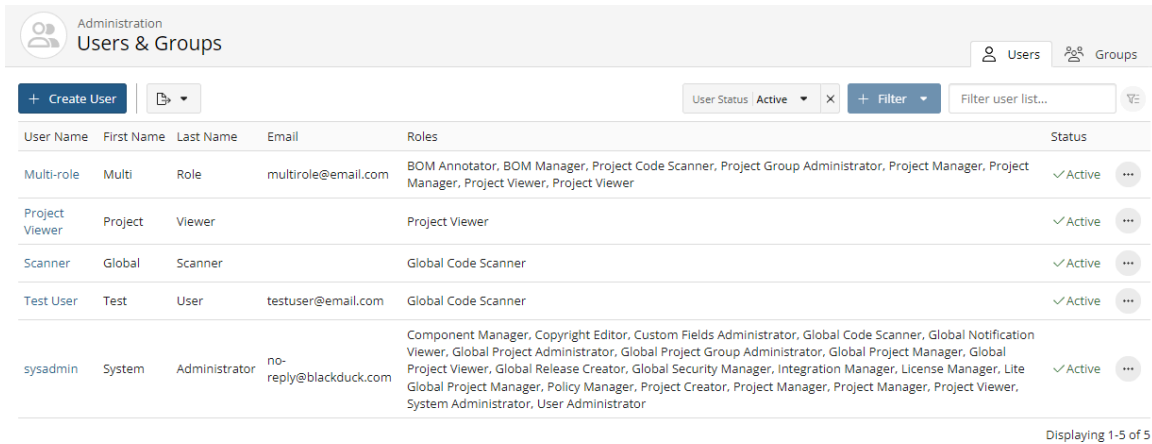
Once you have created a user account, you can add [overall roles](#) to the user account. These overall roles specify what actions the user is able to perform and what information the user can view in Black Duck. Click [here](#) for more information on the tasks that can be performed for each role.

 **Note:** If you do not assign a role to a user, that user has read-only access to Black Duck; this user cannot create projects. However, if a user with no roles is added as a project member, that user will be able to update and delete that project.

To assign an overall role to a user:



2. Select **Users & Groups** to display the **User & Groups Management** page.



User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

The roles assigned to each user appears on the page.


3. Find the user to whom you want to assign a role:

- Select the **Display Inactive Users** check box to include inactive users. Clearing this check box hides all inactive users.
- Filter the users that appear on the page.
- Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.

4. Select the username to display the *Username* page.

5. In the **Overall Roles** section, select the global **roles** that you want to assign to this user account. Deselect any roles that you want to remove from this user account.

The role is automatically assigned or removed. You do not have to save your configuration information.

 **Note:** Users can also obtain roles via user groups. Roles obtained through user groups are not shown in the **Overall Roles** section; instead the **User Groups** section lists roles for each user group.

Viewing your roles


Use the Profile page to view the roles assigned to your user account.

To view your roles:

1. Log in to Black Duck.
2. From the user menu located on the top navigation bar, select **Profile**.

The Profile page appears.

Select **Overall Roles** to view the roles assigned to your user account. Note that this section includes all roles that were assigned to you via user groups.

 **Note:** Users with the User Administrator role can view the roles assigned to a user account by selecting the username in the Users & Groups page.

Black Duck user role matrix

The roles assigned to a user or group determine the tasks that can be performed. You can assign multiple roles (or no roles) to a user or group.

Roles are also assigned to a user when a user is assigned as a member of a project or a project group.

Global roles by task

Task	Roles (details or restrictions)
Manage code scans/Protex BOM files: <ul style="list-style-type: none"> Scan code Upload scans to Black Duck. Map or unmap scans to projects Delete scans 	<ul style="list-style-type: none"> Global Project Administrator (Map/unmap/delete scans only) Global Project Manager (Map/unmap/delete scans only) Global Code Scanner
Create, edit, delete projects	<ul style="list-style-type: none"> Global Project Administrator Global Project Manager Project Creator (edit/delete solely the project created by the corresponding user)

Task	Roles (details or restrictions)
Add or remove users from a project	<ul style="list-style-type: none"> • Global Project Administrator (add users with a defined role) • Global Project Manager (Add users but cannot define their roles. Users added to projects by a global project manager will have read only access to the projects and will not be able to edit or modify the BOM.) • User Administrator (add users with a defined role) • Global Project Group Administrator (add users with a defined role)
Manage projects versions: <ul style="list-style-type: none"> • Create, edit, delete project versions • Edit project or version settings, including tags 	<ul style="list-style-type: none"> • Global Project Administrator • Global Project Manager • Global Release Creator (Create permission only) • Project Creator (See Project Manager role for permissions obtained when creating a project)
Manage custom components	<ul style="list-style-type: none"> • Component Manager • Global Project Administrator • Global Project Manager
Manage licenses: <ul style="list-style-type: none"> • Create, edit, delete custom licenses • Manage KnowledgeBase licenses • Create, edit, delete custom license families • Manage KB and custom license terms 	<ul style="list-style-type: none"> • License Manager
View BOMs: <ul style="list-style-type: none"> • View BOM • Add/edit/view comments • Print BOM • Compare BOMs 	<ul style="list-style-type: none"> • Global Project Administrator (Cannot edit comments created by other users) • Global Project Manager • Global Project Viewer (View all projects only) • Any other user assigned to the project
Manage BOMs: <ul style="list-style-type: none"> • Manually add components; delete manually added components • Ignore components • Review components • Remediate security vulnerabilities • Override policy violations • Remove override of policy violations • Edit licenses, including excluding license from Notices File report, adding an attribution statement, or selecting a different license for a component version • Indicate license term fulfillment status • Manage deep license data • View license conflicts 	<ul style="list-style-type: none"> • Global Project Manager
Manage policy rules:	<ul style="list-style-type: none"> • Policy Manager

Task	Roles (details or restrictions)
<ul style="list-style-type: none"> • Create, edit, or delete policy rules 	
Update Additional (Custom, SBOM) fields	<ul style="list-style-type: none"> • Component Manager (Can only update Component and Component Version custom fields) • Custom Fields Administrator (Can only create, edit, delete Custom Fields) • Global Project Administrator (Update custom field values for project, project version, and BOM)
Create, edit, or delete global remediation statuses	<ul style="list-style-type: none"> • Global Project Administrator • Global Project Manager • Global Security Manager (Must be assigned to a project to view data)
Run project vulnerability reports from the Reports menu	<p>The following roles can create a project vulnerability report for any project:</p> <ul style="list-style-type: none"> • Global Project Manager • Global Project Administrator • Global Project Viewer <p>The following roles must be assigned to a project to create a project vulnerability report:</p> <ul style="list-style-type: none"> • Component Manager • Copyright Editor • Custom Fields Administrator • Global Code Scanner • Global Project Group Administrator • Global Release Creator • Global Security Manager • License Manager • Policy Manager • Project Creator • System Administrator • User Administrator
Create and modify copyright statements	<ul style="list-style-type: none"> • Copyright Editor (Must be assigned to a project to view data)
Run Project version reports: <ul style="list-style-type: none"> • Version Details report • Vulnerability report • Notices File report • Software Bill of Materials (SBOM) report 	<p>The following roles can create a project version report for any project:</p> <ul style="list-style-type: none"> • Global Project Manager • Global Project Administrator <p>The following roles must be assigned to a project to create a project version report:</p> <ul style="list-style-type: none"> • Component Manager • Copyright Editor • Custom Fields Administrator • Global Code Scanner • Global Project Group Administrator • Global Project Viewer

Task	Roles (details or restrictions)
	<ul style="list-style-type: none"> • Global Release Creator • Global Security Manager • License Manager • Policy Manager • Project Creator • System Administrator • User Administrator
Delete Project version reports	<p>The following roles can delete a project version report for any project:</p> <ul style="list-style-type: none"> • Global Project Manager • Global Project Administrator <p>The following roles can only delete project version reports created by themselves:</p> <ul style="list-style-type: none"> • Component Manager • Copyright Editor • Custom Fields Administrator • Global Code Scanner • Global Project Group Administrator • Global Project Viewer • Global Release Creator • Global Security Manager • License Manager • Policy Manager • Project Creator • System Administrator • User Administrator
View information in Dashboard pages	<p>The following roles can view any project from the Dashboard page:</p> <ul style="list-style-type: none"> • Global Project Manager • Global Project Administrator <p>The following roles can only view any projects to which they are associated on the Dashboard page:</p> <ul style="list-style-type: none"> • Component Manager • Copyright Editor • Custom Fields Administrator • Global Code Scanner • Global Project Group Administrator • Global Project Viewer • Global Release Creator • Global Security Manager • License Manager • Policy Manager • Project Creator • System Administrator • User Administrator
Access the Tools page:	All roles

Task	Roles (details or restrictions)
<ul style="list-style-type: none"> Download the scanner Access links to the Community and Customer Education 	
Use the Search function	All roles
Administer Black Duck. Use the Admin menu to:	<ul style="list-style-type: none"> System Administrator
<ul style="list-style-type: none"> View jobs Register Black Duck. Configure LDAP Configure SAML Manage system settings Manage system announcements Configure password requirements 	
Administer users and groups. Use the Admin menu to:	<ul style="list-style-type: none"> User Administrator
<ul style="list-style-type: none"> Manage users, including resetting passwords Manage groups 	
Manage snippets	<ul style="list-style-type: none"> Global Project Manager Global Project Administrator
View issues	<ul style="list-style-type: none"> Global Project Manager Global Project Administrator
Manage project groups:	<ul style="list-style-type: none"> Global Project Group Administrator
<ul style="list-style-type: none"> Create/Edit/Delete project groups Add/Remove members and user groups from project groups 	
Manage Access Tokens	<ul style="list-style-type: none"> User Administrator
View notifications	<ul style="list-style-type: none"> Global Notification Viewer (View notifications for all projects and receives all system notifications regardless of user preferences)
Download the heatmap CSV report	<ul style="list-style-type: none"> System Administrator
View the scan heatmap	<ul style="list-style-type: none"> Global Project Administrator Global Project Manager Global Notification Viewer Global Project Viewer Global Code Scanner
Manage integration servers	<ul style="list-style-type: none"> Integration Manager
Manage lightweight BOMs	<ul style="list-style-type: none"> Lite Global Project Manager

Task	Roles (details or restrictions)
Convert project versions to LTS	<ul style="list-style-type: none"> Global Project Administrator Global Project Manager

Project and Project Group roles


Task	Roles (details or restrictions)
Manage project groups: <ul style="list-style-type: none"> Create/Edit/Delete project groups Add/Remove members and user groups from project groups 	<ul style="list-style-type: none"> Project Group Administrator (below parent group)
Manage code scans/Protex BOM files: <ul style="list-style-type: none"> Scan code Upload scans to Black Duck. Map or unmap scans to projects Delete scans 	<ul style="list-style-type: none"> Project Manager (Can unmap/delete scans from their projects) Project Group Manager (Can unmap/delete scans from their projects) Project Code Scanner (Can map/unmap/delete a code scan to/from projects for which they have access) Project Group Code Scanner
Create, edit, delete projects	<ul style="list-style-type: none"> Project Administrator (Delete/Edit only) Project Group Administrator (must already have access to these projects) Project Manager (cannot create projects but can delete projects to which they are associated)
Manage projects: <ul style="list-style-type: none"> Create, edit, delete project versions Edit project or version settings, including tags 	<ul style="list-style-type: none"> Project Administrator Project Manager (Only projects they manage) Project Group Administrator (must already have access to these projects) Project Code Scanner (Can only create project versions)
Add or remove users or groups to projects	<ul style="list-style-type: none"> Project Administrator (add users with a defined role on projects they administer) Project Manager (Add users but cannot define their roles on projects they administer. Users added to projects by a project manager will have read only access to the projects and will not be able to edit or modify the BOM.) Project Group Administrator (must already have access to these projects)
Manage custom licenses: <ul style="list-style-type: none"> Create, edit, delete custom licenses 	<ul style="list-style-type: none"> BOM Manager
View BOMs: <ul style="list-style-type: none"> View BOM View notifications Add/edit/view comments Print BOM 	<ul style="list-style-type: none"> All roles

Task	Roles (details or restrictions)
<ul style="list-style-type: none"> Compare BOMs 	
<p>Manage BOMs:</p> <ul style="list-style-type: none"> Manually add components; delete manually added components Ignore components Review components Edit licenses, including excluding license from Notices File report, adding an attribution statement, or selecting a different license for a component version Indicate license term fulfillment status Manage deep license data Update custom field information View license conflicts 	<ul style="list-style-type: none"> Project Administrator (Update Project and Project Version custom fields, view BOM custom fields) Project Manager Project Group Administrator (must already have access to these projects, Update Project and Project Version custom fields, view BOM custom fields). BOM Manager
<p>Manage policy violations:</p> <ul style="list-style-type: none"> Override policy violations Remove override of policy violations 	<ul style="list-style-type: none"> Project Manager (Can only manage policy violations if enabled by the system administrator>) Policy Violation Reviewer
Remediate security vulnerabilities	<ul style="list-style-type: none"> Project Manager (Can only remediate security vulnerabilities if enabled by the system administrator) Security Manager (Can only modify remediation for vulnerabilities associated with components)
Update custom field values	<ul style="list-style-type: none"> Project Manager (Can only update BOM Component, Project, and Project Version custom fields) BOM Annotator (Can only update BOM Component custom field) BOM Manager (Can only update BOM Component custom field)
<p>Manage policy rules:</p> <ul style="list-style-type: none"> Create, edit, or delete policy rules 	<ul style="list-style-type: none"> No project or project group level role can perform this task
<p>Run project vulnerability reports from the Report menu:</p> <ul style="list-style-type: none"> Vulnerability Remediation Report Vulnerability Status Report Vulnerability Update Report 	<ul style="list-style-type: none"> All roles
<p>Run Project version reports:</p> <ul style="list-style-type: none"> Version Details report Vulnerability report Notices File report Software Bill of Materials (SBOM) report 	<ul style="list-style-type: none"> All roles
Delete project version reports	<p>The following roles can delete all reports:</p> <ul style="list-style-type: none"> Project Administrator Project Manager


Task	Roles (details or restrictions)
	<ul style="list-style-type: none"> Project Group Administrator (must already have access to these projects) <p>The following roles can delete reports generated by themselves:</p> <ul style="list-style-type: none"> All roles
View information in Dashboard pages	<ul style="list-style-type: none"> All roles
Access the Tools page from which user can:	<ul style="list-style-type: none"> All roles <ul style="list-style-type: none"> Download the scanner Access API documentation
Search	<ul style="list-style-type: none"> All roles
Manage snippets	<ul style="list-style-type: none"> Project Administrator Project Manager BOM Manager
Convert project versions to LTS	<ul style="list-style-type: none"> Project Administrator Project Manager

Managing the Project Manager and Project Group Manager roles

System administrators can define whether users with the Project Manager or Project Group Manager role can manage policy violations (override policy violations or remove overrides) or remediate security vulnerabilities for a project.

 **Note:** This is a global setting: all users with the Project Manager or Project Group Manager role are affected by any changes you make to the role.

To modify the Project Manager or Project Group Manager role:

1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **System Settings**.
4. Click **Roles**.
5. In the **Roles** section, select or clear the **Policy Violation Reviewer** and/or **Security Manager** options.

Roles

The roles checked below will have their permissions applied to the Project Manager and Project Group Manager roles.

Policy Violation Reviewer

Security Manager

Reset


Save

6. Click **Save**.

Authenticating users with LDAP

Authenticating users through an existing LDAP corporate directory helps to facilitate:

- The creation of user accounts. If the user account does not exist, upon successful authentication, the Black Duck user account is created.
- Centralized management of user account details. Each time a user logs in to Black Duck, Black Duck synchronizes with the directory server. If changes were made to mapped attributes, Black Duck updates the user account information.
- (Optional) The creation of groups. If a user is a member of an LDAP group, upon successful authentication, a Black Duck user account, as well as a Black Duck group, is created. The group is populated with the new user.

 **Note:** Note: If the Black Duck group already exists, the Black Duck user account is created, and the group is populated.

To authenticate users with LDAP:

1. Contact your LDAP administrator and gather the following information:

LDAP server details

This is the information that Black Duck uses to connect to the directory server.

- **Server URL** (required): The host name or IP address of the directory server, including the protocol scheme and port, on which the instance is listening.

Example: ldap://<server_name>.<domain_name>.com:339

Click [here](#) for more information on configuring secure LDAP.

- **Authentication Type:** If credentials are required for LDAP access, the authentication type to use: Simple, None, or Digest-MD5.
- **Manager DN** (optional): If your organization does not use anonymous authentication, and requires credentials for LDAP access, the password and either the LDAP name or the absolute LDAP distinguished name (DN) of a user that has permission to read the directory server.

Example of an absolute LDAP DN: uid=ldapmanager,ou=employees,dc=company,dc=com

Example of an LDAP name: jdoe

LDAP users attributes and LDAP attribute mappings

This is the information that the Black Duck uses to locate users in the directory server:

- **User Search Base** (required): The absolute base DN under which users can be located.
Example: `dc=example,dc=com`
- **User Search Filter** (required) The attribute used to match a specific, unique user. The value of this attribute personalizes the user profile icon with the name of the user.
Example: `uid={0}`
- **User DN Pattern** (optional): If some of your users are not located under the absolute base DN for the user search, the user DN pattern is used to match a specific, unique user.
Example: `cn={0},ou=contractors`
- **First Name, Last Name, Email** (optional) The attributes that map to the first name, last name, and email address of users.

LDAP groups

If you are enabling LDAP group synchronization, this is the required information that Black Duck uses to locate user groups in the directory server:

- **Group Search Base** (required): The absolute base DN under which groups can be located.
Example: `ou=groups,dc=example,dc=com`
- **Group Filter** (required) The attribute used to match a unique user member within a given group.
Example: `uniquemember={0}`
- **Group Name Attribute** (required) The attribute that identifies a specific, unique group name.
Example: `cn`

2. Log in to Black Duck as a system administrator.



4. Select **System Settings**.

5. Select **User Authentication**.

6. In the **External Authentication** section:

- Check the **Enable LDAP Configuration** checkbox.
- Enter the server connection and authentication details that Black Duck is to use to connect to the directory server.

7. In the **LDAP User Attributes** section, enter the user attributes values Black Duck is to use to locate users.

Optionally, clear the **Create user accounts automatically in Black Duck** check box to turn off the automatic creation of users when they authenticate with LDAP. This check box is selected by default so users that do not exist in Black Duck are created automatically when they log into Black Duck using LDAP. This applies to new installs and upgrades.

8. (Optional) Enter the attributes that map to user-specific information in the **LDAP Attribute Mappings** section.


9. (Optional) Select **Synchronize LDAP groups** and enter the group attribute values Black Duck is to use to locate groups in the **LDAP Groups** section.
10. (Optional) Enter user credentials in the **Test Connection, User Authentication and Field Mapping** section and click **Test Connection** to test the connection to the directory server.
If the LDAP group synchronization is enabled and configured, the user's first name, last name, email address, and user's LDAP groups are displayed for successful connections.
11. Click **Save**.

Configuring secure LDAP

If you see certificate issues when connecting your secure LDAP server to Black Duck, the most likely reason is that the Black Duck server has not set up a trust connection to the secure LDAP server. This usually occurs if you are using a self-signed certificate.

To set up a trust connection to the secure LDAP server, import the server certificate into the local Black Duck LDAP truststore by:

1. Obtaining your LDAP information.
2. Using the Black Duck UI to import the server certificate.

 **Note:** All hosted customers should secure access to their Black Duck application by leveraging our out-of-the-box support for single sign on (SSO) via SAML or LDAP. Information on how to enable and configure these security features can be found in the installation guides. In addition, we encourage customers that are using a SAML SSO provider that offers two-factor authorization to also enable and leverage that technology to further secure access to their Black Duck application.

Obtaining your LDAP information

Contact your LDAP administrator and gather the following information:

LDAP Server Details

This is the information that Black Duck uses to connect to the directory server.

- (required) The host name or IP address of the directory server, including the protocol scheme and port, on which the instance is listening.
Example: `ldaps://<server_name>.<domain_name>.com:339`
- (optional) If your organization does not use anonymous authentication, and requires credentials for LDAP access, the password and either the LDAP name or the absolute LDAP distinguished name (DN) of a user that has permission to read the directory server.
Example of an absolute LDAP DN: `uid=ldapmanager,ou=employees,dc=company,dc=com`
Example of an LDAP name: `jdoe`
- (optional) If credentials are required for LDAP access, the authentication type to use: simple or digest-MD5.

LDAP Users Attributes

This is the information that Black Duck uses to locate users in the directory server:

- (required) The absolute base DN under which users can be located.
Example: `dc=example,dc=com`
- (required) The attribute used to match a specific, unique user. The value of this attribute personalizes the user profile icon with the name of the user.
Example: `uid={0}`

Test Username and Password

- (required) The user credentials to test the connection to the directory server.

Importing the server certificate

To import the server certificate:

1. Log in to Black Duck as a system administrator.



3. Select **System Settings**.
4. Click **User Authentication**.
5. Check the **Enable LDAP Configuration** checkbox and complete the information in the **LDAP Server Details** section, as described above. In the **Server URL** field, ensure that you have configured the secure LDAP server: the protocol scheme is `ldaps://`.
6. Complete the information in the **LDAP User Attributes** section, as described above.

Optionally, clear the **Create user accounts automatically in Black Duck** check box to turn off the automatic creation of users when they authenticate with LDAP. This check box is selected by default so users that do not exist in Black Duck are created automatically when they log into Black Duck using LDAP. This applies to new installs and upgrades.

7. Enter the user credentials in the **Test Connection, User Authentication and Field Mapping** section and click **Test Connection**.
8. If there are no issues with the certificate, it is automatically imported and the "Connection Test Succeeded" message appears:

Test Connection, User Authentication and Field Mapping

Tests ability to connect and authenticate test-user. Note: test-user credentials are not saved.

✓ Connection Test Succeeded

Test Username *

Denis Bors

Test Password *

.....

Reset

Test Connection

9. If there is an issue with the certificate, a dialog box listing details about the certificate will appear. Do one of the following:
 - Click **Cancel** to fix the certificate issues.

Once fixed, retest the connection to verify that the certificate issues have been fixed and the certificate has been imported. If successful, the "Connection Test Succeeded" message appears.


- Click **Save** to import this certificate.

Verify that the certificate has been imported by clicking **Test Connection**. If successful, the "Connection Test Succeeded" message appears.

About locked out user accounts

A user will be locked out of their account for 10 minutes if they fail to enter the correct password after 10 attempts. After the 10th failed attempt, a message will appear on the login page notifying the user that their account is locked.


Log files contain information by username on successful logins, unsuccessful logins, and account lockouts.

 **Note:** This lockout feature does not apply to users logging in using SAML or LDAP.

15. Managing groups in Black Duck

You can use groups in Black Duck to manage overall roles and project team membership for several user accounts at once instead of managing that information at the individual user account level. You can:

- [Create a group](#)
- [Manage group information](#) such as the group name or status
- [Manage group roles](#)
- [Add a member to a group](#)
- [Remove a member from a group](#)
- [Delete a group](#)

 **Note:** If you are using an external LDAP directory server to authenticate users and have enabled LDAP group synchronization, the Group Management page uses the **Source** column to identify groups that were created in Black Duck (**Internal**) and groups that were created because of LDAP authentication (**LDAP**).

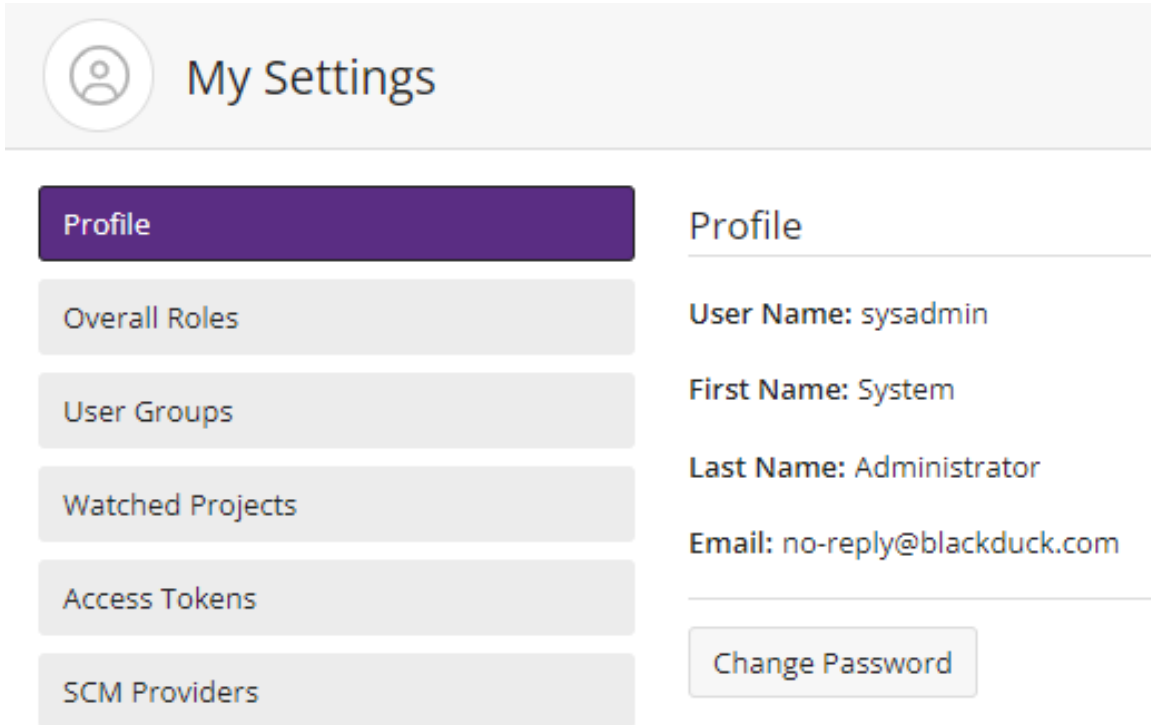
Viewing your groups

You can view the groups you belong to, and the source, status, and roles associated with each group.

To view your groups:

1. Log in to Black Duck.
2. From the user menu located on the top navigation bar, select **Profile**.

The Profile page appears.



My Settings

Profile

Overall Roles

User Groups

Watched Projects

Access Tokens

SCM Providers

Profile

User Name: sysadmin

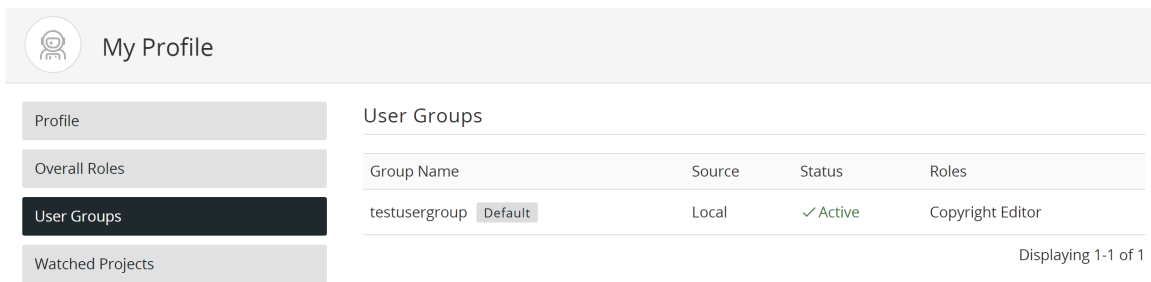
First Name: System

Last Name: Administrator

Email: no-reply@blackduck.com

Change Password

3. Select **User Groups** to view the group in which you are a member.



My Profile

Profile

Overall Roles

User Groups

Watched Projects

User Groups

Group Name	Source	Status	Roles
testusergroup <small>Default</small>	Local	✓ Active	Copyright Editor

Displaying 1-1 of 1

You can [view the groups associated with a particular user](#).

Creating groups

You can create and configure a group with specific roles that will be granted to all members of the group.

If you create a default group, subsequent new users are automatically added to this group and are granted all roles and access to all projects configured for this group. Note that:

- You can have more than one default group.
- Default groups have a status of *Status* - Default.

To create a group:

1. Log in to Black Duck.

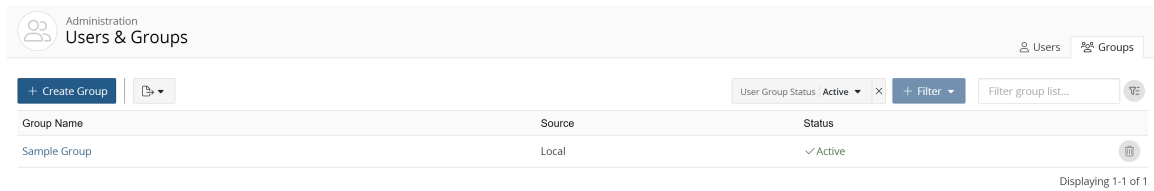
2.



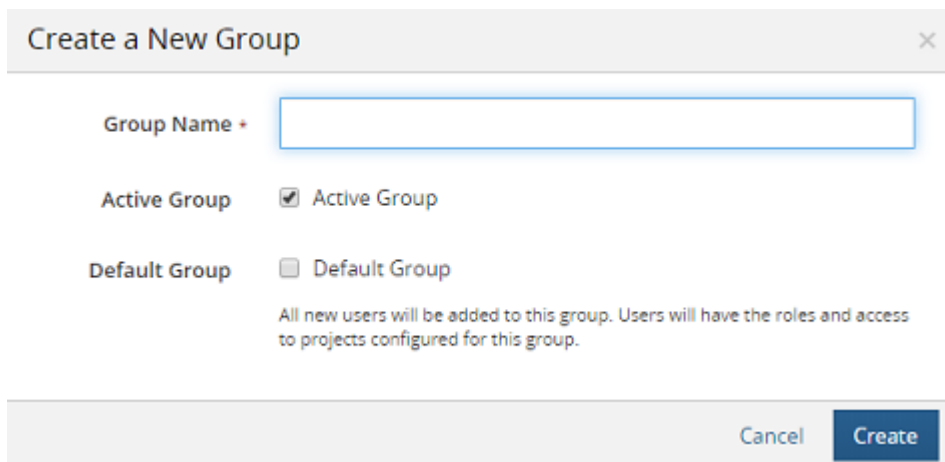
Click **Admin**.

3. Select **Users & Groups** to display the **User & Group Management** page.

4. Click the **Groups** tab.



5. Click **Create Group** to display the Create a New Group dialog box.



6. In the Create a New Group dialog box:

- a. Type the name of the group in the **Group Name** field.
- b. Select whether this group is active or inactive.
- c. Select whether this group is a default group.
- d. Click **Create**. The Group Management page updates to display the new group.

You can now:

- [Add members](#) to the group.
- [Assign roles](#) to the group.

Managing group information

After you have created a group, you can change the group name, status (active/inactive), and/or whether this is a default group.

To manage group information:

1. Log in to Black Duck.



3. Select **Users & Groups** to display the **User & Group Management** page.

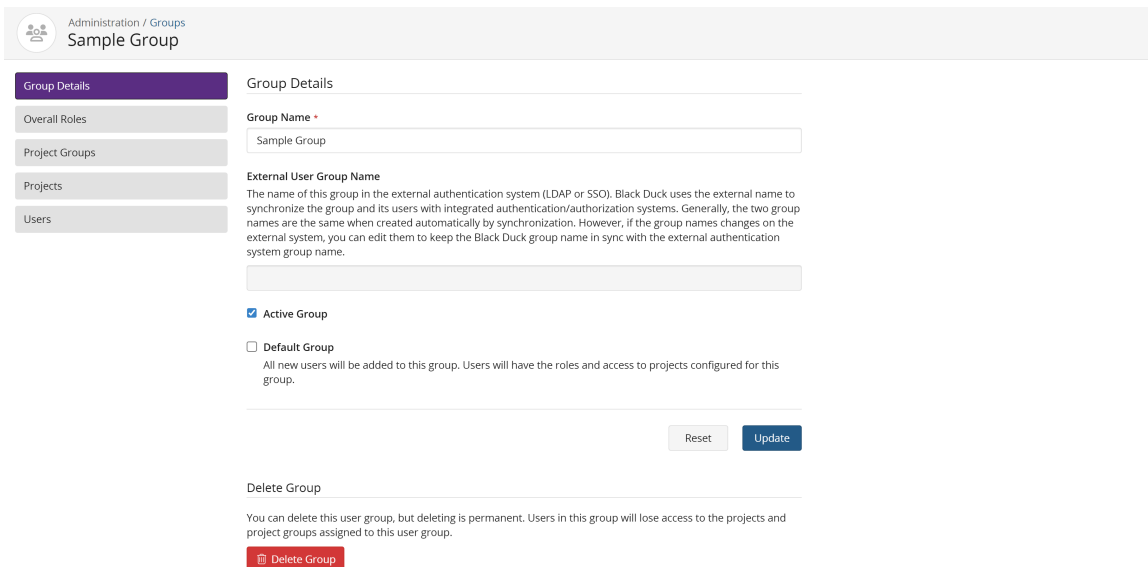
4. Click the **Groups** tab.



5. Find the name of the group whose name you want to modify:

- Select the **Display Inactive Groups** check box to include inactive groups. Clearing this check box hides all inactive groups.
- Filter the groups that appear on the page.
- Sort the list of group names by selecting the column. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more groups than are listed on this page.

6. Select the group name you want to edit to display the *Group Name* page.



7. On the **Group Details** page, type the new group name, change the status, or change whether this is a default group.

Note that if you enabled group synchronization when configuring LDAP or SAML, the name of this group in the external authentication system (LDAP or SSO) appears in the **External Group Name** field. Black Duck uses the external name to synchronize the group and its members with integrated authentication/authorization systems. Generally, the two group names are the same when created automatically by synchronization. However, if the group name changes on the external system, you can edit the name to keep the Black Duck group name in sync with the external authentication system group name.

8. Click **Save** to save the changed information. Black Duck saves the group name and status.
9. Use the other sections on this page to:
 - [Manage group roles](#).
 - [Add or remove](#) group members.
 - [Add or remove](#) projects.

Managing group projects

You can manage the projects assigned to a group using the *Group Name* page.

To assign a project to a group :

1. Log in to Black Duck.

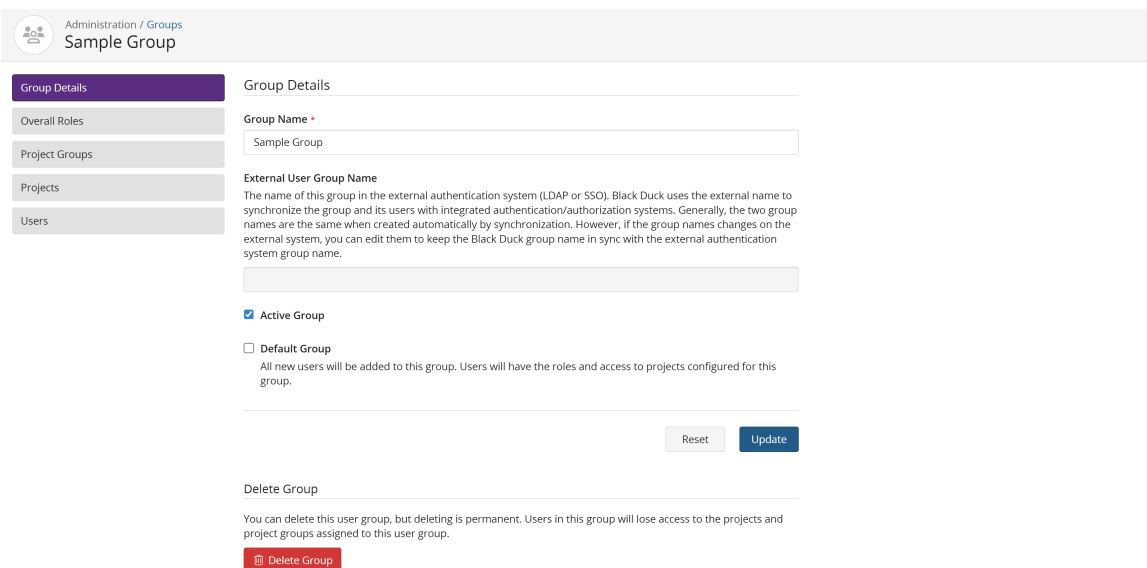


3. Select **Users & Groups** to display the **User & Group Management** page.

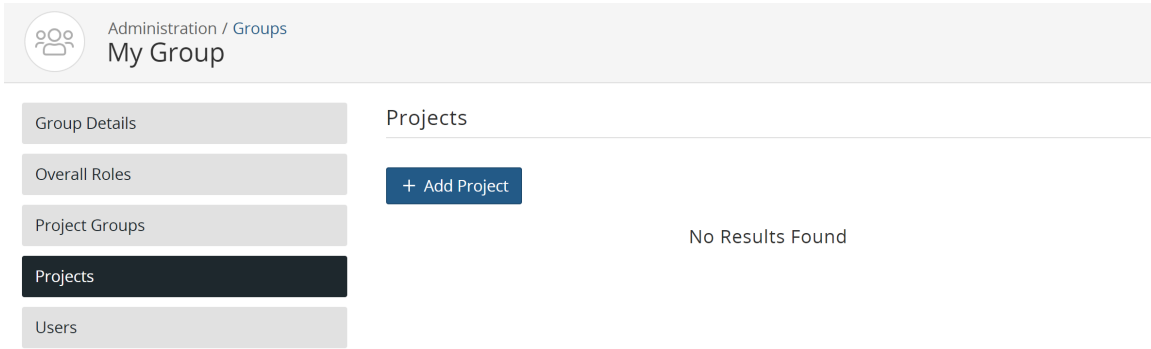
4. Click the **Groups** tab.



5. Select the name of the group to display the *Group Name* page.



6. Click **Projects** in the left-hand menu.



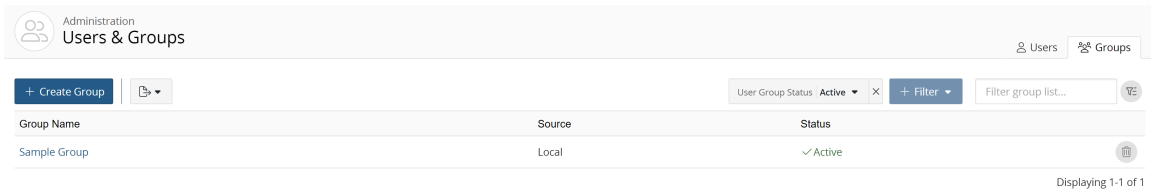
7. Click **Add Project** to display the Add Project dialog box.
8. Enter one or more projects and click **Add**.

To remove a project from a group:

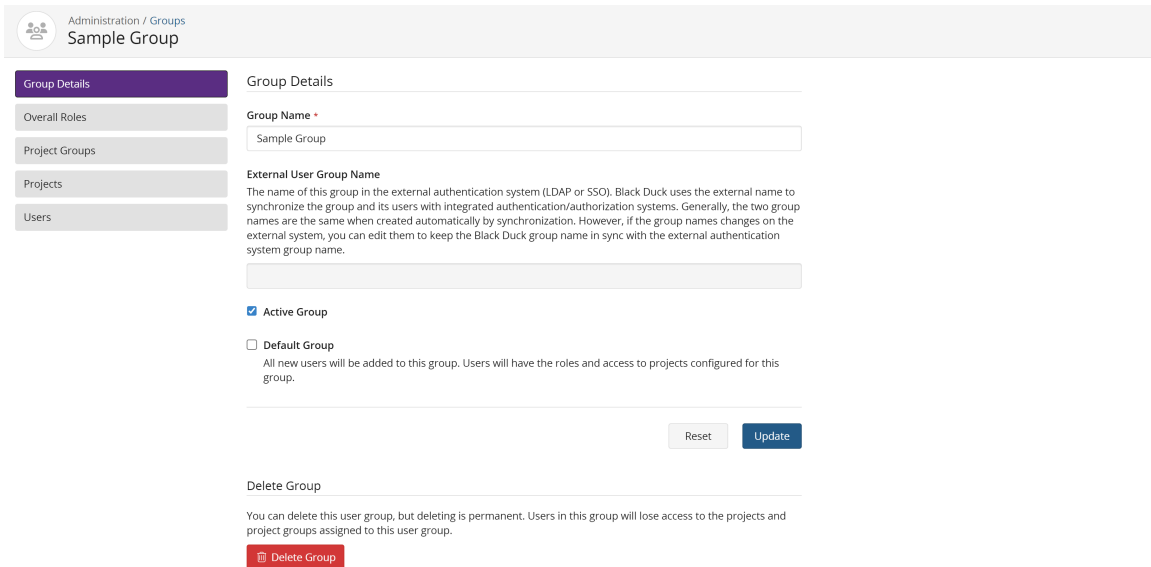
1. Log in to Black Duck.



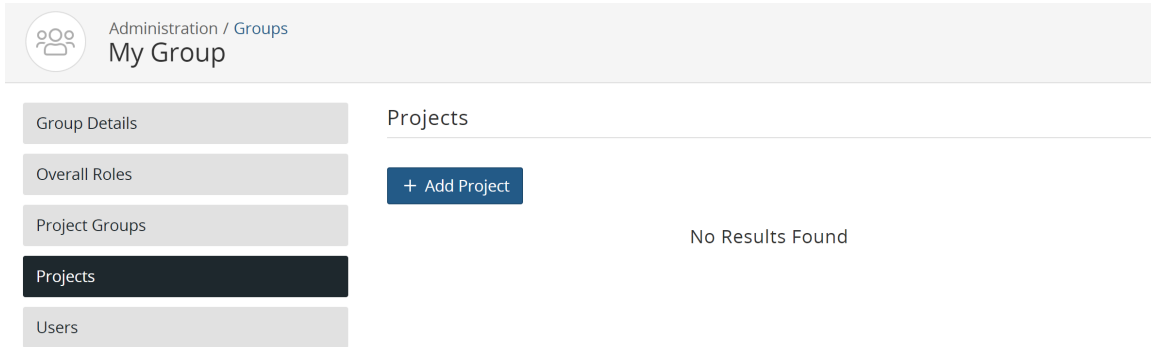
3. Select **Users & Group** to display the **User & Group Management** page.
4. Click the **Groups** tab.




5. Select the name of the group to display the *Group Name* page.



6. Click **Projects**.



7.  in the row of the project you want to remove from the project group.
8. Click **Delete Direct Access** to remove the project.

Managing group roles

Once you have added [overall roles](#) to a group, you can add users to the group, then assign that group to one or more projects. These users will have the overall roles assigned to the group and will be members of all project teams to which the group has been added.

To manage group roles:

1.  Click **Admin**.
2. Select **Users & Groups** and click the **Groups** tab to display the Group Management page.



3. Find the name of the group for which you want to manage roles to display the *Group Name* page:
 - Filter the groups that appear on the page.
 - Sort the list of groups by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more groups than are listed on this page.
4. Select the name of a group to display the *Group Name* page.

The screenshot shows the 'Administration / Groups' interface for a 'Sample Group'. On the left is a navigation menu with options: 'Group Details' (highlighted), 'Overall Roles', 'Project Groups', 'Projects', and 'Users'. The main content area is titled 'Group Details' and contains the following elements:

- Group Name**: A text input field containing 'Sample Group'.
- External User Group Name**: A text input field that is currently empty.
- Active Group**: A checked checkbox.
- Default Group**: An unchecked checkbox with a sub-note: 'All new users will be added to this group. Users will have the roles and access to projects configured for this group.'

At the bottom right of the 'Group Details' section are two buttons: 'Reset' and 'Update'.

Below this section is the 'Delete Group' section, which includes a warning: 'You can delete this user group, but deleting is permanent. Users in this group will lose access to the projects and project groups assigned to this user group.' and a red button labeled 'Delete Group'.

5. Click **Overall Roles** in the left-hand menu.

15. Managing groups in Black Duck • Adding members to a project group

Administration / Groups
Sample Group

Group Details

Overall Roles

Project Groups

Projects

Users

Overall Roles

- Component Manager**
This role can create, update and delete custom components.
- Copyright Editor**
Ability to add and modify copyrights for components.
- Custom Fields Administrator**
This role grants the ability to manage custom fields.
- Global Code Scanner**
This user can run code scans and map them to any project.
- Global Notification Viewer**
This role has read only access to all projects and receives all system notifications regardless of user preferences.
- Global Project Administrator**
This role has access to all project data. This person can create/modify/delete projects and versions, etc.
- Global Project Group Administrator**
This role grants the ability to manage project groups and their members.
- Global Project Manager**
This role grants administration and management privileges to all projects, including editing all BOMs.
- Global Project Viewer**
This role has read only access to all projects.
- Global Release Creator**
This role gives the ability to create releases to any project regardless of membership status.
- Global Security Manager**
Ability to modify default remediation status for vulnerabilities.
- License Manager**
Ability to create/modify/delete licenses.
- Policy Manager**
Ability to create/modify/delete policies.
- Project Creator**
This role can create projects and assumes the Project Manager role for the projects they create.
- System Administrator**
This role will have access to system settings like the registration key, jobs page, LDAP, SSO, etc. This is more for an IT person who installs and manages the hosting of the application.
- User Administrator**
This role grants the ability to manage users and user groups and assign roles to users & groups.

Reset Update

6. Select the roles that you want to assign to all members of this group. Deselect any roles that you want to remove from this group.
The role is automatically assigned to the group. You do not have to save your configuration information.

Adding members to a project group

You can add members to a project group by:

- Managing a project group and adding members to the project group
- Managing a user and adding the user to project groups
- Managing a project and adding members to the project

Note that subsequent users are automatically added to [default groups](#).

Adding members to a project group by managing a project group

1. Log in to Black Duck.



3. Select **Users & Groups** to display the **User & Group Management** page.

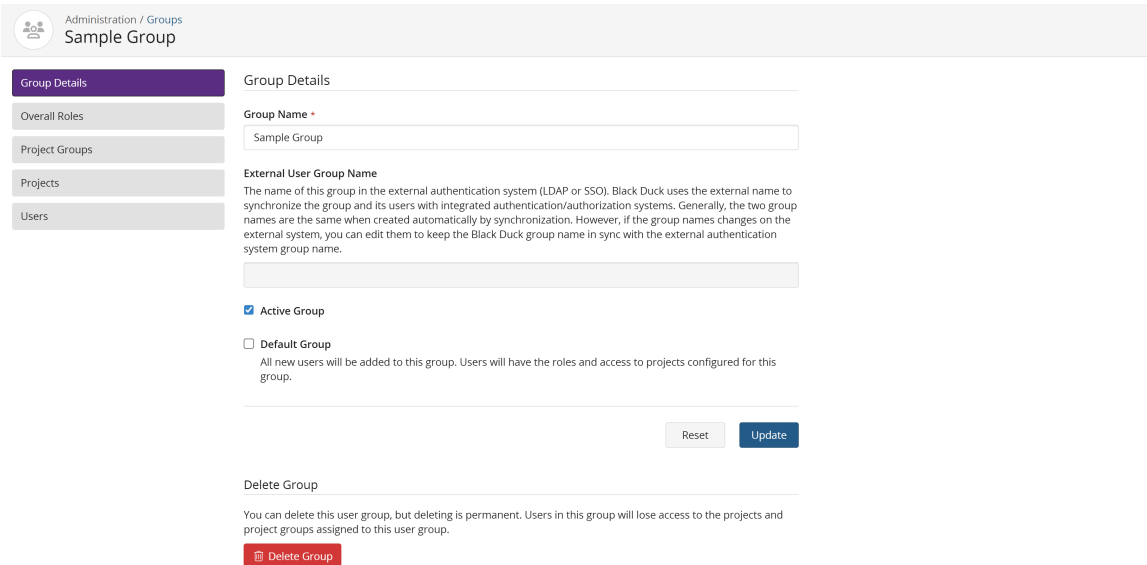
4. Click the **Groups** tab.



5. Find the name of the group for which you want to manage membership:

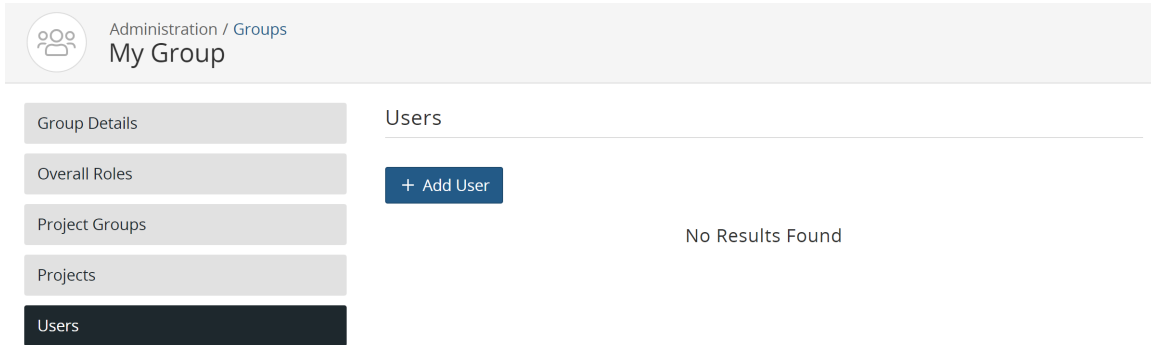
- Select the **Display Inactive Groups** check box to include inactive groups. Clearing this check box hides all inactive groups.
- Filter the groups that appear on the page.
- Sort the list of groups by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more groups than are listed on this page.

6. Select a group to display the *Group Name* page.



7. Click **Users** in the left-hand menu.

15. Managing groups in Black Duck • Adding members to a project group

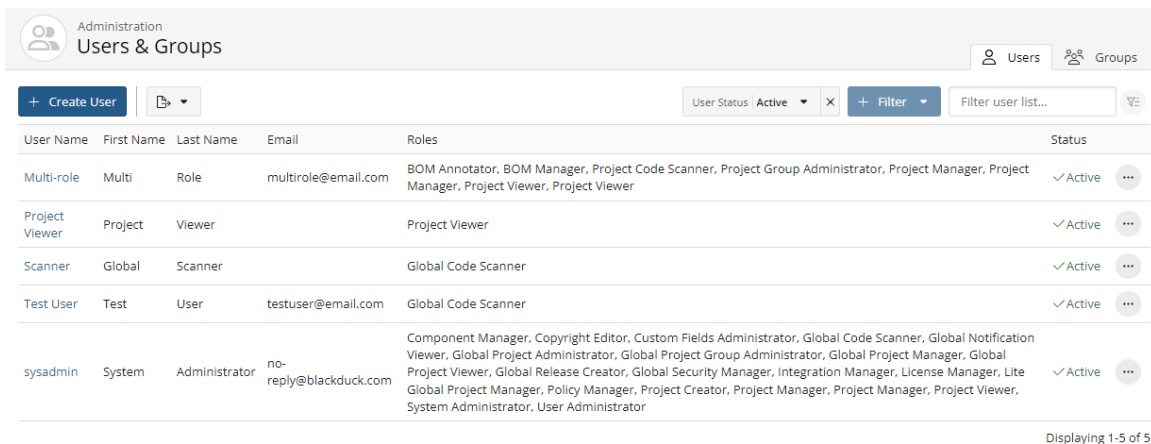


8. Click **+ Add User** to display the Add Users to Group dialog box.
9. Add a user or multiple users by:
 - a. Typing the user name of the user that you want to add to the project team. The list is type-ahead enabled, so you can see a list of available user names that contain the text you have typed.
 - b. Click the **Users** dropdown box to see a list of users.
10. Select any number of users to add to the project group.
11. Click **Add**.

The group member list updates to show the newly-added members.

Adding a member to a project group by managing a user

1. Log in to Black Duck.
2.  Click **Admin**.
3. Select **Users & Groups** to display the **User & Group Management** page.



4. Find the user you want to find:
 - Select the **Display Inactive Users** check box to include inactive users. Clearing this check box hides all inactive users.
 - Filter the users that appear on the page.

- Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.
5. Select the user to display the *Username* page.
 6. Click **Add group** in the **User Groups** section to display the Add Group dialog box.
 7. Begin typing the group name. The list is type-ahead enabled, so you can see a list of available group names that contain the text you have typed.
Select the **Active only** check box to see active groups only.
 8. Select the groups you want this user to join.
 9. Click **Add**.
The group table updates to display the newly-added group(s).
Note that the roles assigned to this user are [determined by the group](#).

Adding a member to a project group by managing a project

1. Log in to Black Duck.
2. Click the project on the Dashboard.
3. Click the **Settings** tab.
4. Click **Groups**.
5. Click the **+ Add User Group** button.
6. Add a user or multiple project groups by:
 - a. Typing the project group name that you want to add to the project team. The list is type-ahead enabled, so you can see a list of available project groups that contain the text you have typed.
 - b. Click the **Group** dropdown box to see a list of project groups.
7. Select the desired [role\(s\)](#) for the chosen project groups.
8. Click **Add**.
The user groups table updates to display the newly-added group(s).

Removing a member from a group

You can remove members from a group by:

- Managing a group and removing members from the group
- Managing a user and removing group membership from the user

To remove a member from a group when managing a group:

1. Log in to Black Duck.

2.  Click **Admin**.

15. Managing groups in Black Duck • Removing a member from a group

3. Select **Users & Groups**.

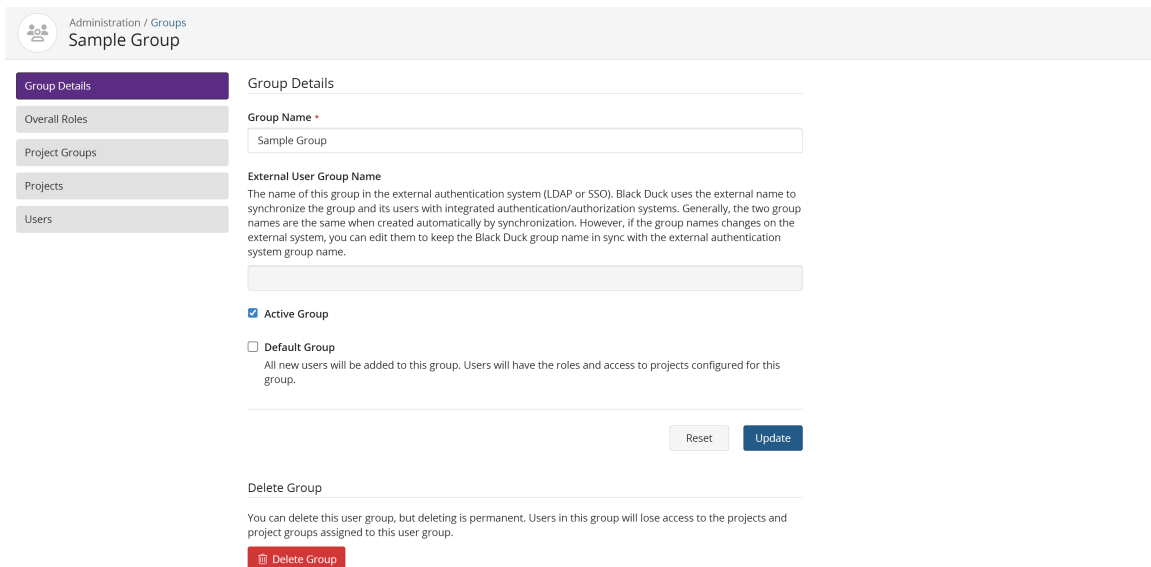
4. Click the **Groups** tab.



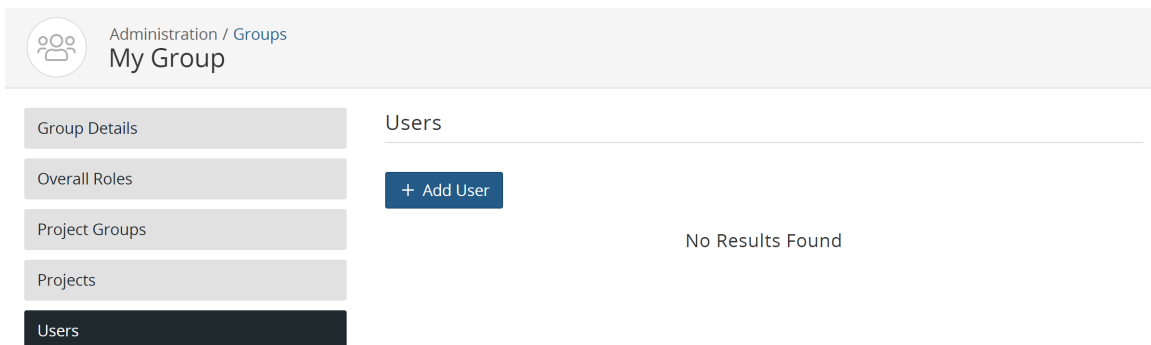
5. Find the name of the group for which you want to manage membership:


- Select the **Display Inactive Groups** check box to include inactive groups. Clearing this check box hides all inactive groups.
- Filter the groups that appear on the page.
- Sort the list of group names by selecting the column. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more groups than are listed on this page.

6. Select the group to display the *Group Name* page.



7. Click **Users** in the left-hand menu.

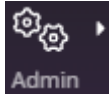


8. Click  in the row of the name of the member you want to remove.
9. In the Remove User from Group dialog box, click **Delete**.

The group member list updates to reflect the updated group membership.

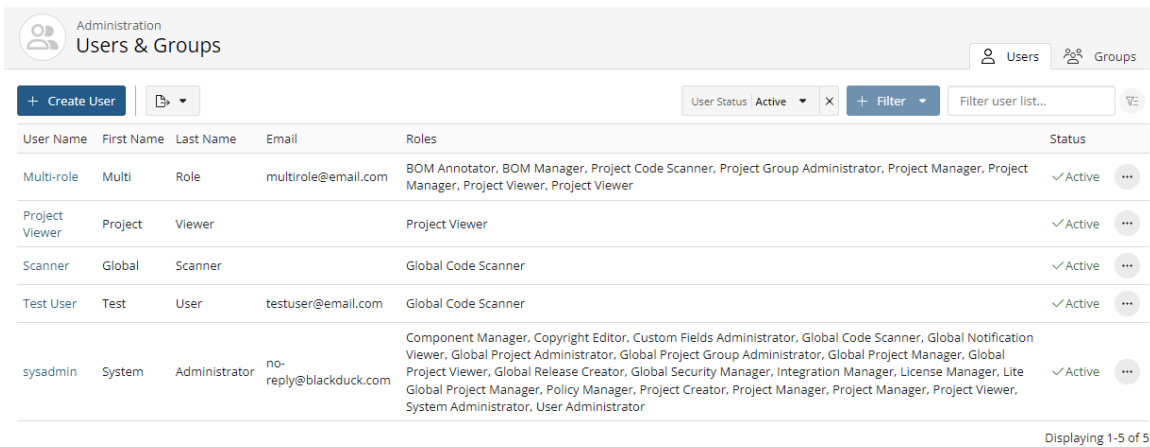
To remove a member from a group while managing a user:

1. Log in to Black Duck.
- 2.




Click **Admin**.

3. Select **Users & Groups** to display the **User & Group Management** page.



User Name	First Name	Last Name	Email	Roles	Status
Multi-role	Multi	Role	multirole@email.com	BOM Annotator, BOM Manager, Project Code Scanner, Project Group Administrator, Project Manager, Project Manager, Project Viewer, Project Viewer	✓ Active
Project Viewer	Project	Viewer		Project Viewer	✓ Active
Scanner	Global	Scanner		Global Code Scanner	✓ Active
Test User	Test	User	testuser@email.com	Global Code Scanner	✓ Active
sysadmin	System	Administrator	no-reply@blackduck.com	Component Manager, Copyright Editor, Custom Fields Administrator, Global Code Scanner, Global Notification Viewer, Global Project Administrator, Global Project Group Administrator, Global Project Manager, Global Project Viewer, Global Release Creator, Global Security Manager, Integration Manager, License Manager, Lite Global Project Manager, Policy Manager, Project Creator, Project Manager, Project Manager, Project Viewer, System Administrator, User Administrator	✓ Active

Displaying 1-5 of 5

4. Find the user you want to find:
 - Select the **Display Inactive Users** check box to include inactive users. Clearing this check box hides all inactive users.
 - Filter the users that appear on the page.
 - Sort the list of users by selecting any of the column names. An arrow next to the column name indicates the direction the list is sorted.
 - Use the pagination bar at the bottom of the list to go to the appropriate page if there are more users than are listed on this page.
5. Select the user to display the *Username* page.
6. Find the group you want to remove for this user in the **User Groups** section and click .
7. In the Remove User from Group dialog box, click **Remove**.
The group list updates to reflect the updated group membership.

Deleting groups

You do not need to remove members from a user group to delete it. When you delete the group, the group membership and permissions are removed from the user's records.

To delete a user group:

1. Log in to Black Duck.

15. Managing groups in Black Duck • Deleting groups

2.



Click **Admin**.


3. Select **Users & Groups** to display the **User & Group Management** page.

4. Click the **Groups** tab.



5. Find the group you want to delete:

- Select the **Display Inactive Groups** check box to include inactive groups. Clearing this check box hides all inactive groups.
- Filter the groups that appear on the page.
- Sort the list of group names by selecting the column. An arrow next to the column name indicates the direction the list is sorted.
- Use the pagination bar at the bottom of the list to go to the appropriate page if there are more groups than are listed on this page.

6. Click  in the row of the group that you want to delete.

7. In the Delete Group dialog box, click **Delete**.

The group is deleted from Black Duck. Users who were assigned to the deleted group no longer have any overall roles that were associated with belonging to that group and no longer have membership on project teams granted through that group.

16. About Project Groups

Black Duck provides the ability to logically group all your projects in the Hub, allowing you to organize which projects belong to which business unit making it easier for you to view risk across the organization. Project groups can contain both projects and other project groups to provide a multi-level hierarchy.

To view the Project Group Management page:

1. Log in to Black Duck.
- 2.



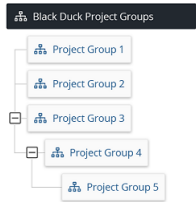
Click **Manage** and select **Project Groups**.

From the Project Group Management page, you will see the root project group for all projects and groups.

A screenshot of the 'Project Group Management' page. The page title is 'Project Group Management'. On the left, there is a tree view showing a hierarchy of project groups: 'Black Duck Project Groups' (root), 'Project Group 1', 'Project Group 2', 'Project Group 3', 'Project Group 4', and 'Project Group 5'. The main content area is titled 'Black Duck Project Groups' and contains a table of groups and projects. The table has columns for 'Description', 'Content', 'Created', and 'Updated'. The 'Content' column shows '3 Subgroups | 7 Projects'. There are controls for '+ Add Group', 'Move', 'Show All Descendants', 'Sort by...', and 'Filter results...'. Below the table, there is a 'Select All Projects' checkbox. The table lists several project groups and projects, including 'Project Group 1', 'Project Group 2', 'Project Group 3', and various 'Monkey' and 'apache-cxf' projects. The 'Created Date' for all items is '8/23/2021'. At the bottom right, it says 'Displaying 1-10 of 10'.

Project Group Hierarchy

The top level of the hierarchy, or the root level, is the main level for all subsequent projects and project groups for your organization. By default, it is named "Black Duck Project Groups", but can be changed at any time.



Ancestor, Parent, and Child Projects and Project Groups

In the example above, you can see a number of levels which may act as individual projects or other project groups. Projects 1, 2, and 3 are considered children of the root level project group. The same can be said for Project 4 and Project 5 in relation to Project 3. Project 5 is also a child of Project 4.

The inverse relationship is called a parent. For example, Project 4's parent is Project 3 and Project 5's parent is Project 4.

An ancestor is any project group existing above the parent for that project group. Project 3 is an ancestor of Project 5.

Members and User Groups

The relationship between projects and project groups matters when assigning members and user groups to project groups. Members and user groups can be assigned to project groups with any number of roles. That assignment will give those users access to the project or project group they are directly assigned to (Direct Access), and to all child projects and project groups of that group with the specified roles unless that assignment is explicitly overridden at the lower levels (Indirect Access). This concept allows for setting users with default access to projects that haven't been created yet. For more details regarding user roles, see [Understanding roles](#).

Black Duck Projects ▶ Project 3

Manage

Members

+ Add Member

User Name	Direct Access	Indirect Access	Status:	
Bom	Yes	No	✓ Active	...
Copyright	Yes	No	✓ Active	...
sysadmin	No	Yes	✓ Active	...

Displaying 1-3 of 3

In the example above, users Bom and Copyright have been added as members with Direct Access. The sysadmin user is a member of the root level project group, therefore has Indirect Access to all child projects and project groups, including Project 3.

Black Duck Projects ▶ Project 3 ▶ Project 4

Manage

Members

+ Add Member


User Name	Direct Access	Indirect Access	Status:	
Bom	No	Yes	✓ Active	...
Copyright	No	Yes	✓ Active	...
sysadmin	No	Yes	✓ Active	...

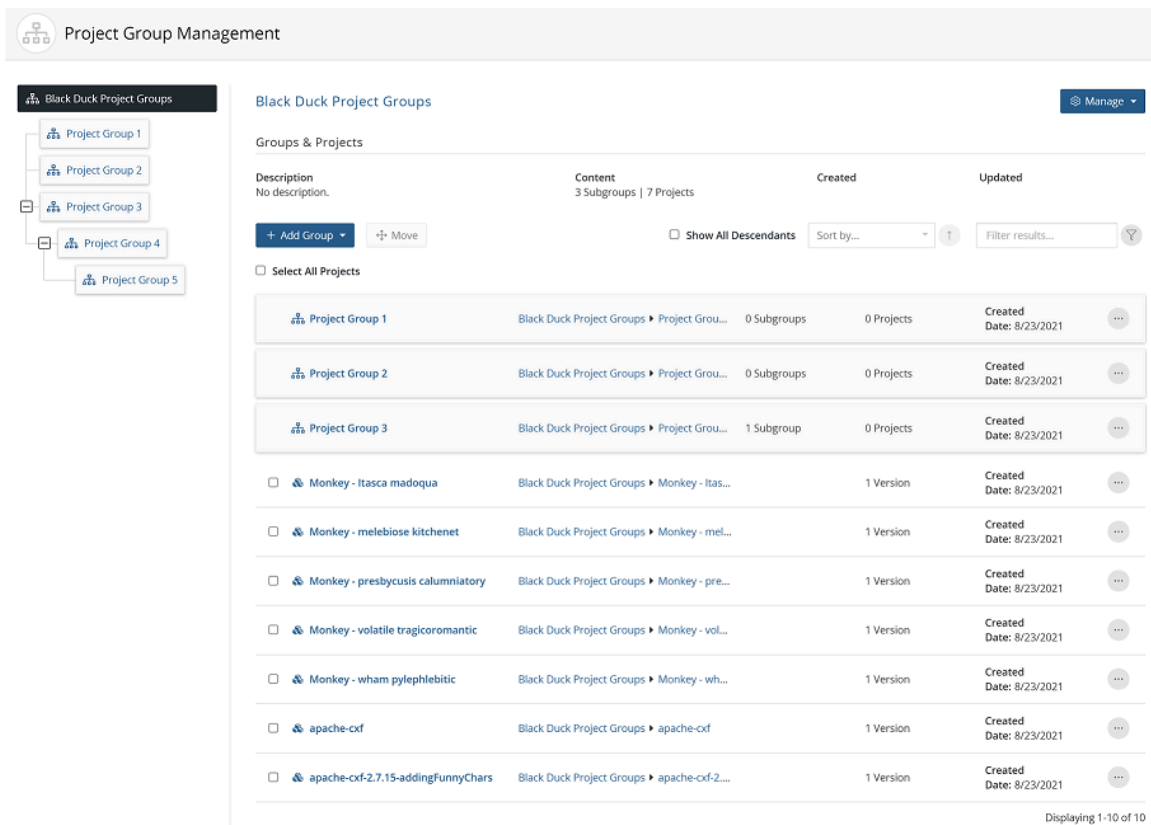
Displaying 1-3 of 3

As a result, users Bom and Copyright now have Indirect Access to Project 4 as seen above. This extends to all children of Project 3. They will hold the same role for all child projects and project groups of Project 3.

Creating a Project Group

You can create a project group by following these steps:

1. Log in to Black Duck.
2. Click .
3. Select **Project Groups** to display the Project Group Management page.



Project Group Management

Black Duck Project Groups


Black Duck Project Groups

Groups & Projects

Description	Content	Created	Updated
No description.	3 Subgroups 7 Projects		
<input type="button" value="+ Add Group"/> <input type="button" value="Move"/> <input type="checkbox"/> Show All Descendants <input type="text" value="Sort by..."/> <input type="text" value="Filter results..."/>			
<input type="checkbox"/> Select All Projects			
Project Group 1	Black Duck Project Groups ▶ Project Grou...	0 Subgroups 0 Projects	Created Date: 8/23/2021
Project Group 2	Black Duck Project Groups ▶ Project Grou...	0 Subgroups 0 Projects	Created Date: 8/23/2021
Project Group 3	Black Duck Project Groups ▶ Project Grou...	1 Subgroup 0 Projects	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - Itasca madoqua	Black Duck Project Groups ▶ Monkey - Itas...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - melebiose kitchenet	Black Duck Project Groups ▶ Monkey - mel...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - presbycusis calumliatory	Black Duck Project Groups ▶ Monkey - pre...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - volatile tragicoromantic	Black Duck Project Groups ▶ Monkey - vol...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - wham pylephlebitic	Black Duck Project Groups ▶ Monkey - wh...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> apache-cxf	Black Duck Project Groups ▶ apache-cxf	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> apache-cxf-2.7.15-addingFunnyChars	Black Duck Project Groups ▶ apache-cxf-2...	1 Version	Created Date: 8/23/2021

Displaying 1-10 of 10

4. Click  and select **Groups and Projects** from the dropdown menu.

5. Click  and select **Create New...** from the dropdown menu.
6. In the Create New Project Group dialog box:
 - a. Type the name of the group in the **Group Name** field. This field is mandatory.
 - b. Type a description for the Project Group in the Description field. This field is optional.
 - c. Click **Save**. The Project Group Management page updates to display the new group.


You can now:

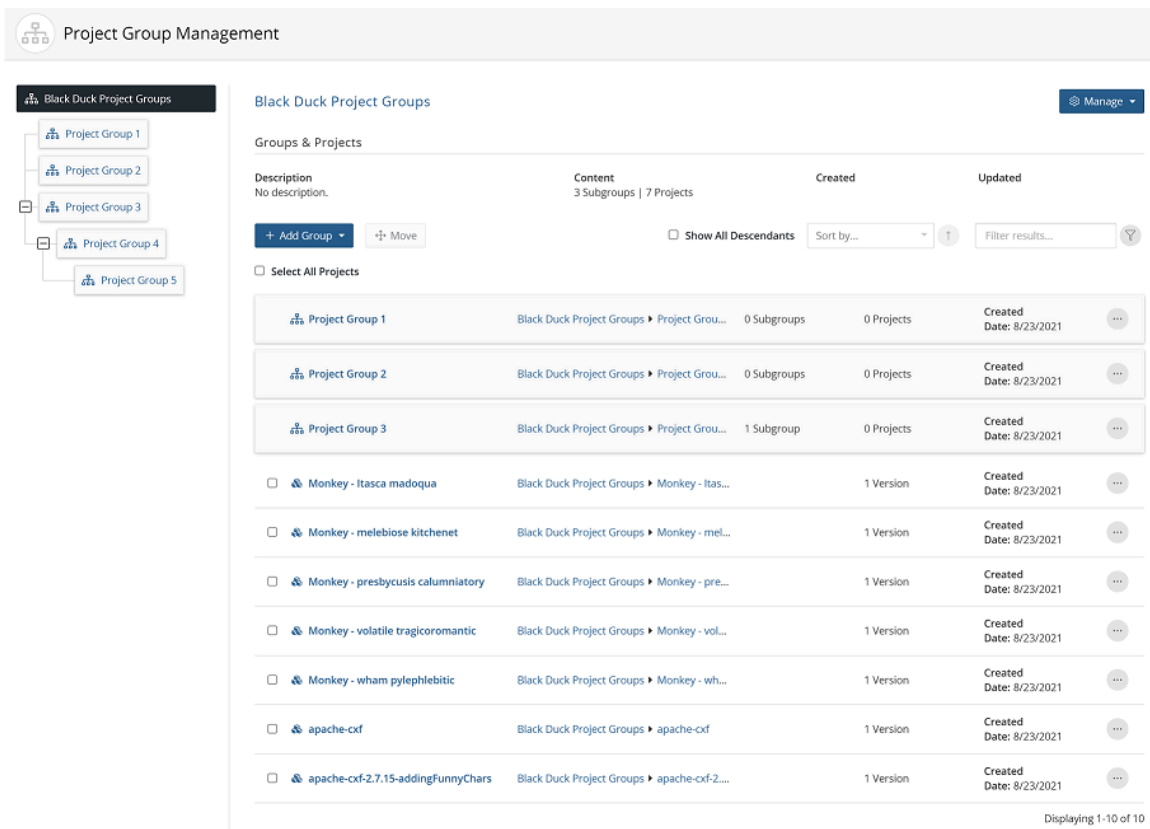
- [Add members and user groups](#) to the project group.
- [Validate the generation of SBOM reports against policies](#).

Editing a Project Group

Once you have created a project group, you can add project and project group children, individual members, and/or user groups. You can also change the project group's name or description as well as set the option to validate the generation of SBOM reports against policies for projects belonging to specific project groups.

To do so, follow the steps listed below:

1. Log in to Black Duck.
2. Click .
3. Select **Project Groups** to display the Project Group Management page.



Project Group Management

Black Duck Project Groups Manage

Black Duck Project Groups

Groups & Projects

Description: No description. Content: 3 Subgroups | 7 Projects. Created: Created. Updated: Updated.

+ Add Group Show All Descendants Sort by: [dropdown] 1 Filter results: [input]


Select All Projects

Description	Content	Created	Updated
Project Group 1	Black Duck Project Groups ▶ Project Grou...	0 Subgroups 0 Projects	Created Date: 8/23/2021
Project Group 2	Black Duck Project Groups ▶ Project Grou...	0 Subgroups 0 Projects	Created Date: 8/23/2021
Project Group 3	Black Duck Project Groups ▶ Project Grou...	1 Subgroup 0 Projects	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - Itasca madoqua	Black Duck Project Groups ▶ Monkey - Itas...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - melebiose kitchenet	Black Duck Project Groups ▶ Monkey - mel...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - presbycusis calumniatory	Black Duck Project Groups ▶ Monkey - pre...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - volatile tragicoromantic	Black Duck Project Groups ▶ Monkey - vol...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> Monkey - wham pylephlebitic	Black Duck Project Groups ▶ Monkey - wh...	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> apache-cxf	Black Duck Project Groups ▶ apache-cxf	1 Version	Created Date: 8/23/2021
<input type="checkbox"/> apache-cxf-2.7.15-addingFunnyChars	Black Duck Project Groups ▶ apache-cxf-2...	1 Version	Created Date: 8/23/2021



Displaying 1-10 of 10

Editing the name or description


By default the root level project group is called "Black Duck Project Groups" but it can be renamed.

1. Click  and select **Settings** from the dropdown menu.
2. Edit the name of the project group in the **Group Name** field. This field is mandatory.
3. Edit the description for the project group in the **Description** field. This field is optional.
4. Click **Save**. The Project Group Management page updates to display the new group.


Adding project sub-groups

1. Click  and select **Groups and Projects** from the dropdown menu.
2. Click  and select **Create New...** from the dropdown menu.
3. In the Create New Project Group dialog box:
 - a. Type the name of the project group in the **Group Name** field. This field is mandatory.
 - b. Type a description for the project group in the **Description** field. This field is optional.
 - c. Click **Save**. The Project Group Management page updates to display the new group.


Removing project sub-groups

1. Select the desired project group from the project group tree in the left-hand panel. This displays all child project groups in the right-hand panel.
2. Click .
3. Select **Delete** from the dropdown menu.
4. Click **Delete** from the confirmation dialog box.


If the project group is a child of a parent group:

1. Select the parent of the desired project group from the project group tree in the left-hand panel. This displays all project sub-groups for that project group in the right-hand panel.
2. Click .
3. Select **Delete** from the dropdown menu.
4. Click **Delete** from the confirmation dialog box.



Moving a project group to a different project group

1. Select the parent of the desired project group from the project group tree in the left-hand panel. This displays all child project groups for that project group in the right-hand panel.
2. Click .
3. Select **Move**
4. Select a project group game from the Group Name dropdown menu presented in the **Move Selected Group to...** dialog box.
5. Click **Save** to confirm the move.



Moving another project group into the selected group

1. Select the project group from the project group tree in the left-hand panel. This will display the details for the project group itself.
2. Click .
3. Select **Move existing...**
4. Select a project group game from the Group Name dropdown menu presented in the **Move Selected Group to...** dialog box. Please note, a project group cannot be moved into the selected project group if it is an ancestor of the selected project group.
5. Click **Save** to confirm the move.



Adding a member to a project group

1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **Members** from the dropdown menu.
3. Click .
4. Type or select a user name from the Users dropdown menu to open a list of members.
5. Select any role(s) that user will have for that project group. For more details regarding user roles, see [Understanding roles](#).
6. Click **Save**.



Removing a member from a project group

1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **Members** from the dropdown menu.
3. Click .
4. Select **Delete Direct Access**.
5. Click **Delete** from the confirmation dialog box.

Editing a member's roles in a project group



1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **Members** from the dropdown menu.
3. Click .
4. Select **Edit Direct Access**.
5. Add or remove any role(s) that user will have for that project group. For more details regarding user roles, see [Understanding roles](#).
6. Click **Save**.

Adding a user group to a project group

1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **User Groups** from the dropdown menu.
3. Click .


4. Type or select a user name from the User Group dropdown menu to open a list of user groups.
5. Select any role(s) that user group will have for that project group. For more details regarding user roles, see [Understanding roles](#).
6. Click **Save**.

Removing a user group from a project group

1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **User Groups** from the dropdown menu.
3. Click .
4. Select **Delete Direct Access**.
5. Click **Delete** from the confirmation dialog box.

Enabling or disabling SBOM report validation

When setting is enabled, the ability to [generate SBOM reports](#) will be disabled if the project has policy violations.

1. Select the desired project group from the project group tree in the left-hand panel.
2. Click  and select **Settings** from the dropdown menu.
3. Scroll to the **Reports** section.
4. Check or uncheck the **Don't generate SBOM reports for projects with policy violations** checkbox.
5. Select either:
 - **Apply setting to all projects in this group only:** Selecting this option will make it so that only this specific project group will have validation enabled when generating a SBOM report.
 - **Apply setting to all projects in this group and child groups:** Selecting this option will make it so this project group and all its child groups will have validation enabled generating a SBOM report.

18. Other administrative tasks

This chapter describes:

- [Viewing project and project version audit information](#)
- [Viewing product registration information.](#)
- [Managing code size limits.](#)
- [Managing user access tokens.](#)
- [Customizing the logo.](#)
- [Viewing jobs.](#)
- [Accessing log files.](#)

Creating system announcements

System Administrators can create custom announcements or messages to be displayed in a variety of locations in Black Duck.

For example, use system announcements to tell your users about upcoming events or if you need to show a disclaimer indicating what happens for unauthorized use.

There are four types of messages that you can create:

- **Login.** A message that appears to the user when they are logging in to Black Duck.
Since this message appears for all users – including unauthenticated users – Black Duck recommends that you do not use this type of system announcement to display sensitive information.
- **Banner.** A message that appears at the top of every page.
- **Footer.** A message that appears in the footer of every page.

Open Source Policy

To read the Synopsys World Wide Open Source Policy, [click here](#).

BLACKDUCK v2020.8.0 | Notices

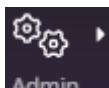
- **Welcome.** A message that appears after the user logs in to Black Duck.
Unlike other announcements, you can provide an option so that users can suppress this message: users will not see this message again unless you edit the message.

Note that you can only create one announcement of each type.

To create a system announcement:

1. Log in to Black Duck with the System Administrator role.

2.



Click **Admin**.

3. Select **Announcements** to display the **Announcements** page.

4. Select the type of announcement.
5. Enter a title for this announcement.
6. Enter the announcement text. If not selected, click **Edit** and enter the text. Click **Preview** to view your announcement as it will appear to your users.

You must use markdown language when creating the announcement. See the next section for more information.

7. For the Welcome announcement, select whether the user can suppress the announcement.

If you select to suppress the announcement, the user will see the following option ("Don't show this again") in the announcement. However, the announcement will reappear to the user if you make any changes to the announcement.
8. Optionally, enter the start and end date for this announcement. By default the start date is today, with no end date, indicating a persistent announcement. Dates are inclusive: the announcement will appear for the date range you select here, including the start and end dates.
9. Select whether to enable the message. Once enabled, the announcement will appear to users once you click **Save**.
10. Click **Save**.

Markdown language

You must use markdown language when creating the announcement.

Click [here](#) for more information on the syntax for markdown language.

The following is the list of allowable tags for system announcements:

- h1, h2, h3, h4, h5, h6

Note that h1 and h2 tags are converted to h3 tags.

- blockquote
- p
- a
- ul

- ol
- nl
- li
- b
- i
- strong
- em
- strike
- abbr
- code
- hr
- b
- table
- thead
- caption
- tbody
- tr
- th
- td
- pre
- iframe
- Anchor tags <a> are only allowed with the following attributes:
 - href
 - name
 - target

Note that images are not allowed.

Viewing project and project version audit information

Black Duck tracks and displays all updates and changes that affect a project and/or project version. Use this information to understand who made changes or the events that caused changes to a project or project version. With this audit trail, you can determine, for example:

- who made changes to the BOM, such as who reviewed a component, added a comment, or ignored a component
- what changes occurred due to a scan, such as what components were added or deleted and what changes occurred due to those components (for example, the vulnerabilities that were added)
- who created or deleted a project version
- when was a policy violation triggered or when was a component no longer in violation,

- when was a policy violation overridden or when was the override reversed
- when did a component in your BOM introduce a new vulnerability
- when was remediation information updated for a vulnerability on a component in your project
- when did someone add or remove users from a project
- when was a snippet match confirmed or ignored

Black Duck provides the following information:

- The object that affected the project or project version, such as a component, vulnerability, or scan
- The type of event, such as vulnerability was found or a component was edited
- Who caused the event in the format User: *username*. If the Black Duck system caused the event (for example components or vulnerabilities found during a scan or an update to Black Duck KnowledgeBase that changed a vulnerability), the column shows User: *blackduck_system*.
- Date and time this event occurred.

The following is an example of a new project and project version created during a scan:

The screenshot shows the Black Duck Projects interface. At the top, there is a header for 'Black Duck Projects 1.0' with a 'Project' tab and 'Versions: 1'. On the right, there are 'Overview' and 'Settings' buttons. Below the header is a sidebar with navigation options: 'Project Details', 'Members', 'Groups', and 'Activity' (which is selected and highlighted in blue). The main content area displays a table of activity events.

Object	Event	Cause	Date and Time
> Project: 1.0	Project Created	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> User: sysadmin	User Role Added	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> User: sysadmin	User Role Added	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> Project Version: Sample Audit Project	Version Created	User: sysadmin	Mon, Mar 4, 2019 12:29 PM

At the bottom right of the table, it says 'Displaying 1-4 of 4'.

Note the following:

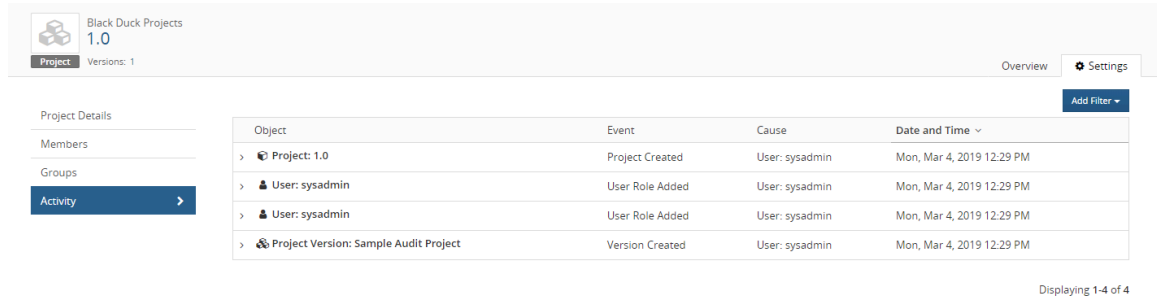
- Information is shown for the past 24 hours with the most recent changes appearing at the top of the table. Use the date filter to view information for different periods of time.
- While the deletion of a project version appears at the project level, deletion of a project will not appear here.

To view audit information:

Audit information appears on the **Settings** tab of the project or project version.

1. Log in to Black Duck.
2. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
3. Do one of the following:
 - To view *project* level audit information, select the **Settings** tab and then select **Activity**.

18. Other administrative tasks • Viewing product registration information



Black Duck Projects
1.0
Project Versions: 1

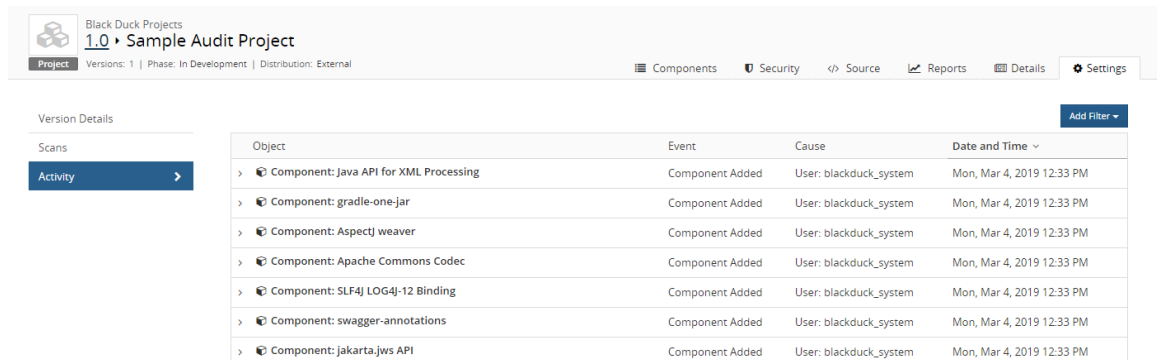
Overview Settings Add Filter

Project Details
Members
Groups
Activity

Object	Event	Cause	Date and Time
> Project: 1.0	Project Created	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> User: sysadmin	User Role Added	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> User: sysadmin	User Role Added	User: sysadmin	Mon, Mar 4, 2019 12:29 PM
> Project Version: Sample Audit Project	Version Created	User: sysadmin	Mon, Mar 4, 2019 12:29 PM

Displaying 1-4 of 4

- To view *project version* level audit information, select the version, select the **Settings** tab, and then select **Activity**.



Black Duck Projects
1.0 • Sample Audit Project
Project Versions: 1 | Phase: In Development | Distribution: External

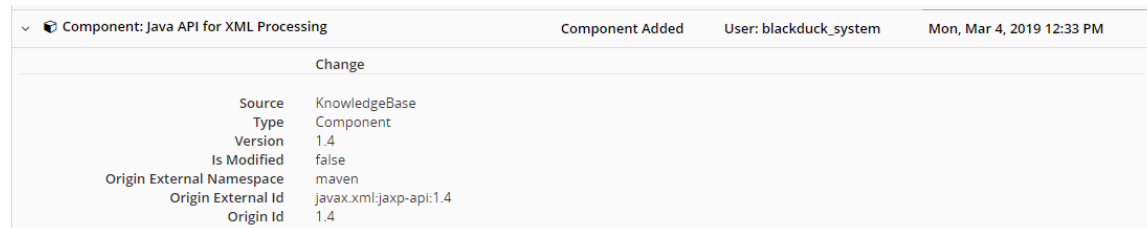
Components Security Source Reports Details Settings Add Filter

Version Details
Scans
Activity

Object	Event	Cause	Date and Time
> Component: Java API for XML Processing	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: gradle-one-jar	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: Aspectj weaver	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: Apache Commons Codec	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: SLF4J LOG4J-12 Binding	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: swagger-annotations	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM
> Component: jakarta.jws API	Component Added	User: blackduck_system	Mon, Mar 4, 2019 12:33 PM

4. From this page:

- Click > located to the left of the object name to view details of this event.



Component: Java API for XML Processing Component Added User: blackduck_system Mon, Mar 4, 2019 12:33 PM

Change	
Source	KnowledgeBase
Type	Component
Version	1.4
Is Modified	false
Origin External Namespace	maven
Origin External Id	javax.xml:jaxp-api:1.4
Origin Id	1.4

- Filter the table to view specific information, such as activity during a specific date range or a specific type of event.

SBOM fields

If [SBOM fields are enabled](#), adding or modifying these fields on the project version or component level will be logged in the Activity tab as well. Click [here](#) for more information on how to edit SBOM fields.

Viewing product registration information

The Product Registration page lists:

Product registration

- Your registration ID

- Status and expiration date and time

Registration features

- Number of users
- Number of projects
- Number of project versions
- Number of codebase KBs/MBs/GBs
- Number of scans
- Maximum scan size

Licensed modules

- **Full Snippet Scanning**

Full snippet scanning enables the scan cli option for `--full-snippet-scan` which performs snippet scanning on all files regardless of any signature matches including all files even if unmodified from last scan. This option has a significant performance and scalability impact and should only be used in extreme situations.

- **Artifactory Integration**

Artifactory Integration is a service that enables scanning of artifacts within a set of configured repositories to identify open source components, using data gathered by Component Scanning.

- **Black Duck Binary Analysis**

Black Duck Binary Analysis (BDBA) is a service that provides enhanced interrogation of binaries to surface the open source components within open source. It also supports expanded file type support including various firmware formats, filesystem/disk images, installation formats and various compression and archive formats.

- **Black Duck Secure Container (BDSC)**

Black Duck Secure Container scanning provides capabilities to identify components within container images, their layers and base images.

- **Black Duck Security Advisory**

The Black Duck Security Advisory (BDSA) is a Black Duck-exclusive vulnerability data feed, sourced and curated by our Security Research team which is part of the Black Duck COSRI (Centre of Open Source Research & Innovation). A BDSA offers deeper coverage for a wider set of vulnerabilities than is available through the NVD (National Vulnerability Database), providing detailed vulnerability insight including severity, impact, exploitability metrics and actionable remediation guidance. The BDSA data for new vulnerabilities is reported an average of three weeks earlier than the NVD's reported data.

- **Component Scanning**

Component Scanning automates the discovery and identification of OSS components in a scan to provide metadata such as license type, security vulnerabilities, and OSS project health for those components. Component scans can be linked to internal project versions to automatically generate BOMs.

- **Cryptography**

Cryptography Management enables the display of cryptographic algorithms, and additional metadata, that are contained in open source components. This data is used to support compliance to security standards and legal export regulations.

- **License Management**

License Management is the feature which allows Black Duck users to edit and maintain the data which can be used to create accurate and compliant open source notice files/reports at a Project/Release level.

- **Match as a Service**

Match as a Service (MaaS) is a service that identifies OSS components, using data gathered by Component Scanning.

- **Notifications**

Black Duck notifications alert your teams when vulnerabilities change or there are policy violations. Your organization can integrate with other platforms using the Notification API.

- **OSS Notices Report**

The Notices Reports feature allows users to create notices (or attribution) reports for their projects. The notice files can then be included with the distribution or incorporated into documentation to satisfy the attribution obligation that exists in the vast majority of open source licenses.

- **Policy Management**

The Policy Management feature enables companies to define rules to govern their use of open source components. With these rules, open source usage can be managed on an exception basis, i.e, as long as open source components meet the policy requirements their usage is allowed, thereby speeding time to market and freeing developers from a cumbersome approval process. Any open source components/versions that fail to meet policy are flagged, enabling a review process to determine if the use of the component should be allowed in the particular application.

- **ReversingLabs**

Using complex binary analysis powered by ReversingLabs, developers and DevOps teams can analyze first party, open source, and commercial software to identify the presence of threats such as malware, maldocs, suspicious files, potentially unwanted applications (PUAs), protestware, and suspicious file structure malformations to help avoid dangerous software supply chain attacks.

- **Risk Management**

Risk Management enables the identification, notification, and remediation of the security, license, and operational risks associated with the OSS components used in your internal projects.

- **SBOM Custom Component Auto-Creation**

SBOM Custom Component Auto-Creation enables custom components to be created automatically if they do not match to an existing KB or custom component upon SBOM import. A Package URL (PURL) must be supplied for the component in the SBOM in order for the custom component to be automatically created.

- **SCM Integration**


SCM Integration enables the configuration and authentication of Source Code Management (SCM) providers. It allows mapping, scanning, and management of SCM repositories in projects. The SCM integration feature must be enabled on your registration key for this to appear on the Product Registration list.

- **Snippets**

This feature enables the option to invoke optimized scans of source files which find OSS usage at the file or code snippet level. These scans work in conjunction with a component scan to produce the best possible Bill of Materials (BOM) for a project. Once discovered, matches can be reviewed and added to a project BOM which pulls in the associated metadata for the component.

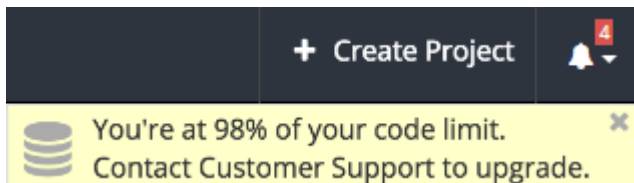
Updating your product registration

Your Black Duck license may restrict the number of users, projects, and/or project versions. If you need more capacity, you can purchase a new license. Once you receive a new license from Black Duck Software, enter the new registration ID information in Black Duck to activate your newly-licensed capacity.

1. Log in to Black Duck as a system administrator.
2.  Click **Admin**.
3. Select **Product Registration** to open the Product Registration page.
4. Type your new registration key in the **Registration ID** field. Be sure that you accept the terms of the End User License Agreement.
5. Click **Save**.

Managing your code size limits

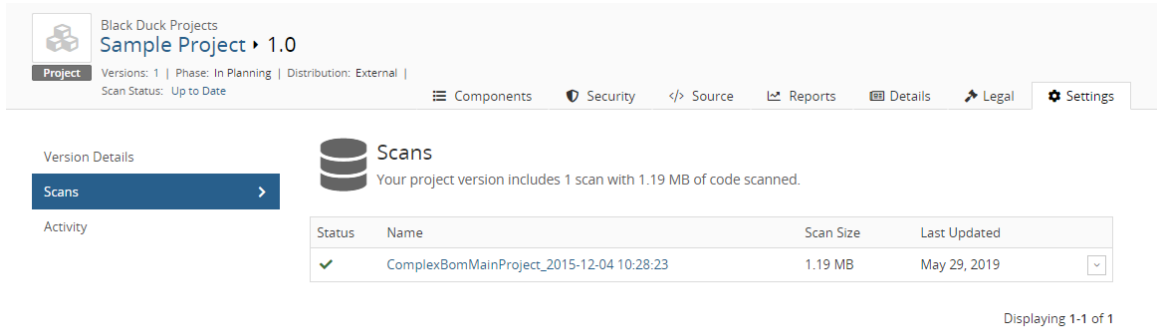
Black Duck will notify you when you are approaching your code size limit (as declared in your license). A notification, such as the following, appears in the UI when you are at 80% or higher of your code size limit:



If you exceed your code size limit, an error message appears when trying to scan (for example, shown in log files in Jenkins or on the screen in Black Duck Detect (Desktop)) or when uploading scans to Black Duck. You will not be able to scan or upload scans if you exceed your code size limit.

When receiving this notification, you can:


- Contact Customer Support to upgrade your service.
- View the scan size for a project version:
 1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
 2. Select the version name which displays the **Components** tab.
 3. Select the **Settings** tab.
 4. Select **Scans** to view the scans mapped to this project version.

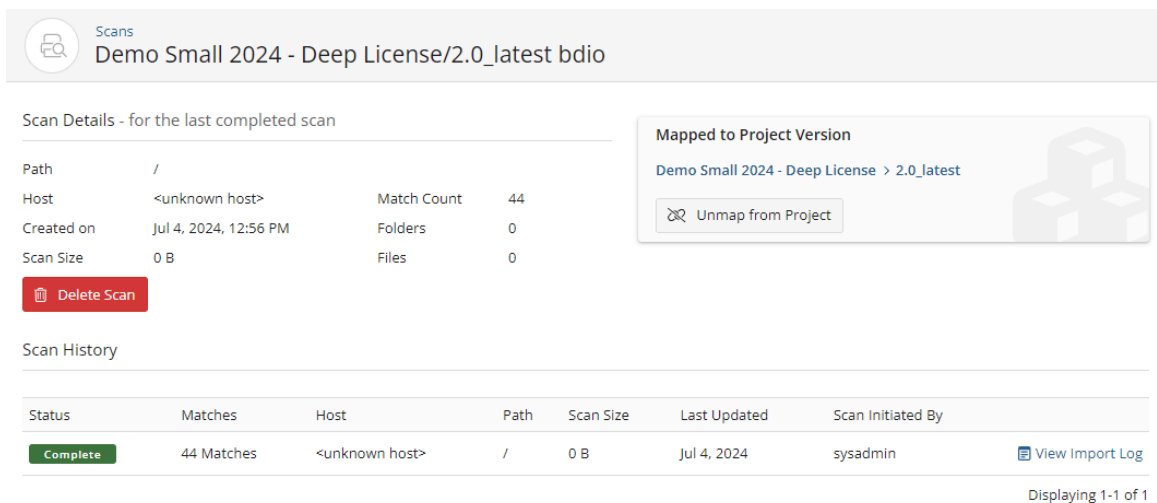


The scan size appears above the list of scans.


- [Delete existing scans](#) to free up space.

To determine the size of a scan:

1.  Click **Scans** to display the Scans page.
2. Select the path of the scan that you want to view the results to open the *Scan Name* page.



The **Scan Details** sections lists the scan size.

-  **Note:** You can view your current usage versus your limit on the Scans page. Values appear in the upper right corner of the page.

Managing user access tokens

Black Duck provides the ability for you to generate one or more “tokens” for accessing Black Duck APIs. These tokens are intended to replace the use of username/password credentials in integration configurations, such as Jenkins or for the Scan Client CLI. With access tokens, if a security breach occurs, the user’s credentials (which might be their SSO or LDAP credentials) are not directly compromised.

Note the following:

- Access tokens can only be created by the current user.

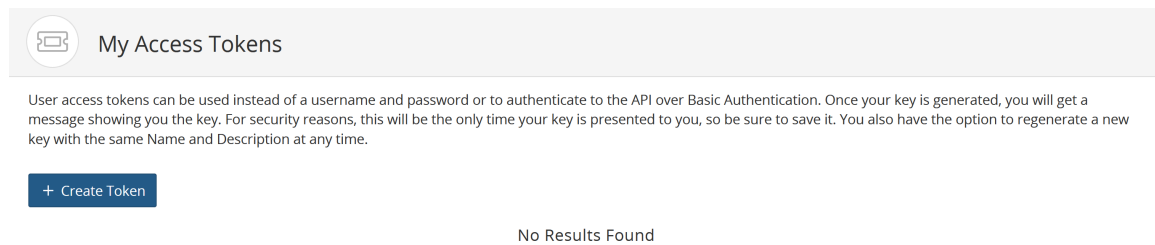
- Access tokens are tied to a user's account; therefore, an access token has the same role as the user who created the token.
- A user can have multiple tokens. Each token must have a unique name.
- Access tokens do not expire, but can be [purged after a set period of inactivity](#).
- If a user is inactivated, their tokens are invalidated.

Refer to the *Getting Started with the SDK* guide for information on using the API keys.

To generate an access token:

1. Log into Black Duck.
2. From the user menu located on the top navigation bar, select **My Access Tokens**.

The My Access Tokens page appears.



3. Click **Create New Token**. The Create New Token dialog box appears.

4. Enter a name, description (optional), and select the scope for this token (read or read and write access). You can only select one access for a token.
5. Click **Create**.

The *Access Token Name* dialog box appears with the access token.

18. Other administrative tasks • Managing user access tokens

- Copy the access token shown in the dialog box. This token can only be viewed here at this time. Once you close the dialog box, you cannot view the value of this token.
- Click **Close**.

To edit an access token:

You can edit the name and description of an access token. You cannot edit the scope (read and/or write access) of a token.

- Log into Black Duck.
- From the user menu located on the top navigation bar, select **My Access Tokens**.

The My Access Tokens page appears.


My Access Tokens

User access tokens can be used instead of a username and password or to authenticate to the API over Basic Authentication. Once your key is generated, you will get a message showing you the key. For security reasons, this will be the only time your key is presented to you, so be sure to save it. You also have the option to regenerate a new key with the same Name and Description at any time.

+ Create Token

Name	Description	Scope
Read and write	Read and write access	write, read

Displaying 1-1 of 1

- Click  in the row of the token you want to revise and select **Edit**.

The Edit User Access Token dialog box appears.

- Edit the name or description and click **Update**.

To regenerate an access token:

You can regenerate a new access token which provides a different key for the same name, description, and access.

- Log into Black Duck.
- From the user menu located on the top navigation bar, select **My Access Tokens**.

The My Access Tokens page appears.


My Access Tokens

User access tokens can be used instead of a username and password or to authenticate to the API over Basic Authentication. Once your key is generated, you will get a message showing you the key. For security reasons, this will be the only time your key is presented to you, so be sure to save it. You also have the option to regenerate a new key with the same Name and Description at any time.

+ Create Token

Name	Description	Scope
Read and write	Read and write access	write, read

Displaying 1-1 of 1

- Click  in the row of the token you want to regenerate and select **Regenerate**.

The Regenerate User Access Token dialog box appears.

- Click **Regenerate** to confirm.

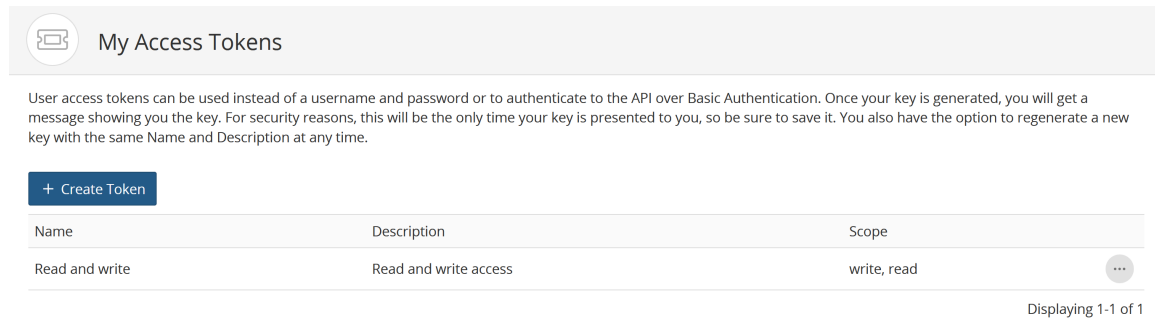
The *Access Token Name* dialog box appears with the new access token.

- Copy the access token shown in the dialog box. This token can only be viewed here at this time. Once you close the dialog box, you cannot view the value of this token.
- Click **Close**.

To delete an access token:

- Log into Black Duck.
- From the user menu located on the top navigation bar, select **My Access Tokens**.

The My Access Tokens page appears.




My Access Tokens

User access tokens can be used instead of a username and password or to authenticate to the API over Basic Authentication. Once your key is generated, you will get a message showing you the key. For security reasons, this will be the only time your key is presented to you, so be sure to save it. You also have the option to regenerate a new key with the same Name and Description at any time.

+ Create Token

Name	Description	Scope
Read and write	Read and write access	write, read


Displaying 1-1 of 1

- Click  in the row of the token you want to remove and select **Delete**.
The Delete User Access Token dialog box appears.
- Click **Delete** to confirm.

Enabling license term fulfillment

BOM Managers, and other users with the appropriate [role](#), manage the fulfillment status for a license term using the **Term Fulfillment** tab in the *Project Version's* **Legal** tab.

By default, these tabs are disabled. System Administrators must enable these tabs for them to appear to these users.

 **Note:** Enabling the **Legal** tab is a global setting. Once enabled, all project versions will display the **Term Fulfillment** tab in the **Legal** tab.

To display the **Legal** and **Term Fulfillment** tabs:

- Log into Black Duck with the System Administrator role.

-  Click **Admin**.

- Select **System Settings**.
- Click **Legal**.
- Set the toggle located in the **Term Fulfillment** section to *Setting is enabled* to display the **Legal** and **Term Fulfillment** tabs. Enabling the setting will not take effect immediately for existing projects.

Terms Fulfillment

Enables Term Fulfillment on the Legal Tab in Project Versions so project users can check off license terms as fulfilled as part of their workflow. Note: this also requires license administrators to indicate which terms require fulfillment. See the documentation on license terms for more information.

Setting is enabled

Set the toggle located in the **Term Fulfillment** section to *Setting is disabled* to remove the **Legal** and **Term Fulfillment** tabs. Note that if you select to enable license conflicts, the **Legal** tab will still appear.

Terms Fulfillment

Enables Term Fulfillment on the Legal Tab in Project Versions so project users can check off license terms as fulfilled as part of their workflow. Note: this also requires license administrators to indicate which terms require fulfillment. See the documentation on license terms for more information.

Setting is disabled


Customizing the logo

You can replace the logo that appears in the header of the user interface:




The maximum height for a logo is 37px and width of 259px.

To change the logo:

1. Log in to Black Duck with the System Administrator role.
2. 
Click **Admin**.
3. Select **System Settings**.
4. Click **Branding**.
5. Click **Upload New Logo** and select the file.

The new logo appears in the header.

 **Note:** To redisplay the Black Duck logo, select **Restore to original**. This link appears on the page after you customize the logo.

Project version auto-deletion

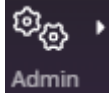
Project version auto-deletion allows you to explore ways to automatically delete project versions according defined criteria. For users with version limits, disk space constraints or database bottlenecks, the buildup of obsolete versions can become problematic to either their process or to their system performance. This feature is helpful if you generate multiple project versions over time which become obsolete over time.

How to enable or disable project version auto-deletion

To enable or disable project version auto-deletion:

1. Log in to Black Duck with the System Administrator role.

- 2.



Click **Admin**.

3. Select **System Settings**.
4. Click **Data Retention**.
5. Click **Project Version Auto-Deletion**.
6. Check or uncheck the **Enable Project Version Auto-Deletion** checkbox.

[Data Retention](#) ▶ Project Version Auto-Deletion

When enabled, project versions are scheduled to be deleted from your system. They need to meet the project version phase and inactivity time conditions, and will only be deleted after the grace period.

[? Learn more about automatic data removal](#)

Enable Project Version Auto-Deletion

Project Version Phases *

- | | | |
|--------------------------------------|--|--------------------------------------|
| <input type="checkbox"/> In Planning | <input checked="" type="checkbox"/> In Development | <input type="checkbox"/> Pre-Release |
| <input type="checkbox"/> Released | <input type="checkbox"/> Deprecated | <input type="checkbox"/> Archived |

Inactivity Period *

Days of inactivity before project versions are scheduled for deletion.

Grace Period *

Once a project version is scheduled for deletion, this is the number of days before it is deleted (as long as the conditions still apply).

Reset

Save

Enabling this feature will cause new data retention fields on the project version's settings page to be displayed.

Black Duck Projects
My Project ▾ Default Detect Version

Project ★ | Phase: In Development | Scans: Up to Date | Status: Up to Date

Components Security </> Source Reports Details Settings

Version Details >

Scans

Activity

Version • Default Detect Version

License Unknown License

Notes

Nickname

Release Date

Phase • In Development

Distribution • External

Data Retention Enabling this checkbox will prevent this Project Version from ever being deleted by the automated data deletion policies setup by your administrator.

Retain Project Version regardless of data retention policies

Save

Project version auto-deletion settings

Project Version Phase Condition: Defines what project version phases are applicable to the auto-deletion process. The default value is *In Development*. Multiple release phases can be selected.


Inactivity Period Condition: Defines the time period in which a version has to be inactive for it to get scheduled for deletion in the future. Inactivity is defined as no scans and no edits for the project version including project version settings, component edits, vulnerability remediations and policy overrides. The default value is 45 days.

Grace Period: Defines how far in the future project versions will be scheduled for deletion (i.e. how long the warning icon shows up in the UI before the project version is actually deleted). The default value is 45 days.

Criteria for project version auto-deletion

The following criteria is used to determine if a project version is a candidate for auto-deletion:

- The project version phase matches any of the release phase values set in **Project Version Phase Condition**.
- The project version is not a sub-project that is included as a component on other project version's BOM.
- The project version is not part of a project in which "Custom scan signatures" are enabled.
- The project version has been inactive during the timeframe specified in the environment variable.
- The project version is not protected from auto-deletion on the project version settings UI in the [Data Retention](#) section.

Any project versions which fall outside of these constraints will not be tagged for auto deletion. Once a project version is tagged for auto deletion, a  icon is displayed on the project's overview screen with the record highlighted in red indicating it is scheduled for deletion.

Black Duck Projects
My Project
Project ★ Watching Project | Versions: 1

Overview Settings

Description
No description.

Additional Fields
-

Created
Mar 10, 2022 by sysadmin

Updated
Mar 10, 2022 by sysadmin

Tags
No Tags

+ Create Version

Filter versions... + Add Filter

Version	Phase	Last Updated	Last Scanned	License	Security Risk	License Risk	Operational Risk
1.0	In Development	2:03 PM	8:38 AM	Unknown License	<div style="width: 10%;"></div>	<div style="width: 20%;"></div>	<div style="width: 30%;"></div>

Displaying 1-1 of 1

After a project has been flagged for deletion, if it subsequently has activity against it, it will be “unflagged” for deletion and the inactivity counter will start again.

The job which sets the autodeletion status and deletes the project versions is the [ScanPurgeJob](#) which runs every 15 minutes.

After project version auto-deletion


Once the elapsed grace period has passed, the project version will be deleted. This will orphan any scans for the project version as they will no longer be associated with a project version. Those scans will be deleted according to the “Cleaning up unmapped code locations” configuration as defined in the install guide (default = 30 days).

Please note, if all project versions for a project are deleted, then the project is also deleted.

Changing the data retention period for unmapped scans

You can change the period of time unmapped scans are retained. By default, the time frame is set to 30 days and can be set to as low as 1 day and to as high as 730 days.

To change the data retention period for unmapped scans:

1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **System Settings**.
4. Click **Data Retention**.
5. Click **Unmatched Scans Retention**.


Data Retention ▶ Unmapped Scans Retention

Unmapped Scan Retention (Days) *

Enter a value from 1 to 365 representing the data retention period in days for unmapped scans. The default period is 30 days. Updating this setting can take several minutes to take effect.

Reset

Save

 **Note:** Changing this setting will take effect the next time [ScanPurgeJob](#) runs, which occurs every 15 minutes.

Protecting scans by project version

You can prevent scans in designated project versions from being unmapped by **Scan Auto-Unmapping** by following these steps:

1. [Navigate to the desired project version.](#)
2. Click the **Settings** tab.
3. In the Version Details pane, find the **Scan Retention** section.
4. Check the **Prevent automatic unmapping of scans** checkbox.

Retain Unmatched File Data

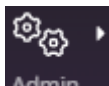
Changing the Retain Unmatched File Data setting globally

You can change whether or not your system retains unmatched files. By default, this setting is not enabled.

To enable the retain unmatched files data setting:

1. Log in to Black Duck with the System Administrator role.

- 2.



Click **Admin**.

3. Select **System Settings**.
4. Click **Data Retention**.
5. Click **Unmatched File Data Retention**.

Data Retention ▶ Unmatched File Data Retention

If enabled, unmatched file data for scans will always be retained. When disabled (default), unmatched file data will be purged.

Retain Unmatched File Data

Purge Now

Selecting one of these options will purge unmatched file data immediately.

 [Purge ONLY Archived Project Version Unmatched File Data](#)


 [Purge ALL Unmatched File Data](#)

Reset

Save

6. Check the **Retain Unmatched File Data** checkbox.

If enabled, unmatched file data for scans will always be retained. When disabled (default), unmatched file data will be purged. Note that the global setting only applies to projects and scans that do not explicitly specify their own setting; similarly, changing the global setting does not affect projects or scans that do specify their own setting.

 **Warning:** Once unmatched files are purged, they cannot be recovered except by restoring from backup.

You can also manually purge unmatched data immediately by clicking either of the following buttons:

Purge ONLY Archived Project Version Unmatched File Data: Clicking this button will purge all unmatched data currently in the [Archived project version phase](#) only.

Purge ALL Unmatched File Data: Clicking this button will purge all purge all unmatched data regardless of its project version phase.

As indicated above, clicking either button only applies to projects and scans that do not explicitly specify their own setting.

Changing the Retain Unmatched File Data setting for a project

You can set a project's setting to have its own policy of whether or not the unmatched file data is purged by following these steps:

1. Select the project name using the **Watching** or **My Projects** dashboard. The *Project Name* page appears.
2. Select the **Settings** tab.
3. Scroll to the **Retain Unmatched File Data** section.

Retain Unmatched File Data

If enabled, unmatched file data for scans will always be retained. When disabled (default), unmatched file data will be purged.

Don't Retain Unmatched File Data

Purge Now

 [Purge ONLY Archived Project Version Unmatched File Data](#)

 [Purge ALL Unmatched File Data](#)

- Select an option from the dropdown menu:

Retain Unmatched File Data: Selecting this option will prevent this project's data from being purged regardless of the global system default setting. This also disables the buttons granting the ability to manually purge unmatched file data.

Don't Retain Unmatched File Data: Selecting this option will allow this project's unmatched file data to be purged. This also enables the buttons granting the ability to manually purge unmatched file data.

System Default: Selecting this option will use the global system default setting as set above. The buttons granting the ability to manually purge unmatched file data will either be enabled or disabled depending on how the system setting is configured.

- Click **Save**.

You can also manually purge unmatched data immediately for this project like from the Retain Unmatched File Data administration page.

BDSA Auto Remediation

When the Black Duck Security Advisory (BDSA) team analyzes a CVE vulnerability, they check to see what component versions are affected by the vulnerability. Sometimes they find that the vulnerability applies to a different set of versions. This feature will give you the ability to automatically ignore CVE vulnerabilities if the BDSA team has found that the vulnerability does not apply to that component version.

This setting only applies to CVE vulnerabilities with a related BDSA vulnerability. If the CVE is mapped to a component version, but its related BDSA is not also mapped to that component version then the system may automatically remediate the CVE vulnerability based on the feature setting.

Administration
System Settings

Branding

Custom Fields

Security Risk Ranking

Legal

Scan

Roles

User Authentication

Synopsys Detect

Product Registration

Data Retention

BDSA Auto Remediation

BDSA Auto Remediation

When enabled, all added or updated project vulnerabilities will be automatically remediated if they have a related BDSA that is not mapped to the same component version.

[Learn More](#)

Enable BDSA auto remediation

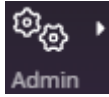
Reset Save Save and Apply

Changing the BDSA Auto Remediation setting

To change the BDSA Auto Remediation setting:

- Log in to Black Duck with the System Administrator role.

2.

Click **Admin**.3. Select **System Settings**.4. Click **BDSA Auto Remediation**.5. Check the **Enable BDSA auto remediation** checkbox to activate the feature or remove the check to disable it.6. Click either the **Save** or **Save and Apply** button:

- **Save:** Clicking this button will save the current setting (on or off) for the BDSA Auto Remediation feature and apply the changes going forward for all new scans. It will not update any existing CVEs.
- **Save and Apply:** Clicking this button will save the current setting and apply the changes going forward for all new scans as well as updating all existing CVE vulnerabilities. The button's behavior changes depending on whether the feature is being activated or deactivated.

If you have activated the BDSA Auto Remediation feature, NEW status CVEs with a related BDSA that are not also mapped to that component version will be auto remediated by changing the remediation status from NEW to IGNORED. The system will also add a message to describe why the vulnerability was remediated

If you have deactivated the BDSA Auto Remediation feature, click the Save and Apply button will undo all auto remediations performed prior. Undoing an auto remediation will change the remediation status of an auto remediated vulnerability from IGNORED to NEW as well as remove the message that was added to the remediation.

Changing this setting will initiate the [AutoRemediateUnmappedJob](#) job which will process the changes above. This process may take some time depending on the quantity of updates needed to perform. You can cancel this job by clicking the Cancel Job button.

BDSA Auto Remediation

 An auto remediation job is currently in progress.

[Cancel Job](#)

When enabled, all added or updated project vulnerabilities will be automatically remediated if they have a related BDSA that is not mapped to the same component version.


[? Learn More](#)

Enable BDSA auto remediation

Reset

Save

Save and Apply

 **Note:** Canceling this process will not revert changes made during the process. The **Save** and **Save and Apply** buttons will also be disabled during this process.

For more information regarding Black Duck Security Advisories and how to remediate vulnerabilities, please refer to:


- [Black Duck Security Advisories](#)
- [Remediating security vulnerabilities](#)

Access Tokens

User Administrators of the Black Duck system need a mechanism to maintain and control access to Black Duck via access tokens. User access is often controlled via Single Sign-On integration, but access tokens are managed independently by Black Duck. Administrators need to ensure security of the system and therefore need the tools to revoke or reset access when required. This page allows the User Administrator to manage all access tokens by either curating the list manually or by setting up an automated purging schedule.

The list of access tokens is composed of the following:

- **Name:** The name of the access token
- **Description:** The description given to the access token.
- **Owner:** The name of the user who created the access token.
- **Usage Count:** The number of times the access token was used.
- **Last Generated:** The date or time the access token was created or regenerated.
- **Last Used:** The date or time the access token was last used.

Administration
Access Tokens




Auto Purge Job

Next auto purge job is scheduled to purge tokens inactive for more than 26 weeks on Tue, Jun 14, 2022 9:30 PM

Maximum weeks of inactivity *

26

Enable Auto Purge Job



Filter by token owner... 

<input type="checkbox"/>	Name	Description	Owner	Usage Count	Last Generated	Last Used
<input type="checkbox"/>	An Access Token	Read and write access	System Administrator	0	11:38 AM	11:38 AM
<input type="checkbox"/>	Other Access Token	Read access only	Sample User	0	1:51 PM	1:51 PM

Displaying 1-2 of 2


Manual access token deletion

To delete access tokens from the list manually:

1. Log in to Black Duck with the User Administrator role.
2.  Click **Admin**.
3. Click **Access Tokens**.
4. Check the box next to any number of access tokens.
5. Click the **Delete** button. A **Delete Token** dialog box will appear.
6. Confirm the access token deletion by clicking the **Delete** button in the dialog box.

Setting up the automated access token purge job

To change the access token purge job setting:

1. Log in to Black Duck with the User Administrator role.
2.  Click **Admin**.
3. Click **Access Tokens**.
4. Set the desired period of time for the **Maximum weeks of inactivity**.
5. Check the **Enable Auto Purge Job** checkbox to activate the feature or remove the check to disable it.
6. Click either the **Save** button.

On-demand access token purge

You can also initiate an on-demand purge immediately by clicking the **Purge Now...** button. Clicking this button will open the Purge Access Tokens Now dialog box.

Purge Access Tokens Now
✕

An on-demand purge will immediately delete all access tokens that have not been used in the specified time frame.

Purge by weeks of inactivity

|
▾

Purge all tokens

Cancel
Next

Select either of the following options:

- **Purge by weeks of inactivity:** Set the time frame for the number of weeks of inactivity.
- **Purge all tokens:** Will delete all created access tokens.

Click the **Next** button to continue the action.

The job responsible for conducting the access token purging is [ApiTokenPurgeCheckJob](#).


Downloading log files and heatmap data

You may need to troubleshoot an issue or provide log files to Customer Support. Users with the System Administrator role can download a zipped file that contains the current log files or the heatmap data for your system.

Please note that it may take a few minutes to prepare the log or heatmap files. Refer to the installation guide for more information on obtaining logs and configuring heatmap data.

Accessing log files


To download the log files from the Black Duck UI:

1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **Jobs**.
4. Select the **System Information** tab.
5. Click **Download Logs (.zip)**.

Accessing heatmap data

You can review and analyze terminal scan trends by downloading the heatmap as a compressed CSV and create the heatmap as a pivot in a spreadsheet program.

To download the heatmap data for your system:


1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Select **Jobs** in the **Diagnostics** section of the Admin menu.
4. Select the **System Information** tab.
5. Click **Download Heatmap (.zip)**.

Viewing heatmap data

The Black Duck Heatmap provides an intuitive and powerful solution to capture, present, analyze, distribute, and automate data analysis, problem detection, problem identification, and applying limited solutions that are mapped to known problems. Statistical data from Black Duck is represented in a matrix with hour of day as one axis and day (or date) of month as the other axis.

The heatmaps are color coded to display the minimum and maximum value in the dataset and to calculate percentages for each value relative to the minimum and maximum. The smallest number is 0% (green color) and the largest is 100% (red color). We then use this percentage to determine the exact color value to use in the heatmap.



 **Note:** Red values in the heatmap does not indicate an error. It is the maximum boundary when determining the amount of scans conducted in the last 30 days.

To view a UI representation of the heatmap:

1. Log in to Black Duck with one of the following roles:


- Global Code Scanner
- Global Notification Viewer
- Global Project Administrator
- Global Project Manager
- Global Project Viewer

2.



Click **Admin**.


3. Select **Heatmaps** in the **Diagnostics** section of the Admin menu.

 **Note:** Heatmap data is populated when the scan is completed and becomes read-only afterwards. Changes made to the projects are not synced with heatmap data.

Filtering the heatmap

You can filter the data displayed in the heatmap by clicking the blue **+ Filter** button on the top right of the page and then using any of the following options. Multiple filters can be added simultaneously to further refine the results:

- **Code Location ID:** Select one or many code location IDs.
- **Code Location Name:** Select one or many code location names.
- **Project Name:** Select one or many project names.
- **Scan Date:** Select a start and end date.
- **Scan Status:** Select from Success, Started, and/or Failure.
- **Scan Type:** Select from the available scan types.
- **Version Name:** Select any version names.

You can edit filters by clicking on an existing filter and adding additional criteria. You can remove an active filter by clicking the  button to the right of the filter.

Creating a Heatmap in Excel

Creating a heatmap in Excel on Mac

Users with the System Administrator role can download a zipped file that contains the current log files.

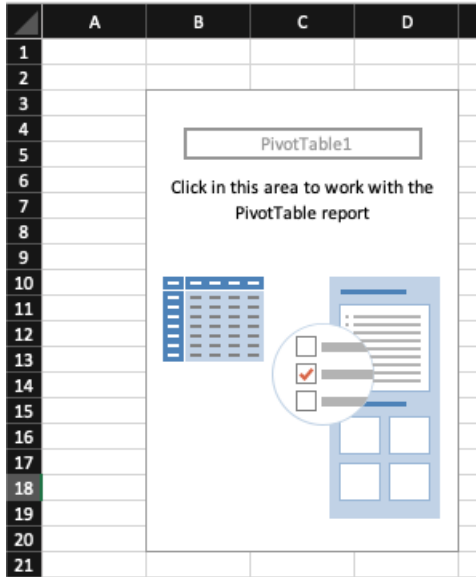
To create a heatmap in Excel:

1. Log in to Black Duck with the System Administrator role.
2. [Download the log files.](#)
3. Extract the logs to a folder.

18. Other administrative tasks • Creating a Heatmap in Excel

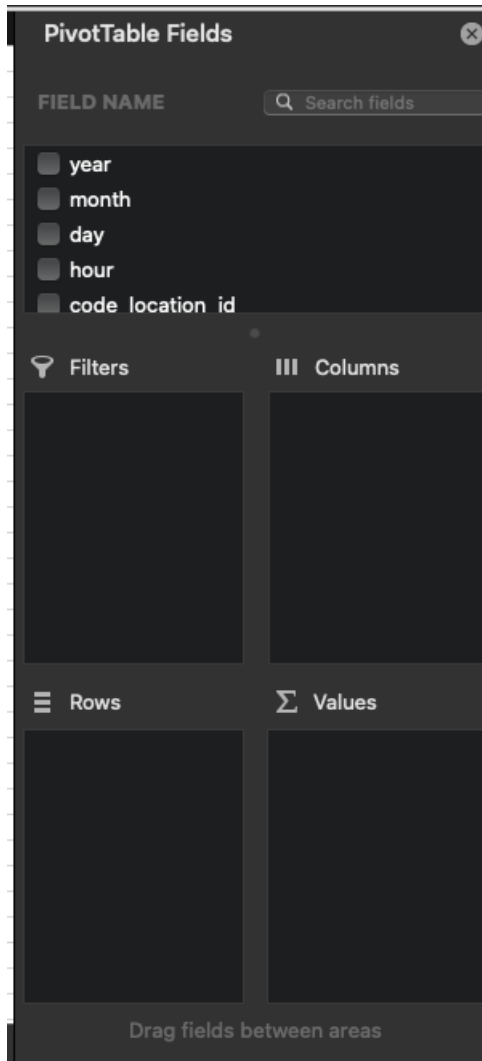
4. Open a new and blank workbook using Microsoft Excel. Using other spreadsheet/workbook programs may yield different results or differ in the steps below.
5. On the first top empty cell of an empty worksheet:
 - a. Click the Data tab > Get Data > From text
 - b. Navigate to the location of your download logs folder/debug folder
 - c. Select the Heatmap CSV file (scan-heatmap-YYYY-MM-#####.csv)
 - d. Click Import
6. From the Text Import Wizard perform following:
 - a. Enable the Delimited radio button
 - b. Start import at row <1>
 - c. Click the Next button
 - d. Uncheck the Tab (as delimiter) box
 - e. Check the Comma (as delimiter) box
 - f. Click the Next button
 - g. Click the Finish button
7. In the Import Data modal:
 - a. Select either the Existing sheet or New Sheet from 'Where do you want to put data'
8. Select the entire data that has been imported
9. Click the Insert tab
10. Click the Table icon
11. In the Create Table modal that appears, click OK. If an alert pop-up appears, select Yes. The imported data should now be a table with the first row as filters. This is the raw data for the Heatmap. Note the Table name in the top left corner of Excel (i.e. Table1).
12. In a new worksheet:
 - a. Click on any cell
 - b. Click the Insert tab
 - c. Click the Pivot icon
13. In the Create PivotTable dialog box, do the following:
 - a. Type the table name (in this case 'Table1') in the *Select a Table or Range* text field
 - b. Select Existing worksheet in the *Choose where to place the PivotTable* section
 - c. Click the OK button

Your worksheet should look like this:



14. Click in the Blank Pivot created, this should open the Pivot column panel.

18. Other administrative tasks • Creating a Heatmap in Excel



15. Drag the following to filter section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project, Version
 - g. Code_Location_Name
16. Drag the Hour field to the Columns section.
17. Drag the Day field to the Rows section.
18. Drag the Scans field to the Values section. It should show as 'Sum of Scans'.
19. Rename Column header from Row Labels to Days
20. Rename 'Column Labels' Filter in second column to Hours

The presentation in Excel should now look like this:

The screenshot shows an Excel PivotTable with the following data:

Sum of scans	Hours	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	3 AM	2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12			2		1		3
13		2		4	1		7
14		2	2				4
Grand Total		12	5	4	5	2	32

The PivotTable Fields task pane on the right shows the following configuration:

- Filters:** year, month, day, hour
- Columns:** code_location_id, code_location_name, version, project, scans
- Rows:** day
- Values:** Sum of scans

21. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))

The close-up shows the PivotTable data with a green selection box highlighting the range from row 5 to row 14 and columns 3 AM to 7 PM. The 'Grand Total' row and column are excluded from the selection.

Sum of scans	Hours	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	3 AM	2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12			2		1		3
13		2		4	1		7
14		2	2				4
Grand Total		12	5	4	5	2	32

22. Click the Home menu item
23. From the Conditional Formatting > Color Scales, select Red - Yellow - Green scale

18. Other administrative tasks • Creating a Heatmap in Excel

Sum of scans		Hours					
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	2			1			3
6	2						2
9		1					1
10	2	2					4
11	2	2				4	8
12			2		1		3
13	2			4	1		7
14	2		2				4
Grand Total	12	5	4	5	2	4	32

24. Select the cells in bottom Grand Total row and apply the same color scale (excluding first and last column). Only Hours data cells).

Sum of scans		Hours					
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	2			1			3
6	2						2
9		1					1
10	2	2					4
11	2	2				4	8
12			2		1		3
13	2			4	1		7
14	2		2				4
Grand Total	12	5	4	5	2	4	32

25. Select the cells in the rightmost Grand Total column and apply the same color scale.

Sum of scans		Hours					
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	2			1			3
6	2						2
9		1					1
10	2	2					4
11	2	2				4	8
12			2		1		3
13	2			4	1		7
14	2		2				4
Grand Total	12	5	4	5	2	4	32

26. Select the bottom right corner cell alone and provide it with a blue background.

Sum of scans		Hours					
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	2			1			3
6	2						2
9		1					1
10	2	2					4
11	2	2				4	8
12			2		1		3
13	2			4	1		7
14	2		2				4
Grand Total	12	5	4	5	2	4	32

27. To analyze the data behind any cell, double click on it.

Maximum Scan Size Heatmap

1. Add a blank new sheet to the workbook.
2. In the new worksheet, click on any cell, click on Insert tab, click on Pivot icon.

3. Click on the Blank Pivot created, this should open the Pivot column panel.
4. Drag the following to filter section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project
 - g. Version
 - h. Code_Location_Name
5. Drag the Hour field to the Columns section.
6. Drag the Day field to the Rows section.
7. Drag avg_scan_size_in_gb field to the Values section
8. Change to show Maximum such that it should display 'Max of avg_scan_size_in_gb':
 - a. Click on the field in the values section. This will launch the PivotTable field.
 - b. Choose Maximum instead of Sum.
 - c. Click the OK button.
9. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))
10. Click the Home tab, Conditional Formatting > Color Scales > Yellow - Green color scale.
11. Select the cells in bottom Grand Total row and apply the same color scale.
12. Select the cells in the rightmost Grand Total column and apply the same color scale.
13. Select the bottom right corner cell alone and provide it with a blue background.

Scan Weight Heatmap

1. Add a blank new sheet to the workbook.
2. In the new worksheet, click on any cell, click on Insert tab, click on Pivot icon.
3. Click on the Blank Pivot created, this should open the Pivot column panel.
4. Drag the following to filter section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project
 - g. Version
 - h. Code_Location_Name

5. Drag the Hour field to the Columns section.
6. Drag scan_weight field to the Values section and change to show Maximum such that it should display 'Avg of scan_weight'. This can be set by clicking on the field in the values section and then choosing Average instead of Sum.
7. Drag the Scans field to the Values section. It should show as 'Sum of Scans'.
8. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))
9. Click the Home tab, Conditional Formatting > Color Scales > Red - White color scale.
10. Select the cells in bottom Grand Total row and apply the same color scale.
11. Select the cells in the rightmost Grand Total column and apply the same color scale.

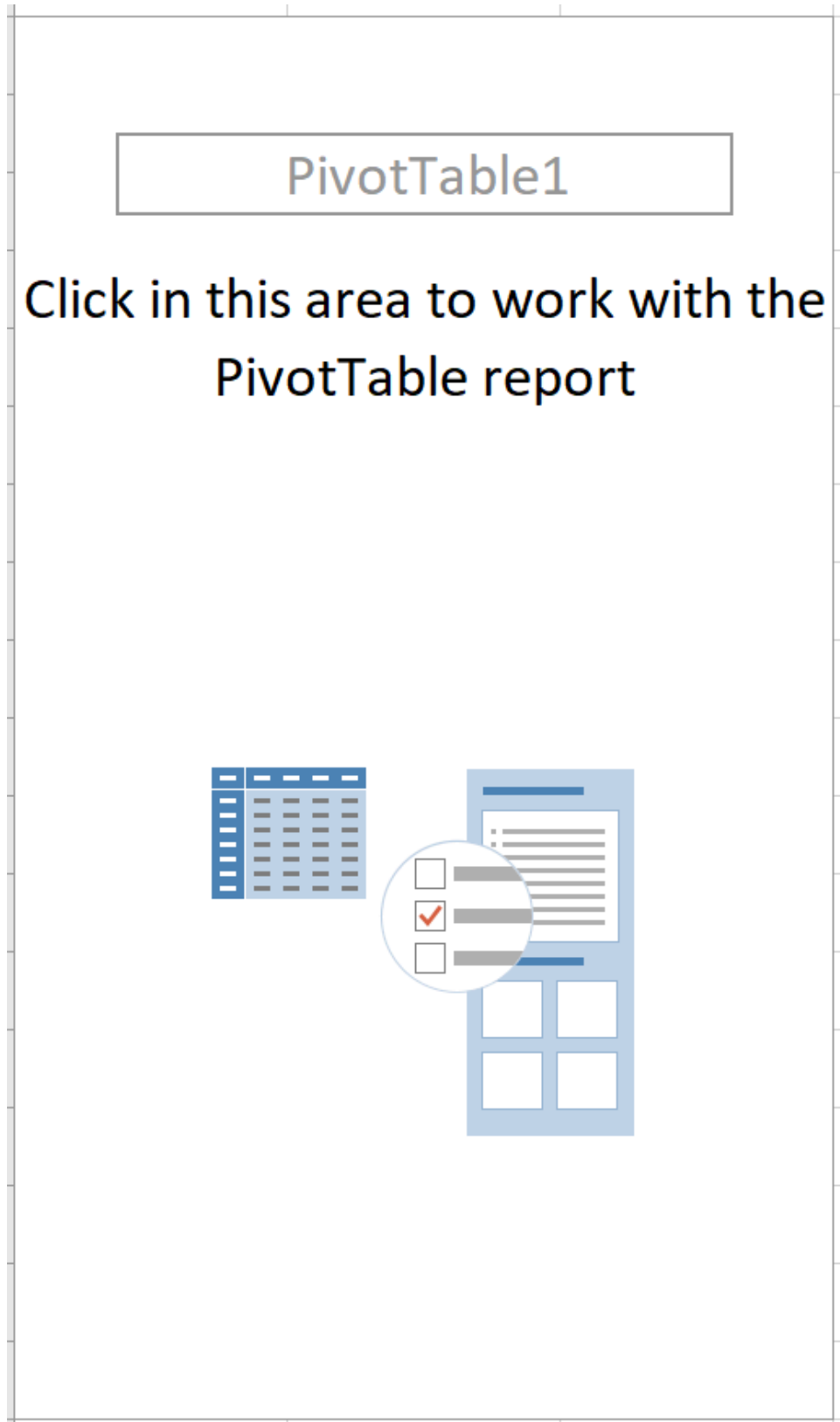
Creating a heatmap in Excel on Windows

Users with the System Administrator role can download a zipped file that contains the current log files.

To create a heatmap in Excel:

1. Log in to Black Duck with the System Administrator role.
2. Extract the logs to a folder.
3. Open a new and blank workbook using Microsoft Excel. Using other spreadsheet/workbook programs may yield different results or differ in the steps below.
4. On the first top empty cell of an empty worksheet:
 - a. Click the Data tab > Get Data > From File > From Text/CSV
 - b. Navigate to the location of your download logs folder/debug folder
 - c. Select the Heatmap CSV file (scan-heatmap-YYYY-MM-#####.csv)
 - d. Click Import
5. From the Import Wizard perform following:
 - a. Ensure that the Delimiter dropdown is set to Comma
 - b. Click the Load button
6. Select the entire data that has been imported
7. Click the Insert tab
8. Click the PivotTable icon
9. In the Create PivotTable modal that appears:
 - a. Ensure the *Select a table or range* radio button is selected with the appropriate selected field
 - b. In the *Choose where you want the PivotTable report to be placed* section, select the New Worksheet
 - c. Click the OK button.

Your worksheet should look like this:



10. Click in the Blank Pivot created, this should open the PivotTable Fields column panel.

11. Drag the following to the Filters section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project, Version
 - g. Code_Location_Name
12. Drag the Hour field to the Columns section.
13. Drag the Day field to the Rows section.
14. Drag the Scans field to the Values section. It should show as 'Sum of Scans'.
15. Rename the column header from 'Row Labels' to Days
16. Rename the 'Column Labels' filter in second column to Hours

The presentation in Excel should now look like this:

The screenshot shows an Excel PivotTable with the following structure:

Sum of scans	Hours	1 AM	2 AM	3 AM	4 AM	5 AM	6 AM	10 AM	1 PM	2 PM	3 PM	4 PM	5 PM	6 PM	7 PM	8 PM	9 PM	10 PM	11 PM	Grand Total					
5	Days												1							3					
6					2															2					
10					2															2					
11				1											4					5					
12										2					1					3					
13													4							4					
14										1	2		1							4					
15										2										4					
16											2									4					
17																				2					
18										4	2	2			2		4	6		20					
19										2	2				5			2		15					
20			2		4															24					
21					2	1	4	1												10					
22					2					1	4	1	2							7					
23					2										5					2					
24																				2					
25																				2					
26																				2					
27																				2					
28																				2					
29							8													51					
30						1	7	9	2											58					
31						2														2					
32																									
33																									
34																									
35																									
36																									
37																									
38																									
39																									
40																									
41																									
42																									
43																									
Grand Total						7	15	13	21	1	4	1	2	11	21	25	14	9	57	20	4	20	12	13	270

The PivotTable Fields task pane on the right shows the following configuration:

- Filters:** year, month, scan_type, status, status_message, project, version, code_location_name
- Columns:** hour
- Rows:** day
- Values:** Sum of scans

17. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))

18. Other administrative tasks • Creating a Heatmap in Excel

Sum of scans	Hours						
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5		2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12				2		1	3
13		2			4	1	7
14		2		2			4
Grand Total		12	5	4	5	2	32

18. Click the Home menu item

19. From the Conditional Formatting > Color Scales, select Red - Yellow - Green scale

Sum of scans	Hours						
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5		2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12				2		1	3
13		2			4	1	7
14		2		2			4
Grand Total		12	5	4	5	2	32

20. Select the cells in bottom Grand Total row and apply the same color scale (excluding first and last column. Only Hours data cells).

Sum of scans	Hours						
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5		2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12				2		1	3
13		2			4	1	7
14		2		2			4
Grand Total		12	5	4	5	2	32

21. Select the cells in the rightmost Grand Total column and apply the same color scale.

Sum of scans	Hours						
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5		2			1		3
6		2					2
9			1				1
10		2	2				4
11		2	2			4	8
12				2		1	3
13		2			4	1	7
14		2		2			4
Grand Total		12	5	4	5	2	32

22. Select the bottom right corner cell alone and provide it with a blue background.

Sum of scans	Hours						Grand Total
Days	3 AM	4 AM	2 PM	4 PM	6 PM	7 PM	Grand Total
5	2			1			3
6	2						2
9		1					1
10	2	2					4
11	2	2				4	8
12			2		1		3
13	2			4	1		7
14	2		2				4
Grand Total	12	5	4	5	2	4	32

23. To analyze the data behind any cell, double click on it.

Maximum Scan Size Heatmap

1. Add a blank new sheet to the workbook.
2. In the new worksheet, click on any cell, click on Insert tab, click on Pivot icon.
3. Click on the Blank Pivot created, this should open the Pivot column panel.
4. Drag the following to filter section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project
 - g. Version
 - h. Code_Location_Name
5. Drag the Hour field to the Columns section.
6. Drag the Day field to the Rows section.
7. Drag avg_scan_size_in_gb field to the Values section
8. Change to show Maximum such that it should display 'Max of avg_scan_size_in_gb':
 - a. Click on the field in the values section. This will launch the PivotTable field.
 - b. Choose Maximum instead of Sum.
 - c. Click the OK button.
9. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))
10. Click the Home tab, Conditional Formatting > Color Scales > Yellow - Green color scale.
11. Select the cells in bottom Grand Total row and apply the same color scale.
12. Select the cells in the rightmost Grand Total column and apply the same color scale.
13. Select the bottom right corner cell alone and provide it with a blue background.

Scan Weight Heatmap

1. Add a blank new sheet to the workbook.
2. In the new worksheet, click on any cell, click on Insert tab, click on Pivot icon.
3. Click on the Blank Pivot created, this should open the Pivot column panel.
4. Drag the following to filter section in this order:
 - a. Year
 - b. Month
 - c. Scan_Type
 - d. Status
 - e. Status_Message
 - f. Project
 - g. Version
 - h. Code_Location_Name
5. Drag the Hour field to the Columns section.
6. Drag scan_weight field to the Values section and change to show Maximum such that it should display 'Avg of scan_weight'. This can be set by clicking on the field in the values section and then choosing Average instead of Sum.
7. Drag the Scans field to the Values section. It should show as 'Sum of Scans'.
8. Select all the cells that fall within the range between the first to last Hours column (do not include filter top row, 'Grand Total' bottom row or last column (Grand Total))
9. Click the Home tab, Conditional Formatting > Color Scales > Red - White color scale.
10. Select the cells in bottom Grand Total row and apply the same color scale.
11. Select the cells in the rightmost Grand Total column and apply the same color scale.

Viewing jobs

You can view all the jobs in the system if you need to troubleshoot an issue and determine if a process ran.

Note that any job older than 30 days is purged from the list.

Possible jobs are:

Job Name	Description
Auto Remediate Unmapped	Auto remediate CVEs with unmapped related BDSAs.
BDIO Data Transfer	Processes scan data and prepares it for the matching process.
BDIO Storage Migration Check	Checks if there are any BDIO files to migrate to the storage service.
BOM Event Cleanup	Cleans up BOM events based on the retention policy.

Job Name	Description
BOM Vulnerability Recomputation Check	Checks if BOM computations are required when certain settings change and starts the necessary jobs.
Check Sigma Tool Version	Checks if the Sigma tool version is up to date and schedules the work if it is not.
Checks Need for Hierarchical BOM Calculation	Checks if hierarchical BOM computations are required and starts the necessary jobs to process them
Hierarchical Version BOM	Creates and updates the hierarchical version BOM.
Job History Statistics	Calculates statistics from the job history.
Journal Partition Maintenance	Creates new database partitions for the project audit trails and drops old partitions. The Journal table is partitioned by month. The first partition is special and contains all existing journal events. Journal events older than 5 years will be purged.
KnowledgeBase Update Check	Initiates updates received from the KnowledgeBase.
License Term Check	Checks if license fulfillment processing is required and starts the necessary jobs.
Update Origin Copyrights	Updates origin copyrights.
Notification Purge Check	Checks if there are notifications that need cleanup and starts the necessary jobs.
Populate License Terms	Updates Black Duck with the latest Black Duck KB license term data.
Purge API Token Check	Determines if the access tokens auto purge job needs to run.
Purge API Token	Deletes inactive access tokens based on configured settings.
Purge Deleted Storage Objects	Removes deleted storage objects from the system and cleans out orphaned records.
Purge Notifications	Manages data retention for existing notifications.
Purge Orphan BOMs	Deletes any BOM data not associated with a project version.
Purge Orphan BOMs Check	Checks to see if any BOM data is not associated with a project version and starts the necessary jobs.
Purge Reports	Manages data retention for existing reports and purges expired reports.
Purge Scan Data - Delete Abandoned Scans	Deletes scans that were started but never completed.
Purge Scan Data - Delete Expired Scans	Deletes scans that are older than the expiration threshold.


18. Other administrative tasks • Viewing jobs

Job Name	Description
Purge Scan Data - Delete Orphaned Scan Identifiers	Deletes orphaned snippet scan identifiers.
Purge Scan Data - Delete Orphaned Scans	Deletes orphaned scans.
Purge Scan Data - Delete Stale Projects	Deletes stale projects.
Purge Scan Data - Delete Stale Releases	Deletes stale releases.
Purge Scan Data - Delete Unmapped Scans	Deletes unmapped scans.
Purge Scan Data - Purge Component Mapping Audit Events	Purges component mapping audit events.
Purge Scan Data - Purge Deleted Scans	Deletes scans that were previously queued for deletion.
Purge Scan Data - Purge Scan Archives	Deletes scan archives that are eligible for deletion.
Purge Scan Data - Purge Unmatched Files	Deletes unmatched files from scans that are eligible for deletion.
Purge Scan Data Scheduler	Schedules job that removes old scan data.
Purge Scan Statistics	Removes old scan statistics.
Report Storage Migration Check	Checks whether there are reports that need to be migrated into the storage service.
Reporting Database Transfer	Migrates Black Duck data to the Black Duck reporting warehouse.
Reporting Database Transfer Scheduler	Schedules job that transfers reporting data to the reporting schemas for customers.
SBOM Report	Generates the SBOM report for a Project Version.
Scan Auto BOM Calculation	Calculates a BOM from a scan.
Scan Statistics	Collects scan statistics shown on the usage: scan completion section on the System Information page.
Scheduled Policy Rule Changes Check	Finds policy violation overrides with scheduled expirations.
Schema Difference Report	Calculates the differences between the current database schema and the ideal as determined at release.
SCM Onboarding daily auto scanning	Schedules nightly job that performs auto scanning of previously onboarded SCM repositories.
SCM Onboarding daily cleanup	Schedules nightly job that cleans up from SCM Onboarding.
Search Dashboard Refresh	Updates the information shown on the Projects and Components Dashboards.
Search Dashboard Refresh Check	Checks if it is time to refresh the Projects and Components Dashboards and starts the necessary job.

Job Name	Description
Snippet BOM Calculation	Calculates a BOM from a snippet scan.
Storage Migration Check	Checks to see if there is migration work to perform.
Storage Pruning Check	Checks if the object storage system has items to prune.
System Maintenance and Reporting	Processes system statistics and registers them with the KnowledgeBase.
Test Performance Dispatch	System diagnostic that tests the performance of job dispatch.
Update KnowledgeBase Data - BDSA Vulnerability Update	Updates BDSA vulnerability information received from the KnowledgeBase.
Update KnowledgeBase Data - Component Update	Updates component information received from the KnowledgeBase.
Update KnowledgeBase Data - Component Version Security Update	Processes component version updates received from the KnowledgeBase.
Update KnowledgeBase Data - License Update	Updates license information received from the KnowledgeBase.
Update KnowledgeBase Data - NVD Vulnerability Update	Updates NVD vulnerability information received from the KnowledgeBase.
Update KnowledgeBase Data - Summary	Issues a summary report about the most recent KnowledgeBase update.
Version BOM Notification	Notifies on various BOM-related events.
Version BOM Notification Check	Checks to see if any version BOM notifications need processing.
Version Bom Computation Check	Checks if any version BOMs need computations and schedules the work.
Version License Report	Creates the Notices File report .
Version Report	Creates the Project Version report .
Version Vulnerability Remediation Report	Creates the Project Version Vulnerability Remediation Report.
Version Vulnerability Status Report	Creates the Project Version Vulnerability Status Report.
Version Vulnerability Update Report	Creates the Project Version Vulnerability Update Report.
Vulnerability Remediation Report	Creates the Vulnerability Remediation Report .
Vulnerability Status Report	Creates the Vulnerability Status Report .
Vulnerability Update Report	Creates the Vulnerability Update Report .
Watchdog	Monitors the job subsystem for errors and reports or fixes issues as they arise.

Viewing jobs

To view a list of jobs and their current statuses:

1. Log in to Black Duck with the System Administrator role.
2.  Click **Admin**.
3. Click **Jobs**.
4. Click the **Jobs** tab to display the Jobs page.

Filtering the Jobs table

You can refine the jobs displayed in the table by selecting one of the following options:

- **Finished**: Displays all finished jobs.
- **Scheduled**: Displays all jobs set to run in your environment.
- **Processing**: Displays all jobs currently processing.

Setting up SCM providers

Black Duck SCM integrations allows for direct communication with SCM providers, allowing Black Duck to automatically obtain repository and branch information when Detect scans are performed on cloned Git repositories, and populating dropdowns and search boxes in order to increase ease of use and data accuracy.

Enabling SCM integration

SCM integration utilizes a service that runs strictly within a Kubernetes environment, either native or Kubernetes in Docker (KinD). The helm charts will need to be used to install Black Duck to use this feature. This feature is not enabled by default in Black Duck and must be activated by adding the feature to your [Product Registration key](#) and then adding the following in your `values.yaml` file:

```
enableIntegration: true
```

 **Note:** Black Duck does not accept self-signed certificates for SCM integrations at this time.

Creating an OAuth App

Before setting up a SCM provider in Black Duck, you must first authenticate the project.

For GitHub and GitHub Enterprise, you must create an OAuth App:

1. Go to <https://github.com/settings/developers> and OAuth Apps and create a new app (or the corresponding URL for GitHub Enterprise).
2. Fill the following fields:
 - **Application Name**
 - **Homepage URL**: The URL of your Black Duck Server
 - **Application Description**
 - **Authorization Callback URL**: `<Homepage URL>/api/scm/github/callback`

3. Click **Save**. This will generate the Client ID to be used in Black Duck.
4. Click **Generate secret**. This will generate a secret string to be used in Black Duck.

For GitLab Self-Managed:


1. Go to <gitlab_server_name>/-/profile/applications. You should see add new application.
2. Fill the following fields:
 - **Name**: provide any name.
 - **Redirect URI**: <bd_server_name>/api/scm/gitlab/callback
3. Uncheck the **Confidential** checkbox.
4. Enable API in the **Scopes** section.

For BitBucket:

1. Go to <bitbucket_server_name>/plugins/servlet/applinks/listApplicationLinks
2. Click **Create Link**.
3. Select **External application**.
4. Select **Incoming** in the **Direction** dialog box and then click OK.
5. Fill the following fields:
 - **Name**: Provide a name.
 - **Redirect URI**: <bd_server_name>/api/scm/bitbucket/callback
6. Check the **Write** checkbox under **Repositories** in the **Application permissions** section.

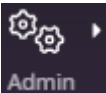
Setting up a GitHub.com SCM integration

To set up a GitHub.com SCM integration:

1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.
3. Click **GitHub.com**.
4. Fill the following fields:
 - Check the **Enable Server** checkbox.
 - Enter the **Client ID** generated from the GitHub website.
 - Enter the **Secret** generated from the GitHub website.
5. Click **Save**.

Setting up a GitHub Enterprise SCM integration


To set up a GitHub Enterprise SCM integration:

1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.

3. Click **GitHub Enterprise**.
4. Click **+ Add Server**.
5. Fill the following fields:
 - **Server Name**: Enter a name for your server.
 - **Server URL**: Enter your GitHub Enterprise server URL.
 - **Client ID**: Enter the **Client ID** generated from the GitHub website.
 - **Secret**: Enter the **Secret** generated from the GitHub website.
 - Check the **Enable Server** checkbox.
6. Click **Create**.


Setting up a GitLab Self-Managed SCM integration

To set up a GitLab Self-Managed SCM integration:

1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.
3. Click **GitLab Self-Managed**.
4. Click **+ Add Server**.
5. Fill the following fields:
 - **Server Name**: Enter a name for your server.
 - **Server URL**: Enter your GitLab Self-Managed server URL.
 - **Client ID**: Enter the **Client ID** generated from the GitLab website.
 - **Secret**: Enter the **Secret** generated from the GitLab website.
 - Check the **Enable Server** checkbox.
6. Click **Create**.

Setting up a GitLab SaaS SCM integration


To set up a GitLab SaaS SCM integration:

1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.
3. Click **GitLab SaaS**.
4. Click **+ Add Server**.
5. Fill the following fields:
 - **Server Name**: Enter a name for your server.
 - **Client ID**: Enter the **Client ID** generated from the GitLab website.
 - **Secret**: Enter the **Secret** generated from the GitLab website.

- Check the **Enable Server** checkbox.
6. Click **Create**.

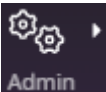
Setting up a Bitbucket SCM integration

To set up a Bitbucket SCM integration:

1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.
3. Click **Bitbucket**.
4. Click **+ Add Server**.
5. Fill the following fields:
 - **Server Name:** Enter a name for your server.
 - **Client ID:** Enter the **Client ID** generated from the Bitbucket website.
 - **Secret:** Enter the **Secret** generated from the Bitbucket website.
 - Check the **Enable Server** checkbox.
6. Click **Create**.

Setting up a Bitbucket Data Center SCM integration

To set up a Bitbucket Data Center SCM integration:


1. Log into Black Duck as a System Administrator.
2.  Click **Admin** and select **Integrations**.
3. Click **Bitbucket Data Center**.
4. Click **+ Add Server**.
5. Fill the following fields:
 - **Server Name:** Enter a name for your server.
 - **Server URL:** Enter your Bitbucket Data Center server URL.
 - **Client ID:** Enter the **Client ID** generated from the Bitbucket Data Center website.
 - **Secret:** Enter the **Secret** generated from the Bitbucket Data Center website.
 - Check the **Enable Server** checkbox.
6. Click **Create**.

Managing SCM providers



Editing SCM integrations

To edit the GitHub.com SCM integration:

18. Other administrative tasks • Setting up SCM providers



1. Log into Black Duck as a System Administrator.
2. 
Click **Admin** and select **Integrations**.
3. Click **GitHub.com**.
4. Edit any of the fields as desired.
5. Click **Update**.

To edit a GitHub Enterprise, GitLab Self-Managed, GitLab SaaS, Bitbucket, or Bitbucket Data Center SCM server:

1. Log into Black Duck as a System Administrator.
2. 
Click **Admin** and select **Integrations**.
3. Click the applicable SCM Provider from the available options.
4. 
Click the **...** icon for the server and select **Edit**.
5. Edit any of the fields as desired.
6. Click **Update**.


Deleting SCM integrations


To delete a GitHub Enterprise, GitLab Self-Managed, GitLab SaaS, Bitbucket, or Bitbucket Data Center SCM server:

1. Log into Black Duck as a System Administrator.
2. 
Click **Admin** and select **Integrations**.
3. Click the applicable SCM Provider from the available options.
4. 
Click the **...** icon for the server and select **Delete**.
5. Click **Delete**.

19. Understanding how to search in Black Duck

You can search using the following categories:

- **Projects:** These are the projects that your company's developers are coding. For Black Duck to index basic project information for search, someone must create one or more projects.
 -  **Note:** The information that you provide about your projects is not shared outside of your company.
- **Components** and the **Black Duck KnowledgeBase:** These are the components that comprise your projects and are viewable in your BOM or it is the components in the Black Duck KnowledgeBase, which is a comprehensive database of open source software (OSS) components.
- **Vulnerabilities:** Security vulnerabilities that impact components and, as a result, which may impact your projects can be searched for by BDSA number, CVE number, or another identifier, for example, "CVE-2014-0160", "Heartbleed", and so on.


You can also perform a global search which searches for your term in all categories. Enter the term in the search field located at the top of the page and press **Enter** or click . Select a category to view the search results.

Note that entering a global search term initiates a new search and resets any filters you previously selected on the **Projects**, **Components**, **Vulnerabilities**, or **Black Duck KnowledgeBase** tabs.

Searching for projects

You can search for project versions that meet your search criteria.

To search for projects:

1. Click  to open the Find page and select the **Projects** tab.
2. Type your search term in the Search field.
3. Optionally, select any filters, as described in the next section, "Using search filters".
4. Optionally, [save this search](#), so that you can easily view them on your dashboard.

The Find page displays the project versions that meet your search criteria.

19. Understanding how to search in Black Duck • Searching for projects

Find

Projects Components Vulnerabilities Black Duck KnowledgeBase

3 Saved Searches Save

31 Results Found
Last Updated at 11:53 AM

Search Term

Clear filters Security Risk: High Security Risk: Critical

Sort by...

34056 1.0

1 Blocker Policy Violation 1 High Security Risk No License Risk 1 Medium Operational Risk

Group: Black Duck Project Groups 3 Components Last Scan: 9/15/2022 Updated: 9/21/2022 License: Unknown License Phase: In Development Distribution: External

anglebrackets 3.1.1

1 Major Policy Violation 2 Critical Security Risks No License Risk 310 High Operational Risks

Group: Black Duck Project Groups 498 Components Last Scan: 9/28/2022 Updated: 10/4/2022 License: Unknown License Phase: In Development Distribution: External

CallGraphProject unspecified

No Policy Violations 1 Critical Security Risk 4 Medium License Risks 15 High Operational Risks


Group: Black Duck Project Groups 23 Components Last Scan: 9/27/2022 Updated: 9/27/2022 License: Unknown License Phase: In Development Distribution: External

chang 1.0

No Policy Violations 1 High Security Risk No License Risk 1 High Operational Risk

Group: Black Duck Project Groups 1 Component Last Scan: 9/28/2022 Updated: 10/5/2022 License: Unknown License Phase: In Development Distribution: External

Citi_springframework_dryrun 5.3.22_after_iscan

You can also type your search term in the Search field located at the top of the application and press **Enter** or click . The Find page appears displaying the search results. Note that entering a global search term initiates a new search and resets any filters you previously selected. Select the **Projects** tab and filters to refine the results, as described below.

About the search results

Search results show all project versions that meet your search criteria. The following information is shown for each project version:

Sample Project 4.0

5 Blocker Policy Violations 1 High Vulnerability 5 High License Risks 11 High Operational Risks

14 Components Last Scan: 10/6/2020 Updated: 10/12/2020 License: Unknown License Phase: In Planning Distribution: External

- Use the bars to quickly view the number of components with the highest level of security, license, or operational risk.

For example, the following shows that while there is a component with lower risk, the highest security risk for this project version is High and that four components in this project version have a high level of security risk as their highest risk level:

1 High Vulnerability

- Hover over the bar to see the number of components for each risk category.


Security Risk

by Component




* Each component is counted once by its highest severity risk


In this example, there is one component that has a high risk level as their highest risk. 10 components that have medium risk as their highest risk level, and six components that have low risk as their highest risk level.

 **Note:** Each component is only counted once and is shown with its highest risk severity level.

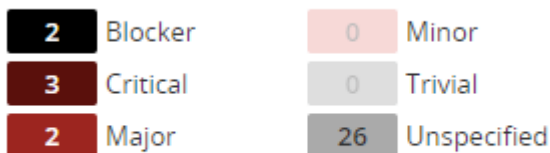
- Use the bar to see the number of components with the highest policy severity level for this project version.

For example, the following shows that while there are components with lower severity levels, the highest policy severity level for this project version is Blocker and there is one component that has Blocker as its highest policy severity level.

 1 Blocker Policy Violation

 **Note:** The text shown states the number of components with the highest policy severity level for this project version, not all policy severity levels affecting this project version.

- Hover over the bar to see the number of components with policy violations by severity level:

Policy Violations by Component

* Each component is counted once by its highest severity risk

- View the number of results found and the time the database was last updated:

14 Results Found
Last Updated at 1:27 PM

- For each project version, the search results also show:
 - Number of components in this project version.
 - Last scan date.
 - When this project version was last updated.
 - License of this project version.

19. Understanding how to search in Black Duck • Searching for projects

- Phase for this project version.
- Distribution of this project version.
- Select the project or version name to view the BOM.

Using search filters

If your search query returns many projects, use filters to narrow your results.

Note that:

- Where necessary, click + to display the filter values; click – to hide them.
- If you select more than one type of filter, Black Duck displays items that match *all* values. If you select more than one value for a specific filter, Black Duck displays items that match either value.

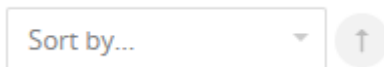
For example, if you use the License Risk filter and select high and medium, the search results display all projects that have high *or* medium license risk. If you select a high License Risk filter and a critical Security Risk filter, the search results display only those projects that meet have a high license risk *and* critical security risks.

Possible project filters are:

- Never Scanned. Select whether this project has never been scanned.
- Watching. Select whether this project is a watched project.
- Security Risk. Select one or more security risk levels.
- License Risk. Select one or more license risk levels.
- Operational Risk. Select one or more operational risk levels.
- Policy Rule. Select a policy rule from the list to find the projects that violate this policy.
- Policy Violations. Severity level of the policy rule.
- IaC Issues: Select from Open, Dismissed, or Not Reported.
- Last Scanned. Select a time period when this project was last scanned.
- Not Scanned Since. Select the time period since this project was last scanned.
- Project Group. Select one or more project groups.
- Release Phase. Select one or more release phases.
- Tags. Select one of more tags.


Sorting the search results

Optionally, you can sort the results that appear on the page by selecting a value from the **Sort by** list:



Note that if you sort the results and save this search, the Dashboard page displays the saved search in the sorted order.


Exporting to CSV

You can export your search results to CSV which converts the individual rows to tabular data. To do so, click the  button and select CSV.

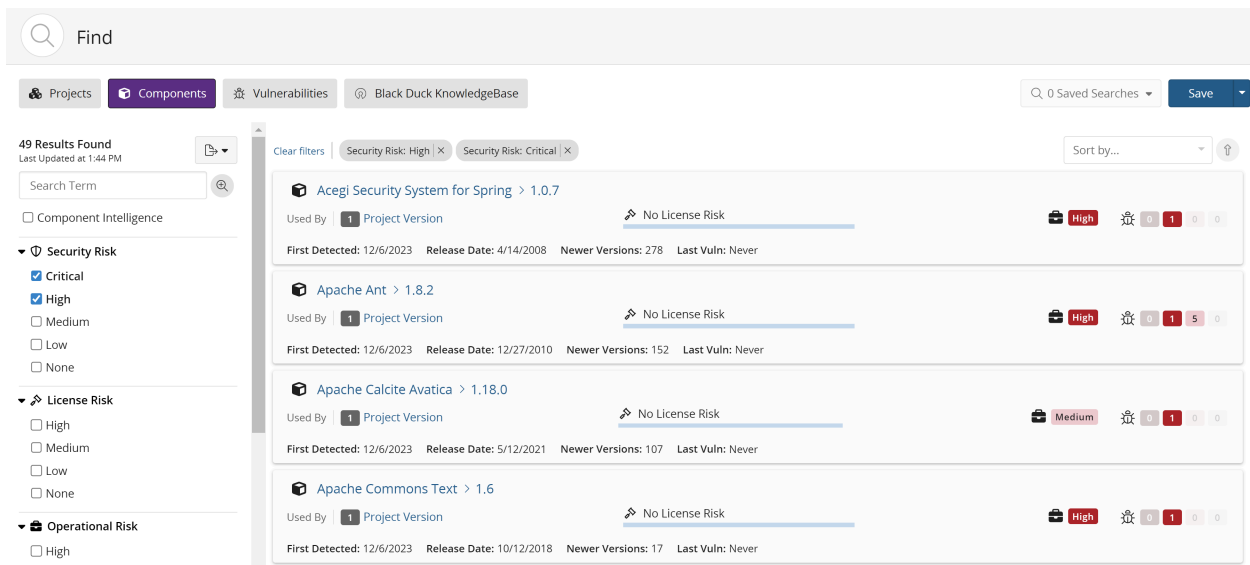
Searching for components

You can search for component versions used in your BOMs and/or components in the [Black Duck KnowledgeBase \(KB\)](#).


To search for components :

1. Click  to open the Find page.
2. Do one of the following:
 - Select the **Components** tab to find component versions used in your projects.
 - Select the **Black Duck KnowledgeBase** tab to search for Black Duck KnowledgeBase components. Note that using the Black Duck KnowledgeBase tab to search for components will not display any custom components used in your projects. Use the Components tab to include these in your search results.
3. Type your search term in the Search field and/or optionally, select any filters, as described in the next section, "Using search filters".
4. Optionally, for component searches, [save this search](#), so that the results appear on the Dashboard page.

The Find page displays the components that meet your search criteria.



You can also type your search term in the Search field located at the top of the application and press **Enter**

or click . The Find page appears displaying the search results. Note that entering a global search term initiates a new search and resets any filters you previously selected. Select the **Components** or **Black Duck KnowledgeBase** tab and filters to refine the results, as described below.

Using search filters

Filters that appear depend on whether you are searching for components used in your BOMs or searching Black Duck KnowledgeBase.


For each filter:

- Where necessary, click + to display the filter values; click – to hide them.
- If you select more than one type of filter, Black Duck displays items that match *all* values. If you select more than one value for a specific filter, Black Duck displays items that match either value.
For example, if you use the License Risk filter and select high and medium, the search results display all components that have high *or* medium license risk. If you select a high License Risk filter and a critical Security Risk filter, the search results display only those projects that meet have a high license risk *and* critical security risks.

KnowledgeBase filters

Use the following filters to narrow your results when searching Black Duck KnowledgeBase:

- Primary Language. Primary language in which the component is written. The filter displays the list of available languages in descending order of frequency of use in components.
- Tags. Available for all components that have tags applied to them to provide additional metadata about the component.
- Commit Activity. Represents the trending commit activity level for the open source component over time.

 **Note:** Primary language, tags, and commit activity information is provided by [Black Duck Open Hub](#).

Component filters

Use the following filters to narrow your results when searching components used in your BOM:


- Component Intelligence. Check to display all components containing suspicious events or incidents where it is highly likely that malware or malicious code has been identified. See [Operational Risk](#) for more information on Component Intelligence.
- Security Risk. Select one or more security risk levels.
- License Risk. Select one or more license risk levels.
- Operational Risk. Select one or more operational risk levels.
- First Detected. Date when the component was first detected by Black Duck (such as by scanning, being manually added to a BOM, and so on).
- License. Select a license from the list.
- License Family. Select a license family from the list.
- Missing Custom Field Data. Select to view the components and/or component versions which have required [custom fields](#) and are missing data.
- Released. Date when the component was released according to the Black Duck KnowledgeBase.

About the search results

Search results show all components that meet your search criteria.

Black Duck KB component search results

The following information is shown for each KnowledgeBase component that meets your search criteria:



Apache Struts Java struts.apache.org

The goal of the Apache Struts project is to encourage application architectures based on the "Model 2" approach, a variation of the classic Model-View-Controller (MVC) design paradigm. Under Model 2, a servlet (or equivalent) manages business logic execution, and presentation logic resides mainly in server pages.

Used By: **12** Project Versions Commit Activity: Decreasing Last Commit: 10/9/2020 Total Versions: 180 11 Tags

- Select the component name to open the [Black Duck KB Component Name](#) page.
- View the number of project versions that use this component as shown by the value next to **Used By**.

Used By | **2** Project Versions

Select **Project Versions** to open the Where Used dialog box.



Used in ×

 Apache MyFaces is being used in 1 Project Version

Project	Phase	Component Version	Security Risk
Sample Project - 4.0	In Development	2.1.5	0 0 1 0

Close

This dialog box lists the projects that use a version of this component.

Column	Description
Project	Name of the project and version that uses this component. Select the project name to display the project version's Components tab.
Phase	Project Phase .
Component Version	Version of this component used in this project version.
Security Risk	Lists the vulnerabilities for each severity level, from left to right: Critical, High, Medium, and Low. 0 3 28 11 Select a value to display the Security tab of Black Duck KB Component Name Version page, which lists the vulnerabilities associated with this version of the component.

- For each component, the search results show:
 - Commit Activity.
 - Last commit date.
 - Total number of versions for this component.
- Select **Tags** to view the tags for this component.
- The URL in the upper right corner is the URL for this component.

Components search results

The following information is shown for each component in your BOM that meets your search criteria.

- Select the component name/version to display the [Component Name Version](#) page.
- View the number of project versions that use this component version as shown by the value next to **Used By**.

Used By | **2** Project Versions

Select **Project Versions** to open the Where Used dialog box.

Project Name	Phase	License	Review Status	Security Risk
Sample Project - 4.0	In Planning	Apache License 2.0	Not Reviewed	0 3 6 0

This dialog box shows the project versions that use this version of the component.

Column	Description
Project Name	Name of the project and version that uses this component version. Select the project name to display the project version's Components tab.
Phase	Project Phase .
License	License for this component version.
Review Status	Whether this component has been reviewed in this project version.
Security Risk	Lists the vulnerabilities for each severity level, from left to right: Critical, High, Medium, and Low. <div style="text-align: center;"> 0 3 28 11 </div> Select a value to display the Security tab of the Black Duck KnowledgeBase Component Name Version page, which lists the vulnerabilities associated with this version of this component.

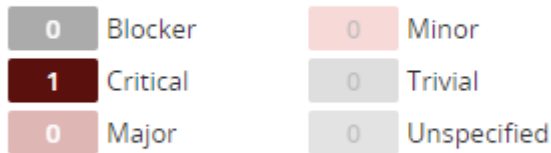
- Use the bar to quickly see the number of components with the highest policy severity level.

1 Critical Policy Violation

Select the bar to see the number of components with policy violations by severity level:

Policy Violations

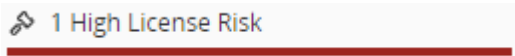
by Component



* Each component is counted once by its highest severity risk

Note: A component is only counted once with the highest policy severity level, not all policy severity levels affecting this component.

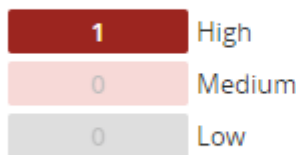
- Use the bar to quickly view the number of components with the highest level of license risk.



Select the bar to view the number of components in each risk category.

License Risk

by Component

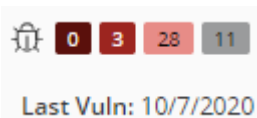


* Each component is counted once by its highest severity risk

- View the operational risk for this component version:



- View the number of vulnerabilities by severity associated with this component version. The **Last Vuln** date is when a vulnerability for this component was last updated in Black Duck (by the Black Duck KnowledgeBase or a user).



Select a value to display the **Security** tab of the Black Duck KB *Component Name Version* page, which lists the vulnerabilities associated with this version of this component.

Identifier	Published	Overall Score
> NVD CVE-2016-3081	Apr 26, 2016	9.3 High
> BDSA BDSA-2018-2905 (CVE-2018-11776)	Aug 22, 2018	8.3 High
> BDSA BDSA-2013-0027 (CVE-2013-4316)	Oct 8, 2018	7.4 High
> BDSA BDSA-2014-0067 (CVE-2013-2251)	May 24, 2018	6.5 Medium
> BDSA BDSA-2017-0903 (CVE-2017-9805)	Sep 6, 2017	6.2 Medium
> BDSA BDSA-2014-0117 (CVE-2014-0094)	Feb 19, 2019	6.2 Medium
> BDSA BDSA-2013-0028 (CVE-2013-1966)	Oct 9, 2018	6.2 Medium
> BDSA BDSA-2017-0367 (CVE-2017-9791)	Aug 9, 2017	6.2 Medium
> BDSA BDSA-2017-0031 (CVE-2017-5638)	Mar 10, 2017	6.2 Medium
> BDSA BDSA-2017-0954 (CVE-2017-12611)	Sep 11, 2017	6.2 Medium
> BDSA BDSA-2013-0052 (CVE-2013-2115)	Aug 14, 2019	6.2 Medium
> BDSA BDSA-2020-2097 (CVE-2019-0230)	Aug 18, 2020	5.9 Medium

- For each component version, the search results also show:
 - Approval status. Status indicates whether this component version has been reviewed.
 - First detected date.
 - Date this component version was released.
 - Number of newer versions.
 - Date when a vulnerability for the component was last updated in Black Duck (such as updates from Black Duck KnowledgeBase, a user manually changing the associated vulnerability, and so on).
- View the number of results found and the time the database was last updated:


14 Results Found
Last Updated at 1:27 PM

Sorting the search results

Optionally, you can sort the results that appear on the page by selecting a value from the **Sort by** list:

Note that if you sort the results and save this search, the Dashboard page displays the saved search in the sorted order.

Exporting to CSV

You can export your search results to CSV which converts the individual rows to tabular data. To do so, click the  button and select CSV.


Searching for vulnerabilities

You can search Black Duck for published security vulnerabilities. Searching by vulnerability is an efficient way to:

- Identify if a new or existing security vulnerability affects a component that is included in your projects.

- Review the severity of the security vulnerability to determine if remediation is required.
- Create a custom vulnerability dashboard so that you can focus on the vulnerabilities that are important to you.

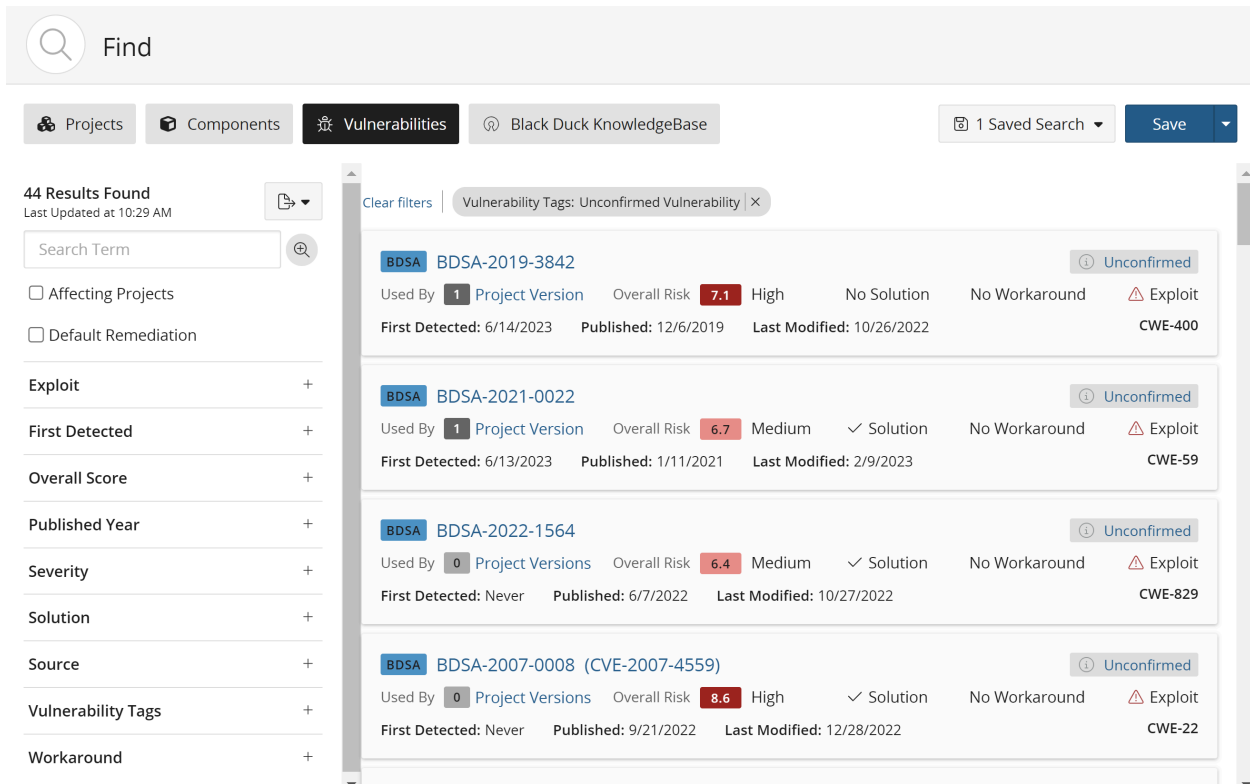
To search for vulnerabilities:

1. Click  to open the Find page and select the **Vulnerabilities** tab.
2. Optionally, type your search term in the Search Term field.
3. Optionally, select any filters, as described in the next section, "Using search filters".

Note that you can enter a search term only, include filters with the search term, or just search using filters.

4. Optionally, [save this search](#), so that the results appear on the Dashboard page.

The Find page displays the vulnerabilities that meet your search criteria.




The screenshot shows the 'Find' page in Black Duck. At the top, there is a search bar with a magnifying glass icon and the word 'Find'. Below the search bar are navigation tabs: 'Projects', 'Components', 'Vulnerabilities' (which is selected), and 'Black Duck KnowledgeBase'. To the right of these tabs, there is a '1 Saved Search' dropdown and a 'Save' button.

On the left side, there is a sidebar with the following information:

- 44 Results Found
- Last Updated at 10:29 AM
- A search term input field with a magnifying glass icon.
- Two checkboxes: 'Affecting Projects' and 'Default Remediation', both of which are unchecked.
- A list of filter categories, each with a '+' sign: 'Exploit', 'First Detected', 'Overall Score', 'Published Year', 'Severity', 'Solution', 'Source', 'Vulnerability Tags', and 'Workaround'.

The main content area displays a list of four vulnerability entries. Each entry has a 'Clear filters' link and a 'Vulnerability Tags: Unconfirmed Vulnerability' tag. The entries are:

- BDSA-2019-3842**: Overall Risk 7.1 (High), No Solution, No Workaround, Exploit. First Detected: 6/14/2023, Published: 12/6/2019, Last Modified: 10/26/2022. CWE-400.
- BDSA-2021-0022**: Overall Risk 6.7 (Medium), Solution checked, No Workaround, Exploit. First Detected: 6/13/2023, Published: 1/11/2021, Last Modified: 2/9/2023. CWE-59.
- BDSA-2022-1564**: Overall Risk 6.4 (Medium), Solution checked, No Workaround, Exploit. First Detected: Never, Published: 6/7/2022, Last Modified: 10/27/2022. CWE-829.
- BDSA-2007-0008 (CVE-2007-4559)**: Overall Risk 8.6 (High), Solution checked, No Workaround, Exploit. First Detected: Never, Published: 9/21/2022, Last Modified: 12/28/2022. CWE-22.

You can also perform a global search by typing your search term in the Search field located at the top of the application and pressing **Enter** or clicking . If not displayed, select the **Vulnerabilities** tab to view your results. Note that entering a global search term initiates a new search and resets any filters you previously selected.

Using search filters

For each filter:

- Where necessary, click **+** to display the filter values; click **-** to hide them.


19. Understanding how to search in Black Duck • Searching for vulnerabilities

- If you select more than one type of filter, Black Duck displays items that match *all* values. If you select more than one value for a specific filter, Black Duck displays items that match either value.


For example, if you use the remediation status filter and select new and needs review, the search results display all vulnerabilities that have a remediation status of new *or* needs review. If you select a remediation status of new and a security filter of high, the search results display only those vulnerabilities that meet have a remediation status of new *and* a high security level.

Use the following filters to narrow your results when searching for vulnerabilities:

- Affecting projects. Selecting this filter searches for vulnerabilities in your projects only. Clearing this filter searches Black Duck KnowledgeBase and your projects.
- Default Remediation. Selecting this filter displays vulnerabilities that are [automatically remediated](#).
- Exploit. Select whether an exploit is available for a vulnerability.
- First Detected. When the vulnerability first appeared in a BOM.
- Overall Score. Enter the minimum overall score value; Black Duck displays vulnerabilities that have this score or higher.
- Published Year. Year the vulnerability was published.
- Severity. The severity levels shown depend on the [selected security configuration](#) as CVSS v2 does not have a critical security level.
- Solution. Select whether a solution is available for a vulnerability.
- Source. BDSA or NVD.
- Vulnerability Tags. Select one or more [vulnerability tags](#).

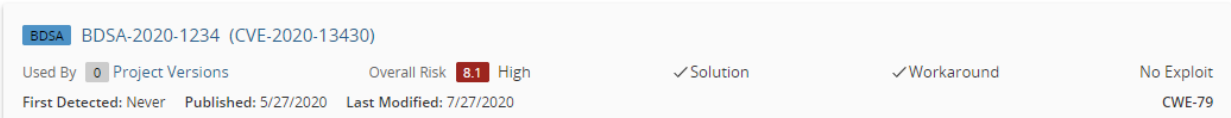
 **Note:** If searching for CISA Known Exploited Vulnerabilities, you must also check the Affecting Projects checkbox to display results.

- Workaround. Select whether a workaround is available for a vulnerability.

 **Attention:** Filter options and tags are not all supported for KnowledgeBase vulnerability queries and some queries are only supported for local vulnerabilities affecting project versions (using the checkbox). This means different behavior may be observed when using the Find page with and without the **Affecting Projects** checkbox enabled.

About the search results

Search results show all vulnerabilities that meet your search criteria. The following information is shown for each vulnerability:



BDSA BDSA-2020-1234 (CVE-2020-13430)				
Used By 0 Project Versions	Overall Risk 8.1 High	✓ Solution	✓ Workaround	No Exploit
First Detected: Never	Published: 5/27/2020	Last Modified: 7/27/2020		CWE-79

- Select the vulnerability ID to view more information on the vulnerability, such as additional score values. You can view National Vulnerability Database (NVD) information by selecting the [CVE number](#) or view Black Duck Security Advisory (BDSA) information by selecting the [BDSA number](#).
- View the number of project versions that affected by this vulnerability next to **Used By**.

Used By | **2** Project Versions

Select **Project Versions** to open the **Affected Projects** tab for the vulnerability which lists the project versions affected by this vulnerability.

Black Duck Security Advisory
Apache HTTPClient Vulnerable to Man-In-The-Middle (MITM) Attack via SSL Hostname Verification Bypass
 BDSA-2014-0126 | CVE-2014-3577 | Published May 30, 2019 | Updated Feb 7, 2020

Overview **Affected Projects** Technical CVE References Settings

Remediate

Project	Component	Component Origin	Status	Target date	Actual date
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 3.1	maven/commons-httpclient:commons-httpclient:3.1	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpClient 4.3.3	maven/org.apache.httpcomponents:httpclient:4.3.3	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.2	maven/org.apache.httpcomponents:httpcore:4.3.2	New	Never	Never
cloudfoundry-identity-parent 3.6.3	Apache HttpComponents Core 4.3.3	maven/org.apache.httpcomponents:httpcore:4.3.3	New	Never	Never

Displaying 1-4 of 4

- View the overall risk score. The search results show the Temporal Score for BDSA vulnerabilities or the Base Score for NVD vulnerabilities and the associated risk level. Note that the score shown and risk level depends on the [selected security rankings](#).

Select the score to view individual scores: temporal, base, exploitability, and impact for BDSA; base, exploitability, and impact for NVD.

- View whether a solution, workaround, or exploit is available:
 - ✓ indicates that there is a solution or workaround available for this vulnerability.
 - ⚠ indicates there is an exploit for this vulnerability.
- For each vulnerability, the search results also show:
 - First Detected.
 - Published date.
 - Last modified date. Note that this date displays the last time the vulnerability was modified in the KnowledgeBase. It does not necessarily mean the vulnerability information was updated itself.
 - Common Weakness Enumeration (CWE) number for this security vulnerability.

Note: Search results are limited to a maximum of 10,000 items.

Exporting to CSV

You can export your search results to CSV which converts the individual rows to tabular data. To do so, click the button and select CSV.

Saving and managing search results

Black Duck gives you the ability to save your search results. This lets you search for projects, components, or vulnerabilities using a variety of attributes, save those searches, and then view dashboards of those saved searches so that you can quickly view the information that is relevant to you.

19. Understanding how to search in Black Duck • Saving and managing search results

The screenshot shows the Black Duck Dashboard interface. At the top, there's a 'Dashboard' header with a menu icon and a 'Summary' link. Below the header, there are tabs for 'Projects' (Watching, My Projects, SCM Projects, Critical security risk projects) and 'Saved Searches'. The main content area is titled 'My Projects' and displays three project cards: 'Sample Project', 'Project Test', and 'webgoat-parent'. Each card shows risk metrics like 'No Policy Violations', '2 Critical Security Risks', '2 High License Risks', and '2 High Operational Risks'. To the right, a 'Results Summary' section shows four donut charts: 'Policy Violations' (0% Blocker, 0% Critical, 0% Major, 0% Minor, 0% Trivial, 0% Unspecified, 100% None), 'Security Risk' (67% Critical, 0% High, 0% Medium, 0% Low, 33% None), 'License Risk' (33% High, 0% Medium, 0% Low, 67% None), and 'Operational Risk' (100% High, 0% Medium, 0% Low, 0% None). The text 'Displaying 1-3 of 3' is visible at the bottom of the project cards.

Saving search results

1. After using the Find page to create the search results you wish to view on your Dashboard, click **Save** on the Find page. The Save Search dialog box appears.


The 'Save Search' dialog box is shown. It has a title bar with the text 'Save Search' and a close button (X). Below the title bar is a label 'Name' followed by an empty text input field. Below the input field is a note: 'Saved searches appear in your dashboard'. At the bottom of the dialog are two buttons: 'Cancel' and 'Save'.

2. Enter a name for these search results and click **Save**.


This saved search now appears in the list of saved searches on your Dashboard page.

- i Tip:** You can also use an existing saved search as a basis for a new saved search. Select a search from the list of saved searches and optionally modify any filters. Click **Save New** and specify a new name for this search. This gives you a new saved search while your existing saved search is not modified.


Editing saved searches

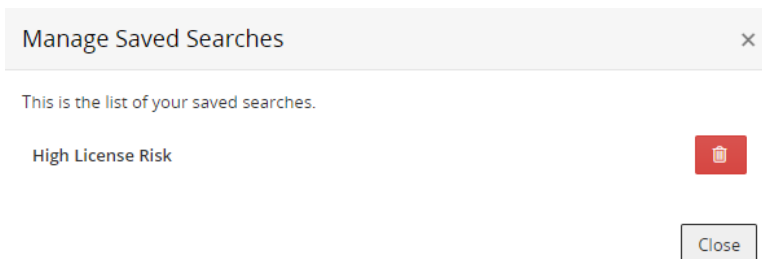
1. Click . The Find page appears.
2. Select the saved search you wish to modify from the list of saved searches. The Find page displays the search results for that saved search.
3. Optionally, modify the filters for this saved search.
4. Click **Update**.


Renaming saved searches

1. Click . The Find page appears.
2. Select the saved search you wish to modify from the list of saved searches. The Find page displays the search results for that saved search.
3. Optionally, modify the filters for this saved search.
4. Click **Update** and select **Rename** from the menu. The Rename Saved Search dialog box appears.
5. Enter the new name of this saved search.
6. Click **Save**.

Deleting saved searches

1. Click . The Find page appears.
2. Select **Manage** from the list of saved searches. The Manage Saved Searches dialog box appears.



3. Click  in the row of the saved search you wish to delete. The saved search is removed from the list of saved searches and from your Dashboard page.
4. Click **Close**.

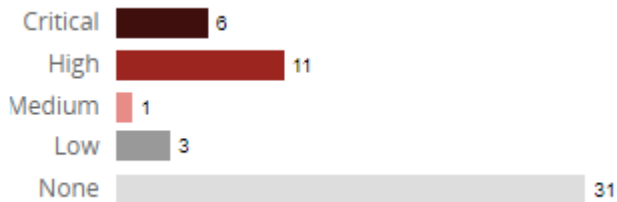
Filtering the data shown in tables

Risk graphs and/or advanced filters are available on some pages to help you filter the information shown in tables.

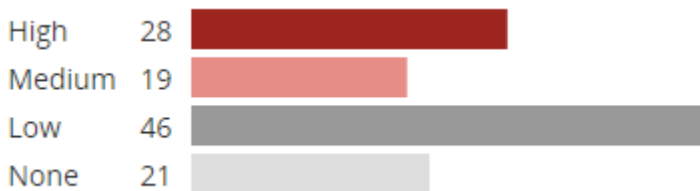
Risk Graphs

Some pages display risk graphs which indicate the number of items in the table shown below the graphs that have that security, license, and/or operational risk at that severity level.

For security risk:



For license and operational risk:



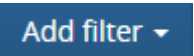
- **Critical** risk: 50% black and 50% red (security risk only)
- **High** risk: 100% red
- **Medium** risk: 50% red
- **Low** risk: 100% gray
- **None**: 50% gray

Select a severity label/graph to filter the table to show only those items that have a specific type and severity of risk.

Advanced Filters

On some pages, tables have an advanced filter feature that provides an easy way to view the data. This feature provides you with a clear view of the filters that are being applied to the table.

To use advanced filters:

1. Click  to view the filters for this table.
2. Select a filter. The filter you selected appears at the top of the table.





3. Select values for this filter and click **OK**. If you select more than one value, Black Duck displays items that match *either* value.
4. Optionally, select additional filters. If you select more than one type of filter, Black Duck displays items that match *all* filters.
5. Black Duck displays items that match all selected filters. For example:

The screenshot shows the Black Duck Dashboard with three risk charts on the left and a table of components on the right. The charts are for Security Risk, License Risk, and Operational Risk. The table has columns for Component Name, Versions, Used count, Security Risk, License Risk, and Operational Risk. A filter is applied to the License Risk column, showing 'None' selected. The table lists components like Kerberos, Angular, Python programming language, SQLite, Lo-Dash, GNU Compiler Collection, PostgreSQL Database Server, morris.js, mixin-deep, JS-YAML, Growl, Scala, minimatch, randomatic, and set-value.


Component Name	Versions	Used count	Security Risk	License Risk	Operational Risk
> Kerberos	2 Versions	2	High	None	Low
Angular	1.3.0-beta.11	1	High	None	Low
Python programming language	2.7.16	1	High	None	Low
SQLite	3.28.0	1	High	None	Low
> Lo-Dash	4 Versions	5	High	None	Low
> GNU Compiler Collection	2 Versions	3	High	None	Low
> PostgreSQL Database Server	4 Versions	41	High	None	Low
morris.js	0.4.3	1	High	None	Low
> mixin-deep	2 Versions	3	High	None	Low
> JS-YAML. Native JS port of PyYAML.	2 Versions	3	High	None	Low
> Growl	2 Versions	2	High	None	Low
> Scala	2 Versions	2	High	None	Low
> minimatch	3 Versions	7	High	None	Low
> randomatic	3 Versions	3	High	None	Low
> set-value	3 Versions	5	High	None	Low

6. Click **X** to remove a filter.


Tip: For pages that have advanced filters and risk charts, advanced filters work with these charts so you can also select and/or clear multiple risk filters by using the graphs. Selecting a risk level displays a filter () icon in the graph and a field appears above the table displaying the values you selected. Click  in the risk graphs or **X** in the filter fields to clear the filter.

20. Working with notifications

Notifications alert you when:


- Security vulnerabilities are published or updated against components that are included in one or more of your projects.
 - [Estimated Security Risks](#) that have been added or removed from components without a version.
 - Actions you perform affect the vulnerabilities in BOM components, such as:
 - Editing, adding, or removing components which have vulnerabilities.
 - Unmapping a scan from a project.
 - Rescanning code or a Docker image.
 - Ignoring or no longer ignoring a component.
 - Modifying file(s) so that they are matched to a different component.
 - Components have violated a policy.
 - Policy violations have been overridden.
 - Components no longer violate a policy.
 - You are approaching or are exceeding your [code size limit](#).
-  **Tip:** You can [remove projects you are watching](#) so that you do not receive notifications for those projects or components in those projects.

Viewing notifications

1. Open the notifications list by selecting .
The list displays all new and previously viewed notifications.
2. To manage the notifications, select **See All Notifications** located at the bottom of the list.
The All Notifications page appears.
3. By default, the page is filtered. Select **Add Filter** to [change these settings](#).

Viewing more information

To view more information on security vulnerabilities and BOM component adjustments:

1. Log in to Black Duck.
2. Open the notifications list by selecting  and select **See All Notifications**.
 - Select a component version to open the **Security** tab of the Black Duck KB [component version page](#).

- Select a vulnerability record (such as CVE-2017-1234) to view the vulnerability details page for that security vulnerability.

To view more information on policy violations and overrides:

1.



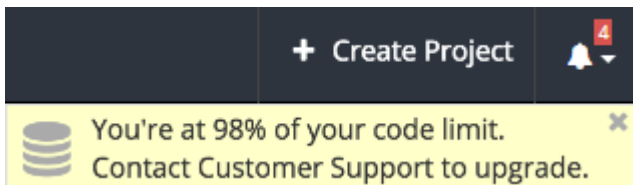
Open the notifications list by selecting  and select **See All Notifications**.

2. Select a policy violation or a policy violation override to open the BOM page.

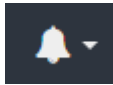
Users with the appropriate role can [override a policy violation](#) or [remove a policy violation that was overridden](#).

To view more information on code limits:

The notification automatically appears at the top of the page when you are close to exceeding your [code size limits](#):



1.



Open the notifications list by selecting .

2. Select **See All Notifications** located at the bottom of the list.

The All notifications page appears.

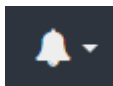
3. To upgrade your code limit, contact Customer Support.

Hiding notifications

You can hide notifications so that they no longer appear in the drop-down list and appeared grayed out on the All Notifications page.

1. Log in to Black Duck.

2.



Open the notifications list by selecting .

3. Click **X** located in the upper right corner of a notification.

You can also use the same method in the All Notifications page.

Click  located in the upper right corner of a notification in the All Notifications page to redisplay the notification.

21. About the Tools page

The Tools page provides download links and links to Black Duck integrations on GitHub, the Black Duck Software Integrity Community, and Black Duck Software Integrity, Customer Education web pages. It is divided into these sections: **Downloads**, **Black Duck Open Source Integrations**, and **Community and Education**, as described below.

To access the Tools page, from the user menu located on the top navigation bar, select **Tools**.

Downloads

This section of the Tools page provides links for Black Duck Detect (Desktop), Black Duck Detect CLI, and Legacy Downloads (the Signature Scanner).


- **Black Duck Detect (Desktop)**. Select the link to download the Mac OS X, Linux, or Windows version of Black Duck Detect (Desktop) from Google Cloud Storage. This tool scans your file system and generates a Bill of Materials (BOM).

Black Duck Detect (Desktop) client systems must meet the following requirements:

- Mac OS X. Version 10.10 or later. A minimum of 8 GB of RAM.
- Windows. Windows 10. A minimum of 8 GB of RAM.
- **Black Duck Detect(CLI)**. Select the link to go to the Integrations Documentation page. From here, select **Black Duck Detect** to view download instructions and documentation for Black Duck Detect, a command line interface (CLI) that integrates with your build jobs to identify package manager dependencies as well as file system matches.
- **Legacy Downloads**. Select **Toggle All** to view the download links for the Linux, Mac OS X, or Windows CLI of the Signature Scanner, a tool to scan your file system and generate a BOM.


The Signature Scanner client systems must meet the following requirements:

- macOS x64. Version 10.10 or later. A minimum of 8 GB of RAM.
- macOS arm64. A minimum of 8 GB of RAM.
- Windows. Windows 10. A minimum of 8 GB of RAM.
- Linux. A minimum of 8 GB of RAM on the supported operating systems.

 **Note:** The Signature Scanner is included with Black Duck Detect. We recommend you use Black Duck Detect to create a more complete Bill of Materials.

Black Duck Open Source Integrations

Clicking this link opens the Black Duck pages on [GitHub](#).

To view Black Duck integrations documentation, from the help menu () located on the top navigation bar, select **Integrations Documentation**.

Community and Education

This section of the Tools page provides the following links:

- **Black Duck Community**. Clicking this link on the Tools page displays an [online resource for customer support, solutions, and information](#).

- **Black Duck Customer Education.** Clicking this link on the Tools page opens a web page that provides more information on education courses for Black Duck Integrity Group products.

22. Black Duck C/CPP Tool

C and C++ projects don't have a standard package manager or method for managing dependencies. It is therefore more difficult to create an accurate BOM for these projects. This leaves Software Composition Analysis tools fewer options than with other languages. The primary options which are available in this context are: file system signatures. Black Duck has a variety of old and new signatures which can be used to build a BOM. In order to effectively use signatures, the tool first needs to know which files to take signatures from. In the past SCA tools have pointed a scanner at a build directory, getting signatures from a subset of files within the directory sub-tree. The problem with this approach is that there are many environmental variables, parameters and switches provided to the build tools, which make reference to files outside of the build directory to include as part of the build. Further, there are, commonly, files within the build directory, which are not part of the build and can lead to false positives within the BOM.

The new Black Duck C/CPP tool avoids the pitfalls described above by using a feature of Coverity called Build Capture. Coverity Build Capture, wraps your build, observing all invocations of compilers and linkers and storing the paths of all compiled source code, included header files and linked object files. These files are then matched using a variety of methods described in the section of this document called "The BOM".

This overview is an excerpt from the documentation that can be found in the [Black Duck C/CPP Tool Documentation Portal page](#) along with other details on how to use the Blackduck-C-CPP tool.

23. Integrating Protex with Black Duck

Black Duck provides the ability to import Protex BOMs into Black Duck.

This feature gives Protex users the ability to use Black Duck to view and manage security vulnerabilities in their existing BOMs. It also provides Black Duck customers the ability to use the greater language support that is available in Protex.

There are three basic methods for importing Protex data into Black Duck:

- Components Only


This option is akin to the mapping that is currently done between Protex and Code Center – the BOM in Code Center only has the list of component/versions and not any of the associated file mappings. Similarly, using this technique to import a Protex BoM into Black Duck only preserves the components/versions. As only the component and version information is being mapped, there is less of a performance impact compared to the other methods.

This is the default output of the [Protex BOM Tool](#).

- Components and Files


This method maps the existing Protex BOM into a comparable BOM within Black Duck, preserving the identified components and the associated file mappings. Note that the resultant BOM in Black Duck is only as good as the identifications that were made manually in Protex, therefore, it is important that the people doing the identification work in Protex pay attention to the versions they are selecting for each component. Historically, for license compliance, having the correct version for a component was less important as licenses rarely changed between versions of the same component. However, for security risk, having the correct version for a component is very important as vulnerabilities are mapped to specific versions of components. Therefore, if you will be using Protex with Black Duck, it is important for you to be aware of this as you are doing your identification work.

The [Protex BOM Tool](#) can export a BOM from Protex and import it directly into Black Duck, mapping it to a specific project and release. Or, the tool can be used to export the BOM into a JSON file which can be later imported into Black Duck using the Black Duck UI.

 **Note:** The component and version identifiers are different between the Protex KB and Black Duck KB. During the import process, Black Duck application will remap each BOM component/version from its Protex KB identifier to the corresponding Black Duck KB identifier. Not all components will have a KB identifier and will therefore not be reflected in Black Duck BOM, for example, custom or local components, or components that do not have a corresponding ID in Black Duck KB.

An [audit log](#) lists the Protex components and licenses that were mapped to Black Duck and provides details around any items that were unable to be mapped between the Protex KB and the Black Duck KB.

To use this method, include the **--include-files** parameter when running the [Protex BOM Tool](#).


 **Note:** Due to the amount of file information contained in many Protex BOMs, there may be some performance impact both during the import process and when navigating to UI pages involving these projects.

- File Metadata > Black Duck Signatures

This method takes the original file metadata that was captured during the Protex scan and imports it into Black Duck such that Black Duck treats it as if the scanner was scanning the files and directories directly. A new Black Duck BOM is created which will likely be different from the original Protex BOM. As the scanner takes advantage of the full context of file and directory information, it can identify the

correct version information for a component. Thus, in many cases you will see more accurate version information using this method and get better results for security use cases.


To use this method, use the `--dryRunWriteDir` and `--include-files` parameters when running the [Protex BOM Tool](#).

 **Note:** For the best results using this approach, archives need to be expanded when running the Protex scan. This may produce longer scan times for some projects depending on the number of archives in the project.

Understanding the Protex BOM integration process

The process for integrating a Protex BOM into Black Duck is:

1. Log in to Black Duck.
2. [Download](#) and install the Protex BOM tool. The Protex BOM tool provides several different ways by which you can import a Protex BOM into Black Duck.
3. [Export](#) the Protex BOM file.

 **Note:** Only projects assigned to the user whose credentials are supplied in the tool will be available for export.

4. If you do not use the Protex BOM tool to import the BOM into Black Duck or map the BOM to a project, then use Black Duck UI to:
 - [Import](#) the Protex BOM file into Black Duck.
 - [Map](#) the Protex BOM to a Black Duck project.

Once the Protex BOM is imported and mapped, you can [view and manage](#) its contents as you manage any other BOM in Black Duck.

Requirements

To import a Protex BOM into Black Duck, you must be running:

- Protex version 7.1.2 or higher
- Black Duck version 2.3 or higher

Note the following:

- Imported Protex data is processed in Black Duck and the Black Duck KB through a new KnowledgeBase matching service. This service converts all Protex Suite IDs to Black Duck KnowledgeBase IDs.
- Matched and unmatched file information is available in Black Duck. The following table lists the Protex discovery type, usage, and the corresponding Black Duck match type:

Protex Discovery Type	Protex Usage	Black Duck
*	Component	Exact
Code Match	File	Exact
Code Match	Snippet	Partial
String Search	Snippet	Partial
Dependency	Snippet	Dependency

- The following Protex BOM components are not available in Black Duck:

- Custom Components
- Custom Licenses

These components are dropped during the import process.

- If you use Protex to make any changes to the Protex BOM, the changes persist when the Protex BOM is reimported to Black Duck: only the changes made in the imported Protex BOM are updated in the Black Duck project.

Downloading the Protex BOM tool

The Protex BOM tool command line interface (CLI) client is packaged as a .zip file. To obtain the zip file, please contact Black Duck Support at <https://community.blackduck.com/s/contactsupport>.

After you unzip the Protex BOM tool client file, use it to [import a Protex BOM](#) into Black Duck.

Exporting a Protex BOM

The Protex BOM tool provides several different ways by which you can import a Protex BOM into Black Duck.

For example, you can use the tool to:

- [Export the Protex BOM from the Protex server](#) and import it into Black Duck using the tool.
- [Export the Protex BOM from the Protex server to a file](#) and manually import it through Black Duck UI.
- [Import a Protex BOM file](#) into Black Duck using the tool.

The tool does not require any specific role in Black Duck or in Protex to use the tool.

By default, the tool outputs component/version data only; use the **--include-files** parameter to include file data.

The Protex BOM tool has these parameters:

Parameter	Description
-?, --help	Shows help for this tool.
-A, ---dest <host: port>	Specifies Black Duck host name and port.
-P, --hub-project <name>	Specifies the name of the Black Duck project to which you want to map this Protex BOM. If the project does not exist, the tool creates the project and maps the BOM to the project.
-R, --hub-release <name>	Specifies the name of the Black Duck project version to which you want to map this BOM. If the version does not exist, the tool creates the version and maps the BOM to this version of the project. If you specify hub-project , hub-release is optional. If you do not specify hub-release , the version defaults to the value of the release parameter.
-S, --secure-dest	Uses HTTPS to connect to the server hosting Black Duck. If you do not include this parameter, HTTP is used.

Parameter	Description
-U, --dest-user <user>	Specifies the username to log in to the Black Duck server.
-W, --dest-password	Forces the tool to prompt you for a password for the Black Duck server. When the tool runs, a prompt appears requesting the password for the specified user. For non-interactive use, set the <code>BD_HUB_PASSWORD</code> environment variable with the password for the Black Duck server. If you set this variable, the dest-password parameter is optional: the tool prompts the user for the password; it does not check the password against the variable.
-a, ---address <host:port>	Specifies the Protex host name and port.
-r --release <name>	Specifies a value to use to identify the current state of the Protex BOM. You can use any value for <name>. Use this parameter to enable viewing multiple "versions" of a Protex BOM in Black Duck. Click here or more information.
--list-projects <SearchQuery>	Lists all Protex project identifiers for all projects to which you have access, one per line, on the console. <SearchQuery> is optional. To export multiple Protex projects, use the output from this parameter to write a script which iterates over multiple project identifiers.
--data <path>	Specifies the path to the Protex BOM file.
--output <path>	Writes the BOM out to a file or directory with the project name.
-p, --project <id or name>	Specifies the Protex project identifier or project name.
-s, --secure	Uses HTTPS to connect to the server hosting Protex. If you do not specify this parameter, HTTP is used.
-u, --user <user>	Specifies the username to log in to the Protex server.
-w, --password	Forces the tool to prompt you for a password. When the tool runs, a prompt appears requesting the Protex server password for the specified user. For non-interactive use, set the <code>BD_PROTEX_PASSWORD</code> environment variable with the password for the Protex server. If you set this variable, the password parameter is optional.
-V, --version	Shows the version information of this tool.
-v, --verbose	Sets the logging level to verbose.
--dryRunWriteDir <dryRunWriteDir>	Specifies the directory to which the Protex BOM Tool outputs a JSON file with the original file metadata used for scanning.
--debug	Shows debug output.
--include-files	Includes the Protex code tree and match details.

By default, the tool generates the Protex BOM to standard out, if you don't specify an output (file) or use the tool to import the BOM to Black Duck.

Exit Statuses

The possible exit statuses are:

- **0:** SUCCESS. The export completed successfully.
- **1:** FAILURE. Generic failure.
- **64:** USAGE. The command to run the tool was used incorrectly, for example, with the wrong number of arguments or a bad syntax.
- **65:** DATA_ERROR. The input data was incorrect.
- **66:** NO_INPUT. An input file (not a system file) did not exist or was not readable.
- **67:** NO_USER. The specified user does not exist.
- **68:** NO_HOST. The specified host does not exist.
- **69:** UNAVAILABLE. A service is unavailable.
- **70:** SOFTWARE. An internal software error has been detected.
- **71:** OS_ERROR. An operating system error has been detected.
- **72:** OS_FILE. A system file does not exist, cannot be opened, or has some sort of error, for example a syntax error.
- **73:** CANNOT_CREATE. An output file cannot be created.
- **74:** IO_ERROR. An error occurred while doing input/output on a file.
- **75:** TEMPORARY_FAILURE. Temporary failure,
- **76:** PROTOCOL. The remote system returned something that was "not possible" during a protocol exchange.
- **77:** NO_PERMISSION. You did not have sufficient permission to perform the operation.
- **78:** CONFIGURATION. Something was found in an unconfigured or misconfigured state.
- **79:** NO_REGISTRATION. Registration to Black Duck or Protex was not valid.

Viewing multiple versions of a Protex BOM in Black Duck

When you import a Protex BOM, Black Duck creates a file (labeled a BOM File in Black Duck UI) that is associated with that BOM. In Black Duck, a BOM File can only be mapped to a single project and version – if you import the Protex BOM again, the new file is added to the existing BOM File.

You may want to view multiple versions, or snapshots, of a Protex BOM in Black Duck. Although Protex does not have project versions, you can use the **release** parameter in the Protex BOM tool to denote a snapshot of your Protex BOM. When you use the **release** parameter, Black Duck creates a new BOM file for that snapshot. You can then map that BOM file to a different project or to a different version of a project. This gives you the flexibility to create multiple snapshots of a single Protex BOM and view them at the same time in Black Duck.

Note that if you specify a value for **release** that has already been used for that Protex BOM, a new BOM File is not created. Instead, the new file will be added to the existing BOM File.

Examples

The following are examples of using the Protex BOM tool:

- [Exporting the Protex BOM and importing it into Black Duck using the export tool](#)

- [Exporting the Protex BOM to a file](#)
- [Importing a Protex BOM from a file](#)

Note that the examples show the required parameters.

Using the Protex BOM tool to map the Protex BOM

In these examples, you have the option of using these parameters to specify the Black Duck project and version that this BOM should be mapped to:

- **hub-project** <name>
- **hub-release** <name>

If you specify a value for the **release** parameter and wish to use the tool to map the Protex BOM, the **hub-release** parameter is optional: if you do not specify a value for **hub-release**, Black Duck project version defaults to the value of **release**.

If you do not specify **hub-project** and **release** or **hub-release**, you must [map the Protex BOM](#) using the Black Duck UI.

Exporting the Protex BOM and importing it into Black Duck using the export tool

This example exports the Protex BOM from the Protex server and imports it into Black Duck using the tool.

1. Open a command prompt.
2. Go to the directory where the tool is installed and run the following command:

Linux example

```
./scan.protex.cli.sh --address <host:port> --user <user> --password --project <id>
--output <path> --dest-address <host:port> --dest-user <user> --dest-password
```

Windows example

```
scan.protex.cli.bat --address <host:port> --user <user> --password --project <id> --
output <path> --dest-address <host:port> --dest-user <user> --dest-password
```

Exporting the Protex BOM to a file

This example exports the Protex BOM from the Protex server to a JSON file. You then need to use the Black Duck UI to [manually import the file](#).

1. Open a command prompt.
2. Go to the directory where the tool is installed and run the following command:

Linux example

```
./scan.protex.cli.sh --address <host:port> --user <user> --password --project <id>
--output <path>
```

Windows example

```
scan.protex.cli.bat --address <host:port> --user <user> --password --project <id> --
output <path>
```

Importing a Protex BOM from a file

This example imports a Protex BOM file into Black Duck using the tool.

1. Open a command prompt.
2. Go to the directory where the tool is installed and run the following command:

Linux example

```
./scan.protex.cli.sh --data <path> --dest-address <host:port> --dest-user <user> --dest-password
```

Windows example

```
scan.protex.cli.bat --data <path> --dest-address <host:port> --dest-user <user> --dest-password
```

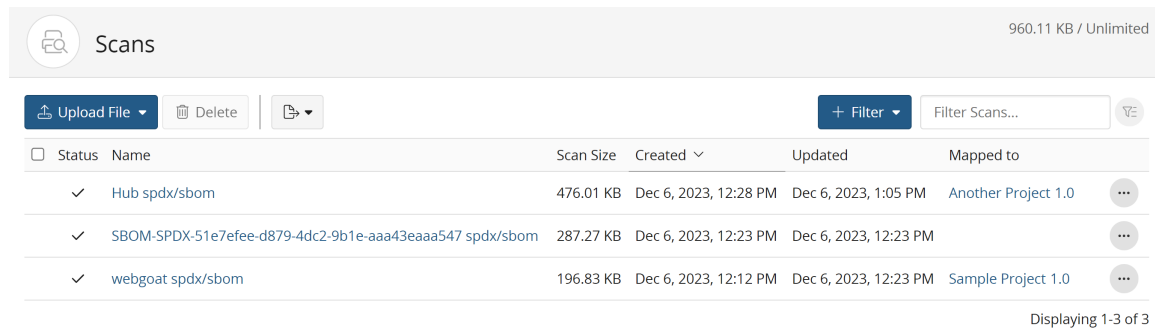
Importing the Protex BOM file

If you output the Protex BOM to a file, you need to import the file into Black Duck.

To import a Protex BOM file:

1. Log in to Black Duck.

2.  Click **Scans**.



Scans 960.11 KB / Unlimited

Upload File Delete Filter Scans...

Status	Name	Scan Size	Created	Updated	Mapped to
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM	
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0

Displaying 1-3 of 3

3. In the Scans page, click **Upload Files**.
4. Use the Upload Files dialog box to locate the Protex BOM file
5. Click **Close**.

If you did not use the Protex BOM tool to automatically map the BOM to a project, use Black Duck to [map the file to a project](#).

Mapping or unmapping a Protex BOM

You must use Black Duck to map the Protex BOM to a project if you did not use the Protex BOM tool to do so.

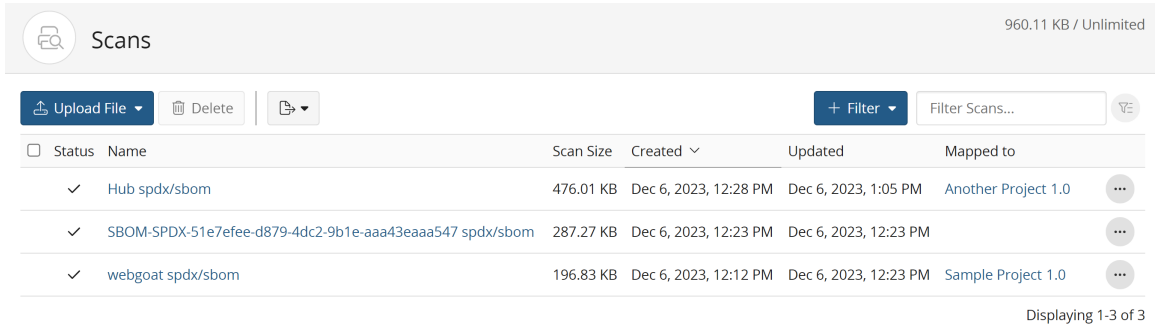
To map a Protex BOM to a project:

1. Log in to Black Duck.

2.  Click **Scans**.


The Scans page appears.

23. Integrating Protex with Black Duck • Mapping or unmapping a Protex BOM



Status	Name	Scan Size	Created	Updated	Mapped to	
✓	Hub spdx/sbom	476.01 KB	Dec 6, 2023, 12:28 PM	Dec 6, 2023, 1:05 PM	Another Project 1.0	⋮
✓	SBOM-SPDX-51e7efee-d879-4dc2-9b1e-aaa43eaaa547 spdx/sbom	287.27 KB	Dec 6, 2023, 12:23 PM	Dec 6, 2023, 12:23 PM		⋮
✓	webgoat spdx/sbom	196.83 KB	Dec 6, 2023, 12:12 PM	Dec 6, 2023, 12:23 PM	Sample Project 1.0	⋮

Displaying 1-3 of 3

3. If you did not use the Protex BOM tool to import the BOM, [use Black Duck's UI to import it](#).
4. Click  and select **Map to Project** in the row of the Protex BOM you want to map.
5. In the Map Scan dialog box, start typing the name of a project to progressively display matches.
6. Select the project version to which you want to map the Protex BOM.
7. Click **Save**.
Black Duck displays the name and version of the project to which you mapped the Protex BOM. Select the link to open the [BOM page](#).


To unmap a Protex:

You can remove the mapping of a Protex BOM.

1. Log in to Black Duck.

2. Click .

The Scans page appears.

3. Click  and select **Unmap from Project** in the row of the Protex BOM that you want to remove the mapping.
4. Click **Remove** to confirm.