OpenText™ Fortify Audit Workbench

User Guide

Version : 25.4.0

PDF Generated on : 05/11/2025

Table of Contents

1. User Guide

Please add the page content

1.1. Change log

The following table lists changes made to this document. Revisions to this document are published only if the changes made affect product functionality.

Software Release / Document Version	**Change**
25.4.0	Updated: Release date and version number
25.2.0	Updated: Release date and version number
24.4.0	Updated: • You can use the encoded authentication token when connecting to Application Security as the decoded token format is deprecated (see Logging in to Application Security)
24.2.0	Updated: • Added a timeout setting for downloading analysis results from Application Security (see Configuring a Connection to Application Security) • The unused metric executable lines of code is no longer displayed in the Project Summary view (see Project Summary View) • Added search modifier `engine priority` (see Search Modifiers) • Added New Issue by Category grouping attribute (see Grouping Issues)

1.2. Introduction

This section contains the following topics:

1.2.1. Product name changes

OpenText is in the process of changing the following product names:

Previous name	New name
Fortify Static Code Analyzer	OpenText™ Static Application Security Testing (OpenText SAST)
Fortify Software Security Center	OpenText™ Application Security
Fortify WebInspect	OpenText™ Dynamic Application Security Testing (OpenText DAST)
Fortify on Demand	OpenText™ Core Application Security
Debricked	OpenText™ Core Software Composition Analysis (OpenText Core SCA)
Fortify Applications and Tools	OpenText™ Application Security Tools

The product names have changed on product splash pages, mastheads, login pages, and other places where the product is identified. The name changes are intended to clarify product functionality and to better align the Fortify Software products with OpenText. In some cases, such as on the documentation title page, the old name might temporarily be included in parenthesis. You can expect to see more changes in future product releases.

1.2.2. About Fortify Audit Workbench

Fortify Audit Workbench complements OpenText™ Static Application Security Testing with a graphical user interface you can use to scan software projects and to organize, investigate, and prioritize the analysis results so that your team can fix security issues quickly and effectively.

From Fortify Audit Workbench, you can view and audit Fortify Project Results (FPR files) from OpenText™ Application Security and Secure Code Plugins. Fortify Audit Workbench issue templates help you sort the results of large scans in a way that works for your business and workflows.

Note

If your Fortify license restricts auditing, then you can scan your code, view audit projects (FPR files), and generate reports from Fortify Audit Workbench, but you cannot audit issues or make any changes to the audit project. You also cannot upload audit projects to Application Security. You can open and review collaborative audits in Application Security, but you cannot make any changes.

1.2.2.1. Audit projects and issue templates

After you initiate a source code scan from Fortify Audit Workbench, OpenText SAST scans and analyzes the code to produce comprehensive results (referred to as an audit project).

In Application Security, an application is a codebase that serves as a container for one or more application versions. A Application Security application version is an instance of the codebase that will eventually be deployed. An audit project is comparable to a Application Security application version in that it represents a snapshot of the codebase.

Issue templates determine how Fortify Audit Workbench (and Application Security) configures and prioritizes the vulnerabilities (issues) uncovered in source code. Fortify Audit Workbench comes with a single basic issue template, which you can use as is, or modify to suit your project needs. You can also import an issue template from Application Security, or create a new issue template from Fortify Audit Workbench.

1.2.2.2. Hybrid 2.0 technology

The Fortify Audit Workbench Hybrid 2.0 technology connects penetration test results directly to source code analysis results to reveal hidden vulnerability relationships and expose their root causes within the source code. This enables your security and development teams to accurately identify and prioritize vulnerabilities, and more productively investigate and remediate security issues in the source code.

1.2.3. Integration with OpenText SAST

You can analyze your code with OpenText SAST from Fortify Audit Workbench. You install OpenText SAST separately from the applications and tools. For instructions on installing OpenText SAST, see the *OpenText™ Static Application Security Testing User Guide*. Updating OpenText Application Security Content also requires a local installation of OpenText SAST.

The OpenText™ Application Security Tools installer (which includes Fortify Audit Workbench) can detect an existing OpenText SAST that is locally installed in the default location or in the same root folder where you installed OpenText™ Application Security Tools. If necessary, you are prompted when you first attempt to analyze your code to select the location of a locally installed OpenText SAST.

1.2.4. Integration with Application Security

Application Security provides a web portal that developers, managers, and security teams can use to share, collaborate, and track remediation of the potential vulnerabilities that OpenText SAST scans uncover. If you connect Fortify Audit Workbench to your Application Security server, you can upload and merge your scan and audit results and share them with your team. This enables you to monitor trends and indicators across multiple application versions.

Integration with Application Security enables you to:

- Upload audit projects (FPR files)
- Perform collaborative application audits
- Manage the security content, which consists of OpenText Secure Coding Rulepacks, custom Rulepacks, and external metadata applied during OpenText SAST scans
- Check for and install available upgrades of OpenText SAST and associated applications (including Fortify Audit Workbench)
- Download issue templates
- Upload new and modified issue templates

See Also

Working with Application Security

Configuring a Connection to Application Security

1.2.5. Related documents

This topic describes documents that provide information about OpenText Application Security Software products.

Note

Most guides are available in both PDF and HTML formats. Product help is available within the Fortify License and Infrastructure Manager (LIM) and the OpenText DAST products.

All products

The following documents provide general information for all products. Unless otherwise noted, these documents are available on the Product Documentation website for each product.

Document / file name	Description
About OpenText Application Security Software Documentation appsec-docs-n-*<version>*.pdf	This paper provides information about how to access OpenText Application Security Software product documentation. **Note** This document is included only with the product download.
OpenText™ Application Security Software System Requirements appsec-sr-*<version>*.pdf	This document provides the details about the environments and products supported for this version of OpenText Application Security Software.
What's New in OpenText Application Security Software <version> appsec-wn-*<version>*.pdf	This document describes the new features in OpenText Application Security Software products.

Document / file name	Description
OpenText Application Security Software Release Notes	This document provides an overview of the changes made to OpenText Application Security Software for this release and important information not included elsewhere in the product documentation.

OpenText ScanCentral DAST

The following document provides information about OpenText ScanCentral DAST. These documents are available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-ScanCentral-DAST.

Document / file name	Description
OpenText™ ScanCentral DAST Configuration and Usage Guide sc-dast-ugd-*<version>*.pdf	This document provides information about how to configure and use OpenText ScanCentral DAST to conduct dynamic scans of Web applications.
OpenText™ Fortify License and Infrastructure Manager Installation and Usage Guide lim-ugd-*<version>*.pdf	This document describes how to install, configure, and use the Fortify License and Infrastructure Manager (LIM), which is available for installation on a local Windows server and as a container image on the Docker platform.
OpenText™ Dynamic Application Security Testing and OAST on Docker User Guide dast-docker-ugd-*<version>*.pdf	This document describes how to download, configure, and use OpenText DAST and Fortify OAST that are available as container images on the Docker platform. The OpenText DAST image is intended to be used in automated processes as a headless sensor configured by way of the command line interface (CLI) or the application programming interface (API). It can also be run as an OpenText ScanCentral DAST sensor and used in conjunction with Application Security. Fortify OAST is an out-of-band application security testing (OAST) server that provides DNS, SMTP, and HTTP/HTTPS services for the detection of OAST vulnerabilities.

ScanCentral SAST

The following document provides information about ScanCentral SAST. This document is available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-software-security-center.

Document / file name	Description
OpenText™ ScanCentral SAST Installation, Configuration, and Usage Guide sc-sast-ugd-<*version*>.pdf	This document provides information about how to install, configure, and use ScanCentral SAST to streamline the static code analysis process. It is written for anyone who intends to install, configure, or use ScanCentral SAST to offload the resource-intensive translation and scanning phases of their OpenText SAST process.

Application Security

The following document provides information about OpenText Application Security (Software Security Center). This document is available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-software-security-center.

Document / file name	Description
OpenText™ Application Security User Guide ssc-ugd-<*version*>.pdf	This document provides Application Security users with detailed information about how to deploy and use Application Security. It provides all the information you need to deploy, configure, and use Application Security. It is intended for use by system and instance administrators, database administrators (DBAs), enterprise security leads, development team managers, and developers. Application Security provides security team leads with a high-level overview of the history and status of a project.

OpenText SAST

The following documents provide information about OpenText SAST (Fortify Static Code Analyzer). Unless otherwise noted, these documents are available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-static-code.

Document / file name	Description

Document / file name	Description
OpenText™ Static Application Security Testing User Guide sast-ugd-*<version>*.pdf	This document describes how to install and use OpenText SAST to scan code on many of the major programming platforms. It is intended for people responsible for security audits and secure coding.
OpenText™ Static Application Security Testing Custom Rules Guide sast-cr-ugd-*<version>*.zip	This document provides the information that you need to create custom rules for OpenText SAST. This guide includes examples that apply rule-writing concepts to real-world security issues. **Note** This document is included only with the product download.
OpenText™ Fortify License and Infrastructure Manager Installation and Usage Guide lim-ugd-*<version>*.pdf	This document describes how to install, configure, and use the Fortify License and Infrastructure Manager (LIM), which is available for installation on a local Windows server and as a container image on the Docker platform.

OpenText Application Security Tools

The following documents provide information about OpenText Application Security Tools. These documents are available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-static-code-analyzer-and-tools.

Document / file name	Description
OpenText™ Application Security Tools Guide sast-tgd-*<version>*.pdf	This document describes how to install application security tools. It provides an overview of the applications and command-line tools that enable you to scan your code with OpenText SAST, review analysis results, work with analysis results files, and more.
OpenText™ Fortify Audit Workbench User Guide awb-ugd-*<version>*.pdf	This document describes how to use Fortify Audit Workbench to scan software projects and audit analysis results. This guide also includes how to integrate with bug trackers, produce reports, and perform collaborative auditing.

OpenText™ Fortify Plugin for Eclipse User Guide ep-udg-*<version>*.pdf	This document provides information about how to install and use the Fortify Plugin for Eclipse to analyze and audit your code.
OpenText™ Fortify Analysis Plugin for IntelliJ IDEA and Android Studio User Guide iap-udg-*<version>*.pdf	This document describes how to install and use the Fortify Analysis Plugin for IntelliJ IDEA and Android Studio to analyze your code and optionally upload the results to Application Security.
OpenText™ Fortify Extension for Visual Studio User Guide vse-ugd-*<version>*.pdf	This document provides information about how to install and use the Fortify Extension for Visual Studio to analyze, audit, and remediate your code to resolve security-related issues in solutions and projects.

OpenText DAST

The following documents provide information about OpenText DAST (Fortify WebInspect). These documents are available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-webinspect.

Document / file name	**Description**
OpenText™ Dynamic Application Security Testing Installation Guide dast-igd-*<version>*.pdf	This document provides an overview of OpenText DAST and instructions for installing and activating the product license.

Document / file name	Description
OpenText™ Dynamic Application Security Testing User Guide dast-ugd-*<version>*.pdf	This document describes how to configure and use OpenText DAST to scan and analyze Web applications and Web services. **Note** This document is a PDF version of the OpenText DAST help. This PDF file is provided so you can easily print multiple topics from the help information or read the help in PDF format. Because this content was originally created to be viewed as help in a web browser, some topics may not be formatted properly. Additionally, some interactive topics and linked content may not be present in this PDF version.
OpenText™ Dynamic Application Security Testing and OAST on Docker User Guide dast-docker-ugd-*<version>*.pdf	This document describes how to download, configure, and use OpenText DAST and Fortify OAST that are available as container images on the Docker platform. The OpenText DAST image is intended to be used in automated processes as a headless sensor configured by way of the command line interface (CLI) or the application programming interface (API). It can also be run as an OpenText ScanCentral DAST sensor and used in conjunction with Application Security. Fortify OAST is an out-of-band application security testing (OAST) server that provides DNS, SMTP, and HTTP/HTTPS services for the detection of OAST vulnerabilities.
OpenText™ Fortify License and Infrastructure Manager Installation and Usage Guide lim-ugd-*<version>*.pdf	This document describes how to install, configure, and use the Fortify License and Infrastructure Manager (LIM), which is available for installation on a local Windows server and as a container image on the Docker platform.
OpenText™ Dynamic Application Security Testing Tools Guide dast-tgd-*<version>*.pdf	This document describes how to use the OpenText DAST diagnostic and penetration testing tools and configuration utilities packaged with OpenText DAST and Fortify WebInspect Enterprise.

Document / file name	Description
OpenText™ Dynamic Application Security Testing Agent Installation and Rulepack Guide dast-agent-igd-<*version*>.pdf	This document describes how to install the OpenText DAST Agent and describes the detection capabilities of the OpenText DAST Agent Rulepack Kit. OpenText DAST Agent Rulepack Kit runs atop the OpenText DAST Agent, allowing it to monitor your code for software security vulnerabilities as it runs. OpenText DAST Agent Rulepack Kit provides the runtime technology to help connect your dynamic results to your static ones.

Fortify WebInspect Enterprise

The following documents provide information about Fortify WebInspect Enterprise. These documents are available on the Product Documentation website at https://www.microfocus.com/documentation/fortify-webinspect-enterprise.

Document / file name	Description
OpenText™ Fortify WebInspect Enterprise Installation and Implementation Guide WIE_Install_<*version*>.pdf	This document provides an overview of Fortify WebInspect Enterprise and instructions for installing Fortify WebInspect Enterprise, integrating it with Application Security and OpenText DAST, and troubleshooting the installation. It also describes how to configure the components of the Fortify WebInspect Enterprise system, which include the Fortify WebInspect Enterprise application, database, sensors, and users.

Document / file name	Description
OpenText™ Fortify WebInspect Enterprise User Guide WIE_Guide_<*version*>.pdf	This document describes how to use Fortify WebInspect Enterprise to manage a distributed network of OpenText DAST sensors to scan and analyze Web applications and Web services. **Note** This document is a PDF version of the Fortify WebInspect Enterprise help. This PDF file is provided so you can easily print multiple topics from the help information or read the help in PDF format. Because this content was originally created to be viewed as help in a web browser, some topics may not be formatted properly. Additionally, some interactive topics and linked content may not be present in this PDF version.
OpenText™ Dynamic Application Security Testing Tools Guide dast-tgd-<*version*>.pdf	This document describes how to use the OpenText DAST diagnostic and penetration testing tools and configuration utilities packaged with OpenText DAST and Fortify WebInspect Enterprise.

1.3. Getting started

The following topics provide an overview of Fortify Audit Workbench, instructions on how to start the tool, and instructions on how to upgrade the Static Code Analyzer and Applications (OpenText SAST, Fortify Audit Workbench, and any plugins or extensions you have installed) as new versions of the products become available.

This section contains the following topics:

1.3.1. Installing Fortify Audit Workbench

You install Fortify Audit Workbench by selecting it as a component when you install OpenText Application Security Tools. For detailed installation instructions, see the *OpenText™ Application Security Tools Guide*.

Fortify Audit Workbench uses the license provided during the OpenText™ Application Security Tools installation.

1.3.2. About upgrades

You can check on the availability of new versions of OpenText SAST or OpenText™ Application Security Tools directly from Fortify Audit Workbench. If a version newer than the one you have installed is available from your Application Security server, you can download it and upgrade your instance.

You can also configure Fortify Audit Workbench to check for, download, and install new versions automatically at startup. Whether you upgrade OpenText SAST or OpenText™ Application Security Tools manually or automatically, Fortify Audit Workbench preserves your data.

To enable upgrades from Fortify Audit Workbench, a Application Security administrator must first set up the automatic upgrade capability on the Application Security server. For instructions on how to do this, see the *OpenText™ Application Security User Guide*.

This section contains the following topics:

1.3.2.1. Upgrading manually

You can check for newer versions of OpenText SAST or OpenText™ Application Security Tools manually, from either the Fortify Audit Workbench **Help** menu or the Options dialog box.

To check for, and (potentially) install, a newer version of OpenText SAST or OpenText™ Application Security Tools:

1. Select **Options > Options**.

   The Options dialog box opens to the **Server Configuration** settings.

2. Under **Audit Workbench Upgrade Configuration**, do the following:

   1. In the **Server URL** box, type the web address for the `installers` folder on your Application Security server (for example, `https://my.domain.com:8080/ssc/update-site/installers`).

      Note

      If your Application Security administrator set up the automatic upgrade capability using an XML file other than `update.xml`, then you must include the XML file in the **Server URL** box (for example, `https://my.domain.com:8080/ssc/update-site/installers/<update_config_file>.xml`).

   2. Click **Check Now**.

      Note

      You can also select **Help > Check for Upgrades** after you set up a Application Security installer web address described in the previous step.

The Fortify Audit Workbench polls the upgrade server for information about the OpenText SAST or OpenText™ Application Security Tools versions available for the platform on which it is running. If a newer version is available, Fortify Audit Workbench prompts you to indicate whether you want to proceed to download and install it.

Important

If you have the OpenText™ Fortify Plugin for Eclipse installed, after you upgrade OpenText Application Security Tools from Fortify Audit Workbench, you must uninstall, and then reinstall the Fortify Plugin for Eclipse.

1.3.2.2. Configuring automatic upgrades

To configure upgrade checks at Fortify Audit Workbench startup:

1. From Fortify Audit Workbench, select **Options > Options**.

   The Options dialog box opens to the **Server Configuration** settings.

2. Under **Audit Workbench Upgrade Configuration**, do the following:

   1. In the **Server URL** box, type the web address for the `installers` folder on your Application Security server (for example, `https://my.domain.com:8080/ssc/update-site/installers`).

      Note

      If your Application Security administrator set up the automatic upgrade capability using an XML file other than `update.xml`, then you must include the XML file in the **Server URL** box (for example, `https://my.domain.com:8080/ssc/update-site/installers/<update_config_file>.xml`).

   2. Select the **Check for upgrades at startup** check box.

3. Click **OK**.

Each time you start Fortify Audit Workbench, it checks the server to determine if a newer OpenText SAST or OpenText™ Application Security Tools version is available and then, if a newer version is available, downloads and installs it.

Important

If you have a Fortify Plugin for Eclipse installed, after you upgrade your OpenText™ Application Security Tools from Fortify Audit Workbench, you must uninstall, and then reinstall the Fortify Plugin for Eclipse.

1.3.3. Sample projects

The OpenText™ Application Security Tools installation includes a sample Fortify Project Results (FPR) file in `<tools_install_dir>/Samples/fprs`.

OpenText also provides several code samples in a separate download in the `Fortify_SCA_Samples_<version>.zip` archive. You can use these sample projects when learning to use OpenText SAST and Fortify Audit Workbench. The ZIP contains two directories: `basic` and `advanced`. Each code sample includes a `README.txt` file that provides instructions on how to scan the code in OpenText SAST and view the output in Fortify Audit Workbench.

The `basic` directory includes an assortment of simple language-specific code samples. The `advanced` directory contains more advanced samples.

1.3.4. Renewing expired licenses

The license for Fortify Audit Workbench expires annually. For information about how to obtain a Fortify license file, see the *OpenText™ Application Security Software System Requirements* document.

To update an expired license:

1. Put the updated Fortify license file in the `<tools_install_dir>` folder.
2. Start Fortify Audit Workbench and verify that it opens successfully.

1.3.5. About starting Fortify Audit Workbench

You can start Fortify Audit Workbench from the start menu on a Windows system. You can start it from the command line on any supported operating system.

This section contains the following topics:

1.3.5.1. Starting Fortify Audit Workbench on Windows systems

To start Fortify Audit Workbench on a Windows system, do one of the following:

- Select **Start > All Programs > Fortify Applications and Tools *<version>* > Audit Workbench**.
- Start Fortify Audit Workbench from the command line:
    1. Open a Command window.
    2. At the prompt, type `auditworkbench`.

1.3.5.2. Starting Fortify Audit Workbench on non-Windows systems

To start Fortify Audit Workbench on a Linux system:

1. Open a command prompt window, and then navigate to the `<tools_install_dir>/bin` directory.
2. At the prompt, type `auditworkbench`.

To start Fortify Audit Workbench on macOS:

- In the `<tools_install_dir>` directory, click `AuditWorkbench.app`.

1.3.6. Changing the appearance

Fortify Audit Workbench comes with a dark or light (default) theme.

To change the appearance:

1. Select **Options > Appearance** and select a theme.

   Note

   To reset the appearance to the default theme, select **Reset Interface**.

2. Restart Fortify Audit Workbench when prompted.

1.3.7. Working with Application Security

You need to configure a connection to Application Security to accomplish any of the following tasks:

- Upload your scan results to Application Security
- Audit applications collaboratively using Application Security
- Update your Fortify Software Security Content from Application Security

This section contains the following topics:

1.3.7.1. Configuring a connection to Application Security

To configure a connection to Application Security, you need the following:

- The web address for your Application Security and if necessary, the proxy server and port number for the connection
- If you connect to Application Security using X.509 SSO, download and deploy the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files to the Java JRE for Eclipse.
- If your Application Security server uses an SSL connection from an internal certificate authority or a self-signed certificate, you must import a self- or locally-signed certificate into the OpenText Application Security Tools keystore. The keystore is in the `<tools_install_dir>/jre/lib/security/cacerts` file.

To configure a connection to Application Security:

1. Select **Options > Options**.
2. In the left pane, select **Server Configuration**.
3. Under **Software Security Center Configuration**, specify the **Server URL**for your Application Security server.
4. If required, specify the proxy server, port number, and optionally credentials for proxy authentication.
5. To change the length of time provided to download analysis results from Application Security, type the timeout value in milliseconds in the **Download Timeout** box.

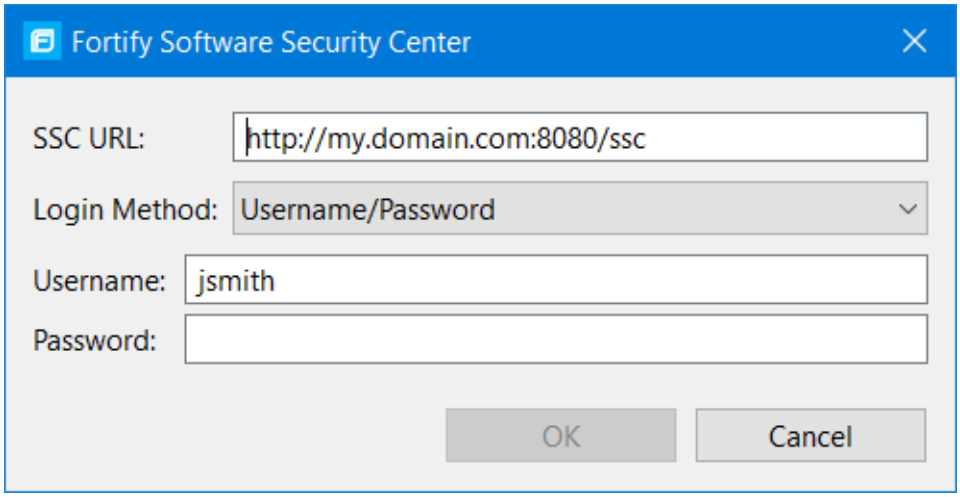
   Setting a value of zero is equivalent to no timeout for the download of analysis results.
6. Click **OK**.

1.3.7.2. Logging in to Application Security

The first time you perform an operation that requires a connection to Application Security such as uploading analysis results or performing a collaborative audit, you are prompted to log in.

To log in to Application Security:

1. If you have not configured a connection to Application Security, in the **SSC URL** box, type the server web address.
2. From the **Login Method** list, select the login method set up for you in Application Security.



3. Depending on the selected login method, do one of the following:

Login method	Procedure
Username/Password	Type your Application Security user name and password.
Authentication Token	In the **Token** box, specify the encoded value of a Application Security authentication token of type ToolsConnectToken. **Note** For instructions about how to create an authentication token, see the *OpenText™ Application Security User Guide*.
X.509 SSO	1. Click **Browse** to the right of **Certificate**. 2. In the Browser for Certificate dialog box, locate the p12 package with the certificate, and then click **Open**. 3. Type the password if required.

4. Click **OK** to connect to Application Security.

1.3.8. Application Security Content

OpenText SAST uses a knowledge base of rules to enforce secure coding standards applicable to the codebase for static analysis. Fortify software security content consists of OpenText Secure Coding Rulepacks and external metadata:

- OpenText Secure Coding Rulepacks describe general secure coding idioms for popular languages and public APIs
- External metadata provides mappings from the Fortify vulnerability categories to alternative categories (such as CWE, OWASP Top 10, and PCI)

OpenText provides the ability to write custom rules that add to the functionality of OpenText SAST and the OpenText Secure Coding Rulepacks. For example, you might need to enforce proprietary security guidelines or analyze a project that uses third-party libraries or other pre-compiled binaries that are not already covered by the OpenText Secure Coding Rulepacks. You can also customize the external metadata to map Fortify issues to different taxonomies, such as internal application security standards or additional compliance obligations. For instructions on how to create your own custom rules or custom external metadata, see the *OpenText™ Static Application Security Testing Custom Rules Guide*.

If you are using collaborative auditing with Application Security, make sure that any custom rules or external metadata changes are also made in Application Security.

Typically, you obtain the current OpenText Application Security Content when you install OpenText SAST.

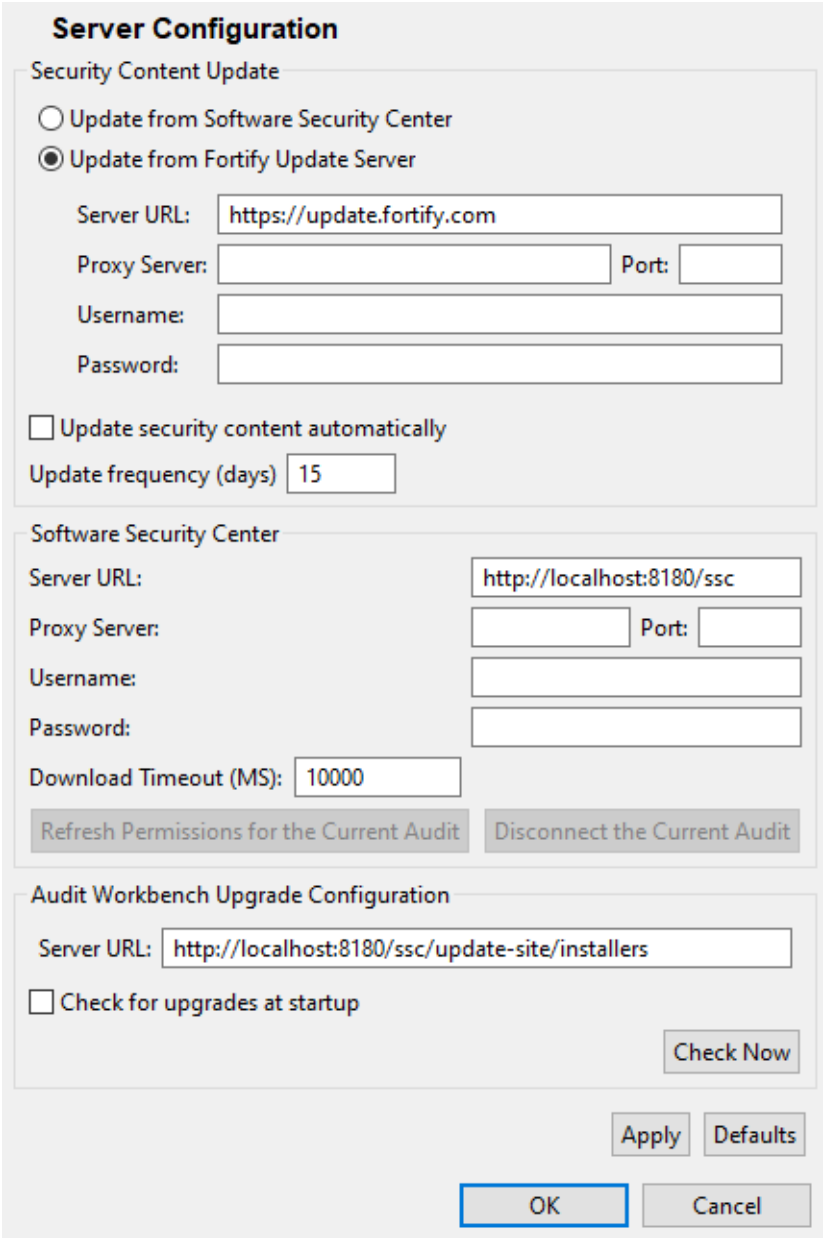
1.3.8.1. Configuring security content updates

You can configure the server from which to update security content and whether to have the security content updated from a server automatically.

To update security content from your local system (if you do not have an internet connection or a Application Security server), see Updating Security Content.

To configure the server from where you will obtain security content:

1. Select **Options > Options**.
2. In the left pane, select **Server Configuration**.



3. To update security content from your Application Security server:

1. Under **Security Content Update**, select **Update from Software Security Center**.
2. Under **Software Security Center**, specify the Application Security server web address and if required, the proxy server, port number, and credentials for proxy authentication.

Note

When you specify proxy information, exclude the protocol from the proxy server (for example, some.secureproxy.com). You must specify a proxy port number.

4. To specify an update server from which to update security content, under **Security Content Update**, do the following:
   1. In the **Server URL**box, type the web address for the update server.
   2. If required, specify the proxy server, port number, and credentials for proxy authentication.

Note

When you specify proxy information, exclude the protocol from the proxy server (for example, some.secureproxy.com). You must specify a proxy port number.

5. To update security content from a server automatically and with a specific frequency:
   1. Select the **Update security content automatically** check box.
   2. In the **Update frequency (days)** box, specify how often to update the security content.
6. Click **OK**.

See Also

Updating Security Content

Importing Custom Security Content

1.3.8.2. Updating security content

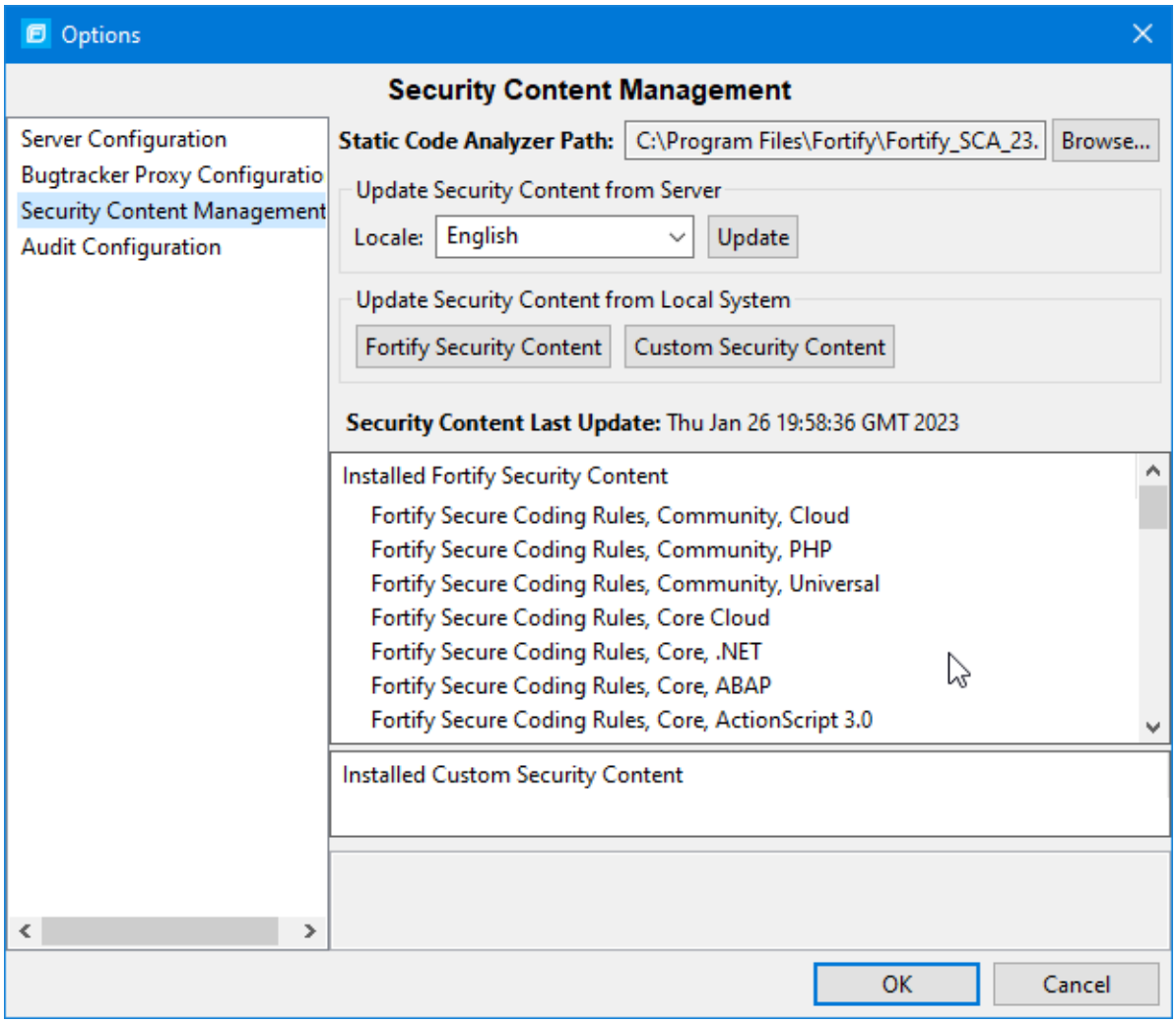
To optimize Fortify Audit Workbench functionality to scan with OpenText SAST, you must have up-to-date security content. You can update Fortify security content from a configured server or from your local system.

Important

To update security content, you must have OpenText SAST locally installed.

To update security content:

1. Select **Options > Options**.
2. In the left pane, select **Security Content Management**.



Note

Scroll to the bottom of the **Installed Fortify Security Content** list to see the external mappings.

Any custom rules and custom external mappings appear in the **Installed Custom Security Content** list.

3. You must provide the location of a locally installed OpenText SAST. If the **Static Code Analyzer Path** shows **<Unavailable>**, do the following:
    1. Click **Browse** to the right of **Static Code Analyzer Path**.
    2. Go to the OpenText SAST installation directory and select the executable file.

        On Windows, the file name is `sourceanalyzer.exe`. On non-Windows systems, the file name is `sourceanalyzer`.
    3. Click **OK**.
4. To update Fortify security content from a server, do the following:
    1. (Optional) From the **Locale** list, select a language.

        OpenText provides security content in English, Simplified Chinese, Traditional Chinese, Japanese, Korean, Spanish, or Brazilian Portuguese. Issue descriptions and recommendations are available in the selected language and the Fortify categories are in English.
    2. Click **Update**.
5. To update Fortify security content from your local system, under **Update Security Content from Local System**, do the following:
    1. Click **Fortify Security Content**.
    2. Navigate to a Fortify security content ZIP file, and then click **Open**.

All existing security content is replaced with the selected Fortify security content. Any existing custom security content is unchanged.

See Also

Importing Custom Security Content

Configuring Security Content Updates

1.3.8.3. Importing Custom Security Content

You can import custom security content to use in your scans.

Note

To import custom external metadata, you must place your external metadata file in the `<sca_install_dir>/Core/config/CustomExternalMetadata` directory.

To import custom rules, do the following:

1. Select **Options > Options**.
2. In the left pane, select **Security Content Management**.
3. Under **Update Security Content from Local System**, click **Custom Security Content**.
4. Select the custom rules files to import (*.xml and *.bin), and then click **Open**.

1.4. Scanning source code

You can scan your source code and view the analysis results in the Fortify Audit Workbench auditing interface.

This section contains the following topics:

1.4.1. Scanning Java projects

The Audit Guide Wizard combines the translation and analysis phases of the scanning process into a single step. Use this wizard to scan small Java projects that have source code in a single directory.

To scan a new Java project:

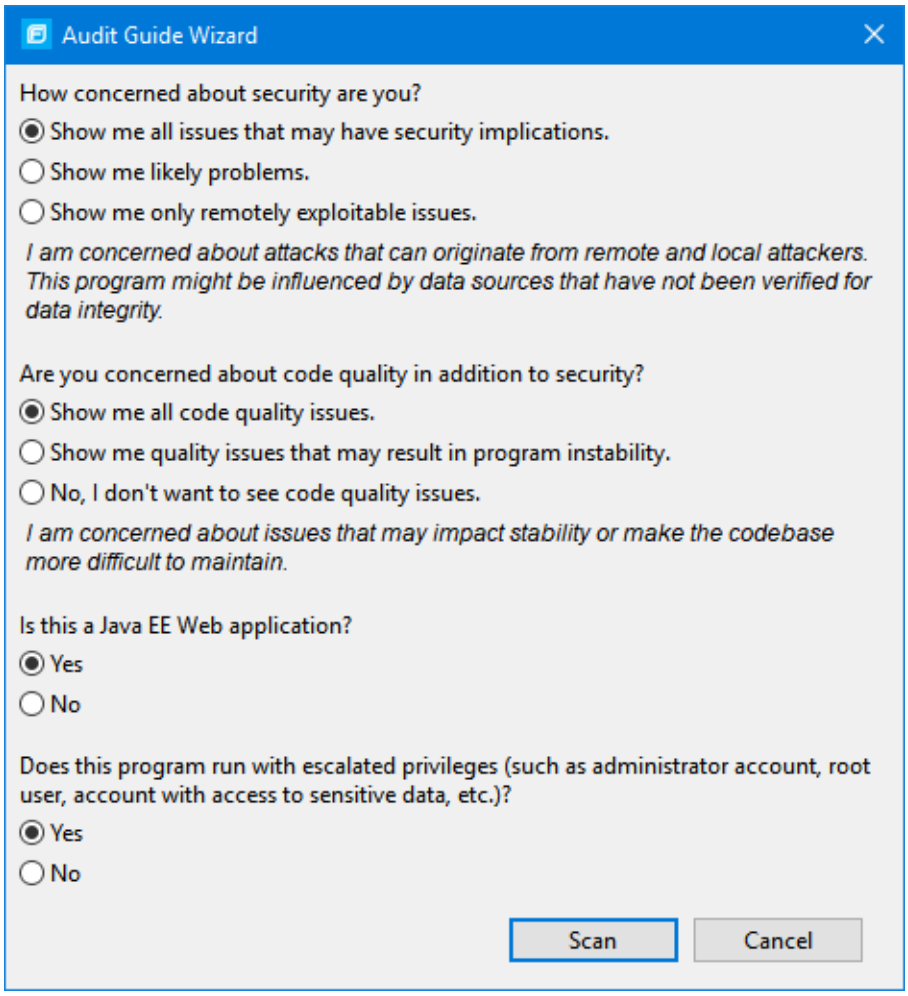
1. Start Fortify Audit Workbench.
2. Under **Start New Project**, click **Scan Java Project**.
3. Select the folder that contains all the source code you want to analyze, and then click **Select Folder**.

   Note

   OpenText SAST sets the build ID to the folder name.

4. Select the Java version used for your project, and then click **OK**.

   The Audit Guide Wizard opens.



5. Select the settings for the types of issues you want to display in the results, and then

click **Scan**.

OpenText SAST analyzes the source code. If OpenText SAST encounters any problems as it scans the source code, Fortify Audit Workbench displays a warning.

6. If a warning is displayed, click **OK**.
7. After the scan is complete, Fortify Audit Workbench displays the analysis results.

   Fortify Audit Workbench stores the analysis results (FPR file) in the following directory:

   - Windows: `C:\Users\<username>\AppData\Local\Fortify\AWB-<version>\<build_ID>`
   - Non-Windows: `<userhome>/.fortify/AWB-<version>/<build_ID>`

Note

OpenText SAST scans started from Fortify Audit Workbench are invoked with the server Java Virtual Machine.

1.4.2. About quick scan mode

Quick scan mode provides a way to quickly scan your projects for critical- and high-priority issues. OpenText SAST performs the scan faster by reducing the depth of the analysis and applying the Quick View filter set. The quick scan settings are configurable. For more details about the configuration of quick scan mode, see the *OpenText™ Static Application Security Testing User Guide*.

Quick scans are a great way to get many applications through an assessment so that you can quickly find issues and begin remediation. The performance improvement you get depends on the complexity and size of the application. Although the scan is faster than a full scan, it does not provide as robust a result set. Other issues that a quick scan cannot detect might exist in your application. OpenText recommends that you run full scans whenever possible.

Note

By default, Fortify Software Security Center does not allow you to upload scans performed in quick scan mode. However, you can configure your Fortify Software Security Center application version so that uploaded audit projects scanned in quick scan mode are processed. For more information, see analysis results processing rules in the *OpenText™ Application Security User Guide*.

To perform a quick scan, follow the steps described in Scanning Large, Complex Projects and select the **Enable Quick Scan Mode** check box. Quick scan is also available when you scan Visual Studio solutions (see Scanning Solutions). Fortify Audit Workbench displays the scan results in its **Project Summary** view. You audit quick scan results just as you audit full scan results.

1.4.3. Scanning large and complex projects

Exceptionally large codebases might require some configuration to ensure a complete scan, including using OpenText SAST to scan the code in smaller sections. While Fortify Audit Workbench enables you to edit OpenText SAST command options, you can handle large, complex scans more successfully directly through the command console. In addition, if a system has memory constraints, OpenText SAST must compete with the Fortify Audit Workbench for resources, which might result in slow or failed scans.

Use the Advanced Static Analysis wizard for projects that have source code in multiple directories, special translation or build requirements, or that have files that you want to exclude from the project.

Note

Fortify Audit Workbench filters out unsupported files within the selected source code directories.

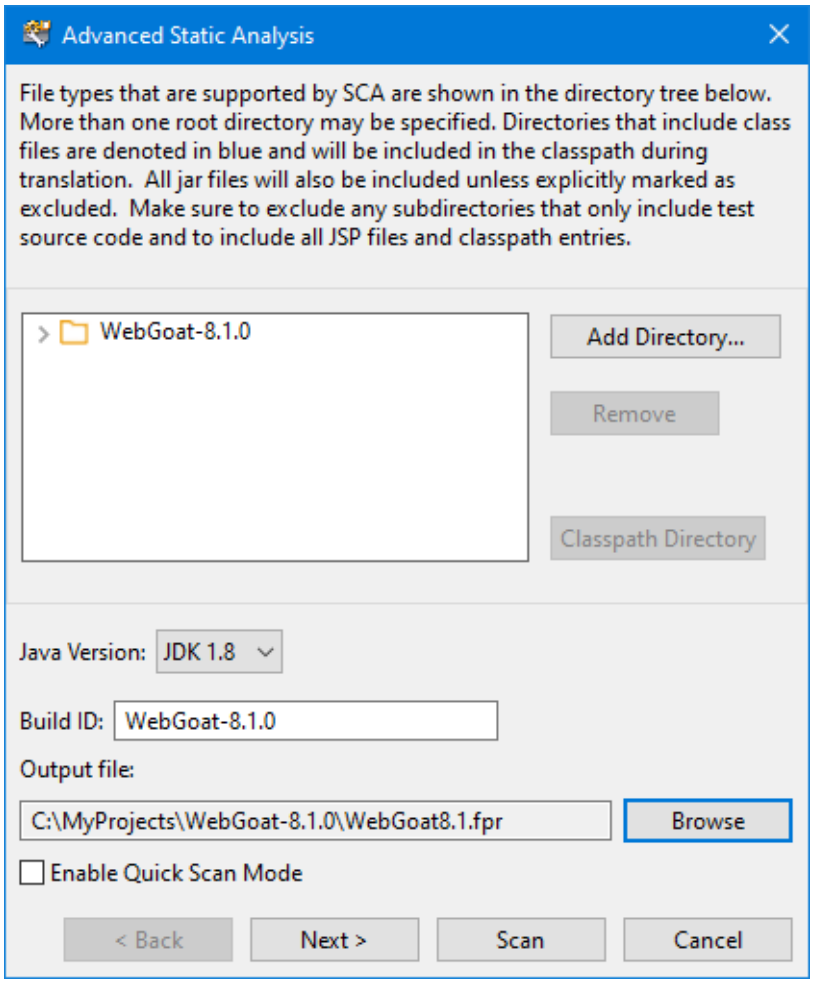
To scan a new project:

1. Start Fortify Audit Workbench.
2. Under **Start New Project**, click **Advanced Scan**.
3. Select the root directory of the project, and then click **Select Folder**.

   The Advanced Static Analysis wizard opens.

Note

The following image shows the wizard options when you select a Java project. The options are different for other programming languages.



The wizard automatically includes all supported files in the scan.

4. (Optional) To add files from another directory:
    1. Click **Add Directory**.
    2. Select the folder that contains the files you want to add to the scan, and then click **Select Folder**.

       The navigation pane displays the directory and Fortify Audit Workbench adds all supported files to the scan. (To remove the directory, right-click the folder, and then select **Remove Root**.)
5. (Optional) To exclude files or directories that contain, for example, test source code, right-click the file or directory, and then click **Exclude**.
6. For Java projects, set the following:
    1. Select the build directories and JAR files, and then click **Classpath Directory**.

       Note

       If you do not select the classpath directory, OpenText SAST uses the `CLASSPATH` environment variable value.

       The folder turns blue, and the files are added to the class path.

2. From the **Java Version** list, select the Java version of the project.

7. In the **Build ID** box, type a build ID.

   The root directory is the default build ID.

8. To specify a different output file path than the default, in the **Output file** box, type the path and file name for the FPR file that OpenText SAST will generate.
9. To perform a quick scan, select the **Enable Quick Scan Mode** check box.

   For information about quick scans, see Quick Scan Mode.

10. Click **Next**.

    The analysis process includes the following phases:

    - During the *clean* phase, OpenText SAST removes files from previous translation of the project.
    - During the *translation* phase, OpenText SAST translates source code identified in the previous page into an intermediate format that is associated with a build ID. The build ID is typically the project.
    - During the *scan* phase, OpenText SAST scans source files identified during the translation phase and generates analysis results, in the Fortify Project Results (FPR) format.

11. (Optional) To skip an analysis phase, clear the **Enable clean**, **Enable translation**, or **Enable scan** check box.

    For example, if the security content has changed but the project has not changed, you might want to skip both the clean and the translation phases so that OpenText SAST scans the project without translating it again.

12. Modify the command-line options for each OpenText SAST analysis phase to suit your requirements.
13. (Optional) To specify the amount of memory OpenText SAST used for analysis:

    1. Click **Configure Memory**.
    2. Adjust the slider to the amount of memory required.

       Note

       Fortify Audit Workbench displays the amount of memory you set for OpenText SAST followed by the amount of memory on your system.

    3. Click **OK.**

14. (Optional) To analyze the source code using an installed custom Rulepack, or to turn off a Rulepack, do the following:

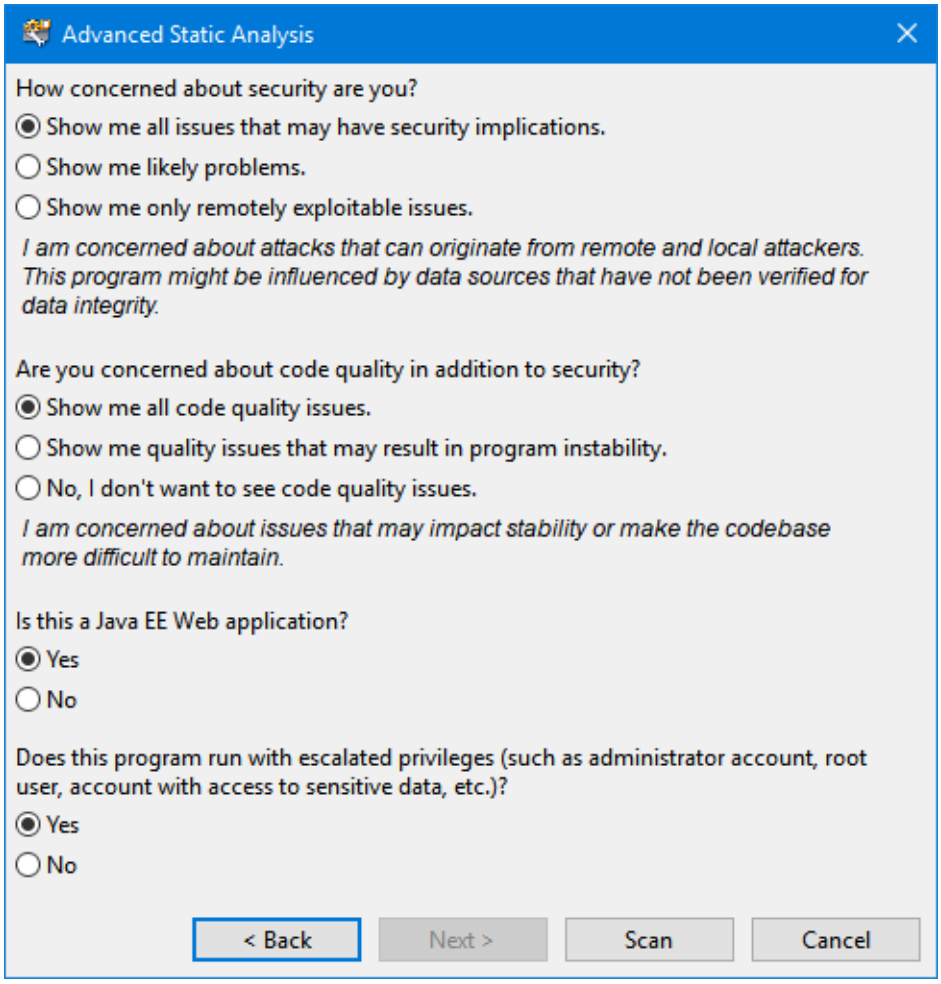
    The Additional Options dialog box opens.

1. Click **Configure Rulepacks**.
2. In the **Installed Fortify Security Content** list, clear the check boxes that correspond to any Rulepacks you want to make unavailable during the scan.

> **Note**
>
> For instructions on how to add custom security content, see Importing Custom Security Content.

3. Click **OK**.

15. From the Advanced Static Analysis wizard, click **Next**.



16. Select your scan settings, and then click **Scan**.

OpenText SAST starts the scan and displays progress information throughout the process. If OpenText SAST encounters any problems scanning the source code, it displays a warning.

After the scan is complete, Fortify Audit Workbench loads the audit project and displays the analysis results.

1.4.4. Scanning Visual Studio solutions

If you have Visual Studio and the Fortify Extension for Visual Studio installed on the same machine as Fortify Audit Workbench, you can analyze Visual Studio solutions and projects.

To scan a Visual Studio solution:

1. Start Fortify Audit Workbench.
2. Under **Start New Project**, click **Visual Studio Build Integration**.

   Note

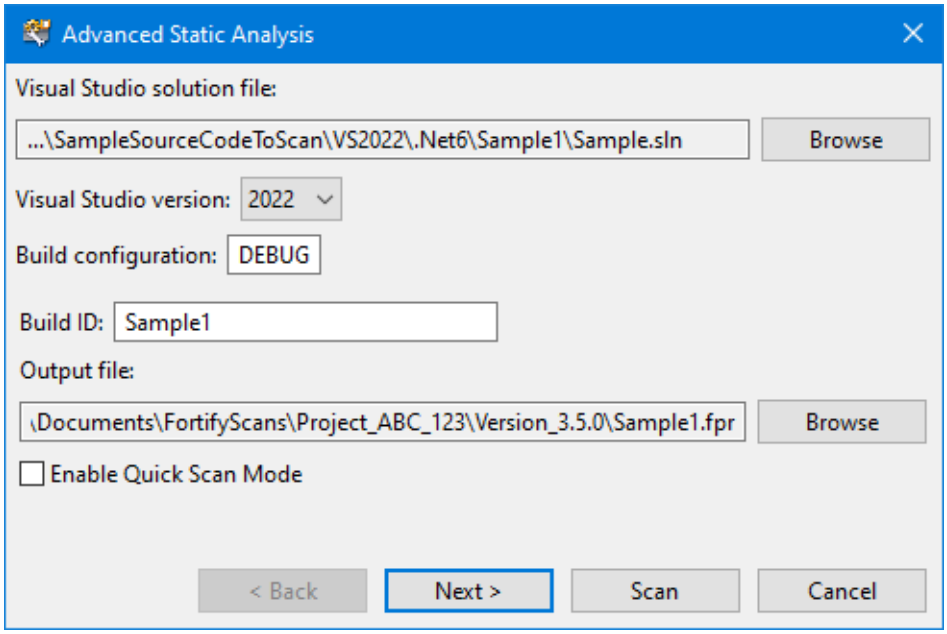
   The **Visual Studio Build Integration** command is only available if you have installed the Fortify Extension for Visual Studio with the OpenText™ Application Security Tools installation.

3. Select the folder that contains the solution you want to analyze, and then click **Select Folder**.

   Note

   OpenText SAST uses the selected folder name as the build ID.

   The Advanced Static Analysis wizard opens.



4. Configure the solution settings as follows:

   1. (Optional) Next to the **Visual Studio solution file** box, click **Browse**. Navigate to and select your Visual Studio solution file.
   2. From the **Visual Studio version** list, select the Visual Studio version used for the solution.

3. In the **Build configuration** box, leave the default value `DEBUG`.
4. (Optional) In the **Build ID** box, type a different build ID.
5. (Optional) To change the output location and file name, click **Browse** to the right of **Output file**.
6. To run the scan in quick scan mode, select the **Enable Quick Scan Mode** check box.
7. Click **Next**.

The Advanced Static Analysis wizard displays details about the OpenText SAST analysis phases for the scan.

- During the *clean* phase, OpenText SAST removes files from previous translation of the project.
- During the *translation* phase, OpenText SAST translates source code identified in the previous page into an intermediate format that is associated with a build ID. The build ID is typically the project.
- During the *scan* phase, OpenText SAST scans source files identified during the translation phase and generates analysis results, in the Fortify Project Results (FPR) format.

5. (Optional) To skip a scanning phase, clear the **Enable clean**, **Enable translation**, or **Enable scan** check box.

   For example, if the Rulepacks have changed but the project has not changed, you might want to skip both the clean and the translation phases so that OpenText SAST scans the project without retranslating the source code.

6. Modify the command-line options for each OpenText SAST phase, if necessary.
7. (Optional) To specify the amount of memory OpenText SAST uses for scanning:

   1. Click **Configure Memory**.
   2. Adjust the slider to the amount of memory required.

      Note

      Fortify Audit Workbench displays the amount of memory you set for OpenText SAST followed by the amount of memory on your system.

   3. Click **OK**.

8. (Optional) To analyze the source code using an installed custom Rulepack, or to turn off a Rulepack, do the following:

   1. Click **Configure Rulepacks**.
   2. In the **Installed Fortify Security Content** list, clear the check boxes that correspond to any Rulepacks you want to make unavailable during the scan.

Note

For instructions on how to add custom security content, see Importing Custom Security Content.

3. Click **OK**.

9. From the Advanced Static Analysis wizard, click **Next**.
10. Select your scan settings, and then click **Scan**.

OpenText SAST starts the scan and displays progress information throughout the process. If OpenText SAST encounters any problems scanning the source code, it displays a warning.

After the scan is completed, Fortify Audit Workbench loads the audit project and displays the analysis results.

1.4.5. Rescanning projects

This section describes how to rescan a project that was translated locally with new or updated rules. Fortify Audit Workbench automatically loads the FPR project settings such as the build ID and source code path and enables you to change the command-line scanning options.

After OpenText SAST completes the scan, Fortify Audit Workbench merges the analysis results with those from the previous scan to determine which issues are new, which have been removed, and which were uncovered in both scans.

To rescan a project:

1. Open an FPR file.
2. Select **Tools > Rescan Project**.

   Note

   You can only rescan a project on the same machine where the project was originally scanned.

   The Rescan Build ID dialog box opens.

3. If the source code has changed since the most recent scan, click **Update Project Translation** to re-translate the project.

   Note

   If the FPR file that you opened was generated by a OpenText SAST scan that was not initiated from Fortify Audit Workbench, then **Update Project Translation** is unavailable.

   Note

   If the source code has changed since the most recent scan, you must update the translation before you rescan the code. Otherwise, a new scan cannot uncover the issues in the updated source code.

4. (Optional) Modify the OpenText SAST scan phase command-line options, as necessary.
5. To perform a quick scan, select the **Enable Quick Scan Mode** check box.
6. (Optional) To change the Rulepacks used to analyze the project:
   1. Click **Configure Rulepacks**.
   2. Click to expand the **Installed Fortify Security Content**.
   3. To add and remove Rulepacks, select or clear the check boxes, as necessary.

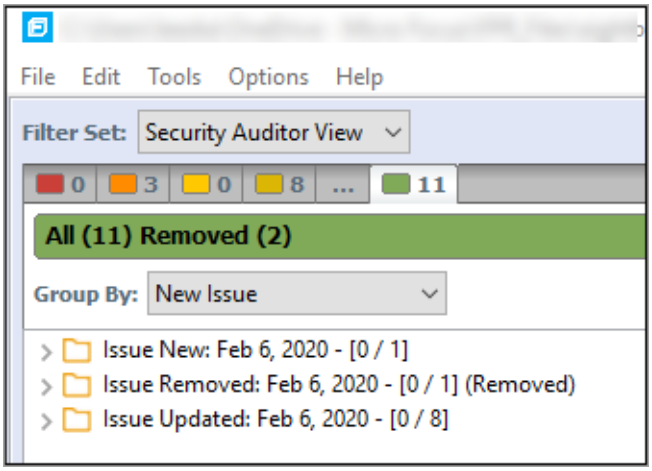
> **Note**
>
> For instructions on how to add custom security content, see Importing Custom Security Content.

4. Click **OK**.

7. Click **Scan**.

After the scan is complete, Fortify Audit Workbench displays the results. Compare the new results with the issues uncovered in the previous scan as follows:

- To display all new issues, select the **All** tab (green), and then, in the **Group By** list, select **New Issue**. Expand the **Issue New** group.
- To display removed issues, select the **All** tab, and then select **Options > Show Removed Issues**.
- To review issues found in both the previous scan and the new scan, select the **All** tab, and then in the **Group By** list, select **New Issue**. Expand the **Issue Updated** group.



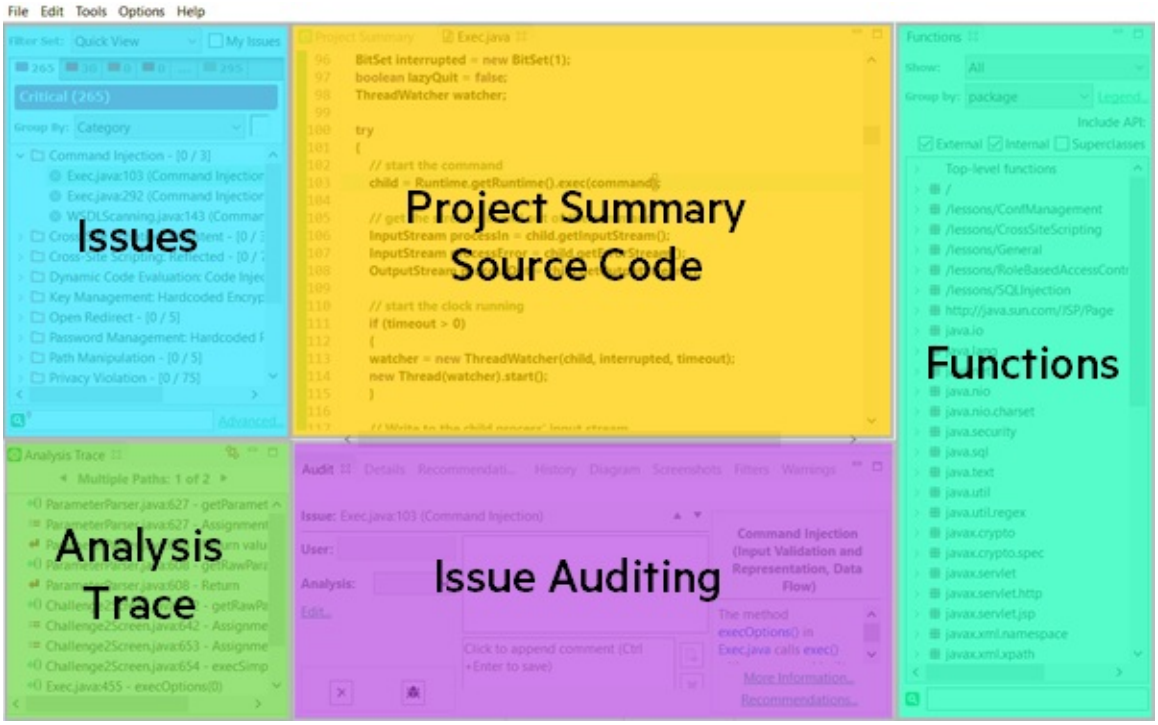
1.5. Viewing analysis results

After a scan is completed, Fortify Audit Workbench displays the analysis results in the Fortify Audit perspective.

This section contains the following topics:

1.5.1. About viewing analysis results

After the scan is complete (or, after you open an existing audit project), summary analysis results are displayed in the **Issues** view and in the **Project Summary** view of the Fortify Audit perspective. The **Analysis Trace** and **Issue Auditing** views are open, but do not contain any information until you select an issue from the **Issues** view.



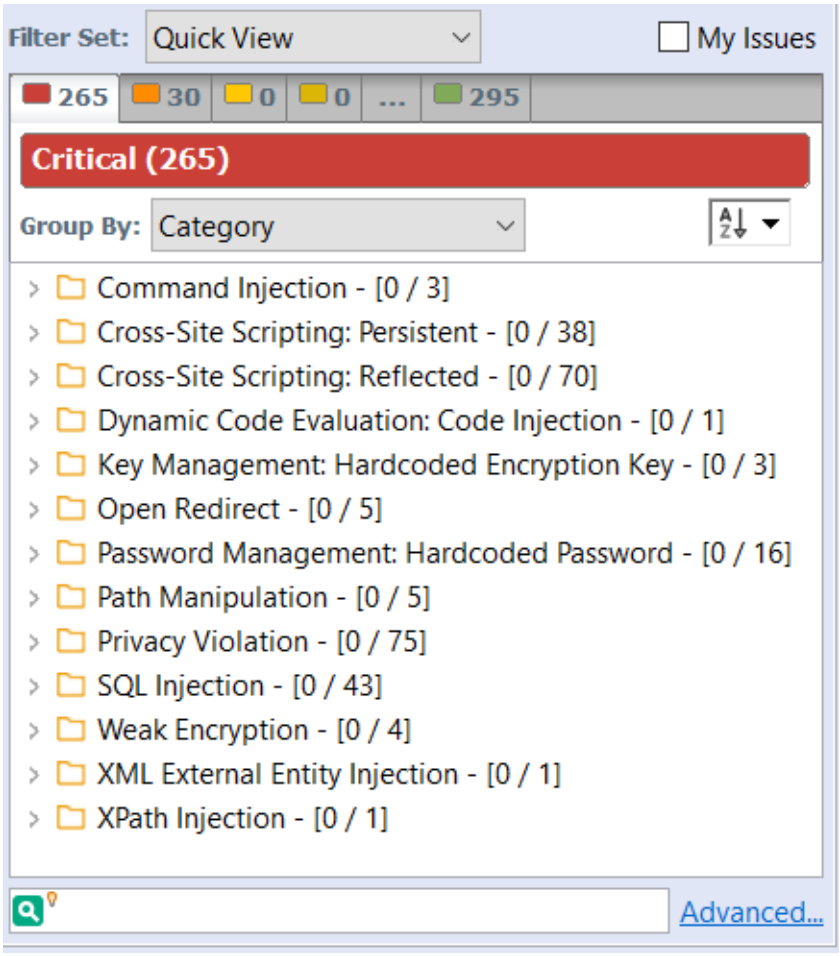
View / Tab	More Information
Issues (top left)	Issues view
Project Summary (top center)	Project Summary View
Source Code (top center)	Source Code Tab
Analysis Trace (bottom left)	Analysis Trace View
Issue Auditing (bottom center)	Issue Auditing View
Functions (right)	Functions View

1.5.1.1. Issues view

The **Issues** view lists the issues detected in the application and provides several ways to group them. The view contains the **Filter Set** list, folders (tabs), the **Group By** list, the **My Issues** check box, and a search box.

Note

In this view, you can right-click an issue and select **Issue Attributes** to see all the attributes associated with the issue such as Analysis tag, analyzer that detected the issue, severity, and more.



This section contains the following topics:

1.5.1.1.1. Filter sets

Fortify Audit Workbench applies filters to sort and display the issues that Static Code Analyzer uncovers. Fortify Audit Workbench organizes filters into distinct *filter sets*.

The selected filter set controls which issues are listed in the **Issues** view. The filter set determines the number and types of containers (folders) that are shown and how and where to display issues. The default filter sets sort the issues by severity into the **Critical**, **High**, **Medium**, **Low**, and **All** folders.

Because filter sets are saved to audit project files, each audit project can have unique filter sets.

Fortify Audit Workbench provides the following filter sets for new projects:

- **Quick View**: This is the default initial filter set for new projects. The Quick View filter set provides a view only of issues in the **Critical** folder (these have a potentially high impact and a high likelihood of occurring) and the **High** folder (these have a potentially high impact and a low likelihood of occurring). The Quick View filter set provides a useful first look at results that enables you to quickly address the most pressing issues.
- **Security Auditor View**: This is the default filter set for projects scanned in earlier product versions. This view shows all security issues detected. The Security Auditor View filter contains no visibility filters, so all issues are shown.

For instructions on how to create custom filter sets, see Configuring Custom Filter Sets and Filters.

If you open an FPR file that contains no custom `filtertemplate.xml` file or if you open an FVDL file or a `webinspect.xml` file, the audit project opens with the Quick View filter set selected.

1.5.1.1.2. Specifying the default filter set

You can change the initial filter set to use for new or opened projects. You can also turn off the default filter set so that Fortify Audit Workbench uses the filter set last enabled in the issue template to display analysis results for new projects.

To select the filter set for new or opened projects:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab on the right.
3. Under **Audit Project Load Mode**, leave the **Default Filter Set** check box selected.

   If you clear the check box, the default filter is loaded. For newly-opened projects, the default filter for FPRs that have no embedded template or the default filter from the embedded template is the Security Auditor View filter set.

4. From the list to the right of the **Default Filter Set** check box, select the filter set to use to display analysis results for new projects.
5. Click **OK**.

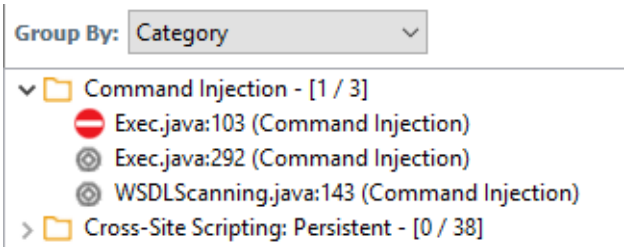
1.5.1.1.3. Folders (tabs)

The color-coded **Critical**, **High**, **Medium**, **Low**, and **All** tabs on the **Issues** view are called folders. You can customize the folders and their settings. The number of folders, names, colors, and the issue list can vary between filter sets and projects.

Note

In Fortify Audit Workbench the term folder *does not* refer to the folder in the issues list.

Within each color-coded folder, issues are grouped into subfolders. At the end of each folder name, enclosed in brackets, is the number of audited issues and the total number of issues in the folder. For example, **Command Injection - [1 / 3]** indicates that one out of three issues categorized as Command Injection has been audited.

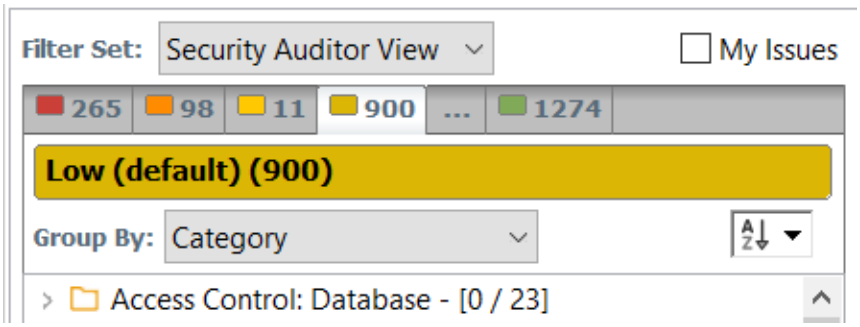


The filter set you select from the **Filter Set** list determines which folders are visible in the **Issues** view. The following table describes the folders that are visible when the **Security Auditor View** filter set is selected.

Folder	Description
Critical	This folder contains issues that have a high impact and a high likelihood of occurring. Issues at this risk level are easy to discover and to exploit and represent the highest security risk to a program. Remediate critical issues immediately.
High	This folder contains issues that have a high impact and a low likelihood of occurring. High-priority issues are often difficult to discover and exploit, but can result in much asset damage. They represent a significant security risk to a program. Remediate these issues with the next patch release.
Medium	This folder contains issues that have a low impact and a high likelihood of exploitation. Medium-priority issues are easy to discover and exploit but often result in little asset damage. These issues represent a moderate security risk to a program. Remediate these issues as time permits.

Low	This folder contains issues that have a low impact and a low likelihood of exploitation. Low-priority issues are potentially difficult to discover and to exploit and typically result in little asset damage. These issues represent a minor security risk to the program. Remediate these issues as time permits.
All	This folder contains all the issues.

An issue is listed in a folder if the folder filter conditions match the issue attributes. Each filter set has a default folder, indicated by **(default)** next to the folder name. If an issue does not match any of the folder filters, the issue is listed in the default folder.



You can create your own folders as you need them. For example, you might group all hot issues for a project into a **Hot** folder and group all warning issues for the same project into a **Warning** folder. For instructions on how to create your own folders, see Creating Folders.

Each folder contains a list of all the issues with attributes that match the folder filter conditions. One folder in each filter set is the default folder, indicated by **(default)** in the folder name.

Note

To show or hide suppressed, hidden, and removed issues, set the user interface preferences from the Options dialog box (see Customizing the View).

1.5.1.1.4. Group By list

You can use the **Group By** list of grouping attributes to sort the issues into subfolders. The grouping attribute you select is applied to all visible folders. To list all issues in the folder without any grouping, select **<none>**.

To customize the existing groups, you can specify which attributes to sort by, add or remove the attributes to create sub-groupings, and add your own grouping options.

The grouping attributes apply to the application instance. You can apply the grouping attributes to any project opened with that instance of the application.

See Also

Grouping Issues

Creating a Custom Grouping Option

1.5.1.1.5. Specifying the default issue grouping

You can change the initial Group By setting to use for new or opened projects.

To select the default Group By setting:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab on the right.
3. Under **Audit Project Load Mode**, select the **Default Issue Grouping** check box.

   If you clear the check box, the default Group By setting is set to Category.

4. From the list to the right of the **Default Issue Grouping** check box, select the grouping you want to use to sort issues.
5. Click **OK**.

1.5.1.1.6. Sorting issues

There are several different ways to sort the issues in the **Issues** View. Select a sort option from the **Sort** list. The following table describes the sort options.

Sort Method	Button	Description
Alphabetical		Sorts the groups and the issues within the groups in alphabetical order
		Sorts the groups and the issues within the groups in reverse-alphabetical order
Group size		Sorts the groups by the number of contained issues from largest to smallest
		Sorts the groups by the number of contained issues from smallest to largest
Last modified date		Sorts the groups and issues in groups by the date last modified by OpenText SAST or the audit/comment date from newest to oldest
		Sorts the groups and issues in groups by the date last modified by OpenText SAST or the audit/comment date from oldest to newest

1.5.1.1.7. Search box

Use the search box to limit the issues displayed in the folder and to search for specific issues. For detailed information about how to use the search box, see Searching for Issues.

1.5.1.2. Project Summary view

The **Project Summary** view provides detailed information about the scan.

To open this view, select **Tools > Project Summary**.

Summary tab

The **Summary** tab shows high-level information about the project. For more information, see Viewing Summary Graph Information.

Note

If the **Summary** tab header indicates that there are warnings in your scan, you can review them in more detail in the Issue Auditing view. For more information, see Warnings Tab.

Certification tab

The **Certification** tab displays the certification status for the analysis results. Results certification is a check to ensure that the analysis results were not altered after OpenText SAST produced them

Build Information tab

The **Build Information** tab displays the following information:

- Build details including the build ID, build label, number of files scanned, source last-modified date, and the date of the scan, which might be different than the date the files were translated
- Total lines of code (Total LOC) scanned

  The total number of lines of code, including blank lines and comments

- List of files scanned with file sizes and timestamps
- Libraries referenced in the scan
- Java class path used in the translation

Analysis Information tab

The **Analysis Information** tab shows the version of OpenText SAST that performed the scan,

details about the computer on which the scan was run, the user who started the scan, scan date, and the time required to scan the code.

The **Analysis Information** tab includes the following subtabs:

- **Security Content**—Lists information about the Rulepacks used to scan the source code
- **Properties**—Displays the OpenText SAST configuration properties used in the scan
- **Commandline Arguments**—Displays the command-line options used to scan the project

1.5.1.2.1. Viewing summary graph information

The summary graph displayed in the **Project Summary** view provides multiple perspectives on the sets of issues, grouped by priority (Critical, High, Medium, and Low) uncovered in a scan. You can drill down in the graph to see detailed information about each issue set, and create various bar charts for issues based on a selected issue attribute.

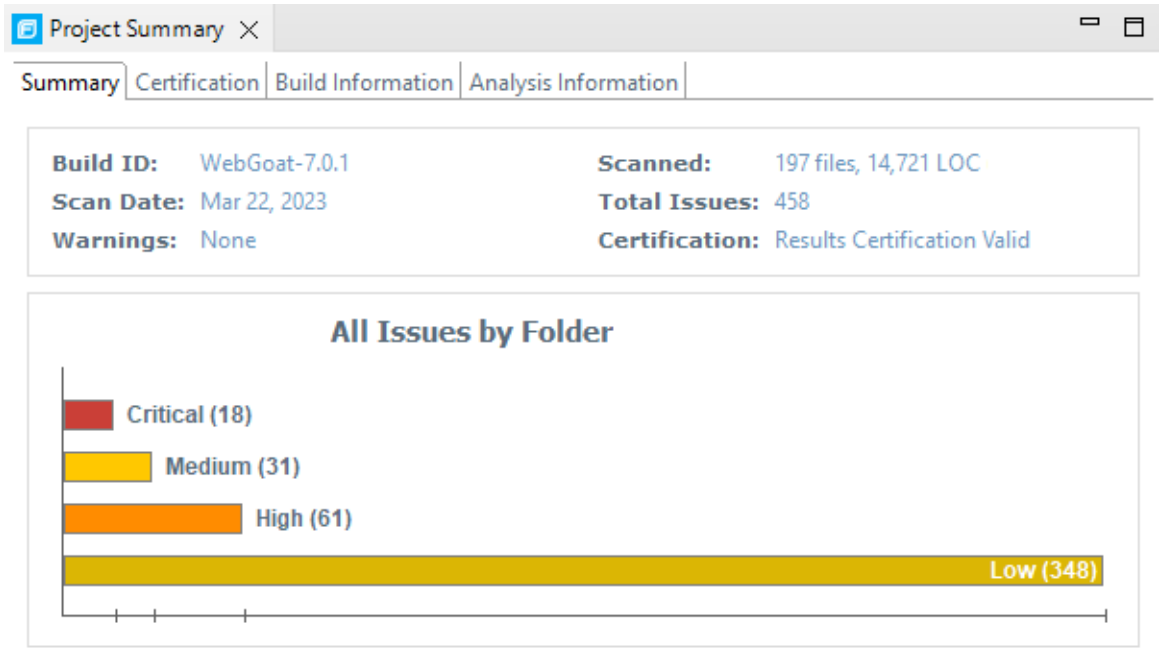
To access details about issue sets in an audit project:

1. Scan your project source code or open an existing audit project.

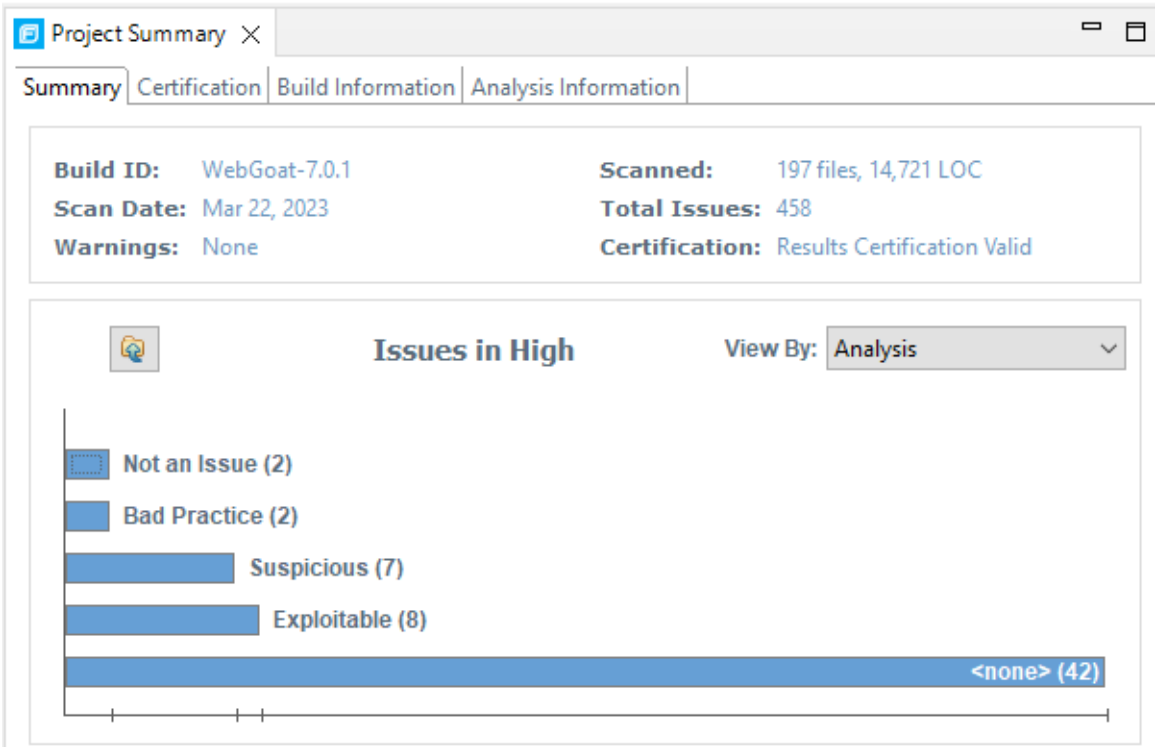
   After the results are loaded, the **Project Summary** view displays the **Summary** tab, which includes the summary graph. The summary graph initially displays issues sorted into the **Critical**, **High**, **Medium**, and **Low** folders.

   Note

   If you change the selection in the **Filter Set** list (**Issues**), the summary graph changes accordingly.



2. To see a different view of the high priority issues, click the **High** bar.

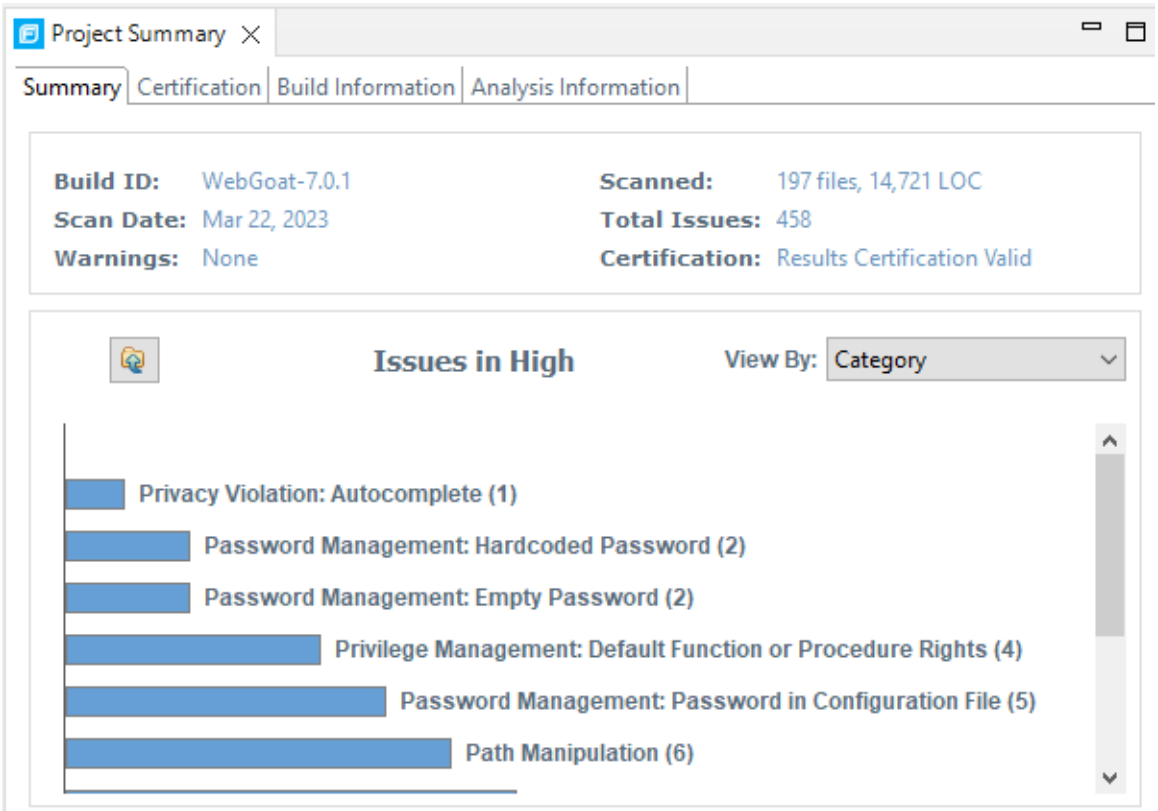


By default, the graph displays high priority issues based on the analysis attribute (assigned analysis values).

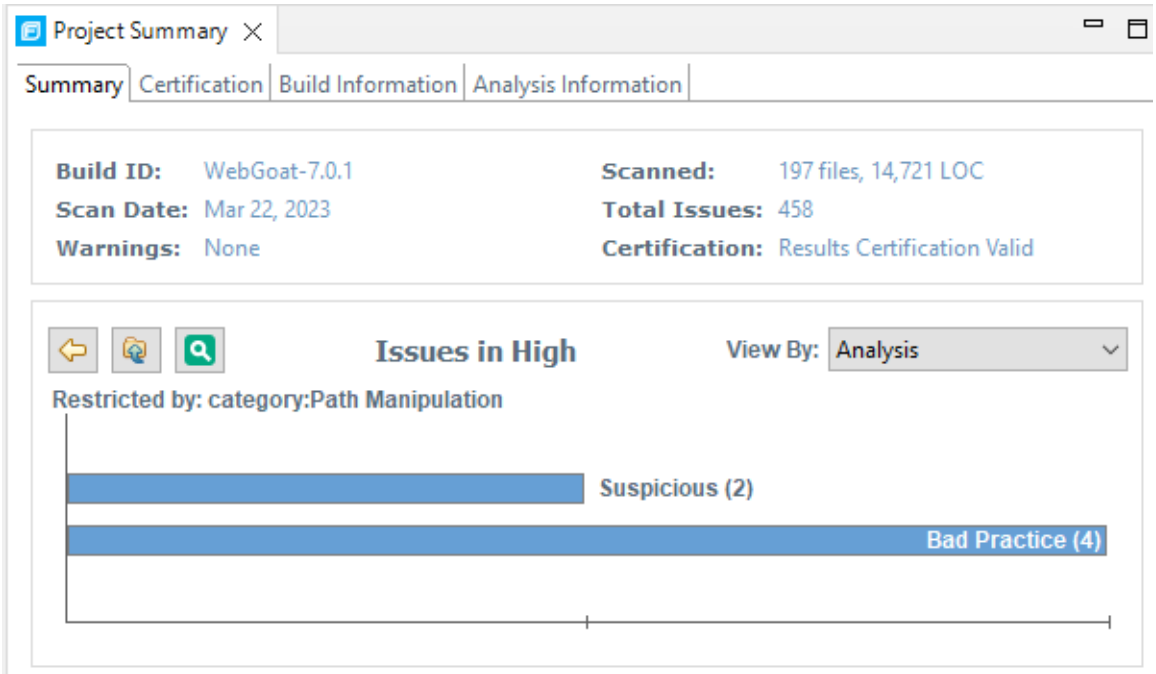
Note

The example here shows information for analysis results that have been partially audited. If these results were from a fresh, unaudited scan, no analysis information would be available. The graph would just display a single bar that represents all (unaudited) high priority issues.

3. To view the high priority issues based on a different attribute, select an item from the **View By** list.

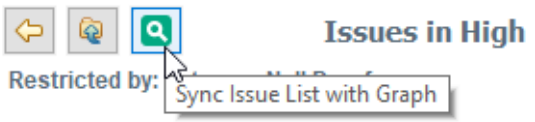


4. On the **Issues in High** bar graph, select a bar for a category that contains multiple issues.

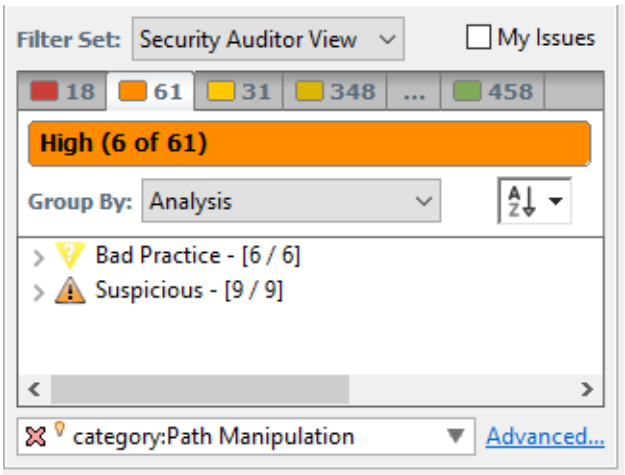


In the example shown here, the **Path Manipulation** bar is selected. You can see that of the six issues, two are marked as Suspicious and four are marked as Bad Practice.

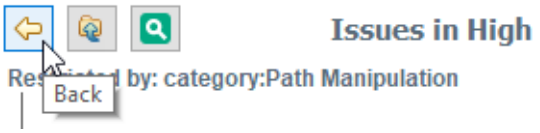
5. To synchronize the issues list with the displayed graphical view, click the **Sync Issue List with Graph** button.



The issue list in the **Issues** view now reflects the selections in the summary graph.



6. To return to the previous view in the summary graph, click the **Back** button.



7. To return to the original summary graph view (issues based on priority), click the **Return to Folder Graph** button.

1.5.1.3. Source Code tab

After you open a project in Fortify Audit Workbench, the top center view displays the **Project Summary** tab. After you select an issue in the **Issues** view to the left, Fortify Audit Workbench adds the source code tab to the top center view. This source code tab shows the code related to the issue selected in the **Issues** view.

If multiple nodes represent an issue in the **Analysis Trace** view (below the **Issues** view), the source code tab shows the code associated with the selected node.

From the source code tab, you can use the context menu commands to:

- Create new issues (**Create New Issue**).

  For more information, see Creating Issues for Undetected Vulnerabilities.

- Create a custom rule (**Generate Rule for Function**).

  For more information, see Writing Rules for Functions.

- Jump to the declaration of a function, class, variable, field, or an argument within source code that OpenText SAST translated (**Jump to Declaration**).
- Locate the file name and line number where a function occurs in the source code (**Find Usages**).

  The search results are displayed in the **Search** tab of the **Issue Auditing** view.

- Refresh the code displayed in source code tab (**Refresh**).

  You might need to use this if the file was modified outside of Fortify Audit Workbench.

- Customize the appearance in the source code tab such as fonts, colors, text edit settings, and so on (**Editor Preferences**).

1.5.1.3.1. About displayed source code

After you open an FPR file in Audit Workbench, the source code tab displays source code that is stored locally. If that source code was updated since the last scan, Fortify Audit Workbench displays the updated source code, even if the latest scan did not use that updated source code.

However, if that source code is updated after you open the FPR file and Fortify Audit Workbench has already started and searched for the source code (even if you close the FPR in Audit Workbench and then re-open it) Fortify Audit Workbench does not look for or display the updated source code. It displays the updated source code only after you quit, and then restart Fortify Audit Workbench.

1.5.1.4. Analysis Trace view

When you select an issue, the **Analysis Trace** view displays the relevant analysis trace. This is a set of program points that show how the analyzer found the issue. For dataflow and control flow issues, the set is presented in the order executed. For dataflow issues, this trace view presents the path that the tainted data follows from the source function to the sink function.

For example, when you select an issue that is related to potentially tainted dataflow, the **Analysis Trace** view shows the direction the dataflow moves in this section of the source code.

The **Analysis Trace** view uses the symbols described in the following table to show how the dataflow moves in this section of the source code or execution order.

Symbol	Description
	Data is assigned to a field or variable
	Information is read from a source external to the code such as an HTML form or a web address
	Data is assigned to a globally scoped field or variable
	A comparison is made
	The function call receives tainted data
	The function call returns tainted data
	Passthrough, tainted data passes from one place to another **Note** This is typically shown as `functionA(x : y)` to indicate that data is transferred from `x` to `y`. The `x` and `y` values are one of the following: • An argument index • `return`—The return value of a function • `this`—The instance of the current object • A specific object field or key
	An alias is created for a memory location
	Data is read from a variable
	Data is read from a global variable

	Tainted data is returned from a function
	A pointer is created
	A pointer is dereferenced
	The scope of a variable ends
	The execution jumps
	A branch is taken in the code execution
	A branch is not taken in the code execution
	Generic
	A runtime source, sink, or validation step
	Taint change

The **Analysis Trace** view can include inductions. Inductions provide supporting evidence for their parent nodes. Inductions consist of:

- A text node, displayed in italics as a child of the trace node. This text node is expanded by default.
- An induction trace, displayed as a child of the text node (a box surrounds the induction trace).

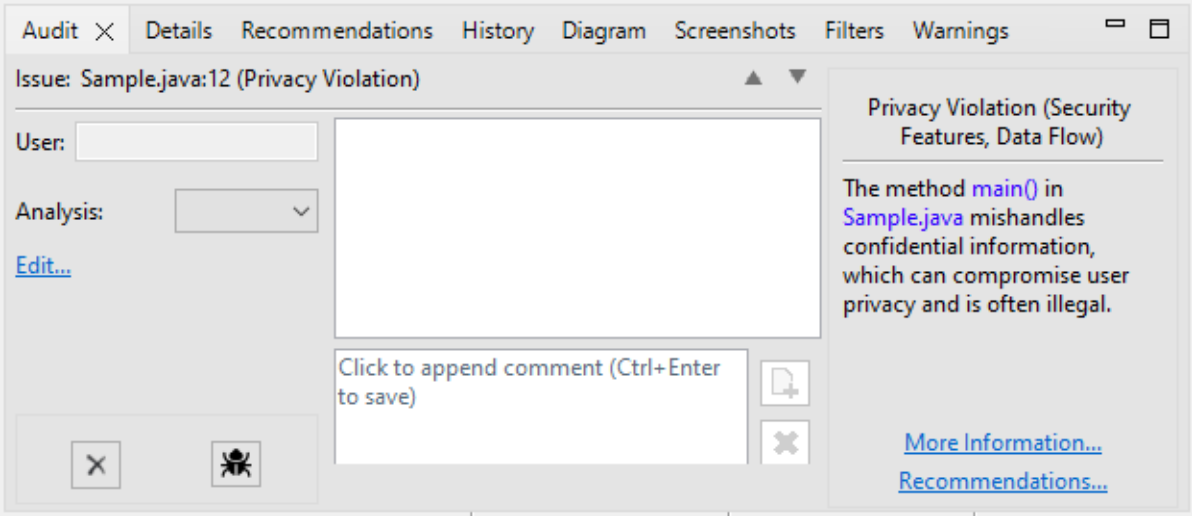
The italics and the box distinguish the induction from a standard subtrace. To display the induction reference information for that induction, click it.

1.5.1.5. Issue auditing view

The Issue Auditing view at the bottom center of the Fortify Audit perspective provides detailed information about each issue on the tabs.

Note

If any of the tabs are not visible, select **Options > Show View** to open them.



This section contains the following topics:

- Audit tab
- Details tab
- WebInspect Agent Details tab
- Recommendations tab
- History tab
- Diagram tab
- Filters tab
- Warnings tab

1.5.1.5.1. Audit tab

The **Audit** tab displays information about the selected issue and enables auditors to add an audit evaluation, comments, and custom tag values. The following table describes the tab interface elements.

Element	Description
Issue	Displays the issue location, including the file name and line number.
User	Displays the name of the user assigned to the issue if the results were uploaded to Application Security and a user was assigned.
Analysis	Displays the audit assessment for the selected issue. To change the assessment, select an item from the list. This is the primary tag. The default primary tag is **Analysis**, but it might be different depending on the custom tag settings in the project configuration. The valid values for **Analysis** are Not an Issue, Reliability Issue, Bad Practice, Suspicious, and Exploitable.
<custom_tagname>	Displays any custom tags if defined for the audit project. These are displayed below the primary tag. If the audit results were submitted to OpenText™ Fortify Audit Assistant in Application Security, then in addition to any other custom tags, the tab displays the following tags: • **AA_Prediction**—Exploitability level that Fortify Audit Assistant assigned to the issue. You cannot modify this tag value. • **AA_Confidence**—Confidence level from Fortify Audit Assistant for the accuracy of its **AA_Prediction** value. You cannot modify this tag value. • **AA_Training**—Whether to include or exclude the issue from Fortify Audit Assistant training. You can modify this value. For more information about Fortify Audit Assistant, see the *OpenText™ Application Security User Guide*.
	Suppresses the issue.
	Unsuppresses the issue (only visible if the issue is suppressed). Suppressed issues are hidden by default. To display suppressed issues, select **Options > Show Suppressed Issues**.
	Provides access to a supported bug tracker.
Comment	Appends additional information about the issue to the comment box.

Rule Information	Shows information, such as the category and kingdom that describes the issue.
More Information	Opens the **Details** tab (see Details Tab).
Recommendations	Opens the **Recommendations** tab (see Recommendations Tab).
Show merge conflicts	Shows merge conflicts in the **Comments** box that might exist after a merge of audit projects. This check box is available only if merge conflicts exist.

1.5.1.5.2. Details tab

The **Details** tab provides an abstract of the issue, a detailed explanation, and examples. The following table describes the tab sections.

Section	Description
Abstract/Custom Abstract	Summary of the issue, including any custom abstracts defined by your organization.
Explanation/Custom Explanation	Description of the conditions in which this type of issue occurs. This includes a discussion of the vulnerability, the constructs typically associated with it, how an attacker can exploit it, and the potential consequences of an attack. This section also includes any custom explanations defined by your organization.
Instance ID	Unique identifier for the issue.
Priority Metadata Values	Priority metadata values for this issue including impact and likelihood.
Legacy Priority Metadata Values	Legacy priority metadata values for the issue including severity and confidence.

Note

For more information about metadata values, see Estimating Impact and Likelihood with Input from Rules and Analysis.

1.5.1.5.3. WebInspect Agent Details tab

The **WebInspect Agent Details** tab displays information about runtime issues that OpenText™ DAST Agent discovered. The following table describes the tab sections.

Section	Description
Request	Shows the path of the request, the referrer address, and the method.
Stack Trace	Shows the order of methods called during execution and line number information. Blue, clickable code links are only displayed for OpenText SAST-scanned code.

1.5.1.5.4. Recommendations tab

The **Recommendations** tab displays suggestions and examples of how to secure the vulnerability or remedy the bad practice. The following table describes the tab sections.

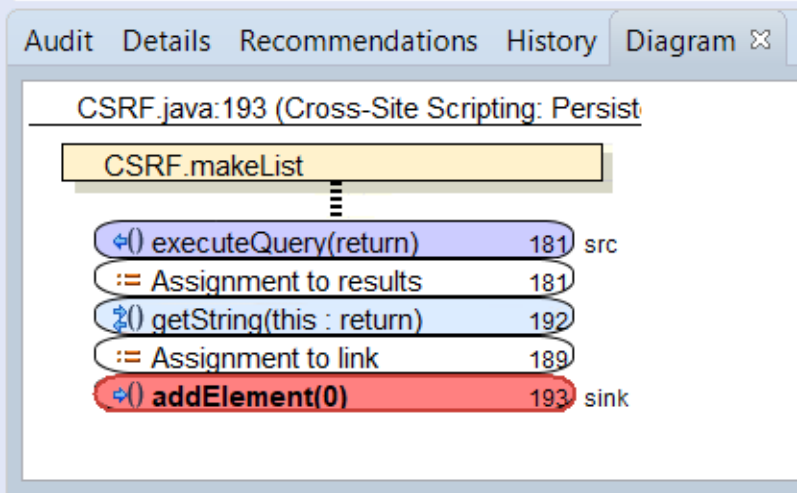
Section	Description
Recommendations/Custom Recommendations	Describes possible solutions for the selected issue. It can also include examples and recommendations defined by your organization.
Tips/Custom Tips	Provides useful information specific to the selected issue, and any custom tips defined by your organization.
References/Custom References	Lists references for the recommendations provided, including any custom references defined by your organization.

1.5.1.5.5. History tab

The **History** tab displays a complete list of audit actions, including details such as the time and date, and the name of the user who modified the issue.

1.5.1.5.6. Diagram tab

The **Diagram** tab displays a graphical representation of the node execution order, call depth, and expression type of the issue selected in the **Issues** view. This tab displays information that is relevant to the rule type. The vertical axis represents the execution order.



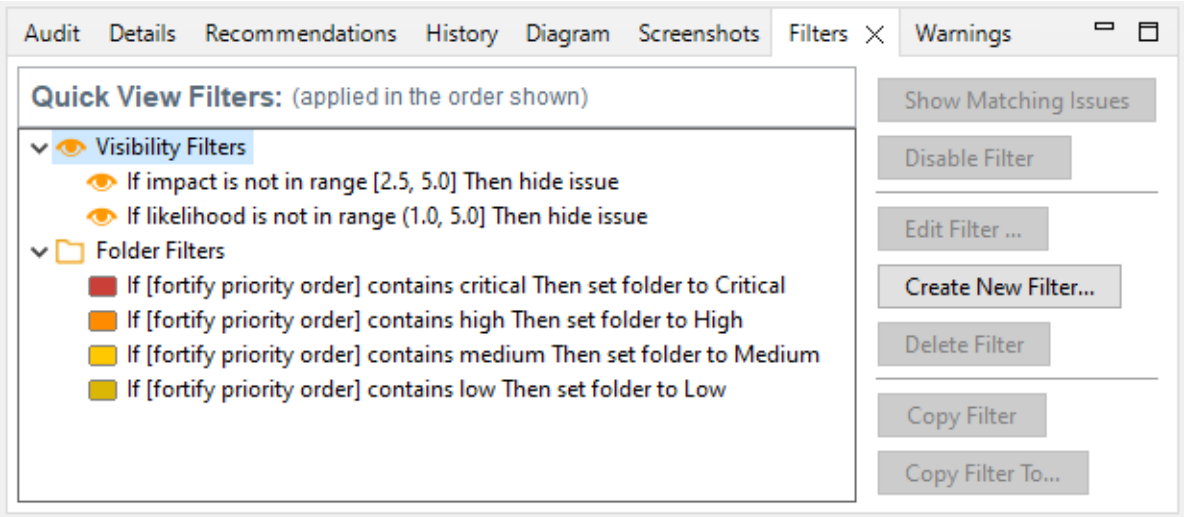
For dataflow issues, the trace starts with the first function to call the taint source, then traces the calls to the source (blue node) and ends the trace at the sink (red node). In the diagram, the source (src) and sink nodes are also labeled. A red X on a vertical axis indicates that the called function finished executing.

The horizontal axis shows the call depth. A line shows the direction that control is passed. If control passes with tainted data through a variable, then the line is red. If control passes without tainted data, the line is black.

The symbols used for the expression type of each node in the diagram are the same symbols used in the **Analysis Trace** view. For a description of the symbols, see Analysis Evidence View.

1.5.1.5.7. Filters tab

The **Filters** tab displays all the filters in the selected filter set.



The following table describes the options to create new filters.

Option	Description
Filters	Displays a list of the visibility and folder filters configured in the selected filter set where: • Visibility filters show or hide issues • Folder filters sort the issues into the folder tabs in the **Issues** view Right-click a filter to show issues that match the filter or to enable, disable, copy, or delete it.
If	Displays conditions for the selected filter. The first list displays issue attributes, the second specifies how to match the attribute, and third is the value the filter matches. **Note** This option is only visible when you create a new filter or edit an existing filter. In this case, a dialog box displays the **If** section.

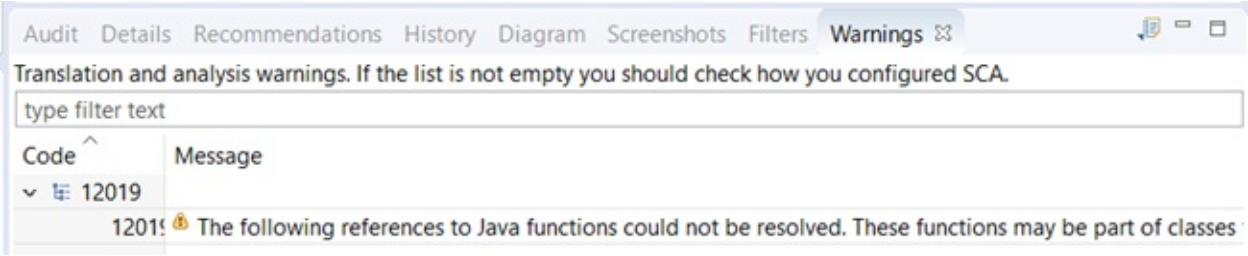
Then	Indicates the filter type, where **Hide Issue** is a visibility filter and **Set Folder to** is a folder filter. **Note** This option is only visible when you create a new filter or edit an existing filter. In this case, a dialog box displays the **Then** section.

See Also

Creating a Filter from the Issue Auditing View.

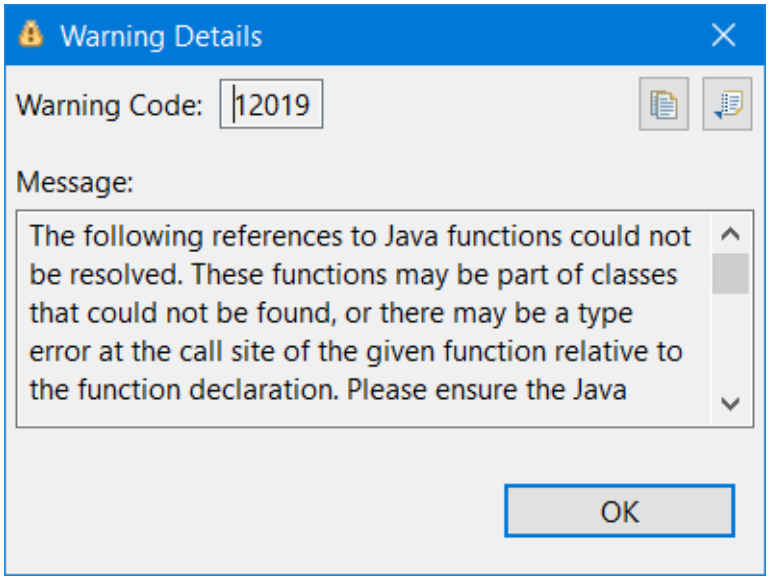
1.5.1.5.8. Warnings tab

The **Warnings** tab lists any warnings that occurred during the analysis.



A common source of warnings are missing references. To resolve this type of warning, make sure that the reference files are either within the project directory structure or in a location known to OpenText SAST. The scan can also issue a warning if a class has no functional content. In this case, the warning is not an issue because an empty class has no impact on a scan.

The following table describes the **Warnings** tab options.

Task	Procedure
See the complete message that is truncated on the tab.	• Double-click the message. 
Copy a warning message to the clipboard.	• Right-click a message, and then select **Copy**.
Save a warning message to a file.	1. Right-click a message, and then select **Export Entry**. 2. Type a name for the file, and then click **Save**. The file includes the audit project name, FPR file location, the warning code, and the warning message.

Save all the warning messages to a file.	1. Click the **Export Warnings** button  . 2. Type a name for the file, and then click **Save**. The file includes the project name, FPR file location, the warning codes, and the warning messages.
Search the warning message	Type the search text in the filter text box.
Modify the text message at the top of the tab.	1. Edit the `<fortify_working_dir>/config/tools/warnings-view.properties` file where `<fortify_working_dir>` is: ◦ Windows: `C:\Users\<username>\AppData\Local\Fortify` ◦ Non-Windows: `<userhome>/.fortify` 2. Edit the text following `message=` to the text you want to display in the **Warnings** tab. Close and reopen the **Warnings** tab to see the updated text.

1.5.1.6. Functions view

The **Functions** view in the top right shows how and where a function occurs in the source code, whether a security rule covers the function, and which rule IDs match the function. The **Functions** view can also list the functions that OpenText SAST identified as tainted source, and the functions that were not covered by rules in the last scan. For detailed information about the **Functions** view, see Using the Functions View.

1.5.1.7. Customizing the Issues view

You can customize the **Issues** view to determine which issues it displays.

To change the **Issues** view:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**.
3. To change your preferences on the **Appearance** tab, select or clear the check boxes described in the following table.

Preference	Description
Show Suppressed Issues	Displays all suppressed issues (off by default).
Show Removed Issues	Displays all issues detected in the previous scan, but are no longer evident in the new **Issues** view. When multiple scans are run on a project over time, vulnerabilities are often remediated or become obsolete. OpenText SAST marks these vulnerabilities as Removed Issues.
Show Hidden Issues	Displays all hidden issues.
Collapse Issues	Shows similar issues based on certain attributes under a shared parent node in the **Issues** view.
Use Short File Names	References the issues in the **Issues** view by file name only, instead of by relative path.
Show Category of Issue	Displays the category of an issue in the **Issues** view and the **Audit** tab.
Show Only My Issues	Displays only issues assigned to you.
Right justify 'All' Folder	Displays the **All** folder aligned on the right.
Display Name in Folder Tabs	Displays the name text in the folder tabs.
Show Abstract	Displays the abstract text in the **Audit** tab.
Show Comments	Displays comments in the **Audit** tab.
Show 'All' Folder in Project Summary Graph	Displays another bar in the chart on the **Summary** tab in the **Project Summary** view.
Include Comments	Displays the history items for comments on the **History** tab.

Parent Fill Opacity	Controls the opacity of the parent tile in Smart View. The setting ranges from 0% opaque on the left to 100% opaque on the right.

Note

To restore the default settings at any time, click **Reset Interface**.

4. To save your preferences, click **OK**.

1.5.2. Searching for issues

You can use the search box below the issues list to search for issues. After you perform a search, the label next to the folder name changes to indicate the number of issues that match the search as a subset of the total.

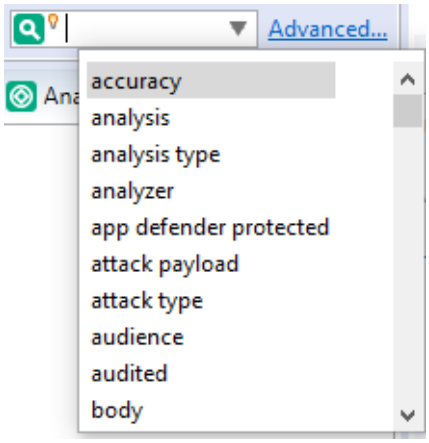
To perform a simple search, do one of the following:

- Type a search query in the search box and press **Enter**.

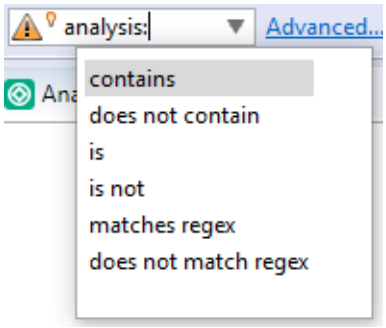
- To select a search query that you used before, click the arrow in the search box, and then select a search query from the list.

To get assistance with composing a search query, do the following:

1. Click in the search box, and then press **Ctrl**+ **Space**.



2. From the displayed list, double-click a search modifier to begin your search query.
3. For assistance to specify the comparison, with your cursor placed after the modifier in the search box, press **Ctrl**+ **Space**.



4. From the displayed list, double-click a comparison to add it to your search query.
5. Type the rest of the search query, and then press **Enter** to perform the search.

The **Issues** view lists all the issues that match your search string.

Creating complex search strings can involve several steps. If you type an invalid search query, the magnifying glass in the search box changes to a warning to notify you of the error. Click the warning sign to view information about the search query error.

The advanced search feature makes it easier to build complex search strings. For a description of this feature and instructions on how to use it, see Performing Advanced Searches.

See Also

Search Syntax

Search Modifiers

Search Query Examples

Performing Advanced Searches

1.5.2.1. Search syntax

To indicate the type of comparison to perform, wrap search terms with delimiters. The following table describes the syntax to use for a search query.

Comparison	Description
contains	Searches for a term without any special qualifying delimiters
equals	Searches for an exact match when the term is wrapped in quotation marks (`""`)
regex	Searches for values that match a Java-style regular expression delimited by a forward slash (`/`) Example: `/eas.+?/`
number range	Searches for a range of numbers using the standard mathematical interval notation of parentheses and/or brackets to indicate whether the endpoints are excluded or included, respectively Example: `(2,4]` indicates greater than two and less than or equal to four
not equal	Excludes issues specified by the string when you precede the string with the exclamation character (`!`) Example: `file:!Main.java` returns all issues that are not in `Main.java`

You can further qualify search terms with modifiers. The syntax for using a modifier is `<modifier>:<search_term>`.

A search query can contain multiple modifiers and search terms. If you specify more than one modifier, the search returns only issues that match all the modified search terms. For example, `file:ApplicationContext.java category:SQL Injection` returns only SQL injection issues found in `ApplicationContext.java`.

If you use the same modifier more than once in a search query, then the search terms qualified by those modifiers are treated as an `OR` comparison. For example, `file:ApplicationContext.java category:SQL Injection category:Cross-Site Scripting` returns SQL injection issues and cross-site scripting issues found in `ApplicationContext.java`.

For complex searches, you can also insert the AND or the OR keyword between your search queries. Note that AND and OR operations have the same priority in searches.

See Also

Search Modifiers

Search Query Examples

Searching for Issues

Performing Advanced Searches

1.5.2.2. Search modifiers

You can use a search modifier to specify to which issue attribute the search term applies. To use a modifier that contains a space in the name, such as the name of the custom tag, you must enclose the modifier in brackets. For example, to search for issues that are new, type `[issue age]:new`.

A search that is not qualified by a modifier matches the search query based on the following attributes: kingdom, primary rule id, analyzer, filename, severity, class name, function name, instance id, package, confidence, type, subtype, taint flags, category, sink, and source.

The following examples describe using the search with and without applying a search modifier:

- To apply the search to all modifiers, type a string such as `control flow`. This searches all the modifiers and returns any results that contain the "control flow" string.
- To apply the search to a specific modifier, type the modifier name and the string as follows: `analyzer:control flow`. This returns all results detected by the Control Flow Analyzer.

The following table describes the search modifiers. A few modifiers have a shortened modifier name indicated in parentheses. You can use either modifier string.

Search Modifier(Issue Attribute)	**Description**
`accuracy`	Searches for issues based on the accuracy value specified (0.1 through 5.0).
`analysis`	Searches for issues that have the specified audit analysis value such as `exploitable`, `not an issue`, and so on.
`[analysis type]`	Searches for issues based on the analyzer product such as `SCA` and `WEBINSPECT`.
`analyzer`	Searches the issues for the specified analyzer such as `control flow`, `data flow`, `structural`, and so on.
`[app defender protected]` (`def`)	Searches for issues based on whether Application Defender can protect the vulnerability category (`protected` or `not protected`).
`[attack payload]`	Searches for issues that contain the search term in the part of the request that caused the vulnerability for penetration test results.
`[attack type]`	Searches for issues based on the type of penetration test attack conducted (URL, parameter, header, or cookie).

`audience`	Searches for issues based on intended audience such as `dev`, `targeted`, `medium`, `broad`, and so on. **Note** This metadata is legacy information that is no longer used and will be removed in a future release. OpenText recommends that you do not use this search modifier.
`audited`	Searches the issues to find `true` if the primary tag is set and `false` if the primary tag is not set. The default primary tag is the Analysis tag.
`body`	Searches for issues that contain the search term in the HTTP message body in penetration test results, which is all the data that is transmitted immediately following the headers.
`bug`	Searches for issues that contain the search term in the information for the filed bug. **Note** This information is discarded each time you restart Fortify Audit Workbench.
`category` (`cat`)	Searches for the specified category or category substring.
`class`	Searches for issues based on the specified class name.
`codesnippet`	Searches for the specified string within the few lines of code that are stored for each vulnerability by default. If code snippets were excluded from the scan results during the analysis, then the search will not return any results.
`comments` (`comment`, `com`)	Searches for issues that contain the search term in the comments added to the issue.
`commentuser`	Searches for issues with comments from a specified user.
`confidence` (`con`)	Searches for issues that have the specified confidence value 0.1 through 5.0 (legacy metadata).
`cookies`	Searches for issues that contain the search term in the cookie from the HTTP query for penetration test results.

`correlated`	Searches for issues based on whether the issues are correlated with another analyzer.
`[correlation group]`	Searches for issues based on whether the issues are in the same correlation group.
`<custom_tagname>`	Searches for issues based on the value of the specified custom tag. You can search a list-type custom tag using a range of values. The values of a list-type custom tag are an enumerated list where the first value is 0, the second is 1, and so on. You can use the search syntax for a range of numbers to search for ranges of list-type custom tag values. For example, `analysis:[0,2]` returns the issues that have the values of the first three analysis values, 0, 1, and 2 (Not an Issue, Reliability Issue, and Bad Practice). To search for a specific date in a date-type custom tag, specify the date in the format: yyyy-mm-dd. To search for issues that have no value set for a custom tag, use `<none>` for the search term. For example, to search for all issues that have no value set in the custom tag labeled Target Date, type: `[Target Date]:<none>`.
`dynamic`	Searches for issues that have the specified dynamic hot spot ranking value.
`[engine priority]`	Searches for issues based on the original priority value determined by the engine that identified the issue.
`file`	Searches for issues where the primary location or sink node function call occurs in the specified file path.
`filetype`	Searches for issues based on the file type such as `asp`, `csharp`, `java`, `jsp`, `xml`, and so on.
`[fortify priority order]`	Searches for issues that have a priority level that matches the specified issue priority. Valid values are `critical`, `high`, `medium`, and `low`.
`headers`	Searches for issues that contain the search term in the request header for penetration test results.
`historyuser`	Searches for issues that have audit data modified by the specified user.
`[http version]`	Searches for issues based on the specified HTTP version such as `HTTP/1.1`.

`impact`	Searches for issues based on the impact value specified (0.1 through 5.0).
`[instance id]`	Searches for an issue based on the specified instance ID.
`[issue age]`	Searches for the issue age, which is `new`, `updated`, `reintroduced`, or `removed`.
`[issue state]`	Searches for audited issues based on whether the issue is an open issue or not an issue (determined by the level of analysis set for the primary tag).
`kingdom`	Searches for all issues in the specified kingdom.
`likelihood`	Searches for issues based on the specified likelihood value (0.1 through 5.0).
`line`	Searches for issues on the primary location line number. For dataflow issues, the value is the sink line number. See also sourceline.
`manual`	Searches for issues that were manually created by penetration test tools, and not automatically produced by a web crawler such as OpenText™ Dynamic Application Security Testing.
`[mapped category]`	Searches for issues based on the specified category that is mapped across the various analyzers (OpenText SAST, OpenText DAST, and OpenText DAST Agent).
`maxconf`	Searches for all issues that have a confidence value equal to or less than the number specified as the search term.
`maxVirtConf`	Searches for dataflow issues that have a virtual call confidence value equal to or less than the number specified as the search term.
`<metadata_listname>`	Searches for issues based on the value of the specified metadata external list. Metadata external lists include `[owasp top ten <year>]`, `[cwe top 25 <version>]`, `[pci ssf <version>]`, `[stig <version>]`, and others.
`method`	Searches for issues based on the method, such as `GET`, `POST`, `DELETE`, and so on.
`minconf`	Searches for all issues that have a confidence value equal to or greater than the number specified as the search term.

`min_virtual_call_confidence` (`virtconf`, `minVirtConf`)	Searches for dataflow issues that have a virtual call confidence value equal to or greater than the number specified as the search term.
`package`	Searches for issues where the primary location occurs in the specified package or namespace. For dataflow issues, the primary location is the sink function.
`parameters`	Searches for issues that contain the search term in the HTTP query parameters.
`primary`	Searches for issues that have the specified primary tag value. By default, the primary tag is the Analysis tag.
`[primary context]`	Searches for issues where the primary location or sink node function call occurs in the specified code context. See also sink and [source context].
`primaryrule` (`rule`)	Searches for all issues related to the specified sink rule.
`probability`	Searches for issues based on the probability value specified (1.0 through 5.0).
`[remediation effort]`	Searches for issues based on the remediation effort value specified. The valid values are whole numbers from 1.0 to 12.0.
`[request id]`	This attribute is not currently used.
`response`	Searches for issues that contain the search term in the response from the protocol used in penetration test results.
`ruleid`	Searches for all issues reported by the specified rule IDs used to generate the issue source, sink and all passthroughs.
`[secondary requests]`	This attribute is not currently used.
`severity` (`sev`)	Searches for issues based on the specified severity value (legacy metadata).
`shortfilename`	Searches for issues where the primary location or sink node function call occurs in file names that contain the specified search term, but not anywhere in its full path. For full path matches, use the modifier file.
`sink`	Searches for issues that have the specified sink function name. See also [primary context].

`source`	Searches for dataflow issues that have the specified source function name. See also [source context].
`[source context]`	Searches for dataflow issues that have the source function call contained in the specified code context. See also source and [primary context].
`sourcefile`	Searches for dataflow issues with the source function call that the specified file contains. See also file.
`sourceline`	Searches for dataflow issues having taint source entering the flow on the specified line. See also line.
`status`	Searches issues that have the status reviewed, not reviewed, or under review.
`suppressed`	Searches for issues based on whether they are suppressed.
`taint`	Searches for issues that have the specified taint flag.
`trace`	Searches for issues that have the specified string in the dataflow trace.
`tracenode`	Enables you to search on the nodes within an issue's analysis trace. Each tracenode search value is a concatenation of the tracenode's file path, line number, and additional information.
`tracenodeAllPaths`	Searches for the specified value in all the steps of analysis trace.
`trigger`	Searches for issues that contain the search term in the part of the response that shows that a vulnerability occurred for penetration test results.
`url`	Searches for issues based on the specified web address.
`user`	Searches for issues assigned to the specified user.

1.5.2.3. Search query examples

The following table contains search query examples.

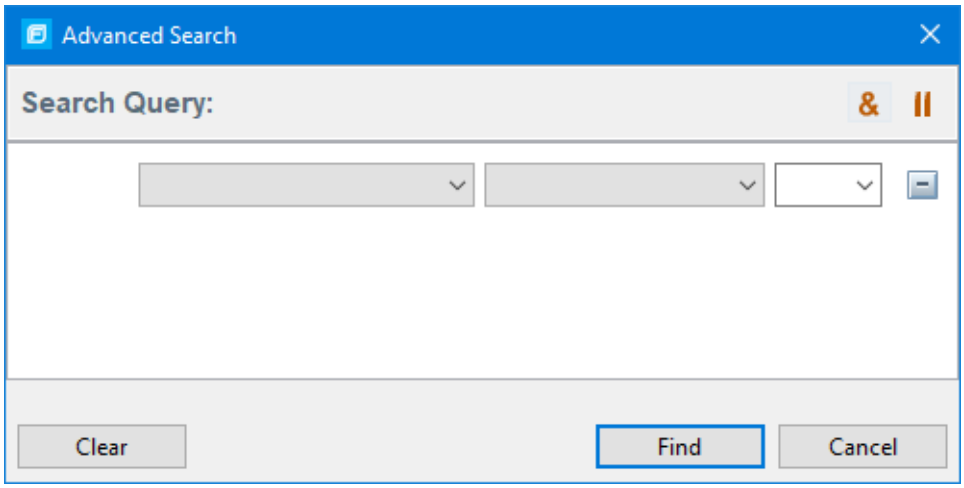
Search Target	Query
All privacy violations in file names that contain `jsp` with `getSSN()` as a source	`category:privacy violation source:getssn file:jsp`
All file names that contain `com/test/123`	`file:com/test/123`
All paths that contain traces with `mydbcode.sqlcleanse` as part of the name	`trace:mydbcode.sqlcleanse`
All paths that contain traces with `cleanse` as part of the name	`trace:cleanse`
All issues that contain `cleanse` as part of any modifier	`cleanse`
All suppressed vulnerabilities with `asdf` in the comments	`suppressed:true comments:asdf`
All categories except for SQL Injection	`category:!SQL Injection`
All issues that have a value specified for a custom tag labeled `version`	`version:!<none>`

1.5.2.4. Performing advanced searches

Use the advanced search feature to build complex search strings.

To use the advanced search feature:

1. To the right of the search box, click **Advanced**.



2. To create your search query:
    1. From the list on the left, select a search modifier.
    2. From the middle list, select the comparison and type.
    3. From the list on the right, select a search term.

    The list for the search term includes the known values in the current scan for the specified attribute. However, you can type any value into this box. To specify an unqualified search term, select **Any Attribute** from the bottom of the modifier list.
3. To add another query row, do one of the following:
    - To add an AND query row, in the top right corner of the dialog box, click the **AND** button .
    - To add an OR query row, in the top right corner of the dialog box, click the **OR** button .
4. Add as many query rows as you need for the search query.
5. To delete a row, to the right of the row, click the **Delete** button . To remove all rows, click **Clear**.
6. To change a query row condition, double-click the current (underlined) query row operator **AND** or **OR**.

    In the following example, you can double-click **AND** to change the query operator to **OR**.

7. Click **Find**.

Note

As you build your search string, the Advanced Search dialog box displays any errors in the status below the search string builder. **Find** is only enabled when the search query is error free.

1.5.3. Working with issues

This section describes how to use Fortify Audit Workbench to review issues.

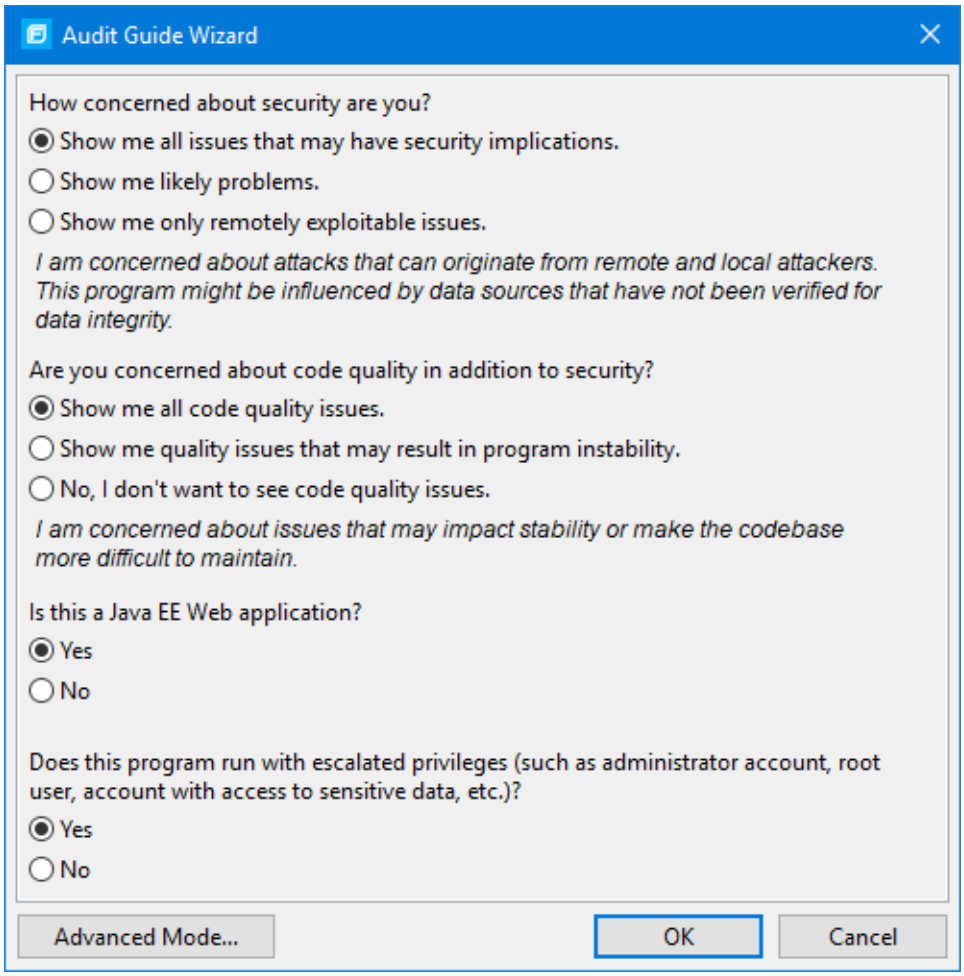
This section contains the following topics:

1.5.3.1. Filtering issues with Audit Guide

You can use the Audit Guide Wizard to filter vulnerability issues in your audit project based on a set of security-related questions.

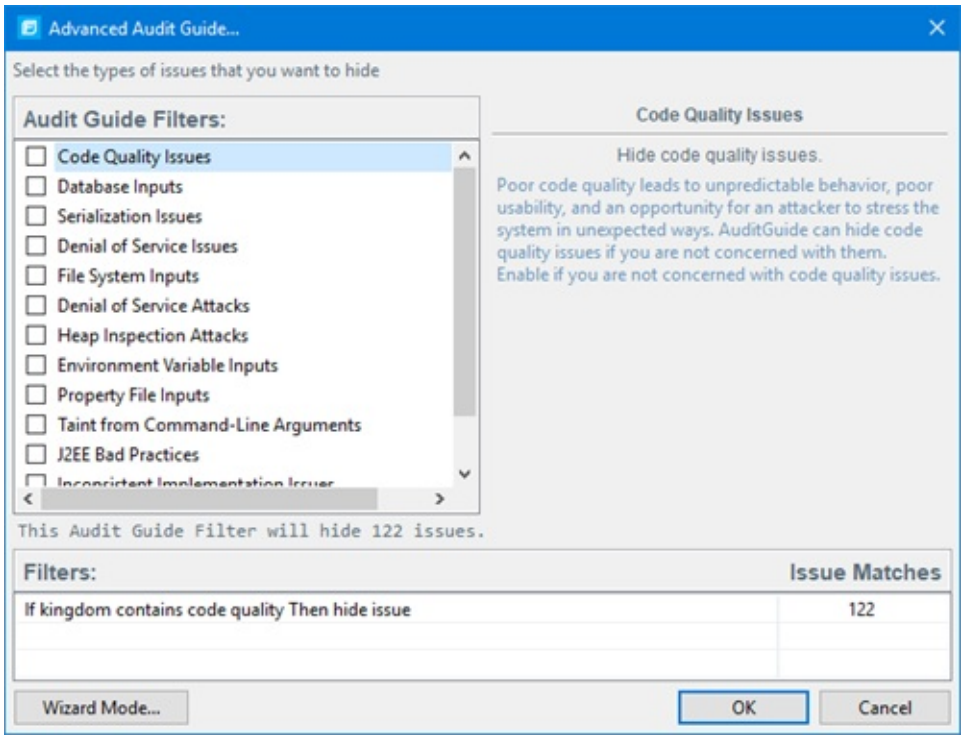
To use the Audit Guide:

1. Select **Tools > Audit Guide**.



2. Make your selections for the types of issues you want to display.
3. To use the advanced filter options, click **Advanced Mode**.

   The Advanced Audit Guide dialog box opens.



1. In the **Audit Guide Filters** list, select the types of issues you want to filter out and ignore.

   As you select items in the **Audit Guide Filters** list, the Audit Guide Wizard also displays the filter details for the selected filter type in the **Filters** table, including the number of issues that match each filter.

2. To see a description of an issue type, click its name in the **Audit Guide Filters** list.

   The Audit Guide Wizard displays a description to the right of the list.

4. Click **OK** to apply your filter selections.

1.5.3.2. Grouping issues

The items visible in the **Issues** view vary depending on the selected issue attribute. The attribute you select from the **Group By** list sorts issues in all visible folders into subfolders.

Use the issue attributes to group and view the issues in different ways. You can view issues with any of the available issue attributes, and you can create and edit customized groups. The following table describes the available issue attributes.

Issue Attribute	**Description**
Analysis	Groups issues by the audit analysis, such as Suspicious, Exploitable, and Not an Issue.
Analysis Type	Groups issues by analyzer product, such as SCA, WEBINSPECT, and SECURITYSCOPE (OpenText DAST Agent).
Analyzer	Groups issues by analyzer group, such as Configuration, Control Flow, Data Flow, Pentest, Semantic, and Structural.
App Defender Protected	Groups issues by whether Application Defender can protect the vulnerability category.
Category	Groups issues by vulnerability category. This is the default grouping.
Category Analyzer	Groups issues by category and then by analyzer.
<custom_tagname>	Groups issues by custom tag.
File Name	Groups issues by file name.
Fortify Priority Order	Groups issues by Critical, High, Medium, and Low based on the issue priority.
Kingdom	Groups issues by the Seven Pernicious Kingdoms classification.
Manual	Groups issues by whether they were manually created by penetration test tools, and not automatically produced by a web crawler such as OpenText DAST.
<metadata_listname>	Groups issues by the alternative metadata external list names (for example, OWASP Top 10 *<year>*, CWE Top 25 *<year>*, PCI SSF *<version>*, STIG *<version>*, and others).

New Issue	Shows which issues are new since the last scan. For example, if you run a new scan, any issues that are new are displayed in the tree under the **Issue New** group and the others are displayed in the **Issue Updated** group. Issues not found in the latest scan are displayed in the **Issue Removed** group.
New Issue by Category	Groups issues that are new since the last scan and then by category. See also New Issue.
Package	Groups issues by package or namespace. Nothing is shown for projects to which this option does not apply, such as C projects.
Priority by Category	Groups issues by Fortify Priority Order and then by category.
Shared Trace Node	Groups issues by the most common paths determined by the Dataflow Analyzer. This grouping helps to maximize the number of issues that you can address by updating one location in the code.
Sink	Groups issues that share the same dataflow sink function.
SmartView	Groups issues with a multiple-level grouping based on the last setting applied in SmartView. By default, groups issues by category, and then by Shared Trace Nodes.
Source	Groups issues that share the same dataflow source functions.
Source File Type	Groups issues by file type. For dataflow issues, the file contains the sink function. **Note** Issues in files with different file extensions that are the same source file type are grouped together (for example, issues in files with the extensions: `html`, `htm`, and `xhtml` are grouped under html).
Taint Flag	Groups issues by the taint flags that they contain.
<none>	Displays a flat view without any grouping.
Edit	Select **Edit** to create a custom grouping option.

The following table describes additional grouping options that are available when you create a custom grouping option (see Creating a Custom Group By Option).

Option	**Description**

Issue State	Groups audited issues by whether the issue is an open issue or not an issue based on the level of analysis set for the primary tag. Values equivalent to Suspicious and Exploitable are considered open issue states.
Primary Context	Groups issues where the primary location or sink node function call occurs in the same code context.
Source Context	Groups dataflow issues that have the source function call contained in the same code context.
Source File	Groups dataflow issues by the source code file where the taint originated.
Status	Groups issues by the audit status (**Reviewed**, **Unreviewed**, or **Under Review**)
URL	Groups dynamic issues by the request web address.

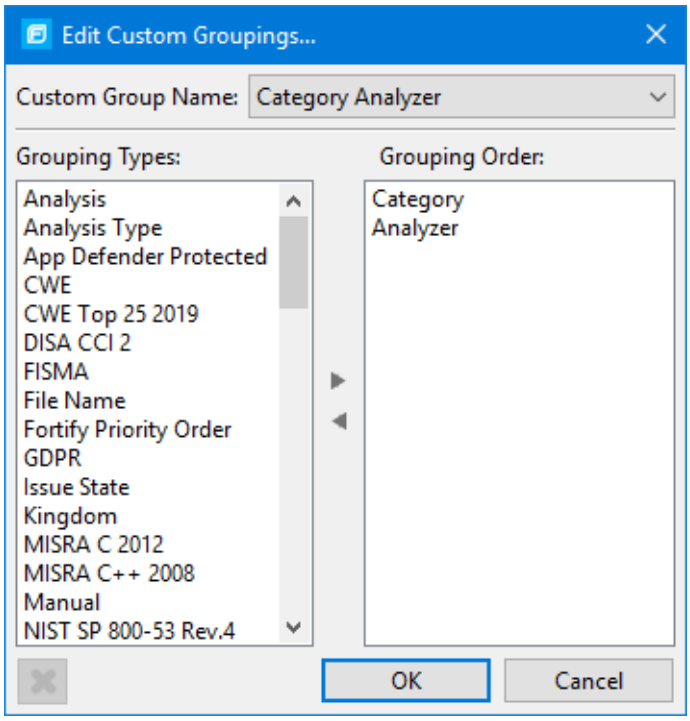
1.5.3.2.1. Creating a custom grouping option

You can create a custom grouping option that groups issues in a hierarchical format in sequential order based on selected attributes.

To create a new grouping option:

1. In the **Group By** list, select **Edit**.

   The Edit Custom Groupings dialog box opens.

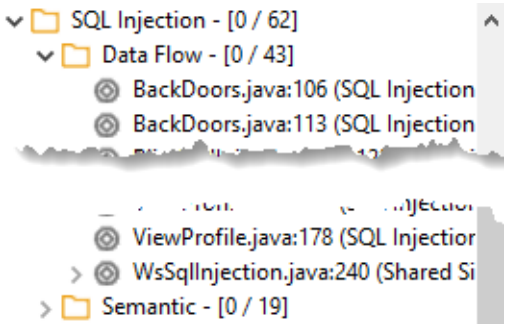


2. To create a custom group by option, do the following:

   1. Select **Create New** from the **Custom Group Name** list.
   2. In the Enter Value dialog box, type a name for the new custom group.
   3. Click **OK**.

3. From the **Grouping Types** list on the left, select a grouping type, and then click the right arrow to move the option to the **Grouping Order** column.

   For example, selecting **Category** and then **Analyzer** creates a list that has top-level nodes that contain the category of the issue, such as SQL Injection, with the issues grouped below by analyzer (such as Dataflow or Semantic).



4. Repeat step 3 to select additional grouping types.
5. To change the order of the grouping types:
    1. In the **Grouping Order** list, select the grouping type that you want to move up or down in the grouping order.
    2. Right-click the selected grouping type, and then select **Move Up** or **Move Down.**
6. To delete a custom grouping, click the **Delete** button .

1.5.3.3. Using Smart view

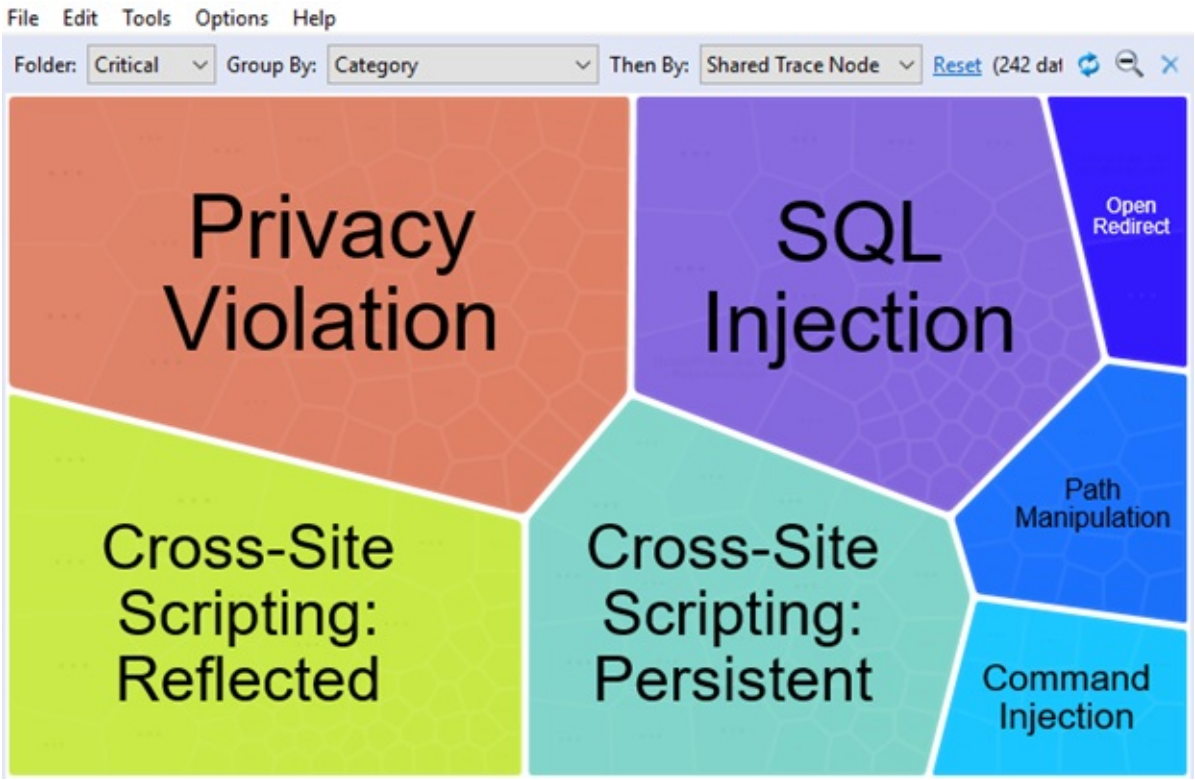
Smart View provides a visual representation of the dataflow issues in your code so that you can quickly identify optimal remediation or triage strategies for multiple issues at once. To use Smart View:

1. Select **Tools > Smart View**.

   Note

   Smart View uses the currently selected folder and grouping option.

   The number of issues for the currently selected folder and grouping selection determines the relative size of the Smart View tiles.

   Note

   You can adjust the opacity of the parent tile. For instructions, see Customizing the Issues View.

2. To filter the issues that are displayed, you can:

   - Select a grouping from the **Folder** list (for example, **Critical**, **High**, **Medium**, **Low**, or **All**)

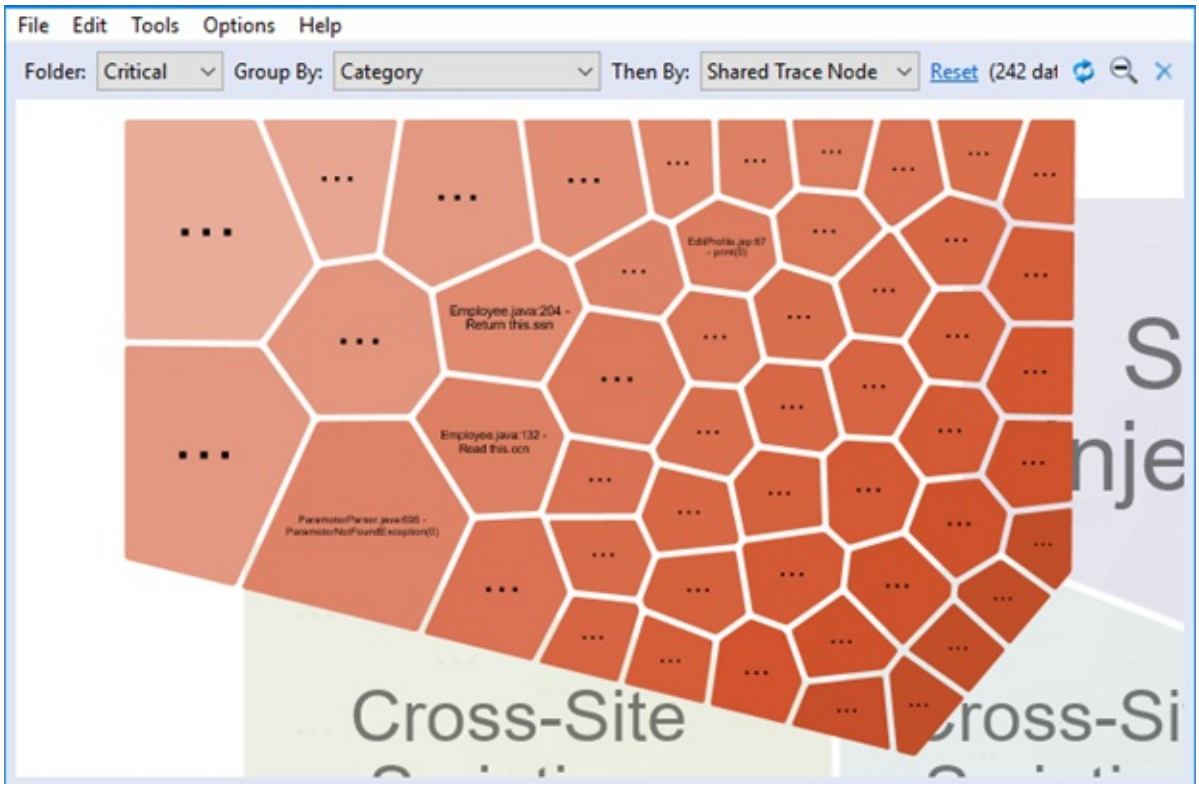
     This list includes any custom folders and folders specific to the current filter set.

- Select a subfolder in the **Group By** list to further sort the issues.
- From the **Then By** list, select whether you are interested in viewing data by **Source**, **Sink**, or **Shared Trace Node**.

  Shared Trace Node is a node (or function) in the code that multiple dataflows pass through.

> **Note**
>
> To reset the display to the default Smart View settings, click **Reset**. This resets **Folder** to **Critical**, **Group By** to **Category**, and **Then By** to **Shared Trace Node**.

3. Click a tile to see the issues in each grouping.



> **Note**
>
> To return to the initial grouping level at any time, click the **Zoom out** button .

4. To see the issues in the auditing interface that share a common dataflow trace node, source, or sink, move your cursor over the tile you are interested in, and then click **View Issues**.

   This closes Smart View and returns you to the auditing interface and displays the issues for the selected grouping. The **Group By** category is set to **SmartView** to indicate that you are viewing the results filtered by the Smart View selection. The search box contains the Smart View symbol and the Smart View search criteria:

.

To return to the primary auditing interface at any time, click the **Exit Smart View** button  .

1.5.3.4. Selectively displaying issues assigned to you

To display only issues assigned to you in the **Issues** view, do one of the following:

- Select the **My Issues** check box.
- Select **Options > Show Only My Issues**.

1.5.3.5. About suppressed, removed, and hidden issues

You can control whether the **Issues** view lists the following types of issues:

- *Suppressed* issues—As you assess successive scans of an application version, you might want to completely suppress some exposed issues. It is useful to mark an issue as suppressed if you are sure that the specific vulnerability is not, and will never be, an issue of concern. You might also want to suppress warnings for specific types of issues that are not a high priority or of immediate concern. For example, you can suppress issues that are fixed, or issues that you plan not to fix. Suppressed issues are not included in the group totals shown in the **Issues** view.
- *Removed* issues—As multiple scans are run on a project over time, issues are often remediated or become obsolete. As it merges scan results, OpenText SAST marks issues that were uncovered in a previous scan, but are no longer evident in the most recent OpenText SAST analysis results as Removed. Removed issues are not included in the group totals shown in the **Issues** view.
- *Hidden* issues—You typically hide a group of issues temporarily so that you can focus on other issues. For example, you might hide all issues except those assigned to you. The individuals assigned to address the issues you have hidden in your view can still access them. The group totals displayed in the **Issues** view include hidden issues.

To hide or show suppressed, removed, or hidden issues in the **Issues** view, from the **Options** menu, select (or clear) one or more of the following:

- **Show Suppressed Issues**
- **Show Removed Issues**
- **Show Hidden Issues**

1.5.3.6. Creating attribute summary tables for multiple issues

You can create a summary table of attributes (for example, in spreadsheet software such as Excel or Google Sheets) for any number of issues that you select from the **Issues** view. You specify the format options, select the issues, and then paste the comma-delimited data into a spreadsheet program to create the summary table.

The table can contain an attributes column followed by a single values column for every issue selected or, the table can display one row per attribute and its corresponding values. Alternatively, you can specify a customized table layout for the values that you copy to your spreadsheet program.

To create a spreadsheet table that contains an attributes column followed by a single values column for each selected issue:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab.
3. Under **Multiple Issues Copy Format**, leave the **[h] List issues in columns** option selected.
4. Select the attributes you want to include from the **Include immutable attributes**, **Include mutable attributes**, and **Include custom tags** check boxes.
5. Click **OK**.
6. From the **Issues** view, use the **Ctrl** or **Shift** key and select all the issues you want to include in a table.
7. With the issues selected, press **Ctrl** + **Alt** + **Shift** + **C**.
8. Start the spreadsheet software, and then paste (**Ctrl** + **V**) the copied data into a single column.

To create a spreadsheet table that displays one row per attribute and its values:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab.
3. Under **Multiple Issues Copy Format**, select the **[v] List issues in rows** option.
4. Select the attributes you want to include from the **Include immutable attributes**, **Include mutable attributes**, and **Include custom tags** check boxes.
5. Click **OK**.
6. From the **Issues** view, use the **Ctrl** or **Shift** key and select all the issues you want to include in a table.
7. With the issues selected, press **Ctrl** + **Alt** + **Shift** + **C**.
8. Start the spreadsheet software, and then paste (**Ctrl** + **V**) the copied data into a single column.

To create a customized table layout for the values that you copy to a spreadsheet program:

1. Select **Options > Options**.

2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab.
3. Under **Multiple Issues Copy Format**, select the **Format manually** option.
4. In the **Attribute value format** box, use the string described in the following table to specify the data layout, format, and separators for the values you want to copy.

String	Function
`[h]`	Columnar format - Attributes are inserted in a single column and the spreadsheet table expands to the right (horizontally) with a new column added for each issue copied in.
`[v]`	Row format - Attributes are inserted in a single row (table header) and a new row populated with values is added for each issue added (table expands vertically).
`%s`	Textual data (you can use the complete `java.util.Formatter` syntax). See the `java.util.Formatter` documentation at https://docs.oracle.com/en/java/index.html.
`,` `;` or tab	Separator symbol - To import the copied value into most spreadsheet programs, you must specify the separator to use in the format field.
`'…'`	Apply the preceding format string to all elements in the selection. This is only valid if the format specification starts with `[h]` or `[v]`.
`%n`	Line separator (platform independent), whether it is the last value for an issue in a row formatted table `[v]` or it is the last value of a given attribute in a columnar formatted table `[h]`.

   For example, to specify which specific attributes you want to copy with the row format (`[v]`), use `[v]%file$s,%category$s,%fortify priority order$s%n`. This copies the three attributes for each selected issue.

5. To see the result of your syntax, look under **Result example**.

   The example shown changes as you change the value in the **Attribute Value Format** box.

   Note

   Examples are not available for complex manual formats.

6. Select the attributes you want to include from the **Include immutable attributes**, **Include mutable attributes**, and **Include custom tags** check boxes.
7. Click **OK**.

1.5.4. About issue templates

OpenText SAST produces comprehensive results for source code analysis. On large codebases, these results can be overwhelming. The issue template assigned to your projects enables you to sort and filter the results to best suit your needs. The filtering and sorting mechanisms appropriate during a given phase in the development process can change depending on the phase of development. Similarly, the filtering and sorting mechanisms might vary depending on the role of the user.

You can sort issues by grouping them into folders, which are logically defined sets of issues presented in the tabs on the **Issues**. You can further customize the sorting to provide custom definitions for the folders into which the issues are sorted. You can provide definitions for any number of folders, whose contents are then defined by filters. Filters can either alter the visibility of an issue or place it into a folder. When used to sort issues into folders, you define the nature of the issues that appear in the customized folders.

You group filters into filter sets and then use the filter sets to sort and filter the issues displayed. An issue template can contain definitions for multiple filter sets. Using multiple filter sets in an audit project enables you to quickly change the sorting and visibility of the issues you are auditing. For example, the default issue template used in the interface provides two filter sets. These filter sets provide an increasingly restrictive view of security-related issues. Defining multiple filter sets for an audit project enables different views for different users, and a customized view does not affect any other views.

In addition to providing sorting and filtering mechanisms, you can customize the auditing process by defining custom tags in the issue template. Auditors associate custom tags with issues during auditing. For example, you can use custom tags to track impact, severity, or priority of an issue using the same names and values used to track these attributes in other systems, such as a bug tracker application.

Issue templates contain the following settings:

- Folder filters—Control how issues are sorted into the folders
- Visibility filters—Control which issues are shown and hidden
- Filter sets—Group folder and/or visibility filters
- Folder properties—Name, color, and the filter set in which it is active
- Custom tags—Specify which audit tags are displayed and the values for each

The issue template applied to an audit project is determined using the following preference order:

1. Template that exists in the audit project
2. Template in `<tools_install_dir>/Core/config/filters/defaulttemplate.xml`
3. Template in `<sca_install_dir>/Core/config/rules/defaulttemplate.xml` or `projecttemplate.xml`
4. Embedded Fortify default template

1.5.5. Configuring custom filter sets and filters

If the filter sets available in Fortify Audit Workbench do not exactly suit your needs, you can create your own, either by using the filter wizard, or by copying and then modifying an existing filter set.

If you are performing collaborative audits in Application Security, you can synchronize your custom filters with Application Security. For more information, see Committing Filter Sets and Folders and Synchronizing Filter Sets and Folders.

This section contains the following topics:

1.5.5.1. Creating a new filter set

To create a new filter set, copy an existing set and modify the settings.

To create a new filter set:

1. Select **Tools > Project Configuration**.
2. Select the **Filter Sets** tab.
3. Next to **Filter Sets**, click the **Add Filter Set** button  .

   The Add New Filter Set dialog box opens.

4. Type a name for the new filter set.
5. Select an existing filter set to copy.
6. Click **OK**.

A new filter set with the same folders, visibility filters, and folder filters as the copied filter set is created.

See Also

Creating a Filter from the Issues View

1.5.5.2. Creating a filter from the Issues view

When a folder list includes an issue that you want to hide or direct to another folder, you can create a new filter using the filter wizard. The wizard displays all the attributes that match the conditions in the filter.

Note

To find the filter that directed the issue to the folder, right-click the issue, and then select **Why is this issue here?** To find the filter that hid an issue, right-click the issue, and then select **Why is this issue hidden?**

To create a new filter from an issue:

1. In the **Issues** view, select a filter set from the **Filter Set** list.
2. Right-click an issue, and then select **Create Filter**.

   The Create Filter dialog box lists suggested conditions.
3. To see all the conditions, select the **Show all conditions** check box.
4. Select the conditions you want to use in the filter.

   You can fine tune the filter later by modifying it on the **Filter** tab.
5. Select the type of filter you want to create, as follows:
   - To create a visibility filter, select **Hide Issue**.
   - To create a folder filter, select **Set Folder to**, and then select the folder name or select **Other Folder** to add an existing folder or create a new one.

     A new folder is displayed in this filter set only.
6. Click **Create Filter**.

   The wizard places the new filter at the end of the filter list. For folder filters, this gives the new filter the highest priority. Issues that match the new folder filter appear in the targeted folder.
7. (Optional) For folder filters, drag the filter higher in the folder filter list to change the priority.

The issues are sorted with the new filter.

Note

The filter is only created in the selected filter set.

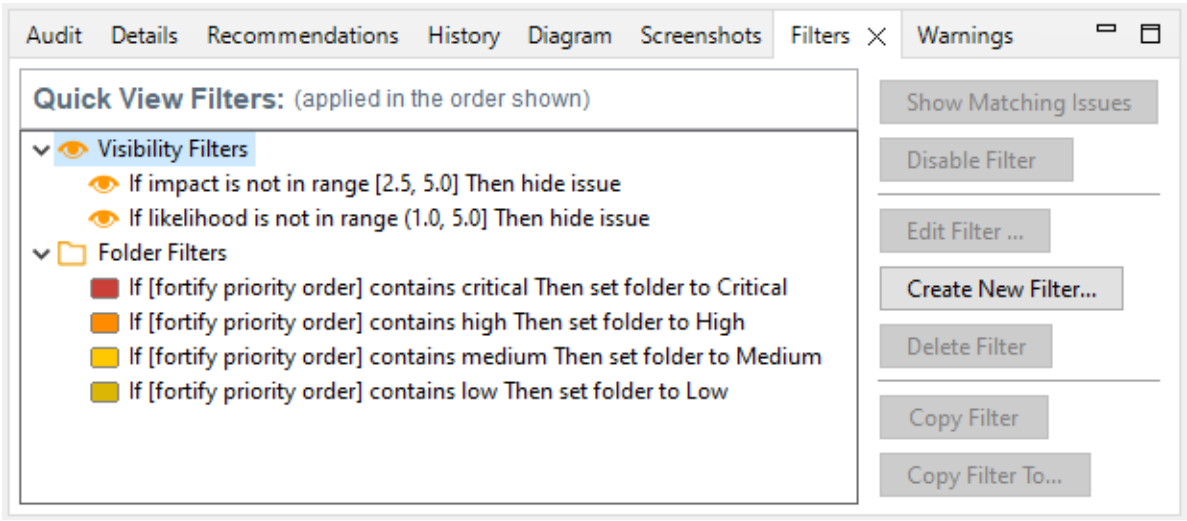
1.5.5.3. Creating a filter from the Issue Auditing view

Use the **Filters** tab in the Issue Auditing view to create visibility filters and folder filters.

Folder filters are applied in order and the issue is directed to the last folder filter it matches in the list.

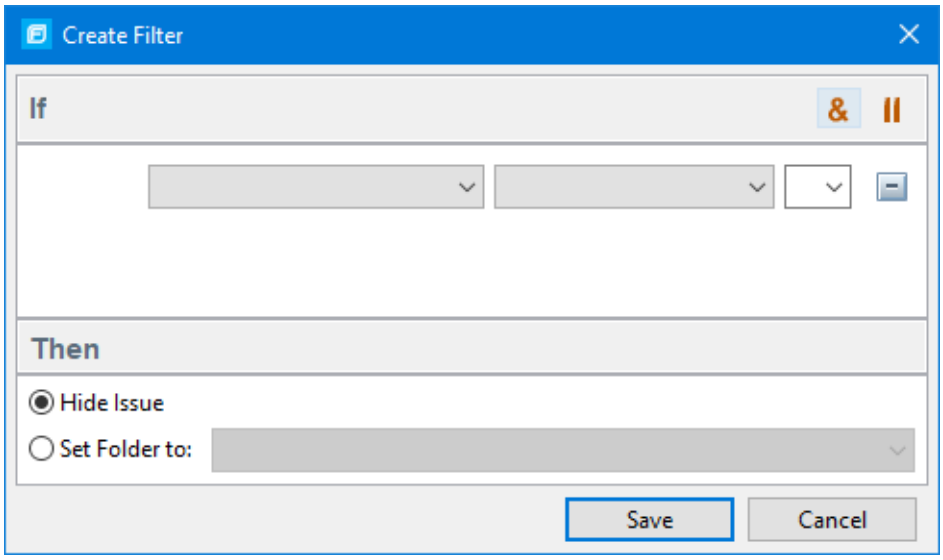
To create a new filter on the **Filters** tab:

1. From the **Filter Set** list, select a filter set.
2. Select the **Filters** tab in the Issue Auditing view.



3. Right-click **Visibility Filters** or **Folder Filters**, and then select **Create New Filter**.

   The Create Filter dialog box opens.



4. From the first list, select an issue attribute.

   For a description of the available issue attributes, see Search Modifiers. The second list is then automatically populated with the available comparison methods.

5. From the second list, select how to match the value.

   The third list contains the possible values for the attribute.

6. Select a value or specify a range as instructed in the **If** line.
7. Set **Then** to one of the following options:

   - To create a visibility filter, select **Hide Issue**.
   - To create a folder filter, select **Set Folder to**, and then select the folder name or select **Other Folder** to add a folder from another filter set or create a new folder.

8. Click **Save**.

   The new filter is displayed at the end of the list. For folder filters, this gives the new filter the highest priority. Issues that match the new folder filter appear in the targeted folder.

9. (Optional) For folder filters, drag the filter higher in the folder filter list to change the priority.

The issues are sorted with the new filter.

Note

The filter is only created in the selected filter set.

1.5.5.4. Copying a filter from one filter set to another

Filter settings are local to a filter set. However, you can copy the filter to another filter set in the audit project. If you copy a folder filter to another set and that folder is not already active in the set, the folder is automatically added.

To copy a filter:

1. In the **Issues** view, select a filter set from the **Filter Set** list.
2. Select the **Filters** tab in the Issue Auditing view.
3. Right-click a filter, and then select **Copy Filter To**.

   The Select a Filter Set dialog box opens with a list of all the filter sets.
4. Select a filter set, and then click **OK**.

   The filter is added to the filter set in the last position.
5. (Optional) For folder filters, you can adjust the order of the filter list by dragging and dropping the filter to a different location in the list.

1.5.5.5. Setting the default filter Set

To specify the default filter set used to view scan findings:

1. In the **Issues** view, click the **Filter Set** list, and then select **Edit**.

   The Project Configuration dialog box opens to the **Filter Sets** tab.

2. In the **Filter Sets** list, select the filter set you want to use as the default for the issue template.
3. Select the **Default filter set** check box, and then click **OK**.

1.5.6. Managing folders

Folders are logical sets of issues that are defined by the filters in the active filter set. Even though a folder can appear in more than one filter set, the contents might differ depending on the filters in that filter set that target the folder. To accommodate filter sets that provide sorting mechanisms with little overlap, you can have filter sets with different folders. Folders are defined independent of the filter sets in which they might appear. For example, a filter set might place low priority issues into a red folder that is labeled "Hot."

This section contains the following topics:

1.5.6.1. Creating a folder

You can create a new folder so that you can display a group of issues you have filtered to the folder. Folders must have unique names.

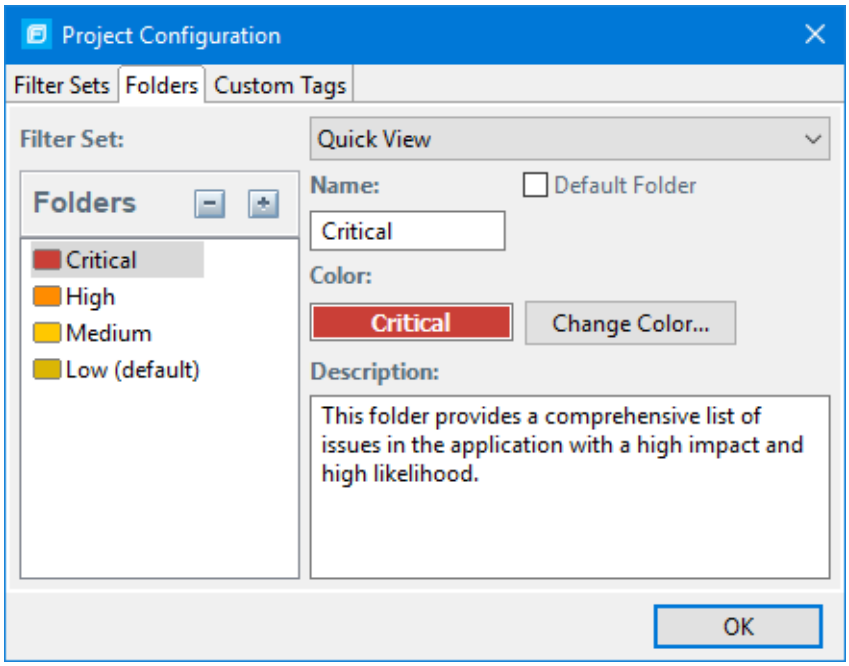
Note

If this functionality is restricted to administrator users, and you are not an administrator, you cannot create folders.

To create a new folder:

1. Select **Tools > Project Configuration**.
2. Select the **Folders** tab.

   The **Folders** pane on the left lists the folders for the filter set selected in the **Folder for Filter Set** list. Fields on the right show the name, color, and description of the selected folder.



3. To associate the folder with an existing filter set, select the filter set from the **Filter Set** list.

   Select **(All Folders)** to create a new folder in the issue template without associating it with a specific filter set. You can associate the folder with an existing filter set later.

Note

Selecting a filter set updates the **Folders** list to display the folders that are associated with the selected filter set.

4. To add a folder:
    1. Next to **Folders**, click the **Add Folder** button  .

       The Add Folder dialog box opens.

       Note

       If you have created folders in other filter sets, the Add New Folder to Filter Set dialog box opens. Click **Create New**.

    2. Type a unique name for the new folder, and then select a folder color.
    3. Click **OK**.

       The folder is added to the bottom of the folder list.
5. In the **Description** box, type a description for the new folder.
6. To change the tab position of the folder on the **Issues** view, drag the folder up or down in the **Folders** list.

   The top position is on the left and the bottom position is on the right.
7. To put all issues that do not match a folder filter into this folder, select the **Default Folder** check box.
8. Click **OK**.

The folder is displayed as a tab with the other folders. If you selected default, all issues that do not match a folder filter are displayed. The new folder is added to the issue template for the audit project.

Note

To display issues in this folder, create a folder filter that targets the new folder. For more information, see Creating a Filter from the Issues View and Creating a Filter from the Issue Auditing View.

1.5.6.2. Adding a folder to a filter set

This section describes how to enable an existing folder in a filter set. Create a new folder that is only included in the selected filter set using the instructions in Creating Folders. To display issues in this folder, create a folder filter that targets the new folder.

To add a folder to a filter set:

1. Select **Tools > Project Configuration**.

   The Project Configuration dialog box opens.

2. Select the **Folders** tab.
3. Click the **Filter Set** list to select the filter set where you want to add a folder.

   The **Folders** list displays the folders in the selected filter set.

4. Next to **Folders**, click the **Add Folder** button  .

   The Add New Folder to Filter Set dialog box opens.

   Note

   If the selected filter set already includes all existing folders, the Create Folder dialog box opens and you can create a new folder for the selected filter set.

5. Select the folder to add to the selected filter set, and then click **Select**.
6. Click **OK**.

The folder is displayed as a tab along with the other folders.

1.5.6.3. Renaming a folder

You can rename a folder. Modifying the name of a folder is a global change reflected in all filter sets.

To rename a folder:

1. Select **Tools > Project Configuration**.
2. Select the **Folders** tab.
3. In the **Filter Set** list, select **(All Folders)**.
4. Select the folder in the **Folders** list.

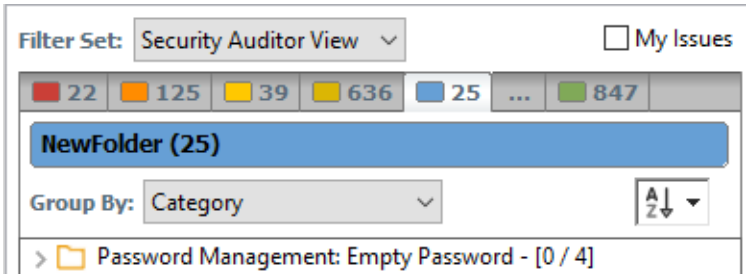
   The folder properties are displayed on the right.

5. Type the new name for the folder.

   The folder name changes in the **Folders** list as you type.

6. Click **OK**.

The new folder name is displayed on the tab.



1.5.6.4. Removing a folder

You can remove a folder from a filter set without removing it from other filter sets.

To remove a folder:

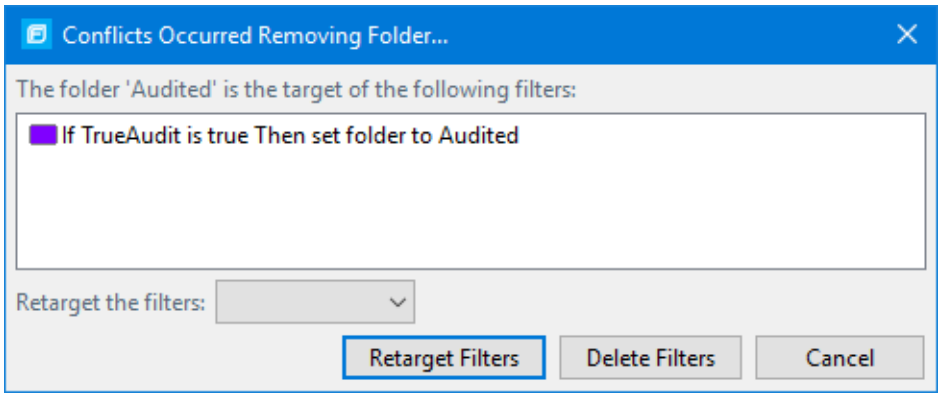
1. Select **Tools > Project Configuration**.
2. Select the **Folders** tab.
3. Select a filter set from the **Filter Set** list.

   The **Folders** list displays the folders in the selected filter set.
4. Select the folder, and then next to **Folders**, click the **Delete Folder** button .

   Note

   The folder is removed only from the selected filter set.

   If the folder is a target of a folder filter, the Conflicts Occurred Removing Folder dialog box opens.

   Do one of the following:

   1. To target the filter to a different folder, select a folder from the **Retarget the filters** list, and then click **Retarget Filters**.
   2. To delete the filter, click **Delete Filters**, and then click **Yes** to confirm the deletion.
5. Click **OK** to close the Project Configuration dialog box.

The folder is no longer displayed as a tab in the **Issues** view.

1.5.7. Configuring custom tags for auditing

To audit code in Application Security, the security team examines project analysis results (FPR) and assigns values to custom tags associated with application version issues. The development team can then use these tag values to determine which issues to address and in what order.

The Analysis tag is provided by default. The **Analysis** tag is a list-type tag and has the following valid values: Not an Issue, Reliability Issue, Bad Practice, Suspicious, and Exploitable. You can modify the **Analysis** tag attributes, change the tag values, or add new values based on your auditing needs.

To refine your auditing process, you can define your own custom tags. You can create the following types of custom tags: list, decimal, string, and date. For example, you might create a list-type custom tag to track the sign-off process for an issue. After a developer audits his own issues, a security expert can review those same issues and mark each as "approved" or "not approved."

You can also define custom tags from Application Security, either directly with issue template uploads through Application Security, or from Fortify Audit Workbench through issue templates in FPR files.

Note

Although you can add new custom tags from Fortify Audit Workbench as you audit a project, if these custom tags are not defined in Application Security for the issue template associated with the application version, then the new tags are lost if you upload the FPR file to Application Security.

You can add the following attributes to your custom tags:

- Extensible—This enables users to create a new value while auditing, even without the permission to manage custom tags.
- Restricted—This restricts who can set the tag value on an issue. Administrators, security leads, and managers have permission to audit restricted tags.
- Hidden (Application Security only)—Use this setting to hide a tag from an application version or issue template.

After you define a custom tag, it is displayed below the **Analysis** tag, which enables you to specify values as they relate to specific issues. Custom tags are also available in other areas of the interface, such as in the **Group By** list to group issues in a folder, in the search field as a search modifier (similarly available as a modifier for filters), and in the project summary graph as an attribute by which to graphically sort issues.

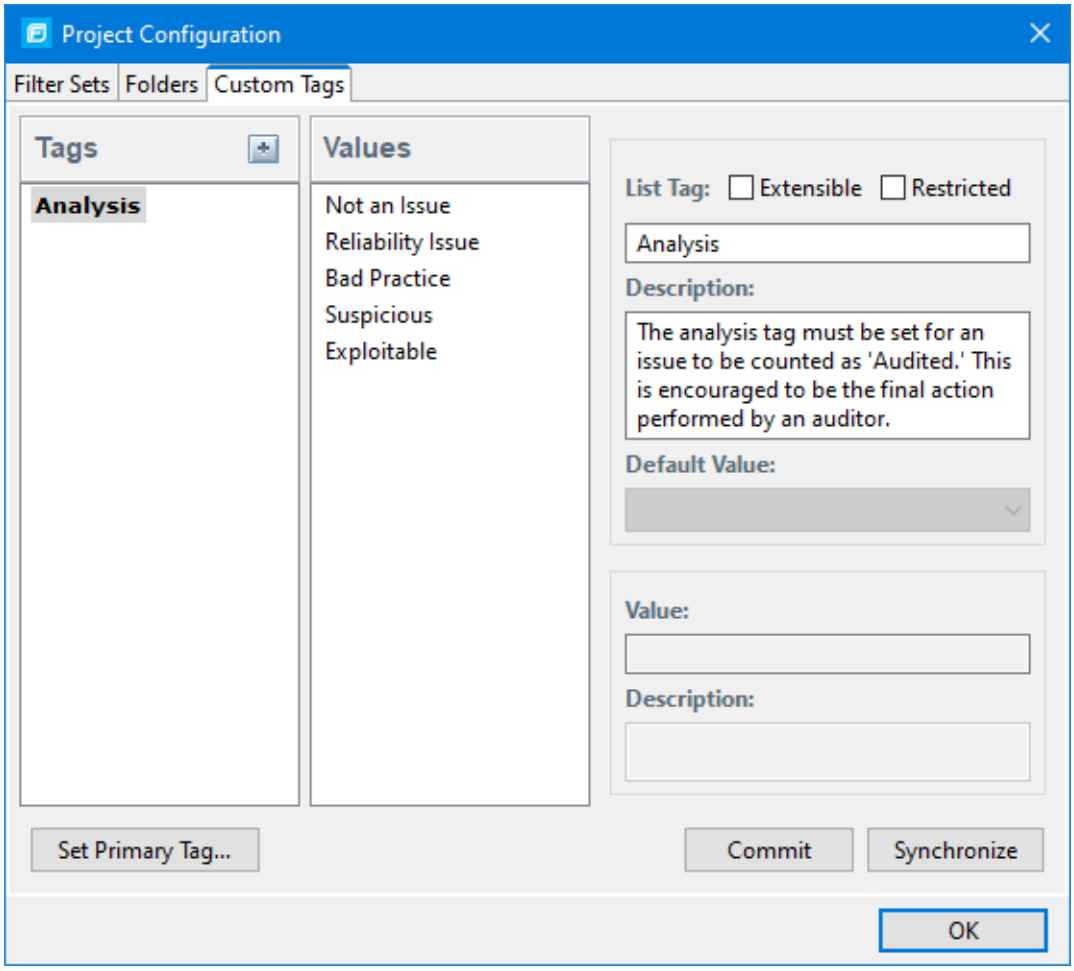
This section contains the following topics:

1.5.7.1. Adding a custom tag

You can create custom tags to use in auditing results. Custom tags are project-wide and are saved as part of an issue template.

To add a custom tag:

1. Select **Tools > Project Configuration**.
2. Select the **Custom Tags** tab.



3. Next to **Tags**, click the **Add Tag** button .

> **Note**
>
> Any previously hidden tags are listed, and you can re-enable them. To create a new tag, click **Create New**.

The Add New Tag dialog box opens.

4. In the **Name** box, type a name for the new tag.

Important

Make sure that the name you specify for a custom tag *is not* a database reserved word.

5. From the **Type** list, select one of the following tag types:
   - **List**—Accepts selection from a list of values that you specify for the tag
   - **Date**—Accepts a calendar date
   - **Decimal**—Accepts a number with a precision of up to 18 (up to 9 decimal places)
   - **Text**—Accepts a string with up to 500 characters (HTML/XML tags and newlines are not allowed)
6. Click **OK**.

   The **Tags** list now includes the new tag.
7. Configure any or all the following optional tag settings:
   - To allow users to add new values for a list-type tag in an audit, leave the **Extensible** check box selected.
   - To allow only administrators, security leads, and managers to set this tag on an issue, select the **Restricted** check box.
   - Type a description of the custom tag in the **Description** box.
   - For a list-type tag, from the **Default Value** list, select the default value for the tag.

     If you do not specify a default value, the default is null.
8. To add a value for a list-type tag, do the following:
   1. From the **Tags** list, select the tag name.
   2. Next to **Values**, click the **Add Value** button  .
   3. In the Enter Value dialog box, type a value, and then click **OK**.
   4. Type a description of the value in the **Description** box.
   5. Repeat steps a through d for each additional value required for the new tag.
9. To make this custom tag the primary tag:

   Note

   You can only set a list-type tag as a primary tag.

   1. Click **Set Primary Tag**.
   2. Select the custom tag from the **Primary Tag** list, and then click **OK**.

   The primary tag name is shown in bold in the **Tags** list. The primary tag determines the audit status for each issue as well as the audit icon in the **Issues** view. By default, the primary tag is **Analysis**.

The **Audit** tab in the Issue Auditing view now displays the new tag and its default value (if you

assigned one).

1.5.7.2. Hiding a custom tag

If you hide a custom tag, it is no longer available on the **Audit** tab in the Issue Auditing view or as a search or filter option.

Note

If you hide a custom tag that was set for any issues, that tag and values are hidden from the issue. If you make the tag available again, the tag and values are restored.

You cannot hide the primary tag.

To hide a custom tag:

1. Select **Tools > Project Configuration**.

   The Project Configuration dialog box opens.

2. Select the **Custom Tags** tab.
3. Select the tag from the **Tags** list.
4. Next to **Tags**, click the **Hide Tag** button .

   This action hides the tag from your available custom tags. You can make this tag available again when you add a custom tag (see Adding a Custom Tag).

5. Click **OK**.

If you hide a tag that has an associated filter, you are prompted to delete the filter.

1.5.7.3. Committing custom tags to Application Security

To commit custom tags to Application Security:

1. With an audit project open, select **Tools > Project Configuration**.
2. Select the **Custom Tags** tab.
3. Click **Commit**.

   Note

   Any list-type custom tags without values are not uploaded to Application Security.

4. If prompted, type your Application Security credentials.

   For information about logging into Application Security, see Logging in to Application Security.

   The Custom Tag Upload dialog box opens.

5. Do one of the following:
   - If the issue template and the application version already exist in Application Security:
     - To upload the custom tags to the global pool and assign them to the application version, click **Yes**.
     - To upload the custom tags to the global pool without assigning them to the application version, click **No**.
     - To prevent uploading the custom tags to Application Security, click **Cancel**.
   - If the issue template does not exist in Application Security:
     - To upload the custom tags to the global pool only in Application Security, click **Yes**.
     - To prevent uploading the custom tags to Application Security, click **No**.

1.5.7.4. Synchronizing custom tags with Application Security

To synchronize custom tags for an audit project that has been uploaded to Application Security.

1. Select **Tools > Project Configuration**.
2. Select the **Custom Tags** tab.
3. Select the custom tag.
4. Click **Synchronize**.
5. If required, type your Application Security credentials.

   For information about logging into Application Security, see Logging in to Application Security.

   The Custom Tag Download dialog box opens.
6. If the application version and the issue template both exist in Application Security, select either **Application Version** or **Issue Template** to specify from where to download the custom tags.
7. To download custom tags from the issue template, click **Yes**.

1.5.8. Issue template sharing

After an issue template is associated with an audit project, all changes made to that template, such as the addition of folders, custom tags, filter sets, or filters, apply to the audit project. The issue template is stored in the FPR when the audit project is saved. For information about how to associate the issue template with an audit project, see "Importing an Issue Template". With issue templates, you can use the same project settings for another project.

This section contains the following topics:

1.5.8.1. Exporting an issue template

Exporting an issue template creates a file that contains the filter sets, folders, and custom tags for the current project. After you export an issue template, you can import it into another audit project file.

To export an issue template:

1. Select **Tools > Project Configuration**.
2. Select the **Filter Sets** tab.
3. Click **Export**.

   The Select a Template File Location dialog box opens.
4. Browse to the location where you want to save the file.
5. Type a file name without an extension.
6. Click **Save**.

Note

If any hidden custom tags exist in the template, you are prompted to indicate whether to include them in the exported issue template. Hidden tags are created anytime you add a custom tag and later delete it. Fortify Audit Workbench saves and hides deleted custom tags so you can easily restore them later. If you do not want hidden tags included in the exported issue template, click **Ignore Tags**.

The current template settings are saved to an XML file.

1.5.8.2. Importing an issue template

Importing an issue template overwrites the audit project configuration settings. The local filter sets and custom tags are replaced with the filter sets and custom tags in the issue template.

To import an issue template:

1. Select **Tools > Project Configuration**.
2. Select the **Filter Sets** tab.
3. Click **Import**.

   The Locate Template File dialog box opens.
4. Select the issue template file to import.
5. Click **Open**.

The filter sets, custom folders, and custom tags are updated.

Note

You can also click **Reset to Default** to return the settings to the default issue template.

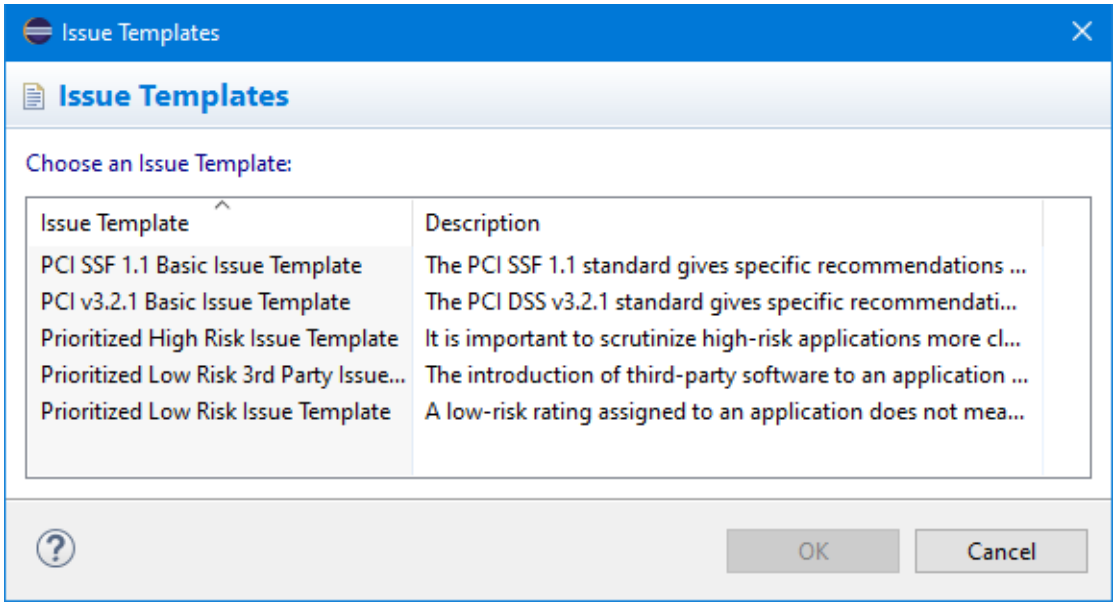
1.5.8.3. Synchronizing filter sets and folders

To download filter sets and folders configured from Application Security:

1. Select **Tools > Project Configuration**.
2. Select the **Filter Sets** tab.
3. Click **Synchronize**.

   A message advises you that downloading filter sets and folders from Application Security overwrites your local filter sets and folders.
4. To proceed with the synchronization, click **Yes**.
5. If required, provide your Application Security credentials, and then click **OK**.

   For information about logging into Application Security, see Logging in to Application Security.

   If the current issue template does not exist in Application Security, do the following:

   1. In the **Issue Template** column, select an issue template name.
   2. Click **OK**.
6. The Fortify Audit Workbench downloads the filter sets and folders from the selected issue template in Application Security, and overwrites your current issue template.

1.5.8.4. Committing filter sets and folders

If you want to upload filter sets and folders to an issue template in Application Security, do the following:

1. Select **Tools > Project Configuration**.
2. Select the **Filter Sets** tab.
3. Select the filter set from the list.
4. Click **Commit**.
5. If required, provide your Application Security credentials.

   For information about logging into Application Security, see Logging in to Application Security.

   The Update Existing Issue Template or Add Issue Template dialog box opens, depending on whether the issue template already exists in Application Security.
6. Do one of the following:

   1. To upload filter sets and folders to the issue template, click **Yes**.
   2. To add the issue template that contains the current set of custom tags to Application Security, click **Yes**.

1.5.9. Advanced configuration

This section contains the following topics:

1.5.9.1. Integrating with a bug tracker application

Fortify Audit Workbench provides a plugin interface to integrate with bug tracker applications. This enables you to file bugs directly from Fortify Audit Workbench . For a list of supported bug tracker applications, see the *OpenText™ Application Security Software System Requirements* document.

To select the plugin to use:

1. Open an audit project.
2. Select **Tools > Select Bug Tracker**.
3. Select a bug tracker from the list, and then click **OK**.

Note

For Jira bug tracker integration, you must restart Eclipse after you change the proxy settings.

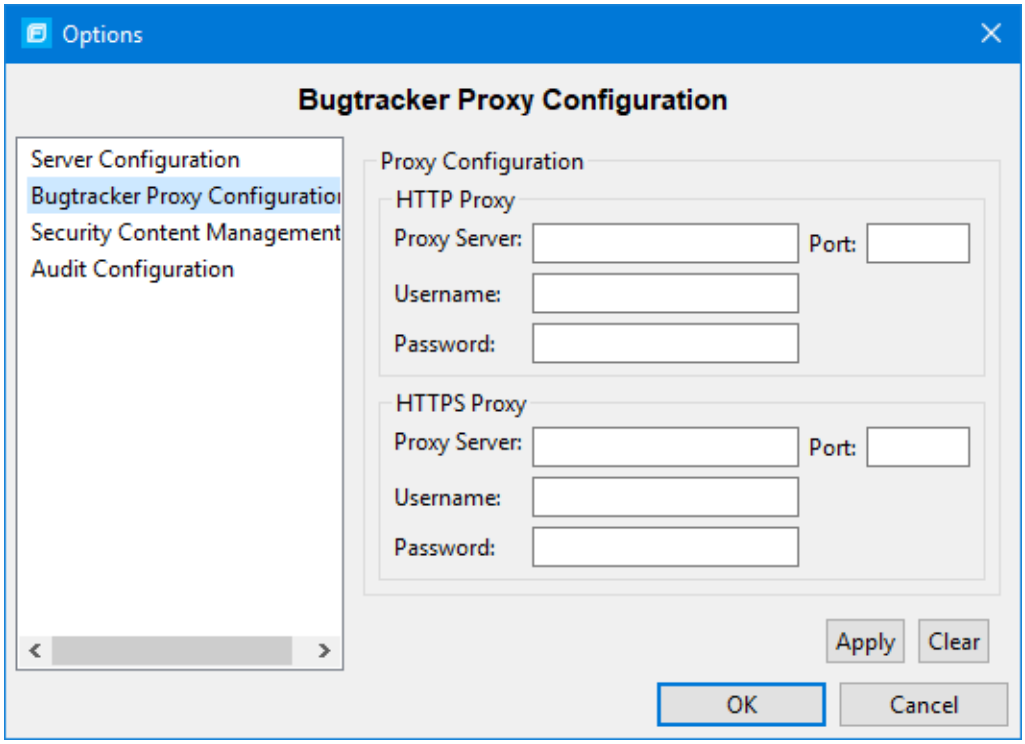
For bug tracker plugin components selected in the OpenText™ Application Security Tools installation, sample source code is available in `<tools_install_dir>/Samples/bugtrackers/BugTrackerPlugin<bug_tracker_app>`, where *<bug_tracker_app>* is the name of the bug tracker application. To write your own plugin, see the instructions in the `README` text file, which is in each bug tracker directory. A JavaDoc includes API information in `<tools_install_dir>/Samples/advanced/JavaDoc/public-api/index.html`.

1.5.9.2. Configuring proxy settings for bug tracker integration

If the bug tracker you use requires a proxy connection, specify the proxy settings. When you submit an issue as a bug, select the **Use proxy** check box. Fortify Audit Workbench provides the proxy settings to the bug tracker plugin.

To configure proxy settings for bug tracker integration:

1. Select **Options > Options**.
2. In the left pane, select **Bugtracker Proxy Configuration**.



3. Under **HTTP Proxy**, specify the proxy server, port number, and optionally credentials for proxy authentication.
4. If the connection uses HTTPS requests, then provide the proxy settings under **HTTPS Proxy**.
5. Click **OK** to save your changes.

1.5.9.3. Public APIs

OpenText publishes public APIs so that you can create custom parsers for pentest tools and services that are not included in the default distribution. The APIs are in (`fortify-public-*.jar`), and you can use them to compile your custom parser.

1.5.9.4. Penetration test schema

OpenText also provides a generic penetration test schema (`pentestimport.xsd`) that you can view in `<tools_install_dir>/Core/config/schemas`. This provides another option for importing additional pentest results. Instead of creating a custom parser for your tool or service, you can translate the results into the Fortify generic format (using XSLT or a similar technology). You can then open or merge these translated results automatically. See Third-Party Penetration Results for more information.

1.6. Auditing analysis results

When OpenText SAST scans application source code, its discoveries are presented as potential vulnerabilities rather than actual vulnerabilities. Every application is unique, and all functionality runs within a context that the development team understands best. No technology can fully determine whether a suspect behavior is considered a vulnerability without direct developer confirmation.

For example, OpenText SAST might discover that a web page designed to display data to the user (for example, a financial transaction record page) appears to allow any authenticated user to request any data with no check of viewing permission. Whether or not this behavior is considered a vulnerability depends entirely on the intended design of the application. If the application is supposed to allow any user to see all data, then the auditor can mark the discovery as a non-issue; otherwise, the auditor can mark the issue as a vulnerability for the team to address.

Note

If your Fortify license restricts auditing, then you can view the analysis results, but you cannot audit issues or make any changes to the audit project.

The topics in this section provide information about how to audit analysis results opened in Fortify Audit Workbench.

This section contains the following topics:

1.6.1. Working with audit projects

After you scan a project, you can audit the analysis results. You can also audit the results of a collaborative audit from Application Security.

This section contains the following topics:

1.6.1.1. Opening an audit project

To open an audit project:

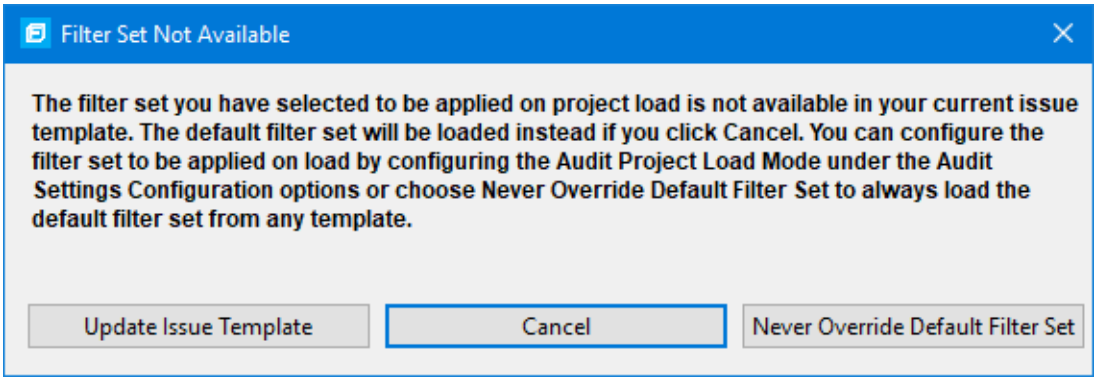
1. Start Fortify Audit Workbench.
2. Select **File > Open Project**.

   The Select Audit Project dialog box opens.

3. Browse to and select the FPR file, and then click **Open**.

1.6.1.1.1. Opening audit projects without the default filter set

If you open an audit project that does not contain the filter set specified as the default filter set for new projects (by default, this is the Quick View filter set), a message is displayed to inform you that the filter set is not available in the audit project's issue template.



The default filter set from the template is loaded at startup, regardless of the setting. This would also happen, for example, with any FPR files downloaded from OpenText™ Core Application Security.

To resolve this, do one of the following:

- To apply the default filter set from the current issue template, click **Cancel**.
- To update the issue template for the project, click **Update Issue Template**.

  After you select **Update Issue Template**, some filter sets that were available before the update, for example Developer View and Critical Exposure, are no longer available.

  A warning is displayed to let you know that you cannot undo the update.

- To ensure that the default filter set for the project is never overridden, click **Never Override Default Filter Set**.

1.6.1.2. Performing a collaborative audit

You can audit a project in Application Security collaboratively with other Application Security users. You can open an application version in Application Security, apply your audit evaluation, and then upload the audit project back to Application Security.

To start a collaborative audit:

1. Start Fortify Audit Workbench.

   If you already have an audit project open, close it.

2. Under **Collaborative Applications**, click **Sign In**.
3. Type your Application Securitylogon credentials.

   For information about logging into Application Security, see Logging in to Application Security.

   Fortify Audit Workbench displays a list of applications that you have permission to access.

4. Select an application version to audit.

   To quickly find an application version, type the name or partial name of an application in the **Search** box. The search is case-insensitive. To clear the search results, clear the **Search** box.

   If necessary, click **Refresh** to update the list of applications in Application Security.

   The audit project file is downloaded from Application Security and opened in Fortify Audit Workbench.

5. Audit the project as described in Evaluating Issues.
6. When you have completed the audit, select **Tools > Upload Audit Project**.

Note

If necessary, you can refresh your Application Security audit permission settings. See Refreshing Permissions From.

See Also

Uploading Audit Results to Application Security

1.6.1.3. Refreshing permissions from Application Security

The Application Security administrator assigns roles to users that determine the actions they can perform in Application Security. When you work on a collaborative audit and the administrator changes your auditing permissions, you might need to refresh the permissions in Fortify Audit Workbench.

To refresh your permissions from Application Security:

1. Select **Options > Options**.
2. In the left pane, select **Server Configuration**.
3. Click **Refresh Permissions for the Current Audit**.
4. Click **OK**.

1.6.1.4. Merging audit data

Audit data includes the custom tags and comments that were added to an issue. You can merge the audit data for your project with audit data from another results file. Comments are merged into a chronological list and custom tag values are updated. If custom tag values conflict (if the same tag is set to different values for a given issue), Fortify Audit Workbench prompts you to resolve the conflict.

Note

Issues are not merged. Merged results include only the issues found in the latest scan. Issues uncovered in the older scan that were not uncovered in the latest scan are marked as Removed and are hidden by default.

Make sure that the projects you merge contain the same analysis information. That is, make sure that the scans were performed on the same source code (no missing libraries or files), the OpenText SAST settings were the same, and the scan was performed using the same security content.

To merge projects:

1. Open a project in Fortify Audit Workbench.
2. Select **Tools > Merge Audit Projects**.
3. Select an audit project (FPR file), and then click **Open**.

   The Progress Information dialog box opens. When complete, the Merge dialog box opens.

   Note

   After you select an FPR, Fortify Audit Workbench might prompt you to choose between the issue template in the current FPR and the issue template in the FPR you are merging in.

4. Click **Yes** to confirm the number of issues added or removed from the file.

   Note

   If the scan is identical, no issues are added or removed.

The project now contains all audit data from both result files.

1.6.1.5. Merging audit data using the command-line utility

You can also use the FPRUtility command-line utility to merge audit data. This utility enables you to merge an audited project, verify the signature of the FPR, or display analysis results information from and FPR. For more information about how to use this utility, see the *OpenText™ Application Security Tools Guide*.

1.6.1.6. Additional metadata

Each issue in Audit Workbench contains additional metadata that is not produced by the internal analyzers. Examples include alternative categories (for example, OWASP, CWE, WASC), and prioritization values that are used in the default filters (for example, impact, accuracy, probability). You can view the metadata attributes through the standard grouping and search mechanisms.

If you open an older FPR that does not contain metadata values, the metadata values for the issues are retrieved from legacy mapping files. These legacy mapping files exist in the `<tools_install_dir>/Core/Config/LegacyMappings` directory, and are indexed by either issue category, or issue category and analyzer. The legacy mapping files are accessed as needed, so each issue in your project must always have metadata values, whether those values come from the FPR, the legacy mapping files, or a combination of the two.

1.6.1.7. Uploading audit results to Application Security

When you work on a collaborative audit and you download the audit project from Application Security, Fortify Audit Workbench retains the application version for the audit project. If you want to upload the audit project to a different application version, you need to disconnect the audit project from Application Security before you upload the results. To disconnect the current audit project from Application Security, select **Options > Options**, click **Server Configuration**, and then click **Disconnect the Current Audit**.

Note

If you created any custom tags or filter sets for your project's issue template, you must first commit them to Application Security before you upload the project so that information is also uploaded. See Committing Custom Tags to Application Security and Committing Filter Sets and Folders for more information.

Note

By default, Fortify Software Security Center does not allow you to upload scans performed in quick scan mode. However, you can configure your Fortify Software Security Center application version so that uploaded audit projects scanned in quick scan mode are processed. For more information, see analysis results processing rules in the *OpenText™ Application Security User Guide*.

To upload results to Application Security:

1. Select **Tools > Upload Audit Project**.
2. If prompted, type your Application Security credentials.

   For information about logging into Application Security, see Logging in to Application Security.

3. If the audit project is not already associated with an application version, select an application version, and then click **OK**.

   Note

   If you see a message that the application version is not committed or does not exist, this indicates that you opened an audit project that was previously associated with an application version that does not exist in Application Security to which Fortify Audit Workbench is currently connected. Disconnect the audit project from Application Security as described previously in this section.

   A message notifies you when the upload is complete.

4. Click **OK**.

Updates you made to issues including comments and tag values (for tags that already exist for the application version in Application Security) are uploaded.

1.6.2. Evaluating issues

To evaluate and assign audit values to an issue or group of issues:

1. Select the issue or group of issues in the **Issues** view (see About Viewing Scan Results).

   Note

   If multiple issues are selected, then this information is displayed on the **Audit** tab as **Issue: Multiple Issues Selected**.

2. Read the abstract on the **Audit** tab, which provides high-level information about the issue, such as the analyzer that found the issue.

   For example, **Command Injection (Input Validation and Representation, Data Flow)** indicates that this issue that the Dataflow Analyzer detected, is a Command Injection issue in the Input Validation and Representation kingdom.

3. Click the **Details** tab to see more details about the issue.
4. On the **Audit** tab, select an analysis value for the issue to represent your evaluation.
5. Specify values for any custom tags defined by your organization.

   To specify a date in a date-type custom tag, click the **Select Date** button to select a date from a calendar.

   For text-type custom tags, you can click the **Edit Text** button ... to see and edit long text strings. This tag accepts up to 500 characters (HTML/XML tags and newlines are not allowed).

6. If the audit results have been submitted to Fortify Audit Assistant in Application Security, then you can specify whether to include or exclude the issue from Fortify Audit Assistant training from the **AA_Training** list.

   Note

   If you select a different value for the analysis tag than the **AA_Prediction** value set by Fortify Audit Assistant, and you select **Include** from the **AA_Training** list, then the next time the data is submitted to Fortify Audit Assistant, it updates the information used to predict whether an issue represents a true vulnerability. For more information about Fortify Audit Assistant, see the *OpenText™ Application Security User Guide*.

7. (Optional) In the **Comments** box, type comments relevant to the issue and your evaluation.

1.6.2.1. Performing quick audits

As you audit issues, you can use a keyboard combination to assign an analysis value to multiple selected issues.

To assign an analysis value to multiple issues simultaneously:

1. In the **Issues** view, select the issues that you want to assign the same analysis value.
2. Press **Ctrl** + **Shift** + **A** (**Cmd** + **Shift** + **A** on macOS).

   Fortify Audit Workbench displays a window in the lower-right corner to indicate you are in **Quick Audit Issue** mode.

   Note

   Do not hold this keyboard combination in the next step.

3. Press one of the following number keys:
   - To assign Not an Issue, press **1**
   - To assign Reliability Issue, press **2**
   - To assign Bad Practice, press **3**
   - To assign Suspicious, press **4**
   - To assign Exploitable, press **5**
   - To assign a custom analysis value configured for your organization, press the number that corresponds to its position in the **Analysis** list on the **Audit** tab.

Fortify Audit Workbench provides keyboard shortcuts for only the first ten values in the **Analysis** list. (To assign the tenth value in the list, you press **Ctrl** + **Shift** + **A**, and then press **0**). If no value is listed for the key you press, no value is assigned.

1.6.2.1.1. Performing quick audits for custom tags

Instead of using the Analysis tag for quick audits, you can use a custom tag your organization has created.

To use a custom tag for quick audits:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**, and then select the **Configuration** tab on the right.
3. Under **Quick Audit Preference**, from the **Attribute to use for quick action audit** list, select a custom tag.

   Note

   Only list-type tags are available to use for quick audits.

   If no custom tags have been created, the list only includes the **Analysis** tag.
4. Click **OK**.

The keyboard shortcut functions just as it does for the Analysis tag values. Fortify Audit Workbench provides keyboard shortcuts for only the first ten values in the list of custom tag values. (To assign the tenth value in the list, you press **Ctrl**+ **Shift** + **A**, and then press **0**). If there is no value in the list for the key you press, no value is assigned.

See Also

Configuring Custom Tags for Auditing

1.6.2.2. Adding screenshots to issues

You can attach a screenshot or other image to an issue. Attached images are stored in the FPR file and are accessible from Application Security. The following image formats are supported:

- GIF
- JPG
- PNG

To add an image to an issue:

1. Select the issue.
2. In the Issue Auditing pane, select the **Screenshots** tab.
3. Click **Add**.
4. In the New Screenshot dialog box, click **Browse** to find and select the file.
5. (Optional) In the **Description** box, type a description.
6. Click **Add**.

1.6.2.2.1. Viewing images

After you add a screenshot to an issue, the image is displayed on the right side of the **Screenshots** tab.

To view a full-size version of an image added to an issue:

1. In the Issue Auditing pane, select the **Screenshots** tab.
2. From the list of screenshots, click the image you want to view.
3. Click **Preview**.

1.6.2.3. Creating issues for undetected vulnerabilities

Add undetected issues that you want to identify as issues to the issues list. You can audit manually configured issues on the **Audit** tab, just as you do other issues.

To create an issue:

1. Select the object in the line of code in the source code tab.
2. Right-click the line that contains the issue, and then select **Create New Issue**.

   The Create New Issue dialog box opens.

3. Select the issue category, and then click **OK**.

The issues list displays the file name and source code line number for the new issue next to a blue icon. The rule information in the **Audit** tab includes `Custom Issue`. You can edit the issue to include audit information, just as you can other issues.

1.6.2.4. Suppressing issues

You can suppress issues that are either fixed or that you do not plan to fix. Suppression marks the issue and all future discoveries of this issue as suppressed. As such, it is a semi-permanent marking of a vulnerability.

To suppress an issue, do one of the following:

- In the **Issues** view, select the issue, and then, on the **Audit** tab in the Issue Auditing view, click the **Suppress** button .
- In the **Issues** view, right-click the issue, and then click **Suppress Issue**.

Note

You can select and suppress multiple issues at the same time.

To display issues that have been suppressed, select **Options > Show Suppressed Issues**.

To unsuppress an issue, first display the suppressed issues, and then do one of the following:

- In the **Issues** view, select the suppressed issue, and then, on the **Audit** tab in the Issue Auditing view, click the **Unsuppress** button  .
- Right-click the issue in the **Issues** view, and then select **Unsuppress Issue**.

Note

You can select and unsuppress multiple issues at the same time.

1.6.3. Submitting an issue as a bug

You can submit issues to your bug tracker application if integration between the applications has been configured.

To submit an issue as a bug:

1. Select the issue in the **Issues** view, and then, on the **Audit** tab, click the **File Bug** button  .
2. If this is the first time you have filed a bug, the Select Bug Tracker Integration dialog box opens. Select a bug tracker application, and then click **OK**.

   For information about configuring the plugin with bug tracker applications, see Bug Tracking System Integration.
3. Specify all required values and review the issue description. Depending on the integration and your bug tracker application, the values include items such as the bug tracker application web address, product name, severity level, summary, and version.
4. If the connection to the bug tracker requires a proxy, select the **Use proxy** check box.

   With this option selected, Fortify Audit Workbench uses the proxy settings specified for bug trackers. For more information, see Configuring Proxy Settings for Bug Tracker Integration.
5. Click **Submit**.

You must already be logged in before you can file a bug through the user interface for bug tracker applications that require a logon. The issue is submitted as a bug in the bug tracker application.

If you use Application Security, you can submit an issue as a bug using a bug tracker application configured through Application Security.

To submit an issue as a bug through Application Security:

1. Select the issue in the **Issues** view, and then, on the **Audit** tab, click the **File Bug** button  .

   The first time you submit a bug, the Select Bug Tracker Integration dialog box opens. Select **Fortify Software Security Center**, and then click **OK**.
2. Specify the values if changes are needed and review the issue description. Depending on the integration and your bug tracker application, the values include items such as the bug tracker application web address, product name, severity level, summary, and version.
3. Click **Submit**.

If your bug tracker application requires you to log in, you must do so before you can file a bug through that interface.

1.6.4. Correlation justification

A correlation occurs when an issue uncovered by one analyzer (OpenText DAST Agent, OpenText SAST, or OpenText DAST) is related directly or indirectly to an issue uncovered by another analyzer.

Correlated events help you identify issues that have a higher probability of being exploited. A vulnerability that is linked to other vulnerabilities might represent an issue that has multiple points of entry. For example, if OpenText DAST scan results are correlated with OpenText SAST scan results, this increases the likelihood that the associated OpenText SAST issues are exploitable.

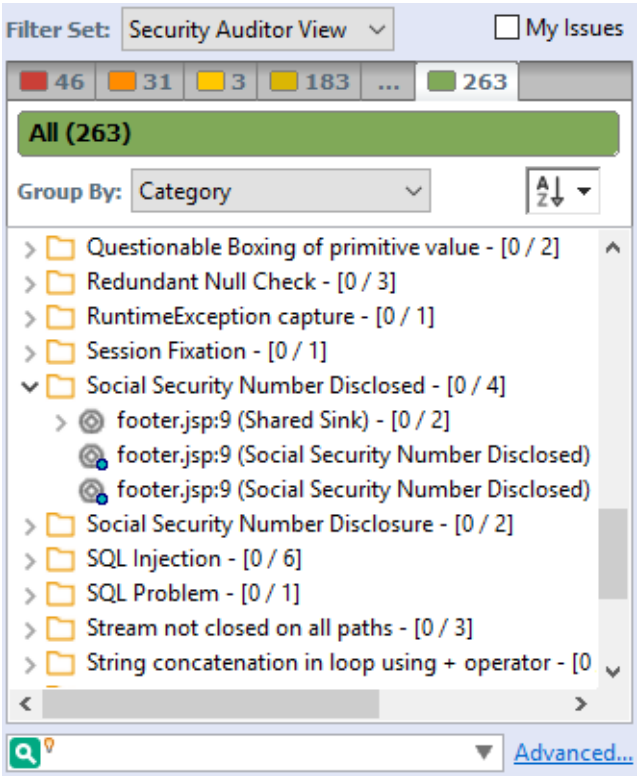
Fortify Audit Workbench provides additional information to help you resolve these correlated issues and mitigate the risks they present. In Fortify Audit Workbench, this additional information is presented as Correlation Justification.

1.6.4.1. Using correlation justification

To use correlation justification:

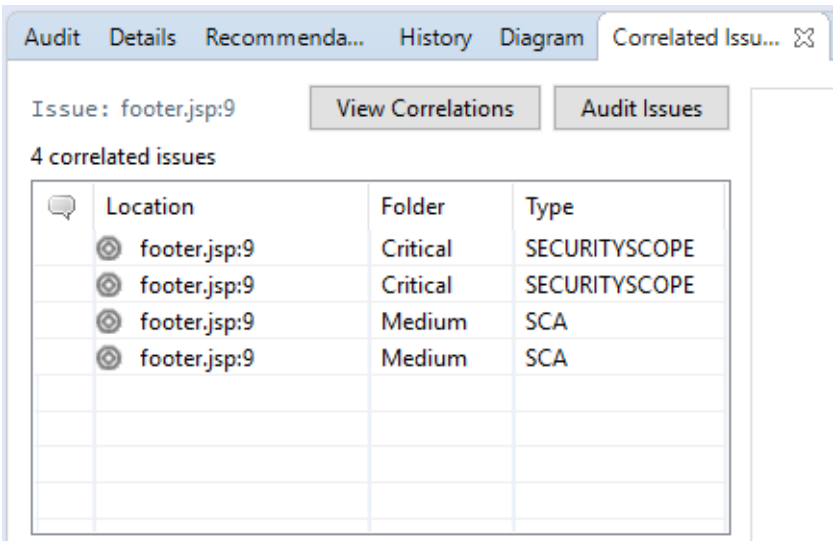
1. In the **Issues** view, select a correlated issue.

   A correlated issue is identified in the issues list by a blue sphere on the issue symbol, as shown below.



2. In the Issue Auditing view, select the **Correlated Issues** tab.

   The **Correlated Issues** tab lists the other issues that are correlated with the issue you first selected.

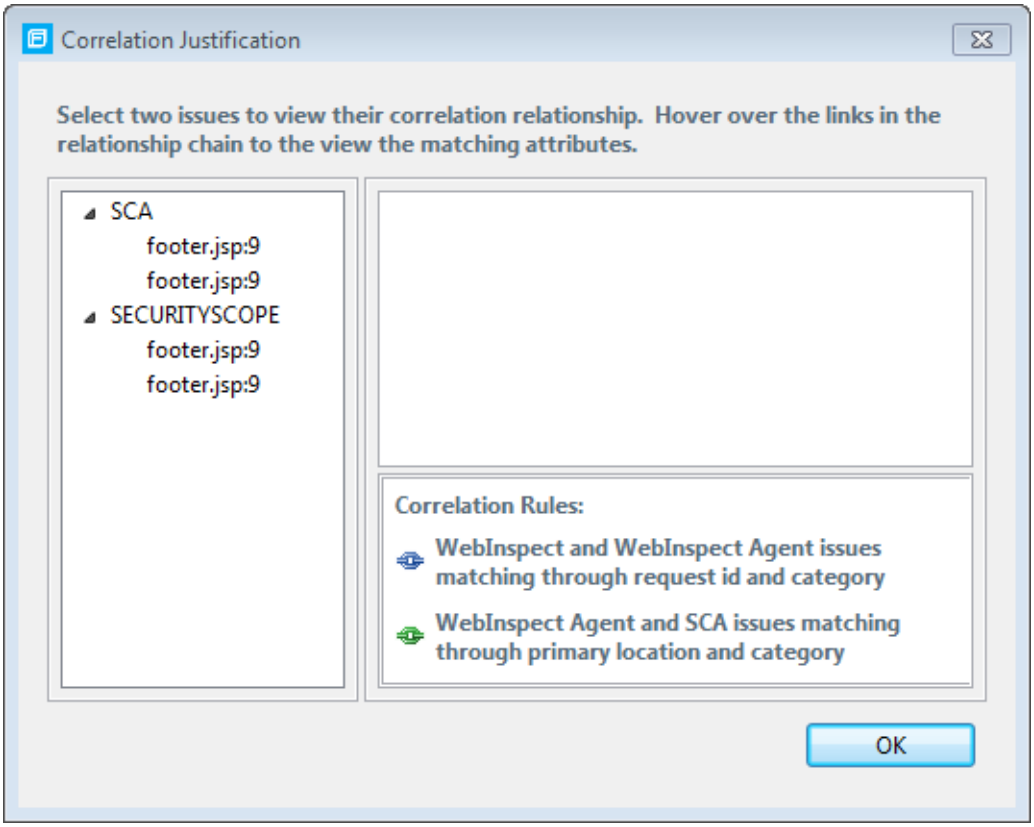


   Because you first selected a correlated issue, the **View Correlations** button is available.

3. Click **View Correlations**.

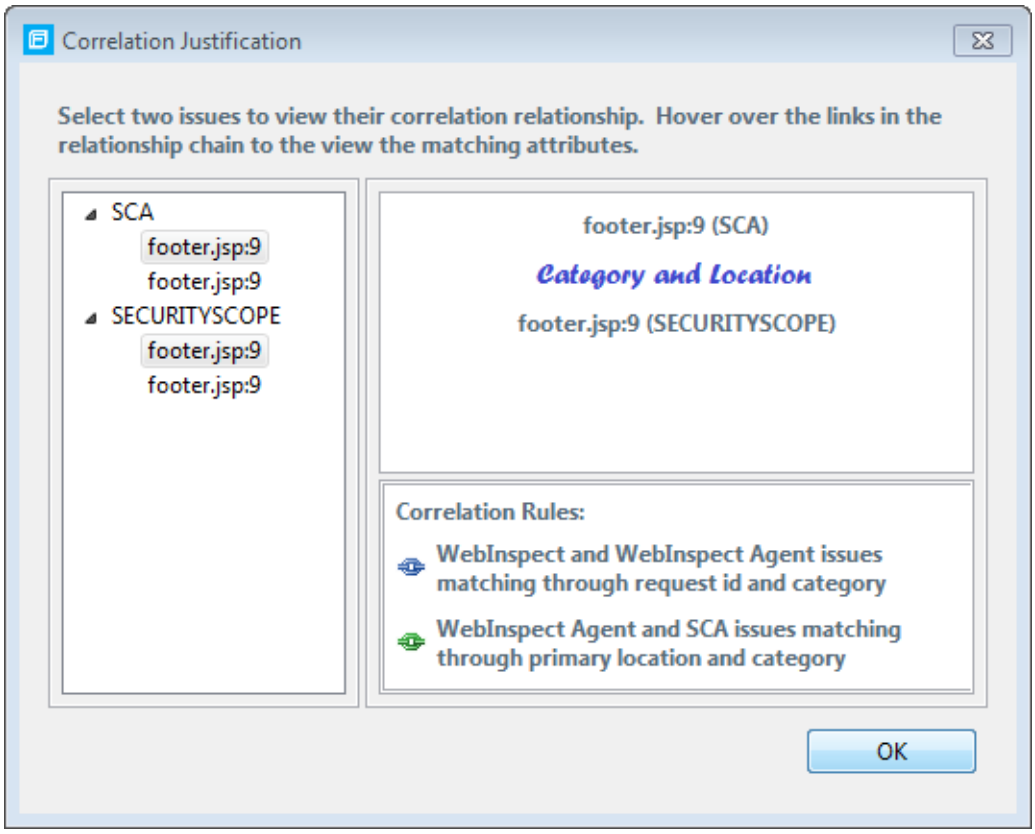
   The Correlation Justification dialog box opens and displays the following three panes:

   - The correlated issues tree on the left displays all correlated issues within a correlated group, sorted based on analyzers.
   - The relationship pane at the top right displays the correlation chain between issues. The chain describes any indirect or direct relationship between the two selected issues.
   - The pane at the bottom right describes each correlation rule in the correlation chain displayed in the relationship pane.



4. To select two issues, press **Ctrl**, and then click each issue.

   The relationship pane displays the two issues and their relationships.



5. To inspect the attributes that correlate the issues, move your cursor to each link in the relationship pane.
6. Click **OK**.

Use correlation justification to gain insight into code vulnerabilities and understand why certain issues are correlated. This can help to reduce the time it takes to remediate the issues.

1.6.5. Penetration test results

Fortify Audit Workbench supports import of XML for dynamic issues from OpenText DAST or from your own custom parser that produces results in an XML file.

To create your own parser, write a class that implements the `com.fortify.pub.issueparsing.AnalysisFileParser` interface from the Fortify public API. It can use any of the classes and utilities from `<tools_install_dir>/Core/lib/fortify-public-<version>.jar`. The Fortify public API documentation is in `<tools_install_dir>/Samples/advanced/JavaDoc/public-api/index.html`. The section for parsing scans and creating issues is in the `com.fortify.pub.issueparsing` package.

1.6.5.1. Viewing penetration test results

Pentest issues have an `analyzer` attribute equal to `pentest`, and an `analysis type` attribute that reflects the tool or service (for instance, OpenText DAST issues have the `WEBINSPECT` analysis type. You can view these attributes through the standard grouping and search mechanisms.

After you select a pentest issue, Fortify Audit Workbench displays the penetration test details on the **Pentest Details** tab. The following table lists the penetration test details.

Pentest Detail	**Description**
Request	Click the question mark button to view the full request.
Path	Web address without the context and parameters.
Referer	Referer header in the request.
Method	Either GET or POST.
Parameters	Parameters included in the HTTP query.
Cookies	Cookies included in the HTTP query.
Attack Type	Type of pentest attack conducted (web address, parameter, header, or cookie).
Attack Payload	Part of the request that causes the vulnerability.
Trigger	Part of the response that shows that a vulnerability occurred. To view the full response, click the question mark button next to the trigger.

1.7. Generating analysis reports

Fortify Audit Workbench provides two types of analysis reports:

- Issue reports based on the Business Intelligence and Reporting Technology (BIRT) system
- Legacy reports based on user-configurable report templates

This section contains the following topics:

1.7.1. Issue reports

You can generate issue reports based on the BIRT system from Fortify Audit Workbench or from the command line. For information on how to generate issue reports from the command line using the BIRTReportGenerator utility, see the *OpenText™ Static Application Security Testing User Guide*.

The following table describes the issue reports available.

Report Template	Description
CWE Top 25	This report lists the most widespread and critical weaknesses that can lead to serious software vulnerabilities (based on the National Vulnerability Database).
CWE/SANS Top 25	This report details issues related to the CWE/SANS Top 25 Most Dangerous Programming Errors and provides information about where and how to fix the issues. It describes the technical risk posed by unremediated issues discovered during analysis and provides an estimate of the development effort needed to test, verify, and fix them.
Developer Workbook	This report provides the information a developer needs to understand and fix the issues discovered during an application audit.
DISA CCI 2	This report provides a standard identifier for policy-based requirements that connect high-level policy expressions and low-level technical implementations.
DISA STIG	This report addresses DISA compliance based on STIG violations and provides information about where and how to fix the issues. It describes the technical risk posed by unremediated issues and provides an estimate of the development effort required to test, verify, and fix them.
FISMA Compliance: FIPS 200	This report addresses FISMA compliance related to FIPS-200 through controls specified in NIST SP 800-53. It details policy violations and provides information about where and how to fix the issues. It describes the technical risks posed by unremediated violations and provides an estimate of the development effort required to test, verify, and fix them.
GDPR	This report groups all detected issues that are relevant to privacy under the EU General Data Protection Regulation (GDPR) legislation. Use this as a framework to help identify and protect personal data as it relates to application security.

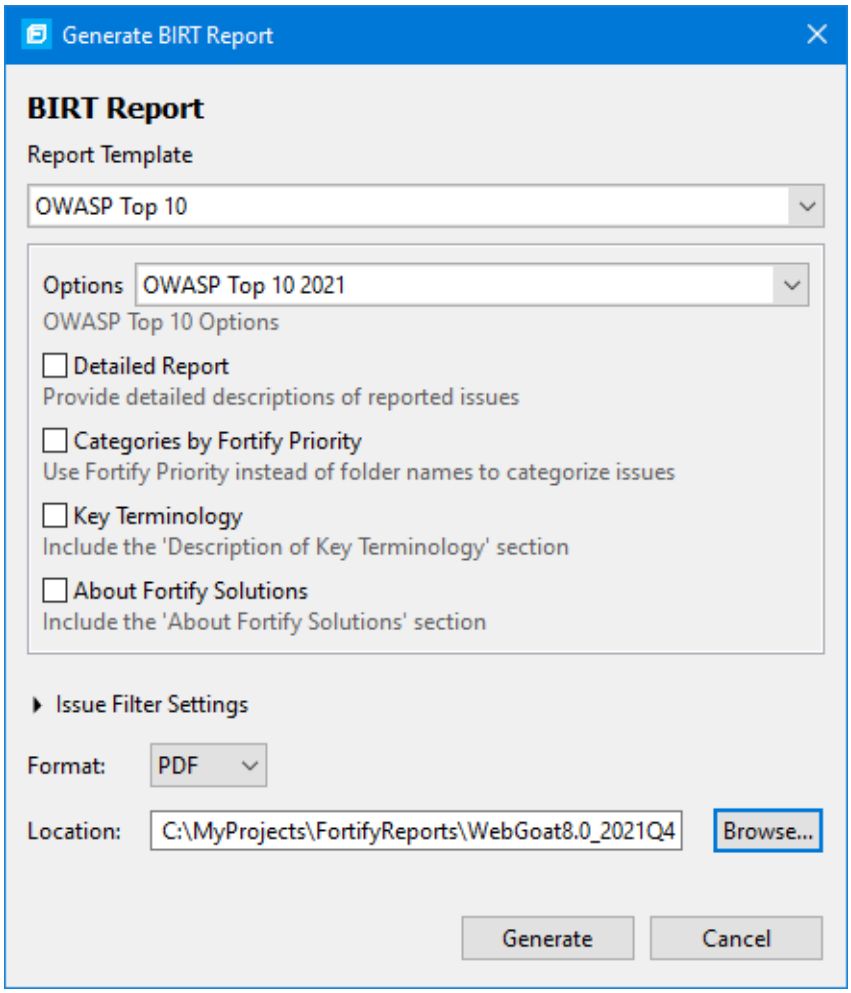
MISRA	This report addresses compliance with either the Motor Industry Software Reliability Association (MISRA) C or C++ guidelines. The results focus on the security relevant guidelines and can help to create a compliance matrix for MISRA. This report describes the technical risk posed by the unremediated issues discovered during analysis and an provides an estimate of the development effort needed to test, verify, and fix them.
OWASP API Top 10	This report focuses on weaknesses affecting Web APIs and is intended to be used in combination with other standards and best practices to thoroughly capture all relevant risks. For example, you can use it in combination with the OWASP Top 10 to identify issues related to input validation such as injections.
OWASP ASVS	This report groups detected issues based on the OWASP Application Security Verification Standard requirements for secure development.
OWASP MASVS 2.0	This report groups detected issues based on the OWASP Mobile Application Security Verification Standard requirements for secure mobile application development.
OWASP Mobile Top 10	This report details the top ten OWASP mobile-related issues and provides information about where and how to fix them. It describes the technical risk posed by the unremediated issues discovered during analysis and gives an estimate of the development effort required to test, verify, and fix them.
OWASP Top 10	This report details the top ten OWASP-related issues and provides information about where and how to fix them. It describes the technical risks posed by unremediated issues discovered during analysis and gives an estimate of the development effort required to test, verify, and fix the issues.
PCI DSS Compliance: Application Security Requirements	This report summarizes the application security portions of PCI DSS. It includes tests for 21 application security-related requirements across sections 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is either "In Place" or "Not In Place."
PCI SSF Compliance: Secure Software Requirements	This report summarizes the application security portions of PCI SSF. It includes tests for 23 application security-related control objectives across Control Objective sections 2, 3, 4, 5, 6, 7, 8, and A.2 of PCI SSF and reports whether each control objective is "In Place" or "Not In Place."

1.7.1.1. Generating issue reports

To generate an issue report:

1. Select **Tools > Reports > Generate BIRT Report**.

   The Generate BIRT Report dialog box opens.



2. From the **Report Template** list, select the type of report you want.
3. From the **Options** list, select the template version (if multiple versions are available).
4. Select the information to include in the report.

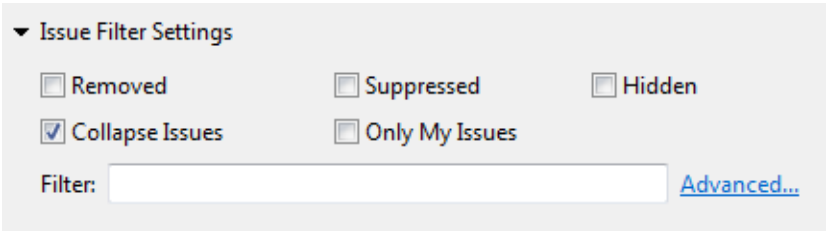
   Note

   Not all options are available for all report templates.

   1. To include detailed descriptions of reported issues, select the **Detailed Report** check box.
   2. To categorize issues by Fortify Priority instead of folder names, select the **Categories By Fortify Priority** check box.
   3. To include descriptions of key terminology in the report, select the **Key Terminology** check box.

4. To include the About Fortify Solutions section in the report, select the **About Fortify Solutions** check box.

5. To filter information from the report, click **Issue Filter Settings**.



You can filter the issues as follows:

- Click **Removed** to include removed issues in the report.
- Click **Suppressed** to include suppressed issues in the report.
- Click **Hidden** to include hidden issues in the report.
- Click **Collapse Issues** to collapse issues of the same sink and type into a single issue.
- Click **Only My Issues** to include only issues assigned to your user name.
- Click **Advanced** to build a search query to further filter the issues to include in the report. For more information about the search modifiers, see Search Modifiers.

6. From the **Format** list, select the format for the report.

   You can save the report in the following formats: Portable Document Format (PDF), HTML, and Microsoft Word (DOC).

7. To specify an alternative location to save the report, click **Browse**, and then select a directory.
8. Click **Generate**.
9. If a report with the same file name already exists, you are prompted to either:

   - Click **Overwrite** to overwrite the existing report.
   - Click **Append Version Number** to have the report saved to a file with a sequential number appended to the file name (for example: `buildABC CWESANSTop25(1).pdf`).

1.7.2. Legacy reports and templates

The legacy reports include user-configurable report templates. Report templates provide several optional sections and subsections that gather and present specific types of data. For detailed descriptions of the report templates, see Legacy Report Components. You can generate legacy reports from Fortify Audit Workbench or from the command line using the ReportGenerator utility. For information on how to generate legacy reports from the command line, see the *OpenText™ Static Application Security Testing User Guide*.

The following sections provide information about the default reports and report templates, instructions on how to modify existing reports, and how to create your own reports.

This section contains the following topics:

1.7.2.1. Generating Legacy Reports

After you select a report template and specify report settings, you generate the report to view the results. You can save the report results in PDF or XML format.

To run a report:

1. Select **Tools > Reports > Generate Legacy Report**.
2. Select a report template from the **Report** list.
3. (Optional) Make changes to the report section settings.
4. Click **Save Report**.

   The Save Report dialog box opens.
5. Make any necessary changes to the report details, including its location and format.
6. Click **Save**.

   Fortify Audit Workbench generates the report in the format you selected.

1.7.2.2. Legacy report templates

This section describes how to select and edit a legacy report template. You can modify legacy report templates from the Generate Legacy Report dialog box, or you can edit report templates directly in XML (see Report Template XML Files). If you or another user have edited or created other default report templates, you might not see the default report templates described in this section.

The legacy report templates include:

- **Fortify Developer Workbook**—Provides a comprehensive list of all categories of issues found and multiple examples of each issue. This report also gives a high-level summary of the number of issues in each category.
- **Fortify Scan Summary**—Provides high-level information based on the category of issues that OpenText SAST found as well as a project summary and a detailed project summary.
- **Fortify Security Report**—A mid-level report that provides comprehensive information on the analysis performed and the high-level details of the audit that was performed. It also provides a high-level description and examples of categories that are of the highest priority.
- **OWASP Top Ten *<year>***—Provides high-level summaries of uncovered vulnerabilities organized based on the top ten issues that the Open Web Security Project (OWASP) has identified.

The following sections describe how to view report templates and customize them to address your reporting needs.

1.7.2.3. Selecting legacy report sections

You can choose sections to include in the report.

To select the sections that you want to include in a report:

1. Click a section title to view the contents of the section.

   The section details are displayed to the right of the dialog box.

2. To include a section in the report, select the section title check box in the list on the left side.
3. To remove a section from the report, clear the check box next to the section title.

For instructions on how to edit each section, see Editing Report Subsections.

1.7.2.4. Opening legacy report templates

To open a report template:

1. Select **Tools > Reports > Generate Legacy Report**.

   The Generate Legacy Report dialog box opens.

2. From the **Report** list, select a report template to open.

The Generate Legacy Report dialog box displays the report template settings.

1.7.2.5. Editing legacy report subsections

When you select a section title, you can edit the contents that are displayed in the report. You can edit text, add or change text variables, or customize the issues shown in a chart or results list.

Editing text subsections

To edit a text subsection:

1. Select the check box next to the subsection title to include this text in the report.

   A description of the text is displayed below the subsection title.

2. Click **Edit Text**.

   The text box displays the text and variables to include in the report.

3. Edit the text and text variables.

As you edit text subsections, you can insert variables that are defined when you run the report. The following table describes these variables.

Variable	Description
`$AUDIT_GUIDE_SUMMARY$`	List of filters created with answers to Audit Guide Wizard questions
`$CLASSPATH_LISTING$`	JAR files used in the scan, one relative path per line
`$COMMANDLINE_ARGS$`	Complete list of command-line options (same format as project summary)
`$FILE_LISTING$`	List of scanned files, each in the format: `<relative_file_path> # Lines # kb <timestamp>`
`$FILTERSET_DETAILS$`	List of filters the current filter set uses
`$FILTERSET_NAME$`	Name of the current filter set
`$FORTIFY_SCA_VERSION$`	OpenText SAST version
`$LIBDIR_LISTING$`	Libdirs specified for the scan, one relative path per line
`$TLOC$`	Total lines of code

`$NUMBER_OF_FILES$`	Total number of files scanned
`$PROJECT_BUILD_LABEL$`	Build label of project
`$PROJECT_NAME$`	Build ID
`$PROPERTIES$`	Complete list of properties set for the analysis phase (same format as project summary)
`$RESULTS_CERTIFICATION$`	Complete certification detail with a list of validity on a per file basis (same format as project summary)
`$RESULTS_CERTIFICATION_`SUMMARY$	Short description of certification (same format as project summary)
`$RULEPACKS$`	Complete list of Rulepacks used for the analysis (same format as project summary)
`$SCAN_COMPUTER_ID$`	Hostname of machine on which the scan was performed
`$SCAN_DATE$`	Date of analysis with the default format style for the locale
`$SCAN_SUMMARY$`	Summary of codebase scanned in format `# files, # lines of code`
`$SCAN_TIME$`	Time of analysis phase
`$SCAN_USER$`	Username for the user who performed the scan
`$SOURCE_BASE_PATH$`	Source base path of codebase
`$TOTAL_FINDINGS$`	Number of issues, not including suppressed and removed issues
`$VERSION_LABEL$`	Label of the scanned project (available only if the Fortify Static Code Analyzer `-build-label` option was used in the scan)
`$WARNINGS$`	Complete list of warnings that occurred
`$WARNING_SUMMARY$`	Number of warnings found in scan

Editing results list subsections

To edit a result list subsection:

1. Select the check box next to the subsection title to include this text in the report.

   A description of the results list is displayed below the subsection title.

2. Click the issues list heading to expand the options.

3. Select the attributes used to group the results list.

   If you group by category, the recommendations, abstract, and explanation for the category are also included in the report. For the list of attributes to group by, see Grouping Issues.

4. (Optional) To refine the issues shown in this subsection with a search query, click **Advanced**.

   For information about the search syntax, see Searching for Issues.

5. Select or clear the **Limit number of Issues in each group** check box.
6. If you selected the check box, type the number of issues to display per group.

Editing chart subsections

To edit a chart subsection:

1. Select the check box next to the subsection title to include this text in the report.

   A chart description is displayed below the subsection title.

2. Select the attributes used to group the chart data.

   For the list of attributes to group by, see Grouping Issues.

3. (Optional) To refine the issues shown in this subsection with a search query, click **Advanced**.

   For information about the search syntax, see Searching for Issues.

4. Select the chart format (table, pie, or bar).

1.7.2.6. Saving legacy report templates

You can save the current report settings as a new template that you can select later to run more reports.

To save settings as a report template:

1. Select **Tools > Generate Legacy Report**.

   The Generate Report dialog box opens.

2. Select the report template from the **Report** list.
3. Make changes to the report section and subsection settings.
4. Click **Save as New Template**.

When you select the report template name from the **Report** list, the report settings are displayed in the Generate Report dialog box.

Saving changes to legacy report templates

You can save changes to a report template so that your new settings are displayed as the defaults for that template.

To save changes a report template:

1. Select **Tools > Generate Legacy Report**.

   The Generate Report dialog box opens.

2. Select the report template to save as the default report template from the **Report** list.
3. (Optional) Make changes to the report section and subsection settings.
4. Click **Save Settings as Default**.

1.7.2.7. Report template XML files

Report templates are saved as XML files. You can edit the XML files to make changes or to create new report template files. When you edit the XML files, you can choose the sections and the contents of each section to include in the report template.

The default location for folder that contains report template XML files is:

`<tools_install_dir>/Core/config/reports/`
To customize the logos used in the reports, you can replace `header.png` and `footer.png` in this directory.

Adding legacy report sections

You can add report sections by editing the XML files. In the XLM structure, the `ReportSection` element defines a new section. It includes a `Title` element for the section name, and it must include at least one `Subsection` element to define the contents of the section in the report. The following XML is the `Results Outline` section of the Fortify Security Report:

```
  <ReportSection enabled="true" optionalSubsections="true">
 <Title>Results Outline</Title>
 <SubSection enabled="true">
  <Title>Overall number of results</Title>
  <Description>Results count</Description>
  <Text>The scan found $TOTAL_FINDINGS$ issues.</Text>
 </SubSection>
 <SubSection enabled="true">
  <Title>Vulnerability Examples by Category</Title>
  <Description>Results summary for critical and high priority issues.
   Vulnerability examples are provided by category.
  </Description>
  <IssueListing limit="1" listing="true">
   <Refinement>[fortify priority order]:critical OR
    [fortify priority order]:high</Refinement>
   <Chart chartType="list">
    <Axis>Category</Axis>
   </Chart>
  </IssueListing>
 </SubSection>
</ReportSection>
```

In the previous example, the `Results Outline` section contains two subsections. The first subsection is a text subsection named `Overall number of results`. The second subsection is a results list named `Vulnerability Examples by Category`. A section can contain multiple

subsections.

Adding report subsections

In the report sections, you can add subsections or edit subsection content. Subsections can generate text, results lists, or charts.

Adding text subsections

In a text subsection, you can include the `Title` element, the `Description` element, and the `Text` element. In the `Text` element, you can provide the default content, although you can edit the content before you generate a report. For a description of the text variables available to use in text subsections, see Editing Report Subsections. The following XML is the `Overall number of results` subsection in the `Results Outline` section:

```
  <SubSection enabled="true">
 <Title>Overall number of results</Title>
 <Description>Results count</Description>
 <Text>The scan found $TOTAL_FINDINGS$ issues.</Text>
</SubSection>
```

In this example, the text subsection is titled `Overall number of results`. The text to describe the purpose of the text is `Results count`. The text in the text field that the user can edit before running a report uses one variable named `$TOTAL_FINDINGS$`.

Adding results list subsections

In a results list subsection, you can include the `Title` element, the `Description` element, and the `IssueListing` element. In the `IssueListing` element, you can define the default content for the limit and set `listing` to `true`. You can include the `Refinement` element either with or without a default statement, although you can edit the content before you generate a report. To generate a results list, the `Chart` element attribute `chartType` is set to `list`. You can also define the `Axis` element. The following XML is the `Vulnerability Examples by Category` subsection in the `Results Outline` section:

```
    <SubSection enabled="true">
   <Title>Vulnerability Examples by Category</Title>
   <Description>Results summary of the highest severity issues.
    Vulnerability examples are provided by category.</Description>
   <IssueListing limit="1" listing="true">
    <Refinement>[fortify priority order]:critical OR
     [fortify priority order]:high</Refinement>
    <Chart chartType="list">
     <Axis>Category</Axis>
    </Chart>
   </IssueListing>
</SubSection>
```

In this example, the results list subsection is titled `Vulnerability Examples by Category`. The text to describe the purpose of the subsection is `Results summary of the highest severity issues. Vulnerability examples are provided by category`. This subsection lists (`listing=true`) one issue (`limit="1"`) per Category (the `Axis` element value) where there are issues that match the statement `[fortify priority order]:critical OR [fortify priority order]:high` (the value of the `Refinement` element).

Adding charts subsections

In a chart subsection, you can include the `Title` element, the `Description` element, and the `IssueListing` element. In the `IssueListing` element, you can define the default content for the limit and set `listing` to `false`. You can include the `Refinement` element either with or without a default statement, although you can edit the content before generating a report. To generate a pie chart, the `Chart` element's attribute `chartType` is set to `pie`. The options are `table`, `pie`, and `bar`. You can change this setting before you generate the report. You can also define the `Axis` element.

The following code shows an example of a chart subsection:

```
    <SubSection enabled="true">
   <Title>New Issues</Title>
   <Description>A list of issues discovered since the previous
    analysis.</Description>
   <Text>The following issues have been discovered since the
    last scan.</Text>
   <IssueListing limit="-1" listing="false">
    <Refinement />
    <Chart chartType="pie">
     <Axis>New Issue</Axis>
    </Chart>
   </IssueListing>
  </SubSection>
```

In this subsection, a chart (`limit="-1" listing="false"`) has the title `New Issues` and a text section that contains the text `The following issues have been discovered since the last scan`. This chart includes all issues (the `Refinement` element is empty) and groups the issues on the value of `New Issues` (the value of the `Axis` element). This chart is displayed as a pie chart (`chartType="pie"`).

1.8. Using the Functions view

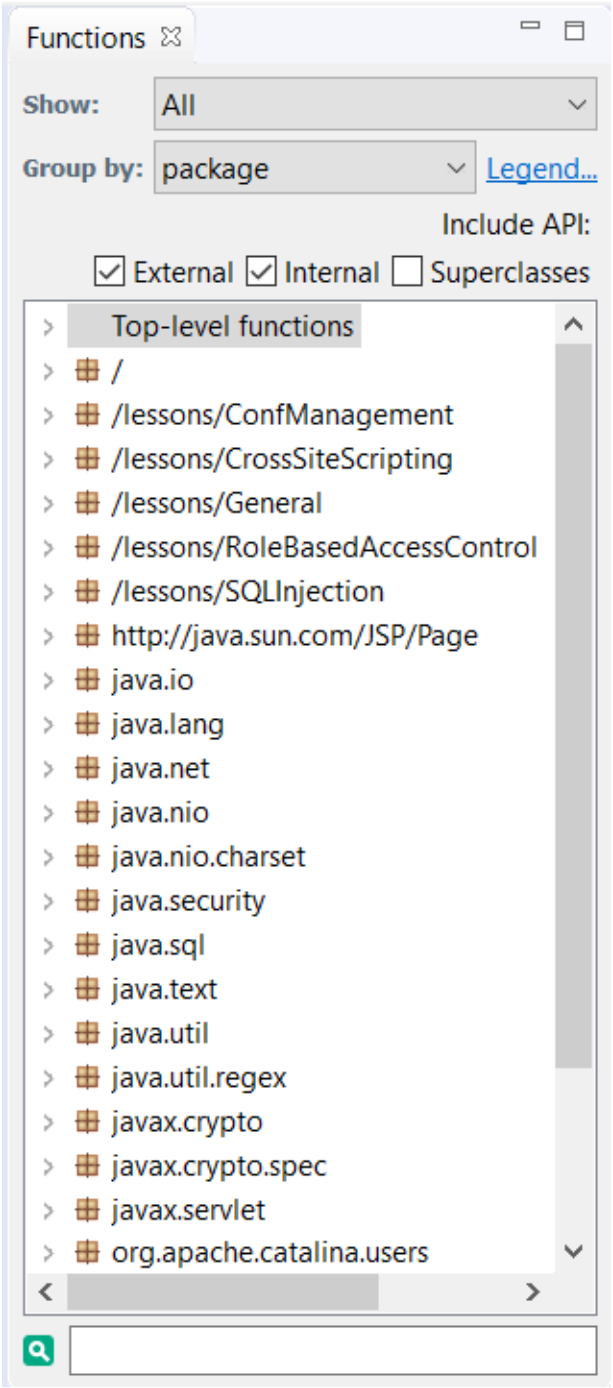
OpenText SAST identifies all functions declared or called in your source code. You can use the **Functions** view in Fortify Audit Workbench to determine where a function is located in the source code, whether a security rule covered the function, and which rule IDs matched the function. You can also list the functions that OpenText SAST identified as tainted source and view only the functions *not* covered by rules applied in the most recent scan.

This section contains the following topics:

1.8.1. Opening the Functions view

To open the **Functions** view:

1. Select **Options > Show View > Functions**.



   Fortify Audit Workbench displays the **Functions** view in the top-right.

2. To view coverage information about top-level (global) functions, expand the **Top-level functions** node.
3. To view descriptions of the symbols displayed to the left of each function, click **Legend**.
   - **Function Not Covered by Rules**

This function does not have any rules associated with it.

Note

It is not necessary to have a rule for every function in an application because not all functions have a security impact.

- **Function Covered by Rules**

  This function is covered by one or more rules; however, the rules are never triggered.

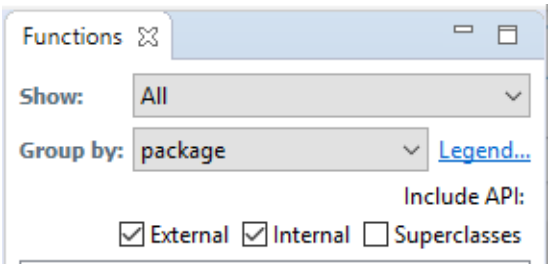
- **Function Covered by Rules and has Matching Rules**

  This function is covered by one or more rules and at least one of them triggered. This does not necessarily mean an issue has been found. For example, a tainted data source rule matches the source function but the tainted data that entered the function does not reach a sink.

1.8.2. Sorting and Viewing functions

To change the order of, or to hide or show functions:

1. Open the **Functions** view.



2. From the **Show** list, select one of the following:
    - To display all functions, select **All**.
    - To display functions not covered by rules, select **Not Covered by Rules**.
    - To display functions that the Rulepack used in the most recent scan has identified as a source of tainted data, select **Taint Sources**.
3. From the **Group By** list, select one of the following sorting methods:
    - To sort functions based on package, select **package**.
    - To sort listed functions by class, select **class**.
    - To sort listed functions alphabetically, select **function**.
4. Under **Include API**:
    - To show functions in external classes, select the **External** check box.
    - To show functions in internal classes, select the **Internal** check box.
    - To show functions in superclasses, select the **Superclasses** check box.

Fortify Audit Workbench updates the **Functions** view.

1.8.3. Locating functions in source code

From the **Functions** view, you can list the file name and line number where the function occurs in the source code.

To show where a function is located in the code:

1. In the **Functions** view, right-click a function, and then select **Find Usages**.

   The **Search** view (at center bottom) lists the file locations and line numbers in which the function is used.

2. To jump to a line of code where the function is used, click the corresponding row in the **Search** view.

1.8.4. Synchronizing the Functions view with the Analysis Trace view

You can synchronize the **Functions** view with the **Analysis Trace** view so that, after you select an issue or a trace node from the **Analysis Trace** view, the **Functions** view automatically displays the class that contains the selected item of evidence. This makes it easy for you to inspect other methods in that class, other classes in that package, and so on.

To synchronize the **Functions** view with the **Analysis Trace** view:

1. In the Functions view, from the **Group By** list, select **class**.
2. In the top-right corner of the **Analysis Trace** view, click the **Synchronize with Functions View** button  .

   The **Functions** view displays the class that contains the item you selected in the **Analysis Trace** view.

The **Synchronize with Function View** button toggles synchronization. To turn off synchronization, click the **Synchronize with Functions View** button again.

1.8.5. Locating classes in source code

To see where classes are used in the source code:

1. In the **Functions** view, right-click a class button  , and then select **Find Usages**.

   The **Search** view (at center bottom) lists the file locations and line numbers in which the class is used.

2. To jump to a line of code where the class is used, click the corresponding row in the **Search** view.

For functions defined in the source code, you can open the declaration in the **Source** view by right-clicking a function and then selecting **Open Declaration**. The source code is displayed with the line highlighted. Alternatively, you can double-click functions to display the declaration.

1.8.6. Determining which rules matched a function

You can display the rule ID for all the rules that matched a function. When rules match a function, a green circle is displayed next to it.

OpenText SAST can match a rule to functions without finding an issue related to the rule. For example, a tainted data source rule matches the source function but the tainted data entering at that function does not reach a sink.

Note

To use the rule ID to locate related issues, see Searching for Issues, or create visibility or folder filters.

To display the rule IDs:

1. Open a project in Fortify Audit Workbench.
2. Open the **Functions** view.
3. Right-click a function, and then select **Show Matched Rules**.

The **Search** view (at center bottom) lists the rule IDs with the vulnerability category name (if applicable) and the Rulepack file name.

1.8.7. Writing rules for functions

You can use the Custom Rule Wizard from the **Functions** view to create a rule for a function.

To write a rule for a function:

1. Open a project in Fortify Audit Workbench.
2. Open the **Functions** view.
3. To create a rule:
    1. Right-click a function, and then select **Generate Rule for Function**.

       The Custom Rule Wizard opens.

    2. Select the rule that best matches the behavior or vulnerability category.
    3. Provide the information the wizard directs, and save the new rule to a custom Rulepack.
4. To rescan the translated files with the custom Rulepack:
    1. Select **Options > Options**.
    2. In the left pane, select **Security Content Management**.
    3. Click **Import Custom Security Content**.
    4. Browse to and select the custom Rulepack, and then click **Open**.
    5. Click **OK** to close the Options dialog box.
    6. Click **Scan**.

    After the scan is completed, the project is updated.
5. Click **OK**.
6. To verify that the rule matched the function:
    1. Right-click the function, and then select **Show Matched Rules**.
    2. Verify that at least one rule ID matches the ID of the rule you created.

The function is now covered by a custom Rulepack and is displayed with a green circle next to it.

1.8.8. Creating custom cleanse rules

You can create custom cleanse rules for specific functions from Fortify Audit Workbench.

To create a cleanse rule for a function:

1. Right-click the function, and then select **Generate Rule for Function**.

   The Custom Rule Wizard opens.

2. In the templates list, expand the **DataflowCleanseRule** folder, and then select **Generic Validation Rule**.
3. Click **Next**.
4. On the **Rule Language** step, select the source code language, and then click **Next**.
5. On the **Validation Function Information** step, type the regular expressions for the package, class, and function.
6. Verify that the information is correct, and then click **Next**.
7. Select the argument to cleanse, and then click **Next**.
8. Select the Rulepack to which you want to add the rule, and then click **Finish**.

1.9. Troubleshooting

The following topics provide information on how to troubleshoot problems you might encounter working with Fortify Audit Workbenchthe Fortify Plugin for Eclipse and how to report an issue to Customer Support.

This section contains the following topics:

1.9.1. Creating archive logs for Customer Support

You can have Fortify Audit Workbench create an archive file that you can later send to Customer Support to help resolve any support issues that might arise. The file includes your Fortify Audit Workbench logs and system properties.

To create an archive of your Fortify Audit Workbench logs and system properties:

1. In the Fortify Audit Workbench menu bar, select **Help > Contact Fortify Product Support**.
2. In the Create Fortify support archive? dialog box, click **Yes**.
3. Navigate to the folder where you want to save the archive file.
4. Accept the default file name displayed in the **File name** box, or change it.
5. Click **Save**.
6. To contact Customer Support and supply the archive file, follow the instructions in the Save Successful dialog box.
7. Click **OK**.

1.9.2. Using the Debug option

If you encounter errors, you can enable the debug option to help troubleshoot.

To enable debugging:

1. Navigate to the `<tools_install_dir>/Core/config` directory and open the `fortify.properties` file.
2. You can either enable debug mode for all OpenText™ Application Security Tools or for specific applications. Remove the comment tag (#) from in front of the property and set the value to `true`.

Property	Description
`com.fortify.Debug`	If set to `true`, all the OpenText™ Application Security Tools run in debug mode.
`com.fortify.awb.Debug`	If set to `true`, Fortify Audit Workbench runs in debug mode.
`com.fortify.eclipse.Debug`	If set to `true`, the Fortify Plugin for Eclipse runs in debug mode.
`com.fortify.VS.Debug`	If set to `true`, the Fortify Extension for Visual Studio runs in debug mode.

1.9.3. Locating log files

For help diagnosing a problem, provide log files to Customer Support. To package the log files in a zip, see Creating Archive Logs for Customer Support.

On Windows systems, the default Fortify log files are the following directories:

- `C:\Users\<username>\AppData\Local\Fortify\sca<version>\log`

  The log files in this directory are only available if you analyze the code with OpenText SAST.

- `C:\Users\<username>\AppData\Local\Fortify\AWB-<version>\log`
- `C:\Users\<username>\AppData\Local\Fortify\AWB-<version>\workspace\.metadata\.log`

On Linux and macOS systems, the default Fortify log files are the following directories:

- `<userhome>/.fortify/sca<version>/log`

  The log files in this directory are only available if you analyze the code with OpenText SAST.
- `<userhome>/.fortify/AWB-<version>/log`
- `<userhome>/.fortify/AWB-<version>/workspace/.metadata/.log`

1.9.4. Addressing the org.eclipse.swt.SWTError error

On Linux systems, Fortify Audit Workbench can fail to start, resulting in the following error:

```
org.eclipse.swt.SWTError: No more handles [gtk_init_check()
failed]
```

If you see this error, check to make sure that X11 is configured correctly and that your `DISPLAY` variable is set.

1.9.5. Out of Memory errors

The following two scenarios can trigger out-of-memory errors in Fortify Audit Workbench.

Scenario	More Information
Opening or auditing a large and complex FPR file	Allocating More Memory for
Running a scan on large and complex project	Allocating More Memory for

As a guideline, assuming no other memory-intensive processes are running, do not allocate more than two thirds of the available system memory.

1.9.5.1. Allocating additional memory for Fortify Audit Workbench

To increase the memory allocated for Fortify Audit Workbench, set the environment variable `AWB_VM_OPTS`. (For example, set `AWB_VM_OPTS=-Xmx4G` to allocate 4 GB to Fortify Audit Workbench.) If you choose to set `AWB_VM_OPTS`, do not allocate more memory than is physically available. Over-allocation degrades performance.

In Fortify Audit Workbench, issue information is persisted to disk. This persisted information is reloaded on demand and thereby decreases the required memory footprint of Fortify Audit Workbench. To prevent out-of-memory errors, you can set a value in the `fortify.properties` file to take advantage of the information persisted to disk functionality. Set the property as follows:

```
com.fortify.model.PersistDataToDisk=true
```

1.9.5.2. Allocating additional memory for OpenText SAST

To increase the memory allocated for OpenText SAST, do one of the following:

- In the Advanced Static Analysis wizard, increase the amount of memory OpenText SAST uses for scanning. This passes the memory allocation option to OpenText SAST. This method does not require restarting Fortify Audit Workbench. See "Scanning Large, Complex Projects".
- Before your start Fortify Audit Workbench, set the environment variable `SCA_VM_OPTS`. For example, to allocate 32 GB to OpenText SAST, set the variable to `-Xmx32G`.

Note

If you choose to set `SCA_VM_OPTS`, do not allocate more memory than is physically available. Overallocation degrades performance.

1.9.6. Specifying memory for external processes

You can specify how much memory external processes such as iidmigrator use by setting the `com.fortify.model.ExecMemorySetting` property in the `<tools_install_dir>/Core/config/fortify.properties` file. Fortify Audit Workbench uses the iidmigrator tool when merging analysis results. The default memory setting for iidmigrator is 1800 MB. The value for this property setting specifies the maximum heap size.

If you set the value of this property and you have OpenText SAST installed, then in addition to updating the file for OpenText™ Application Security Tools, make sure to apply the same property change in the `<sca_install_dir>/Core/config/fortify.properties` file.

1.9.7. Saving a project that exceeds the maximum removed issues limit

When you save a project that has more than the maximum number of removed issues, Fortify Audit Workbench displays following warning message:

> Your project contains more than *<removed_issues_limit>* removed issues.
> Would you like to persist them all, or limit the number to *<removed_issues_limit>*?
> If you limit the number, audited removed issues will take precedence over unaudited ones.

Click **Limit** to limit the number of issues to the maximum or click **Save All** to save all the removed issues. The `com.fortify.RemovedIssuePersistanceLimit` property controls the maximum number of removed issues *<removed_issues_limit>*. See the *OpenText™ Application Security Tools Guide* for more information.

To configure how Fortify Audit Workbench handles this issue for future occurrences:

1. Select **Options > Options**.
2. In the left pane, select **Audit Configuration**.
3. Select the **Configuration** tab.
4. Under **Save Audit Project Options**, specify one of the following configuration settings:
    - **Limit removed issues to the maximum number**
    - **Save all removed issues every time**
    - **Prompt me next time**
5. Click **OK**.

1.9.8. Resetting the default views

If you have closed or moved views, such as the **Issues** view or the **Audit** tab, you can reset the user interface to restore the views to the default state.

To reset the user interface to the default state:

1. Select **Options > Options**.
2. In the left pane, click **Audit Configuration**.
3. On the **Appearance** tab, click **Reset Interface**.

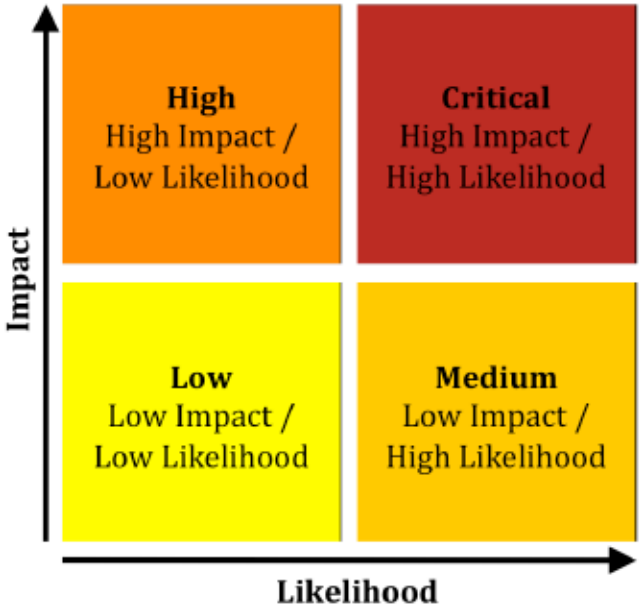
1.10. Static analysis results prioritization

The following topics describe how OpenText SAST automatically prioritizes the scan results displayed in Fortify Audit Workbench.

This section contains the following topics:

1.10.1. About results prioritization

OpenText SAST divides static analysis findings into four risk quadrants: critical, high, medium, and low. Membership in each quadrant depends on whether the finding has a high or low impact and high or low likelihood of occurring.



When OpenText SAST produces a results file, automated processing and human review can convert issues into findings. Findings, which represent specific problems with the codebase, sometimes map one-to-one with issues. However, in other cases, multiple related issues might be combined into a single finding. For example, every form that submits a request without including a unique token might produce an issue related to Cross-Site Request Forgery (CSRF), but these issues are more useful when they are combined into a single finding that indicates the application is vulnerable to CSRF attacks.

On occasion, the static analysis process goes wrong. Depending on the rules and the analysis algorithms used, a static analysis can produce false positives (reported vulnerabilities where no vulnerabilities exist) or false negatives (unreported vulnerabilities) or both.

1.10.2. Quantifying risk

Because it is not possible to determine if or when an organization will suffer consequences related to a vulnerability, OpenText SAST takes a probabilistic approach to prioritizing vulnerabilities. Risk is defined quantitatively, as follows:

`risk = impact x likelihood`

The risk that a vulnerability poses is equal to the impact of the vulnerability multiplied by the likelihood that the impact will occur. Impact is defined as the negative outcome resulting from a vulnerability and likelihood as the probability that the impact will happen.

Impact can come in many forms. For example, an organization might lose money or reputation because of a successful attack, or it might lose business opportunity because the presence of a vulnerability causes a system to fail a regulatory compliance check.

Two factors contribute to the likelihood that a vulnerability will cause harm:

- The probability that the vulnerability will be discovered (by an attacker or an auditor)
- The conditional probability that, once found, the vulnerability will be exploited

These probabilities change as the computer security field advances. New vulnerability assessment techniques make it easier to find vulnerabilities, and new attack techniques increase the set of vulnerabilities that attackers can exploit. Progressively better vulnerability prevention, mitigation, and recovery strategies help counterbalance these advances.

For example, consider Race Condition: Singleton Member Field vulnerabilities, which occur when code assigns a value associated with a user session to a member variable of a singleton object in a web application. Under the singleton model, the same class instance services all requests, therefore values from one user session can spill over into another user's session. The following code demonstrates a singleton member field race condition:

```
public class GuestBook extends HttpServlet {
  String name, password;
  protected void doPost (HttpServletRequest request,
   HttpServletResponse response) {
    name = request.getParameter("username");
    password = request.getParameter("password");
    if (DBUtils.lookupUser(username, password)) {
        accessSensitiveResources();
    }
  }
}
```

Although this vulnerability is simple to exploit after it is found, it can be difficult to find race conditions because successful attacks often depend on very precise timing. Therefore, this class of vulnerability has a low likelihood of occurring, which primarily reflects the difficulty

involved in finding the vulnerability.

For an example of a vulnerability whose likelihood is primarily governed by how difficult it is to exploit, consider HTTP Header Manipulation, which occurs when unvalidated user input is included in an HTTP response header and can enable cross-site scripting, HTTP response splitting, and cache poisoning, among other attacks. The following code demonstrates a header manipulation vulnerability:

```
String author = request.getParameter(AUTHOR_PARAM);
Cookie cookie = new Cookie("author", author);
cookie.setMaxAge(cookieExpiration);
response.addCookie(cookie);
```

In this case, identifying a vulnerable application is often quite simple because the vulnerability is evident in web traffic returned from the server. Crafting a meaningful exploit, however, typically involves a deep understanding of the application's business logic, ready access to a pool of legitimate users, and in some cases, a working knowledge of the network topography between the server and the users. Therefore, this class of vulnerability has a low likelihood because it is difficult to exploit.

1.10.3. Estimating impact and likelihood with input from rules and analysis

OpenText SAST estimates the impact of a discovered vulnerability based on its type. The impact value is associated with the static analysis rule that defines the vulnerability. In this way, results can indicate that a category such as cross-site scripting has a higher impact than a category such as null pointer dereference.

To compute the likelihood portion of the risk equation, OpenText SAST draws on values from the rules used for analysis, the analysis process itself, and from a human auditor (if an individual has reviewed the results.) The likelihood of a finding is computed by combining the accuracy of the rule and the confidence in the analysis with the probability that the vulnerability will be discovered and acted upon, as follows:

```
likelihood = accuracy x confidence x probability
```

For the purpose of weighing static analysis results, an accuracy measure is associated with each rule applied by the analysis engine. This number represents the possibility that the rule will correctly identify a vulnerability.

For example, the rule that OpenText SAST uses to identify the member field race condition has a high accuracy because it precisely identifies assignments to a member field of a singleton object. Conversely, the rule used to identify cross-site request forgery has a low accuracy because it identifies potentially vulnerable form submissions and relies on a human auditor to determine whether the form submissions are susceptible to cross-site request forgery.

During static analysis, OpenText SAST might have to make assumptions about the way the code behaves at runtime. The more assumptions OpenText SAST makes, the more likely it is that a result is incorrect.

The term *confidence* is used to estimate the possibility that OpenText SAST correctly applies the rule. For example, OpenText SAST reports reflected cross-site scripting vulnerabilities in a JSP where data from a request parameter is echoed directly to the page with high confidence. Conversely, OpenText SAST reports a persistent cross-site scripting issue where data read from a database into a class selected at runtime using dependency injection is rendered in the presentation tier with low confidence.

To represent the probability that the vulnerability is discovered and acted upon (with action potentially coming the form of an exploit), OpenText SAST associates a probability measure with each category of vulnerability identified by the rules. For example, cross-site scripting vulnerabilities carry a higher probability than member field race conditions because they are more likely to be discovered and exploited.

From a programmer's perspective, some bugs are harder to fix than others. Modifying a single line of code in a self-contained method is easier than modifying the result of a sequence of

calls that span the program. The term *remediation effort* describes the relative amount of effort required to fix and verify a finding.

OpenText SAST provides a remediation effort with each finding it reports. For example, member field race conditions have a small remediation effort, while cross-site request forgery, which often involves major changes to a website, has a high remediation effort.

To avoid implying too much precision where little exists, OpenText SAST limits values of impact, accuracy, confidence, and probability to a decimal range of from 0.1 to 5.0 and scales the calculated likelihood value to the same range. It then defines high values for impact and likelihood as those at 2.5 and above [2.5,5.0] and low values as those below 2.5 (0,2.5).

OpenText SAST does not provide units for remediation effort because the absolute cost of remediating different vulnerabilities differs from one organization to another. Instead, remediation effort estimates the relative cost to remediate one kind of finding versus another, thereby enabling a comparison of the effort required to remediate different vulnerabilities or vulnerabilities across more than one project.

The following table provides sample impact, accuracy, confidence, and probability values for the four vulnerabilities mentioned in this section along with the resulting risk calculations and corresponding remediation effort for each vulnerability category.

Category	**Impact**	**Accuracy**	**Confidence**	**Probability**	**Risk**
Race Condition: Singleton Member Field	4	5	5	3	Impact = 4 (High) Likelihood = (5 · 5 · 3)/25 = 3 (High) Risk = Critical Estimated remediation effort = 5
Cross-Site Request Forgery	2	1	5	2	Impact = 2 (Low) Likelihood = (1 · 5 · 2)/25 = <1 (Low) Risk = Low Estimated remediation effort = 12

Cross-Site Scripting: Reflected	5	5	5	5	Impact = 5 (High) Likelihood = (5 · 5 · 5)/25 = 5 (High) Risk = Critical Estimated remediation effort = 1
Cross-Site Scripting: Persistent	5	5	1	5	Impact = 5 (High) Likelihood = (5 · 1 · 5)/25 = 1 (Low) Risk = Medium Estimated remediation effort= 1

1.11. Legacy report components

The following sections provide information about the content and organization of the legacy report templates, which you can either modify or use as provided. Each report template includes several sections and subsections. The subsections provide charting and other data collection and presentation options.

This section contains the following topics:

1.11.1. Fortify Security Report

The Fortify Security Report is a high-level report that includes comprehensive analysis information and high-level details of the corresponding audit. This report also includes a high-level description and examples of the categories that have the highest priority. The following table lists Fortify Security Report sections and their corresponding subsections.

Section	Subsection
Executive Summary Presents an overview of the scan. This includes an overview of issues, an overview of issues by Fortify Priority Order, and recommendations for issue remediation. This section is designed for management and project managers.	**Issues Overview** Editable overview of the issues, including the date of the scan, number of issues, name of the project, scan summary, and total number of detected issues.
	Issue Summary by Fortify Priority Order Issues are categorized into the following four risk quadrants based on whether they have a high or low impact, and high or low likelihood of being exploited: • Critical - High impact and high likelihood. Critical issues are easy for the attacker to discover and exploit to result in extensive asset damage. • High - High impact but low likelihood. High priority issues are often difficult to discover and exploit, but can result in extensive asset damage. • Medium - Low impact but high likelihood. Medium priority issues are easy to discover and exploit, but often result in little asset damage. • Low - Low impact and low likelihood. Low priority issues are difficult to discover and exploit and typically result in little asset damage. You can present this information in table, pie chart, or bar chart.

	Recommendations and Conclusions High-level recommendations about how to remediate the issues listed in the Issue Summary by Fortify Priority Order subsection. You can edit the text in this subsection.
Project Summary Provides project summary information such as the codebase, scan information, results certifications, and so on.	**Code Base Summary** Summary of the analyzed codebase. You can edit the text element of this subsection.
	Scan Information Analysis details. You can edit the text element of this subsection.
	Results Certification Results certifications summary. You can edit the text element of this subsection.
	Attack Surface Attack surface summary. You can edit the text element of this subsection.
	Filter Set Summary Summary of the filter set used in the report. You can edit the text element of this subsection.
	Audit Guide Summary Summary of the audit guide. You can edit the text element of this subsection.
Results Outline Provides an outline of the results that OpenText SAST produced during the scan.	**Overall number of results** Total number of results that OpenText SAST produced during the scan. You can edit the text element of this subsection.
	Vulnerability Examples by Category Results summary of highest-level issues by category.

Detailed Project Summary Provides a detailed project summary.	**Files Scanned** List of all scanned files. You can edit the text element of this subsection.
	Reference Elements List of all libraries that OpenText SAST used in the translation phase of analysis. You can edit the text element of this subsection.
	Rulepacks List of Rulepacks that OpenText SAST used in the analysis. You can edit the text element of this subsection.
	Properties List of properties that OpenText SAST set in the analysis phase. You can edit the text element of this subsection.
	Commandline Arguments List of all options that OpenText SAST used in the translation phase of analysis. You can edit the text element of this subsection.
	Warnings List of all warnings issued during both the translation and analysis phases of the scan. You can edit the text element of this subsection.
Issue Count by Category Provides a chart of Issues by category. This chart is configurable.	**Issues By Category** Chart of issues by category. You can present the information in a table, pie chart, or bar chart.
Issue Breakdown by Analysis Provides a chart of issues by analysis. This chart is configurable.	**Issue By Analysis** Chart of issues by analysis. You can present the information in a table, pie chart, or bar chart.
New Issues Provides a chart of all new issues. This chart is configurable.	**New Issues** Chart of new issues. You can present the information in a table, pie chart, or bar chart.

1.11.2. Fortify Developer Workbook report

The Fortify Developer Workbook report provides a high-level summary of the vulnerabilities detected during a scan. This includes a report summary and an issue summary by Fortify Priority Order. This report is designed for developers. The following table lists Fortify Developer Workbook report sections and their corresponding subsections.

Section	Subsection
Report Overview Provides a high-level overview of report findings.	**Report Summary** Editable overview of the issues, including the date of the scan, name of the project, scan summary, and total number of detected issues.
	Issue Summary by Fortify Priority Order Issues charted based on Fortify Priority Order. You can present the information in a table, pie chart, or bar chart.
Issue Summary Provides the number and categories of vulnerabilities.	**Overall number of results** Total number of vulnerabilities. You can edit the text element of this subsection.
	Issues by Category Chart of issues based on category. You can present the information in a table, pie chart, or bar chart.
Results Outline Provides an outline of the results that OpenText SAST produced during the scan.	**Vulnerability Examples by Category** Results summary of highest-level issues by category.

1.11.3. OWASP Top Ten reports

The OWASP Top Ten reports provide high-level summaries of uncovered vulnerabilities organized based on the top ten issues identified by the Open Web Security Project (OWASP) for years 2017 and 2021. These reports include the sections and subsections described in the following table.

Section	Subsection
Report Overview Provides a high-level overview of report findings.	**Report Summary** Editable overview of vulnerabilities, including the date of the scan, the project name, and the total number of vulnerabilities.
	Issue Summary Chart of issues grouped by a selected attribute such as category, kingdom, or analysis type. You can present the information in a table, pie chart, or bar chart.
Issue Breakdown by OWASP Top Ten *<year>* Provides a chart of issues organized by OWASP top ten security risks.	**Issue Breakdown by OWASP Top Ten *<year>*** Chart of issues grouped by a selected attribute such as category, kingdom, or analysis type. You can present the information in a table, pie chart, or bar chart.
Results Outline Provides an outline of the results that OpenText SAST produced during the scan.	**Vulnerabilities by OWASP Top Ten *<year>*** List of the vulnerabilities organized by the OWASP Top Ten. You can select the listing to further refine and organize the vulnerabilities that Fortify Audit Workbench provides in the report.

1.11.4. Fortify scan summary report

The Fortify scan summary report type provides high-level information based on the category of issues that OpenText SAST found as well as a project summary and a detailed project summary. The following table provides descriptions of the report sections and subsections.

Section	Subsection
Issue Count by Category Provides a chart of issues by category.	**Issues By Category** Chart of issues grouped by a selected attribute such as category, kingdom, or analysis type. You can present the information in a table, pie chart, or bar chart.
Project Summary Provides project summary information, including codebase summary and general scan information.	**Code Base Summary** Summary of the codebase that OpenText SAST scanned, including the location of the code, the number of files, lines of code, and the build label. You can edit the text element of this subsection.
	Scan Information Scan information, including the OpenText SAST version, machine name, and the name of the user who ran the scan. You can edit the text element of this subsection.
	Results Certification Results certifications information, including the results certification summary and the details of the results certification. You can edit the text element of this subsection.
Detailed Project Summary Provides detailed project summary information including the files scanned, reference elements, and so on.	**Files Scanned** Lists all files that OpenText SAST scanned. You can edit the text element of this subsection.
	Reference Elements List of libraries that OpenText SAST used during the translation phase. You can edit the text element of this subsection.
	Rulepacks List of Rulepacks that OpenText SAST used during the analysis. You can edit the text element of this subsection.
	Properties List of properties that OpenText SAST set during the analysis phase. You can edit the text element of this subsection.

	Commandline Arguments List of all options that OpenText SAST used in the analysis phase. You can edit the text element of this subsection.
	Warnings List of all warnings issued during both the translation and analysis phases of the analysis. You can edit the text element of this subsection.