

OpenText™ Fortify Extensions for Visual Studio Code

Software Version: 23.1.0

User Guide

Document Release Date: August 2023

Software Release Date: August 2023

Legal Notices

Open Text Corporation

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Copyright Notice

Copyright 2020-2023 Open Text.

The only warranties for products and services of Open Text and its affiliates and licensors (“Open Text”) are as may be set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Open Text shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Trademark Notices

“OpenText” and other Open Text trademarks and service marks are the property of Open Text or its affiliates. All other trademarks or service marks are the property of their respective owners.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number
- Document Release Date, which changes each time the document is updated
- Software Release Date, which indicates the release date of this version of the software

This document was produced on August 01, 2023. To check for recent updates or to verify that you are using the most recent edition of a document, go to:

<https://www.microfocus.com/support/documentation>

Contents

Preface	5
Contacting Fortify Customer Support	5
For More Information	5
About the Documentation Set	5
Fortify Product Feature Videos	5
Change Log	6
Chapter 1: Getting Started	7
Software Requirements	7
Fortify Extensions for Visual Studio Code (Analysis) Requirements	7
Fortify Remediation Extension for Visual Studio Code Requirements	9
Installing the Fortify Extensions for Visual Studio Code	9
Related Documents	10
Fortify ScanCentral SAST	10
Fortify Software Security Center	10
Fortify Static Code Analyzer	11
Chapter 2: Analyzing your Code	12
Uploading Code to Fortify on Demand for Assessment	12
Performing a Local Analysis with Fortify Static Code Analyzer	14
Performing an Analysis Remotely with Fortify ScanCentral SAST	16
Chapter 3: Remediating your Code	18
Opening Fortify Software Security Center Application Versions	19
Viewing and Selecting Issues	21
Grouping Issues	22
Customizing Issue Visibility	24
Searching for Issues	25
Search Modifiers	26

Search Query Examples	31
Viewing Issue Information	32
Audit Tab	32
Analysis Trace	34
Recommendations Tab	36
Details Tab	36
History Tab	37
Comments Tab	37
Locating Issues in your Source Code	37
Adding Audit Information	37
Assigning Users to Issues	37
Assigning a Tag to an Issue	38
Adding Comments to Issues	38
Suppressing Issues	39
Send Documentation Feedback	40

Preface

Contacting Fortify Customer Support

Visit the Support website to:

- Manage licenses and entitlements
- Create and manage technical assistance requests
- Browse documentation and knowledge articles
- Download software
- Explore the Community

<https://www.microfocus.com/support>

For More Information

For more information about Fortify software products:

<https://www.microfocus.com/cyberres/application-security>

About the Documentation Set

The Fortify Software documentation set contains installation, user, and deployment guides for all Fortify Software products and components. In addition, you will find technical notes and release notes that describe new features, known issues, and last-minute updates. You can access the latest versions of these documents from the following OpenText Product Documentation website:

<https://www.microfocus.com/support/documentation>

To be notified of documentation updates between releases, subscribe to Fortify Product Announcements on the OpenText Community:

<https://community.microfocus.com/cyberres/fortify/w/fortify-product-announcements>

Fortify Product Feature Videos

You can find videos that highlight Fortify products and features on the Fortify Unplugged YouTube channel:

<https://www.youtube.com/c/FortifyUnplugged>

Change Log

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

Software Release / Document Version	Changes
23.1.0	Updated: <ul style="list-style-type: none">• This document has been updated to include the new Fortify Remediation Extension for Visual Studio Code that you can use to audit and remediate your code by reviewing analysis results on Fortify Software Security Center from VS Code (see "Remediating your Code" on page 18)• The Fortify Extension for Visual Studio was updated to include a link to the Fortify Remediation Extension for Visual Studio Code (see "Opening Fortify Software Security Center Application Versions" on page 19)
22.1.0	Updated: <ul style="list-style-type: none">• Minor edits

Chapter 1: Getting Started

This document describes how to install and use:

- The Fortify Extension for Visual Studio Code to analyze your project with Fortify Static Code Analyzer to uncover any security issues.

You can analyze your project with a locally installed Fortify Static Code Analyzer, upload the project to Fortify on Demand, or upload the project to Fortify ScanCentral SAST. When you analyze your project with Fortify ScanCentral SAST, you have the option to upload the analysis results to Fortify Software Security Center.

- The Fortify Remediation Extension for Visual Studio Code to review analysis results from Fortify Software Security Center so you can resolve security-related issues in VS Code.

This section contains the following topics:

- [Software Requirements](#) 7
- [Installing the Fortify Extensions for Visual Studio Code](#) 9
- [Related Documents](#)10

Software Requirements

This topic describes the Fortify Software that the Fortify extensions for VS Code work with and the requirements for each task.

Fortify Extensions for Visual Studio Code (Analysis) Requirements

To analyze your code, make sure the following requirements are met depending on the type of analysis you are using.

Software	Version	Requirements
Fortify on Demand	N/A	To upload your project to Fortify on Demand for assessment, make sure that you have the following:

Software	Version	Requirements
		<ul style="list-style-type: none">• Fortify ScanCentral SAST standalone client installed and included in the PATH environment variable• Fortify on Demand credentials• An application in Fortify on Demand with static scan settings configured
Fortify Static Code Analyzer	20.2.0 or later	<p>To scan your project locally with Fortify Static Code Analyzer, you must either:</p> <ul style="list-style-type: none">• Make sure that the PATH environment variable includes the sourceanalyzer executable• Have the full path to the Fortify Static Code Analyzer installation directory <p>Make sure that your system meets the system requirements for the Fortify Static Code Analyzer version you are using as described in the <i>Fortify Software System Requirements</i> document in Fortify Static Code Analyzer and Tools Documentation.</p>
Fortify ScanCentral SAST	20.2.0 or later	<p>To scan your project remotely with Fortify ScanCentral SAST, make sure that you have one of the following:</p> <ul style="list-style-type: none">• A ScanCentral Controller URL• A Fortify Software Security Center URL and an authentication token of type ScanCentralCtrlToken <p>For languages that are supported for analysis and system requirements for the Fortify ScanCentral SAST version you are using, see the <i>Fortify Software System Requirements</i> document in Fortify Static Code Analyzer and Tools Documentation.</p>

Software	Version	Requirements
Fortify Software Security Center	20.2.0 or later	To upload analysis results to Fortify Software Security Center after an analysis with Fortify ScanCentral SAST, make sure you have the following: <ul style="list-style-type: none">• An application version that exists in Fortify Software Security Center• Optionally, an authentication token of type CIToken This token is only required if the connection to Fortify ScanCentral SAST uses a ScanCentral Controller URL.

Fortify Remediation Extension for Visual Studio Code Requirements

To open analysis results on Fortify Software Security Center and audit and remediate your code, you must have the following:

- A Fortify Software Security Center URL.
The Fortify Software Security Center version must correspond with the Fortify Remediation Extension for Visual Studio Code version. The version number format is *<major>.<minor>.<patch>* (for example, 23.1.0). The *<major>* and *<minor>* portions of the Fortify Software Security Center and the Fortify Remediation Extension for Visual Studio Code version numbers must match. For example, versions 23.1.0 and 23.1.1 correspond.
- A user account on the Fortify Software Security Center server that has permission to access application versions.
To log into Fortify Software Security Center, you can use a user name and password or an authentication token of type ToolsConnectToken.
- To audit issues in the analysis results, your user account must have audit permissions.
- To add comments to issues or assign custom tags that require comments, your user account must have the permission to comment on issues.

Installing the Fortify Extensions for Visual Studio Code

You can install the extensions on a computer running Windows, Linux, or macOS. Install either extension from the Visual Studio Marketplace. See the Visual Studio Code documentation for instructions about how to install an extension. You can install the extension that best fits your needs or install both extensions.

Related Documents

This topic describes documents that provide information about Fortify software products.

Note: You can find the Fortify Product Documentation at <https://www.microfocus.com/support/documentation>. Most guides are available in both PDF and HTML formats.

Fortify ScanCentral SAST

The following document provides information about Fortify ScanCentral SAST. Unless otherwise noted, this document is available on the Product Documentation website at <https://www.microfocus.com/documentation/fortify-software-security-center>.

Document / File Name	Description
<i>Fortify ScanCentral SAST Installation, Configuration, and Usage Guide</i> SC_SAST_Guide_<version>.pdf	This document provides information about how to install, configure, and use Fortify ScanCentral SAST to streamline the static code analysis process. It is written for anyone who intends to install, configure, or use Fortify ScanCentral SAST to offload the resource-intensive translation and scanning phases of their Fortify Static Code Analyzer process.

Fortify Software Security Center

The following document provides information about Fortify Software Security Center. Unless otherwise noted, this document is available on the Product Documentation website at <https://www.microfocus.com/documentation/fortify-software-security-center>.

Document / File Name	Description
<i>Fortify Software Security Center User Guide</i> SSC_Guide_<version>.pdf	<p>This document provides Fortify Software Security Center users with detailed information about how to deploy and use Software Security Center. It provides all of the information you need to acquire, install, configure, and use Software Security Center.</p> <p>It is intended for use by system and instance administrators, database administrators (DBAs), enterprise security leads,</p>

Document / File Name	Description
	development team managers, and developers. Software Security Center provides security team leads with a high-level overview of the history and current status of a project.

Fortify Static Code Analyzer

The following documents provide information about 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
<i>Fortify Static Code Analyzer User Guide</i> SCA_Guide_<version>.pdf	This document describes how to install and use Fortify Static Code Analyzer to scan code on many of the major programming platforms. It is intended for people responsible for security audits and secure coding.
<i>Fortify Static Code Analyzer Applications and Tools Guide</i> SCA_Apps_Tools_<version>.pdf	This document describes how to install Fortify Static Code Analyzer applications and tools. It provides an overview of the applications and command-line tools that enable you to scan your code with Fortify Static Code Analyzer, review analysis results, work with analysis results files, and more.
<i>Fortify Static Code Analyzer Custom Rules Guide</i> SCA_Cust_Rules_Guide_<version>.zip	This document provides the information that you need to create custom rules for Fortify Static Code Analyzer. This guide includes examples that apply rule-writing concepts to real-world security issues. Note: This document is included only with the product download.
<i>Fortify License and Infrastructure Manager Installation and Usage Guide</i> LIM_Guide_<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.

Chapter 2: Analyzing your Code

The Fortify Extensions for Visual Studio Code provides three ways to analyze your source code to detect security vulnerabilities.

- Upload your currently opened project to Fortify on Demand for static assessment.
- Run a locally installed version of Fortify Static Code Analyzer on the currently opened project. To view the analysis results, open the Fortify Project Results (FPR) file in Fortify Audit Workbench.
- Run a remote analysis using Fortify ScanCentral SAST.
You can upload the analysis results to a Fortify Software Security Center server.

The following sections describe any prerequisites for each analysis method and the instructions for how to use it.

This section contains the following topics:

Uploading Code to Fortify on Demand for Assessment	12
Performing a Local Analysis with Fortify Static Code Analyzer	14
Performing an Analysis Remotely with Fortify ScanCentral SAST	16


Uploading Code to Fortify on Demand for Assessment

The Fortify on Demand task supports packaging of JavaScript and TypeScript projects for scanning.

You must have the standalone Fortify ScanCentral SAST client on the system where Fortify Extensions for Visual Studio Code is installed to upload code to Fortify on Demand. You can obtain the Fortify ScanCentral SAST client from the Fortify on Demand Tools page. For instructions on how to install the Fortify ScanCentral SAST client, see the README file included in the ZIP archive.

Important! Before you upload your code to Fortify on Demand, you must first configure the static scan settings in the Fortify on Demand portal.

To upload the opened project to Fortify on Demand for assessment:

1. If the extension is not open, click **Fortify**  in the activity bar.
2. Click **Fortify on Demand** in the side bar.
3. From the **API root URL** list, select your data center's API root URL.

- (Optional) Select the **Use proxy** check box to connect through a proxy and provide the settings described in the following table.

Field	Description
Proxy host	Type the name of the proxy server. Exclude the protocol from the proxy host (for example, some.secureproxy.com).
Proxy port	Type the port of the proxy server.
Use HTTPS	Select the check box to connect using HTTPS.
Use proxy credentials	Select the check box if the proxy server requires authentication. Type the account credentials for the proxy server.

- Select an authentication method and provide the relevant credentials described in the following table.

Authentication method	Procedure
API credentials	<ol style="list-style-type: none">In the Key box, type the API key.In the Secret box, type the API secret.
User credentials	<ol style="list-style-type: none">In the Username box, type the account username.In the Password box, type the account password.In the Tenant ID box, type the tenant ID.
Personal access token	<ol style="list-style-type: none">In the Username box, type the account username.In the Secret box, type the personal access token.In the Tenant ID box, type the tenant ID.

- In the **Release ID** box, type the Fortify on Demand release ID.

The release ID is configured when you save your static scan settings in the Fortify on Demand portal. You can find the release ID in the application release URL. In the example URL: <https://ams.fortify.com/Releases/258262/Overview>, the release ID is 258262. You can also find the release ID in the Fortify on Demand Static Scan Setup page.

Note: The release must have saved scan settings in the portal in order for the release ID to be used as a token.

- From the **Entitlement preference** list, select the entitlement preference. If multiple entitlements are available, the scan will use the oldest entitlement. If the release has an active subscription, the scan will use the active subscription.
- Select the **Purchase entitlement** check box to purchase an entitlement if none is available for the specified entitlement preference. If the purchase entitlements feature is not enabled for the

tenant, the Fortify Extensions for Visual Studio Code log will display an error message.


9. From the **Remediation preference** list, select whether to run the scan as a remediation scan.
10. From the **Action for In-Progress scan** list, select the action to take if the release has an in-progress scan:
 - **Do Not Start Scan**—Do not start a new scan and fail the task
 - **Cancel In Progress Scan**—Cancel the scan in progress and start a new scan (if the scan in progress can be automatically canceled)
 - **Queue**—Queue the scan (if the scan queue limit has been reached, the scan will be canceled)
11. Click **Upload**.

If the project is successfully uploaded, the Fortify Extensions for Visual Studio Code log displays a 200 OK status code and the scan ID. The Fortify on Demand Scans pages display a new scan for the release.

Performing a Local Analysis with Fortify Static Code Analyzer

You must have Fortify Static Code Analyzer locally installed.

To scan the opened project with Fortify Static Code Analyzer:

1. If the extension is not open, click **Fortify**  in the activity bar.
2. Click **Static Code Analyzer** in the side bar.
3. In the **Static Code Analyzer executable path** box, type the path to the Fortify Static Code Analyzer executable or click **Browse** to find the file on your system.

Type `sourceanalyzer` to use the executable that is in the PATH environment variable. This is the default.

4. In the **Build ID** box, type a unique identifier for the analysis.
5. (Optional) In the **Scan results location (FPR)** box, type a name for the Fortify Project Results file (for example, `MyProjectA.fpr`).

If you do not provide an analysis results file name, then Fortify Extensions for Visual Studio Code uses the name of the current project folder for the FPR file and saves the FPR in the current project folder.

Note: If you do not specify a path for the FPR, then on Windows the FPR is saved in `C:\Users\\AppData\Local\Programs\Microsoft VS Code`. On macOS, the default path is `/Users/<username>/projectRoot`.

6. (Optional) To specify a custom location for the Fortify Static Code Analyzer log file, type a file name (or a full path) in the **Log location** box.

By default, the Fortify Extensions for Visual Studio Code saves the log file in the following location:

- Windows: C:\Users*<username>*\AppData\Local\Fortify\sca*<version>*\log
- Non-Windows: *<userhome>*/.fortify/sca*<version>*/log

where *<version>* is the version of Fortify Static Code Analyzer that you are using.

After the scan is complete, you can click **Open** to the right of **Log location** to see the log file.

7. (Optional) To add additional Fortify Static Code Analyzer translation options:
 - a. Select the **Add translation options** check box.
 - b. Type Fortify Static Code Analyzer translation options.
See the *Fortify Static Code Analyzer User Guide* in [Fortify Static Code Analyzer and Tools Documentation](#) for detailed information about the available translation options.
8. (Optional) To add additional Fortify Static Code Analyzer scan options:
 - a. Select the **Add scan options** check box.
 - b. Type Fortify Static Code Analyzer scan options.
See the *Fortify Static Code Analyzer User Guide* in [Fortify Static Code Analyzer and Tools Documentation](#) for detailed information about the available scan options.
9. To download Fortify security content before the scan, select the **Update security content** check box.




If you are using a Fortify Rulepack update server other than <https://update.fortify.com> or if you require proxy information for the connection, you must use the Fortify Static Code Analyzer post-install tool (scapostinstall) to configure this information before you can update Fortify security content. See the *Fortify Static Code Analyzer User Guide* in [Fortify Static Code Analyzer and Tools Documentation](#) for more information.
10. Click **Scan**.

When the scan is complete, Fortify Extensions for Visual Studio Code displays the **FPR path** in an information message .

Update security content


Cancel

Scan

 FPR path: my_proj_scan.fpr  

Source: Fortify Extension for Visual Studio Code (Extension)

Open

 Execution of sourceanalyzer -b myproj -Dcom.fortify.sca.hoa.Enable=tr...


To view the analysis results in Fortify Audit Workbench, click **Open** to the right of **Scan results location (FPR)** box. For information about using Fortify Audit Workbench, see the *Fortify Audit Workbench User Guide* in [Fortify Static Code Analyzer and Tools Documentation](#).

Performing an Analysis Remotely with Fortify ScanCentral SAST

You must have a properly configured Fortify ScanCentral SAST installation. For more information, see the *Fortify ScanCentral SAST Installation, Configuration, and Usage Guide* in [Fortify Software Security Center Documentation](#).

You can connect to Fortify ScanCentral SAST using either a ScanCentral Controller URL or a Fortify Software Security Center URL for a server that is integrated with Fortify ScanCentral SAST.

To upload the opened project for analysis by Fortify ScanCentral SAST:

1. If the extension is not open, click **Fortify**  in the activity bar.
2. Click **ScanCentral SAST** in the side bar.
3. In the **ScanCentral executable location** box, type the path to the Fortify ScanCentral SAST executable or click **Browse** to find the file on your system.

The standalone Fortify ScanCentral SAST client can be installed anywhere on your system. By default, Fortify Extensions for Visual Studio Code looks for the Fortify ScanCentral SAST executable in the Fortify Static Code Analyzer installation (`<sca_install_dir>/bin`) directory.

4. Under **Controller connection**, select how you want to connect to Fortify ScanCentral SAST. Do one of the following:

- Select **Controller URL** and then in the **Controller URL** box, type the ScanCentral Controller URL.

The format for the ScanCentral Controller URL is: `<protocol>://<controller_host>:<port>/scancentral-ctrl` (for example: `https://myControllerHost.com:8443/scancentral-ctrl`).

- Select **SSC URL**, and then provide the following:

- i. In the **Software Security Center URL** box, type the server URL.

The format for the Fortify Software Security Center URL is: `<protocol>://<ssc_host>:<port>[/ssc]` (for example: `http://my.domain.com:8080/ssc`).

- ii. In the **Controller token** box, paste the decoded value for a Fortify Software Security Center authentication token of type ScanCentralCtrlToken.

5. (Optional) In the **Notification email** box, type an email address to which the ScanCentral Controller will send job status notifications.
6. (Optional) To upload the analysis results to Fortify Software Security Center:
 - a. Select the **Upload scan results to Software Security Center** check box.
 - b. Specify an existing application name and application version.

- c. Do one of the following:
 - If you are using the Fortify Software Security Center URL for the connection to Fortify ScanCentral SAST, select the **Use Controller token** check box.
 - In the **Continuous integration token** box, paste the decoded value for a Fortify Software Security Center authentication token of type CIToken.
7. (Optional) To specify additional Fortify Static Code Analyzer translation or scan options. select the **Additional options** check box to add any translation and scan options.

See the *Fortify Static Code Analyzer User Guide* in [Fortify Static Code Analyzer and Tools Documentation](#) for information about the available translation and scan options.

8. (Optional) To specify a custom location for the Fortify ScanCentral SAST log file, type a file name (or a full path) in the **Log location** box.

By default, the Fortify Extensions for Visual Studio Code saves the log file in the following location:

- Windows: C:\Users*<username>*\AppData\Local\Fortify\scancentral-*<version>*\log
- Non-Windows: *<userhome>*/.fortify/scancentral-*<version>*/log

where *<version>* is the version of Fortify ScanCentral SAST that you are using.

After the scan is complete or if any errors occurred, you can click **Open** to the right of **Log location** to see the log file.

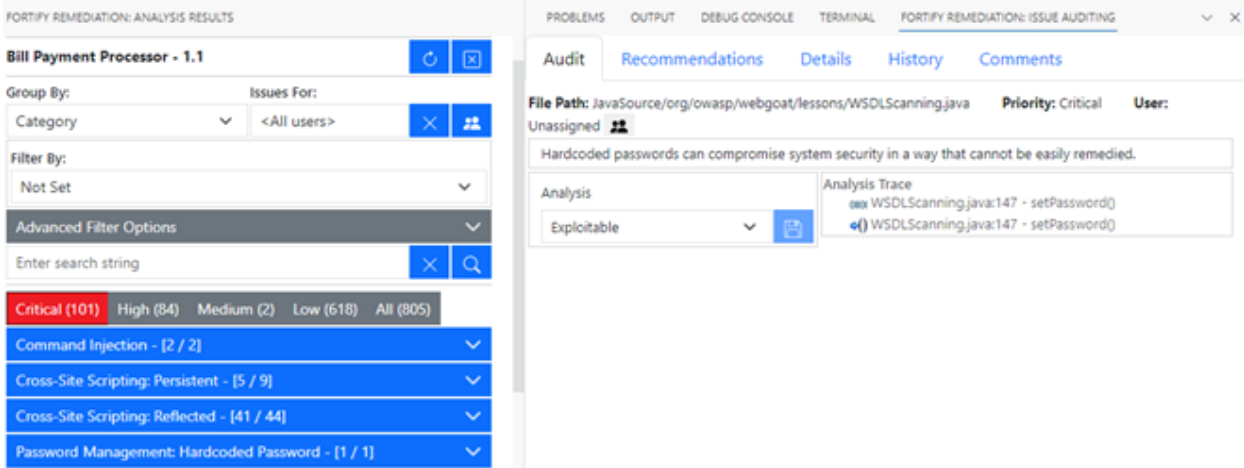
9. Click **Scan**.

When the scan request to Fortify ScanCentral SAST is complete, Fortify Extensions for Visual Studio Code displays the status in an information message.

You can view the analysis results on Fortify Software Security Center if you uploaded them to the server.

Chapter 3: Remediating your Code

After you open an application version on Fortify Software Security Center, the Fortify Extension for Visual Studio Code displays the analysis results in the **Analysis Results** view. This view displays all security issues, organized into tabs, which by default correspond to Fortify priority values. For example, the **Critical** folder contains all critical issues for a project and the **Low** folder contains all low-priority issues. Filters available for the application version determine which issues are visible. After you select an issue in the **Analysis Results** view, the **Issue Auditing** panel displays five tabs that provide information specific to the selected issue.



To remediate issues, the project you have open in VS Code must correspond to the application version you opened from Fortify Software Security Center (see ["Opening Fortify Software Security Center Application Versions"](#) on the next page).



This section contains the following topics:

- [Opening Fortify Software Security Center Application Versions](#) 19
- [Viewing and Selecting Issues](#) 21
- [Grouping Issues](#) 22
- [Customizing Issue Visibility](#) 24
- [Searching for Issues](#) 25
- [Viewing Issue Information](#) 32
- [Locating Issues in your Source Code](#) 37
- [Adding Audit Information](#) 37


Opening Fortify Software Security Center Application Versions

To view the analysis results, you must first connect to Fortify Software Security Center and open an application version.

To open an application version:

1. If the Fortify Remediation Extension for Visual Studio Code is not open, do one of the following:
 - Click **Fortify Remediation**  in the activity bar.
 - If you have the Fortify Extension for Visual Studio installed, click **Fortify**  in the activity bar, and then click **Remediation** in the side bar.
If necessary, the extension is automatically installed.
2. Configure the Fortify Software Security Center connection settings in VS Code **Settings**:
 - a. Open the VS Code **Settings** and search for `Fortify Remediation`.
 - b. In the **Software Security Center URL** box, type the URL for your Fortify Software Security Center server.
 - c. (Optional) From the **Login Method** list, select a default login method.
 - d. (Optional) Select **Save Token** to save the authentication token value after a successful login for future connections to Fortify Software Security Center.


Note: For the **Username/Password** login method, the user name is always saved for future connections. The password is never saved.

3. Click **Fortify Remediation**  in the activity bar.
The **Analysis Results** view opens in the side bar.

FORTIFY REMEDIATION: ANALYSIS RESULTS

Connect to Software Security Center

Login method

Username/Password 

User

User name is required

Password

Password is required



Connect

4. From the **Login method** menu, select the login method set up for you in Fortify Software Security Center.
5. Depending on the selected login method, follow the procedure described in the following table.

Login method	Procedure
Username/Password	Type your Fortify Software Security Center user name and password.
Authentication Token	Specify the decoded value of a Fortify Software Security Center authentication token of type ToolsConnectToken.


6. Click **Connect** to connect to Fortify Software Security Center.
The **Select Application Version** displays the application versions that your user account has permission to access.
7. Select an application name and version, and then click **Open**.

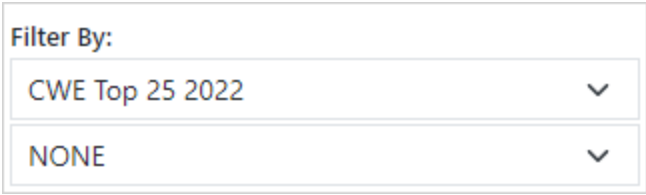
The Fortify Remediation Extension for Visual Studio Code displays the analysis results for the selected Fortify Software Security Center application version (see ["Viewing and Selecting Issues" on the next page](#)).

Note: To open a different application version on the same Fortify Software Security Center server to which you are already connected, click **Close application** . To switch to a different Fortify Software Security Center instance, select **Log out**  and then reconnect to Fortify Software Security Center as described in this topic.

Viewing and Selecting Issues

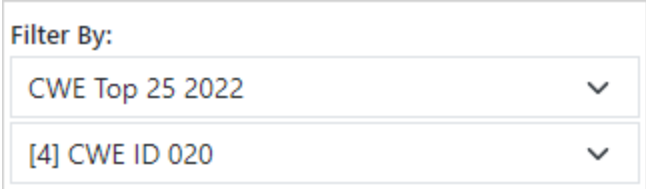
To view and select issues in an opened application version:

1. From the **Group By** list, select an attribute for sorting issues in all visible folders into groups. The default grouping is **Category**. For a description of the available **Group By** attributes, see ["Grouping Issues" on the next page](#).
2. By default, issues assigned to your Fortify Software Security Center user account are shown. From the **Issues For** list, you can do either of the following:
 - To show issues for all users, click **Clear** .
 - Select a Fortify Software Security Center user name.
3. To filter the issues within the selected grouping:
 - a. From the **Filter By** list, select a filter category.



The screenshot shows a 'Filter By:' dropdown menu with two visible options: 'CWE Top 25 2022' and 'NONE'. Each option has a downward-pointing chevron icon to its right.

- b. To refine the issues further, select a filter option from the list below the selected filter category.

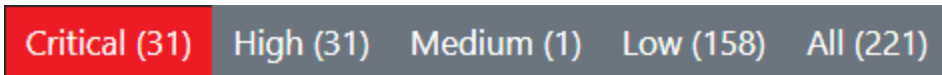


The screenshot shows a 'Filter By:' dropdown menu with two visible options: 'CWE Top 25 2022' and '[4] CWE ID 020'. Each option has a downward-pointing chevron icon to its right.

4. To apply a filter set to the issues, click **Advanced Filter Options**, and then from **Filter Set**, select one of the following filter sets to apply to issues:
 - Select **Security Auditor View** to list all issues relevant to a security auditor.
 - Select **Quick View** to list only 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).

Note: The filter sets available depend on the issue template assigned to the application version you opened.

5. Click a tab to view the associated issues.



Note: The tabs shown depend on your **Group By**, **Issues For**, and **Filter Set** selections. It is possible that not all tabs are shown. The tabs shown also depend on the issue template associated with the application version.

- The **Critical** tab contains issues that have a high impact and a high likelihood of exploitation. We recommend that you remediate critical issues immediately.
- The **High** tab contains issues that have a high impact and a low likelihood of exploitation. We recommend that you remediate high issues with the next patch release.
- The **Medium** tab contains issues that have a low impact and a high likelihood of exploitation. We recommend that you remediate medium issues as time permits.
- The **Low** tab contains issues that have a low impact and a low likelihood of exploitation. We recommend that you remediate low issues as time permits (your organization can customize this category).
- The **All** tab contains all issues.

Within each tab, issues are grouped by the **Group By** selection. After each grouping name, enclosed in brackets, is the number of audited issues and the total number of issues in the group. For example, **Command Injection - [1 / 3]** indicates that one issue out of three categorized as Command Injection was audited.

6. Click to expand a grouping and view the issues it contains.
The Fortify Extension for Visual Studio Code retrieves the corresponding issues from Fortify Software Security Center.
7. Select an issue to view its details in the **Issue Auditing** panel.

Grouping Issues

The issues visible in the **Analysis Results** view vary depending on the selected grouping attribute. The value you select from the **Group By** list sorts issues in all visible folders into subfolders. Use the **Group By** attributes to group and view the issues in different ways. The following table describes the available **Group By** attributes.

Attribute	Description
Analysis	Groups issues by the Analysis tag value assigned, such as Suspicious, Exploitable, and Not an Issue.
Analysis Type	Groups issues by analyzer product, such as SCA, WEBINSPECT, and SECURITYSCOPE (WebInspect Agent).

Attribute	Description
Analyzer	Groups issues by analyzer group, such as Control Flow, Data Flow, Pentest, 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 setting.
<custom_tagname>	Groups issues by the selected custom tag value assigned.
Engine Priority	<p>Groups issues based on the original priority value determined by the engine that identified the issue.</p> <p>Note: This is only available in Fortify Software Security Center version 22.2.0 or later.</p>
File Name	Groups issues by file name.
Folder	Groups issues by folders defined in the issue template.
Fortify Priority Order	Groups issues by Critical, High, Medium, and Low based on the issue priority.
Introduced date	Groups issues by the date the issue was first detected.
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.
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 Fortify WebInspect.
<metadata_listname>	Groups issues using the alternative metadata external list names (for example, CWE, OWASP Top 10 <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 NEW group and the others are displayed in the UPDATED

Attribute	Description
	group. If removed issues are visible, issues not found in the latest scan are displayed in the REMOVED group.
Package	Groups issues by package or namespace. Nothing is shown for projects to which this option does not apply, such as C projects.
Primary Context	Groups issues where the primary location or sink node function call occurs in the same code context.
Priority Override	Groups issues by the Priority Override tag value assigned.
Sink	Groups issues that share the same dataflow sink function.
Source	Groups issues that share the same dataflow source functions.
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).
Taint Flag	Groups issues by the taint flags that they contain.
URL	Groups dynamic issues by the request URL.

Customizing Issue Visibility

You can customize the issues list in the **Analysis Results** view to determine which issues the Fortify Extension for Visual Studio Code displays.

To customize the display of hidden, removed, and suppressed issues:

1. In the **Analysis Results** view, expand the **Advanced Filter Options** section.
2. Under **Issue Visibility**, select or clear the following options:
 - To display all hidden issues, select **Show Hidden**.

Note: The visibility filter settings in the issue template associated with the application version determine which issues are hidden.

- To display all the issues removed since the previous analysis, select **Show Removed**.

- To display all suppressed issues, select **Show Suppressed**.

Note: Users who audit issues can suppress specific types of issues that are not considered high priority or of immediate concern. For example, auditors can suppress issues that are fixed, or issues that your organization plans not to fix.

The Fortify Remediation Extension for Visual Studio Code displays issues based on your selection.

Note: You can also specify issue visibility options in the Visual Studio Code Settings for **Fortify Remediation**.

Searching for Issues

You can use the search box above 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 indicate the type of comparison to perform for a search in the **Analysis Results** view, wrap the search terms with delimiters. The following table shows the syntax to use for the search string.

Comparison	Description
contains	Searches for a term without any qualifying delimiters
equals	Searches for an exact match if the term is wrapped in quotation marks (" ")
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 equals	Excludes issues specified by the string by preceding the string with an 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>`. For more information, see ["Search Modifiers" on the next page](#).

A search string 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 string, 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.

Search Modifiers

You can use a search modifier to specify to which attribute of an issue 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 string based on the following issue 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 result that contains the specified 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	Description
<code>accuracy</code>	Searches for issues based on the accuracy value specified (0.1 through 5.0).
<code>analysis</code>	Searches for issues that have the specified audit analysis value, such as <code>exploitable</code> , <code>not an issue</code> , and so on.
<code>[analysis type]</code>	Searches for issues based on the analyzer product such as SCA and WEBINSPECT.
<code>analyzer</code>	Searches the issues for the specified analyzer such as <code>control flow</code> , <code>data flow</code> , <code>structural</code> , and so on.
<code>[app defender protected]</code> (def)	Searches for issues based on whether Application Defender can protect the vulnerability category (<code>protected</code> or <code>not protected</code>).
<code>[attack payload]</code>	Searches for issues that contain the search term in the part of the request that caused the vulnerability for penetration test results.

Search modifier	Description
[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 the intended audience, such as dev, targeted, medium, broad, and so on. Caution! This metadata is legacy information that is no longer used and will be removed in a future release. Fortify recommends that you not use this search modifier.
audited	Searches for issues based on whether the primary tag is set (true or false). 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.
category (cat)	Searches for the specified category or category substring.
class	Searches for issues based on the specified class name.
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 those detected by 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.

Search modifier	Description
	<p>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, <code>analysis:[0,2]</code> 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).</p> <p>To search for a specific date in a date-type custom tag, specify the date in the format: <code>yyyy-mm-dd</code>.</p> <p>To search for issues that have no value set for a custom tag, use <code><none></code> as the search term. For example, to search for all issues that have no value set in the custom tag labeled Target Date, type: <code>[Target Date]:<none></code>.</p>
<p><code>[engine priority]</code></p>	<p>Searches for issues based on the original priority value determined by the engine that identified the issue.</p> <p>Note: This modifier is only available in Fortify Software Security Center version 22.2.0 or later.</p>
<p><code>file</code></p>	<p>Searches for issues where the primary location or sink node function call occurs in the specified file path.</p>
<p><code>[fortify priority order]</code></p>	<p>Searches for issues that have a priority level that matches the specified issue priority. Valid values are <code>critical</code>, <code>high</code>, <code>medium</code>, and <code>low</code>.</p>
<p><code>headers</code></p>	<p>Searches for issues that contain the search term in the request header for penetration test results.</p>
<p><code>historyuser</code></p>	<p>Searches for issues that have audit data modified by the specified user.</p>
<p><code>[http version]</code></p>	<p>Searches for issues based on the specified HTTP version such as <code>HTTP/1.1</code>.</p>
<p><code>impact</code></p>	<p>Searches for issues based on the impact value specified (0.1 through 5.0).</p>

Search modifier	Description
[instance id]	Searches for an issue based on the specified instance ID.
[issue age]	Searches for the issue age, which is either 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. Also see "sourceline" on page 31 .
manual	Searches for issues that were manually created by penetration test tools, and not automatically produced by a web crawler such as Fortify WebInspect.
[mapped category]	Searches for issues based on the specified category that is mapped across the various analyzers (Fortify Static Code Analyzer, Fortify WebInspect, and Fortify WebInspect 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 (for example, [owasp top 10 <year>], [cwe top 25 <year>], [pci ssf <version>], [stig <version>], and others).
method	Searches for issues based on the method, such as GET, POST, and so on.

Search modifier	Description
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. Also see "sink" below and "[source context]" on the next page .
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.
response	Searches for issues that contain the search term in the response from the protocol used in penetration test results.
severity (sev)	Searches for issues based on the specified severity value (legacy metadata).
sink	Searches for issues that have the specified sink function name. Also see "[primary context]" above .
source	Searches for dataflow issues that have the specified source function name. Also see "[source context]" on the next page .

Search modifier	Description
[source context]	Searches for dataflow issues that have the source function call in the specified code context. Also see "source" on the previous page and "[primary context]" on the previous page .
sourcefile	Searches for dataflow issues with the source function call that the specified file contains. Also see file .
sourceline	Searches for dataflow issues having taint source entering the flow on the specified line.
status	Searches issues that have the status reviewed, not reviewed, or under review.
suppressed	Searches for suppressed issues.
taint	Searches for issues that have the specified taint flag.
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 URL.
user	Searches for issues assigned to the specified user.

Search Query Examples

The following table contains search query examples.

Search task	Example query
Find privacy violations in file names that contain jsp with getSSN() as a source	category:"privacy violation" source:getssn file:jsp
Find file names that contain com/test/123	file:com/test/123
Find issues that contain cleanse as part of any modifier	cleanse

Search task	Example query
Find suppressed vulnerabilities with asdf in the comments	suppressed:true comments:asdf
Find all categories except for SQL Injection	category:!SQL Injection
Find issues that have a value specified for a custom tag labeled version	version:!<none>

Viewing Issue Information

After you select an issue in the **Analysis Results** view, the Fortify Extension for Visual Studio Code displays the issue-specific content in the in the **Issue Auditing** panel on the **Audit**, **Recommendations**, **Details**, **History**, and **Comments** tabs.

Audit Tab

The **Audit** tab provides a dashboard of analysis information for the selected issue.

Note: Any changes you make on the **Audit** tab are automatically uploaded to the application version in Fortify Software Security Center.

The following table describes the **Audit** tab features.

Element	Description
Priority	The Fortify priority value determined for the selected issue.

Element	Description
	<p>If Fortify Software Security Center has Priority Override enabled and the priority value was changed, then the current priority value is displayed with the original Fortify priority value in parentheses. For instructions on how to change the priority override, see "Assigning a Tag to an Issue" on page 38.</p>
User	<p>User assigned to the selected issue. For instructions on how to assign a user to an issue, see "Assigning Users to Issues" on page 37.</p>
Analysis	<p>Your assessment of the selected issue. To change the assessment, select an item from the list. This is the primary tag defined in Fortify Software Security Center for the application version. The default primary tag is Analysis, but your organization might have a different tag designated as the primary tag.</p>
<custom_tagname>	<p>Any custom tags your organization has defined in Fortify Software Security Center. If available, these are displayed below the primary tag. For information on how to make changes to these tags, see "Assigning a Tag to an Issue" on page 38.</p> <p>If the audit results have been submitted to Fortify Audit Assistant in Fortify Software Security Center, then in addition to any other custom tags, the tab displays the following tags:</p> <ul style="list-style-type: none"> • 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. This is a percentage expressed in values that range from 0.000 to 1.000. For example, a value of 0.982 indicates a confidence level of 98.2 percent. You cannot change this tag value. • AA_Training—Whether to include or exclude the issue from Fortify Audit Assistant training. You can modify this value <p>For more information about Fortify Audit Assistant, see the <i>Fortify Software Security Center User Guide</i> in Fortify Software Security Center Documentation.</p>
Suppress	<p>Suppresses the issue. For information about suppressing issues, see "Suppressing Issues" on page 39.</p>
Analysis Trace	<p>Items of evidence that the analyzer uncovered. The analysis trace evidence is presented in the order it was discovered. For descriptions of the analysis trace icons, see "Analysis Trace" on the next page.</p>










See Also




["Adding Audit Information" on page 37](#)

Analysis Trace

The analysis trace on the **Audit** tab is presented in sequential order. For dataflow issues, this trace is a presentation of 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 box shows the direction of the dataflow in this section of the source code.

The analysis trace box uses the icons described in the following table to show how the dataflow moves in this section of the source code or execution order.

Icon	Description
	Data is assigned to a field or variable
	Information is read from a source external to the code (HTML form, URL, and so on)
	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 parameter to another
	Note: This is typically shown as <code>functionA(x : y)</code> to indicate that data is transferred from x to y. The x and y values are one of the following: <ul style="list-style-type: none">• An argument index• <code>return</code>—The return value of a function• <code>this</code>—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

Icon	Description
	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 box can contain 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.

Recommendations Tab

The **Recommendations** tab provides suggestions and examples on how to secure a vulnerability or remedy a bad practice. The following table describes the sections on this tab.

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.

Details Tab

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

Section	Description
Abstract/Custom Abstract	Summary of the selected issue, including any custom abstracts defined by your organization.
Explanation/Custom Explanation	Description of the conditions under which an issue of the selected type occurs. This includes a discussion of the vulnerability, the constructs typically associated with it, ways in which attackers can exploit it, and the potential ramifications of an attack. This section also includes any custom explanations defined by your organization.
Instance ID	Unique identifier for the issue.
Primary Rule ID	Identifier for the primary rule used to uncover 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.

History Tab

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

Comments Tab

The **Comments** tab displays all comments that were submitted for the issue. To add a comment for an issue, see ["Adding Comments to Issues" on the next page](#).

Locating Issues in your Source Code


You can use the Fortify Remediation Extension for Visual Studio Code to locate security-related issues in your code. Make sure that the revision of the source code open in VS Code corresponds to the application version you opened on Fortify Software Security Center.

To locate issues in the source code, do one of the following:

- Select an issue in the **Analysis Results** view.
- From the **Audit** tab, select a line in the Analysis Trace.


VS Code places the focus on the line of code that contains the selected security-related issue.

Adding Audit Information

After you select and review an issue, you can add audit information on the **Audit** tab. To see any updates to the audit results made in Fortify Software Security Center, click **Refresh** .

Assigning Users to Issues

To assign a user to an issue:

1. From the **Analysis Results** view in the side bar, select an issue.
2. In the **Issue Auditing** panel, click the **Audit** tab.
3. Click **Select User** , select a user name, and then click **Save**.

To leave the issue unassigned, click **Unassign User** .

The Fortify Remediation Extension for Visual Studio Code makes the update to the application version in Fortify Software Security Center.



Assigning a Tag to an Issue

To assign a custom tag value to an issue:

1. From the **Analysis Results** view, select an issue.
2. From the **Analysis** list on the **Audit** tab, select a value that reflects your evaluation of this issue.
This is the primary tag as defined in Fortify Software Security Center. The default primary tag is **Analysis**, but your organization might have a different tag designated as the primary tag.
3. If the priority override capability is enabled on Fortify Software Security Center, you can override the priority value for the issue as follows:
 - a. From the **Priority Override** list, select the preferred priority value.
 - b. Explain why you changed the value in the comment box outlined in red.
4. If custom tags defined for the project exist, provide values for them.

The Fortify Remediation Extension for Visual Studio Code displays all custom tags assigned to the application; however, you can only provide values for tags that your Fortify Software Security Center user account has permission to edit.

Use the following instructions to provide a value for a custom tag:

- For text- and decimal-type custom tags, type the value in the box. Text-type custom tags accept up to 500 characters (HTML/XML tags and newlines are not allowed).
 - For date-type custom tags, type a date or click **Select Date**  to select a date from a calendar.
5. If a tag requires a comment, then after you provide a value for the tag, then you must type a comment in the box outlined in red.
 6. Click Save .

The Fortify Remediation Extension for Visual Studio Code makes the updates to the application version in Fortify Software Security Center.

Adding Comments to Issues

The comments tab in the **Fortify Remediation: Issue Auditing** panel displays any comments submitted for the selected issue.

To add a comment to an issue:

1. From the **Analysis Results** view in the side bar, select an issue.
2. In the **Issue Auditing** panel, click the **Comments** tab.
3. In the **Enter comment** box, type your comment.
4. Click **Save**.

The Fortify Extension for Visual Studio Code makes the update to the application version in Fortify Software Security Center.

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:

1. In the **Analysis Results** view, select the issue.
2. In the **Issue Auditing** panel, click the **Audit** tab, and then click **Suppress**.

By default, Fortify Remediation Extension for Visual Studio Code automatically refreshes the **Analysis Results** view issue list and hides suppressed issues.

To display issues that have been suppressed:

- In the **Analysis Results** view, expand **Advanced Filter Options**, and then select **Show Suppressed**.

Note: You can review suppressed issues by searching for them using the suppressed search modifier.

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

1. In the **Analysis Results** view, select the suppressed issue.
Each suppressed issue is tagged with an "S" icon.
2. In the **Issue Auditing** panel, click the **Audit** tab, and then click **Unsuppress**.

Send Documentation Feedback

If you have comments about this document, you can [contact the documentation team](#) by email.

Note: If you are experiencing a technical issue with our product, do not email the documentation team. Instead, contact Customer Support at <https://www.microfocus.com/support> so they can assist you.

If an email client is configured on this computer, click the link above to contact the documentation team and an email window opens with the following information in the subject line:

Feedback on User Guide (Fortify Extensions for Visual Studio Code 23.1.0)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to fortifydocteam@microfocus.com.

We appreciate your feedback!