OpenText™ Fortify Azure DevOps Extension

Fortify Azure DevOps Extension User Guide

Version : 10.0

PDF Generated on : 26/09/2025

Table of Contents

1. Fortify Azure DevOps Extension User Guide

Software Version: 10.0

Document Release Date: September 2025

Software Release Date: September 2025

1.1. 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**
10.0	Added: • Upgraded NodeJS version from 10 to 20 for all tasks and updated related npm packages • Abillity to automatically download the ScanCentral SAST client from the Controller. For more information, see Using the Fortify Static Code Analyzer Install task. • Support for SAST_LOCATION environment variable for ScanCentral SAST client 25.4. • Support for encoded SSC authentication tokens. • Ability to show the Skip build option when you select **None** in the **Build Tool** list in a ScanCentral SAST task
9.5	Added: • Added an option to automatically detect build tools for the project files being scanned. For more information, see Adding a Fortify ScanCentral SAST Assessment task. • Added DotNet as a build tool option in Adding a Fortify ScanCentral SAST Assessment task. • Added ability to specify Fortify Static Code Analyzer translation and scan options, and to exclude files and directories from the package in Adding a Fortify ScanCentral SAST Assessment task. • Added an option to download debug logs in a ZIP file in Adding a Fortify ScanCentral SAST Assessment task. Updated: • Added information on configuring `SCANCENTRAL_JAVA_HOME` environment variables for Fortify ScanCentral SAST client in Requirements for the Fortify ScanCentral SAST task. Removed: • The Targeted Visual Studio environment field has been removed from the Fortify Static Code Analyzer Install task (see Using the Fortify Static Code Analyzer Install task). • Removed the chapter *Getting started with Fortify WebInspect*.
9.4	Updated: • Streamlined Fortify on Demand topics and removed discontinued Fortify on Demand tasks. Added: • Adding a DAST Automated assessment task
8.10	Updated: • Added information about Fortify ScanCentral SAST client requirements and troubleshooting (see Requirements for the Fortify ScanCentral SAST Task and Troubleshooting the Fortify ScanCentral SAST Task)

8.9	Updated: • Added information on packaging files required for Debricked open source scans (see Adding a Static Assessment Task)

1.2. Introduction

The Fortify Azure DevOps Extension (formerly the Fortify VSTS Extension) adds static and dynamic analysis to your continuous integration (CI) and continuous delivery (CD) builds. This integration helps you identify application vulnerabilities earlier in the software development lifecycle.

This document describes how to use the Fortify Azure DevOps Extension. This document assumes that you have a working knowledge of Azure DevOps and know how to use Azure Pipelines for your CI/CD solutions. This extension includes the tasks described in the following table.

Note

If you use any Fortify Azure DevOps task that requires access to an external server such as Fortify Software Security Center or Fortify ScanCentral (SAST or DAST) and the server's certificates are self-signed, then you must extend the node.js predefined root certificate authority (CA) with extra certificates. Do this by setting the NODE_EXTRA_CA_CERTS environment variable. For more information, see the node.js command-line options documentation.

Task (version)	Description	More information
Fortify Static Code Analyzer Install (8.x)	The Fortify Static Code Analyzer Installation task automatically installs and configures Fortify Static Code Analyzer.	Getting Started with Fortify Static Code Analyzer
Fortify Static Code Analyzer Assessment (7.x)	The Fortify Static Code Analyzer Assessment task enables you to run Fortify Static Code Analyzer as a build step. After the analysis is complete, the scan results are available as a Fortify Project Results (FPR) file. You can publish the FPR as a build artifact. To review the scan results, download this artifact and open it in either Fortify Audit Workbench or Fortify Software Security Center. You can also configure the task to upload the scan results to a Fortify Software Security Center server.	Getting Started with Fortify Static Code Analyzer
Fortify on Demand Static Assessment 9.x)	The Fortify on Demand Static Assessment task submits a static scan request and uploads code to Fortify on Demand as a build step. The scan results are available in Fortify on Demand.	Getting started with Fortify on Demand
FoD DAST Automated (2.x)	The FoD DAST Automated task submits an automated dynamic scan request to Fortify on Demand as a build step. The scan results are available in Fortify on Demand.	Getting started with Fortify on Demand
Fortify ScanCentral SAST Assessment (7.x)	The Fortify ScanCentral SAST Assessment task submits a static scan request to a ScanCentral SAST Controller (using a ScanCentral SAST client) as a build step. You can also configure the task to upload the scan results to Fortify Software Security Center.	Getting Started with Fortify ScanCentral SAST

Fortify ScanCentral DAST Assessment (7.x)	The Fortify ScanCentral DAST Assessment task submits a dynamic scan request to Fortify ScanCentral DAST as a build step. You can view the scan results in Fortify Software Security Center.	Getting Started with Fortify ScanCentral DAST
Fortify WebInspect Dynamic Assessment (7.x) The "Fortify WebInspect Dynamic Assessment" task no longer exists. Removed in 9.5	The Fortify WebInspect Dynamic Assessment task automatically submits a dynamic scan request to Fortify WebInspect as a build step. Fortify WebInspect scans your Web application or Web services for vulnerabilities based on the settings specified in the Scan Settings file.	Getting Started with Fortify WebInspect

1.3. Getting started with Fortify Static Code Analyzer

To configure the Fortify Azure DevOps Extension to use Fortify Static Code Analyzer, you must have experience using Fortify Static Code Analyzer in a standalone environment. For detailed information about how to use Fortify Static Code Analyzer, see *OpenText™ Fortify Static Code Analyzer User Guide* in Fortify Static Code Analyzer and Tools Documentation.

This section contains the following topics:

- Requirements for Fortify Static Code Analyzer tasks
- Installing Fortify Static Code Analyzer
- Using the Fortify Static Code Analyzer Install task
- Adding a Fortify Static Code Analyzer Assessment task
- Troubleshooting the Fortify Static Code Analyzer Assessment task

1.3.1. Requirements for Fortify Static Code Analyzer tasks

Make sure that you have the following information needed to configure the Fortify Static Code Analyzer installation and complete the preparation steps before you run a scan on your application:

- A Fortify license file (`fortify.license`)
- To run Fortify scans in your build definitions, you must first set up a build agent pool of agents that are configured with all the prerequisites to build the application.

  To prepare an agent for the analysis, install the required build software based on your target application's source code, and then confirm that you can successfully build your application on the agent.

  > **Note**
  >
  > The Fortify Static Code Analyzer tasks are not supported on Microsoft-hosted agents. OpenText recommends a minimum of 16 GB of RAM and a quad-core processor to run Fortify Static Code Analyzer.

- To scan .NET projects, the agent must have a full installation of Visual Studio and devenv included in the path environment variable or a supported version of .NET SDK and .NET Framework. For more information, see Fortify Software System Requirements.

  One way to do this is to launch the Developer Command Prompt and run the agent's `configureAgent` or `runAgent` scripts to connect to Azure DevOps.

- You can perform the scan phase on the local agent or remotely using Fortify ScanCentral SAST. To run a scan with Fortify ScanCentral SAST, you must have the following:
  - A Fortify Software Security Center server that is configured to integrate with ScanCentral SAST Controller
  - A Fortify Software Security Center authentication token of type CIToken
- To trigger a build failure based on scan results produced with Fortify ScanCentral SAST, you must use Fortify ScanCentral SAST version 22.1.0 or later (see Adding a Fortify Static Code Analyzer Scan as a Build Step).
- To upload the scan results to Fortify Software Security Center, you must have a Fortify Software Security Center authentication token of type CIToken.
- To perform the scan using Fortify ScanCentral SAST and to upload scan results to Fortify Software Security Center, you need to set up an Azure DevOps service connection to Fortify Software Security Center.

  Create a **Generic** service connection and provide the Fortify Software Security Center server URL and the encoded value of a Fortify Software Security Center authentication token of type CIToken. Leave the **username** box empty.

1.3.2. Installing Fortify Static Code Analyzer

To install Fortify Static Code Analyzer, you have the following two options:

- Using the SCA Install Task

  This installs Fortify Static Code Analyzer with built-in defaults.

- Use the Fortify Static Code Analyzer installer manually on your agent machines.

  This option gives you more control over your installation. For installation instructions, see the *OpenText™ Fortify Static Code Analyzer User Guide* in Fortify Static Code Analyzer and Tools Documentation.

1.3.3. Using the Fortify Static Code Analyzer Install task

The Fortify Static Code Analyzer **Install** task automatically installs and configures Fortify Static Code Analyzer on the target agents.

Perform this install task one time for each agent (or when you upgrade to a new version of Fortify Static Code Analyzer). OpenText recommends that you create a build definition dedicated to setting up agents. You must target this build step to each agent you plan to enable in your build pool.

Before you use the **Fortify Static Code Analyzer Install** task:

- Make sure that you can successfully build your application on the agent where you are installing Fortify Static Code Analyzer.
- You must have both the Fortify Static Code Analyzer installer executable and the `fortify.license` file available using an addressable file path on the agent.
- Make sure that the agent's work directory is close to the root to avoid issues with the Windows maximum path length limitation (MAX_PATH).

This task can:

- Install Fortify Static Code Analyzer unless it is already installed.
- Configure the installation with a user-provided `fortify.license` file.
- Automatically download the Fortify ScanCentral SAST client from the Controller.
- Install the latest Fortify Security Content allowed by the Fortify license.

To configure the Fortify Static Code Analyzer install task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Find and add the Fortify Static Code Analyzer **Install** task.
4. Provide the information described in the following table.

Field	**Description**
Display name	Type a name for the task.
Fortify SCA installer path	Type the full path to the Fortify Static Code Analyzer installer on the agent. For example, `C:\<location_on_agent>\OpenText_SAST_Fortify_windows-x64_<version>.exe`.
Fortify SCA license file	Type the full path to the `fortify.license` file on the agent. For example, `C:\`*`<location_on_agent>`*`\fortify.license`.
Update Fortify Security Content	(Optional) Select whether to update the Fortify Security Content.
Proxy host	(Optional) Specifies a proxy host required for connection to the Fortify Rulepack update server.
Proxy port	(Optional) Specifies a proxy port required for connection to the Fortify Rulepack update server.

1.3.4. Adding a Fortify Static Code Analyzer Assessment task

Use the **Fortify Static Code Analyzer Assessment** task to run Fortify Static Code Analyzer as a build step. After you run the build and the scan is complete, the scan results are available as a Fortify Project Results (FPR) file. You can publish the FPR and Fortify Static Code Analyzer log files as build artifacts. To review the scan results, download the FPR artifact and open it in either Fortify Audit Workbench or Fortify Software Security Center. You can also configure the task to upload the FPR to an existing Fortify Software Security Center server for enterprise vulnerability management.

To configure a Fortify Static Code Analyzer Assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Add the **Fortify Static Code Analyzer Assessment** task.
4. Provide the general information described in the following table.

Field	**Description**
Display name	Type a name for the task.
Fortify SCA license file	(Optional) Provide the path to a Fortify license file. If specified, it overwrites the `fortify.license` file on the build agent where Fortify Static Code Analyzer is currently installed. This path must be the location of a Fortify license file that is different than where Fortify Static Code Analyzer is already installed. **Note** The user running the agent should have the proper permission to write to the Fortify Static Code Analyzer installation directory.
Build ID for Fortify SCA	Type a unique identifier for the scan.
Update Fortify Security Content	(Optional) Select whether to update your installed Fortify Security Content by downloading the latest Fortify Secure Coding Rulepacks and metadata from the Fortify Rulepack update server.
Run SCA clean	(Optional) Select whether to remove any temporary files from a previous scan for the specified build ID.
Enable verbose logging	(Optional) Select whether to send verbose status messages to the console and to the log file.
Enable debug logging	(Optional) Select whether to include debug information in the log file, which is useful for Customer Support to help troubleshoot issues.

5. To run translation, configure the following settings under **Translation Options**:
    1. Select the **Run Fortify SCA translation** check box.

    2. From the **Application type** list, select the type of project you want to analyze.

       The configuration settings dynamically change based on your selection.

    3. Specify the information required to translate the application.

Application Type	Description
.NET	In the **Projects for Fortify SCA analysis** box, type the relative path to the solution or project file name.
Java	Specify the classpath, source version, sourcepath, source files, build tool options, source files (this can be a build file), and any other additional files to include in the scan.
Other	Specify any build tool options, source files, and any other additional files to include in the scan.

    4. (Optional) In the **Additional Fortify SCA translation options** box, specify any additional Fortify Static Code Analyzer translation options. For example, the following option excludes test files from the translation:

```
-exclude **tests/**
```

See *OpenText™ Fortify Static Code Analyzer User Guide* in Fortify Static Code Analyzer and Tools Documentation for more information about translation options.

6. To run a scan, configure the following settings under **Scan Options**:

    1. Select the **Run Fortify SCA scan** check box.
    2. From the **Scan type** list, select whether you want to perform a local scan or a remote scan using Fortify ScanCentral SAST.
    3. (Optional) In the **Additional Fortify SCA scan options** box, specify any additional scan options.
    4. (Optional) In the **Custom Rulepacks** box, specify custom rules.

       Specify custom rules files (*.xml or *.bin) separated by spaces or specify a directory that contains custom rules.

    5. If you selected a scan type of **ScanCentral** in step b, then in the **Fortify SSC service connection** box, specify an Azure DevOps service connection to Fortify Software Security Center. For more information, see Requirements for Fortify Static Code Analyzer Tasks.
    6. To upload the scan results to Fortify Software Security Center, do the following:

        1. Select the **Upload results to SSC** check box.
        2. If you have not already done so, in the **Fortify SSC service connection** box, specify an Azure DevOps service connection to Fortify Software Security Center. For more information, see Requirements for Fortify Static Code Analyzer Tasks.
        3. Specify an application version that exists in Fortify Software Security Center by providing one of the following:

            - An application name and an application version name.
            - A Fortify Software Security Center application version ID.

> **Note**
>
> If you provide both application name and version and an application ID, the extension uses the application ID for the upload regardless of the selected application version type.

4. (Optional) To connect to Fortify Software Security Center with a proxy server, specify the proxy information.

   > **Note**
   >
   > Use the following syntax for the **Proxy URL**:
   > `<protocol>://<address>:<port>`

5. (Optional) To trigger a build failure based on the scan results, type a search query in the **Build failure criteria** box.

   For example, the following search query causes the build to fail if any critical issues exist in the scan results:

   ```
   [fortify priority order]:critical
   ```

   See OpenText™ Fortify Software Security Center User Guide in Fortify Software Security Center Documentation for a description of the search query syntax.

   By default, the task returns a warning when the build failure criteria is met. To fail the build instead, select **FAIL** from the **Task results when build failure criteria is met** list.

6. (Optional) To specify how long to poll Fortify Software Security Center to determine if FPR processing is finished, type the time in minutes in the **Polling timeout** box.

   If no value or a value of 0 is specified, polling continues until FPR processing finishes or stops due to errors. The valid values are 0–10080.

7. (Optional) To specify how frequently to poll Fortify Software Security Center to determine if the FPR processing is finished, in the **Polling interval** box, specify an interval (in minutes).

   The valid values are 1–60 and the default value is 1 minute.

   > **Important**
   >
   > If the FPR processing requires approval, then this step will not complete until approval is granted through Fortify Software Security Center.

   As an alternative to uploading scan results to Fortify Software Security Center, you can add a standard Azure DevOps **Publish Pipeline Artifact** build step to collect the scan results and log files.

> **Note**
>
> To ensure that you obtain scan log files when you publish artifacts, make sure that you select the **Continue on error** check box in the task configuration. Otherwise, if the assessment fails, the artifact collection task does not start.

1.3.5. Troubleshooting the Fortify Static Code Analyzer Assessment task

Unable to Find sourceanalyzer

The agent running the scan must have the location of Fortify Static Code Analyzer included in the execution path. By default, the Fortify Static Code Analyzer installer adds itself to the path.

If you see this error, make sure that the Fortify Static Code Analyzer installation location is part of the OS execution path. You might need to restart your agent to pick up changes made to the OS path.

Unable to Connect to Fortify Software Security Center for Upload

- Make sure that your application name, version name, and service connection are correctly configured.
- If your Fortify Software Security Center is configured to use HTTPS, make sure that the JDK keystore in the Fortify Static Code Analyzer installation is configured to accept the Fortify Software Security Center server certificate.

1.4. Getting started with Fortify on Demand

A Fortify on Demand account is required to use the Fortify Azure DevOps Extension with Fortify on Demand.

This section contains the following topics:

- Adding Fortify on Demand credentials in Azure DevOps
- Setting up a Fortify on Demand Static Assessment
- Setting up a DAST Automated assessment
- Setting Up a Fortify on Demand Dynamic Assessment
- Troubleshooting Fortify on Demand tasks

1.4.1. Adding Fortify on Demand credentials in Azure DevOps

Before adding a Fortify on Demand task to your pipeline, you need to obtain appropriate Fortify on Demand credentials and add them in Azure DevOps. Service connections are used to manage Fortify on Demand credentials in Azure DevOps. You can create a Fortify service connection to store Fortify on Demand credentials.

To add your Fortify on Demand credentials in Azure DevOps:

1. In an Azure DevOps project, navigate to the project settings .
2. Under **Pipelines**, select **Service connections**.
3. Click **New service connection**.
4. Select **Fortify** from the list and click **Next**.

   The Add Fortify service connection window appears.

5. Select the method of authentication:

   - **Basic Authentication**: requires personal access token with the `api-tenant` scope
   - **Token Based Authentication**: requires API key with the `api-tenant` scope

   See the Fortify on Demand documentation for instructions on creating a personal access token and API key.

6. Complete the following fields:

Field	Description
Connection name	Specify a name for your service connection.
API URL	Specify your data center's API root URL: ◦ US: https://api.ams.fortify.com ◦ EMEA: https://api.emea.fortify.com ◦ APAC: https://api.apac.fortify.com ◦ SGP: https://api.sgp.fortify.com ◦ FedRAMP: https://api.fed.fortify.com ◦ Trial: https://api.trial.fortify.com
Portal URL	Specify your data center's domain URL.
Proxy Host (optional)	Specify the URL of the proxy server.
Proxy Port (optional)	Specify the port of the proxy server.
Username, Personal Access Token, Tenant ID API Key, API Secret	◦ If you selected **Basic Authentication**, specify the account username, personal access token, and tenant code. ◦ If you selected **Token Based Authentication**, specify the API key and secret.

7. Click **OK**.

   Your new service connection is saved.

1.4.2. Setting up a Fortify on Demand Static Assessment

Perform the following tasks to set up a Fortify on Demand Static Assessment:

- Download and install the Fortify ScanCentral SAST client on the agent. See Downloading and Installing the Fortify ScanCentral SAST Client. This part is optional if you are using a Microsoft-hosted agent.
- Configure static scan settings. You can configure scan settings from the Fortify on Demand portal before submitting the assessment or from Azure DevOps as part of the task settings.
- Add the Fortify on Demand Static Assessment task to a pipeline in an Azure DevOps project. See Adding a Static Assessment Task.

1.4.2.1. Downloading and installing the Fortify ScanCentral SAST client

A stand-alone Fortify ScanCentral SAST client is offered for automatically packaging all necessary dependencies and source code required for static scanning and the files required for Debricked open source scanning. The following languages are supported: .NET and .NET Core (MSBuild projects), Apex, Classic ASP, ColdFusion, Dockerfiles, Go, Java (Gradle and Maven projects), Javascript/Typescript, PHP, Python, and Ruby.

The latest version of the Fortify ScanCentral SAST client is available from the Tools page in the portal. Installation instructions are available in the README.txt file stored in the zip file.

Important

The stand-alone Fortify ScanCentral SAST client is a component of the on-premises Fortify ScanCentral SAST software and is used to package code to send to a Controller for scanning. Fortify Azure DevOps Extension uses only the packaging feature of the Fortify ScanCentral SAST client. Details that are relevant to packaging your source code has been provided.

For more information about using the Fortify ScanCentral SAST client, see the Fortify Software Security Center Documentation. Select the documentation version that corresponds to your installed version.

- Software requirements: "Fortify ScanCentral SAST Client Software Requirements" in *Fortify Software System Requirements*
- Supported build tools: "Fortify ScanCentral SAST Sensor Languages and Build Tools" in *Fortify Software System Requirements*
- Command-line options: "Package Command" in *Fortify ScanCentral SAST Installation, Configuration, and Usage Guide*

1.4.2.2. Adding a Static Assessment task

You can add the **Fortify on Demand Static Assessment** task to your build pipeline using the classic editor or the YAML editor in Azure DevOps. The following instructions describe how to add a static assessment task to a build pipeline using the YAML editor.

> **Note**
>
> The **Fortify on Demand Static Assessment** task does not support release pipelines.

To add a static assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Find and select **Fortify on Demand Static Assessment** from the task list.

   The static assessment task settings appear.

4. Complete the following fields:

Field	**Description**
Source code location	Specify the path on the agent where the source code files are located. You can use predefined variables for the source code directory, such as `$(Build.SourcesDirectory)`. Do not use `$(Build.ArtifactStagingDirectory)` or `$(Build.ArtifactDirectory)`, as these locations can cause errors when compressing the source code prior to transmission.
ScanCentral file location	Specify the path on the agent where the Fortify ScanCentral SAST client executable is located. For example, `C:\Program Files\Fortify_ScanCentral_Client_21.1.0_x64\bin`. If the field is left empty, the latest version of the Fortify ScanCentral SAST client will automatically be downloaded on the agent. **Note** The Fortify ScanCentral SAST version and the installed Java version must be compatible. If the Java version is incompatible, the task will fail.
Fortify Connection	Select an existing service connection or click **+New** to add a new service connection. For more information, see Adding Fortify on Demand credentials in Azure DevOps.

5. In the **Application/Release Options** section, select the method of identifying the release from the **Pick a Release** list:
   - **Release ID**
   - **BSI Token**
   - **New Application and Release**
6. Follow the procedure for the selected method:

Method	**Procedure**

Release Id	In the **Release ID** field, specify the release ID. **Note** The release must have saved scan settings in the portal in order for the release ID to be used as a token.
BSI token	In the **Build Server Integration Token** field, specify the BSI token.
New Application and Release	Complete the following fields to create an application and/or release: ◦ **Application Name**: specify the application name. If a unique value is provided, an application will be created. **Note** If you are working with an existing application, updates to application settings will be applied where applicable. ◦ **Business Criticality**: select the business criticality. ◦ **Application Attributes**: specify required and optional application attributes as `<attributeName1>: <attributeValue1>; <attributeName2>: <attributeValue2>; ...` ◦ **Application Type** (not applicable to existing applications): select the application type. ◦ **Microservice Application** (not applicable to existing applications): select the check box to scan the application as a microservice application. The microservice feature must be enabled for the tenant. ◦ **Microservice Name**: If the application consists of microservices, specify the microservice name. If a unique value is provided, a microservice will be created. **Note** An application can have a maximum of 10 microservices. ◦ **Release Name**: specify the release name. A unique value must be provided. ◦ **SDLC Status**: select the SDLC status. ◦ **Owner ID**: specify the owner ID.

7. In the **Entitlement Options** section, complete the following fields:

Field	**Description**
Entitlement Options	Select the method of determining the entitlement to use: ◦ **User-selected entitlement**: the user specifies the entitlement. Provide the entitlement ID in the **Entitlement ID** field. ◦ **Auto-selected entitlement**: Fortify on Demand determines the entitlement. 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.
Entitlement Preference	Select the entitlement preference.

Purchase Entitlements	(Optional, available for **Auto-selected entitlement**) Select the check box to purchase an entitlement if none is available for the specified entitlement preference. The purchase entitlements feature must be enabled for the tenant.

8. In the **Scan Options** section, complete the following fields:

Note

Updates to scan settings are retained for subsequent scans.

Field	Description
Choose Scan Settings Source	Select the method of specifying the scan settings: ◦ **Create/Override Existing Scan Settings if any** (required if you are creating a release) Complete the following fields: ▪ **Assessment Type Id**: specify the assessment type ID ▪ **Audit Preference**: select the audit preference ◦ **Use Existing Saved Scan Settings**
Action if Scan In Progress	If the release has an in progress scan, select the action to take: ◦ **Do Not Start Scan**: do not start a new scan and fail the task ◦ **Cancel Scan In Progress**: cancel the scan in progress and start a new scan (if the scan in progress scan can be automatically canceled) ◦ **Queue**: queue the scan (if the scan queue limit has been reached, the scan will be canceled)
Remediation Preference Type	Select whether to run a remediation scan.
Build Type	Select the method of packaging the application files. All selections except for **None** invoke the Fortify ScanCentral SAST client to package the application files.

9. Follow the procedure for the selected build type:

Field	Procedure
Go (ScanCentral)	**Open Source Component Analysis**: select the check box to include open source component analysis.[1]

Maven, Gradle	Complete the following fields: ◦ **Technology Stack**: select the technology stack.[2] ◦ **Language Level**: select the language level.[2] ◦ **Open Source Component Analysis**: select the check box to include open source component analysis.[1,2] ◦ **Build Command**: (Optional) specify custom build parameters for preparing and building a project. For example, to invoke a Gradle build before packaging: `-Prelease=true clean customTask build` ◦ **Build File**: (Optional) specify the path on the agent where the build file is if you are not using a default name such as `build.gradle` or `pom.xml`. For example, `myCustomBuild.gradle` ◦ **Include Tests**: (Optional) select the check box to include the test source set (Gradle) or a test scope (Maven) with the scan. ◦ **Skip Build**: (Optional) select the check box to disable the project preparation build step before packaging.
DotNet, MSBuild	**Important** Packaging using MSBuild is only available on Windows agents. The MSBuild executable must be added to the PATH environment variable. You can set the environment variable by running the Batch Script task before the Static Assessment task. Set `filename` to the path of `VsDevCmd.bat` and `modifyEnvironment` to `true`. For detailed instructions on configuring the Batch Script task, see https://docs.microsoft.com/en-us/azure/devops/pipelines/tasks/utility/batch-script?view=azure-devops. If you are using a Microsoft-hosted agent, see https://docs.microsoft.com/en-us/azure/devops/pipelines/agents/hosted?view=azure-devops&tabs=yaml to determine the path of `VsDevCmd.bat`. For example, for the Windows Server 2019 with Visual Studio 2019 agent, the path is `C:\Program Files (x86)\Microsoft Visual Studio\2019\Enterprise\Common7\Tools\VsDevCmd.bat`. Complete the following fields: ◦ **Technology Stack**: select the technology stack.[2] ◦ **Language Level**: select the language level.[2] ◦ **Open Source Component Analysis**: select the check box to include open source component analysis.[1,2] ◦ **Build Command**: (Optional) specify custom build parameters for preparing and building a project. ◦ **Build File**: specify the path on the agent where the build file is located. For example, `mySolution.sln`. ◦ **Skip Build**: (Optional, MSBuild only) select the check box to disable the project preparation build step before packaging. **Note** **Skip Build** is not supported in Fortify ScanCentral SAST versions 21.1.2 and later.
PHP (ScanCentral)	**Open Source Component Analysis**: select the check box to include open source component analysis.[1]

Python	Complete the following fields: ◦ **Python Version**: select the language level.[2] ◦ **Open Source Component Analysis**: select the check box to include open source component analysis.[1,2] ◦ **Python Virtual Environment**: Specify the Python virtual environment location. ◦ **Python Requirements File**: specify the Python project requirements file to install and collect dependencies.
None	Complete the following fields: ◦ **Technology Stack**: select the technology stack.[2] ◦ **Language Level**: if applicable, select the language level.[2] ◦ **Open Source Component Analysis**: select the check box to include open source component analysis.[1,2]

1. If your tenant has Debricked entitlements, OpenText recommends using version 22.1.2 or later of the Fortify ScanCentral SAST client, which packages the files required for a Debricked open source scan. Otherwise, manually generate the files and include them in the payload. For instructions on generating these files, see the Fortify on Demand documentation.

2. Available if you are configuring scan settings.

10. In the **Poll Options** section, complete the following fields:

Field	**Description**
Polling Interval	Specify the length of time in minutes between polling for static and open source scan statuses and results. The default value is 1. A value of 0 disables polling. **Note** Polling stops once either the static or open source scan is canceled, paused, or completed.
Action if Failing Policy	Select whether to complete the task and throw a warning or fail the task based on the application security policy set by your organization.

11. Click **Add**.

    The YAML code for the task is added to your pipeline. The YAML code by default specifies the latest version of the extension.

12. Save the settings.

If a scan is successfully submitted during the pipeline run, the task will be marked as succeeded. If the scan is rejected, the build logs will display the appropriate error message.

1.4.3. Setting up a DAST Automated assessment

Perform the following tasks to set up a DAST Automated assessment:

- Prepare your web application. See Preparing a web application for DAST Automated assessment.
- Configure dynamic scan settings. You can configure scan settings from the Fortify on Demand portal before submitting the assessment or from Azure DevOps as part of the task settings.
- Add the **FoD DAST Automated** task to a pipeline in an Azure DevOps project. See Adding a DAST Automated assessment task.

1.4.3.1. Preparing a web application for DAST Automated assessment

The first step in a DAST Automated assessment is to prepare your web application. For instructions on preparing the web application, see the Fortify on Demand documentation.

1.4.3.2. Adding a DAST Automated assessment task

You can add the **FoD DAST Automated** assessment task to your pipeline using the classic editor or YAML editor in Azure DevOps. The following instructions describe how to add a DAST Automated assessment to a build pipeline through the YAML editor.

> **Note**
>
> You can use the classic editor or YAML editor to define build pipelines; use the classic editor to define release pipelines.

To add a DAST Automated assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Select **FoD DAST Automated** from the list.

   The DAST Automated assessment task settings appear.

4. Complete the following fields:

Field	**Description**
Fortify Connection	Select an existing service connection or click **+New** to add a new service connection. For more information, see Adding Fortify on DemandCredentials in Azure DevOps.

5. In the **Application/Release Options** section, select the method of identifying the release from the **Pick a Release** list:
   - **Release ID**
   - **New Application and Release**
6. Follow the procedure for the selected method:

Method	**Procedure**
Release Id	In the **Release ID** field, specify the release ID. **Note** The release must have saved scan settings in the portal in order for the release ID to be used as a token.

New Application and Release	Complete the following fields to create an application and/or release: ◦ **Application Name**: specify the application name. If a unique value is provided, an application will be created. **Note** If you are working with an existing application, updates to application settings will be applied where applicable. ◦ **Business Criticality**: select the business criticality. ◦ **Application Attributes**: specify required and optional application attributes as `<attributeName1>: <attributeValue1>; <attributeName2>: <attributeValue2>; ...` ◦ **Application Type** (not applicable to existing applications): select the application type. ◦ **Microservice Application** (not applicable to existing applications): select the check box to scan the application as a microservice application. The microservice feature must be enabled for the tenant. ◦ **Microservice Name**: If the application consists of microservices, specify the microservice name. If a unique value is provided, a microservice will be created. **Note** An application can have a maximum of 10 microservices. ◦ **Release Name**: specify the release name. A unique value must be provided. ◦ **SDLC Status**: select the SDLC status. ◦ **Owner ID**: specify the owner ID.

7. In the **Entitlement Options** section, complete the following fields:

Field	Description
AssessmentType Id	Specify the DAST Automated assessment type ID.
Entitlement ID	Specify the entitlement ID that the assessment will use.
Entitlement Frequency	Specify the entitlement frequency: **Single Scan**, **Subscription**. Note that microservice applications are restricted to subscriptions.

8. In the **Scan Options** section, complete the following fields:

Field	Description

Choose Scan Settings Source	Select how scan settings are specified: ◦ **Create/Override Existing Scan Settings if any** (required if you are creating a release) **Note** Updates to scan settings are retained for subsequent scans. ◦ **Use Existing Saved Scan Settings**
Scan Type	Select the dynamic scan type: ◦ **Website**: this scan is similar to a Dynamic Website scan. ◦ **Workflow Driven**: this scan is similar to a Dynamic Website scan that utilizes a workflow macro. ◦ **API**: this scan is similar to a Dynamic API scan.

9. If you selected **Create/Override Existing Scan Settings if any**, complete the following fields. Otherwise, skip to the next step. Fields are not described in the order of presentation in the UI.

Scan type	**Field**	**Description**
All scan types	Environment Facing	Select whether the site is internal or external.
All scan types	Time Zone	Select your location's time zone, which is used to schedule the scan's start time.
All scan types	Request False Positive Removal (optional)	Select the check box to request false positive removal by the testing team once per application. **Important** Login macro generation and false positive removal are an optional service that is available once per application and consumes 1 additional assessment unit. **Important** If you want to request both login macro generation and false positive removal, you must select both options together; once a scan that includes either option has completed, both options will be disabled for subsequent scans.

API	API Type	Select the API definition type: **Postman**, **OpenApi**, **Graph QL**, **GRPC**. Perform the relevant task based on your API definition type: **Note** OpenAPI Specification versions 2.0 and 3.0 are supported.
		Postman Specify the file ID of the uploaded file in the **Postman Collection** field.
		OpenAPI Select **File** or **URL to the OpenAPI specification** and perform the relevant task based on your selection. ○ **File** 1. Specify the file ID of the uploaded file in the **OpenApi Json File** field. 2. If the API requires authentication, provide the API key value in the **API Key** field. **Note** The supported security scheme is API key. Multiple API keys in requests are not supported. ○ **URL to the OpenAPI specification** 1. Specify the OpenAPI document URL in the **OpenApi Url** field. 2. If the API requires authentication, provide the API key value in the **API Key** field. **Note** The supported security scheme is API key. Multiple API keys in requests are not supported.

		GraphQL Select **File** or **URL** and perform the relevant task based on your selection. ○ **File** 1. Specify the file ID of the uploaded file in the **GraphQL Json File** field. 2. Select the API scheme in the **API Scheme Type** field: **HTTP**, **HTTPS**, **HTTP and HTTPS**. 3. Specify the URL or hostname In the **API Host** field. 4. Specify the directory path for the API service in the **API Service Path** field. ○ **URL** 1. Provide the GraphQL introspection endpoint URL in the **GraphQL Url** field. 2. Select the API scheme in the **API Scheme Type** field: **HTTP**, **HTTPS**, **HTTP and HTTPS**. 3. Specify the URL or hostname In the **API Host** field. 4. Specify the directory path for the API service in the **API Service Path** field. **Note** The GraphQL API must have introspection enabled to download the schema contents for the scan.
		gRPC 1. Specify the file ID of the uploaded file in the **GRPC Proto File** field. 2. Select the API scheme in the **Scheme Type** field: **HTTP**, **HTTPS**, **HTTP and HTTPS**. 3. Specify the URL or hostname In the **API Host** field. 4. Specify the directory path for the API service in the **API Service Path** field.
Website	Dynamic Site URL	Provide your site's URL.

Website	Scope	Select one of the following options: ○ **Scan entire host (*<URL>*)** (default): the entire host will be scanned **Example**: Given https://foo.com/home, the following URLs will be included: ▪ https://foo.com/ ▪ https://foo.com/contact-us.html ▪ https://foo.com/folder/ ▪ https://foo.com/folder/folder2/page.aspx ▪ https://foo.com/home/folder/ ▪ https://foo.com/home/index.html ○ **Restrict scan to a URL or sub folder**: only the directory denoted by the last slash in the URL and its subdirectories will be scanned. **If you select this option, make sure the last slash denotes the directory to which you want the scan to be restricted.** **Example**: Given https://foo.com/home/, the following URLs will be excluded: ▪ https://foo.com/ ▪ https://foo.com/folder/ ▪ https://foo.com/contact-us.html ▪ https://foo.com/folder/folder2/page.aspx
Website	Redundant Page Direction (optional)	Select the check box to enable comparison of page structure to determine the level of similarity, allowing the sensor to identify and exclude processing of redundant resources. **Important** Redundant page detection works in the crawl portion of the scan. If the audit introduces a session that would be redundant, the session will not be excluded from the scan.
Website	Exclude URLs (optional)	Specify full or partial URLs to exclude URLs matching the strings as `<Url1>; <Url2>; <Url3>; ...`The field is not case-sensitive.

Options depend on scan type	Scan Policy	Select the policy (collection of vulnerability checks and attack methodologies that the sensor deploys against a Web application): **Standard**: A standard scan includes an automated crawl of the server and performs checks for known and unknown vulnerabilities such as SQL Injection and Cross-Site Scripting as well as poor error handling and weak SSL configuration at the web server, web application server, and web application layers. **Criticals and Highs**: Use the Criticals and Highs policy to quickly scan your web applications for the most urgent and pressing vulnerabilities while not endangering production servers. This policy checks for SQL Injection, Cross-Site Scripting, and other critical and high severity vulnerabilities. It does not contain checks that may write data to databases or create denial-of-service conditions, and is safe to run against production servers. **Passive Scan**: The Passive Scan policy scans an application for vulnerabilities detectable without active exploitation, making it safe to run against production servers. Vulnerabilities detected by this policy include issues of path disclosure, error messages, and others of a similar nature. **API**: The API policy contains checks that target various issues relevant to an API security assessment. This includes various injection attacks, transport layer security, and privacy violation, but does not include checks to detect client-side issues and attack surface discovery such as directory enumeration or backup file search checks. All vulnerabilities detected by this policy may be directly targeted by an attacker. This policy is not intended for scanning applications that consume Web APIs.
Website API	Timebox Scan Duration (Hours)	Specify the maximum duration of the scan. If the scan is not completed at the end of the specified duration, the scan is terminated and partial results are available. If the scan is completed during the specified duration, then complete results are available. Incremental scanning is not supported.
All scan types	Network Authentication (optional)	Select the check box if network authentication is required. Provide the authentication type, username, and password. **Note** The scan will be canceled if network authentication fails.

Website	Site Authentication (optional)	Select the check box if site authentication is required. Specify the file ID of the uploaded file in the **Login Macro File for Site Authentication** field. **Note** Make preparations so that the user credentials remain valid for the scan duration, such as increasing the password expiration duration. The scan will be canceled if site authentication fails.
Website	RequestForLoginMacroCreation(optional)	Select the check box to request generation of a login macro by the testing team once per application. Upon scan completion, the login macro will be available for download on the Scans page. **Important** Login macro generation and false positive removal are an optional service that is available once per application and consumes 1 additional assessment unit. **Important** If you want to request both login macro generation and false positive removal, you must select both options together; once a scan that includes either option has completed, both options will be disabled for subsequent scans.

10. In the **Poll Options** section, complete the following fields:

Field	Description
Polling Interval	Specify the length of time in minutes between polling for static and open source scan statuses and results. The default value is 1. A value of 0 disables polling. **Note** Polling stops once either the static or open source scan is canceled, paused, or completed.
Action if Failing Policy	Select whether to complete the task and throw a warning or fail the task based on the application security policy set by your organization.

11. Click **Add**.

The YAML code for the task is added to your build pipeline. The YAML code specifies the latest version of the extension.

12. Save the settings.

If a scan is successfully submitted during the pipeline run, the task will be marked as succeeded. If the scan is rejected, the build logs will display the appropriate error message.

1.4.4. Setting Up a Fortify on Demand Dynamic Assessment

Perform the following tasks to set up a Fortify on Demand dynamic assessment:

- In Fortify on Demand, configure dynamic scan settings. See Setting Up a Dynamic Scan.
- In an Azure DevOps project, configure a Fortify on Demand dynamic assessment task. See Configuring a Dynamic Assessment Task.

1.4.4.1. Configuring a Dynamic Scan

After preparing your website for a dynamic assessment, you need to complete the Dynamic Scan Setup page. You only need to configure the dynamic scan settings once per release as the settings are carried over to the next scan. You can edit settings as needed for subsequent assessments.

To configure a dynamic scan:

1. Select the Applications view.

   Your Applications page appears.

2. Click the name of the application.

   The Application Overview page appears.

3. Click **Start Scan** for the release that you want to have assessed and select **Dynamic**.

   The Dynamic Scan Setup page appears.

4. Complete the required fields. All other fields are optional or set to default values.

Field	Description
Assessment Type	Select the assessment type. Only assessment types allowed by the organization's security policy are displayed. The SLO of the selected assessment type appears below the field. **Note** The **Dynamic+ Web Services** assessment is used for testing web services where an OpenAPI definition or Postman collection is not available.
Dynamic Site URL	Type your site's URL. This field is available for Dynamic Website, Dynamic+ Website, and Dynamic+ Web Services assessments.
Entitlement	Select the entitlement that the assessment will use. The field displays entitlements that are valid for the selected assessment type, including those available for purchase. If the release has an active subscription, only options that do not consume entitlements are displayed.
Time Zone	Select your location's time zone, which is used to schedule the scan's start time.
Environment Facing	Select whether the site is internal or external.

5. If needed, you can configure additional scan settings in the sections appearing below the required fields. The sections that are available depend on the assessment type selected.

   Scope (Dynamic Website, Dynamic+ Website, Dynamic+ Web Services)

   1. To edit the scope of the scan, click **Scope**.

2. Complete the fields as needed.

Field	Description
Scan entire host (*<URL>*)	Select one of the following options: - **Scan entire host (*<URL>*)** (default): the entire host will be scanned **Example**: Given https://foo.com/home, the following URLs will be included: - https://foo.com/ - https://foo.com/contact-us.html - https://foo.com/folder/ - https://foo.com/folder/folder2/page.aspx - https://foo.com/home/folder/ - https://foo.com/home/index.html - **Restrict the scan to the URL directory and subdirectories**: only the directory denoted by the last slash in the URL and its subdirectories will be scanned. **If you select this option, make sure the last slash denotes the directory to which you want the scan to be restricted.** **Example**: Given https://foo.com/home/, the following URLs will be excluded: - https://foo.com/ - https://foo.com/folder/ - https://foo.com/contact-us.html - https://foo.com/folder/folder2/page.aspx
Restrict scan to URL directory and subdirectories	
Allow HTTP (:80) and HTTPS (:443)	Select the check box to allow both HTTP and HTTPS scanning of the site (default). **Example**: Given https://foo.com/home, if the **Scan entire host** option is selected, http://foo.com/ and its subdirectories will be included. If the **Restrict scan to URL directory and subdirectories** option is selected, only http://foo.com/home and its subdirectories will be included.
Allow form submissions during crawl	Select this option to allow form submissions during the crawl of the site (default). This uncovers additional application surface area that can then be examined for a more thorough scan. Deselecting this option does **not** prevent form submissions during the vulnerability checks. Detection of many critical vulnerabilities, such as SQL injection and cross-site scripting, requires form submissions. To exclude specific web functionalities from form submissions, specify those URLS in the **Exclude URLS that contain** field.
Exclude URLS that contain	(Optional) Type a full or partial URL and click to exclude URLs matching the string from testing. Add a new entry for each string. The field is not case-sensitive. By default, Fortify Azure DevOps Extension the None does not scan URLs outside the provided hostname, such as subdomains (https://www.foo.com, https://dev.foo.com) or offsite domain (https://bar.com). **Example**: https://foo.com/login.html, login.html

(Authentication (Dynamic Website, Dynamic+ Website, Dynamic+ Web Services)

1. To edit the authentication settings, click **Authentication**.

2. Complete the fields as needed.

Field	Description

Form Authentication	(Optional) Select the check box if form authentication is required. Provide user names and passwords for at least two users. To add more credentials, use the **Additional Notes** field at the bottom of this form. If available, select the **Generate unique authentication** check box if self-registration is required.
Network Required	(Optional) Select the check box if network authentication is required and provide a username and password.
Additional Authentication Instructions	(Optional) Select the check box if additional authentication is required, such as an account number or tenant code, and type instructions. **Important** Fortify Azure DevOps Extension The None does not support multi-factor authentication. Examples include authentication controls involving SMS messages, email verifications, CAPTCHA, OATH Tokens, and physical tokens.

Web Services (Dynamic Web Services)

For information on preparing web services project files suitable for automated testing, see Preparing Web Services Project Files.

1. To add instructions for scanning web services utilized by the site, click **Web Services**.
2. Select the API definition type: **Postman Collection (File)**, **Postman Collection (URL)**, **OpenAPI (File)**, **OpenAPI (URL)**.

   Note

   OpenAPI Specification versions 2.0 and 3.0 are supported.

3. Perform the relevant task based on your API definition type:

API Definition Type	**Procedure**
Postman Collection (File)	1. Click **...** and browse to and select the Postman collection file. The JSON file format is accepted. If a file already exists, you can use the existing file or upload a new file.

Postman Collection (URL)	1. Provide the Postman collection URL.
	2. If authentication is needed to access the URL, provide the header name in the **Header Name** and the credentials in **Header Value** fields. For example, provide `Authorization` in **Header Name** and `Bearer <token>` in **Header Value**. Not that this is separate from the credentials used to authenticate requests.
	Examples:
	`X-API-Key: <apikey>`
	`Authorization: <apikey>`
	`Authorization: Bearer <token>`
	Note
	If the credentials are passed as a query parameter, include it in the URL.

OpenAPI (File)	1. Click ... and browse to and select the OpenAPI document file. The JSON file format is accepted. If a file already exists, you can use the existing file or upload a new file. 2. If the API requires authentication, provide the API key value in the **API Key** field. **Note** The supported security scheme is API key. Multiple API keys in requests are not supported.
OpenAPI (URL)	1. Provide the OpenAPI document URL. 2. If the API requires authentication, provide the API key value in the **API Key** field. **Note** The supported security scheme is API key. Multiple API keys in requests are not supported.

4. In the **Additional Instructions** field, type additional instructions.

Web Services (Dynamic+ Web Services)

For information on preparing web services project files suitable for automated testing, see Preparing Web Services Project Files.

1. To add instructions for scanning web services utilized by the site, click **Web Services**.

2. Complete the fields as needed.

Field	**Description**

Web Service Type	1. Select the web service type: **SOAP**, **REST**. 2. Upload a project file, such as a WSDL file or API definition file, that contains working sample data. The JSON, WSDL, TXT, and XML file formats are accepted.
Additional Instructions	(Optional) Type additional instructions, such as required headers, tokens, or authentication mechanisms.
Username, Password API Key, Password	(Optional) Provide the username and password or API key and password.

Scheduling & Availability (all assessments)

1. To edit the scan frequency and site availability settings, click **Scheduling & Availability**.
2. Complete the fields as needed.

Field	Description
Repeat Frequency	Select the scan's repeat frequency: **Do not repeat** (default), **2 weeks**, **1 month**, **2 months**, **3 months**, **4 months**, **6 months**, **12 months**. If you are requesting a single scan, keep the default value. Scheduled recurring scans are automated and subjected to the following stipulations: ▪ Scheduling of a scan occurs seven days before the calculated scan date, which is determined by the start date of the previous scan and the repeat frequency. For example, if a monthly scheduled scan starts on the 5th of the month, the next scan will be scheduled for the 5th of the next month. ▪ The entitlement is deducted at the time of scheduling. ▪ A scan will only be scheduled if a valid entitlement for the selected assessment type exists at the time of the scheduling. ▪ If a scan is canceled, no further scans will be scheduled. ▪ If a scan is still in progress when the next scan is to be scheduled, Fortify Azure DevOps Extension the None will attempt once a day to reschedule the next scan until the scan date has passed. For example, if a monthly scheduled scan that starts on the 5th of the month is still in progress by the 5th of the next month, the next rescheduling attempt will take place seven days before the 5th of the month after that.
Site Availability	Select the check boxes to indicate when the environment is available for testing. Use the local time of the time zone specified above. You must provide a minimum of a four hour window of availability during the week. Pausing and resuming testing causes the scan to take longer than the standard SLO. Contact the support team for more information if you have site availability constraints.

Additional Details (Dynamic Website, Dynamic+ Website, Dynamic+ Web Services)

1. To add additional details about the scan, click **Additional Details**.
2. Complete the fields as needed.

Field	Description

User agent	Select the user agent type that will be used for the site: **Desktop browser** (default), **Mobile browser**
Concurrent request threads	Select the number of concurrent requests that will be used for the scan: ▪ **Standard** (default): 5 crawl requestor threads, 10 audit requestor threads, 20 second request timeout ▪ **Limited**: 2 crawl requestor threads, 3 audit requestor threads, 5 second request timeout Selecting the **Limited** option will reduce the scan load but will also cause the scan to take longer than the standard SLO.
Additional Notes	(Optional) Type additional information that the testing team needs to know before starting the assessment. **Note** Free form exclusions and whitelist notes have been migrated to this field.
Additional Documentation	(Optional) Upload documentation (30 MB limit) that facilitates testing of the application. Uploaded files are displayed in the **Uploaded Files** section below. Supported file types: DOC, DOCX, PPT, TXT, PDF, PPTX, ZIP, XLS, XLSX, CSV.
Generate WAF Virtual Patch	**Note** Contact support to enable the option. (Optional) Select the check box to request a WAF virtual patch from Fortify WebInspect. Once the assessment is complete, you can download the file on the Scans page
Request pre-assessment conference call	(Optional, Dynamic Premium and Dynamic+ assessments only) Select the check box to request a pre-assessment conference call. The check box is cleared after the assessment is completed. **Note** You cannot request a pre-assessment conference call for a scan scheduled within 72 hours.

6. Once you have configured the scan settings, click **Save**.

   ○ If the form is complete, the **Setup Status** is marked as **Valid**.

   ○ If the form is incomplete, the **Setup Status** is marked as **Incomplete**. A list of the issues appears at the top of the page. You can hover over the **x** icon next to **Setup Status** to display the list.

1.4.4.2. Adding a Dynamic Assessment Task

You can add the **Fortify on Demand Dynamic Assessment** task to your pipeline using the classic editor or YAML editor in Azure DevOps. The following instructions describe how to add a dynamic assessment to a build pipeline through the YAML editor.

> **Note**
>
> Build pipelines can be defined using the classic editor or YAML editor; release pipelines can be defined using the classic editor.

To add a dynamic assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Select **Fortify on Demand Dynamic Assessment** from the list.

   The dynamic assessment task settings appear.

4. Complete the following fields:

Field	Description
Display name	Type a name for the task.
The root API Url	Type the API root URL of your Fortify on Demand data server.
Release Id	Type the release ID.

5. In the **Authentication Methods** section, complete the following fields:

Field	Description
API Authentication Type	1. Select the method of authentication: **API Key/Secret** or **Personal Access Token**. 2. Provide the API key and secret or your account username, personal access token, and tenant ID. OpenText recommends using secret build variables to specify the Fortify on Demand credentials.
Proxy host	(Optional) Type the URL of the proxy server.
Proxy port	(Optional) Type the port of the proxy server.

6. In the **Entitlement Options** section, complete the following fields:

Field	Description
Entitlement Preference	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.
Purchase Entitlements	(Optional) Select the check box to purchase an entitlement if none is available for the specified entitlement preference. The purchase entitlements feature must be enabled for the tenant.

Prefer Remediation	Select the check box to run a remediation scan if one is available.

7. Click **Add**.

   The YAML code for the task is added to your build pipeline. The YAML code specifies the latest version of the extension.

8. Save the settings.

If a scan is successfully submitted during the pipeline run, the task will be marked as succeeded. If the scan is rejected, the build logs will display the appropriate error message.

1.4.5. Troubleshooting Fortify on Demand tasks

Task fails with error "SyntaxError: Use of const in strict mode"

Problem: The task fails with the following error:

```
const tl = require('vsts-task-lib/task');
^^^^^
SyntaxError: Use of const in strict mode
```

Cause: The version of `node.exe` in the VSO agent folder is earlier than 5.0. To check the version of `node.exe` installed for the agent, search for "node.exe" in the VSO agent folder, then run `[path to node.exe]\node -v`.

Solution: Manually update the node in the VSO agent folder to version 5.0 or later.

Static Assessment task fails with error "The process 'C:\hostedtoolcache\windows\scancentral\21.1.2\x64\bin\scancentral.bat' failed with exit code 1"

Issue: The Static Assessment task fails with the following error: `The process 'C:\hostedtoolcache\windows\scancentral\21.1.2\x64\bin\scancentral.bat' failed with exit code 1`. The ScanCentral log contains the following error: `java.io.IOException: Cannot run program "msbuild.exe": CreateProcess error=2, The system cannot find the file specified`.

Cause: The MSBuild executable was not added to the PATH environment variable.

Solution: Set the environment variable by running the Batch Script task. For more information, see Adding a Static Assessment Task.

1.5. Getting started with Fortify ScanCentral SAST

You can submit your project to Fortify ScanCentral SAST for remote static analysis (translation and scan). You can also upload and view the results in Fortify Software Security Center. See Adding a Fortify ScanCentral SAST Assessment as a Build Step. With this task, you do not need to install Fortify Static Code Analyzer on the Azure DevOps agent.

Note

To run the translation locally and offload only the scan phase to Fortify ScanCentral SAST, use the Fortify Static Code Analyzer Install task and the Fortify Static Code Analyzer Assessment task (see Getting Started with Fortify Static Code Analyzer Tasks).

This section contains the following topics:

- Requirements for the Fortify ScanCentral SAST task
- Adding a Fortify ScanCentral SAST Assessment task
- Troubleshooting the Fortify ScanCentral SAST task

1.5.1. Requirements for the Fortify ScanCentral SAST task

Make sure that your environment meets the requirements described in this section to use Fortify ScanCentral SAST task in your build. This section also includes preparation steps and information required to have on hand to use the task.

- The Fortify ScanCentral SAST task is available with Fortify ScanCentral SAST versions 20.2.0 or later.
- To trigger a build failure based on the scan results, you must use Fortify ScanCentral SAST version 22.1.0 or later (see Upload results to SSC).
- Fortify ScanCentral SAST runs on a Java Virtual Machine. Make sure that you have a Java Virtual Machine installed on the agent. You can use the Java tool installer task in your pipeline to install it.

  > **Note**
  >
  > You can run the Fortify ScanCentral SAST Assessment task on a Microsoft-hosted agent that might already have a Java Virtual Machine installed.

- Java 17 must be installed on the agent for Fortify ScanCentral SAST client version 24.2.0 or later.
- For Fortify ScanCentral SAST client version 24.2.0 or later, set the `SCANCENTRAL_JAVA_HOME` environmental variable to Java version 17. For Fortify ScanCentral SAST client version 23.2.0 or earlier, set the `SCANCENTRAL_JAVA_HOME` to Java version 11.
- To connect to Fortify ScanCentral SAST, you must have one of the following:
  - The Fortify ScanCentral SAST Controller URL
  - The Fortify Software Security Center URL and a Fortify Software Security Center authentication token of type CIToken (the task determines the Controller information from Fortify Software Security Center)

    Define an Azure DevOps variable that contains the decoded value of this token. By default, the extension uses a variable with the name `ScanCentral.SscCiToken`.
- If the Fortify ScanCentral SAST Controller or Fortify Software Security Center URL uses SSL with a self-signed or untrusted certificate, you might need to add the certificate to the trusted certificates as follows:
  - On the agent's certificate store—To allow the Fortify Azure DevOps Extension to download and install the Fortify ScanCentral SAST client. See the Azure DevOps documentation for how to run with a self-signed certificate.
  - In the Java keystore—To allow the Fortify ScanCentral SAST client to connect to Fortify ScanCentral SAST Controller and Fortify Software Security Center. Use the Java keytool to import a trusted certificate.
- Define an Azure DevOps variable that contains value of the Fortify ScanCentral SAST `client_auth_token` property for the Controller. By default, the extension uses a variable with the name `ScanCentral.ClientToken`.
- Your project must be in one of the supported languages. For a list of languages that are supported for project translation, see the *Fortify Software System Requirements* in Fortify Software Security Center Documentation.

1.5.2. Adding a Fortify ScanCentral SAST Assessment task

Use the **Fortify ScanCentral SAST Assessment** task to perform a remote Fortify Static Code Analyzer analysis using Fortify ScanCentral SAST as part of your build. The project is automatically packaged and then uploaded to Fortify ScanCentral SAST for security analysis. You can also upload the scan results to Fortify Software Security Center.

This task automatically installs a Fortify ScanCentral SAST client from the Fortify ScanCentral SAST Controller on the agent if it is not already installed. In addition, if the Controller version you are using is newer than the Fortify ScanCentral SAST client already installed on the agent, then the task automatically installs the newer version. Make sure that you have enabled auto-updates of Fortify ScanCentral SAST clients from the Controller. The Fortify ScanCentral SAST client is installed in the Azure DevOps Pipelines tool cache.

For detailed information about how to use Fortify ScanCentral SAST, see OpenText™ Fortify ScanCentral SAST Installation, Configuration, and Usage Guide in Fortify Software Security Center Documentation.

To configure a Fortify ScanCentral SAST Assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Find and add the **Fortify ScanCentral SAST Assessment** task.
4. In the **Server Information** section, provide the information described in the following table.

Field	Description
ScanCentral Controller URL	(Optional) Type the URL for the Fortify ScanCentral SAST Controller. The correct format for the Controller URL is: `*<protocol>*://*<controller_host>*:*<port>*/scancentral-ctrl` (for example: `https://myControllerHost.com:8443/scancentral-ctrl`). **Note** If you do not provide the Controller URL, then you must provide the SSC URL and the SSC continuous integration token.
ScanCentral client authentication token	Type a defined variable that contains the value of the `client_auth_token` property for the Fortify ScanCentral SAST Controller. This secures the Controller for authorized clients only. See OpenText™ Fortify ScanCentral SAST Installation, Configuration, and Usage Guide in Fortify Software Security Center Documentation for more information.
SSC URL	(Optional) Type the URL for the Fortify Software Security Center server. **Note** The **SSC URL** is required if you are uploading the scan results to Fortify Software Security Center and if you do not provide a Fortify ScanCentral SAST Controller URL.

SSC continuous integration token	Type a defined variable that contains the decoded value of a Fortify Software Security Center authentication of type CIToken. **Note** The **SSC continuous integration token** is required if you provide an **SSC URL** and if you are uploading scan results to Fortify Software Security Center.
Upload results to SSC	(Optional) To upload the scan results (FPR file) to Fortify Software Security Center, do the following: 1. Select **Upload results to SSC**. 2. Specify an application version that exists in Fortify Software Security Center by providing one of the following: ▪ An application name and an application version name. ▪ A Fortify Software Security Center application version ID. **Note** If you provide both application name and version and an application ID, the extension uses the application ID for the upload regardless of the selected application version type. 3. (Optional) To trigger a build failure based on the scan results, type a search query in the **Build failure criteria** box. For example, the following search query causes the build to fail if any critical issues exist in the scan results: `[fortify priority order]:critical` See OpenText™ Fortify Software Security Center User Guide in Fortify Software Security Center Documentation for a description of the search query syntax. By default, the task returns a warning when the build failure criteria is met. To fail the build instead, select **FAIL** from the **Task results when build failure criteria is met** list. 4. (Optional) To specify how long to poll Fortify Software Security Center to determine if FPR processing is finished, type the time in minutes in the **Polling timeout** box. If no value or a value of 0 is specified, polling continues until FPR processing finishes or stops due to errors. The valid values are 0–10080. 5. (Optional) To specify how frequently to poll Fortify Software Security Center to determine if the FPR processing is finished, in the **Polling interval** box, specify an interval (in minutes). The valid values are 1–60 and the default value is 1 minute.

5. In the **Translation Options** section:

    ◦ Select **Automatically detect build tool** option to automatically detect the build tool.

Note

On Windows agents, if you select the **Automatically detect build tool** option, the default build tool is MSBuild.

On Linux agents, if you select the **Automatically detect build tool** option, the default build tool is DotNet.

If you want to use DotNet as the build tool on a Windows agent, you must explicitly select **DotNet** from the **Build tool** list.

- If you selected **DotNet**, **Gradle**, **Maven**, or **MSBuild** in the **Build tool** list, provide the information described in the following table.

Field	**Description**
Build command	(Optional) Type any custom build commands to prepare and build the project. If not specified, the default build command is used.
Build file	(For DotNet, Gradle, or Maven) Type the name of the build file if it is different than the default of `build.snl`, `build.gradle`, or `pom.xml`. (For MSBuild) Type the name of the build file.
Additional Fortify SCA translation options	Specify a list of Fortify Static Code Analyzer translation options separated by a new line (one per line).
Excludes	Specify the relative paths of files or directories to exclude from the package separated by a new line (one per line).
Skip build	Select whether to skip the build invocation that prepares the generated sources and libraries before the project information is packaged for submission to Fortify ScanCentral SAST.
Include test	(For Gradle and Maven projects only) Select whether to include the test source set (Gradle) or a test scope (Maven) with the scan.
Exclude disabled projects	(For MSBuild projects only) Select whether to skip projects that are either explicitly excluded from the build in the solution or skipped during the build due to platform and configuration settings. **Note** This setting is only valid with Fortify ScanCentral SAST versions 21.1.2 and earlier.

- If you selected **none** for the build tool, provide the information described in the following table.

Field	**Description**
Skip build	Select the check box to disable the project preparation build step before packaging.

Include node_modules dependencies	(Optional) Select whether to restore dependencies to the node_modules directory before the scan.
Python version	(Optional) Select the Python version for Python projects.
Python requirements file	(Optional) Type the name of the Python project requirements file used to install and collect dependencies. Use only this Python field if you have no preference for the Python version used or there is only one Python version installed and on the PATH.
Python virtual environment	(Optional) Type the location (directory) of the Python virtual environment. Specify this together with the Python requirements file to have dependencies restored before the scan.
PHP version	(Optional) Type the PHP version used in the project.
Translate Apex project	Select this option if your project consists of Apex and Visualforce code.
Translate SQL project	Select this option if your project is an SQL project and then select if your project is **PL/SQL** or **T-SQL**.

6. (Optional) In the **Scan Options** section, provide the information described in the following table.

Field	Description
Filter file	Type the name of a filter file to filter out specific vulnerability categories, rules, and vulnerability instances from the analysis. For more information, see OpenText™ Fortify Static Code Analyzer User Guide in Fortify Static Code Analyzer and Tools Documentation.
Issue template	Type an issue template to include for the scan. An issue template determines how issues uncovered in your project are filtered and sorted.
Custom Rulepacks	Specify any custom rules files (`*.xml`) separated by spaces or specify a directory that contains custom rules.
Additional Fortify SCA scan options	Specify a list of Fortify Static Code Analyzer scan options separated by a new line (one per line).

7. (Optional) In the **Advanced Options** section, provide the information described in the following table.

Field	Description
Notification email	Type the email address to which the Fortify ScanCentral SAST Controller will send notifications.
Sensor pool UUID	To target a specific sensor pool for the scan, specify the sensor pool UUID. You can obtain the UUID for sensor pools from the ScanCentral SAST **Sensor Pools** page in Fortify Software Security Center. By default, Fortify ScanCentral SAST uses the default sensor pool as defined in Fortify Software Security Center.

Wait for scan to finish	Select whether to have this task wait until the scan is complete and the results are downloaded to the DevOps agent. If selected, then you can provide the following: ◦ In the **Results file** box, type a name for the Fortify results file (FPR). For example, `MyProjectA.fpr`. The file is saved in the working folder unless you specify an absolute path. ◦ In the **Log file** box, type a name for the local log file. The file is saved in the working folder unless you specify an absolute path. ◦ Select **Overwrite** to replace any existing results file (*.fpr) or log file with new data. Otherwise, existing files are not overwritten and the results are not downloaded to the agent. A message will indicate if this happens.
Quiet	Select this option to prevent execution statements from being written to stdout during the build.
Download debug logs	Select this option to generate a ZIP file that includes debug log files from clients, sensors, and Fortify Static Code Analyzer. **Note** If you select this option, the process waits for the completion of the scan in order to download the log files.

1.5.3. Troubleshooting the Fortify ScanCentral SAST task

Unable to open the FPR file in the email notification from Fortify ScanCentral SAST

You can use Postman or cURL (available on Windows 10) to download the FPR or log file mentioned in the email notification from Fortify ScanCentral SAST.

To use Postman to download the FPR or log file:

1. Copy the URL for the FPR or the log file from the notification email.
2. Paste the URL in the Postman URL text field and then add `fortify-client` in the HTTP header.
3. Click **Send and Download**.
4. Save the file.

Unsupported class version error

If Fortify ScanCentral SAST client is run with an unsupported Java version (see Requirements for the Fortify ScanCentral SAST Task), the following message appears in the log file:

```
java.lang.UnsupportedClassVersionError: com/fortify/scancentral/launcher/Launcher has been compiled by a more recent version of the Java Runtime...
```

If you are using Fortify ScanCentral SAST client version 22.2.0 and later, set the SCANCENTRAL_JAVA_HOME environment variable to point to the supported Java version. Alternatively, make sure that the correct Java version is installed on the agent. If multiple Java versions are available on the agent, make sure the pipeline that runs the Fortify ScanCentral SAST task has PATH or JAVA_HOME environment variables that point to the supported Java version.

Failure with a self-signed certificate error

You are connecting to Fortify ScanCentral SAST Controller or Fortify Software Security Center using SSL with a self-signed or untrusted certificate. Add the certificate to to both the agent certificate store and the Java keystore (see Requirements for the Fortify ScanCentral SAST Task).

1.6. Getting started with Fortify ScanCentral DAST

You can submit your Web application to Fortify ScanCentral DAST for a dynamic scan and view the results in Fortify Software Security Center. See Adding a Fortify ScanCentral DAST Scan as a Build Step.

This section contains the following topics:

- Requirements for the Fortify ScanCentral DAST task
- Adding a Fortify ScanCentral DAST Assessment task

1.6.1. Requirements for the Fortify ScanCentral DAST task

Make sure that your environment meets the requirements described in this section to use Fortify ScanCentral DAST task in your build. This section also includes preparation steps and information required to have on hand to use the task.

Fortify ScanCentral DAST Requirements

- You must use Fortify Software Security Center and Fortify ScanCentral DAST version 20.2.0 or later.
- You must have the Fortify ScanCentral DAST API URL.
- If the ScanCentral DAST API uses SSL with a self-signed or untrusted certificate, verify that the ScanCentral DAST API URL is accessible from the Azure DevOps agent. You might need to add the certificate to the trusted certificates on the agent.
- You must have a CI/CD identifier for the Web application you want to scan.

1.6.2. Adding a Fortify ScanCentral DAST Assessment task

Use the **Fortify ScanCentral DAST Assessment** task to perform a scan of your Web application as part of your build. After you run the build and the scan is complete, the scan results are available in Fortify Software Security Center. For more information about configuring and using Fortify ScanCentral DAST, see OpenText™ Fortify ScanCentral DAST Configuration and Usage Guide in Fortify ScanCentral DAST Documentation for versions 20.2.0 and later.

To configure a Fortify ScanCentral DAST Assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Find and add the **Fortify ScanCentral DAST Assessment** task.
4. Provide the information described in the following table.

Field	**Description**
ScanCentral DAST API URL	Specify the URL and port where the DAST API service runs in the format `<protocol>://<DAST_API_hostname>:<port>/api` or `<protocol>://<DAST_API_IP_address>:<port>/api`.
CI/CD identifier	Specify a scan settings identifier GUID. This is also known as the Settings Identifier.
SSC continuous integration token	Specify an Azure DevOps variable that contains the decoded value of a Fortify Software Security Center authentication token of type CIToken.
Overrides	(Optional) Fortify ScanCentral DAST scan setting overrides (JSON format).

1.7. Getting started with Fortify WebInspect

- Install an agent on a Virtual Machine.
- Install an instance of Fortify WebInspect on the agent.
- Configure and start the Fortify WebInspect API on the agent.
- Create a Scan Settings file on the agent to be used during the scan.

For more information about how to install and configure Fortify WebInspect, see the installation and the user guide in Fortify WebInspect Documentation.

This section contains the following topics:

- Setting up a Fortify WebInspect Dynamic Assessment
- Troubleshooting the Fortify WebInspect Dynamic Assessment task

1.7.1. Setting up a Fortify WebInspect Dynamic Assessment

To configure a Fortify WebInspect Dynamic Assessment task:

1. In an Azure DevOps project, navigate to your existing build pipeline.
2. Click **Edit**.
3. Find and add the **Fortify WebInspect Dynamic Assessment** task.
4. In the **Scan Settings** box, type the name of the settings file to use in the scan.
5. In the **WebInspect API** box, type `http://<hostname>:<port>/`, where *<hostname>* and *<port>* identify where the WebInspect API is installed.

   > **Important**
   >
   > You must specify the WebInspect API location. The task will not start without this information.

6. In the **Scan Results** box, type the location where you want the scan results written.

For more information about the WebInspect API, see the API documentation at `http://<hostname>:<port>/webinspect/api` on the agent where Fortify WebInspect is installed. If you used the default settings when configuring the Fortify WebInspect API, then type `http://localhost:8083/webinspect/api`.

1.7.2. Troubleshooting the Fortify WebInspect Dynamic Assessment task

If the Fortify WebInspect Dynamic Assessment task fails to start, you might need to stop the Fortify Monitor program on the agent and restart it with Administrator privileges.

For more info, visit https://docs.microfocus.com