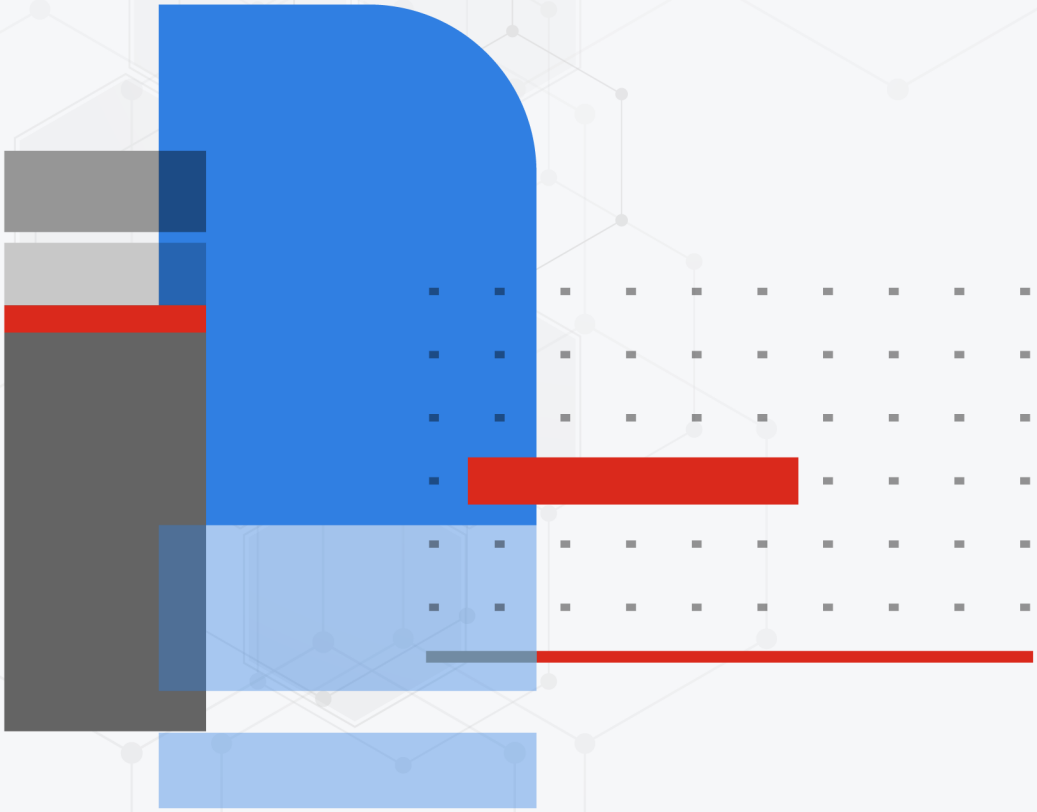
User Guide

FortiDevSec 24.4.0



FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO LIBRARY
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/training-certification

FORTINET TRAINING INSTITUTE
https://training.fortinet.com

FORTIGUARD LABS
https://www.fortiguard.com

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

December 18, 2024
FortiDevSec 24.4.0 User Guide
68-244-1108129-20241218

TABLE OF CONTENTS

Change log

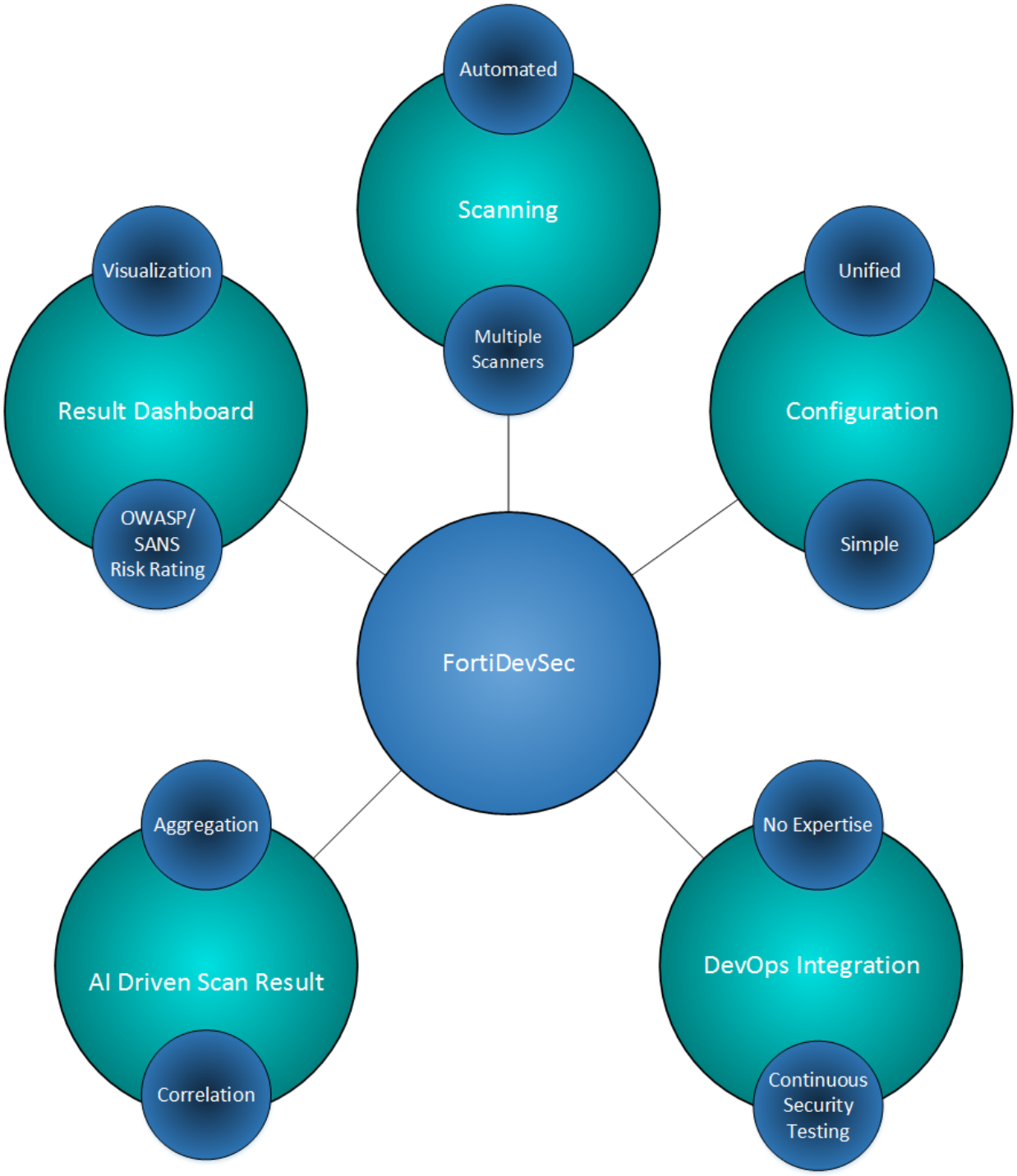
Date	Change description
2024-12-18	FortiDevSec version 24.4.0 release document.
2025-02-19	Updated the following topics: • CI/CD Tools • Running the Security Scan • Frequently Asked Questions (FAQs)
2025-03-11	Updated Modifying the Vulnerability Status.
2025-03-27	Updated Email Notification topic.

Introduction

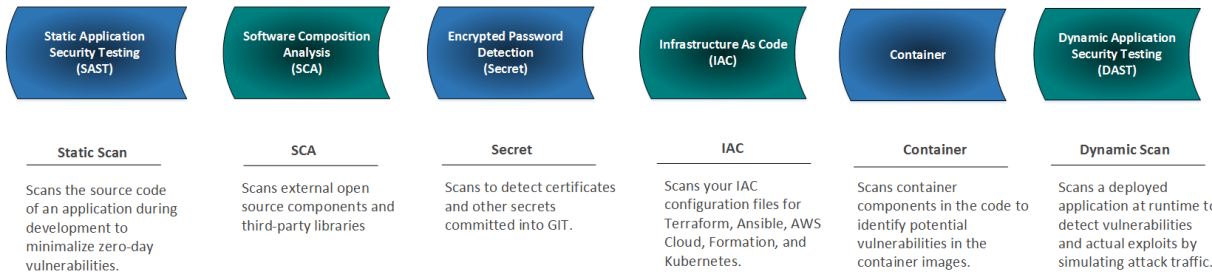
The realm of application security involves tools and techniques to protect applications from attacks and violations. Due to the huge advancement in hacking techniques and cyber-attack methodologies even modern complex applications contain unassessed security risks and vulnerabilities that may lead to substantial harm to your organization/business. Evaluating the security risks associated with applications and assessing the security weaknesses allows you to mitigate the potential risk to your organization with appropriate remedial measures.

What is FortiDevSec

FortiDevSec is a cloud-based automated application security tool that performs intensive and comprehensive scans for an accurate vulnerability assessment of your application. It integrates continuous application security testing into major DevOps Continuous Integration (CI)/Continuous Deployment (CD) environments, embedding itself into the process of developing and deploying applications to evaluate and detect security gaps that you can mitigate/remediate in the course of the Software Development Lifecycle (SDLC). The automated scanning process resides in your CI/CD pipeline and allows you to scan your applications without manual intervention and is completely non-intrusive with no disruptions to your setup. The easy-to-understand application security assessment approach of FortiDevSec allows you to build secure applications and involves a simple 3-step procedure that facilitates application scanning with minimal know-how of the application security domain.



FortiDevSec packages multiple security scanners into a single solution that includes source code scanners, static, dynamic, SCA, Secret, IaC, and Container scanners. The FortiDevSec scanning process automatically determines the relevant scanner type(s) based on the scanned applications across multiple languages and frameworks. FortiDevSec provides zero effort deployment and saves you the overhead of installing and managing multiple scanners and plugging these into your setup individually.



The FortiDevSec application scanning is a simple procedure that includes creating a single unified configuration file and running the scanner CLI. The *fdevsec.yaml* file integrates basic and advanced configurations for all security scanners and application languages avoiding fragmentation or multiple configuration steps.

The architecture of FortiDevSec integrates continuous application security testing into your DevOps CI/CD workflow and adopts a minimalistic approach towards the security testing procedure enabling DevOps personnel to integrate and run comprehensive application security scans without any security domain expertise. It seamlessly integrates with all major Devops CI/CD platforms to find security issues during the SDLC.

The scan result is aggregated and correlated for all applications across different scan types using advanced Artificial Intelligence (AI)/Machine Learning (ML) and uploaded in the FortiDevSec cloud providing a detailed insight into the scanned applications with a complete view of security risks. The applications are assigned standardized risk rating based on Open Web Application Security Project (OWASP) and SysAdmin, Audit, Network and Security (SANS) factors. FortiDevSec associates each vulnerability to an *OWASP Top 10* category or a *SANS Top 25* rank by assessing the CWE IDs. The AI driven scan results and risk rating methodology prioritize the detected vulnerabilities based on the assessed severity with minimum false positives and noise. The interactive and customizable dashboard user interface is organized to display scan statistics in a distinctive way with ease of accessibility, navigation, and data filtering.

The high vulnerability detection rate and their intelligent prioritization in the FortiDevSec scan result offers robust risk determination capabilities that facilitate prompt response and appropriate remedial measures for the identified risks. You can configure the risk rating criteria for your application and based on the result analysis, you can manage the scan findings in the GUI by assigning a suitable status to each vulnerability.

Support Matrix

Supported Scanners

Scanner	Description
SAST	Scans the source code of an application during development to minimize zero-day vulnerabilities. The application languages supported for SAST are *Shell, Java, Ruby on Rails, Python, Golang, PHP, JavaScript*/*NodeJS*, *C*, *C++*, *C# .Net*, and *TypeScript*.
SCA	Scans for vulnerabilities in the open-source libraries/components used by the application. The programming languages supported by the SCA scanner are *Java, Javascript, Ruby, Python, Golang, C# .Net and PHP*. SCA supports scanning multiple Git repositories within same directory.

Scanner	Description
	Also, SCA scans for *Outbreak Alerts* and *Supply Chain Attacks* identified by *FortiGuard Labs Threat Research*.
Secret	Scans hard coded secrets such as passwords, API keys, and tokens in git repository commits. See FortiDevSec Secret Scanner.
IaC	Scans your IaC configuration files for *Terraform, Cloud Formation, Docker* and *Kubernetes*, to detect configuration issues.
Container	Scans container components to identify potential vulnerabilities.
DAST	Scans a deployed application at runtime to detect vulnerabilities. The DAST scanner supports scanning of assets/targets hosted on both the internal network of an organization and the external/public network using FortiDAST proxy server. See FortiDAST Proxy Server. The DAST scanner allows you to configure a full or a quick scan using the FortiDAST, for more information see FortiDAST Scanner. • **Quick Scan** : A quick scan is fast mode scanning that provides vulnerability assessment based on limited testing/scraping of the static pages of your asset. These pages are scraped by searching and extracting URLs from HTML tags and attributes. • **Full scan**: A full scan provides vulnerability assessment based on complete testing/scraping of the static and dynamic pages of your asset. The Crawler also performs browsing simulation such as clicking of buttons, links, and images to test the interaction between the dynamic pages and the browser. This mode of vulnerability assessment takes longer than a quick scan.

Supported CI/CD Pipeline Tools

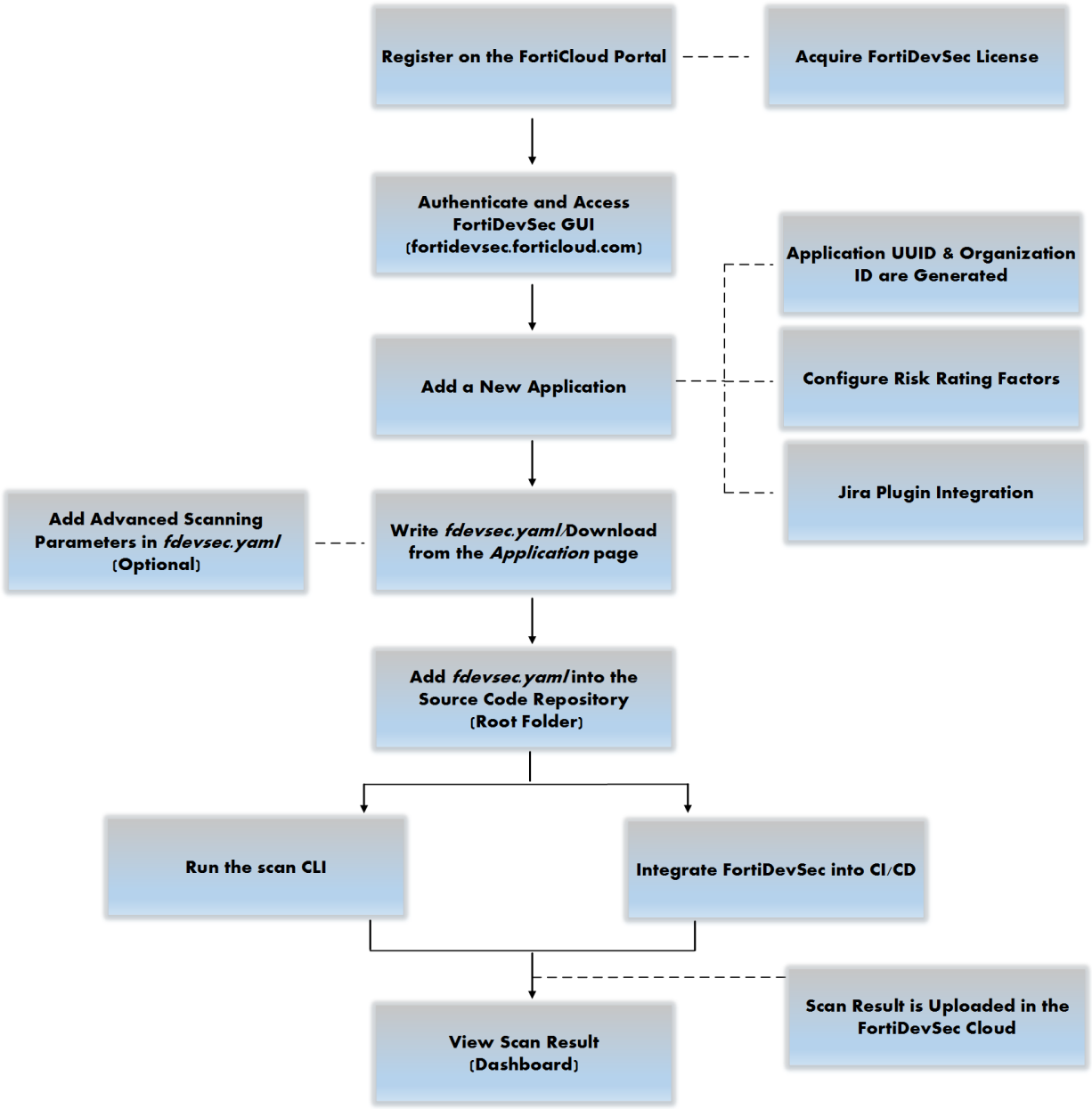
Support for the following CI/CD tools is available. For more information, see Running the Security Scan

- AWS CodePipeline
- Azure DevOps
- Bamboo
- CircleCI
- Drone CI
- GCP Cloud Build
- GitHub Actions
- GitLab
- Jenkins
- Travis CI
- Bitbucket
- JFrog (for GitLab and GitHub projects)

How FortiDevSec Works

You can scan your applications by integrating FortiDevSec into your CI/CD setup. When you run the scan, FortiDevSec automatically determines the open-source scanners to use based on the application language. It uses Docker images for the required scanners and uploads the scan results in the FortiDevSec cloud.

- To quickly scan your application, see section Beginner's Tutorial.
- For detailed configuration and scanning procedure, see section Scanning an Application.



FortiDevSec Secret Scanner

FortiDevSec scans your git repository commits for hard coded secrets such as passwords, API keys, and tokens. Secret scanner will not scan the files in the git repository.

Note:

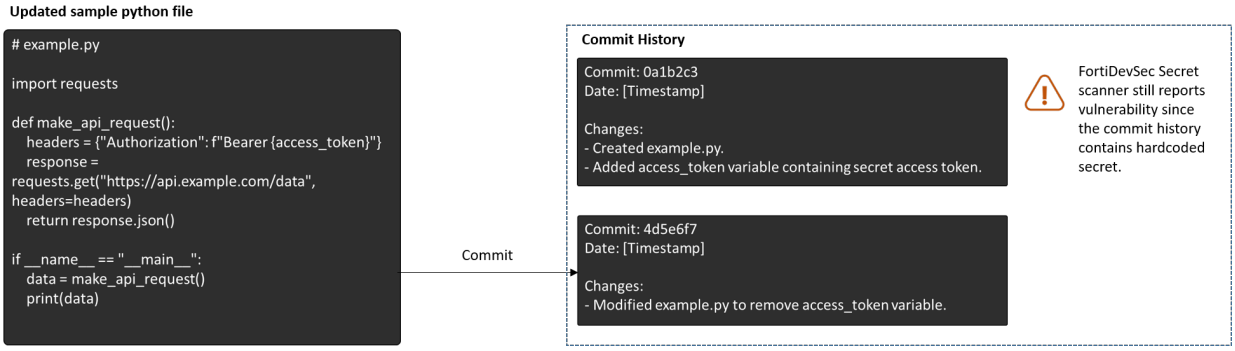
- Vulnerabilities will continue to be flagged as long as commits containing secrets remain in the git repository/branch.

Example:

The following example demonstrates how FortiDevSec secret scanner works.

Consider a sample python script file *example.py*. This file contains a hardcoded secret variable called *access_token*. When you commit the Python file to a git repository and run the FortiDevSec secret scan, FortiDevSec will identify and report the vulnerability.

Even if the access token is subsequently removed from the example.py file, it may still be identified by the secret scanner due to its presence in the commit history.



It is recommended to execute the secret scan on local development branches before pushing any commits containing secrets to the git repository.

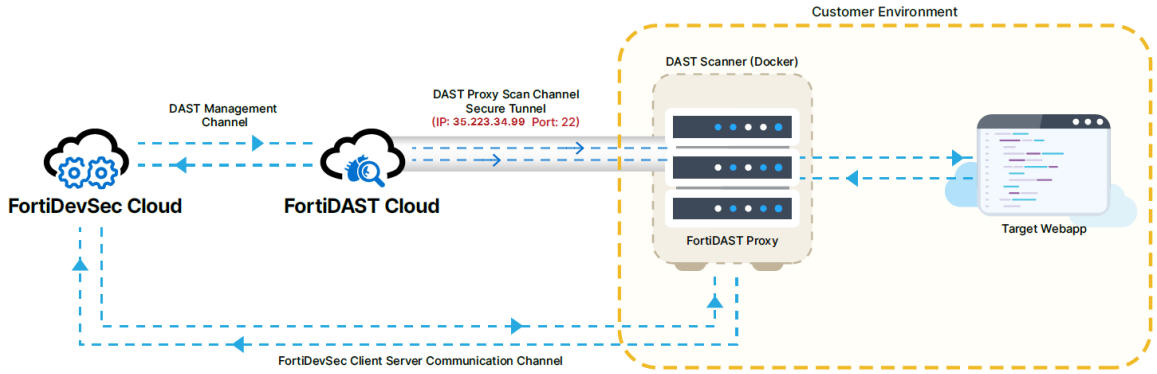
To address and eliminate any reported secret vulnerabilities identified by FortiDevSec, remove the specific commit from the respective git repository/branch.

FortiDAST Proxy Server

The FortiDAST uses a proxy server running in the DAST docker container for asset/application access. When the container is invoked with the required configurations, it interacts with the FortiDAST cloud through REST APIs, and begins the scanning process automatically. This involves authorizing the asset, initiating the scan, waiting for the scan to complete, and stopping the container after the scan is complete.

Notes:

- Ensure that the SSH service is enabled and the firewall allows the communication between the FortiDAST proxy and FortiDAST Cloud (*IP: 35.223.34.99* and *Port: 22*).
- Ensure that the asset/application is reachable from the FortiDAST docker's network through which you are performing the scan.



Licensing

You are required to acquire a license to access FortiDevSec. FortiDevSec licensing is subscription based and is available in the following configurations:

License	Validity in years	Maximum users	Maximum assets/apps supported for DAST scanner
FortiDevSec Standard	1	5	5
FortiDevSec FortiDAST Add-on (Optional)	1	n/a	5*

* In addition to FortiDevSec Standard license.

Contact the Fortinet *Customer Support* team for license purchase.

Notes:

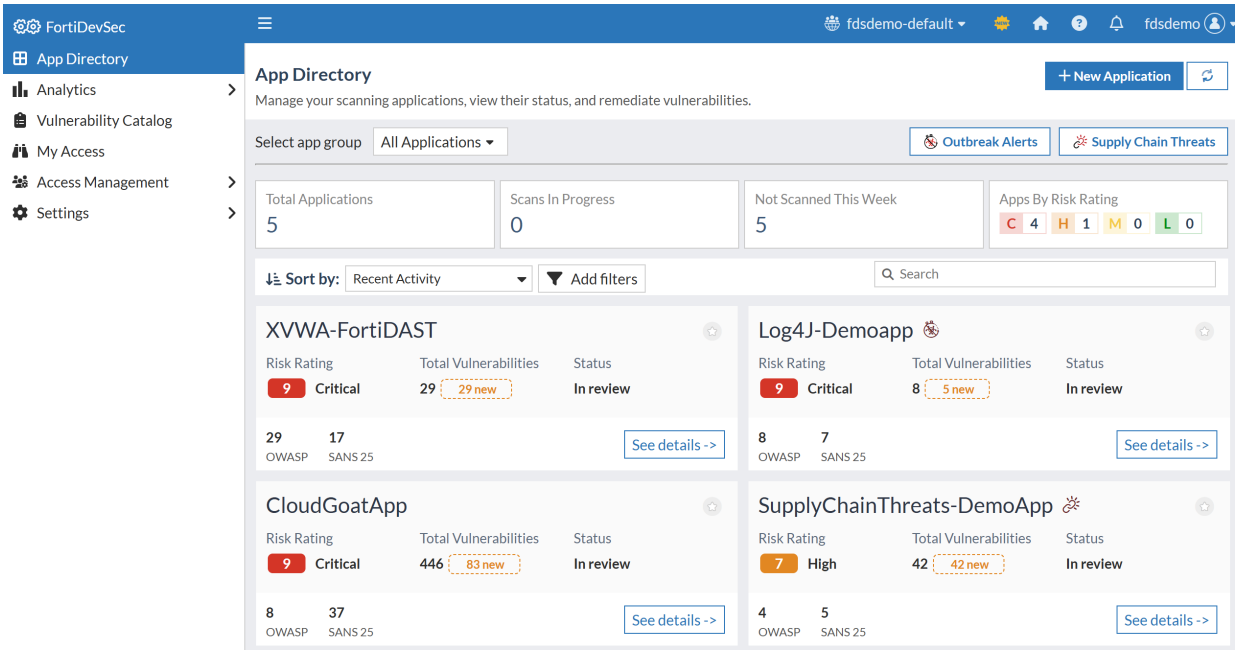
- After license expiry, FortiDevSec allows you to scan existing applications for a grace period of 1 month, but you cannot view the scan results on the portal GUI. During grace period you will not be able to perform DAST scans or add any new assets for scanning.
- Each FortiDevSec Standard license comes with five users support : One Master user and Four additional users, which can be Sub-users, IAM users, or IDP users.
- FortiDevSec licenses are stackable, you can add more users by purchasing co-term contracts.
- For DAST scanning, you can add assets only using FortiDevSec with FortiDevSec Standard/ FortiDAST Add-on license. To add assets directly in FortiDAST, you need FortiDAST license.

Viewing License Information

Click **Organization ID** in the header and select **Org licenses**, to view the licensing information currently in use.

User Interface Overview

The FortiDevSec user interface provides a streamlined approach to managing your application security. The home page contains six sections accessible from the left navigation menu.



Section	Description
App Directory	The *App Directory* section allows you to access detailed scan results, view critical FortiGuard Outbreak Alerts, access vulnerability insights, and stay informed about potential supply chain attacks. See App Directory
Analytics	The *Analytics* section allows you to gain a comprehensive view of your security posture with organization-wide risk metrics (Current Status), historical trend analysis (Historical

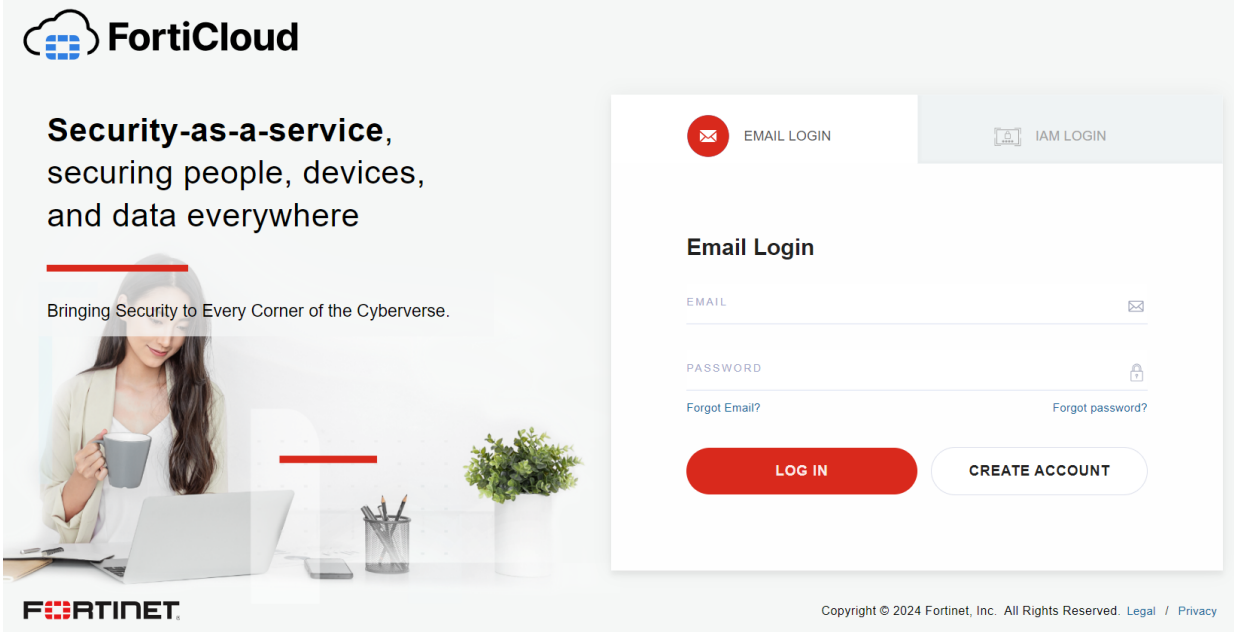
Section	Description
	Insights), and in-depth, application-specific data (Application Level). See Analytics
Vulnerability Catalog	The *Vulnerability Catalog* page displays a list of all vulnerabilities for the selected scanned application, allowing you to filter, group, and explore in detail. See Vulnerability Catalog.
My Access	The *My Access* page provides insights into your groups, access rights, join requests, and available shared groups to join. See My Access.
Access Management	The *Access Management* section allows you to manage member groups, application groups, and group requests, as well as control member permissions at the application group level. See Access Management.
Settings	The *Settings* section allows you to customize email notifications for important security events and product updates, and securely manage API tokens for integration with your existing workflows. See Settings.

Signing-on for FortiDevSec

This release provides single sign-on support for FortiDevSec along with FortiCloud suite of products. FortiDevSec is accessible via the *FortiCloud* GUI - https://support.fortinet.com. Eventually you are redirected to the FortiDevSec login page.

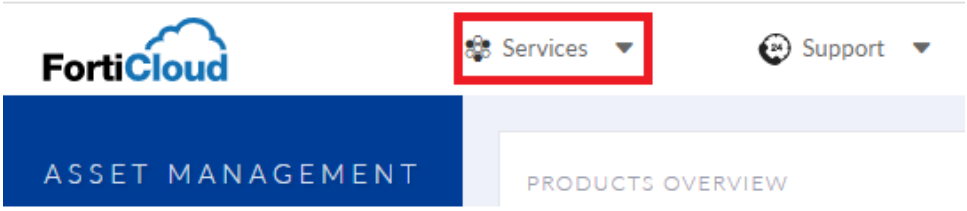
Registering on FortiCloud

Prior to using FortiDevSec, you are required to register on the *FortiCloud* portal. Use the https://support.fortinet.com access link to register on the *FortiCloud* portal. A security code is emailed to the address specified during registration; use the code to complete registration and activate your account.

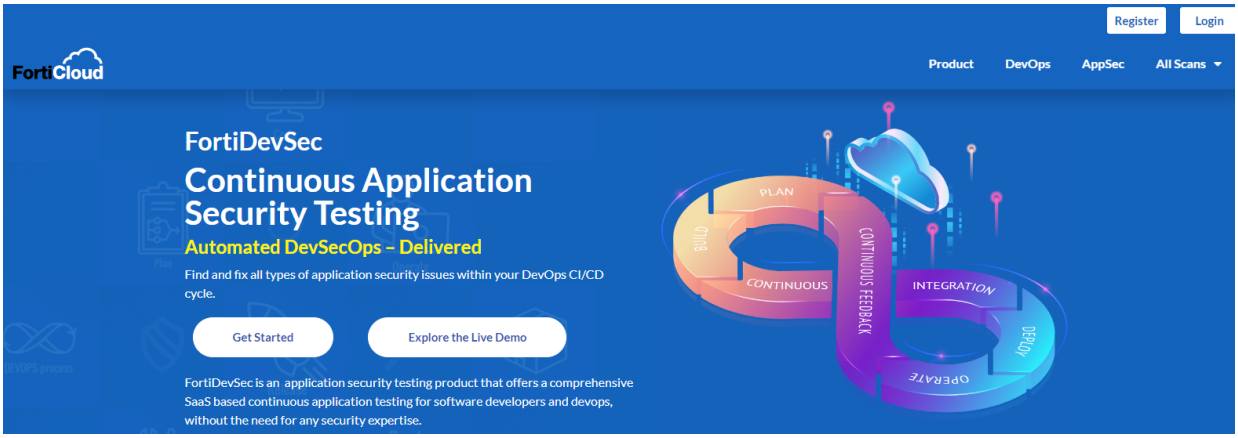


Accessing FortiDevSec

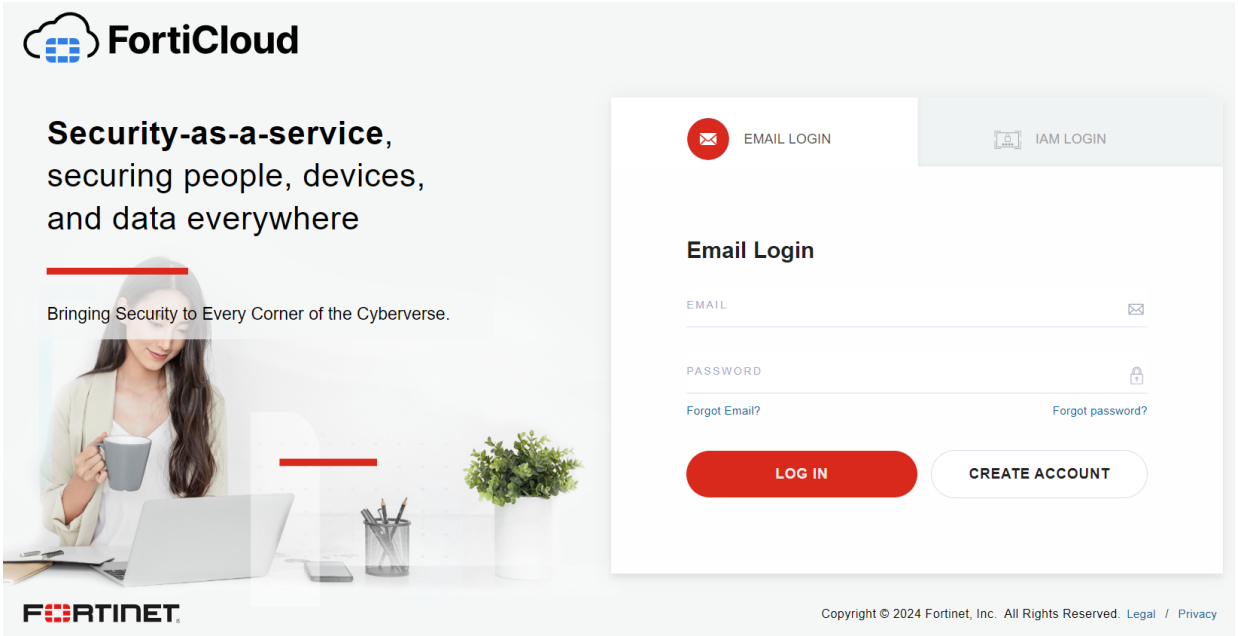
Any user registered on https://support.fortinet.com can access FortiDevSec. Once you login into *FortiCloud*, click on **Services**, a banner with Fortinet products is displayed.



Select FortiDevSec and you are redirected to the FortiDevSec portal, *https://fortidevsec.forticloud.com/*.



You can explore a demo of FortiDevSec with live application scan data or login.



Note: Click *New* icon in the header to view the new features delivered in a release.

User Management

FortiDevSec user management allows you to create and manage users who have access to your FortiDevSec account. FortiDevSec user management is based on organizations (ORG IDs). After purchasing the license, the account you use to register FortiDevSec license will be assigned an ORG ID. This account will be the master account for your ORG ID.

Only one ORG ID will be assigned to each account. If you register additional licenses using the same account, the licenses will be added to your existing ORG ID. To get a new ORG ID, you must use a different account.

The master account user can add sub-users. Master and sub-users can access the applications created under that ORG ID. One sub-user can be a part of multiple master accounts. When logging in, the sub-user can select which sub-account they want to access.

FortiDevSec supports user credentials created in the Identity & Access Management (IAM) portal. On FortiCloud, you can create IAM users and External IdP roles, and use them with FortiDevSec.

For more information about using the IAM portal, see the Identity and Access Management Administration Guide.

Once users are created, you can define granular access to applications using application and member groups. See Access Management.

- Permission profiles
- IAM users
- External IdP roles

Permission Profiles

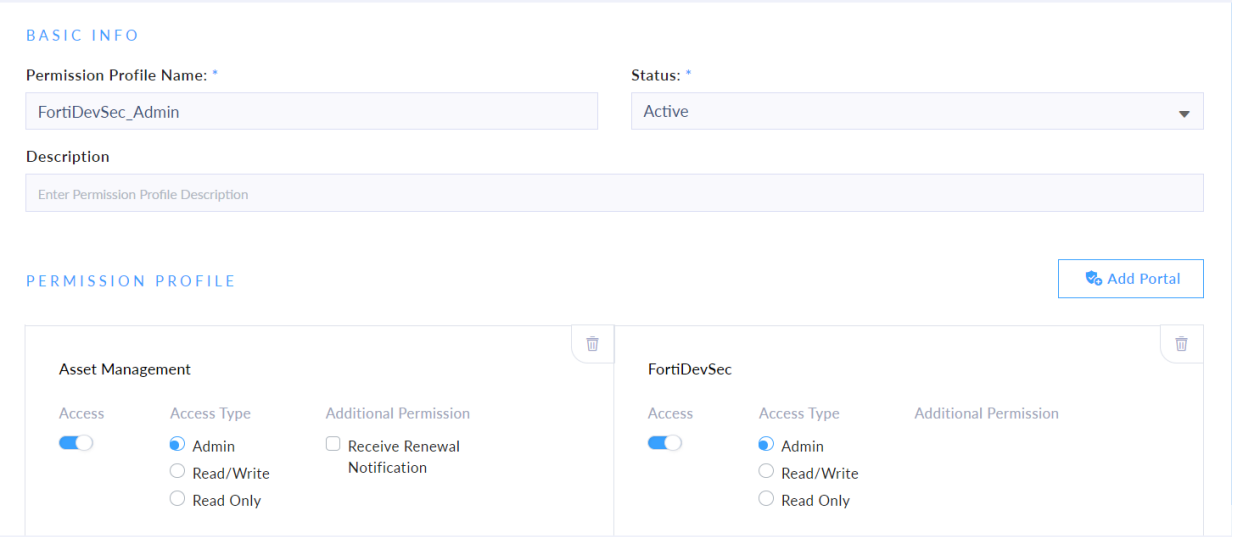
Before you can create IAM users, user groups or external IdP roles, you must create a permission profile. Permission profiles define the level of portal access and permissions a user has. Permission profiles allow you to explicitly enable or disable access to FortiCloud portals and grant portal-specific permissions for the enabled portals.

Adding Permission Profiles

Access the **Identity & Access Management** (IAM) service from the FortiCloud portal. Configure the Cloud Management & Services permissions to enable access to FortiDevSec.

1. Navigate to **Permission Profiles**.
2. Click **Add New**.
3. Click **Add Portal** and select FortiDevSec from the list and click Add.
4. Configure the required permissions for FortiDevSec:
   a. Toggle **Access** to allow access to FortiDevSec.
   b. Select the required Access Type: **Admin**.

For more information about permission profiles, see Permission profiles in the Identity and Access Management (IAM) guide on the Fortinet Documents Library.

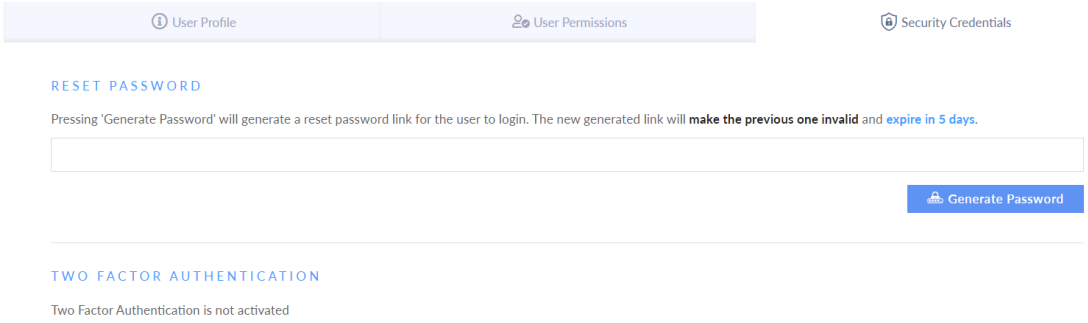


IAM Users

The IAM user type provides more control and flexibility when assigning user permissions. Save time creating new users by applying the permissions of an existing user to a new user or adding the user to a group.

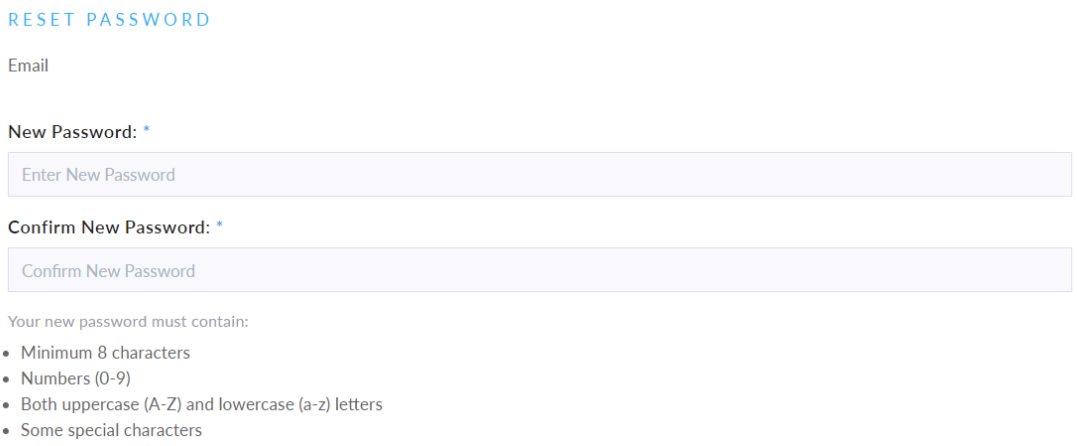
Adding an IAM user:

1. Go to FortiCloud (https://support.fortinet.com/), and log in.
2. From the Services menu, select IAM. The IAM portal is displayed.
3. Create a new IAM user.
   For more information, see Adding IAM Users in the Identity and Access Management (IAM) guide on the Fortinet Documents Library.
4. Add an IAM user group, and add the user to it.
   For more information, see Adding IAM User Groups in the Identity and Access Management (IAM) guide on the Fortinet Documents Library.
5. Generate an IAM user login password.
   a. Go to IAM Users, and click the full name of the user. The User Profile tab is displayed.
   b. Go to the Security Credentials tab, and click Generate Password.



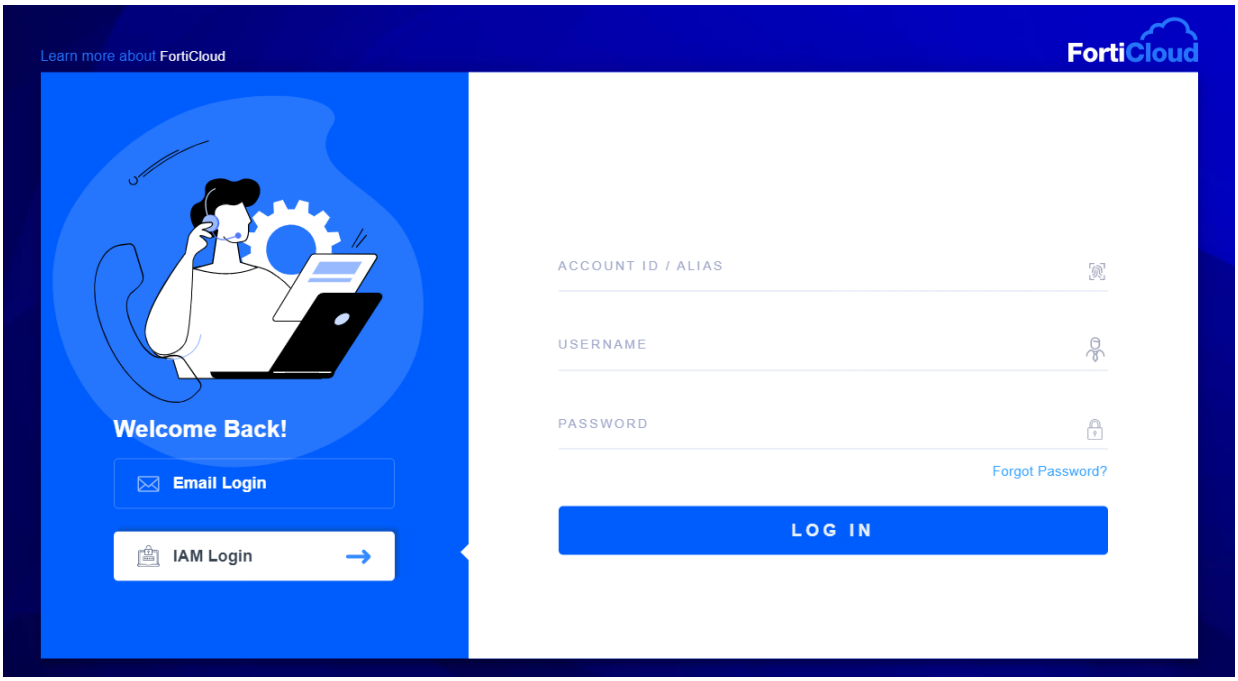
   c. Click **Copy Reset Link** to copy the link and paste it in a new tab.

d. Set the new password and click **Submit**.



e. Share the credentials with the IAM user.

6. The IAM user can use the credentials to log in to FortiCloud.
   After logging in to FortiCloud, the IAM user has access to FortiDevSec portal.

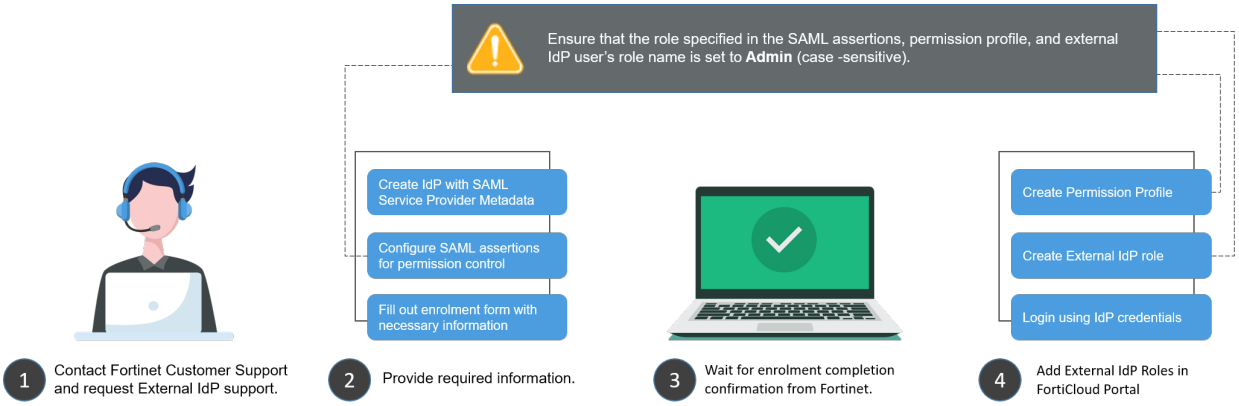


External IdP Roles

FortiDevSec supports integration of third-party Identity Provider (IdP) services to log-in and manage networks. This feature is useful for enterprises that need to secure their user credentials and hence provision FortiDevSec access through their own Identity Provider. The external IdP initiated Security Assertion Markup Language (SAML) assertion consisting of specific IdP attributes is used by FortiCloud/FortiDevSec to verify the user account details and grant required access.

Configuring External IdP

The graphic below depicts an overview of configuring the external IdP.



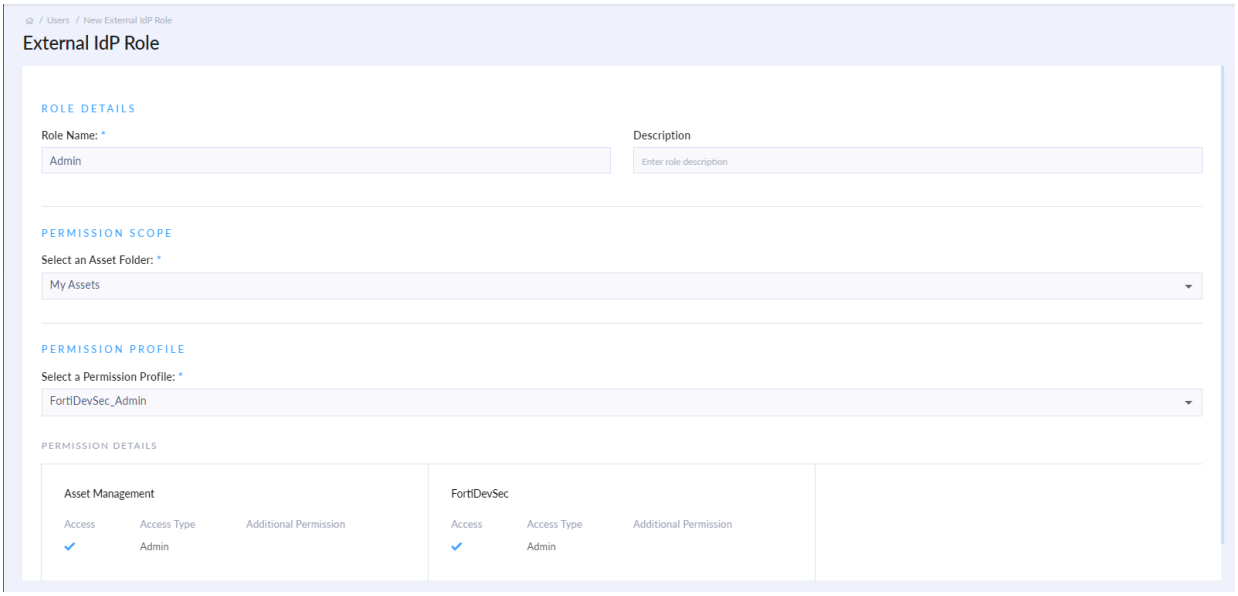
External IdP authentication is offered in conjunction with FortiCare and FortiAuthenticator. To configure external IdP support perform the following steps:

1. Contact the Fortinet *Customer Support* team and request External IdP support.
2. Provide the required information and initiate enrollment with the appropriate FortiCare accounts.
   a. Create an IDP with SAML Service Provider Metadata. You'll need to provide specific URLs for SP Entity ID, Login URL, and Relay State. Support for SAML 2.0 and IDP-initiated assertion response is necessary. The following is an example where *company* is the unique name of your organization.
   ```
   SP Entity ID http://customersso1.fortinet.com/saml-idp/proxy/
   {company}/metadata/
   SP Login URL https://customersso1.fortinet.com/saml-idp/proxy/
   {company}/saml/?acs
   Relay State https://customersso1.fortinet.com/saml-idp/proxy/
   {company}/login/
   ```
   b. Configure the SAML assertions with the *username* and *role* attributes for permission control in FortiCloud.
   Notes:
   - FortiDevSec currently supports a single user role named **Admin** and a single access type named **Admin**.
   - The role name configured must exactly match the role attribute defined in FortiDevSec. For FortiDevSec the role attribute defined is **Admin**(case-sensitive).

   c. In the enrollment form, provide specific information to Fortinet, such as, the SAML Metadata file, company name, contact information, and the Fortinet master account registered in FortiCloud that the IdP requires to connect to.
3. Wait for confirmation of enrollment completion from Fortinet. After successful enrollment, configure external IdP roles in FortiCloud to grant the required access.
4. To add an external IdP role, access the **Identity & Access Management** (IAM) service from the FortiCloud portal and perform the following steps:
   Note: Ensure the permission profile is created before adding an external IdP role. See, Adding Permission Profiles.
   a. Navigate to **Users > Add New** and click **Extenal IdP User**.
   b. Enter **Admin**(case-sensitive) as **Role Name** and **Description** (optional).

c. Select an asset group from the **Asset Permissions** list.
d. Select the Permission profile.
e. Click **Add Role**.
After the role is created, it is listed on the on the **Manage External IdP Roles** page. You can enable/disable or delete a created role. Select the role and click on the required option.

Note: FortiCloud's IdP user modifications require a minimum of 15 minutes before they are reflected on FortiDevSec.



Beginner's Tutorial

This tutorial aims at using FortiDevSec to run a security scan on your application quickly.

- Automated Application Scanning
- Manual Application Scanning

Automated Application Scanning

This tutorial aims to automate a security scan on your application in a CI/CD environment. Ensure that the Prerequisite is met, see section Scanning an Application for more details.

- Adding a New Application
- Integrating the fdevsec.yaml
- CI/CD Configurations
- Viewing the Scan Result

Adding a New Application

Login into the FortiDevSec portal and add a new application for your organization.

1. Click on the **New Application** tab and enter the application name.
2. Click **Next** and the **App Setup** information is displayed, download the *fdevsec.yaml* file from the application page.

You can *optionally* configure the risk ratings for your application. See section Adding a New Application for detailed procedure.

Integrating the *fdevsec.yaml*

Integrate the *fdevsec.yaml* into your CI/CD as defined in the next step (based on the CI/CD tool). This tutorial uses only the mandatory parameters in the configuration file, you can add optional (advanced) parameters to make your scan more precise.

The application languages are automatically detected and FortiDevSec runs the appropriate scans.

See section Configuring the Scanner (fdevsec.yaml) for detailed procedure.

CI/CD Configurations

Integrate scan configurations into your CI/CD tool. See Running the Security Scan.

Viewing the Scan Result

The dashboard of the FortiDevSec portal lists the applications, click on your application to view and analyze comprehensive details of the detected vulnerabilities.

See section Viewing the Scan Result for more details.

Manual Application Scanning

This tutorial aims to run a security scan for your application manually in your source code through the CLI. Ensure that the Prerequisite is met, see section Scanning an Application for more details.

- Adding a New Application
- Integrating the fdevsec.yaml
- Running the Scan
- Viewing the Scan Result

Adding a New Application

Login into the FortiDevSec portal and add a new application for your organization.

1. Click on the **New Application** tab and enter the application name.
2. Click **Next** and the **App Setup** information is displayed, download the *fdevsec.yaml* file from the application page.

You can *optionally* configure the risk ratings for your application. See section Adding a New Application for detailed procedure.

Integrating the *fdevsec.yaml*

Add the *fdevsec.yaml* file into the root folder of your source code. This tutorial uses only the mandatory parameters in the configuration file, you can add optional (advanced) parameters to make your scan more precise.

The application languages are automatically detected and FortiDevSec runs the appropriate scans.

See section Configuring the Scanner (fdevsec.yaml) for detailed procedure.

Running the Scan

Navigate to the root folder of the source code and run this command.

Use the SAST command for static source scan.

```
docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_sast:latest
```

Use the DAST command for dynamic application scan.

```
docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_dast:latest
```

See section Running the Security Scan for detailed procedure.

Viewing the Scan Result

The dashboard of the FortiDevSec portal lists the applications, click on your application to view and analyze comprehensive details of the detected vulnerabilities.

See section Viewing the Scan Result for more details.

Scanning an Application

Perform these procedures to scan your applications for vulnerabilities.

Prerequisite

Prior to running a SAST or DAST scan from a host/machine through the CI/CD pipeline (automatic/manual), ensure that the Docker engine is installed in that host/machine, and has required user access/permission to run the Docker. To install the Docker engine across different platforms, see Docker.

System Requirements

The system requirements for a FortiDevSec default scan can vary depending on the following factors.

- **Codebase size**: Factors like number of lines of code, number of files, and size of individual files can all impact resource utilization..
- **Code complexity**: Factors like nesting of control flow statements, dependencies between functions and files, complex looping structures, and use of advanced algorithms can all impact resource utilization.
- **Scanner configuration**: The number of scanners used and whether scans are run serially or in parallel impact resource utilization.
- **Life of the repository**: Older repositories with a larger number of commits may require more processing power for secret scanning due to the increased data volume.

Following are the hardware requirements for the system where you will run SAST or DAST scans.

Component	Minimum Requirements
CPU	2 cores
Memory	4 GB
Storage	10 GB or more free disk space.

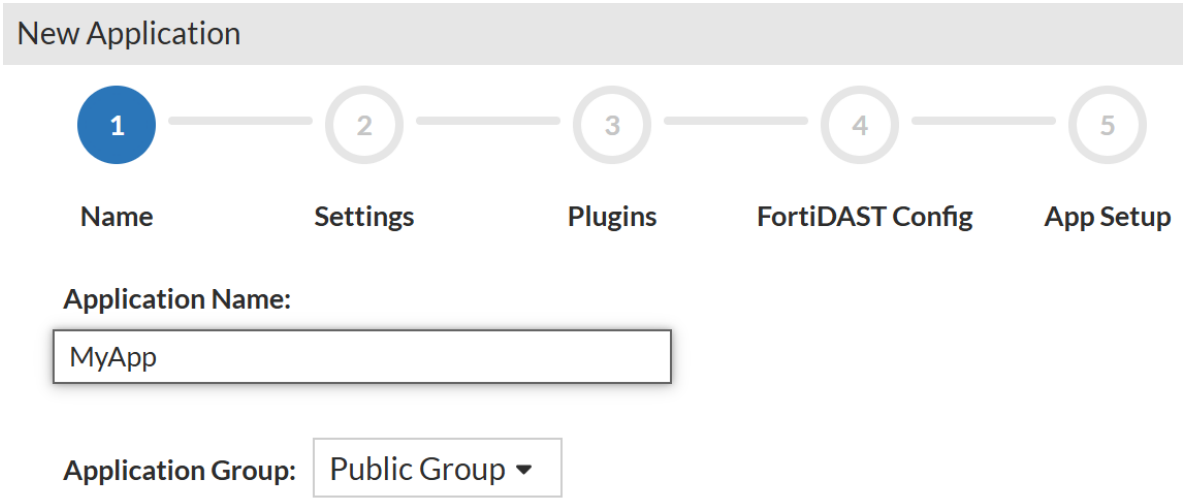
Notes:

- The minimum requirements are derived from an environment with an average repository size of 1 GB, a file count of 2,000, and a vulnerability count of 10,000.
- You might need to scale resource allocation based on the size and complexity of your repository.

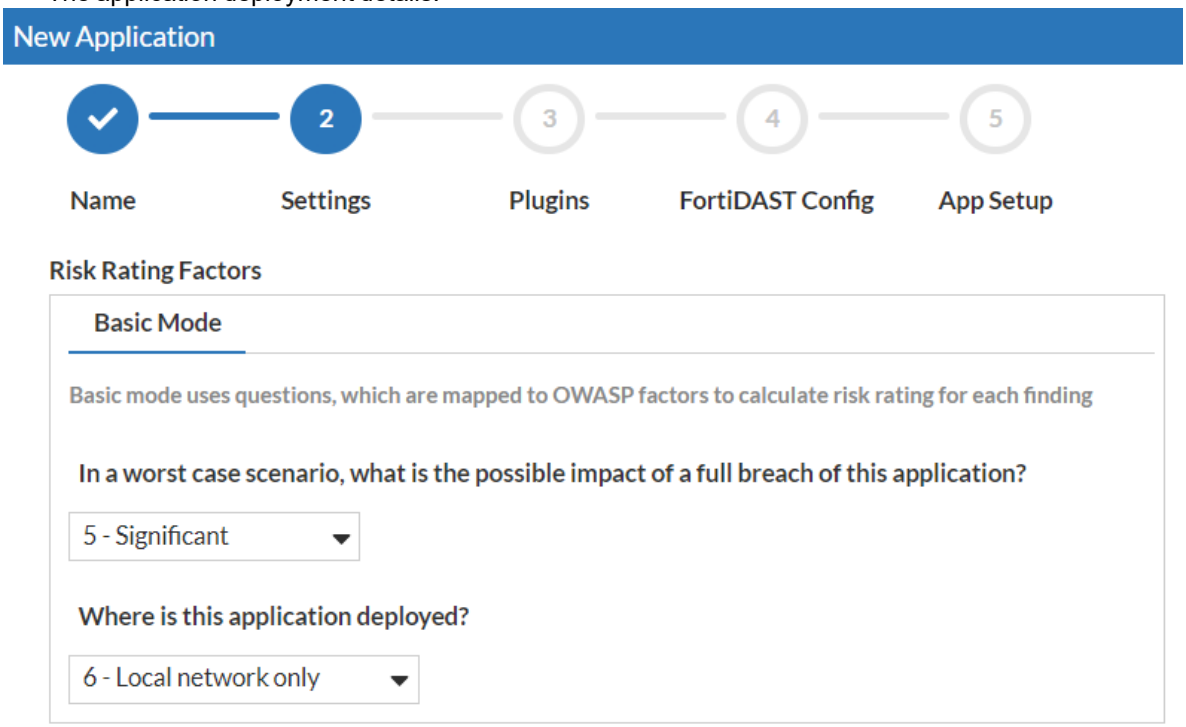
Adding a New Application

Adding your application in the FortiDevSec GUI to perform vulnerability scan testing.

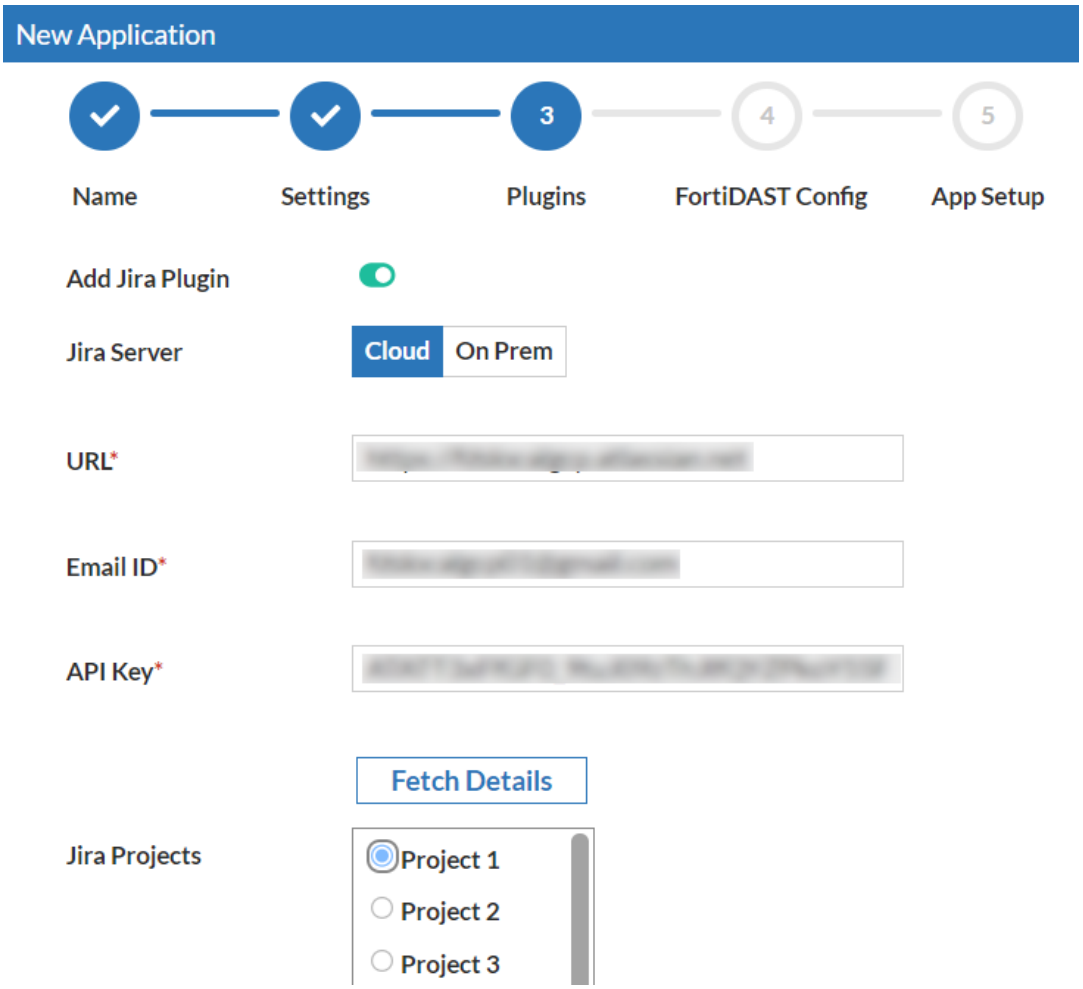
1. In **App Directory** page, click **+New Application**.
2. In **New Application** window, enter **Application Name** and select **Application Group** from the drop down. See Application Groups.

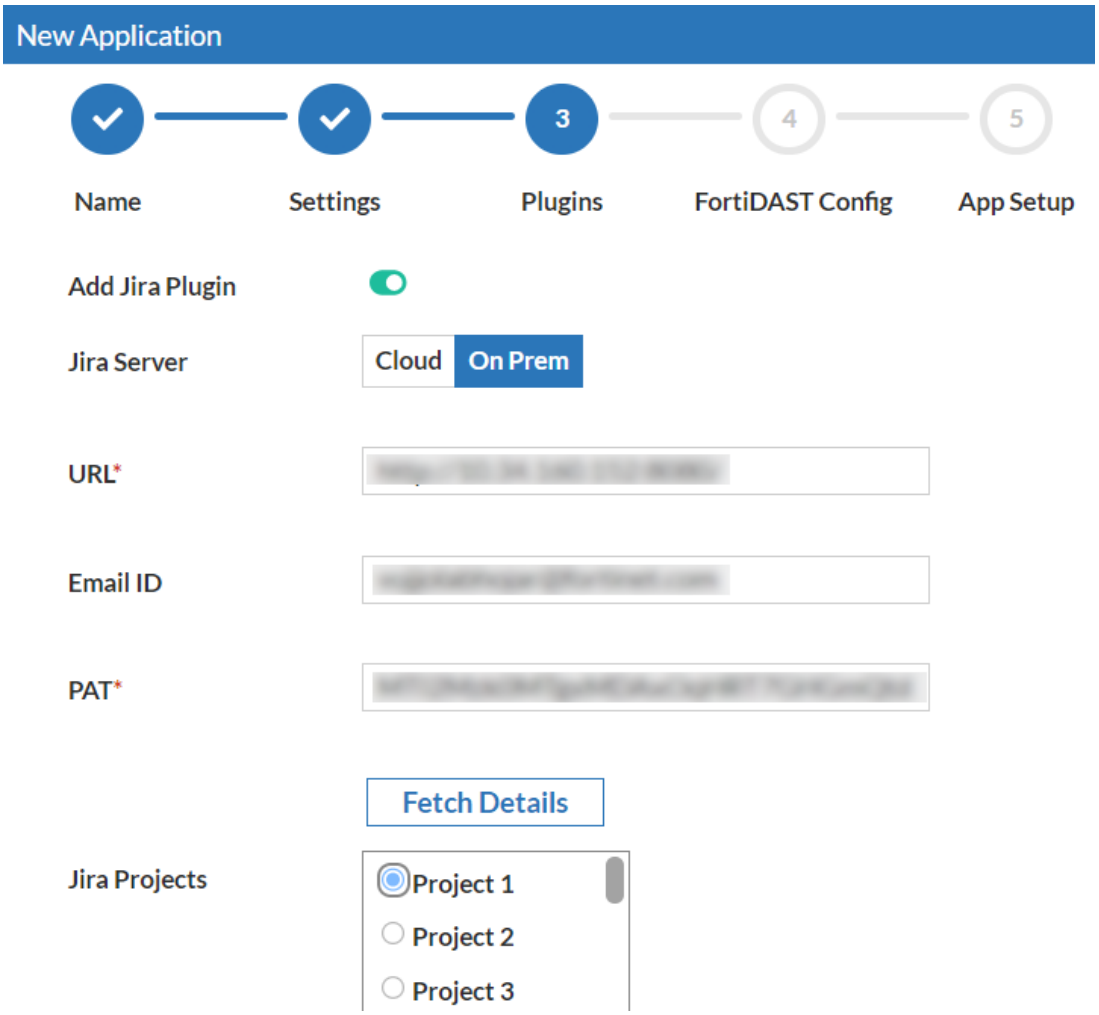


3. In the **Settings** panel you can configure the risk rating factors based on questionnaire for this application .
Note: If you do not configure the risk rating factors then the displayed default settings are applied.
The following data is associated with the OWASP factors to calculate risk rating for your application.
   - Possible impact in case of a full breach of this application.
   - The application deployment details.



4. Click **Next**.
5. You can enable and **Add Jira Plugin**, configure the Jira cloud server or the on-premise solution and select the project to integrate it with this application.

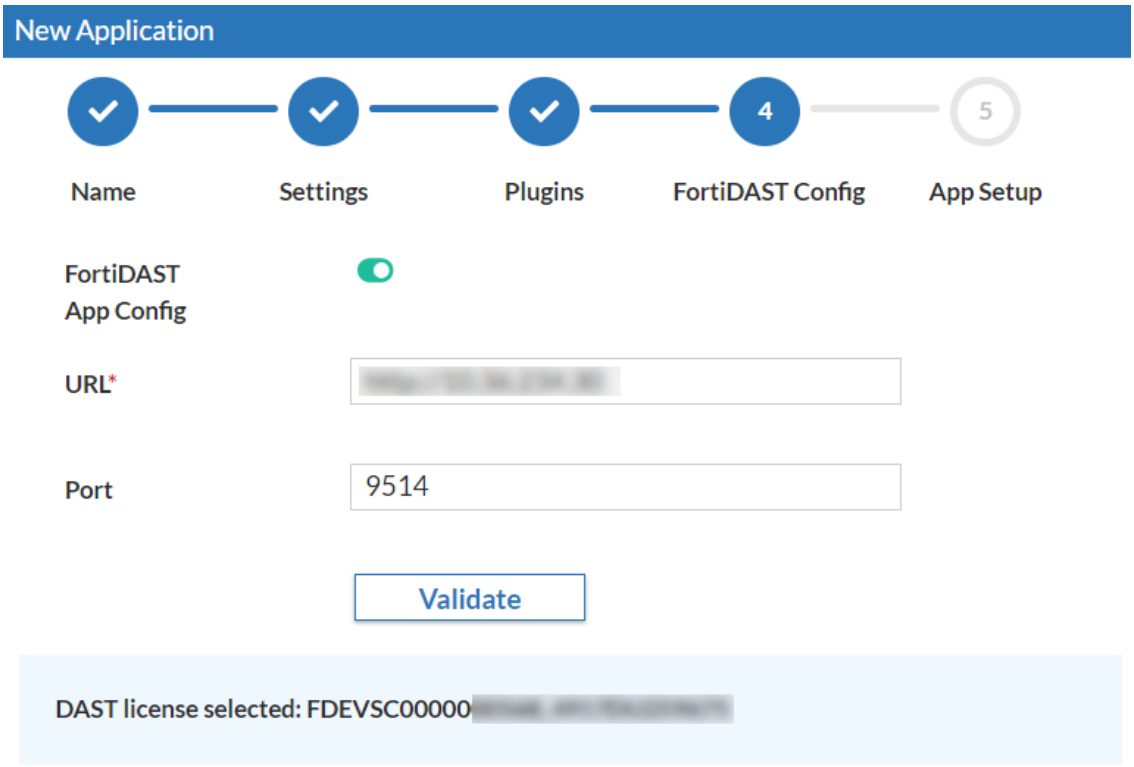

New Application
3
4
5
Name
Settings
Plugins
FortiDAST Config
App Setup
Add Jira Plugin
Jira Server
Cloud
On Prem
URL*
Email ID*
API Key*
Fetch Details
Jira Projects
Project 1
Project 2
Project 3



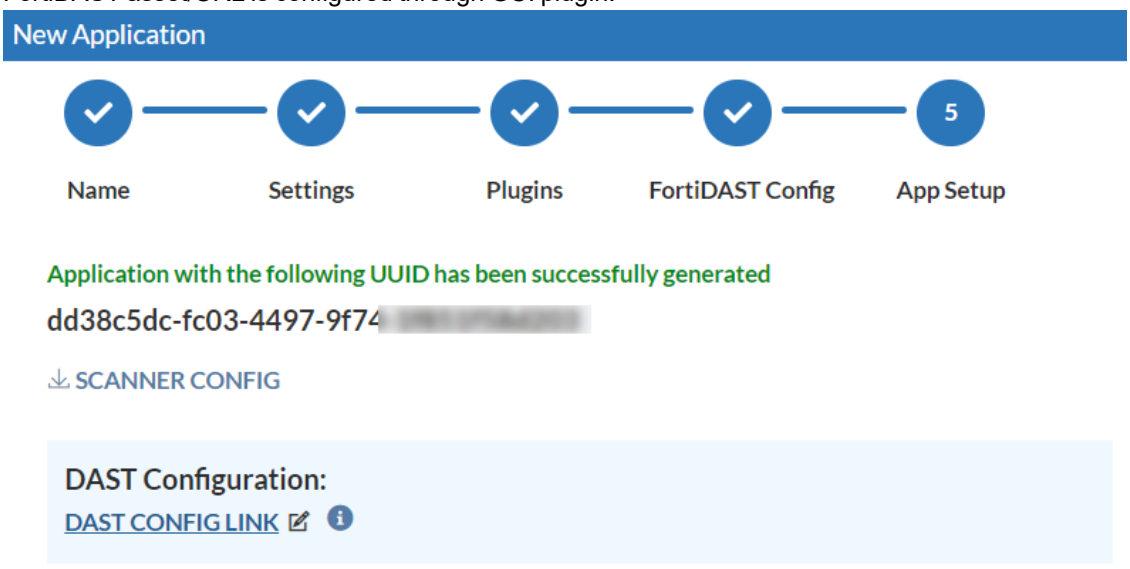
6. You can enable and configure FortiDAST scanning.
   a. Enter the target application **URL** and **port number**.
   b. Click **Validate**. FortiDevSec checks the entered URL format and verifies if the URL is already a target within FortiDAST.
      - If the URL exists, link to existing configuration is displayed.
      - If not, FortiDevSec adds the new URL as a target and associates it with a valid license.

      Once the URL and license is validated, the DAST license selected is displayed.

c. Click **Next**.



7. After an application is created, success message is displayed.
   - To configure the DAST scan in FortiDAST:
     i. Click **DAST Config Link**.
     ii. FortiDAST configuration page opens in a new tab.
     iii. Configure DAST scan as needed. See Configuring FortiDAST Scanner.
   - To download *fdevsec.yaml* file, click **Scanner Config**.
     Note: To perform DAST scan, uncomment the dast configuration in fdevsec.yaml file even when FortiDAST asset/URL is configured through GUI plugin.



Click **Done** and your application is listed in the dashboard.

Configuring the Scanner (fdevsec.yaml)

Check-in or add the *fdevsec.yaml* file into the root folder of the application source code.

Note: Do NOT modify the name and format of this file.

FortiDevSec automatically detects your application languages and runs the relevant scans. However, to run DAST scans additional parameters are required in *fdevsec.yaml*, these are described later on in this section. You can also optionally add advanced settings to *fdevsec.yaml* file as per your requirements.

You can also configure the scanner using command line arguments. See Command Line Arguments.

However, you can configure the scanner either by using the fdevsec.yaml file or command-line arguments, but not both simultaneously.

The following is a sample *fdevsec.yaml* file, the contents of this file vary based on different application scanning requirements.

Note: Ensure that proper indentation is maintained while configuring *fdevsec.yaml* file.

```
version: v1

id:
  org: 6a4d32db-6751-441a-88fe-9b4793717cde
  app: aa8a393b-afc6-47d7-84d2-b7011f1d0012

# Optional parameters.
scanners:
  - sast
  - dast
  - secret
  - sca
  - iac
  - container

languages:
  - python
  - javascript

exclude_path:
  - <directory path or name that must be excluded>

dast:
  url: <your.url.com>
  full_scan: true #true|false

resource:
```

```
  serial_scan: true #true|false

fail_pipeline:
  risk_rating: <1-9>
```

The following are the mandatory and optional parameters for *fdevsec.yaml*.

Parameter	Description
Mandatory parameters	
org	A unique ID associated with your organization.
app	A unique ID that identifies the applications within the organization.
Optional Parameters	
scanners	This identifies the type of scanner to test the applications. The supported values are **sast**, **dast**, **sca**, **secrets**, **iac**, and **container**. **Notes:** • If this parameter is unspecified, FortiDevSec runs only static scans. • To run DAST scan, use DAST image with the `url` parameter specified in the configuration file.
languages	Specify the language that you want to scan. The supported values are **java**, **javascript**, **python**, **golang**, **php**, **ruby**, **c++**, **shell**, **c#**, **c**, and **typescript**. FortiDevSec automatically detects the language if this parameter is not specified. **Notes:** • Specifying languages as javascript also scans NodeJS code. • If rails framework is not found in the source code repo, ruby scanner will not generate results.
exclude_path	Specify the directory path or name that must be excluded from the scan. Exclude path is supported for all scanners except *dast* and *container*.
dast	Specify these parameters if you intend running a DAST scan on your application. • `url` - The URL where your application is hosted. • `full_scan` - The supported values are **true** and **false**. The default value for `full_scan` is **true**. When set to **true**, a full DAST scan is run and when set to **false**, a basic scan is run. **Note**: You can configure the FortiDAST scanner with specific parameters for testing your asset (URL). For details on scanner configuration see the FortiDAST documentation.
resource	When `serial_scan` is set to **true**, the scans run consecutively and when set to **false**, multiple scans run parallel. The default value of `serial_scan` is **true**.

Parameter	Description
fail_pipeline	Specify the `risk_value` parameter if you intend to fail CI/CD pipeline based on your risk tolerance level. If the resulting risk rating value after scan is greater than or equal to the defined value, the CI/CD pipeline fails. The CI/CD pipeline tool will automatically detect the failure and will stop the pipeline process. • `risk_value` - The supported value is a number in the range of **1–9**; 1 indicates the lowest and 9 the highest risk rating level.

Running the Security Scan

You can automate a security scan on your application in a CI/CD environment or run a security scan for your application manually in your source code through the CLI terminal.

- Automated Scanning
- Manual Scanning
- Downloading the Required Language Scanners

The FortiDevSec SAST/DAST scanner Docker image is built for Linux. To run the FortiDevSec Linux-based Docker container on Windows, you must install Docker Desktop, which supports **Windows Subsystem for Linux 2 (WSL 2)**.

Automated Scanning

You can integrate scan configurations into your CI/CD tool and automate the application scan testing for the following. Ensure that *fdevsec.yaml* file is checked into the root folder of your source code. See CI/CD Tools.

Manual Scanning

Ensure the *fdevsec.yaml file* is added to the root directory of your source code before initiating a manual scan. See Configuring the Scanner (fdevsec.yaml).

To run a scan **manually**, navigate to the root folder of the source code and run the following command.

```
docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_sast:latest
```

In this command a SAST (`/fdevsec_sast:latest`) scan is run, modify the value to DAST (`/fdevsec_dast:latest`) if required.

```
docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_dast:latest
```

Notes:

- Verify DAST configuration in *fdevsec.yaml* file before perfoming the DAST scan.
- The SAST scanner docker image is bundled with SCA, Secret, IaC and Container scanners.
- Scanner docker images must be updated using `docker pull <image>` command to the latest version to use the latest features.
- If not configured using GUI plugin, the DAST/FortiDAST asset scan configuration details can only be added by logging in to FortiDAST after performing the initial scan.
- APP ID and ORG ID must not be modified when scan is in progress.
- FortiDevSec container scans currently cannot scan private images requiring Docker login.

The following image depicts a sample command for SAST.

```
devopsuser@User1:~/Repos/OWASPBenchmark$ docker run --pull always --rm --mount
type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
latest: Pulling from fdevsec_sast
Digest: sha256:8419af98214170eb2dfe7dfbdbc99d4b4b51447a14e7f184aac297ff3e47aef1
Status: Image is up to date for registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
registry.fortidevsec.forticloud.com/fdevsec_sast:latest
2022/02/03 06:33:57 Loaded scan config for Org ID: d9d3dc20-9372-4188-
884fb18a5c75fe5c
2022/02/03 06:33:57 Languages configured in conf file: [java]
2022/02/03 06:34:02 Scanners configured in conf file: [sast]
2022/02/03 06:34:03 Total enabled scanners: 1
2022/02/03 06:34:03 Running parallel scan as per user config.
Scanning Progress: [##################################################] 100% 1/1
2022/02/03 06:37:25 FortiDevSec SAST scanner done, exiting.
```

The following image depicts a sample command for DAST.

```
devopsuser@Dev:~/Repo/OWASPBenchmark$ docker run --pull always --rm --mount
type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
latest: Pulling from fdevsec_dast
Digest: sha256:8419af98214170eb2rgt3fbdbc99d4b4b51447a14e7f184aac297ff3e47aef1
Status: Image is up to date for registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
2022/02/03 08:37:19 Loaded scan config for Org ID: d9d3dc20-9372-4188-884f-
b18a5c75fe5c
2022/02/03 08:37:19 Scanners configured in conf file: [dast]
2022/02/03 08:37:20 Response Status: 202 Accepted
2022/02/03 08:37:20 Total enabled scanners: 0
2022/02/03 08:37:20 No scanners specified.
2022/02/03 08:37:20 FortiDevSec DAST scanner done, exiting.
```

Downloading the Required Language Scanners

FortiDevSec uses multiple language scanner images to scan your application, you can optionally download these scanner image files on your machine based on the configured or detected languages when you run a scan. This reduces the overhead of downloading all scanner images each time FortiDevSec scans your application. Create a directory on your machine and grant full access. Consider the following example.

```
mkdir scanner_downloads
chmod 777 scanner_downloads
```

Run the following command to scan your application (indicating the directory) and download the scanner images. This example indicates the directory **`scanner_downloads`** created earlier.

```
docker run --pull always -ti --rm --mount type=bind,source="$(pwd)",target=/scan
--mount type=bind,source="$(pwd)"/scanner_downloads,target=/scanner
registry.fortidevsec.forticloud.com/fdevsec_sast:latest
```

Note: Replace the **scanner_downloads** directory in the above command with the name of the newly created directory where you have downloaded the language scanner images.

Command Line Arguments

The FortiDevSec allows you to configure scanner by providing command line arguments as an alternative to configuring yaml file. To pass command line arguments to the FortiDevSec scanner, append `main s` to the scanner run command, followed by the desired arguments.

You can configure the scanner either by using the fdevsec.yaml file or command-line arguments, but not both simultaneously.

Command Format:

- SAST scan: docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_sast:latest **main s** --arg1 value1 --arg2 value2
- DAST scan: docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_dast:latest **main s** --arg1 value1 --arg2 value2

Example:

Following is an example command to run a SAST scan with the specified arguments.

```
docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_sast:latest main s --org-id 9df9dc0f-
0000-4cf7-8d91-81e194fdd727 --app-id 613a0004-b08f-40e1-a5c8-6702f2b5027b -l
python -l java -l c -S=true --scanner sast --scanner sca
```

where,

- `docker run --pull always --rm --mount type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_sast:latest` is a command to run a SAST scan from the application root folder.
- `main s` is used to enable arguments.
- `--org-id 9df9dc0f-0000-4cf7-8d91-81e194fdd727` specifies the organization for the scan.
- `--app-id 613a0004-b08f-40e1-a5c8-6702f2b5027b` specifies the application for the scan.
- `-l python -l java -l c` configures the scanner to analyze Python, Java, and C languages.
- `-S=true` enables serial scan mode.
- `--scanner sast --scanner sca` specifies the use of both the SAST and SCA scanners.

Notes:

- The command provided is an example and may need adjustments based on your specific environment and requirements.
- Arguments are case-sensitive.

- Multiple arguments can be combined in a single command.
- `--org-id` and `--app-id` are mandatory arguments.

Available Arguments:

The following are the mandatory and optional arguments. Replace **value** with actual value based on your environment. Run **--help** command for a comprehensive list of available arguments and their descriptions.

Argument Format	Description
Mandatory arguments	
--org-id value or **-o** value	A unique ID associated with your organization.
--app-id value or **-a** value	A unique ID that identifies the applications within the organization.
Optional arguments	
--scanner value or **-s** value	This identifies the type of scanner to test the applications. The supported values are **sast**, **dast**, **sca**, **secrets**, **iac**, and **container**. **Notes:** • If this parameter is unspecified, FortiDevSec runs only static scans. • To run DAST scan, use DAST image with the `url` parameter specified in the configuration file.
--language value or **-l** value	Specify the language that you want to scan. The supported values are **java**, **javascript**, **python**, **golang**, **php**, **ruby**, **c++**, **shell**, **c#**, **typescript**, and **c**. FortiDevSec automatically detects the language if this parameter is not specified. **Note:** Specifying languages as javascript also scans NodeJS code.
--exclude_path value or **-e** value	Specify the directory path or name that must be excluded from the scan. Exclude path is supported for *Golang* and *Python* languages.
--url value or **-u** value **--fullscan** value or **-f** value	Specify these arguments if you intend running a DAST scan on your application. • `--url` - The URL where your application is hosted. • `--fullscan` - The supported values are **true** and **false**. The default value for `--fullscan` is **true**. When set to **true**, a full DAST scan is run and when set to **false**, a basic scan is run. **Note**: You can configure the FortiDAST scanner with specific parameters for testing your asset (URL). For details on scanner configuration see the FortiDAST documentation.
--serial-scan value or **-S** value	When `--serial-scan` is set to **true**, the scans run consecutively and when set to **false**, multiple scans run parallel. The default value of `--serial-scan` is **true**.

Argument Format	Description
--risk_rating value or **-r** value	Specify the `--risk_value` argument if you intend to fail CI/CD pipeline based on your risk tolerance level. If the resulting risk rating value after scan is greater than or equal to the defined value, the CI/CD pipeline fails. The CI/CD pipeline tool will automatically detect the failure and will stop the pipeline process. • `--risk_value` - The supported value is a number in the range of **1–9**; 1 indicates the lowest and 9 the highest risk rating level.

Viewing the Scan Result

The FortiDevSec scan result for application vulnerability scanning is populated in a comprehensive *App Directory* dashboard. To view scanned application results see App Directory.

App Directory

The FortiDevSec *App Directory* dashboard that provides an insight into the scanned applications categorizing the findings based on the scanners used with the calculated risk rating indicators. You can view the detected vulnerabilities' details per application. A **Sample** application is generated on the GUI and first time product users are guided by an interactive product tour, to discover the product and its configurations.

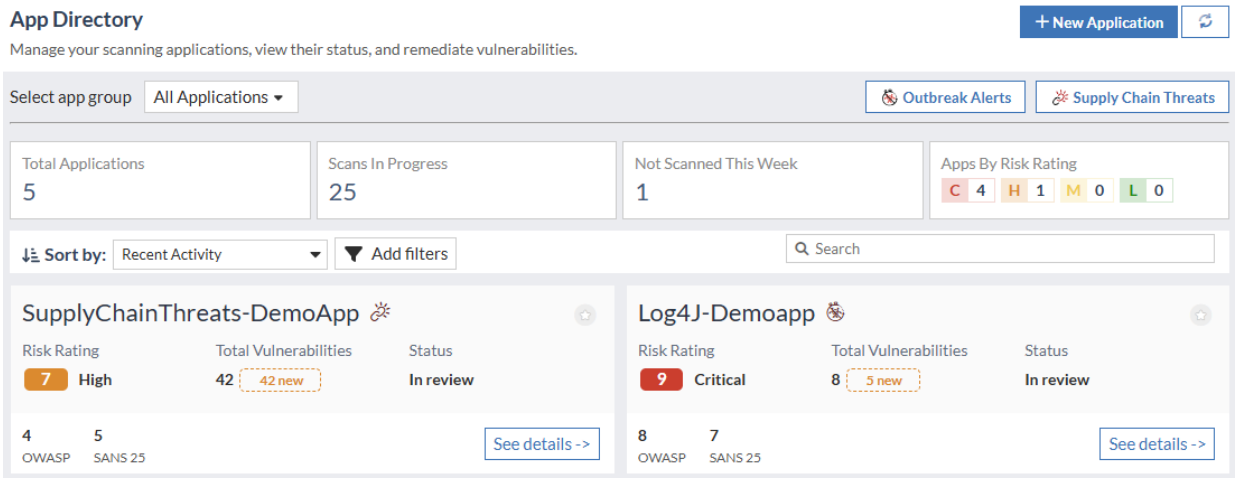
The **Supply Chain Threats** includes the list of all supply chain threats detected. The **Outbreak Alerts** includes the list of all the FortiGuard outbreak alerts identified after performing the application scan.

The widgets provide a summary of scanned applications, including the following information.

- *Total applications*: The total number of applications scanned.
- *Scan In Progress*: The number of applications currently being scanned.
- *Not Scanned This Week*: The number of applications that have not been scanned in the current week.
- *Apps By Risk Rating*: The number of applications categorized by each risk rating. Click count displayed next to the risk rating severity to filter the applications based on the severity.

You can perform the following actions in *App Directory* page.

- Use the **Sort by** dropdown to sort applications by different criteria.
- Click **Add filters** to filter the list of applications based on various criteria. See Filtering Applications.
- Use the **Search** field to find specific applications.
- Click **+New Application** to add a new application to FortiDevSec. See Adding a New Application.
- Click refresh icon to manually refresh *App Directory* page data.
- Use **Select app group** dropdown to filter applications based on their assigned application groups. See Application Groups.



- Viewing Supply Chain Threats
- Viewing Outbreak Alerts
- Filtering Applications
- Viewing Scanned Applications

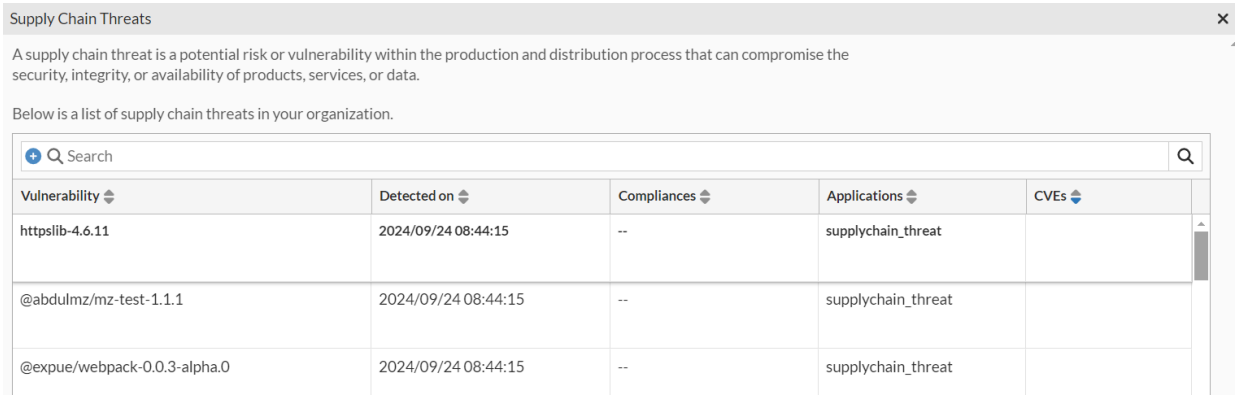
Viewing Supply Chain Threats

Supply chain threats refer to security risks that arise from vulnerabilities present in the components, software libraries, or dependencies used in the development of an application. These threats can be exploited by attackers to compromise the overall security of the application and potentially gain unauthorized access to sensitive data or systems.

Click **Supply Chain Threats** in the *App Directory* page to view the all the supply chain threat alerts. The following fields are displayed for each supply chain threat.

- *Vulnerability*: The name of the specific vulnerability detected.
- *Detected on*: The date and time when the threat was first identified.
- *Compliances*: Indicates whether the vulnerability is related to OWASP or SANS standards.
- *Applications affected*: A list of applications that are impacted by the vulnerability.
- *CVEs*: A list of Common Vulnerabilities and Exposures (CVEs) associated with the threat. Click on any CVE to view the corresponding page on the NVD (National Vulnerability Database) website.

Note: FortiDevSec SCA currently detects Supply Chain threats only from *Python OSS* ecosystems.



Viewing Outbreak Alerts

FortiGuard Outbreak Alerts provides key information about on-going cybersecurity attack with significant ramifications affecting numerous companies, organizations and industries.

Click **Outbreak Alerts** in the *App Directory* page to view the all the FortiGuard outbreak alerts detected. The following fields are displayed for each outbreak alert.

- *Severity*: The level of risk associated with the attack.
- *Vulnerability*: The name of the specific vulnerability detected. Click the vulnerability name to access the corresponding FortiGuard page for more details.
- *Detected on*: The date and time when the outbreak was first detected.
- *Applications*: A list of applications that are affected by the vulnerability.
- *CVEs*: A list of Common Vulnerabilities and Exposures (CVEs) related to the outbreak. Click on any CVE to view the corresponding page on the NVD (National Vulnerability Database) website.

Note: Outbreak alerts is currently supported only for SCA and Container scanners.

Filtering Applications

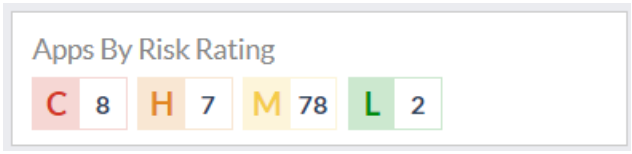
You can filter the applications listed in the *App Directory* based on various criteria, allowing you to focus on specific sets of applications for review.

- Filtering by Risk Rating
- Filtering by Additional Criteria

Filtering by Risk Rating

The Apps by Risk Rating widget on the App Directory dashboard provides a quick overview of applications categorized by their risk rating (Critical, High, Medium, Low). You can use this widget to filter applications by risk level.

1. Go to *App Directory* > *Apps by Risk Rating* widget.
2. Click count displayed next to the desired risk rating severity(*C - Critical*, *H - High*, *M - Medium*, *L - Low*). This filters the application list to only show applications with that specific risk rating severity.
3. To disable the filter, click on the previously selected risk rating again.



Filtering by Additional Criteria

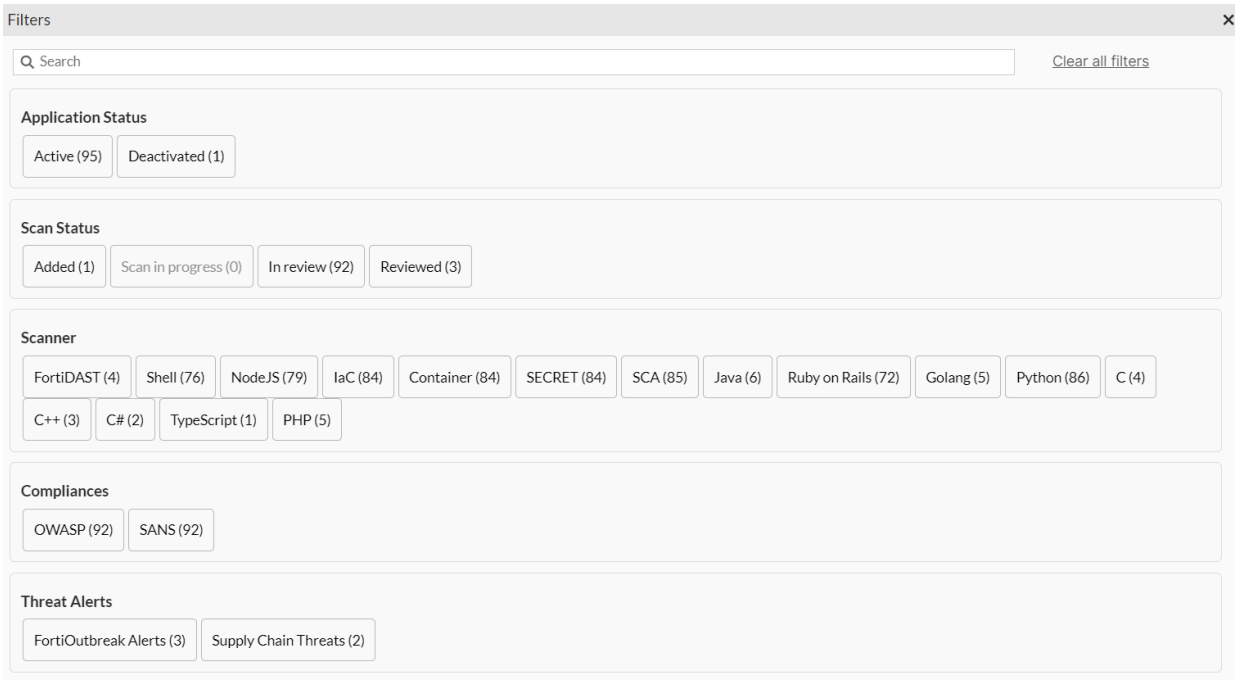
For more granular filtering options, click *Add filters* on the *App Directory* page. The *Filters* window displays the following filters.

- *Application Status*: Filter applications based on their current status.
- *Scan Status*: Filter applications based on their scan status.
- *Scanner*: Filter applications based on the scanner used during the last scan.
- *Compliances*: Filter applications based on compliance frameworks such as *OWASP* or *SANS*.
- *Threat Alerts*: Filter applications based on the presence *FortiOutbreak Alerts* or *Supply Chain Threats*.

Within each filter section, select the desired options to narrow down the application list. Also, you can use the *Search* field to find a specific filter.

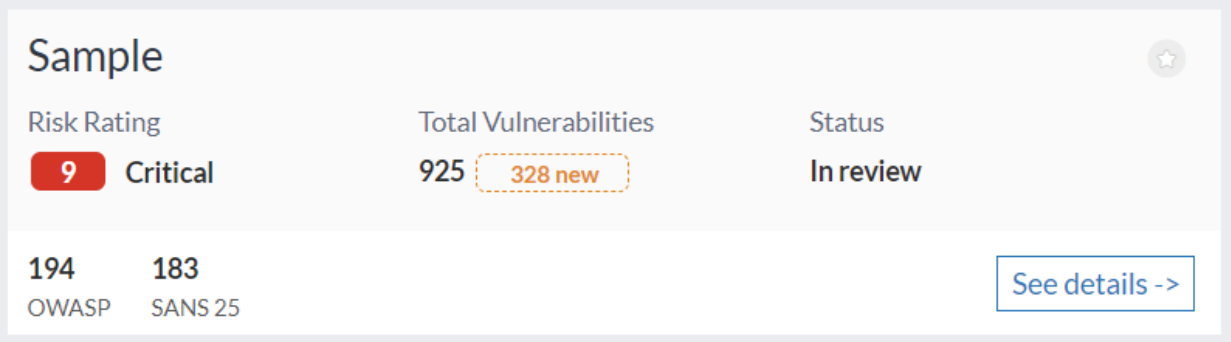
Once you have selected the filter/s, click the *OK* to apply the filters and update the displayed applications.

To remove all applied filters and return to the unfiltered application list, click the *Clear all filters.*



Viewing Scanned Applications

The application panel lists all the scanned applications with basic details.



You can analyze the following information specific to each application.

- The risk rating assigned by FortiDevSec for this application.
- The total number of vulnerabilities detected and the number of new vulnerabilities detected since the last scan.
- The vulnerability counts for both *OWASP* and *SANS* categories.
- The current status of the application.
- The presence of an supply chain threat alert icon indicates that the application has a supply chain vulnerability that requires attention.
- The presence of an outbreak alert icon indicates that the application has vulnerability that requires immediate attention.

Click *See details* to view scan details.

Viewing Scanned Application Details

In this page details such as, the scanner types used with a break-up of the number of vulnerabilities found by each scanner and the associated risk rating are displayed. In this example, there are a total of 925 vulnerabilities found and categorized based on the scanners that detected them.

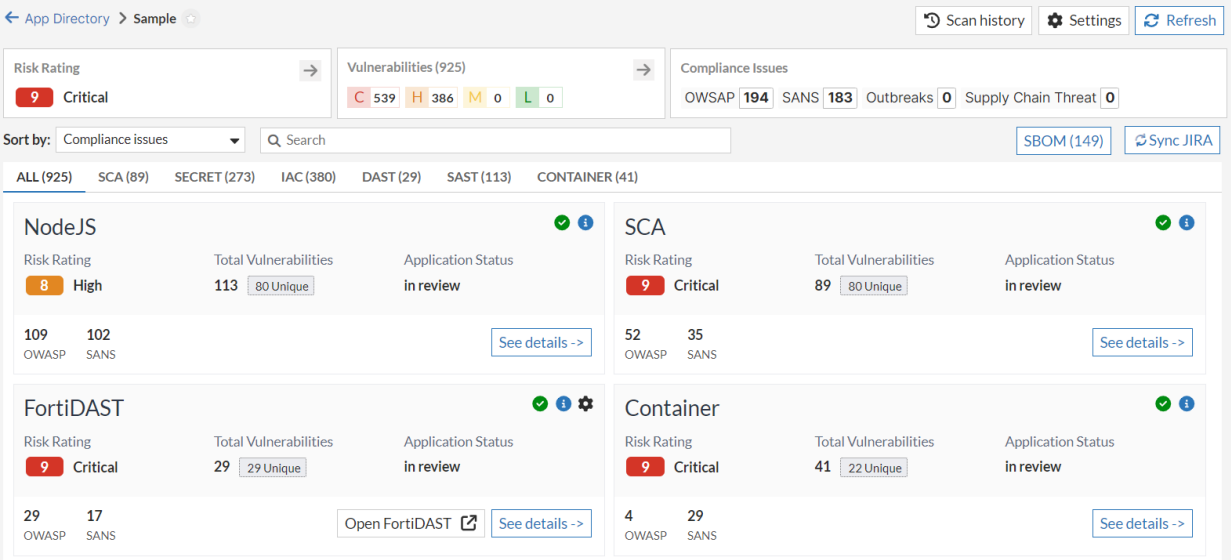
The widgets provide a summary of the selected application, including the following information.

- *Risk Rating*: The overall risk rating for the selected application. Click arrow to modify the risk rating settings for the selected application.
- *Vulnerabilities*: The total number of vulnerabilities and vulnerabilities categorized by severity. Click count displayed next to the vulnerability severity to filter the applications based on the severity. Click arrow to view all the detected vulnerabilities in *Vulnerability Catalog* page for the selected application. See Vulnerability Catalog.
- *Compliance Issues*: The number of vulnerabilities detected for each category including *OWASP*, *SANS*, *Outbreaks* and *Supply Chain Threats*.Click desired category count to view additional details.

You can perform the following actions in *App Directory* page.

- Click **Scan history** to view the scan history of the application such as the type of scanners used for various scans, the scan duration, total number of vulnerabilities found, and the associated risk.
- Click **Settings** to view and modify the selected application details. See Viewing Application Details.
- Click **Refresh** icon to manually refresh *App Directory* page data.

- Use the **Sort by** dropdown to sort scanners by different criteria.
- Use the **Search** field to find specific scanner.
- View scanner details. See Viewing Scanner Details.
- Click **SBOM** to view the Software Bill of Materials for the selected application.
- Click **Sync JIRA** to manually synchronize the vulnerabilities from the JIRA plugin.



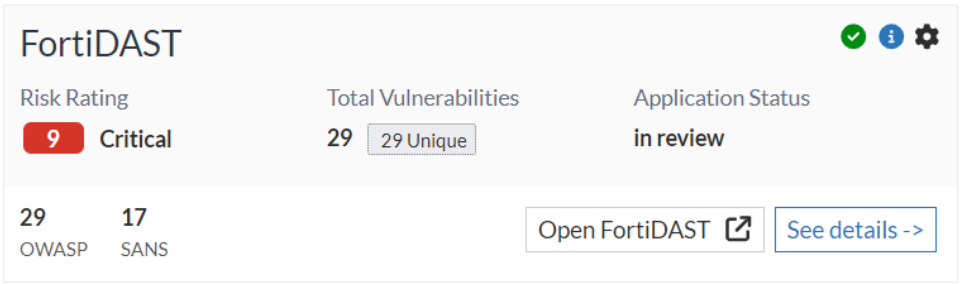
Viewing Scanner Details

For each scanner type in scanned application details page, you can analyze the following information.

- The risk rating assigned by FortiDevSec for this scanner.
- The total number of vulnerabilities detected and the number of unique vulnerabilities detected since the last scan.
- The vulnerability counts for both *OWASP* and *SANS* categories.
- The current status of the selected application.
- The presence of an supply chain threat alert icon indicates that the application has a supply chain vulnerability that requires attention.
- The presence of an outbreak alert icon indicates that the application has vulnerability that requires immediate attention.
- Hover over to view CI/CD and build related information.

Click **See details** for any scanner type to view detailed vulnerability information on the *Vulnerability Catalog* page. The selected application and scanner type will be applied as filters to display relevant vulnerabilities. See Vulnerability Catalog.

In FortiDAST scanner details, click gear icon to view the configuration page or click **Open FortiDAST** to view the scan results page in *FortiDAST* portal.



Viewing Application Details

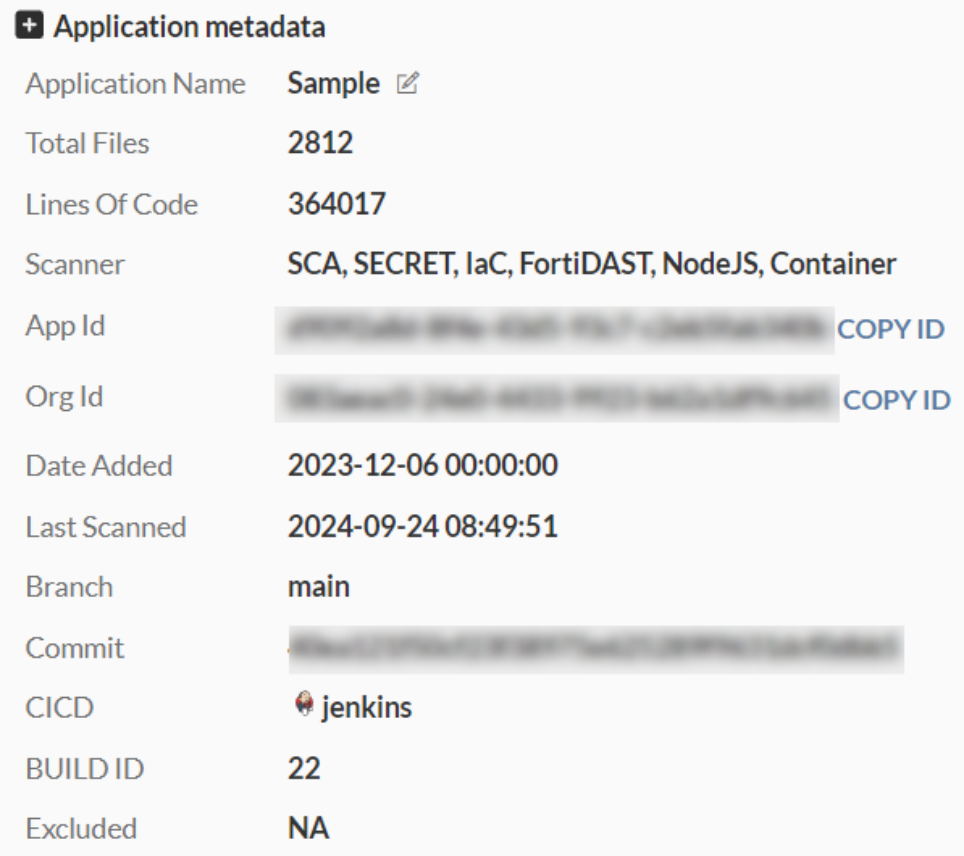
The Application details window displays detailed application information including the following sections.

- Application Metadata
- Application Plugins
- Actions

Application Metadata

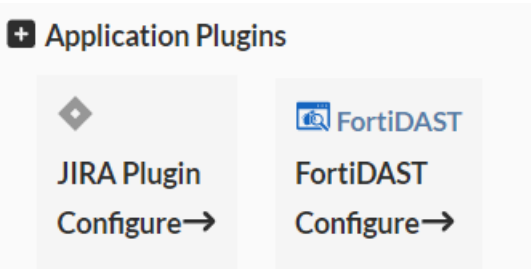
This section displays key data about the application:

- *Application name*: The name assigned to the application. Click edit icon to edit the name.
- *Total files*: The total number of files scanned within the application.
- *Lines of code*: The total number of lines of code analyzed within the application.
- *Scanner*: The types of scanners used to analyze the application (e.g., SCA, SECRET, IaC, FortiDAST, NodeJS, Container).
- *App ID*: A unique identifier assigned to the application. Click **COPY ID** to copy the APP ID.
- *Org ID*: The identifier of the organization to which the application belongs.
- *Date added*: The date and time when the application was first added to the system.
- *Last scanned*: The date and time of the most recent scan performed on the application.
- *Branch*: The specific branch of the code repository from which the application was analyzed.
- *Commit*: The unique identifier of the commit within the code repository that corresponds to the scanned version of the application.
- *CICD*: The CI/CD pipeline used for building and deploying the application.
- *BUILD ID*: The identifier of the specific build within the CI/CD pipeline that was scanned.
- *Excluded*: Any file paths or directories that were excluded from the scan.



Application Plugins

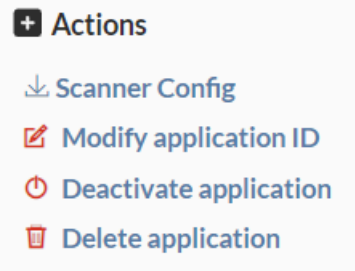
This section allows you to enable and configure JIRA and FortiDAST scan target with FortiDevSec, click **Plugins**. See Plugins.



Actions

This section provides options for managing the application.

- **Download configuration file** - Click **Scanner Config** to download the *fdevsec.yaml* file.
- **Modify App ID** - You can modify the application ID, the new ID is displayed in this **Details** panel instantly. Ensure that you update the modified application ID in any existing *fdevsec.yaml* file.
- **Deactivating/Deleting the Application** - You can deactivate an application wherein no modification is allowed to the application vulnerability findings, but you are allowed to view them. You can delete an application (that is not being scanned) from the dashboard only after deactivating it.



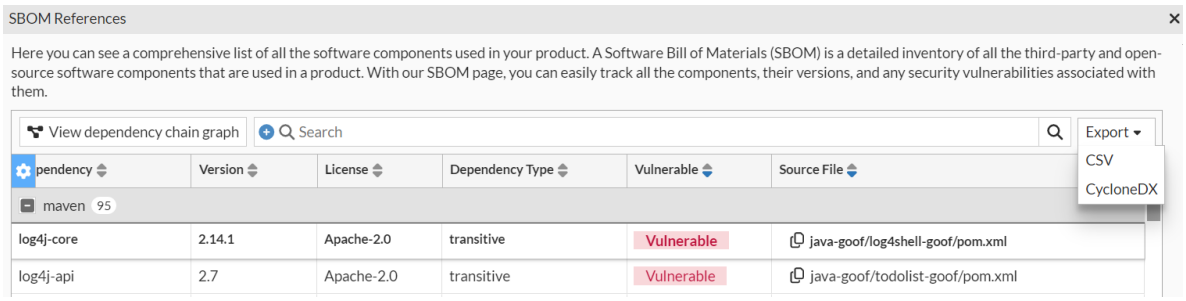
Viewing Software Bill of Materials(SBOM)

A Software Bill of Materials (SBOM) is a detailed inventory that includes all the third-party and open-source software components used in the product. FortiDevSec SBOM references page presents a complete list of all the software components used in your product and helps you easily track these components, their versions, and any security vulnerabilities they may have.

Perform the following steps to view SBOM.

1. In the FortiDevSec *Dashboard* > *Applications*, click **See details** in the desired application which contains secret scan to view scan details.
2. In scanned application details page, click **SBOM**.
3. **SBOM References** window is displayed. The components are grouped based on their ecosystem and the following fields are displayed for each component.

Field	Description
Dependency	The name of the third party library being used.
Version	The version of the library being utilized.
License	Displays license information for the dependency. Licenses with known risks are highlighted.
Vulnerable	Notifies whether the library is vulnerable or non-vulnerable.
Source File	The file path where the library name and version are mentioned and utilized.



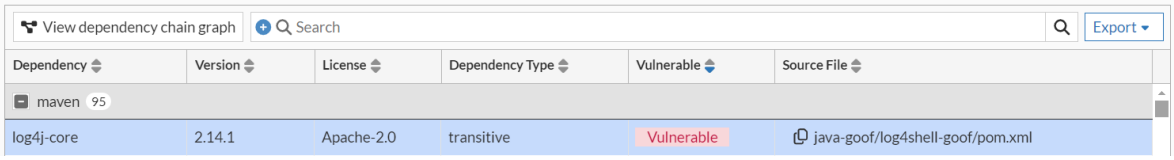
4. Click **Export** and choose **CSV** to save the list of all components in a Microsoft Excel file. Alternatively, select **Cyclone DX** to export the list in the *Cyclone DX JSON* format.

Viewing Dependency Chain Graph

The Dependency chain graph window offers a comprehensive view of your software component's dependency relationships. Analyze both direct and transitive dependencies to identify potential issues.

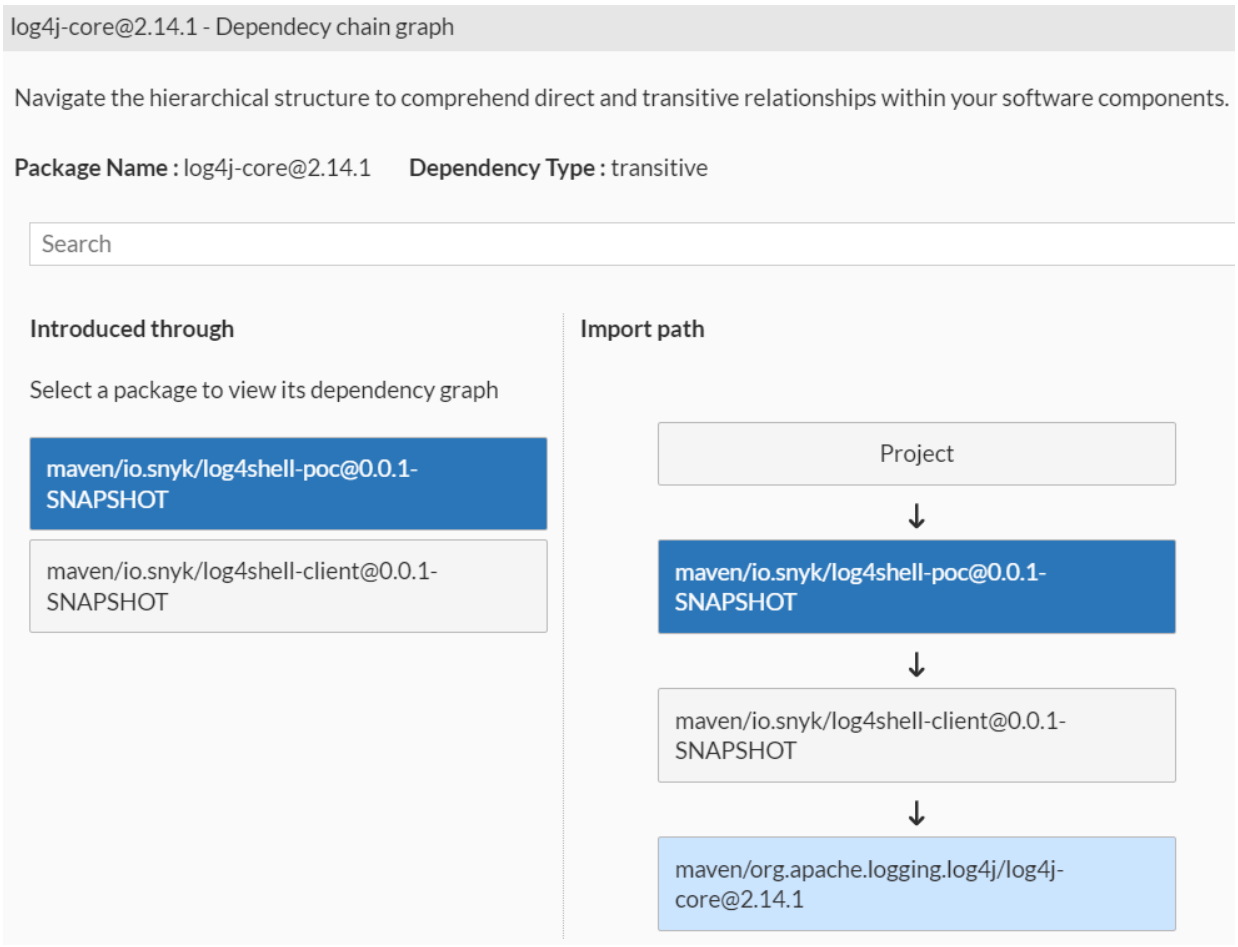
Perform the following steps to view dependency chain graph.

1. In the FortiDevSec*Dashboard* > *Applications*, click the desired application name or the number of vulnerabilities which contains secret scan to view scan details.
2. Click **SBOM** in the SCA scanner widget. **SBOM References** window is displayed.
3. Select the software component and click **Dependency graph**.



4. The following information is displayed in Dependency chain graph window.

Field	Description
Package Name	Package name of the selected software component.
Dependency type	Type of dependency, **transitive** or **direct**. If the selected software component contains both direct and transitive dependency then **transitive** will be displayed as dependency type. A **direct** dependency is a component that you directly reference in your code. A **transitive** dependency is a component that your selected component indirectly relies on through its direct dependencies.
Introduced through	Provides information about the path through which a package was introduced.
Import path	Dependency chain graph for the selected package from the **Introduced through** section.


log4j-core@2.14.1 - Dependecy chain graph
Navigate the hierarchical structure to comprehend direct and transitive relationships within your software components.
Package Name : log4j-core@2.14.1
Dependency Type : transitive
Search
Introduced through
Select a package to view its dependency graph
maven/io.snyk/log4shell-poc@0.0.1-SNAPSHOT
maven/io.snyk/log4shell-client@0.0.1-SNAPSHOT
Import path
Project
maven/io.snyk/log4shell-poc@0.0.1-SNAPSHOT
maven/io.snyk/log4shell-client@0.0.1-SNAPSHOT
maven/org.apache.logging.log4j/log4j-core@2.14.1

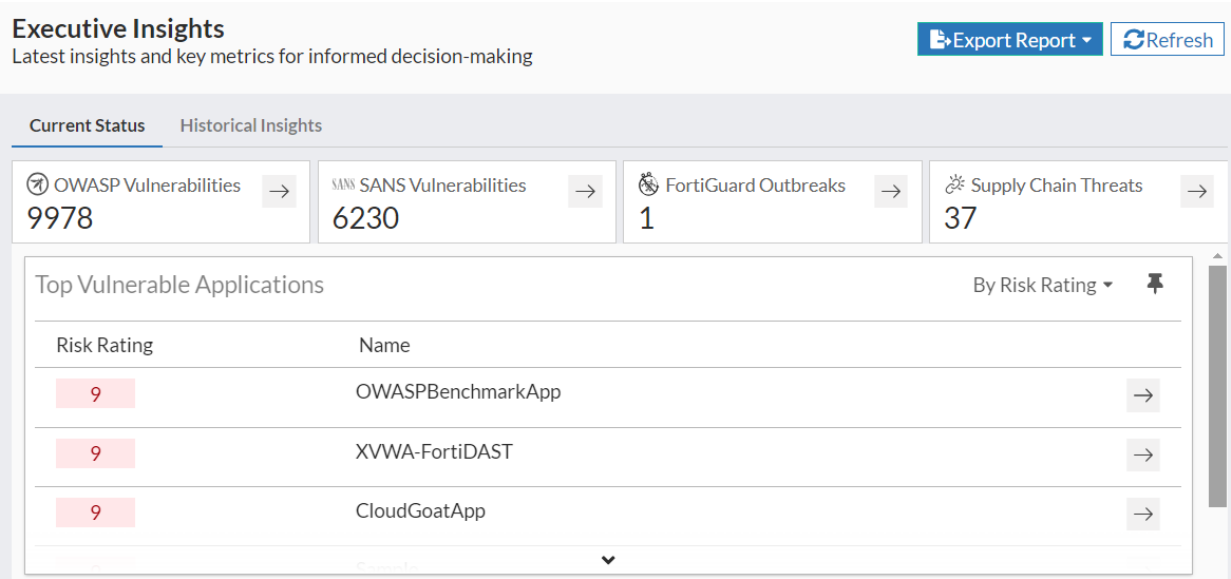
Analytics

The *Analytics* section provides insights into your application security posture. You can view overall trends and vulnerabilities at the organization level and drill down into specific applications for more detailed analysis.

Note: Perform rescan for the previously scanned applications to ensure all the analytics data is displayed correctly.

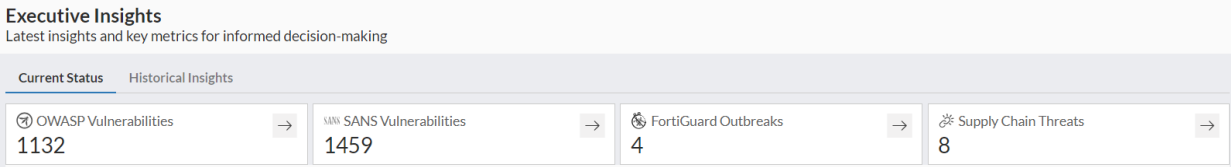
Organization Level

The *Organization Level* analytics page allows you to gain a high-level overview of security risks across all applications within your organization. You can use this page to view current vulnerabilities, track historical trends, and evaluate the overall health of your applications security. Click *Refresh* to manually update the analytics data.



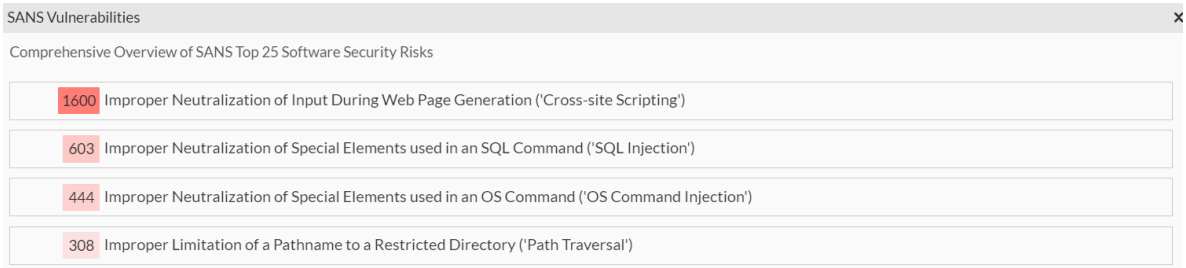
Current Status

The *Current Status* tab displays real-time data on your organization's security posture. You can identify the most prevalent vulnerabilities, assess the risk distribution of your applications, and get a quick overview of your remediation efforts. The following information is displayed.

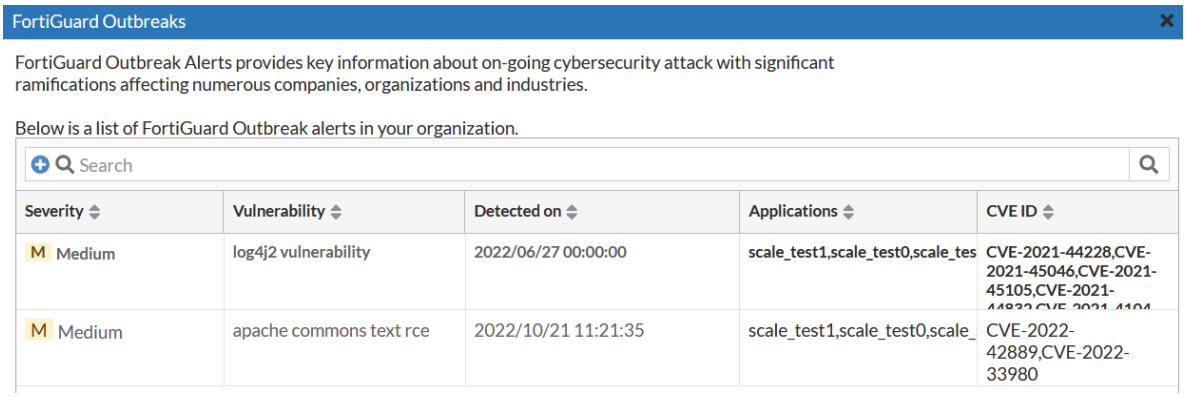


- **OWASP Vulnerabilities** - Displays the total number of vulnerabilities detected across your organization that align with the OWASP Top 10 list. Click arrow to view detailed information.

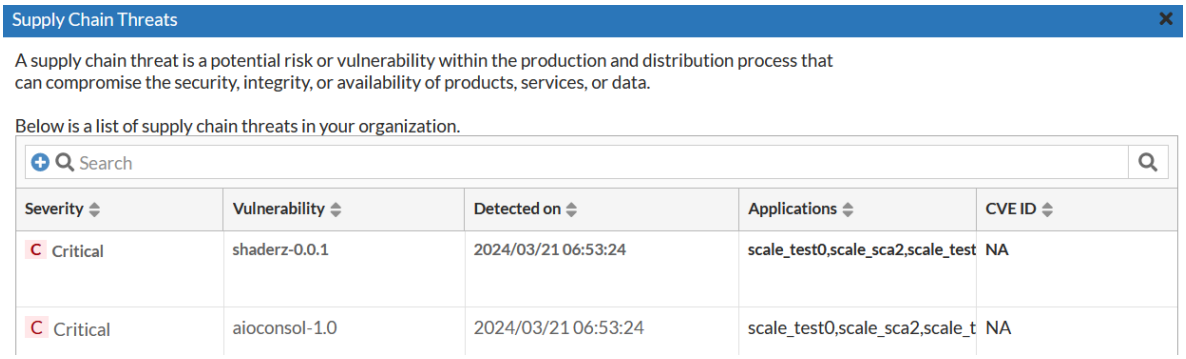
- **SANS Vulnerabilities**- Displays the total number of vulnerabilities detected across your organization that align with the SANS Top 25 list. Click the arrow to view detailed information.



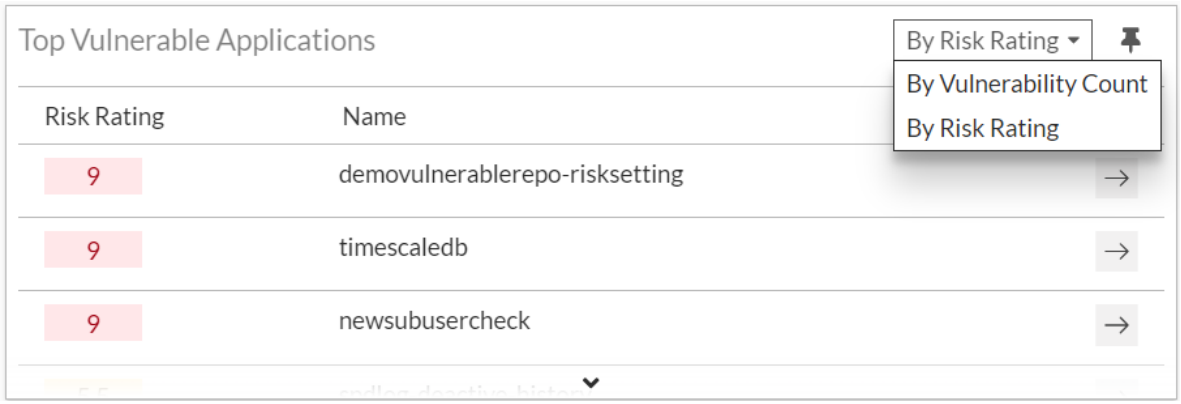
- **FortiGuard Outbreaks** - Displays the total number of *FortiGuard Outbreak Alerts* detected across all your applications. *FortiGuard Outbreak Alerts* identified by *FortiGuard Labs* provide critical information about ongoing cybersecurity attacks with significant potential impact. Click arrow to view detailed information.



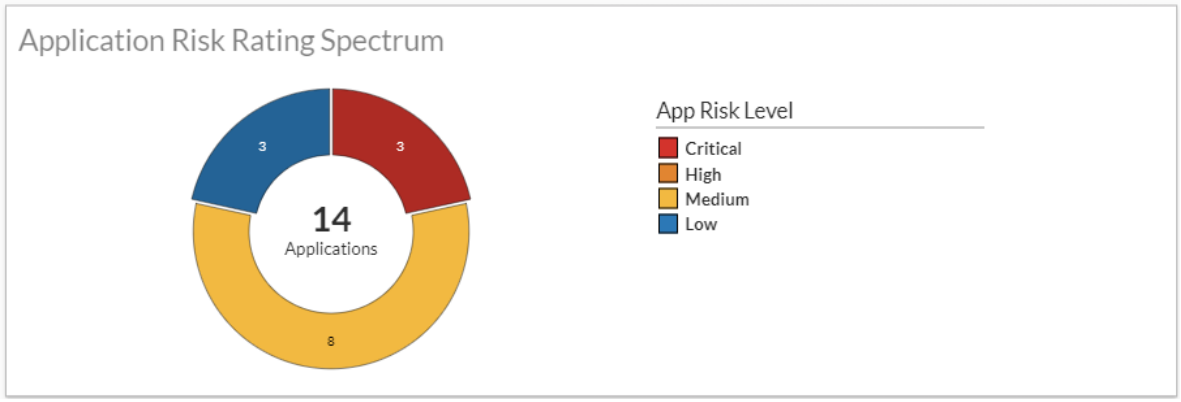
- **Supply Chain Threats** - Displays the total number of supply chain threats detected across all your applications. A supply chain threat is a potential risk or vulnerability within the production and distribution process that can compromise the security, integrity, or availability of products, services, or data. Click arrow to view detailed information.



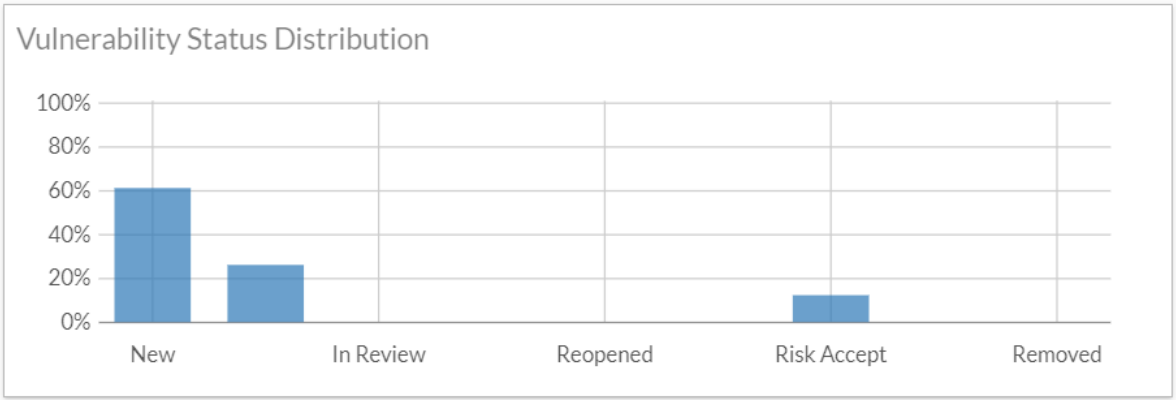
- **Top Vulnerable Applications** - Lists applications with the highest number of vulnerabilities or the highest risk ratings. Use the dropdown to sort *By Risk Rating* or *By Vulnerability Count*. Click on an application name or the arrow to view the application's scan details. See Viewing Scanned Application Details.



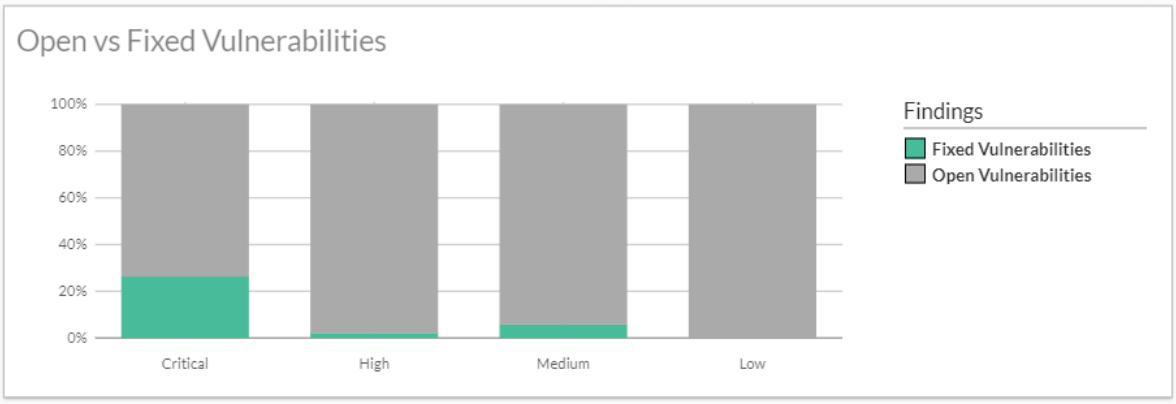
- **Application Risk Rating Spectrum**: A donut chart showing how many applications fall into each risk level (Critical, High, Medium, and Low).



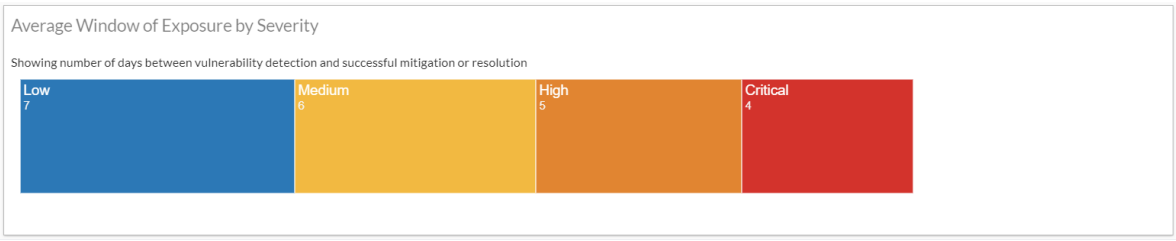
- **Vulnerability Status Distribution**: Chart showing the percentage of applications with vulnerabilities in different statuses (New, Confirmed, In Review, Reopened, Fixed, Risk Accepted, False Positive, and Removed).



- **Open vs Fixed Vulnerabilities**: Chart displaying the percentage of open and resolved vulnerabilities, grouped by severity.



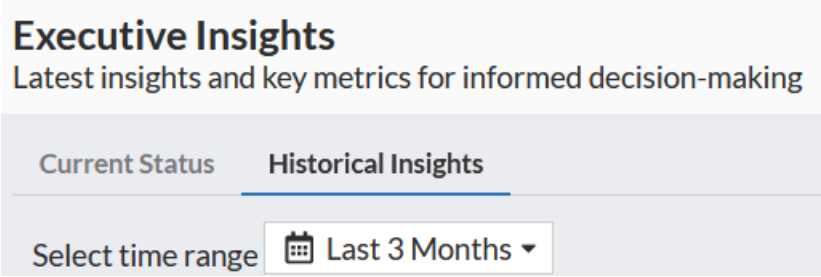
- **Window of Exposure by Severity**: Measures the average time (in days) between when a vulnerability is detected and when it's successfully fixed, grouped by severity.



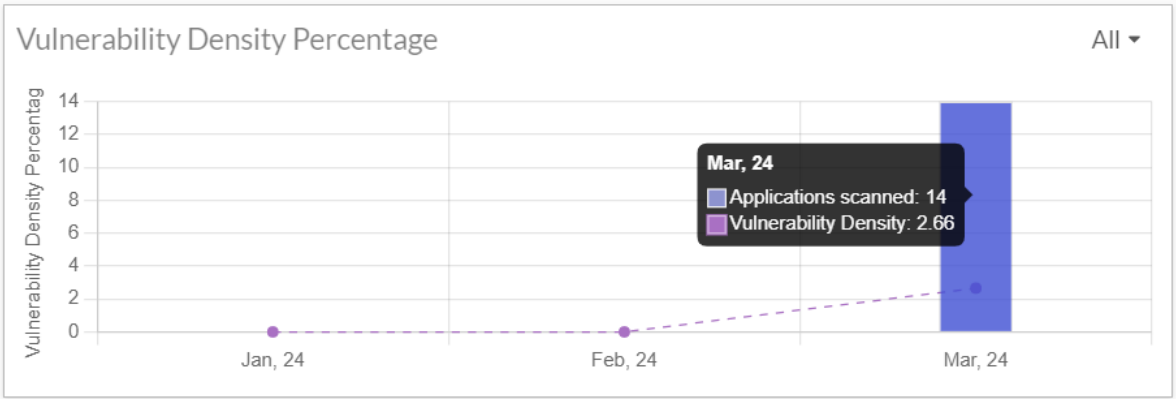
Historical Insights

The *Historical Insights* tab in the Analytics section provides a view of how your organization's security posture has evolved over time. You can analyze historical patterns, and measure the progress of your security improvements. The following information is displayed.

- **Select time range** - Choose between *3*, *6*, and *12* months to define the period for historical analysis.

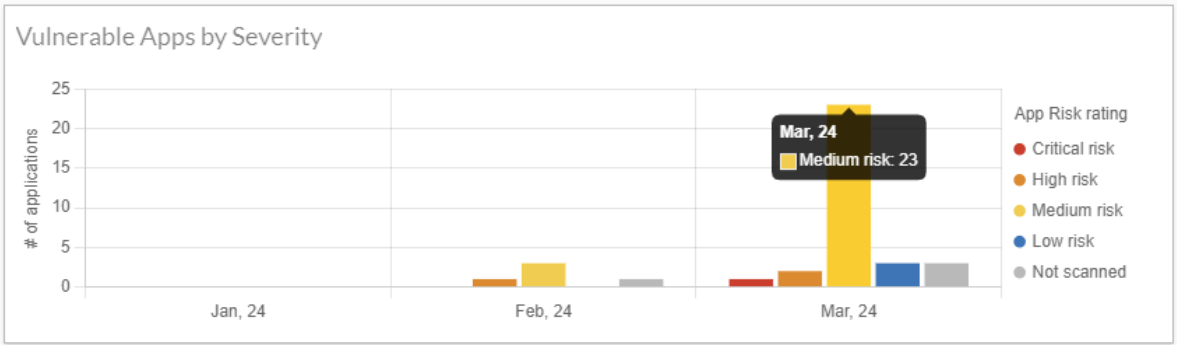


- **Vulnerability Density Percentage** - Displays the vulnerability density (number of findings divided by the number of lines of code scanned) over the selected time period, grouped by severity. Use the dropdown to select specific severity levels (*All*, *Critical*, *High*, *Medium*, *Low*).

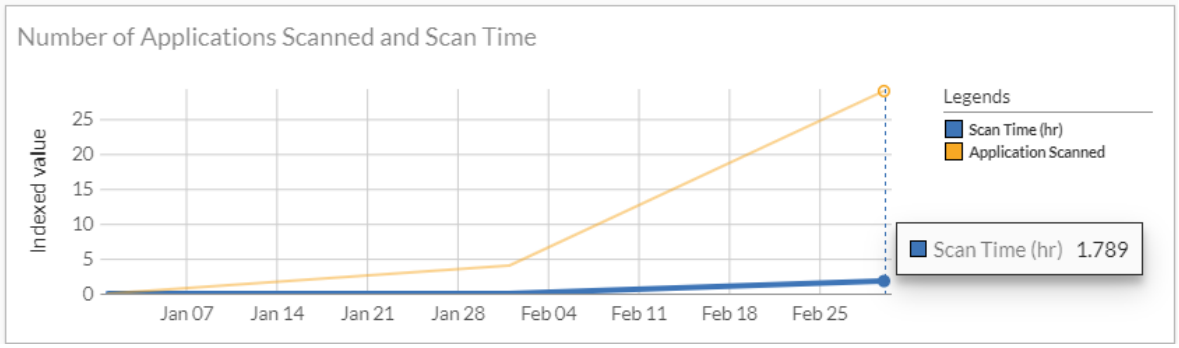


- **Vulnerable Apps by Severity** - Displays trends in the number of vulnerable applications over time, grouped by severity level. Click legend items to filter the information displayed based on application risk rating.

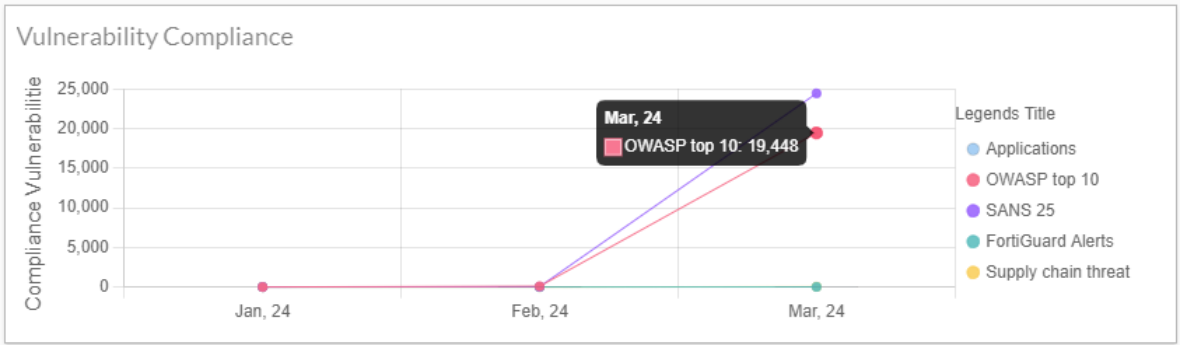
  Note: *Not scanned* refers to applications that were scanned in the previous month but have not been scanned within the currently selected time period.



- **Number of Applications Scanned and Scan Time** - Displays line chart showing how many applications were scanned and the average scan duration (in hours) over time.



- **Vulnerability Compliance**: Displays the number of vulnerabilities detected over time, categorized by *OWASP Top 10*, *SANS Top 25*, *FortiGuard Alerts*, and *Supply Chain Threats*.

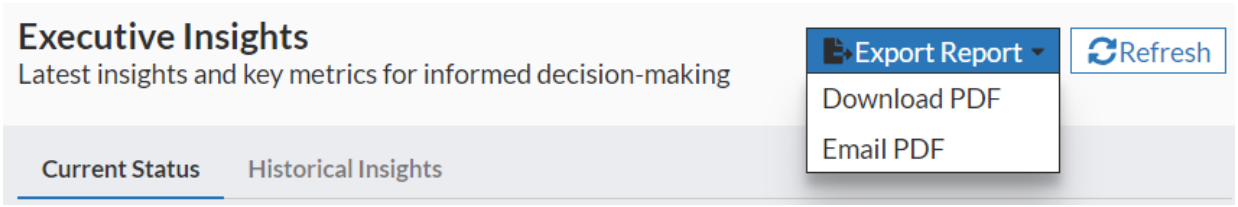


Exporting Reports

You can export a report containing current status and historical insights data for your organization. This report provides a high-level overview of your organization's application security posture. There are two ways to export the report.

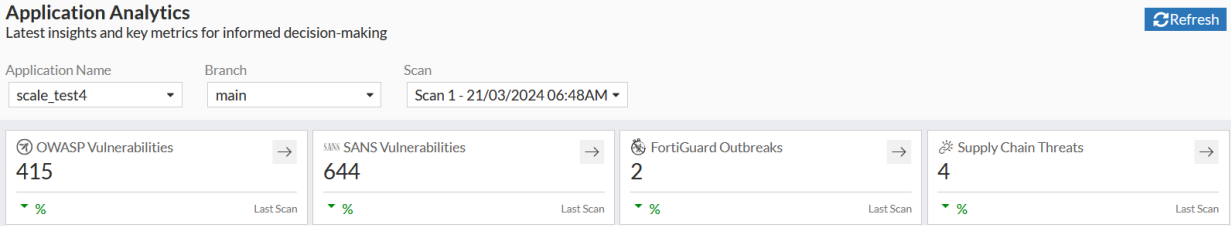
1. **Download PDF**: Click *Export Report* > *Download PDF* to download the report in PDF format.
2. **Email PDF**: Click *Export Report* > *Email PDF* to email the report as a PDF attachment.

- Receiving the report via email can take up to 5 minutes.
- The report is only sent to the organization owners. Sub users will not receive the email report.



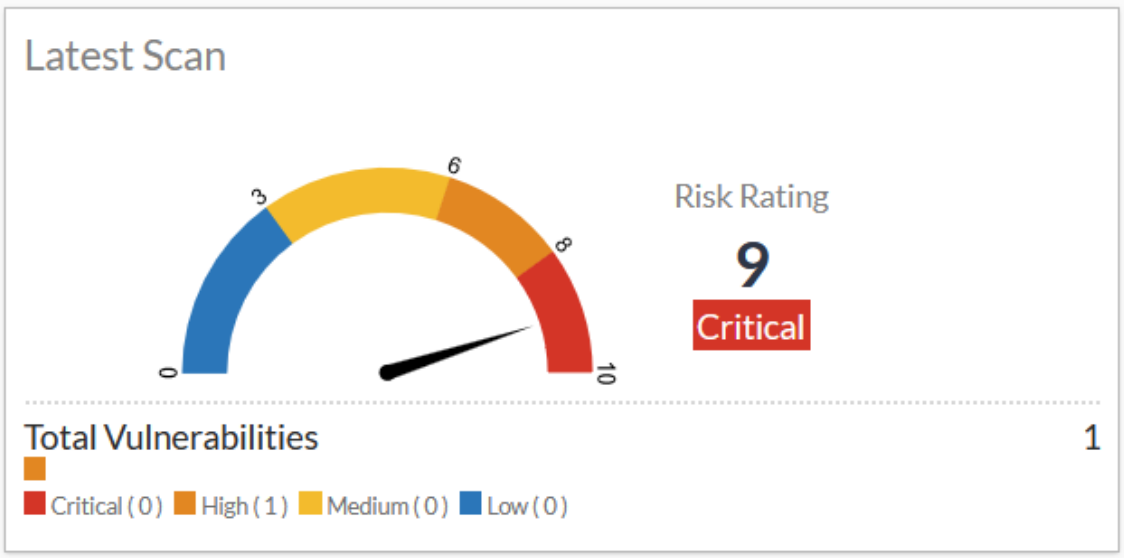
Application Level

The *Application Level* page in the *Analytics* section provides an in-depth analysis of the security posture of a specific application. Select the application, relevant code branch, and the specific scan result that you want from the *Application*, *Branch*, and *Scan* dropdown respectively. To refresh the application scan information, click *Refresh*.



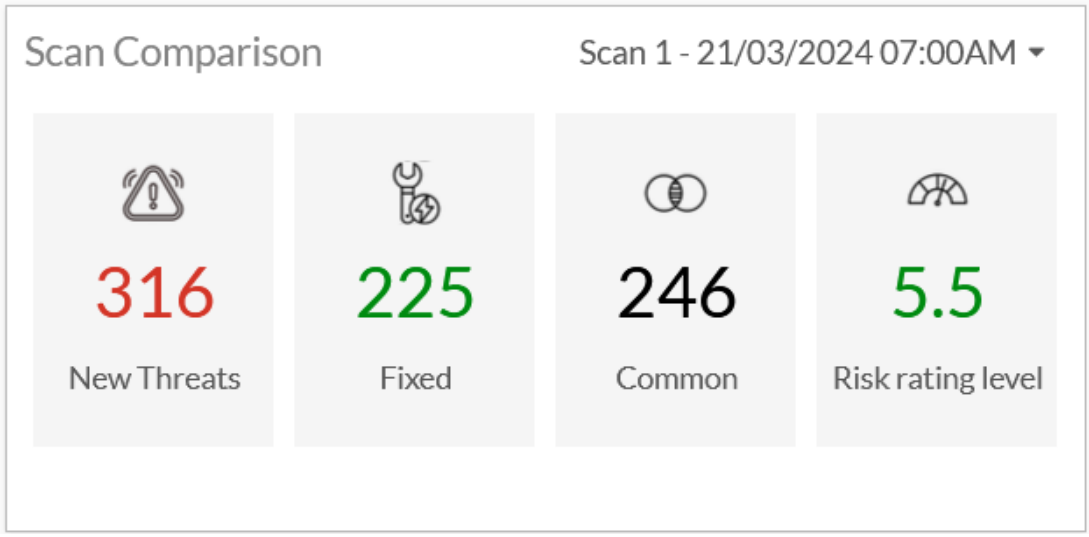
The following information is displayed for the selected application.

- **OWASP Vulnerabilities** - Displays the total number of vulnerabilities detected that align with the OWASP Top 10 list. Percentage change indicates the increase/decrease in the number of vulnerabilities since the previous scan. Click arrow to view detailed information in list or distribution graph formats.
- **SANS Vulnerabilities**- Displays the total number of vulnerabilities detected that align with the SANS Top 25 list. Percentage change indicates the increase/decrease in the number of vulnerabilities since the previous scan. Click the arrow to view detailed information in list or distribution graph formats.
- **FortiGuard Outbreaks** - Displays the total number of *FortiGuard Outbreak Alerts* detected. *FortiGuard Outbreak Alerts* identified by *FortiGuard Labs* provide critical information about ongoing cybersecurity attacks with significant potential impact. Percentage change indicates the increase/decrease in the number of vulnerabilities since the previous scan. Click arrow to view detailed information.
- **Supply Chain Threats** - Displays the total number of supply chain threats detected. A supply chain threat is a potential risk or vulnerability within the production and distribution process that can compromise the security, integrity, or availability of products, services, or data. Percentage change indicates the increase/decrease in the number of vulnerabilities since the previous scan. Click arrow to view detailed information.
- **Latest Scan** - Displays risk meter, risk rating, severity level, and vulnerability counts by severity.



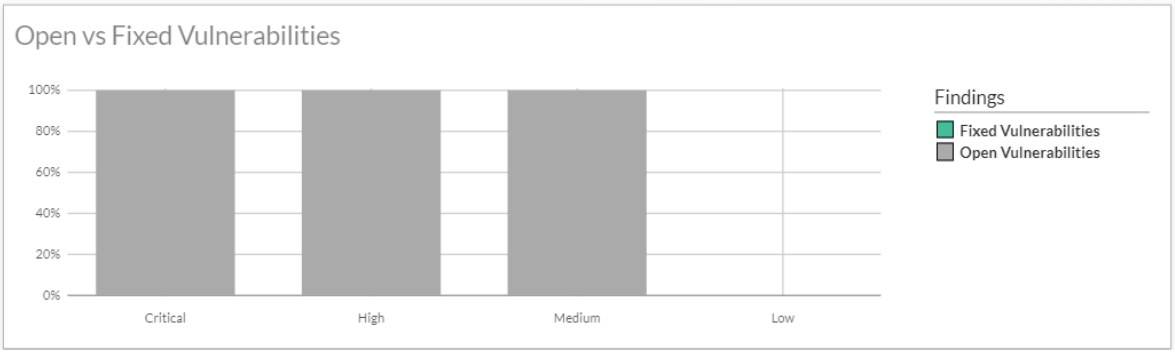
- **Scan Comparison** - Displays comparison of the latest scan with the previously selected scan. You can select the scan result of any prior scans from the dropdown for comparison. The following metrics are provided.
  - **New Threats** - Vulnerabilities detected in the latest scan but not the previous one.
  - **Fixed** - Vulnerabilities present in the previous scan that have been resolved.

    Note: When resolving a vulnerability, address all similar occurrences to ensure they are fixed.
  - **Common** - Vulnerabilities present in both scans.
  - **Risk rating level** - Indicates if the risk rating has increased, decreased, or remained the same (the risk rating level is not color coded to indicate severity).

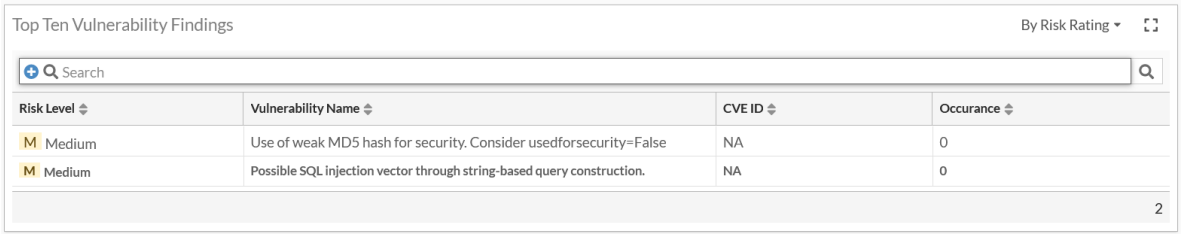


- **Vulnerability Status Distribution** - Displays the percentage of vulnerabilities in each status (*New, Confirmed, In Review, Reopened, Fixed, Risk Accepted, False Positive, Removed*). Select the severity checkbox to filter data.

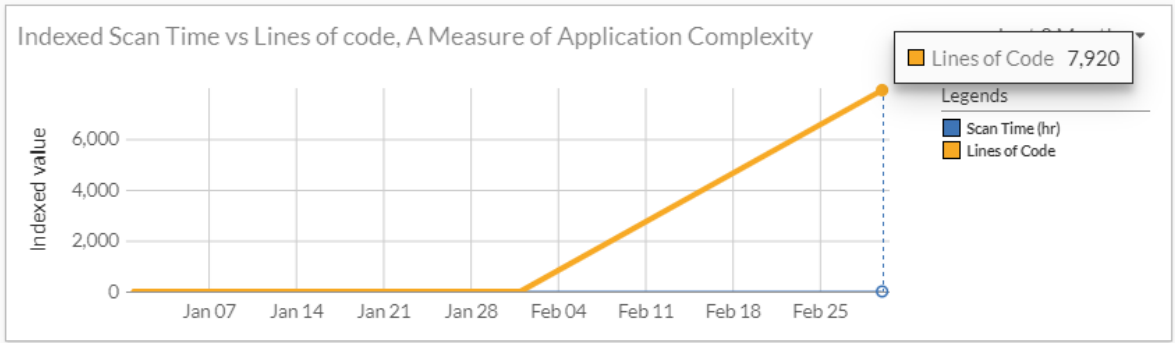
- **Open vs. Fixed Vulnerabilities** - Displays the percentage of open and fixed vulnerabilities grouped by severity.



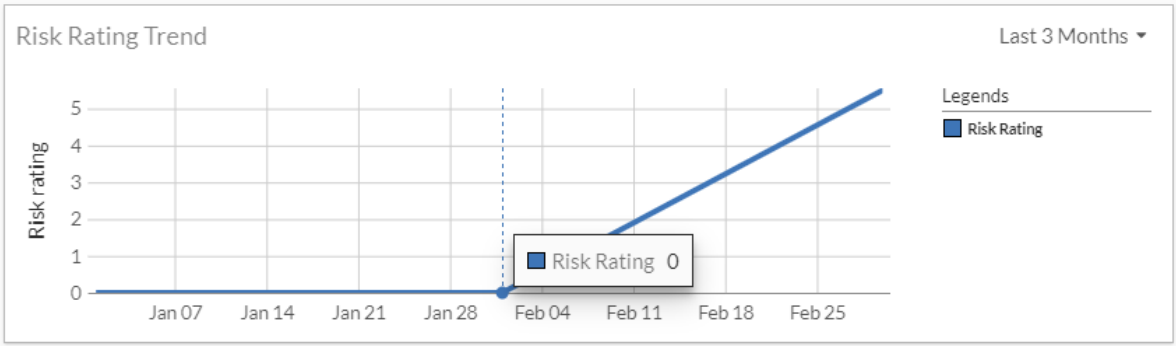
- **Top Ten Vulnerability Findings** - Lists top ten vulnerability findings with the highest count or the highest risk ratings. Use the dropdown to sort *By Risk Rating* or *By Vulnerability Count*.



- **Indexed Scan Time vs Lines of Code** - Displays the number of lines of code scanned and the average scan time in hours over the selected period for the application. Choose between 3, 6, or 12 months from the dropdown.



- **Risk Rating Trend** - Displays how the application's risk rating has changed over time. Choose between 3, 6, or 12 months from the dropdown.



Vulnerability Catalog

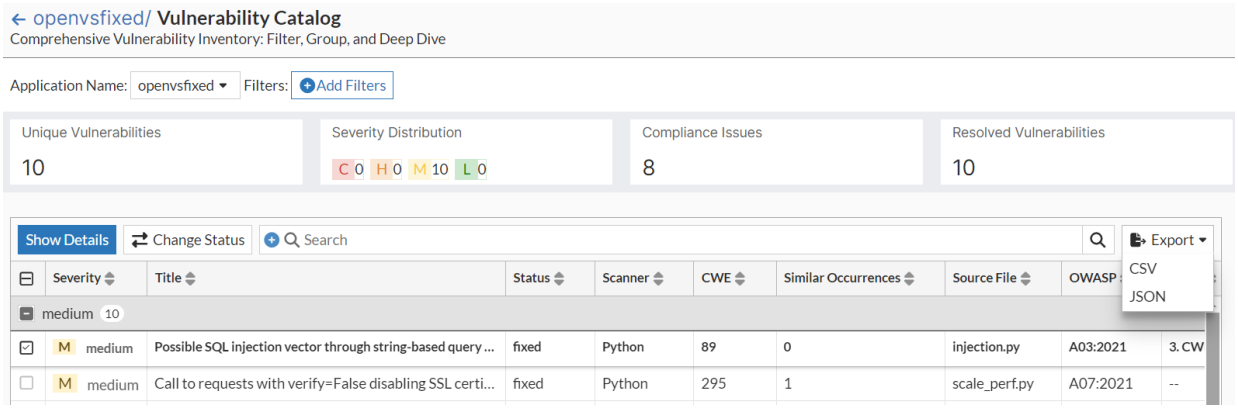
The Vulnerability Catalog page provides a comprehensive view of all vulnerabilities identified within a selected application. You can view, search, sort, filter, and change the status of these vulnerabilities.

The following summary widgets offer a quick overview of the vulnerability data.

- *Unique Vulnerabilities*: Displays the total number of distinct vulnerabilities. Each unique vulnerability may have multiple instances, but it is counted only once in this summary.
- *Severity Distribution*: Displays the count of vulnerabilities based on their severity levels.
- *Compliance Issues*: Indicates the number of vulnerabilities that are related to OWASP or SANS standards.
- *Resolved Vulnerabilities*: Displays the number of vulnerabilities that have been resolved. A vulnerability is considered resolved when its status is changed to one of the following: *fixed*, *false positive*, *removed*, or *risk accepted*.

You can perform the following actions within the Vulnerability Catalog.

- Choose an application from the **Application Name** dropdown to view its specific vulnerabilities.
- Use the **Search** bar to find vulnerabilities by name or other criteria. Additionally, you can add filterable columns to search for vulnerabilities based on specific fields.
- Click **Export** and select either **CSV** or **JSON** to save the list of vulnerabilities in the desired format.
- Apply filters to narrow down the list of vulnerabilities based on various criteria, such as severity, status, or compliance. See Filtering Vulnerabilities.
- View vulnerability details. See Viewing Vulnerability Details.
- Modify the status of vulnerabilities. See Modifying the Vulnerability Status



The following fields are displayed for each vulnerability. The vulnerabilities are grouped by severity. To customize the visible columns, hover over the first column header and click the gear icon. From the displayed menu, select or deselect the column names you want to add or remove.

- *Severity*: The level of risk associated with the vulnerability.
- *Title*: A brief description of the vulnerability.
- *Status*: The current status of the vulnerability.
- *Scanner*: The type of scanner that detected the vulnerability.
- *CWE*: The Common Weakness Enumeration (CWE) identifier that categorizes the vulnerability.

- *Similar Occurrences*: The number of times this vulnerability has been detected in other applications or environments.
- *Source File*: The specific file within the application where the vulnerability was found.
- *CVE*: The Common Vulnerabilities and Exposures (CVE) identifier, if applicable.
- *Finding ID*: A unique identifier assigned to the vulnerability.
- *FortiGuard Outbreaks*: Indicates whether the vulnerability is associated with any known FortiGuard outbreak alerts.
- *Last Scanned*: The date and time of the last scan that detected the vulnerability.
- *Line Number*: The specific line within the source file where the vulnerability was identified.
- *OWASP*: The OWASP category.
- *SANS*: SANS Top 25 category.
- *Supply Chain Threats*: Indicates whether the vulnerability is associated with supply chain threats.

Filtering Vulnerabilities

You can filter the displayed findings based on specific criteria. in the *Vulnerability Catalog* page, select an application and click **Add Filters**. The following filters are available.

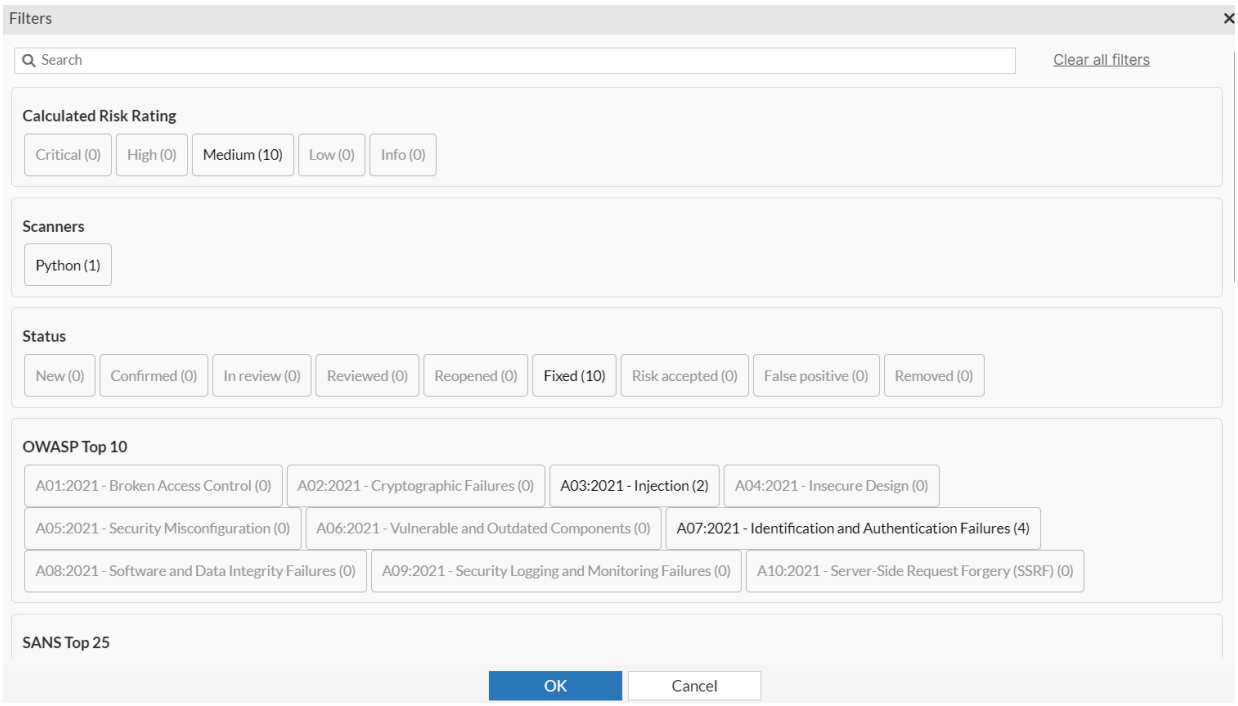
- **Calculated Risk Rating** - Filtered based on the assigned risk rating.
- **Scanners** - Filtered based on the scanners used in the latest scan.
- **Status** - Filtered based on the status.
- **Category** - Filtered based on the specific application.
- **Files** - Filtered based on the specific files.
- **Directory** - Filtered based on the specific directories.
- **OWASP Top 10** - Filters based on the specific OWASP vulnerability.
- **SANS top 25** - Filters based on the specific SANS vulnerability.
- **Images** - Filters based on the image files.

Within each filter section, select the desired options to narrow down the application list. Also, you can use the *Search* field to find a specific filter.

Once you have selected the filter/s, click the *OK* to apply the filters and update the displayed applications.

To remove all applied filters and return to the unfiltered application list, click the *Clear all filters.*

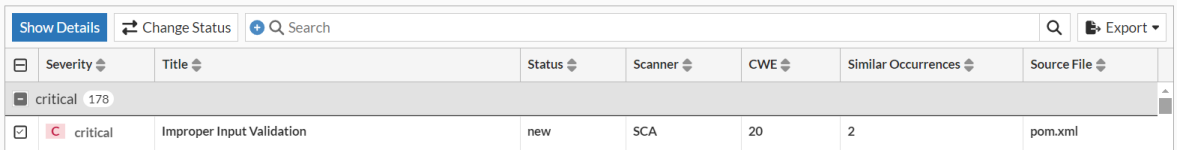
Note: To export a specific type of vulnerability, select the desired filters and click **Export**.



Viewing Vulnerability Details

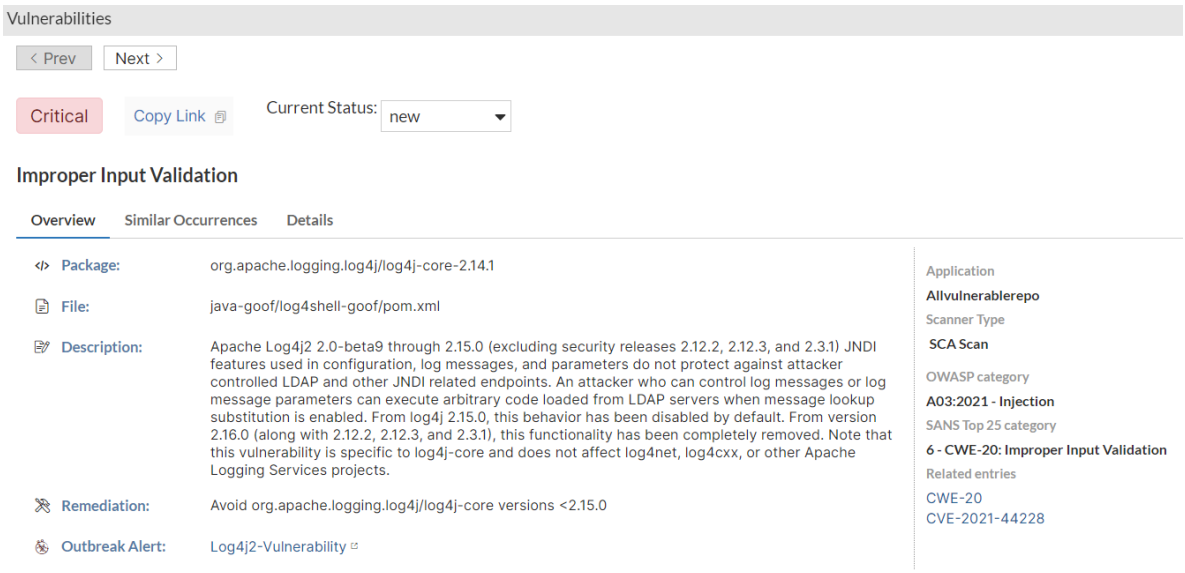
Perform the following steps to view vulnerability details.

1. In the *Vulnerability Catalog* page, select an application.
2. Select a desired vulnerability.
3. Click **Show Details** to view the detailed information.

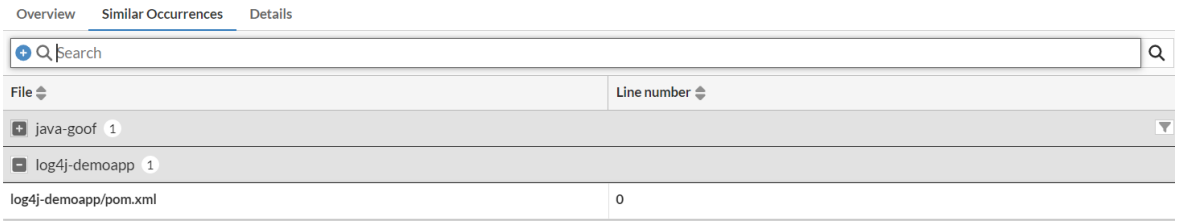


The following information is displayed.

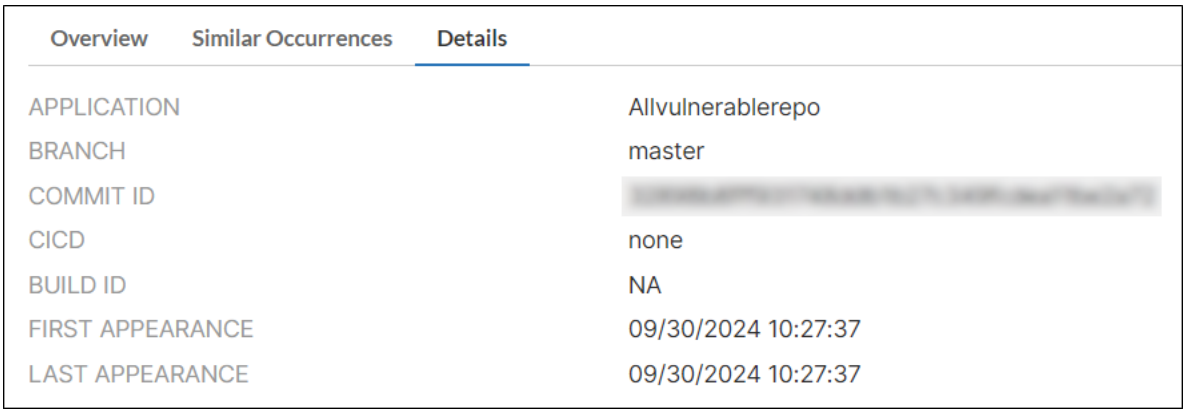
- The severity level assigned to the vulnerability by FortiDevSec
- The current status of the vulnerability. You can change the status by selecting a different option from the dropdown menu.



- The **Overview** section contains the following information.
  - The name of the application in which the vulnerability was found.
  - The name of the software **package** or component where the vulnerability was found.
  - The associated **file** and the **line number** that the vulnerability is found in.
  - The **Issue** description and the associated **CWE** and **CVE** (if any). Click on the CWE/CVE link to view details.
  - The **Remediation** provides information (if available) on how to fix/avoid the vulnerability.
  - The associated **OWASP Top10** or **SANS Top 25** category.
  - The **Outbreak Alert** lists the alerts found in the application. Click outbreak alert link to navigate to the FortiGuard Outbreak Alert page for in-depth analysis.
  - The **Supply Chain** provides information on the supply chain threat detected.
- The **Similar Occurrences** section contains the number of times this vulnerability has been detected in other applications or environments. Click on each instance to view its details. Expand each instance to see additional information. You can also use the search field to find occurrences within other files.



- The **Details** section contains the following information.
  - The **CI/CD** and **Build** details.
  - The history of the vulnerability is also displayed that includes the time of its first and last appearance.



The vulnerabilities details page for the SECRET scanner contains the following additional information in Overview tab.

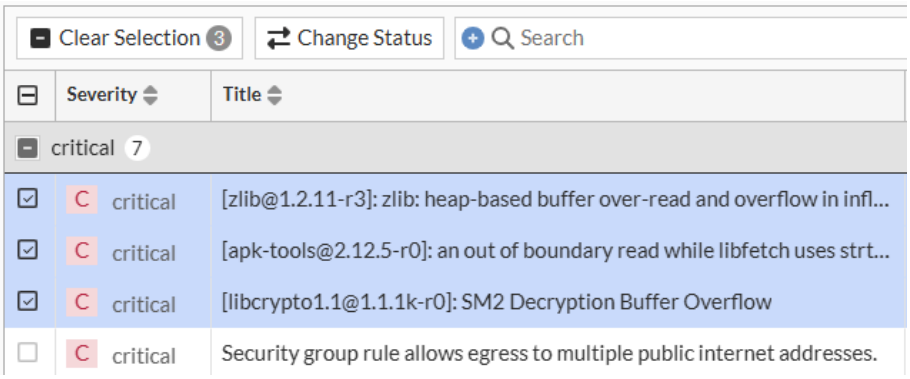
- The **Secret Type** displays the type of secret detected.
- The **Secret Status** displays the current status of the secret detected.
  - JWT - **Expired** or **Valid**
  - AWS - **Exploitable** or **Not Exploitable**
- The **Detected In** provides information on where the secret was detected.
  Note: Secrets are first detected in git commits and then searched for in files.
- The **Code** field includes the following information.
  - **Hash** - Git commit hash.
  - **By** - Details of the user who has committed the change.
  - The last line in code field contains the commit message added by the user. If the GIT commit message is more than 200 characters it is truncated.
- The **Associated Issues** displays fingerprint for other components of the exploitable AWS credential within the same file.

Modifying the Vulnerability Status

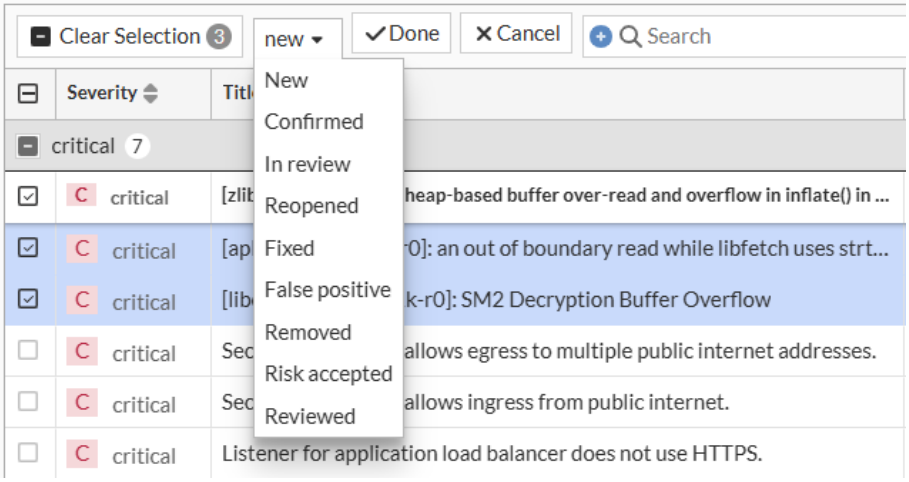
You can modify the status of each vulnerability or of all vulnerabilities.

Perform the following steps to change the status of vulnerability.

1. In the *Vulnerability Catalog* page, select an application
2. Select the check box next to the desired vulnerability or multiple vulnerabilities.
3. Click **Change Status**.



4. From the drop-down menu, choose the desired status to apply.



5. Click **Done**.

The status of the selected vulnerabilities changes.

The following status types are supported.

- **New**: This is a new vulnerability detected by the scan.
- **Confirmed**: This is a real vulnerability and requires a fix.
- **In Review**: This vulnerability is currently in review/looked into for further action.
- **Reviewed**: This vulnerability review is complete.
- **Reopened**: This is a fixed vulnerability detected again in the rescan and requires to be addressed.
- **Fixed**: This vulnerability is fixed and does not appear in the next scan result.
- **Risk Accepted**: This vulnerability is an accepted risk and continues to exist without any potential damage.

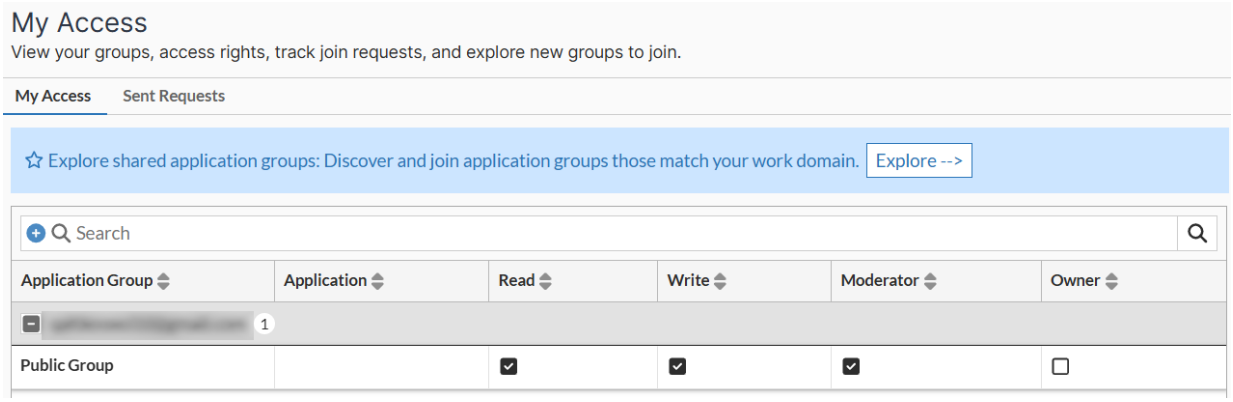
- **False Positive**: This vulnerability is a potential flaw in the scanner or is indicative of a unique feature of the application.
- **Removed**: This vulnerability is overlooked in the application.

My Access

The *My Access* page provides an overview of your current access rights within FortiDevSec. It displays your membership in various application groups, along with the associated permissions and access levels.

The *My Access* page consists two tabs.

- My Access
- Sent Requests



Note: **Sent Requests** and **Explore** options are available only for users with *Read*, *Write*, and *Moderator* access.

My Access

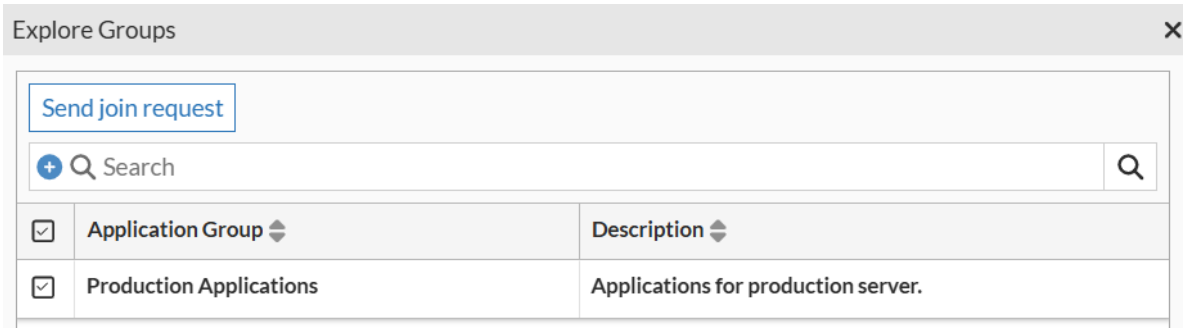
The *My Access* tab presents a detailed view of your group memberships, organized by member groups. Each row in the table represents an application group, providing the following information.

- *Application Group*: The name of the application group.
- *Applications*: A list of applications included within the group.
- *Read*: Indicates whether you have read access to the group and its applications.
- *Write*: Indicates whether you have write access to the group and its applications.
- *Moderator*: Indicates whether you have moderator permissions for the group and its applications.
- *Owner*: Indicates whether you are the owner of the group and its applications.

To explore available shared groups and send join requests:

1. In **My Access** tab, click **Explore**. A list of available shared groups will be displayed.
2. Select the desired group and Click the **Send Join Request** button to initiate the request.
3. The request will be sent to the application group owner or moderator for approval or rejection. You can

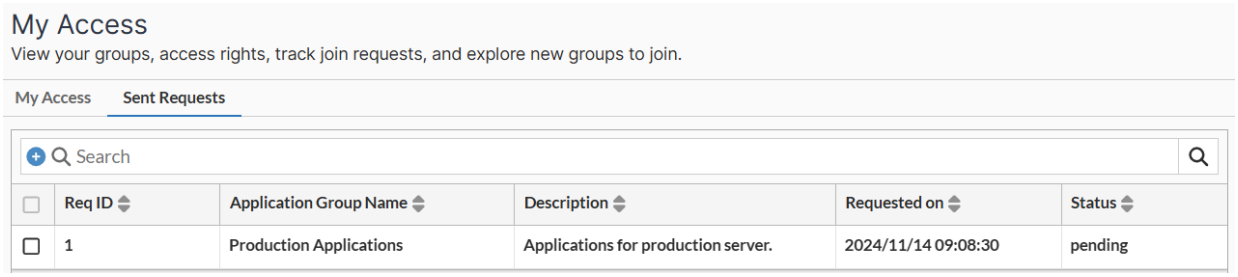
track the status of your request in the **Sent Requests** tab.



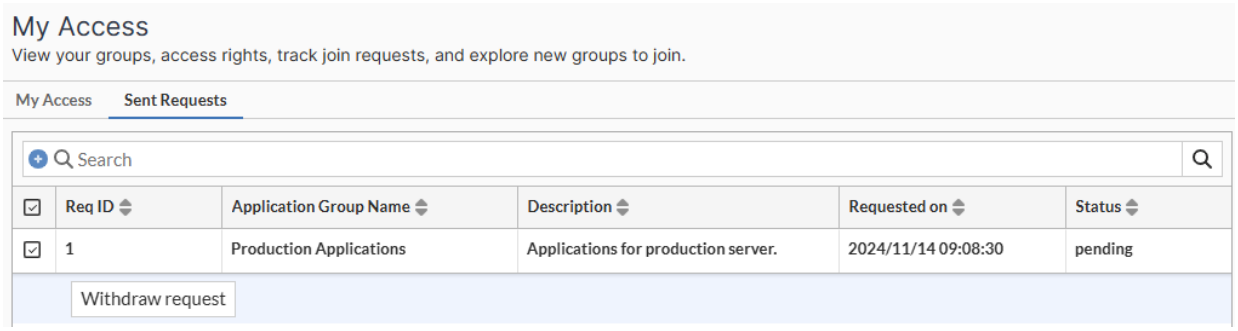
Sent Requests

The *Sent Requests* tab provides a list of all the shared group join requests you have sent. For each request, the following information is displayed.

- *Request ID*: A unique identifier for the request.
- *Application Group Name*: The name of the application group you requested to join.
- *Description*: A brief description of the application group.
- *Requested On*: The date and time when the request was sent.
- *Status*: The current status of the request (*Pending*, *Approved*, or *Rejected*).



If you no longer need a sent request that is currently *Pending*, you can withdraw it. To do so, navigate to **My Access > Sent Requests**, select the desired request, and click **Withdraw request**.



Access Management

The *Access Management* section allows you to manage user access and permissions within your FortiDevSec organization. You can create and manage application and member groups, assign specific access rights to members within these groups, and control shared group requests. This allows you to maintain a secure and controlled environment, ensuring that only authorized users have access to specific applications and resources.

- Groups
  - Member Groups
  - Application Groups
- Group Requests
- Access Control
- User Permissions

Member Groups

The *Member Groups* page allows you to create, manage, and view details of member groups within your FortiDevSec organization. Member group is a collection of users who share similar access privileges. Only master users can create member groups. See User Permissions.

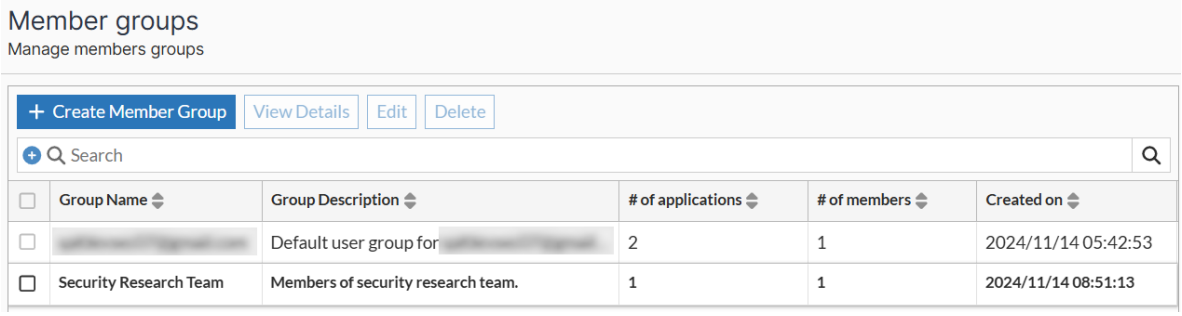
- Viewing Member Groups
- Creating a Member Group
- Editing a Member Group
- Viewing Member Group Details
- Deleting a Member Group

Viewing Member Groups

The *Member Group* page displays a table listing all the member groups. The following information is displayed.

- *Group Name*: The name of the group.
- *Group Description*: A description of the group.
- *# of Applications*: The number of applications accessible to the group.
- *# of Members*: The number of users in the group.

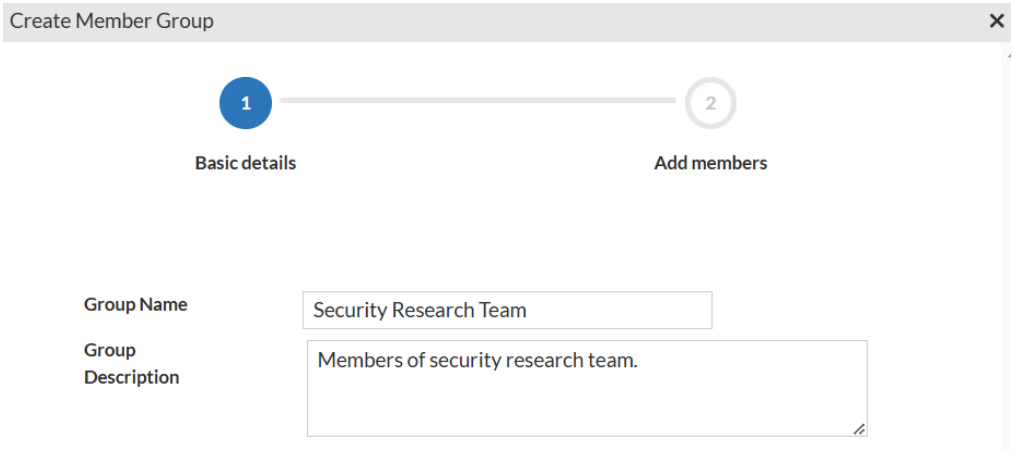
- *Created on*: The date the group was created.



Creating a Member Group

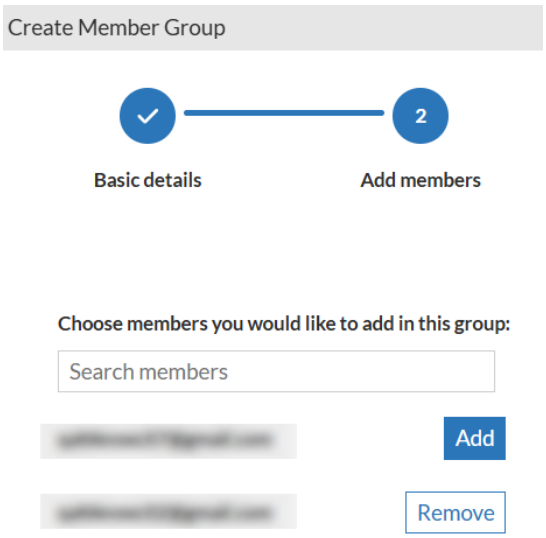
Perform the following steps to create a new Member Group.

1. Navigate to **Access Management > Groups > Member Groups**.
2. Click **+ Create Member Group**.
3. Provide a **Group Name** and **Description** for the group. Click **Next**.



4. Add members to the group by searching for the desired users and clicking **Add** next to their names.

   Note: Users must be created in FortiCloud (*IAM*, *IdP*, or *Sub-users*) and must log in to FortiDevSec at least once before they can be added to a member group.



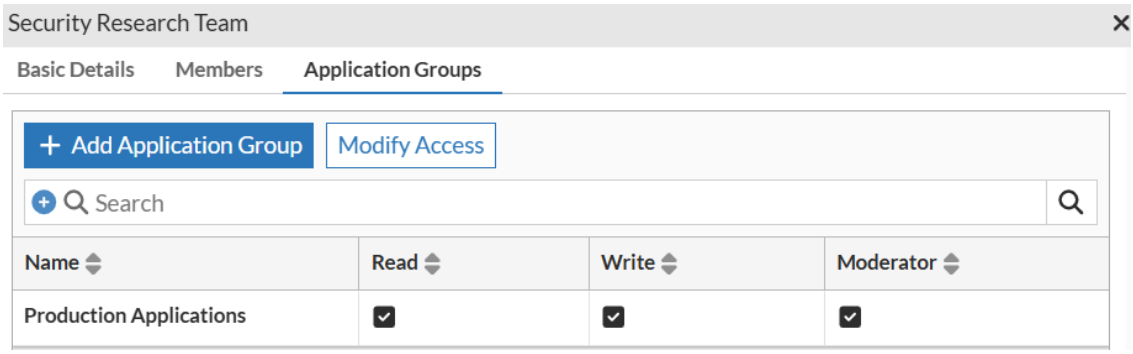
5. Click **Submit** to create the group.

Note: If a member is added to two different member groups with different permissions for the same application group, the member will be assigned the highest permission level granted across all member groups.

Editing a Member Group

Perform the following steps to modify the Member Group.

1. Navigate to **Access Management > Groups > Member Groups**.
2. Select a member group from the list. Click **Edit**.
3. In **Basic details** tab, edit the group name and description.
4. In **Members** tab, click **Modify Members** to add or remove members.
5. In **Application Groups** tab, modify application group access.
   a. To add an application group, click **Add Application Group**, select the desired group, choose the access type (*Read*, *Write*, or *Moderator*), and click **Add**.
   b. To modify access, select an existing application group, click **Modify Access**, change the access type, or click **Remove** to remove the group.



6. Click **Update** to save the changes.

Viewing Member Group Details

Perform the following steps to view a Member Group details.

1. Navigate to **Access Management > Groups > Member Groups**.
2. Select a member group from the list.
3. Click **View Details** to access detailed information about the group, including:
   - Basic information (name, description)
   - A list of application groups associated with the member group
   - A list of members belonging to the group

Deleting a Member Group

Perform the following steps to delete a Member Group.

1. Navigate to **Access Management > Groups > Member Groups**.
2. Select a member group from the list. Click **Delete**.

Application Groups

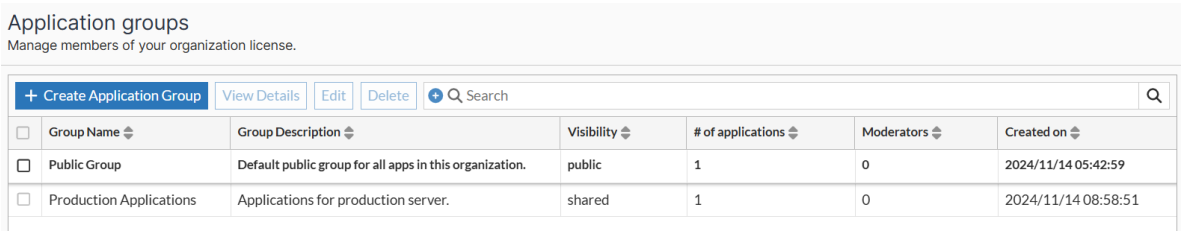
The *Application Groups* page allows you to create and manage application groups. These groups can be used to organize and control access to applications added to FortiDevSec. Only master users can create application groups. See User Permissions.

- Viewing Application Groups
- Creating a New Application Group
- Editing an Application Group
- Viewing Application Group Details
- Deleting an Application Group

Viewing Application Groups

The Application Group page displays a table listing all the application groups. The following information is displayed.

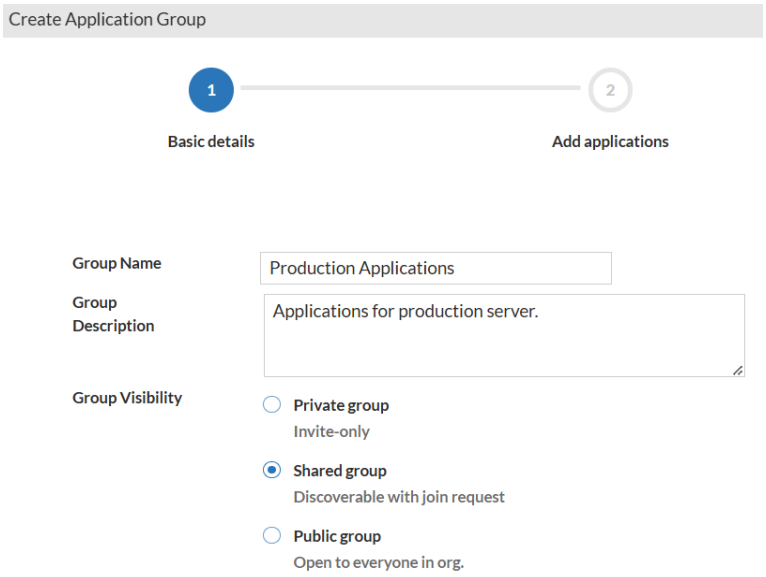
- *Group Name*: The name of the group.
- *Group Description*: A description of the group.
- *Visibility*: The visibility settings for the group.
- *# of Applications*: The number of applications in the group.
- *# of Moderators*: The number of moderators for the group.
- *Created on*: The date the group was created.



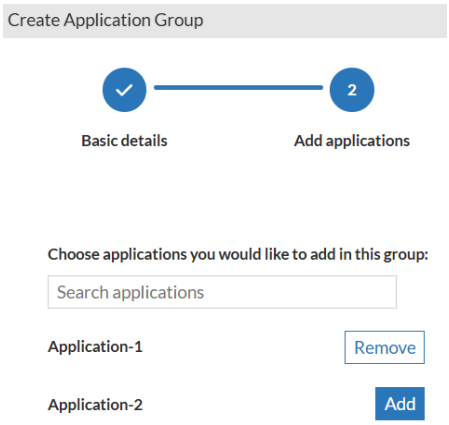
Creating a New Application Group

Perform the following steps to create a new application group.

1. Navigate to **Access Management > Groups > Application Groups**.
2. Click **Create Application Group**.
3. Enter a **Group Name** and **Group Description**.
4. Select the Group Visibility and click **Next**.
   - *Private group*: Accessible only to members of specific member groups, as determined by the owner or moderator.
   - *Shared group*: Discoverable by users, who can request to join the group. Requests are reviewed and approved/rejected by owners or moderators.
   - *Public group*: Accessible to all users in the organization.



5. Search for the desired application or click **Add** next to the application name to add it to the group.

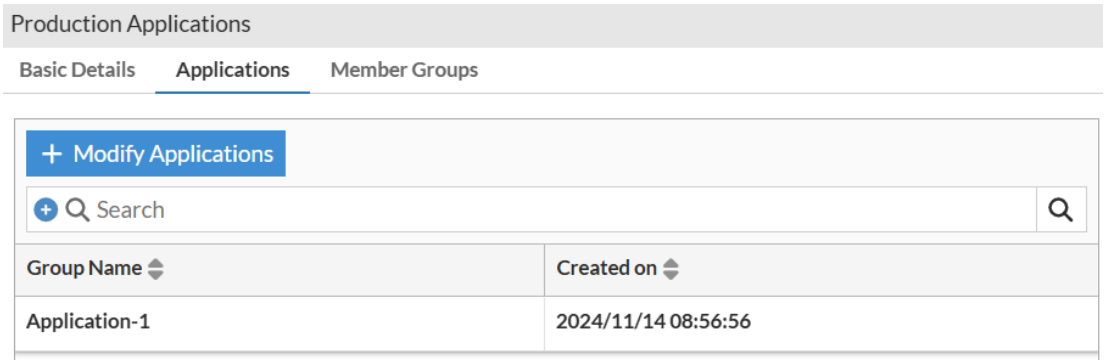


6. Click **Submit**.

Editing an Application Group

Perform the following steps to modify application group details.

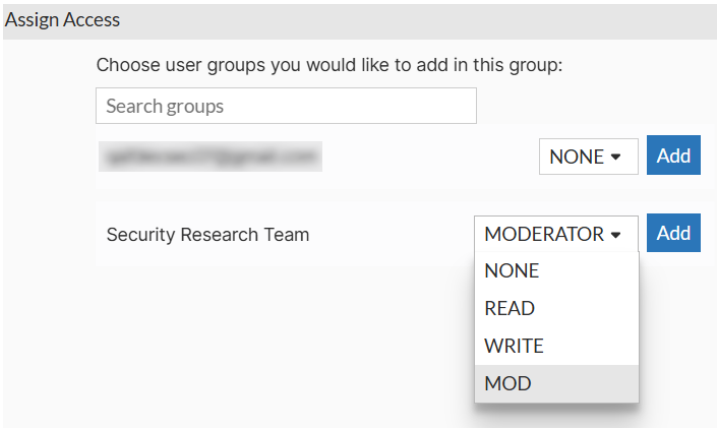
1. Navigate to **Access Management > Groups > Application Groups**.
2. Select an **Application Group** and click **Edit**.
3. In **Basic Details** tab, modify the basic details (name, description, and group visibility) as needed.
4. In the **Applications** tab,
   a. Click **Modify Applications**.



   b. Click **Add** to add new applications or click **Remove** to remove existing applications.

   Note: An application can only belong to one application group at a time. If you attempt to add an application to a new group while it's already a member of another group, it will be automatically removed from the original group.

   c. Click **Add**.
5. In the **Member Group** tab,
   a. To add a member group, click **Add Member Group**.
   b. Select the desired member group and access type (*Read*, *Write*, *Moderator*).
   c. Click **Add**.



   d. To modify access permissions or remove a member group, click **Modify Access** or **Remove**.
   e. Click **Done**.
6. Click **Update**.

Viewing Application Group Details

Perform the following steps to view application group details.

1. Navigate to **Access Management > Groups > Application Groups**.
2. Select an **Application Group** and click **View Details**.

3. View detailed information about the group, including:
   - *Group Visibility*: The group's visibility setting.
   - *Description*: A description of the group.
   - *Applications*: A list of applications in the group, including their creation date and member group access.
   - *Member Group Access*: A list of member groups with their access permissions.

Deleting an Application Group

Perform the following steps to delete an application group.

1. Navigate to **Access Management > Groups > Application Groups**.
2. Select an **Application Group**.
3. Click **Delete**.

Note: To delete a group, it must be empty. If the group contains applications, you must either remove them or move them to a different group.

Group Requests

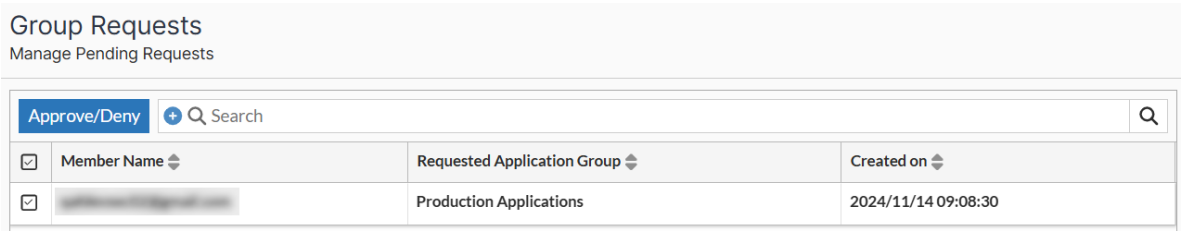
The *Group Requests* page allows master users and administrators to manage pending requests to join shared groups. These requests can be initiated by IAM, IDP, and sub-users.

- Viewing Group Requests
- Managing Group Requests

Viewing Group Requests

The *Group Requests* page displays a list of pending requests, including the following information for each.

- *Member Name*: The email id of the user who submitted the request.
- *Requested Application Group*: The name of the application group the user wants to join.
- *Created on*: The date the request was created.

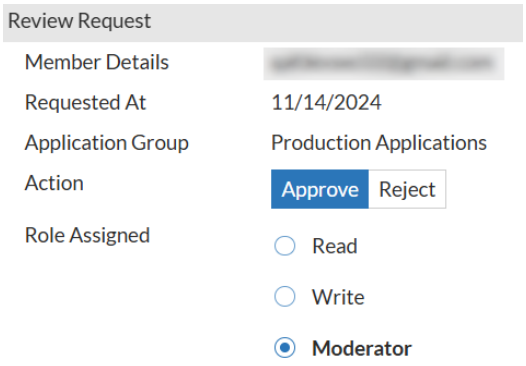


Managing Group Requests

Perform the following steps to approve or reject a request.

1. Navigate to **Access Management > Group Requests**.
2. Select a request from the list.
3. Click **Approve/Deny**.

4. In **Review Request** window,
   - To approve, click **Approve** and select a **Role Assigned** (*Read*, *Write*, or *Moderator*).
   - To reject, click **Reject** and provide a comment explaining the reason for rejection.



5. Click **Submit**.

A notification email will be sent to the sub-user who submitted the request, informing them of the decision. Users can check the status of their requests in **My Access > Sent Requests**.

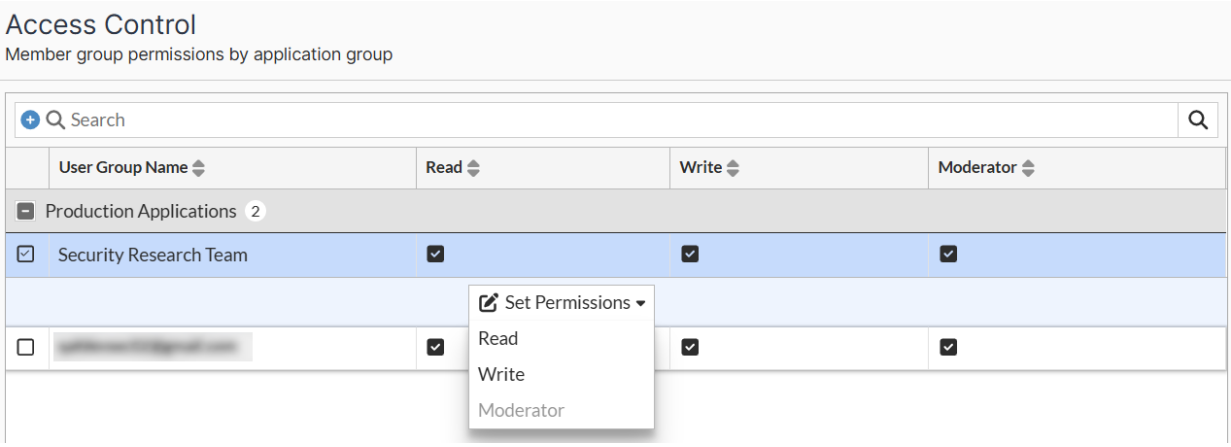
Access Control

The *Access Control* page provides granular control over member group permissions for each application group. This page displays a list of member groups for each application group, along with their current access permissions.

- *User Group Name*: The name of the member group.
- *Read*: Indicates whether the group has read-only access.
- *Write*: Indicates whether the group has write access.
- *Moderator*: Indicates whether the group has moderator access.

To modify a member group's permissions for a specific application group:

1. For the desired Application Group, select the **Member Group** you want to modify. Only the master user can modify the permissions on this page. See User Permissions.
2. Choose a new access permission from the **Set Permissions** dropdown.



User Permissions

In FortiDevSec, user permissions are granularly controlled to ensure secure access to applications and data. These permissions are assigned based on user roles and group memberships, providing a flexible and efficient way to manage access control. Master users are the owners of member and application groups.

Application permissions

Following are the permissions for different user roles at the application level.

Action	Permissions			
	Owner	Moderator	Write	Read
Create an application within a *Private* application group if you are a member of that group.	✓	✓	✓	✗
Create an application within a *Shared* application group if you are a member of that group.	✓	✓	✓	✗
Update *Risk Rating* of an application that is a part of *Private* or *Shared* application group.	✓	✓	✓*	✗
Change the *Vulnerability Status* in the *Vulnerability Catalog* page.	✓	✓	✓*	✗

*Only when the user with write access is the owner of the application.

Settings permissions

Following are the settings permissions for applications within *Private* or *Shared* application groups.

Action	Permissions			
	Owner	Moderator	Write	Read
Deactivate an application.	✓	✓	✓*	✗
Delete an application.	✓	✓	✗	✗
Configure *FortiDAST* plugin.	✓	✓	✓	✗

Action	Permissions			
	Owner	Moderator	Write	Read
Configure *Jira* plugin.	✓	✓	✓	✗
Modify application name.	✓	✓	✓*	✗
Modify application ID.	✓	✓	✗	✗

*Only when the user with write access is the owner of the application.

Member group permissions

Following are the permissions for different user roles at the member group level.

Action	Permissions			
	Owner	Moderator	Write	Read
Create a member group.	✓	✗	✗	✗
Add *Application Groups* and modify access permissions on the *Member Group Edit* page.	✓	✗	✗	✗
Modify member group name.	✓	✗	✗	✗
Add members to a member group.	✓	✗	✗	✗
Delete members from a member group.	✓	✗	✗	✗
Delete a member group.	✓	✗	✗	✗
Update permissions of a member group	✓	✓*	✗	✗

* A user with moderator access can only modify the permissions of other member groups with lower privileges. Both the moderator's group and the target group must have access to the same application group.

Application group permissions

Following are the permissions for different user roles at the application group level.

Action	Permissions			
	Owner	Moderator	Write	Read
Create a application group.	✓	✗	✗	✗
Add an application to application group.	✓	✓	✗	✗
Delete an application from application group.	✓	✓*	✗	✗
Modify application group name.	✓	✓	✗	✗

Action	Permissions			
	Owner	Moderator	Write	Read
Modify application group visibility.	✓	✓	✗	✗
Delete application group.	✓	✗	✗	✗
Add *Member Groups* and modify access permissions on the *Application Group Edit* page.	✓	✗	✗	✗

* Only when a user with moderator access is part of that application group.

Group request permissions

Following are the group request permissions.

Action	Permissions			
	Owner	Moderator	Write	Read
Approve shared application group join request.	✓	✓*	✗	✗

* Only when a user with moderator access is part of requested application group.

Access control permissions

Following are the access control permissions.

Action	Permissions			
	Owner	Moderator	Write	Read
Modify member group permissions in *Access Control* page.	✓	✗	✗	✗

Settings

The Settings section allows you to configure customized email notifications and manage API tokens.

- Email Notification
- API Access

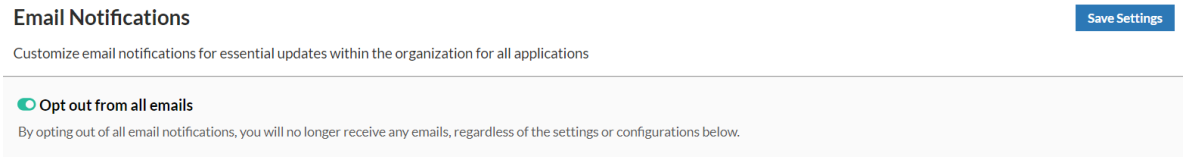
Email Notification

The *Email Notification* page allows you to customize email notifications for important events, vulnerability alerts, and product updates related to your applications within FortiDevSec.

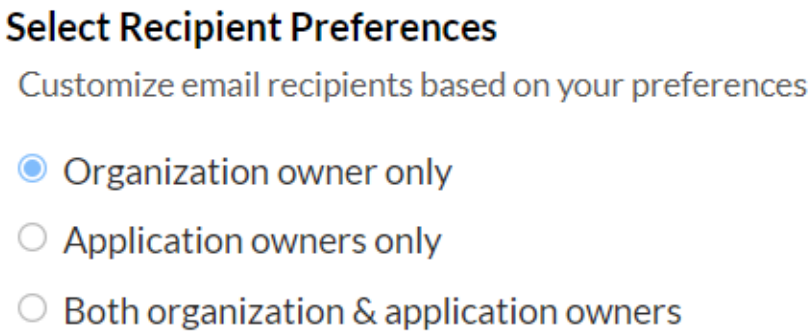
- Master users can be organization or application owners. Sub-users, IAM, and IdP users can only be application owners.
- IAM and IdP users are not currently supported for email notifications.
- Only the *Master User* can configure email notifications. IAM, IDP and sub users don't have the permission to configure email notifications.

Click *Save Settings* after making any changes to apply your notification preferences. The following settings can be configured.

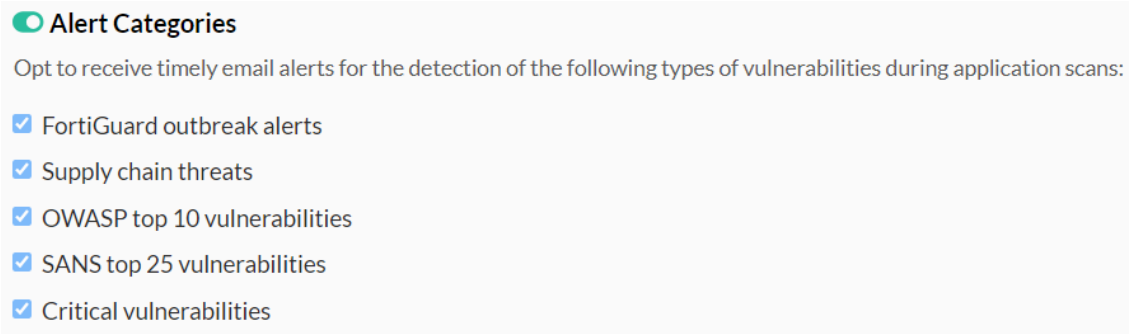
- **Opt out from all emails** - Toggle this setting to disable all email notifications, overriding any other settings.



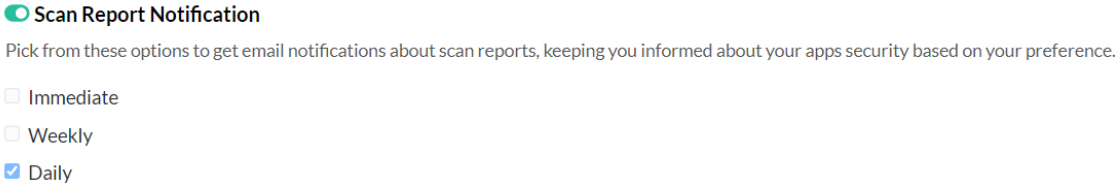
- **Select Recipient Preferences** - Choose who receives email notifications.
  - **Organization owner only** -Only the organization owner will receive email notifications.
  - **Application owners only** - Only the application owner will receive email notifications.
  - **Both organization & application owners** - Both organization owners and application owners will receive email notifications.



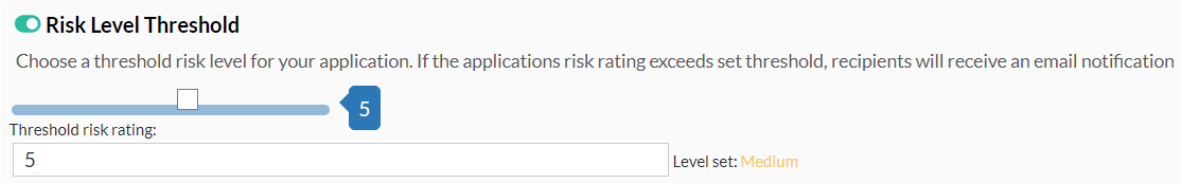
- **Alert Categories** - Enable email alerts for the following vulnerability types. The email notification will be sent after each scan completes.
  - **FortiGuard outbreak alerts** - Receive notifications about critical vulnerabilities identified by FortiGuard Labs.
  - **Supply chain threats** - Receive notifications about vulnerabilities introduced through third-party dependencies in your applications.
  - **OWASP Top 10 vulnerabilities** - Receive notifications about vulnerabilities belonging to the OWASP Top 10 list.
  - **SANS Top 25 vulnerabilities** - Receive notifications about vulnerabilities classified within the SANS Top 25 list.
  - **Critical vulnerabilities** -Receive notifications about high-risk vulnerabilities that could lead to severe compromise.



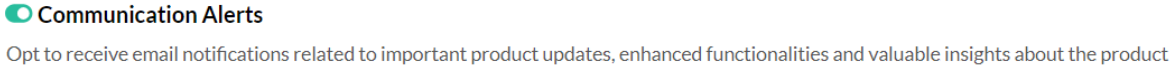
- **Scan Report Notifications** - Choose how often you want to receive scan report summaries. .
  - **Daily** - Receive a consolidated email notification summarizing scans performed each day.
    Notes:
    - Currently, the only supported option is *Daily*.
    - Reports are sent only to the organization owners. You must select *Organization owners only* or *Both organization & application owners* as recipients to receive daily reports.



- **Risk Level Threshold** - Set a risk rating threshold (from 0 to 9). If an application's risk rating exceeds this threshold, selected recipients will receive an email notification.



- **Communication Alerts** - Enable this option to receive emails about product updates, new features, and other relevant FortiDevSec announcements.



Note: By default, email notifications are configured with the following settings:

- **Recipients**: Organization owners
- **Risk Level Threshold**: 3
- **Communication Alerts**: Enabled

API Access

The API Access page allows you to generate and manage API tokens. These tokens enable secure integration of FortiDevSec data into your existing systems, and tools.

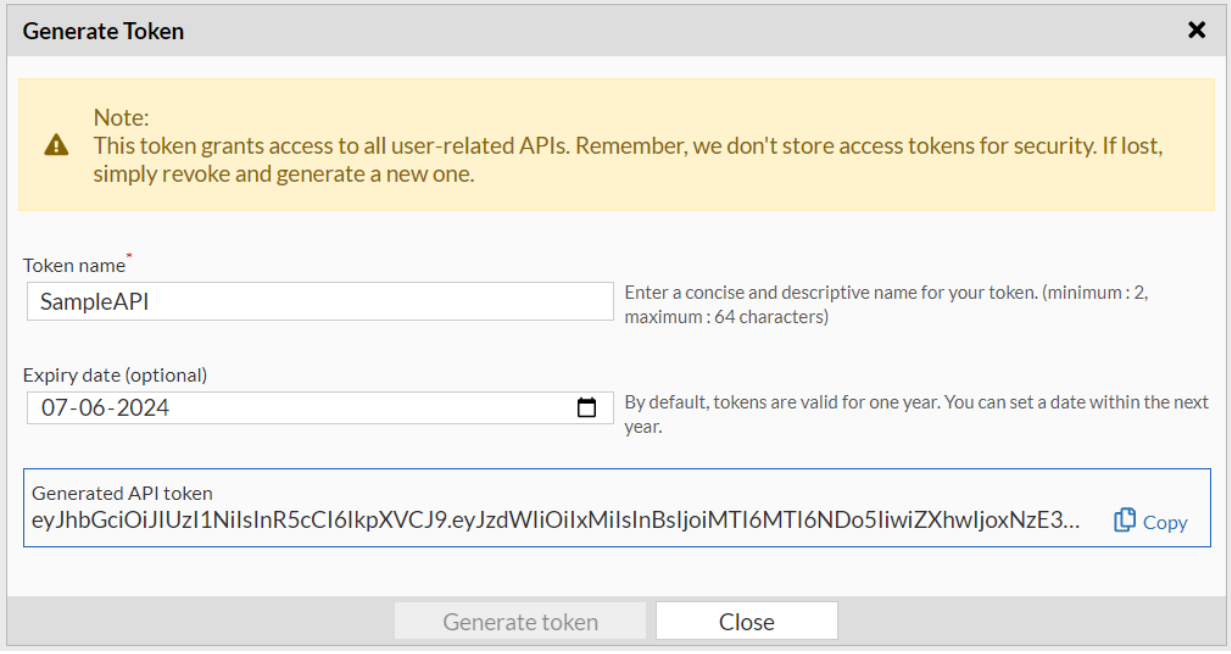
Note: Only the *Master User* has access to create API keys in the *API Access* page. IAM, IDP and sub users don't have the permission to create API key.

Generate API Token

Perform the following steps to generate API token.

1. Click the *Generate API Token* button. *The Generate Token* dialog opens.
2. Enter the following information.
   a. Token Name
   b. Expiry Date(Optional)
3. Click *Generate Token*.

The generated API Token is displayed. This is the only time it will be visible. Make sure to copy and store it in a secure location.



Manage API Tokens

The API Access page lists previously generated tokens with the following information:

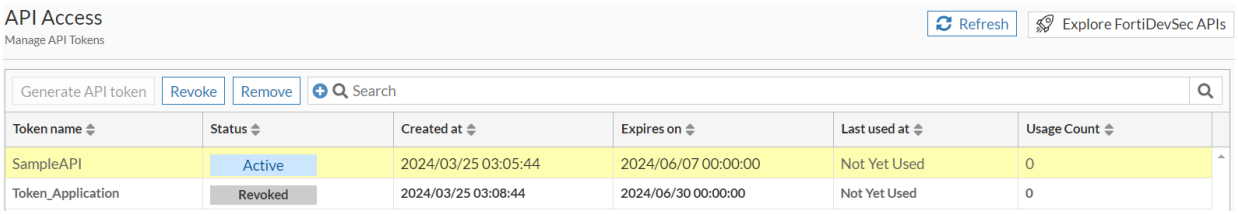
- **Token Name**
- **Status** - Active, Revoked, or Expired
- **Created at** - Date and time of token creation
- **Expires on** - Expiry date of the token
- **Last Used at** - Date and time the token was last used in an API request.
- **Usage Count** - Tracks the number of times the token has been used.

To disable a token preventing further use, select the token you want from the list and click *Revoke*.

To delete a token permanently, select the token you want from the list and click *Remove*.

To refresh the token information, click *Refresh*.

To view the FortiDevSec API documentation, click *Explore FortiDevSec APIs*.



Notes:

- FortiDevSec does not store API tokens for security purposes. If you lose a token, you must generate a new one.
- Revoke tokens that are no longer needed or might be compromised.

CI/CD Tools

The FortiDevSec SAST/DAST scanner Docker image is built for Linux. To run the FortiDevSec Linux-based Docker container on Windows, you must install Docker Desktop, which supports **Windows Subsystem for Linux 2 (WSL 2)**.

FortiDevSec supports scanning in the following CI/CD tools.

- AWS CodePipeline
- Azure DevOps
- Bamboo
- CircleCI
- Drone CI
- GCP Cloud Build
- GitHub Actions
- GitLab
- Jenkins
- Travis CI
- Bitbucket
- JFrog GitHub
- JFrog GitLab

AWS CodePipeline

Following is a sample code segment that can be configured in *buildspec.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
version: 0.1
phases:
    install:
        commands:
            - echo "Entered the install phase..."
        finally:
            - echo "This always runs even if the update or install command fails"
    pre_build:
        commands:
            - echo "Entered the pre_build phase..."
        finally:
            - echo "This always runs even if the login command fails."
    build:
        commands:
            - echo "Entered the build phase..."
            - echo "Build started on `date`"
        finally:
```

```
            - echo "This always runs even if the install command fails"
    post_build:
        on-failure: CONTINUE
        commands:
            - echo "Entered the post_build phase..."
            - echo "Build completed on `date`"
            - echo "Running FortiDevSec SAST scanner..."
            - env | grep -E "CODEBUILD_CI|CODEBUILD_BUILD_NUMBER|CODEBUILD_RESOLVED_
SOURCE_VERSION" > /tmp/env
            - "docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest"
```

Azure DevOps

Following is a sample code segment that can be configured in *azure-pipelines.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
trigger:
  - main

pool:
  vmImage: 'ubuntu-latest'

stages:
- stage: SAST
  displayName: 'Static Application Security Testing (SAST) Stage'
  jobs:
  - job: RunSAST
    displayName: 'Run SAST'
    steps:
    - task: Bash@3
      displayName: 'Install and Run SAST'
      inputs:
        targetType: 'inline'
        script: |
          env | grep -E "AZURE_HTTP_USER_AGENT|BUILD_BUILDID|BUILD_
SOURCEBRANCHNAME|BUILD_SOURCEVERSION" > /tmp/env
          docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in *azure-pipelines.yml* file to perform a DAST scan.

```
trigger:
  - main

pool:
  vmImage: 'ubuntu-latest'

stages:
- stage: DAST
  displayName: 'Dynamic Application Security Testing (DAST) Stage'
```

```
  jobs:
  - job: RunDAST
    displayName: 'Run DAST'
    steps:
    - task: Bash@3
      displayName: 'Install and Run DAST'
      inputs:
        targetType: 'inline'
        script: |
          env | grep -E "AZURE_HTTP_USER_AGENT|BUILD_BUILDID|BUILD_
SOURCEBRANCHNAME|BUILD_SOURCEVERSION" > /tmp/env
          docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source="$PWD",target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

Bamboo

Following is a sample code segment that can be configured in *bamboo.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
--

version: 2
plan:
    project-key: MYAPP
    name: Build the myapp
    key: MYAPP
 stages:
  -scan the myapp stage:
       jobs:
          -- Scan
Scan:
   tasks:
       - clean        # To keep the working directory clean
     -script:
      - env | grep -E "bamboo_buildNumber|bamboo_repository_branch_name|bamboo_
repository_revision_number" > /tmp/env
      - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in *bamboo.yml* file to perform a DAST scan.

```
--

version: 2
plan:
    project-key: MYAPP
    name: Build the myapp
    key: MYAPP
 stages:
  -scan the myapp stage:
       jobs:
```

```
          -- Scan
Scan:
   tasks:
       - clean        # To keep the working directory clean
     -script:
      - env | grep -E "bamboo_buildNumber|bamboo_repository_branch_name|bamboo_
repository_revision_number" > /tmp/env
      - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

CircleCI

Following is a sample code segment that can be configured in *circleci/config.yml* file to perform a SAST scan. Refer to the Orb Registry page to use the latest version.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
version: 2.1
jobs:
    SAST:
        machine: yes
        steps:
          - checkout
          - run: |
                env | grep -E "CIRCLECI|CIRCLE_BUILD_NUM|CIRCLE_BRANCH|CIRCLE_SHA1" >
/tmp/env
                docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
workflows:
    Scans:
        jobs:
          - SAST
```

Following is a sample code segment that can be configured in *circleci/config.yml* file to perform a DAST scan.

```
version: 2.1
jobs:
    DAST:
        machine: yes
        steps:
          - checkout
          - run: |
                env | grep -E "CIRCLECI|CIRCLE_BUILD_NUM|CIRCLE_BRANCH|CIRCLE_SHA1" >
/tmp/env
                docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
workflows:
    Scans:
        jobs:
          - DAST
```

Drone CI

Following is a sample code segment that can be configured in *drone.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
---
kind: pipeline
type: exec
name: SCAN

platform:
    os: linux
    arch: amd64

steps:
#Run FortiDevSec SAST Scanner, once the build step is done.
- name: SAST
  commands:
  - env | grep -E "DRONE|DRONE_BUILD_NUMBER|CI_COMMIT_BRANCH|CI_COMMIT_SHA" >
/tmp/env
  - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in *drone.yml* file to perform a DAST scan.

```
---
kind: pipeline
type: exec
name: SCAN

platform:
    os: linux
    arch: amd64

steps:
#Run FortiDevSec SAST Scanner, once the build step is done.
- name: DAST
  commands:
  - env | grep -E "DRONE|DRONE_BUILD_NUMBER|CI_COMMIT_BRANCH|CI_COMMIT_SHA" >
/tmp/env
  - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

GCP Cloud Build

Following is a sample code segment that can be configured in *cloudbuild.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
steps:

# Run FortiDevSec SAST Scanner, once the build step is done.
- name: 'gcr.io/cloud-builders/docker'
  entrypoint: bash
  args: ['-c','docker run --pull always --rm --env GCP_CLOUDBUILD_CI=$GCP_CLOUDBUILD_
CI --env BUILD_ID=$BUILD_ID --env BRANCH_NAME=$BRANCH_NAME --env COMMIT_SHA=$COMMIT_
SHA --mount type=bind,source=$(pwd),target=/scan
registry.fortidevsec.forticloud.com/fdevsec_sast:latest']
```

Following is a sample code segment that can be configured in *cloudbuild.yml* file to perform a DAST scan.

```
steps:

# Run FortiDevSec DAST Scanner, once the deploy step is done.
- name: 'gcr.io/cloud-builders/docker'
  entrypoint: bash
  args: ['-c','docker run --pull always --rm --env GCP_CLOUDBUILD_CI=$GCP_CLOUDBUILD_
CI --env BUILD_ID=$BUILD_ID --env BRANCH_NAME=$BRANCH_NAME --env COMMIT_SHA=$COMMIT_
SHA --mount type=bind,source=$(pwd),target=/scan
registry.fortidevsec.forticloud.com/fdevsec_dast:latest']
```

GitHub Actions

Following is a sample code segment that can be configured in *main.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
name: FortiDevSec Scanner CI
on:
    push:
        branches: [ master ]
    pull_request:
        branches: [ master ]
jobs:
    build:
        runs-on: ubuntu-latest
        steps:
        - uses: actions/checkout@v2
        - name: SAST
          run: |
            env | grep -E "GITHUB_ACTIONS|GITHUB_RUN_NUMBER|GITHUB_REF_NAME|GITHUB_
SHA" > /tmp/env
            docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in *main.yml* file to perform a DAST scan.

```
name: FortiDevSec Scanner CI
on:
    push:
        branches: [ master ]
    pull_request:
        branches: [ master ]
```

```
jobs:
    build:
        runs-on: ubuntu-latest
        steps:
        - uses: actions/checkout@v2
        - name: SAST
          run: |
            env | grep -E "GITHUB_ACTIONS|GITHUB_RUN_NUMBER|GITHUB_REF_NAME|GITHUB_
SHA" > /tmp/env
            docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

GitLab

Following is a sample code segment that can be configured in *gitlab-ci.yml* file to perform a SAST scan using **Shell executor** provided by the GitLab Runner.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
SAST:
  stage: build
  script:
    - env_file=`mktemp`
    - env | grep -E "GITLAB_CI|CI_BUILD_ID|CI_DEFAULT_BRANCH|CI_COMMIT_SHA|CI_
PIPELINE_IID" > $env_file
    - docker run --pull always --rm --env-file $env_file --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
    - rm $env_file
  tags:
    - devsecops
```

Following is a sample code segment that can be configured in *gitlab-ci.yml* file to perform a DAST scan using **Shell executor** provided by the GitLab Runner.

```
DAST:
  stage: build
  script:
    - env_file=`mktemp`
    - env | grep -E "GITLAB_CI|CI_BUILD_ID|CI_DEFAULT_BRANCH|CI_COMMIT_SHA|CI_
PIPELINE_IID" > $env_file
    - docker run --pull always --rm --env-file $env_file --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
    - rm $env_file
  tags:
    - devsecops
```

Jenkins

Following is a sample code segment that can be configured in **Jenkins > (Your App) > Configure > Add build step > Execute Shell** to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
env | grep -E "JENKINS_HOME|BUILD_ID|GIT_BRANCH|GIT_COMMIT" > /tmp/env
docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in **Jenkins > (Your App) > Configure > Add build step > Execute Shell** to perform a DAST scan.

```
env | grep -E "JENKINS_HOME|BUILD_ID|GIT_BRANCH|GIT_COMMIT" > /tmp/env
docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

Travis CI

Following is a sample code segment that can be configured in *.travis.yml* file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
language: python
python:
  - "3.6"
services:
  - docker
jobs:
  include:
    - stage: SAST
      script:
        - env | grep -E "TRAVIS|TRAVIS_BUILD_ID|TRAVIS_BRANCH|TRAVIS_COMMIT" >
/tmp/env
        - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
```

Following is a sample code segment that can be configured in *.travis.yml* file to perform a DAST scan.

```
language: python
python:
  - "3.6"
services:
  - docker
jobs:
  include:
    - stage: DAST
      script:
        - env | grep -E "TRAVIS|TRAVIS_BUILD_ID|TRAVIS_BRANCH|TRAVIS_COMMIT" >
```

```
/tmp/env
        - docker run --pull always --rm --env-file /tmp/env --mount
type=bind,source=$PWD,target=/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
```

Bitbucket

Following is a sample code segment that can be configured in your configuration file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
pipelines:
default:
       - step :
                runs-on:
                        - self.hosted
                        - linux
                name: Build and Scan
                services:
                        - docker
                script:
                        - env_file=`mktemp`
                        - env | grep -E "BITBUCKET_PROJECT_UUID|BITBUCKET_BUILD_
NUMBER|BITBUCKET_BRANCH|BITBUCKET_COMMIT" > $env_file
                        - docker run --pull always --rm --env-file $env_file -v
"$(pwd)":/scan registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
                        -rm $env_file
```

Following is a sample code segment that can be configured in configuration file to perform a DAST scan.

```
pipelines:
default:
       - step :
                runs-on:
                        - self.hosted
                        - linux
                name: Build and Scan
                services:
                        - docker
                script:
                        - env_file=`mktemp`
                        - env | grep -E "BITBUCKET_PROJECT_UUID|BITBUCKET_BUILD_
NUMBER|BITBUCKET_BRANCH|BITBUCKET_COMMIT" > $env_file
                        - docker run --pull always --rm --env-file $env_file -v
"$(pwd)":/scan registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
                        -rm $env_file
```

JFrog GitHub

Following is a sample code segment that can be configured in your configuration file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
name: sast
on:
       push:
               branches:
                       - main
jobs:
       run-container:
               runs-on:ubuntu-latest
               steps:
                       -name:Checkout code
                         uses:actions/checkout@v2
                       -name:Setup JFrog CLI
                         uses:jfrog/setup-jfrog-cli@v3
                       env:
                               JF_UR:${{ secrets.JF_URL }}
                               JF_ACCESS_TOKEN:${{ secrets.JF_ACCESS_TOKEN }}

                       -name: Run Docker Container
                       run:|
                               env_file=`mktemp`
                               env | grep -E "JFROG_CLI_BUILD_NUMBER" > $env_
file
                               docker run --pull always --rm --mount
type=bind,source="$(pwd)",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_
sast:latest
                               rm $env_file
```

Following is a sample code segment that can be configured in configuration file to perform a DAST scan.

```
name: dast
on:
       push:
               branches:
                       - main
jobs:
       run-container:
               runs-on:ubuntu-latest
               steps:
                       -name:Checkout code
                         uses:actions/checkout@v2
                       -name:Setup JFrog CLI
                         uses:jfrog/setup-jfrog-cli@v3
                       env:
                               JF_UR:${{ secrets.JF_URL }}
                               JF_ACCESS_TOKEN:${{ secrets.JF_ACCESS_TOKEN }}

                       -name: Run Docker Container
                       run:|
                               env_file=`mktemp`
                               env | grep -E "JFROG_CLI_BUILD_NUMBER" > $env_
file
                               docker run --pull always --rm --mount
type=bind,source="$(pwd)",target=/scan
```

```
registry.fortidevsec.forticloud.com/fdevsec_
dast:latest
                                rm $env_file
```

JFrog GitLab

Following is a sample code segment that can be configured in your configuration file to perform a SAST scan.

Note: Make sure to update the parameters in the sample code according to your environment before using it.

```
default:
       image:docker:latest

include:
       -remote:"https://releases.jfrog.io/artifactory/jfrog-
cli/gitlab/v2/.setup-jfrog-unix.yml"

jfrog-docker-build:
       variables:
               IMAGE_NAME:sample.jfrog.io/jfrog-gitlab-docker/jfrog-
docker-example-image:$CI_PIPELINE_IID
               JFROG_CLI_BUILD_NAME:JFROG_CLI_BUILD_NAME
               JFROG_CLI_BUILD_NUMBER:$CI_PIPELINE_IID

       tags:
               -gitlab-org-docker
       services:
               -docker:dind
       script:
               -env_file=`mktemp`
               -env | grep -E "JFROG_CLI_BUILD_NUMBER" > $env_file
               -docker run --pull always --rm --env-file $env_file --
mount type=bind,source="$(pwd)",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_sast:latest
               -rm $env_file
```

Following is a sample code segment that can be configured in configuration file to perform a DAST scan.

```
default:
       image:docker:latest

include:
       -remote:"https://releases.jfrog.io/artifactory/jfrog-
cli/gitlab/v2/.setup-jfrog-unix.yml"

jfrog-docker-build:
       variables:
```

```
                IMAGE_NAME:sample.jfrog.io/jfrog-gitlab-docker/jfrog-
docker-example-image:$CI_PIPELINE_IID
                JFROG_CLI_BUILD_NAME:JFROG_CLI_BUILD_NAME
                JFROG_CLI_BUILD_NUMBER:$CI_PIPELINE_IID

        tags:
                -gitlab-org-docker
        services:
                -docker:dind
        script:
                -env_file=`mktemp`
                -env | grep -E "JFROG_CLI_BUILD_NUMBER" > $env_file
                -docker run --pull always --rm --env-file $env_file --
mount type=bind,source="$(pwd)",target=/scan
registry.fortidevsec.forticloud.com/fdevsec_dast:latest
                -rm $env_file
```

Integrations

FortiDevSec supports the following integrations.

- FortiDevSec on Google Cloud Platform
- FortiDevSec Visual Studio Code Extension
- Plugins
  - Jira
  - FortiDAST App Config

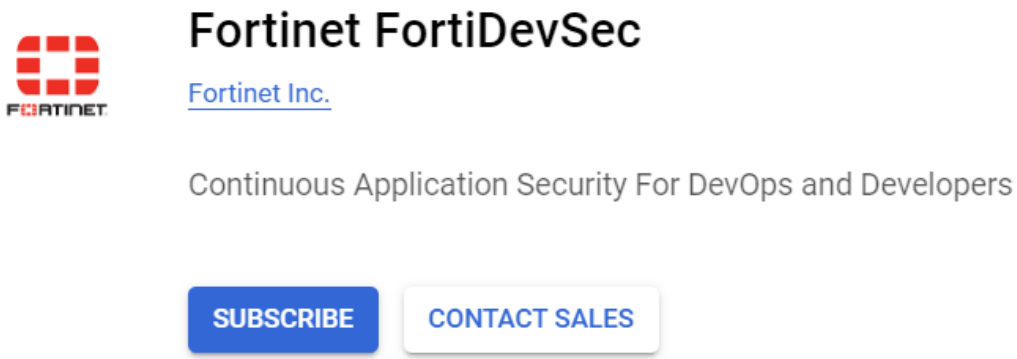
FortiDevSec on Google Cloud Platform

FortiDevSec is available for subscription on the Google Cloud Platform (GCP) Marketplace. FortiDevSec's subscription-based model ensures that you can take advantage of it's capabilities while paying only for what you use, making it an efficient and cost-effective solution for businesses of all sizes.

Prior to using FortiDevSec, you are required to register on the FortiCloud portal. See Registering on Forti Cloud.

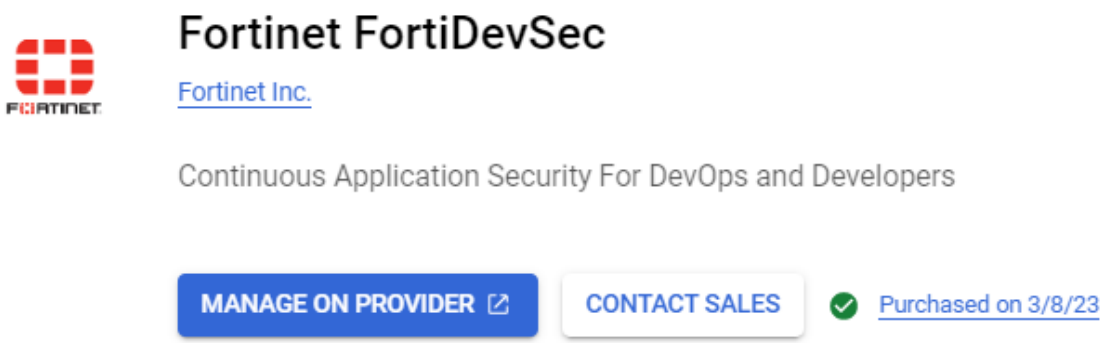
To subscribe to FortiDevSec in the GCP Marketplace, perform the following steps:

1. Navigate to the Google Cloud Platform (GCP) Console at console.cloud.google.com.
2. Click **Marketplace** icon in the left-hand navigation menu.
3. Search **FortiDevSec** in Google Cloud Marketplace and click FortiDevSec
4. On the FortiDevSec details page, review the application information and click **SUBSCRIBE**.



5. Choose the appropriate subscription plan based on your usage.
   Note: Only the monthly subscription plan is supported currently. Auto-renewal is enabled by default and can be disabled if required.
6. Review the terms and conditions of the subscription agreement.
7. Click **SUBSCRIBE.** Usually it takes few hours for GCP to activate the subscription.
8. The **MANAGE ON PROVIDER** will be displayed only after successful activation of the subscription. Click

MANAGE ON PROVIDER to launch FortiDevSec.



Notes:

- Ensure that you log in to FortiDevSec through GCP when logging in for the first time.
- The usage for the FortiDevSec subscription is calculated based on identifying active users as individuals who have made a commit within the previous 90 days in the repository.
- If you have only a FortiDevSec GCP license, you cannot perform FortiDAST scans. However, you can purchase a standalone FortiDAST license using the same email registered for your FortiDevSec GCP license. Once you have done this, you will be able to initiate FortiDAST scans through FortiDevSec.

In the FortiDevSec GUI, click **Organization** on the portal GUI homepage, to view the licensing information currently in use.

FortiDevSec Visual Studio Code Extension

The *FortiDevSec Visual Studio Code Extension* allows you to scan your code, view results, and explore vulnerability details directly from Visual Studio(VS) Code IDE.

Prerequisites

Ensure the following prerequisites are met before installing *FortiDevSec Visual Studio Code Extension*.

- Internet connection for accessing FortiDevSec services during use.
- A valid FortiDevSec license. See Licensing.
- Docker installed and available to run as non-root user. To install the Docker engine across different platforms, see Docker.

Supported operating systems and architecture

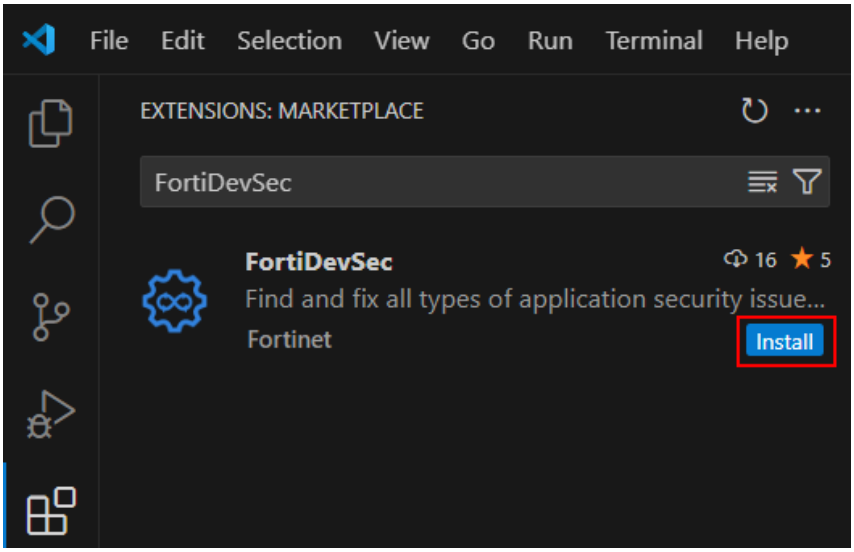
You can use the *FortiDevSec Visual Studio Code Extension* in the following environments:

- Linux - AMD64
- Windows - AMD64
- macOS - AMD64

Installing FortiDevSec Visual Studio Code Extension

Perform the following steps to install the FortiDevSec extension in VS Code.

1. Open **VS Code**.
2. Click the **Extensions** icon in the **Activity Bar** on the left side of VS Code.
3. Use the search bar at the top of the **Extensions** view to find the **FortiDevSec** extension.
4. Click **Install**. FortiDevSec icon is displayed in the activity bar.



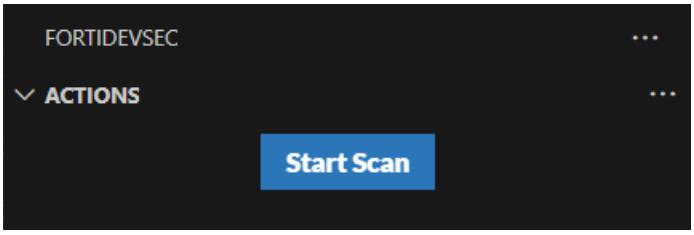
For more information on installing extensions, see Visual Code Studio User Guide.

Initiating Scan

Once the FortiDevSec extension is installed successfully, perform the following steps to run security scan.

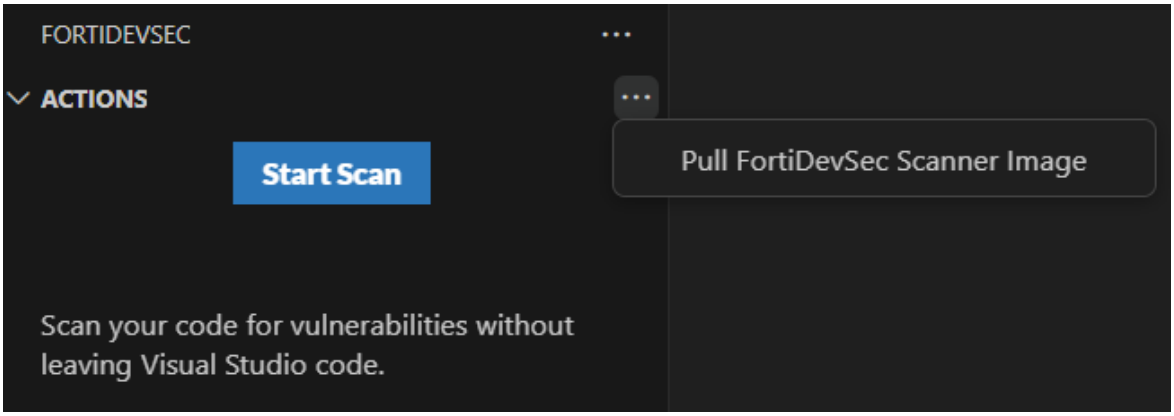
1. Login to FortiDevSec UI portal.
2. Add a new application. See Adding a New Application.
3. Download the **fdevsec.yaml** file.
4. Copy the downloaded fdevsec.yaml file to the root directory of your repository.
   Note: Ensure that the Docker is running on your system.

5. Open your repository in VS Code.
6. Click FortiDevSec icon in the activity bar.
7. Click the **Start Scan** button to initiate the scan.



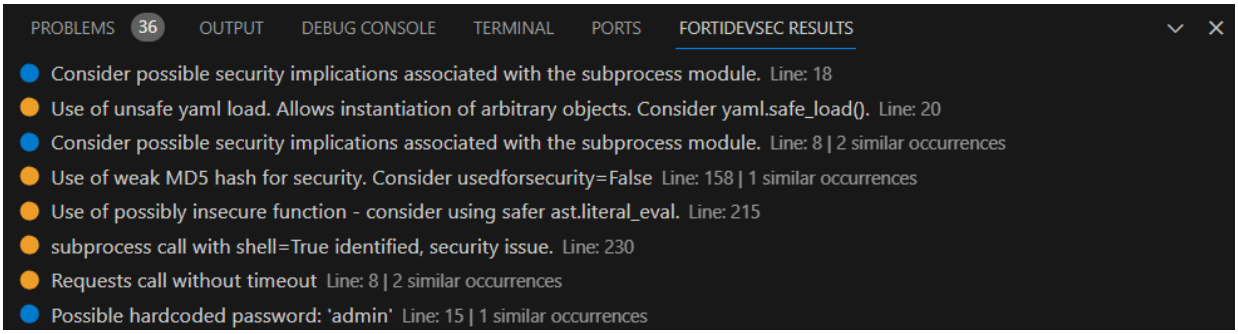
Note: Keep Visual Studio Code IDE open while the FortiDevSec scan runs. Closing VS Code IDE before the scan finishes prevents downloading results from the FortiDevSec cloud.

To pull the latest FortiDevSec scanner images, click the ellipsis menu in the **Actions** section and click **Pull FortiDevSec Scanner Image**.

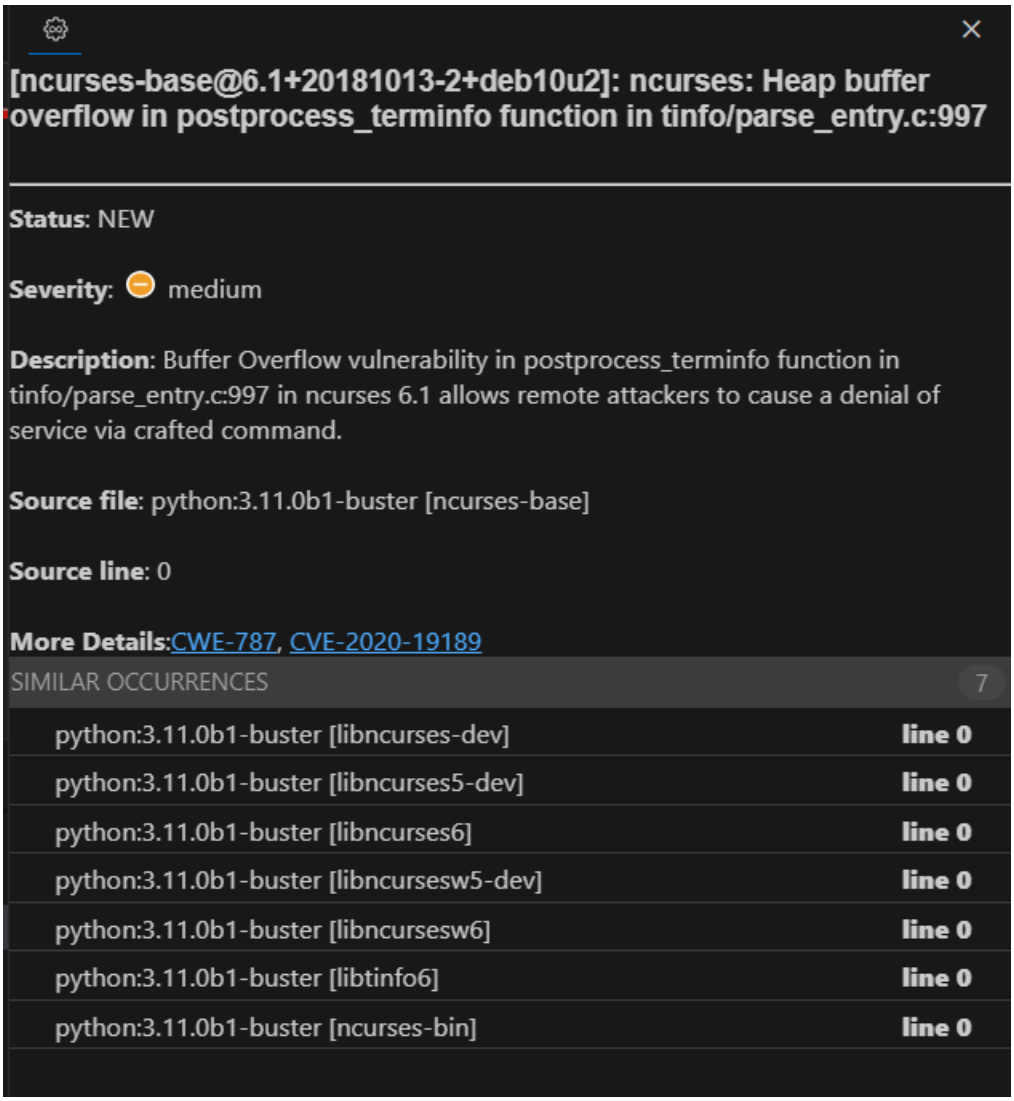


Viewing Scan Results

Upon completion of the scan, the results will be downloaded automatically. The **FortiDevSec Results** view will then display the detected vulnerabilities.



Clicking a vulnerability reveals its detailed view in the right pane. The detailed view includes the following information.

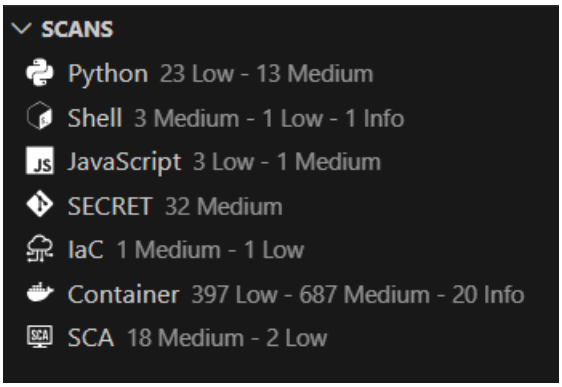


Field	Description
Status	Current status of the vulnerability.
Severity	Risk rating assigned by FortiDevSec.
Source file	The associated file and the line number that the vulnerability is found in.
Source line	
More Details	Displays the associated **CWE** and **CVE** (if any). Click on the CWE/CVE link to view details.
Similar Occurrences	The number of similar occurrences that the vulnerability is found in, click on each instance to view details.

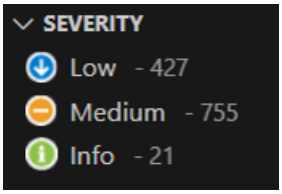
Filtering Scan Results

You can filter the scan results based on scan type or severity.

- **Scan Type**: Click the desired scan type in the **Scans** section. To clear the filter, click anywhere within the blank area of the **Scans** section.



- **Severity**: Click the preferred severity level in **Severity** section. To remove the severity filter, click anywhere within the blank area of the **Severity** section.



  You can combine both **Scans** and **Severity** to filter the scan results. For example, select python for scan type and medium for severity to view results specific to python and are of medium severity.

Plugins

You can configure FortiDevSec plugins during a new application creation or in the scanned applications details page. FortiDevSec supports the following plugins.

- JIRA
- FortiDAST App Config

Jira

FortiDevSec allows you to integrate Jira projects for unified bug management. Jira integration is optional and avoids the overhead of maintaining the detected vulnerabilities in multiple systems.

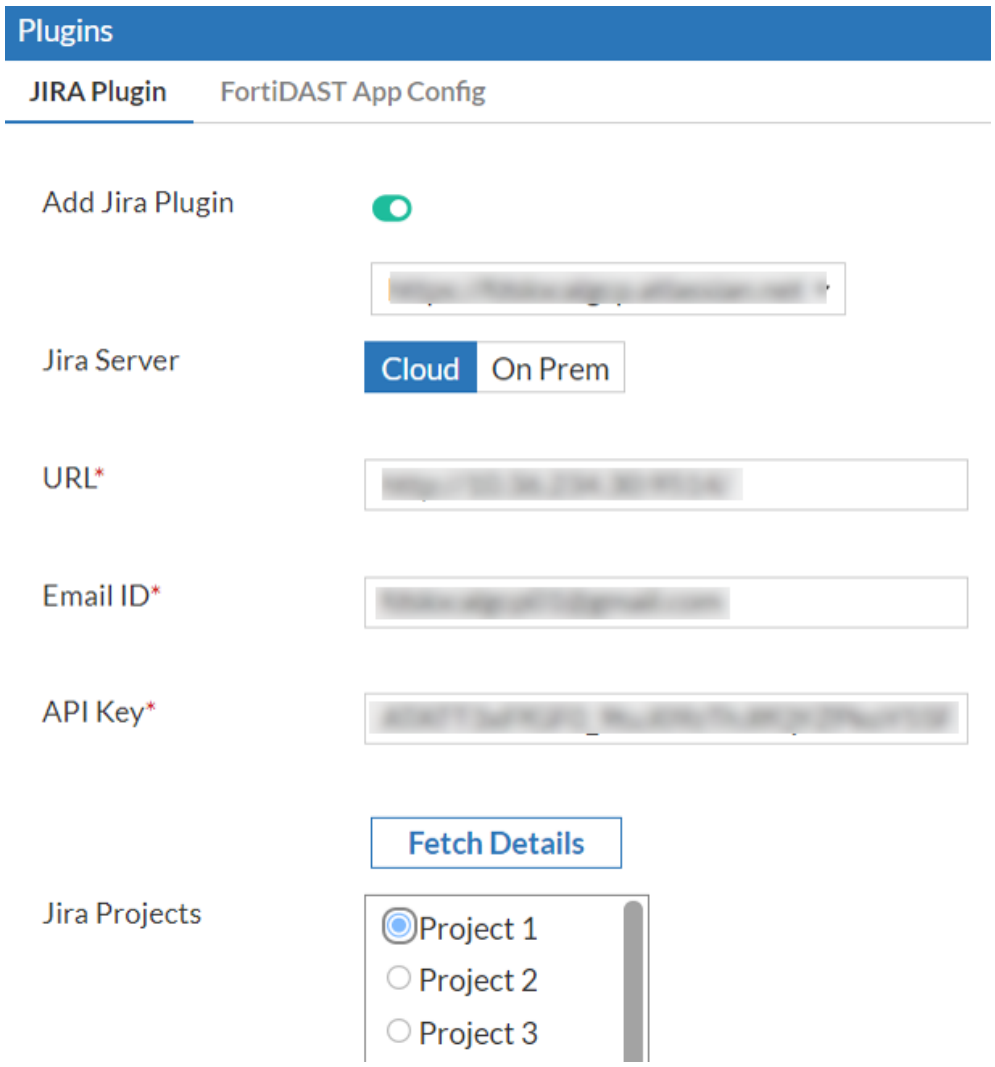
Before integrating Jira with FortiDevSec, use the following links to generate the **API Key** and **Personal access token (PAT)** for Jira cloud and on-prem respectively.

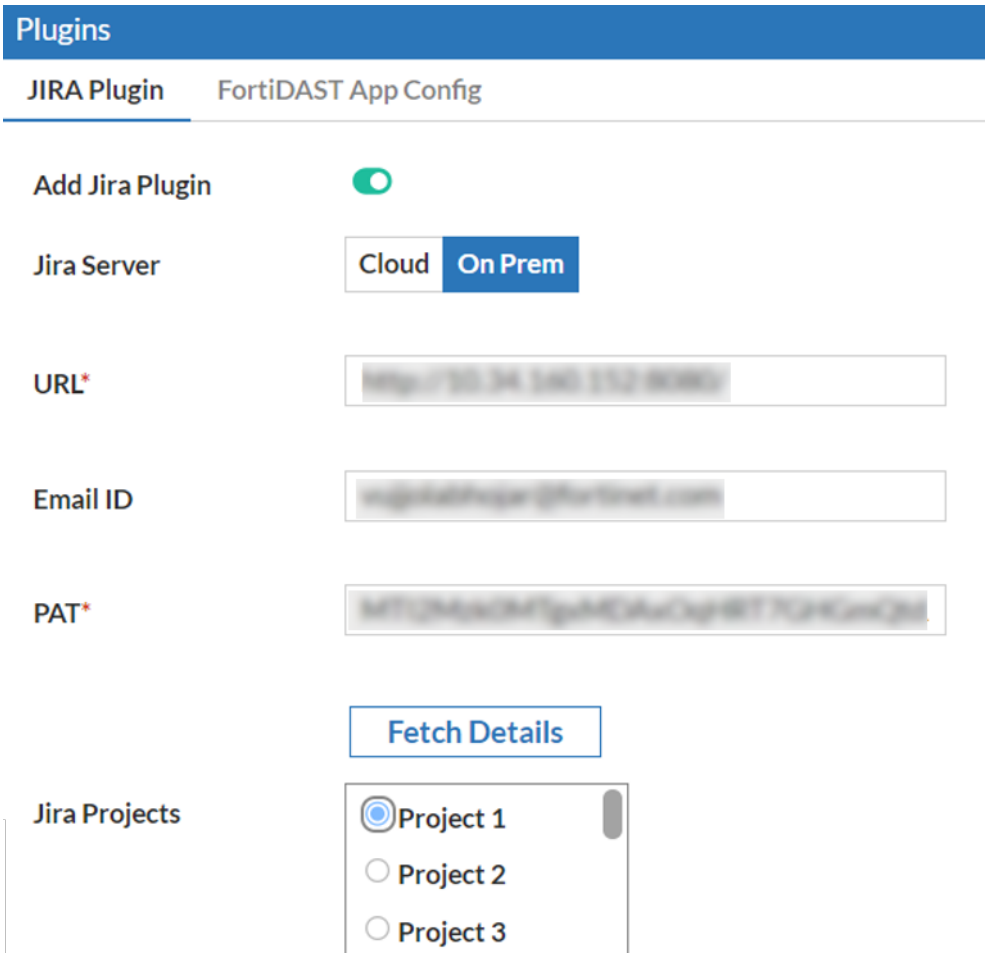
- Jira Cloud - https://id.atlassian.com/manage-profile/security/api-tokens
- Jira On-prem - https://confluence.atlassian.com/enterprise/using-personal-access-tokens-1026032365.html

To add Jira integration to new application, see Adding a New Application.

To add/update Jira integration to an existing application:

1. In the *FortiDevSec* > *App Directory* page, click the desired application .
2. In the scanned application details page, click **Settings**.
3. in *Application details* page, under *Application Plugins* section, click **JIRA Plugin Configure**.
4. Toggle the **Add Jira Plugin**, if not already enabled. If Jira plugin is already enabled, update the required fields and click **OK**.
5. Select the **Cloud** or **On Prem** option for the Jira Server.
6. Enter the **URL**.
7. Enter the **Email ID**.
8. Enter **API Key**, if cloud server is selected. Enter **PAT** if on prem server is selected.
9. Click **Fetch Details** and select the Jira projects.
10. Click **OK**.





Notes:

- Jira plugins are supported per FortiDevSec application.
- When vulnerability issues are reported in the FortiDevSec GUI, the associated Jira project is also updated.
- The vulnerability updates in Jira are synchronized to FortiDevSec prior to scanning the application. You can also synchronize the Jira updates manually.
- FortiDevSec relies on the built-in Jira priority system: **Highest, High, Medium, Low,** and **Lowest**. Please configure your Jira project integrated with FortiDevSec, to align with these priorities.

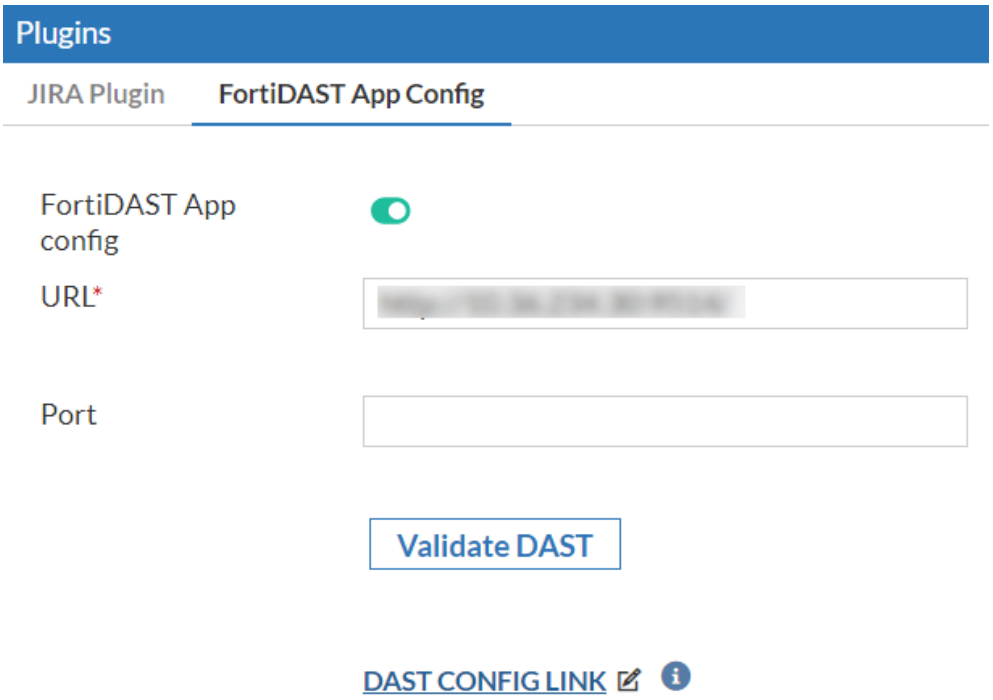
FortiDAST App Config

FortiDAST App Config plugin allows you to configure the target asset for DAST scanning in FortiDevSec GUI. Once the target asset URL validated you can perform additional configuration in FortiDAST portal.

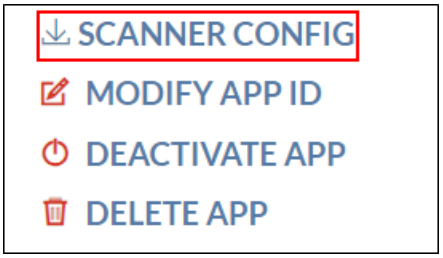
To configure FortiDAST scanning for a new application, see Adding a New Application.

To add/edit FortiDAST configuration for an existing apllication:

1. In the FortiDevSec dashboard, click the desired application.
2. In the scanned application details page, click **Settings**.
3. in *Application details* page, under *Application Plugins* section, click **FortiDAST Configure**.
4. Toggle the **FortiDAST App config**, if not already enabled. If FortiDAST plugin is already enabled, update the required fields and click **OK**.
5. Enter the target asset **URL** and **port** number.
6. Click **Validate DAST**.
7. Once the URL is validated, click **DAST Config Link** to perform additional configuration in FortiDAST portal. See Configuring FortiDAST Scanner.



8. Once you complete the configuration in FortiDAST portal, return to the FortiDevSec tab and click **OK**.
9. To download the updated yaml configuration file, click **Scanner Config**.
   Note: To perform DAST scan, uncomment the dast configuration in fdevsec.yaml file even when FortiDAST asset/URL is configured through GUI plugin.



10. Perform the scan. See Running the Security Scan.

Notes:

- You can disable or modify the URL at any point after app creation from the scanned application details page. The configured URL will be updated accordingly, but the last scan results will be displayed until the next scan.
- If the URL is not configured or disabled in the GUI, the URL mentioned in the YAML file or sent through the command line will be used for the scan.
- If you have already configured a URL in the FortiDevSec GUI and enabled it, but configured a different URL using the YAML config or command line, the scanner will give you an error message because the two URLs do not match. To fix this, you must either update the URL in the YAML config or command line to match the one in the GUI, or disable the URL scan option in the GUI.

Frequently Asked Questions (FAQs)

Can I run a DAST scan on the web applications hosted on the same local host?

Yes, but you need to specify the correct hostname or IP address of the web application, which a scanner Docker container or FortiDAST can resolve. Do not use *localhost* or *127.0.0.1* in the URL as this does not work.

Do I require a FortiDAST license to run a DAST scan?

The FortiDAST license is included in the FortiDevSec standard license and supports up to a maximum of 5 assets/apps. Use FortiDevSec FortiDAST Add-on license to expand upon the standard license to support additional assets/apps. See Licensing.

Note: If you already own the FortiDAST standard license, it can be used to expand upon the FortiDevSec standard license to perform DAST scanning of additional assets/apps.

Do I need to install a Docker engine in the host/machine to run a SAST/DAST scan?

Yes, since the FortiDevSec SAST and DAST scanners are docker images, you are required to install a Docker engine in that host/machine with the required user access/permission, to scan (automatic/manual) through the CI/CD pipeline. See Prerequisite.

When do the vulnerabilities from FortiDevSec get populated to the configured Jira project for the FortiDevSec application?

The identified vulnerabilities in the FortiDevSec application are populated to the configured Jira project only after the scan or rescan of that application. Issues are pushed in batches after each individual scanner finishes.

Note: Only the bug tracking project template, under the (Jira) software development, is currently supported for exporting vulnerabilities from FortiDevSec to Jira.

Does any change of the vulnerability status in the FortiDevSec application get synchronized to the configured Jira Project?

No. Only the vulnerability status updated in the configured Jira Project (Cloud or On-Prem) are synchronized to the FortiDevSec application. Status updates from FortiDevSec to Jira is not currently supported.

How does the status mapping work between the FortiDevSec vulnerabilities and Jira project issues?

The status mapping is as follows.

Jira Status	FortiDevSec Status
TO DO	New
DONE	Fixed
IN REVIEW/IN PROGRESS	Confirmed

How to generate the API Key for Jira Cloud and PAT (Personal Access Token) for Jira On-Prem, required for Jira integration with FortiDevSec application?

Use the following links to generate the API Key and PAT for Jira Cloud and On-Prem respectively.

- Jira Cloud - https://id.atlassian.com/manage-profile/security/api-tokens
- Jira on-Prem - https://confluence.atlassian.com/enterprise/using-personal-access-tokens-1026032365.html

What happens when I revoke the API Key in Jira cloud?

FortiDevSec will not be able to add or update the vulnerabilities in Jira, so you need to generate a new API key.

Will the vulnerability status updates done in JIRA automatically be synchronized with FortiDevSec?

No, you must manually synchronize JIRA updates using the **Sync** option in FortiDevSec. See Viewing the Scan Result.

What happens if I delete the sample app from the dashboard as an Org owner (master user)?

Deleting the sample app from the organization owner's account will remove it for all users in the organization, including users managed by both Identity and Access Management (IAM) and an Identity Provider (IdP).

How long are scan results retained in the Consolidated Result API?

The Consolidated Result API allows you to download results from recent scans for up to 10 days. After 10 days, a rescan is required to access the latest results.

What happens if a member is added to two different member groups with different permissions for the same application group?

The member will be assigned the highest permission level granted across all member groups.

Can an application belong to multiple application groups?

No, an application can only be a member of one application group at a time.

Do IAM and IDP users receive email notifications about shared group request decisions?

No, IAM and IDP users do not receive email notifications regarding the approval or rejection of their shared group requests.

Do moderators have the permission to change application group visibility?

Yes, moderators have the authority to change the visibility of an application group. This allows them to manage the group's accessibility and control who can join.

I am getting an "Unable to Download" error when trying to export SBOM vulnerabilities in CycloneDX format after upgrading FortiDevSec. What should I do?

A rescan is required after upgrading FortiDevSec to export SBOM results in CycloneDX format.

Why do I get the error "docker: image operating system 'linux' cannot be used on this platform" when trying to run the FortiDevSec scanner Docker image on Windows?

The FortiDevSec SAST/DAST scanner Docker image is built for Linux. To run the FortiDevSec Linux-based Docker container on Windows, you must install Docker Desktop, which supports **Windows Subsystem for Linux 2 (WSL 2)**.

www.fortinet.com