FORTINET

Administration Guide

FortiSOAR 7.6.2

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

April, 2025
FortiSOAR 7.6.2 Administration Guide
00-400-000000-20210113

TABLE OF CONTENTS

Change Log **10**
Overview **11**
Common Tasks 11
Tasks and Permissions 12
Guide to setting up FortiSOAR 13
System Configuration **15**
General 17
Configuring System and Cluster Health Monitoring 17
Configuring Comments 19
Setting the formats for Date and DateTime fields on the FortiSOAR UI 20
Customizing the behavior of the FortiSOAR Setup Guide 20
Setting a language other than English for your FortiSOAR system 22
Configuring Themes 25
Configuring Default Country Code 25
Configuring Navigation Preferences 25
Enabling Light Mode Setting 25
Application Configuration 26
Purging audit logs, executed playbook logs, and recycle bin records, and reclaiming unused disk space 26
Configuring the logging level for Playbook Execution Logs 34
Configuring Playbook Recovery 35
Configuring playbook log movement to historical storage 35
Configuring the Simplified Expression View 36
Configuring the default timezone for exporting reports 36
Managing user listings in People Lookup fields 36
Enabling MIME type validations for file uploads 37
Log Forwarding 37
Persisting the FortiSOAR logs 39
Environment Variables 40
Branding 41
System Fixtures 43
Audit Log 46
Viewing Audit Log 49
Viewing User-Specific Audit Logs 54
Viewing Audit Log in the detailed view of a record 54
Purging Audit Logs 57
Troubleshooting issues with Audit Logs 59
Audit Logs not processed when file size exceeds 64MB 60
License Manager 60
Notifications 61
Adding Delivery Rules 63
Setting up Notification Channels 67
Working with Delivery Rules and Notification Channels 70
Modifying the 'Notify On Pending External Manual Input' delivery rule post- 72

upgrade to release 7.3.0 or later
Purging of notifications 73
Data Archival 74
Methods of Setting Up Data Archival 75
Setting up an External Database for Data Archival 75
Configuring various settings for Data Archival 77
Viewing and Searching Archived Records 79
Blocking specific HTML tags and attributes 81
Security Management 83
Important Concepts 83
Authentication versus Authorization 83
Users and Appliances 83
Teams and Roles 84
Security Management Menus 84
Team Hierarchy 84
Teams 85
Roles 85
Users 85
Access Keys 85
Authentication 85
Password Vault 86
Configuring Team Hierarchy 86
Relationships 86
Using the Editor 88
Configuring Teams 91
Editing Teams 91
Configuring Roles 93
Default Roles 95
Modules in the Roles Page 97
Adding Roles 98
Assigning Roles to Users and Appliances 99
Configuring User Profiles 100
Adding Users 100
User Profiles 102
Managing Access Keys 112
Managing API key-based authentication for appliances 112
Managing HMAC authentication for appliances 117
Appliance Profile 119
Playbook Appliance 120
Troubleshooting 121
Configuring Authentication 122
Configuring Accounts 122
Configuring Two-Factor Authentication (2FA) 125
Configuring LDAP / AD 130
Configuring SSO 132
Configuring FortiSOAR authentication with a RADIUS server 185
Configuring the Password Vault Manager 189

Delete Users 192
Application Editor 194
Module Editor 194
Creating a New Module 195
Modifying an existing module 199
Saving your changes 210
Viewing your changes 210
Publishing modules 211
Picklist Editor 211
Creating or modifying a picklist 212
Pre-Processing Rules 214
Adding a new 'Drop' type pre-processing rule 215
Adding a new 'Update' type pre-processing rule 219
Supported Operations on the Pre-Processing Rules Page 223
Navigation Editor 224
Modifying the Navigation bar 225
Correlation Settings 227
Recommendation Engine 230
Permissions required 231
Record Similarity, Field Suggestions, and Playbook Suggestions 232
Phishing Classification 245
Export and Import Wizards 252
Permissions required 253
Export Wizard 253
Import Wizard 268
Recycle Bin 282
Permissions Required 282
Using the Recycle Bin 283
Behavior of Recycle Bin in the case of MSSP environments 284
Segmented Network Support 286
Overview 286
FortiSOAR Agent CLI 286
Invoke connector actions using FSR agents in segmented networks 287
Minimal permissions required 287
Installing a connector on an FSR agent 287
Configuring connectors 291
Running remote actions 291
Upgrading an FSR Agent 296
Upgrading the configuration on an FSR Agent machine 298
Running unauthenticated manual inputs in segmented networks 299
Troubleshooting 300
Files to be used for troubleshooting 300
The FSR agent remains stuck in the "Remote Node Unreachable" state after users update the certificates or if they change the authentication strategy 300
Deactivated FSR agent does not come back to the connected state even after activating the FSR agent 300
FSR Agents configuration page displays a "Agent <Agent UUID> does not 301

exist" error when you click the Export Config link
Post restore to release 7.3.0, switching the SME of any Tenant or FSR Agent, the status of the tenant or agent is displayed as "Remote Node Unreachable" 301
FortiSOAR Admin CLI 302
Prerequisites 302
FortiSOAR Admin CLI - Usage 302
CLI commands used for forwarding FortiSOAR logs 312
High Availability and Disaster Recovery support in FortiSOAR 314
Overview 314
High Availability Types supported with FortiSOAR 314
High Availability with an internal PostgreSQL database 314
High Availability with an externalized PostgreSQL database 316
Cluster Licensing 317
Viewing and updating the license of an HA cluster 318
Prerequisites to configuring High Availability 319
Process for configuring High Availability 320
Steps to configure FortiSOAR HA cluster with an internal PostgreSQL database 320
Steps to configure FortiSOAR HA cluster with an external PostgreSQL database 321
Takeover 321
RabbitMQ clustering across all active HA nodes 322
Usage of the csadm ha command 323
Overview of nodes in a FortiSOAR HA cluster 327
Checking replication between nodes in an active-passive configuration 327
Installation of connectors on nodes in a HA cluster 327
Changing the hostname of primary and secondary nodes in an HA cluster 328
Changing the hostname of the primary node 328
Changing the hostname of a secondary/passive node 329
Upgrading an HA cluster 329
Health Checks in FortiSOAR HA nodes 329
Load Balancer 330
Setting up HAProxy as a TCP load balancer fronting the two clustered nodes 330
Configuring FortiSOAR in FortiADC 331
Using the Gobetween load balancer 337
Behavior that might be observed while publishing modules when you are accessing HA clusters using a load balancer 341
Extending support for two NICs on a FortiSOAR appliance for controlled traffic routing 341
Section 1: Rocky Linux or RHEL changes for multihoming (MultiNIC) 343
Section 2: FortiSOAR changes for Multihoming 345
Setting up a High Availability FortiSOAR cluster in the AWS Cloud with Aurora as the external database 347
Configuration Details 347
Verifying FortiSOAR functionality with the Aurora external database 350
Verifying FortiSOAR cluster failover to another region 350
FortiSOAR Nodes Hydration 351

Upgrading Hydrated FortiSOAR Nodes 352
Tunables 352
Best practices 353
Best practices for changing the shared memory size on HA nodes after setting up an HA cluster 353
Monitoring health of HA clusters 354
Understanding HA Cluster Health Notifications 354
Troubleshooting issues based on the notifications 357
Sample scale test that were done in the lab to understand the behavior of 'csadm ha get-replication-stat' 359
Disaster Recovery 363
FAQs 363
Troubleshooting 364
Failure to create an HA cluster 364
Timeout failure while creating an HA cluster 365
Unable to add a node to an HA cluster using join-cluster, and the node gets stuck at a service restart 366
Fixing an HA cluster whose Primary node is halted and then resumed 366
Unable to join a node to an HA cluster when a proxy is enabled 366
Changes made in nodes in an active-active cluster fronted with a load balancer take some time to reflect 367
Post Takeover the nodes in an HA cluster do not point to the new active primary node 367
After performing the leave-cluster operation, the license is not found on a secondary node 367
The leave-cluster operation fails at the "Starting PostgreSQL Service" step when a node in the cluster is faulted 369
Resetting the password for an instance that is part of active/active cluster causes the other instances of that cluster to not able to log in to FortiSOAR 369
The database of a secondary node in the HA cluster is out of synch with the database of the primary node 370
Failure of the 'leave-cluster' command on a secondary node that had its database externalized before joining the HA cluster, and that is currently using an internal database 370
Common failures observed during the join-cluster operation 371
Elasticsearch Configuration 374
Externalization and Authentication of Elasticsearch 374
Migration of Elasticsearch data 376
Troubleshooting Tips 376
FortiSOAR Search Errors 376
Optimizing the reindexing of Elasticsearch data 377
Externalization of your FortiSOAR PostgreSQL database 378
Prerequisites 378
Externalizing FortiSOAR databases 379
Setting up an externalized database on the cloud 381
Setting up an externalized database without data migration 382
Installing the pg_squeeze and pg_repack extensions 383

Installing pg_squeeze ... 384
Installing pg_repack ... 386
Troubleshooting ... 386
Unable to edit connector configuration on new externalized FortiSOAR instances ... 386
Backing up and Restoring FortiSOAR ... 388
Prerequisites ... 388
Backup Process ... 388
Data that is backed up during the backup process ... 389
Prerequisites to running the backup process ... 389
Performing a backup ... 389
Restore Process ... 390
Restoring data ... 391
Restoring a backup of an FortiSOAR instance with an externalized database to an instance that is not externalized ... 392
Backup and Restore process for FortiSOAR High Availability systems ... 392
Regenerating RabbitMQ certificates when you are creating an HA cluster using a restored node ... 393
Backup and Restore process for the External Secure Message Exchange systems 393
Troubleshooting Backup and Restore Issues ... 394
Post-restore the FSR agent status is not updated as "Remote Node Unreachable" ... 394
About FortiSOAR ... 395
Downloading FortiSOAR logs ... 397
Monitoring FortiSOAR ... 398
Benefits of monitoring ... 398
Manually setting up monitoring for each FortiSOAR component ... 399
Monitoring uptime of FortiSOAR ... 399
Monitoring FortiSOAR services ... 399
Monitoring databases ... 400
Monitoring Disk Space Utilization ... 400
Monitoring CPU and Memory Utilization ... 401
Monitoring connectors ... 401
Monitoring workflows ... 401
FortiSOAR integration with FortiMonitor ... 403
Setting up a FortiSOAR instance to be monitored using FortiMonitor ... 403
Monitoring FortiSOAR using FortiMonitor ... 404
Components that are monitored out-of-the box by FortiMonitor on FortiSOAR 404
FortiSOAR-specific components that are monitored by FortiMonitor ... 405
OnSight Collector ... 407
Frequently Asked Questions ... 408
Q. How to add queued playbooks as a metric for monitoring using the FortiSOAR template ... 408
Q. How to change the monitoring interval for the License expiry metric ... 410
Q. How to modify the timings of generating a Takeover Incident ... 411
Q. How to remove stale entries on FortiMonitor for FortiSOAR metrics after a 413

takeover
Debugging, Troubleshooting, and Optimizing FortiSOAR 415
List of logs used for troubleshooting FortiSOAR 415
List of key FortiSOAR services and processes 418
Configurations required for exporting of records with unsupported character sets in the PDF format 420
Regenerating Self-Signed certificates 421
Additional settings for record similarity and field predictions 422
Configure the maximum number of records to be fetched in one call 422
Change the default value of some of the user profile parameters 423
Security considerations for Websockets 425
Backup of the most recent System View Template (SVT) before saving the edited SVT 425
Automation added to fetch Google Chrome updates 426
Troubleshooting Tips 426
Your Workflow data size has increased 426
Error displayed while performing a search operation in FortiSOAR 426
Reindexing FortiSOAR modules for search 427
FortiSOAR crashing with "out of memory" errors 427
Changing Postgres worker memory 427
Crashing of the Tomcat server when the 'Detailed' mode for Log Forwarding is enabled 428
Recommendation Engine does not work and displays an "ML Service not running" error 428
Enabling PostgreSQL logging to identify API requests that are taking longer than expected 429

Change Log

Date	Change Description
2025-07-25	Added a note clarifying that the external database must meet the specifications of a FortiSOAR node to the Externalization of your FortiSOAR PostgreSQL database chapter.
2025-05-28	Updated the steps in the *Setting up an externalized database without data migration* topic of the Externalization of your FortiSOAR PostgreSQL database chapter.
2025-04-29	Initial release of 7.6.2

Overview

Use the administration guide to understand how to customize and administer FortiSOAR, including system, security and user management, and configuring templates.

When you log on to FortiSOAR for the first time as a csadmin user, you will be mandated to change your password. This enhances the security of your csadmin account and prevents unauthorized parties from accessing the administration account for FortiSOAR. New passwords that are set must contain at least 8 characters, one lower-case alphabet, one upper-case alphabet, one digit, and any one of the following special characters ~ ! @ # $ % ^ & * | ? _
Ensure that you note down your csadmin password since if you forget your initial csadmin password, then you have to request FortiSOAR to reset this password. Also, when you are changing your csadmin password, you must ensure that you also update the email ID that is specified for csadmin, which by default is set to `soc@fortinet.com` (which is not a valid email ID). You can change the email ID by clicking the **User Profile** icon ( ) to open the `User Profile` page and change the email address in the **Email** field. Once you set a valid email ID in the user profile, then you would be able to reset your password, whenever required, by clicking the **Forgot Password** link on the login page.

Also, note if you want to move any file from and to a FortiSOAR system, then you must install SCP (`yum install openssh-clients -y`) or any SCP client. This is required since the `openssh-clients` package has been removed from FortiSOAR for security compliance.

From release 7.2.0 onwards, the Incident Response modules have been removed from the FortiSOAR platform and moved to the SOAR Framework Solution Pack (SP). The SOAR Framework Solution Pack (SP) is the **Foundational** Solution Pack that creates the framework, including modules, dashboard, roles, widgets, etc., required for effective day-to-day operations of any SOC. As the Incident Response modules, i.e., Alerts, Incidents, Indicators, and War Rooms are not part of the FortiSOAR platform, it becomes essential for users to install the SOAR Framework SP to optimally use and experience FortiSOAR's incident response. For detailed information about the SOAR Framework SP, see the SOAR Framework SP documentation.
From release 7.2.0 onwards, the SOAR Framework Solution Pack is installed by default with the fresh installations of FortiSOAR.

Common Tasks

Some of the common task that an administrator can perform are:

- License management
- System configuration
- Security management
- User management
- Appliance management
- Password Vault management
- Playbook configuration
- Application management

You can perform administration tasks using the **Settings** ( ) icon in the upper right-hand corner near the **User Profile** icon.

Apart from the above tasks, you can install the SLA Management Solution Pack to track and manage SLA for alerts and incidents in FortiSOAR. For more information see the SLA Management Solution Pack documentation.

Tasks and Permissions

To manage different modules, appropriate rights must be assigned to users. In FortiSOAR, modules are applied to roles, for example, the `Security` module is applied to the `Security Administrator` role. Role permissions are based on the Create, Read, Update, and Delete model (CRUD). Each module within FortiSOAR has explicit CRUD permissions that you can modify and save within a single Role.

For example, to perform all tasks for system configuration, you must be assigned a role that has `CRUD` permissions on the `Application` module, or to be able to add and manage users, you must be assigned a role that at the minimum has `Create` and `Update` permissions on the `People` module.

By default, FortiSOAR has at least one role in place after installation, the `Security Administrator`.

Task	Permissions required on the module
System configuration: Customizing FortiSOAR and configure several default options used throughout the system, including setting up authentication mechanisms and configuring dashboards and templates.	Create, Read, Update, and Delete (CRUD) permissions on `Application` module. Default Role - Application Administrator.
Security management: Managing teams and roles.	CRUD permissions on `Security` module. Default Role - Security Administrator. The security administrator role also has CRUD permissions on the `Secure Message Exchange` and `Tenants` modules, so that this role can configure multi-tenant systems.
User management: Adding and removing users and editing their permissions.	CRUD permissions on `People` module.
Appliances management: Managing appliances	CRUD permissions on `Appliances` module.

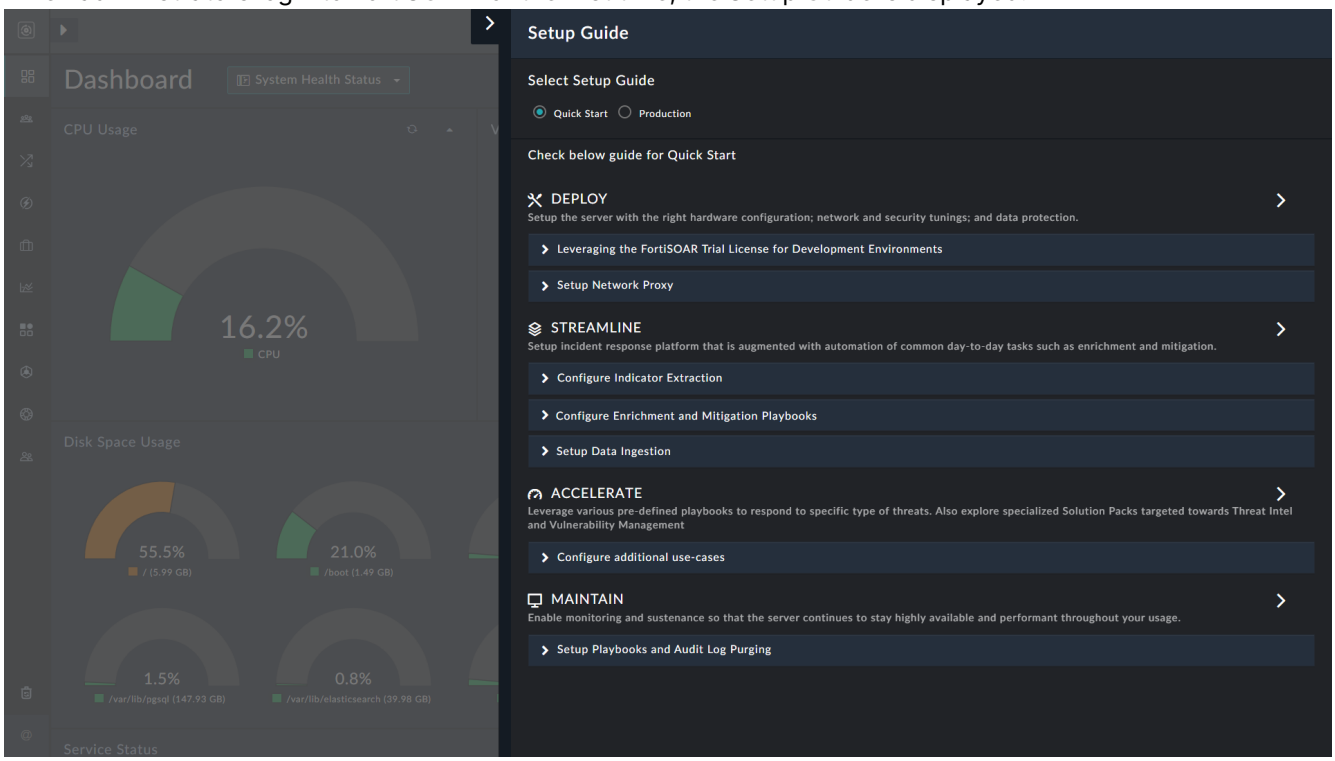
and access keys.	
Password Vault management: Integrating with third-party external vaults to manage sensitive data	CRUD permissions on `Connectors` module and Read permission on `Application` module.
Playbook management: Configuring playbook collections and playbooks	CRUD permissions on `Playbook` module. Default Role - Playbook Administrator.

Guide to setting up FortiSOAR

The Setup Guide designed to assist administrators, whether they are new or experienced, in configuring FortiSOAR according to best practices. It provides guidance on essential configurations and the installation of necessary solution packs for optimal performance, such as setting up network proxies, enabling audit and playbook log purging, configuring enrichment, and mitigation playbooks. For details, including permissions required to view the Setup Guide, see the Setup Guide Widget documentation.

The minimum permissions required to view and use the 'Setup Guide' are 'Read' and 'Update' permissions for both `Security` and `Application` and 'Read' permission for `Widget` and `Solution Pack`. Additionally, ensure that the **Enable Setup Guide** option is selected on the System Configuration page, which is the default setting.

When administrators log into FortiSOAR for the first time, the Setup Guide is displayed:



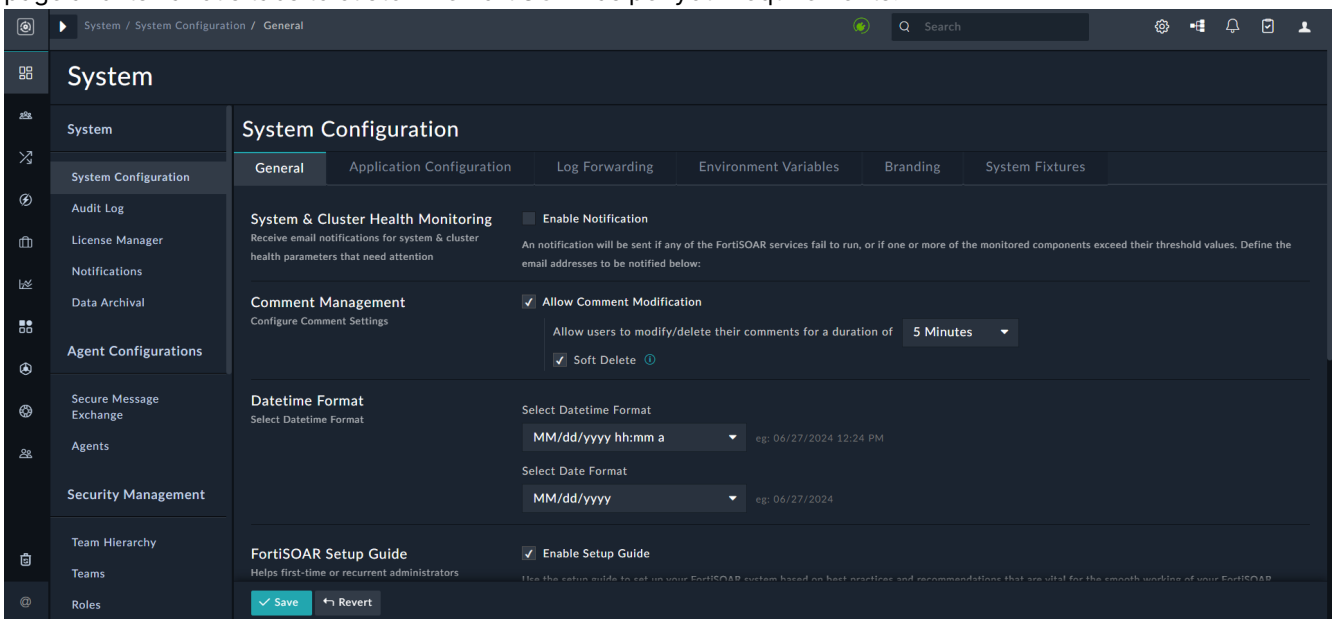
To minimize the Setup Guide, click the **>** arrow. To reopen the Setup Guide, click the **Setup Guide** icon in the top-right corner of FortiSOAR:

To hide the FortiSOAR Setup Guide icon, clear the **Enable Setup Guide** option on the `System Configuration` page. For more information, see the System Configuration chapter.

System Configuration

You can customize FortiSOAR and configure several default options used throughout the system, including the way FortiSOAR gets displayed to the users and the way notifications are sent to the users. To configure the system, you must be assigned CRUD permissions to the `Application` module. The Application module is assigned by default to the `Application Administrator` role. For information about roles, refer to the *Default Roles* section in the Security Management chapter.

Click the **Settings** ( ) icon to open the `System` (`System Configuration`) page. Use the `System Configuration` page and its various tabs to customize FortiSOAR as per your requirements:



Click **Settings > Audit Log**to open the `Audit Log` page. Use the `Audit Log` page to view a chronological record of all actions across FortiSOAR. For more information, see Audit Log.

Click **Settings > License Manager**to open the `License Manager` page. Use the `License Manager` page to update your license and view the details of your FortiSOAR license. For more information, see License Manager.

Click **Settings > Notifications**to open the `Notifications` page. Use the `Notifications` page to configure rules for delivery of notifications and also configure notification channels. For more information, see Notifications.

Click **Settings > Data Archival**to open the `Data Archival` page. Use the `Data Archival` page to configure settings for data archival and to search for records in the data archives. For more information, see Data Archival.

Click **Settings > Environment Variables**to open the `Environment Variables` page. Use the **Environment Variables** tab on the `System Configuration` page to add proxies to serve HTTP, HTTPS, or other protocol requests from FortiSOAR or define environment variables. For the procedure for configuring proxy settings and defining environment variables is included in the *Configuring Proxy Settings and environment variables* topic in the *Additional configuration settings for FortiSOAR* chapter of the "Deployment Guide."

Use the **General** tab on the `System Configuration` page to edit several default options found throughout the system, especially in the user profile. These include the following:

- Default notifications for system and cluster health monitoring
- Default Comment Modification
- Default DateTime Format
- Behavior of the FortiSOAR Setup Guide
- Setting a language other than English for your FortiSOAR UI
- Default theme
- Default country code
- Default navigation bar style
- Enable light mode setting for the 'Grid' widget across modules

For more information, see General and for more information on user profile configuration, refer to the *User Profiles* section in the Security Management chapter.

You can modify all the default values on a per-user basis on any user's Profile page.

Use the **Application Configuration** tab on the `System Configuration` page to configure various administrative options in FortiSOAR. These include the following:

- Purging of audit logs and executed playbook logs and cleaning the database
- Settings for recycle bin records including, purging of recycle bin records and restricting permanent deletion recycle bin records
- Default Playbook Execution Logging Levels
- Default Playbook Recovery Options
- Default settings for Playbook Log Movement
- Default rendering of expressions in the Playbook Designer (Simplified Expression View)
- Default timezone for exporting reports
- Manage user listings in 'People' Lookup fields
- Enable MIME type validation for file uploads

To enable sending system notifications, including requests for resetting passwords, and also for sending emails outside FortiSOAR you must configure the SMTP connector. For more information, see the SMTP Connector document.

For more information, see Application Configuration.

Use the **Log Forwarding** tab on the `System Configuration` page to forward FortiSOAR application and audit logs to your central log management server. For more information, see Log Forwarding.

Use the **Environment Variables** tab on the `System Configuration` page to configure proxy settings for FortiSOAR and to define any other environment variables. For more information, see Environment Variables.

Use the **Branding** tab on the `System Configuration` page to customize FortiSOAR branding based on your license type. For more information, see Branding.

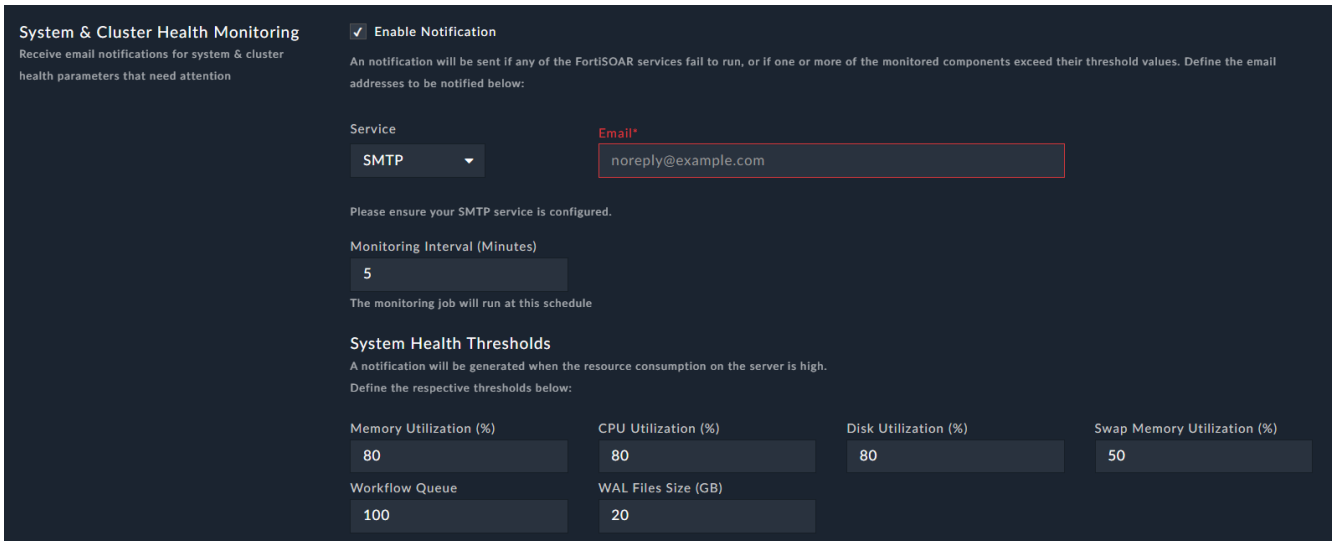
Use the **System Fixtures** tab on the `System Configuration` page to view the links to various playbook collections, email templates, and the self agent and self tenant pages, which are included by default with FortiSOAR. For more information, see System Fixtures.

General

On the `General` page, you can configure settings, especially those related to user profiles that apply across FortiSOAR. You can edit the settings and then click **Save** to apply the changes or click **Revert** to undo your changes.

Configuring System and Cluster Health Monitoring

You can set up system monitoring for FortiSOAR, both in the case of a single node system and in High Availability (HA) clusters. To receive email notifications of any FortiSOAR service failure, or of any monitored threshold exceeding the set threshold, etc., click the **Enable Notification** checkbox in the `System & Cluster Health Monitoring` section.



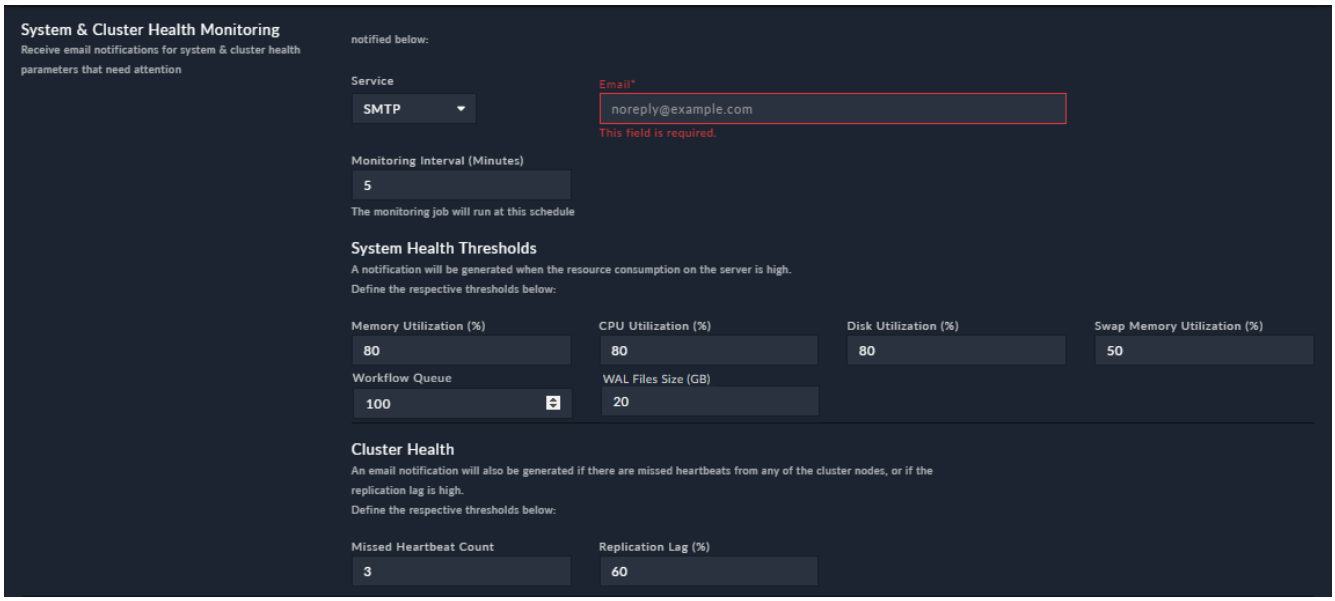
Once you click the **Enable Notification** checkbox, from the **Service** drop-down list, select the service to be used for notifications. You can choose between **SMTP** or **Exchange**. In the **Email** field, specify a comma-separated list of email addresses of users who should receive email notifications in case of any FortiSOAR service failures, or of any monitored threshold exceeding the set threshold, etc.

The email that is sent for high CPU consumption also contains information about the processes that are consuming the most memory.

In the **Monitoring Interval (Minutes)** field, specify the interval in minutes at which you want to monitor the system and perform the health check of the HA cluster. By default, the system is monitored every **5** minutes.

In the `System Health Thresholds` section, you can set the thresholds, in percentages, for Memory Utilization (80% default), CPU Utilization (80% default), Disk Utilization (80% default), Swap Memory Utilization (50% default), Workflow Queue, i.e., the value of the celery queue size, and WAL Files Size (GB), which by default is set as 20 GB. The default value of the workflow queue is set at 100. If the thresholds set are reached or crossed for any of the monitored parameters, an email notification is sent to all the specified email addresses.

If you have an HA environment, then additionally, RabbitMQ certificates, Nginx certificates, and self-signed PostgreSQL certificates are also monitored and notifications can be sent to specified users when any of the certificates is nearing expiry.
Heartbeat failures and replication lags between nodes of your HA cluster can also be monitored and notifications can be sent to specified users in the event of heartbeat failures and high replication lags between nodes of your HA cluster. You can specify values for these parameters in the `Cluster Health` section:



In the **Missed Heartbeat Count** field, specify the number of missed heartbeats after which notifications of failure will be sent to all the specified email addresses.

You cannot specify a value lesser than 3 in the Missed Heartbeat Count field.

In the **Replication Lag** field, specify the lag value for the replication lag between nodes. By default, this is set to **3**, i.e., 3GB. If the replication lag threshold is reached or crossed, then an email notification is sent to all the specified email addresses.

Some examples of how Monitoring Interval (Minutes) and Missed Heartbeat Count values help you in monitoring heartbeats between nodes in an HA cluster:

Case 1

If you have set the Monitoring Interval to 5 minutes and the Missed Heartbeat Count to 3, this means that when the heartbeat is missed (the `cyops-ha service` is down) for the last >=15 minutes (monitoring interval * missed heartbeat count), the heartbeat missed notification will be sent to all the email addresses that you have specified in the **Email** field.

The cluster health check is performed based on the monitoring interval specified. For example, if you specify 3 minutes in the **Monitoring Interval (Minutes)** field, then the HA cluster health check will be run every 3 minutes.

Notifications are sent based on the multiplication of the values that you have set in the monitoring interval and the missed heartbeat count. For example, if you have set the monitoring interval to 3 and missed heartbeat

count to 4, and if the heartbeat is missed for the last >=12 minutes, then heartbeat missed notifications will be sent to all the email addresses that you have specified in the **Email** field.

Case 2

If you have had no heartbeats missed for the last >=15 minutes, considering the monitoring interval that is set to 5 minutes and the missed heartbeat count set to 3, however, there is a service down or a service connectivity failure found in the health check, then a notification for service down or service connectivity failure will be sent to all the email addresses that you have specified in the **Email** field.

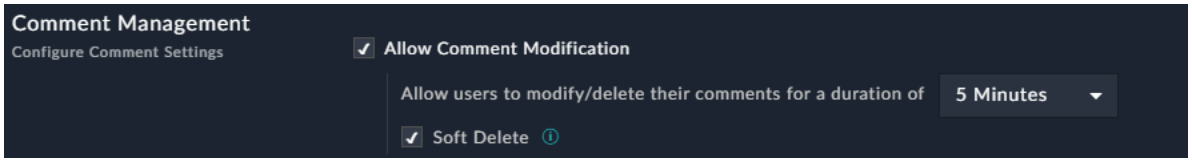
For more information on HA, see the High Availability and Disaster Recovery support in FortiSOAR chapter.

Configuring Comments

A user who has `Security Update` permissions can edit comments of any FortiSOAR user, and a user who has `Security Delete` permissions can delete comments of any FortiSOAR user. There is no time limit for the Security user to update or delete comments.

Users can edit and delete their own comments in the "Collaboration" window or in the Comments widget, if you (the administrator) has enabled the settings for comment modification and if the user has appropriate CRUD permissions on the `Comments` module.

To allow users to edit and delete their own comments, click the **Settings** icon, which opens the `System Configuration` page. On the **Application System Configuration** tab, in the `Comment Modification` section, select the **Allow Comment Modification** checkbox.

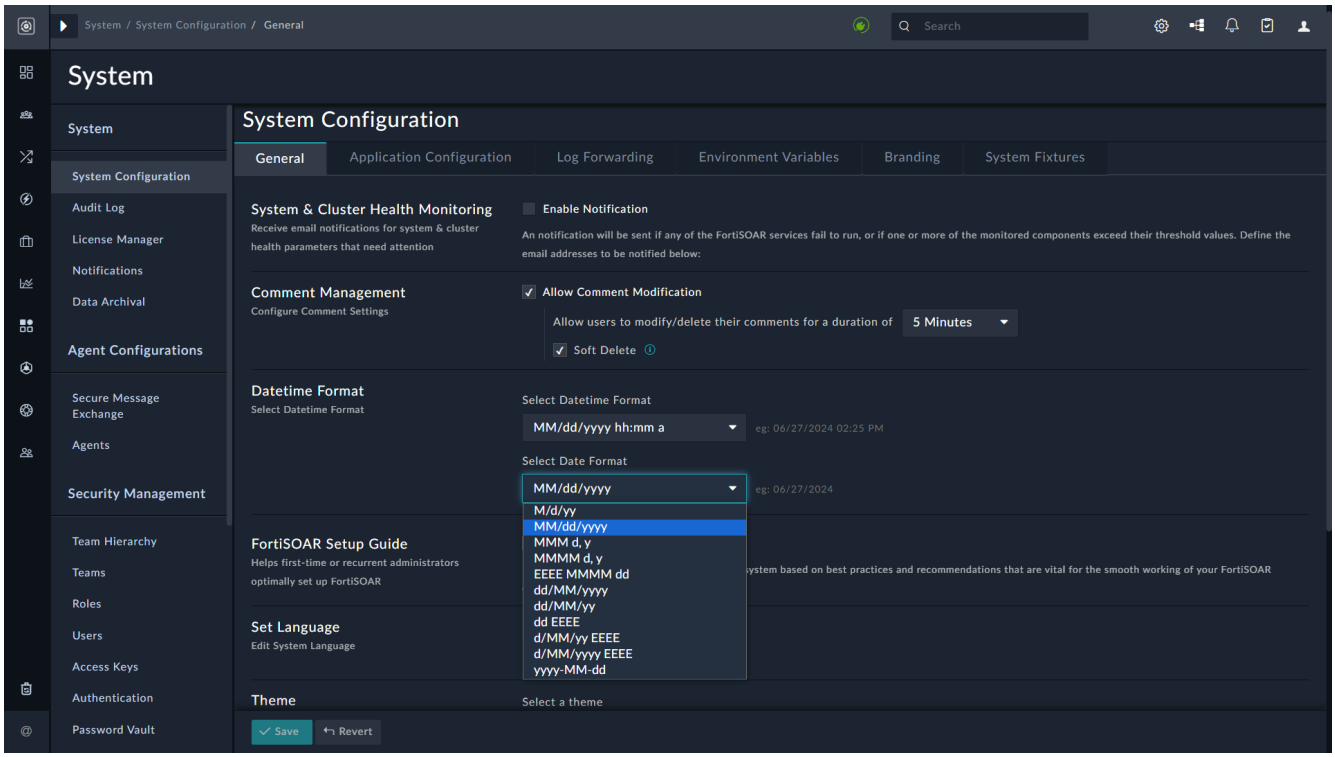


You can also specify the time until when the user can edit or delete their comments in the **Allow users to modify /delete their comments for a duration of** field. For example, if you select 1 minute from this field, then users can edit and delete their comments until 1 minute after which they have added the comment. By default, the **Allow users to modify/delete their comments for a duration of** field is set to 5 minutes. Users cannot edit or delete their comments after the time specified in the **Allow users to modify/delete their comments for a duration of** field.

You can also specify the behavior of the comment "delete" action, i.e., when a user deletes a comment, you can choose to permanently delete the comment or flag the comment for deletion, i.e., **Soft Delete**. If you choose to keep the **Soft Delete** checkbox checked (default), then the comments will be soft deleted, i.e., on the UI you will see `--Comment Deleted--` instead of the comment. In case you have cleared the **Soft Delete** checkbox, you will not see anything on the UI since the comment has been permanently deleted.

Setting the formats for Date and DateTime fields on the FortiSOAR UI

Using standard syntax (https://angular.io/api/common/DatePipe), you can customize the formats for Date and DateTime fields on the FortiSOAR UI.
To configure the formats for the DateTime and Date fields, in the `Datetime Format` section, from the **Select Datetime Format** drop-down list, select the required DateTime format. Similarly, from the **Select Date Format** drop-down list, select the required Date format and click **Save**.The FortiSOAR UI now displays all DateTime and Date fields in the set format.

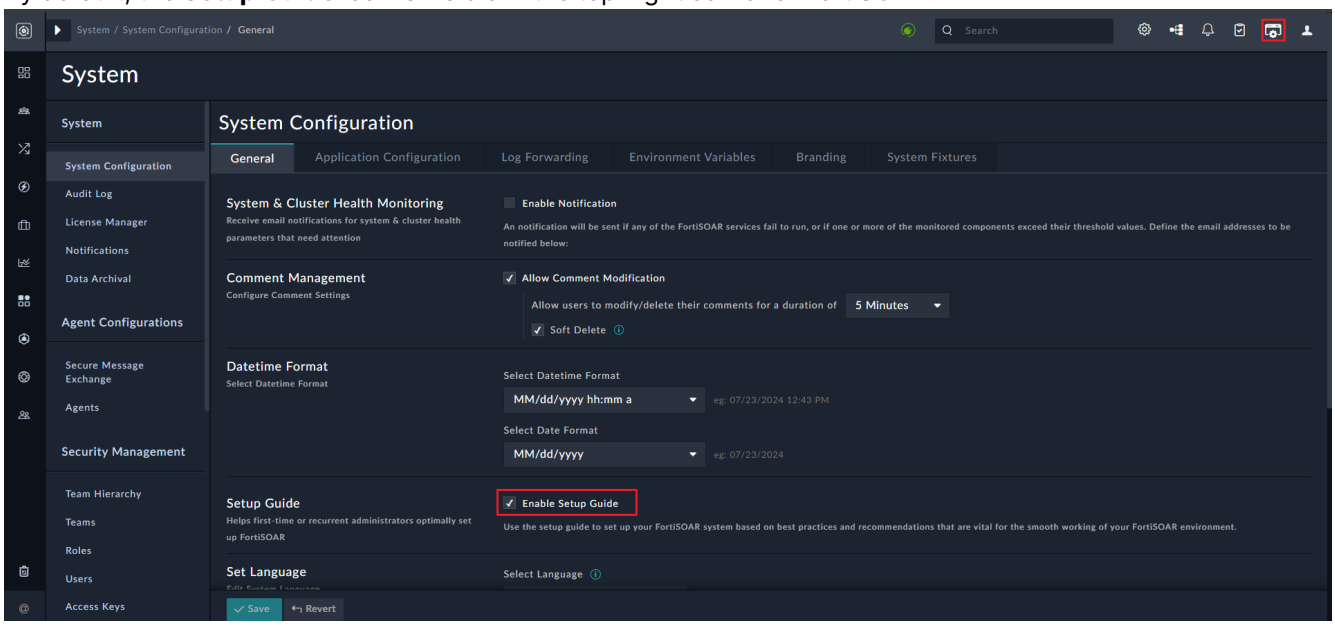


For example, if you set the DateTime format to '`HH:mm EEEE MMMM dd`', all the DateTime fields on the FortiSOAR UI, such as Resolved Date, Ack Due Date, etc., will be present in this format, i.e., as '15:24 Wednesday August 30'.

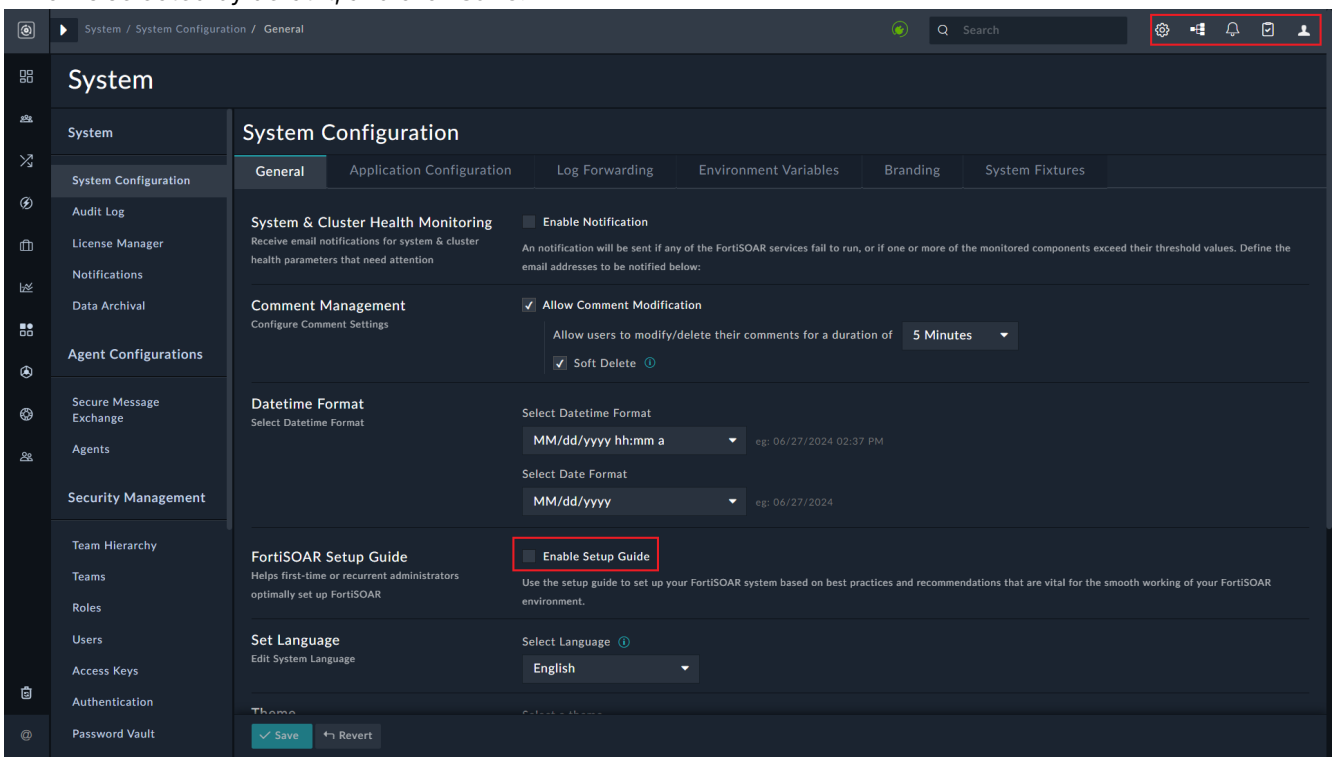
Customizing the behavior of the FortiSOAR Setup Guide

The FortiSOAR Setup Guide helps first-time or recurrent administrators of FortiSOAR to optimally set up FortiSOAR based on best practices. For more information on the FortiSOAR Setup Guide, see the Overview chapter and the Setup Guide documentation.

By default, the **Setup Guide** icon is visible in the top-right corner of FortiSOAR:



To hide the **Setup Guide** icon, go to the `System Configuration` page, clear the **Enable Setup Guide** option, which is selected by default, and click **Save**:



Once the updated setting is saved, you will observe that the Setup Guide icon are not visible in the top-right corner of FortiSOAR.

Setting a language other than English for your FortiSOAR system

Support for 'Internationalization' has been added to the FortiSOAR platform in release 7.5.0, enabling FortiSOAR to meet the linguistic, cultural, and other needs of a particular locale. The default language set for the FortiSOAR UI is English, with additional support now available for the following languages:

- Japanese (Preview)
- Korean (Preview)
- Simplified Chinese (Preview)

'Preview' has been added to languages other than English as the translations have been done using translation tools, which might result in inaccuracies or incompleteness in the translations. Your feedback is valuable for improving their accuracy and quality. Additionally, changing the language from English to another language might affect the user experience on the FortiSOAR UI to some extent, such as labels exceeding their width or misalignment of buttons.

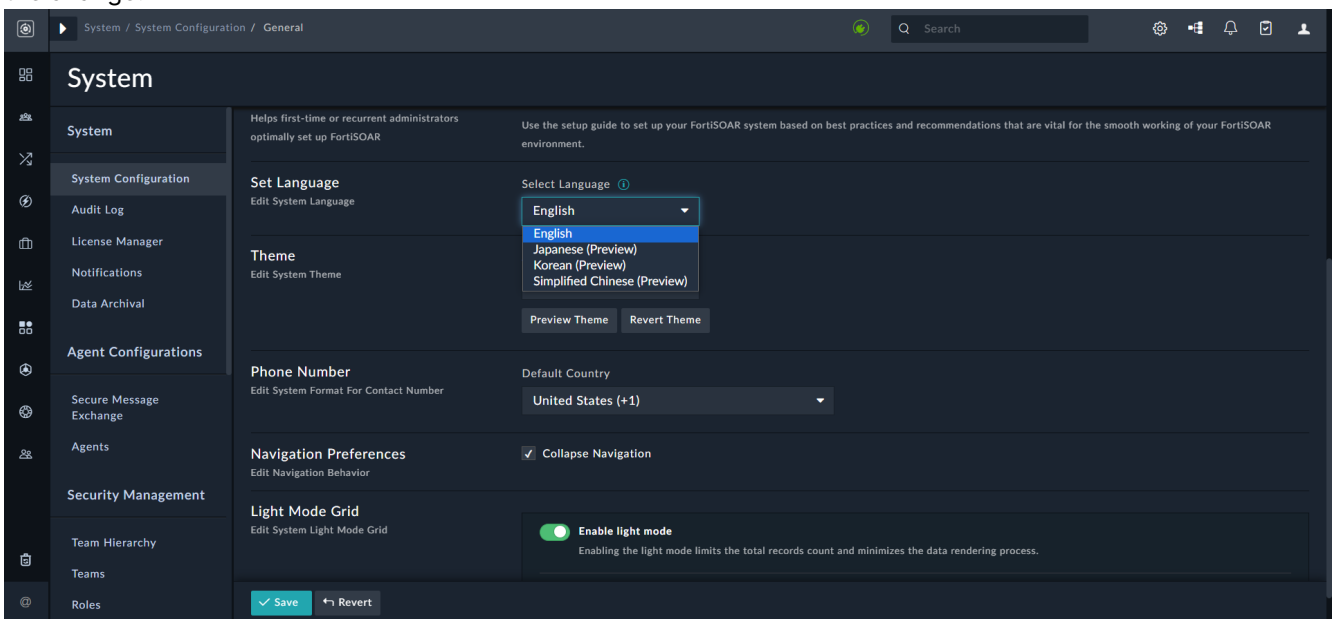
To support internationalization, FortiSOAR uses a system widget named "Language Pack" that includes the supported languages. This widget is automatically installed during the installation or upgrade of FortiSOAR to release 7.5.0 or later.

The "Language Pack" widget is a system widget that cannot be uninstalled, and you should not modify it. Making changes to it can result in translation issues, causing the FortiSOAR UI to appear in English. English serves as the fallback language for FortiSOAR. Any content that is not translated will be shown in English.

Administrators must be assigned the 'Read' or 'Usage' permission on 'Widgets' (in addition to other necessary permissions) to modify the global language settings. Without this permission, the FortiSOAR UI remains in English, regardless of the language you set.

To change the language in FortiSOAR for all users, switch from English to any of the supported languages by clicking the **Settings** icon, which opens the `System Configuration` page. In the `Set Language` section, select the language you want from the **Select Language** drop-down list for the FortiSOAR UI then click **Save** to apply

the change:



If your FortiSOAR instance is part of a high-availability cluster, changing the language on the `System Configuration` page on the primary node will apply the selected language translation on the secondary nodes when users log out and log back in on secondary nodes. Similarly, if the language is changed on a secondary node, users must log out and log back in on the primary node and other secondary nodes for the language changes to be reflected.

Users can set the language for their FortiSOAR instance in their profile. The language set in the user profile takes precedence over the language set by an administrator. For example, if the administrator sets the FortiSOAR language to 'Korean' using this process, but a user sets their profile language to 'Japanese', the user's FortiSOAR UI will be displayed in Japanese. For information on setting the language in a user profile, refer to the *Overview* chapter in the "User Guide."

When changing the language in FortiSOAR from English to any of the supported language, all static text in the FortiSOAR UI gets displayed in the selected language. Static text includes labels of fields, tabs, and buttons, titles of dialog boxes, descriptions, static message content, such as content displayed in confirmation, error, or warning messages, etc.

Content that can be changed by the user, gets displayed in English and not in the selected language, some examples of such type of content include:

- Names or titles of playbooks, connectors, and built-in widgets such as charts, grids, tabs, etc.
- Tab names, headings, subheadings, etc., in the record detail view and list view as these are user-editable fields.
- Values of picklist items and navigation menu items.
- For modules (mmd) on the `Module Editor` page, the name of a module and the title of a field must be in English, as the `Type` (for modules) and `Field API Key` (for fields) are auto-generated and must be in English as they are used to identify the module or field in the API.

Non-default grid columns are not translated until the user resets the columns to their default settings.
When changes are made to the order of grid columns or when a column is added or deleted from a module, these changes are stored as user preferences. To have the field names translated to the user or global preferred language, the columns need to be reset to their default settings.

Apart from the above the following content also gets displayed in English and not in the selected language:

- Syslogs forwarded from a FortiSOAR system.
- Playbook step results.
- Grid view exported in the CSV format.
- Tooltips, error messages, and toaster messages that are generated from the backend such as tooltips on the `Module Editor` page or the health check disconnected error message for the Recommendation Engine.

For information on various services, functions, etc., added for multilingual support for widgets created starting from release 7.5.0, see the "Widget Development Guide."

Editing locale files for existing modules

You can modify the files for the supported locales (Japanese, Korean, Simplified Chinese) for existing modules using the Export and Import Wizards. For details on these wizards, see the *Export and Import Wizards* topic in the Application Editor chapter. If you want to update some translations of modules, such as the 'Alerts' Module, follow these steps:

1. Export the 'Alerts' module and download the .zip file of the export.
2. Extract the downloaded .zip file to your local disk. A sample folder structure of the exported 'Alerts' module is as follows:
```
exportTemplate folder
--+ modules
---+ alerts
-----+ languages
-------+ en.json
-------+ ja.json
-------+ ko.json
-------+ zh_cn.json
-----+ detail-layout.json
-----+ form-layout.json
-----+ list-layout.json
-----+ mmd.json
--+ picklists
--+ records
--+ info.json
```
3. Open the language json file you want to edit for the module. For example, to edit the Japanese language file, open the `ja.json` file.
4. Update the necessary translations and save the file.
5. Create a .zip file of the Alerts module with the updated translations.
6. Import the zip file to your FortiSOAR instance using the Import Wizard.
7. After a successful import, you will see the updated translations for the 'Alerts' module.

Configuring Themes

You can configure the FortiSOAR theme that will apply to all the users in the system.

Non-admin users can change the theme by editing their user profile. Changes made by a non-admin user to the theme are applicable only to those users who have not changed their default user profile settings.

There are currently three theme options, **Dark**, **Light**, and **Space**, with **Space** being the default. On the `System Configuration` page, select the theme that you want to apply across FortiSOAR. Click **Preview Theme** to view how the theme would look and click **Save** to apply the theme.

To revert the theme to the default, click **Revert Theme**.

Configuring Default Country Code

You can configure country code format for contact numbers that will apply to all users in the system. In the `Phone Number` section, select the **Default Country** and thereby the default country code that you want to apply across FortiSOAR and click **Save** to apply the code.

Configuring Navigation Preferences

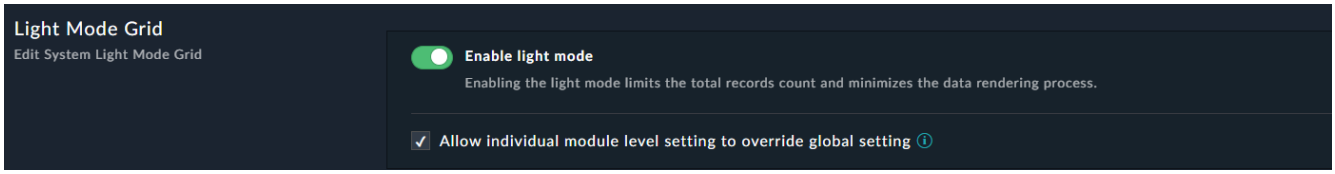
You can configure the behavior of the left navigation bar across FortiSOAR. You can choose whether you want the left navigation bar to collapse to just display icons of the modules or expand to display both icons and titles of modules. In the `Navigation Preferences` section, click **Collapse Navigation** to collapse the left navigation bar and click **Save** to apply the behavior of the left navigation bar across the system.

Enabling Light Mode Setting

You can enable the 'Light Mode' for the 'Grid' widget across modules by toggling the **Enable light grid** to **Enabled** (default is **Disabled**). This lighter version of the grid widget enhances performance and usability.

The 'Light Mode' is not applicable to 'Notifications' grids.

For details on the 'Light Mode' for Grids, see the '`Grid`' topic in the Dashboards, Templates, and Widgets chapter of the User Guide.



Additionally, to honor the module-level light mode settings, select the **Allow individual module level setting to override global setting** option. For example, if the light grid is enabled globally, but you want to exclude the

'Indicators' module, then you can clear the 'Enable Light Mode' checkbox for the grid in that module. This action will result in the Indicators module using a regular grid. By default, the **Allow individual module level setting to override global setting** option is unchecked, enforcing global light-mode settings across all modules and overriding any module-specific settings.

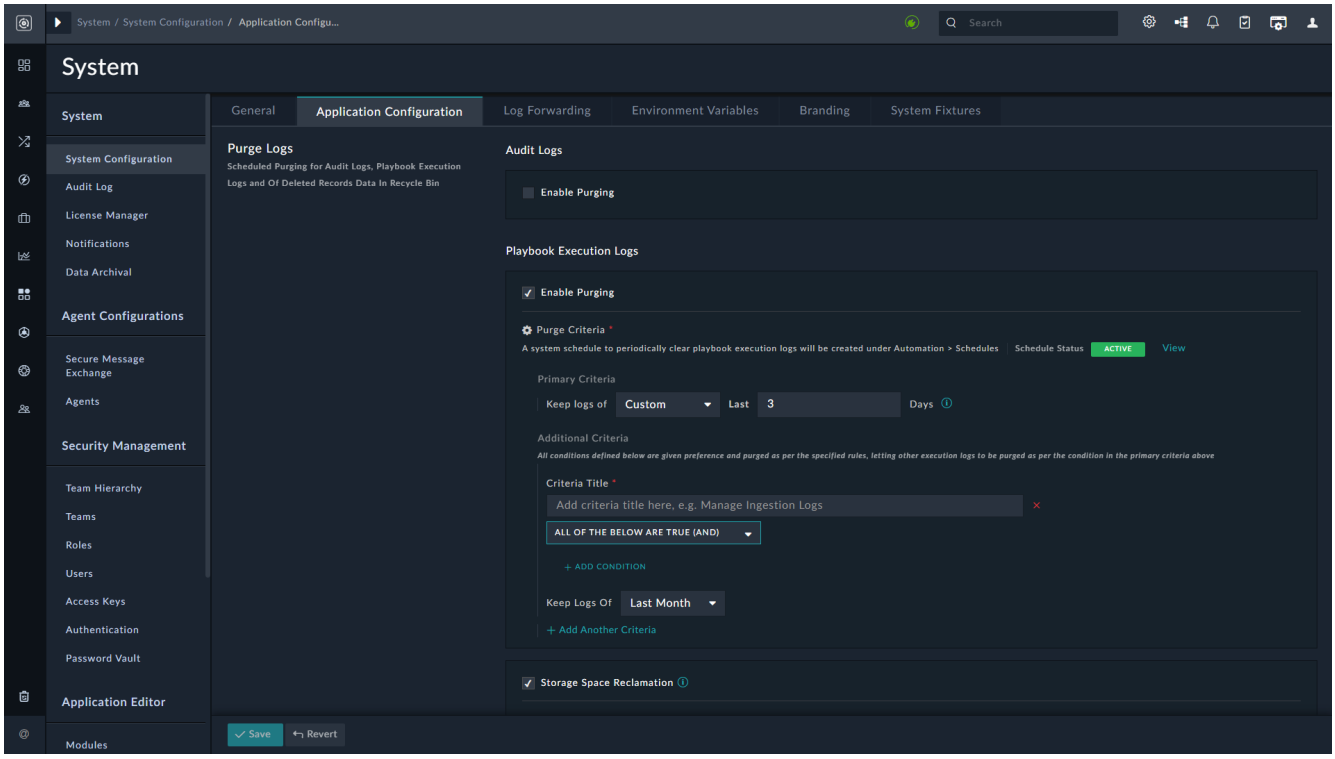
Application Configuration

Click the **Application Configuration** tab on the `System Configuration` page to open the `Application Configuration` page. On the `Application Configuration` page, you can configure various administrative settings that will apply across FortiSOAR. You can edit the settings and then click **Save** to apply the changes or click **Revert** to undo your changes.

Purging audit logs, executed playbook logs, and recycle bin records, and reclaiming unused disk space

You can schedule purging, on a global level, for both audit logs and executed playbook logs. Click the **Settings** icon, which opens the `System Configuration` page, and click the **Application Configuration** tab. In the `Purge Logs` section, you can define the schedule for purging both Audit Logs and Executed Playbook Logs.

By default, audit logs are not purged, but logs for executed playbooks are purged. The purge function excludes 'Recent' playbook logs and playbooks executed on the same day when purging 'Historical' logs.



The Purging activity deletes logs permanently, and you cannot revert this operation.

Purging logs deletes old records from the respective tables; however, it does not free up the PostgreSQL database space, which could cause space and performance issues in FortiSOAR. Therefore, from version 7.0.2 onward, FortiSOAR provides you with an option to reclaim unused disk space. This activity clears all the empty rows in tables and indexes, which helps in improving the performance by optimizing the space. By default, this cleanup activity is run "Weekly at 02:01 AM (UTC) on Sunday"; however, you have the ability to update the schedule of this cleanup activity as per your requirement.

In release 7.6.0, FortiSOAR has optimized the process of reclaiming unused disk space by utilizing advanced PostgreSQL utilities. These enhancements address issues such as database bloat and performance slowdowns in operations such as playbook searches, caused by extensive data retention and the accumulation of a large number of records over time. This is often a result of PostgreSQL space quickly filling up when numerous playbooks are executed, even in 'Info' mode. The process reclaims disk space from PostgreSQL by optimally managing data.

The configuration file `/opt/cyops-workflow/sealab/sealab/config.ini` contains parameters for using advanced PostgreSQL utilities, i.e., `pg_squeeze` and `pg_repack`. To modify default values, edit the `config.ini` file and then restart the services using the `systemctl restart uwsgi celeryd fsr-workflow celerybeatd` command:

- `SQUEEZE_SOFT_TASK_TIMEOUT`: This parameter defines the soft time limit for the `pg_squeeze` task. The soft time limit allows the task to catch exceptions and clean up before being terminated; the hard timeout is not catch-able and force terminates the task. If the soft time limit is exceeded and cleanup is stuck then the hard time limit (expected to be higher than the soft time limit) will force terminate the `pg_squeeze` task as a safety net and to avoid forever running of task. The default soft time limit is 3600 seconds.
- `SQUEEZE_TASK_TIMEOUT`: This parameter sets the hard time limit for the `pg_squeeze` task. When the task exceeds this duration, the system will throw an exception. The hard time limit should always be higher than the `SQUEEZE_SOFT_TASK_TIMEOUT` and is used to override the `CELERYD_TASK_TIME_LIMIT` for the '`pg_squeeze`' task. The default hard time limit is 3800 seconds.
- `SQUEEZE_EXECUTION_TIME`: The system performs a periodic cleaning task in addition to the scheduled one. You can adjust the UTC time for this task by changing the `SQUEEZE_EXECUTION_TIME` parameter in the `/opt/cyops-workflow/sealab/sealab/config.ini` file. It is recommended to configure '`pg_squeeze`' to run when the ingestion rate is low. The default setting for the `SQUEEZE_EXECUTION_TIME` parameter is '`22-6`', indicating that the cleaning task will occur from 10 PM to 6 AM.
- `USE_PG_SQUEEZE` and `USE_PG_REPACK`: In fresh installations from release 7.6.0 onwards, the default settings for '`USE_PG_SQUEEZE`' and '`USE_PG_REPACK`' parameters are set to '**true**'. However, in upgrades to release 7.6.0, the '`USE_PG_SQUEEZE`' and '`USE_PG_REPACK`' keys are set to '`false`', which means that the upgraded systems will still use full vacuum for cleaning up workflow execution history, which can block of all workflow operations.
  NOTE: Before changing the workflow execution cleanup behavior by setting '`USE_PG_SQUEEZE`' and '`USE_PG_REPACK`' keys to '`true`', in upgraded instances, it is recommended to run a full vacuum first.

In High Availability environments, settings related to `pg_squeeze` and `pg_repack`, such as `USE_PG_SQUEEZE`, `USE_PG_REPACK`, `SQUEEZE_EXECUTION_TIME`, etc. must be modified on all cluster nodes.

In release 7.6.1, FortiSOAR introduces 'playbook log movement' to optimize workflow logs storage. This feature automatically moves playbook logs to historical storage once playbooks complete execution. Historical storage supports active storage by retaining only active and awaiting playbooks in the active storage, while completed playbook logs are transferred to historical storage. This process reduces the size of the active storage, enhancing overall performance. Additionally, the reclaim disk space operation runs more efficiently, saving time and system resources due to the reduced table size. Currently, completed playbook logs are moved to historical storage every 15 minutes, while logs for failed or terminated playbooks are transferred every 60 minutes. You can adjust these timings in the setting for Playbook Log Movement.

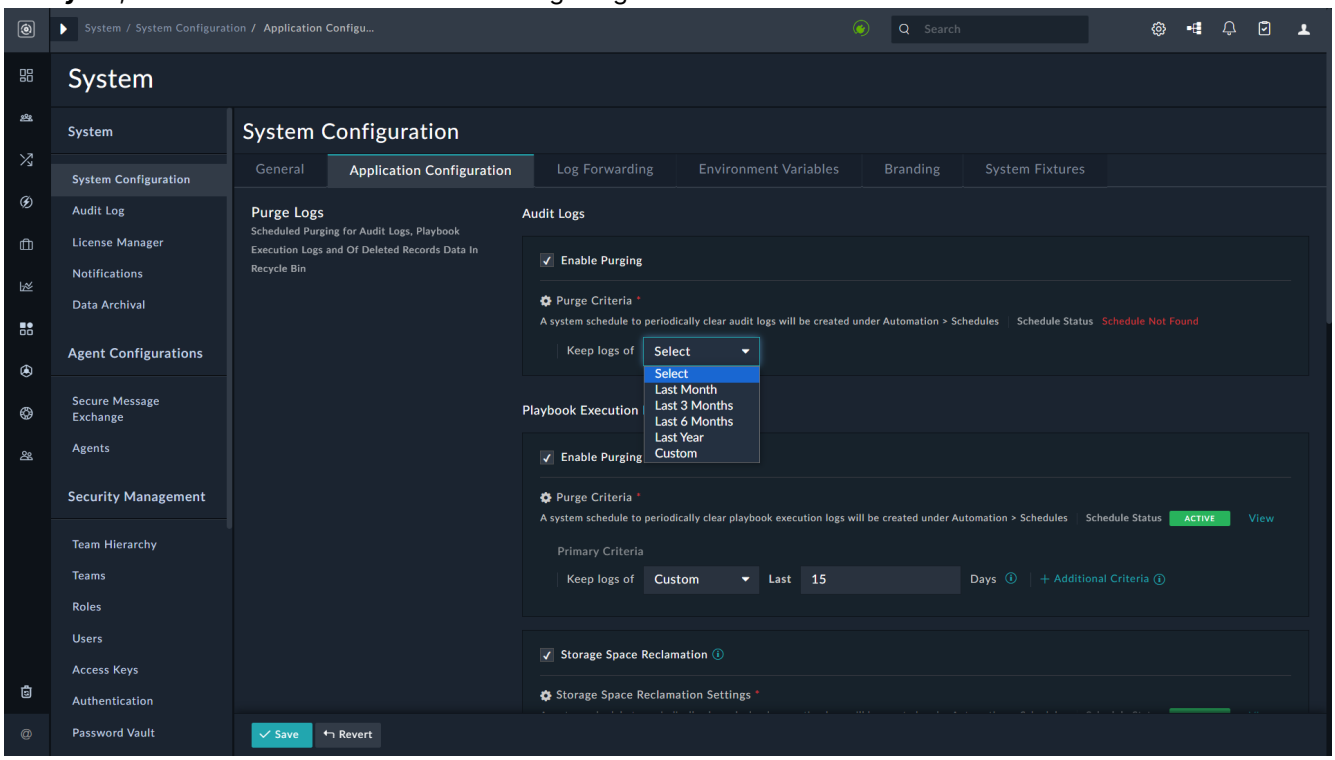
For more information on optimizing settings see the Debugging, Troubleshooting, and Optimizing FortiSOAR chapter and for debugging and optimizing playbooks and workflows, see the *Debugging and Optimizing Playbooks* chapter in the "Playbooks Guide."

Scheduling purging of audit logs

To purge **Audit Logs**, you must be assigned a role that has a minimum of `Read` permission on the `Security` module, `Read` permission on the `Application` module, and `Delete` permissions on the `Audit Log` module.

To enable purging of Audit logs, select the **Enable Purging** checkbox that appears in the `Audit Logs` section.

Once you select the **Enable Purging** checkbox, you require to define the schedule for purging of audit logs. To specify the time for which you want to retain the logs, you must select the appropriate option from the **Keep logs Of**drop-down list. You can choose from the following options: **Last month**, **Last 3 months**, **Last 6 months**, **Last year**, or **Custom** as shown in the following image:



If you choose **Custom**, then you must specify the *number of days* for which you want to retain the logs.

For purging purposes, 1 month is considered as 30 days and 1 year is considered as 365 days.

The purging schedule clears all logs that belong to a timeframe earlier than what you have specified.

For example, if you want to retain audit logs for a month, then select **Last month** from the **Keep logs of** drop-down list. Once you save this setting all audit logs that are older than 1 month (30 days) will be cleared, and this will be an ongoing process, as the audit log records will all be time-stamped and the ones older than 30 days will be purged.

By default, the purge schedule job, runs every midnight (UTC time) and clears all logs that have exceeded the time duration that you have specified. If you want to run the purging activity at a different time of the day or for a different duration, you can do so by editing the schedule of purging on the `Schedules` page (**Automation** > **Schedules**) once you enable purging of the logs.

Scheduling purging of executed playbook logs

To purge **Executed Playbook Logs**, you must be assigned a role that has a minimum of `Read` and `Update` permissions on the `Security` module, `Read` and `Update` permissions on the `Application` module, and `Delete` permissions on the `Playbooks` module.

Purging of executed playbook logs based on date or days criteria

Executed Playbook Logs are purged by default, and therefore the **Enable Purging** checkbox is already selected in the `Executed Playbook Logs` section. By default, any executed playbook logs that are older than 15 days are purged. You can change time for which you want to retain the playbook execution logs by selecting the appropriate option from the **Keep logs Of** drop-down list, as is the case with audit logs.

A system schedule, named "Purge Executed Playbook Logs" is also already created and active on the `Schedules` page. This schedule runs every midnight (UTC time) and clears all logs that have exceeded the time duration that is specified. If you want to run the purging activity at a different time of the day or for a different duration, you can do so by editing this schedule.

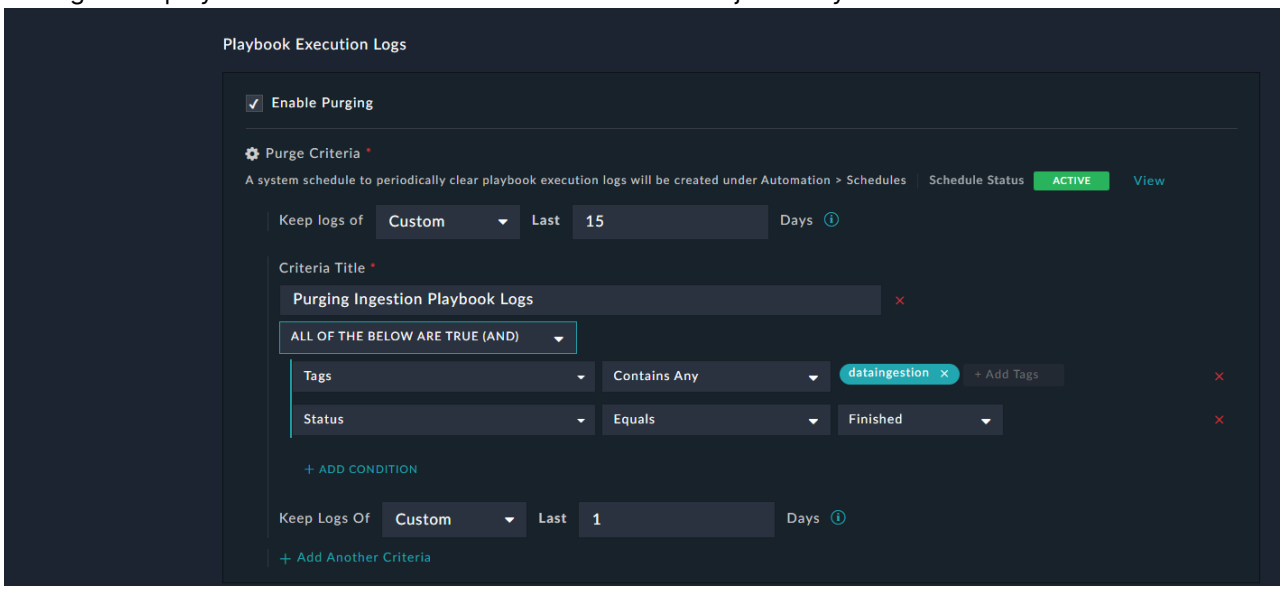
To optimize playbook log purging, you must set the purge schedule interval to greater than one day.

Purging of executed playbook logs based on criteria other than date or days

You can purge executed playbook logs based on some complex query condition that involves multiple parameters and not just the date or days criteria. For example, clearing logs of ingestion playbooks that have completed their execution. Being able to clear logs based on these criteria is useful since ingestion playbooks are generally scheduled and they can occupy a major chunk of playbook history in the database. Therefore, this feature provides you with an option to build desired queries for purging executed playbook logs and scheduling the purging.

To add the custom criteria, based on the clearing ingestion playbook that have completed their execution example, do the following:

1. Click the **+Additional Criteria** link.
2. In the **Criteria Title** field, enter the title of the criteria based on which you want to purge the executed playbook logs. For example, `Purging Ingestion Playbook Logs`.
3. Select the logical operator, **All of the below are True (AND)**, or **Any of the below is True (OR)**. For our example, we require the AND operator, since we want to purge all playbooks that contain the "ingestion" tag and whose status is finished, so select **All of the below are True (AND)**.
4. Click the **Add Condition** link to add conditions for purging the executed playbook logs:
   From the **Select a field** drop-down, select **Tags**, from the **Operator** drop-down list select **Contains Any** and in the **Add Tags** field, enter `dataingestion`. Click the **Add Condition** link again, and from the **Select a field** drop-down, select **Status**, from the **Operator** drop-down list select **Equals**, in the Status drop-down list select **Finished**.
   You can add additional conditions or criteria as per your requirements.
5. Schedule the purging of the executed playbooks logs based on the above-specified criteria by selecting the appropriate option from the **Keep Logs Of** drop-down list. You can choose from the following options: Last month, Last 3 months, Last 6 months, Last year, or Custom.
   For our example, we choose the **Custom** option and specify `1` for days, which means that keep the logs for the ingestion playbooks that have finished their execution for just 1 day in the database.



6. To save the criteria for purging executed playbook logs, click **Save**.

Points to be considered while setting multiple purging criteria

If you have added multiple purging criteria, then the purge functionality purges logs sequentially.
For example, if you have defined the following criteria

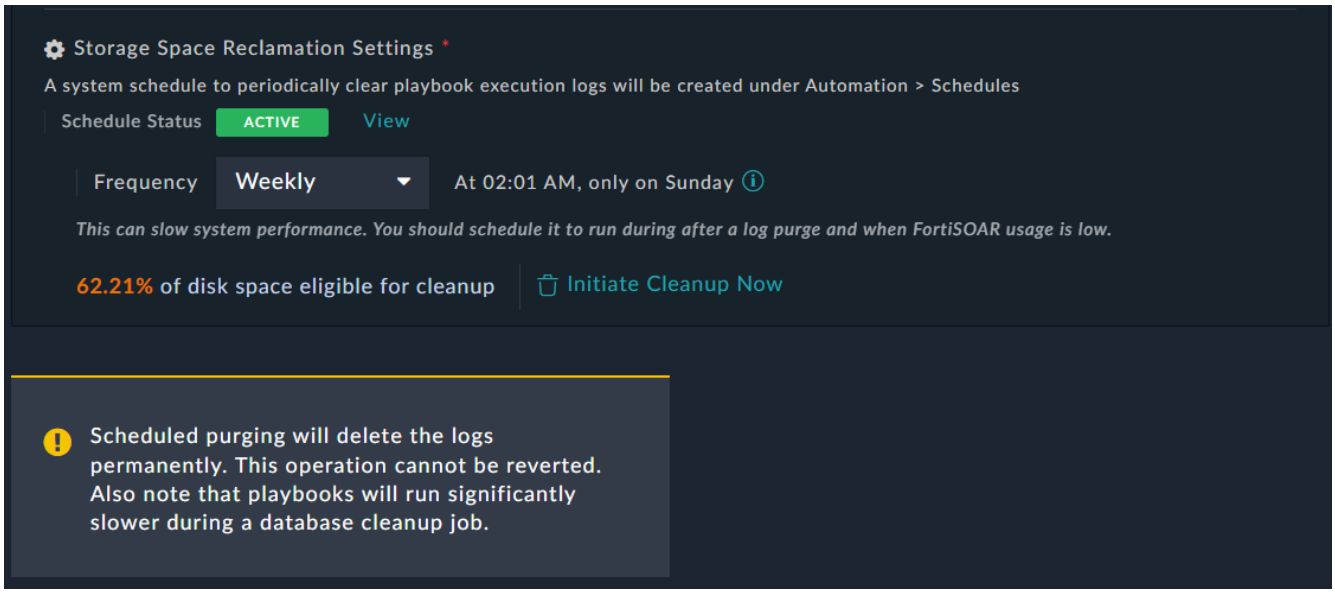
- Default: Keep logs of the last 2 days.
- If 'Playbook Execution Status = Failed', then keep logs for last 1 day.
- If Tags contain Ingest, then keep logs for last 1 day.

In such a scenario, logs are purged as follows:

1. Retains logs for the last 2 days only, and purges the remaining logs.
2. From the logs of the last 2 days, looks for logs that have 'Playbook Execution Status = Failed', and keeps such logs for the last 1 day only.
3. Looks for logs that have 'Tags' containing 'Ingest', and keeps such logs for the last 1 day only.

Scheduling storage space reclamation

To reclaim unused space, ensure that the **Storage Space Reclamation** option is selected (default). This activity clears all the empty rows in tables and indexes, which helps in improving the performance by optimizing the space.

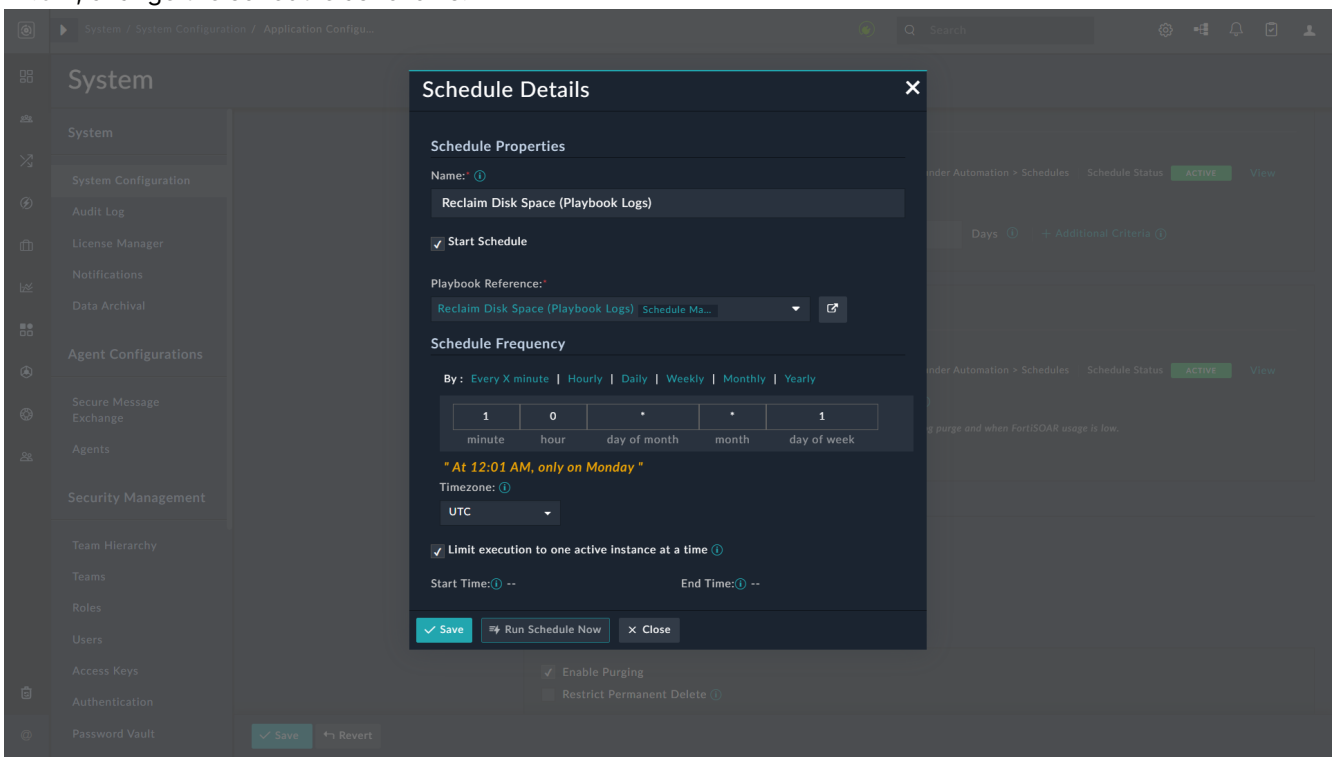


You can schedule regular storage space reclamation to ensure that space gets regularly freed up in the PostgreSQL database as per your requirements.

It is recommended to schedule the cleanup process after purging logs and during non-production hours when system usage is lower. This is because as the process pauses the execution and operations of all playbooks and slows down your system's performance.

In the `Storage Space Reclamation Settings` section, you can select the frequency of running the storage space reclamation activity. You can choose between running this activity **Weekly** or **Daily** or to some custom frequency based on your requirements. By default, a system schedule named "Reclaim Disk Space (Playbook Logs)" is created in **Automation** > **Schedules** to periodically clear the playbook execution logs "Weekly at 02:01 AM on Sunday".
To change this schedule to a custom frequency, click the **View** link to display its Schedule Details and edit the schedule as per your requirement, and then click **Save**. For example, to run this activity "Weekly on Mondays at

12:01", change the schedule as follows:



You can view the % of disk space that is being used for playbook logs that is eligible for cleanup and which can be reclaimed, and you can also immediately initiate a cleanup of playbook logs by clicking the **Initiate Cleanup Now** link, which displays a confirmation dialog. Clicking **Confirm** on the dialog immediately starts the disk space reclamation activity.

System Settings for Recycle Bin

Scheduling purging of recycle bin records

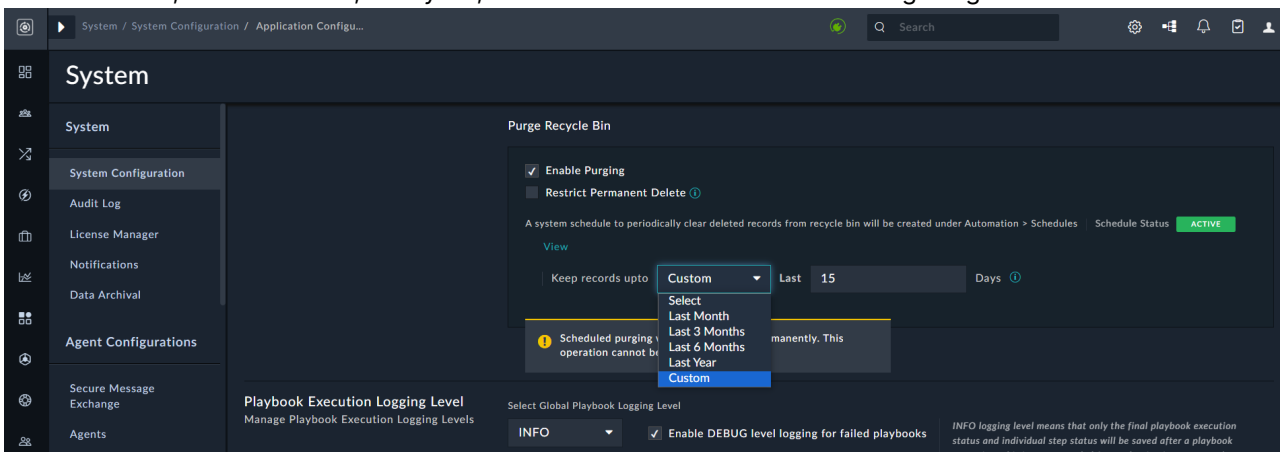
You can schedule the purging of recycle bin records to periodically clear the soft-deleted records from the recycle bin. For more information, see the Recycle Bin chapter.

To purge Recycle Bin records, you must be assigned a role that has a minimum of `Read` and `Update` permissions on the 'Application' module. To enable purging of recycle bin records do the following:

1. Click the **Settings** icon, which opens the `System Configuration` page.
2. On the `System Configuration` page, click the **Application Configuration** tab.
3. In the `Purge Recycle Bin` section, select **Enable Purging**.
4. Once you select the **Enable Purging** checkbox, you require to define the schedule for purging of recycle bin records.
5. In the `Purge Recycle Bin` section, set up the purging schedule, which is a system schedule named 'Recycle Bin Cleanup' that runs periodically as per the time-frame you have configured and purges recycle bin records:

a. Click **View** beside `Schedule Status`, which is set to *Inactive*, to open the `Schedule Details` dialog.
b. Click **Start Schedule** to begin the schedule immediately, or you can also set the Start Time and End Time for the schedule.
c. In the `Schedule Frequency` section, choose the frequency of running this schedule.
For example, to purge recycle bin records daily at 6:00 am, click **Daily** and then in the hour field enter 6 and in the minute field enter 0.
d. From the **Timezone** drop-down list, select the timezone in which you want the schedule to run. By default, this is set as UTC.
e. If you want to ensure that you do not rerun the workflow, if previous scheduled instance is yet running, then click **Limit execution to one active instance at a time**.
f. (Optional) From the **Start Time** field, select the date and time from when the schedule should start running.
g. (Optional) From the **End Time** field, select the date and time till when the schedule should run, i.e., the date and time to stop the schedule. For more details on schedules, see the *Schedules* chapter in the "User Guide."
h. Click **Save** to save the schedule.
Once the schedule is saved, you can see that the `Schedule Status` has changed to *Active*:

6. To specify the time for which you want to retain the recycle bin records, you must select the appropriate option from the **Keep records upto** drop-down list. You can choose from the following options: Last month, Last 3 months, Last 6 months, Last year, or Custom as shown in the following image:



If you choose **Custom**, then you must specify the number of days for which you want to retain the recycle bin records.
For example, if you want to retain the recycle bin records for a month, then select **Last month** from the **Keep records upto** drop-down list. Once you save this setting all recycle bin records that are older than 1 month (30 days) will be cleared, and this will be an ongoing process, as the records will all be time-stamped and the ones older than 30 days will be purged.

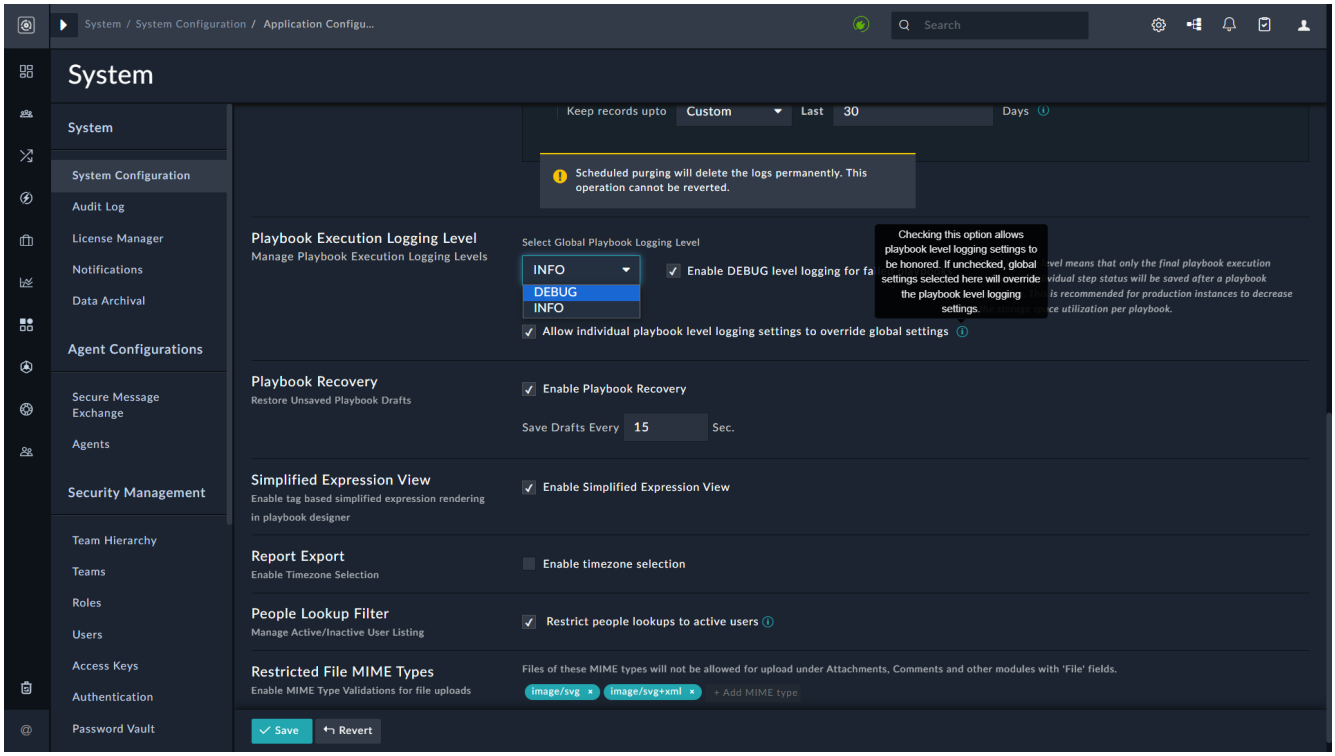
Restricting access for permanent deletion of recycle bin records to only "admin" users

To restrict access for permanently deleting records to only users with the 'Administrator' role, i.e., users having `Delete` permission on the 'Application' module, select the **Restrict Permanent Delete** option. If you enable this settings, then non-admin users can soft-delete records but cannot permanently delete them, which would help prevent accidental deletion of records by non-admin users. For more information on the Recycle Bin, see the Recycle Bin chapter.

Configuring the logging level for Playbook Execution Logs

You can define the logging levels for your playbook execution logs, both globally as well as at the individual playbook level. On the `System Configuration` page, you can choose either `INFO` or `DEBUG` as the global playbook logging level.

`INFO` is set as the default global playbook logging level for fresh installations of FortiSOAR. If you are upgrading FortiSOAR, then the `DEBUG` mode is set as the default playbook logging level to ensure that there is no data loss.



- **INFO**: At the '`INFO`' (default) level, only the final playbook execution status and individual playbook step status information is available after playbooks have completed their execution. It is recommended that you keep the logging level at `INFO` for production instances and in scenarios where you want to use storage space efficiently.
  When you retain the default `INFO` logging level, the `Enable DEBUG level logging for failed playbooks` checkbox is selected by default, which means that '`DEBUG`-level' logging is set for failed playbooks and users do not need to rerun the playbook to view the exact reason for playbook failures.
- **DEBUG**: At the '`DEBUG`' level, detailed logging is enabled that includes additional execution information like step input, output, configurations and other details.

  Enabling `DEBUG` level logging can quickly fill up the storage space. It is recommended to use it only while designing or debugging playbooks, or use this option wisely only for certain playbooks where this data might be useful

Retaining the selection of the **Allow individual playbook log level logging settings to override global settings** checkbox honors the logging level that has been set at the individual playbook level. If you clear this checkbox, or do not change the logging level at the individual playbook level (default is `INFO`), then the global playbook logging level gets applicable. This is useful if you want to temporarily switch logging for the entire environment.

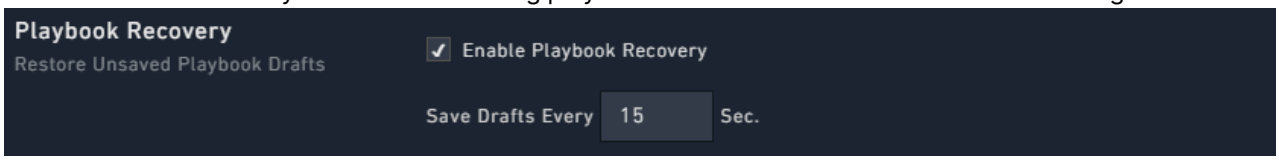
You can set the logging level of individual playbooks, even if you clear the **Allow individual playbook log level logging settings to override global settings** checkbox; however, at the time of playbook execution the global playbook logging level gets applied.

Configuring Playbook Recovery

Use the autosave feature in playbooks to recover playbook drafts in cases where you accidentally close your browser or face any issues while working on a playbook.

In the `Playbook Recovery` section, you can define the following:

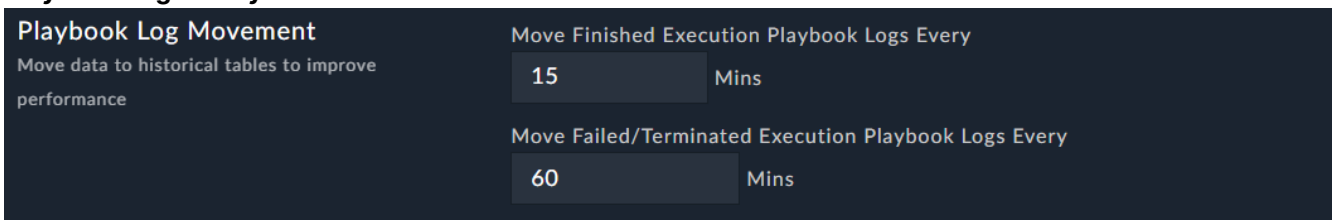
- If you do not want FortiSOAR to save playbook drafts, clear the **Enable Playbook Recovery** option. By default, this option is checked.
- In the **Save Drafts Every** field, enter the time, in seconds, after which FortiSOAR will save playbook drafts. By default, FortiSOAR saves playbook drafts **15** seconds after the last change.
  The minimum time that you can set for saving playbook drafts is **5** seconds after the last change.



Configuring playbook log movement to historical storage

In release 7.6.1, FortiSOAR introduces 'playbook log movement' to optimize the storage of workflow logs by transferring playbook logs to historical storage after playbooks complete their execution. By default, completed playbook logs are moved to historical storage every 15 minutes, while data for failed or terminated playbooks is transferred every 60 minutes.

You can customize these settings in the `Playbook Log Movement` section. To adjust the timing for moving completed playbook logs, update the **Move Finished Execution Playbook Logs every <> Mins** field. To modify the timing for transferring failed or terminated playbook logs, update the **Move Failed/Terminated Execution Playbook Logs every <> Mins** field:



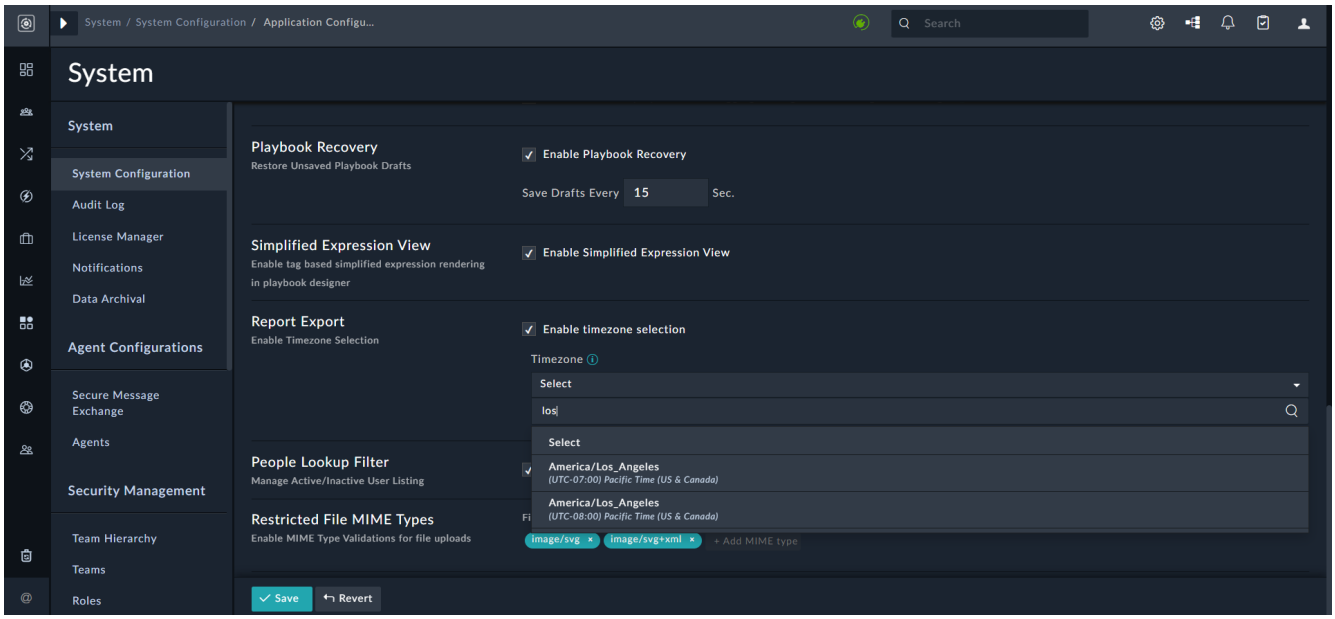
Configuring the Simplified Expression View

For fresh installations of FortiSOAR release 7.4.0 or later, by default, the **Simplified Expression View** option is selected on the `Application Configuration` page, rendering a simplified expression based on tags rather than the full Jinja expression in the playbook designer.

For upgraded FortiSOAR systems, by default, the **Simplified Expression View** option is cleared. If you deselect this option, then the complete Jinja expressions are displayed in the playbook designer. For more information, see the "Playbooks Guide."

Configuring the default timezone for exporting reports

You can define a timezone that will be used by default for exporting reports. This timezone will be applied by default to all reports that you export from the Reports page. To apply the default timezone, click the **Enable Timezone Selection** option in the `Report Export` section in the `Application Configuration` page. Then from the **Timezone** drop-down list, search for and select the timezone in which you want to export the report. For example, if you want to search for the timezone of Los Angeles, you can type `los` in the search box below the Timezone field to find the correct timezone, as shown in the following image:

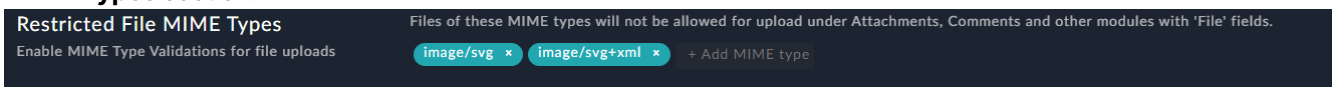


Managing user listings in `People` Lookup fields

You can choose to display only active users in 'people lookup' fields, such as `assignedTo`, across FortiSOAR. To manage user listings in people look up fields, ensure that the **Restrict people lookups to active users** checkbox in the `People Lookup Filter` section is selected. If the **Restrict people lookups to active users** checkbox is cleared then both active and inactive users will be displayed in people lookup fields.

Enabling MIME type validations for file uploads

You can specify the MIME types that will be disallowed from being uploaded in Attachments, Comments, or any other modules that have fields of type 'File'. Using this option, administrators can restrict potentially malicious files of types such as .exe, .bat, etc. to be uploaded into FortiSOAR, which users can later download. FortiSOAR has not added this restriction as defaults since there could be business use cases such as where users as part of automation read the file being sent to them in emails, and then upload the same to FortiSOAR to be used in the future for different operations like sandboxing, etc. Therefore, administrators can enable MIME type validations for file uploads as per their organization's policies by adding MIME types in the **Restricted File MIME Types** section:



From release 7.2.1 onwards, by default, SVG files are disallowed from being uploaded in any modules that have fields of type 'File'. If you want to allow uploading of SVG files, then you can remove the image/svg and image/svg+xml tags from the **Restricted File MIME Types** section.

In addition to restricting MIME types, you can also block specific HTML tags and attributes from being added to HTML content in Rich Text fields. For more information, see Blocking specific HTML tags and attributes.

Log Forwarding

Many organizations use an external log management server (syslog server) to manage logs and maintain all logs at a single place, making analysis efficient. FortiSOAR application logs and audit logs can be forwarded to your central log management server that supports a Rsyslog client, using both the FortiSOAR UI and the `csadm` CLI. You can also select the category of the logs you want to forward to the external log management server. For information about configuring forwarding of logs to an external log management server using the CLI, see the FortiSOAR Admin CLI chapter.

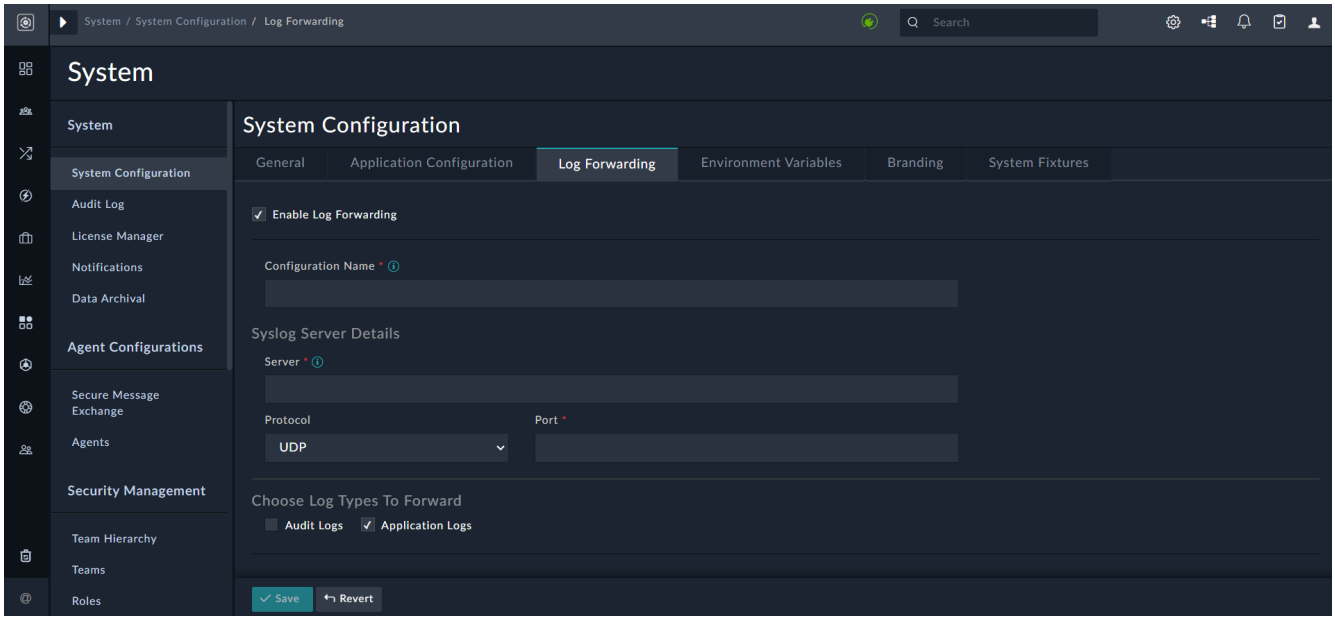
If you have a FortiSOAR HA setup, then note that Syslog settings are not replicated to the passive node. If you want to forward logs from the passive node, you must enable it manually using the `csadm log forward` command.

You could also send FortiSOAR logs to a SIEM, since all SIEMs support syslog ingestion, and which would help you achieve the following

- Ease High Availability (HA) troubleshooting since now you can use consolidated logs instead of having to go individual nodes to debug HA issues.
- Ability to forward FortiSOAR logs to your SIEM, if you have a policy of setting up log forwarding to SIEM for all your production devices.
  Once the logs are in the a SIEM, you can further configure rules for raising alerts for specific failure, making system monitoring more effective.

Click **Settings** > **System Configuration** and then click the **Log Forwarding** tab to open the `Log Forwarding` page. Use the `Log Forwarding` page to setup, modify, and enable or disable your syslog forwarding of

FortiSOAR logs to your central syslog server. To enable syslog forwarding, click the **Enable Log Forwarding** check box.



Once you select the **Enable Log Forwarding**, you require to fill in the details of the syslog server to which you want to forward the FortiSOAR logs, the type of logs to forward, etc.

You can configure only a single syslog server.

---

1. In the **Configuration Name** field, add the name of the configuration in which you want to store the log forwarding configuration details.
   Note: The name that you specify must not have any special characters, underscores, or spaces.
2. In the `Syslog Server Details` section, enter the following details:
   a. In the **Server** field enter the DNS name or IP address of the syslog server to which you want to forward the FortiSOAR logs.
   b. From the **Protocol** drop-down list, select the protocol that you want to use to communicate with the syslog server. You can choose between **UDP**, **TCP**, or **RELP**.
   c. In the **Port** field enter the port number that you want to use to communicate with the syslog server.
   d. (Optional) To securely communicate with the syslog server, click **Enable TLS**.
   Once you click **Enable TLS**, in the **Certificate** field, you must enter your CA certificate.
   If you have a client certificate for your FortiSOAR client, then in the **Client Certificate** and **Client Key** fields, you must enter the client certificate and the client key.
3. In the `Choose Log Types To Forward` section, choose the types of FortiSOAR logs you want to forward to the syslog server.
   Application Logs include OS logs, and this checkbox is selected by default. To also forward FortiSOAR audit logs, click the **Audit Logs** checkbox. Once you select audit logs, you can define the following:
   a. From the **Specify Audit Log Detail Level** drop-down list, select the amount of data, **Basic** or **Detailed** that you want to forward to the syslog server. Basic (default and recommended) sends high-level details of the event per audit log, whereas Detailed sends detailed information about the event per

audit log.

b. In the `Configure Audit Log Forward Rules` section, define the rules to forward audit logs:
From the **Record Type** drop-down list, select the record types such as, Alerts, Incidents, etc. whose audit logs you want to forward to the syslog server.
From the **User** drop-down list, select the users such as, CS Admin etc., whose audit logs you want to forward to the syslog server.
From the **Operation** drop-down list, select the operations such as Create, UpdateConfig, Delete, etc., whose audit logs you want to forward to the syslog server.
From the **Playbooks** drop-down list, select the operations such as Generate Incident Summary Report, Playbook Execution History Cleanup, etc., whose audit logs you want to forward to the syslog server.
To add more rules, click the **Define More Rules** link.
Important: If you do not define rules, then all the audit logs will be forwarded.

4. Once you have completed configuring syslog forwarding, click **Save**.
FortiSOAR performs validations such as, whether the syslog server is reachable on the specified port etc. before adding the syslog server.
Once the syslog server is added, you can update or remove the configuration as per your requirements.

Troubleshooting Log Forwarding issues

- For High Availability clusters in FortiSOAR release 7.6.0 or earlier, log forwarding settings configured from the primary node do not reflect on the secondary node(s).
To resolve this issue, run the following commands on the secondary node:
  a. `echo | openssl s_client -connect localhost:8444 -showcerts 2>/dev/null | sed -n '/-----BEGIN CERTIFICATE-----/,/-----END CERTIFICATE-----/p' > self_signed_certificate.crt`
  b. `cat self_signed_certificate.crt >> /opt/cyops-auth/.env/lib64/python3.9/site-packages/certifi/cacert.pem`
- If audit logs are not being forwarded to the syslog server, follow the troubleshooting steps outlined in the Technical Tip: Audit logs are not being forwarded article.

Persisting the FortiSOAR logs

If your external log management server goes down, then the FortiSOAR logs generated during that time period will not be sent by FortiSOAR to your syslog server. If you want to persist the logs for the time frame when external log management server is down and send those logs when server comes back online, you need to do the following:

In the `/etc/rsyslog.d/00-rsyslog-fortisoar-settings.conf` file, add the following contents after the `#### add the server details after this ####` line:

```
#### add the server details after this ####
$ActionQueueType LinkedList
$WorkDirectory /home/csadmin/.offline-rsyslogs/
#
# for the workdir mentioned above, make sure you run
# chown -R -t syslogd_var_lib_t /home/csadmin/.offline-rsyslogs/
#
$ActionQueueMaxDiskSpace 1gb # 1gb space limit (You can change this value)
```

```
$ActionQueueFileName fortisoar-offline-rsyslog
$ActionResumeRetryCount -1
$ActionQueueSaveOnShutdown on
```

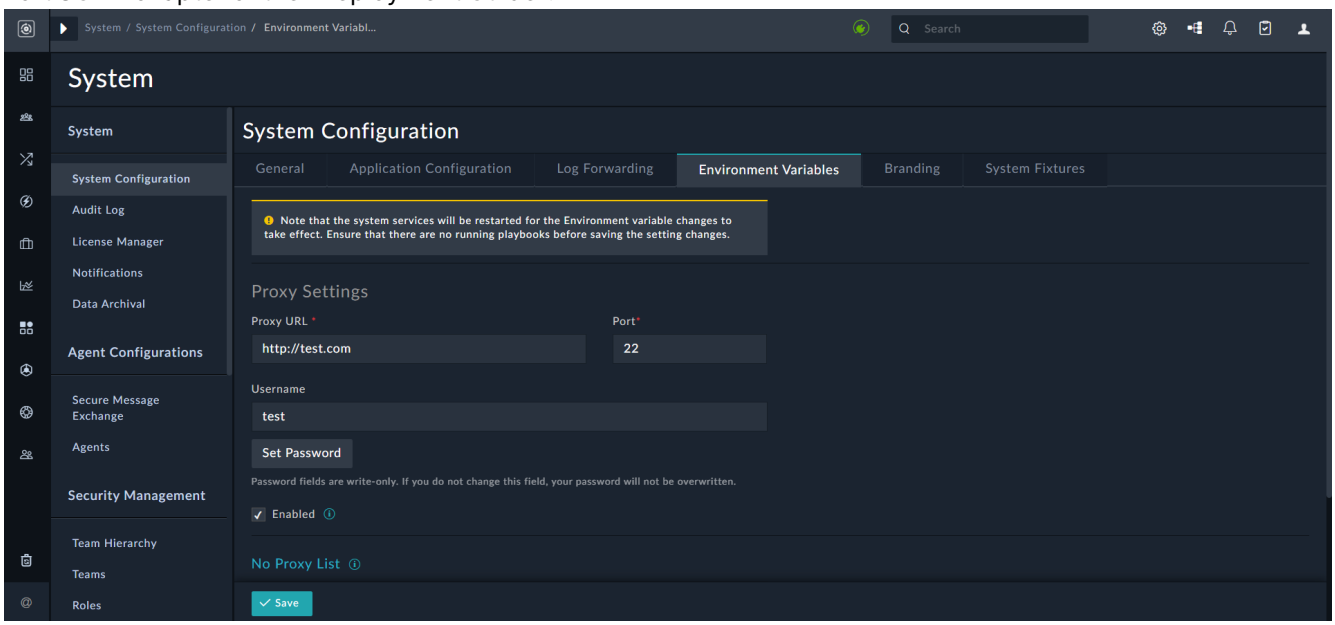
Next, run the following commands:

```
mkdir -p /home/csadmin/.offline-rsyslogs/
chcon -R -t syslogd_var_lib_t /home/csadmin/.offline-rsyslogs/
systemctl restart rsyslog
```

Environment Variables

You can use the **Environment Variables** tab on the `System Configuration` page to configure proxy settings for FortiSOAR and to define any other environment variables.

The procedure of how to configure proxy settings and define environment variables is included in the `Configuring Proxy Settings and environment variables` section in the *Additional configuration settings for FortiSOAR* chapter of the "Deployment Guide".

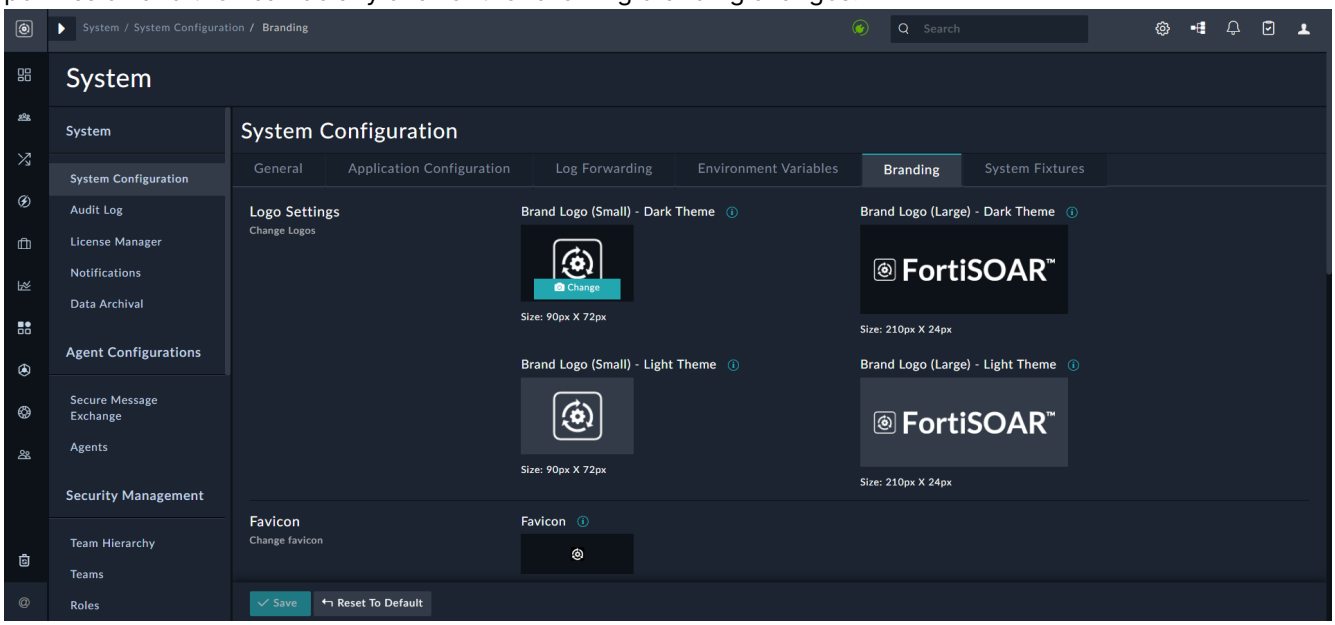


External web pages that you open (for example, from a link included in the description field of an alert) or view (for example, using the iFrame Widget) in FortiSOAR goes through the configured proxy server if you have configured the proxy in the web browser's settings. If the proxy is not configured in the web browser's settings, then the external web pages are opened directly without using the configured proxy server.

Branding

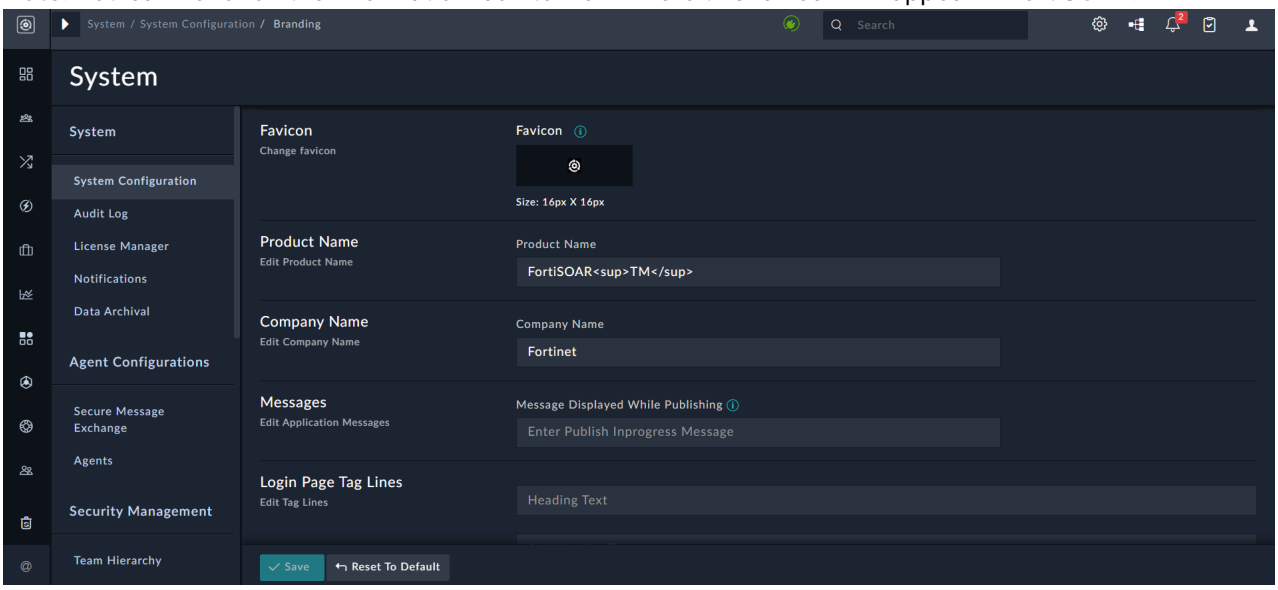
You can customize branding of FortiSOAR as per your requirement. Branding is not bound based on licensing, i.e., all customers can customize FortiSOAR branding as per their requirements.

To customize your branding in FortiSOAR, you must have a role which has a minimum of **Application Update** permission and then can do any or all of the following branding changes:

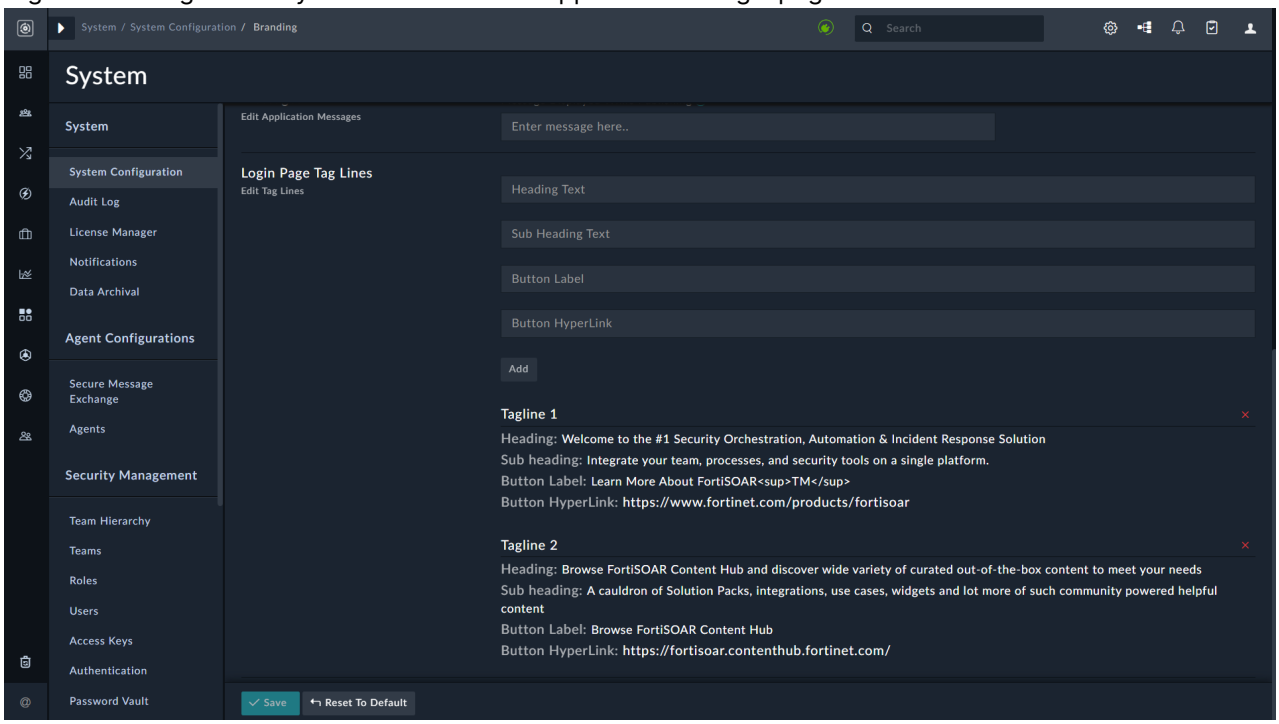


- **Changing Logos**: You can update the FortiSOAR logo to reflect your logo in the FortiSOAR UI. However, note that the maximum file size for a logo is 1 MB. Also, the different logos have specific dimensions, and if the image files exceed the specified dimensions, then the FortiSOAR UI displays relevant error messages and prevents users from uploading images that exceed the file size and/or have the wrong dimensions.
  You can update your logo in the `Logo Settings` section:
  Brand Logo (Small) - Dark Theme and Brand Logo (Large) - Dark Theme: Click the FortiSOAR logos and browse to the logos that you want to display in FortiSOAR Dark or Steel theme in two dimensions: Small (90px X 72px) and Large (210px X 24px).
  Brand Logo (Small) - Light Theme and Brand Logo (Large) - Light Theme: Click the FortiSOAR logos and browse to the logos that you want to display in FortiSOAR Light theme in two dimensions: Small (90px X 72px) and Large (210px X 24px).
  Note: You can hover on the information icon to view where these logos will appear in FortiSOAR.
- **Changing the Favicon**: To change the favicon that is displayed in FortiSOAR, click the FortiSOAR favicon and browse to the icon that you want to display as a favicon. Dimensions of favicon must be 16px X 16px.

Note: You can hover on the information icon to view where this favicon will appear in FortiSOAR.



- **Editing the Product Name**: To change the name of the product displayed in the FortiSOAR UI, in the **Product Name** field, enter the name of the product that you want to display in the UI.
- **Editing the Company Name**: To change the name of the company displayed in the FortiSOAR UI, in the **Company Name** field, enter the name of the company that you want to display in the UI.
- **Editing Application Messages**: To change the default message, "System Publish Underway, Please Wait" that users see during the publish operation, in the **Message Displayed While Publishing** field enter the customized message to be displayed to user during the publish operation.
- **Editing the Login Tagline**: To change the customized messages or taglines that appears to all users on their login screen, you can deselect the default tagline(s) by clicking the red cross that appears beside the tag line. The tagline that you deselect will not appear on the login page.



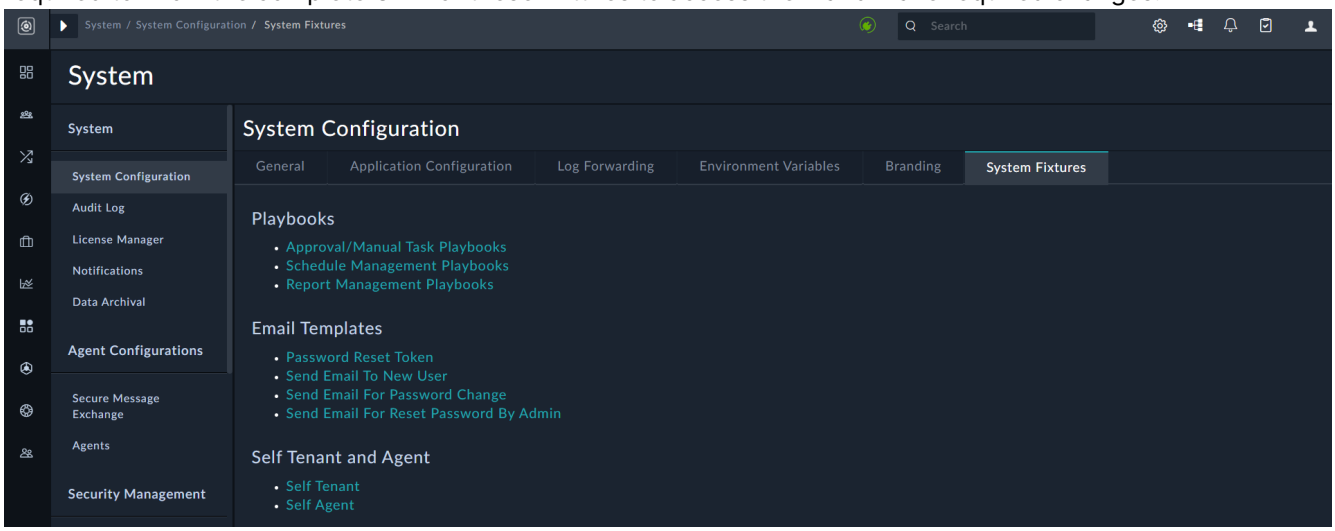
You can then add your own tag line in the `Login Page Tag Lines` section as follows:

In the **Heading Text** field: Enter the heading for your tagline.
In the **Sub Heading Text** field: Enter the sub-heading for your tagline.
In the **Button Label and Button Hyperlink** fields: If you want to add a button on your login page, which on clicking by the user, navigates the user to another web page, then enter the label of the button and the URL of the other web page in the **Button Label** and **Button Hyperlink** fields respectively.
Once you complete adding your tag line, click **Add**.

To save your branding updates, click **Save**, to reset the branding to its default, click **Reset to Default**.

System Fixtures

The FortiSOAR UI includes links in the `System Fixtures` page to the various playbook collections, email templates, the self agent, and the self tenant pages, which included by default when you install your FortiSOAR instance. Click the **System Fixtures** tab on the `System Configuration` page to view the links to the system playbook collections and templates. Administrators can click these links to easily access all the system fixtures to understand their workings and make changes in them if required. In the previous versions, administrators required to know the complete URL for these fixtures to access them and make required changes.



You can see these fixtures when you install FortiSOAR and the SOAR Framework solution pack. As you install other solution packs, you might see system fixtures added by those respective solution packs.

The following fixtures are included:

Playbooks:

- **Approval/Manual Task Playbooks** collection: Includes a collection of system-level playbooks that are used to automate approvals and manual tasks, such as triggering the resume playbook when the input is received for manual tasks.
- **Schedule Management Playbooks** collection: Includes a collection of system-level playbooks that are used for the scheduler module and used for various scheduler actions such as scheduling playbook execution history cleanup, audit log cleanup, etc.

- **Report Management Playbooks** collection: Includes a collection of system-level playbooks that are used to manage generation of FortiSOAR Reports, such as exporting reports, generating reports, generating reports based on a schedule, etc.

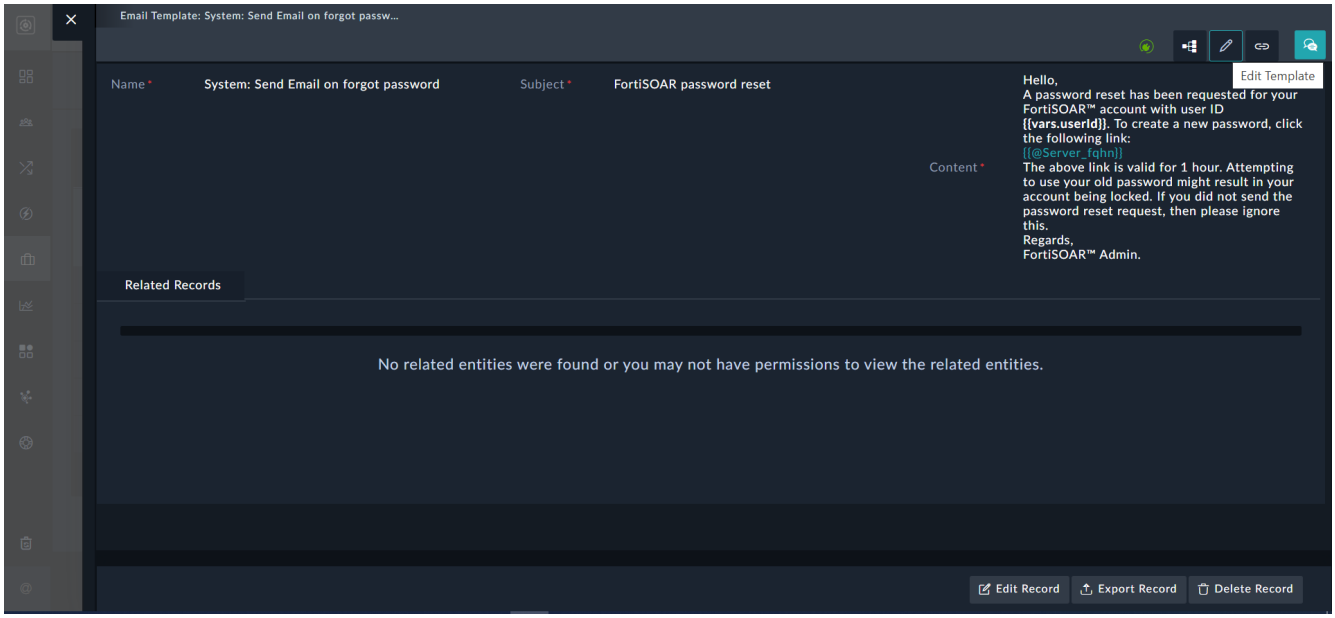
Playbooks that contain a reference to the `approvalHost` global variable fail with the '`approvalHost variable undefined`' error, since the `approvalHost` global variable is removed from release 7.2.0 onwards. To resolve this error, replace the `approvalHost` global variable in the playbook with the `Server_fqhn` global variable.

For more information on system-level playbooks, see the *Introduction to Playbooks* chapter in the "Playbooks Guide."

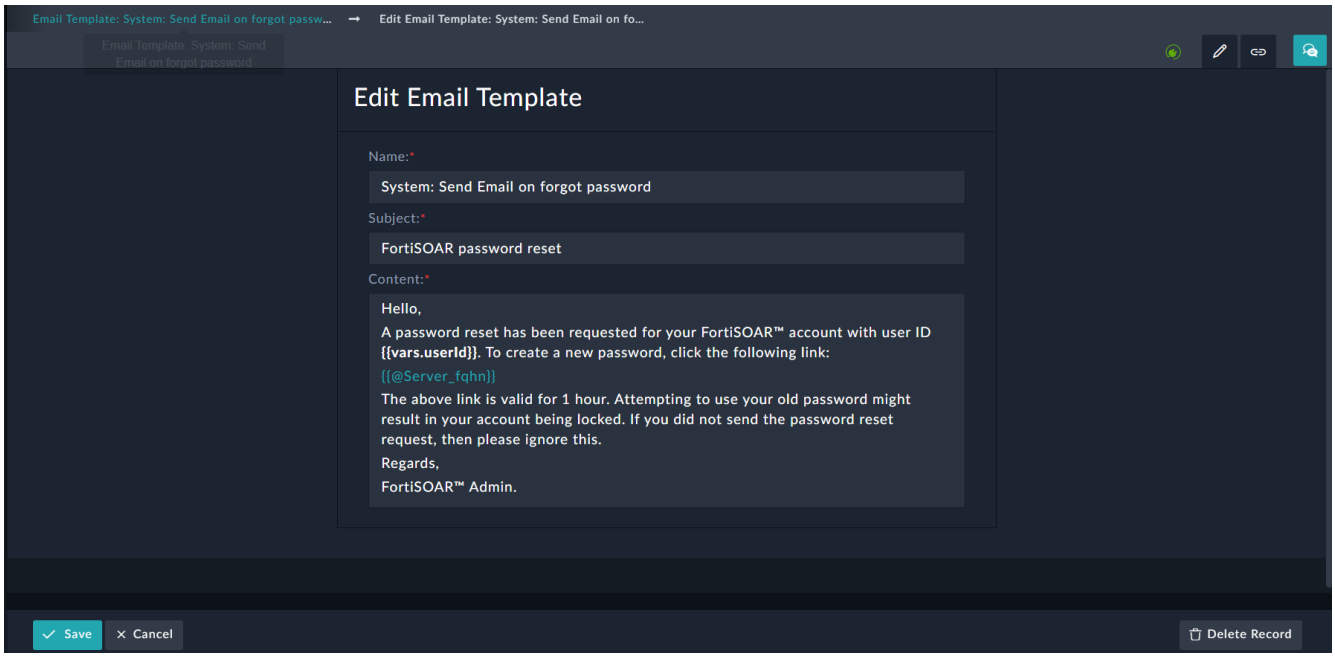
Email Templates

- **Password Reset Token**: Includes an email template that is sent to FortiSOAR users' who forget their password and click on the **Forgot Password** link, so that they can reset their password. This email contains a link that the user can use to create their new password.
- **Send Email To New User**: Includes an email template that is sent to a new FortiSOAR users' and it contains a link that the new user can use to create their own new password.
- **Send Email For Password Change**: Includes an email template that is sent when a user requests for a change in their FortiSOAR password.
- **Send Email For Reset Password By Admin**: Includes an email template that is sent to FortiSOAR users' whose password has been reset by an administrator.

To modify the content of the email templates, click the email template whose content you want to change, for example, click **Password Reset Token** to open the email template. Click the **Edit Record** button to edit the contents of the template. You can also click the **Edit Template** icon to edit the structure of the email or click **Actions** to perform actions on the record.



To change the content of the email, click the **Edit** icon, or click the **Edit Record** to open the email template in a "form" format in which you can change the contents of the email as per your requirement, and then click **Save** to save your changes.

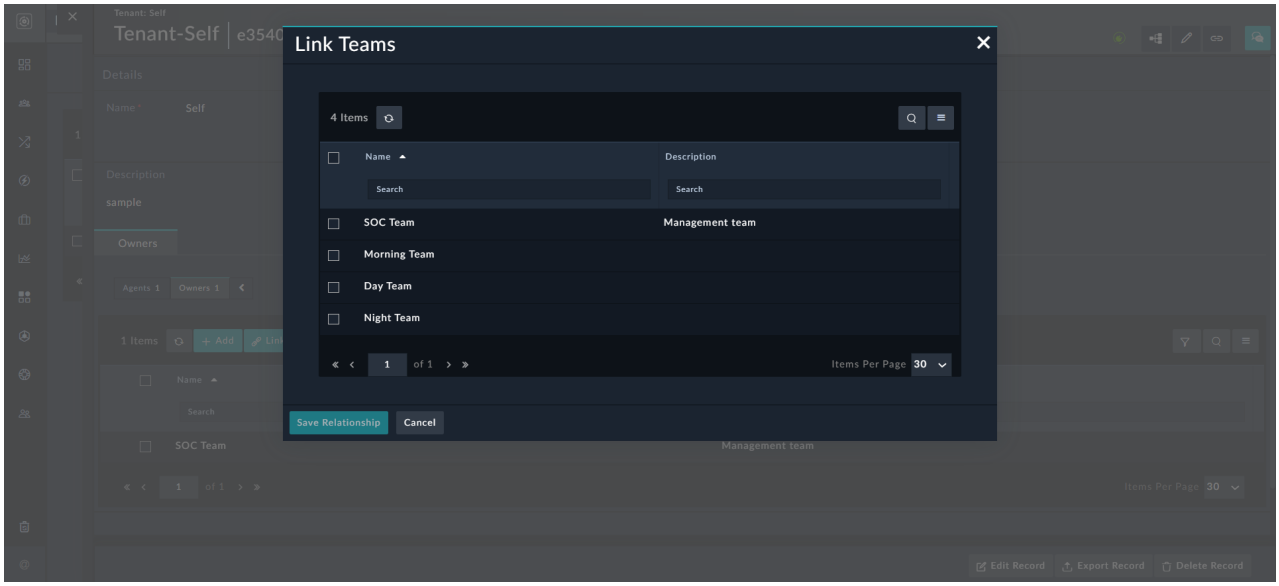


In case you have deleted the email templates, and you require to update or modify the default email templates, then you require to edit the `mailtemplate.yml` file located at `/opt/cyops/configs/cyops-api/`

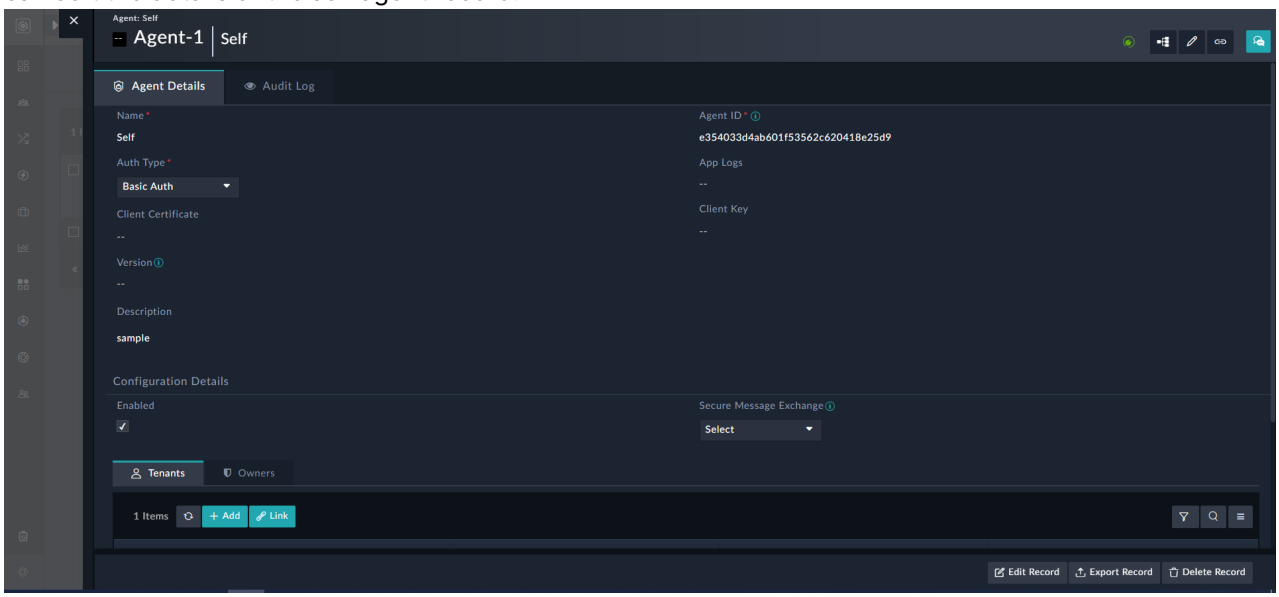
Self Tenant and Agent

This section contains dedicated pages for the self tenant and self agent, making it easier to update details of the self tenant and agent such as remapping teams linked to the self tenant or tenants linked to the self agent.

- **Self Tenant**: Click the **Self Tenant** link to access the self tenant in its own page in a new window, where you can edit the details of the self tenant record:



- **Self Agent**: Click the **Self Agent** link to access the self tenant in its own page in a new window, where you can edit the details of the self agent record:



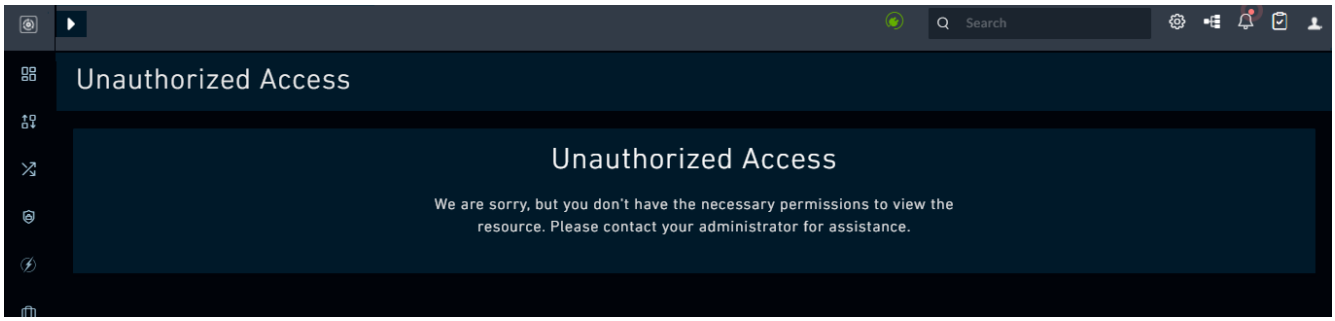
Audit Log

You can view the historical record of activities across FortiSOAR using the Audit Log, the User-Specific Audit Logs, and the graphical representation of the Audit Log in the detail view of a record.

Audit Log Permissions

- To view your own audit logs, you must have a role with a minimum of `Read` permission on the `Audit Log Activities` module. To view audit logs of all users, you must have a role with a minimum of `Read` permission on the `Security` and `Audit Log Activities` modules.
- To filter audit logs based on users you must have a role with a minimum of `Read` permission on the `People`, `Appliances`, and `Audit Log Activities` modules.
- To delete your own audit logs, you must have a role with a minimum of `Delete` permission on the `Audit Log Activities` module. To delete audit logs of all users, you must have a role with a minimum of `Delete` permission on the `Security` and `Audit Log Activities` modules.
  Note: The `Delete` permission on the `Audit Log Activities` module will be removed for both `csadmin` and playbook appliances roles, and also this will not be enabled (checked) by default for the **Full App Permissions** role. Therefore, if you want any user or role to have the right to delete audit logs, you must explicitly assign the `Delete` permission on the `Audit Log Activities` module to that particular user or role.

If you cannot access the Audit Log, you must ask your administrator for access. FortiSOAR displays an error, as shown in the following image, if you do not have access to Audit Logs:



You can view historical record of activities across FortiSOAR using the following options:

- **Audit Log**: Audit Log displays a chronological list of all the actions across all the modules of FortiSOAR. Click **Settings** > **Audit Log**to open the `Audit Log` page.
- **User-Specific Audit Logs**: User-Specific Audit Logs displays a chronological list of all the actions across all the modules of FortiSOAR for a particular user.
- **Viewing Audit Log in the detailed view of a record**: You can view a graphical presentation, or grid view, of all the actions performed on that particular record. The audit log is displayed in a graphical format using the `Timeline` widget.

Audit Logs include data such as, recording the name of the user who had deleted the record, linking and delinking events, picklist events, and model metadata events (including changes made in model metadata during the staging phrase). Free text search, additional filtering criteria, the ability to quickly add auditing for a new service and lazy loading has also been implemented in audit logs.

Audit logs also contain operations related to playbooks such as trigger, update, terminate, resume, create and delete playbook versions, etc. From version 6.4.1 onwards, the operations such as `Snapshot Created` and `Snapshot Deleted` operations are renamed to `Version Created` and `Version Deleted` since snapshots have been renamed to versions. However, for older audit logs in cases of an upgrade, you will see the old audit log entries with the older names such as `Snapshot Created` or `Snapshot Deleted`.

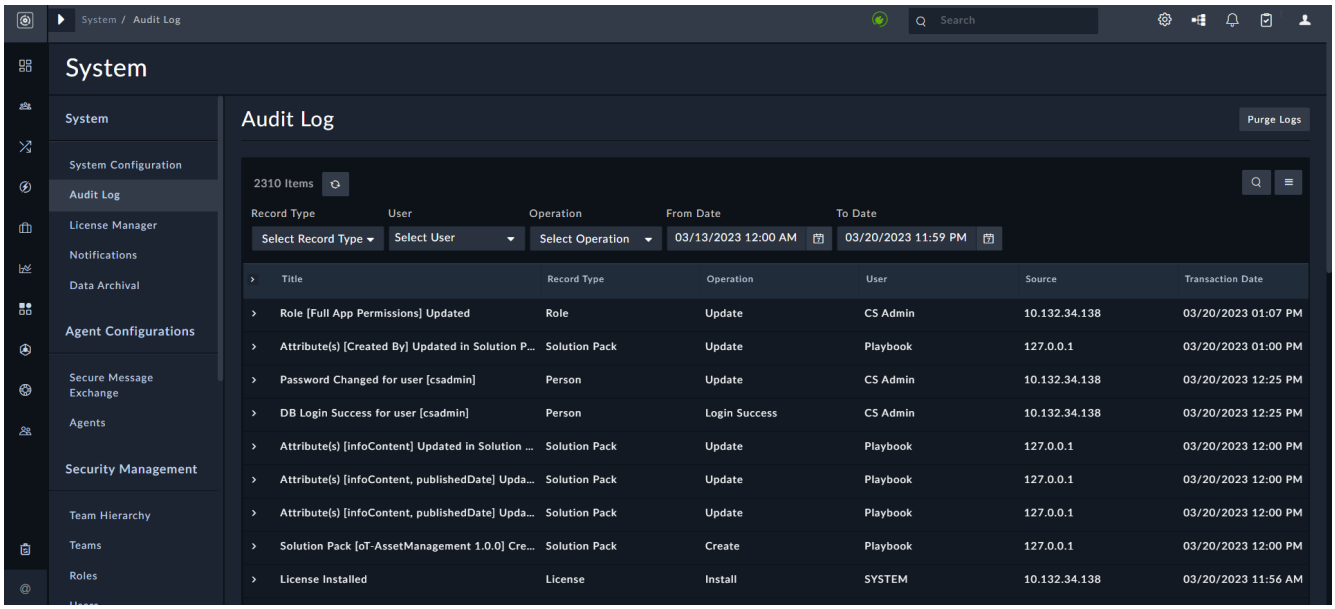
The data included in the audit log also contains the following types of entries:

- Users' login success or failures and logout events. The login event includes all three supported login types, which are DB Login, LDAP Login, and SSO Login.
- Users' login with an invalid username.
- Locked users's attempts to log on to FortiSOAR.
- Locking of users' account in the event of multiple failed login attempts.
- Inactive users's attempts to log on to FortiSOAR.
- Forced log out events by an administrator using the UI
- Forced log out events by an administrator using the CLI
- Change in user's access type, i.e., Named to Concurrent, or vice-versa, by an administrator
- Triggering of the workflow execution history cleanup job, i.e., the 'pg_squeeze' and 'pg_repack' tasks.
- Success or failure entry every time the storage space reclamation job is run along with the information about the date and time of the job run and table name on which the job is run.
  Success entry when the storage reclamation job is successfully completed: "Workflow Execution history cleanup job Completed".
  Failure entry on failure of the storage reclamation job: "Workflow Execution history cleanup job Failed". This occurs mainly due to insufficient space.
- Success or failure entry for every pg_squeeze' and 'pg_repack' run.
- Creating, updating, and deleting rules and channels.

- Deleting system notifications.
- Purging of system notifications.
- Soft deletion of records.
- Restoring of records from Recycle Bin.
- Creation of audit records each time a pre-processing rule matches a record. These records include the rule name, UUID, record information and date.
- All operations involving the creation, update, or deletion of pre-processing rules are audited for tracking purposes. Failures in executing pre-processing rules are also audited.
- Records that are dropped due to the 'Drop' pre-processing rule and records that are updated due to the 'Update' pre-processing rule are also audited.
- All operations related to API key, such as regenerating an API key, creating an API key, updating an API key, etc.
- Success or failure entry every time the data archival service is run.
- Setting of the archival option for a module.
- Changing configuration of rules
- Creating, updating, or deleting playbook blocks
- Creating, activating, deactivating or deleting an MFA method
- Adding global configuration for an MFA method
- Mandating 2FA for all users
- Activating grace period for each MFA user
- Locking user account after expiration of grace period

If you have a field, in a module, whose `Singular Description` attribute value contains a `.` or `$` then the Audit Logs replace the `.` or `$` with an `_`. For example, if you have a field SourceID whose singular description you have specified as `Source.ID`, then in this field will appear as `Source_ID` in Audit Logs.

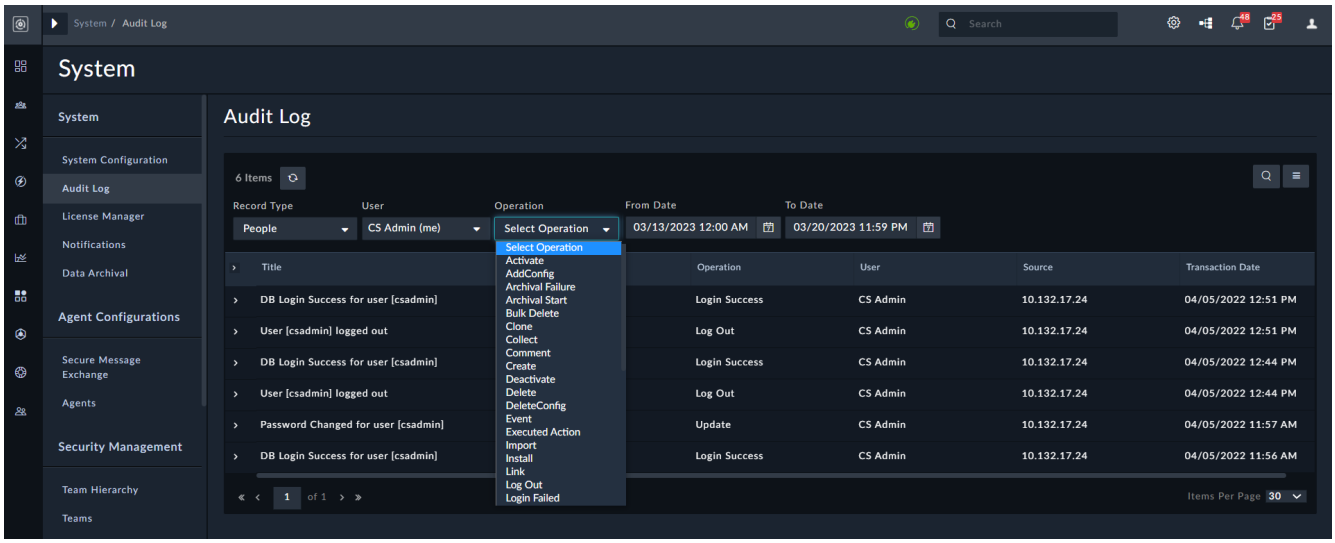
You can purge Audit Logs using the **Purge Logs** button on the top-right of the `Audit Log` page. You will see the **Purge Logs** button only if you have **Delete** permissions on the `Audit Log Activities` module.



You can also use the Audit Log Purge API to purge audit logs on an automated as well as an on-demand basis. For more information, see the *API Methods* chapter in the "API Guide."

Viewing Audit Log

Use the Audit Log to view a chronological list of all the actions across all the modules of FortiSOAR. To view the `Audit Log` page, you must have access to the `Audit Log Activities` module. Click **Settings** > **Audit Log** to open the `Audit Log` page. The `Audit Log` page filters audit logs, by default, to only show audit data from the previous 8 days. User can modify the date filter to suit their needs:



The audit log displays users' login success or failures and logout events. The login event includes all three supported login types, which are DB Login, LDAP Login, and SSO Login. You can filter the audit logs to display the audit logs for a particular record type, such as Alerts, Rules, API Key, Incidents, Tasks, etc., by selecting the

record type from the **Record Type** drop-down list. You can also filter audit logs on users, operations, and data range, apart from modules.

To filter audit logs on for a particular user, select the user from the **Select User** drop-down list.

To filter audit logs on for a particular operation, select the operation from the **Select Operation** drop-down list. You can choose from the operations such as, Comment, Create, Delete, Link, Login Failed, Snapshot Created, Trigger, Unlink, Update, etc.

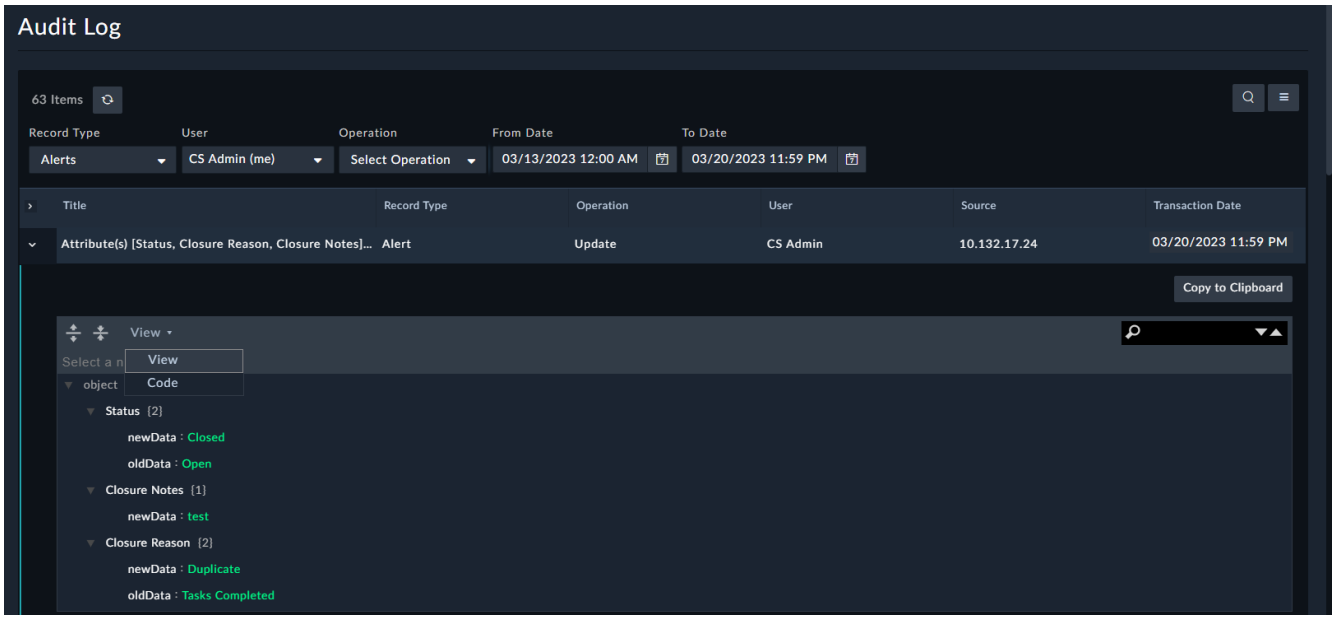
You can also filter audit logs for a particular date range by selecting the **From Date** and **To Date** using the calendar icon.

You can also search for audit logs using free text search. Click the **Search** icon and enter a search criterion in the **Search Logs** field to search the audit logs.

The Audit Log displays the following historical information for each record:

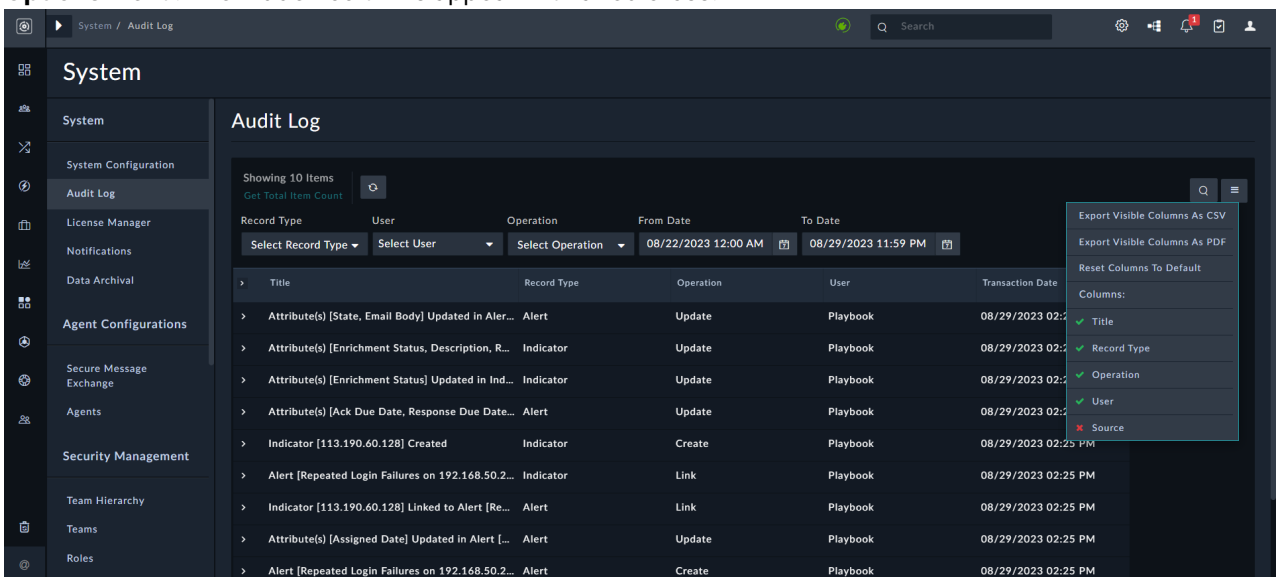
- `Title`: Title of the record on which the action was performed.
  Note: In case of Approval playbooks the playbook audit log displays the Approval Description field, which represents the name of the approval record, in the Title field. In this case, the Title field will be displayed in the format `Approval [Approval Description] Operation Performed`. For example, `Approval [Approval Test] Created`.
- `Record Type`: Type (module) of the record on which the action was performed such as Alerts, Incidents, Configs, Indicators, etc.
- `Operation`: Operation that was performed.
- `User`: User who performed the operation such as the name of the user who performed the operation, or if the operation was performed by the system, playbooks, agents.
  Note: From version 7.0.2 onwards, operations performed by FortiSOAR such as the creation of schema for various modules (Alerts, Events, etc.) are added as a global audit log user filter named '`SYSTEM`', and is applicable for audits generated from version 7.0.2 onwards. Audits done by the `SYSTEM` user prior to version 7.0.2 will not be visible when filtered by the `SYSTEM` user.
- `Source`: Source IP address where the operation that was performed.
- `Transaction date`: Date and time that the record was updated in the format `DD/MM/YYYY HH:MM`.

To view the details of an audit log entry, click the expand icon (**>**) in the audit entry row. Details in the audit log entry are present in the JSON format, and include the old data and updated (new) data for a record, in case of an `update` operation, and all attributes and their details, such as ID and type, for a record, in case of a `create` operation. You can copy the data using the **Copy to Clipboard** button.

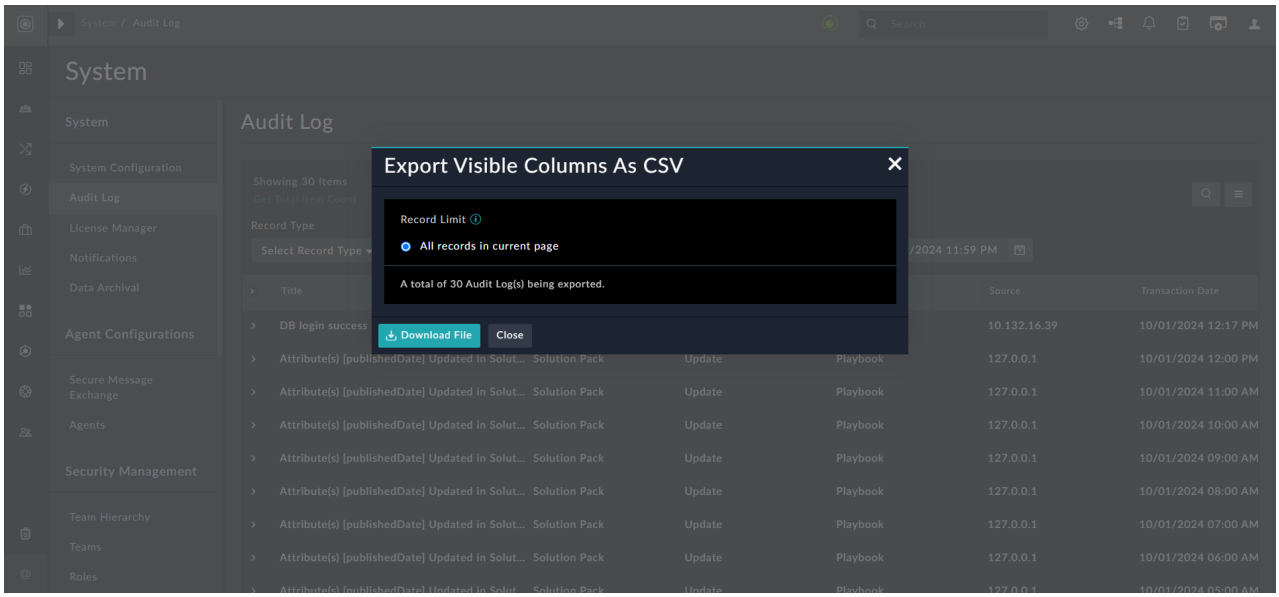


You can perform the following operations on the `Audit Log` page, by clicking the **More Options** icon ( ) to the right of the table header:

- **Export Visible Columns As CSV**: Use this option to export visible columns of the audit log to a `.csv` file. **Note**: You can hide columns by deselecting a column from the list of columns present within the **More Options** menu. The hidden columns appear with a red cross.



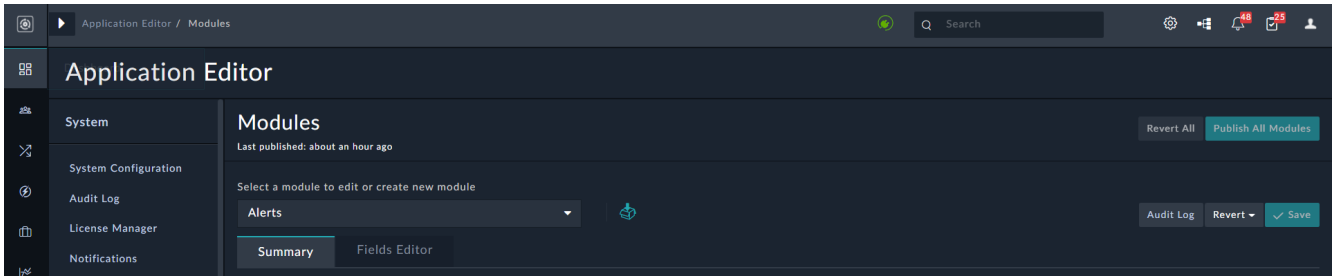
Once you click this option, the `Export Visible Columns As CSV` dialog is displayed:



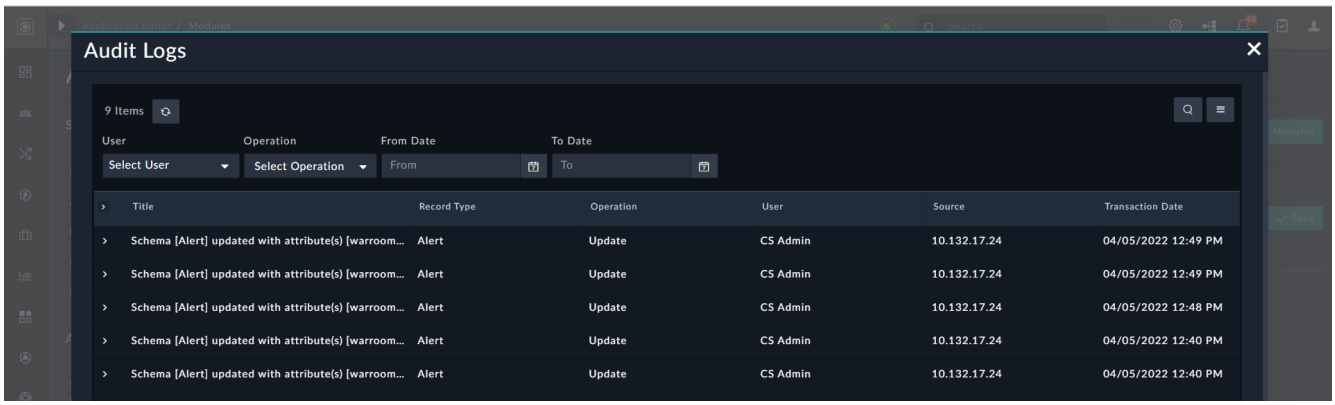
In this dialog, the **All records in current page** option is selected by default, which exports visible column data of the records on the current page. If you clear this option, then visible column data of all the records is exported.
Click **Download File** to download the CSV file containing the exported records.

- **Export Visible Columns As PDF**: Use this option to export visible columns of the audit log to a `.pdf` file.
  Note: If you want to export records from the record's grid and view panel that contain unsupported character sets such as Korean or Chinese, then you need to perform additional configurations. For more information, see the 'Configurations required for exporting of records with unsupported character sets in the PDF format' topic in the Debugging, Troubleshooting, and Optimizing FortiSOAR chapter.
- **Reset Columns To Default**: Use this option to reset the audit log fields to the default fields specified for the audit log.

You can view logs specific to a particular module, by clicking **Settings** > **Modules** (in the `Application Editor` section) and from the **Select a module to edit or create new module**drop-down list, select the module whose audit log you want to view, and then click the **Audit Logs** button.



You view the same details and perform the same actions as mentioned earlier on the `Audit Logs` Dialog. You can filter the audit logs for modules on users, operations, and date range. For example, you can filter logs which have an **Create** operation performed on a particular record type (module), as shown in the following image:



Similarly, you can also view logs specific to a particular picklist, go to **Settings** > **Picklists** (in the `Application Editor` section). From the **Select a picklist or edit or create a new picklist** drop-down list select the picklist whose audit log you want to view and click the **Audit Logs** button. You view the same details and perform the same actions as mentioned earlier on the `Audit Logs` Dialog. You can filter the audit logs for picklists on users, operations, and date range.

Audit logs also include the auditing of the following actions so that you can get comprehensive records of all activities across FortiSOAR :

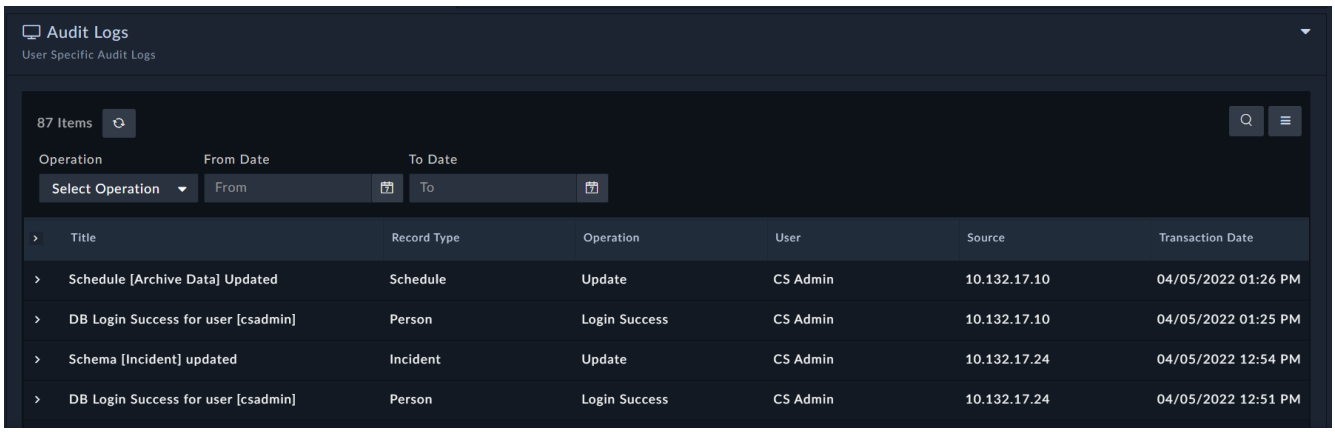
- Schedule actions - Create, update and delete, start, and stop
- Rules actions - Create, update, delete, activate, and deactivate
- Dashboard/Report actions - Create, update, and delete
- Navigation actions - All
- License Update actions - All
- SVT Update actions - All
- Role - Modification (Create and Delete already gets audited)
- Team - Modification (including team hierarchy updates), User Link/Unlink (Create and Delete already gets audited)

From version 7.0.0 onwards, the following fields have been added to audit and system logs to provide more information about your FortiSOAR system and to help *FortiAnalyzer* (FAZ) integration:

- **vd**(enterprise|master|tenant) - The value of this field is "enterprise" for an enterprise setup, "master" for the master node in a multi-tenant setup, and "tenant" for the tenant node in a multi-tenant setup.
- **level**(emerg|alert|crit|err|warning|notice|info|debug) - The severity level of the event.
  Note that the following audit operations will be considered as 'warning' severity operations: Delete, Unlink, Terminate, Version Deleted, Uninstall, DeleteConfig, Deactivate and Replication Failed. All other audit operations are considered as 'info' severity.
- **devid** - FortiSOAR's SNO, i.e., the same serial number from the license file.
- **datetime** - Event timestamp in the 'epoch' format. This is applicable only for audit logs.
- **type**(AuditLog|System Log) - The Log type.
  Sample Audit log: `2021-02-19T06:17:24.958779+00:00 fsrprimary fortisoar-audit-log: CEF:0|Fortinet Inc|FortiSOAR|7.0.0|Alert Deleted|Alert Deleted|1|devid="FSRVMPTM20000061" vd="enterprise" level="warning" type="Audit Log" msg="Alert [1] Deleted " src="192.168.56.1" suid="3451141c-bac6-467c-8d72-85e0fab569ce" suser="CS Admin" end=1613634028029 playbookName="" playbookId="" eventTimeStr="18 Feb 2021 07:40:28.029"`

Viewing User-Specific Audit Logs

Use the User-Specific Audit Logs to view the chronological list of all the actions across all the modules of FortiSOAR for a particular user. Users can view their own audit logs by clicking the **User Profile** icon and selecting the **Edit Profile** option and clicking the **Audit Logs** panel. Administrators who have a minimum of `Read` access on the `Audit Log Activities` module along with access to the `People` module, which allows them to access a user's profile, can view **User Specific Audit Logs**. The user-specific audit log also displays user's login success or failures and logout events. The login event includes all three supported login types, which are DB Login, LDAP Login, and SSO Login.



Use the same filtering and searching techniques mentioned in the Viewing Audit Log section. You can filter the user-specific audit logs on record types (modules) and date range.

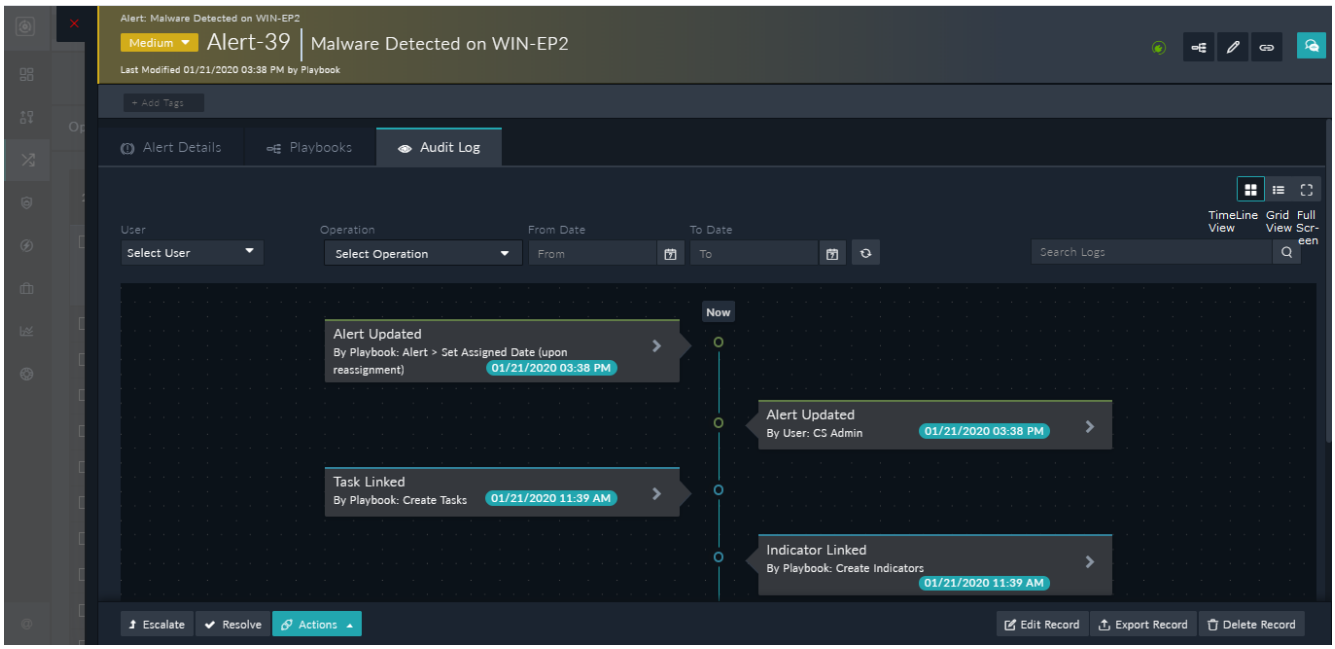
The user-specific audit logs display the same information as the audit log, and you can also perform the same actions here as you can perform in case of audit logs. For more information, see the Viewing Audit Log section.

Viewing Audit Log in the detailed view of a record

Use the `Audit Log` tab, which is present in the detail view of a record, to view the graphical presentation of all the actions performed on that particular record. The `Audit Log` tab uses the `Timeline` widget to display the graphical representation of the details of the record. You cannot edit the `Timeline` widget. For more information about widgets, see the *Using Template Widgets* topic in the "User Guide."

You can toggle the view in the **Audit Log** tab to view the details in both the grid view and the timeline (graphical) view. Use the same filtering and searching techniques mentioned in the Viewing Audit Log section. You can filter the user-specific audit logs on record types and date range.

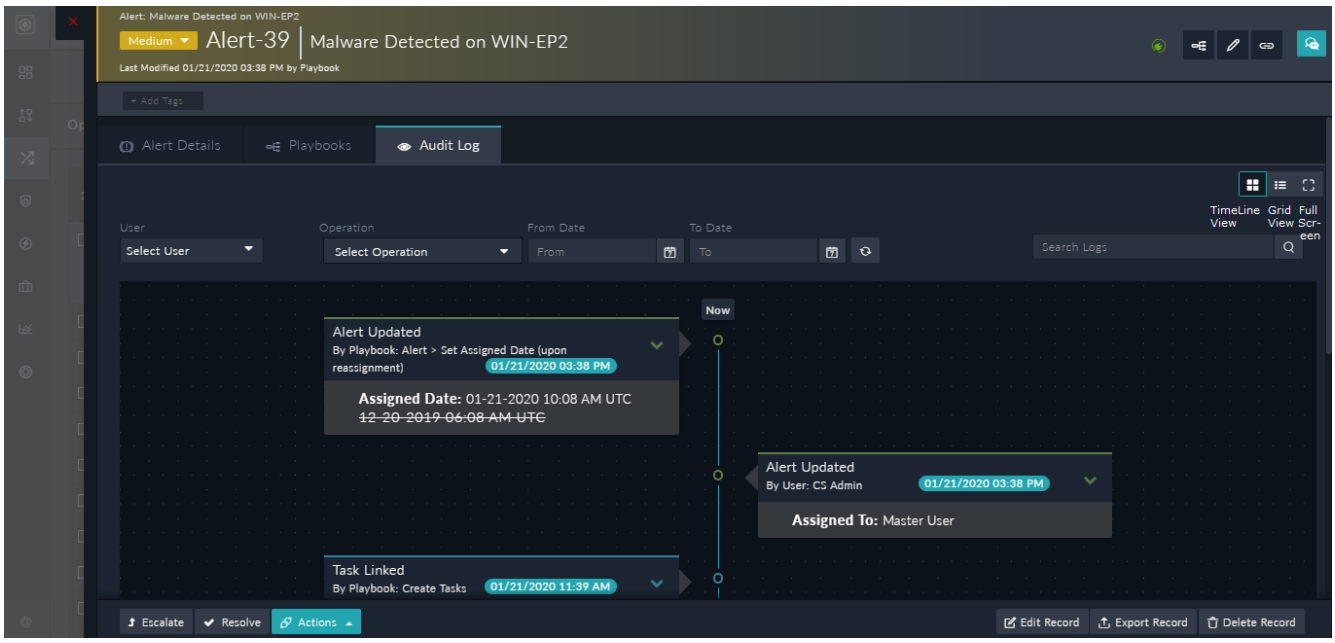
Click a record within a module to open the detail view of a record and then click the **Audit Log** tab to view the graphical representation, or grid view of the details of the record, as shown in the following image:



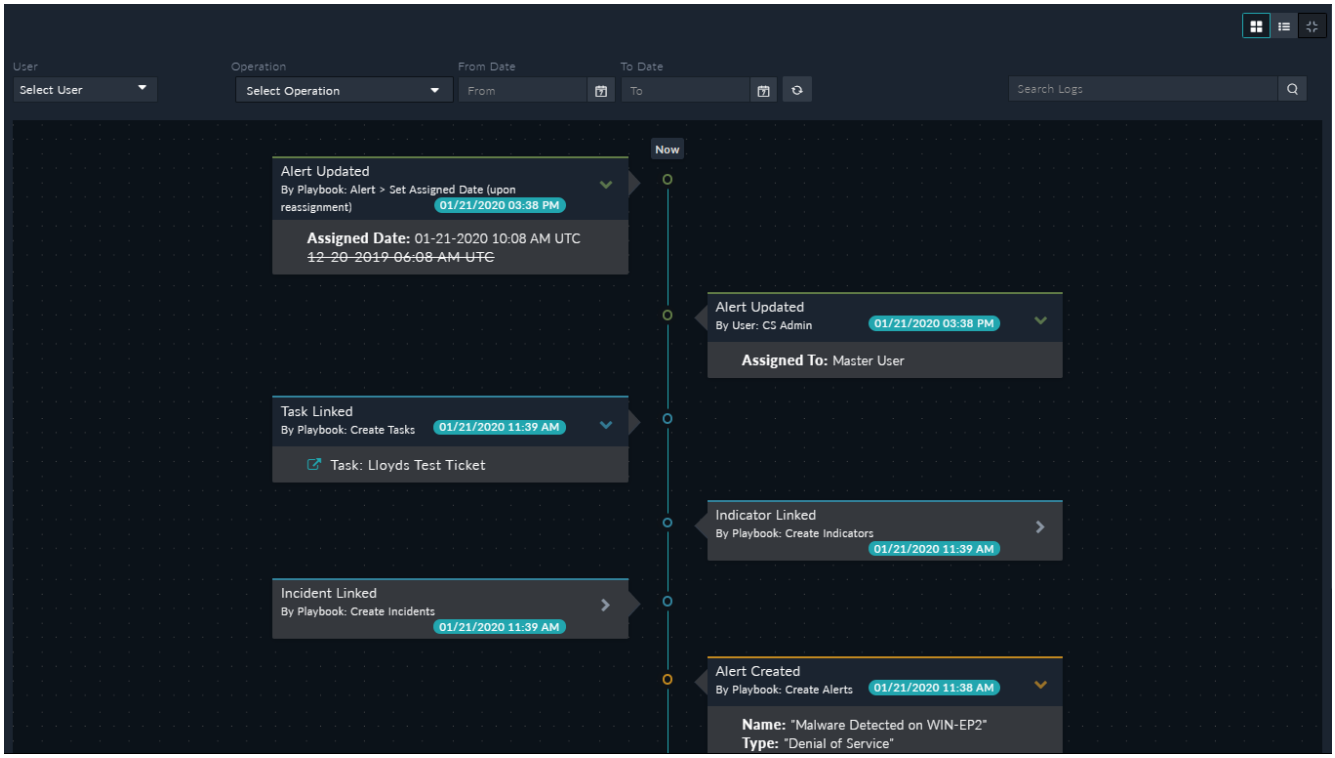
A timeline item mentions the action performed on the record, such as `Created`, `Updated`, `Commented`, `Attached`, or `Linked`, the name of the person who has made the update, and the date and time that the update was made.

In the timeline, you might see some records created by `Playbook`. This signifies that the record was created by a workflow entity, such as a Playbook or a Rule.

When you update any detail in a record, then you can click the refresh button in the timeline to view the updates in timeline immediately. To view the complete details of the updates made at a particular timeline item, click the arrow (**>**) present to the right of the item. The following image displays the details shown for a specific timeline item:

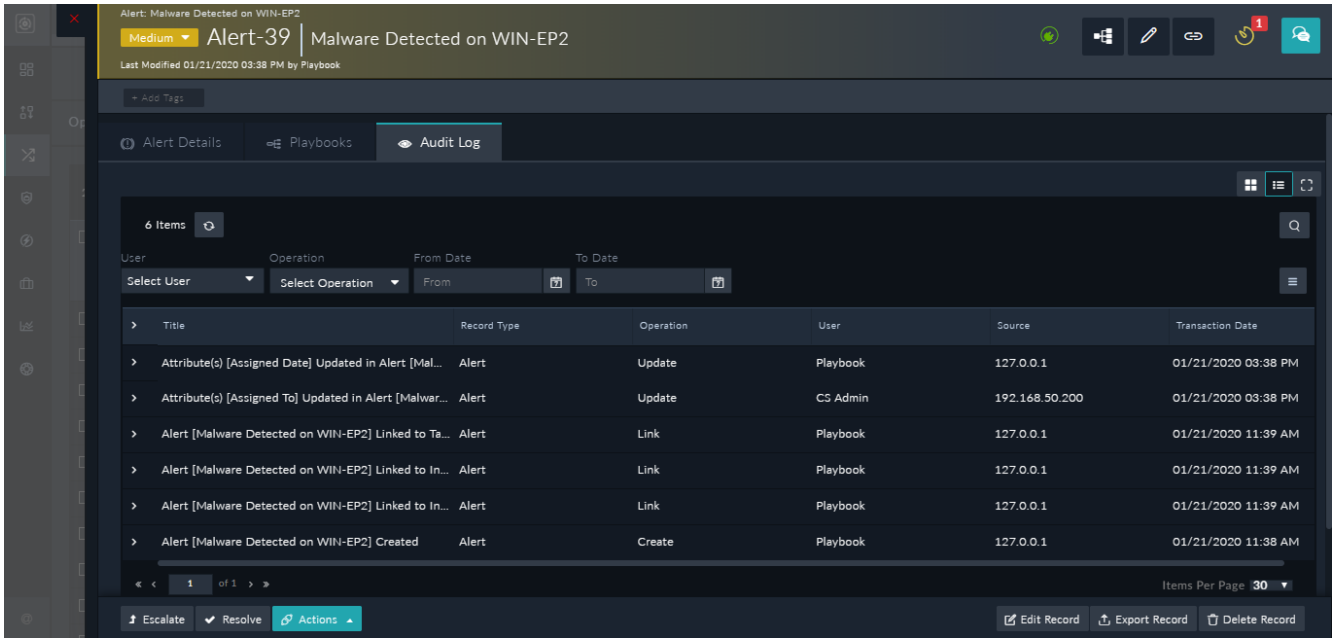


You can toggle between the expanded and collapsed view of the audit log tab, using the **Full-screen Mode** icon. To move to a full screen view of the audit log, click the **Full-screen Mode** icon, which opens the audit log in the full screen as shown in the following image:



To exit the full screen, press ESC.

You can toggle between the timeline view and grid view in the Audit Log tab. The grid view in the detailed view of a record appears as shown in the following image:

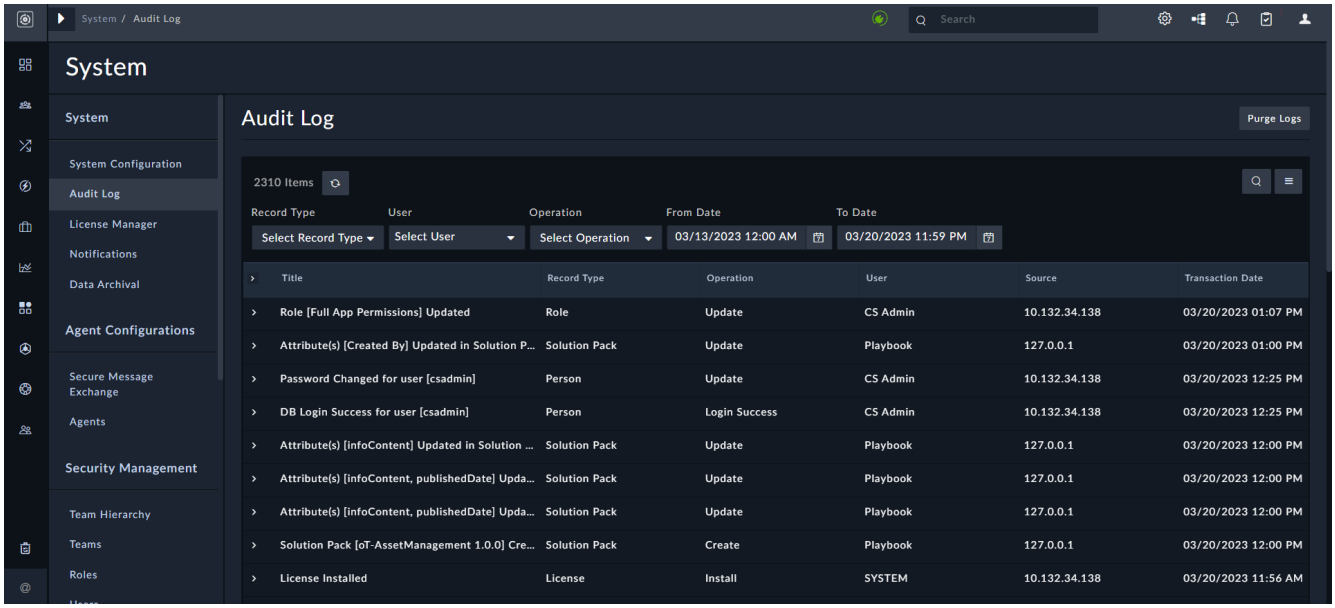


The grid view also displays the same information as the audit log, and you can also perform the same operations here as you can perform in case of audit logs. For more information, see the Viewing Audit Log section.

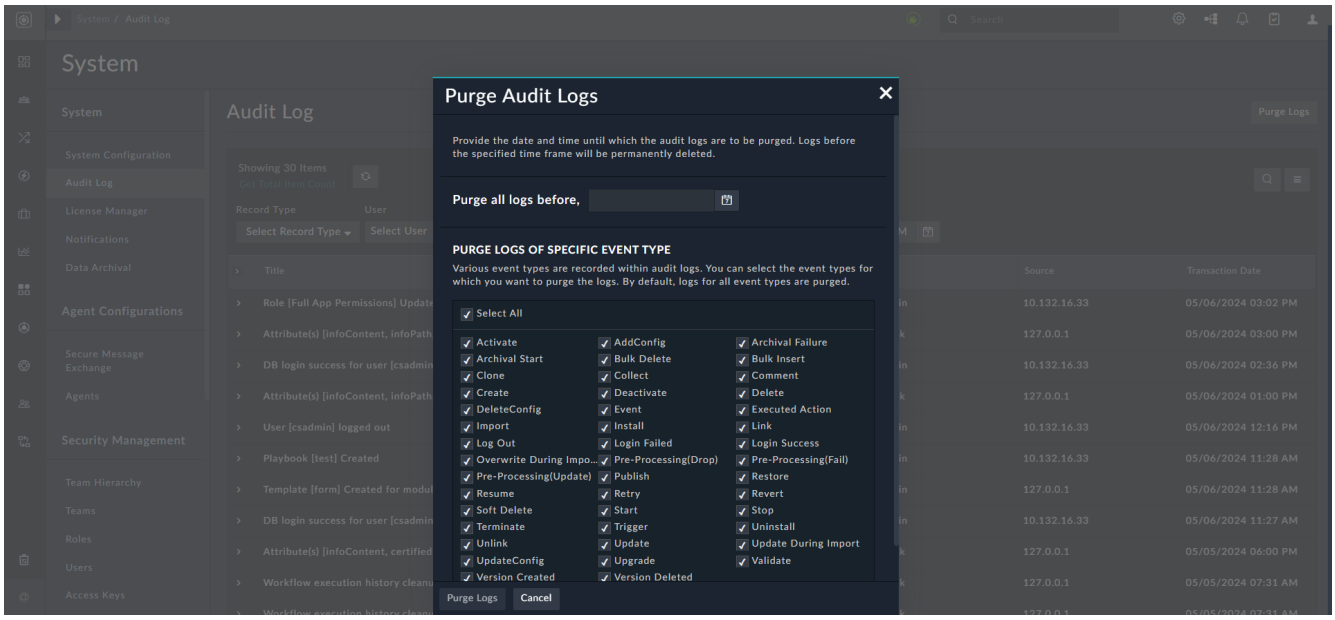
Purging Audit Logs

You can purge Audit Logs using the **Purge Logs** button on the top-right of the `Audit Log` page. Purging audit logs allows you to permanently delete old audit logs that you do not require and frees up space on your FortiSOAR instance. You can also schedule purging, on a global level, for both audit logs and executed playbook logs. For information on scheduling Audit Logs and Executed Playbook Logs, see Purging audit logs, executed playbook logs, and recycle bin records, and reclaiming unused disk space.

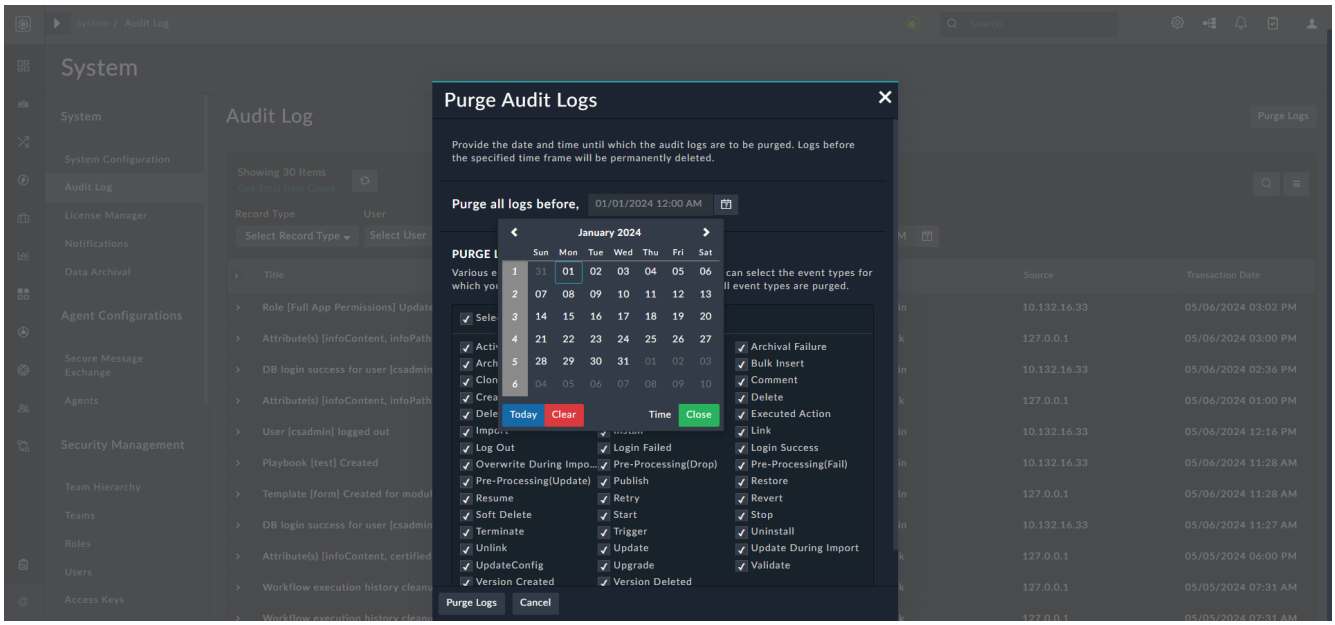
To purge Audit Logs, you must be assigned a role that has a minimum of `Read` permission on the `Security` module and `Delete` permissions on the `Audit Log Activies` module.



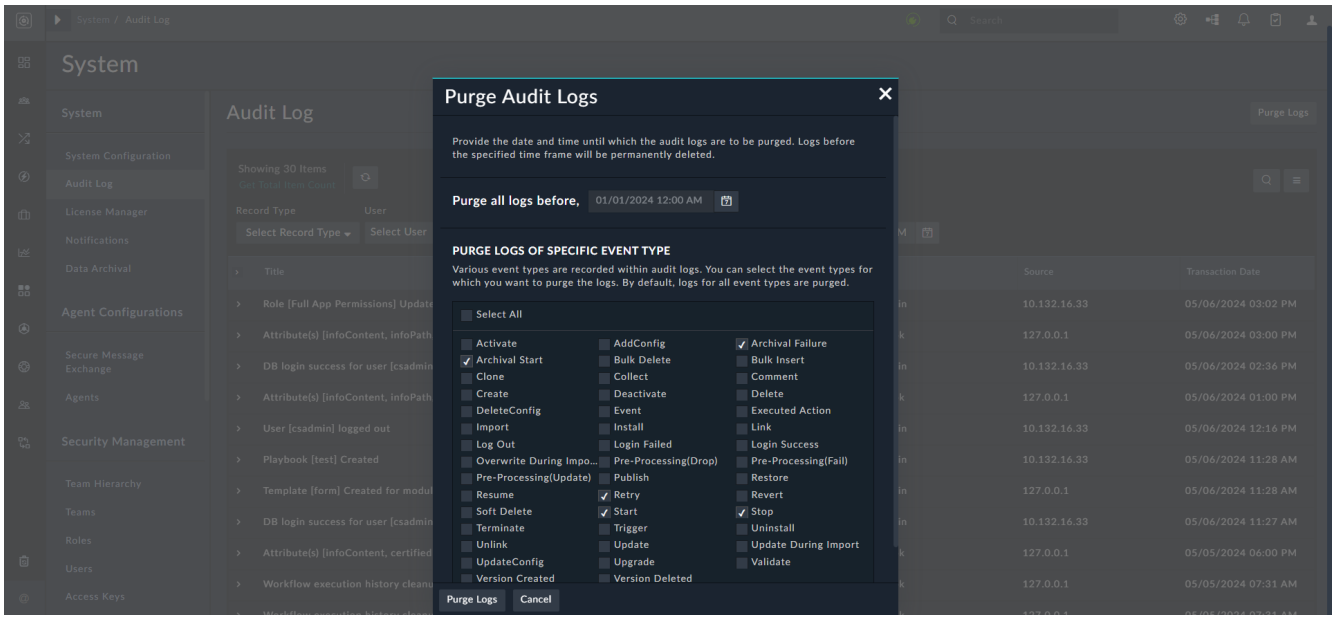
To purge Audit Logs, click the **Purge Logs** button on the `Audit Log` page, which displays the `Purge Audit Logs` dialog:



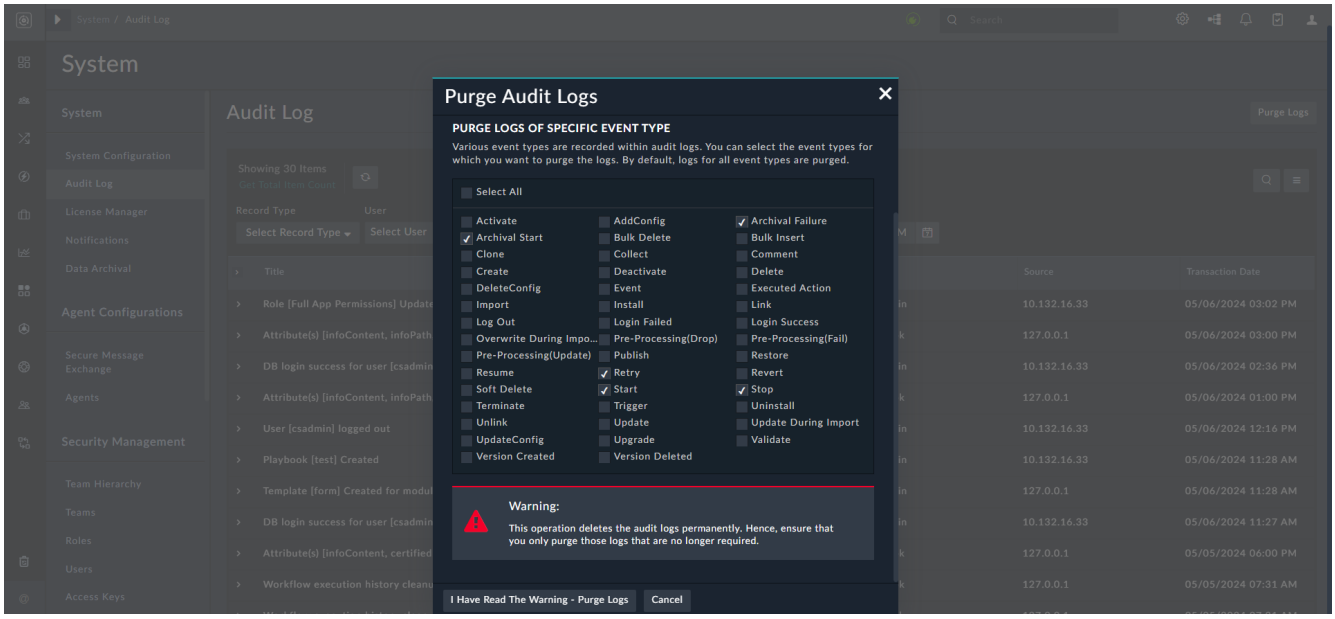
In the **Purge all logs before,** field, select the time frame (using the calendar widget) before which you want to clear all the audit logs. For example, if you want to clear all audit logs before `January 1st, 2024, 12:00 AM`, then select this date and time using the calendar widget.



By default, logs of all events are purged and the **Select All** checkbox is selected. However, if you want to purge just a few event types such as "Archival Start", "Archival Failure", "Retry", "Stop", and "Start", clear the **Select All** checkbox, and then select only these event types, as shown in the following image:



To purge the logs, click the **Purge Logs** button, which displays a warning as shown in the following image:



Click the **I Have Read the warning - Purge Logs** to continue the purging process.

Troubleshooting issues with Audit Logs

For troubleshooting audit log issues use the auditlog.log located at `/var/log/cyops/cyops-gateway/auditlog.log`.

Audit Logs not processed when file size exceeds 64MB

When audit logs exceed 64MB, the gateway logs contain an error such as: 'Caused by: java.lang.IllegalStateException: Message body is too large (65040321), maximum configured size is 62000000. See ConnectionFactory#setMaxInboundMessageBodySize if you need to increase the limit.'

Resolution

To resolve this issue, follow the steps below:

1. SSH to your FortiSOAR instance with 'root' access.
2. Open the `/opt/cyops/configs/cyops-gateway/gateway.properties` file.
3. Append the `max_inbound_message_body_size` variable to the `gateway.properties` file. Set its value to a limit higher than the default 64MB. For example, to increase the limit to 128MB:
   `max_inbound_message_body_size=128`
4. Restart the tomcat service:
   `systemctl restart cyops-tomcat`

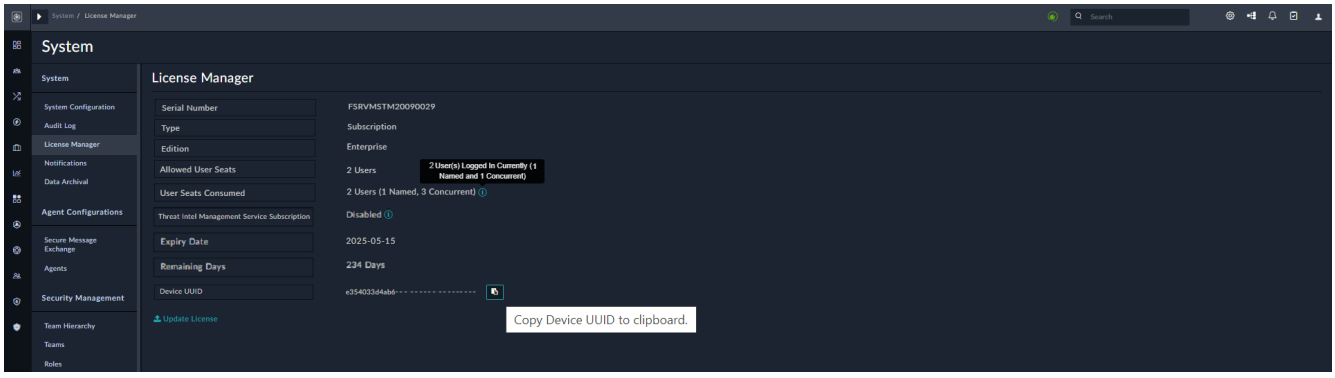
License Manager

FortiSOAR enforces licensing using the License Manager. The License Manager restricts the usage of FortiSOAR by specifying the following:

- The maximum number of active named users in FortiSOAR at any point in time.
- The type and edition of the license.
- The expiry date of the license.
- The restrictions on usage of Threat Intelligent Management (TIM) features.

For details of the FortiSOAR licensing process, including deploying your FortiSOAR license for the first time, see the *Licensing FortiSOAR* chapter in the "Deployment Guide."

Click **Settings > License Manager** to open the `License Manager` page as shown in the following image:



You can use the `License Manager` page to view your license details and to update your license. FortiSOAR displays a message about the expiration of your license 15 days prior to the date your license is going to expire. If you license type is **Evaluation** or **Perpetual**, then you must update your license within 15 days, if you want to

keep using FortiSOAR. To update your license, click **Update License** and either drag-and-drop your updated license or click and browse to the location where your license file is located, then select the file and click **Open**. If your license type is **Subscription**, you must renew your subscription.

For more information on licensing and for details about the various parameters on the `License Manager` page, see the *Licensing FortiSOAR* chapter in the "Deployment Guide."

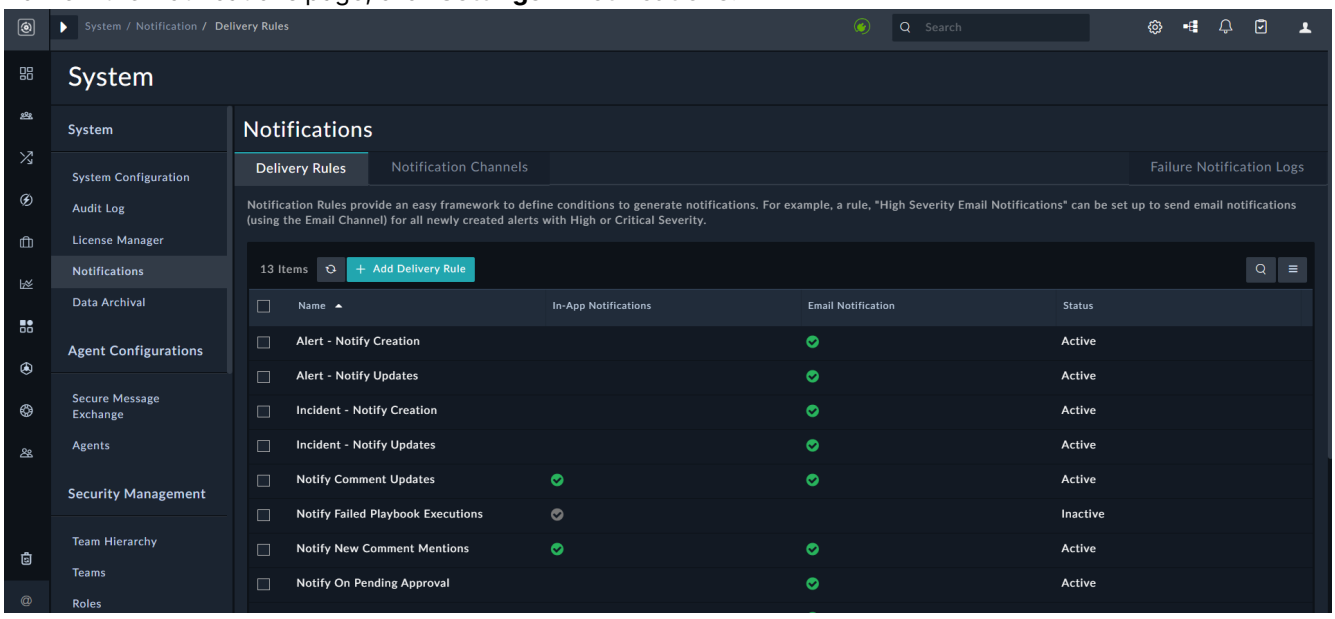
Notifications

FortiSOAR is built on the premise of humans being in the loop for effective orchestrated investigation; therefore, ensuring timely action-centric (or informational) notifications are generated becomes vital in achieving this goal. To achieve this goal, in release 7.2.0, FortiSOARintroduces a common framework for diverse notifications, such as email notifications, UI notifications from various services (like alerts/incidents/tasks assignments), Comments @mentions, workflow failures, etc. This common notification framework makes it possible for users to have complete control of the setup and consumption of notifications, i.e., how and when they want to consume notifications, and what notifications they want to consume.

From release 7.2.0 onwards, there is a single 'Notifications' page using which you can enable, disable, or edit the notifications settings and add, edit or deactivate notification channels.

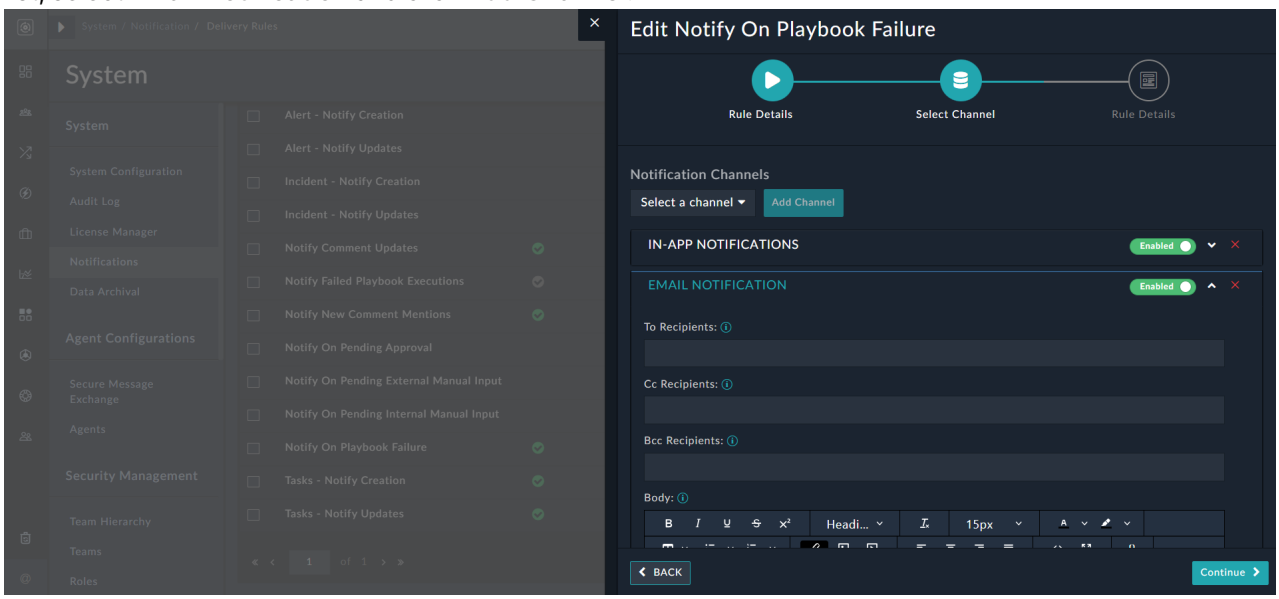
To view the 'Notifications' page, you must be assigned `Read` permission on the `Notifications Rules` module and `Update` permission on the `Application` module. Similarly, to perform actions such as configuring delivery rules, or notification channels, you must be assigned appropriate `CRUD` permissions on the `Notifications Rules` module.

To view the Notifications page, click **Settings** > **Notifications**:



On the Notifications page, you will see the **Delivery Rules**, **Notification Channels**, and **Failure Notification Logs** tabs:

- The `Delivery Rules` page contains default rules that are included by default in FortiSOAR and which have already been created for some common use cases, such as notifications to be generated when an approval request is created - 'Pending Approval Notification', or when comments that have @mentions are updated or added, or for playbook failures, etc. It also specifies the channel used for sending the notifications. Some examples of jinja expressions that can be used while creating or adding notifications are included in the Usage examples of Jinja Expressions in Notifications topic.
  Note: To edit an existing rule, you can click the row of the rule to open the edit <named of the rule> page, make the required changes, and then click **Update**.
  For example, if you want playbook failure notifications to be sent using the email channel in addition to the default In App notifications, click the **Notify On Playbook Failure** rule to display the `Edit Notify On Playbook Failure` dialog, then in the `Select Channel` page, from the **Notification Channels** drop-down list, select **Email Notification** and click **Add Channel**:



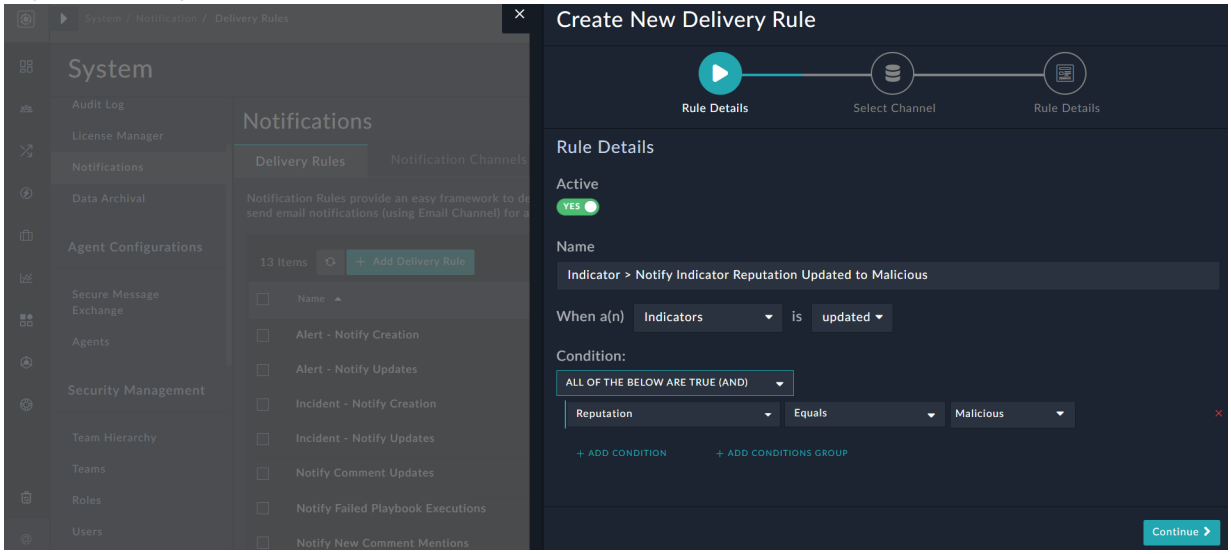
  Note: Ensure that you have added appropriate notification details to the Email channel before you add the Email channel for playbook failures. For details on how to set up a notification channel, see the Setting up Notifications Channels topic.
- The `Notification Channels` page lists the channels that can be used to send notifications. The default channels are In-App Notifications and Email Notifications.
- The `Failure Notification Logs` page contains a list of notification failures with their respective error messages using which you can debug the notification failure. Once you have fixed the issues in the notification, you can select that particular log entry and click **Retry Notification**.
  To purge failure notification logs, click the **Purge Logs** button on the `Failure Notification Logs` page, which displays the **Purge Failed Notification Logs** dialog. In this dialog, select the time frame (using the calendar widget) before which you want to clear all the failure notification logs.

If you have upgraded to release 7.2.0 or later from a release prior to 7.2.0, then users will receive two emails for a single notification. This is because the upgraded FortiSOAR system would have both the new notification rules and the system playbooks that triggered notifications prior to 7.2.0. To resolve this issue, the earlier system playbooks that used to send the notifications emails must be deactivated and the notification rules must be customized to suit your requirements.

Adding Delivery Rules

You can add your own custom delivery rules based on which notifications will be created and delivered as follows:

1. Click **Settings** > **Notifications > Delivery Rules**.
2. On the `Delivery Rules` page, click **Add Delivery Rule**, to open the 'Create New Delivery Rule' wizard.
3. On the `Rule Details` screen, you can define the rules for generating the notifications:
   a. To create the rule in the 'Active' state, leave the Active toggle button as **YES.**
   b. In the **Name** field, add a name that describes the purpose of the rule. For example, if you want to generate a notification if the reputation of an indicator is updated to malicious, then add the name as `Indicator > Notify Indicator Reputation Updated to Malicious`
   c. Add the rule for generating notifications:
   `When an Indicator is updated`
   Then add the additional conditions for generating the notification. In our example, we want to generate a notification only if the reputation of the indicator is updated to 'Malicious' is added. Therefore, in the `Condition` section, choose the logical operator **AND** or **OR** and then add the condition as:
   `Reputation Equals Malicious`

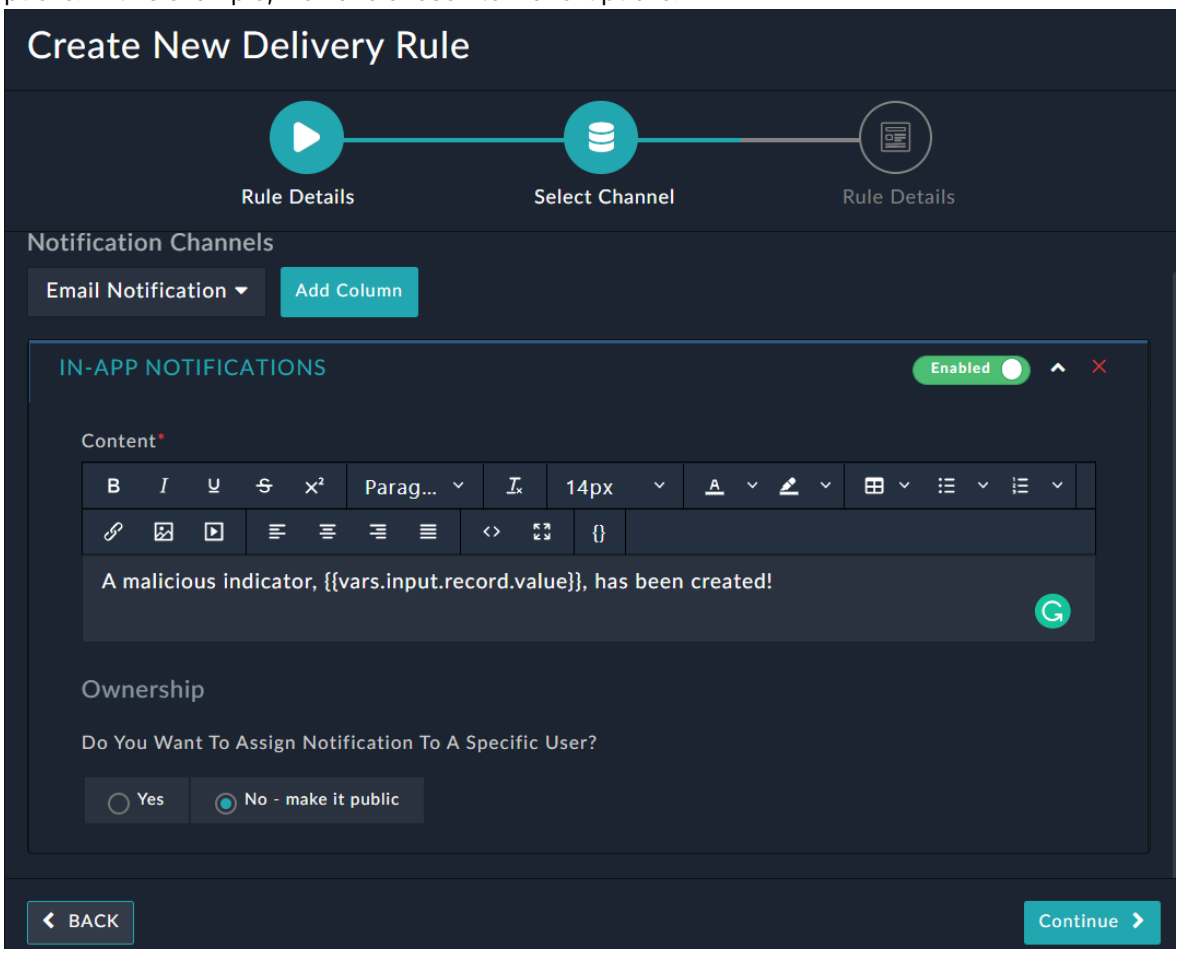


   You can add additional conditions as per your requirement.
   NOTE: The 'Playbook Status' filter includes a list of all playbook statuses, allowing you to build notifications based on playbook statuses, such as sending notifications when playbooks are in the "Finished with Error" state.
   d. Once you have completed adding the details for the rule, click **Continue**.
4. On the `Select Channel` screen using which users can consume the notifications.
   Note: You can choose only those channels that have been created. In-App Notifications and Email Notification are channels that are set up by default. For more information, see the Setting up Notifications Channels topic.
   a. From the **Notification Channels** drop-down list, select the channel using which you want to deliver the notifications. If you want to deliver notifications using the **Notifications** icon present on the top-right corner in FortiSOAR, select **In-App Notifications** and click **Add Column**. You can further configure the settings for this notification:

i. To enable or disable this notification, you can toggle the **Enabled** button.

ii. In the **Content** field, add the content that you want to display as part of the notification. For example:
`A malicious indicator, {{vars.input.record.value}}, has been created!`

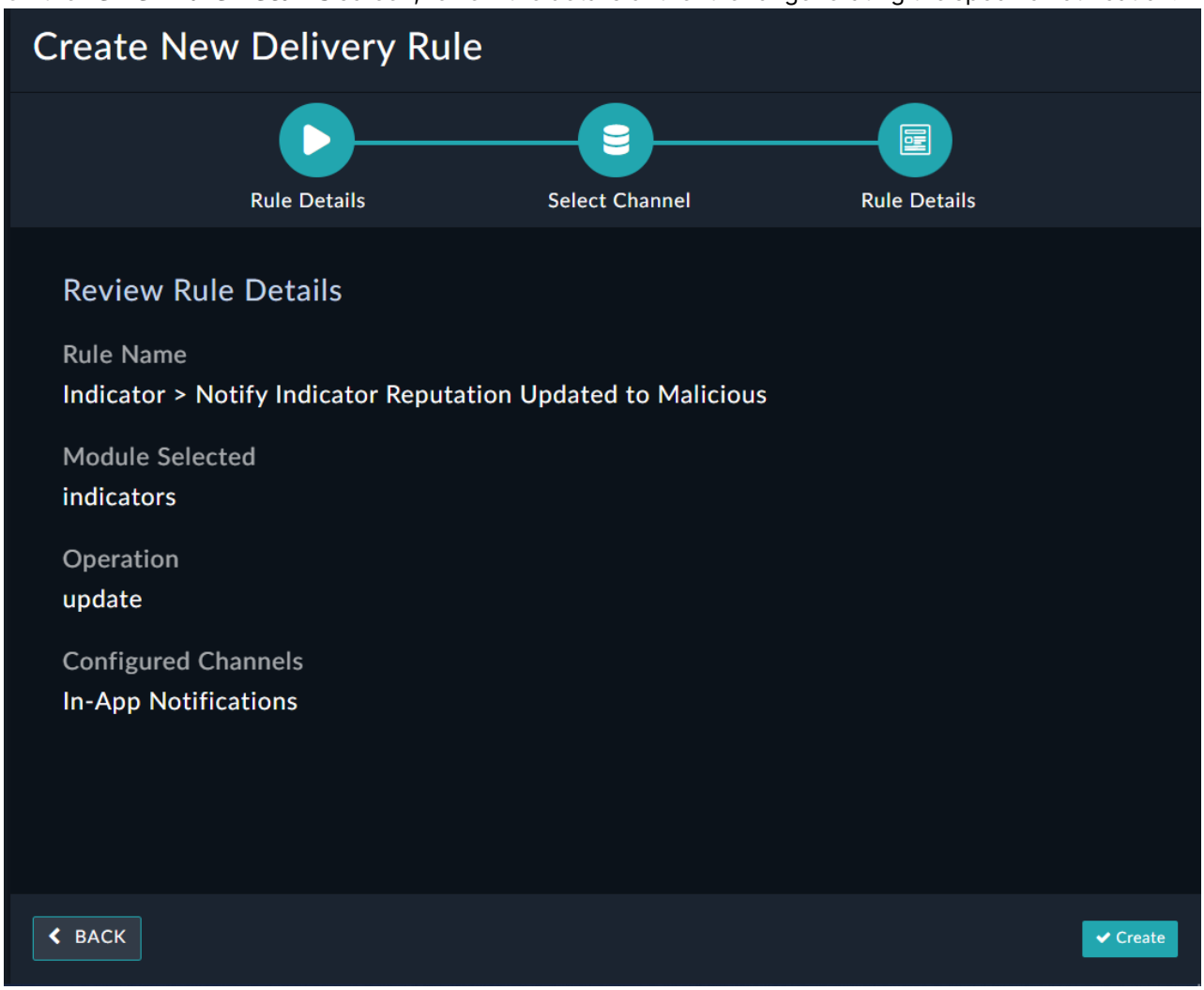
iii. In the `Ownership` section, you can choose to assign this notification to a specific user or to make it public. In this example, we have chosen to make it public.



Similarly, you can choose to deliver notifications from other notification channels, such as **Email Notification**.

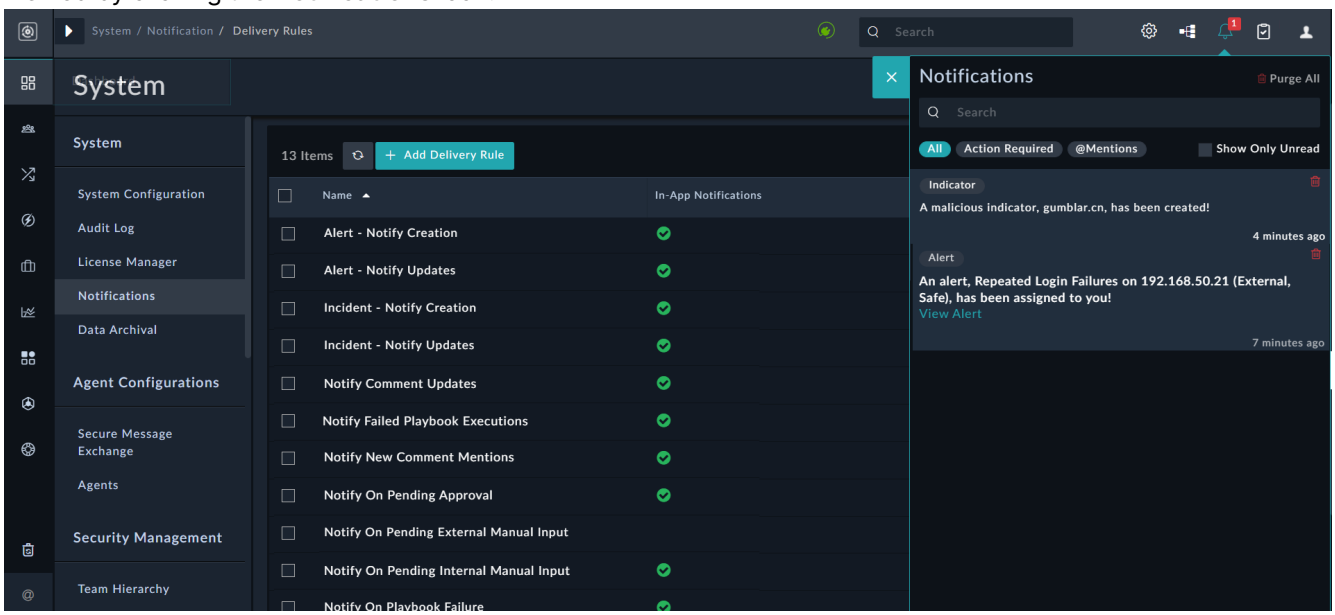
b. Once you have completed setting up the notification channels, click **Continue**.

5. On the `Review Rule Details` screen, review the details of the rule for generating the specific notification:



Once you are satisfied, click the **Create** button.

You will observe that whenever a malicious indicator is added, a notification will be generated, that can be viewed by clicking the **Notifications** icon:



Usage examples of Jinja Expressions in Notifications

Following are some examples of jinja expressions that can be used while adding or customizing notifications:

Jinja expressions to get records details:

- To get the name of a record:
  `{{ vars.input.record.name }}`
- To get the severity of a record:
  `{{ (vars.input.record.severity | fromIRI).itemValue }}`
- To get the URL to access a record:
  ```
  {% set idParts = vars.input.record['@id'].split('/') %}
  {{globalVars.Server_fqhn}}{{ '/modules/alerts/' + (idParts | last) }}
  ```

Jinja expressions to get the email address of the record owner based on the notification channels:

- To fetch the record owner's email address while customizing the "Email" channel when you are adding or customizing a rule:
  `{{vars.input.record.assignedTo}}`
- To fetch the record owner's email address while customizing "any" channel apart from the "Email" channel when you are adding or customizing a rule:
  `{{(vars.input.record.assignedTo | fromIRI).email}}`

Jinja expressions to customize "Manual Inputs":

- To fetch the title of the manual input:
  `{{vars.input.record.input.schema.title}}`
- To access the parent record of the record that contains the manual input trigger:
  `{{ vars.input.record.record }}`
- To get the 'type' of record that contains the manual input trigger:
  `{{(vars.input.record.record | fromIRI).type.itemValue }}`

- To print the IRI of the parent record of the record that contains the manual input trigger:
  `{% if vars.input.record.record != '' %} Record Link - {{vars.input.record.record.split('/')[-2]}}{% endif %}`
- To fetch the email address of the owner in the case of an external or unauthenticated manual input:
  `{{vars.input.record.owner_details.emailRecipients}}`

Jinja expressions to get comment record details:

- To get the 'first name' of the user who added the comment:
  `{{(vars.input.record.createUser | fromIRI).firstname }}`
- To get the 'last name' of the user who added the comment:
  `{{(vars.input.record.createUser | fromIRI).lastname }}`

Setting up Notification Channels

Notification channels define the mechanism using which notifications are delivered to users. By default, the 'In-App Notifications' and 'Email Notification' (using the SMTP integration) channels are set up. Users cannot delete these default notification channels and they also cannot edit the In-App Notification channel.

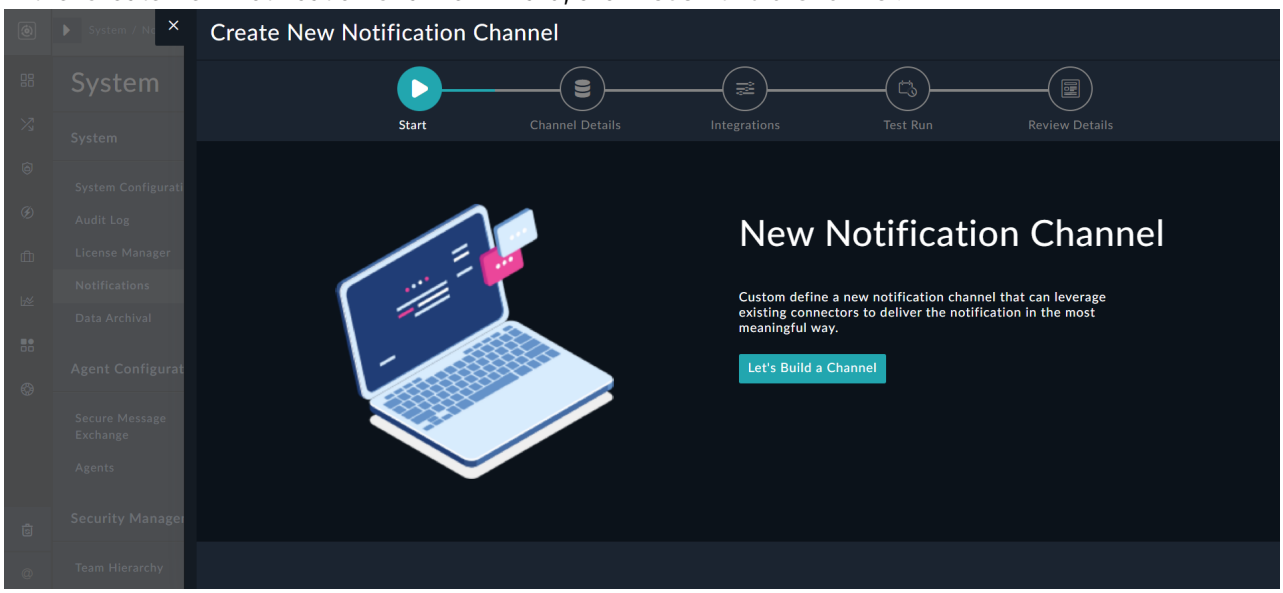
You can set up other channels to deliver notifications using desktop applications, slack, microsoft teams, mobile applications, etc. You could also edit the existing 'Email Notification' channel to use Exchange or any other email server to send email notifications.

To deliver notifications to users using notification channels, you first have to ensure that you have configured the integration (connector) that will be used for the channel. For example, to use **Exchange** as the 'Email Notification' channel (SMTP is pre-configured), ensure that you have configured the **Exchange** connector and its 'Health Check' displays 'Available'. 'In-App Notifications' do not require any configuration.

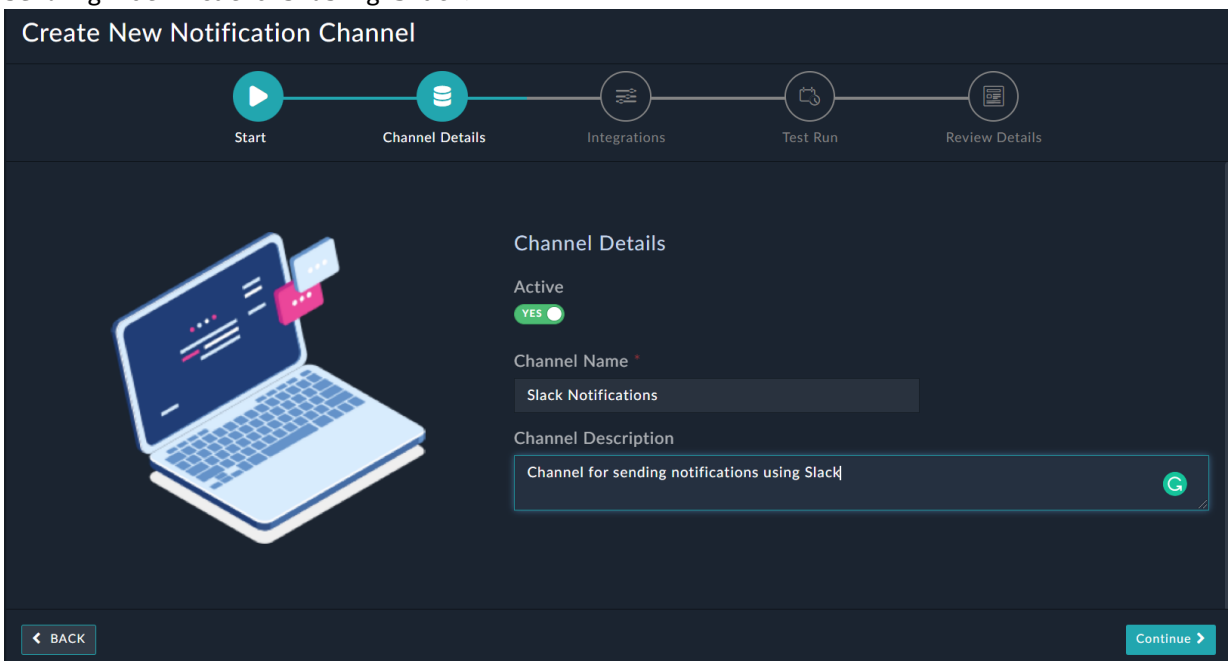
To set up a new channel for delivering notifications, for example, a Slack channel, do the following:

1. Ensure that you have installed and configured the Slack connector and its 'Health Check' displays 'Available'. Also, ensure that you have set up a channel in Slack for receiving the notifications.
   For information on connectors, see the *Introduction to connectors* chapter in the 'Connectors Guide.'
2. Click **Settings** > **Notifications> Notification Channels**.
3. On the `Notification Channels` page, click **Add**, to open the 'Create New Notification Channel' wizard.

4. In the 'Create New Notification Channel' wizard, click **Let's Build a Channel**.



5. On the `Channel Details` screen, specify the details of the notification channel:
   a. To create the notification channel in the 'Active' state, leave the Active toggle button as **YES.**
   b. In the **Name** field, add the name of the channel, for example, `Slack Notifications`.
   c. In the **Channel Description** field, add a brief description of the channel, for example, `Channel for sending notifications using Slack`.



   d. Once you have completed adding the details for the channel, click **Continue**.
6. On the `Integrations` screen, configure the connector using which you want to send notifications to the users:

a. From the **Choose Suitable Connector** drop-down list, select **Slack**.

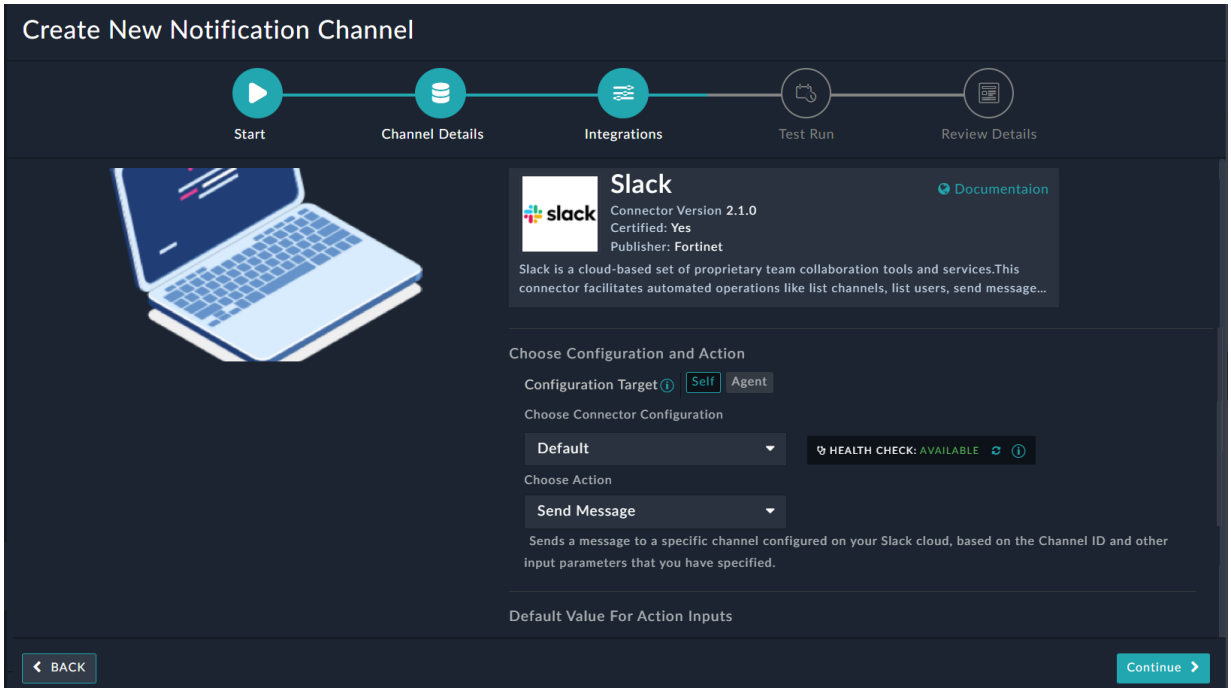
b. In the `Choose Configuration and Action` section, configure the following:

   i. Toggle Configuration Target to Agent if you want to configure the connector for a remote FSR Agent, or retain the configuration for the current FortiSOAR node, i.e., Self (default). For more information on agents and how to run remote actions using FSR Agents, see the Segmented Network support in FortiSOAR chapter.

   ii. From the **Choose Connector Configuration** drop-down list, select the connector configuration that you want to use to send notifications and ensure that the 'Health Check' displays **Available**. For our example, select **Default**.

   iii. From the **Choose Action** drop-down list, select the action that you want to perform using the Slack connector. For our example, select **Send Message**. The Send Message action will send the notifications to the specific channel that you have set up on your Slack Cloud.

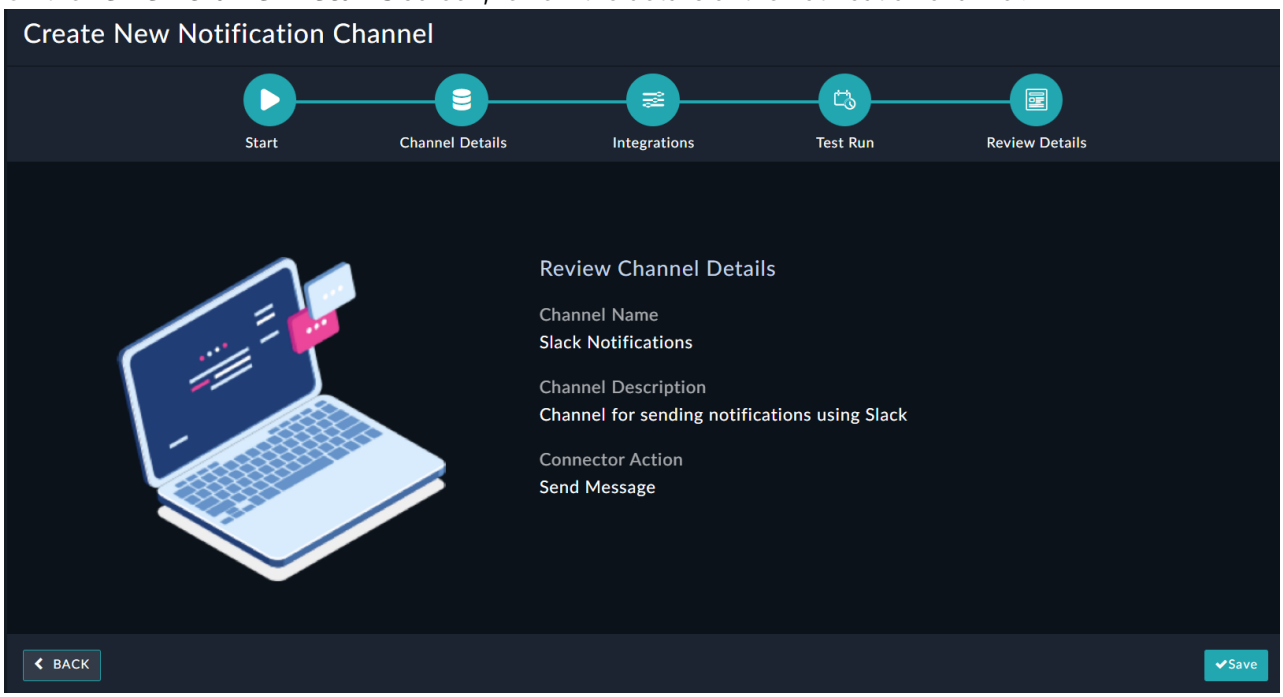
c. In the `Default Value for Action Inputs` section, enter the ID of the channel (or user) that you have created for receiving notifications from FortiSOAR, and the default message that you want to include for notifications.



d. Once you have completed setting up the integration, click **Continue**.

7. On the `Test Run` screen, you can test your integration by providing sample inputs and clicking **Trigger Test Notification**.
   Clicking Trigger Test Notification sends a live notification to the Slack channel that you have configured for receiving notifications.
   To move to the next screen, click **Continue**.

8. On the `Review Channel Details` screen, review the details of the notification channel:



Once you are satisfied, click the **Save** button.

Working with Delivery Rules and Notification Channels

The default notification channel for emails is SMTP; however, you can choose to change this to Exchange or any other email service provider. If you set, for example, Exchange as the default notification channel, then all notifications such as workflow failures, creation and updates of alerts, incidents, etc, actions pending for some user actions, etc., will all be delivered using Exchange. In such a case, you must do the following:

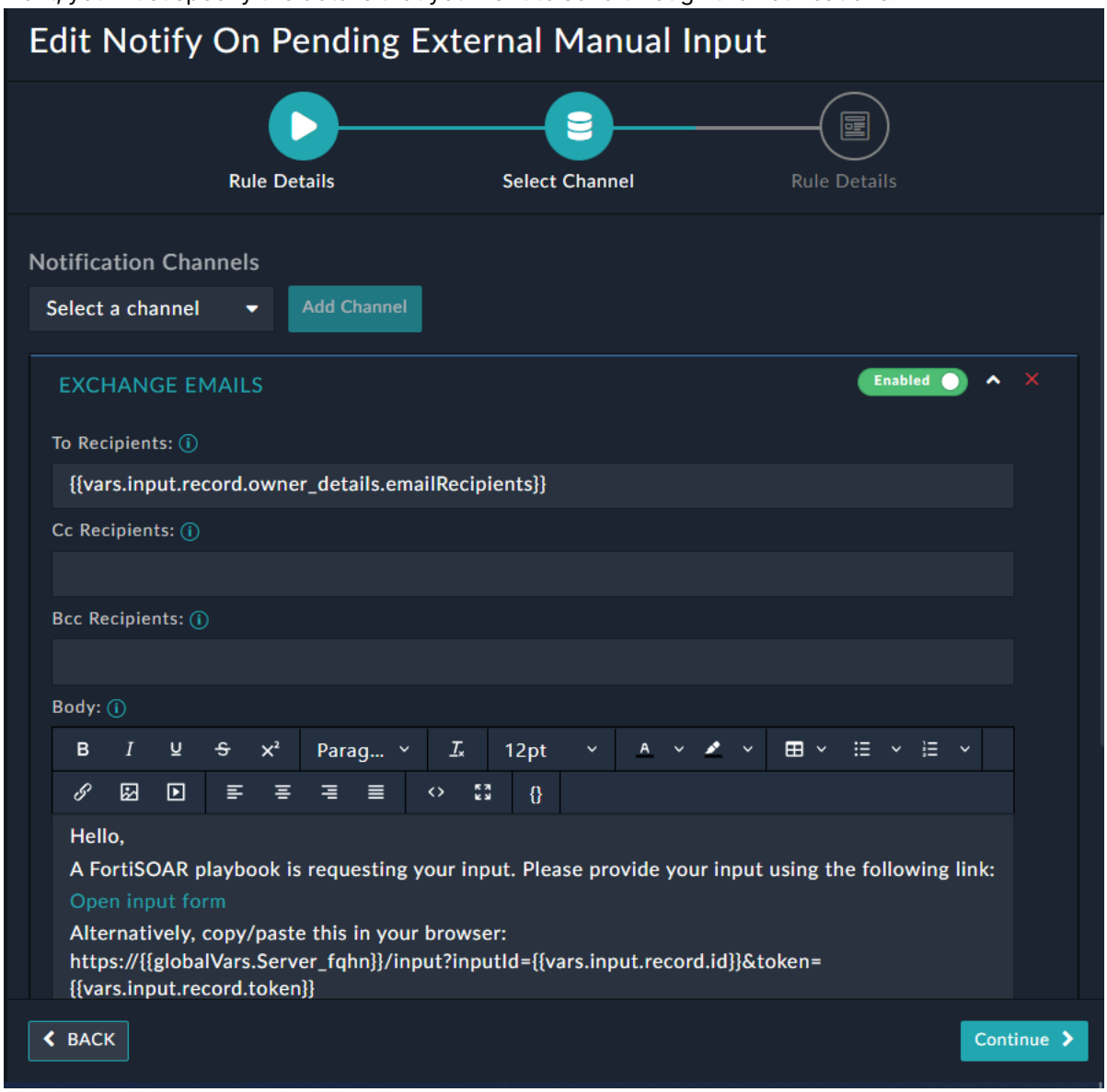
1. Configure the **Exchange** connector and its 'Health Check' displays 'Available'.
2. Navigate to **Notifications** > **Notifications Channel** and click the **Email Notification** row. In the **Update Email Notification Channel** wizard, edit the email notification channel to use the Exchange connector instead of the SMTP connector. The Update Email Notification Channel wizard is exactly like the Create New Notification Channel wizard, steps for which are described in the Setting Up Notifications Channels topic.

If however, you want to use SMTP as the default notification channel and use Exchange only for specific use cases, for example, to use Exchange in cases where you want to request decisions or other inputs from either FortiSOAR or non-FortiSOAR users; then, or you want Approvals notifications sent using Exchange, then you have to update the respective delivery rules, i.e., Notify On Pending External Manual Input, Notify On Pending Internal Manual Input, or Notify On Pending Approval Notification. For our example, we will use exchange for sending pending external manual input notifications:

1. Ensure that you have configured the Exchange connector and set up a new notification channel using the Exchange connector. For steps on setting up a notification channel, see Setting Up Notifications Channels.
2. Update the delivery rule to use Exchange instead of SMTP for delivering pending external manual input notifications:

a. Navigate to **Notifications** > **Delivery Rules** and click the **Pending External Manual Input Notification** row.

b. In the **Edit Pending External Manual Input Notification** wizard, leave the **Rules Details** screen unchanged and click Continue. On the **Select Channel** screen, remove **Email Notification**, which is the default SMTP notification channel, and add the notification channel that you have set up for Exchange. Next, you must specify the details that you want to send through the notifications:



On this screen, you can also customize the email content, including adding custom Jinja input in the email body that gets sent to the configured email addresses. You can also specify a list of record IRIs or file IDs for attachments that you want to add to the email in the **Attachment IRI List** field.
Once you have specified all the details, click **Continue**.

c. On the **Review Rule Details** screen, review the details of the rule you have set up, and if satisfied click **Update**.
This updates the Pending External Manual Input Notification rule to send pending external manual input notifications using Exchange.

Disabling a Delivery Rule

If you have upgraded your FortiSOAR system to release 7.3.x from a release prior to 7.3.0, you can disable the `Notify On Pending Manual Input` rule, if you want to disable the manual input notification from the 'Notifications Panel'; else you will receive the manual input notification in both the 'Notifications Panel' and the 'Pending Tasks'. In the case of a fresh installation of FortiSOAR 7.3.x, notifications for manual input are visible on only the **Manual Input** tab in the 'Pending Tasks' panel.

In the case of a fresh installation of FortiSOAR 7.4.0 (or later), by default, the `Notify On Pending Manual Input` rule is removed, and if you upgrade to FortiSOAR 7.4.0, then by default, the `Notify On Pending Manual Input` rule is in the 'Inactive' state.

To disable the `Notify On Pending Manual Input` rule do the following:

1. Click **Settings** > **Notifications** > **Delivery Rules**.
2. On the `Delivery Rules` page, click the `Notify On Pending Manual Input` rule.
3. On the **Rules Details** screen, toggle **Active** to **No**.
4. Click **Continue** on the **Select Channel** screen, and then click **Update** on the **Rules Details** screen.

You can disable any rule on your FortiSOAR system using these steps.

Modifying the 'Notify On Pending External Manual Input' delivery rule post-upgrade to release 7.3.0 or later

From FortiSOAR 7.3.0 onwards, you can choose to run unauthenticated manual inputs in segmented networks using FSR agents. To achieve this, the 'Notify On Pending External Manual Input' delivery rule has been updated in the case of fresh installations of FortiSOAR 7.3.0. However, if you have upgraded your FortiSOAR instance to release 7.3.0 or later from a release prior to 7.3.0, you must modify the 'Notify On Pending External Manual Input' rule to allow the running of manual inputs on FSR agents as follows:

1. Open the `Notifications` page by clicking **Settings** > **Notifications**.
2. On the **Delivery Rules** tab, click the **Notify On Pending External Manual Input** delivery rule row to edit this rule.
3. On the `Edit Notify On Pending External Manual Input` dialog, click **Continue** on the `Rule Details` screen.
4. On the `Select Channel` screen, expand the '**Email**' channel or any other channel used to send the notifications.
5. In the `Body` section, do the following:
   a. Select the **Open Input Form** text, and then click the **Insert/Edit Link** button to edit the manual input link.
   b. In the `Insert/Edit Link` dialog in the **URL** field, enter the following URL:
   `https://{{vars.input.record.server_fqhn | ternary(vars.input.record.server_fqhn, globalVars.Server_fqhn)}}/{{vars.input.record.agent_id | ternary('', 'input')}}?inputId={{vars.input.record.id}}&token={{vars.input.record.token}}`
   and then click **Save**.

c. Replace the existing URL text that is present in the **Body** section after the 'Alternatively, copy/paste this in your browser:' sentence with the following URL:
`https://{{vars.input.record.server_fqhn | ternary(vars.input.record.server_fqhn, globalVars.Server_fqhn)}}/{{vars.input.record.agent_id | ternary('', 'input')}}?inputId={{vars.input.record.id}}&token={{vars.input.record.token}}`
and then click **Save**.

6. On the Select Channel screen, click **Continue**.
7. On the Rule Details screen, review the rule and then click **Update** to update the rule to send external manual input notification on both your FortiSOAR instance as well as your FSR agent.

Purging of notifications

By default FortiSOAR, runs a system schedule, every day at midnight (00:00 hrs) to purge all system notifications, both read and unread, that are older than 14 days. The cron expression for this system schedule is present in the /opt/cyops-workflow/sealab/sealab/config.ini file as follows:
`PURGE_NOTIFICATION_SCHEDULE: {'minute': '0', 'hour': '0', 'day_of_week': '*', 'day_of_month': '*', 'month_of_year': '*'}`
NOTE: The days of the week are represented by integers from 0 to 6, with 0 representing Sunday and 6 representing Saturday. Therefore the day_of_week property accepts * or [0-6] characters.

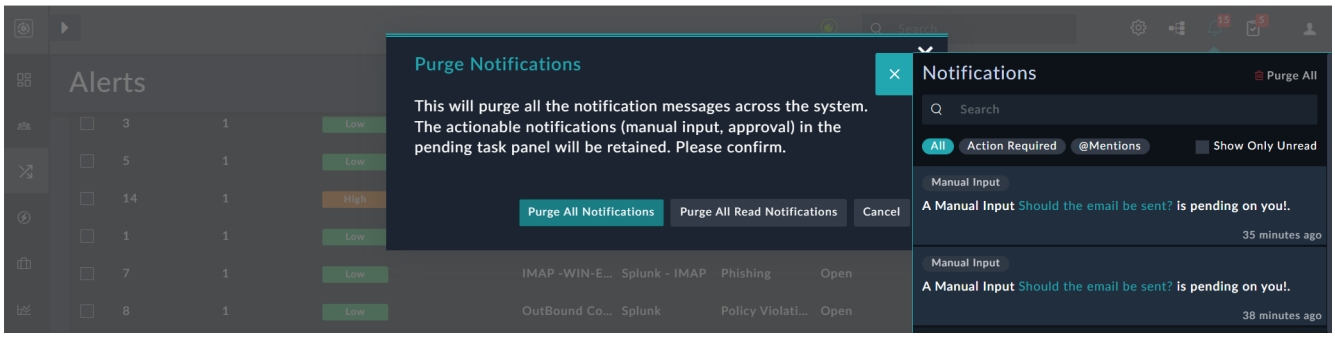
You can update this cron expression if you want to change the default schedule timing window of 12 am, and then run the following command:
`$ sudo -u nginx /opt/cyops-workflow/.env/bin/python /opt/cyops-workflow/sealab/manage.py default_schedules`
Then restart the services using the following command:
`systemctl restart celeryd celerybeatd fsr-workflow`

It is highly recommended that you do not update the system schedule for purging notifications.

Similarly, the KEEP_SYSTEM_NOTIFICATION_DAYS parameter in the config.ini sets the default number of days that read system notifications are retained in the system, which by default is set to 7 days as follows:
`KEEP_SYSTEM_NOTIFICATION_DAYS = 7`
You can change this setting as per your requirements. After you have completed making changes to this setting, you must restart the default schedules using the following command:
`$ sudo -u nginx /opt/cyops-workflow/.env/bin/python /opt/cyops-workflow/sealab/manage.py default_schedules`
Next, you must restart the services using the following command:
`systemctl restart celeryd celerybeatd fsr-workflow`

Users with a minimum of Update permissions on the Security Module also use the FortiSOAR UI to purge notifications. To purge notifications, click the Notifications icon on the top-right corner of the FortiSOAR UI to display the Notifications Panel. Then click the **Purge All** icon to display the Purge Notifications dialog. Click **Purge All Notifications** to delete all notifications or click **Purge All Read Notifications** to delete all read

notifications:



Data Archival

The volume of data ingested in SOAR platforms grows exponentially as SOCs across the globe automate most of their procedures using SOAR solutions. Over years of usage, as the volume of data grows, which can adversely affect the performance of a SOAR platform as follows:

- Database queries become slower and more and more resource-intensive.
- UI becomes very sluggish as more time is needed to fetch the required data.
- More disk space for storage on the primary system.

Historical data is not required for day-to-day investigation. However, it cannot be discarded completely as organizations need it for audit and compliance reviews, and also for occasional references.

To solve the mentioned issues and to retain historical data for the long term by preserving it in your data lake, FortiSOAR provides a way to archive data. The data archival solution in FortiSOAR provides the following advantages:

- Improves the overall performance of your FortiSOAR instance by helping to lessen the primary data volume.
- Ability to move historical data that is not accessed frequently to less expensive storage.
- Ability to configure timeframes for data to be kept at primary and for data to be archived.
- Ability to search the archive for certain specific records or record attributes.
- Ability to archive certain specific types of records. For example, critical and closed alerts and incidents.

If you are upgrading your FortiSOAR instance from a release prior to 7.2.0 to a 7.2.0 or later release, then ensure that your externalized PostgreSQL database does not contain a database with the name 'data archival', else there might be conflicts with the data archival feature.

The following modules cannot be archived:

- Agents
- Appliances
- Approvals
- People
- Routers

- Saved Reports
- Tenants

Methods of Setting Up Data Archival

- **External Database** (Recommended): Use an external database for data archival purposes.
- **Internal Database**: Use an internal database for data archival purposes; this can be used for testing purposes.
- **Syslog Forwarding**: You can also archive data to an external Syslog server. You can choose to enable Syslog forwarding in addition to archiving data on databases or can set it up as a sole data archival destination.
  Important: Data that you archive using only Syslog Forwarding cannot be searched using FortiSOAR.

You can use a combination of methods for archiving your data. For example, you can choose to archive data in an external Syslog server, in addition to storing that in an internal or external database. If you choose to archive data in both the external Syslog server and a database, then any record that is archived will be forwarded to the Syslog server and will also be stored as a row in the database. You can use such a combination if you want an archival strategy in which you want to forward records to an external Syslog server to keep data there for a longer timeframe say 20 years, but also keep a record in the database for a shorter timeframe say 2 years so that the recent historical records are searchable and for very old records you can go back to the Syslog server.

Setting up an External Database for Data Archival

To set up an external database for data archival, do the following:

1. On the externalized data archival database run the following commands:
   a. To ensure that the data archival server allows connections, open the firewall port:
   ```
   # firewall-cmd --add-service=postgresql --permanent
   # firewall-cmd --reload
   ```
   b. To ensure that the `pg_hba.conf` file trusts the FortiSOAR server for incoming connections:
   Add the following entry to the file `/var/lib/pgsql/16/data/pg_hba.conf` file:
   ```
   local all all md5
   host all all ip/subnetmask md5
   ```
   For example, if the ip/subnetmask of your externalized PostgreSQL database is xxx.xxx.xxx.xxx/xx then add the following to the `pg_hba.conf` file:
   ```
   local all all md5
   host all all xxx.xxx.xxx.xxx/xx md5
   ```
   c. To ensure that the `postgresql.conf` file trusts the FortiSOAR server for incoming connections:
   Make the following changes to the `/var/lib/pgsql/16/data/postgresql.conf` file:
   ```
   listen_addresses = '*'
   port = 5432
   ```
   d. Restart PostgreSQL using the following command:
   ```
   # systemctl restart postgresql-16
   ```
   e. Create a `cyberpgsql` user using the following command:
   ```
   sudo -u postgres psql -U postgres -c "CREATE USER cyberpgsql WITH SUPERUSER PASSWORD
   'changeme'";
   ```
   Important: The password of the `cyberpgsql` must not contain the following characters: `'^$`.

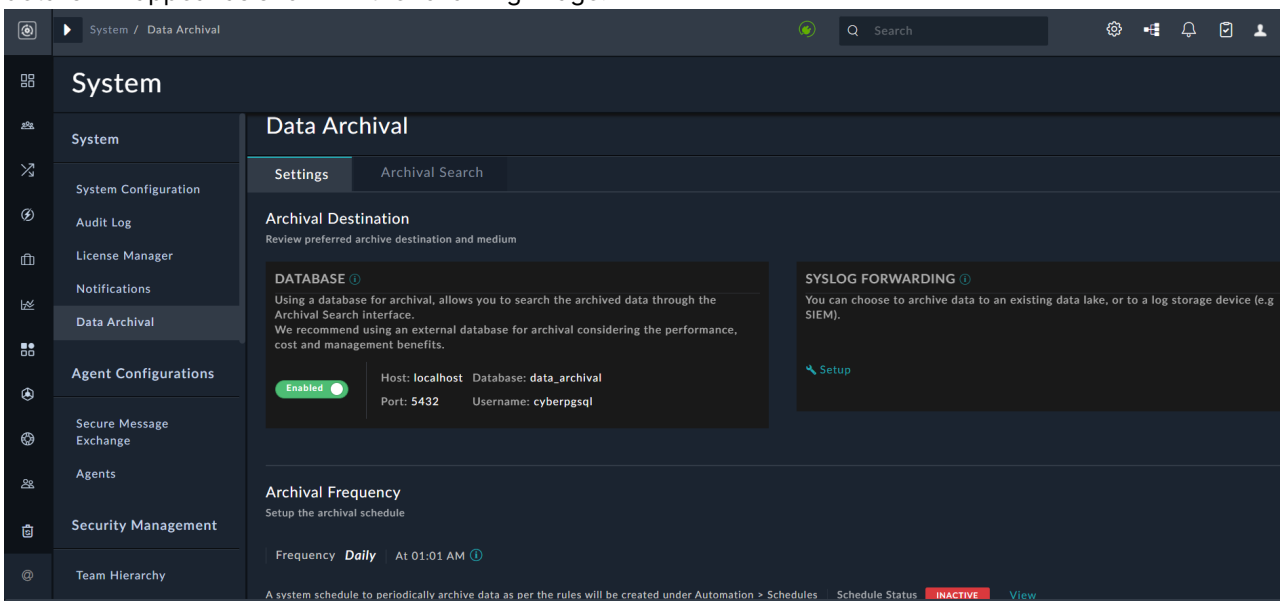
   f. Enable the `hstore` and `uuid-osop` extensions (this is a *recommended* step) using the following commands:
   ```
   sudo -u postgres psql -U postgres -d postgres -c "CREATE EXTENSION IF NOT EXISTS hstore;"
   sudo -u postgres psql -U postgres -d postgres -c 'CREATE EXTENSION IF NOT EXISTS "uuid-ossp";'
   ```
2. SSH to your FortiSOAR VM and log in as a *root* user to perform the next steps.
3. Create the `db_external_config.yml` file at the following location `/opt/cyops/configs/database/`
   Use the following command to create the `db_external_config.yml` file:
   ```
   # cp /opt/cyops/configs/database/db_config.yml /opt/cyops/configs/database/db_external_config.yml
   ```
   Note: If you have already externalized your FortiSOAR databases, then a `db_external_config.yml` file will already be present.
4. Edit the `db_external_config.yml` file to add the details for the data_archival external database as follows:
   In the `postgres_archival` section:
   a. Set the `pg_archival_external` parameter to "true".
      This parameter determines whether or not the Postgres archival database needs to be externalized. If it is set to "true", then the Postgres archival database is externalized, and if set to "false" (default), then the Postgres archival database is not externalized.
   b. Update the value of the data archival host (`pg_archival_host`) and data archival port (`pg_archival_port`) (if needed) parameters.
   c. Add the encrypted password that you have set on your remote Data Archival server in the `pg_archival_password` parameter.
      You can encrypt your passwords by running the `csadm db --encrypt` command as a *root* user. For more information on `csadm`, see the FortiSOAR Admin CLI chapter.
5. Check the connectivity between the FortiSOAR local instance and remote data archival database using the `csadm db --check-connection` command.
6. To externalize the data archival database, type the following command:
   ```
   # csadm db --archival-externalize
   ```
   Or run the following command to externalize all the FortiSOAR databases, including the data archival database:
   ```
   # csadm db --externalize
   ```
   Once you run the above command, you will be asked to provide the path in which you want to save your database backup file.
   Note: If you run the `# csadm db --externalize` option more than once (i.e., you are running the option again after the first time), then `csadm` will display a message such as:
   `The databases already exist in postgresql, do you want to delete these databases (y/n):` If you want to externalize your PostgreSQL database again you must type `y`.
7. After you have completed externalizing your PostgreSQL database, you should restart your schedules.

You can choose to externalize both the main PostgreSQL database and your archival database to the same external database or a different external database. However, if you have externalized your main PostgreSQL database, then you must externalize your archival database, i.e., the external database for data archival cannot be set to 'localhost'. For more information, see the Externalization of your FortiSOAR PostgreSQL database chapter.

Configuring various settings for Data Archival

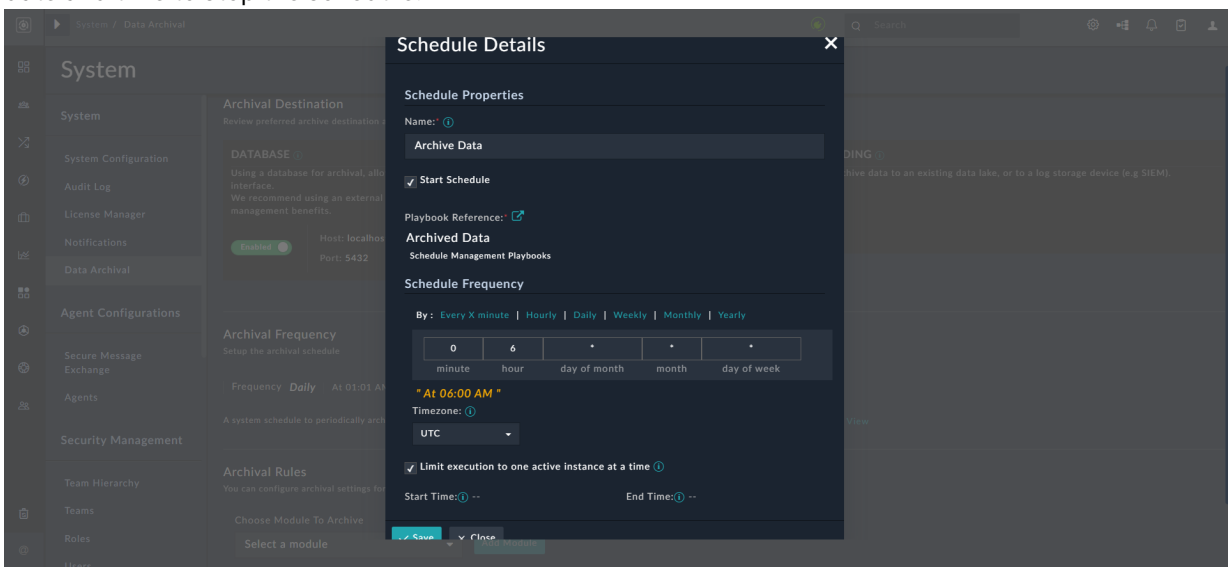
Important: To configure various settings for data archival, such as archival frequency, rules for archival, etc., you must have `Update` permission on the `Application` module and `Create`, `Read`, `Update`, and `Delete` permissions on the `Data Archival` module.

1. Log onto FortiSOAR and in `System` click the **Data Archival** option.
2. On the `Data Archival` page, in the **Setting** tab, in the `Archival Destination` section, you can view the details of the database that you have set up for data archival. Details include the host, port, and name of the database and the username that you have set up for the database. In the case of an internal database, details will appear as shown in the following image:



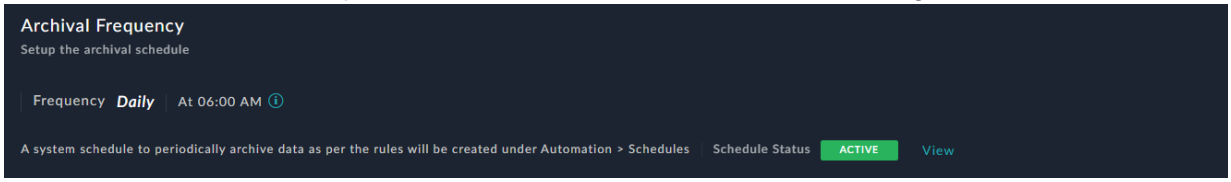
   If you have set up an external database for data archival, its details are shown in the Database section. For information on how to set up an external database for data archival, see the Setting up an External Database for Data Archival topic.
   If you want to archive data to an external Syslog server, either in addition to the external or internal database or as a sole archival destination, then to update the Syslog configurations, you must have `Security Update` permissions and you must ensure that the destination Syslog server IP is reachable from the FortiSOAR instance and should accept TCP/UDP data in port that is set up for communication.
   To configure Syslog, click **Setup** in `Syslog Forwarding` and enter the following details in the `Archival Syslog Setting` dialog:
   a. In the **Server** field, enter the IP or hostname of the Syslog server that you want to set up for data archival.
   b. From the **Protocol** drop-down list, choose UDP or TCP as the protocol to be used to communicate with the Syslog server.
   c. In the **Port** field, enter the port number to be used to communicate with the Syslog server.
   d. Click **Save** to save the Syslog details.
3. In the `Archival Frequency` section, set up the archival schedule, which is a system schedule that runs periodically as per the timeframe you have configured and archives data:

a. Click **View** beside `Schedule Status`, which is set to *Inactive*, to open the `Schedule Details` dialog.
b. Click **Start Schedule** to begin the schedule immediately, or you can also set the Start Time and End Time for the schedule.
c. In the `Schedule Frequency` section, choose the frequency of running this schedule.
For example, to run the data archival daily at 6:00 am, click **Daily** and then in the hour field enter 6 and in the minute field enter 0.
d. From the **Timezone** drop-down list, select the timezone in which you want the schedule to run. By default, this is set as UTC.
e. If you want to ensure that you do not rerun the workflow, if previous scheduled instance is yet running, then click **Limit execution to one active instance at a time**.
f. (Optional) From the **Start Time** field, select the date and time from when the schedule should start running.
g. (Optional) From the **End Time** field, select the date and time till when the schedule should run, i.e., the date and time to stop the schedule.



For more details on schedules, see the *Schedules* chapter in the "User Guide."

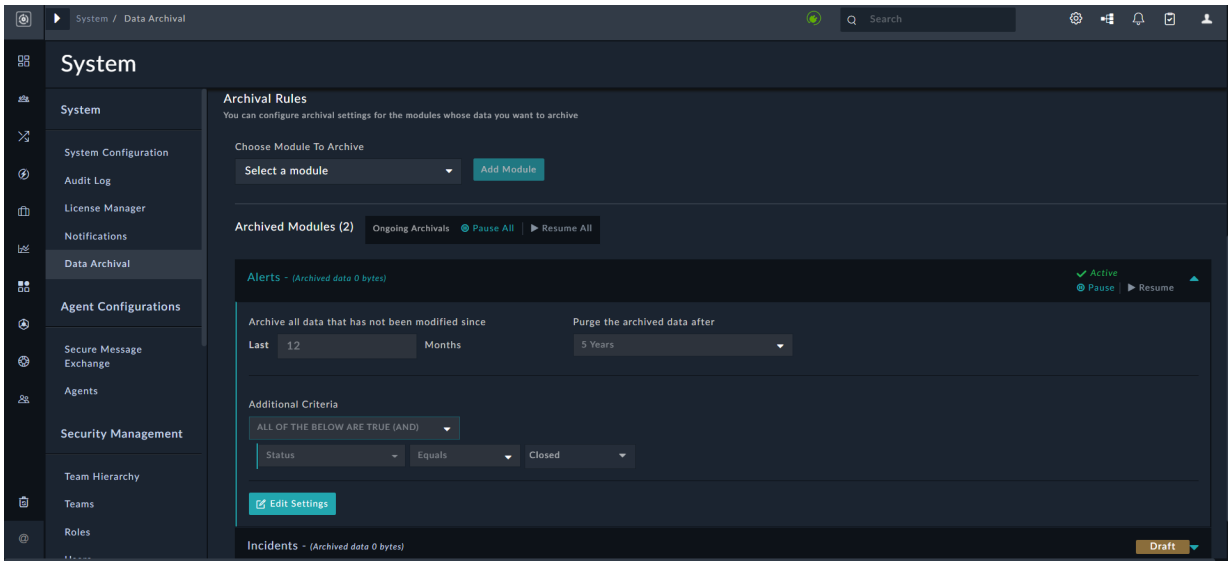
h. Click **Save** to save the schedule.
Once the schedule is saved, you can see that the `Schedule Status` has changed to *Active*:



4. In the `Archival Rules` section, you can choose the modules that you want to archive as well as set up rules for archival:
a. From the **Choose Module To Archive** drop-down list, select the module whose records you want to archive and click **Add Module**.
For example, select **Alerts** and click **Add Module**.
b. In the **Archive all data modified earlier than** field, enter the number of months earlier than which you want to archive the records. For example, if you enter `12` in this field, then it means that all records, which were modified earlier than 12 months will be archived.
Note: The value that you mention in this field must be in multiples of 3.

c. From the **Retain archived data for** drop-down list, choose the number of years the archived data should be retained in the specified archival destination.

d. In the `Additional Criteria` section, add conditions that refine the selection of data to be archived. For example, if you want to archive closed alerts only, then click **Add Condition** and from the **Select a field** drop-down list, select **Status**, from the **Operator** drop-down list, select **Equals**, and then select **Closed**.



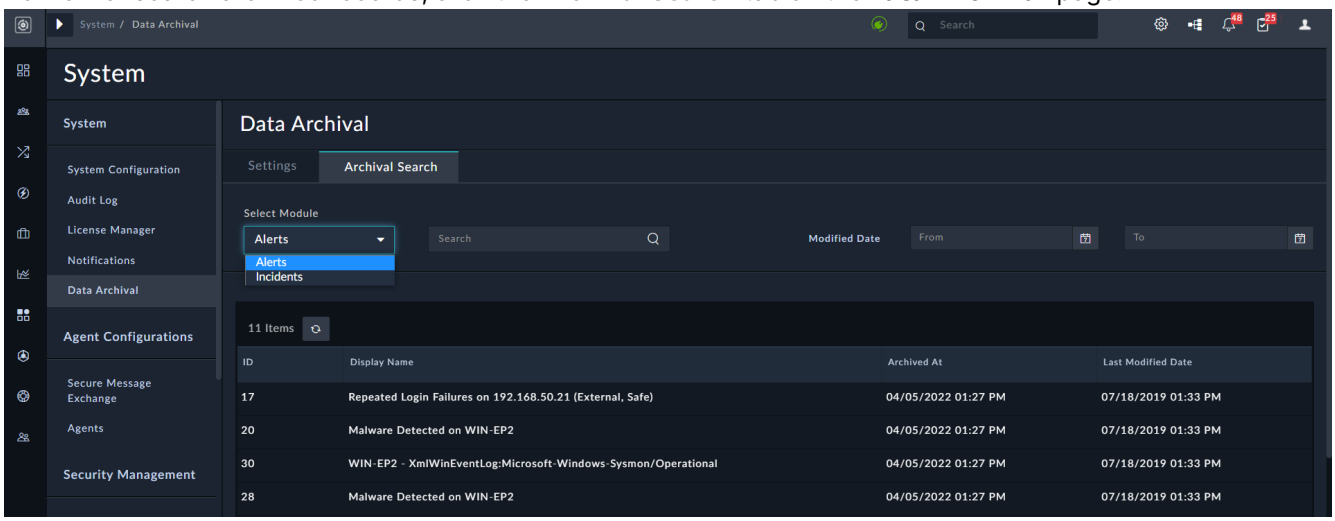
e. Click **Save and Start Archival** to save the settings and start data archival for the specified module. Settings are saved in the **Draft** mode if you do not click **Save and Start Archival**.
In the `Archived Modules` section, you can click **Pause All** or **Resume All** to pause or resume archival for ongoing archival for all modules. Additionally, in the row of particular module, for example Alerts, you can click **Pause** or **Resume** to pause or resume archival of that particular module.
You can also edit the settings for any archival, by clicking **Edit Settings**.

A record gets archived along with their entire relationships, but the actual relationship records get archived according to their own schedule. For example, if an alert has an indicator as its related record, then the alert is archived along with its indicator relationship, i.e., the indicator's value and reputation; however, the actual indicator record will be archived according to its own schedule.
Also, unique Constraints will not be considered across the primary and archived tables. For example, `Alerts` have unique constraints defined on “source Id” fields, once an alert is moved to the archival, new alerts with the same “source Id” can be created. Additionally, there are no constraints on data residing in an archival.

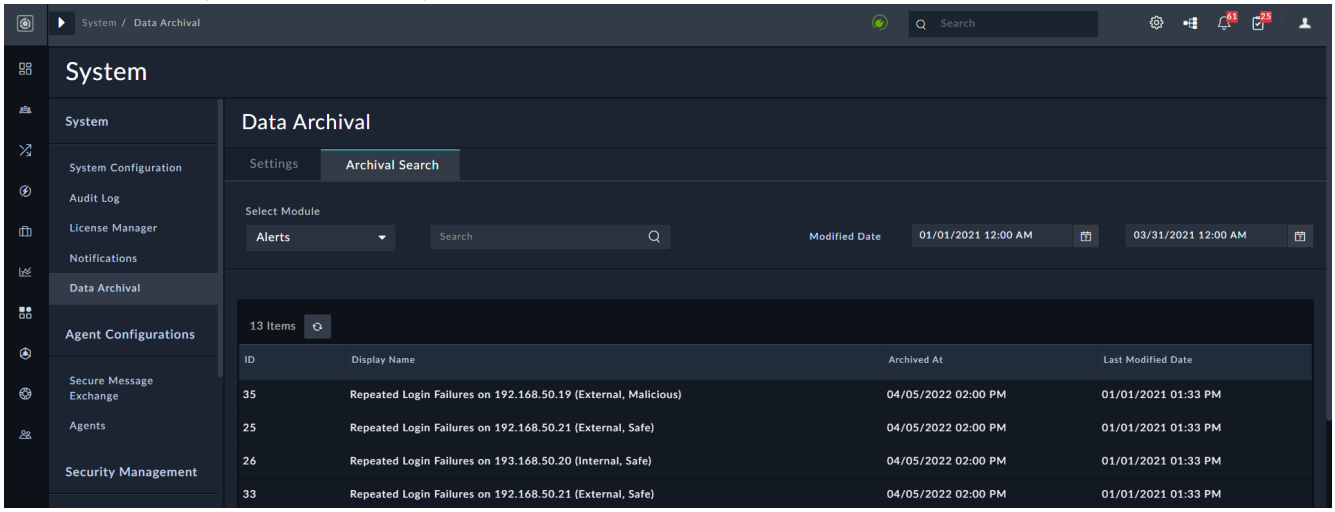
Viewing and Searching Archived Records

To view archived records, you must have permissions on the module to view the archived records, for example, to view archived alert records, you must have permissions on the `Alerts` module. Team ownership and user ownership of the records as at the time of archival is carried forward and honored while rendering and searching across archived data.

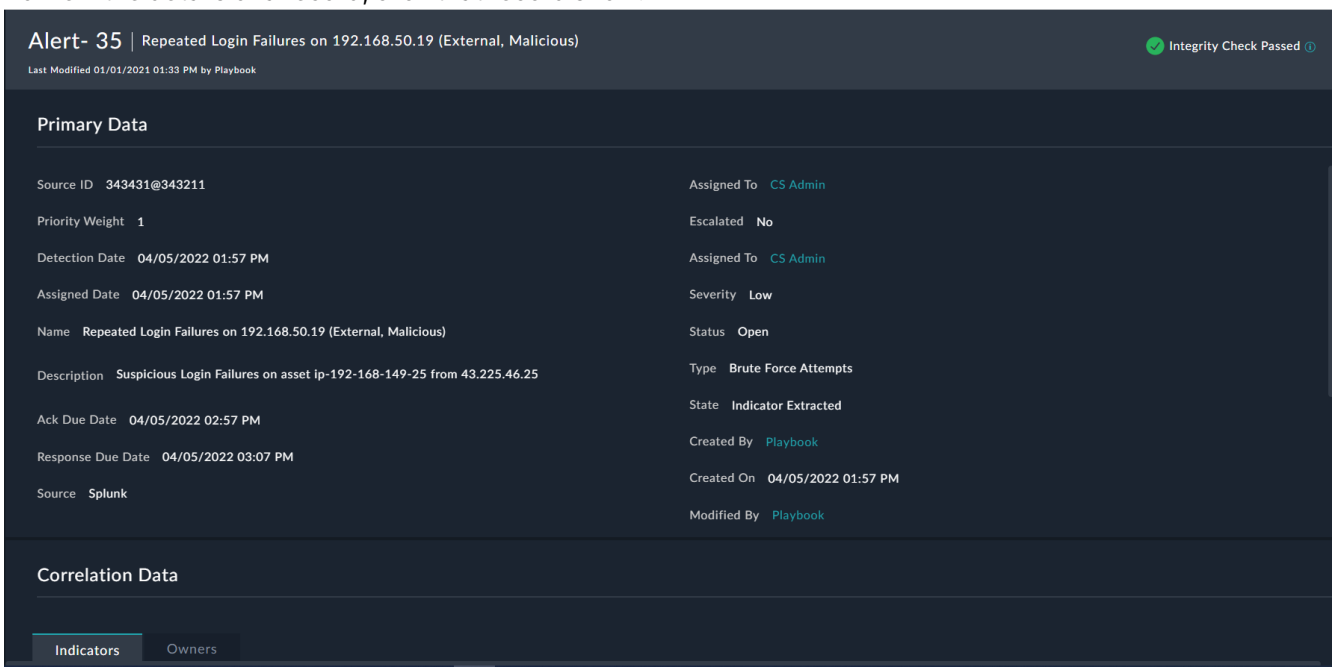
To view or search archived records, click the **Archival Search** tab on the `Data Archival` page:



You can search for archived records based on the module of the record, and the ID or Display Name of the record. Additionally, you can filter archived records based on their modified date and time. For example, to filter archived records from the 'Alerts' module whose modified date is between '1st January 2021' to '31st March 2021', select **Alerts** from the **Select Module to Search** drop-down list, and in the **Modified Date To** calendar, select 1st January 2021 and similarly, from the **From** calendar select 31st March 2021:



To view the details of a record, click that record's row:



The record detail view displays the primary data of the record and it contains only those fields whose values are not null or which are lookup fields. The `Correlation Data` section displays the relationship data of the record in their respective tabs. The archived record also contains an integrity check (required for auditing), which checks that the archived record has not been tampered with or modified (in the database itself). The **Integrity Check Passed** check indicates that the signature of the record's current state matches the original value, and there has been no tampering of the archived record.

Blocking specific HTML tags and attributes

Rich Text fields can be used to accept and render the HTML input making it possible for users to inject HTML code into FortiSOAR to insert content such as a link or an image. These can be used to trick a user into taking harmful actions or into believing misinformation. To avoid this, administrators can choose from the following options:

- Use the Text Area field instead of the Rich Text field.
- If you need to use the Rich Text Field, then you can choose between the following options:
  - Configure an iFrame or an iFrame sandbox (using the iFrame Widget) if you want to render HTML content
  - Block specific HTML tags and attributes if you want to add, edit, and render the HTML content with restrictions on a few tags or attributes.
    Note: By default, safe HTML tags and attributes are allowed.

To add the configuration for allowing or disallowing specific HTML tags and attributes, do the following:

1. SSH to your FortiSOAR VM and log in as a *root* user.
2. Open the `config.json` file using `vi /opt/cyops-ui/vendor/config.json`.

3. Update the `config.json` as per your requirements and then save the file.
   Note: The tags and attributes that you restrict in the `config.json` will apply to all instances of the Rich Text Field throughout your FortiSOAR instance.

Example of the `config.json` file:

```
{
    "markdown": {
        "allowedHTMLTagsAndAttrs": {
            "HTMLTags": ["button"],
            "HTMLAttrs": "",
            "AllowSVGContent": true
        },
        "blockHTMLTagsAndAttrs": {
            "HTMLTags": ["img","style”],
            "HTMLAttrs": [“style”]
        }
    },
    "pdfmake": {
        "font": {
            "normal": "Roboto-Regular.ttf",
            "bold": "Roboto-Medium.ttf",
            "italics": "Roboto-Italic.ttf",
            "bolditalics": "Roboto-MediumItalic.ttf"
        }
    }
}
```

Security Management

FortiSOAR gives you the power to assign levels of accessibility to users with Role-Based Access Control (RBAC) combined with Team membership. You can grant access to specific modules in FortiSOAR to users based on their Role Permissions. Users exercise their permissions in conjunction with their Team membership(s). Appliances are governed by the same authorization model.

The security model within FortiSOAR achieves the following four essential security goals:

- Grants users the level of access necessary based on your desired organization structure and policies.
- Supports sharing of data for collaboration while still respecting your team boundaries.
- Supports data partitioning and prevents users from accessing data that is not explicitly meant for them.
- Restricts external applications and scripts (appliances) from using the API beyond the requirements for accomplishing the desired RESTful actions.

The following sections describe several vital concepts you must keep in mind while working with the FortiSOAR security model. In-depth discussion and examples might be found in the individual Knowledge Base sections.

Important Concepts

Authentication versus Authorization

The FortiSOAR security model consciously treats authentication and authorization separately.

- Authentication defines your ability to log in and access FortiSOAR. FortiSOAR enforces authentication based on a set of credentials.
- Authorization governs users' ability to work with data within FortiSOAR *after* authentication is complete. You control authorization by assigning teams and roles to users.

This is an important distinction since when you are setting up user accounts, you must always define both the authentication and desired authorization for a user. Otherwise, once a user logs onto FortiSOAR, the user might be presented with a blank screen due to lack of authorization.

Users and Appliances

Users represent a discrete individual human who is accessing the system. Users are differentiated from Appliances in that they receive a time-expiring token upon login that determines their ability to authenticate in the system. The Authentication Engine issues the token after users have successfully entered their credentials and potentially a 2-factor authentication. By default, tokens are set to have a lifespan of 30 minutes before being regenerated.

The default 2-Factor authentication consists of a username and password for the primary authentication, and a unique code sent using an SMS or Voice message for the secondary authentication. The secondary

authentication method is not mandatory but highly recommended. You can configure the authentication methods on a per-user basis. Use the **System Configuration** menu to configure the system defaults for the secondary authentication.

The 2-Factor Authentication can be different for each user, but you can set it at a default preference level across the system. You can also allow a non-admin user to update their own 2-Factor Authentication mechanism. However, this is not recommended.

Appliances represent non-human users. Appliances use Hash Message Authentication Code (HMAC) to authenticate messages sent to the API. HMAC construction information is based on a public / private key pair. Refer to the "API Guide" for instructions for generating the HMAC signature.

For HMAC authentication the timestamp must be in UTC format.

Teams and Roles

Teams and Roles are closely aligned with a data table design. Teams own specific records, which are rows in a table. Roles govern permissions on the columns within that table around Create, Read, Update, and Delete (CRUD) activities.

Teams define ownership of discrete records within the database. A record can have more than one Team owner. Users can belong to multiple teams allowing them to access all records, which are owned by their assigned teams.

Roles define users' ability to act upon data within a CRUD permission set on any module in the system.

You must be assigned a role that has CRUD permissions on the Security module to be able to add, edit and delete teams and roles.

Security Management Menus

The Security Management Administration menu is split into the following areas:

Team Hierarchy

Use the `Team Hierarchy` menu to edit the relationships between teams defined within the system. You can also add and delete teams using the Team Hierarchy page.

Teams

Use the `Teams` menu to add new teams and edit user membership in bulk within each Team. You can also define membership within teams on an individual basis, using the individual user or appliance profile.

Roles

Use the `Roles` menu to create and define roles within the system. You assign roles based on CRUD permissions defined across all modules. You can assign roles in the User or Appliance profiles only. Currently, you cannot bulk assign roles.

FortiSOAR implements RBAC for playbooks; for example, for uses to run playbooks, administrators require to assign roles that have the `Execute` permission on the `Playbooks` module to such users.

Users who do not have Execute permissions will not be shown the **Execute** buttons for the module records, for example alert records. Execute actions include actions such as **Escalate**, **Resolve**, or any actions that appear in the **Execute** drop-down list.

Users

Use the `Users` menu to create and manage existing users. Each user has a profile with contact information including email and phone numbers plus additional reference information. You can assign teams and roles to users and control a user's state from the user's profile. User states are `Active`, `Unlocked`, `Inactive`, and `Locked`.

You must be assigned a role that has Create, Read, and Update (CRU) permissions on the People module to be able to add users and edit their user profiles. You cannot delete a user using the FortiSOAR UI, though you can make a user “Inactive” to stop that user from using the system. However, you can delete users using a script, for more information, see the Delete Users on page 192 section.

Access Keys

Use the `Access Keys` menu to manage keys for authenticating automation scenarios and using FortiSOAR APIs. Automation can now utilize API keys (from release 7.5.0 onwards) or HMAC authentication.

Authentication

Use the `Authentication` menu to configure various authentication settings in FortiSOAR, including setting session and idle timeouts and settings options for user accounts. You can also setup and manage the LDAP / AD integration and Single Sign-On (SSO) integration within your environment. When you use an external server to

perform authentication, you must have an administrative username and password to perform searches to import users. FortiSOAR supports the FreeIPA LDAP authentication.

FortiSOAR supports the following methods of authentication: Database users, LDAP users, and SSO.

Even if you configure SSO, you can still provision database and LDAP users.

Password Vault

Use the `Password Vault` menu to manage integrations with external vaults such as "Thycotic Secret Server" and "CyberArk" that are used by organizations to securely store their sensitive data and credentials. You can also use the `Password` field in the connector configuration instead to securely store and manage sensitive data, such as keys, API Keys, tokens, or credentials.

Configuring Team Hierarchy

Teams represent groups of "owners." If you are a member of a Team that owns a record, then you can apply any Role permissions you have on that record.

There is no direct connection between your Team and Role. If your Team or Teams own a record, you can do whatever you are permitted to do by your Role or Roles. If you are on multiple Teams, you have the permissions provided by your Roles across all those Teams.

Teams only provide ownership of records. Team Relationships extend ownership from one Team's members to another Team's members. Team Relationships are almost a form of "sudo" to borrow from Linux concepts, where you are effectively acting as if you were a member of another Team though you might not be explicitly on the roster of that Team.

> Team Relationships do not govern any user permissions. A user's Role or Roles determines their permissions. If you have extended ownership of a record AND sufficient privileges for that record module, then you can exercise those permissions on the extended ownership record.

If your Team has the appropriate relationship, you can work with a record owned by another Team as if you were on that Team, even though your Team may not be identified as an owner.

> All user actions in the system are audited, so there is no way for a user to work on a record from another team through a relationship that is not known and recorded.

Relationships

Teams govern record ownership within the FortiSOAR Security Model. Team Hierarchy reflects how team ownership relates between discrete teams.

Use the Team Hierarchy editor to define team relationships in accordance with each team's relationships with other teams in the system. The possible team relationships are shown in the following table:

Relationship Type	Description
Parent	Parent Teams are virtual owners of the records of the Child Team. A Parent team can act on those records as if they were a member of the Child Team.
Sibling	Sibling Teams can act on each other's records as if they were each members of the same team.
Child	Child Teams are the opposite of Parent Teams. Members of the Parent Teams can act on the records owned by the Child Team, but members of the Child Team cannot act records owned by the Parent teams.

A simple organization chart cannot capture the relationships in this definition. The real structure looks more like a mind map. This was a conscious design decision to support more advanced Team relationship use cases, such as allowing for internal investigations among existing users without alerting the user and providing Legal persona with their own permissions during Incidents.

Records created by 'nth' level of team hierarchy will be visible to parent teams. For example, records created by grandchildren teams will be visible to the grandparent teams.

There is **no inheritance** in relationships among 'Children Teams'; therefore, just because two teams, for example, Team A and Team B, are Children of Team Z, their parent, does not imply that Team A and Team B are each other's siblings or are related to one another in any other way. If you want Team A and Team B to be Siblings, you must explicitly define that relationship.

Honoring RBAC for related records

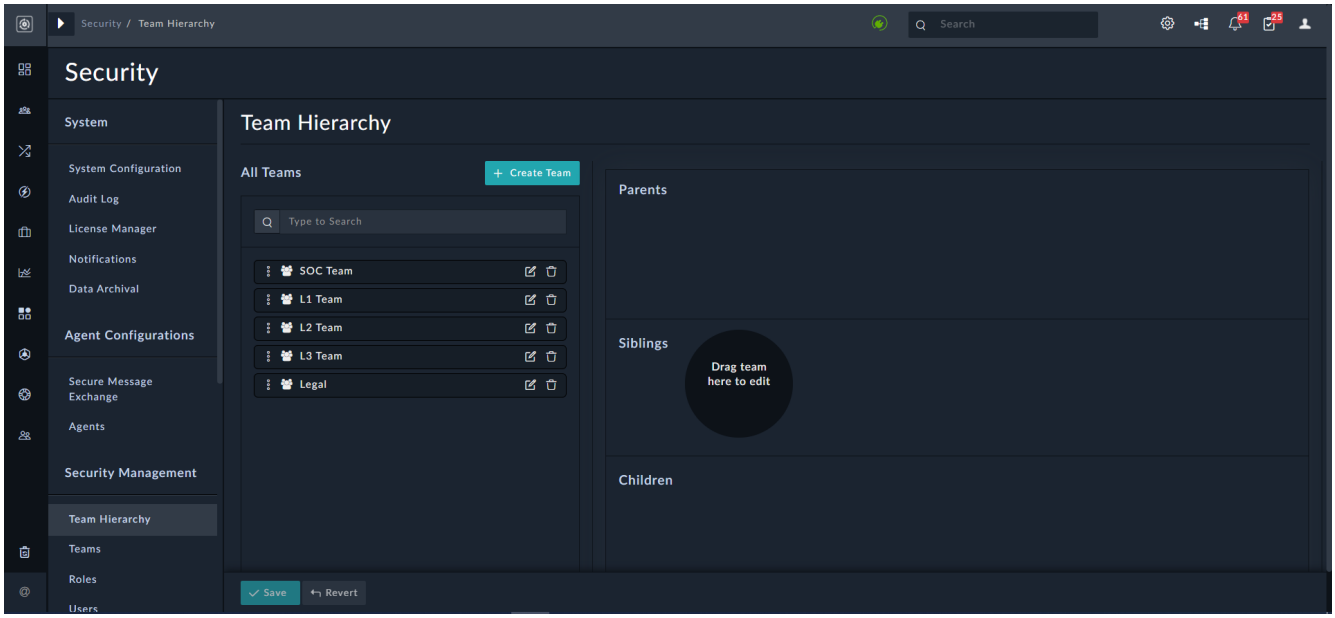
Teams govern record ownership within FortiSOAR. However, in the case of related records display the RBAC is ignored when viewing records in context of their parent records. For example, if users have access to an alert records that is linked to 5 indicator records, among which the user does not have access to one indicator record, still the user is able to view all 5 indicator records in the context of alert records in alert details page under the relation tab. Note that when users click the related records to which they do not have access, users will see an "`Access Denied`" message.
However, if you want to honor the RBAC for related records, i.e., display only those related records to which users have access when viewing records in context of their parent records, then do the following:

1. Open and edit the `/opt/cyops-api/config/parameters_prod.yaml` file.
2. Update the `ignore_rbac_for_related_record` parameter to change its value to 'false' (default is 'true'):
   `ignore_rbac_for_related_record: false`
3. Run the following command to complete the change:
   ```
   systemctl restart php-fpm && sudo -u nginx php /opt/cyops-api/bin/console cache:clear &&
   systemctl restart php-fpm
   ```

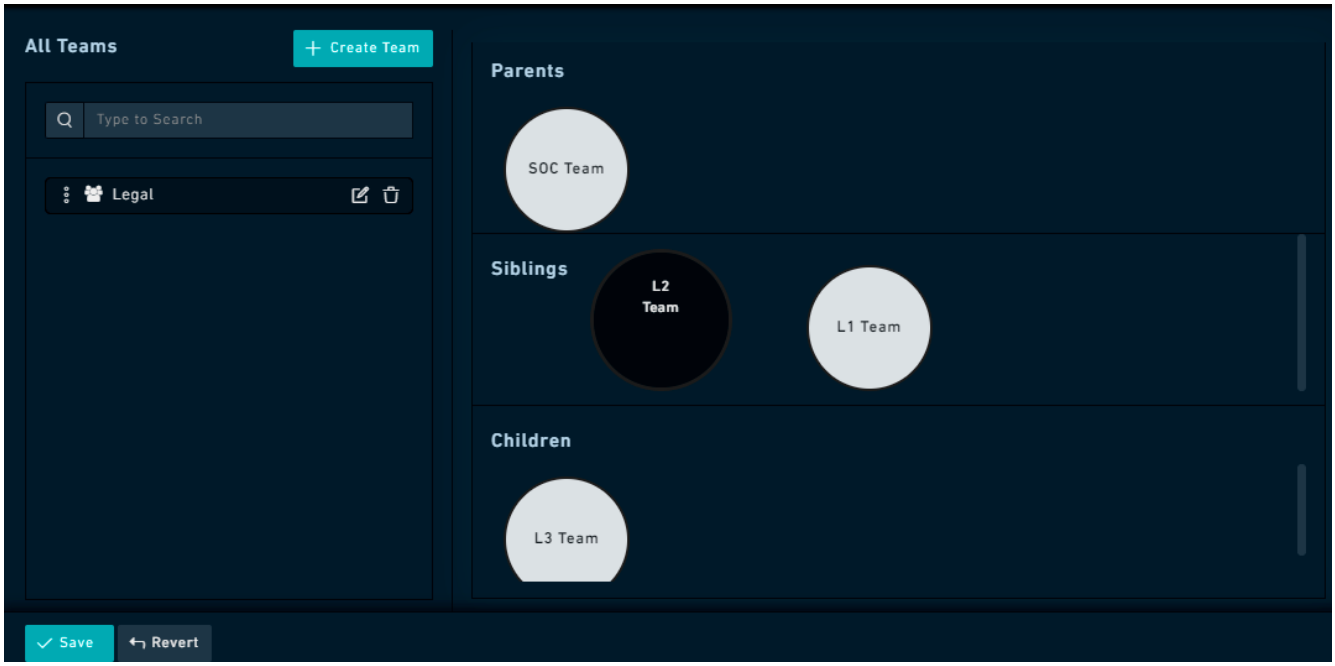
Using the Editor

The Team Hierarchy Editor is built to centralize around one team at a time and to define how that Team relates to all other teams in the system. The Central Team is referred to as the "Team in Focus" for this document. Click **Settings** > **Team Hierarchy** to open Team Hierarchy Editor.

The Team Hierarchy Editor has the All Teams menu and three swim lanes used to define the three relationship types, which are Parent, Sibling, and Child.



To edit the relationships of any team, you must first bring that team in focus. To bring a team in focus, you must drag and drop that team to the **Drag team here to edit** area or double click that Team's title in the **All Teams** menu.



Once you have put a Team 'in focus' on the Hierarchy Editor, the relationships that the team in focus has with all other teams is displayed in the respective swim lanes. For example, in the image above, the team in focus is the **L1 Team**. The L1 Team has SOC Team as the Parent team, L2 Team as its Sibling team and L3 Team as its Child team.

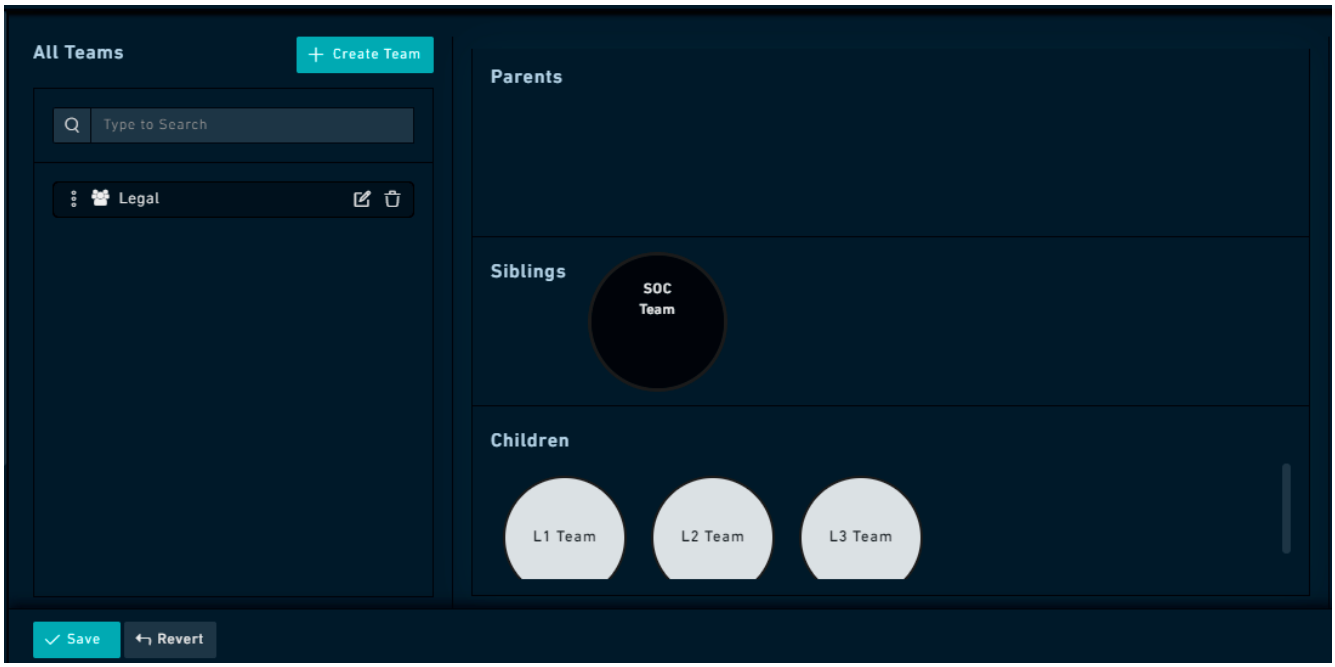
To edit the relationships, drag and drop Team bubbles or the Team titles in All Teams onto the appropriate swim lane. Changes are staged until you click **Save**. Once you click **Save**, changes immediately go into effect.

Following is an illustrative example of what is possible in this model:

Example

The SOC Team is the Parent of L1, L2, and L3 so the members of the SOC team can act across all records of the L1, L2, and L3 teams as if they are a member of all teams.

> Note you can achieve the same result by adding managers to every team in the organization. However, managers would then never be able to own any records exclusively.



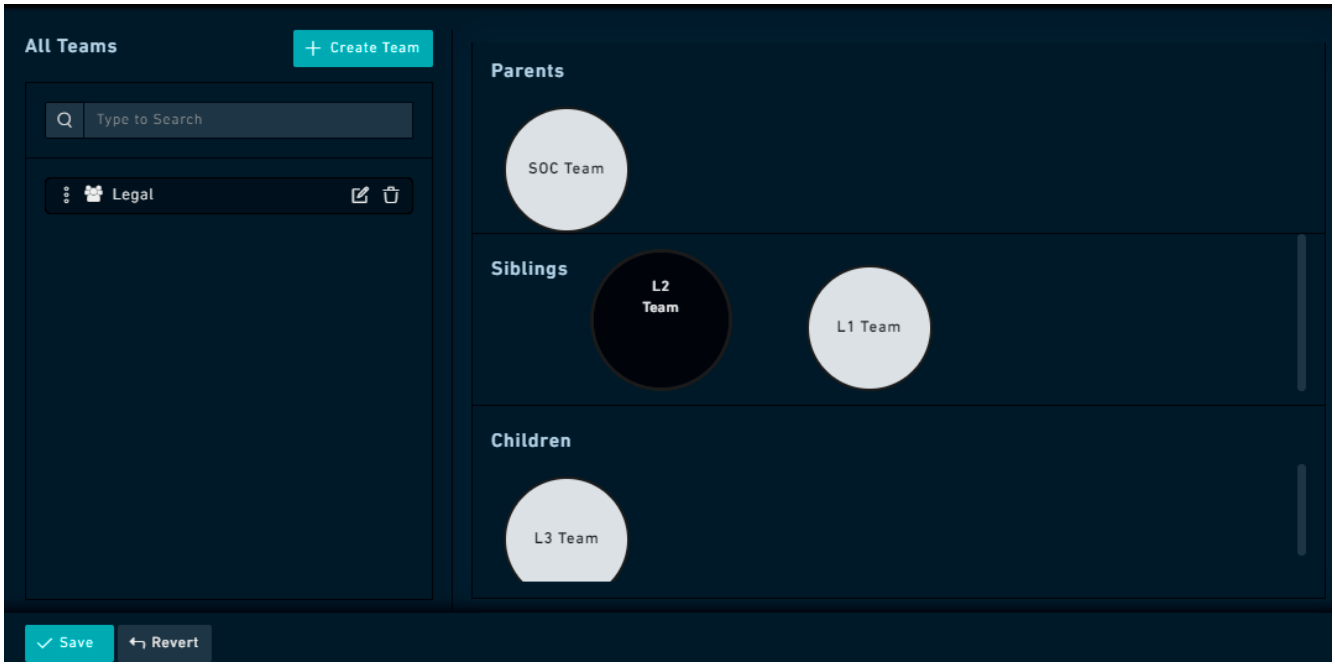
The Legal Team is unrelated to all other Teams in this case, which means that the SOC Team team is isolated from all the Legal Team's records and vice-versa. If the Legal Team were related to the SOC Team team, you would have seen the relationship in one of the swim lanes.

> The Security Module governs the Role for editing all Teams and Team hierarchies. Anyone with Read access to the Security Module can see all the Teams and Roles within the system.
>
> We recommend you provide Security Module permissions with caution as anyone with the Role can see any relationship in the system and would be alerted if any investigation into their activities were initiated at the Team level.

To summarize the relationships in this view, the SOC Team:

1. Effectively own all records of L1, L2, and L3
2. Own none of Legal

Now let's turn to a different team. If you were to focus L2 Team, you would find a slightly different case. We know that the SOC Team are a Parent Team, so we expect to see that relationship inverted. Beyond the relationship between SOC Team and the L2 Team, no other relationships are implied until you put L2 as the Team in Focus.

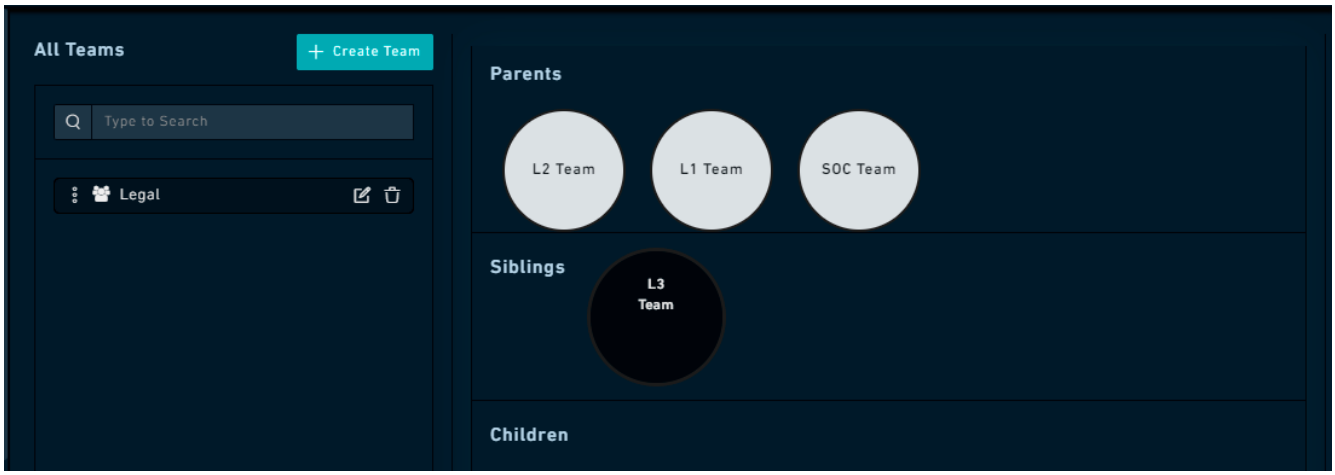


When L2 is the Team in Focus, you see that there is another set of relationships governing that Team. The L1 Team is a Sibling of L2, though **that is not** because the Teams are both Children of the SOC Team. The Sibling relationship has been explicitly defined between L2 and L3. You also see that the L3 Team is a Child of L2.

To summarize the relationships in this view, the L2 Team can:

1. Effectively own all records of L1 and L3
2. Own none of SOC Team records

The final piece of the example comes from placing L3 as Team in Focus. We know some things already about L3, namely that the SOC Team and L1 Teams are Parent Teams. But we do not know about L2.



When L3 is in focus, we see the expected relationships between the SOC Team and L2 Teams, but we now see that L1 is also a Parent.

To summarize the relationships in this view, the L3 Team can:

1. Effectively own only their own records
2. Own none of SOC Team, L2, or L1 records

Configuring Teams

Use the `Teams` page to manage members of a team centrally. You can assign a user to multiple teams; in fact, you can assign a user to be a part of all the teams.

By default, FortiSOAR has at least one team in place after installation, the **SOC Team**. We recommend that you do not modify the default teams and instead add new teams, as per your requirements.

There is no limit to how many Teams you can have in the system. Teams do not necessarily have to represent a specific team within your organization, but instead, Teams represent a group of users who own a set of records. In this way, you can think of Teams as row ownership within a table. The records are rows, and at least one and potentially more than one Team must own that row.

Whenever you add a new team, you must update the Playbook (called `WFUSER` in previous versions) assignment. `Playbook` is the default appliance in FortiSOAR that gets included in a new team. Only a user with `CRUD` access to the `Appliances` module can update the `Playbook` assignment, to ensure that the appliance has the necessary role to perform data read or write to modules. If the `Playbook` does not have appropriate permissions, then Playbooks will fail.

Editing Teams

Click **Settings** > **Teams** to open the `Teams` page. Use the Team Editor to create new teams and to assign users in bulk to teams. You can quickly move users between teams by selecting users who are designated to be Team

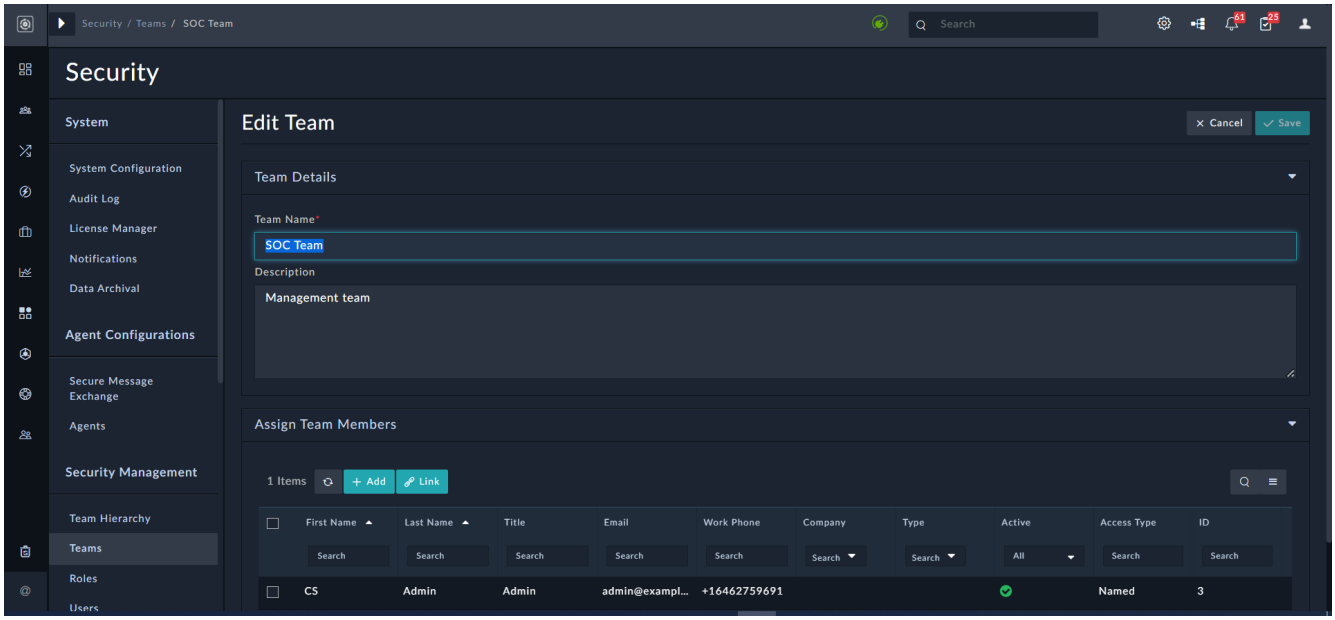
Members. You can use filtering and searching techniques to assign users to teams easily.

You can perform the following operations on the Teams page:

- Add a team
- Delete a team
- Clone a team
- Edit team details, including editing the name and description of the team and changing the assignment of users within a team

To Delete or Clone a team, on the `Teams` page, select the team you want to delete or clone, and click **Delete** or **Clone**.

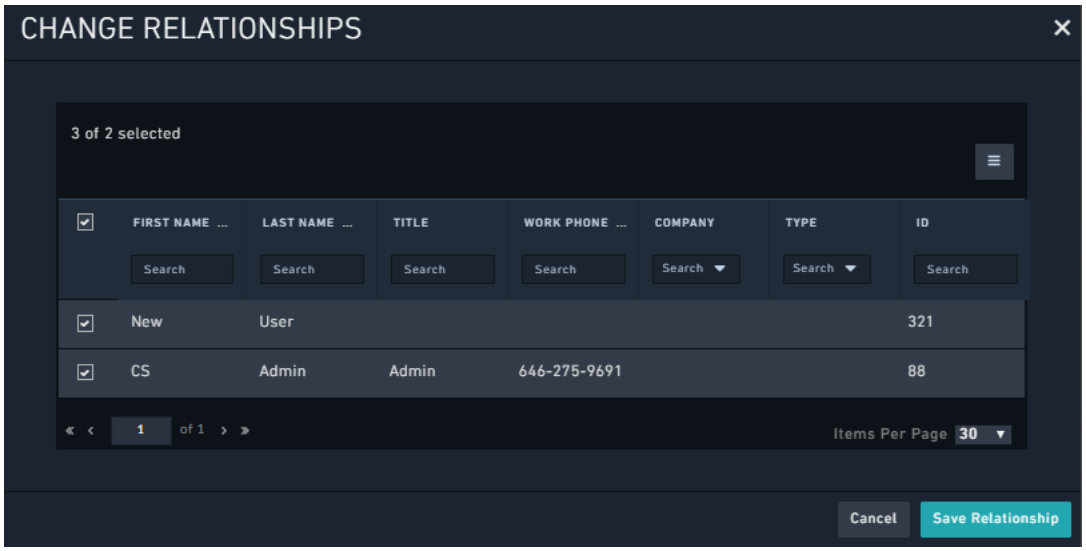
To edit a team, on the `Teams` page, click the team you want to edit. On the `Edit Team` page, you can change the name and description of the team and edit members. Members of a team are "linked" using the **Link** button on the `Assign Team Members` grid.



To add a user and then immediately assign that user to a team click **Add**.

To add members to a team, click **Link**, which brings up the `Change Relationships` modal window. The Change Relationships window displays all the users within the system. Click the checkbox on the user row to add the user to the team. To remove members from a team, click the checkbox on the user row. Click **Save Relationship** to complete your actions and add or remove members from a team.

Team membership takes effect immediately upon saving across the system.



Configuring Roles

The Roles menu allows you to define and modify all the roles within the application. Roles are not hard-coded in the system; therefore, Role editing is a sensitive permission and must be carefully governed by system users.

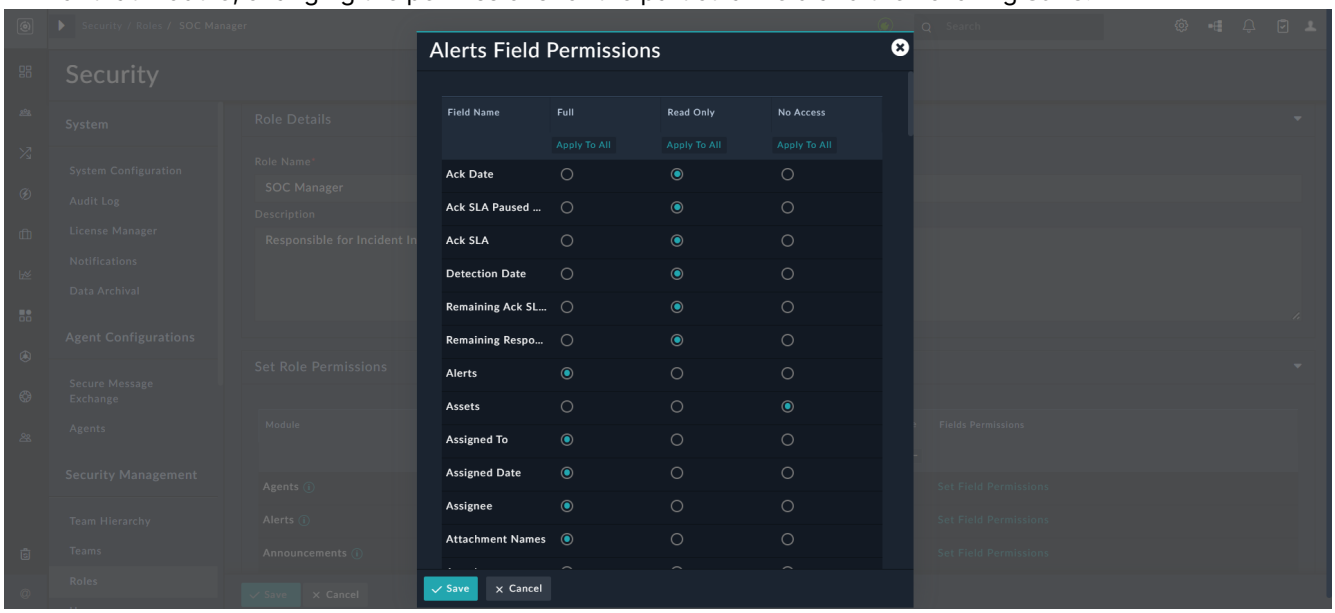
Any user that requires to work with FortiSOAR and records within FortiSOAR must be assigned a Role with a minimum of `Read` permission on the `Application`, `Audit Log Activities`, and `Security` modules.

Use the `Roles` page to add and edit RBAC permissions within FortiSOAR. Role permissions are based on the Create, Read, Update, and Delete model (CRUD). Each module within FortiSOAR has explicit CRUD permissions that you can modify and save for a particular role.

Release 7.2.1 has enhanced the 'Set Role Permissions' grid on the Roles page to make permission assignment efficient, intuitive, and logical:

- Automatic logical selection of permissions:
  - Selecting the 'Create' permission for a module leads to the automatic selection of the 'Read' permission for that module.
  - Selecting the 'Update' permission for a module leads to the automatic selection of the 'Read' and 'Create' permissions for that module.
- Locks the column headers of the 'Set Role Permissions' grid to ensure that is always displayed, which improves the usability of the grid.

You can also explicitly assign permissions for each field within a module by clicking the **Set Field Permissions** link for that module, changing the permissions for the particular field and then clicking **Save**:

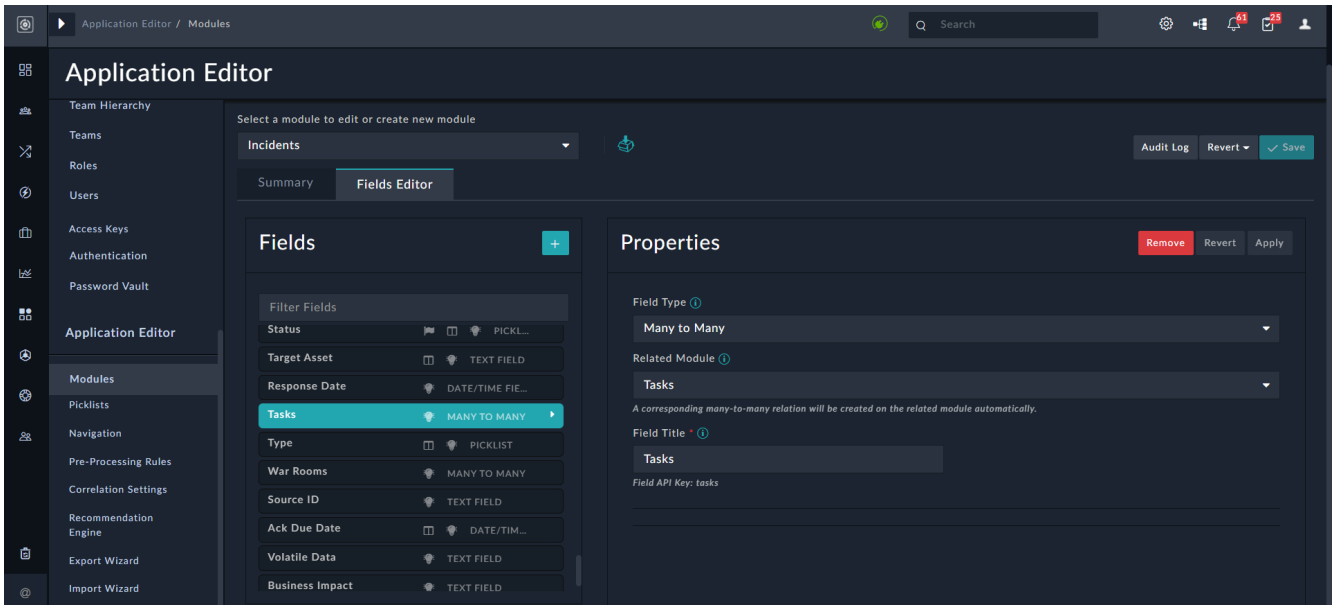


A user can have more than one role applied to their RBAC model. Each role granted to a user is additive to the users' overall RBAC permission set. Therefore, a users' RBAC permissions is an aggregation of all the CRUD permissions granted to them by each Role they are assigned.

Example 1: If you assign roles of `Security Administrator` and `Application Administrator` to User A, then User A will have CRUD permissions on both the `Security` and `Application` modules.
Note that the `Security Administrator` role also has CRUD permissions on the `Secure Message Exchange` and `Tenants` modules, so that this role can configure multi-tenant systems.

Example 2: If you had assigned the role of `Application Administrator` to User B, then User B gains all the CRUD permissions on the `Application` module and this user can configure your FortiSOAR system.

Example 3: If you want a user to work with Incident records, then you must assign that user with CRUD permissions on the `Incident` module, apart from that you must also assign the user a Role that has a minimum of `Read` access on all the related modules. To view the related modules, click **Settings** > **Modules**. Select the module whose records you want the user to work on, for example, **Incidents** from the **Select a module to edit or create new module** drop-down list. Click the **Fields Editor** tab to view all the fields and related modules, such as `Indicators` and `Tasks`, as shown in the following image. In this case, when you add a user to work in the `Incident` module, you must also assign the user a Role that has a minimum of `Read` access on the `Indicators` and `Tasks` modules.



Default Roles

By default, FortiSOAR has at least one role in place after its installation, the Security Administrator. Apart from the Security Administrator role, FortiSOAR generally also has the following default roles defined, after the installation of the SOAR Framework Solution Pack (SP):

- **SOC Manager** - manages the investigation of incidents and other containment and remediation tasks.
- **Security Administrator** - administers Teams and Roles and is responsible for creating the appropriate team structure and building and assigning roles.
- **Application Administrator** - given full access to application-wide features, so that they can configure the system and customize FortiSOAR as required.
- **Full App Permissions** - generally, this role is defined as one that has full permissions across FortiSOAR, i.e., a *root* user. You can define this role as per your requirements. Use this role carefully.
- **Playbook Administrator** - manages playbooks and connectors and also has permission to the `Security` module.
- **SOC Analyst** - triages alerts, filters false positives, investigates incidents, and performs other remediation and containment tasks.

Apart from the default roles, you can also create roles as per your requirements such as a **FortiSOAR Agent** that contains agent permissions, i.e., agent appliances are auto-assigned to this role. You can add this role directly to users so that they get access to agents. Agent appliances are auto-assigned to this role, and by default have access (CRU permissions) to `Files` and `Attachments`.

All Roles are "soft" roles, meaning none of the default Roles are hardcoded. You can add, modify, reassign permissions, and delete roles at will, but use this power with extreme caution.

We recommend that you do not modify or delete the default roles (and teams) and instead add new roles (and teams), as per your requirements.

SOC Manager

The SOC Manager role is given complete access to the `Alerts` and `Incident` modules and modules associated with the investigation of incidents, such as `Approvals`, `Assets`, `Communications`, `Indicators`, `Tasks`, `War Rooms`, etc, and also `Notification Rules`, `Schedules`, `Reporting`, etc. These users are responsible for investigating incidents and performing remediation and containment activities.

Security Administrator

The Security Administrator role starts by having full CRUD permissions across the `Security` module. This allows the Security Administrator to add and manage Roles and Teams within the application. The security administrator role also has CRUD permissions on the `Secure Message Exchange` and `Tenants` modules, so that this role can configure multi-tenant systems.

The Security Administrator should be assigned to someone who has been tasked with the responsibility for building and maintaining the role and team structure for your organization.

"Do not remove the Security Administrator Role"
We recommend you never remove the Security Administrator role. If you remove the Security Administrator role, you must ensure that at least one other role with an assigned user has the Security module enabled if you always want to maintain access to edit teams and roles within the application. You can assign the Security Module to another role, or another user, as required.

Playbook Administrator

The Playbook Administrator has access to the Orchestration and Playbooks component. Only users who have explicitly been given a minimum of `Read` access to Playbooks can see this component on the left navigation bar. For users to have full privileges to manage playbooks, they must be given `Read`, `Create`, `Update`, `Delete`, and `Execute` permissions.

System-level playbooks are also configured and should remain in place permanently. These are tagged as 'system, dev' and are now in a hidden Collection.

Application Administrator

The Application Administrator is granted access to configure application settings, found in the `Application Editor` section on the `Settings` screen.

All users must have Read privileges to the Application module to be able to use the application interface. Non-human users, API users, can be restricted from entering into the application GUI by not giving them any access to the Application module.

Full App Permissions User

Full App Permission user is a "root" user, who has full permissions across FortiSOAR. However, data partitioning is still in effect depending on the Team to which the Full App Role user belongs. The result of data partitioning is that a user with the Full App Permissions role might not see all the data within the application unless they have made their Team a Parent of all other Teams in the Application.

SOC Analyst

The SOC Analyst role is given access to the `Alerts` and `Incident` modules, and modules associated with alerts, such as `Comments`, `Attachments`, `Indicators`, `Tasks`, `War Rooms`, etc, and also `Schedules`, `Reporting`, etc. These users are responsible for alert triaging, false-positive filtering, investigating incidents and performing other remediation and containment tasks, and escalating potentially malicious alerts to incidents for review by the SOC team.

Modules in the Roles Page

Modules are discrete areas or record sets within the application. Some modules represent discrete record tables while some represent areas of modification within the administrator's panel.

Not all modules present in the Roles menu are available in the interface. Some of the modules are administrative or for particular purposes, such as the Files module.

Table Modules

Table modules are record sets that are editable within the FortiSOAR UI from a component level, i.e., these are all the modules that are listed in the Roles Editor, which is used to set module-specific permissions. Components, which include Incident Management, Vulnerability Management, Resources, etc., consist of a logical grouping of modules. For example, the Incident Management component contains modules such as Alerts, Incidents, Tasks, etc., and the Vulnerability Management component contains modules such as Vulnerabilities, Assets, and Scans. Each of these individual modules is accessible within the navigation menus.

Users can access and modify modules if they are given appropriate CRUD permissions to those modules within FortiSOAR. For example, if a user requires to modify alerts and incidents, that user must be assigned a role that at the minimum has 'Read' and 'Update' permissions on the "Alerts" and "Incidents" modules.

Administration Modules

Administration modules refer to specific areas of administration within the application. These are generally accessible in the `Settings` menu, with discrete tabs for each of the menu options.

Some of the admin modules found in the system, in alphabetical order, are:

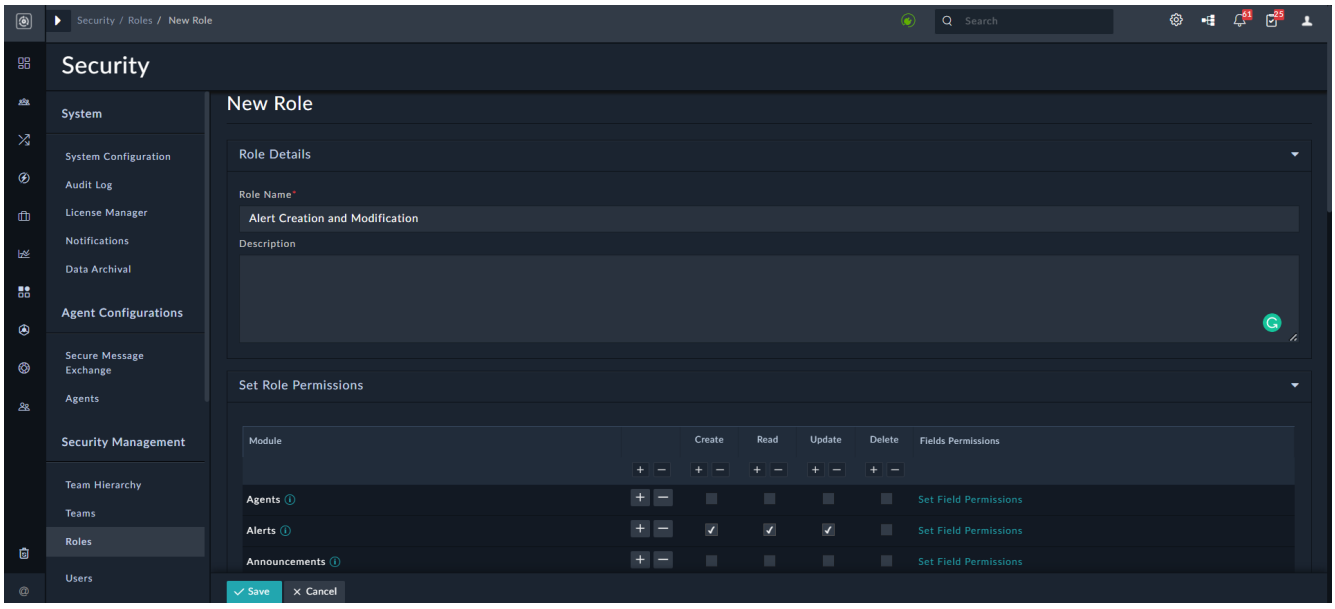
- **Appliances** - the ability to manage appliances from the **Appliances** item.
- **Application** - the ability to change system defaults used throughout the system from the **System Configuration** item.
- **People** - the ability to manage human users from the **Users** item.
- **Playbook** - the ability to access and manage the Orchestration and Playbooks component in the left navigation menu.
- **Password Vault** - the ability to integrate with external vaults such as "Thycotic Secret Server" and "CyberArk" to securely store their sensitive data and credentials.
- **Security** - the ability to manage teams and roles from the **Team Hierarchy**, **Teams**, and **Roles** item.

Adding Roles

To add a new role, click **Settings** > **Roles** to open the `Roles` page. Click **Add** to open the `New Role` page enter the role name and description in the respective fields. In the `Set Role Permissions` grid, the Module column displays the name of the various modules to which you can assign permissions. Each of the `Create`, `Read`, `Update`, and `Delete` columns have checkboxes that allow you to assign specific permissions for each module. The Playbook module has an additional `Execute` permission that is required for users to execute actions and playbooks.

Whenever you add a new role by default, the `Read` permission for "Application" will be selected.

For example, if you require to create a user who needs to add and modify alerts and their associated tasks, you can create a new role as shown in the following image:

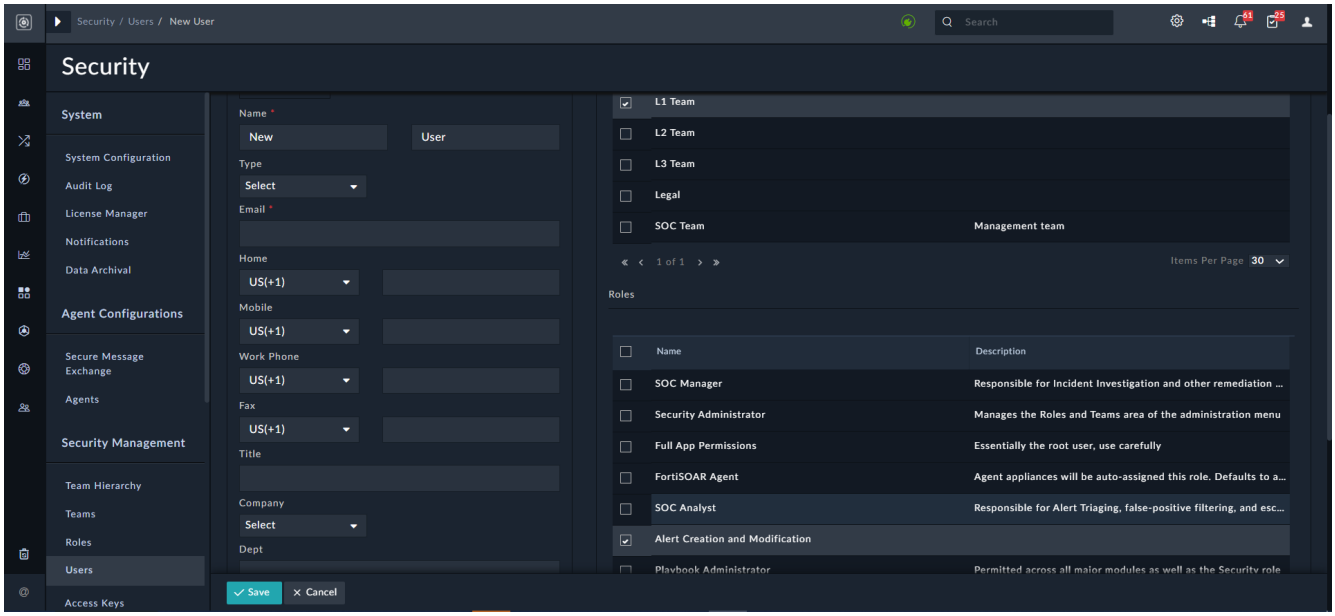


Assigning Roles to Users and Appliances

You cannot assign roles in bulk to Users or Appliances. You must assign roles directly assigned to users at the time of creating or updating user or appliance profiles.

To assign a role to a user, click **Settings** > **Users** to open the `Users` page. The `Users` page displays a list of users (active and inactive) for the organization. On the Users page, click the username to whom you want to assign the role. On the `Edit User` page, select the role(s) from the **Roles** table in the `Team and Role` section that you want to assign to the user, and click **Save**. If there are more than five roles in the system, the Roles table becomes scrollable.

For example, you can assign the Alerts creation and modification role to New User as shown in the following image:



Roles can be added or removed at any time from any profile. When permissions to a Role is changed, then enforcement begins immediately. However, as the UI is built upon login, some aspects of the UI for navigation might still be available until the UI is refreshed or logged out. For instance, if Playbook privileges are removed from your user, you will still be able to see the Playbooks navigation button in the UI, but when you navigate to it, you will be notified that you are not authorized to view that page (401 error).

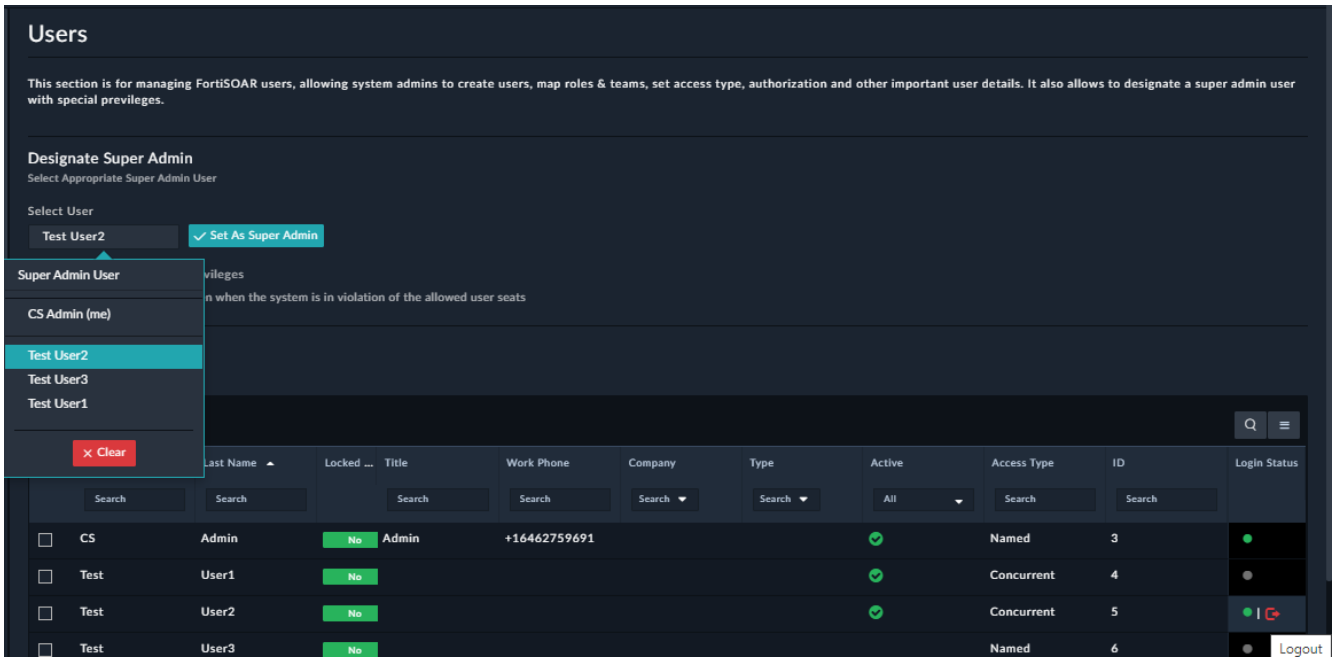
Users who are assigned roles having permissions to the 'People' module, but who do not have access to the 'User Id' field, i.e., the **User Id** field is set as '**No Access**' in the `People Field Permissions` dialog, are unable to see 'Locked', 'User Id', and 'Login Status' fields for users listed on the Users page in FortiSOAR. For information on setting field permissions for modules, see the Configuring Roles section

---

Configuring User Profiles

Adding Users

To add a new user, click **Settings** > **Users** to open the `Users` page. The Users page displays details such as name, title, lock status, ID, etc for all the created users in your FortiSOAR system.

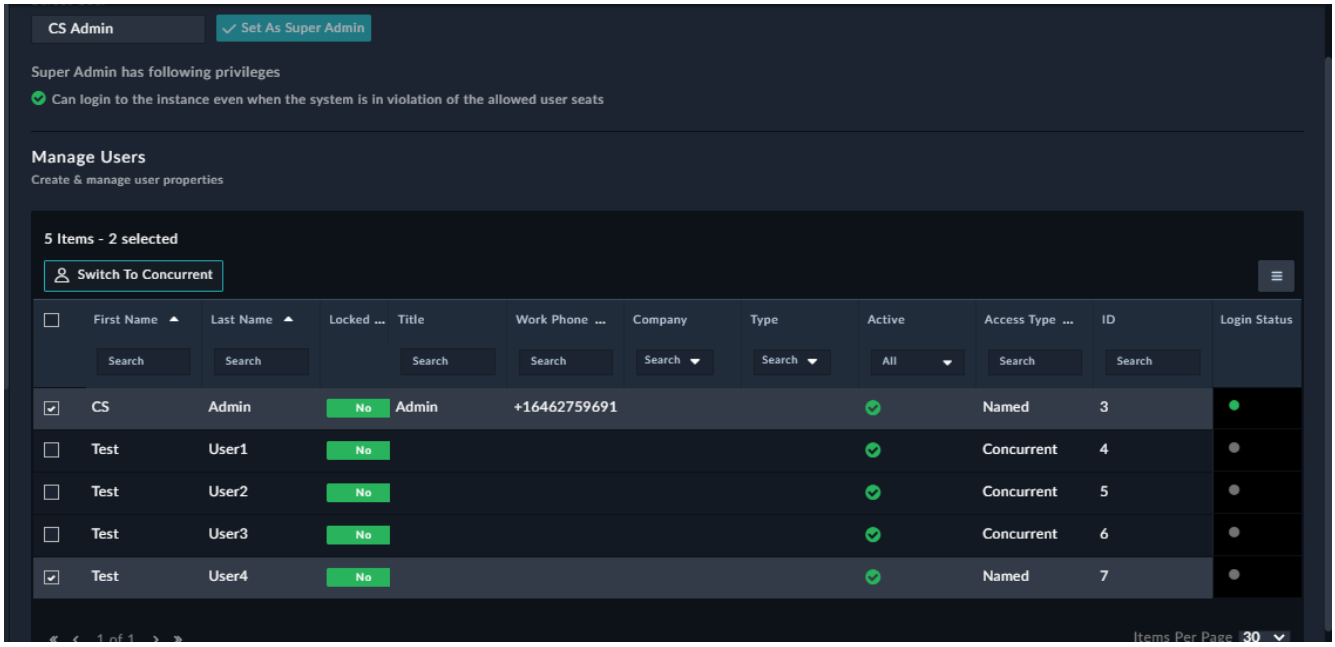


From version 7.0.1 onwards, you can select any active user as a "*Super Admin*." A *Super Admin* user has a privilege of being able to log into FortiSOAR even when the system is in violation of the "user seat entitlement", i.e., the number of named users exceeds the number of seats you have bought for FortiSOAR. By default, this user is set as '*CS Admin*'. Prior to version 7.0.1, the csadmin user could log into FortiSOAR, even when there was a license validation failure. However, there could be customer environments where the *csadmin* user is kept inactive or has been deleted, and therefore ability to choose any active user as a *Super Admin* is introduced in version 7.0.1. On the `Users` page, in the `Designate Super Admin` section, from the **Select User** drop-down list, which displays only active users, select the user that you want to designate as the *Super Admin* and click **Set As Super Admin**.

The user who is chosen as the *Super Admin* user must have `Read` and `Update` permissions on the `Security` and `People` modules, so that in case of a license violation the *Super Admin* is able to update the access type of the users. Also, the *Super Admin* user cannot be marked as '*Inactive*' or deleted.

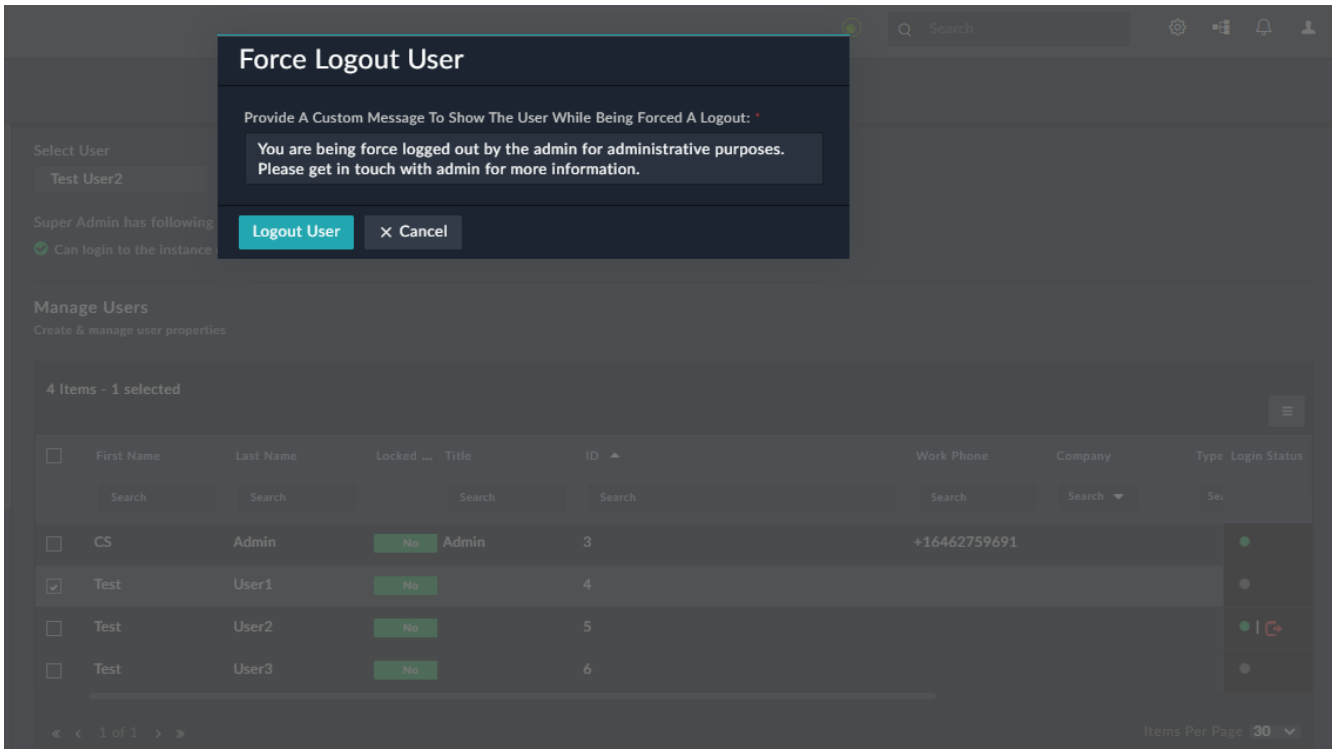
The `Users` page displays the access type, i.e., named or concurrent, and login status of all the users. For more information on 'Concurrent Users', see the *Licensing FortiSOAR* chapter in the "Deployment Guide." You can selectively update users' access type, i.e., Concurrent users to Named users, and vice-versa, at any time. You can also bulk update the access type of users from Named to Concurrent by selecting the users in the Grid on

the `Users` page whose access type you want to change from named to concurrent, then click the **Switch to Concurrent** button, and then click **Confirm** on the `Confirmation` dialog.



You might need to bulk update the user access from Named to Concurrent, when you have upgraded FortiSOAR.

You can forcefully log out selective '**Concurrent**' users from the system, by clicking the **Logout** icon in the row of a particular user. For example, if you want to log out Test User2, then click **Logout** in the *Test User2* row. FortiSOAR will display a `Confirmation` dialog, click **Confirm** to display a `Force Logout User` dialog, in which you can add a custom message that will be displayed to the user when the user is being forcefully logged out, and then click **Logout User**:



A `Named` user can never be forcefully logged out of any system.

To add a new user, click **Add** and enter the user details on the `New User` page and click **Save** to save the new user profile.

The **Username** field is **mandatory** and **case sensitive** and it cannot be changed once it is set. It is also recommended that all new users should change their password when they first log on to FortiSOAR, irrespective of the complexity of the password assigned to the users.

Use the SMTP connector to configure SMTP, which is required to complete the process of adding new users. The SMTP connector is used to send email notifications. If you have not set up the SMTP connector, the user gets created. However, the password reset notification link cannot be sent to the users, and therefore the process remains incomplete. For more information on the SMTP connector, see the "SMTP Connector" document.

User Profiles

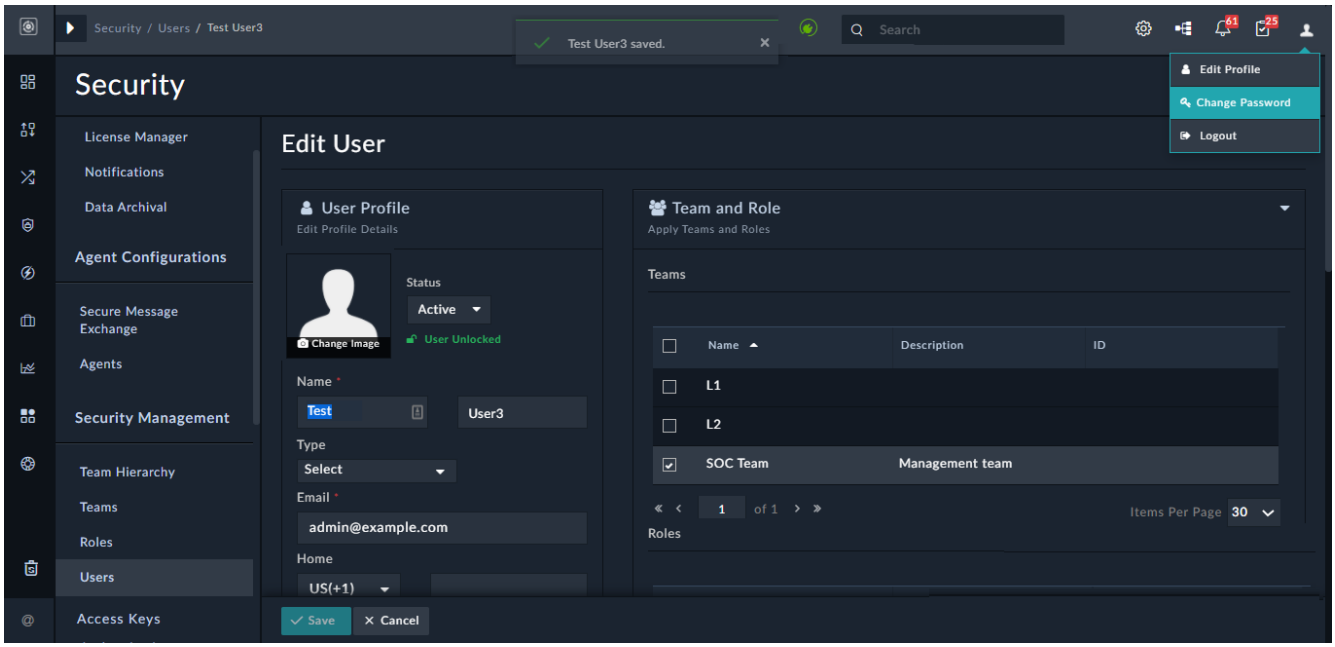
All users within the system have a profile. Each user has access to their own profile so that they can update specific information about them by clicking the **User Profile** icon ( ).

The user profile includes users' name, email, username, password, phone numbers, and access type, i.e., Named or Concurrent. Users can also view the team and roles they belong to as well as update their theme. Users can also view their own audit logs, which display a chronological list of all the actions that you have performed across all the modules of FortiSOAR.

You must change your password when you log on to FortiSOAR for the first time. To change your password, click the **User Profile** icon and then select the **Change password** option. You can also change your password at any time using this option.

The **Username** field is mandatory that cannot be edited once it is set.

To edit user profiles, you must be assigned a role that has a minimum of **Read**, **Create** and **Update** permission on the `People` module. Click **Settings** > **Users** to open the `Users` page and click the user whose profile you want to edit. This opens the `Edit User` page, where you can edit the user's profile, including the user's email ID, name, phone and fax numbers, users' teams, roles, 2-factor authentication settings, notification settings, and theme settings. You can also see their login history.

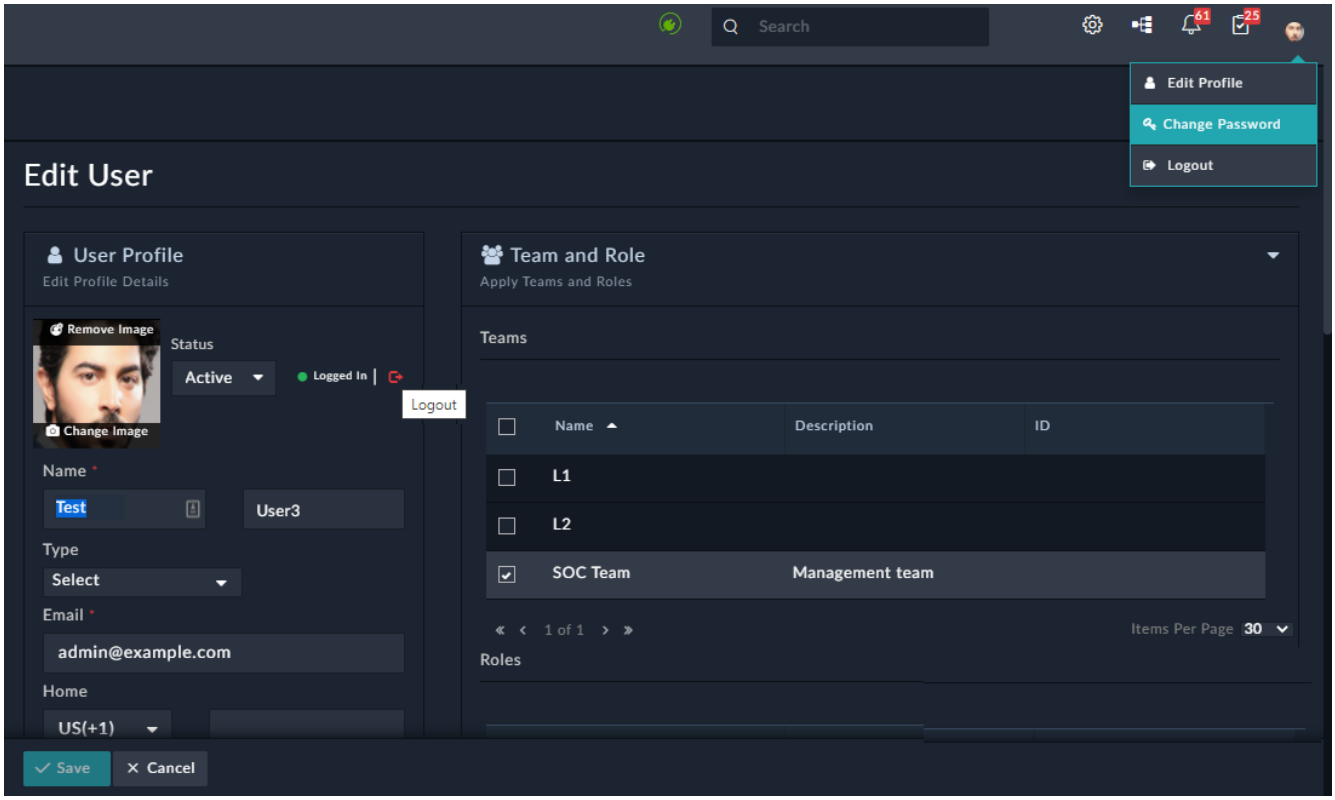


You can upload the user's profile picture by clicking **Change Image**, which opens the `Upload a Profile Picture` dialog, where you can drag-and-drop the profile image file, or click the **Import** icon and browse to the image file to import the profile image file to FortiSOAR, and then click **Save Profile Image** to add the profile image. Once the profile image is added, the same can be removed at anytime by clicking the **Remove Image** button that appears on the profile image.
A user is one whose `People` record is `Active`. If you have **Read** and **Update** permissions on both the `Security` and `People` modules, you can edit a user's Active or Inactive status on their profile page. If you change a user's status to *Inactive*, you stop that user from using the system upon expiration of their issued token.

Locked users are those who get temporally locked out of FortiSOAR when they have exceeded the number of authentications tries allowed within a one-hour period. By default, users' can enter incorrect credentials, username and/or password 5 times, while logging into FortiSOAR, before their account gets locked for 30

minutes. From version 7.0.0 onwards, administrators cannot lock a user using the FortiSOAR UI; however, administrators can unlock a locked user using the UI by clicking the **Unlock** checkbox on that user's profile page and then clicking **Save**, or locked users can wait for 30 minutes before their account gets unlocked. Security Administrators can also change the default values for the different parameters, such as number of attempts before the user account is locked, etc. For information about these parameters. see the Debugging, Troubleshooting, and Optimizing FortiSOAR on page 415 chapter.

From version 7.0.1 onwards, you can also forcefully log out selective users from the system, by clicking the **Logout** icon on the user's profile page:



If you face issues with user preferences such as applying filters on the grid or column formatting within a grid, click the **More Options** icon ( ) and click on the **Reset Columns To Default** option.

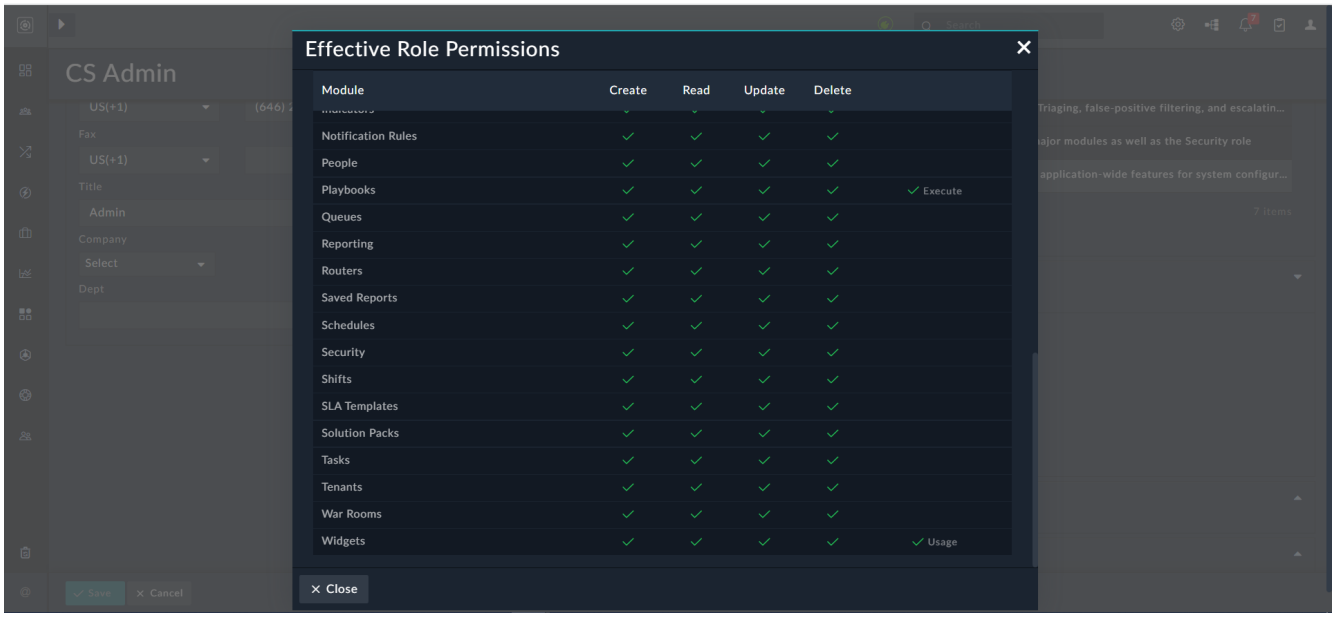
Teams and Roles

If you are editing your own profile and you have no access to the `People` module, then you can only view to which teams and roles you belong.

If you are assigned a role with **Read**, **Create**, and **Update** permissions on the `People` module then:

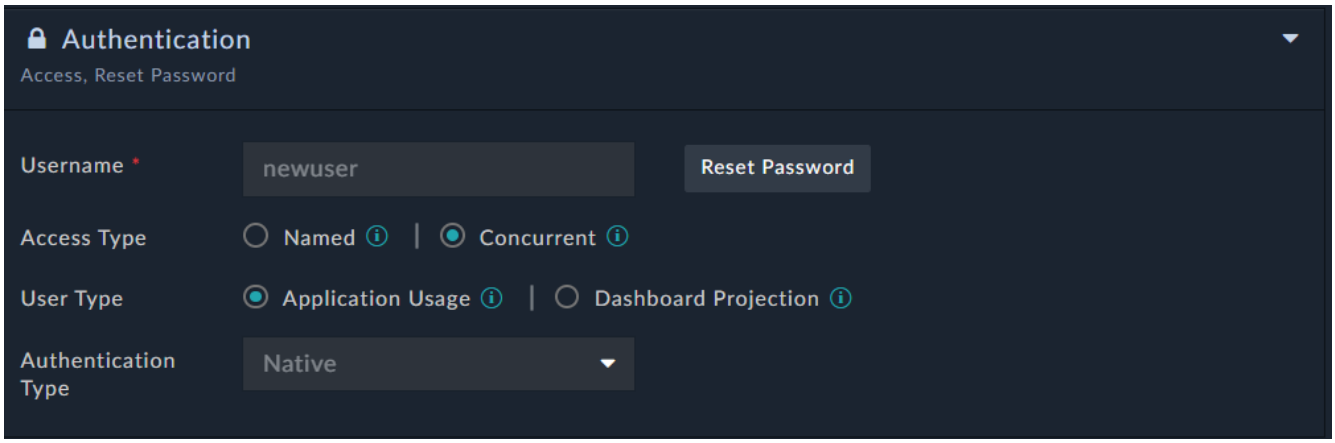
- You can assign roles to users by selecting the roles from the **Roles** table in the `Team and Role` section on the `Edit User` page. If there are more than five roles in the system, the Roles table becomes scrollable.
- You can assign teams to users by selecting the teams from the **Teams** table in the `Team and Role` section on the `Edit User` page. If there are more than five teams in the system, the Teams table becomes scrollable.

In release 7.2.1, FortiSOAR provides administrators — users with a minimum of 'Security' `Read` permissions — with the ability to view the aggregated list of effective permissions based on different roles assigned to a given user. To view the consolidated permissions list for a specific user, click **Settings** > **Users**. On the `Users` page in the `Manage Users` section, click the row of the user whose consolidated permissions you want to view. On the user's profile page, in the `Roles` section, click **View Effective Role Permissions**:

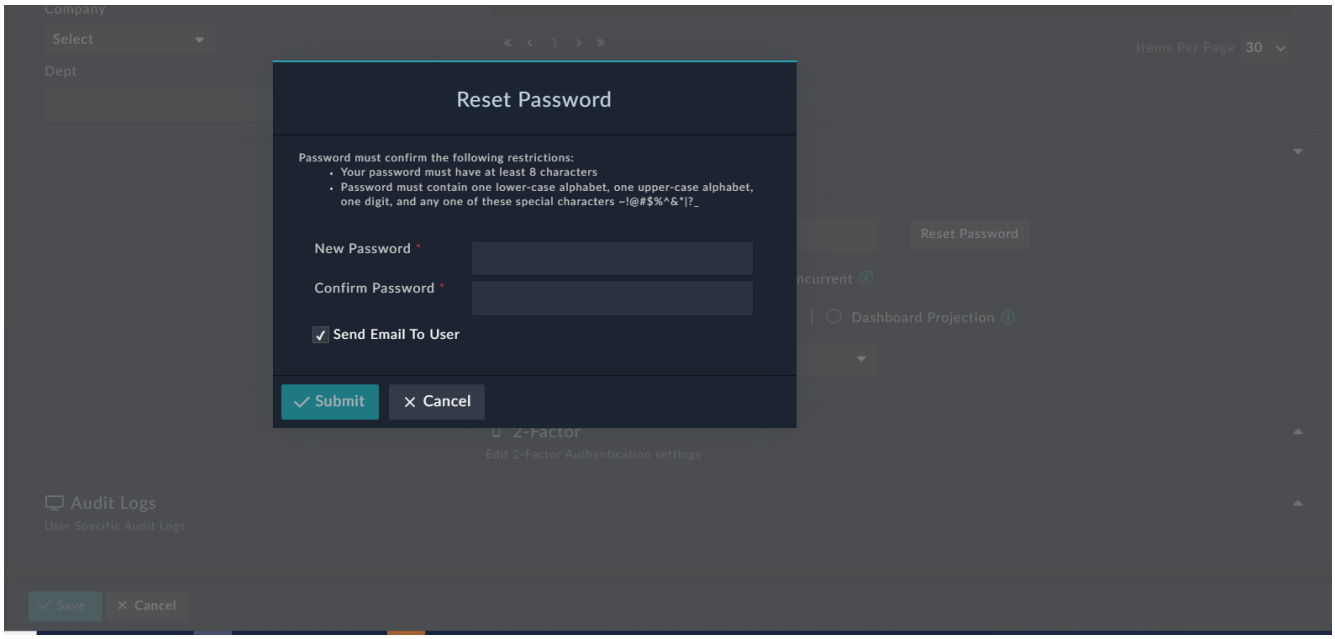


Authentication

An administrator who is assigned a role with **Read**, **Create** and **Update** permissions on the `People` module and **Read** and **Update** permission on the `Security` module can reset passwords for users on the `User Profile` page. To reset passwords, open the profile page of the user whose password you want to reset and go to the **Authentication** section.



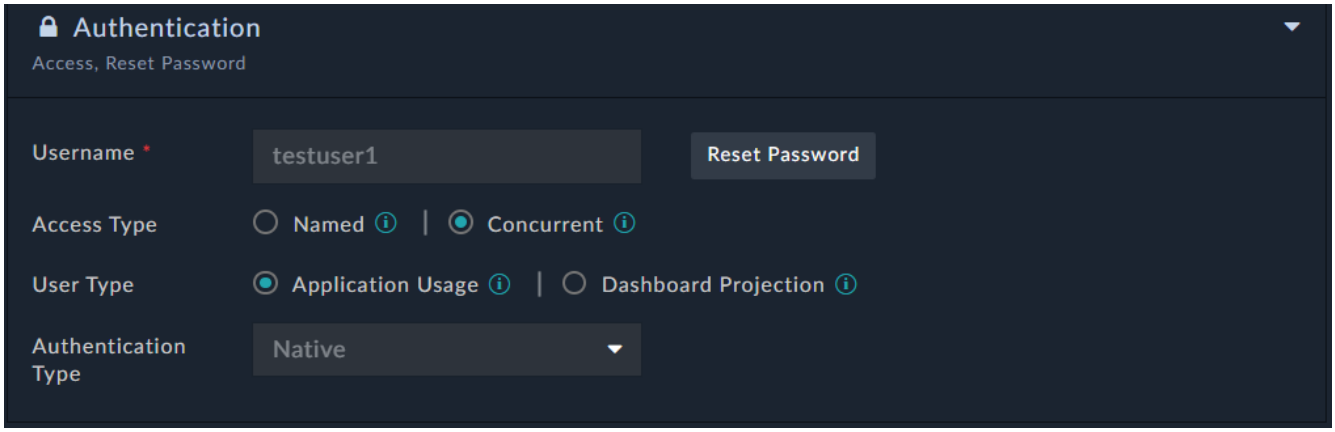
Click the **Reset Password** button to reset the password for a user. Clicking **Reset Password** displays the `Reset Password` dialog, in which you must enter the new password in the **New Password** field and re-enter the same password in the **Confirm Password** field.



Select the **Send Email to User** check box to send an email notification to the user whose password you have reset. The email notification tells the user that the administration has changed their password and the user must contact their administrator for the new password or reset their password using the Forgot Password option on the FortiSOAR login page. For more information on the Forgot Password option, see the *Regenerating your password* topic in the "User Guide." Click **Submit** to reset the users' password.

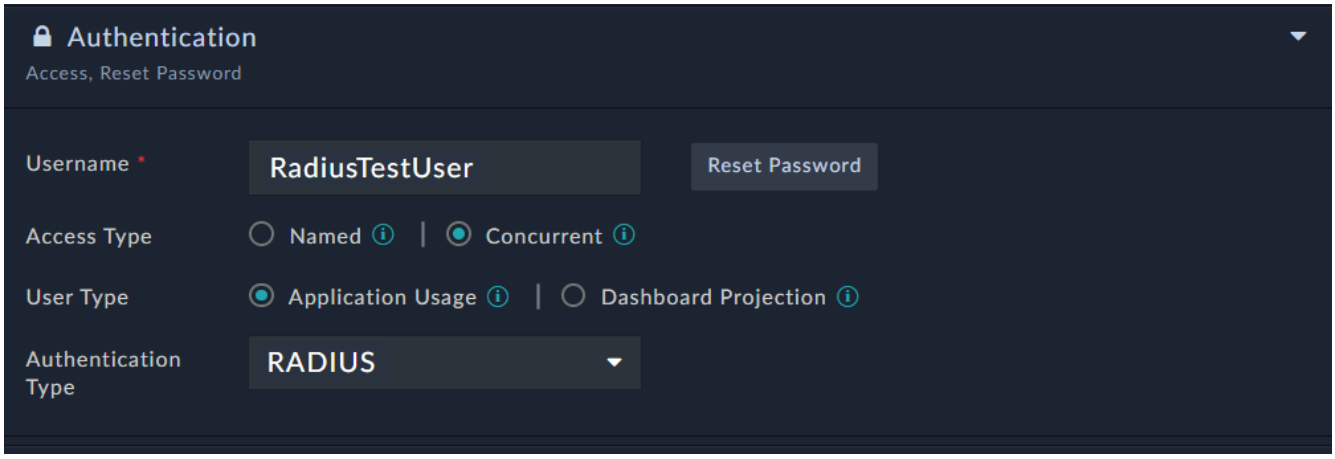
By default, users' can click the **Reset Password** link 10 times before actually resetting their password, after which users' will not get a new link to reset their password for 12 hours. A Security Administrator can change these default values, see the Debugging, Troubleshooting, and Optimizing FortiSOAR on page 415 chapter for further details.

You can configure the 'Access Type' of users', i.e., whether the user is a **Named** user or a **Concurrent** user. A 'Named' user has a FortiSOAR seat permanently reserved, i.e., such a user can always log onto FortiSOAR except in case of a license violation. However, a 'Concurrent' user can log in only when there is a concurrent seat available. You can also selectively change users' access type, i.e., Concurrent users to Named users, and vice-versa, at any time, or you can bulk change users access type from Named to Concurrent. For more information, see the Adding Users on page 100 section.



The 'User Type' is set for users based on their FortiSOAR usage. Select **Application Usage** for users who will access FortiSOAR regularly for their tasks and who regularly need to interact with the FortiSOAR system. Select **Dashboard Projection** if this user seat will be primarily used to project information on display screens such as in a control room, and supports extended idle timeout to prevent screen timeouts. Based on the user types, you can different re-authentication times, see Configuring Session and Idle timeouts for details.

In the **Authentication Type** drop-down list, you can create either a 'Native' (default) user or a 'RADIUS' user. Users with their authentication type set to **RADIUS** can log in to FortiSOAR using their RADIUS credentials:



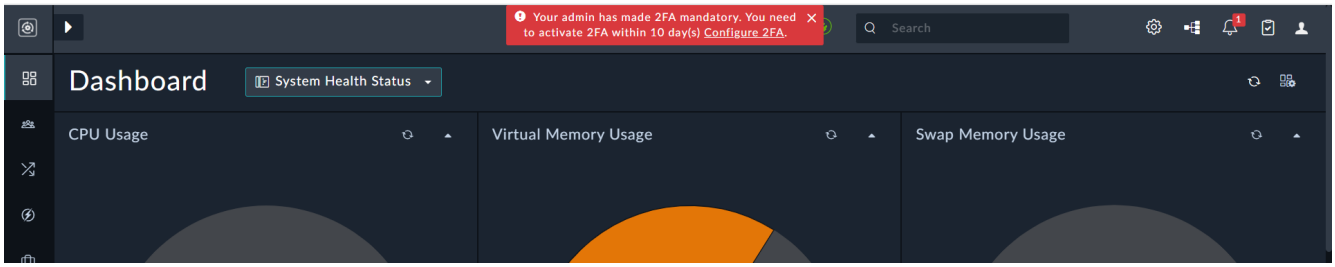
Administrators must setup RADIUS configuration before users can perform authentication. The steps for setting up RADIUS are present in the Configuring FortiSOAR authentication with a RADIUS server topic. Also, note that the username of the RADIUS user that you create in FortiSOAR must be the same as the username specified on the RADIUS server since FortiSOAR performs a lookup for the user before making the RADIUS authentication request. You can also import RADIUS users in bulk into your FortiSOAR system, details for which are present in the Importing RADIUS users in bulk topic.
While logging on to FortiSOAR, the user credentials are first authenticated against primary radius server, and if the authentication against primary server succeeds, the user gets logged in to FortiSOAR. If the connection to the primary radius server fails, then the credentials are authenticated against secondary server, and if this authentication succeeds, the user gets logged in to FortiSOAR; else the log in attempt fails with the appropriate error message.

2-Factor

The **2-Factor** authentication menu displays the 2FA method that has been setup by the FortiSOAR administrator. Currently, out-of-the-box, FortiSOAR supports TeleSign and Google Authenticator for 2FA authentication. For more information, see Configuring Two-Factor Authentication (2FA).

All FortiSOAR users must enable 2FA in their user profiles if the FortiSOAR administrator has enabled 2FA using any of the supported verification methods and mandated that it be used by all FortiSOAR users; otherwise, they are blocked from logging onto FortiSOAR. Keep in mind that there is a 10-day grace period before this enforcement becomes strict. Users who have not enabled their 2FA after the 10-day grace period are prevented

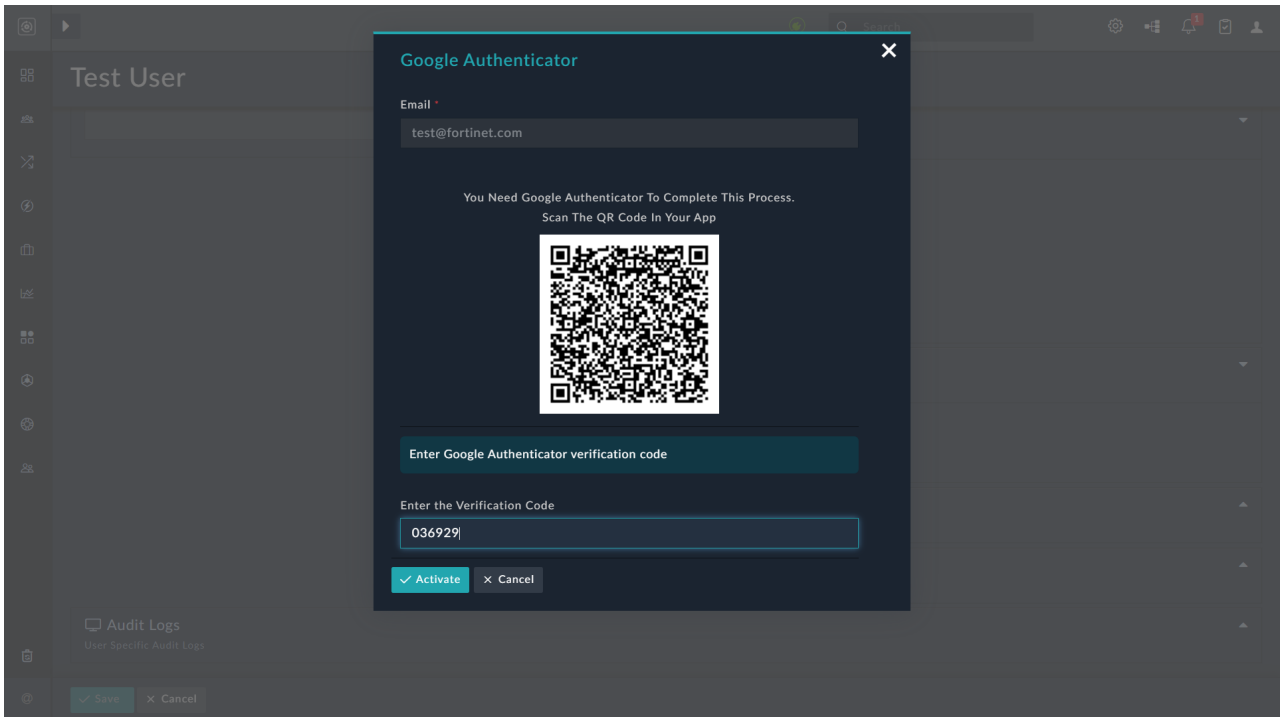
from accessing FortiSOAR. FortiSOAR displays the following warning about the grace period:



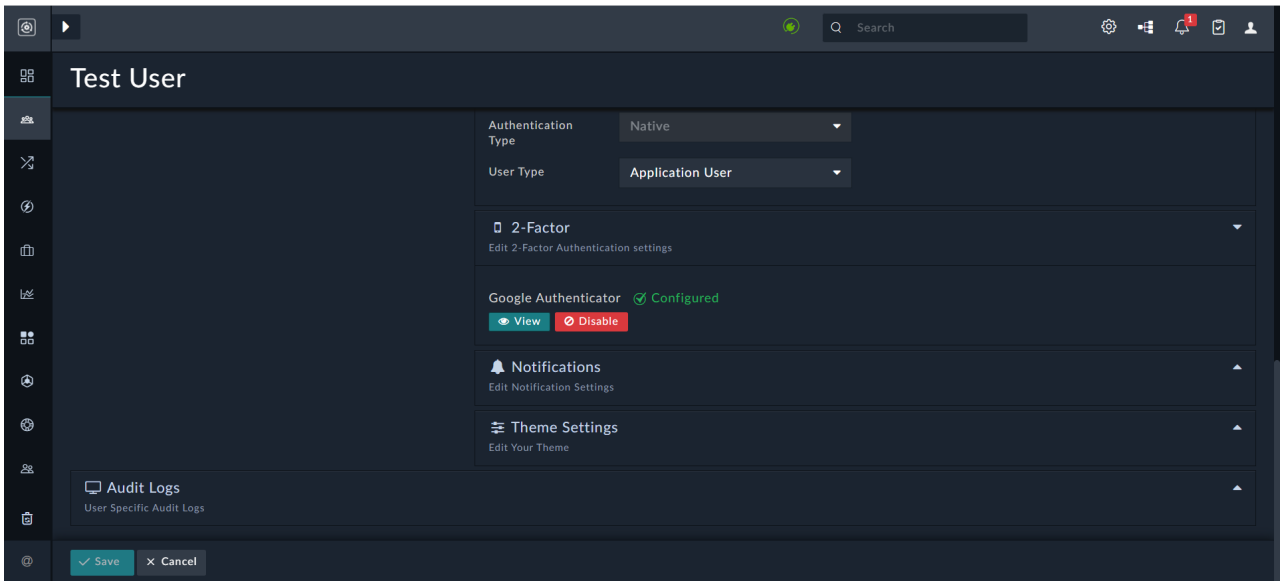
If the FortiSOAR administrator has configured Google Authenticator as the 2FA method, then in the user profile, the 2-Factor section displays Google Authenticator. To configure Google Authenticator, you need Google Authenticator to complete the verification process and send the one-time password (OTP) code to the user's device. Google Authenticator can be installed on a mobile device or any other device as this application is also available as a browser extension.

To configure Google Authenticator, do the following:

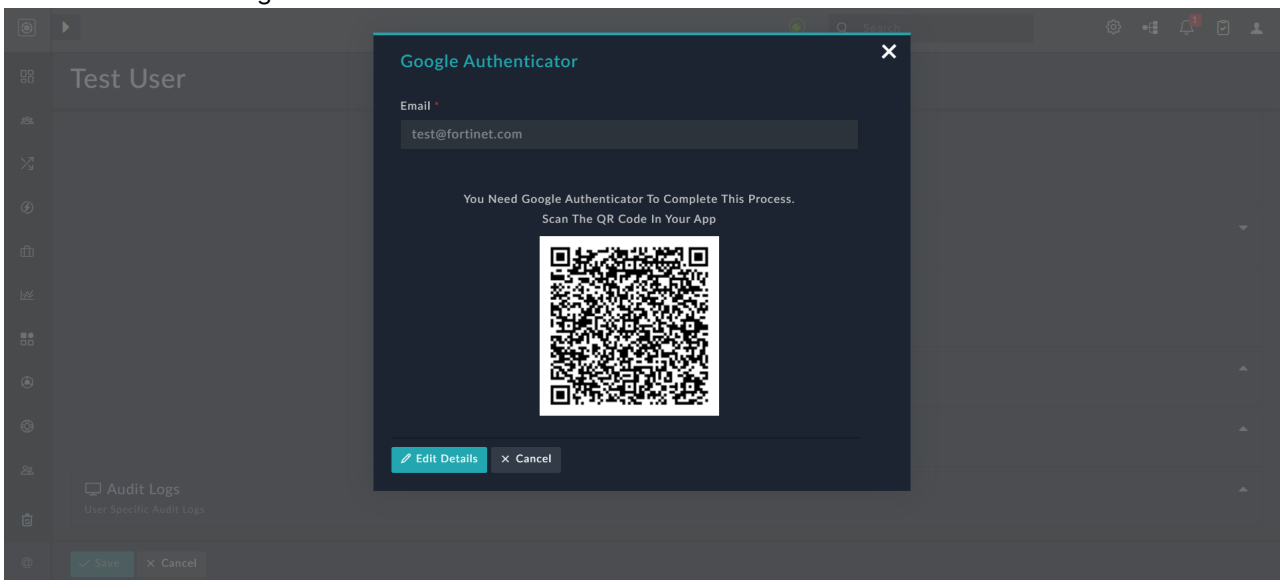
1. Click **Configure**, which displays a dialog box in which you can enter the email address to be used for setting up Google Authenticator for the user.
2. Click **Get OTP** to display a QR code.
3. Scan this QR code using Google Authenticator to complete the process and obtain the verification code.
4. In the **Enter the Verification Code** field, enter the verification code provided by Google Authenticator, and click **Activate**:



This activates 2FA for the user account:



To disable 2FA for the user account, click **Disable**, or to edit 2FA for the user account, click **View**.
NOTE: Users are prevented from disabling their configured 2FA once 2FA is mandated by their administrator.
Clicking **View** displays a Google Authenticator dialog in which you can click **Edit Details** to update the account details and again click **Get OTP**:



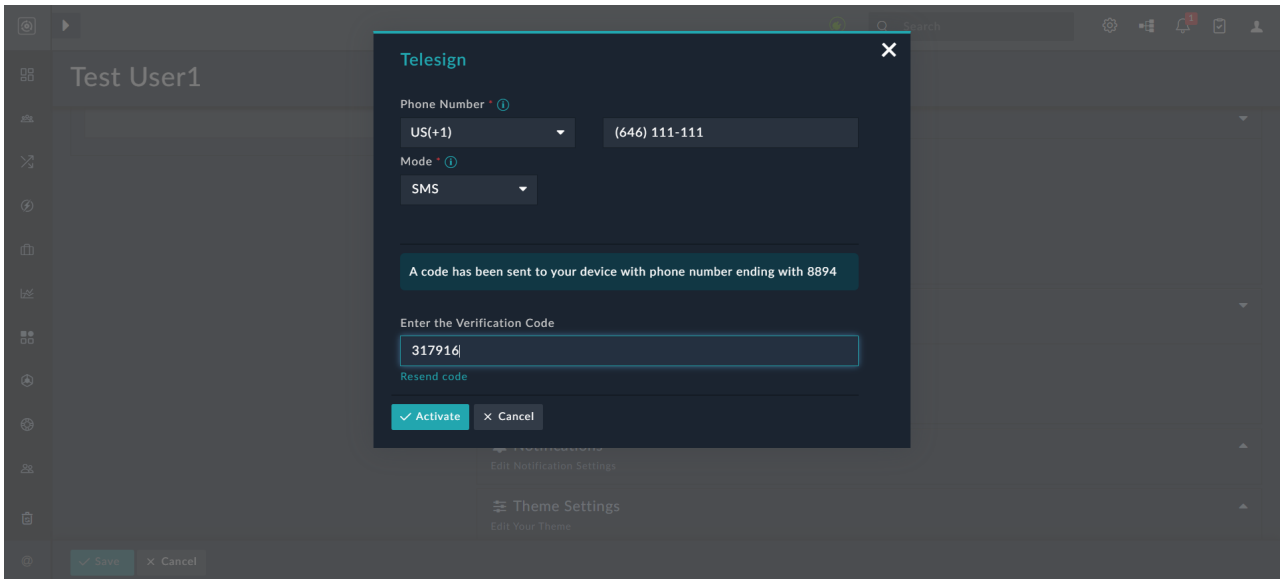
5. Once you enter the OTP in the dialog, click **Activate** to activate the 2FA configuration for the user account with the updated details.

If the FortiSOAR administrator has configured Telesign as the 2FA method, then in the user profile, the 2-Factor section displays Telesign. You need a TeleSign account to set up 2-Factor Authentication (2FA) so that users can log into FortiSOAR and receive their one-time passwords (OTPs) on their mobile devices.

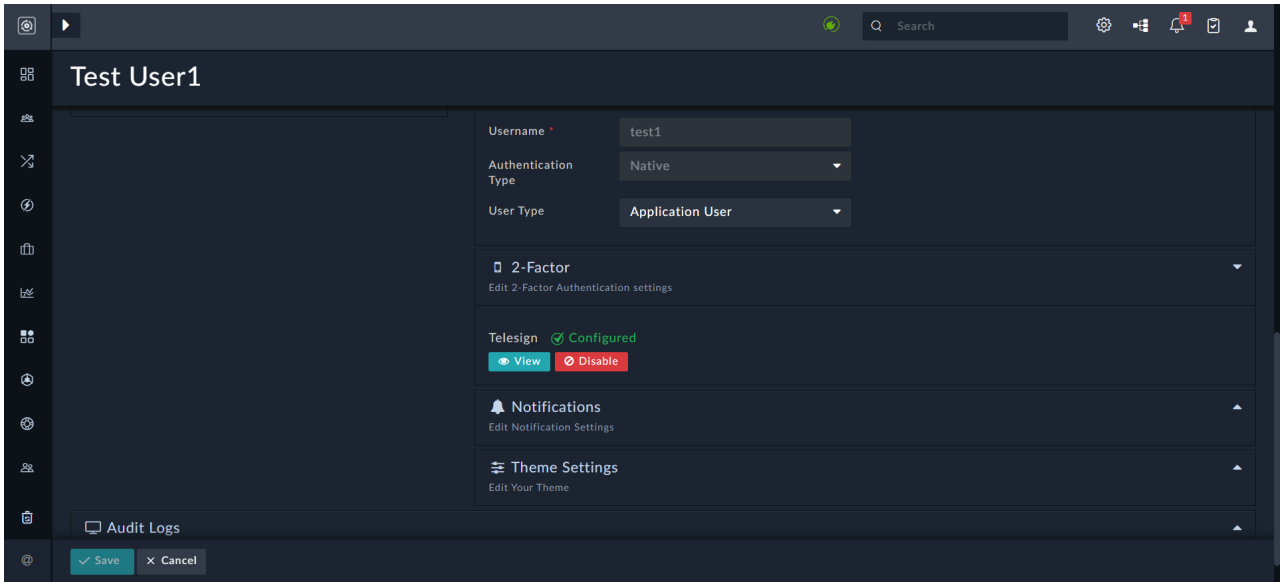
To configure Telesign, do the following:

1. Click **Configure**, which displays a dialog box with the user's workplace phone number by default. The user's Telesign account will be set up using this phone number. To set up the user's Telesign account, you can alter the default phone number and enter any other phone number.

2. Choose the method for sending the verification code to the designated phone number from the **Mode** drop-down list, and then click **Get OTP**.
3. In the **Enter the Verification Code** field, enter the verification code provided by Telesign, and click **Activate**:

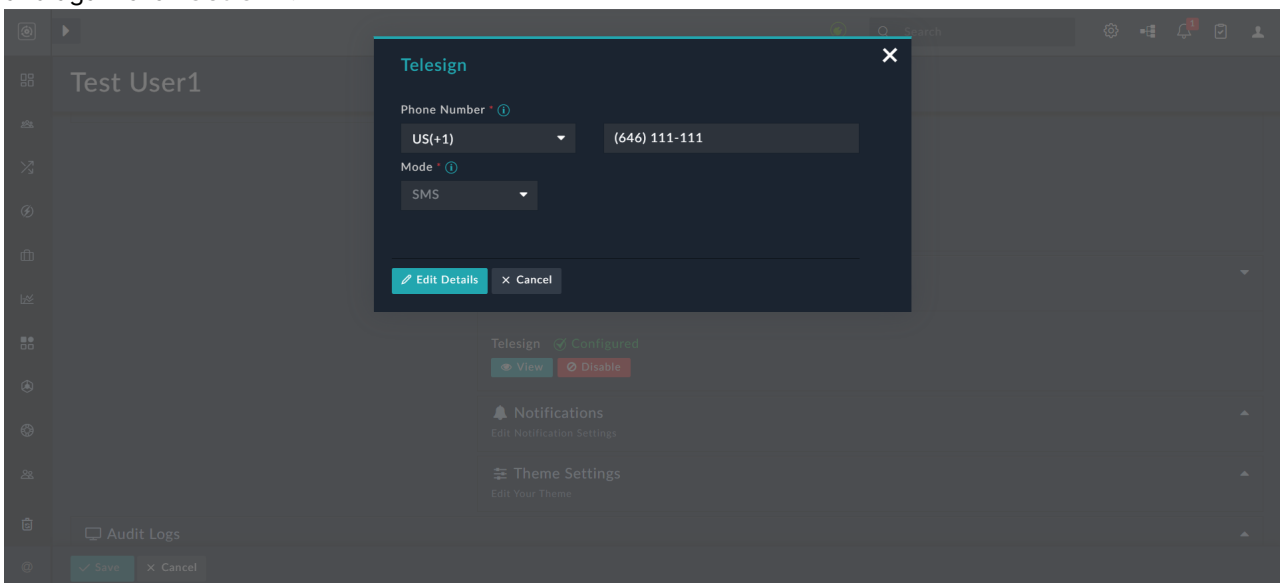


This activates 2FA for the user account:



To disable 2FA for the user account, click **Disable**, or to edit 2FA for the user account, click **View**.
NOTE: Users are prevented from disabling their configured 2FA once 2FA is mandated by their administrator.
Clicking **View** displays a Telesign dialog in which you can click **Edit Details** to update the account details

and again click **Get OTP**:



4. Once you enter the OTP in the dialog, click **Activate** to activate the 2FA configuration for the user account with the updated details.

Notifications

You can determine how you want to send notifications. To send system notifications using emails, select the **Enable System notification on email** checkbox. To send emails when a user is tagged, using @, in comments, select the **Enable email notifications for @ mentions in comments** checkbox.

Theme Settings

You can update your FortiSOAR theme, if you have appropriate rights, using the **Theme Settings** menu on the `Edit User` page. There are currently three theme options, **Dark**, **Light**, and **Space**, with **Space** being the default. Click **Preview Theme** to see the Theme as it would look and save the profile to apply the theme. To go back to the original theme, after previewing the theme, click **Revert Theme**.

History

The `History` menu contains the authentication history for the user and displays the ten most recent authentication attempts and their outcome.

Audit Logs

The `User Specific Audit Logs` panel displays a chronological list of all the actions that a user has performed across all the modules of FortiSOAR. Click the **Audit Logs** panel to view the list of actions. The audit log displays users' login success or failures and logout events. The login event includes all three supported login types,

which are DB Login, LDAP Login, and SSO Login. Audit logs also contains user-specific terminate and resume playbook events.

Managing Access Keys

Automation can utilize HMAC authentication or API key-based authentication (from release 7.5.0 onwards). API key authentication is also beneficial in outbound Threat Intelligence Management feed distributions, particularly for clients such as firewalls that only support basic authentication.

Access keys are also subject to the same authorization model as users, meaning that you must add the created access keys to a team and assign them roles to perform any actions within the system. Access keys have a few key differences compared to Users (People); the most significant one being that access keys use either HMAC verification or API keys for access purposes instead of using a token issued from the Authentication Engine. The Authentication Engine uses the HMAC signature to validate the Public and Private key pair or the API key that is issued at the time of creating access keys. Access keys also do not have a login ID and do not count towards your license limit.

API keys or appliances are generally used for authenticating to FortiSOAR while calling Custom API Endpoint triggers, for example, while configuring auto-forwarding of events and alerts from a SIEM to FortiSOAR, you can use an API key or appliance, otherwise you might require to add a user password, in plain text, in the configuration files.

Similar to users, you must assign appropriate roles to access keys and add access keys as members of the teams that would be running playbooks, so that appliances can access or modify any data within the system. Team hierarchy and other restrictions that apply to users also apply to access keys.

As a good security practice, we recommend that you limit the role and team of an access key to provide only the minimum level of privilege needed to perform its operations.

You can directly use the API key to perform various operations using APIs, see the *Access Keys* chapter in the "API Guide."

To perform various operations using the API key, you must be assigned appropriate permissions on the `Appliances` module. Also note that the API keys cannot be used to perform all operations related to the API key such as creating or updating API keys and all authentication (`/auth`) operations.

Managing API key-based authentication for appliances

An API key is an alphanumeric string that is commonly used to secure and manage access to APIs. It serves as a token for authentication, allowing control of access and tracking of API usage.

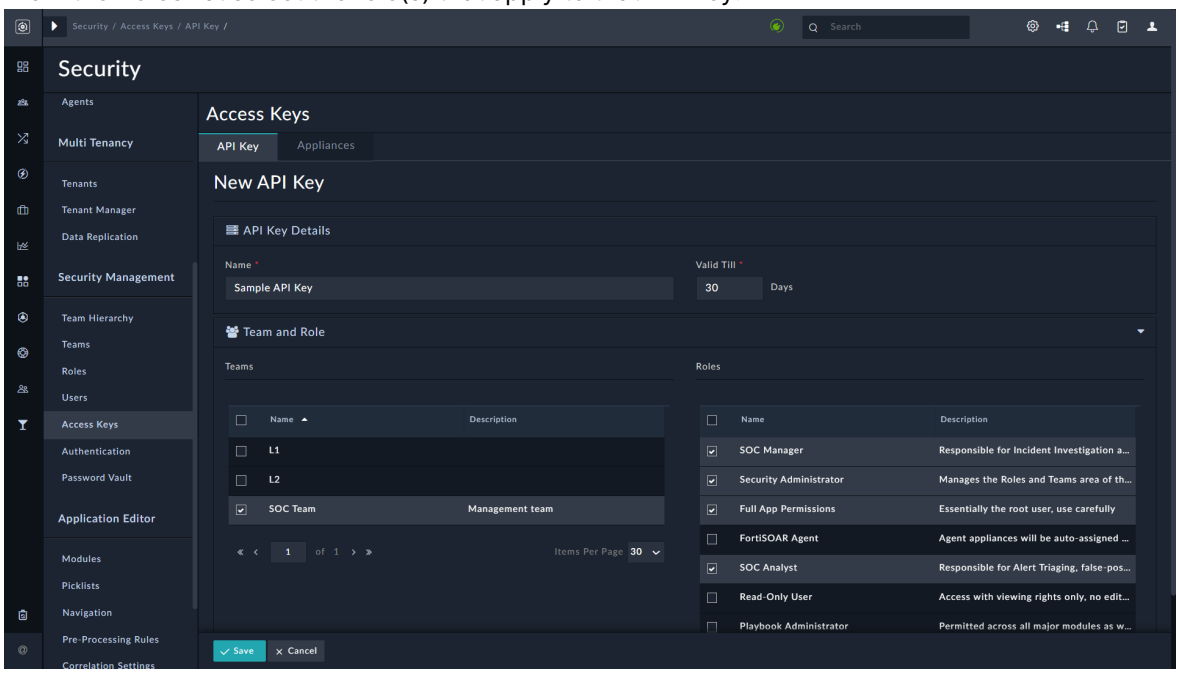
From release 7.5.0 onwards, FortiSOAR supports API key-based authentication in addition to HMAC-based authentication.

The 'API key' records cannot be owned by a team or a user.

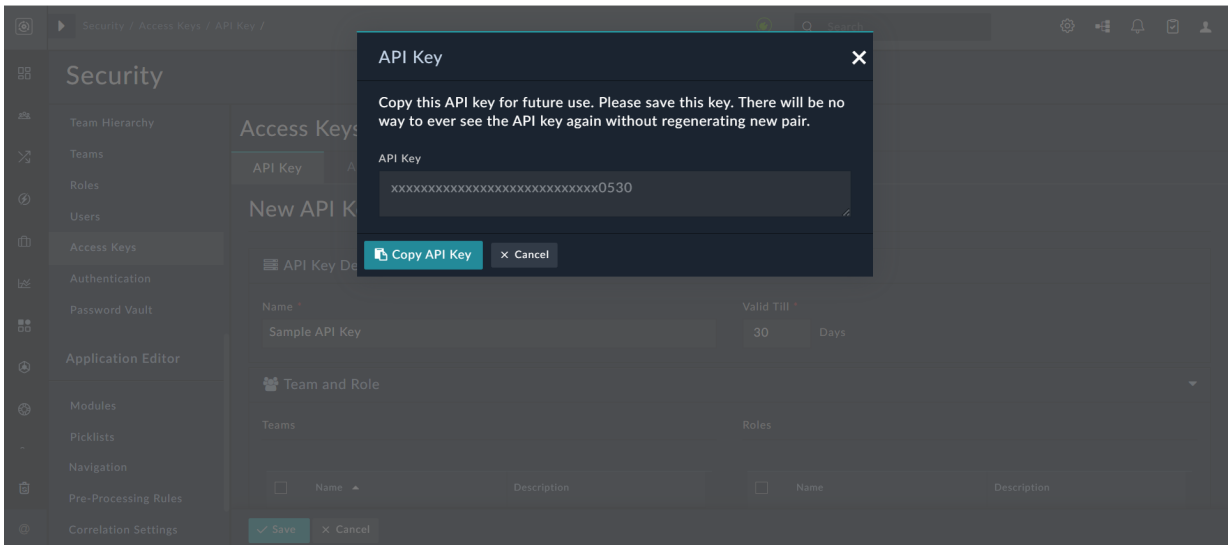
Creating a new API key

To create a new API key, follow these steps:

1. Click **Settings** > **Access Keys**. Click the **API Key** tab and click **Add** to create a new API key.
2. On the `New API Key` page, enter the required details:
   a. In the **Name** field enter a name to identify the API key.
   b. In the **Valid Till** field, specify the number of days until the API key is valid.
   NOTE: The minimum number of days that you can specify for the validity of an API key is 1 day and the maximum validity that you can specify is 365 days. It is recommended to avoid using the same API key an extended period to reduce vulnerability. Keys should be generated periodically to mitigate the risks of unauthorized access.
   c. You need to associate the API key with appropriate Teams and Roles to access and use any information from an FortiSOAR system. To assign the teams and roles to the API key, do the following:
      i. In the `Team and Role` section, from the **Teams** list select the team(s) that apply to that API key.
      ii. From the **Roles** list select the role(s) that apply to that API key.



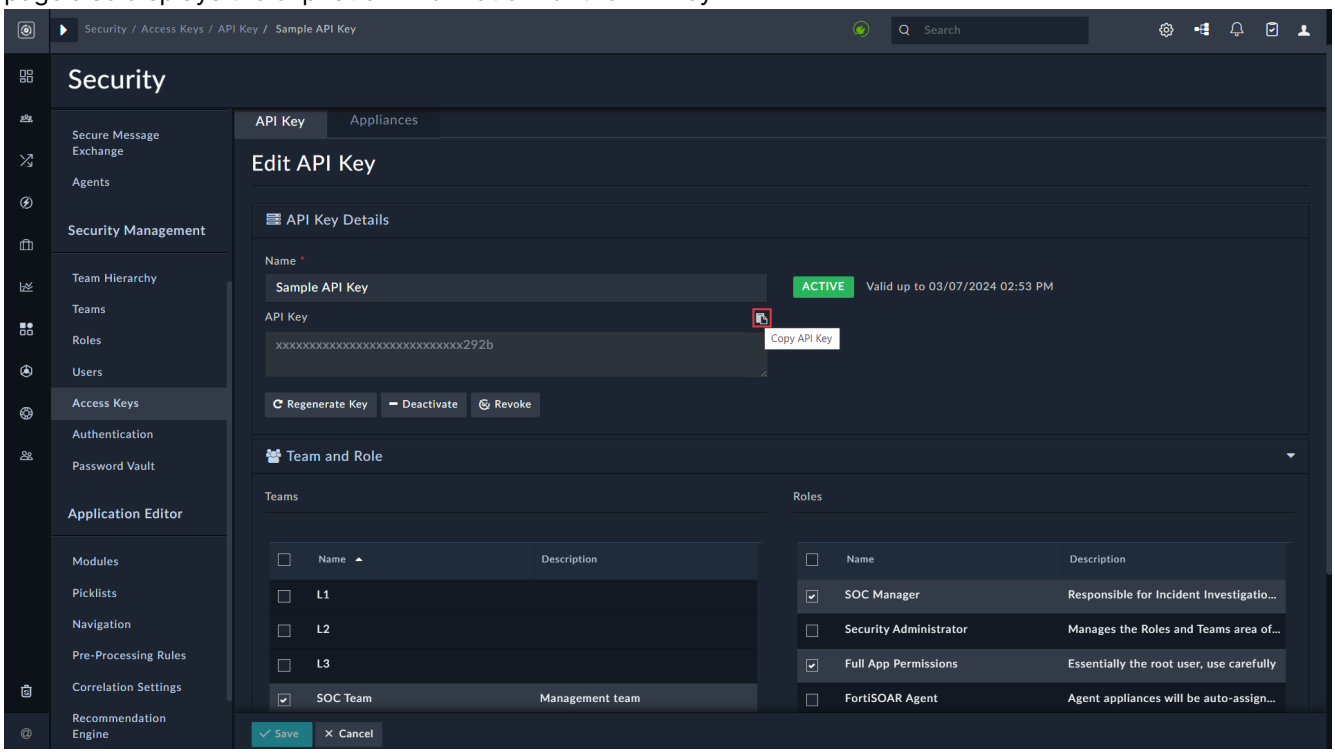
   d. Click **Save** to generate a new API key.
   On clicking **Save**, FortiSOAR displays the masked API key in a modal window:



IMPORTANT: The retrieval mode set by the administrator when the API key is created determines whether the API key can be retrieved. By default, the retrieval mode is disabled, and the API key is displayed only once; therefore, you must click **Copy API Key** to save it for future reference. For information on the retrieval mode setting, see the Setting the API key retrieval mode topic.

Operations supported for an API key

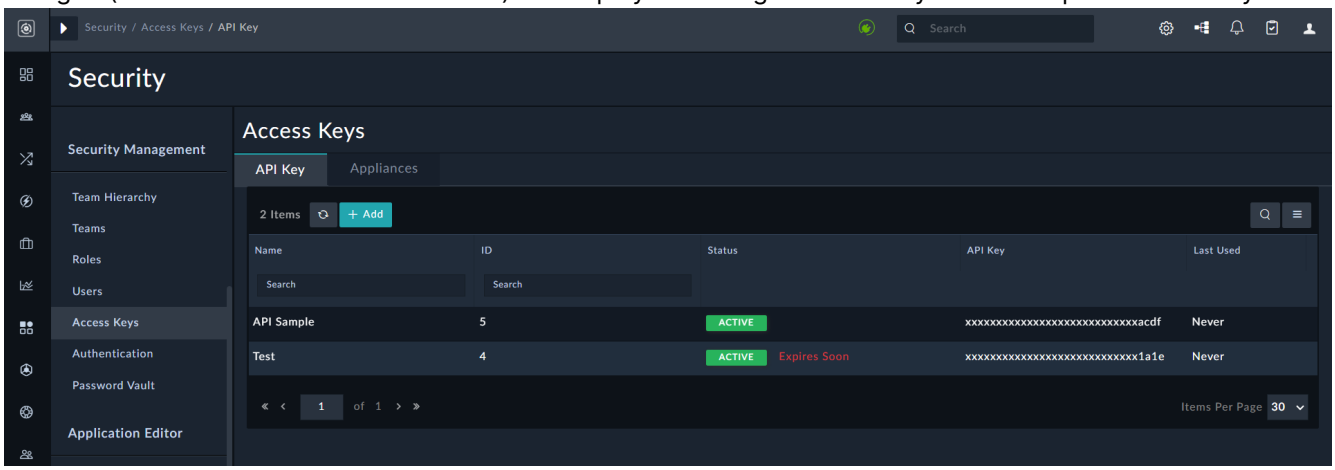
You can retrieve, reset validity, regenerate, deactivate, or revoke an API key on the `Edit API Key` page. This page also displays the expiration information for the API key:



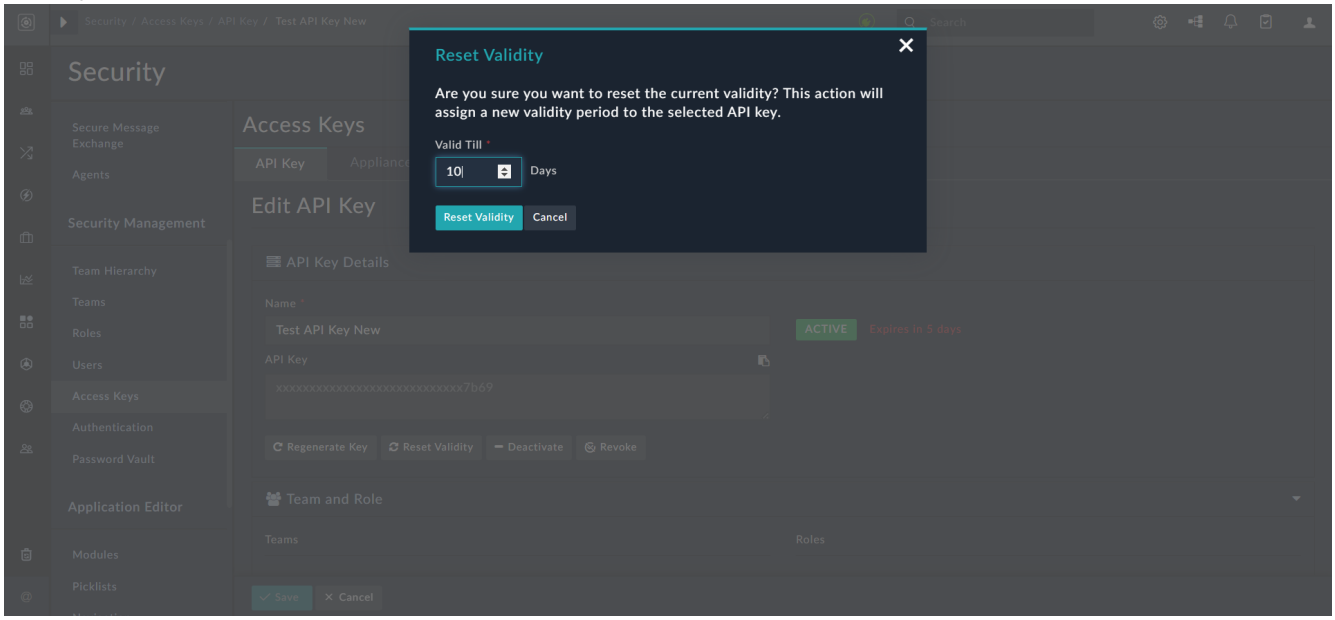
To access the `Edit API Key` page, click **Settings** > **Access Keys**. Then, click on the row containing the API key that you want to retrieve, reset validity, regenerate, deactivate or revoke.

To retrieve the API key, click **Copy API Key**, which copies the API key to the clipboard. The administration can determine whether an API key is retrievable, and only when the administrator has enabled the API retrieval mode, will you be able to see the Copy API Key option. For information on the retrieval mode setting, see the Setting the API key retrieval mode topic.

Users are not notified when any API key is about to expire; they need to check the API key grid to see its status. The grid (and detail view of the API record) will display a warning if the API key is set to expire within 7 days:



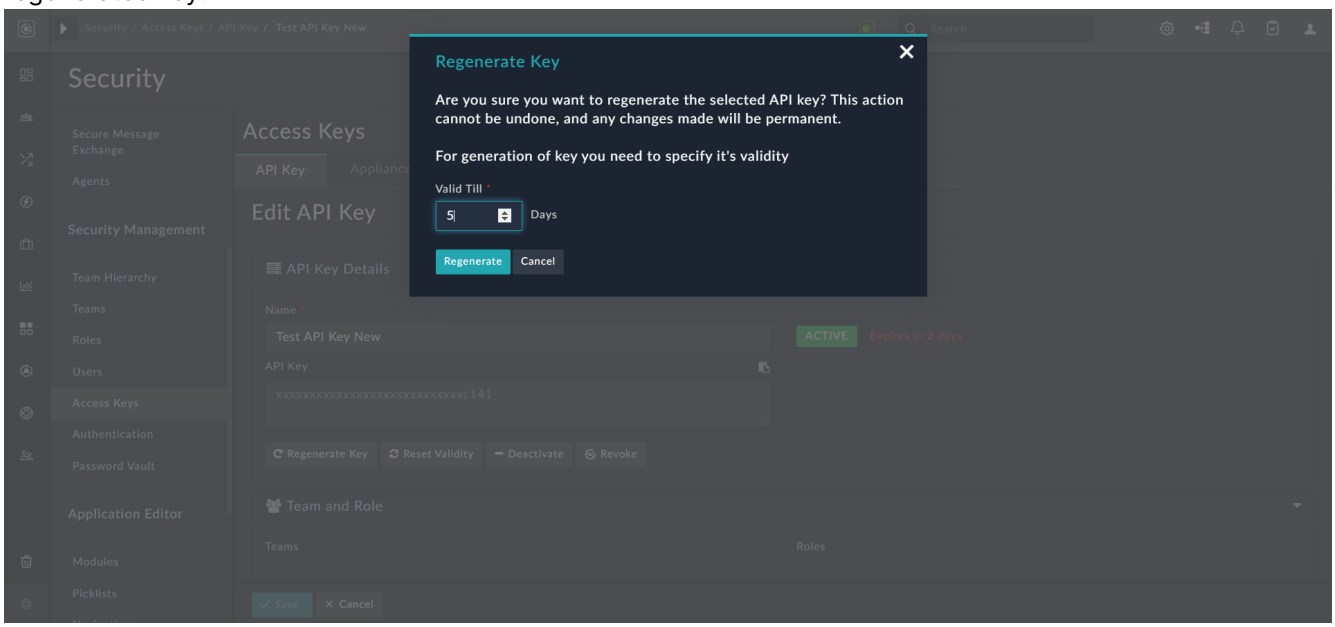
The option to reset the validity of an API key is only available if the current validity is set to expire within 7 days. To reset the validity, click on the row of the API key to display the `Edit API Key` page, and then click **Reset Validity** to display the `Reset Validity` dialog in which you can specify the new validity period for the selected API key:



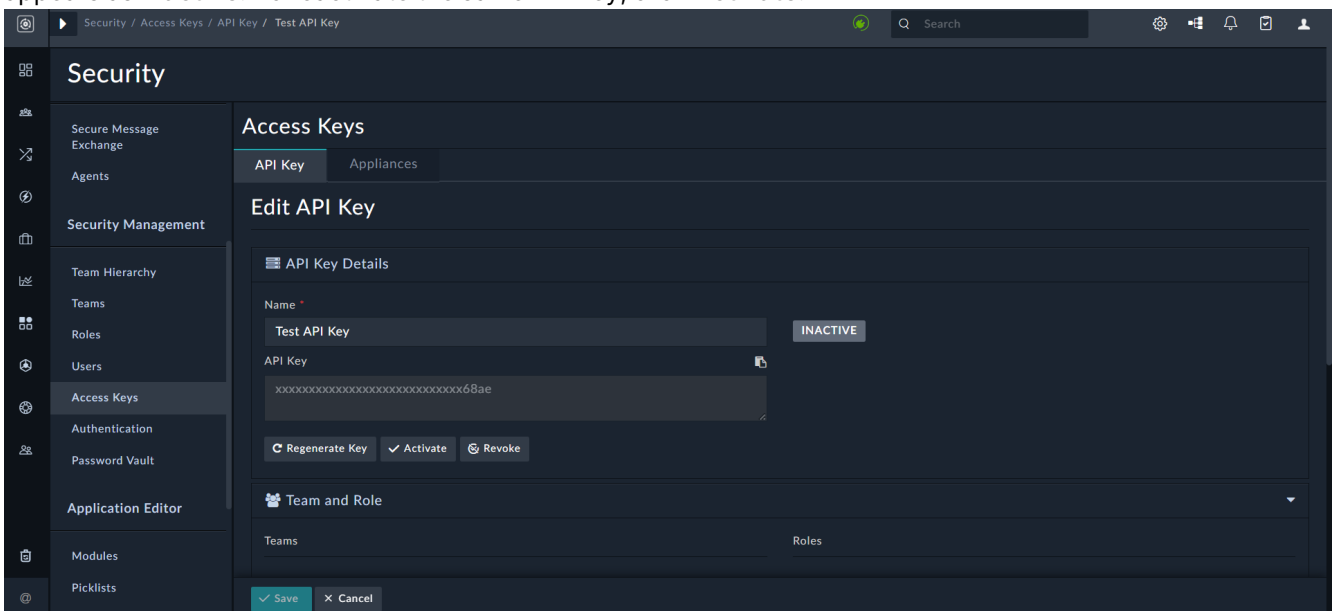
You might need to regenerate an API key, if for example, the API key is lost or the user did not save the API key when it was generated. To regenerate the API key, click **Regenerate Key**.

Regenerating an API key is an irreversible operation. Once you regenerate an API key, the previous keys are invalidated.

Clicking **Regenerate Key** displays the `Regenerate Key` dialog where you must specify the validity of the regenerated key:

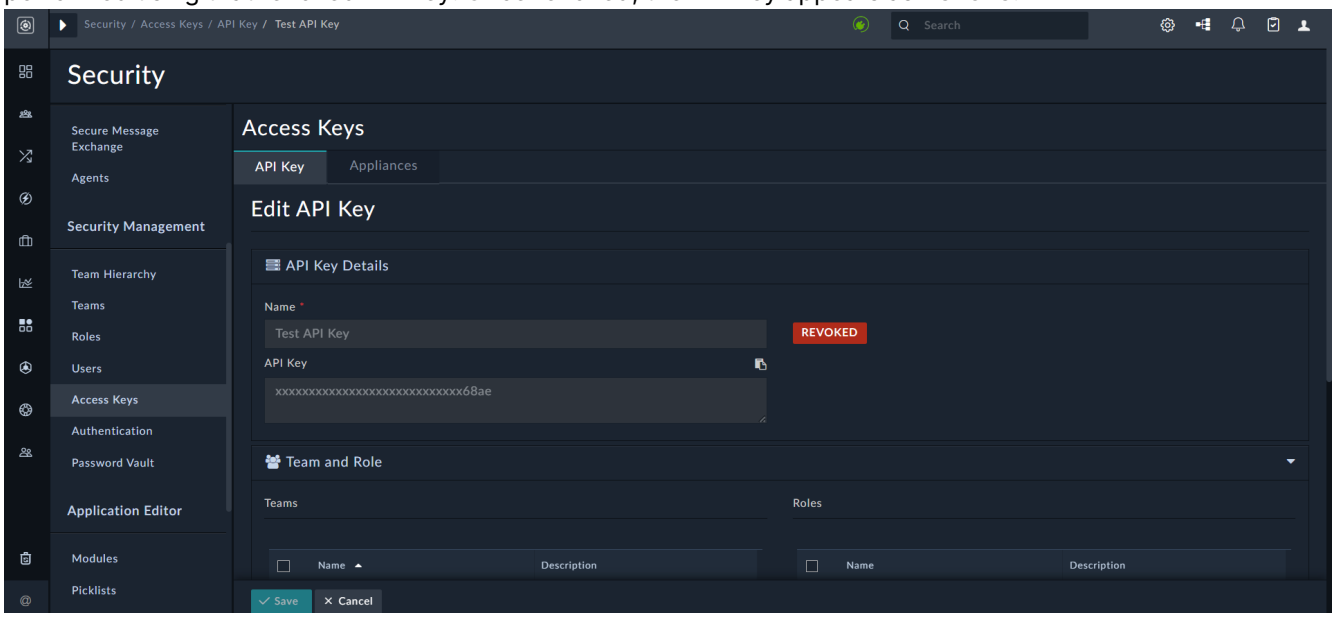


If you want to deactivate an API key for a specific period, click **Deactivate**. Once deactivated, the API key appears as **Inactive**. To reactivate the same API key, click **Activate**:



If you want to permanently deactivate an API key and revoke its access, click **Revoke**. Note that 'Revoke' is an irreversible operation, and it makes the API unusable permanently and no further operations are allowed to be

performed using that revoked API key. Once revoked, the API key appears as **Revoke**:



To use an API key, the API key must be placed in the 'Header' with the Authorization key and the format for identifying the user is of type API_KEY %api_key%:

```
{
        'Authorization' : 'API-KEY <Here goes the api key>'
}
```

For more information on API key, see the *Access Keys* chapter in the "API Guide."

Managing HMAC authentication for appliances

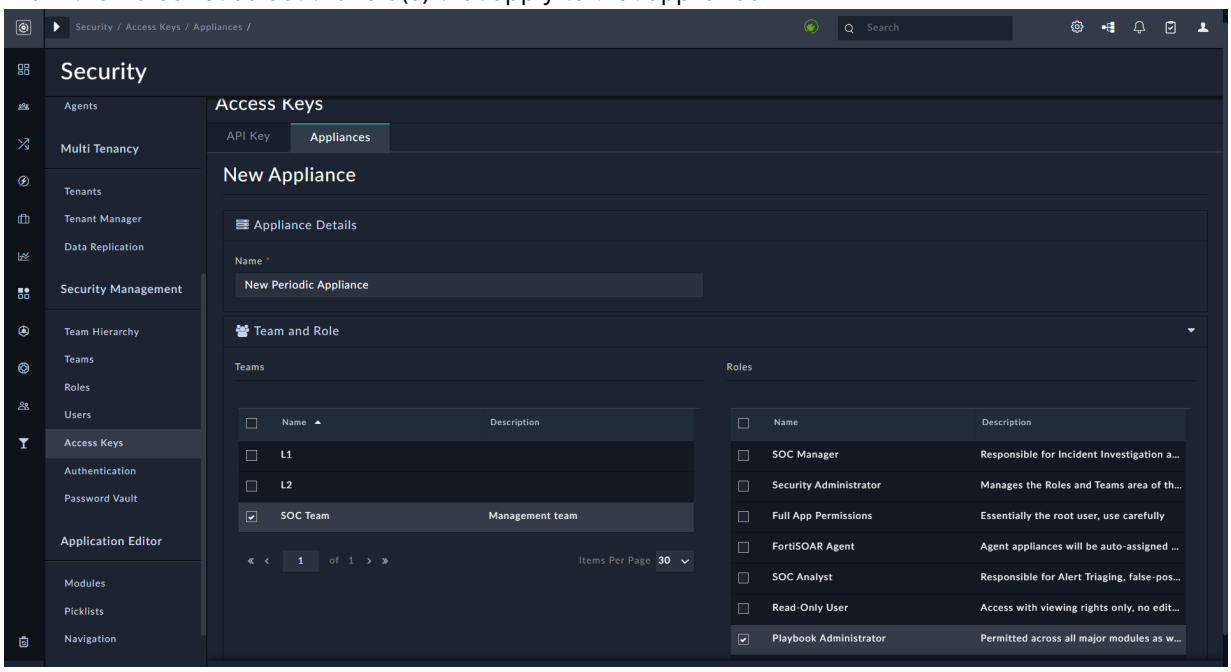
Automation can utilize HMAC authentication or API key-based authentication (from release 7.5.0 onwards).

Creating a New Appliance

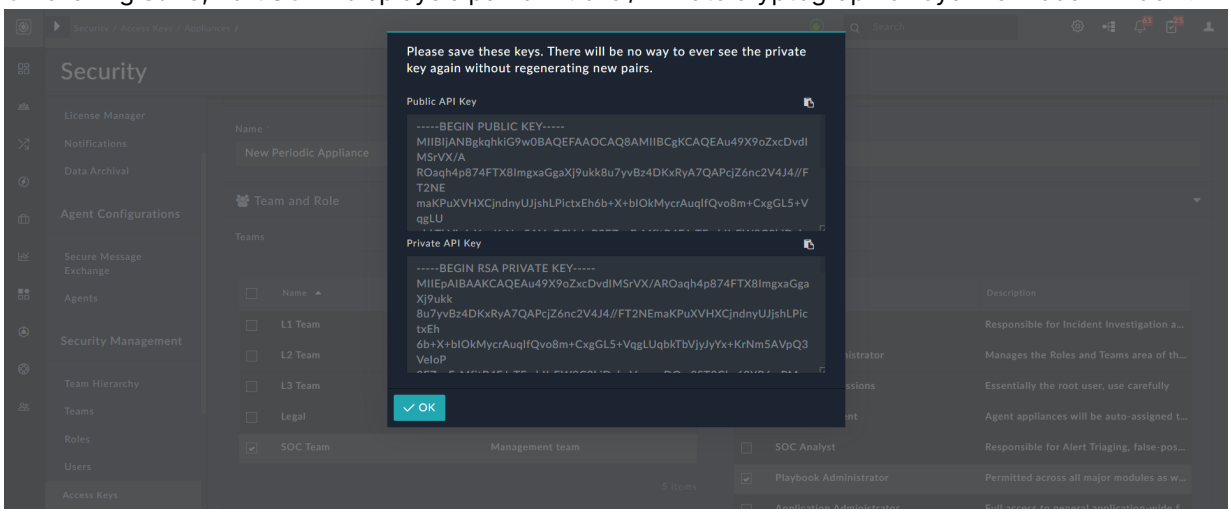
To create a new appliance, follow these steps:

1. Click **Settings** > **Access Keys**. Then, click the **Appliance** tab and click **Add** to create a new appliance.
2. On the `New Appliance` page enter the required details:
   a. In the **Name** field enter a name to identify the appliance.
   b. In the `Team and Role` section, from the **Teams** list select the team(s) that apply to that appliance.

c. From the **Roles** list select the role(s) that apply to that appliance.



d. Click **Save** to generate a new Appliance.
On clicking **Save**, FortiSOAR displays a pair of Public / Private cryptographic keys in a modal window:

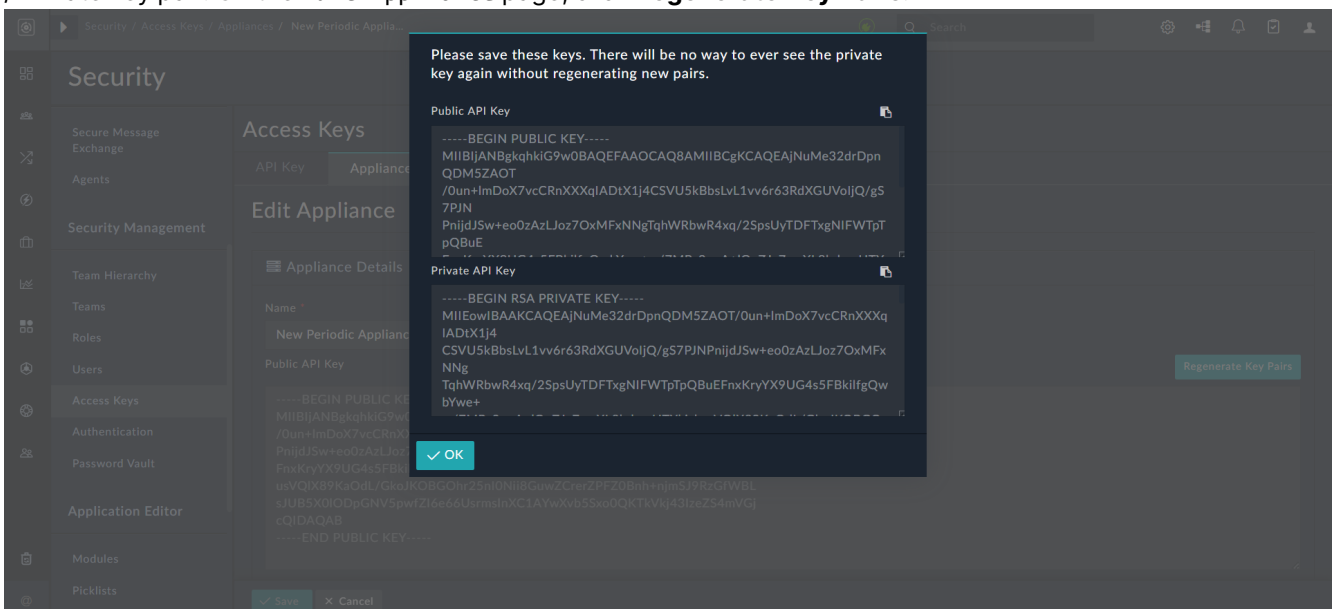


IMPORTANT: When the Public / Private key pair is generated, the Private key is shown only once. You must ensure to copy this key and keep it somewhere safe for future reference. If you lose this key, it cannot be retrieved.

Regenerating Appliance Keys

To regenerate a Public / Private key pair for an appliance, click **Settings** > **Access Keys** and then select the **Appliances** tab. On the `Appliances` page, click the row of the appliance for which you want to regenerate Public

/ Private key pair. On the `Edit Appliance` page, click **Regenerate Key Pairs**:



To regenerate appliance keys from the CLI for all system appliances, use the following command:
`/opt/cyops-auth/.env/bin/python /opt/cyops-auth/defaultappliance.py generate-appliance-keys`

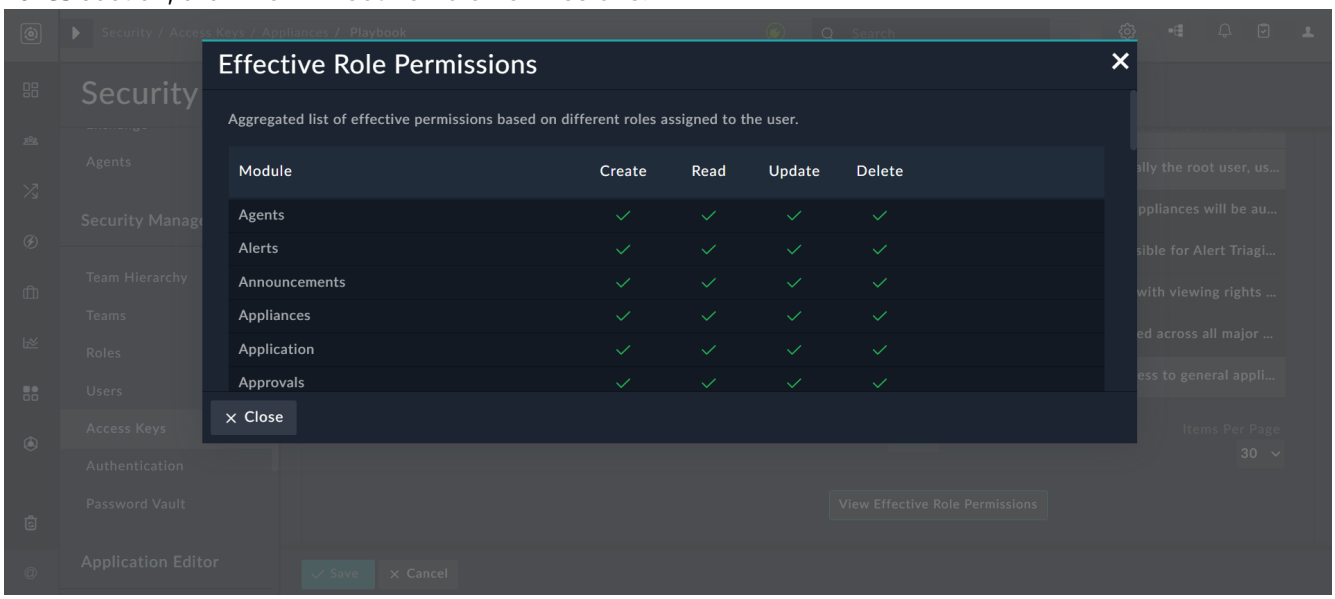
To regenerate an appliance key from the CLI for a specific appliance, use the following command:
`/opt/cyops-auth/.env/bin/python /opt/cyops-auth/defaultappliance.py generate-appliance-keys --uuid <uuid_of_the_specific_appliance>`

Appliance Profile

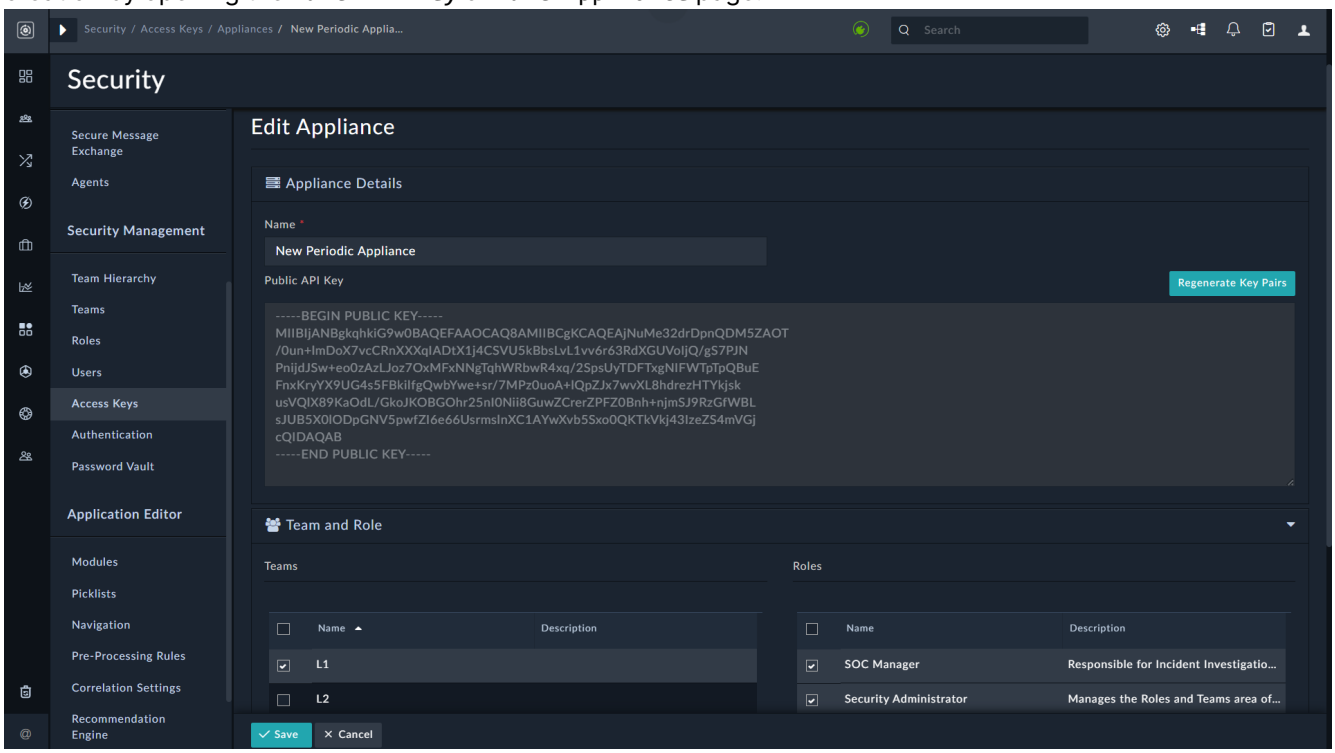
FortiSOAR allows administrators — users with a minimum of 'Security' `Read` permissions — to view the aggregated list of effective permissions based on different roles assigned to a specific appliance.

To view the consolidated permissions list for a particular appliance, click **Settings** > **Access Keys** and then select either the **API Key** or **Appliances** tab. On the `API Key` or `Appliances` page, click the row of the appliance for which you want to view the consolidated permissions. On the `Edit Appliance` or `Edit API Key` page, in the

`Roles` section, click **View Effective Role Permissions**:



A 'Security Admin' can also modify details such as the teams and roles of API keys or appliances after their creation by opening the `Edit API Key` or `Edit Appliance` page.

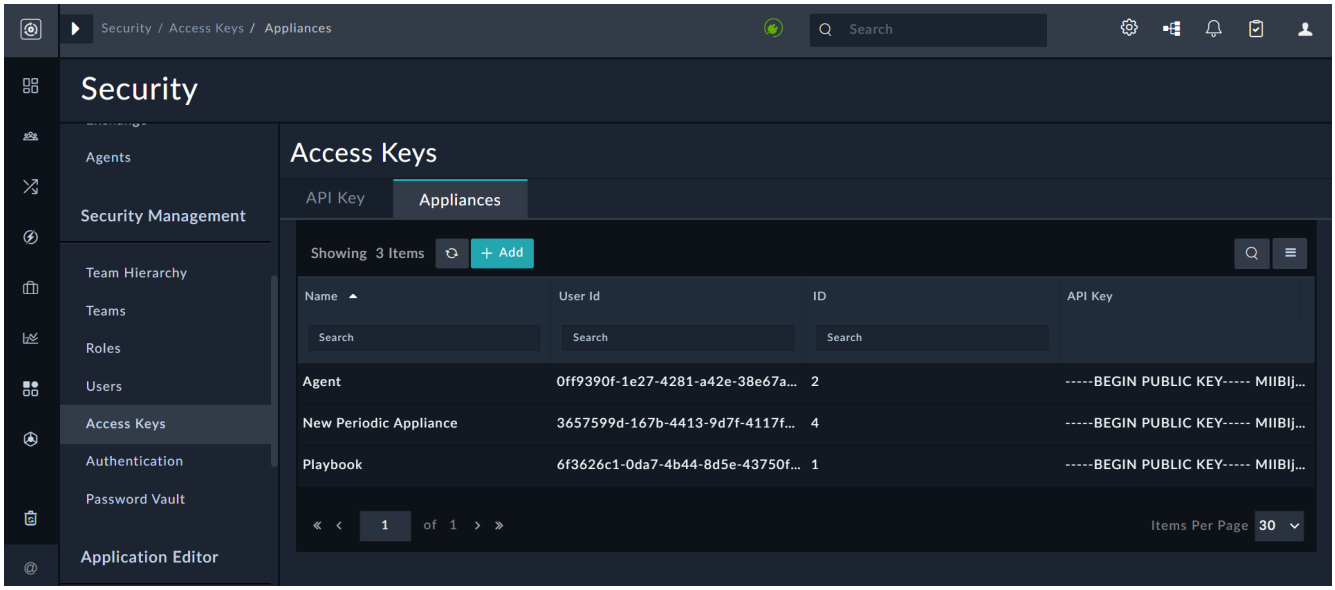


Playbook Appliance

By default, a user named `Playbook` is created as an appliance user and is assigned to the `SOC Team`. This appliance is utilized by the FortiSOAR workflow service to authenticate to the API service when a workflow step

is run to read, create, update, or delete records. Therefore, Therefore, it requires all necessary permissions on the modules accessed by playbooks. Additionally, when a record is inserted by a workflow, such as a Playbook or a Rule, the inserted record is owned by the teams associated with the appliance that created the record. For example, if a playbook or workflow inserts a new incident record, then the `Created By` field of this record displays the name of the appliance or API key that has executed the playbook, which by default is `Playbook`, and the record will be owned by the team assigned to the appliance, which by default is `SOC Team`. If multiple teams are assigned to the appliance, then all of those teams will be listed as 'owners' of this newly inserted record. Example to explain this is, if you have created an appliance named `QA` that is been assigned `SOC Team` and `Team A` as its teams. Now if a playbook that inserts an alert record is executed using the **QA** appliance, then the `Created By` field of this newly inserted alert record will display `QA` and its owners will be `SOC Team` and `Team A`.

As a good security practice, we recommend that you limit the role and team of a Playbook Appliance to provide only the minimum level of privilege needed to perform its operations.



You must however assign the new playbook appliance with a minimum of `Read` permission on the `Playbook` module to allow a new appliance user to run playbooks without encountering permission denied errors. Additionally, appropriate permissions must be assigned on other modules such as `Alerts`, based on the playbooks intended to be run using the appliance.

Troubleshooting

Getting an HMAC failure

Resolution: If an HMAC failure occurs, ensure that the server time for the application server is synchronized with that of the FortiSOAR server. You can synchronize both servers to a common NTP server, such as time.apple.com, to ensure time synchronization.

Configuring Authentication

Click **Settings** > **Authentication** to configure various authentication settings in FortiSOAR, including setting session and idle timeouts, settings options for user accounts, setting for API key retrieval, configuring LDAP / AD, configuring SAML to enable users to use sign-on (SSO), and configuring authentication with RADIUS server.

FortiSOAR supports the following methods of authentication: Database users, LDAP users, and SSO.

Even if you configure SSO, you can still provision database and LDAP users.

To configure authentication settings, you must be assigned a role that at a minimum has **Read** and **Update** permissions on the `Security` module.

Configuring Accounts

Configuring Session and Idle timeouts

Click **Settings** > **Authentication** to open the `Account` page. On the `Account` page, in the `Session & Idle Timeout` section, you can configure the following settings for session and idle timeouts:

Setting	Description
Idle Timeout	The number of minutes an 'Application Usage' user can be idle on FortiSOAR after which the 'Idle Warning' dialog is displayed. The default value is 30 minutes. For user types, see the Authentication topic.
Idle Timeout Grace Period	The number of seconds an 'Application Usage' user is given to view the 'Idle Warning' dialog after which FortiSOAR logs the user out. The default value is 60 seconds.
Token Refresh	The number of minutes after which the session token is refreshed. The default value is 60 minutes, and recommended value is 30 minutes. **Note**: The token refresh time must always be set to less than 120 minutes. This is needed as the maximum token alive time is 120 minutes, before which the token must be refreshed.
Reauthenticate Dashboard Projection User	The number of hours after which a dashboard projection user is forced to be re-authenticated. The default value is 24 hours.
Reauthenticate Application Usage User	The number of hours after which an application usage user is forced to be re-authenticated. The default value is 24 hours.

Notes:

- In the case of the **Token Refresh** option, users do not get logged out from the FortiSOAR UI; instead, the token gets refreshed automatically once the session token refresh time is reached.
- In the case of the **Reauthenticate Dashboard Projection User** and **Reauthenticate Application Usage User** options, users are automatically logged out of the FortiSOAR UI once the reauthentication time is reached. To log back into FortiSOAR, users need to re-enter their credentials.

Enabling custom password policies for users configured with Basic Authentication

When a new password is set up it must contain the following:

- At least 8 characters
- one lower-case alphabet
- one upper-case alphabet
- one digit
- Any one of the following special characters ~ ! @ # $ % ^ & * | ? _

Apart from the above default rules, you can also set up custom password policies, which enforces the following additional restrictions on the passwords that users can create:

- Password must not be one of 10 previous passwords.
- Password must not contain the username of the user.
- Password must not have been changed in the last 1 day.

By default, the custom password policy is disabled. If you want to enable the custom password policy, you need to do the following on your FortiSOAR instance:

1. Edit the `das.ini` file as a *root* user using an SSH session:
   `vi /opt/cyops-auth/utilities/das.ini`
2. Add the`[PASSWORD]` section at the end of the `das.ini` file as follows:
   ```
   [PASSWORD]
   validator = custom
   ```
   This enables the custom password policy for your FortiSOAR instance.
3. If you want to change any of the parameters for the default or custom password policy, you require to edit the **custompwdvalidator.py** file located at `/opt/cyops-auth/validationutils/custompwdvalidator.py`
   For example, if you want to change the default length of the password that users can set from 8 characters to 10 characters, in the `custompwdvalidator.py` file update the following:
   ```
   if len(password) < 8:
       message = "Password must be at least 8 character long"
       logger.error(message)
       return False, message
   ```
   To
   ```
   if len(password) < 10:
       message = "Password must be at least 10 character long"
       logger.error(message)
       return False, message
   ```
   Similarly, you can also update a custom policy. For example, if you do not want to enforce the "Password must not contain the username of the user" policy, you can comment out or remove the following code from the `custompwdvalidator.py` file:
   ```
   if loginid.lower() in password.lower():
       message = "username not allowed in password"
   ```

```
    logger.error(message)
    return False, message
```

Important: If you make any changes to the `validate()` function in the `customepwdvalidator.py` file, ensure you make the corresponding update in the `password_policy()` function in the same file.

Optionally, if you want to update the "Password must not be one of 10 previous passwords" custom policy to "Password must not be one of 12 previous passwords", you can run the following command:

```
/opt/cyops/scripts/api_caller.py --method PUT --endpoint https://localhost/api/auth/config
--payload '{"option":"history","value": 12}'
```

The value of the **value** parameter in the `--payload` determines the number of passwords that users cannot use, for example in the above command it is set to '12'.

Or, if you want to update the "Password change not allowed within 1 day of last password change" custom policy to "Password change not allowed within 2 days of last password change", you can run the following command:

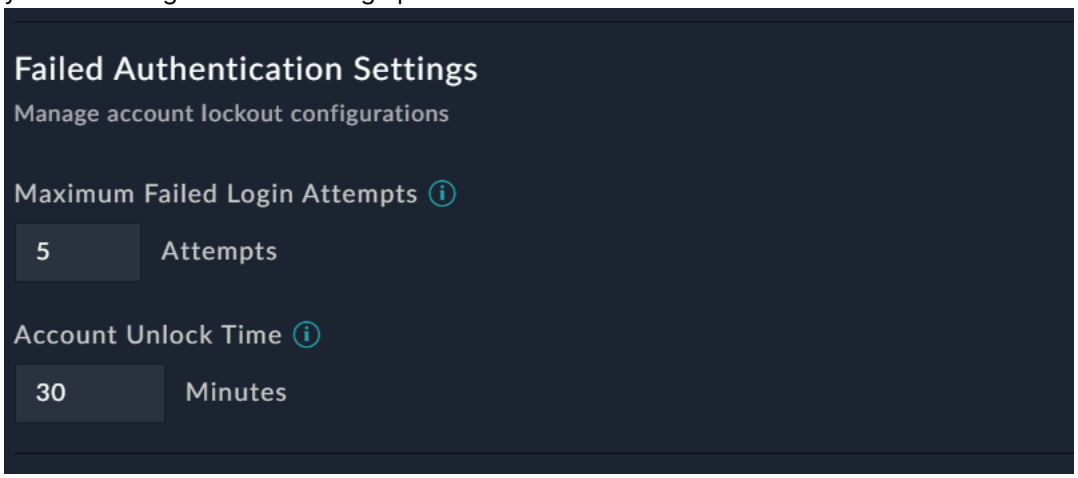
```
/opt/cyops/scripts/api_caller.py --method PUT --endpoint https://localhost/api/auth/config
--payload '{"option":"min_password_age","value": 2}'
```

The value of the **min_password_age** parameter in the `--payload` determines how often users can change their passwords, for example in the above command it is set to '2', i.e., users cannot change their password within a two-day time frame.

4. If you make any changes in the `custompwdvalidator.py` file, then you must restart the `cyops-auth` service:
   `systemctl restart cyops-auth`

Configuring account lockout configurations

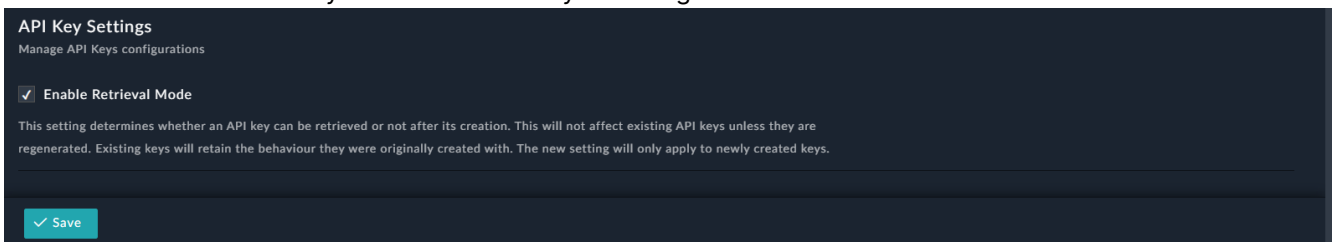
Click **Settings** > **Authentication** to open the `Account` page. In the `Failed Authentication Settings` section, you can configure the following options for account lockouts:



- **Maximum Failed Login Attempts**: Specify the number of times that users can enter an incorrect password while logging into FortiSOAR before their account gets locked. By default, this is set to 5 (times).
- **Account Unlock Time**: Specify the duration, in minutes, after which the user accounts get automatically unlocked, in cases where user accounts were locked due to exceeding the number of failed login attempts. By default, this is set to 30 (minutes).

Setting the API key retrieval mode

Click **Settings** > **Authentication** to open the `Account` page. In the `API Key Settings` section, you can choose to allow retrieval of the API key after its creation by selecting the **Enable Retrieval Mode** checkbox:



Whether or not an API key can be retrieved is dependent on the **Enable Retrieval Mode** settings at the time of API key creation. For example, if you have enabled the retrieval mode and then created an API key, that API key will always be retrievable, even if you disable the retrieval of API key setting later. Also, note that this setting does not affect existing API keys unless they are regenerated. By default, the retrieval mode is disabled.

Configuring Two-Factor Authentication (2FA)

In FortiSOAR release 7.4.2, generic two-factor authentication (2FA) implementation is introduced to allow users to effectively implement and customize 2FA methods according to their requirements. For more information on the available 2FA APIs, see the `Two-Factor Authentication (2FA) Methods` topic in the *API Methods* chapter of the "API Guide". Out-of-the-box, FortiSOAR supports Telesign and Google Authenticator as the authentication vendors.

Click **Settings** > **Authentication** to open the `Account` page. On the `Account` page. Click **2FA** to configure two-factor authentication for user accounts as follows:

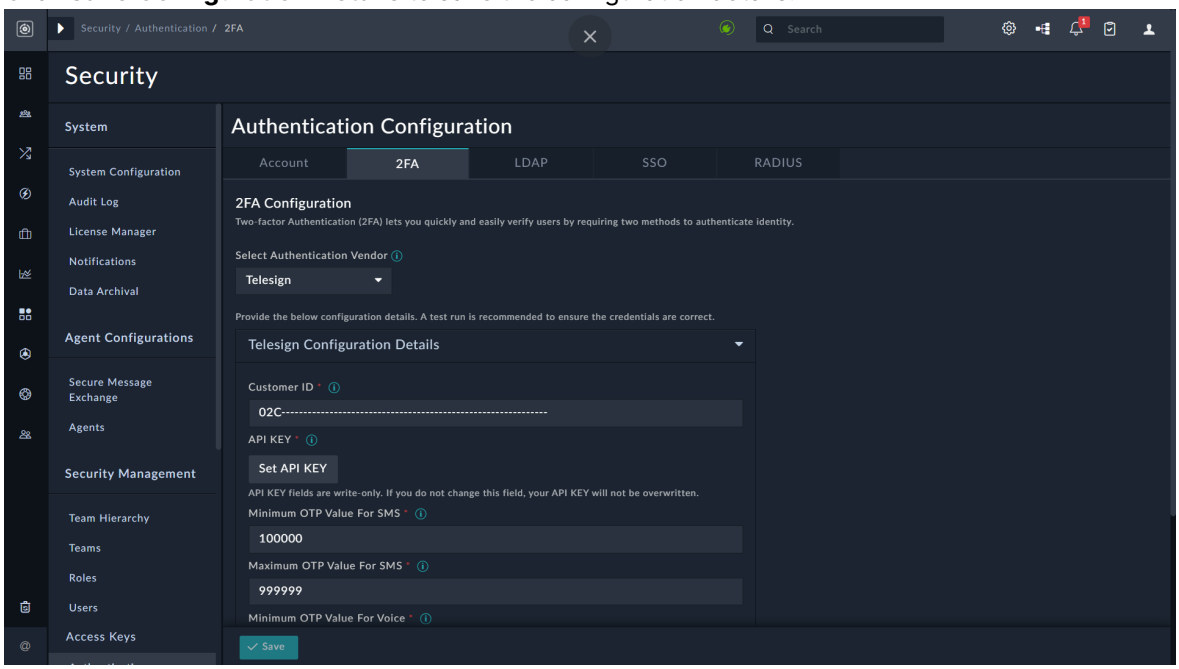
To configure 2FA, do the following:

1. In the `2FA Configuration` section, from the **Select Authentication Vendor** drop-down, select the authentication vendor to be used for providing the 2FA authentication tokens for logging on to FortiSOAR. Out-of-the-box, FortiSOAR supports **Telesign** and **Google Authenticator** as the authentication vendors.
2. Provide the details to configure the selected authentication method, i.e., Telesign or Google Authenticator. **NOTE**: Currently, you can only configure one level of authentication. Multi-level authentication is not supported, i.e., you cannot configure both Telesign and Google Authenticator.
   a. If you select **Telesign**, then you must specify the following details of your Telesign account to configure 2-Factor Authentication (2FA) for FortiSOAR users in the `Telesign Configuration Details` section:
      i. In the **Customer ID** field, enter the customer ID that has been provided to you for using TeleSign.
      ii. In the **Set API Key** field, enter the API Key that has been provided to you for using TeleSign.
      iii. In the **Minimum OTP Value For SMS** field, enter the smallest number to be used as the authentication token while sending the text message to the user's configured phone number.
      iv. In the **Maximum OTP Value For SMS** field, enter the largest number to be used as the authentication token while sending the text message to the user's configured phone number.

v. In the **Minimum OTP Value For Voice** field, enter the smallest number to be used as the authentication token while sending the voice message to the user's configured phone number.

vi. In the **Maximum OTP Value For Voice** field, enter the largest number to be used as the authentication token while sending the voice message to the user's configured phone number.

vii. Click **Save Configuration Details** to save the configuration details:

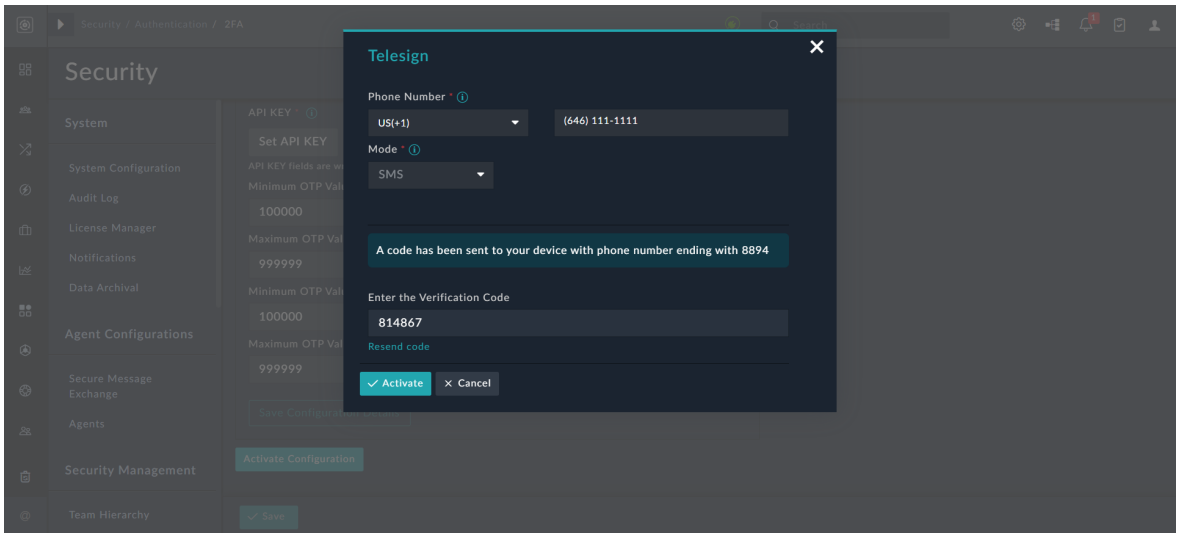


Note: FortiSOAR runs a health check on the provided configuration and, in the event of a failure, throws an error. The configuration details are saved only after the health check is successful.

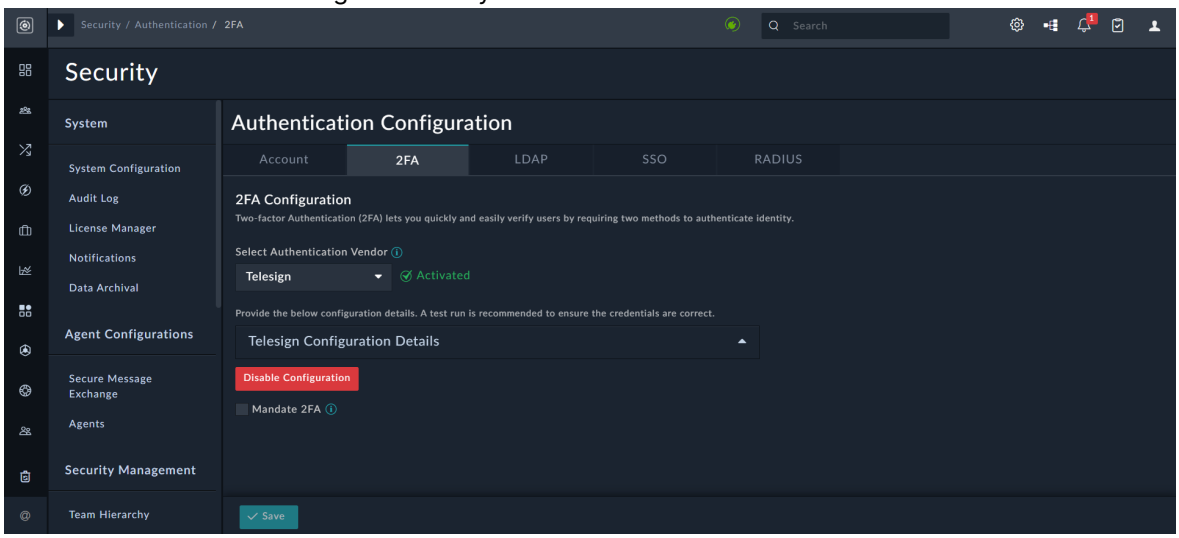
viii. Once the configuration details are saved, click **Activate Configuration** to verify the provided details.

Clicking **Activate Configuration** displays a dialog box in which you are required to enter the phone number with its country code in the **Phone Number** field. The country code is picked up by default from the value set in the **Default Country** field on the `System Configuration` page. Then, from the **Mode** drop-down list, select the mode, **SMS** or **Voice** to receive the authentication token, and then click **Get OTP**.

If the configuration details are correct, then you will receive the verification code, which you can enter in the **Enter Verification Code** field and click **Activate**:

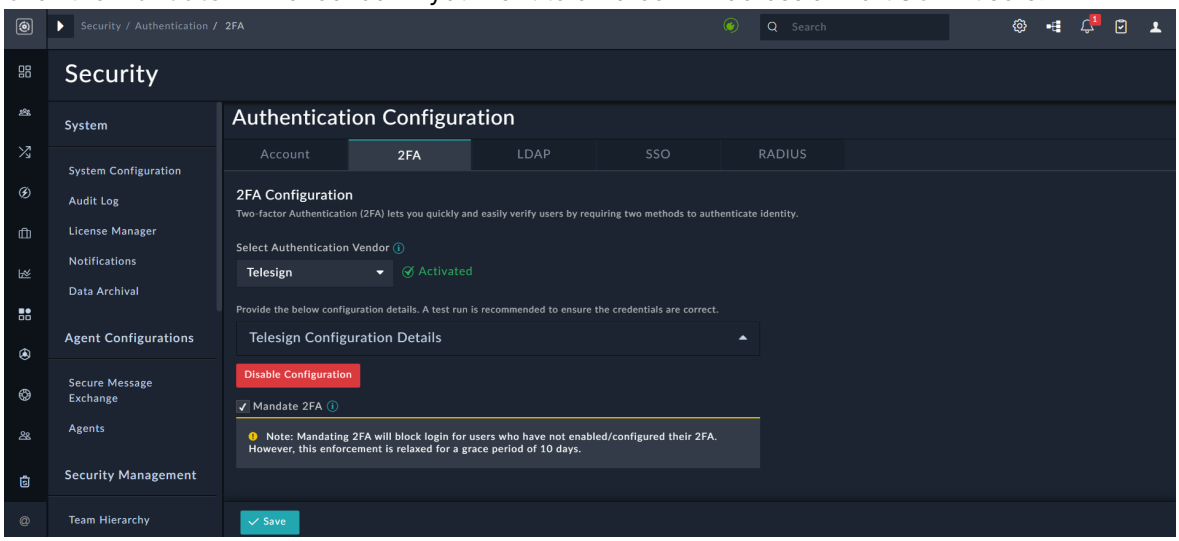


This activates the 2FA configuration for your account and for FortiSOAR user accounts.



If you want to disable 2FA for FortiSOAR, click **Disable Configuration**.

ix. Click the **Mandate 2FA** checkbox if you want to enforce 2FA across all FortiSOAR users:



By choosing this option, all FortiSOAR users are required to have 2FA enabled in their user profiles. For more information on enabling 2FA for users, see 2-Factor on page 107.
Selecting this option blocks users who do not have 2FA enabled from logging on to FortiSOAR. Keep in mind that there is a 10-day grace period before this enforcement becomes strict. Users who don't have 2FA enabled after the 10-day grace period are prevented from accessing FortiSOAR. However, keep in mind that administrators can reset the accounts of such users enabling them to access FortiSOAR *once* and enable their 2FA. Resetting user accounts is done by administrators by selecting Reset in the 2-Factor section of the user's profile. Additionally, users cannot disable their configured 2FA once 2FA is mandated.

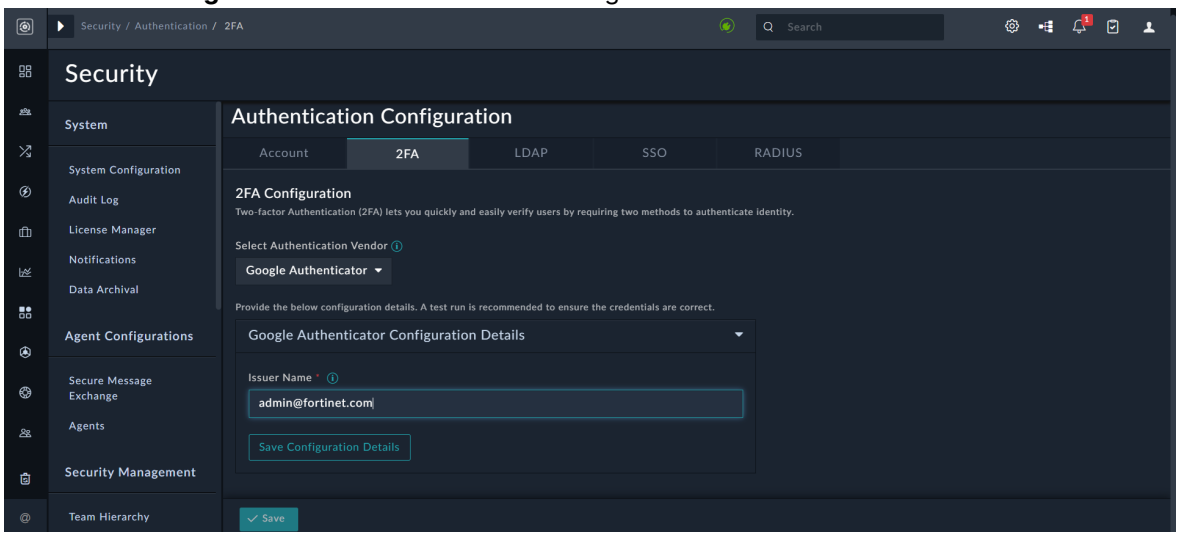
x. Click **Save** to save the account configuration.

b. If you select **Google Authenticator**, then you need to have Google Authenticator installed on your device for activating and configuring 2-Factor Authentication (2FA) for FortiSOAR users:

i. Click **Google Authenticator Configuration Details** to display the Issue Name field.

ii. In the **Issuer Name** field, enter the name of the provider or service with which you want to associate the Google Authenticator account.

iii. Click **Save Configuration Details** to save the configuration details:



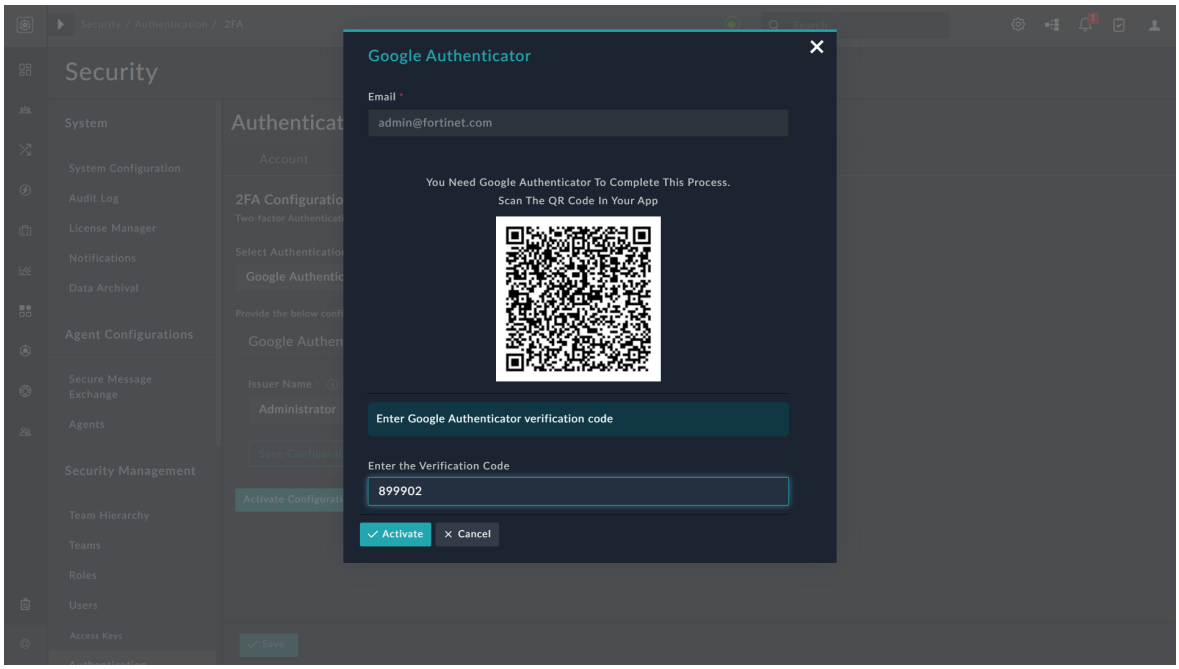
iv. Once the configuration details are saved, click **Activate Configuration** to verify the provided details.

v. In the **Email** field, enter an email address to be used for setting up Google Authenticator.

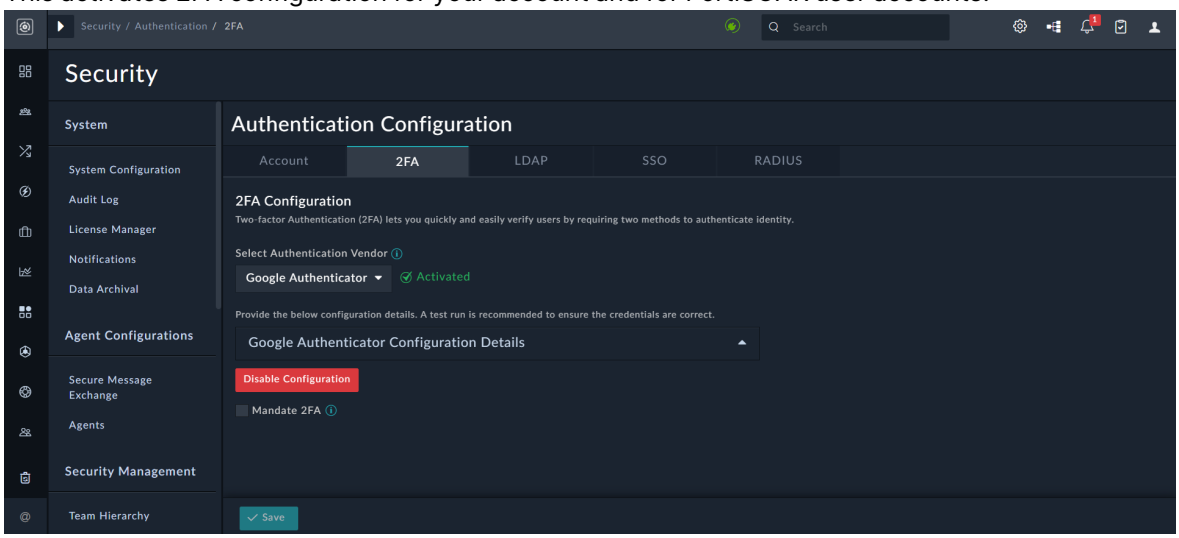
vi. Click **Get OTP** to display a QR code.

vii. Scan this QR code using Google Authenticator to complete the process and get the verification code.

viii. In the **Enter the Verification Code** field, enter the verification code provided by Google Authenticator and click **Activate**.

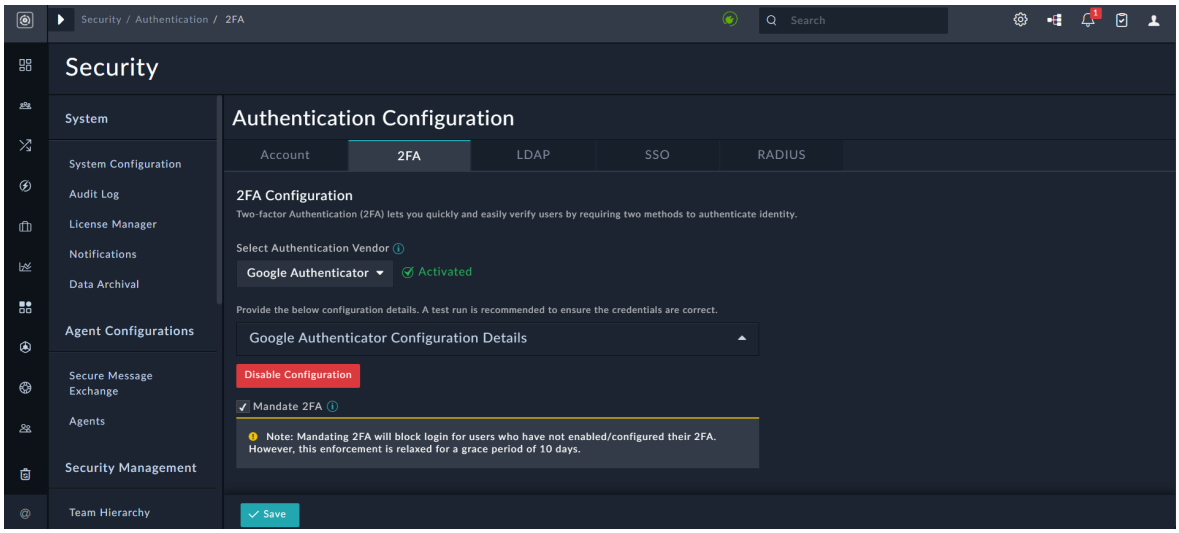


This activates 2FA configuration for your account and for FortiSOAR user accounts.



If you want to disable 2FA for FortiSOAR, click **Disable Configuration**.

ix. Click the **Mandate 2FA** checkbox to enforce 2FA across all FortiSOAR users:



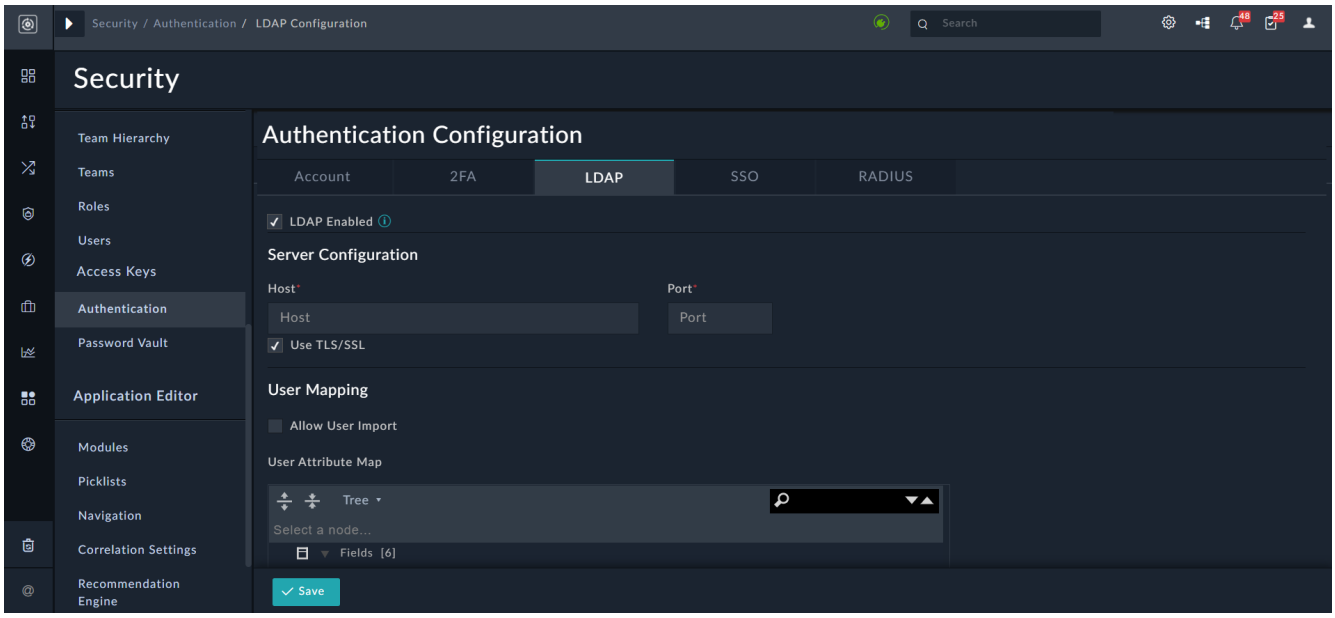
By choosing this option, all FortiSOAR users are required to have 2FA enabled in their user profiles. For more information on enabling 2FA for users, see 2-Factor on page 107.
Selecting this option blocks users who do not have 2FA enabled from logging on to FortiSOAR. Keep in mind that there is a 10-day grace period before this enforcement becomes strict. Users who don't have 2FA enabled after the 10-day grace period are prevented from accessing FortiSOAR. However, keep in mind that administrators can reset the accounts of such users enabling them to access FortiSOAR *once* and enable their 2FA. Resetting user accounts is done by administrators by selecting Reset in the 2-Factor section of the user's profile. Additionally, users cannot disable their configured 2FA once 2FA is mandated.

x. Click **Save** to save the account configuration.

Configuring LDAP / AD

Use the `Authentication` menu to setup, modify, and turn on or off your LDAP / AD authentication provider. Click **Settings** > **Authentication** to open the `Account` page. Click the **LDAP** tab and click the **LDAP Enabled** checkbox, if you want to enable LDAP authentication for FortiSOAR.

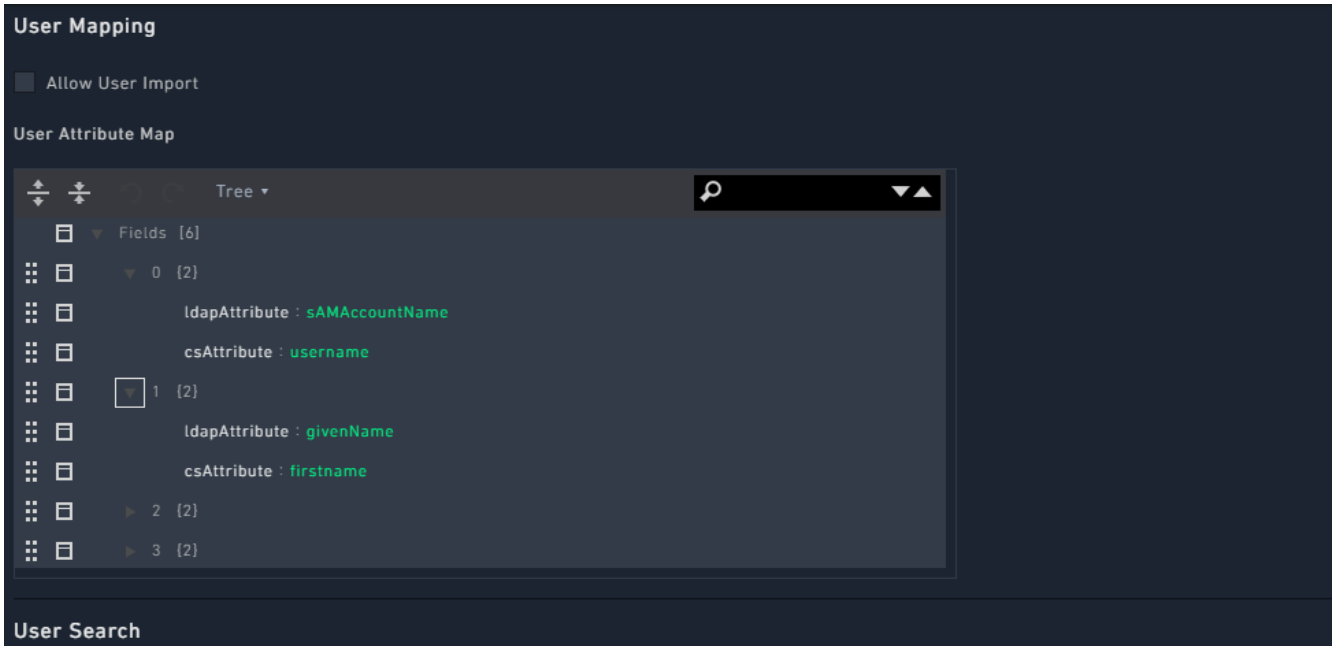
Enter the hostname and port of your LDAP / AD authentication server. Click **Use TLS/SSL** and then provide a user search the directory and import users. You can add users either by mapping users using the `User Attribute Map`, or search for users in the directory and then import users.



User Attribute Map

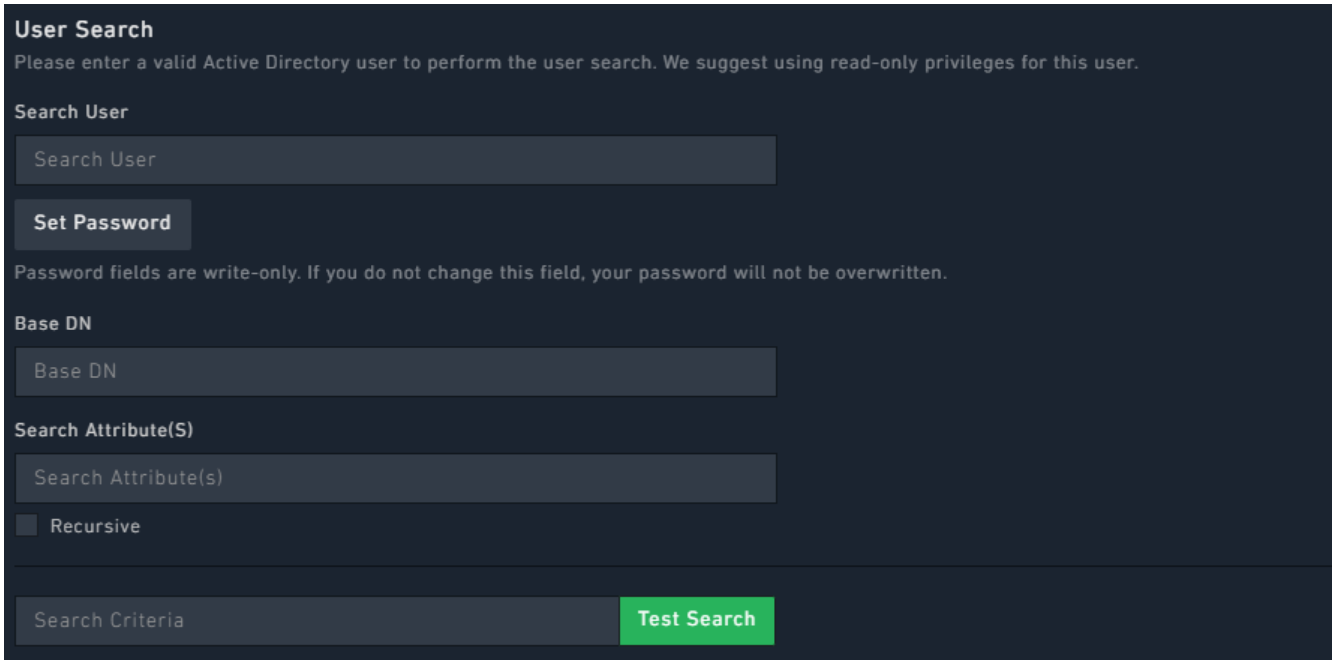
To map users, configure the **User Attribute Map**. FortiSOARprovides you a default user attribute map array that contains the most common combination of field mappings. You can modify the mappings based on your own LDAP container fields by editing the map.

In the `User Attribute Map`, under `Fields`, click the editable field name (right-side field name), to map it to your LDAP fields. The non-editable field name (left-side field name) is the FortiSOAR attribute.



User Search

You must have valid administrative username and password to search the LDAP / AD resource for user information. You do not have to use admin credentials, but at a minimum, you must have user credentials to access and import all desired user containers.



Search User: Searches LDAP / AD for a user in the format `CN=UserName,CN=Users,DC=XXXX,DC=XXX`.

Base DN: Base DN for user search in the format `CN=Users,DC=XXX,DC=XXX`.

Search Attribute (s): Attribute for searching a user, for example, `sAMAccountName`.
Check the **Recursive** checkbox for recursively searching for users.

Search Criteria: Criteria for searching a user, for example, `SOCMembers`.

Once you have added the credentials in the `User Search` section, click **Allow User Import** to configure your environment to look in the LDAP / AD resource for **all new users**.

If you want to add local users, you must clear the `Allow User Import` checkbox to revert your system to the local user import in the Users administration menu.

Configuring SSO

Use the `Authentication` menu to setup, modify, and turn on or off your SSO configuration. Click **Settings**> **Authentication** to open the `Account` page. Click the **SSO** tab and click the **SAML Enabled** check box if you want to enable SAML for FortiSOAR. You must configure SAML in FortiSOAR to enable users to use single sign-on.

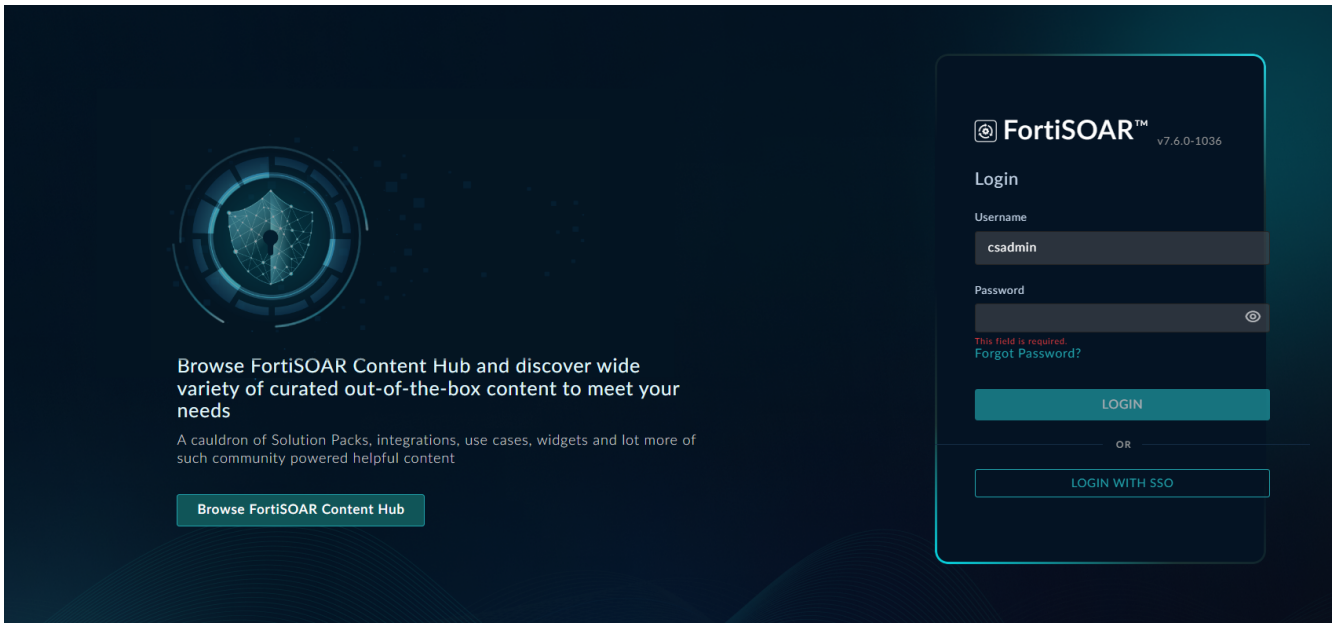
Configuring SAML is a two-way process:

- The IdP configuration must be added to the FortiSOAR UI, see the Configuring SAML in FortiSOAR. topic.
- Similarly, the Service Provider (SP) configuration that is present in the FortiSOAR UI must be made at the Identity Provider (IdP). Steps for configuring the various SPs are detailed in the following sections:
  - For FortiAuthenticator (FAC): Configuring SAML in FortiAuthenticator section.
  - For OneLogin: Configuring SAML in OneLogin section.
  - For Auth0: Configuring SAML in Auth0 section.
  - For Okta: Configuring SAML in Okta section.
  - For Google: Configuring SAML in Google section. You must have an administrator account for your G Suite account.
  - For Microsoft Entra ID (formerly Azure AD): Configuring SAML in Microsoft Entra ID section.
  - For Active Directory Federation Services (ADFS): Configuring SAML in ADFS section. For specific information about the values, you need to add for the SSO configuration, see Configuring FortiSOAR for ADFS.

Introduction to SAML

Security Assertion Markup Language (SAML) is an XML-based, open standard data format for exchanging authentication and authorization data between parties, particularly between an identity provider and a service provider. The single most important requirement that SAML addresses are web browser single sign-on (SSO).

By using SAML, FortiSOAR does not require to store user credentials, and FortiSOAR is independent of the underlying authentication mechanism used by a user. Once you complete making all the SAML configurations on both the FortiSOAR and Identity Provider (IdP) side, then the FortiSOAR login page will display a **Login with SSO** button. Users can then log on to FortiSOAR using the **Login with SSO** button that is present on the FortiSOAR login page.



Once the user clicks the **Login with SSO** button, the user is redirected to a third-party identity provider login page, where the user must enter their credentials and get themselves authenticated. Once a user successfully logs on to FortiSOAR, the user profile automatically gets created. The User profile is created based on the configurations you have set while Configuring SAML in FortiSOAR. For example, when the user is created, the

user is assigned the default team and role based on what the administrator configured during SAML configuration. Users can update their profile by editing their user profile.

You can map the role and team of SSO users in FortiSOAR based on their roles defined in the IdP. Thereby you can set different roles for different SSO users, i.e., you can set the role of an SSO user in FortiSOAR based on the role you have defined in your IdP. For more information, see Support for mapping roles and teams of SSO users in FortiSOAR.

The default access type set for all SSO users is 'Concurrent'. Administrators can change the access type for the user later, if needed. For more information about user access types, see the *Licensing FortiSOAR* chapter in the "Deployment Guide."

Benefits of SAML

User experience: SAML provides the ability for users to securely access multiple applications with a single set of credentials entered once. This is the foundation of the federation and single sign-on (SSO). Using SAML, users can seamlessly access multiple applications, allowing them to conduct business faster and more efficiently.

Security: SAML is used to provide a single point of authentication at a secure identity provider, meaning that user credentials never leave the firewall boundary, and then SAML is used to assert the identity to others. This means that applications do not need to store or synchronize identities, which in turn ensures that there are fewer places for identities to be breached or stolen.

Standardization: The SAML standardized format is designed to interoperate with any system independent of implementation. This enables a more open approach to architecture and identity federation without the interoperability issues associated with vendor-specific approaches.

SAML Principles

Roles

SAML defines three roles: Principal (generally a user), Identity Provider (IdP), and the Service Provider (SP).

Principal: The principal is generally a user that has an authentic security context with IdP and who requests a service from the SP.

Identity Provider (IdP): IdP is usually a third-party entity outsourced to manage user identities, or in other terms, an IdP is a user management system. The IdP provides user details in the form of assertions. Before delivering the identity assertion to the SP, the IdP might request some information from the principal, such as a username and password, to authenticate the principal. SAML specifies the assertions between the three parties: in particular, the messages that assert identity that is passed from the IdP to the SP. In SAML, one identity provider can provide SAML assertions to many service providers. Similarly, one SP might rely on and trust assertions from many independent IdPs.

Service Provider (SP): The SP maintains a security wrapper over the services. When a user request for a service, the request first goes to the SP, who then identifies whether a security context for the given user exists. If not, the SP requests and obtains an identity assertion from the IdP. Based on this assertion, the service provider makes the access control decision and decides whether to perform some service for the connected principal.

Attribute Mapping

Each IdP has its own way of naming attributes for a user profile. Therefore, to fetch the attribute details for a user from an IdP into the SP, the attributes from the IdP must be mapped to attributes at the SP. This mapping is taken care in a separate part at the SP. If the attribute mapping is not set correctly, the SP sets default values for mandatory attributes like First Name, Last Name, and Email.

Prerequisites to configuring SAML

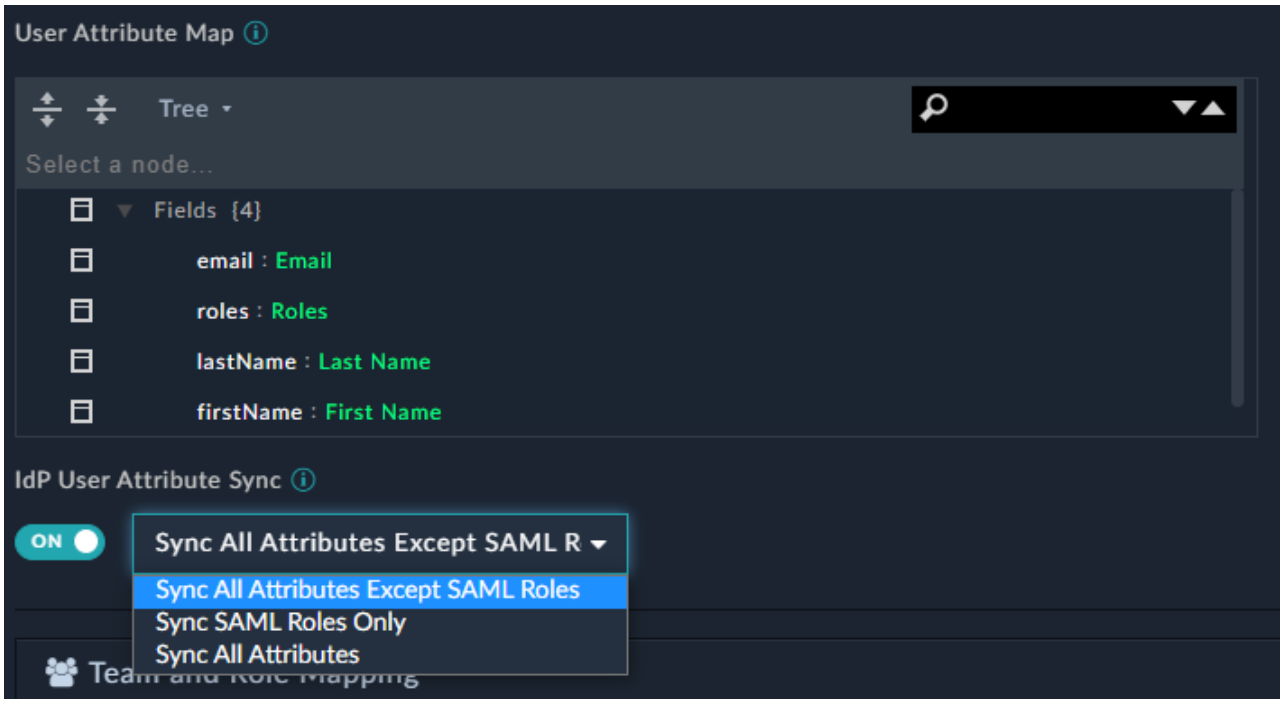
- Ensure that you are assigned a security administrator role that at a minimum has `Read`, `Create` and `Update` permissions on the `Security` module. You also require to have `Read` permissions for Teams and Roles.
- Ensure that you have enabled SAML in your FortiSOAR instance. To enable SAML, log on to FortiSOAR, click **Settings**. In the `Security Management` section click **Authentication** to open the `Authentication Configuration` page. Click the **SSO** tab and click the **SAML Enabled** checkbox.

Configuring SAML in FortiSOAR

Configuring SAML is a two-way process. The SP configuration that is present in the FortiSOAR UI must be made at the IdP. Similarly, the IdP configuration must be added to the FortiSOAR UI.

This section covers configuring SAML with IdPs such as, FortiAuthenticator (FAC) OneLogin, Auth0, Okta, Google, and Active Directory Federation Services. (ADFS), which are the IdPs that have been tested with FortiSOAR. You can use a similar process to configure any other IdP that you use.

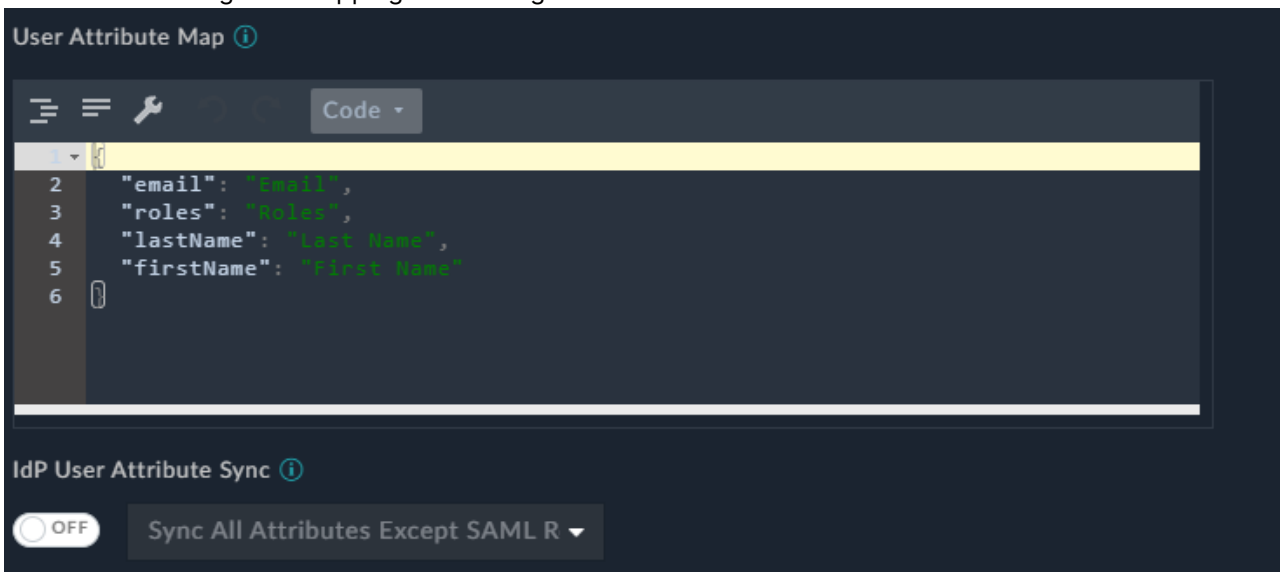
1. Log on to FortiSOAR as an administrator.
2. Click **Settings** > **Authentication** > **SSO**.
3. To enable SAML for FortiSOAR, click the **SAML Enabled** check box.
4. In the `Identity Provider Configuration` section, enter the IdP details.
   Get the details for FortiAuthenticator (FAC) from the Configuring SAML in FortiAuthenticator section.
   Get the details for OneLogin from the Configuring SAML in OneLogin section.
   Get the details for Auth0 from the Configuring SAML in Auth0 section.
   Get the details for Okta from the Configuring SAML in Okta section.
   Get the details for Google from the Configuring SAML in Google section. You must have an administrator account for your G Suite account.
   Get the details for Microsoft Entra ID (formerly Azure AD) from the Configuring SAML in Microsoft Entra ID section.
   For information on Configuring SAML in FortiSOAR for Active Directory Federation Services (ADFS) from the Configuring SAML in ADFS section. For specific information about the values, you need to add for the SSO configuration, see Configuring FortiSOAR for ADFS.
5. From the **Provision User** drop-down list, select the user creation strategy, i.e., choose either **At Sign-in (Default)** or **Pre-provision**. For more information see the Pre-provisioning SAML users section.
6. Map the user attributes received from the IdP with the corresponding attributes of FortiSOAR.
   Use the **User Attribute Map** to map the attributes received from the IdP with the corresponding attributes required by FortiSOAR. FortiSOAR requires the firstname, lastname and email attributes to be mapped.
   In the `User Attribute Map`, under `Fields`, in the **Tree** view, click the editable field name (right side field name), to map it to the attribute that will be received from the IdP. The non-editable field name (left-side field name) is the FortiSOAR attribute. For example, in the following image, you map the FortiSOAR attribute `firstName` to the IdP attribute `First Name`.



If you want to sync the user attributes and/or SAML roles between FortiSOAR users and their IdP profile, toggle **IdP User Attribute Sync** to **ON** and then select one of the following options:

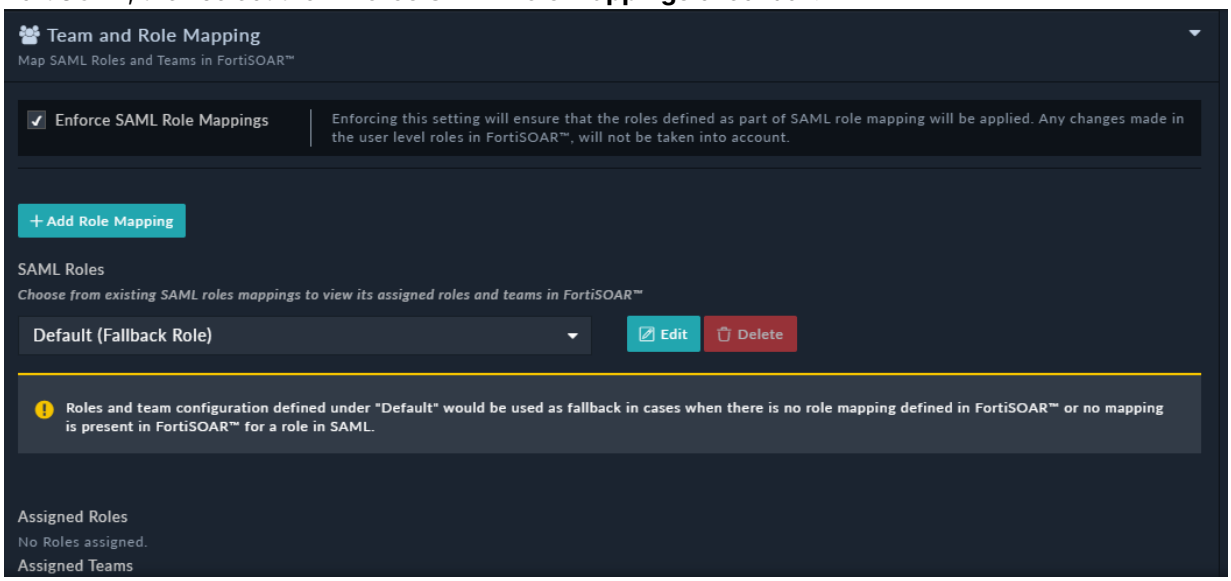
- **Sync All Attributes Except SAML Roles**: Selecting this option synchronizes all user attributes except *SAML Roles* between FortiSOAR users and their IdP profile.
- **Sync SAML Roles Only**: Selecting this option synchronizes only the SAML Roles between FortiSOAR users and their IdP profile.
- **Sync All Attributes**: Selecting this option synchronizes all user attributes and SAML Roles between FortiSOAR users and their IdP profile.

You can also change the mapping in the using the **Code** View:

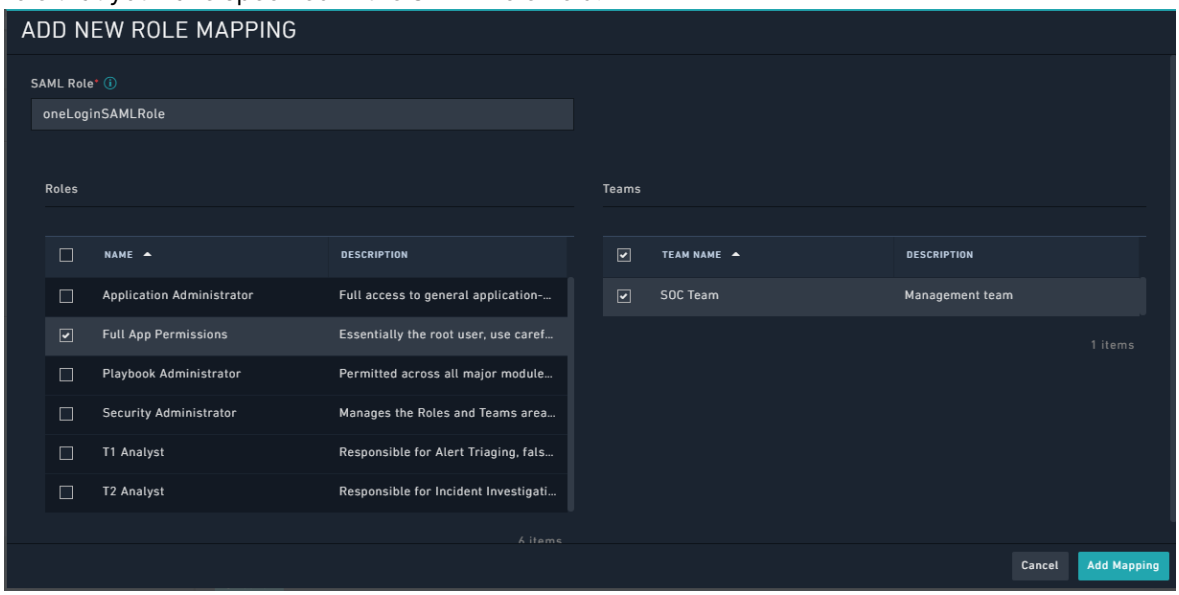


IMPORTANT: The firstname and lastname attributes will be set to their default value, "Change Me", if you forget to configure these attributes or set them with incorrect values when setting up the user attributes mapping.

7. To map roles that you have defined in your IdP (see Support for mapping roles and teams of SSO users in FortiSOAR) to teams and roles in FortiSOAR, do the following:
   a. If you want to ensure that roles defined as part of SAML role mapping will be applied to SSO users in FortiSOAR, then select the **Enforce SAML Role Mappings** checkbox.



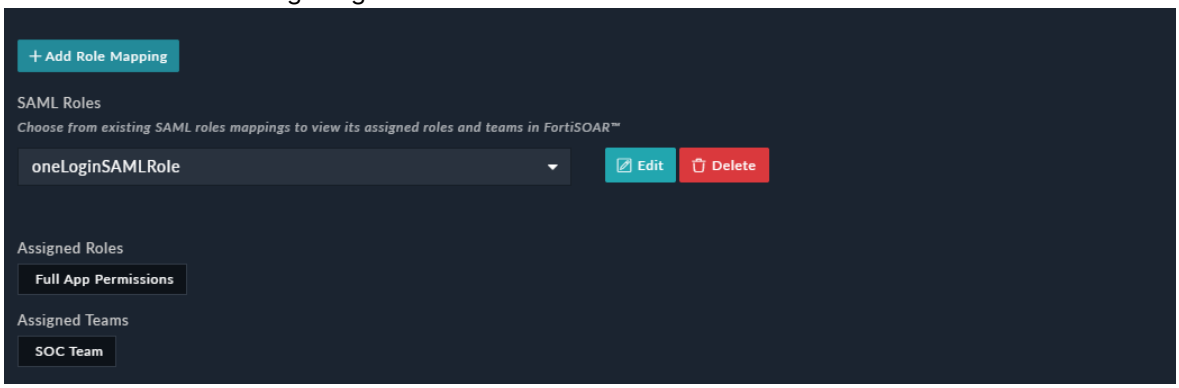
   b. To map a role in the IdP to a FortiSOAR-role and optionally a team in FortiSOAR, in the `Team and Role Mapping`section, click **Add Role Mapping**.
   c. In the `Add New Role Mapping` dialog, do the following:
      i. In the **SAML Role** field, add the name of the roles that you have defined in your IdP.
      Note: The name that you have specified in your IdP, and the name that you enter in this field must match exactly, including the matching the case of the name specified.
      ii. From the `Roles` column, select the FortiSOAR role(s) that you want to assign to the role that you have specified in the SAML Role field.
      iii. (Optional) From the `Teams` column, select the FortiSOAR teams(s) that you want to assign to the role that you have specified in the SAML Role field.



   Once you assign the default team and roles to users, all user profiles created contain this team and

role assigned to them.
If you do not assign the default team and roles to users, and you have also not defined a **Default (Fall Back Role)**, details given in a further step in this procedure, then all user profiles are created without team or role information and will have only basic access. In this case, users will require to request the administrator for appropriate access and privileges.

iv. Click **Add Role Mapping**.
This adds the mapped role in the SAML Roles drop-down list in the `Team and Role Mapping`section as shown in the following image:



As shown in the above image, the `oneLoginSAMLRole`, i.e., the role defined in the IdP has been mapped to the `Application Administrator` role and the `SOC Team` in FortiSOAR.

d. To define a default role (and optionally teams) that will be assigned to the SSO user if you have not set up mapped roles of SSO users in FortiSOAR, or if FortiSOAR receives a response from the IdP that does not contain any roles, or receives a response that does not map to any of the FortiSOAR roles, do the following:

i. From the **SAML Roles** drop-down list, select **Default (Fall Back Role)** and click **Edit**.

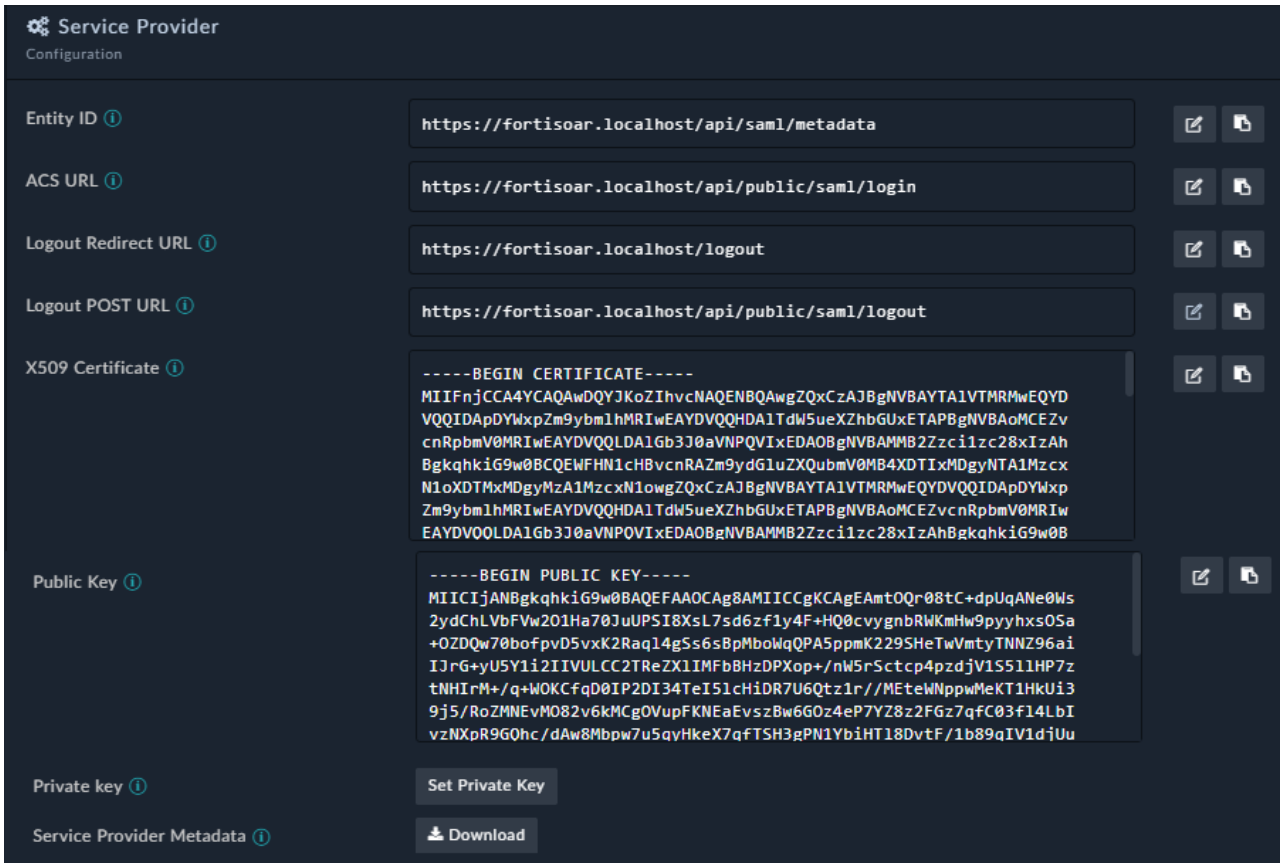
ii. In the `Update Role Mapping` dialog, from the `Roles` column select the role(s) that you want to assign to the default role. You can also optionally select the team(s) that you want to assign to the default role from the `Teams` column and click **Update Mapping**.

e. (Optional) To delete or update an existing role do the following:

i. To update an existing role, select the role from the **SAML Roles** drop-down list and click **Edit** and in the `Update Role Mapping` dialog, you can update the name of the mapped SAML role, and the mapped FortiSOAR roles and teams.
Once you have completed modifying the existing role as per your requirement, click **Update Mapping**.

ii. To delete an existing role, select the role from the **SAML Roles** drop-down list and click **Delete**. FortiSOAR displays a confirmation dialog, click **Confirm** to delete the role.

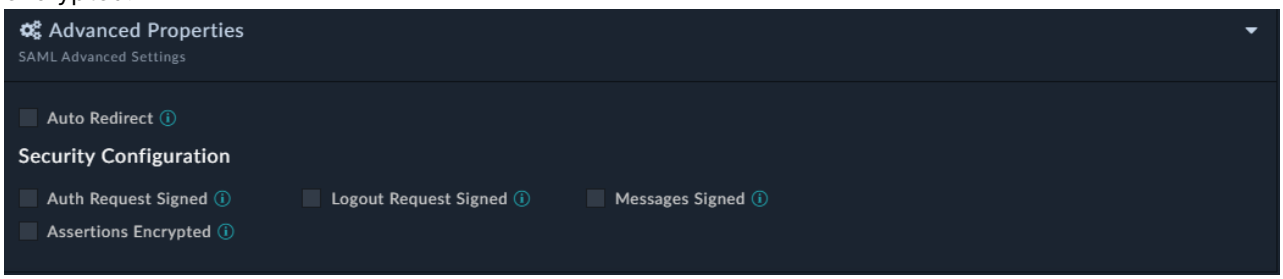
8. Add the information provided in the `Service Provider` section to `Configuration` section your IdP.
This information is pre-configured. However, you can edit the fields, such as **Entity ID** (hostname) within this section. This is especially useful if you are using an alias to access FortiSOAR. You can also edit the certificate information and the private and public keys of your service provider, which is useful in cases where you want to use your own certificates.



For OneLogin, enter this information in the `Configure IdP` step. See the Configuring SAML in OneLogin section for more details.
For Auth0, enter this information in the `Configure IdP` step. See the Configuring SAML in Auth0 section for more details.
For Okta, enter this information in the `Configure IdP` step. See the Configuring SAML in Okta section for more details.

9. (Optional) Configure advanced settings for SAML.
   Prior to version 7.0.0, users were required to click the **Use Single Sign On (SSO)** link to get redirected to the SSO login page or login using SSO active session. However, there are some organizations that have policies, which require direct redirection to the SSO login page, if SSO is configured. Therefore, in version 7.0.0 an **Auto Redirect** checkbox is introduced. Select the **Auto Redirect** checkbox to redirect users directly to the SSO login page or automatically log the user into FortiSOAR in case the SSO session is active. If you leave the **Auto Redirect** checkbox cleared, then FortiSOAR directs users to the FortiSOAR login page, where users can click the **Use Single Sign On (SSO)** link to get redirected to the SSO login page or login using SSO active session.
   If you have selected the **Auto Redirect** checkbox, i.e., enabled SSO auto-redirect, administrator can yet access the FortiSOAR login page to configure or troubleshoot issues with the portal, by adding `auto_redirect=false` to the URL. For example, `https://<hostname>/login/?auto_redirect=false`
   Select the **Auth Request Signed** checkbox if your IdP requires FortiSOAR to send signed authentication requests.
   Select the **Logout Request Signed** checkbox if your IdP requires FortiSOAR to send signed logout requests.
   Select the **Messages Signed** checkbox if you want messages coming from your IdP to be signed.
   Select the **Assertion Encrypted** checkbox if you want assertions within the SAMLResponse to be

encrypted.



10. Click **Save** to complete the SAML configuration in FortiSOAR.

Changing the hostname using the `csadm` command does not change hostname part in 'Service Provider' details in SAML configurations. Therefore, if you have changed the hostname, you must manually update the hostname in 'Service Provider' details in the SAML Configuration page.

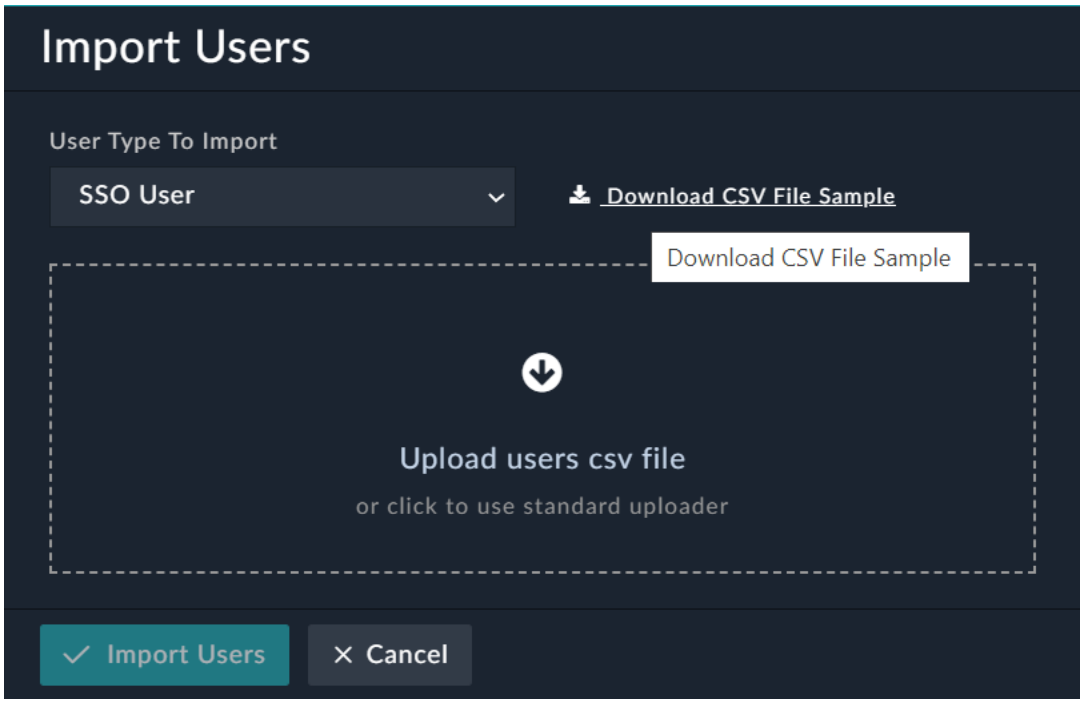
Pre-provisioning SAML users

From version 7.0.2 onwards, you can choose the strategy to create SSO users between **At Sign-in (Default)** or **Pre-provision**. The At Sign-in (Default) strategy creates users at login, i.e., user accounts are created automatically on the first SSO login of the user login. The Pre-provision strategy requires the user account to be created prior to login. Pre-provisioning users enables you to limit the SSO authentication only for pre-created users, which enables you to:

- Pre-decide roles mapping for SSO users. Now you can create users and then you can manually intervene to map the user to the correct role.
- Minimize the issues administrators can face due to incorrect or partial configuration mapping at the IdP's end.
- Control creation of users by mistake.

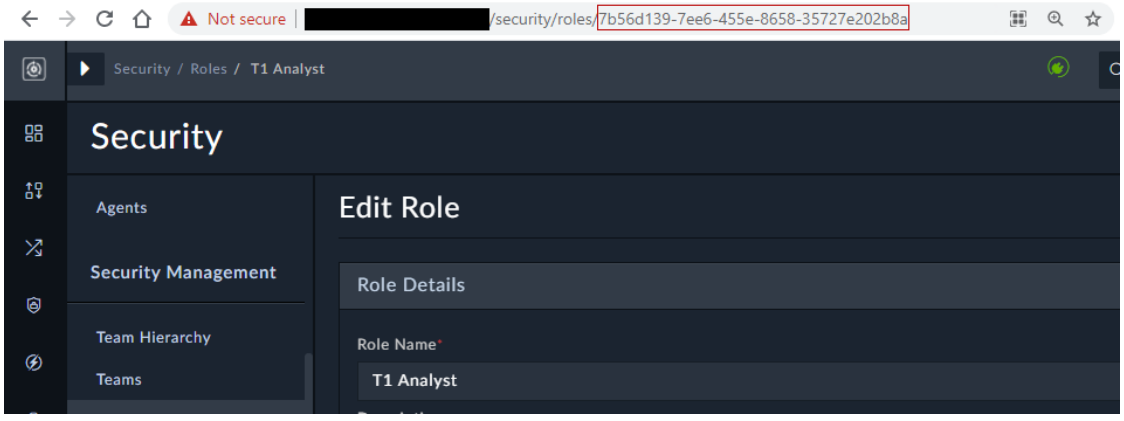
The process of pre-provisoning users is as follows:

1. Ensure that you have selected the **Pre-provision** option from the **Provision User** drop-down list on the `SSO` page (**Settings** > **Authentication** > **SSO**).
2. To pre-provision users you need to provide user details in a CSV file as follows:
   a. From the left menu, click **Users**, and on the `Users` page, click the **Import Users** button.
   Note: The **Import Users** button will be visible only if SSO or RADIUS is enabled.
   b. In the `Import Users` dialog, do the following:
      i. From the **User Type To Import** drop-down list, select **SSO User**.
      ii. Click the **Download CSV File Sample** link to download the sample CSV file (`SSO User_Template.csv`).



The sample CSV file contains an example of the user details you need to provide. You need to provide the following user details in the CSV file:

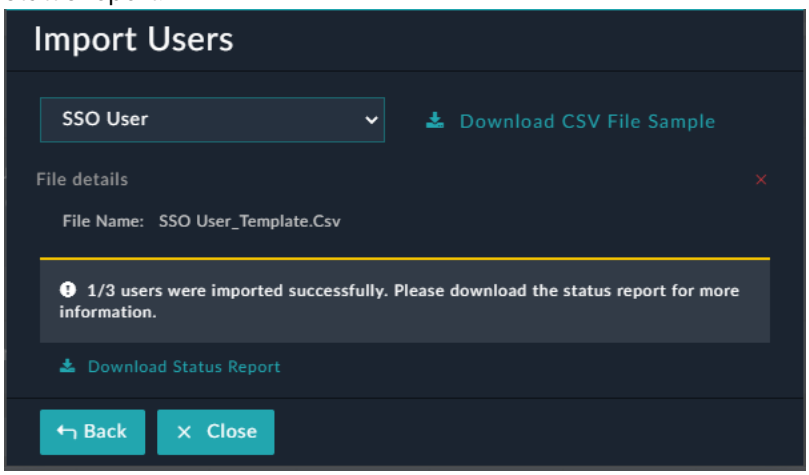
- username: Name of the SSO user.
- email: Email address of the SSO user
- firstname: (Optional) First name of the SSO user.
- lastname: (Optional) Last name of the SSO user.
- phonemobile: (Optional) Mobile number of the SSO user.
- roles: (Optional) Role (s) that you want to assign to the SSO user. To assign a role to the user you need to provide the UUID of that role. To get the UUID of a role, click **Settings** > **Security Management** > **Roles**, and then click the role that you want to assign to the user. For example, click **T1 Analyst**, which opens the `Edit Role` page, and then from the address bar, copy the UUID (as shown in the following image) and paste it in the roles column in the CSV file.



Note: You can assign multiple roles to the user by using the pipe symbol (|) to separate the UUID of each role.

- teams: (Optional) Team (s) that you want to assign to the SSO user. To assign a team to the user you need to provide the UUID of that team. To get the UUID of a team, click **Settings** > **Security Management** > **Teams**, and then click the team that you want to assign to the user, and then from the address bar copy the UUID of the team, similar to the process described for roles.
  Note: You can assign multiple teams to the user by using the pipe symbol (|) to separate the UUID of each team.
- accessType: Access type (Named or Concurrent) that you want to assign to the SSO user. If you do not specify any access type for the user, then the user will be assigned as a 'Concurrent' user.

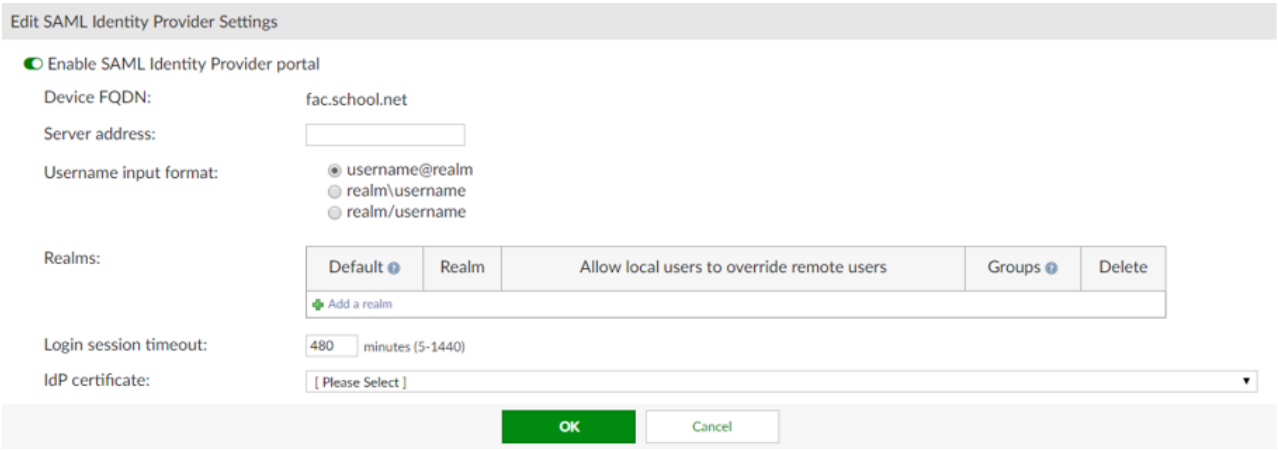
c. Once you complete filling the user details in the CSV file, click the **Import User** button, and in the `Import Users` dialog, drag and drop the csv file or click the **import** icon to import the CSV file, and then click the **Import Users** button.
If there are no issues in the import, then all the SSO users get created and they can log into FortiSOAR.
If there are any issues in the CSV files, such as not providing all the information required to create SSO users, then such users are not created. To get information about the users not created, download the status report.



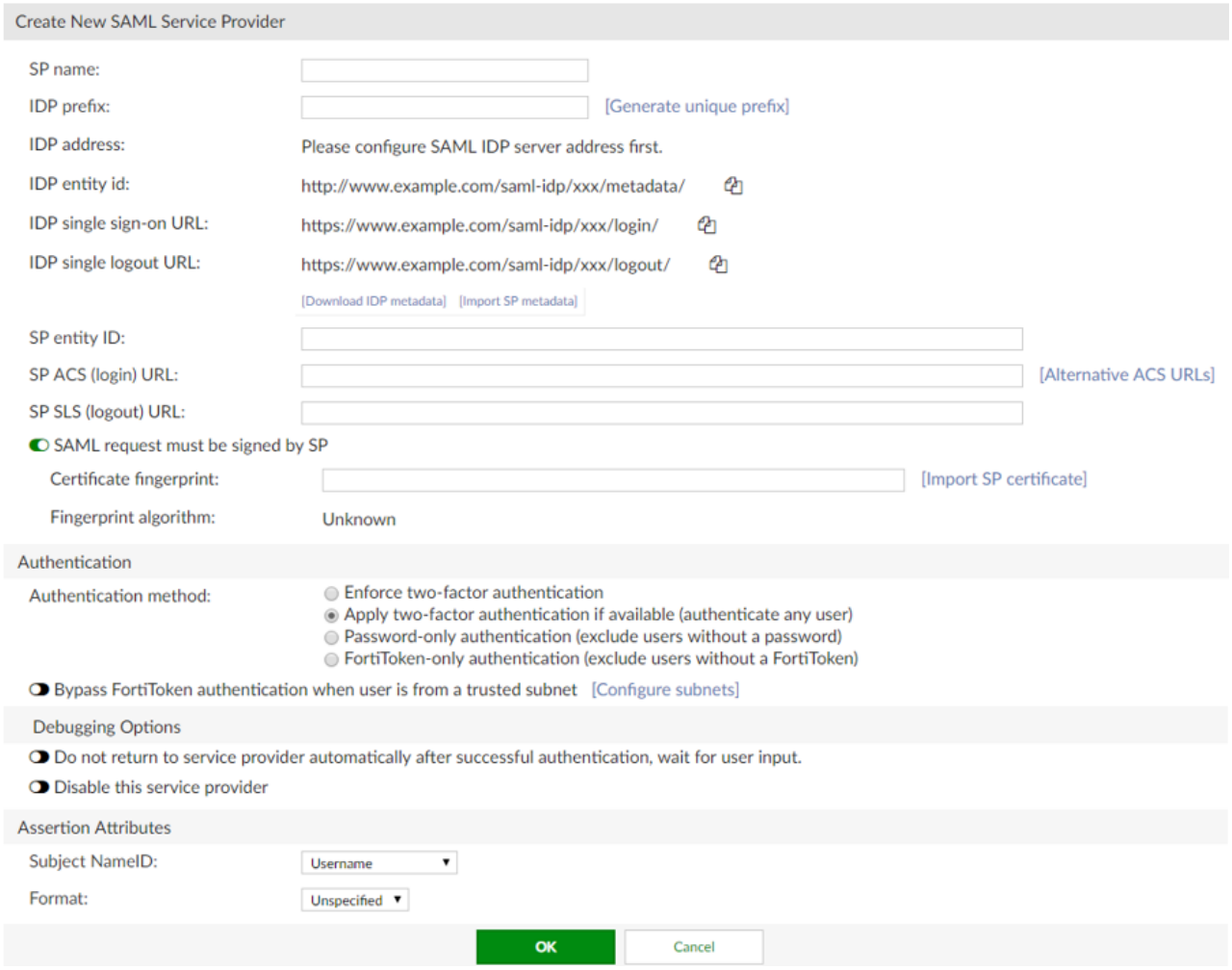
Configuring SAML in FortiAuthenticator

1. Log on to FortiAuthenticator (FAC) as an administrator.

2. Configure IdP. To configure general SAML IdP portal settings, navigate to **Authentication** > **SAML IdP** > **General**, and then select **Enable SAML Identity Provider portal**.



3. In the `Edit SAML Identity Provider Settings` section, enter the following details:
   - **Device FQDN**: To configure this setting, you must enter a Device FQDN in the System Information widget in the Dashboard.
   - **Server address**: Enter the IP address, or FQDN, of the FAC device.
   - **Username input format**: Select one of the following three username input formats:
     - username@realm
     - realm\username
     - realm/username
   - **Realms**: Select **Add a realm** to add the default local realm with which the users will be associated. Use **Groups** and **Filters** to add specific user groups.
   - **Login session timeout**: Set the user's login session timeout limit between 5 - 1440 minutes (one day). The default is 480 minutes (eight hours).
   - **IDP certificate**: Select a certificate from the dropdown menu.
4. Configure a Service Provider. Navigate to **Authentication** > **SAML IdP** > **Service Providers**, and then click **Create New**.



In the `Create New SAML Service Provider` section, enter the following information:

- **SP name**: Enter the name of the Service Provider (SP).
- **IDP prefix**: Enter a prefix for the IDP that will be appended to the end of the IDP URLs.
  Alternatively, you can select **Generate unique prefix** to generate a random 16 digit alphanumeric string.
- **IDP address**: To configure the IDP address (and IDP settings below), you must have already configured the server's address in **Authentication** > **SAML IdP** > **General**.
- **SP entity id**: Enter the entity ID of the SP. Retrieve the SP entity id, SP ACS URL, and SP SLS URL from FortiSOAR by navigating to **Settings** > **Authentication** > **SSO**. Then click the **Service Provider Configuration** section to get these details.
  Alternatively, you can download the metadata of the SP from FortiSOAR and import the same here.
- **SP ACS (login) URL**: Enter the Assertion Consumer Service (ACS) login URL of the SP.
- **SP SLS (logout) URL**: Enter the Single Logout Service (SLS) logout URL of the SP.
- **SAML Attributes**: Map the User attributes to SAML attributes. This is needed so that users created in FortiSOAR have the correct details.
  SAML attributes must be configured as shown in the following image:

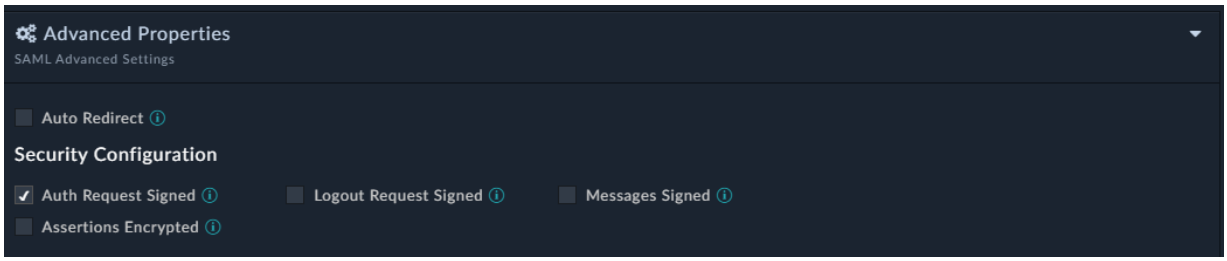
SAML Attribute	User Attribute	Actions
First Name	First name	
Last Name	Last name	
Email	Email	
Roles	FAC local group	

Important: You must not change the **SAML Attribute** names as these are the attribute names expected by FortiSOAR. You can change the **User Attribute** names as per your requirement.
The remaining fields can be left unmodified, or can be modified as per your requirement.
You must download the IdP metadata.

5. In FortiSOAR, navigate to **Settings** > **Authentication** > **SSO**, and then enter the following details in the `Identity Provider Configuration` section:

- **Entity ID**: Enter the **IDP entity id** from the `Create New SAML Service Provider` section mentioned in step 4.
- **Single Sign On URL**: Enter the **IDP single sign-on URL** from the `Create New SAML Service Provider section` mentioned in step 4.
- **Single Logout Request URL**: Enter the **IDP single logout URL** from the `Create New SAML Service Provider` section mentioned in step 4.
- **X509 Certificate**: Retrieve the signing certificate from IDP metadata that you have downloaded in step 4 and enter it in this field. The signing certificate is located under the `<md:KeyDescriptor use="signing">` key in the metadata xml file.

- **Advanced Properties**: In the `Security configuration` section, ensure that the **Auth Request Signed** checkbox is enabled:



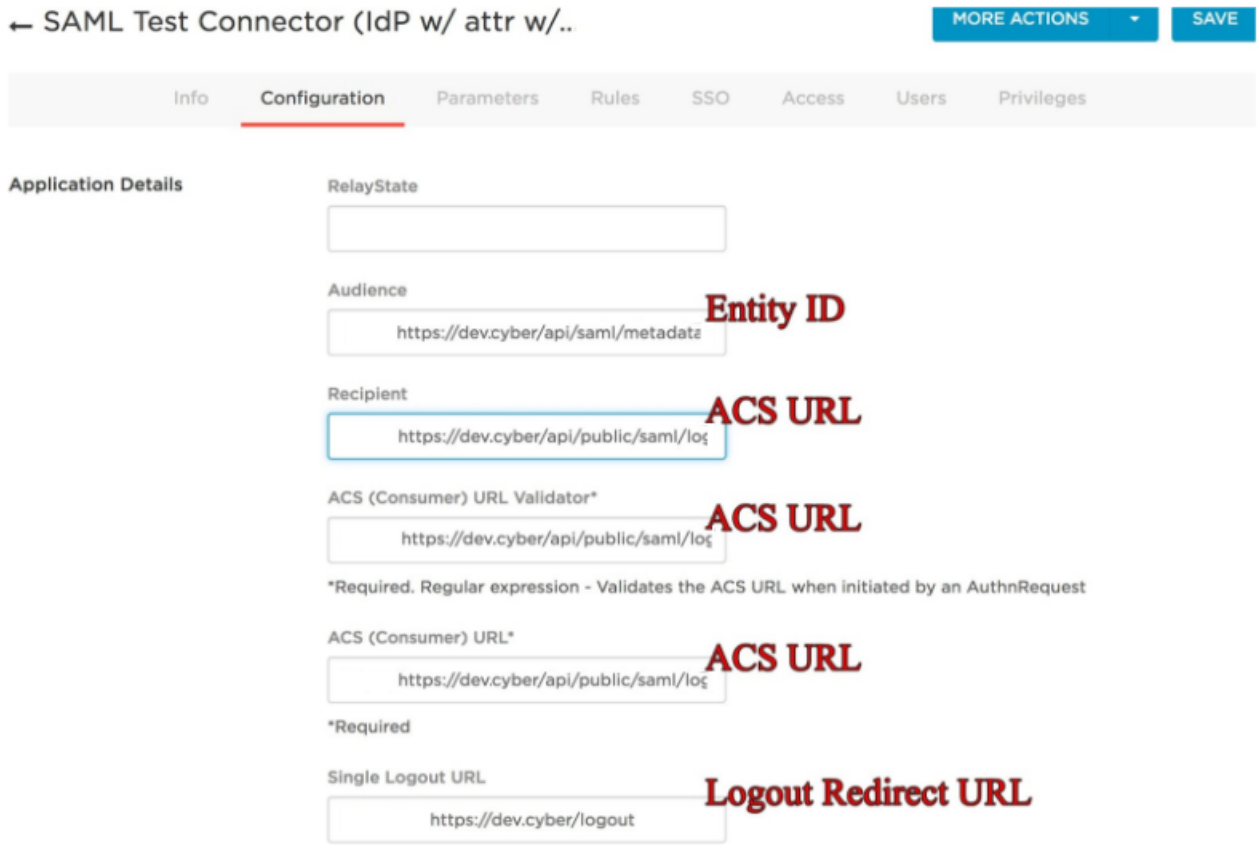
- For **Team and Role Mapping**, the Role name can be given as the 'User Group' name from FortiAuthenticator that is present in **Authentication** > **User Management** > **User Groups**. You can utilize an existing Group or create a new one as per your requirement. The login user should be from the same group as mentioned in 'Team and Role' mapping.

6. Click **Save** in FortiSOAR to save the changes to the IdP configuration.

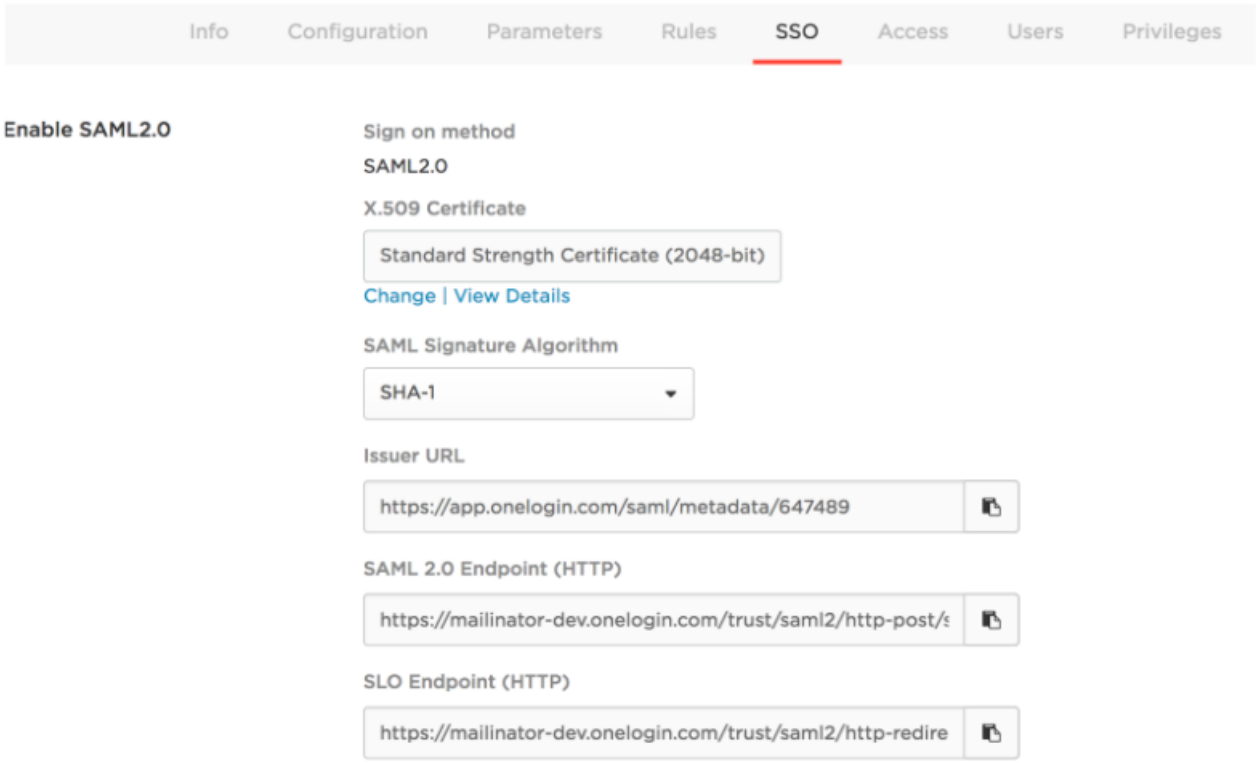
Configuring SAML in OneLogin

1. Log on to OneLogin as an administrator.
2. Create a new application in OneLogin. Navigate to **APPS** > **Company Apps** > **ADD APP**. In the Find Applications section, search for `saml` and select **SAML Test Connector (IDP w/attr w/sign response)**. Save the application.

3. Configure IdP. On the SAML Test Connector (IDP w/attr w/sign response), click the **Configuration** tab and enter your SP details as shown in the following image:



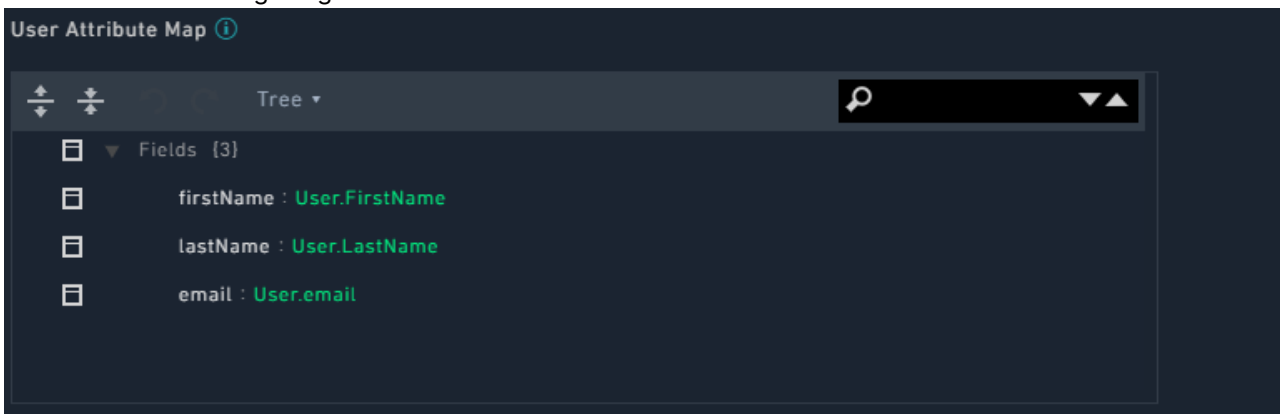
4. Get SSO details. On the SAML Test Connector (IDP w/attr w/sign response), click the **SSO** tab and you will see the SSO details of OneLogin (IdP) as shown in the following image:



5. Add the SSO details shown in step 4 in FortiSOAR. To add the SSO details, log on to FortiSOAR, click **Settings** > **Authentication** > **SSO**. In the `Identity Provider Configuration` section, enter the IdP details

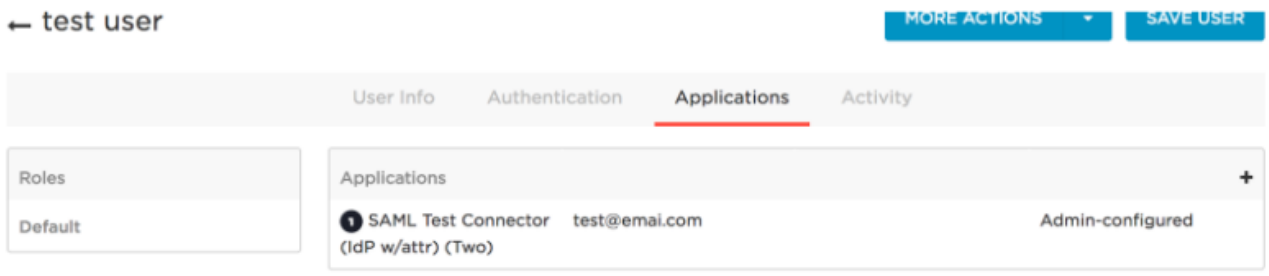
as shown in the following image:

6. Add the default user attribute mapping for OneLogin in FortiSOAR by updating the **User Attribute Map** as shown in the following image:



Note: You can change the default user attribute mapping if required.

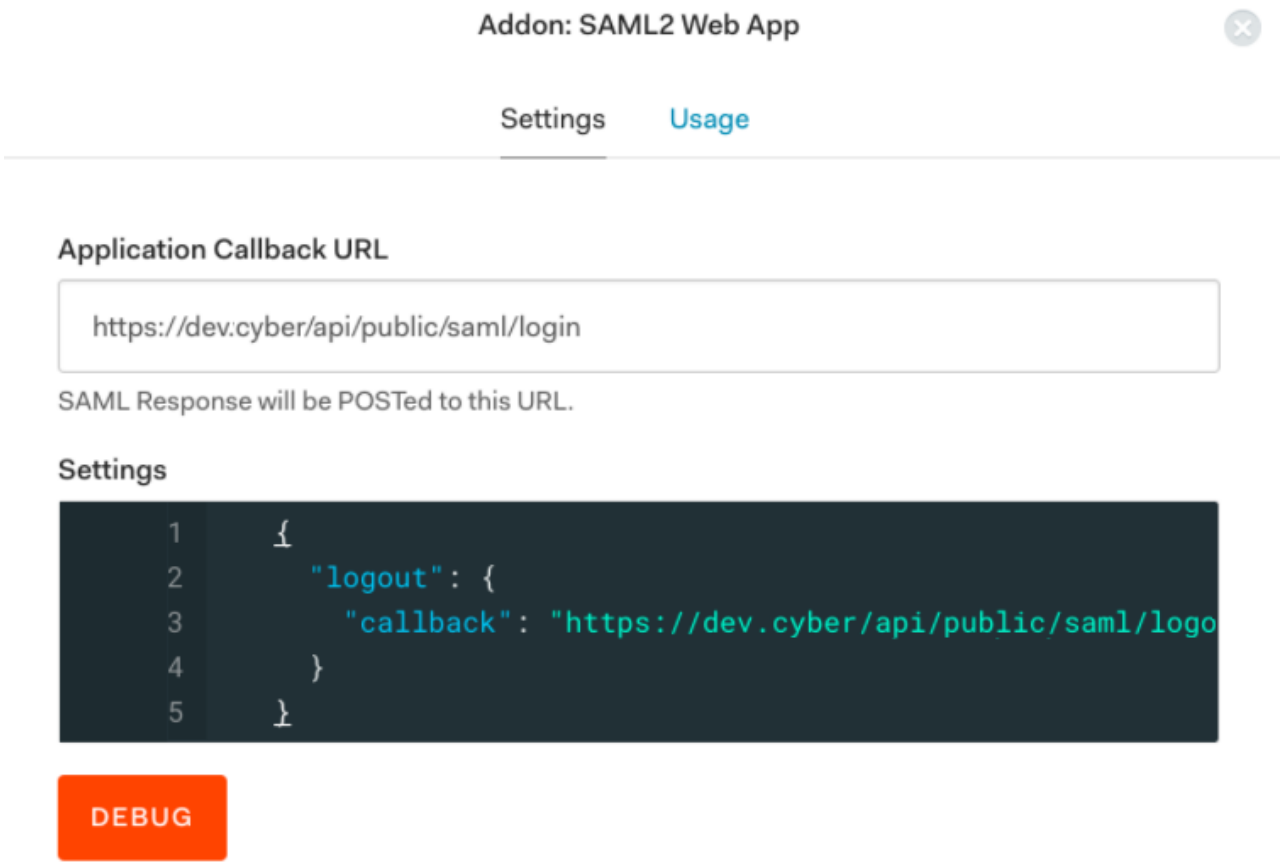
7. Click **Save** in FortiSOAR to save the changes to the IdP configuration and user attribute mapping.
8. Create a new user in OneLogin. Log on to OneLogin as an administrator and navigate to the **USERS** main menu and create a new user by clicking on **NEW USER** and entering relevant details. Once the user is created, open the user details, click the **Applications** tab and select the application created in step 2.
Note: While attaching the application to the user, the 'SAVE' button might be disabled. To enable the Save button, click any field and then press space or any key and then clear the space or character using backspace.



Once the user is created, you must assign the user a password by clicking **MORE ACTIONS**.

Configuring SAML in Auth0

1. Log on to Auth0 as an administrator.
2. Create a new application in Auth0. In the `Clients` section, create a new client by selecting **Regular Web Applications**.
3. Configure IdP (Auth0). In Auth0, go to the **Addon** tab of the application you have created in step 1 and select **SAML2 WEB APP**. On the `Settings` page that appears, in the **Application Callback URL** field enter the `ACS URL` from your SP configuration. In the **Settings** field, uncomment the logout portion and set the **callback** field to the value that is present in the **Logout POST URL** field that is present in the `Service Provider` section on the FortiSOAR `SSO` page, as shown in the following image:



4. Get SSO details. Download `Identity Provider Metadata`, by navigating to **App configuration** > **Addons** > **SAML2** > **Usage** > **Identity Provider Metadata**. Click **Download**. The Identity Provider Metadata appears

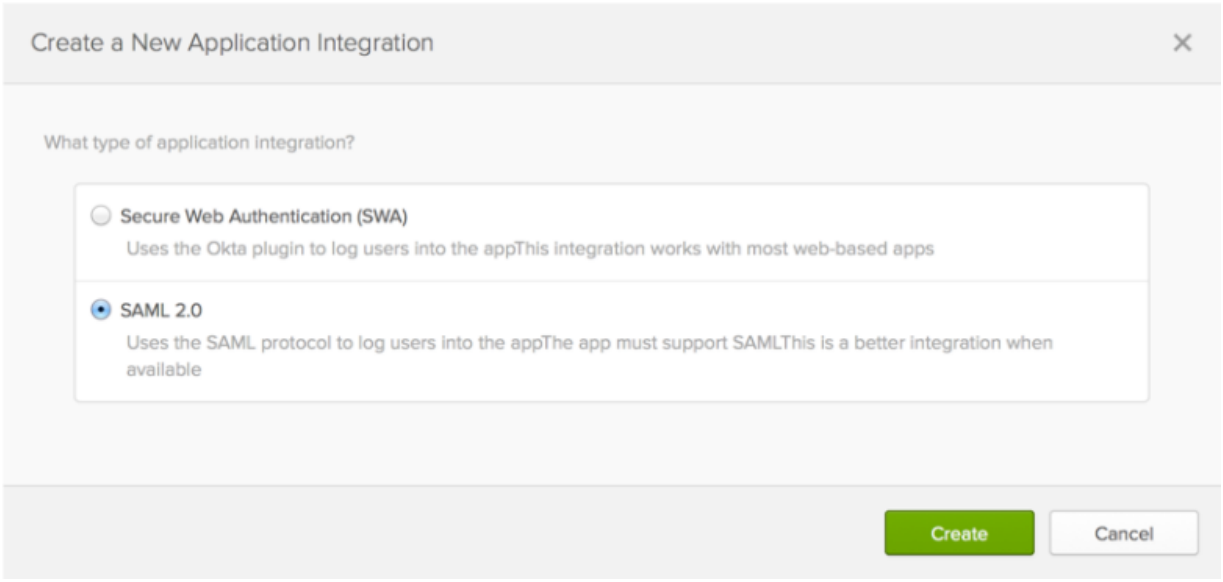
as shown in the following image:

```
<EntityDescriptor entityID="urn:o1084360.auth0.com" xmlns="urn:oasis:names:tc:SAML:2.0:metadata">
  <IDPSSODescriptor protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <KeyDescriptor use="signing">
      <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
        <X509Data>
          <X509Certificate>MIIC/
zCCAeegAwIBAgIJZAzJWzHeGWUiMA0GCSqGSIb3DQEBCwUAMB0xGzAZBgNVBAMTEm8xMDg0MzYwLmF1dGgwLmNvbTAeFw0xNzA0MDEwNzMwMDBaFw0zMDEyMDkwNzMwMDBaMB0xGzAZBgNVBAMTEm8xMDg0
MzYwLmF1dGgwLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKhOggZ4r3jq2iAgrFNZv0IoEJZVKA6mhmpfFiqs8vAliUbtEireTpSfSO1SP/
YaW7ODamSBZrb06VRCmt+LzsGwsXPJTZDwqORYraA3w4dSp5mnC7VLjyTrmMazRbcWo6Egq6N6v+OE48z0Qtd/
Fb5wTdi8yKmxVV8xNYIEhRdcTIRoTjYZ0oc1j26BX7xOe3wYBIzly0JzkRCkZjpEOFZMIeC8cmMEJdDS3UEV/4nYsgLVi4CB/Y9WWf4kbyLE1pTAKWBsbdgbBk5aYRxu1qNhu3ZuUT7AV/
PEgXoBJIsFJiru37ORVp0Smi4F/Ji2rZk85Loj3hG0+G6CFYkzKnMCAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUuRCNNTfAOfiQNeZdDE0nQUwza9QwDgYDVR0PAQH/
BAQDAgKEMA0GCSqGSIb3DQEBCwUAA4IBAQBDGs3tN5W9wByql+hHZ83268Cow3t8iTeWmkWG9PLYZIL6AviKgN7XaBvHos05/
Kf0p5A1MoXKJ46QkUIEhusDIufBGNC7i3c3UpZYgaLcIDrf5BXPjUYCQW+ogiQPCuadrZjeImgAmaMsV/ChEucbYUD/
mDwUqLc3RQ+O+cBHtfQ0eGSivAogmObbkoB3xWwL1hUn+XI3UEC3zlLTNj72FXadDtS7Pp9p4acIOnmlkR/
YnqOB1MxUNLcM7ainvSwgwSU6zu81PUZkhufbBVnVA2QXHx0zrKVENWhLBBf2Dbn9W0kPychGxDrGmTCBF+VZTqdZf/n9a7EO0AGbK7Nw</X509Certificate>
        </X509Data>
      </KeyInfo>
    </KeyDescriptor>
    <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect" Location="https://o1084360.auth0.com/samlp/
o9ApfoancBKS5VqQv0DphA1FLyWHtgZm/logout"/>
    <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://o1084360.auth0.com/samlp/
o9ApfoancBKS5VqQv0DphA1FLyWHtgZm/logout"/>
    <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</NameIDFormat>
    <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:persistent</NameIDFormat>
    <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:transient</NameIDFormat>
    <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect" Location="https://o1084360.auth0.com/samlp/
o9ApfoancBKS5VqQv0DphA1FLyWHtgZm"/>
    <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://o1084360.auth0.com/samlp/
o9ApfoancBKS5VqQv0DphA1FLyWHtgZm"/>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
FriendlyName="E-Mail Address" xmlns="urn:oasis:names:tc:SAML:2.0:assertion"/>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
FriendlyName="Given Name" xmlns="urn:oasis:names:tc:SAML:2.0:assertion"/>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
FriendlyName="Name" xmlns="urn:oasis:names:tc:SAML:2.0:assertion"/>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
FriendlyName="Surname" xmlns="urn:oasis:names:tc:SAML:2.0:assertion"/>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"
FriendlyName="Name ID" xmlns="urn:oasis:names:tc:SAML:2.0:assertion"/>
  </IDPSSODescriptor>
</EntityDescriptor>
```

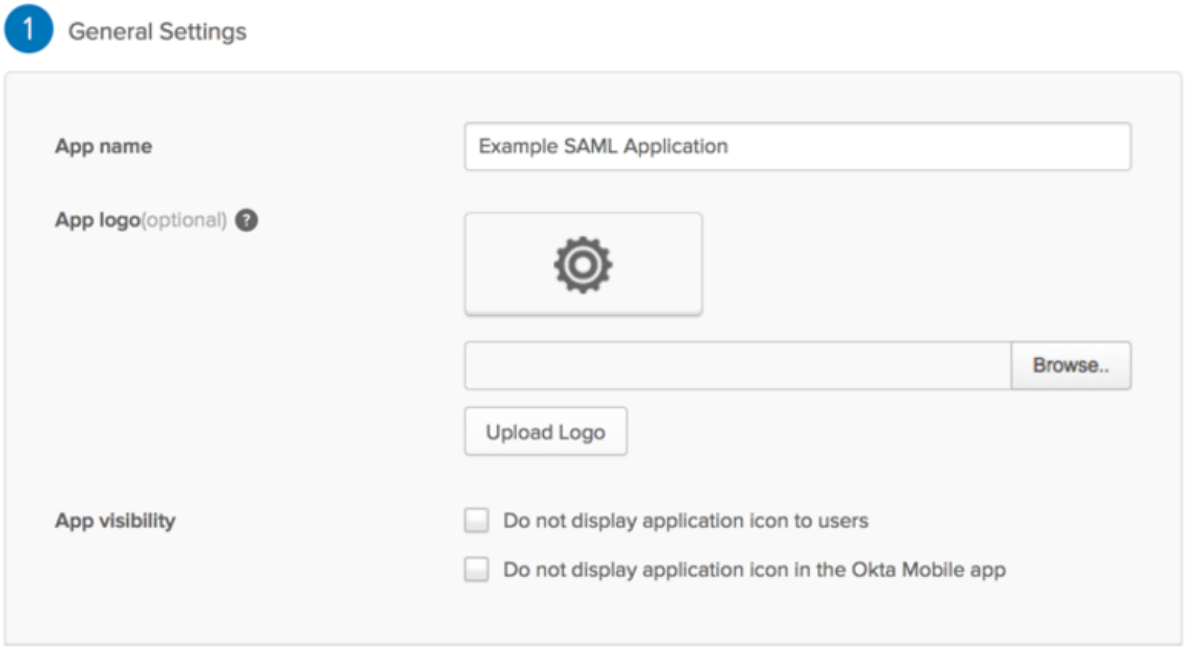
5. Add the SSO details shown in step 4 in FortiSOAR. To add the SSO details, log on to FortiSOAR, click **Settings** > **Authentication** > **SSO**. In the `Identity Provider Configuration` section, use the Identity Provider Metadata to fill in the **Entity ID**, **Single Sign On URL**, **X509 Certificate**, and **Single Logout Request URL** details.
   Based on Identity Provider Metadata screenshot in step 4, you would fill in the SSO details in FortiSOAR as follows:
   - In the **Entity ID** field enter the following value that you get from the Identity Provider Metadata:
     `urn:o1084360.auth0.com`
   - In the **Single Sign On URL** field enter the following value that you get from the Identity Provider Metadata:
     `https://o1084360.auth0.com/samlp/o9ApfoancBKS5VqQv0DphA1FLyWHtg`
   - In the **Single Logout Request URL** field enter the following value that you get from the Identity Provider Metadata:
     `https://o1084360.auth0.com/samlp/o9ApfoancBKS5VqQv0DphA1FLyWHtgZm/logout`
   - In the **X509 Certificate** field enter the following value that you get from the Identity Provider Metadata:
     ```
     zCCAeegAwIBAgIJZAzJWzHeGWUiMA0GCSqGSIb3DQEBCwUAMB0xGzAZBgNVBAMTEm8xMDg0MzYwLmF1dGgwLmNvbTAeFw0xNzA0MDEwNzMwMDBaFw0zMDEyMDkwNzMwMDBaMB0xGzAZBgNVBAMTEm8xMDg0
     MzYwLmF1dGgwLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKhOggZ4r3jq2iAgrFNZv0IoEJZVKA6mhmpfFiqs8vAliUbtEireTpSfSO1SP/
     YaW7ODamSBZrb06VRCmt+LzsGwsXPJTZDwqORYraA3w4dSp5mnC7VLjyTrmMazRbcWo6Egq6N6v+OE48z0Qtd/
     Fb5wTdi8yKmxVV8xNYIEhRdcTIRoTjYZ0oc1j26BX7xOe3wYBIzly0JzkRCkZjpEOFZMIeC8cmMEJdDS3UEV/4nYsgLVi4CB/Y9WWf4kbyLE1pTAKWBsbdgbBk5aYRxu1qNhu3ZuUT7AV/
     PEgXoBJIsFJiru37ORVp0Smi4F/Ji2rZk85Loj3hG0+G6CFYkzKnMCAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUuRCNNTfAOfiQNeZdDE0nQUwza9QwDgYDVR0PAQH/
     BAQDAgKEMA0GCSqGSIb3DQEBCwUAA4IBAQBDGs3tN5W9wByql+hHZ83268Cow3t8iTeWmkWG9PLYZIL6AviKgN7XaBvHos05/
     Kf0p5A1MoXKJ46QkUIEhusDIufBGNC7i3c3UpZYgaLcIDrf5BXPjUYCQW+ogiQPCuadrZjeImgAmaMsV/ChEucbYUD/
     mDwUqLc3RQ+O+cBHtfQ0eGSivAogmObbkoB3xWwL1hUn+XI3UEC3zlLTNj72FXadDtS7Pp9p4acIOnmlkR/
     YnqOB1MxUNLcM7ainvSwgwSU6zu81PUZkhufbBVnVA2QXHx0zrKVENWhLBBf2Dbn9W0kPychGxDrGmTCBF+VZTqdZf/n9a7EO0AGbK7Nw
     ```
   - Click **Save** in FortiSOAR to save the changes to the IdP configuration and user attribute mapping.

Configuring SAML in Okta

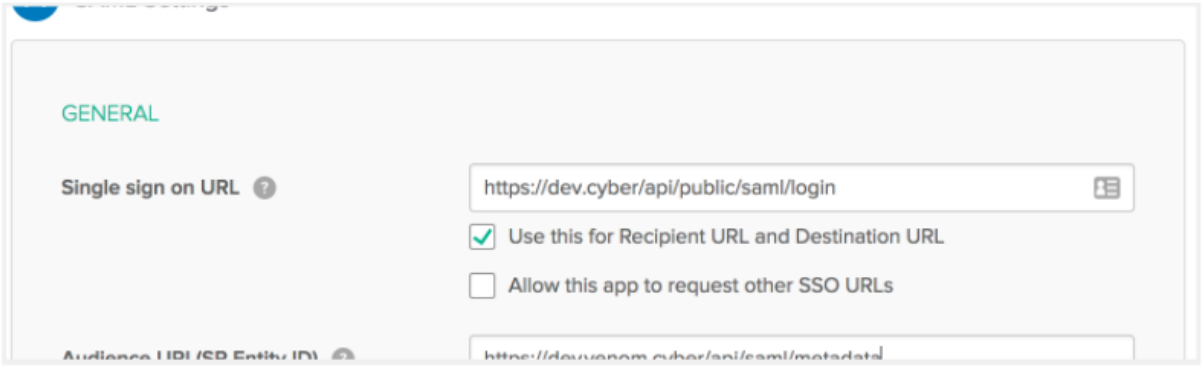
1. Log on to Okta as an administrator.
   If you don't have an Okta organization, you can create a free Okta Developer Edition organization.
2. Create a new application in Okta and configure IdP in the application.
   - In Okta, click the blue **Admin** button.
   - On the `Applications` tab, click **Add Applications** > **Create New App**.
   - On the `Create a New Application Integration` dialog, select **SAML 2.0** and click **Create**.



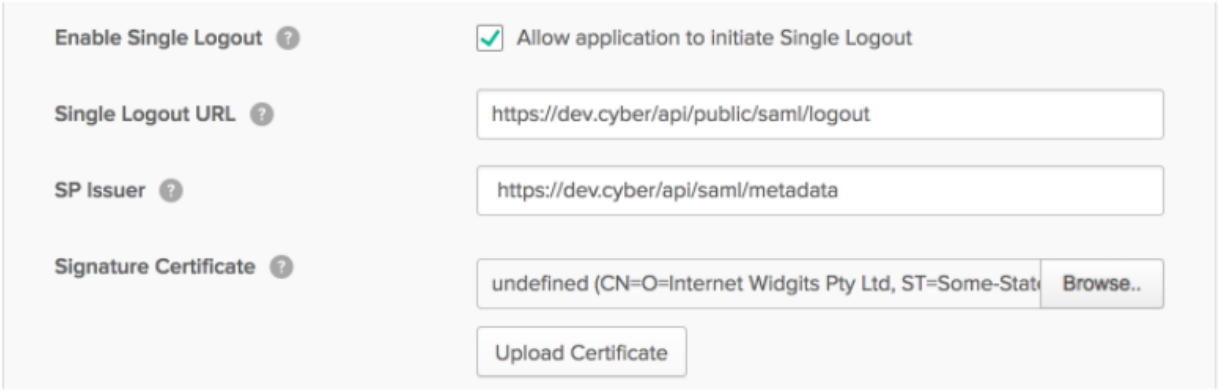
3. Configure IdP.
   - In the newly created application, on the `General Settings` dialog, in the **App name** field, enter the application name and click **Next**.



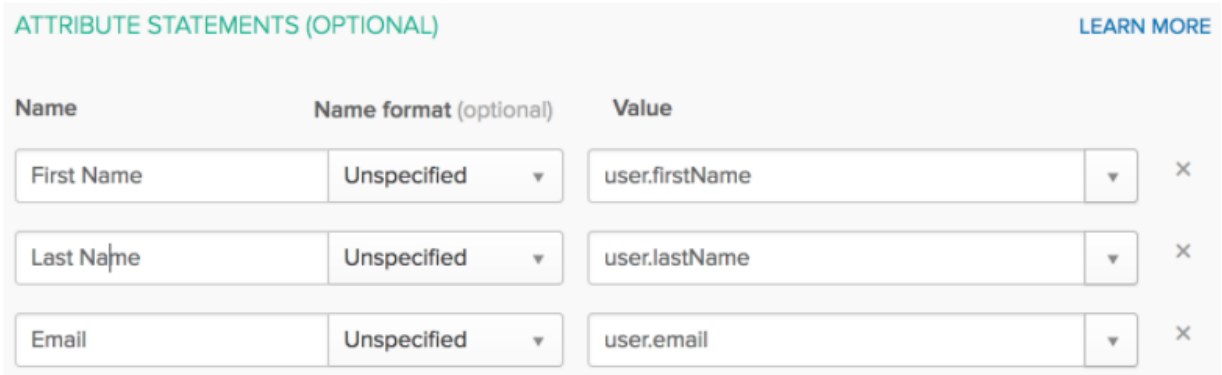
- On the `Configure SAML` dialog, in the `SAML Settings` section, in the **Single Sign On URL** field, enter or paste the `SP ACS URL` and in the **Audience URI** field, enter or paste the `SP Entity ID`.



- Click **Show Advanced Settings**.
- Select the **Enable Single Logout** checkbox.
  In the **Single Logout URL** field, enter or paste the SP Logout POST URL.
  In the **SP Issuer** field, enter or paste the SP Entity ID.
  In the **Signature Certificate** field, browse to where you have downloaded the SP X509 certificate and click **Upload Certificate**.

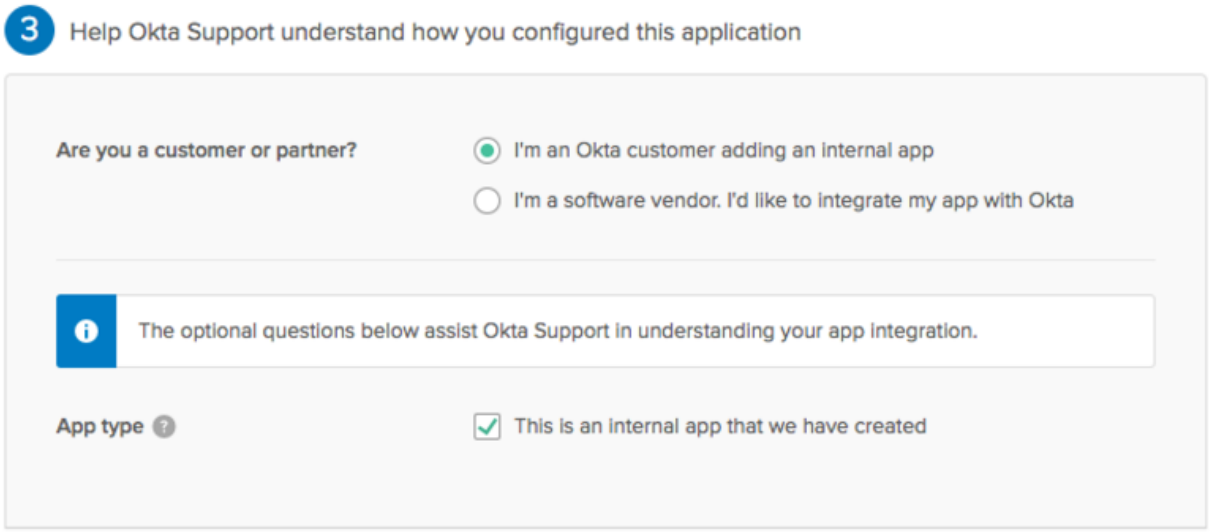


- In the `ATTRIBUTE STATEMENTS (OPTIONAL)` section, set the mapping as shown in the following image:



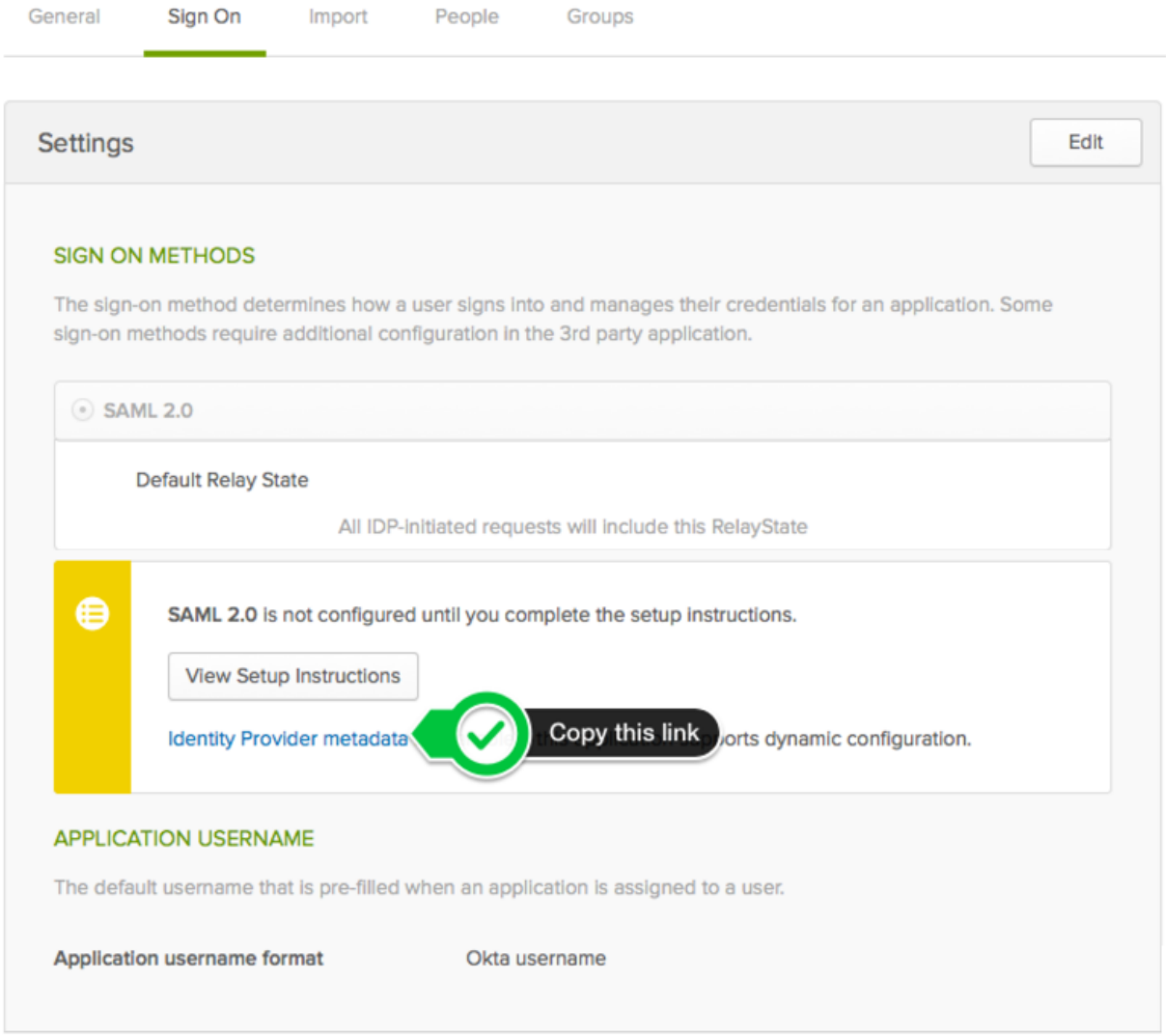
  Note: You must remember the attribute names specified in the above image. You will require to map these user attribute names while configuring the **User Attribute Map** on the SSO page in FortiSOAR.
- Click **Next**.

- On the `Help Okta Support understand how you configured this application` dialog, select **I'm an Okta customer adding an internal app**, and **This is an internal app that we have created**.

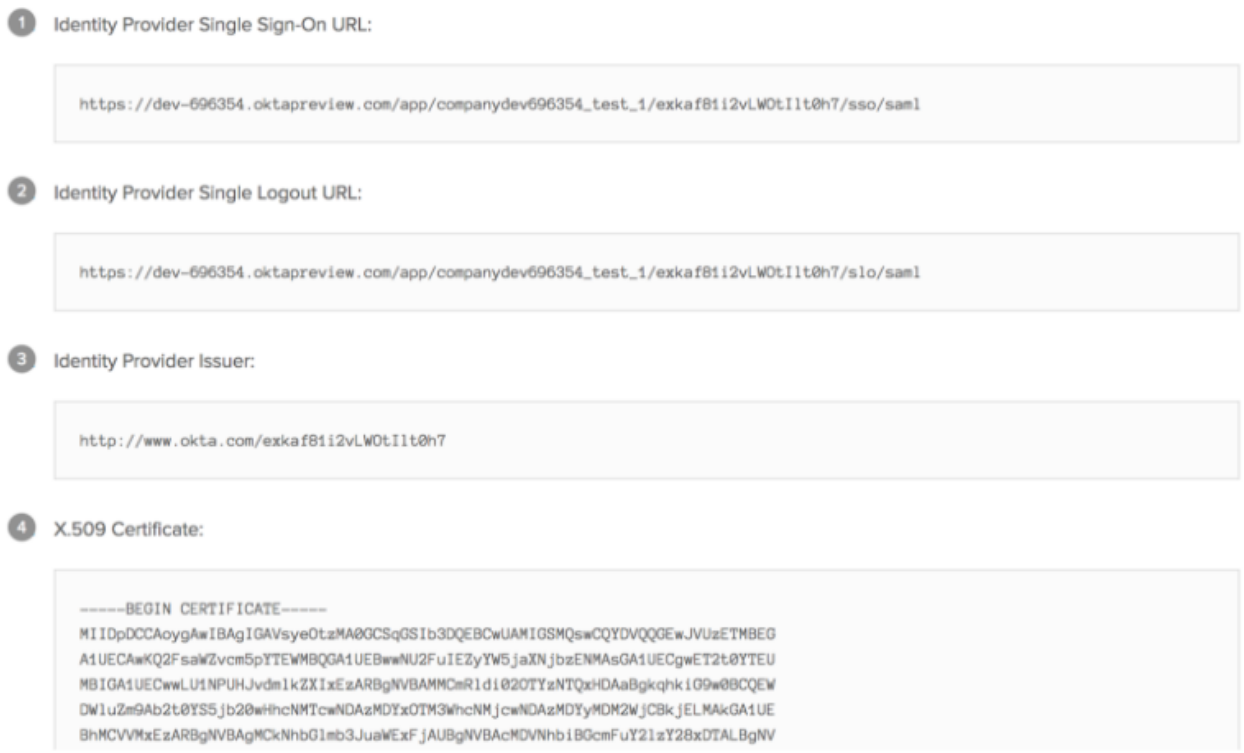


- Click **Finish**.
  The `Sign On` tab of your newly created SAML application gets displayed. Keep this page open in a separate tab or browser window as you will require the information present on this page to complete

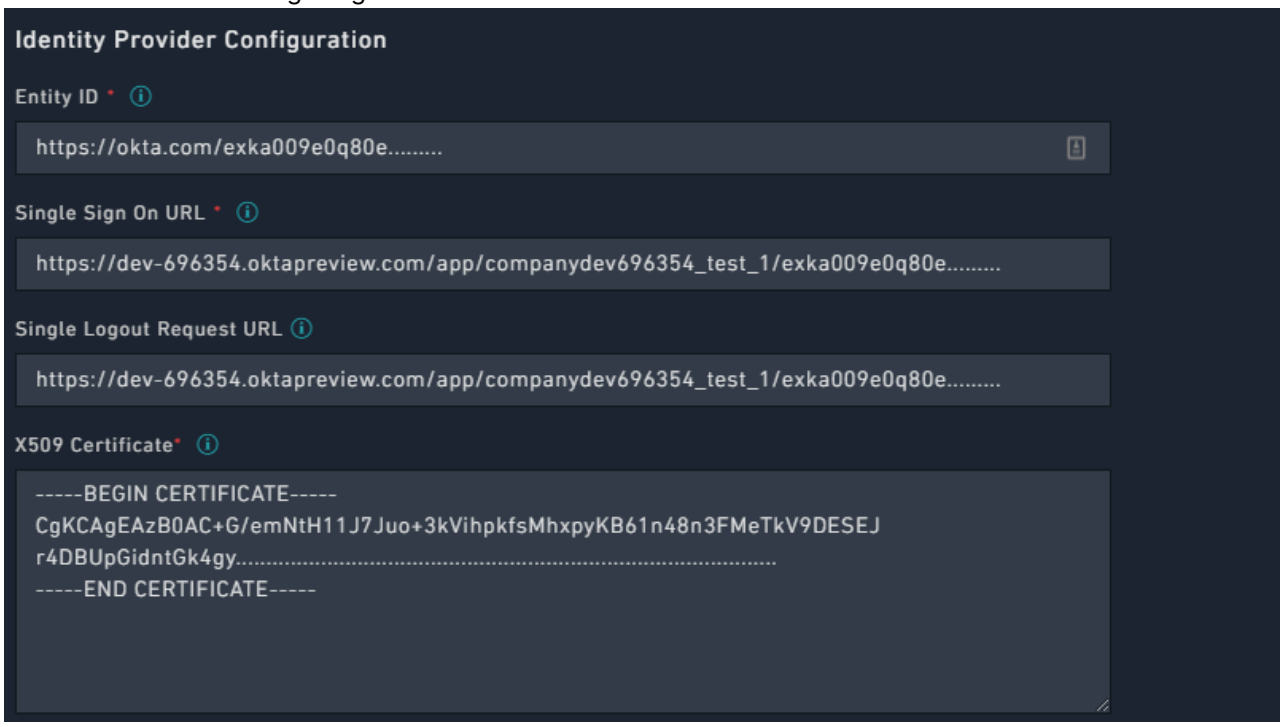
the `Identity Provider Configuration` section in FortiSOAR.



4. Get SSO details. Click **View Setup Instructions** and information as shown in the following image:

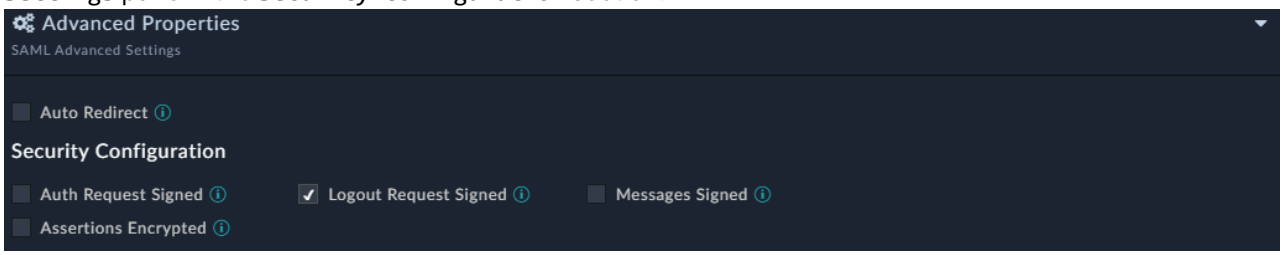


5. Add the SSO details shown in step 4 in FortiSOAR. To add the SSO details, log on to FortiSOAR, click **Settings** > **Authentication** > **SSO**. In the `Identity Provider Configuration` section, enter the IdP details as shown in the following image:

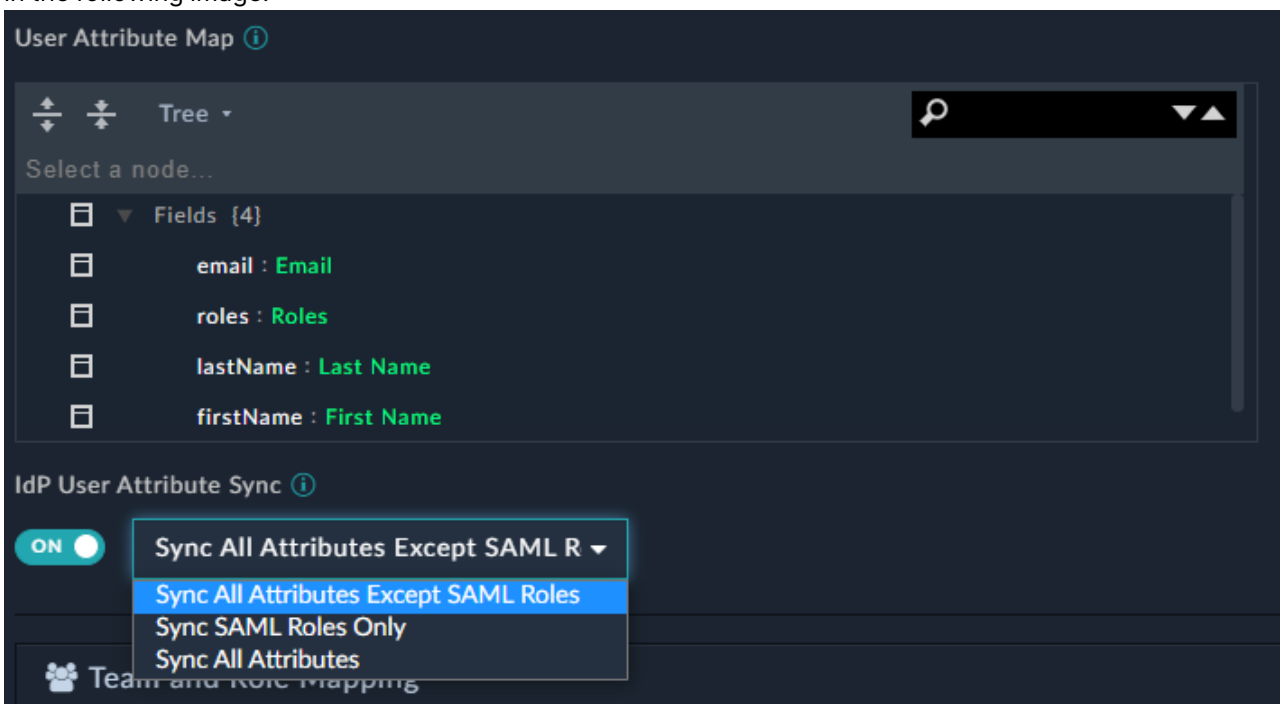


Note: The LogoutRequest message for Okta must be signed for Single Logout (SLO). Therefore, you must

select the **Logout Request Signed** checkbox that is present in the `Advanced Properties SAML Advanced Settings` pane in the `Security Configuration` section.

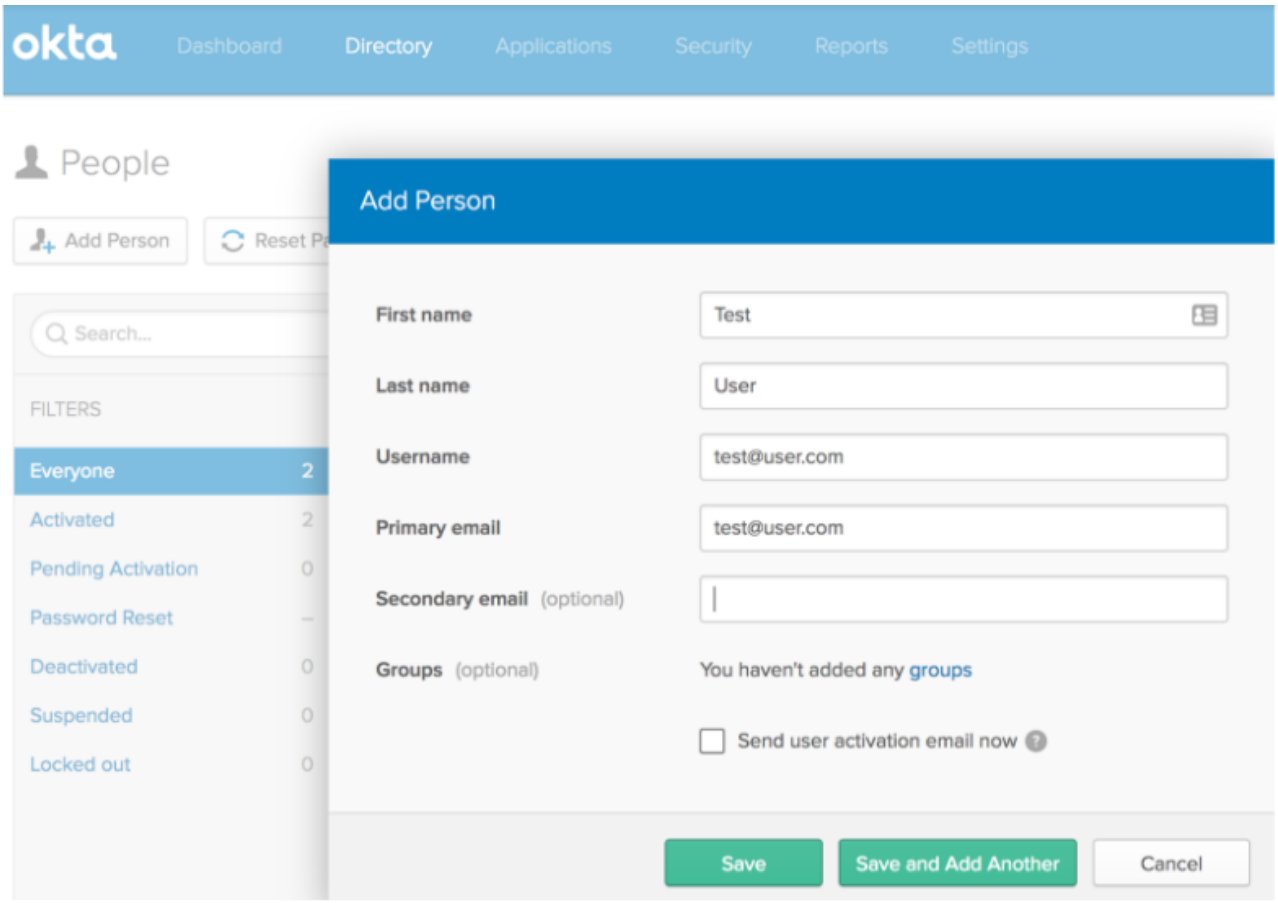


6. Add the default user attribute mapping for Okta in FortiSOAR by updating the **User Attribute Map** as shown in the following image:

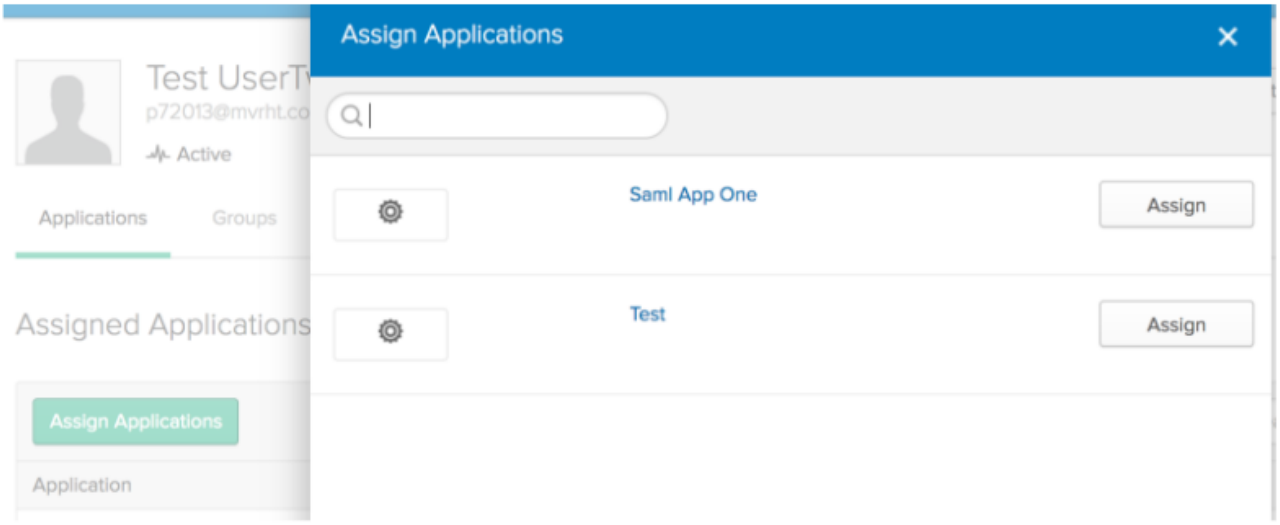


Note: The IdP keys, the keys on the right side, are obtained from the `ATTRIBUTE STATEMENTS (OPTIONAL)` section in Okta, as specified in step 3. You can change the default user attribute mapping later if required.

7. Click **Save** to complete the SSO configuration in FortiSOAR.
8. Create a new user in Okta. Log on to Okta as an administrator and navigate **Directory** > **People** > **Add Person** and enter all the user details.

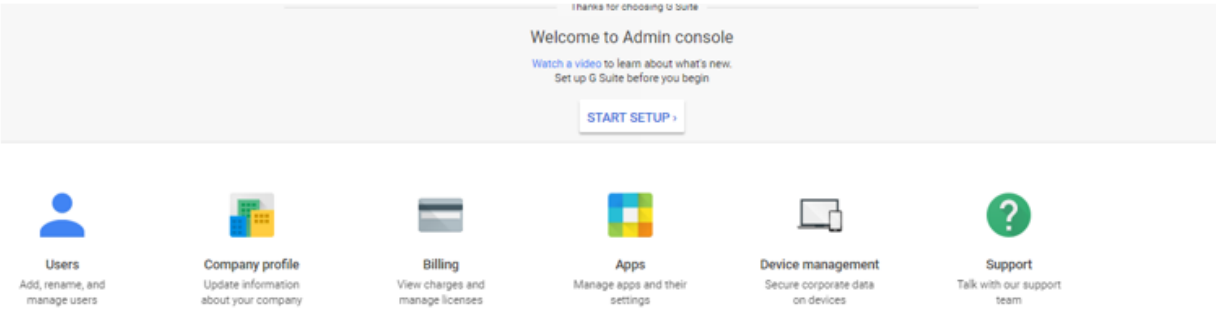


Once the user is created and activated successfully, you can assign this user to the SAML application that you have created. Click on a user to get the user details, and then assign the user to an application using the `Assign Applications` dialog as shown in the following image:

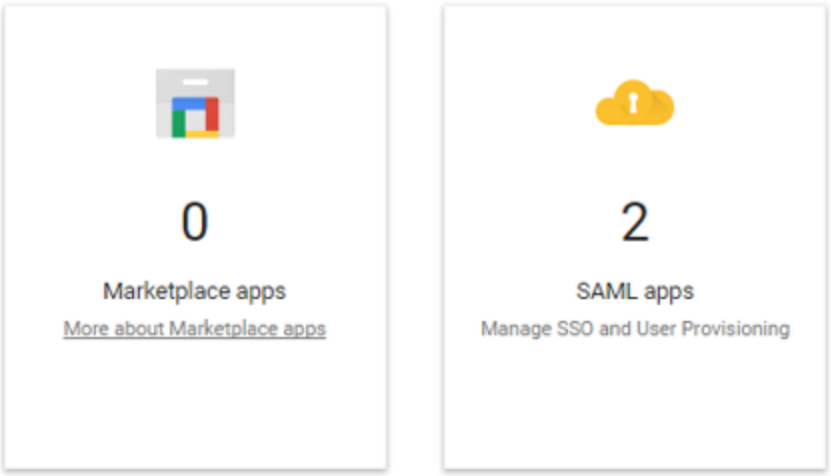


Configuring SAML in Google

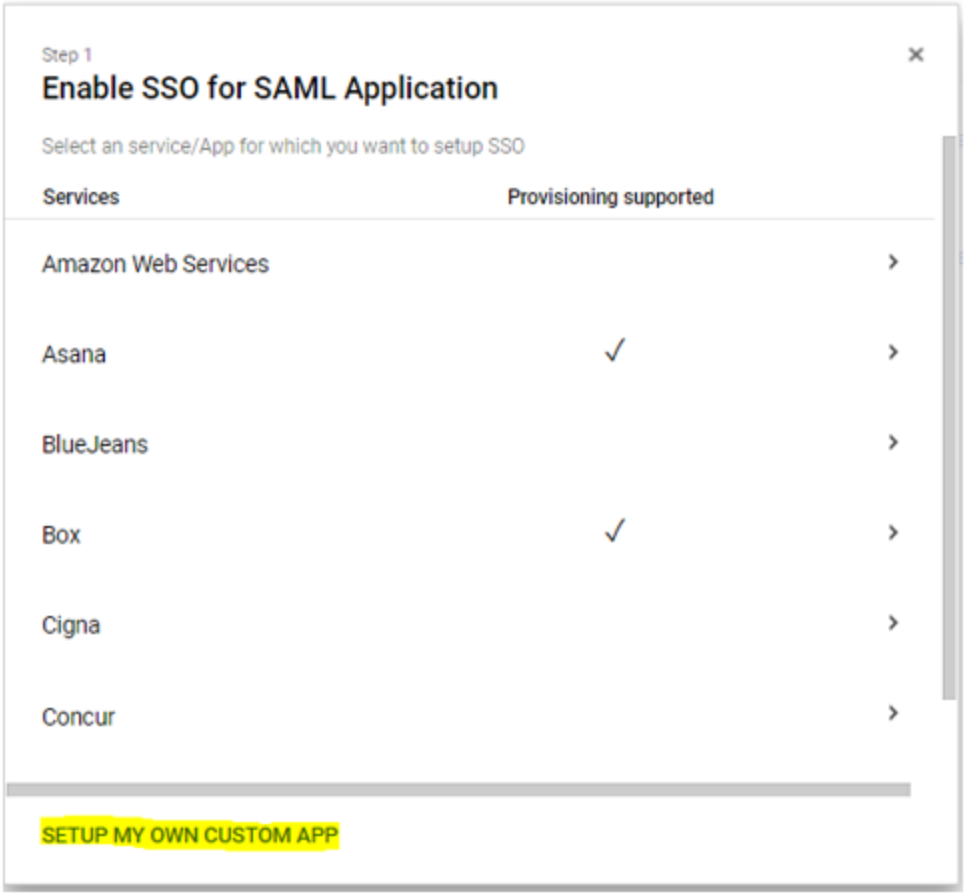
1. Ensure that you have Administrator access for your G Suite account and log on to G Suite using the admin account.
2. Configure IdP.
   - On your `Admin` console, click **Apps**.



   - Click **SAML apps**. On the `SAML` page, click **+** on the right bottom corner, to add a new SAML Application.

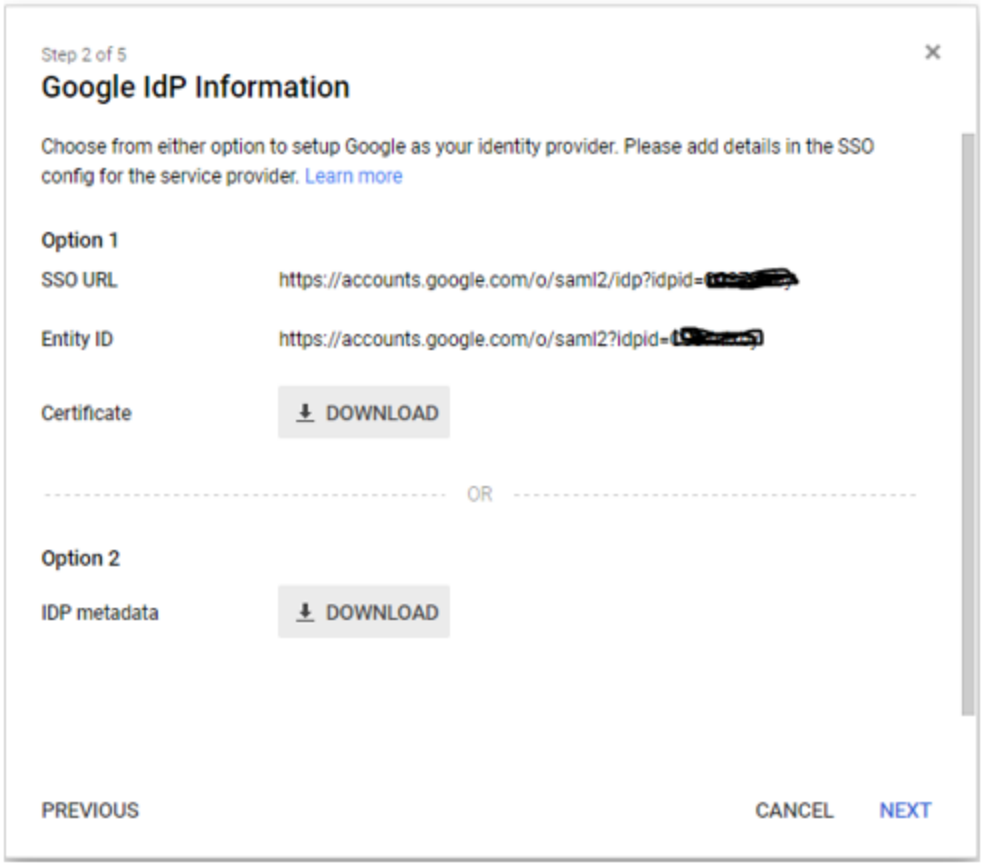


- On the `Enable SSO for SAML Application` page, click **SETUP MY OWN CUSTOM APP**.



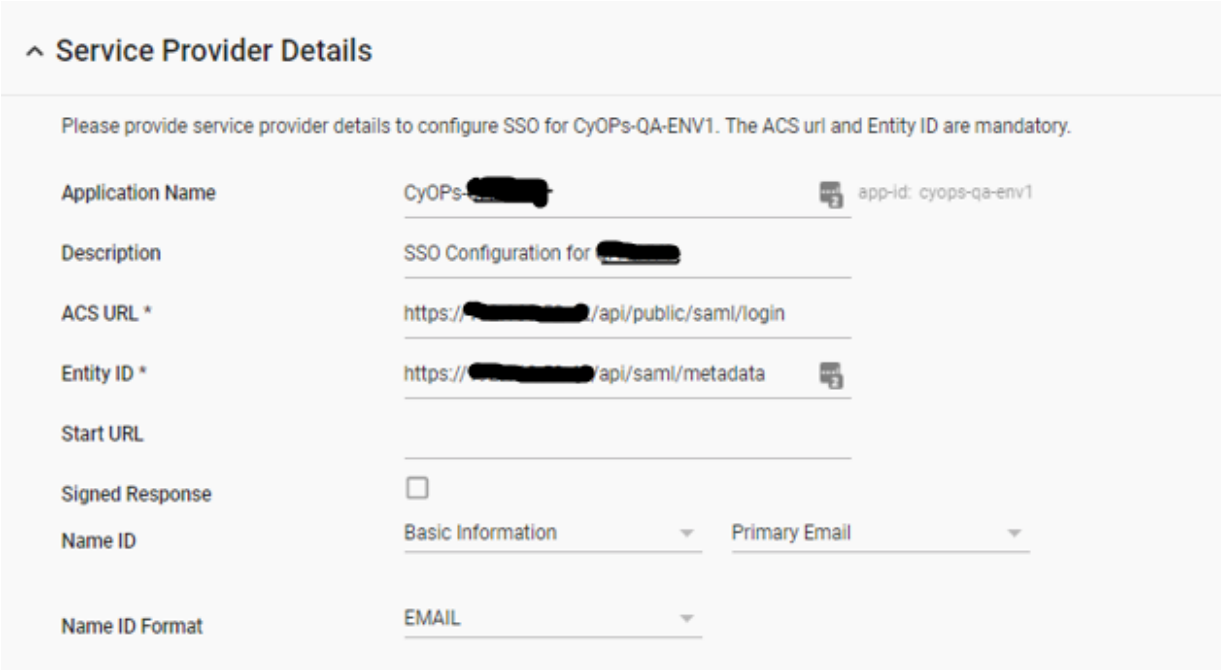
- Click **Next** to display the Google IdP information. Save the Google IdP information and download the `Certificate`.

You will require the IdP information for Google to configure SSO within FortiSOAR.

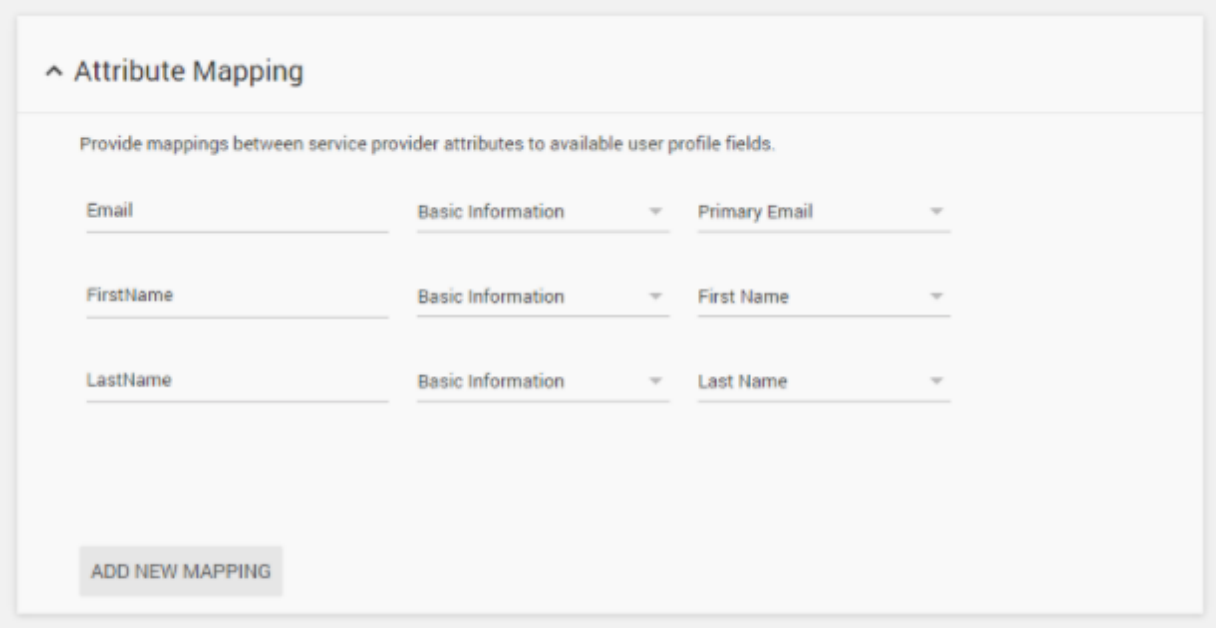


- Click **Next** and add basic information about the App, such as **Name** and Description and then click **Next**.
- On the `Service Provider Details` page, enter the **Entity ID** and **ACS URL** from the `Service Provider` section in FortiSOAR. Log on to FortiSOAR and navigate to **Settings** > **Authentication** > **SSO**, go to the

`Service Provider` section to get the details. See Configuring SAML in FortiSOAR.



- Click **Next** and add more attribute mapping as required.

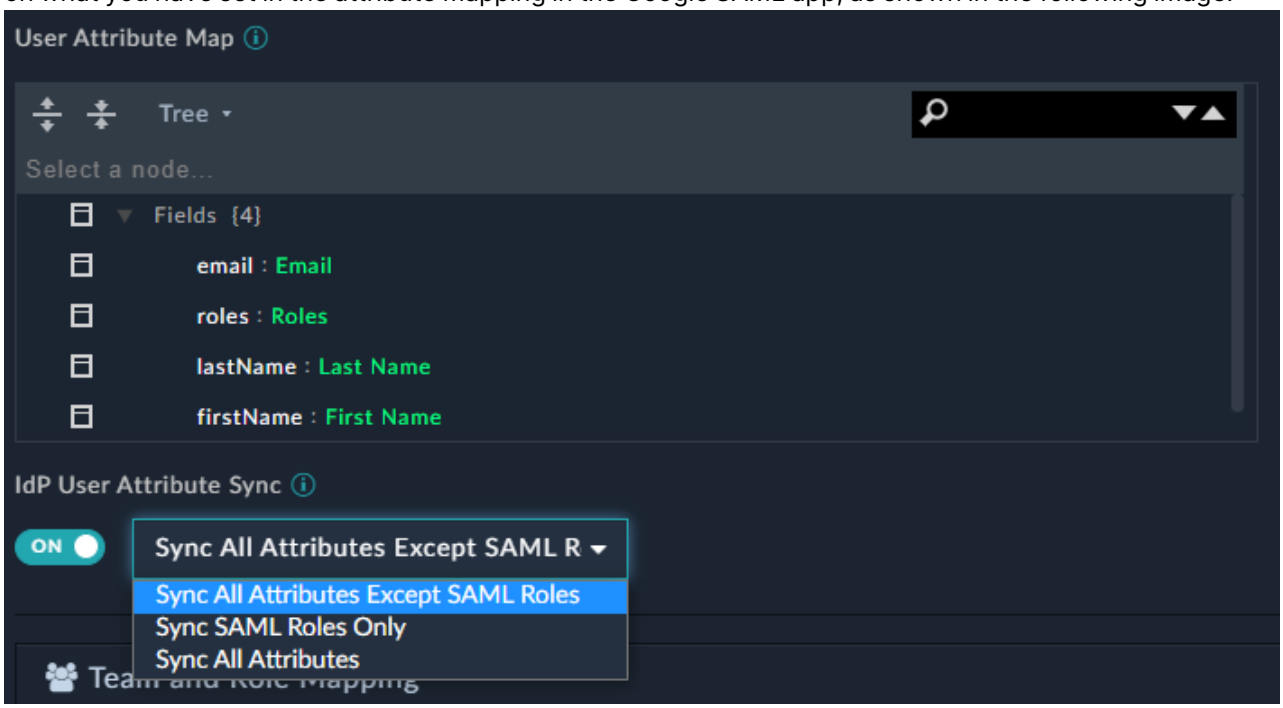


- Save the app configuration and click **Exit**.
- Set up user access for the Google SAML App, see Set up your own custom SAML application.

3. Add the SSO details saved in step 2 in FortiSOAR. To add the SSO details, log on to FortiSOAR, click **Settings**> **Authentication** > **SSO**. In the `Identity Provider Configuration` section, enter the Google IdP details and certificate as shown in the following image:

Note: Google SAML app does not provide a Logout URL. Therefore, users remain logged into their Google account even if they log off from FortiSOAR.
In FortiSOAR the **Single Logout Request URL** field is optional and can be left blank.

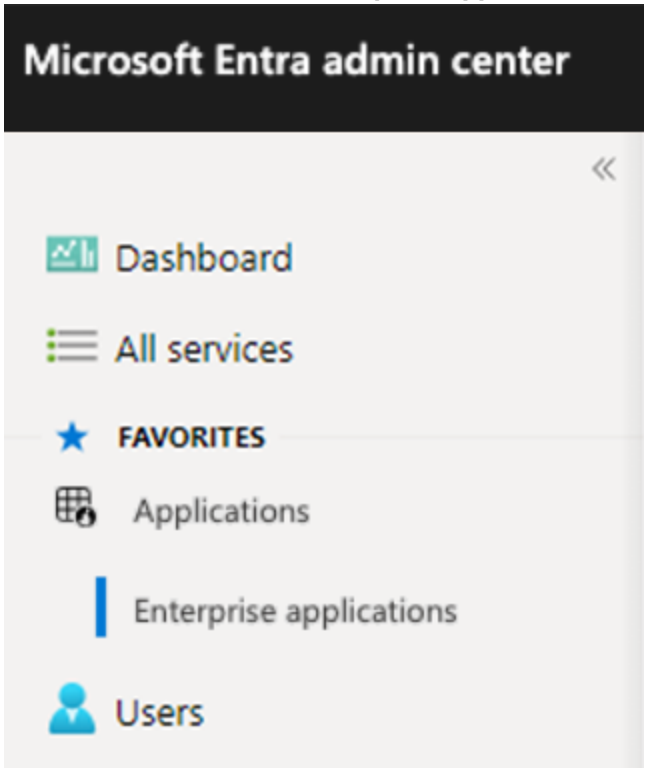
4. Add the default user attribute mapping for Google in FortiSOAR by updating the **User Attribute Map**, based on what you have set in the attribute mapping in the Google SAML app, as shown in the following image:



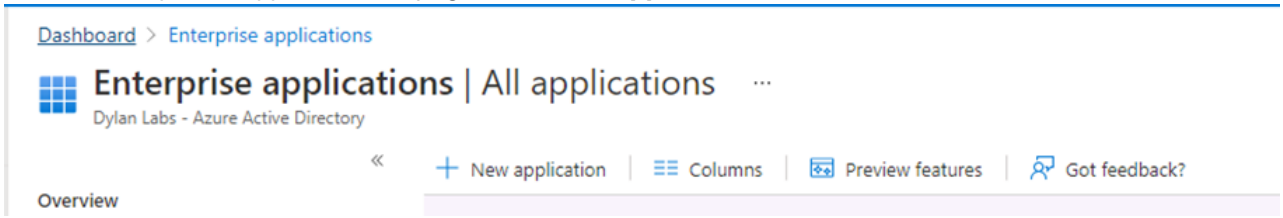
5. Click **Save** in FortiSOAR to save the changes to the IdP configuration.

Configuring SAML in Microsoft Entra ID (formerly Azure AD)

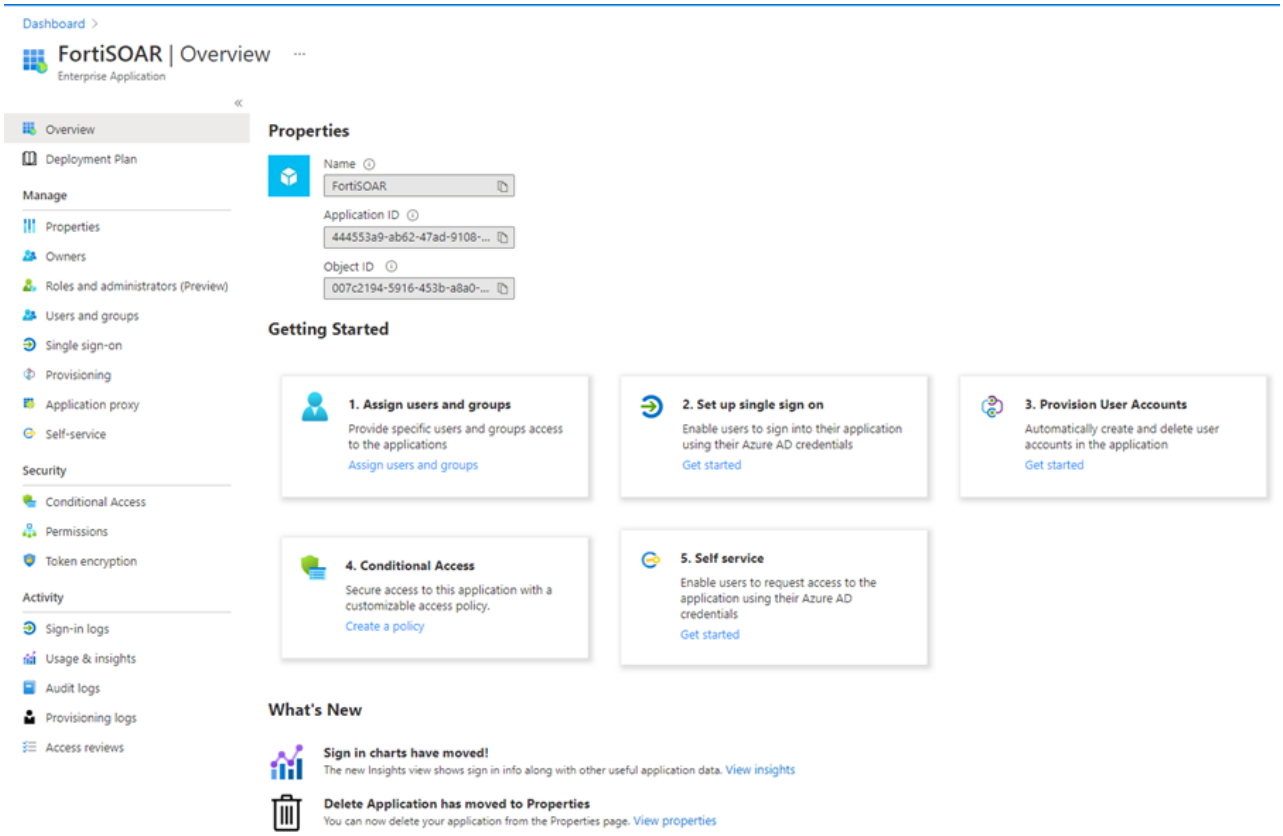
1. Open the Microsoft Entra ID (formerly Azure Active Directory/Azure AD) portal: https://entra.microsoft.com/#view/Microsoft_AAD_IAM/TenantOverview.ReactView (formerly https://aad.portal.azure.com/).
2. From the left menu, click **Enterprise Applications**.



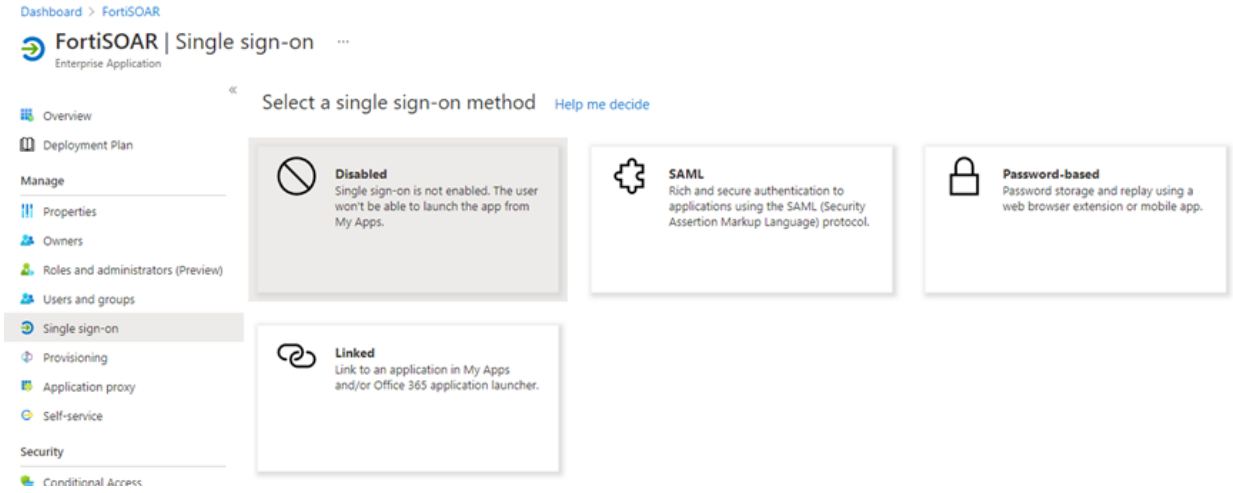
3. On the `Enterprise Applications` page, click **New Application**.



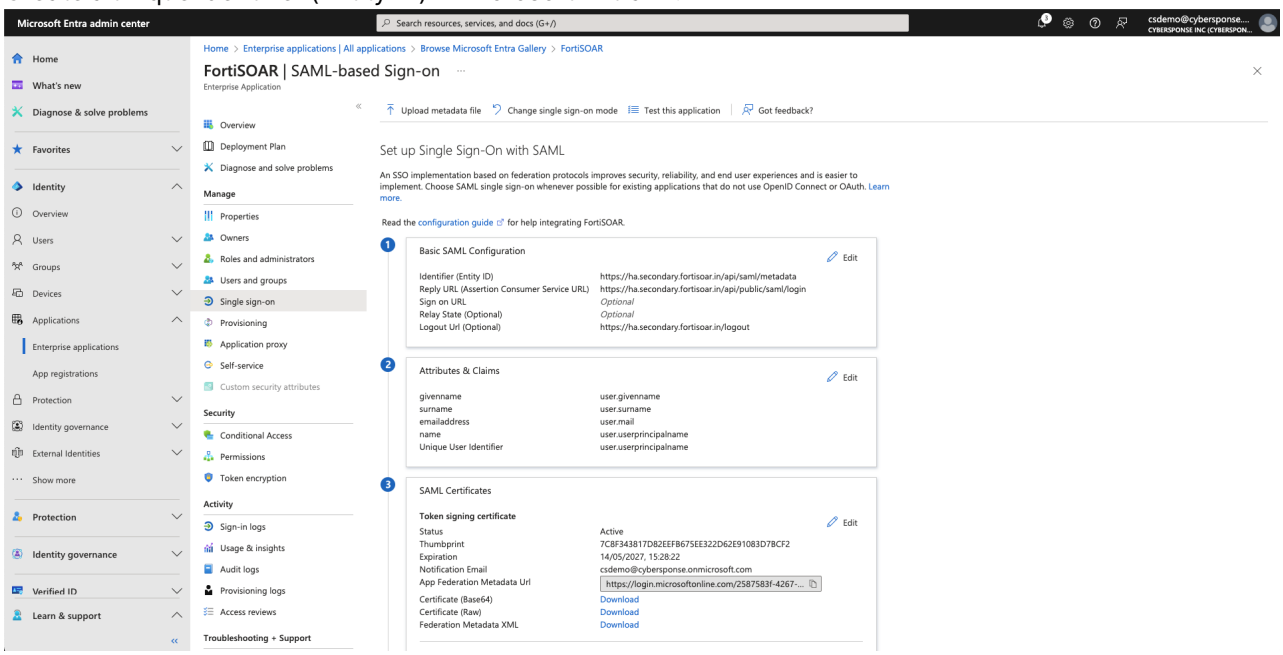
4. Click **Create Your Own Application**.
5. Enter `FortiSOAR` in the **Name** field and and click **Integrate any other application you don't find in the gallery (Non-gallery)**, and then click **Create**.
6. Follow the steps mentioned in the `Getting Started` section such as adding users/groups, creating custom roles for SAML Role mapping, etc.



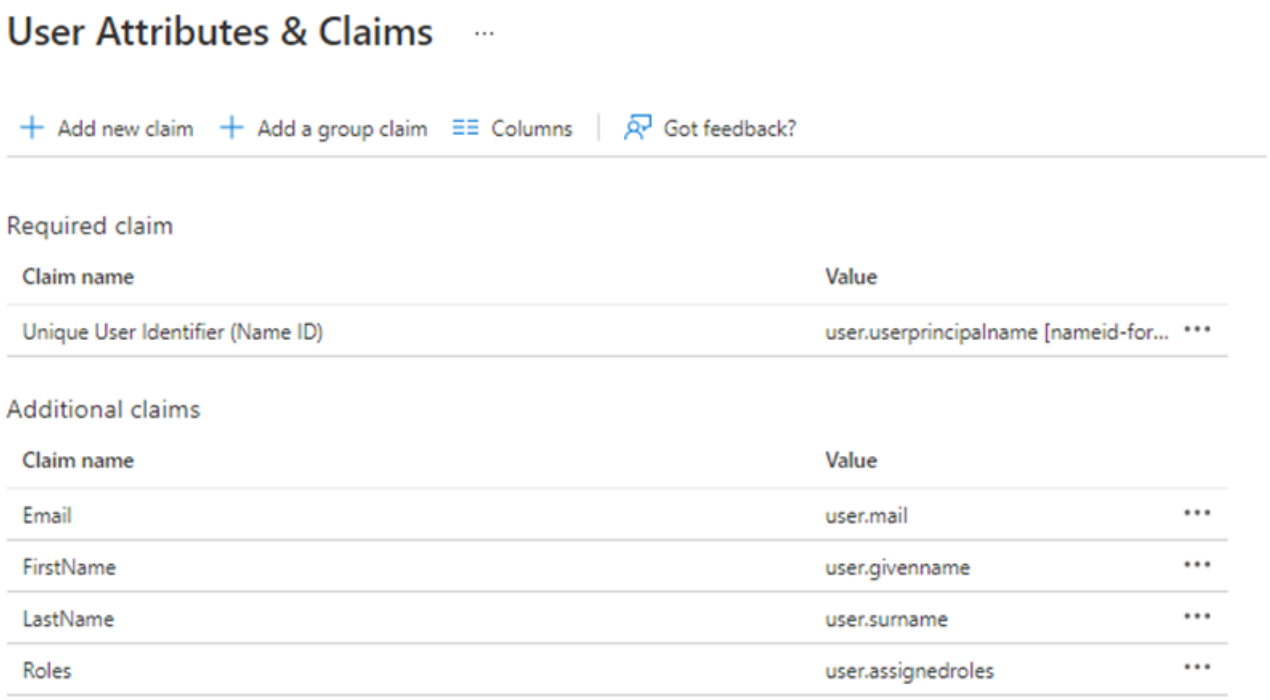
7. Click **Single sign-on**, and then **SAML**.



8. Create a unique Identifier (Entity ID) in Microsoft Entra ID.

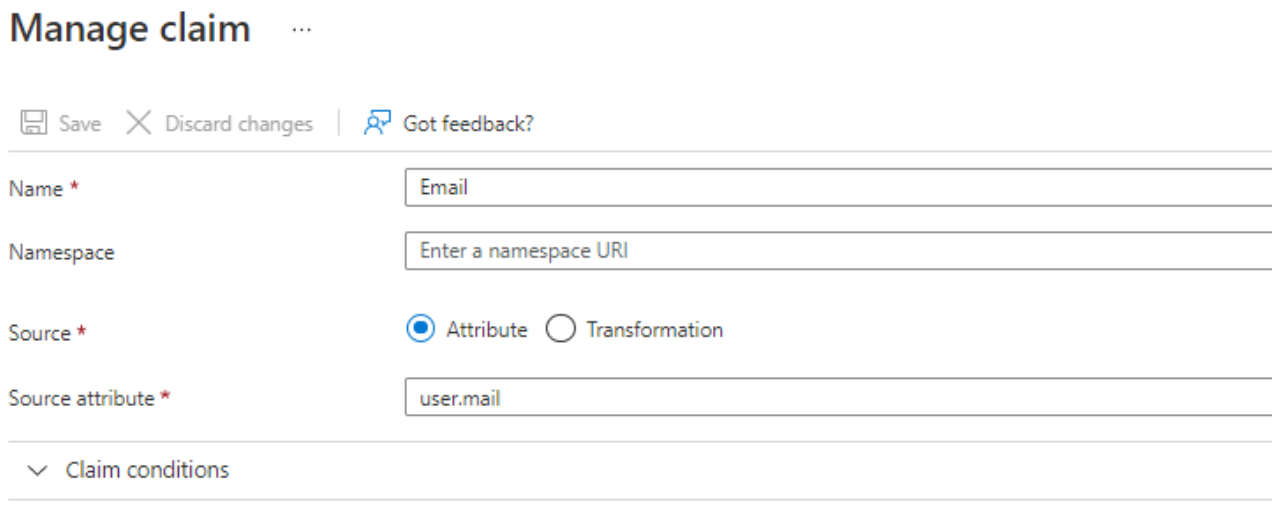


9. Modify user attributes in Microsoft Entra ID as shown in the following image:

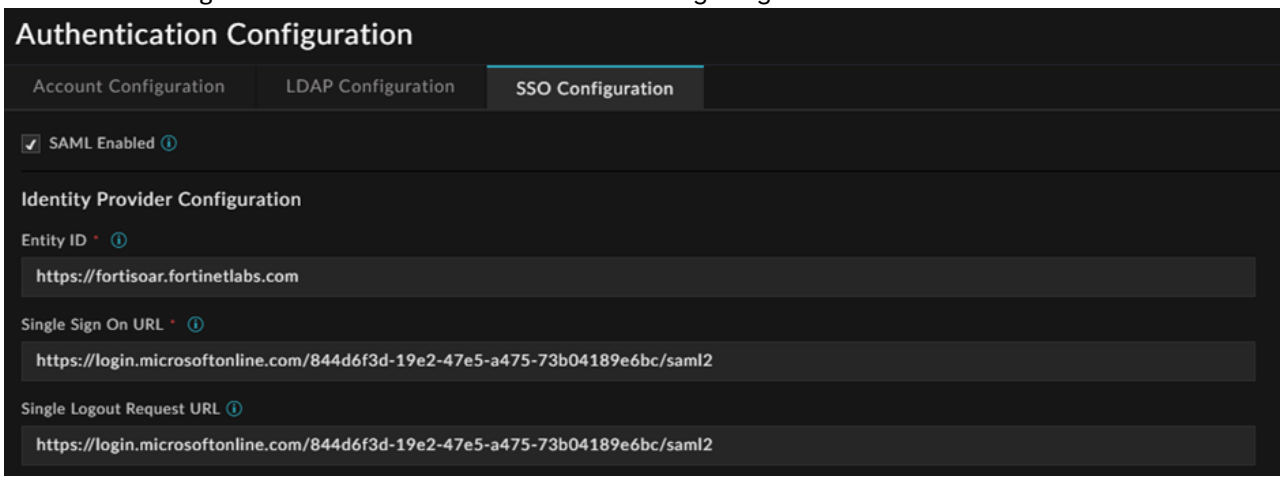


Important: For the Microsoft Entra ID attributes and claims, it's recommended that you delete the `namespace` section for each attribute, else it generates a URL. The following image is an example of the

Email attribute whose namespace is blank:

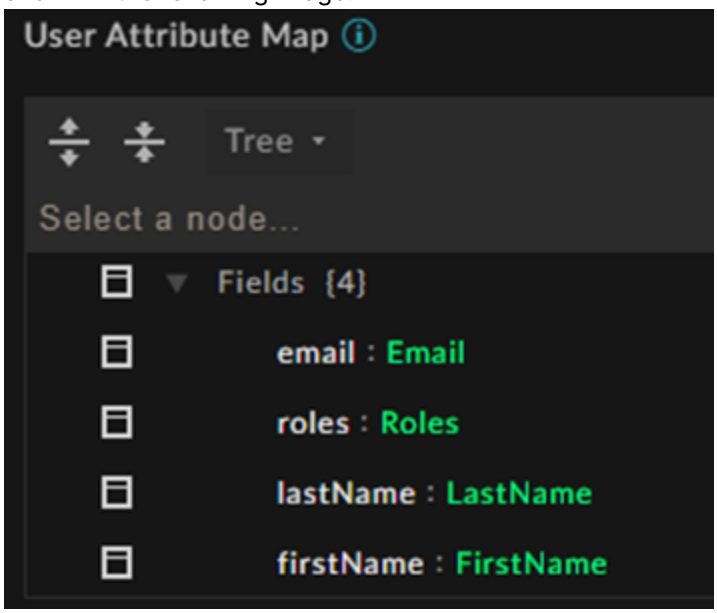


10. In FortiSOAR, navigate to **Settings** > **Authentication** > **SSO**, and then enter the IdP details in the `Identity Provider Configuration` section as shown in the following image:

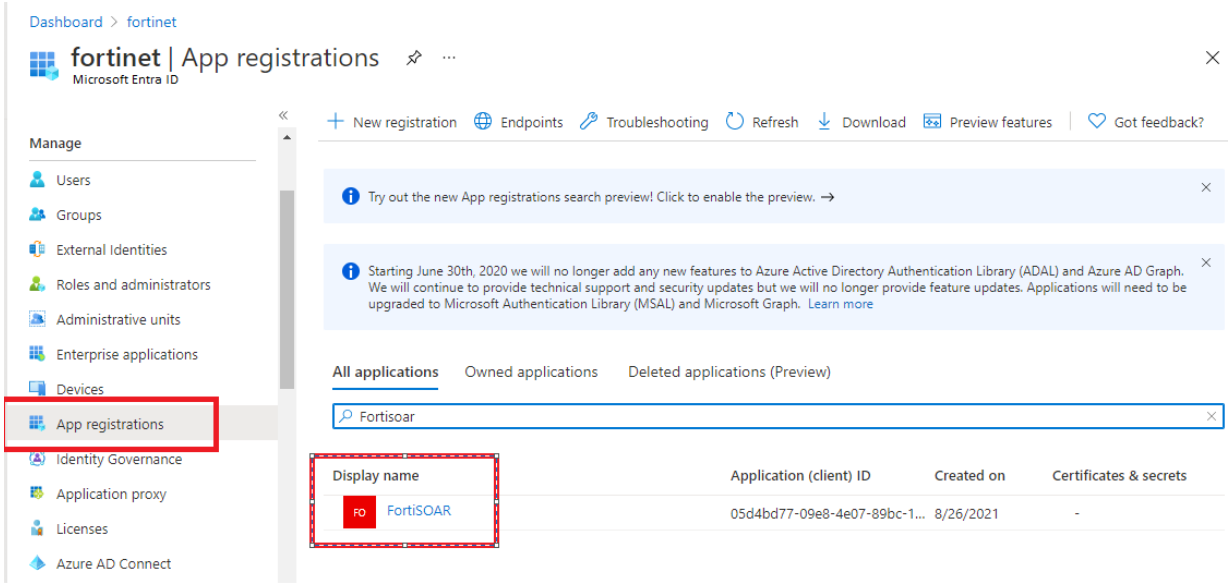


Enter the details such as the Entity ID, Single Sign On URL, Single Logout Request URL, etc as set up in Microsoft Entra ID.
In the **X509 Certificate**field, paste the text that you copied from the downloaded Microsoft Entra ID certificate.

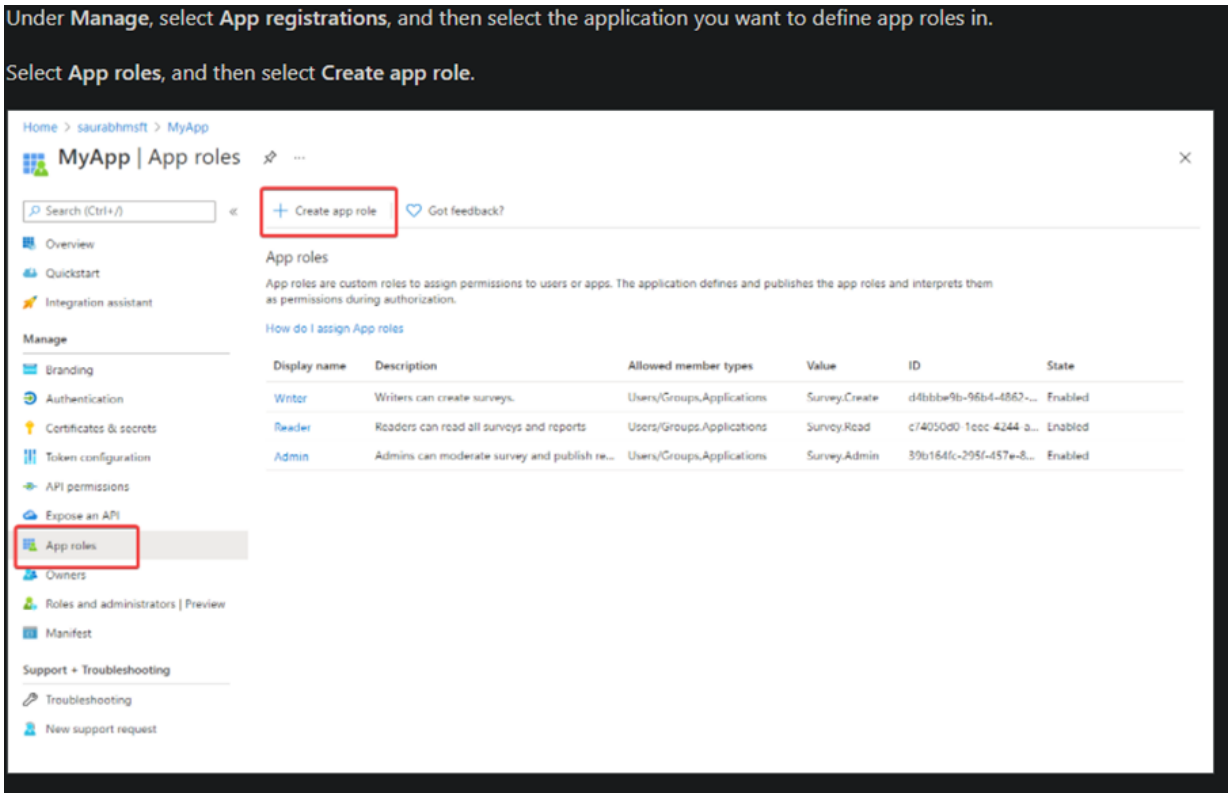
11. Add the user attribute mapping for Microsoft Entra ID in FortiSOAR by updating the **User Attribute Map** as shown in the following image:



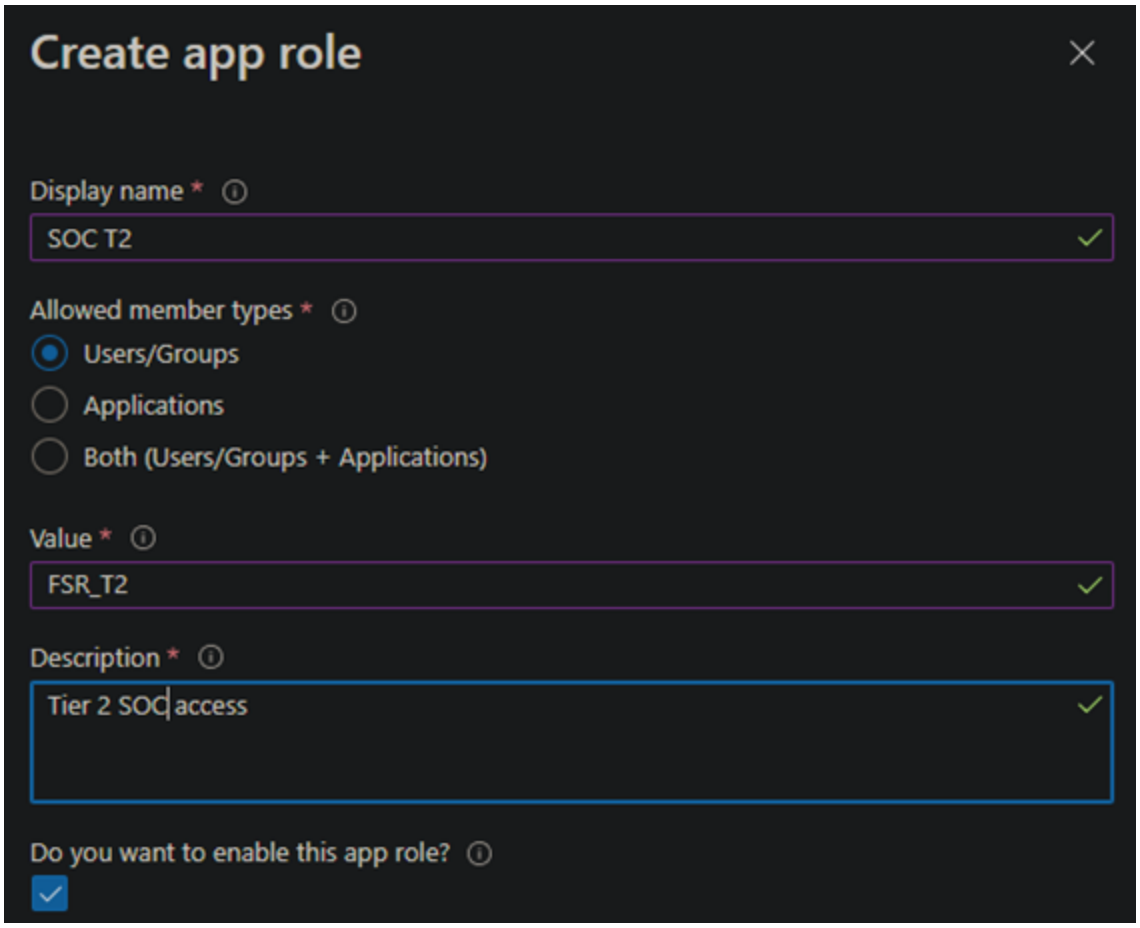
12. In the `Team and Role Mapping` section map the teams and roles of SSO users to the teams and roles created in Microsoft Entra ID IdP:
    a. To create roles in Microsoft Entra ID, open the Microsoft Entra ID portal.
    b. From the left menu, click **Microsoft Entra ID** and then click **App registrations**. In the `All applications` section, search for the name of your FortiSOAR application:



    c. Click App Roles and then click Create app role. for example, create the `FSR_T2 Analyst` in Microsoft Entra ID, as shown in the following image:

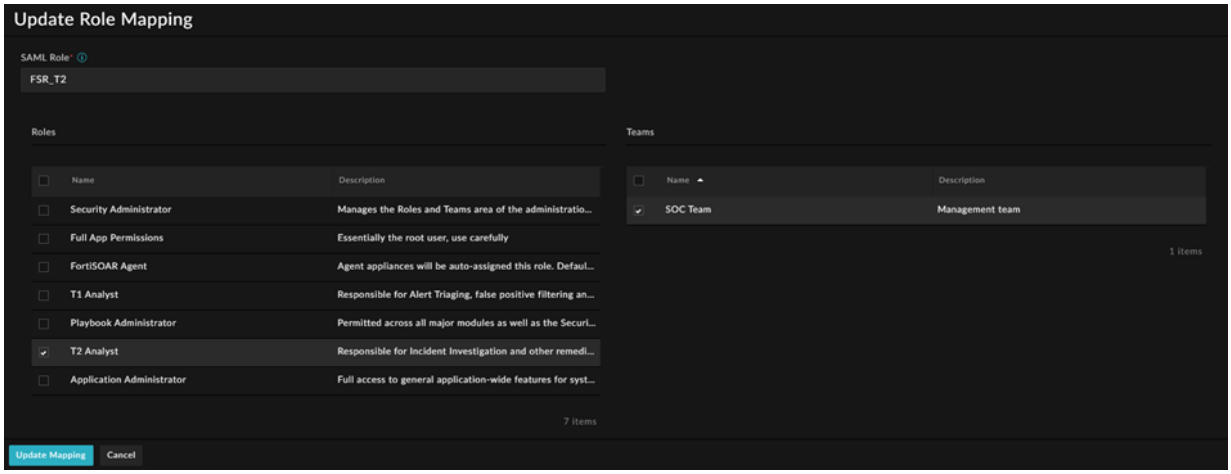


Then enter the values in the `Create app role` dialog, and click **Apply**.

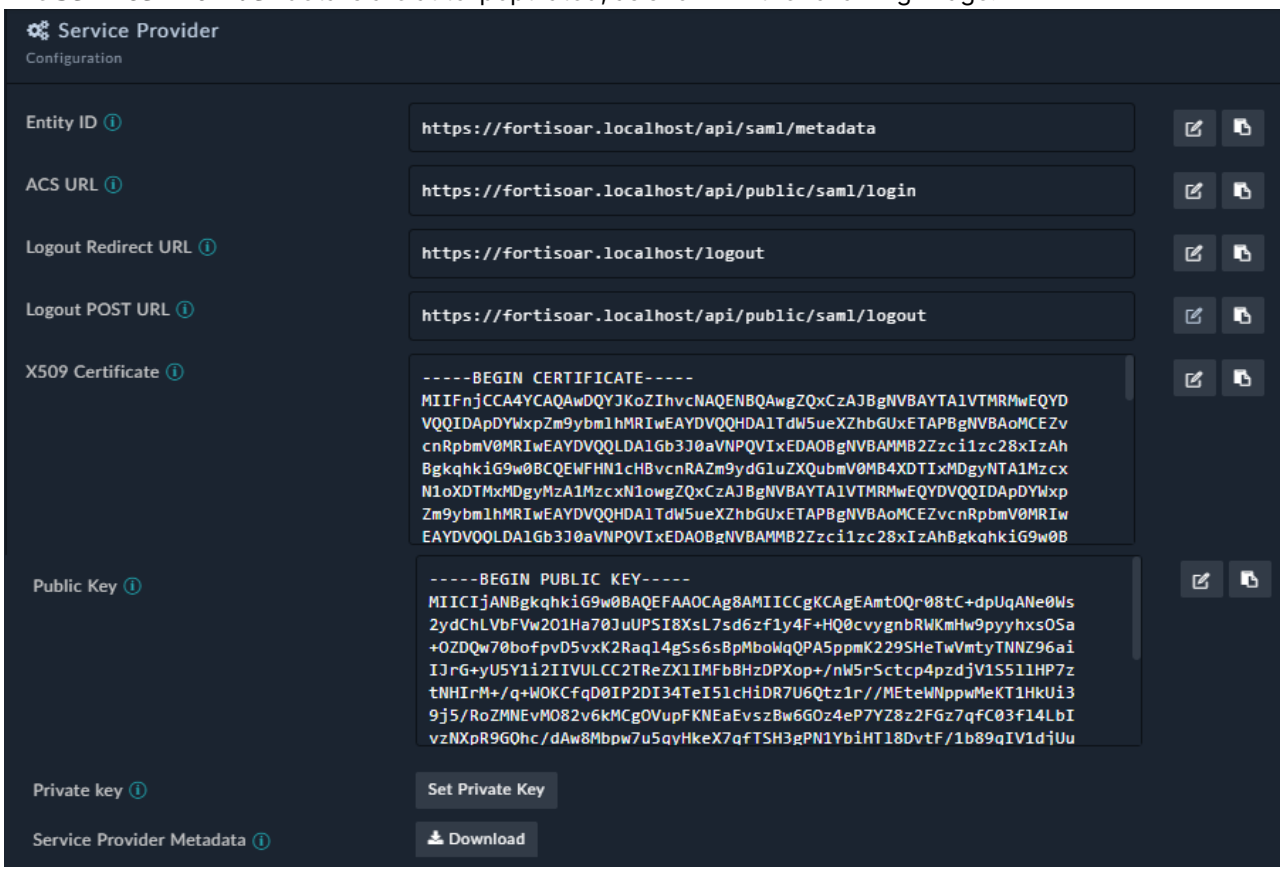


This creates the `FSR_T2 Analyst` role in Microsoft Entra ID. You need to perform this step for each role that you want to map in FortiSOAR.

d. Log on to FortiSOAR and on the `SSO` page, and map the roles that you have created in Microsoft Entra ID to FortiSOAR roles. For example, `FSR_T2 Analyst` role can be mapped to the T2 Analyst role in FortiSOAR:



13. The `Service Provider` details are auto-populated, as shown in the following image:



Note: These settings can be kept as is and only need to be updated for DNS name change if you want to keep a different DNS name for FortiSOAR than the one set in FortiSOAR (as hostname).

14. Click **Save** in FortiSOAR to save the changes to the IdP configuration.

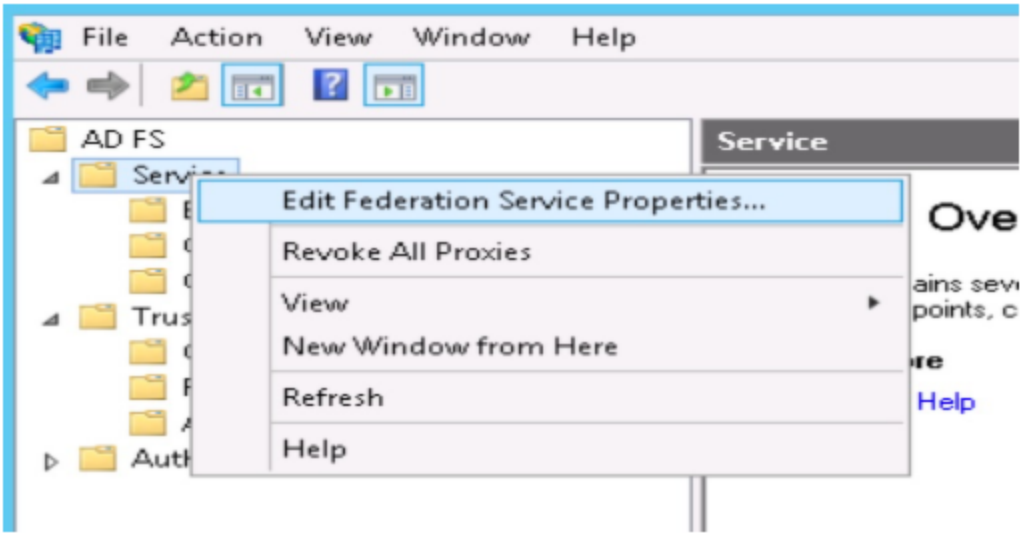
Configuring SAML in ADFS

If you change the hostname for your FortiSOAR system, you will require to delete the old ADFS configuration and re-configure ADFS.

General ADFS Setup

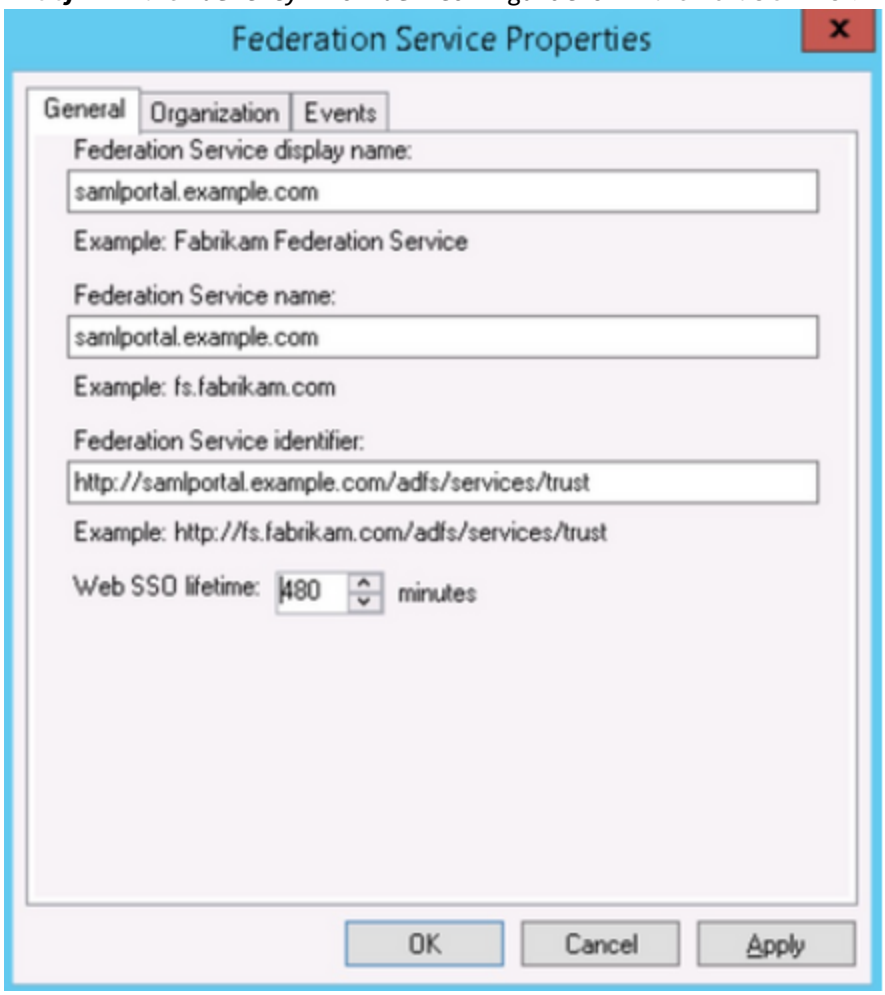
This procedure uses `ADFS 3.0` and uses `samlportal.example.com` as the ADFS website. The values you use in your setup will be based on your ADFS website address. See ADFS integration with SAML 2.0 for more information.

1. Log on to the ADFS server and open the management console.
2. Right-click **Service** and click **Edit Federation Service Properties**.



3. On the `Federation Service Properties` dialog, in the `General Settings` tab, confirm that the DNS entries and certificate names are correct. Note the `Federation Service Identifier`, since you will use as the

Entity ID in the `Identity Provider Configuration` in the FortiSOAR UI.

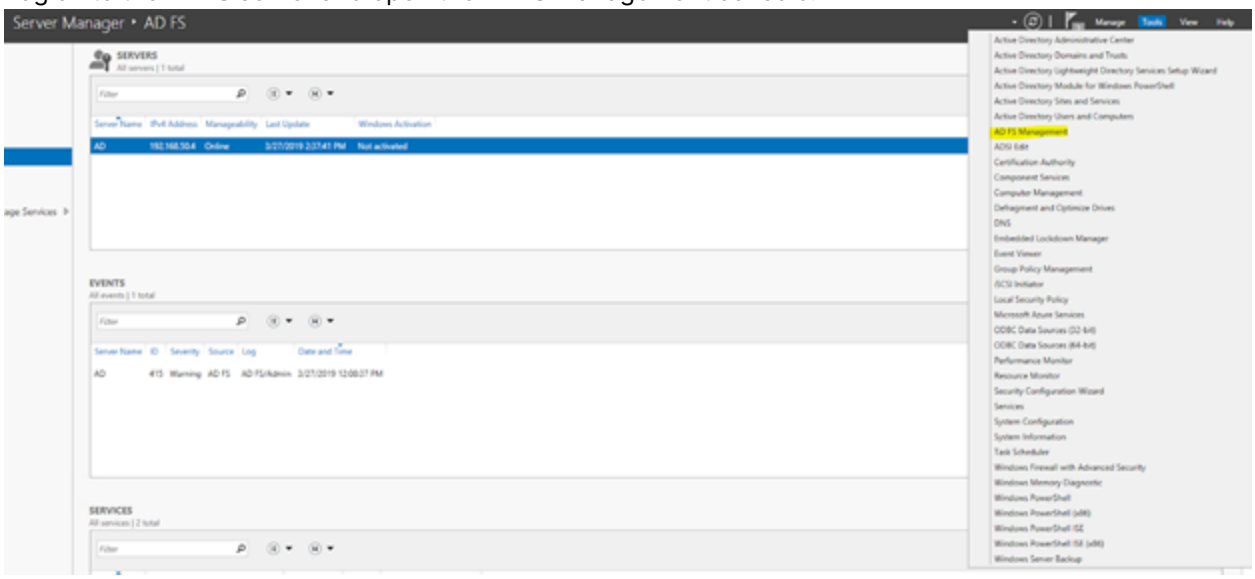


4. In the `Services` panel, browse to `Certificates` and export the Token-Signing certificate using the following steps.
   a. Right-click the certificate and select **View Certificate**.
   b. Select the **Details** tab and click **Copy to File**, which opens the `Certificate Export` wizard.
   c. On the Certificate Export Wizard, click **Next**.
   d. Select **Base-64 encoded binary X.509 (.cer)**, and then click **Next**.
   e. Select where you want to save the Token-Signing certificate and provide a name to the certificate, and then click **Next**.
   f. Click **Finish**.
   g. Copy the contents of the Token-Signing certificate and paste the contents in the **X509 Certificate** area in the `Identity Provider Configuration` in the FortiSOAR UI.

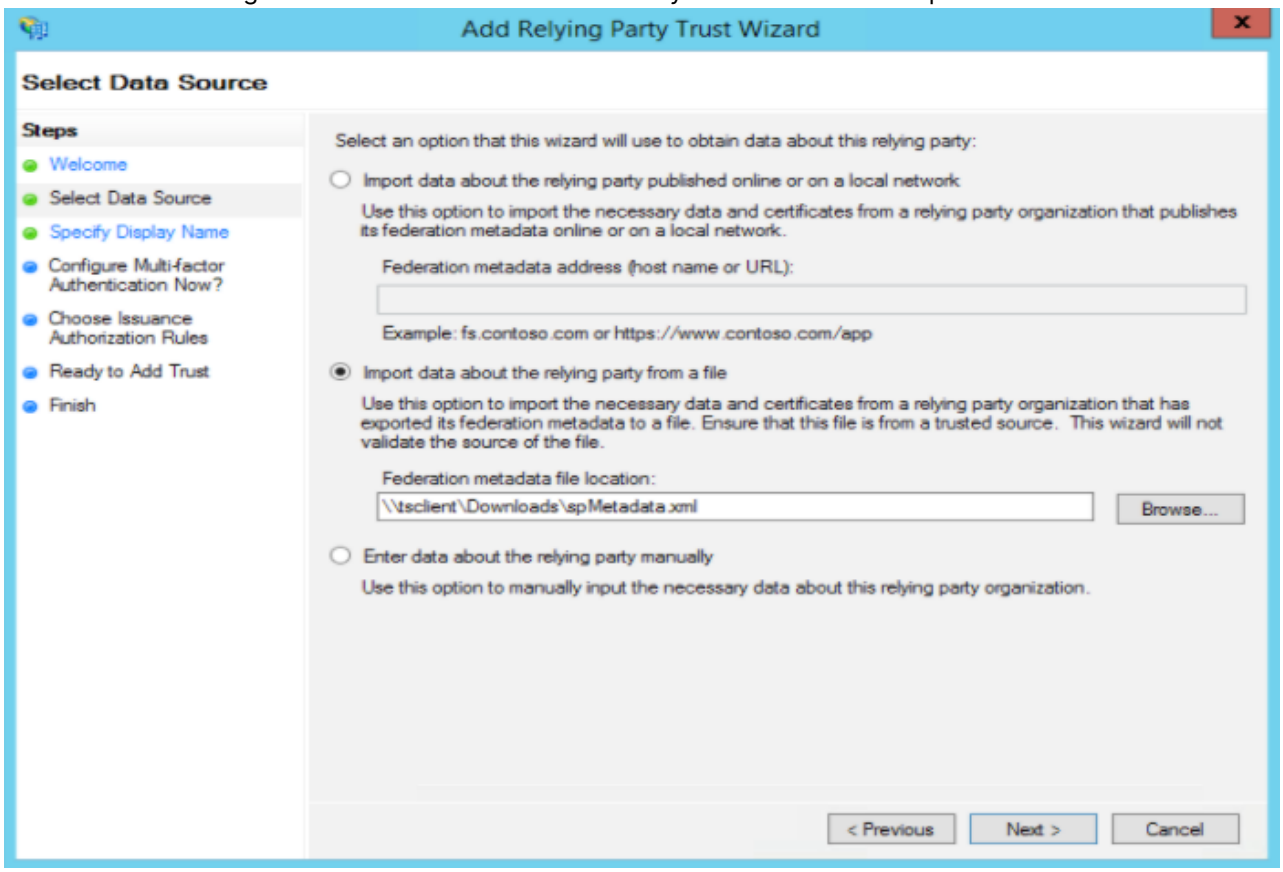
Configuring ADFS Relying Party Trust

1. Log on to FortiSOAR as an administrator.
2. Click **Settings** > **Authentication** > **SSO** and download the SAML metadata file by clicking **Download** in the `Service Provider Configuration` section.

3. Log on to the ADFS server and open the ADFS management console.



4. Expand **Trust Relationships** and right-click **Relying Party Trust** and select **Add**.
5. On the `Add Relying Party Trust Wizard`click **Start**.
6. In the `Select Data Source` panel, select the **Import data about the relying party from a file** option and click **Browse** to navigate to the SAML metadata file that you have saved in Step 2 and then click **Next**.



7. In the `Specify Display Name` panel set the display name and then click **Next**.
8. (Optional) In the `Configure Multi-factor Authentication Now?`panel configure MFA and then click **Next**.

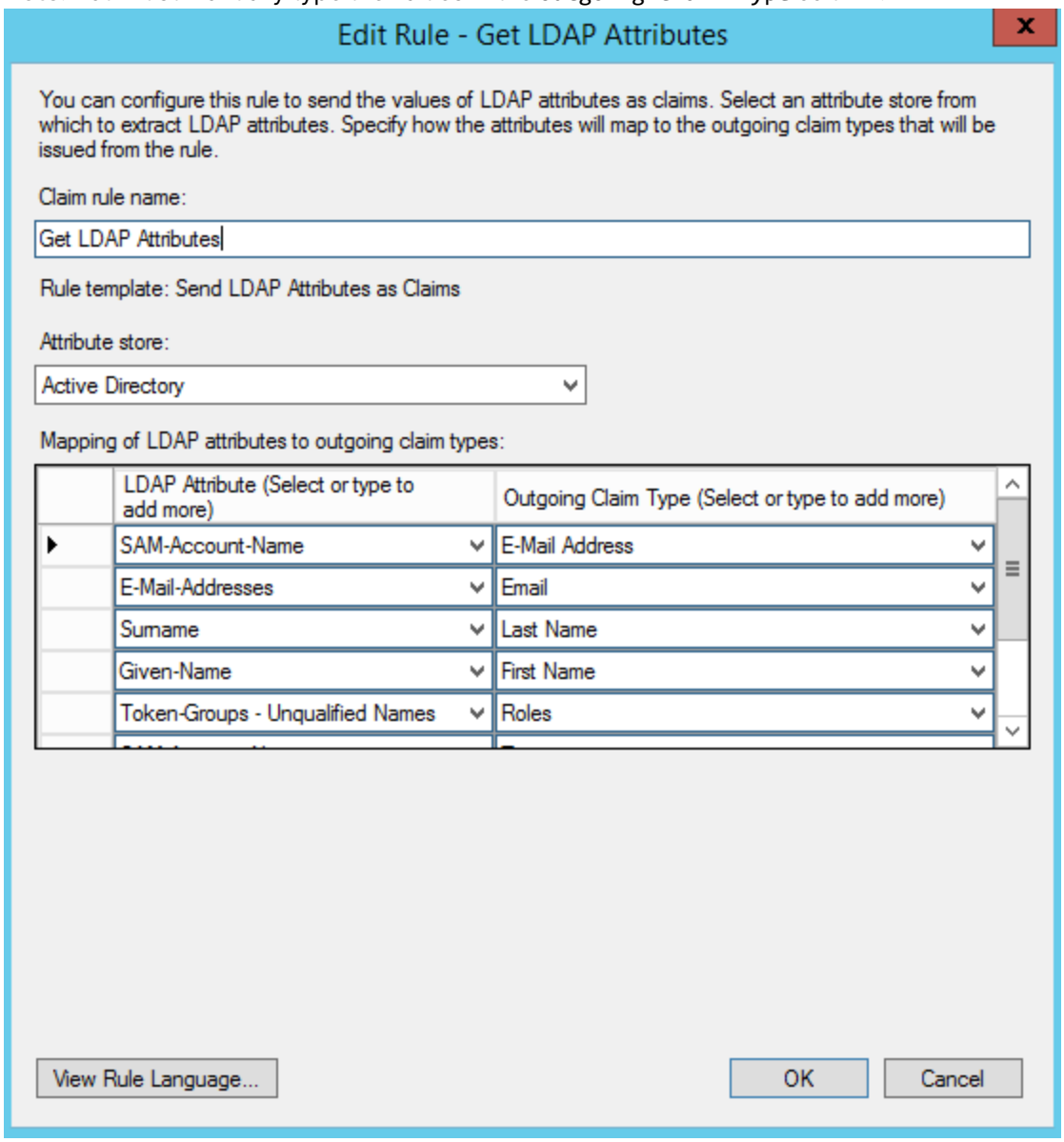
9. In the Choose Issuance Authorization Rules panel, select the **Permit all users to access this relying party** option and then click **Next**.
10. In the Ready to Add Trust panel, click **Next**.
11. In the Finish panel, ensure that the **Open the Edit Claim Rules dialog** statement is selected and then click **Close**. This opens the Edit Claim Rules Wizard in which you can immediately add and configure rules as mentioned in the next section, or if you have closed Edit Claims Rules then use the steps mentioned in the next section to open Edit Claim Rules and add and configure rules.

Configuring ADFS Relying Party Claim Rules

You must edit the claim rules to enable communication with FortiSOAR SAML

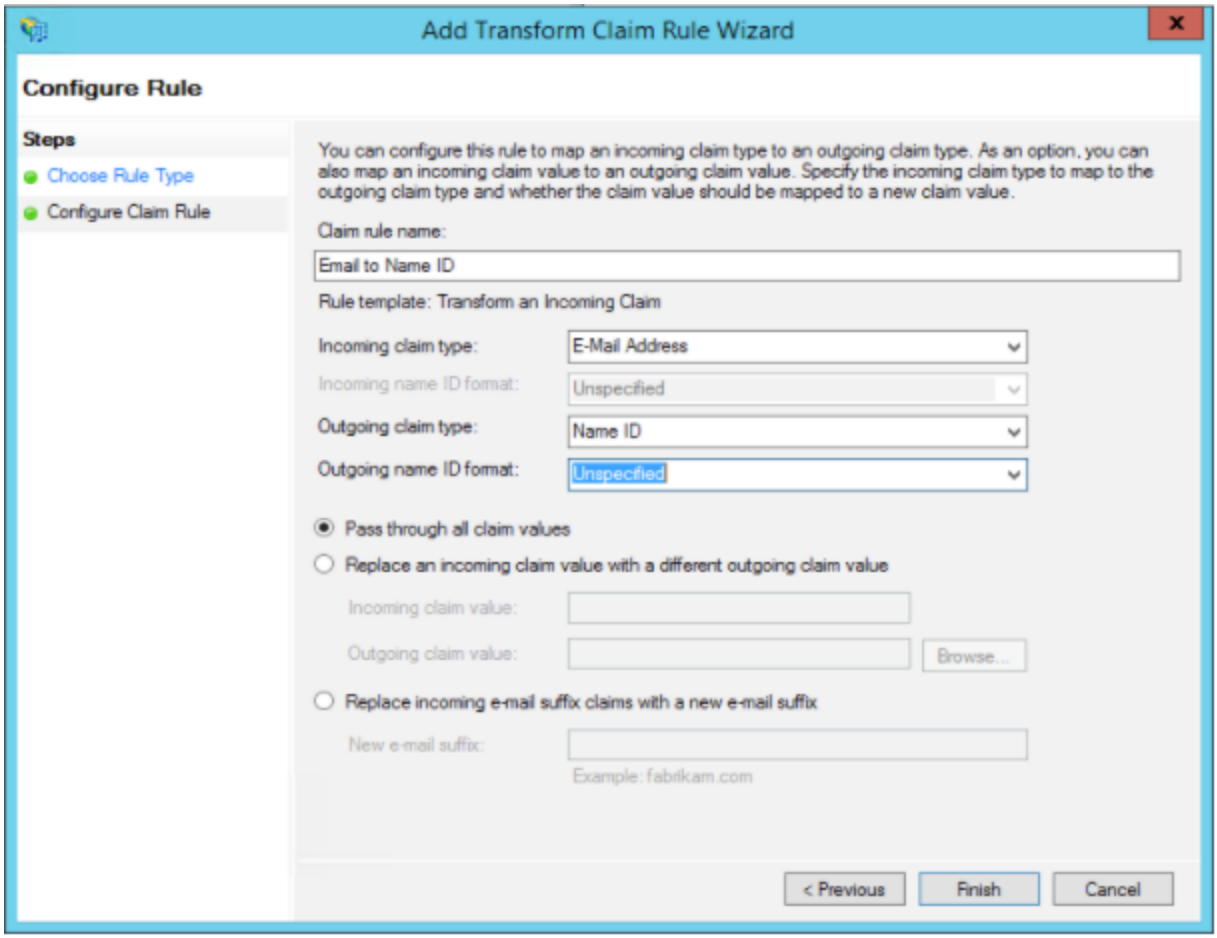
1. Log on to the ADFS server and open the management console.
2. Right-click the relying party trust (as configured in the previous section) and select **Edit Claim Rules**.
3. Click the **Issuance Transform Rules** tab and select **Add Rules**.
4. Select **Send LDAP Attribute as Claims** as the claim rule template to use and then click **Next**.
5. On the Configure Claim Rule dialog, in **Claim rule name**, enter a name to the claim rule. For example, name the claim rule as Get LDAP Attributes.
6. From the Attribute store drop-down list, select Active Directory.
7. In the Mapping of LDAP attributes to outgoing claim types section, map the following values:
   a. Select **SAM-Account-Name** from the LDAP Attribute column and map that to **E-Mail Address** in the Outgoing Claim Type column.
   b. Select **E-Mail-Addresses** from the LDAP Attribute column and map that to **Email** in the Outgoing Claim Type column.
   Note: You must manually type the values in the Outgoing Claim Type column.
   c. Select **Surname** from the LDAP Attribute column and map that to **Last Name** in the Outgoing Claim Type column.
   Note: You must manually type the values in the Outgoing Claim Type column.
   d. Select **Given-Name** from the LDAP Attribute column and map that to **First Name** in the Outgoing Claim Type column.
   Note: You must manually type the values in the Outgoing Claim Type column and the values that you specify in the Outgoing Claim Type column must match the what you enter in the right-side field in the **User Attribute Map** in the Identity Provider Configuration in the FortiSOAR UI.
   e. Select **Token-Groups - Unqualified Names** from the LDAP Attribute column and map that to **Roles** in the Outgoing Claim Type column.

Note: You must manually type the values in the `Outgoing Claim Type` column.



8. Click **Finish** and select **Add Rules**.
9. Select **Transform an Incoming Claim** as the claim rule template to use and then click **Next**.
10. On the `Add Transform Claim Rule Wizard`, in **Claim rule name**, enter a name to the claim rule. For example, name the claim rule as `Email to Name ID`.
11. From the **Incoming claim type** drop-down list, select **E-Mail Address**, from the `Outgoing claim type` drop-down list, select **Name ID** and select the **Pass through all claim values** option and click **Finish** and

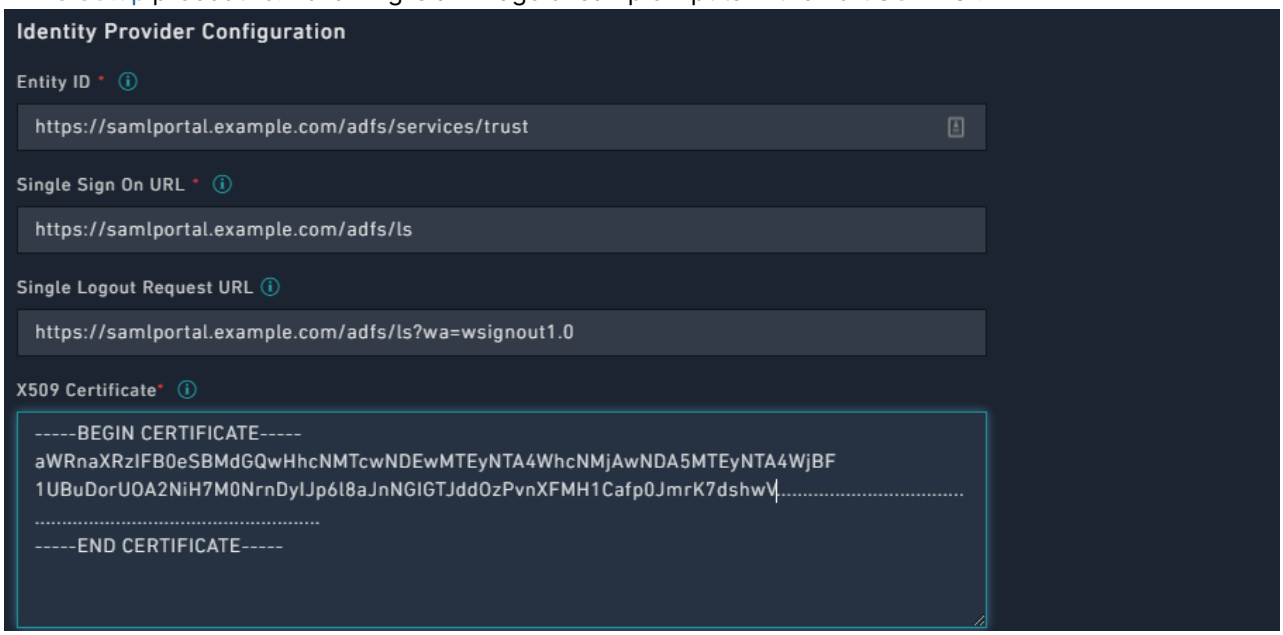
then click **OK**.



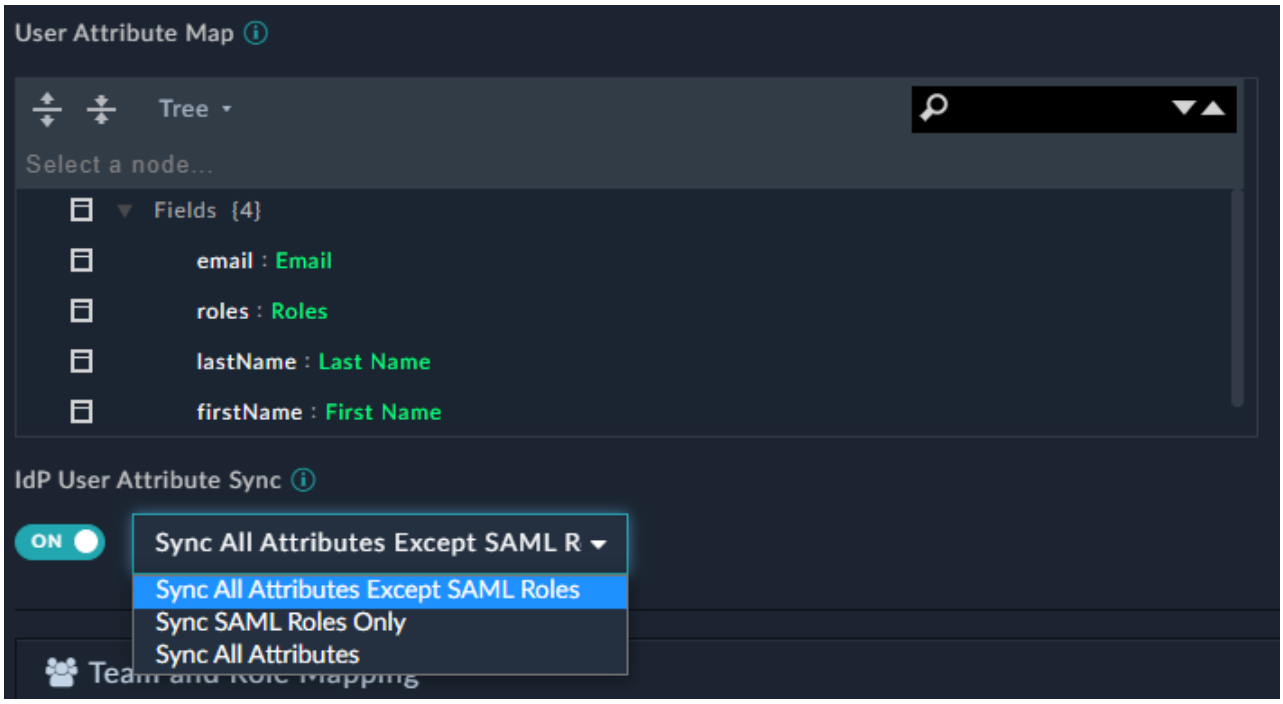
Configuring FortiSOAR for ADFS

1. Log on to FortiSOAR as an administrator.
2. Click **Settings** > **Authentication** > **SSO**.
3. To enable SAML for FortiSOAR, click the **SAML Enabled** check box.
4. In the `Identity Provider Configuration` section, enter the IdP details.
   Enter the **Entity ID** as the one that you had noted in Step 3 of the General ADFS Setup procedure. For example, `https://samlportal.example.com/adfs/services/trust`
   Enter the **Single Sign On URL** as `<server_address>/adfs/ls`. For example, `https://samlportal.example.com/adfs/ls`
   Enter the **Single Logout Request URL** as `<server_address>/adfs/ls?wa=wsignout1.0`. For example, `https://samlportal.example.com/adfs/ls?wa=wsignout1.0`
   In the **X509 Certificate** area, paste the contents of the certificate you exported in Step 8 of the General

ADFS Setup procedure. Following is an image of sample inputs in the FortiSOAR UI:



5. Map the user attributes received from the ADFS (IdP) with the corresponding attributes of FortiSOAR.
   Use the **User Attribute Map** to map the attributes received from the ADFS with the corresponding attributes required by FortiSOAR. FortiSOAR requires the firstname, lastname and email attributes to be mapped. The ADFS attributes that you need to map are the names that you specify as values in the `Outgoing Claim Type` column in the management console of ADFS. For more information, see Configuring ADFS Relying Party Claim Rules.
   In the `User Attribute Map`, under `Fields`, click the editable field name (right side field name), to map it to the attribute that will be received from the IdP. The non-editable field name (left-side field name) is the FortiSOAR attribute. For example, in the following image, you map the FortiSOAR attribute `firstName` to the IdP attribute `First Name`.



If you want to set any of the optional configurations, see Configuring SAML in FortiSOAR.

6. Click **Save** to complete the SAML configuration in FortiSOAR.

Support for mapping roles and teams of SSO users in FortiSOAR

You can map the role and team of SSO users in FortiSOAR based on their roles defined in the IdP. Thereby you can set the role of an SSO user in FortiSOAR based on the role you have defined in your IdP.

To achieve this FortiSOAR has added a new configuration in the `SSO` page where you can map the role that you have specified in the IdP to a FortiSOAR role and team. The relationship between the IdP role and the FortiSOAR role is one to many, i.e., one IdP role can map to multiple FortiSOAR roles.

SAML supports attribute-based authorization. Therefore, you should configure attribute `roles` in your IdP that will contain roles of your SSO users on the IdP.

If you have not set up mapped roles of SSO users in FortiSOAR, or if FortiSOAR receives a response from the IdP that does not contain any roles, or receives a response that does not map to any of the FortiSOAR roles, then the SSO user will be assigned the default roles.

Configuring IdPs to send the SSO user role information to FortiSOAR

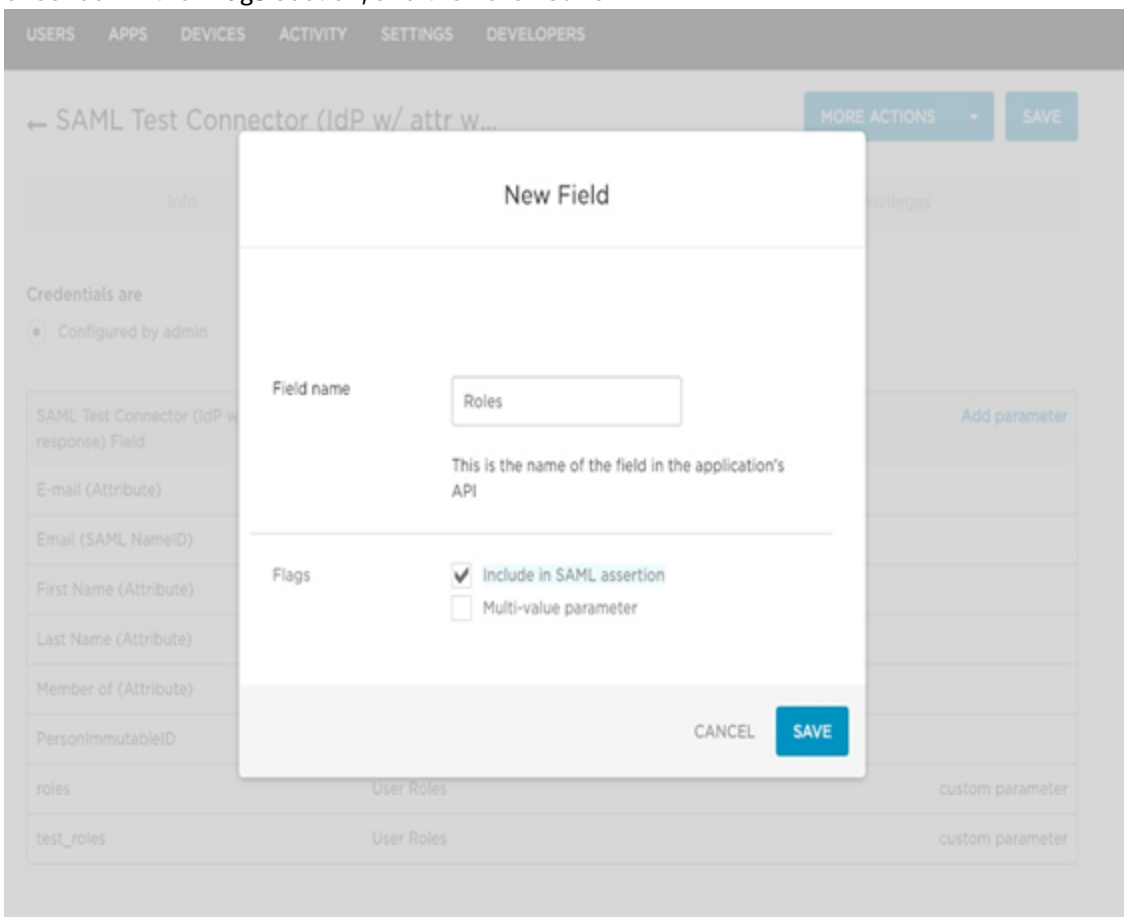
The following sections define how you can configure IdPs, i.e., **OneLogin**, **Okta**, or **Auth0** to send the SSO user role information to FortiSOAR when the user is logging on to FortiSOAR (SSO login).

For mapping of roles in ADFS, see the Configuring ADFS Relying Party Claim Rules section.

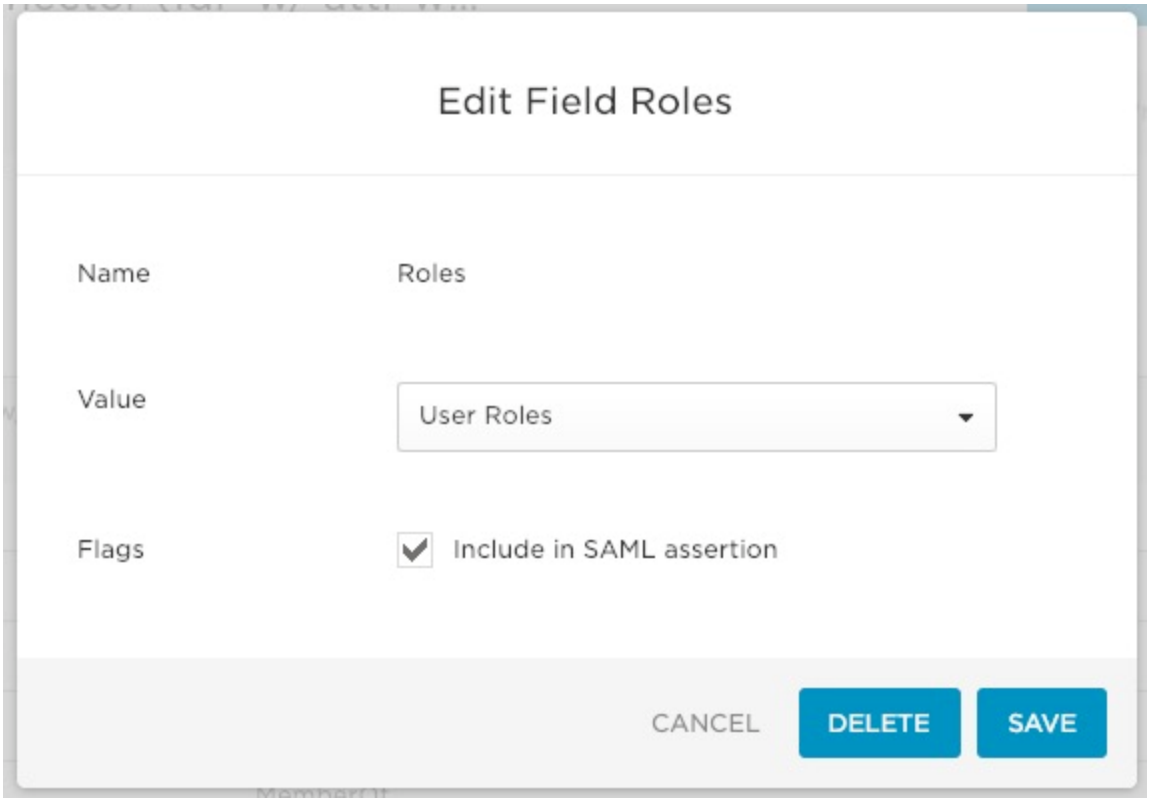
For any other IdP, configure roles as per the IdP requirements and contact the IdP support personnel if you face any issues.

OneLogin

1. Log on to OneLogin as an administrator.
2. Navigate to the SAML app that you have created by clicking **APPS** in the administration panel. Open the SAML app and in the `App Configuration` screen, go to the `Parameters` section and click **Add Field**, which displays the New Field dialog.
3. In the `New Field` dialog, in the Field name type `Roles`, ensure that you check **Include in SAML assertion** checkbox in the `Flags` section, and then click **Save**.

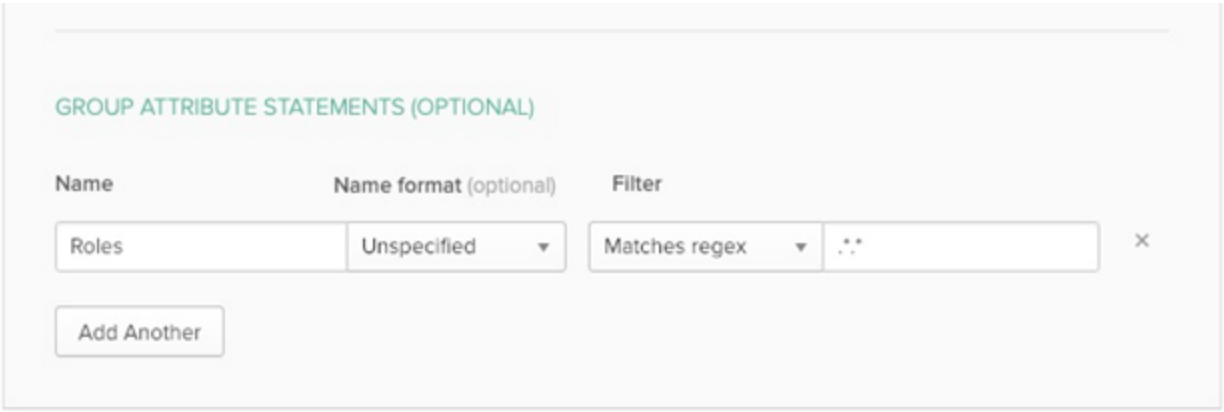


4. In the next dialog, i.e., the `Edit Field Roles` dialog, from the **Value** drop-down list, select **User Roles** and click **Save**.



Okta

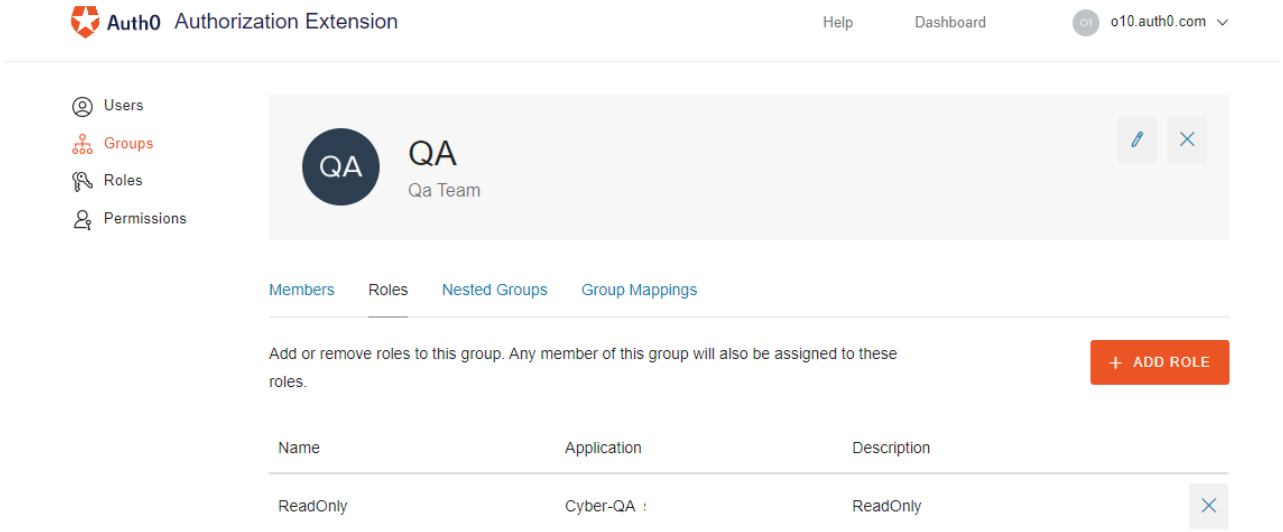
1. Log on to Okta as an administrator.
2. Navigate to the SAML app that you have created and edit the SAML settings.
3. In the GROUP ATTRIBUTE STATEMENTS (OPTIONAL) section set the following:
   Name: Set as **Roles**.
   Filter: Set as **Matches regex** `*.*`



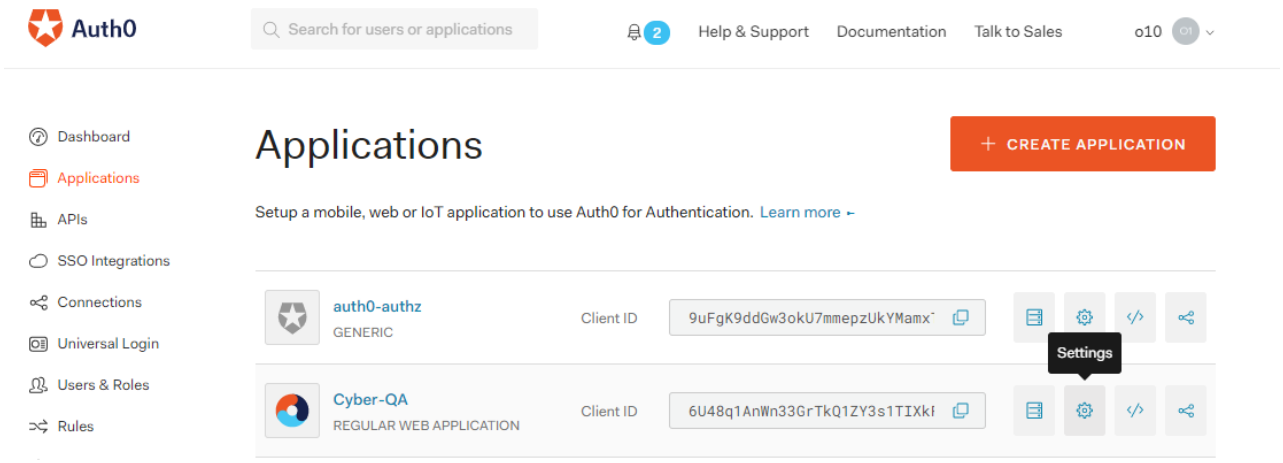
4. Click **Next** and complete the setup.

Auth0

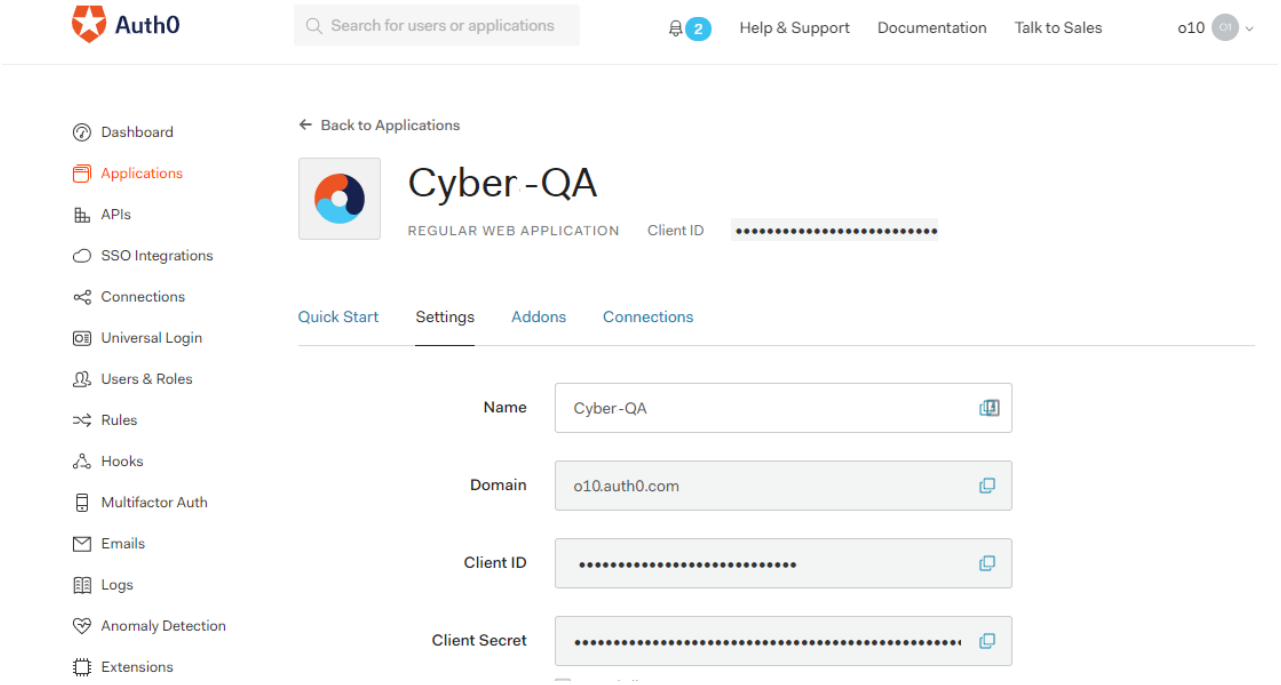
1. Log on to Auth0 as an administrator and in the left menu click **Authorization**.
2. On the `Authorization Extension` page, create a new group and associate required members (users) and roles with this group.



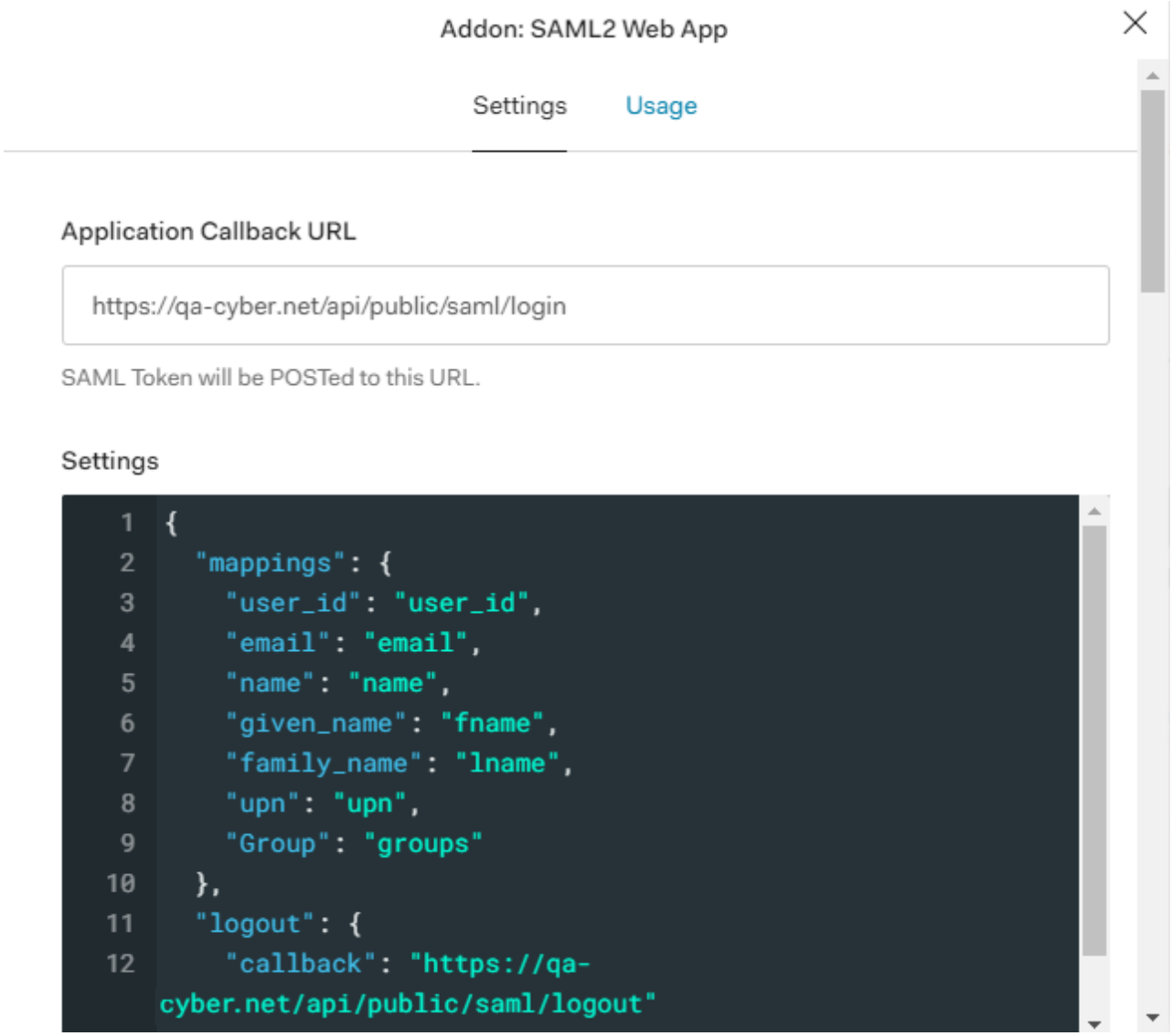
3. Navigate back to the main menu (`Dashboards` page) and click **Applications**.
4. Create a new application, or click on the **Settings** icon of the application whose settings you want to edit:



This opens the `Setting` page for the application:



5. Click the **Addons** tab and click **SAML2** and enter the required details on the **Settings** tab for the application you have created:



6. Click **Save** to save the settings of the application.

Troubleshooting SAML issues

Unable to login to FortiSOAR when ADFS SAML is configured

If you are unable to login to FortiSOAR when ADFS SAML is configured and the default certificates are failing, and if you find the `"The revocation function was unable to check revocation for the certificate."` error in the ADFS logs, then you must turn off the certificate revocation check using the following steps:

1. Enter `Powershell` in the "Administrator" mode of the ADFS system.
2. Run the following commands: (RelyingPartyTrustName should be in double quotes):
   ```
   Set-AdfsRelyingPartyTrust -TargetName "<RelyingPartyTrustName>" -
   SigningCertificateRevocationCheck None
   Set-AdfsRelyingPartyTrust -TargetName "<RelyingPartyTrustName>" -
   ```

`EncryptionCertificateRevocationCheck None`
This turns off the certificate revocation check and now you should be able to login to FortiSOAR.

SAML users face issues while trying to login to FortiSOAR when the certificate gets expired or replaced on ADFS IDP

When the certificate gets expired or replaced on ADFS IDP, then the SAML users get the following errors which trying to log into FortiSOAR:

```
Fri Oct 22 06:46:26 AST 2021
                        There was an unexpected error (type=Internal Server Error, status=500).
                        Processing samlservice sso response failed with error: Signature validation failed. SAML
Response rejected

                        [root@lincon ~]# tail /var/log/cyops/cyops-gateway/saml.log
                        22-10-2021 05:09:38.351 [http-nio-8080-exec-110] ERROR
c.onelogin.saml2.authn.SamlResponse.isValid - Signature validation failed. SAML Response rejected
                        22-10-2021 05:16:34.567 [http-nio-8080-exec-114] ERROR
c.onelogin.saml2.authn.SamlResponse.isValid - Signature validation failed. SAML Response rejected
```

Resolution

To resolve this issue, get the newly deployed certificate and log in to FortiSOAR. Navigate to **Settings** > **Authentication** > **SSO**. In the `Identity Provider Configuration` section, replace the contents of the **x509 Certificate** field with the contents of the new certificate.

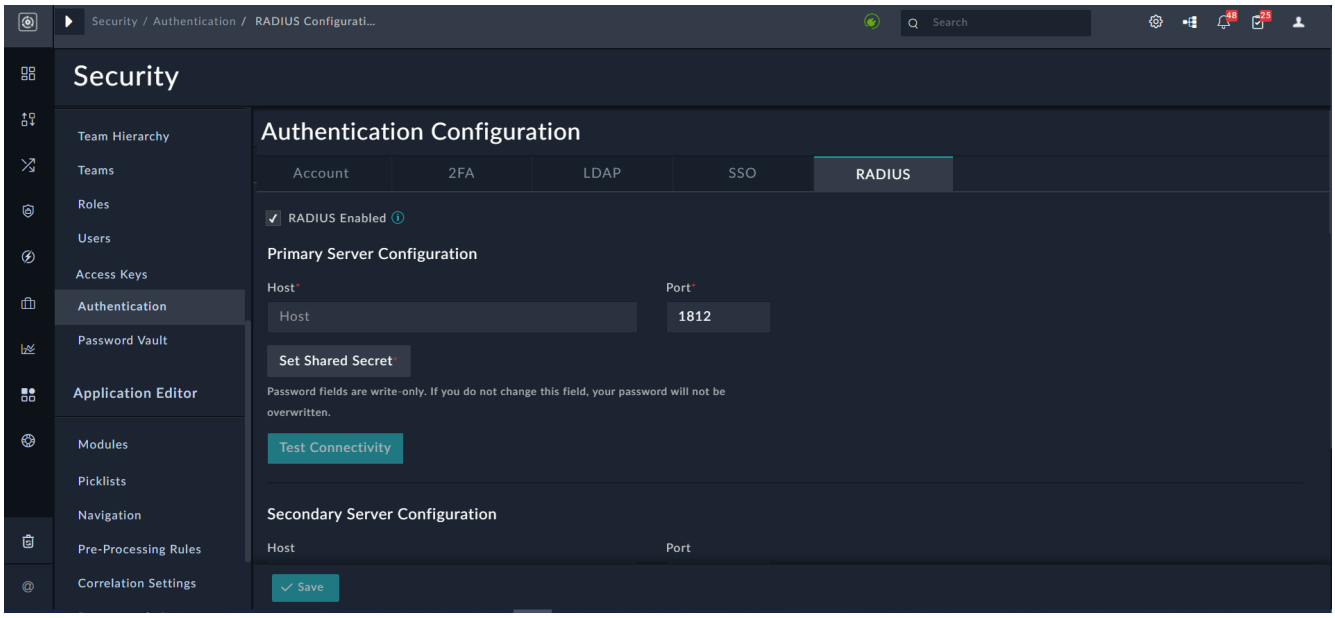
Configuring FortiSOAR authentication with a RADIUS server

From release 7.2.0 onwards, FortiSOAR enables you to authenticate users using a RADIUS server, i.e., users can enter their RADIUS credentials to log into FortiSOAR.

Use the `Authentication` menu to setup, modify, and turn on or off authentication with a RADIUS server.

To create or update a RADIUS configuration, administrators must have '`Security Update`' permissions.

Click **Settings** > **Authentication** to open the `Account` page. Click the **RADIUS** tab and click the **RADIUS Enabled** checkbox, if you want to authenticate users using a RADIUS server.



Once you click the **RADIUS Enabled** checkbox, configure your primary and optionally a secondary RADIUS server as follows:

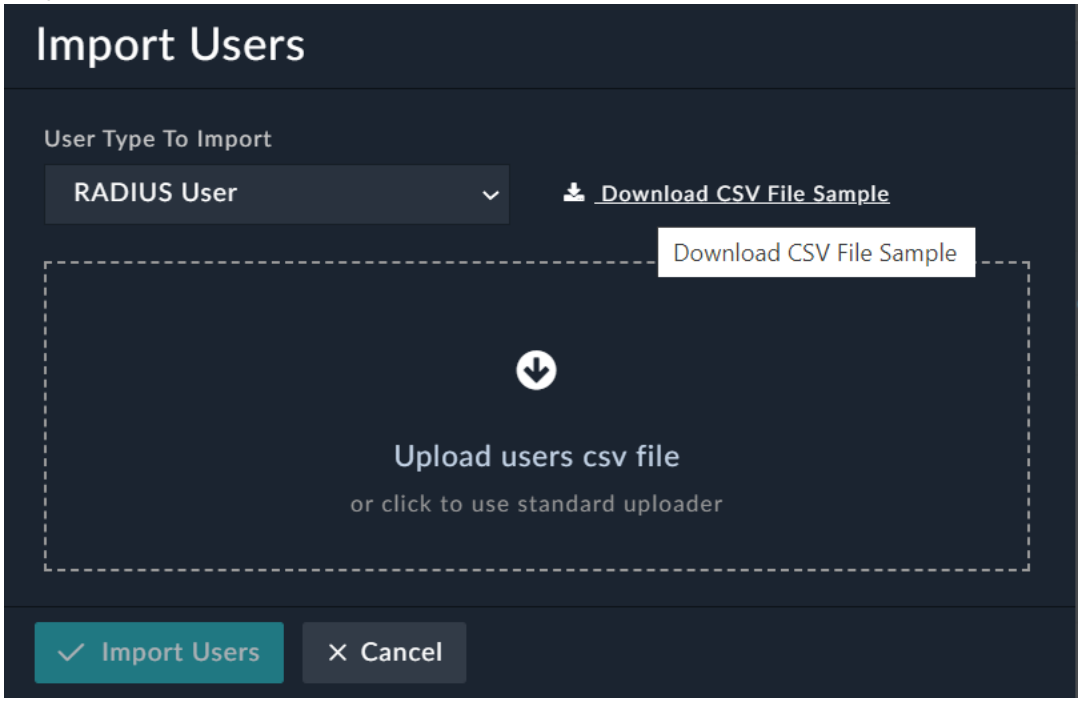
1. In the `Primary Server Configuration` section, enter the following details for your primary RADIUS server:
   a. In the **Host** field, enter the IP address of the primary RADIUS server that you will use to authenticate users.
   b. In the **Port** field, enter the port number where the primary RADIUS server listens for authentication requests. Defaults to 1812.
   c. In the **Shared Secret** field, enter the secret code that is known to only the client (FortiSOAR in this case) and the primary server.
   To check the RADIUS configuration of the primary server, click the **Test Connectivity** button. Clicking **Test Connectivity** opens the **Enter Test Credentials** dialog in which you can enter the username and password used to connect to the primary server. If the connection succeeds, then a success message gets displayed, and if the connection fails, then an appropriate error message gets displayed.
2. In the `Secondary Server Configuration` section, enter the following details for your secondary RADIUS server:
   a. In the **Host** field, enter the IP address of the secondary RADIUS server that you will use to authenticate users.
   b. In the **Port** field, enter the port number where the secondary RADIUS server listens for authentication requests. Defaults to 1812.
   c. In the **Shared Secret** field, enter the secret code that is known to only the client (FortiSOAR in this case) and the secondary server.
   To check the RADIUS configuration of the secondary server, click the **Test Connectivity** button. Clicking **Test Connectivity** opens the **Enter Test Credentials** dialog in which you can enter the username and password used to connect to the secondary server. If the connection succeeds, then a success message gets displayed, and if the connection fails, then an appropriate error message gets displayed.
3. To save the configurations for your primary and secondary RADIUS servers, click **Save**.

You can use the Export and Import Wizards to export and import your Radius configurations across instances. For more information on export and import wizards, see the 'Export and Import Wizards' topic in the Application Editor chapter.

Importing RADIUS users in bulk

You can import RADIUS users in bulk into your FortiSOAR system. To import users, you must enable RADIUS authentication on the `RADIUS` page (**Settings** > **Authentication** > **RADIUS**). Once you have enabled RADIUS authentication, do the following to import users:

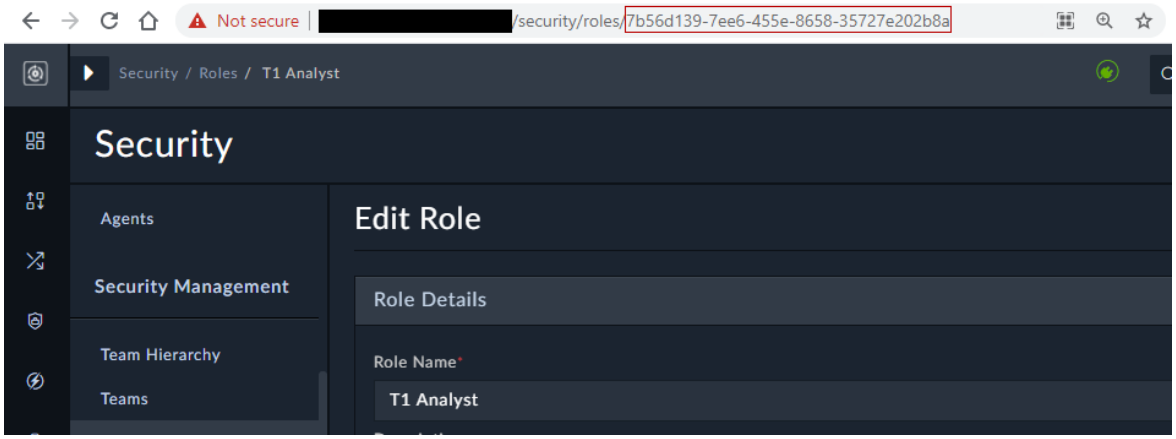
1. From the left menu, click **Users**, and on the `Users` page, click the **Import Users** button.
   Note: The **Import Users** button will be visible only if RADIUS or SSO is enabled.
2. In the `Import Users` dialog, do the following:
   a. From the **User Type To Import** drop-down list, select **RADIUS User**.
   b. Click the **Download CSV File Sample** link to download the sample CSV file (`RADIUS User_Template.csv`).



The sample CSV file contains an example of the user details you need to provide. You need to provide the following user details in the CSV file:

- username: Name of the RADIUS user.
- email: Email address of the RADIUS user
- firstname: (Optional) First name of the RADIUS user.
- lastname: (Optional) Last name of the RADIUS user.
- phonemobile: (Optional) Mobile number of the RADIUS user.
- roles: (Optional) Role (s) that you want to assign to the RADIUS user. To assign a role to the user you need to provide the UUID of that role. To get the UUID of a role, click **Settings** > **Security Management** > **Roles**, and then click the role that you want to assign to the user. For example, click T1 Analyst, which opens the Edit Role page, and then from the address bar, copy the UUID (as

shown in the following image) and paste it in the roles column in the CSV file.

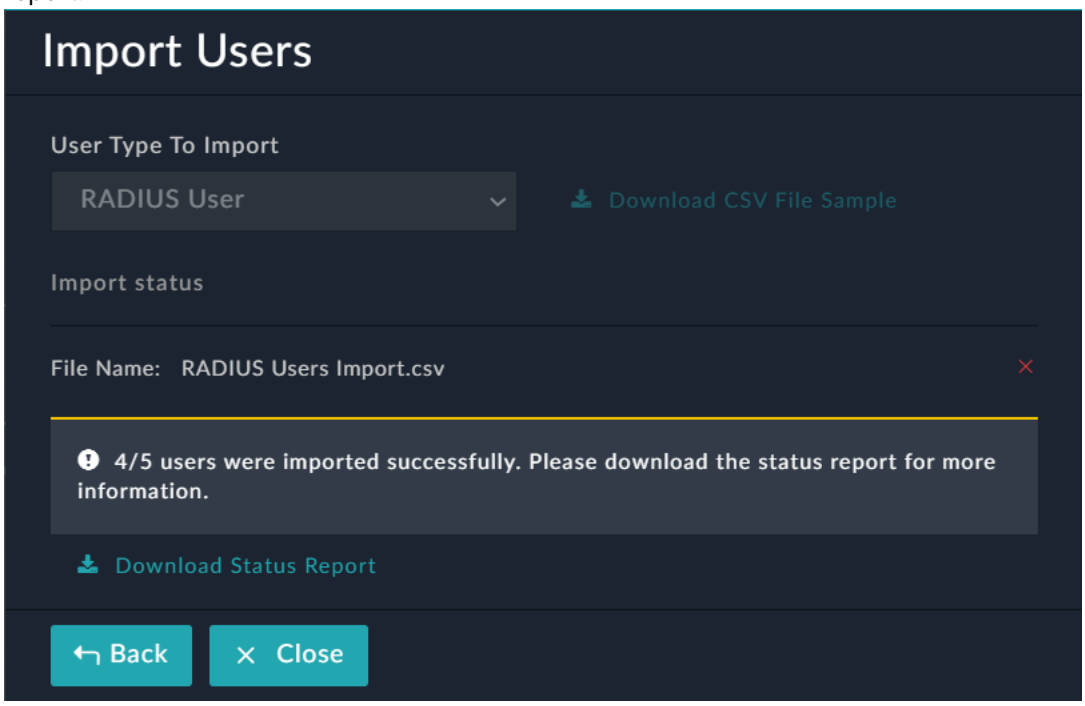


Note: You can assign multiple roles to the user by using the pipe symbol (|) to separate the UUID of each role.

- teams: (Optional) Team (s) that you want to assign to the RADIUS user. To assign a team to the user you need to provide the UUID of that team. To get the UUID of a team, click **Settings** > **Security Management** > **Teams**, and then click the team that you want to assign to the user, and then from the address bar copy the UUID of the team, similar to the process described for roles.
  Note: You can assign multiple teams to the user by using the pipe symbol (|) to separate the UUID of each team.
- accessType: Access type (Named or Concurrent) that you want to assign to the RADIUS user. If you do not specify any access type for the user, then the user will be assigned as a 'Concurrent' user.

3. Once you complete filling the user details in the CSV file, click the **Import User** button, and in the `Import Users` dialog, drag and drop the CSV file or click the import icon to import the CSV file, and then click the **Import Users** button.
   If there are no issues in the import, then all the RADIUS users get created and they can log into FortiSOAR.
   If there are any issues in the CSV file, such as not providing all the information required to create RADIUS users, then such users are not created. To get information about the users not created, download the status

report.



Configuring the Password Vault Manager

FortiSOAR integrates with external vaults such as "Thycotic Secret Server" and "CyberArk" that are used by organizations to securely store their sensitive data and credentials. Integration with external vaults also enables users to periodically change system credentials in their central vaults and automatically have configurations fetch those passwords using the vault.

To configure the Password Vault, you must be assigned a role that has `Read` permissions on the 'Connector' module, `Read` permissions on the 'Security' module, and `Update` permission on the 'Application' module.
To install and configure the connector for using the vault, you must be assigned a role that has `Create`, `Read`, `Update`, and `Execute` permissions on the 'Connector' module and `Read` permission on the 'Application' module.

FortiSOAR must have a connector created for a vault for you to be able to use an external vault in FortiSOAR. FortiSOAR has integrated with Thycotic Secret Server and CyberArk, and therefore we have a Thycotic Secret Server and CyberArk connectors in the Content Hub.

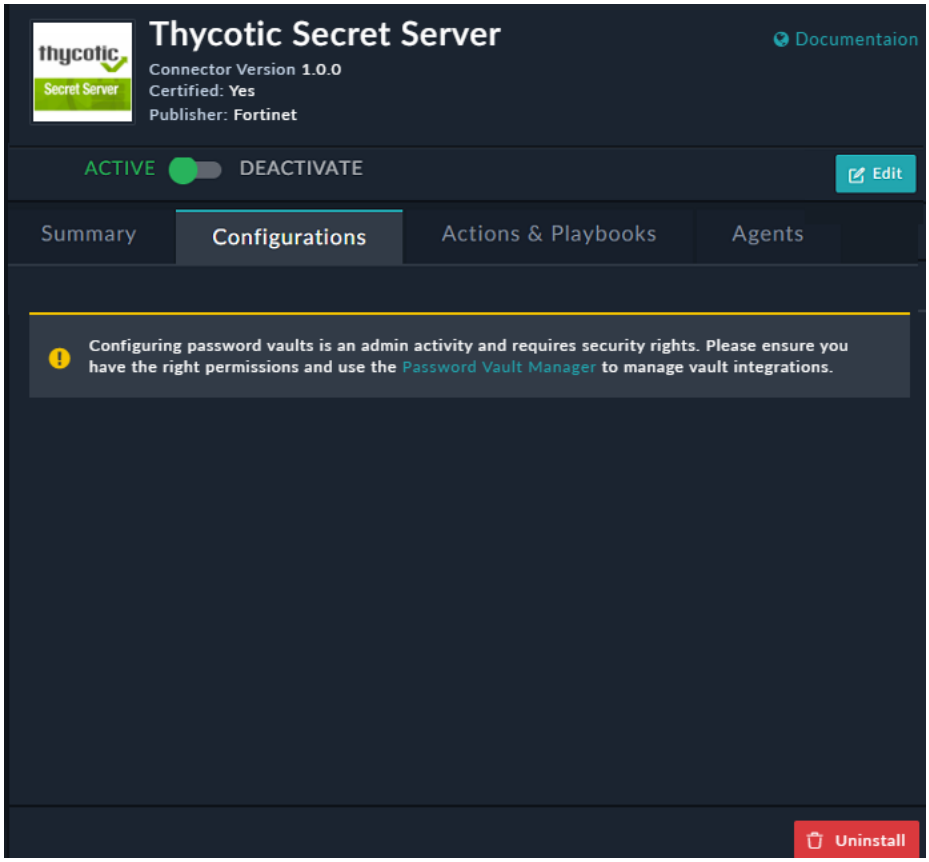
To use a vault in FortiSOAR, you must first install the connector from the Content Hub. For more information on installing a connector, see the *Introduction to Connectors* chapter in the "Connectors Guide."

Once you have installed the connector, you must configure the connector.

To install and configure the connector for using the vault, you must be assigned a role that has `Create`, `Read`, `Update` (CRU) permissions on the 'Connector' module and `Read` and `Update` permissions on the 'Security' module, `Read` and `Update` permission on the 'Application' module.

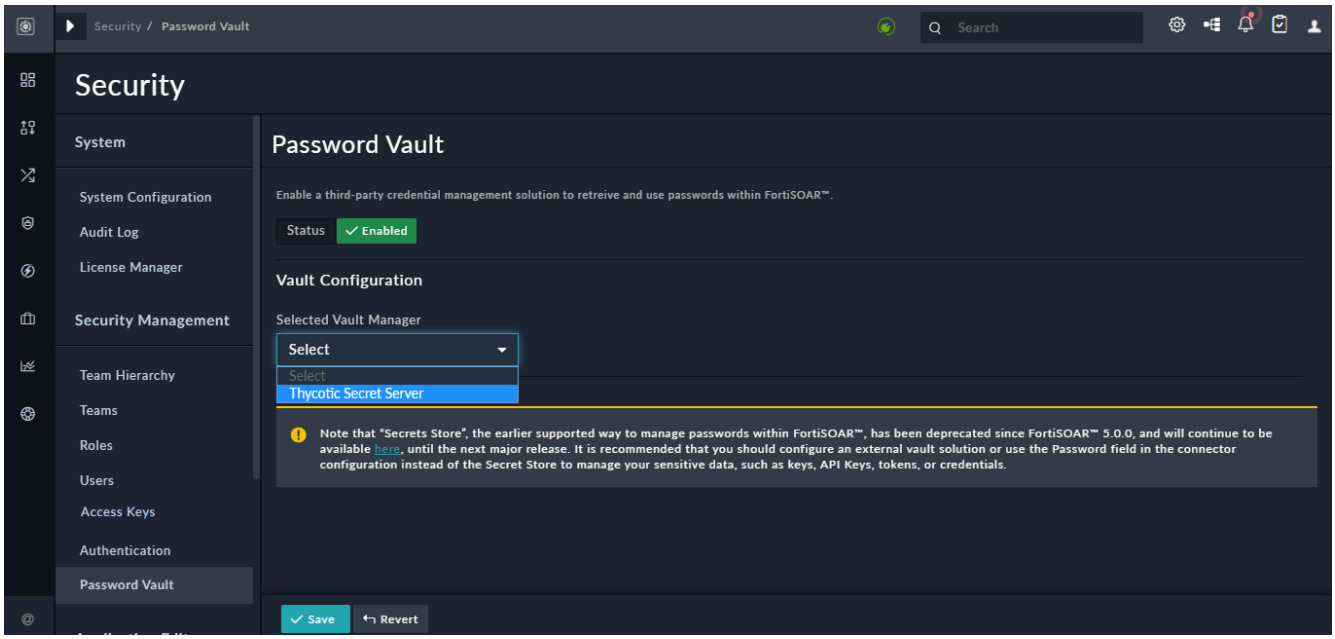
This section describes configuring the "Thycotic Secret Server" connector. You can configure "CyberArk", or any other vault connector that gets integrated with FortiSOAR in the future in a similar manner.

Important: You cannot configure a connector that is integrated with an external vault on the `Connector Configuration` dialog as is the case with other connectors. Once you installed the connector and if you have appropriate permissions, the following `Connector Configuration` dialog is displayed in the case of Thycotic Secret Server:

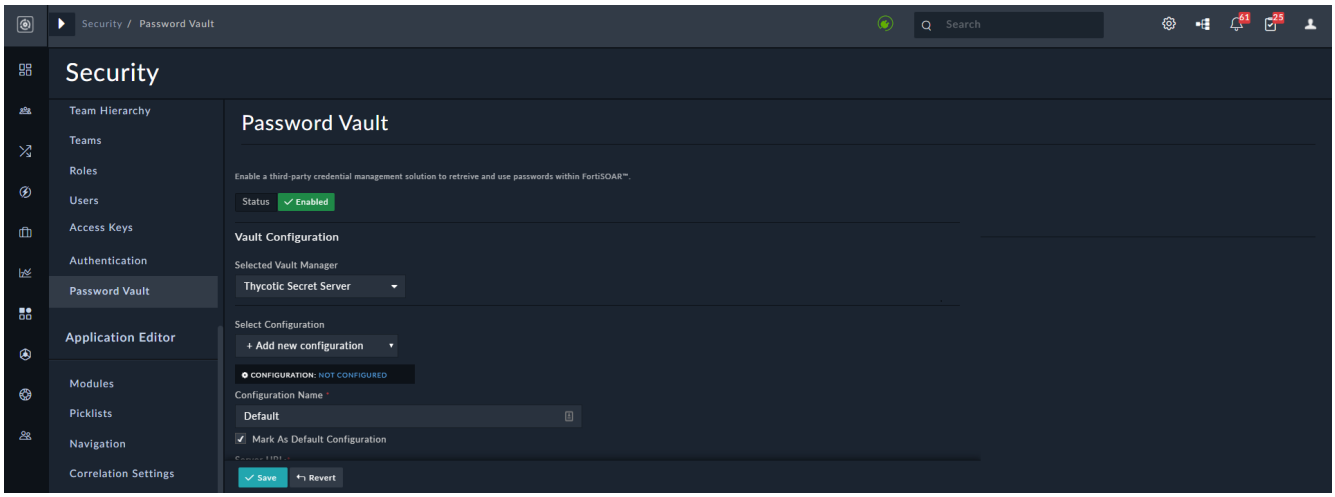


You can open the Password Vault Manager by either clicking the **Password Vault Manager** link on the connector configuration dialog or by clicking **Settings** > **Password Vault**.

On the `Password Vault` page, click the **Disabled** button to enable integration with external vaults and configure the selected vault. From the **Selected Vault Manager** drop-down list, select the vault that you want to use.

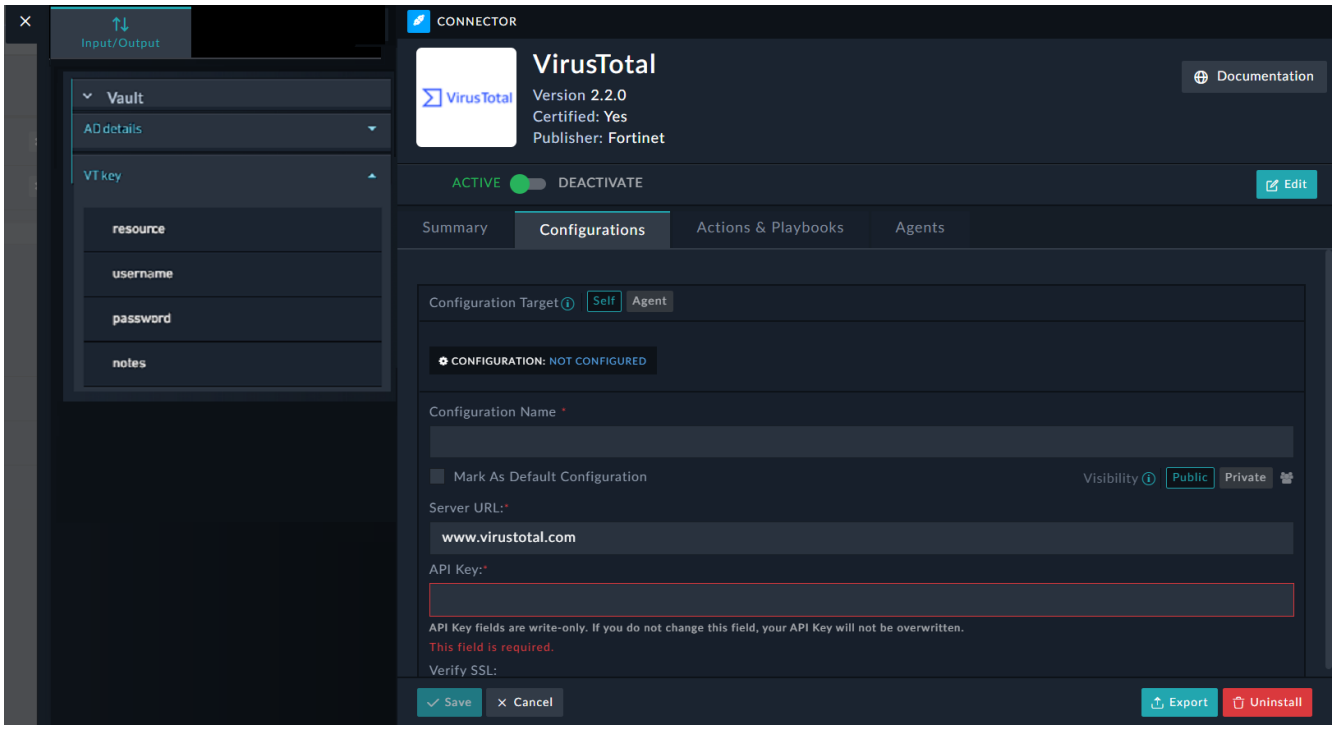


In our example, select Thycotic Secret Server, configure the connector and then click **Save**.



Note: You can add multiple configurations for a vault; however, you must select a particular configuration for integration with FortiSOAR. Similarly, you might have multiple vaults, but you can only have one vault integrated with FortiSOAR.

Credentials (passwords, keys, tokens, etc) that you have stored in the vault are not visible to the users. However, once you have configured your vault, then users can use the credentials stored in the vault in connector configurations. For example, if you have a user who is creating a playbook that requires access to VirusTotal, a 3rd party integration, and you do not want to provide the VirusTotal API key to users, you can store the credentials in an external vault. Users can then select the vault credentials in the connector configuration steps by clicking the Password or Set API Key field, which then displays the `Dynamic Values` dialog > **Input/Output** tab from which you can select the required credentials, as shown in the following image:



For more information on Dynamic Values, see the *Dynamic Values* chapter in the "Playbooks Guide."

You can also continue to use the `Set Password` field in the connector configuration to securely store and manage sensitive data, such as keys, API Keys, tokens, or credentials.

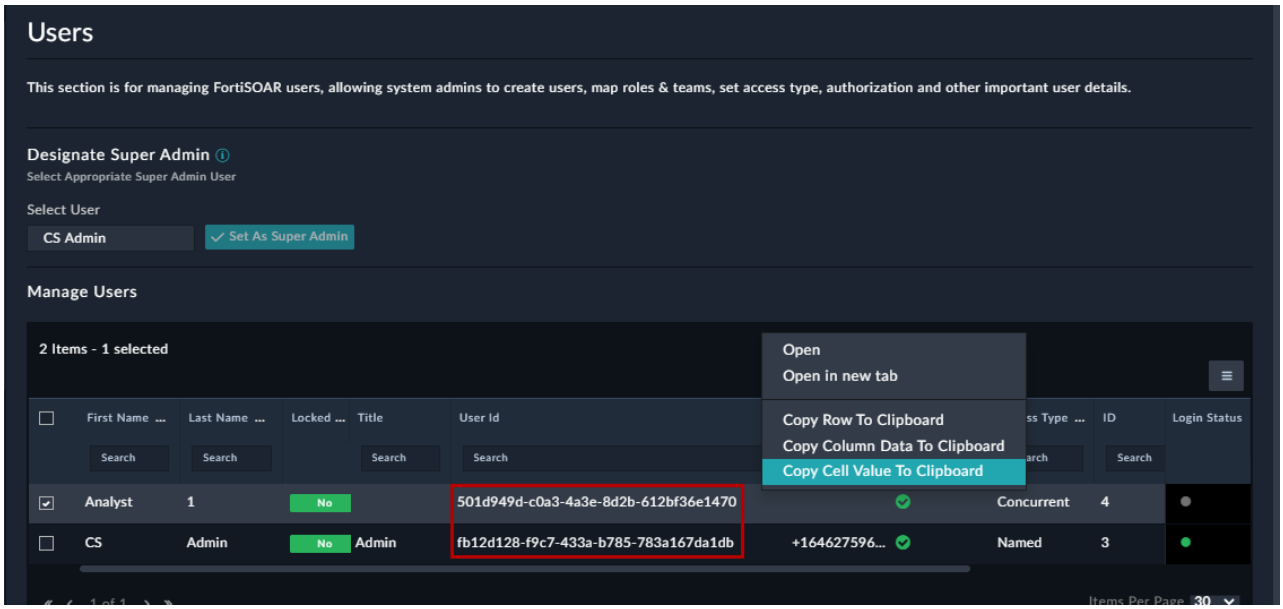
Delete Users

Apart from the above functions that an administrator can perform on the FortiSOAR UI, administrators can also delete users using a script.

It is highly recommended that you use this script to delete or cleanup users during the initial stages of setting up your FortiSOAR system. If you delete users who have been using FortiSOAR for a while, then the records for which the deleted user was the only owner, will also be lost forever.

To delete users, perform the following steps:

1. SSH to your FortiSOAR VM and login as a *root* user.
2. Enter the `userId` of the user(s) that you want to delete in the `usersToDelete.txt` file, which is located at `/opt/cyops/scripts/`. The `usersToDelete.txt` file is an empty text file in which you can enter the ID of users that you want to delete.
   The UserID of the users are displayed in the **User Id** column on the `Users` page in the FortiSOAR UI as shown in the following image:



To copy the UserID, right-click the 'User Id' cell and from the context menu, select the **Copy Cell Value To Clipboard** option.

3. Navigate to the `/opt/cyops/scripts/` directory.
4. Run the following command:
   `# /opt/cyops/scripts/userDelete`
   Important: The User Delete script deletes users in the local database and does not work for externalized databases.

Application Editor

Use the Application Editor to configure data models contained in modules, to export and import configurations, visually display the nodes related to a particular record, customize your Picklist values, and the left navigation bar.

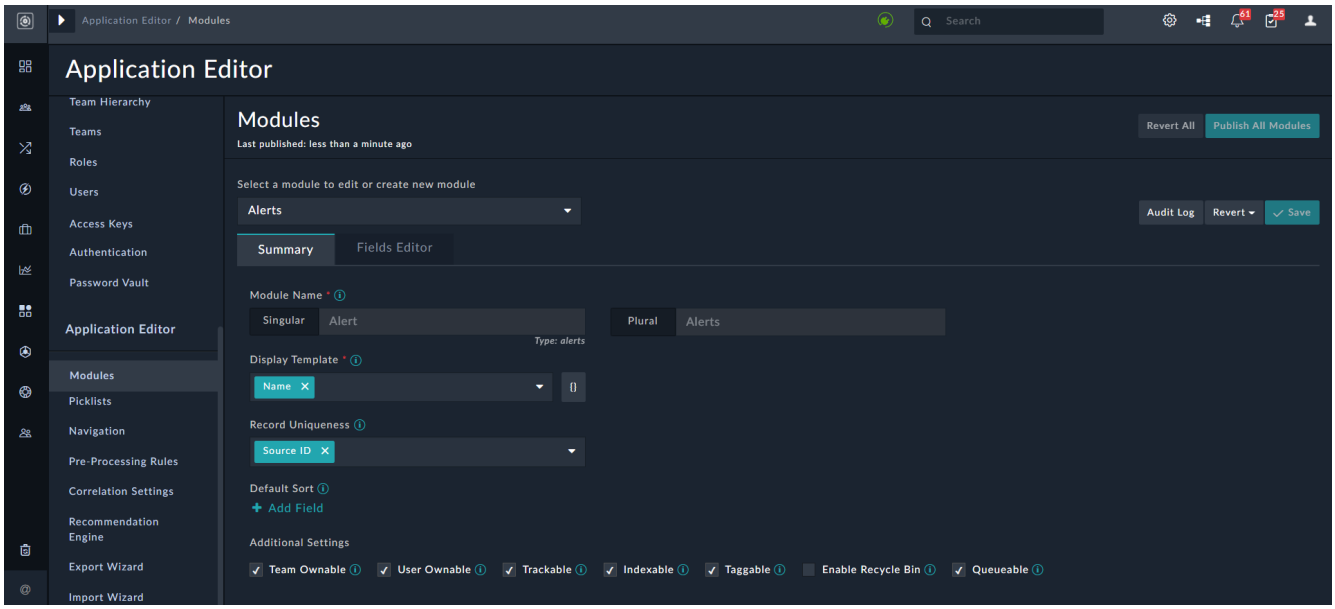
The Application Editor has following tools for this purpose:

- Module Editor - for editing the data models in a module
- Picklist Editor - for changing picklist values and color associations
- Navigation Editor - for modifying the navigation links and hierarchy in the left navigation bar
- Pre-Processing Rules - for helping uses define rules to detect and drop duplicate records based on predefined criteria before storing the incoming records in the database.
- Correlation Settings - for configure the display of the Visual Correlation widget.
- Recommendation Engine - for predicting and assigning field values based on Artificial Intelligence/Machine Learning (AI/ML)
- Configuration Manager - for exporting and importing configurations across environments

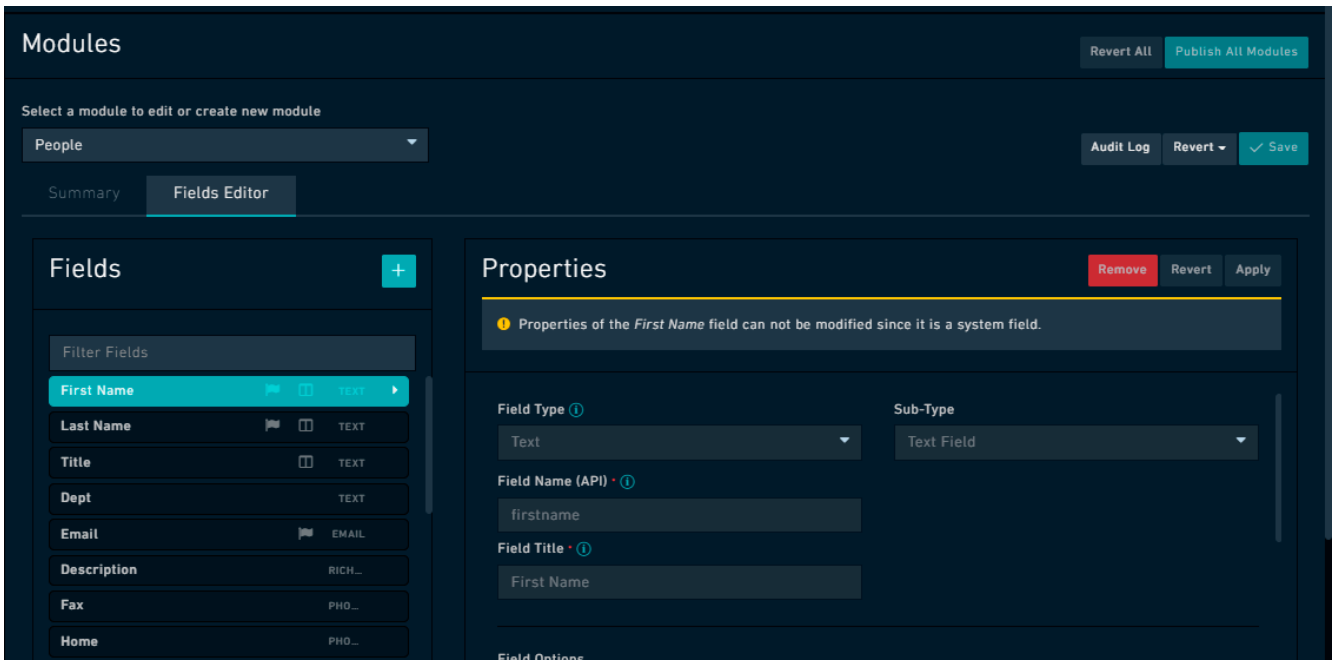
To edit these settings, users must be assigned a role that has at a minimum of 'Read' and 'Update' permissions on the "Application" module.
If you want a user to be able to add modules also, then those users must be assigned a role that has at a minimum of 'Read,' 'Create,' and 'Update' permissions on the "Application" module.
To delete picklists or navigation items, you must have 'Delete' permissions on the "Application" module.
These privileges must be granted carefully as unintended application modification could result in data loss.

Module Editor

Use the Module Editor to add new modules and to add new fields and edit existing fields within a module. You can open the module editor by clicking **Settings** and in the `Application Editor` section, click **Modules**. This displays the `Modules` page.



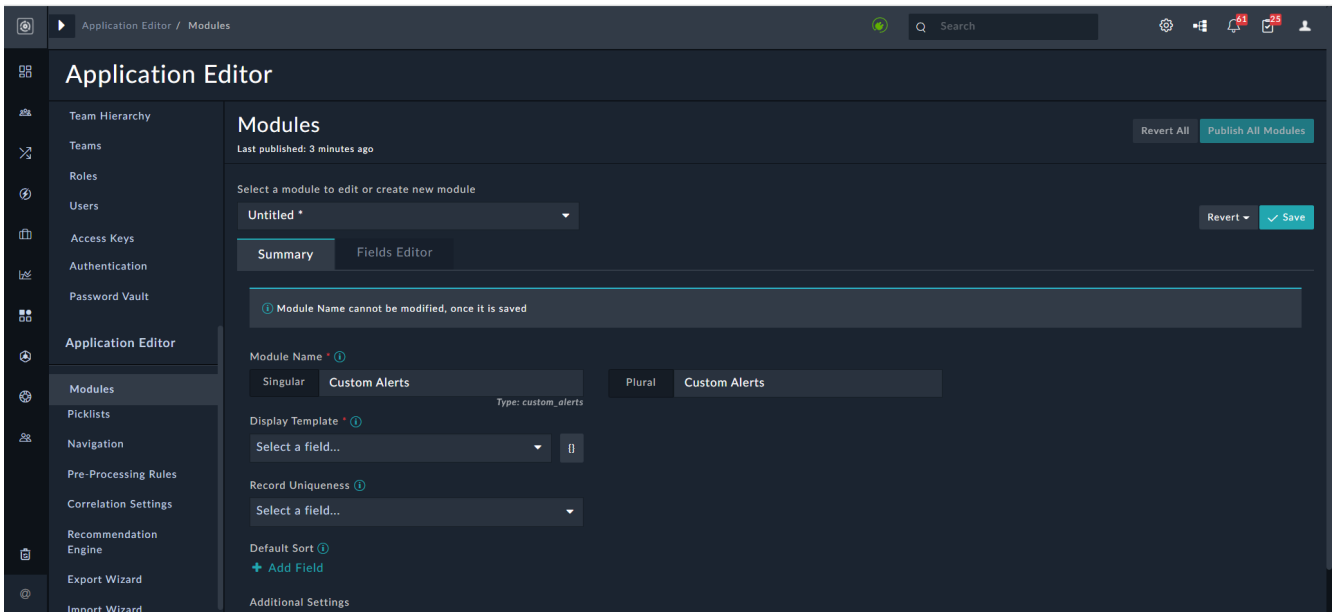
Important: Some fields in some modules are system fields and only FortiSOAR can create system fields. Changing the properties of the system fields could lead to issues with the working of FortiSOAR. Therefore, you cannot modify any properties of these fields, and they appear as read-only when you select them in the **Fields Editor** tab. An example of this type of field is the `First Name` field in the `People` Module as shown in the following image:



Creating a New Module

To create a new module, click **Settings** > **Modules**. This displays the `Modules` page. Use the **+Create new module** option that appears at the top of the editor to define the properties of the module. By default, when

choosing the Module Editor, the ability to define the new modules is available.

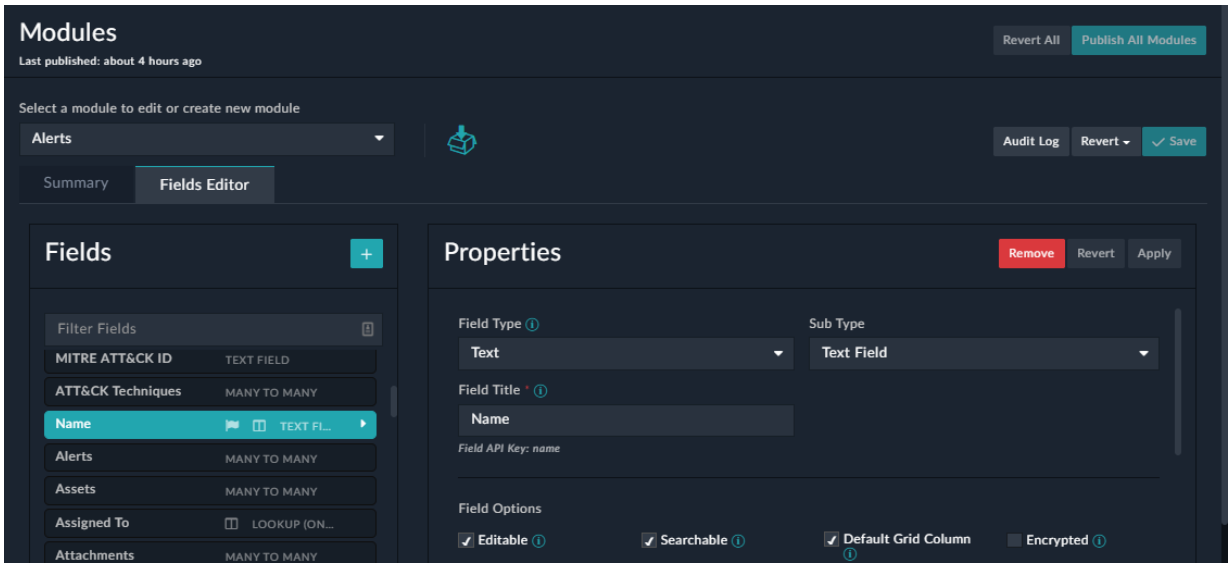


Bear in mind there are requirements to realize the addition of the new module, notwithstanding the need to allow for the interface to recognize this module.

From version 7.0.2 onwards, the process of creating a new module has been simplified so that users needs to provide only the **Module Name** while creating the module. Legacy fields such as table name, which could cause issues if incorrect inputs were provided are auto-generated. Simplification of module creation enhances user experience and also eliminates errors that occur due to incorrect inputs. In addition to this in case of the many to many relation field type, fields are by default added with same module name to the other side of related module, which simplifies the process of creating many to many relation fields. Also, now users only need to select the **Related Module** and provide the value in the **Field Title** field to create relation fields.

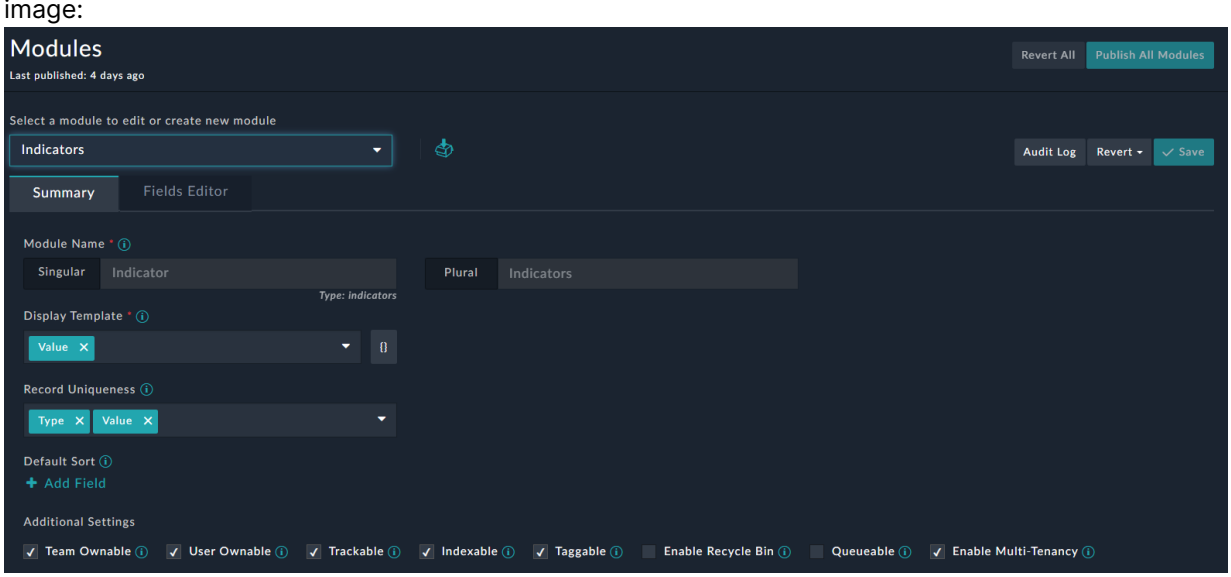
To add a module, do the following:

1. Click **Settings** and click **Modules** in the `Application Editor` section, to open the Module Editor. This displays the `Modules` page.
2. On the **Summary** tab configure the following fields for defining a module using the **Module Editor**.
   Note: To add any field to the `Summary` view, you must ensure that you have already added the field using the `Field Editor`. See the Modifying an existing module section for information on how to add and edit fields.
   a. **Module Name**: The name of the module itself, when it is in the singular context (individual record). For example, an Incident refers to an individual record in a module, whereas the module is Incidents.
   Once add the name of the module in the **Singular** context, the name of the module in the **Plural** context gets automatically populated.
   Note: The module 'Type' is also automatically populated based on the name you specify in the Singular field. Type is used to identify the module in the API.
   b. **Display Template**: This field uses an Angular Template expression to display a record appropriately. This template specifies the fields that will be displayed when a record from this module is referenced in the application. Fields of a module can be specified in the display template as `{{ field_name_api }}`.
   Important: Ensure that you add the fields that you specify in **Display Template** in the module that you are creating or updating. For example, if you have added `{{ name }}` in **Display Template**, then ensure that you have added **Name** as a field in the module, its **Field Title** (**Field API Key**)attribute will be set as `name`.



You can also add an attribute of a picklist as part of the Display Template. See the following Display Template section for more details.

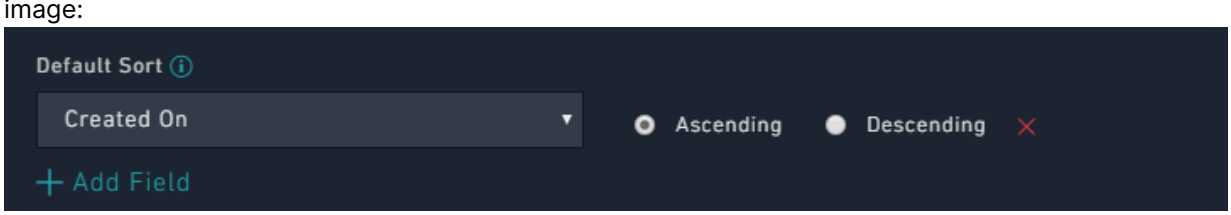
c. **Record Uniqueness**: This section allows you to choose one or more published fields from the existing module that would be used to define a unique condition. This ensures that only unique records will be created in FortiSOAR and any record that matches the unique field or the combination of unique fields would not be created.
For example, *Type + Value* is a unique combination in the `Indicators` module as shown in the following image:



This will ensure that indicator records will be created with unique "Type and Values" values.
If you want to add any other field to for part of the unique combination of fields, then from the `Record Uniqueness` section, select that field from the **Select a Field** drop-down list, and then click **Add Field**. Similarly, if you want to remove any field from the unique combination, you need to click the red **X** on that field.

d. **Default Sort**: Click **Add Field** to add a field based on which the list of records in the module will be sorted, either in the ascending or descending order. For example, indicators will be sorted based on when they were created, if you add **Created On** in the Default Sort section, as shown in the following

image:



e. **Team Ownable**: Records that are ownable can be owned by a team or multiple teams. An example of a module that you should make ownable is Alerts. When a new record is created in an ownable module, the ownership is assigned by default to the user or appliance that created the record. For example, if User X, who belongs to both Team A and Team B, creates a record named 'Alert 1', the record will be automatically associated with both Team A and Team B. Consequently, all users in these teams will have access to 'Alert 1'. Similarly, if a record named 'Alert 2' is created by a Playbook Appliance, the ownership will be assigned to the Playbook Appliance Team, even if no team is explicitly specified during record creation.
If you do not select this option, then the records are not ownable and will be visible to all users who have access to that module, regardless of their of the user's team. An example of a module that should not be ownable is Addresses.
Note: Records without owners are visible to all users who have access to the module. If you change a module from not ownable to ownable, then the records created before you have changed the ownership are visible to "All". However, until owners (teams) are assigned these records are read-only, meaning their fields cannot be edited. Also, users' who are editing the same record (with no owners) must assign their team to the records to ensure that the records continue to be visible to those users and/or teams.

f. **User Ownable**: Records that are user ownable can be owned by Users or Appliances. An example of a module whose records you should make user ownable is Alerts. If you do not select this option, then the records within this module will be visible to all users.

g. **Trackable**: Selecting this option ensures that FortiSOAR tracks the date that the record was created, user who has created the record, date that the record was modified, and user who has modified the record on all records in the module.
Important: Once a module has been created, you must not modify the `Type` and `Table Name` fields. You can edit the Singular and Plural names whenever required. However, note that the Singular name plus an `s` is used for API endpoint generation during module creation. Changing the Singular name could disrupt existing API calls to the endpoint.

h. **Indexable**: Selecting this option ensures that this module has fields that can be indexed.

i. **Taggable**: Selecting this option ensures that the selected module is taggable, i.e., you can enter tags to records in this module making it easier to search and filter records in the module.

j. **Enable Recycle Bin**: Selecting this option enable soft deletion of selected module's records, i.e., records that are deleted from this module move to the recycle bin from where they can be restored, if needed. For more information on recycle bin, see the Recycle Bin chapter.
Note: You cannot select the **Enable Recycle Bin** checkbox for system modules, i.e., for the 'People', 'Appliances', 'Agents', 'Approvals', 'Tenants', 'Routers', 'Comments', and 'Saved Reports' modules. Therefore, recycle bin for these modules cannot be configured and records of these modules are always permanently deleted.

k. **Queueable**: Selecting this option ensures that this module is displayed as an option for auto-assignment while creating or configuring queues, i.e., records of this module can be auto-assigned using queues. For more information, see the *Queue, Shift, and Leave Management* chapter in the "User Guide."

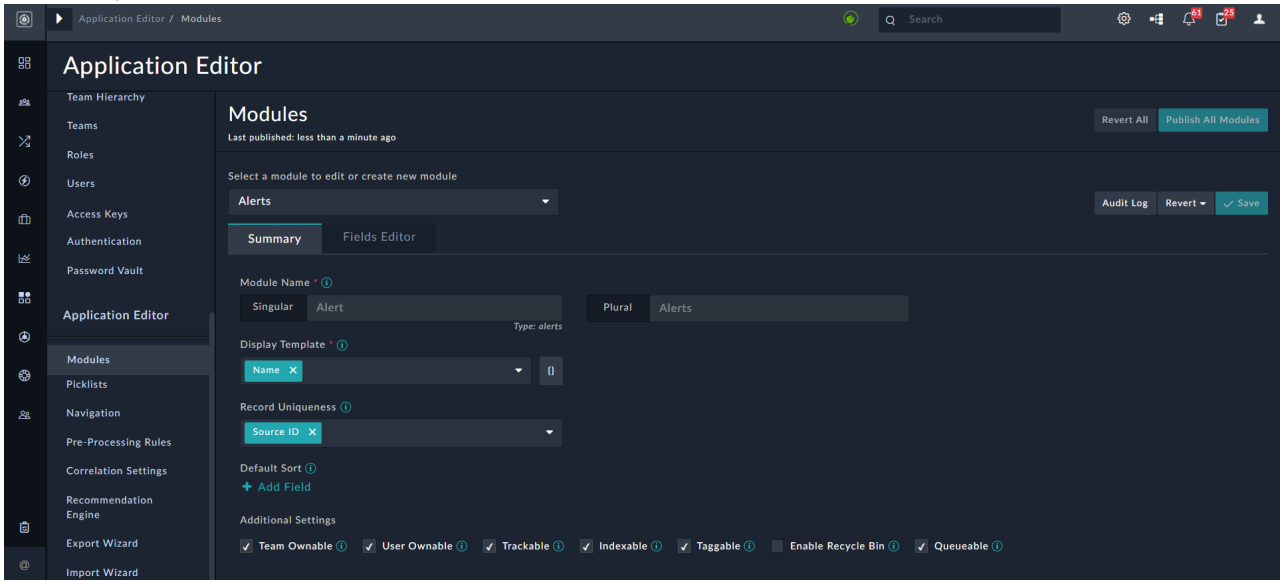
   l. **Enable Multi-Tenancy**: (Only applicable to multi-tenant setups): Selecting this option enables replication of data for the selected module across peers' setups. Enabling this option also adds the 'Tenant' field to the schema for the selected module, which can then be used to associate records with specific tenants in the system.
   NOTE: You cannot enable multi-tenancy for system modules such as 'Agents', 'Appliances', 'People', etc.

3. On the **Fields Editor** tab, add the required fields to the module and click **Save**. See the Modifying an existing module section for information on how to add and edit fields.
4. On the Summary tab, click **Save** to save the changes to save the changes to the module and publish the module to reflect the changes in the system. For information on the Save operation, see the Saving your changes section. For information on publishing, see the Publishing Modules section.
   To clear any changes made in the interface since the last `Save` event, click **Revert**.

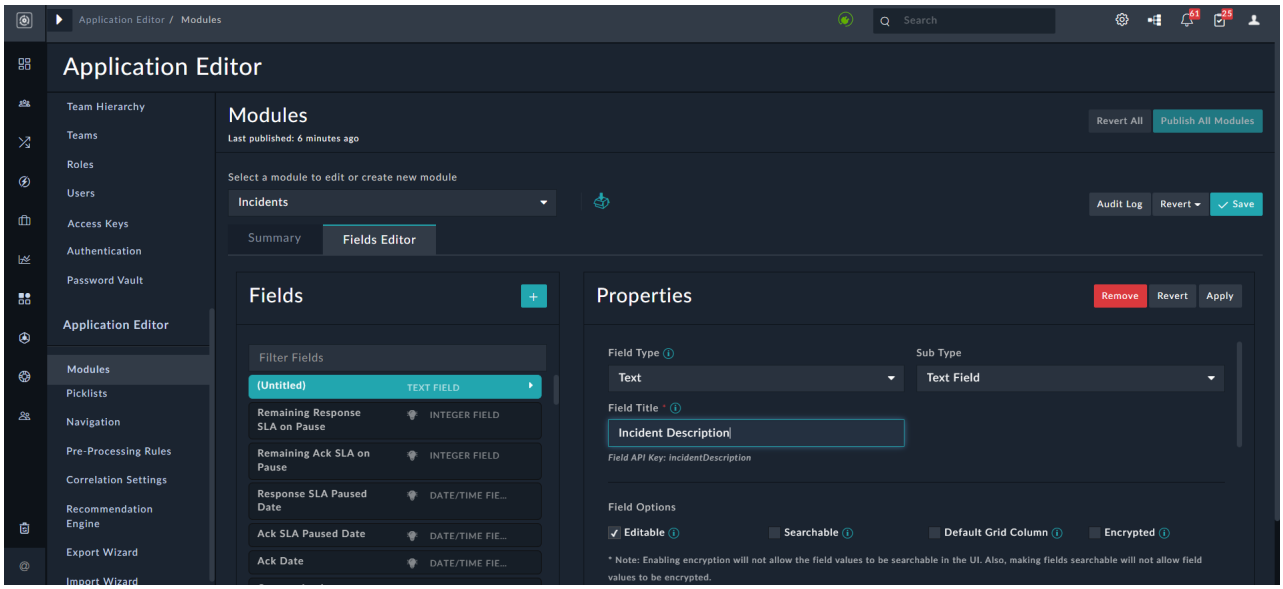
Modifying an existing module

To modify an existing module, do the following:

1. Click **Settings** and click **Modules** in the `Application Editor` section, to open the Module Editor.
   This displays the `Modules` page.
2. To edit an existing module, from the **Select a module to edit or create new module** drop-down list, select the module.
   For example, select `Alerts`.



3. To add fields to the module, or to edit any of the fields belonging to the module, add or update the fields on the **Fields Editor** tab on the `Modules` page.



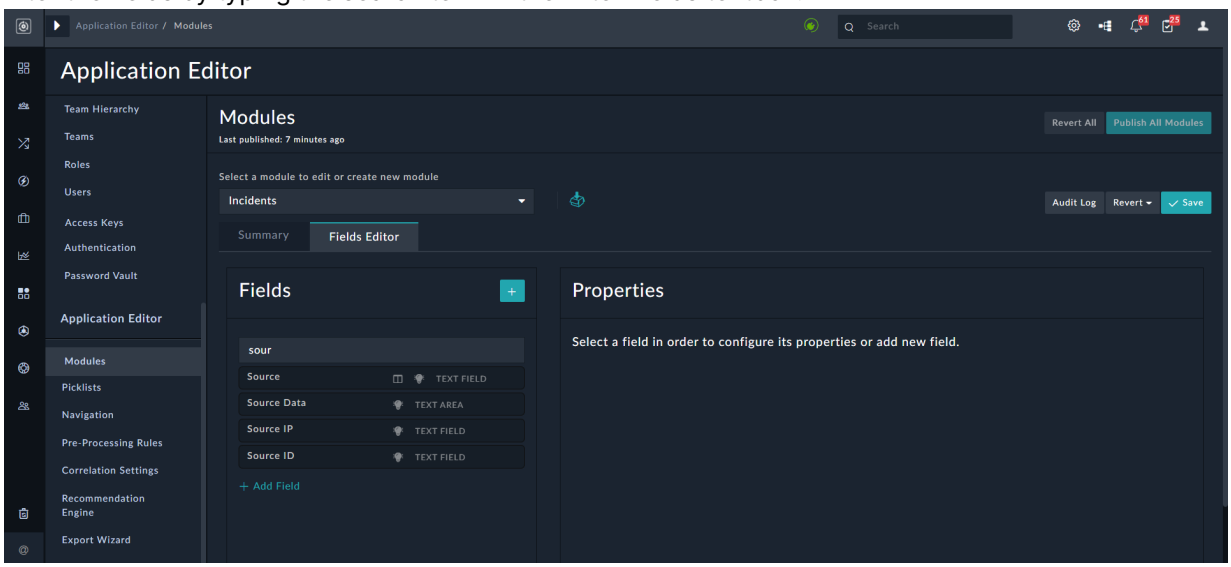
Some fields in some modules are system fields and changing properties of these fields could lead to issues with the working of FortiSOAR. Therefore, you cannot modify any properties of these fields and they appear as read-only when you select them in the **Fields Editor** tab. An example of this type of field is the `First Name` field in the `People` Module.

From version 7.0.2 onwards, one-time edits are allowed for read-only fields, in the following cases:

- If the value of the field had not been set, i.e., it is 'null' or 'blank'
- If the field name is "tenant", and the field value is "self"

One-time edits of read-only fields is required since there are scenarios where the value of the fields is not known at the time of record creation. An example of this is that at the time of record (alert) creation, the tenant to be mapped to the alert is not uniquely identifiable. Only after extraction and enrichment of the alert can the tenant be correctly determined. Therefore, now it is possible to change the tenant attribute of a record one time, if the current value is pointing to the local tenant ("Self").

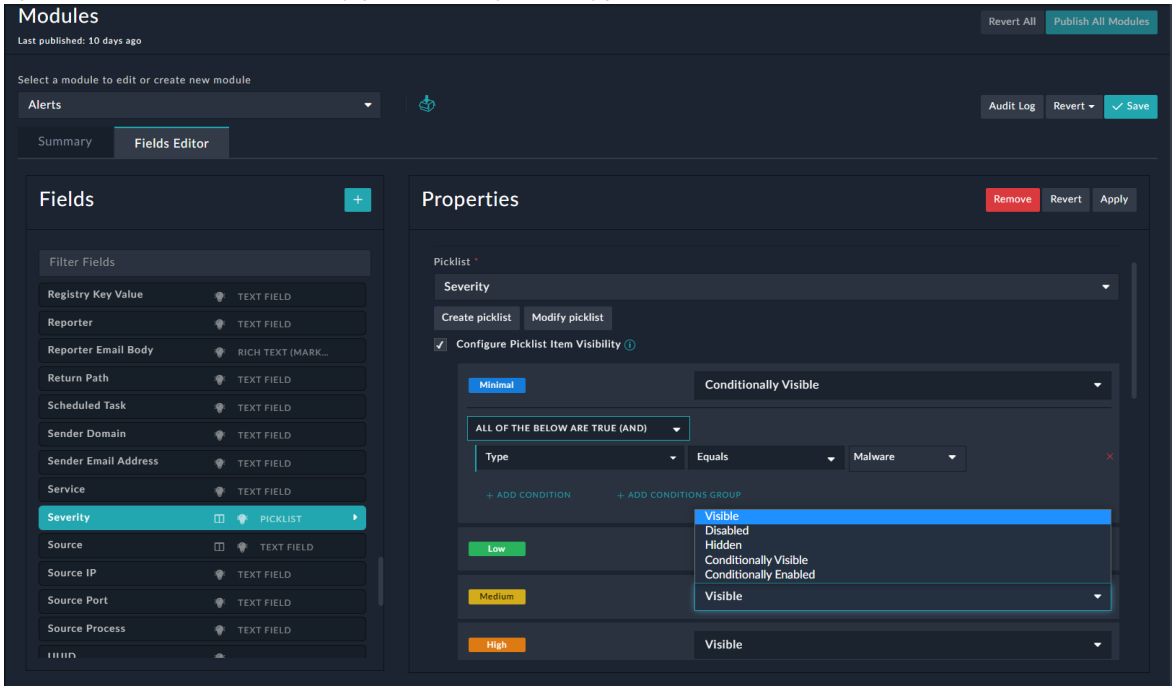
a. To add fields, on the `Modules` page, click the **Fields Editor** tab and click the Add (**+**) icon beside `Fields`. **Note**: The Fields Editor pane displays the list of fields that have been defined for the module. You can filter the fields by typing the search term in the Filter Fields textbox:



b. Define the following properties for the newly added field:

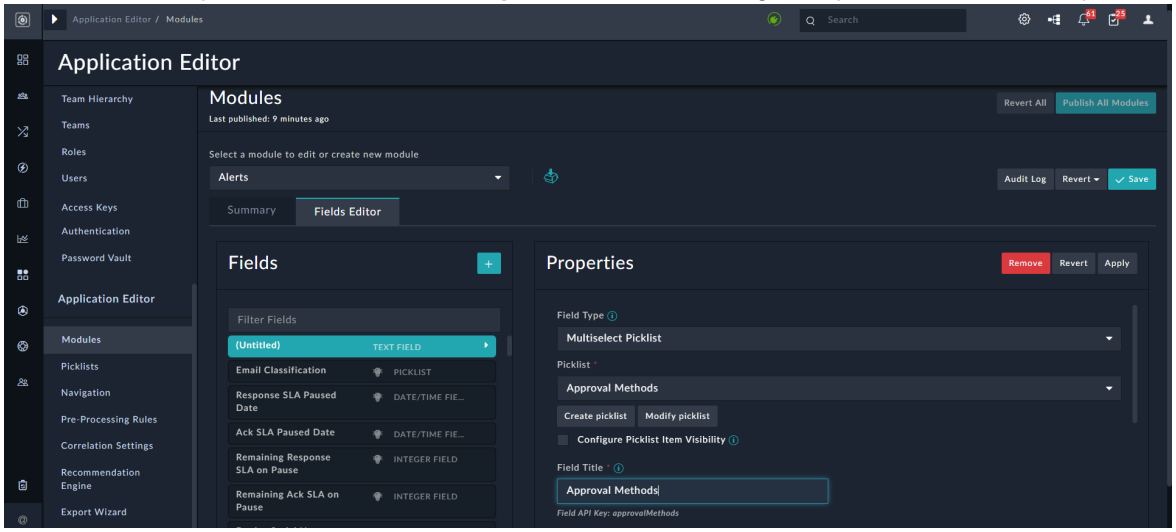
- **Field Type**: The type of field; it specifies the type of form used to render this attribute. Examples of field types are text, file field, checkbox, integer, decimal, date/time, picklist, and relationship fields. It is recommended that when you create a field of type `Integer`, then you should set its default value as "zero".
  Use the JSON field type for fields such as, `Source Data`, which commonly store data in the JSON format to allow playbooks to get to the JSON data directly, without having to use a JSON parse step. You can also define the height of the JSON field in pixels by editing your record template.
  Based on the **Field Type** that you select, you can see an additional field. For example, if you select the field type as `Picklist`, a **Picklist** drop-down list will appear, and you must select the picklist associated with the field or click **Create new picklist** to add a new picklist. Use the **Configure Picklist Item Visibility** checkbox to hide (or display) a picklist items for the field based on specified conditions. Once you check this checkbox, the picklist items are displayed and you can choose whether the item should be Visible, Disabled, Hidden, Conditionally Visible, or Conditionally Enabled.
  - **Visible**: The picklist item is always visible.
  - **Disabled**: The picklist item is visible but not selectable when adding or editing the field.
  - **Hidden**: The picklist item is hidden when adding or editing the field; however the picklist item is searchable.
  - **Conditionally Visible**: The picklist item is visible only when the specified conditions are met; else it is hidden.
  - **Conditionally Enabled**: The picklist item is visible and selectable only when the specified conditions are met; else it is disabled.

  If you choose **Conditionally Visible** or **Conditionally Enabled**, you require to define the condition when the picklist item should be visible or enabled. An example would be displaying the 'Minimal' option, in case of the Severity picklist, only if the type of alert is Malware:



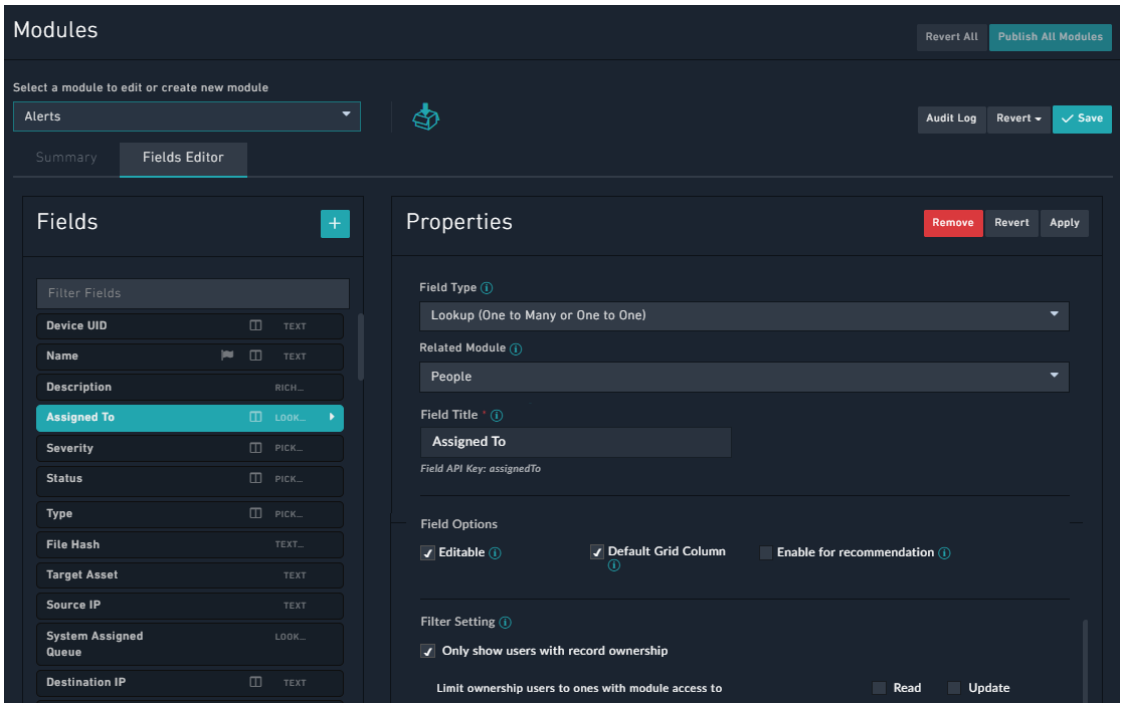
FortiSOAR also supports a special type of picklist, called "Multiselect Picklist". You can use the

multiselect picklist for fields that can contain more than one value. For example, you can have an alert be assigned more than one "Type", i.e., an alert can be of type Brute Force Alert and Malware. In such a case you can assign **Multiselect Picklist** as the *Field Type* for the "Type" field.
To create a new picklist click **Create new picklist** and then assign the picklist items to that picklist:

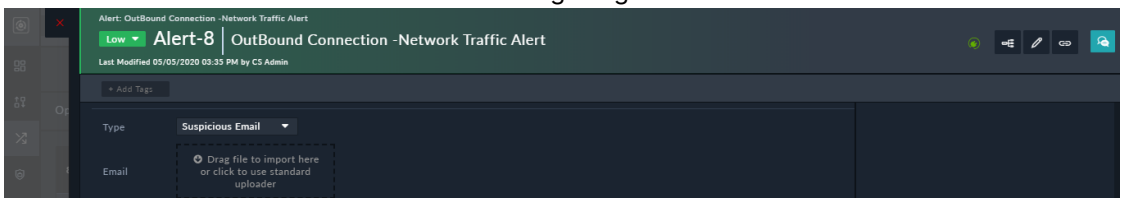


You can also modify an existing picklist, for example **AlertType**, by adding or removing picklist items or changing the properties of the picklist itemsselect an existing picklist
If you select `Lookup (One to Many or One to One)`, `Many to One`, or `Many to Many`, a **Related Model** drop-down list will appear, and you must select the related module, and provide the **Field Title** before you publish the module.Also, in case of the `Many to Many` relation field type, fields are by default added with same module name to the other side of related module.
Notes with respect to Relationship fields:

- If you have added a `Many to One` relationship field in a module, then it is recommended that you set up a `Lookup` relation field on the related module for the modules to work as expected. If you do not set up a `Lookup` relation field on the related module, the 'Publish' operation fails with an appropriate error message displayed on the FortiSOAR UI. The error message displayed is as follows: "`For many-to-one "<nameOfRelationshipField>" field in "<nameOfModule>" module there is no lookup field present in "<nameOfRelatedModule>" module.`"
- In the case of `Lookup (One to Many or One to One)` fields that have `People` as a related module contain the `Filter Setting` section. If you select the **Only show users with record ownership** checkbox in the `Filter Settings` section, then the list of users displayed in the lookup on the UI is restricted to include only those users who have record ownership. Further, you can also limit the list of users displayed in the lookup based on permissions given to the user on the module using the Limit ownership users to ones with module access to option. In the **Limit ownership users to ones with module access to** option, you can choose to display users who have **Read** access or **Update** (**Read + Update**) access.



Example: In the Alerts module, we have an **Assigned To** field, which is of type `Lookup (One to Many or One to One)`, with **People** as the related module. In this case by default, all users will be displayed in the Assigned To lookup, when you open an alert record. However, this could include users who belong to other teams, and who, therefore, would not have access to the record, even if you assign that record to that user.
Therefore, to restrict the users to only those users who have access to the record, you can select the **Only show users with record ownership** checkbox. You can further restrict users displayed in the **Assigned To** lookup based on the module access. For example, if you want to display only those users who can update the record, in the **Limit ownership users to ones with module access to field**, select the **Update** checkbox (once you select the Update checkbox, the Read checkbox is automatically checked).
You can drag and drop files in the "File Field" type field. An example of a "File Field" is "Email"; therefore, if you have an **Email** field in your record, then you can drag and drop an email to attach it to the record as shown in the following image:

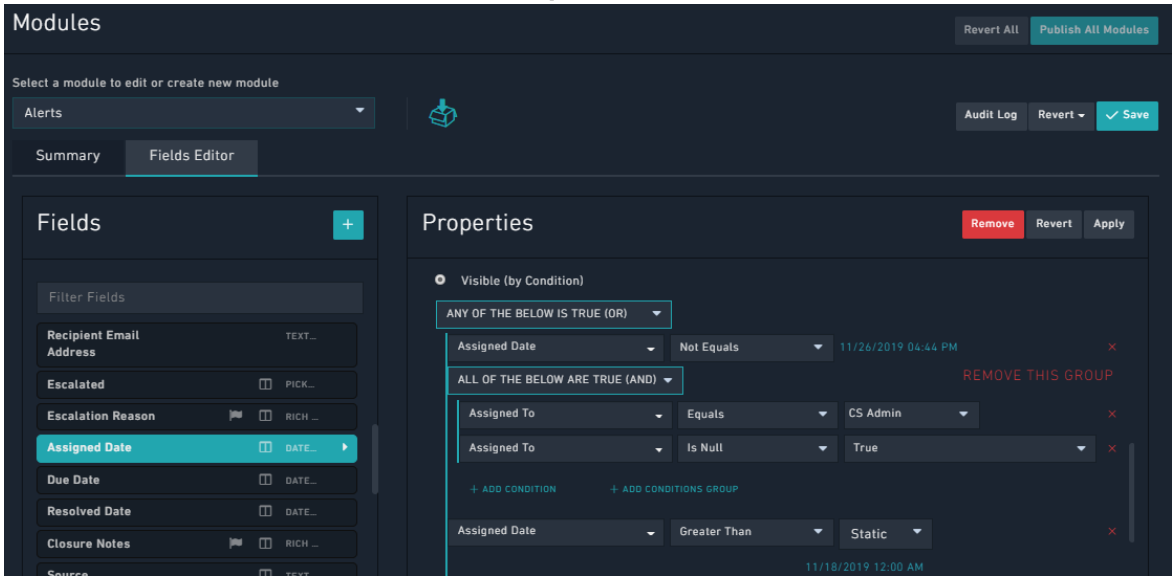


- **Sub Type**: The "Sub Type" field can be used along with the **Text** and **DateTime** (from release 7.2.0 onwards) type fields.
  When you select **DateTime** in the `Field Type`, then the **Sub Type** drop-down list is displayed, from which you can select either **Date Field** or **Date/Time Field** as the sub types. Select the sub type as **Date** for fields that do not require the time component, and only require the date to be displayed. For **Date** fields, the time is considered as 12 A.M.
  Note: From release 7.2.0 onwards, `DateTime` fields, such as 'Created On', 'Modified On', etc. are stored with milliseconds precision (earlier it was seconds). If you have upgraded to release 7.2.0 or later from a release earlier to 7.2.0, and users add new values to any existing `DateTime` fields, then

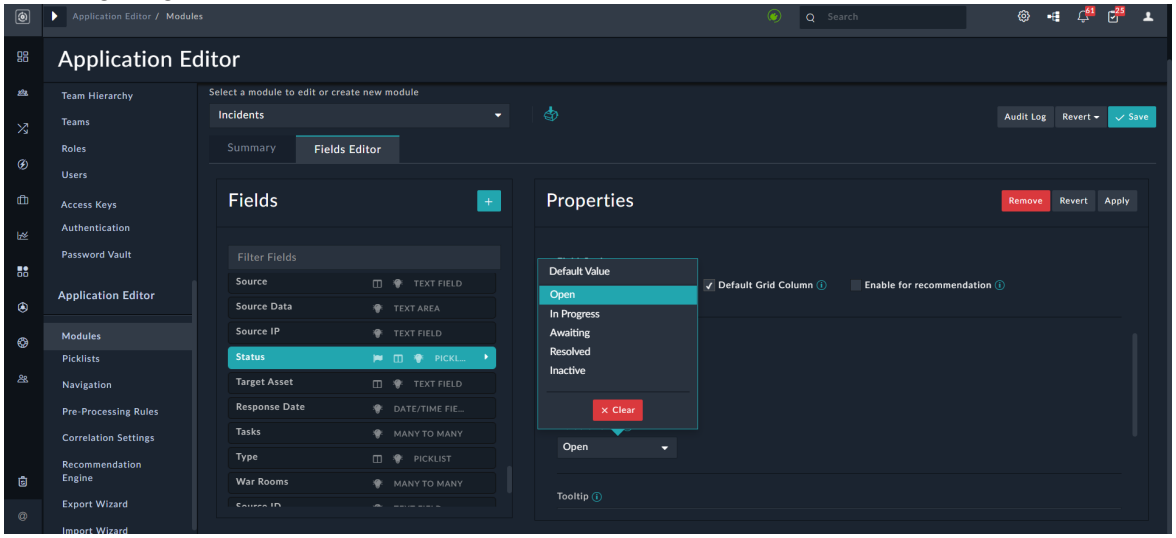
those are stored with milliseconds precision; however, values in the `DateTime` fields that were present before the upgrade would yet be present in the old format, i.e., stored with the seconds precision.
When you select **Text** in the `Field Type`, then the **Sub-Type** drop-down list is displayed, from which you can select the sub-types such as, Text Field, Rich Text (Markdown), Rich Text (HTML), Text Area, IPv4, IPv6, Domain, URL, or Filehash. The sub-type field enforces the format of data that the user can enter in that field. For example, in a Rich Text (HTML) or Rich Text (Markdown) fields, you can use formatting options or you can use the IP address and domain field types to lookup threat intelligence tools and whois info. You can also add images in the richtext fields (or in the Richtext Content widget); however, in case of .svg files, the Chrome browser does not restrict expansion of XML entities inside SVGs, which can rendering of image tag with any link in the href attribute that could case a client-side denial of service. Therefore, administrators can choose to not load SVG files enabled in HTML or Markdown Editors by updating the `/opt/cyops-ui/vendor/config.json` file using `SSH`. In the `config.json` file, update the value of the `AllowSVGContent` parameter to 'false'. In such a case the SVG will render as "`<\\img src="https:\/\/link.svg">`" in a richtext field, such as the 'Description' or the 'Comment' field.
Important: When you select Rich Text (HTML) or Rich Text (Markdown) for a field that expects data to be populated from external data sources using data or email ingestion, then ensure that the System View Template (SVT) of that field is set to 'IFrame' to prevent any XSS attack attempts or overriding of the parent CSS styling. For information on setting SVTs, see the *Dashboards, Templates, and Widgets* chapter in the "User Guide."
Note: For all modules, the default rich text editor is set to "Markdown", i.e., the Rich Text (Markdown) is selected for rich text fields. You can change the editor from markdown to HTML by selecting the appropriate sub-type for a field and then publish the module for the changes to reflect.

- **Field Title**: A short descriptive name describing the item.
  The**Field API Key** is automatically created based on the value of Field Title field. The field can be alphanumeric and starts with a lower-case alphabet and does not contain any spaces, underscores or any special characters. Note that the value of this field does not get changed once the field has been created and the module has been published. This is because there is no migration path from the old name to the new name, so you risk data loss if this value is changed. Therefore even if you change the Field Title value the value of the Field API Key will not be changed.
  Note: If you have a field, in a module, whose `Field Title` (`Singular Description`) attribute value contains a `.` or `$`, then the Audit Logs replace the `.` or `$` with an `_`. For example, if you have a field SourceID whose singular description you have specified as `Source.ID`, then in this field will appear as `Source_ID` in Audit Logs.
- **Editable**: Selecting this option allows you to modify the field after the creation of a module record. If this option is not selected, then you cannot modify the initial value after the record is created.
- **Delete Associations With Parent**: Selecting this option cascades the deletion of a parent record to all the associated records for the `Many to One` relation type fields. For example, `Events` and `Alerts` have a `Many to One` relationship, i.e., one alert could have many associated events. If you select this option, then if an alert is deleted all its associated events also get deleted. Another example would be the case where one alert has multiple comments, and selecting this option cascade deletes the comments that are associated with a deleted alert.
  Warning: Select this option with caution since RBAC is ***Not*** applied for child records. For example, if you have `Delete` permissions on the `Alerts` module but not on the `Events` module, and you have selected this option, then deleting an alert with associated events, leads to the associated events getting deleted, irrespective of the permissions on the `Events` modules.
- **Searchable**: Selecting this option makes this field searchable in the grid view.

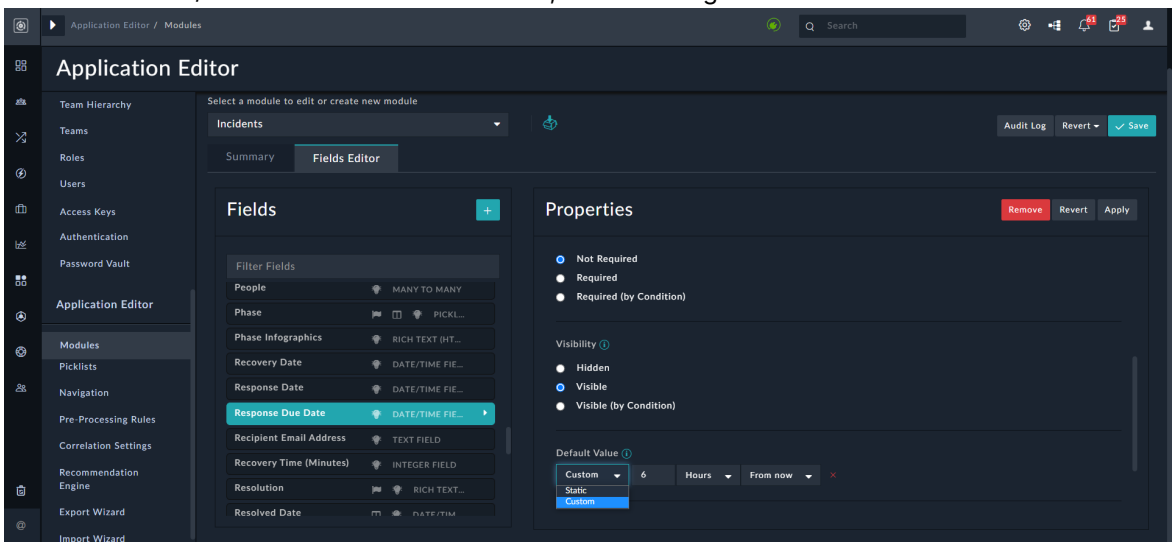
- **Default Grid Column**: Selecting this option makes the field appear as a column by default in the grid view. The order of the grid columns is defined by order of the fields in the Field Editor list. For information about grids, see the *Dashboards, Templates, and Widgets* section in the "User Guide."
- **Enable for recommendation**: Selecting this option enables this field to accept values generated from the recommendation engine. Once you select this option, then an 'Auto populate' checkbox will appear beside this field while configuring data ingestion and also while creating or updating records using playbooks. This option appears for fields of type 'Picklist' and 'Lookup'.
- **Encrypted**: Selecting this option enables encrypting of field values before storing in the database for enhanced security. FortiSOAR UI will continue to display the non-encrypted values. Currently, Text Fields, Email Fields, Rich Text Area and Text Area fields can be encrypted. FortiSOAR uses AES-256 encryption to encrypt and store the values in the database.
  Important: Once you enable encryption you cannot search the field values in FortiSOAR UI. Filters also will not work on encrypted fields. You also cannot use the upsert functionality for fields that are encrypted.
- **Required**: Specifies whether the field is a required field.
  The options are: **Not required**, **Required**, or **Required (by condition)**.
  Once you select **Required (by condition)**, FortiSOAR displays the Condition Builder options where you must add the necessary condition.
  Note: FortiSOAR also supports advanced date operations and nested conditions for the **Required (by condition)** fields i.e., the **Add Condition Group** link is now available for these fields.
  Important: Do not choose the **Visibility = Hidden** option for **Required (by condition)** fields.
- **Visibility**: Specifies whether the field is visible or not.
  The options are: **Hidden**, **Visible** and **Visible (by condition)**.
  If you select the **Hidden** option, then the field is only accessible at the API level and not shown in the UI.
  If you select the **Visible** option, then the field is displayed on the UI. If you select the **Visible (by condition)** option, then the field is displayed on the UI only if the specific conditions are met.
  Once you select **Visible (by condition)**, FortiSOAR displays the Condition Builder options where you must add the necessary condition.
  Note: FortiSOAR also supports advanced date operations nested conditions for the **Visible (by condition)** fields i.e., the **Add Condition Group** link is now available for these fields:



- **Default Value**: Specifies the default value of this field. Once you specify a value in this field, then this value will be displayed, by default when you add a record in the selected module.
  For example, if you want the status of a newly created alert to be set to **Open**, by default, then select the **Status** field and from the **Default Value** drop-down list, select **Open** as shown in the following image:

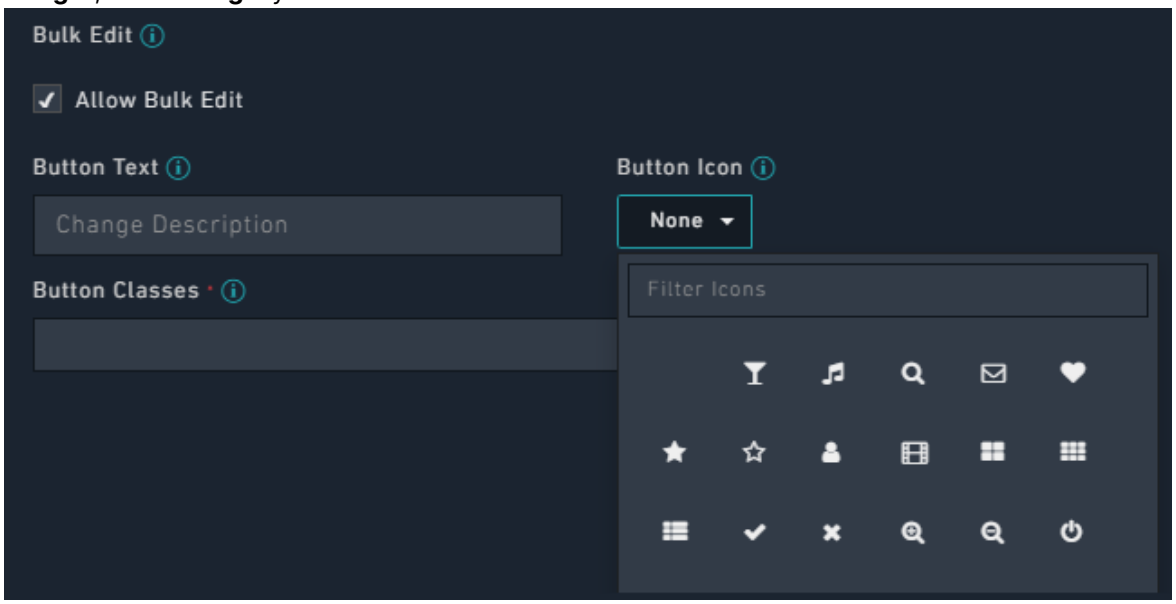


  Once you set the default value, then whenever you add a new record in the `Alerts` module (in our example), then by default **Open** will be displayed in the **Status** field.
  In the **Default Value** field for the `Date/Time` field type you can specify either a Static date/time or a Custom date/time. If you select **Static**, click the **Select Date** icon to display the Calendar and select the required date/time. If you select **Custom**, then you can specify a date/time relative to the current date/time such as 1 hour from now, or 3 hours ago.



  Note: In case you have upgraded to version 5.0.0 or later, then you will have to reselect your datetime default values, since the new datetime format is not backward compatible. You will be able to see the older applied datetime default value in the FortiSOAR fields. However, if you want to edit the default field, then you will have to specify the datetime again in the **Default Value** field.
- **Tooltip**: Brief definitions that you can optionally add to fields. This definition is displayed when you click the information (**i**) icon of the field that has tooltip information added while creating, updating, or viewing records.

- **Length Constraints**: In case of a Text field with sub-type set as Text Field or Text Area, you can specify length constraints by clicking the **Add minimum/maximum range** checkbox, if you want to override the default field length constraints by providing a minimum-maximum range for a field. Once you select the **Add minimum/maximum range** checkbox, you can specify the minimum character length for the field in the **Minimum** field and the maximum character length for the field in the **Maximum** field. You can enter any number from `0` to the maximum character length that is applicable for that field in the database. FortiSOAR will display a validation message if the maximum character length for the field is exceeded, or if the minimum character length for the field is not met.
- **Bulk Edit**: Selecting the **Allow Bulk Edit** option to allow bulk edit operations on the selected field. For example, if you have selected the **Severity** field, in the `Alerts` module, and have clicked `Allow Bulk Edit`, this means the users can select multiple records in the grid view of the `Alerts` module and change the severity of those records to a particular severity level.
  You must enter the following details for the button that you want to use for the bulk edit operation:
  Button Text: In the **Button Text** field, type the name of the label that will be displayed on the bulk action button. For example, type **Change Type**.
  Button Icon: From the **Button Icon** drop-down list, select the icon that will be displayed on the bulk action button. If you do not want an icon to be displayed, select **None**.
  Button Classes: From the **Button Classes** drop-down list, select from the **Default**, **Primary**, **Danger**, or **Warning** styles.

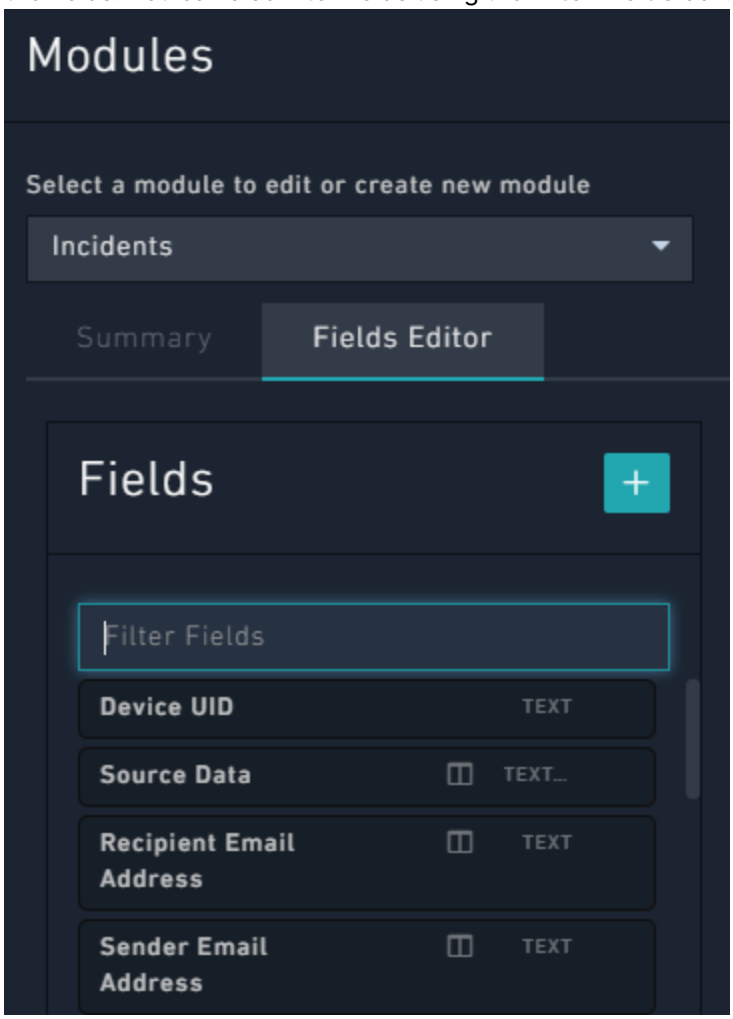


  Once you save the changes and publish the module, a **Change Severity** button is added to the `Alerts` module in the action bar. For more information on how to use the bulk action button, see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide". You can add the bulk edit action button for any other fields, such as **Status**, **Assigned To**, and **Type**.

c. Click **Apply** to add the field or click **Revert** to clear any changes made to the field since the last `Save` event or click **Remove** to remove the field.
For information on the save and revert operations, see the Saving your changes section.

4. You can also define the order of the default grid columns, which is defined by the order of the fields in the `Fields Editor` list. Fields are listed in the **Fields** column and you can drag-and-drop the fields to sequence

the fields. You can also filter fields using the **Filter Fields** box.



5. Click **Save** to save the changes to the module or click **Revert** > **Revert to last saved** to clear any changes made in the interface since the last Save event or click **Revert** > **Revert to last published** to clear any changes made since the last Publish event. For fields, you can revert only to the last published instance. For information on the save and revert operations, see the Saving your changes section.

Once you have completed making modifications to the module, you must publish the modules to reflect the changes in the system. This takes the system down for up to a few minutes while the changes are made. See the Publishing Modules section for more information.

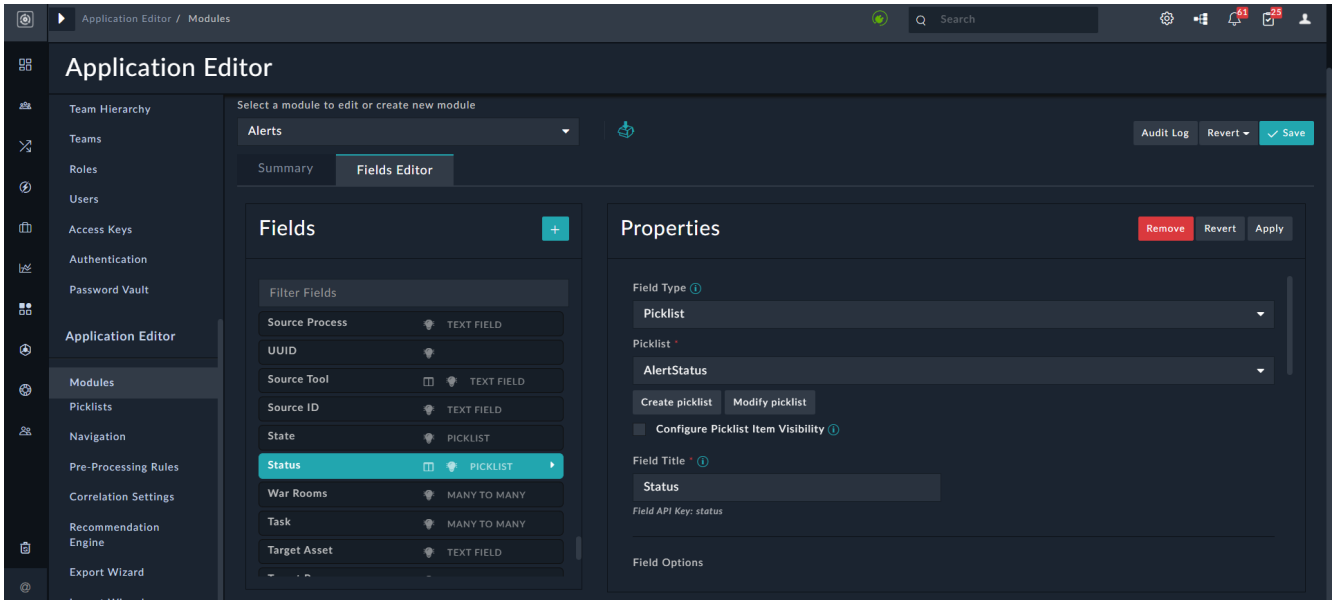
The Module Editor changes the relational database schema, therefore for changes to go live in the environment, you must perform a Publish to the database. This temporarily takes the application offline while the database operations are being performed. All users in the application must save their work prior to this occurring before this occurs or you risk data loss.

Display Template

A module's Display Template refers to one or more fields in the data model itself that is used to display a record in the general interface. Certain widgets, or visualizations in the interface, use the Display Template to identify records to the user. This template specifies the fields that will be displayed when a record from this module is referenced in the application. Fields of a module can be specified in the display template as `{{ <*name of the module's field*>_name}}`. If multiple fields are part of the display template, then you can specify multiple fields as `{{field_name1,field_name2,…}}`.

If you were to use just the name of the module itself, such as Incidents, every Incident record in the interface would include a label named `Incident`. So, users see the Incident label with every record, which is not helpful. Therefore, we use Templates with a language to describe how to label your modules.

You can also include an attribute, such as `itemValue`, of a picklist field in the Display Template, add the following jinja: `{{picklistFieldName.itemValue}}` . For example if you want to include the status of the alert in the Display Template, then the picklist to be used would be `AlertStatus`, and the picklist field name would be `status`:



Therefore you require to add the following jinja: `{{status.itemValue}}` in the Display Template.

Example

We have taken the `Asset` module as an example. Assets represent computing resources, typically on the network. Assets generally have a hostname, IP Address, or MAC Address. We are using the hostname as an example.

To create a **Display Template** for the Assets module with the hostname, you must ensure that you have already added the `hostname` field using the `Field Editor`. Once the hostname is added as a field in the Assets module, use the following expression in the Display Template field:

```
{{ hostname }}
```

The double curly braces, `{{ }}`, is used to identify a variable, specifically a field name. In our case, we are calling the `hostname` field. Any expression in the Display Template field that uses a field from the module data fields must use the double curly braces surrounding the field name. When the record is displayed, the hostname of each asset is used in the interface, so users know the asset they are selecting. For example, a user can know that they are selecting an `HR Server`.

Assets have a unique situation. They might only have one of those pieces of identifiable information depending on what is known about a resource. A laptop on a DHCP network might have only a MAC address. If we set the Asset Display Template to always be a hostname, in many cases the asset might have a blank Display Template in the interface. For these situations, the Angular Template expression allows you the flexibility to modify the format of the Display Template in a way that can account for variation. Taking the asset example, asset information is used to help users identify the asset when in the interface. Because using only a single asset field could potentially lead to a lot of blank Display Templates, we use multiple fields such as hostname, IP address and Mac address:

```
{{ hostname }} {{ ip }} {{ mac }}
```

This Display Template expression instructs the system to use all three fields based on their field names. If a field is not present, it displays as blank. In some cases, depending on what is known about the asset, the Display Template will include all three pieces of information, in others just one.

You can further extend this to display static information that identifies the parts of the Display Template. The following example includes the static text of `Hostname:IP:` and `MAC:` in every Display Template. This might be redundant but is an option.

```
Hostname:{{ hostname }} IP:{{ ip }} MAC:{{ mac }}
```

We recommend that to keep things simple, most of the time you would want to use the following expression for a Display Template, assuming you always create a `name` field for a module:

```
{{ name }}
```

This ensures that the Display Template points to whatever is in the `name` field on any module record. If you create `name` as a required field, then it will always be populated.

Saving your changes

Whenever you make any changes to a module or a field, you must stage those changes by saving. At the top-right of the Module Editor is the **Save** button, which applies any changes made to the staged data. To update the database and make your changes to go live, you must `Publish` the updated modules.

The **Revert** button clears any changes made in the interface since the last `Save` event. If you go into a module and realize that you have edited the wrong field, use **Revert** to clear the changes. However, once you press **Save**, you require to undo the changes manually.

Viewing your changes

Editing any of the fields of a module does not mean those fields are accessible immediately within the UI or the API. The fields must be first represented in the database. The templates included might automatically discover

these fields, or these fields might need to be added manually to the template to specify their location within the interface. However, you can set the grid defaults within the attribute data for the model itself.

To update the database and make your changes to go live, you must `Publish` the updated modules.

Publishing modules

Whenever you change a field or a module and click **Save**, the change is staged but is not yet live in the system. You must perform a `Publish` to ensure that the changes are made in the system.

You initiate a publish action by clicking the **Publish All Modules** button at the top-right of the Module Editor page. Publishing pushes the changes that you have made to fields and modules to the database. Up until the Publish point, all changes to the data model in the Module Editor are saved as metadata, which is information that describes the structure of other information.

A Publish is the point at which the changes are truly irreversible, meaning that an unintended field deletion could cause irretrievable data loss. Use Publish carefully and verify changes before Publishing to avoid any problems.

"Publishing is a sensitive operation"
We recommend that you send a prior notification to all users of a publish since while the publish is in progress users are unable to work. We also recommend reviewing each staged change to ensure that only the desired changes are going to take effect.

If there is any error during the publish operation, FortiSOAR displays a meaningful error message at the top of the module editor, so that it becomes easier for you to resolve issues.

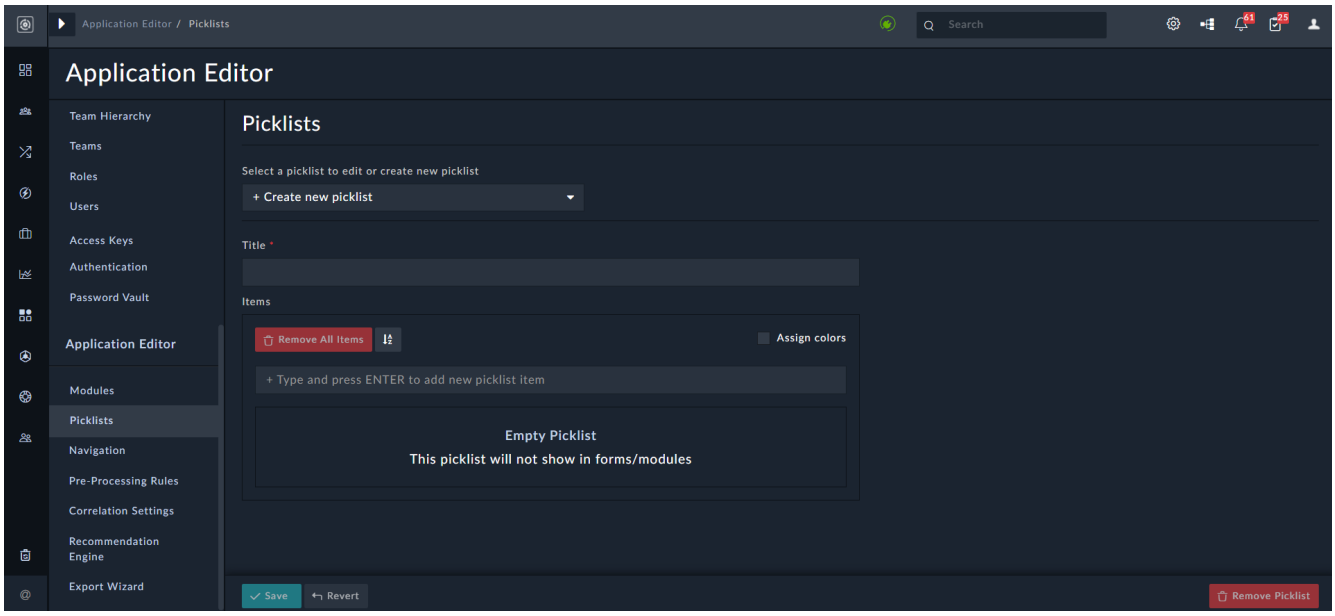
If you have not selected an appropriate field from the **Related Model** drop-down list, for a `Many to One`, or a `Many to Many` field, then the publish operation will display an error.

Picklist Editor

Use the Picklist Editor to change the values of any picklist within the modules and add new picklists that might be referenced by a field in any module.

Unlike the Module Editor, changes made in the Picklist Editor are immediately live once they are saved. This is because Picklists names and Picklist values are records in the database.

A UUID (Universally Unique Identifier) identifies picklist values, which means if you modify any of the names or colors of an existing picklist value, the original data is preserved. Therefore, all records that contain that picklist value retains a reference to the UUID for the picklist. This means that if you want to change an Incident Category of `Theft` to `Physically Stolen`, you could make that change on the existing `Theft` value and any records with the value of `Theft` would now display `Physically Stolen`.

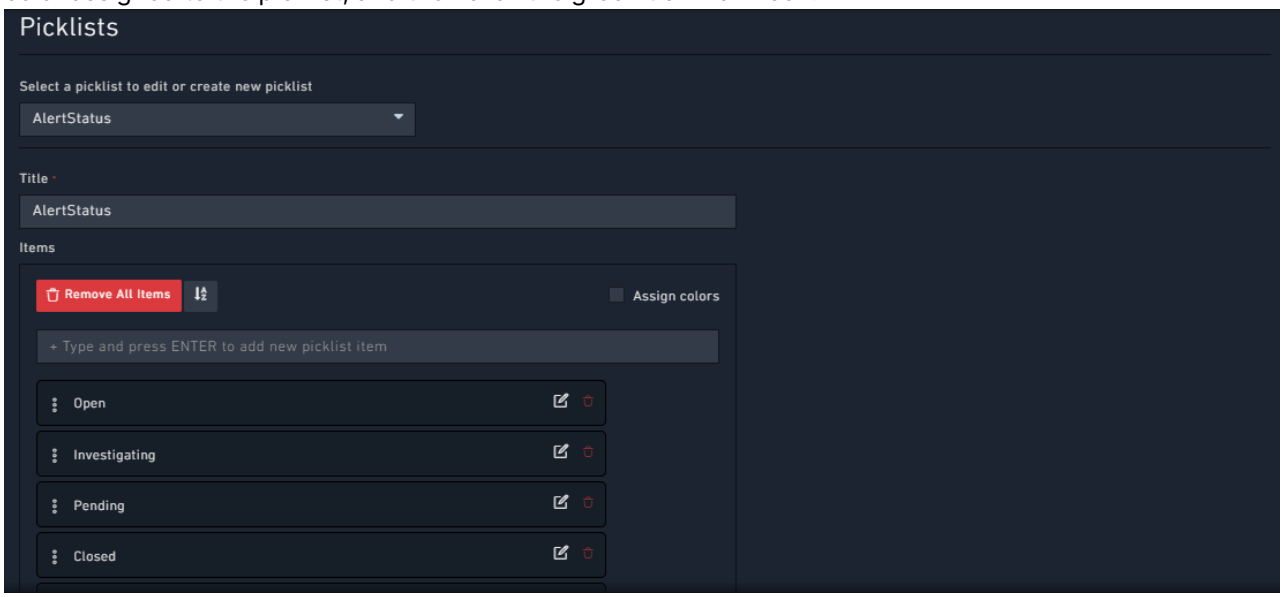


Creating or modifying a picklist

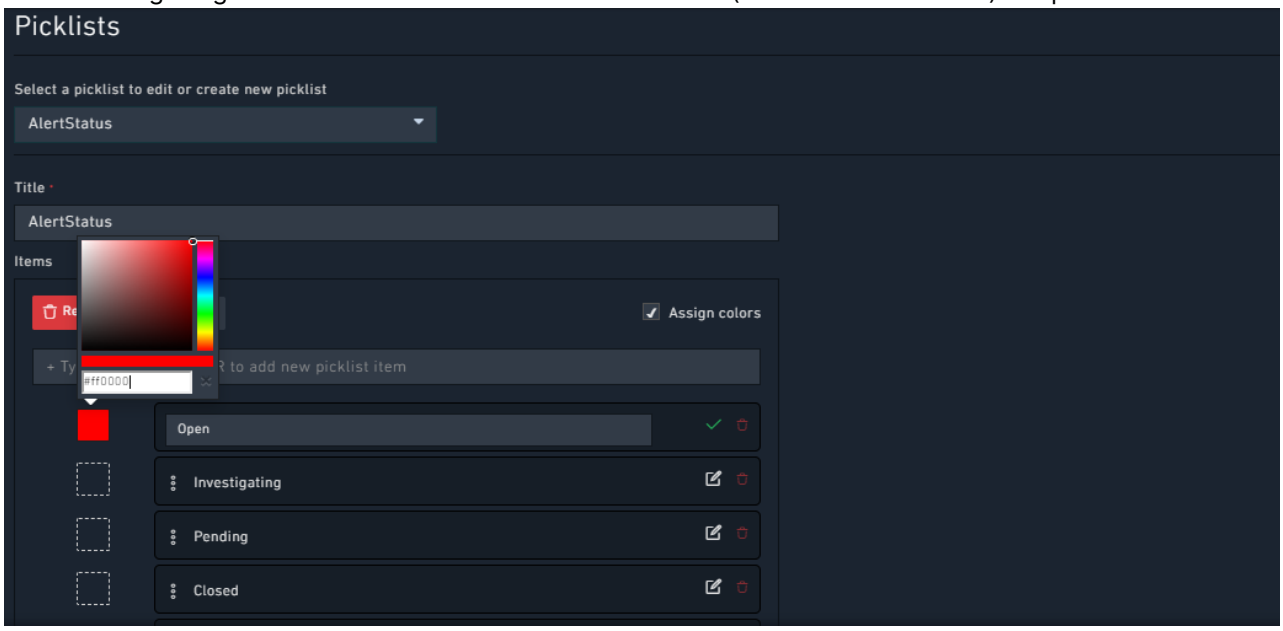
To add or modify a picklist:

1. Click **Settings** and in the `Application Editor` section, click **Picklists**.
   This displays the Picklist Editor.
2. Add or edit an existing picklist. To add a picklist, use the **+Create new picklist** option that appears at the top of the editor to define the properties of the picklist. Start by entering a title for the new picklist in the **Title** field.
   Or To edit an existing picklist, from the **Select a picklist to edit or create new picklist** drop-down list, select the picklist.
   For example, select `AlertStatus`.
3. In the `Items` section, in the **+ Type and Press ENTER To Add New Picklist Item** field, enter the name of the new picklist item and press `Enter`.
   Note: The `{}()/\@:` special characters are not allowed in the picklist item's name.
   For example, for the `AlertStatus` picklist, add items such as **Open**, **False Positive**, and **Verified**.
   Or
   To edit a picklist item, click the **Edit** icon that appears on the item row, update the name of the item or the

color assigned to the picklist, and then click the green tick mark icon.

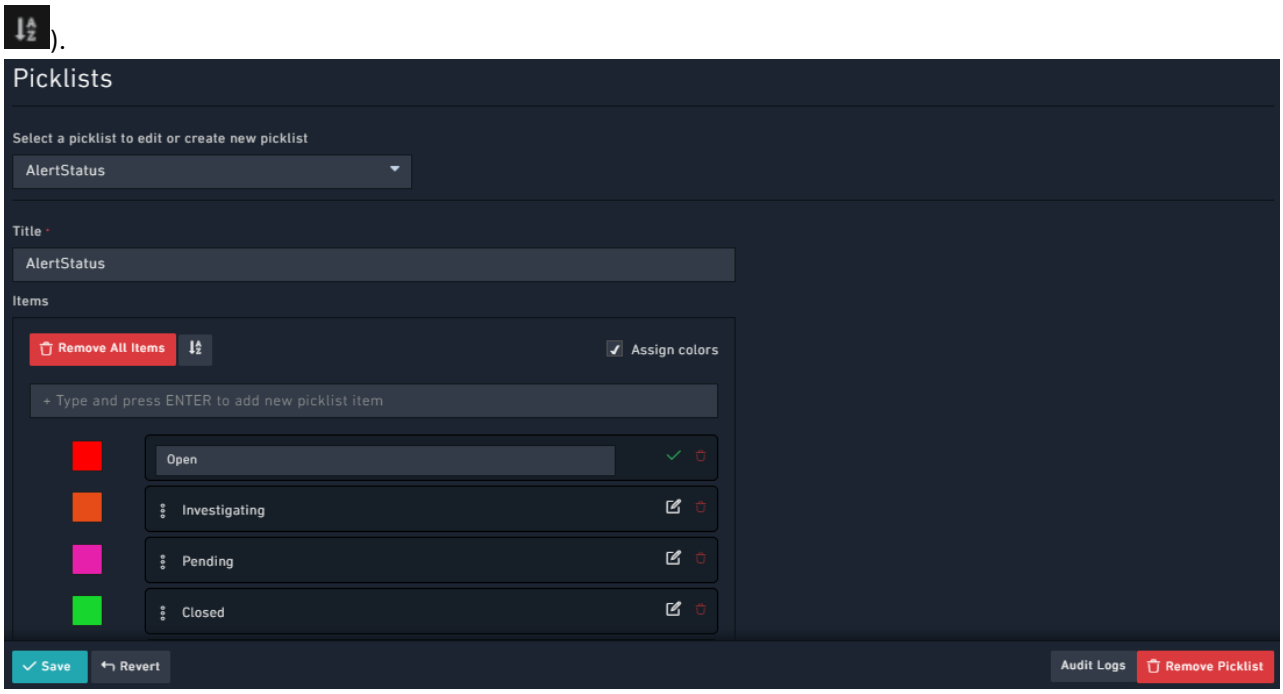


4. (Optional) You can add colors to any picklist, by checking the **Assign Colors** checkbox. Once the Assign colors checkbox is enabled, you can assign each item in the picklist a color.
   Use the color picker box that appears next to each item in the picklist or enter the hexadecimal code for the color to edit the colors. You can use any valid HTML color.
   You can set the picklist color by directly entering the hexadecimal code of the color and assigning that as the picklist color, or by using an API, or you could choose colors by clicking in the color picker.
   The following image shows how to enter a hexadecimal code (`#ff0000` for red color) in a picklist:



   Note: Multiple items in a picklist can have the same color.

5. (Optional) You can also remove all items from the picklist by clicking the **Removing All Items** button, and you can also change the sort order of the picklist items clicking the **Ascending or Descending order** icon (



6. Click **Save** to save the changes made to the picklist or click **Revert** to clear any changes made to the picklist since the last Save event or click **Remove Picklist** to remove the picklist.
   You can also click the **Audit Log** button to view logs specific to a particular picklist. For more information on Audit Logs, see the System Configuration chapter.

Pre-Processing Rules

Playbooks were previously used to detect duplicate records; however, the lack of precision in playbook design often leads to an influx of redundant alerts in a SOAR system, resulting in an abundance of records and increased workflow load. To address this issue, FortiSOAR has implemented a rule-based pre-processing feature that gets triggered before an incoming record is stored in the database, providing the flexibility to make decisions such as dropping records based on predefined criteria.

Additionally, the implementation of a post-processing rule adds efficiency to record management, by linking similar records based on specified similarity criteria. This post-processing rule facilitates intelligent linking of records, reduces reliance on resource-intensive playbooks and optimizes system performance. In summary, these rule-based pre- and post-processing features enhance the control and efficiency of the SOAR platform. For information on the API for pre-processing rules, see the *API Methods* chapter in the "API Guide." Starting from release 7.6.0, a new pre-processing rule called "Enforcing File Attachments for File Indicators" is included by default. This rule ensures that indicators of type 'file' are only created when a file is attached.

Some advantages of adding pre-processing rules include:

- Ability to configure different types of rules for identifying duplicate records
- Avoid the need to create custom playbooks for removing duplicate records.
- Avoid the execution of playbooks for records with similar attributes.

To view the 'Pre-Processing Rules' page, you must be assigned `Read` permission on the `Preprocessing Rules` module and `Update` permission on the `Application` module. Similarly, to perform actions such as adding new pre-processing rules, editing these rules, changing their status, etc, you must be assigned appropriate `CRUD` permissions on the `Preprocessing Rules` module.

Processing of rules

- Pre-processing of rules does not apply to bulk operations.
- If multiple rules are matched for a record, the processing occurs as follows:
  - Once a single rule is matched for a record, the processing stops and exits; none of the other rules are evaluated.
  - Rules are sorted first by their 'Priority', with the highest priority rules evaluated first, starting with P1 as the highest and P5 as the lowest, and then by their 'Created' date.

You can export and import your pre-processing rules using the Export and Import Wizards.

Adding a new 'Drop' type pre-processing rule

Users can add pre-processing rules to act as a gatekeeper that help FortiSOAR detect duplicate records getting ingested into the system, thereby filtering incoming records and preventing the unnecessary accumulation of redundant alerts. The pre-processing rule can also prevent execution of playbooks for the same records, leading to a reduction of the load on the system. For example, users can define a pre-processing rule that prevents records associated with internal security campaigns from being created, offering a preemptive measure to prevent unnecessary entries into the SOAR platform.

You can choose to 'Drop' or 'Update' incoming records based on the defined pre-processing rule.

In MSSP environments, replicated records are not evaluated by the 'Drop' rule. However, update rules are still evaluated on replicated records. To prevent this, add the '`tenant=self`' condition to each update rule on both the master and tenant nodes.

To add a pre-processing rule that drops incoming records associated with internal security campaigns or those that have a specific source IP address, follow these steps:

1. Click **Settings** and in the `Application Editor` section, click **Pre-Processing Rules**.
2. On the `Pre-Processing Rules` page, click **+ Add Pre-Processing Rule** to display the `Create New Pre-processing Rule` wizard and define the rule:
   a. In the `Rule Details` dialog, enter details for the rule:
      i. Toggle the **Active** button to set the state of the rule as 'Active' (default) or 'Inactive'.
      ii. In the **Rule Name** field, enter a name to identify the rule.
      NOTE: Rule names must be unique within the system.

iii. (Optional) In the **Description** field, enter a brief description of the rule.

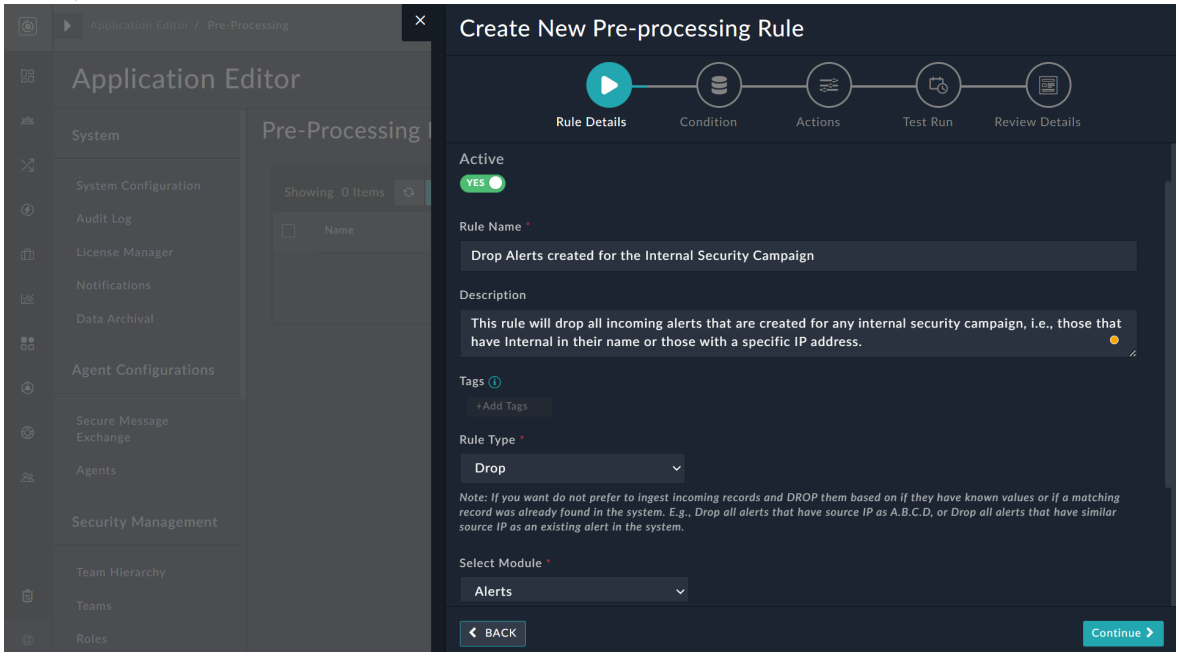
iv. (Optional) In the **Tags** field, add keywords that you can use to reference the rule.

v. From the **Rule Type** drop-down list, select the action that requires to be performed on similar incoming records. You can choose between **Drop** or **Update**. For our example, select **Drop**.

vi. From the **Module** drop-down list, select the module on whose records you want to run the pre-processing rule. For our example, select '**Alerts**'.
NOTE: System modules such as 'People', 'Appliances', 'Agent', etc., will not be included in the **Module** drop-down list as rules cannot be configured for these modules.

vii. From the **Priority** drop-down list, select the priority of the rule, which determines the order of rule execution.

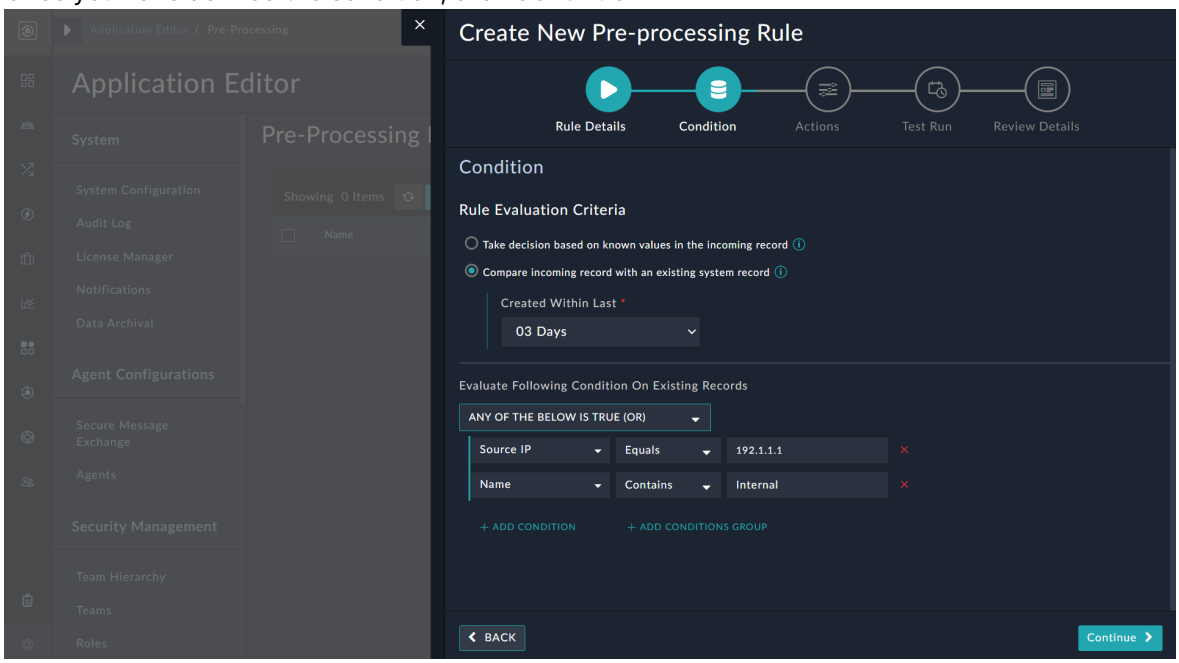
viii. (Optional) In the **Set Rule Expiry Date** field, select the date when this rule will expire and no longer be used to detect duplicate records being ingested into the system. For example, if you have created an internal security campaign for a month, you can set the expiration date to be the same.
Once you have entered all the details, click **Continue**.



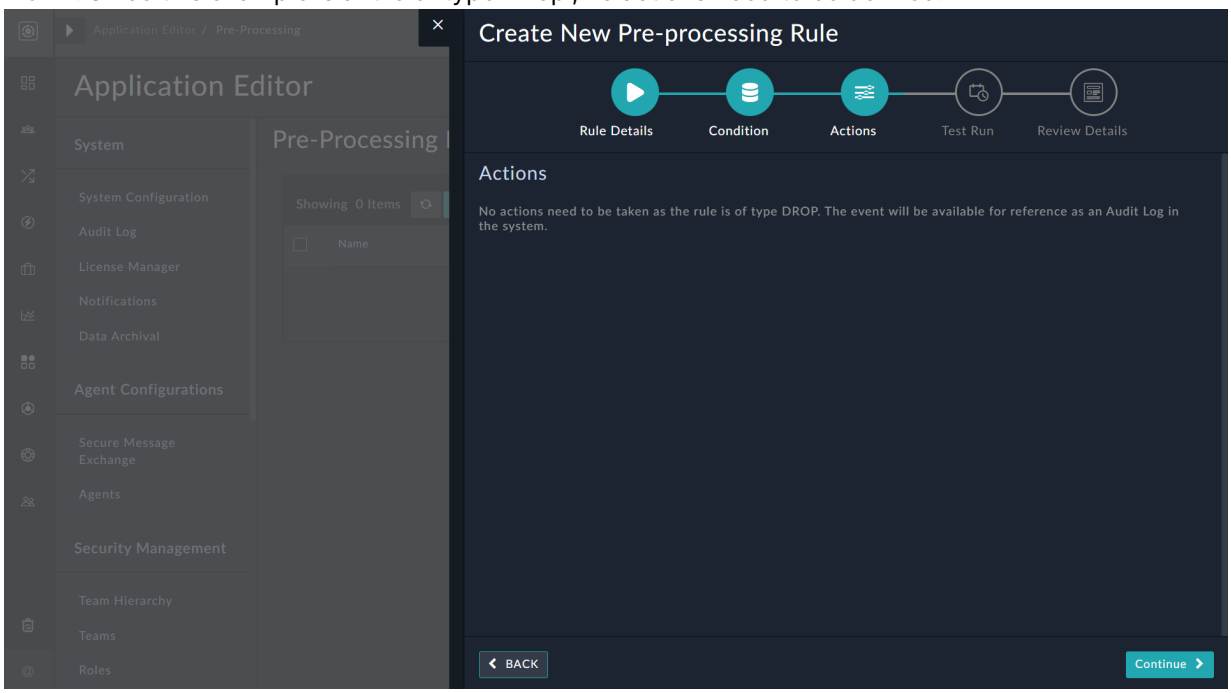
b. In the `Condition` dialog, specify the condition to filter incoming records.

i. If you have selected the 'Drop' action, you can evaluate the criteria based on the following conditions:
Select the **Take decision based on known values in the incoming record** option to take the decision to drop the incoming record based on the values of the record that is being ingested into the system.
OR
Select the **Compare incoming record with an existing system record** option to take the decision to drop the incoming record after comparing its values with existing records in the system.
If you select this option, then from the **Created Within Last** drop-down list, select the number of days within which the existing records were created and that need to be compared with the incoming record. You can choose between 1 to 7 days.
For example, if you choose **03 days**, the incoming record's values will be compared with the values of records created in the system in the last 3 days.

ii. In the `Evaluate Following Condition On Existing Records` section, define the condition for comparing the values of the incoming records. For our example, define the condition as `Source IP Equals 192.1.1.1` OR `Name Contains Internal`.
NOTE: Negative operators such as 'Not Equals', 'Does Not Contain', 'Does Not Match', 'Is Not In List', etc. cannot be selected in the conditions defined for evaluating existing records in the case of the 'Drop' rule type. This is due to the possibility that numerous records could be dropped when using negative operators.
Also note that if you have selected the **Compare incoming record with an existing system record** option, you can select the 'Already Exists' operator to compare the values of incoming records with existing records for condition evaluation.
Once you have defined the condition, click **Continue**.



c. On the `Actions` dialog, click **Continue**.
NOTE: Since this example is a rule of type 'Drop', no actions need to be defined.

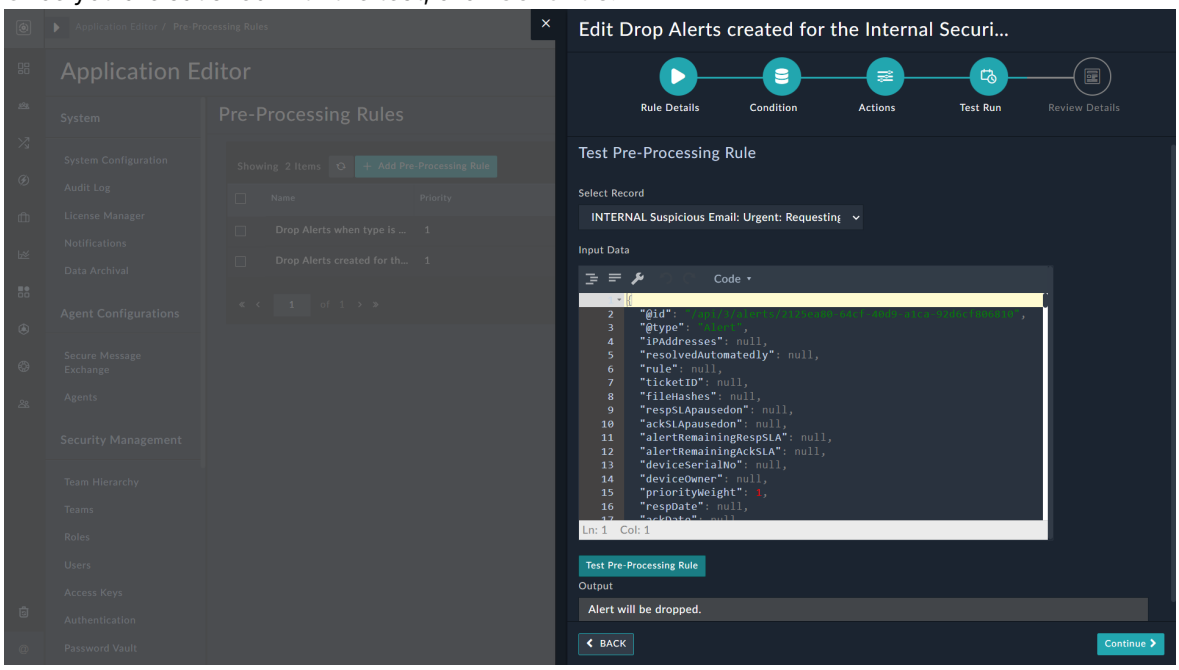


d. On the `Test Run` dialog, test the created pre-processing rule.
IMPORTANT: In case you choose the '**Already Exists**' operator while specifying the condition for the 'Drop' rule, the test result will consistently show as 'Not Dropped'. This is because no database query is made by the test functionality to evaluate the 'Already Exists' condition.

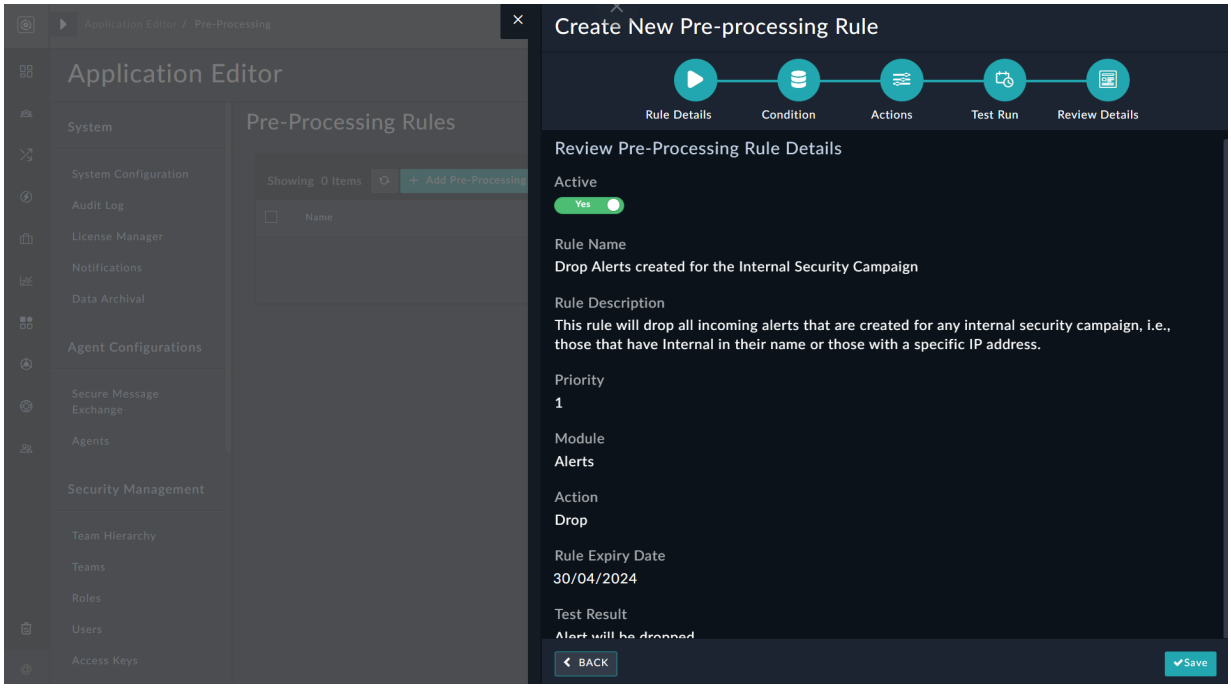
i. From the **Select Record** drop-down list, select a record that will act as the incoming record.
Once a record is selected, the JSON format of that record will populate the **Input Data** field.

ii. Click the **Test Pre-Processing Rule** button to evaluate this record according to the defined pre-process rule.
The **Output** field will display the result of the evaluation.
Since we have selected a record that contains internal in its name, the output displays the result as "Alert will be dropped".

Once you are satisfied with the test, click **Continue**.



e. On the `Review Details` dialog, review the details of the rule. If your are satisfied, click **Save** to save this rule.

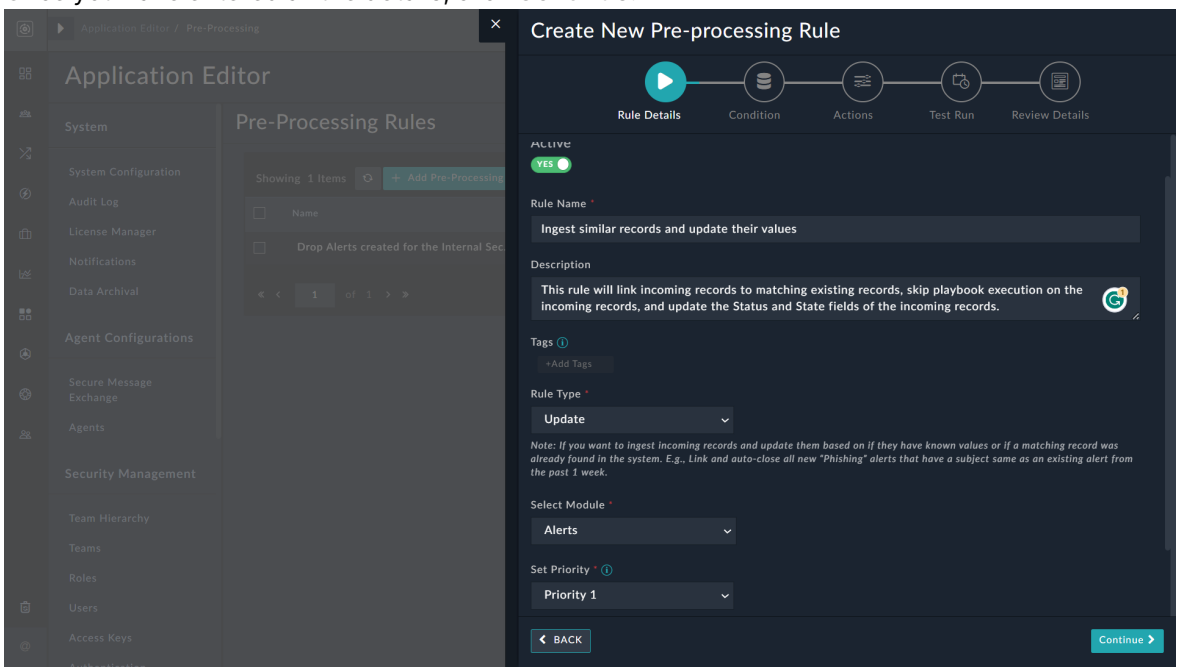


Adding a new 'Update' type pre-processing rule

To add a pre-processing rule that ingests incoming records and updates them based on the defined rule, such as linking incoming records to matching existing records, skipping playbook execution on the incoming records,

and updating the fields of the incoming records, such as updating the State of the incoming record to 'Similar Alerts Correlated' and marking its Status as 'Closed', follow these steps:

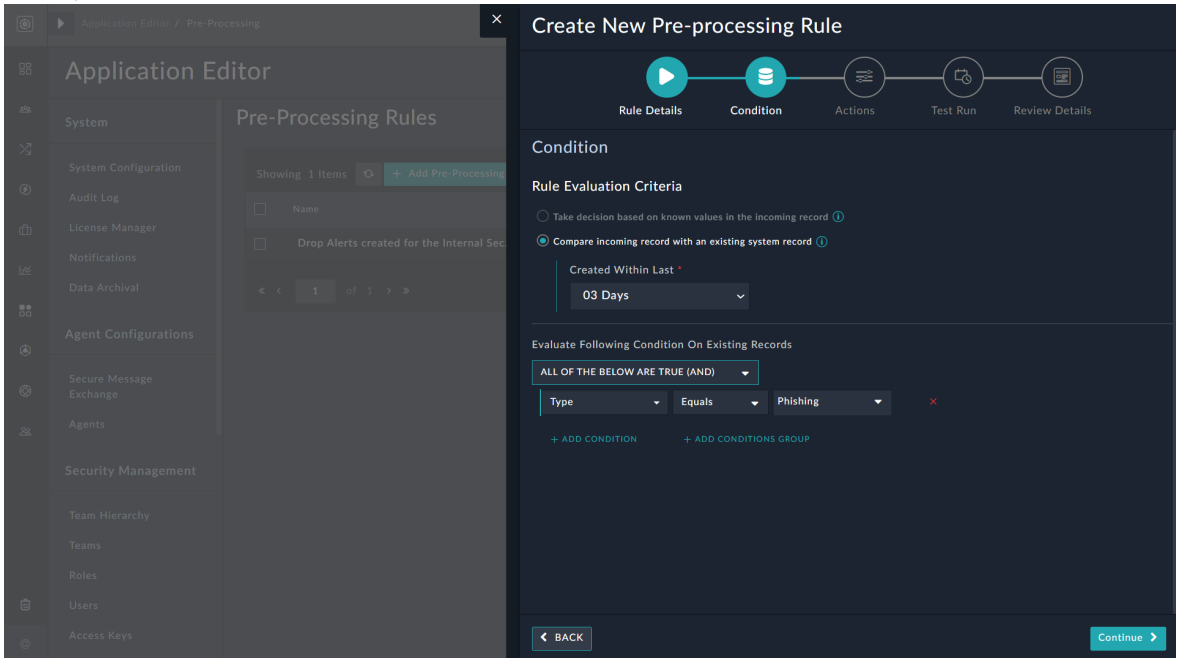
1. Click **Settings** and in the `Application Editor` section, click **Pre-Processing Rules**.
2. On the `Pre-Processing Rules` page, click **+ Add Pre-Processing Rule** to display the `Create New Pre-processing Rule` wizard and define the rule:
   a. In the `Rule Details` dialog, enter details for the rule:
      i. Toggle the **Active** button to set the state of the rule as 'Active' (default) or 'Inactive'.
      ii. In the **Rule Name** field, enter a name to identify the rule.
      NOTE: Rule names must be unique within the system.
      iii. (Optional) In the **Description** field, enter a brief description of the rule.
      iv. (Optional) In the **Tags** field, add keywords that you can use to reference the rule.
      v. From the **Rule Type** drop-down list, select the action that requires to be performed on similar incoming records. You can choose between **Drop** or **Update**. For our example, select **Update**.
      vi. From the **Module** drop-down list, select the module on whose records you want to run the pre-processing rule. For our example, select '**Alerts**'.
      NOTE: System modules such as 'People', 'Appliances', 'Agent', etc., will not be included in the **Module** drop-down list as rules cannot be configured for these modules.
      vii. From the **Priority** drop-down list, select the priority of the rule, which determines the order of rule execution.
      viii. (Optional) In the **Set Rule Expiry Date** field, select the date when this rule will expire and no longer be used to detect duplicate records being ingested.
      Once you have entered all the details, click **Continue**.



   b. In the `Condition` dialog, specify the condition to filter incoming records.
      i. If you have selected the 'Update' action, then you can filter incoming records using the **Compare incoming record with an existing system record** condition. The decision to update the incoming record is taken after comparing its values with existing records in the system. Additionally, from the **Created Within Last** drop-down list, you must select the number of days within which the existing records were created and that need to be compared with the incoming record. You can

choose between 1 to 7 days. For example, if you choose **03 days**, the incoming record's values will be compared with the values of records created in the system in the last 3 days.

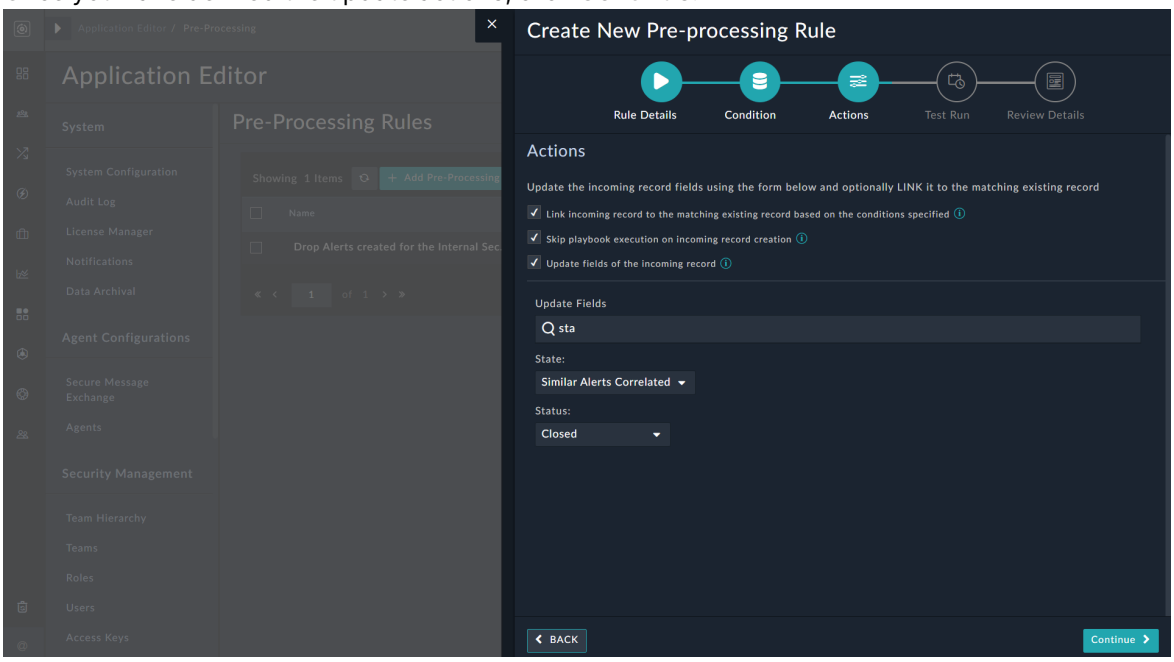
ii. In the `Evaluate Following Condition On Existing Records` section, define the condition for comparing the values of the incoming records. For our example, define the condition as `Type Equals Phishing`.
Once you have defined the condition, click **Continue**.



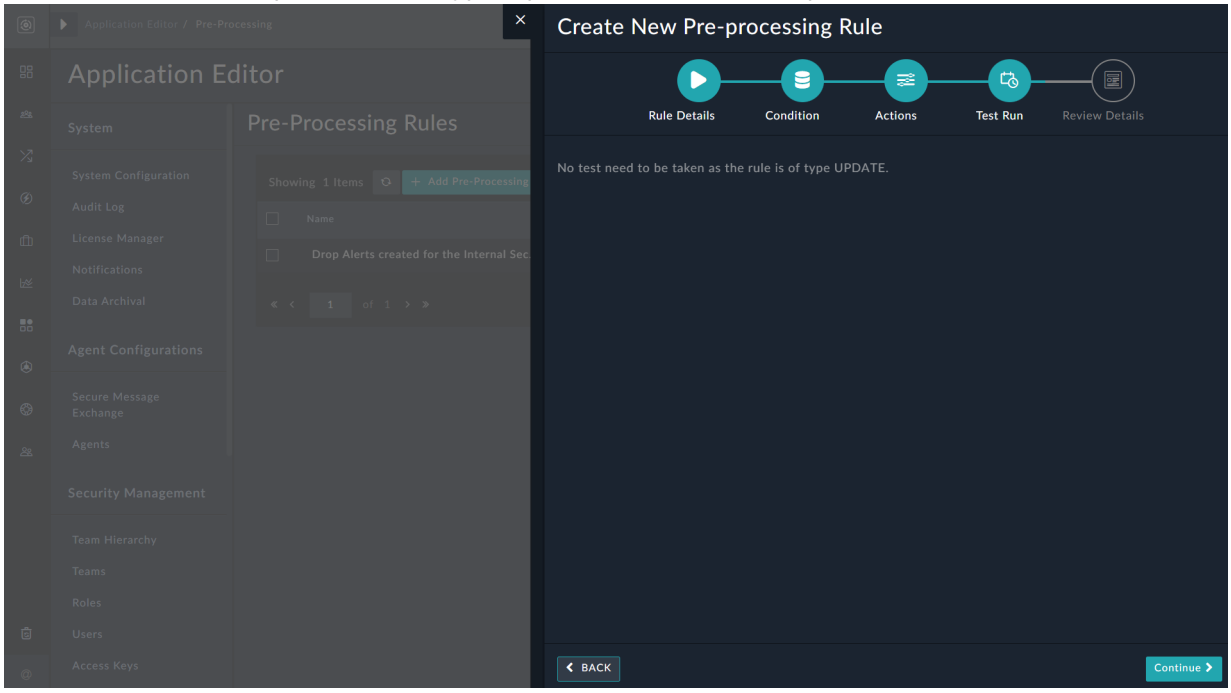
c. On the `Actions` dialog, select how you want to update similar incoming records with the following options:

- Select the **Link incoming record to the matching existing record based on the conditions specified** option to link the similar incoming record to the matching existing records.
NOTE: This type of linking will work only on modules that support many-to-many relationships on fields within the same module. For example, it is possible to link an alert to another alert or an incident to another incident; however, it is not possible to link a people record to another people record.
- Select the **Skip playbook execution on incoming record creation** option to prevent playbooks, such as the 'On Create' playbooks from being executed on the incoming record.
- Select the **Update fields of the incoming record** option to display a list of the primary fields of the selected module and update the values of the incoming records fields based on the specified values.
You can use the **Update Fields** search box to search through the list of fields.
For our example, update the **State** field to **Similar Alerts Correlated** and the **Status** field to **Closed**.

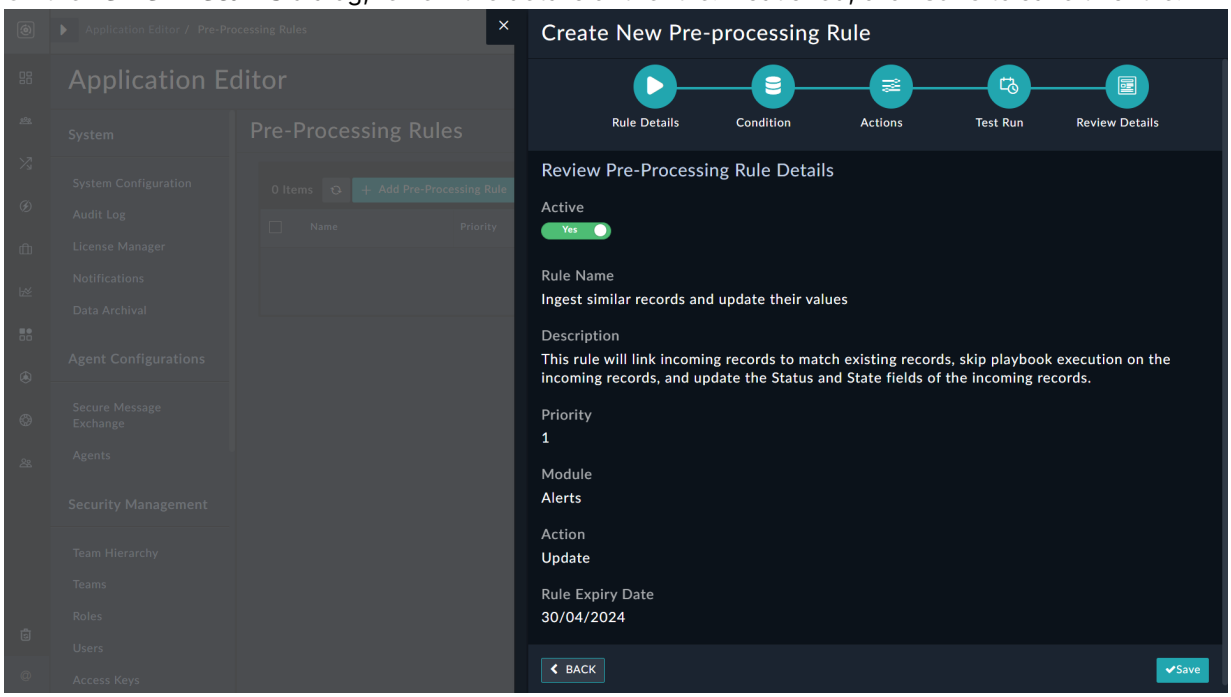
Once you have defined the update actions, click **Continue**.



d. On the `Test Run` dialog, click **Continue**.
NOTE: Since this example is a rule of type 'Update', no test run is required.



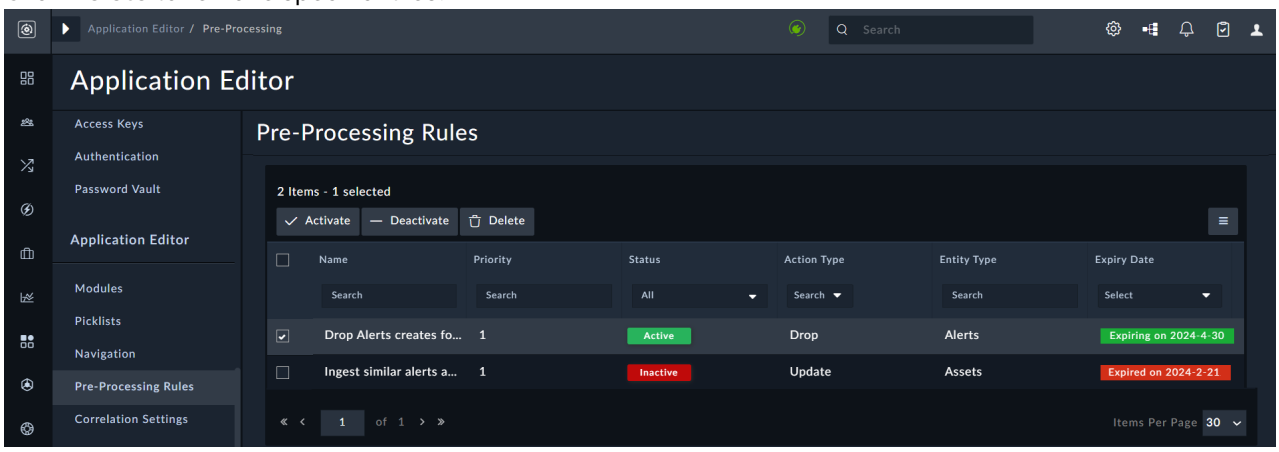
e. On the `Review Details` dialog, review the details of the rule. If satisfied, click **Save** to save this rule.



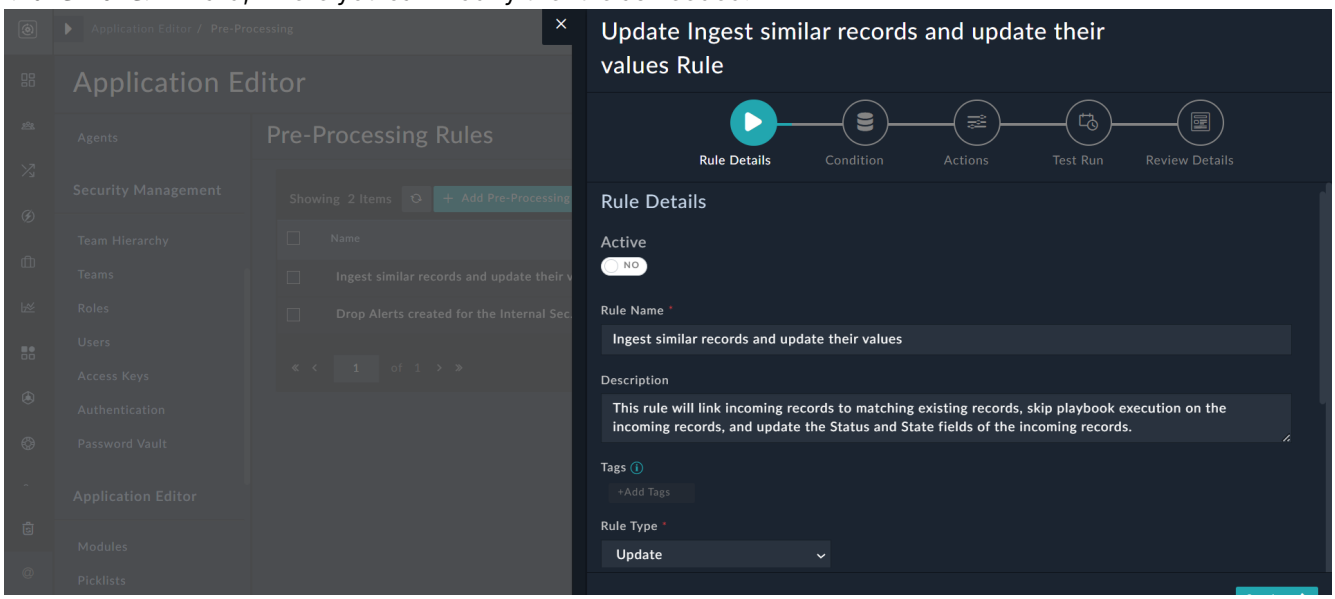
Supported Operations on the Pre-Processing Rules Page

Click **Settings** > **Pre-Processing Rules** in the `Application Editor` section to open the `Pre-Processing Rules` page. On the `Pre-Processing Rules` page, you can search for rules by name, entity type, etc, and filter them by priority, status, action type, etc. Additionally, you can select the rules to perform the following operations:

- Click **Activate** to make specific rules 'Active'
- Click **Deactivate** to make specific rules 'Inactive'
- Click **Delete** to remove specific rules.



To edit a specific pre-processing rule, click its row on the `Pre-Processing Rules` page. This opens the `Update <Rule Name>` wizard, where you can modify the rule as needed:

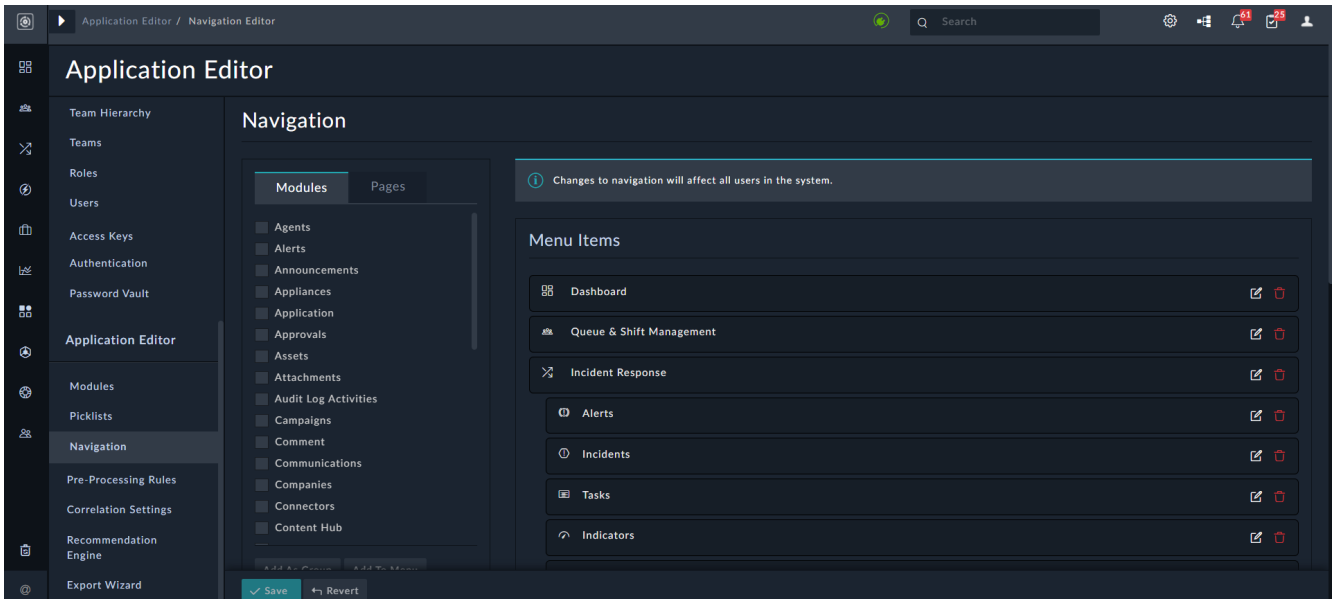


Navigation Editor

Pages are iFramed resources that are accessible from the application interface by the user, such as resource pages and wikis within the local environment or on an accessible website link. Pages must currently be added in the `modules` API to be present to add in the Editor.

Use the Navigation Editor to modify the system Navigation bar, present on the left-side of the application interface.

Changes that you make to the left navigation bar using the Navigation Editor affects all users. Currently, these changes cannot be made at a user-specific level.



There are two types of Navigation values:

- Single-level navigation item, in which case an icon and title on the Navigation bar represent a module or page
- Two-level navigation item, in which case an icon and title reveal a menu of additional options. Secondary navigation items might only have a name, not an icon.

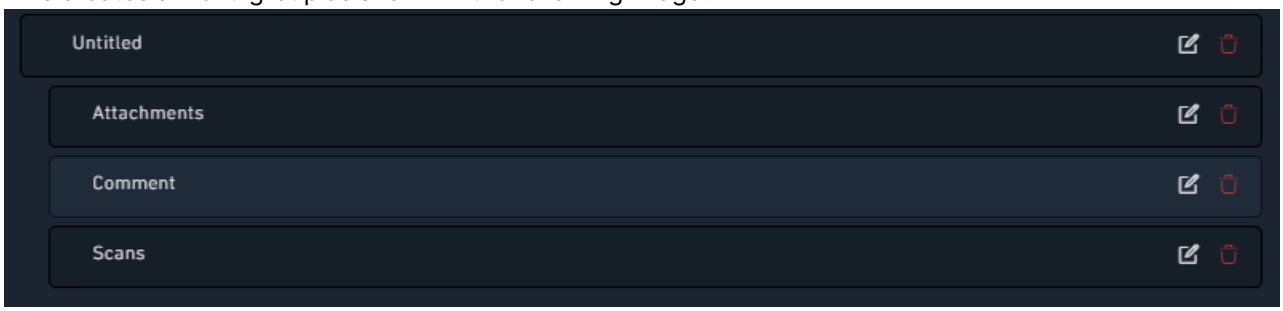
You can add an external HTML page in an iFrame or a new tab and display that page as part of the left-navigation in FortiSOAR.

Modifying the Navigation bar

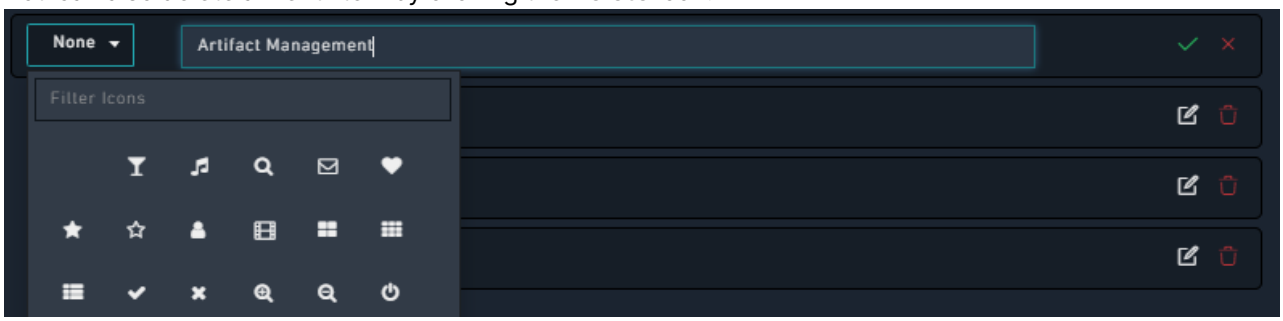
To modify the Navigation bar:

1. Click **Settings** and in the `Application Editor` section, click **Navigation**. This displays the **Navigation Editor**.
2. Add or modify the navigation bar:
   To add a single-level item, select module or pages by clicking the **Modules** or **Pages** tab, and click **Add To Menu**. Single level items on the menu must represent a 1:1 relationship with a module or page.
   To add a two-level item, select modules or pages by clicking the **Modules** or **Pages** button, and click **Add As Group**.
   The second-level navigation item is not a hyperlink or capable of referencing a given module or page. Only the sub-items in the group can be linked as a module or page. Clicking any two-level Navigation group shows and hides the sub-items.
   For example, you want to create a menu-group named Artifacts Management that has Attachments, Comment, and Scans as the menu items. You select the Attachments, Comment, and Scans modules and click **Add As Group**.

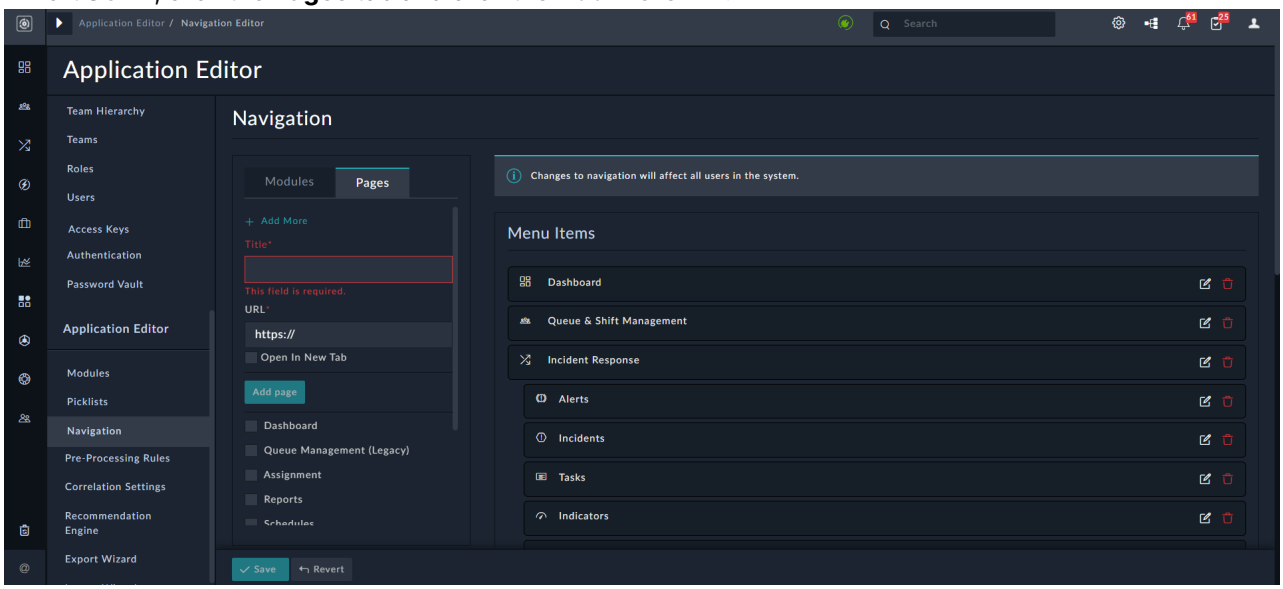
This creates a menu group as shown in the following image:



3. Edit the menu items by clicking the **Edit** icon that appears on the item row, update the name of the menu item or replace the icon of the first-level item in menu group, and click the green tick mark icon.
   You can replace icons by choosing icons from the icon selector at the left of each Navigation item.
   You can also delete a menu item by clicking the **Delete** icon.



4. Drag-and-drop modules or module groups to change the order of the navigation items in the Navigation bar.
   Note: The top item of the navigation is always the default login page. By default, this is the dashboard page. However, you can modify this to make any other page the home page.
5. To add an external HTML page in an iFrame or a new tab and display that page as part of the left-navigation in FortiSOAR, click the **Pages** tab and click the **Add More** link.



   a. In the **Title** field, enter the name for the HTML page that you would want to display in the left navigation menu.
      For example, if you want to add a link to the Google website as part of your left-navigation in

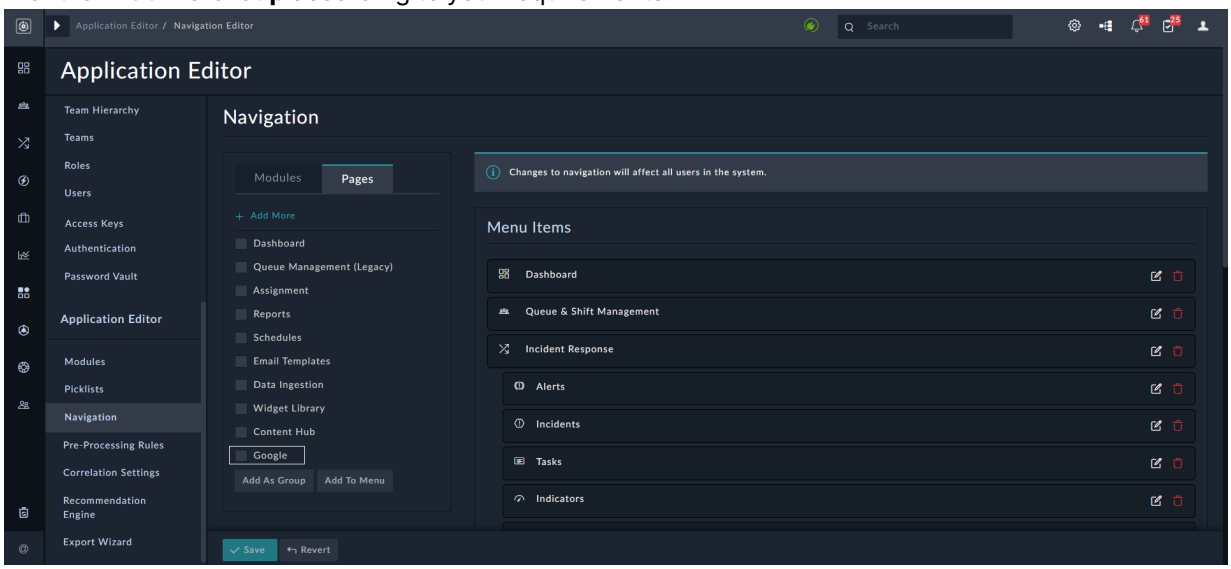
FortiSOAR, enter `Google` in the title field.

b. In the **URL** field, enter the URL for the HTML page that you want to display in an iFrame or new tab. For our example, enter `https://www.google.com`.

c. (Optional) If you want to open the page in a new tab, click the **Open in New Tab** checkbox. If the **Open in New Tab** checkbox is unchecked (default) the page will open in an iFrame in FortiSOAR.

d. Click **Add Page**.

e. On the **Pages** tab, select the page you have just added, **Google** in our example, and then click **Add To Menu** or **Add As Group** according to your requirements.



6. Click **Save** to save the changes made to the menu items or click **Revert** to clear any changes made to the menu items since the last Save event.

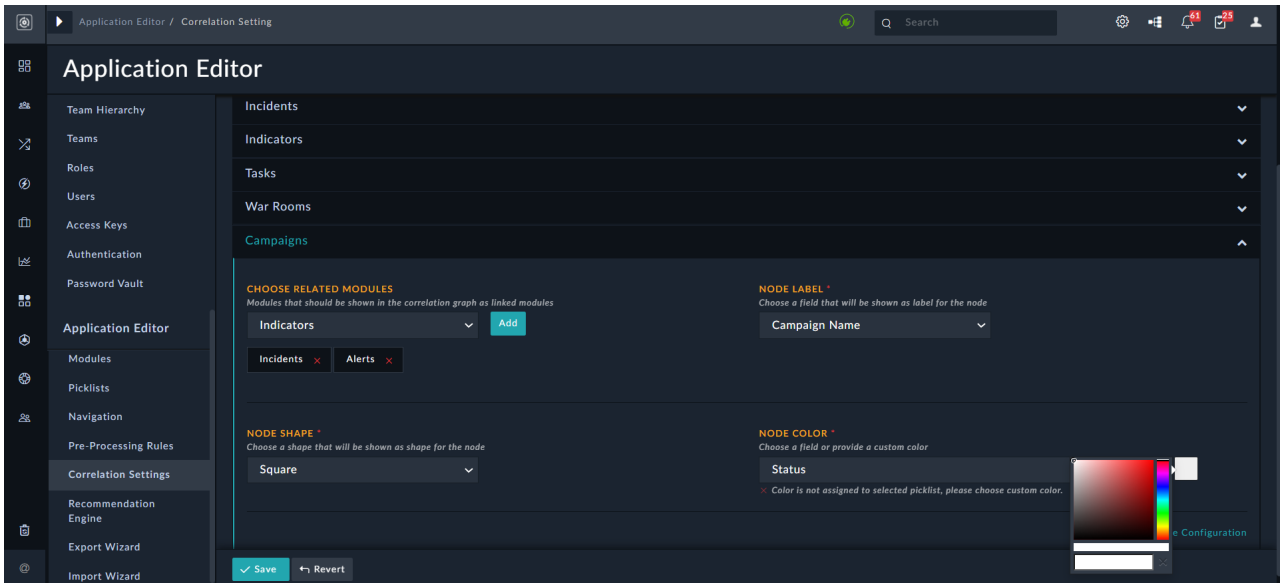
Correlation Settings

If you want to use the Visual Correlation widget to visually display the nodes related to a particular record, then you have to configure the display of the various related nodes on the `Visual Correlation Setting` page.

The following procedure is an example where you are configuring the display of tasks that have associated records.

1. Click **Settings** and in the `Application Editor` section, click **Correlation Settings**.
2. On the `Visual Correlation Setting` page, from the **Choose Modules To Define Correlation View Configurations** drop-down list, select the module for which you want to define visual correlations and then click **Add and Configure**.
   Note: FortiSOAR has pre-configured modules such as, Alerts, Assets, Incidents, Indicators, War Rooms, etc. The default depth of the nodes displayed is "3", i.e., if you start from the Alerts node, you view its related indicators and if those indicators have related assets, you can view those related assets. However, if that asset also has a related incident, then you can double-click on that asset node and see its related incident and so on.
3. To configure the `Campaigns` module, from the **Choose Modules To Define Correlation View Configurations** drop-down list, select **Campaigns** and click **Add and Configure**.

4. From the **Choose Related Modules** drop-down list, select the modules that should be shown in the correlation graph as linked modules, and then click **Add Related Modules**.
   Note: The modules that are pre-configured already have related modules configured, for example, the Alert module has Alerts, Incidents, Indicators, Vulnerabilities, and Assets configured as related modules.
   For our example, we require to add `Alerts`, `Assets`, and `Incidents`, as related modules to the `Tasks` module.
   Note: If you add a module that has not been configured for correlation, for example, the `Comments` module, then you will require to configure that module also before correlations can work. For example, if you choose the `Comments` module, then you must also configure the Node Label, Node Shape, and Node Color for that module.
5. From the **Node Label** drop-down list, select the field that will be shown as a label for the node.
   For our example, choose **Campaign Name**.
6. From the **Node Shape** drop-down list, select a shape that will be shown as the shape of the node.
   For our example, choose **Square**.
   You can also choose to specify a custom icon as the shape of the node. In this case choose **Custom Icon** from the **Node Shape** drop-down list and click **Upload** to display the `Upload an Image` dialog. In the `Upload an Image` dialog, drag-and-drop the icon file, or click the **Import** icon and browse to the icon file to import the icon file into FortiSOAR and then click **Save Image** to add the custom icon.
   You can also change the background shape of the custom icon by clicking the shapes present under **Background shape of custom icon**.
   You can also change the custom icon by clicking **Change**, which again displays the `Upload an Image` dialog.
   Note: The custom icon should be 15px X 15px and the file size must be less than 10KB.
7. From the **Node Color** drop-down list, select the field that will conditionally determine the color of the node.
   For the pre-configured modules, such as alert, this field is set by default. For example, for the `Alerts` module, this is set as **Severity**. For the `Campaign` module, select `Status`.
   Note: In case of picklists the color of the node is determined by the color that you have assigned to the value of the picklist item. For example, if you have chosen the **Priority** as the picklist, then the colors that you have assigned to the selected picklist value will be used as the node color. For example, if the Priority is set as Urgent, then the node color will appear as Red, if its High, then the node color will appear as Orange, if its Medium, then the node color will appear as Yellow, etc. Therefore, if the task in which you have added the Visual Correlation widget is Critical, then its node color will be Red. For information on how to add the Visual Correlation widget in the records, see the *Dashboards, Templates, and Widgets* chapter in the "User Guide."
   You can also assign a custom color for the node by using the **Choose custom color** picker.



If you do not specify any color, the node will appear with its default color.

8. Next, you require to define how the related record node will also be displayed. For our example, we have chosen the `Alerts` and `Incidents` modules, as related modules all of which are pre-configured modules, so we do not require to configure any module.
9. Click **Save** to save the settings for visual correlation.

Now, you can add the Visual Correlation Widget in the detail view of the `Campaign` records (see the *Dashboards, Templates, and Widgets* chapter in the "User Guide") to view the Correlations graph. The following image displays the "Visual Correlations" graph, which is an example of a Visual Correlation Widget being added to the detail view of `Tasks` records:

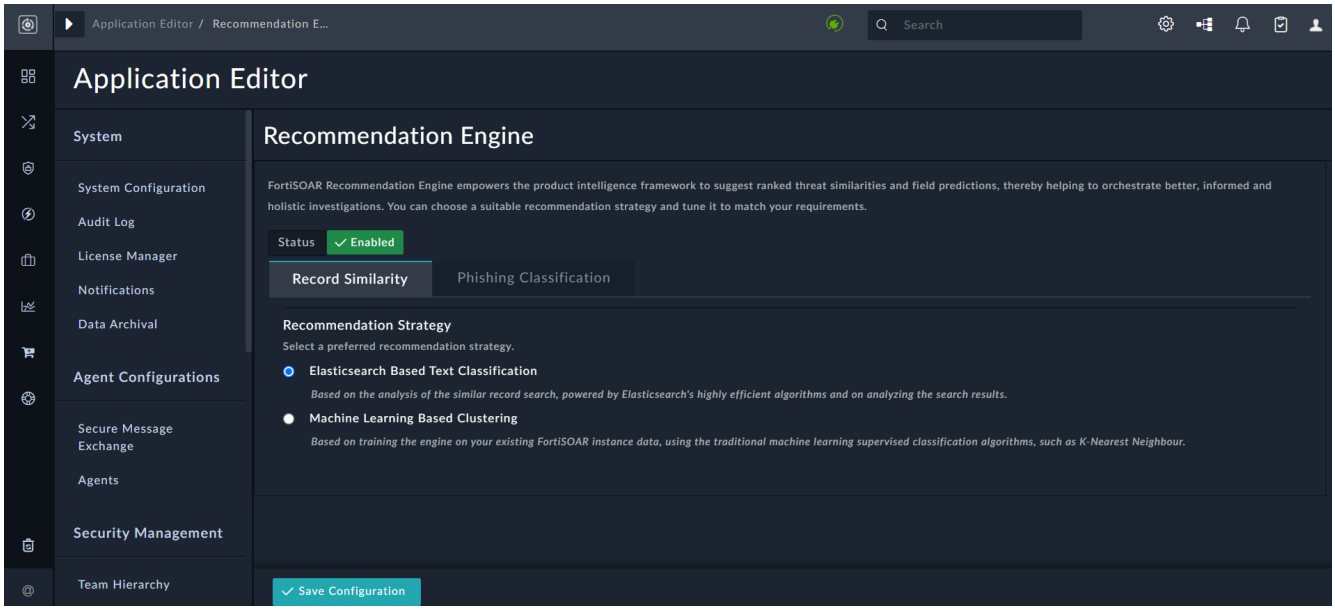
As you can see in the above image, the Correlations graph has a title which is **Tasks: Correlated Records**. The title can be specified by the user when they are adding the Visual Correlation Widget. The task for which correlated records are displayed is shown as a square node, whose color is determined by its severity, Red in

this case since the task has priority set as "Urgent". The name of the task displayed as the label of the node. The associated records are displayed as various nodes, which is determined by the legend given in the left of the graph. The color of the nodes is determined by their "Severity" in case of Incidents and Alerts and "Asset Criticality" in case of Assets. For example, linked records whose severity is critical appear in red, high appear in orange, those that are medium appear in yellow, those that are low appear in green, and so on.

Recommendation Engine

FortiSOAR's Recommendation Engine empowers the product intelligence framework to suggest ranked threat similarities and field suggestions, thereby helping to orchestrate better, informed, and holistic investigations. You can choose a suitable recommendation strategy and tune it to match your requirements. The Recommendation Engine analyzes your existing record data, recommends similar records, and predicts and assigns field values in records. From release 7.6.2, the Recommendation Engine also helps in predicting 'Phishing' emails, which improves triaging and the overall investigation process.

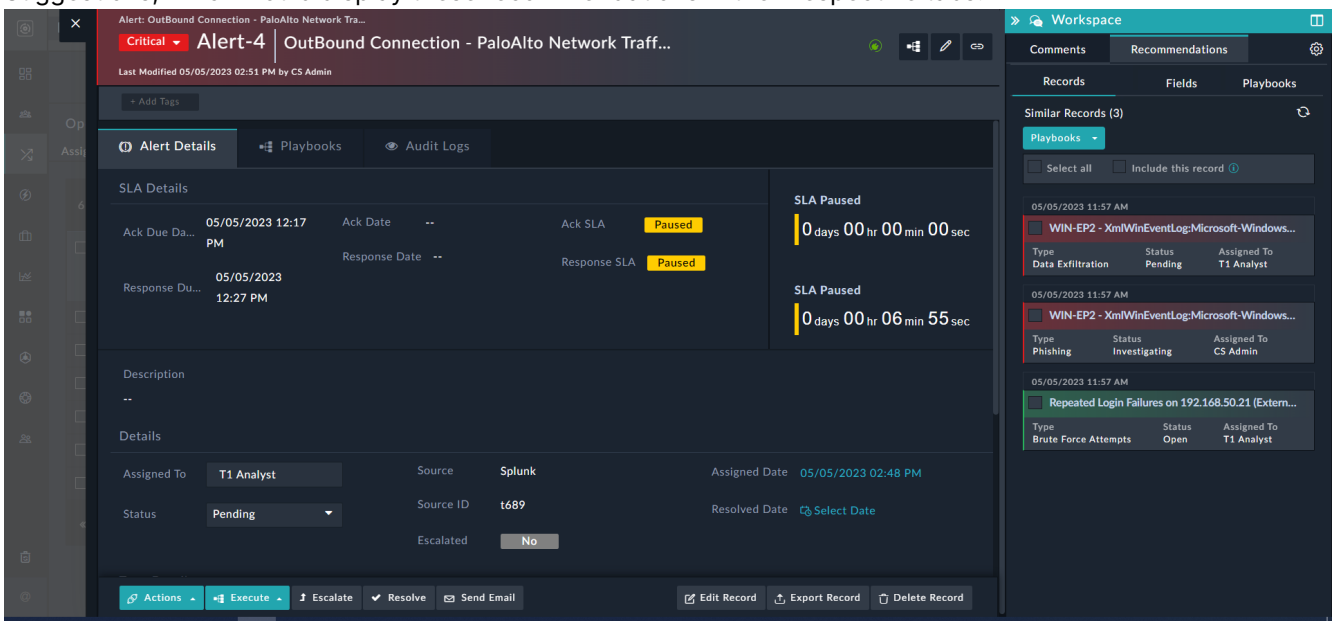
To view the 'Recommendation Engine' settings, click **Settings**, and in the `Application Editor` section, click **Recommendation Engine**:



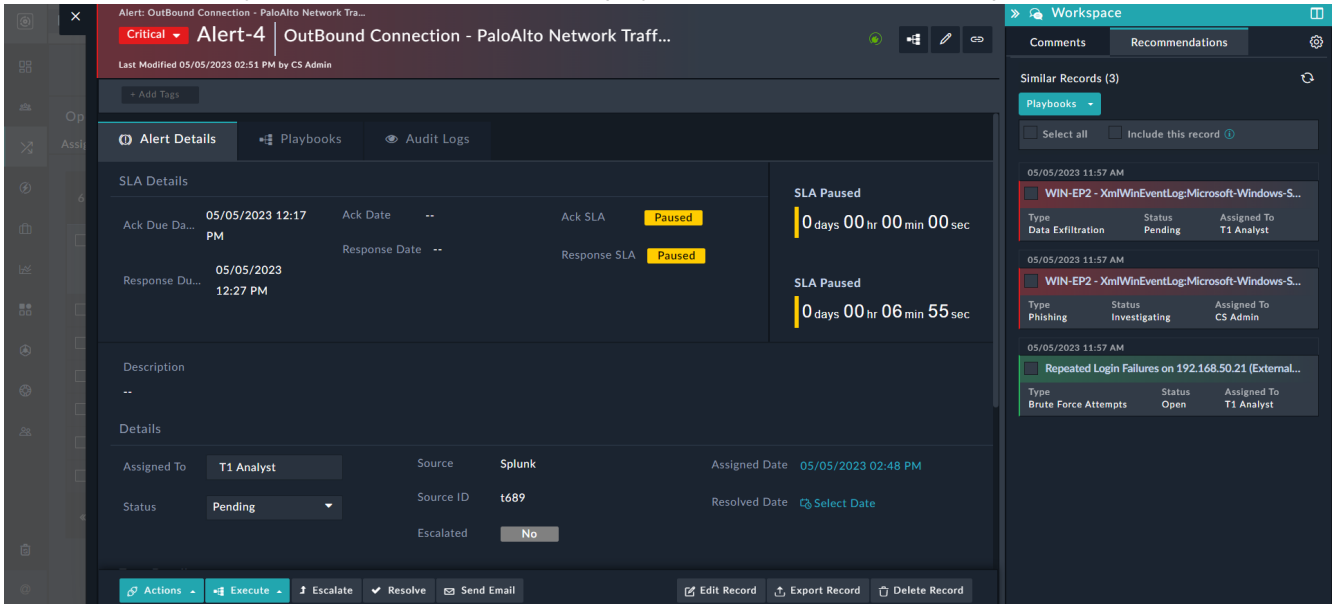
If you do not want FortiSOAR to predict or assign values to fields, suggest similar playbooks, display similar records, or predict 'Phishing Emails', then you can disable the 'Recommendation Engine' by toggling the **Status** button to 'Disabled'. By default, the 'Recommendation Engine' is enabled, i.e., the **Status** button is set to '**Enabled**'.

After you have upgraded your FortiSOAR instance to release 7.3.0 or later from a release prior to 7.3.0, recommendations using the ML Engine connector do not work until the running of the scheduled nightly re-training job. If you want to leverage recommendations before the scheduled run, manually trigger the re-training job by clicking the **Train** button.

You can choose to set up all the recommendations, i.e., Field Suggestions, Record Similarity, and Playbook Suggestions, which would display these recommendations in their respective tabs:



Or, you can choose to set up only one or two recommendations, for example, only Record Similarity, which would mean that only the Similar Records tab is displayed in the Recommendations pane:



Permissions required

To work with the Recommendation Engine and get record similarity, field suggestions, and phishing classification using the ML engine, you must be assigned a minimum of `Read` permission on the `Security` module and on the module on which you require recommendations, and `Read`, `Update`, and `Execute` permissions on the `Connectors` module.

Record Similarity, Field Suggestions, and Playbook Suggestions

FortiSOAR provides you with the 'Recommendation Engine' that analyzes your existing record data using different algorithms to recommend similar records and playbooks, and predict and assign field values in records. It is based on finding similarities of patterns in historical data.

FortiSOAR provides you with two strategies for record similarity:

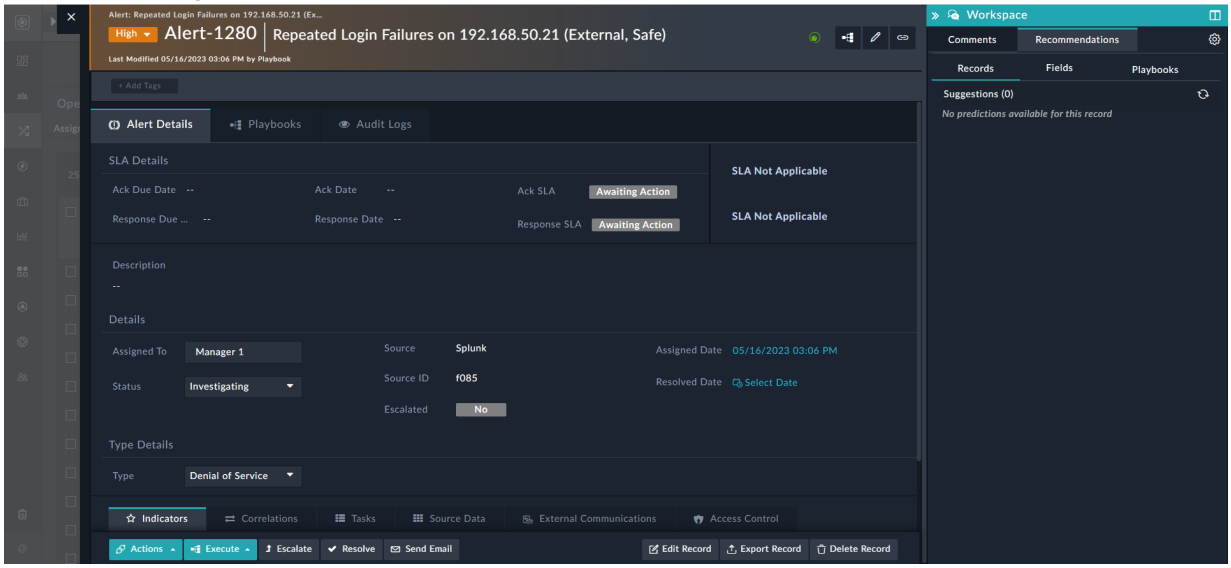
- **Elasticsearch Based Text Classification**, which is based on analysis of similar records search using Elasticsearch's efficient algorithms to analyze the search results. This is the default recommendation engine.
  Note: By default, Elasticsearch-based recommendations do not work on a FortiSOAR Docker instance due to size limitations. To know how to resolve this issue, see the `Deploying FortiSOAR on a Docker platform` chapter in the "Deployment Guide."
- **Machine Learning Based Clustering**, which is based on training the ML engine using the data existing on your FortiSOAR instance, and it uses traditional machine learning (ML) supervised classification algorithms such as 'K-Nearest Neighbors'.

Elastic Search Based Text Classification

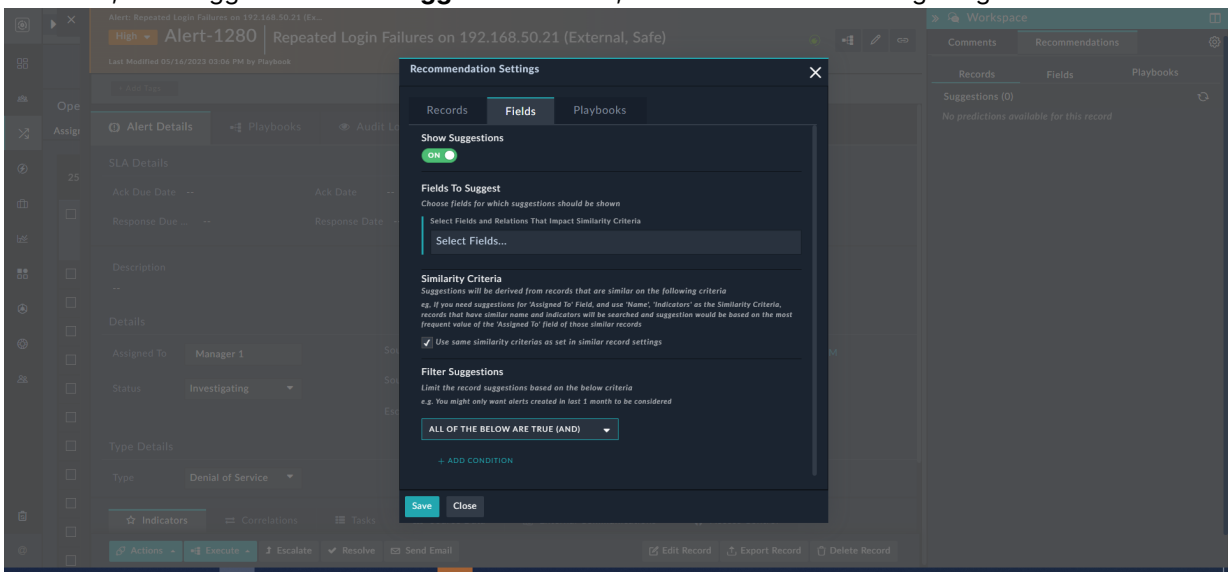
On the **Recommendation Engine** > **Record Similarity** page, select either **Elasticsearch Based Text Classification** (default) as the recommendation strategy. For **Elasticsearch Based Text Classification** you do not need to configure anything, and FortiSOAR continues to predict and assign field values and display similar records as earlier releases. However, you can set up the record similarity and field suggestion in the detail view of records based on which similar records and field suggestions get displayed to all other users.

Setting Up Field Suggestions

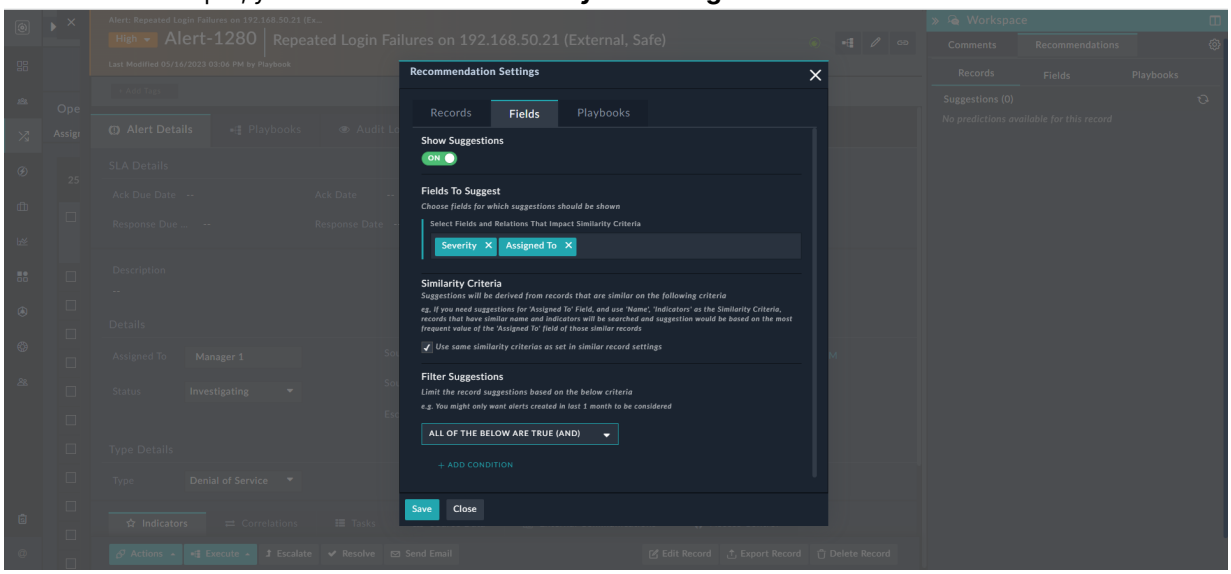
1. Open the Detail view of the alert record.
   a. Click the **Workspace** icon, and then click the **Recommendations** tab:



2. To display suggested values of fields and specify the criteria for the same, do the following:
   a. Click the **Settings** icon ( ) on the **Recommendations** tab.
   b. In the `Recommendations Settings` dialog, click the **Fields** tab, ensure that the **Show Suggestions** is toggled to **On**, which is the default
   If it is not, then toggle the **Show Suggestions** to **On**, as shown in the following image:

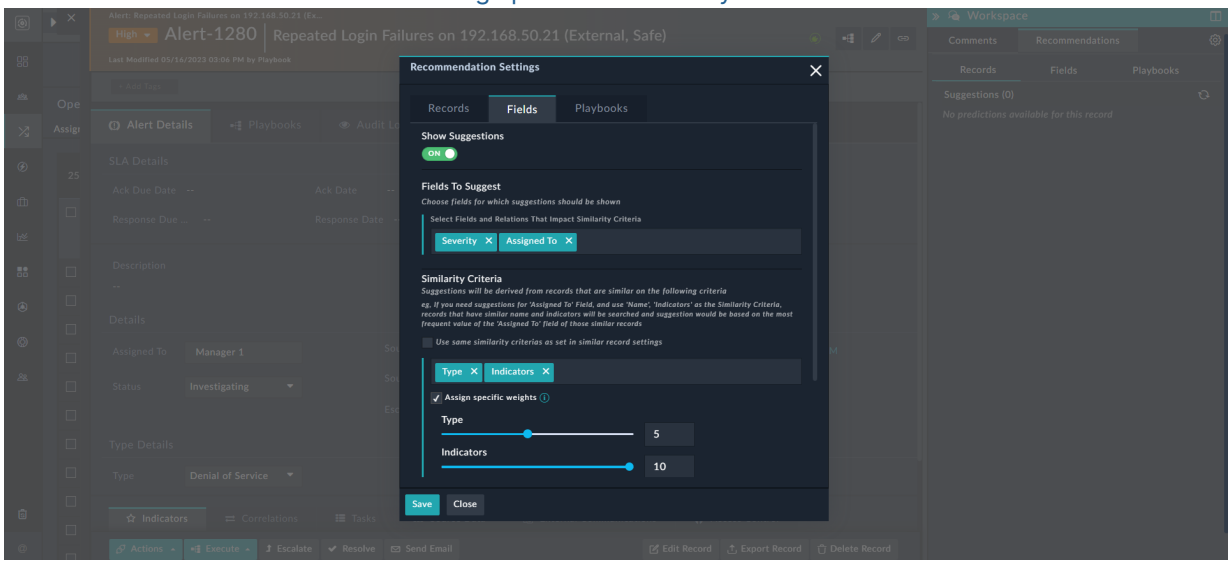


   c. In the `Fields To Suggest` section, choose the fields for which you want FortiSOAR to predict the field values. For example, you can choose the **Severity** and **Assigned To** fields:



   Important: Fields for which you want to predict values should not be on-change fields, i.e., fields that require some workflow or playbooks to be run if the value of the field is changed, as in this case even if the value of the field gets updated, the workflow will remain incomplete. An example of such a type of field would be the **Escalated** field in the "Alerts" module. If you have added *Escalated* in the `Field Suggestions`, then even though you can change the value of the Escalated field in the record as per the field suggestions, which we are assuming is set to "Yes"; the complete Escalate workflow is not completed. In this case, even though the Escalated value of the alert record is set to Yes; however, the alert is not escalated to an incident, i.e., no corresponding Incident is created, and therefore the 'Escalate' workflow remains incomplete.

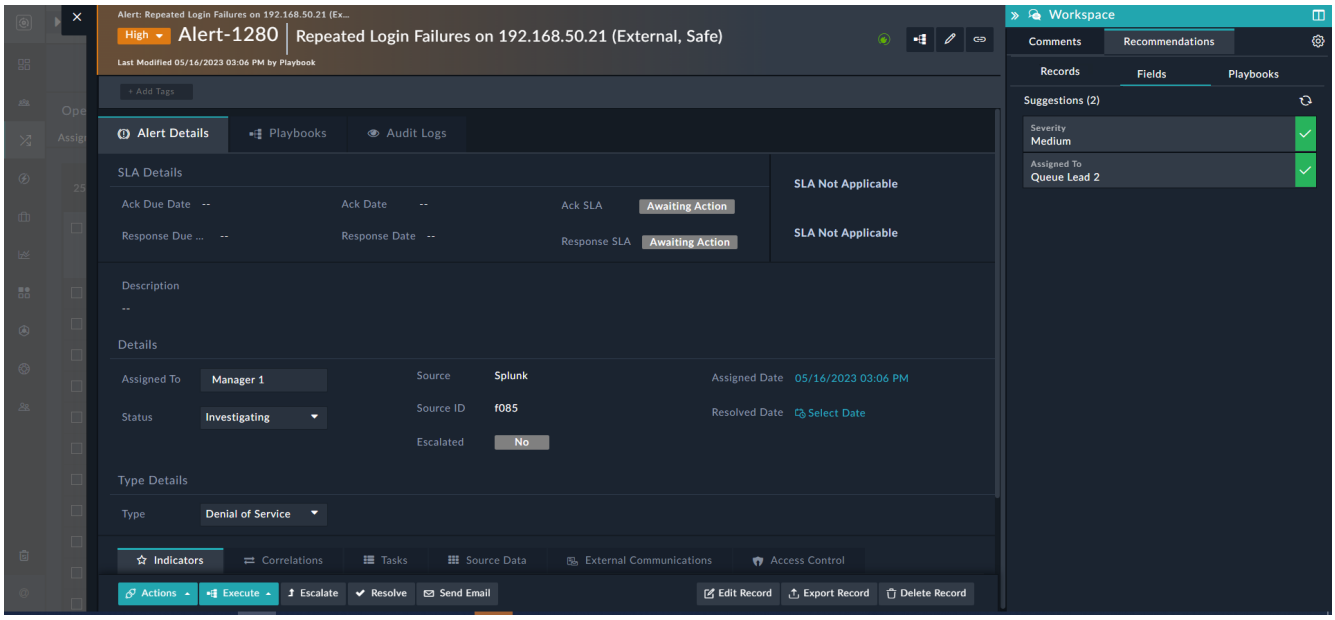
d. To use the same criteria to form the field value suggestion as you have defined for similar records, ensure that the **Use the same similarity criteria as set in similar record settings** checkbox is selected (default).
If you want to use different criteria from the field value suggestion, then clear the **Use the same similarity criteria as set in similar record settings** checkbox. Then, in the `Similarity Criteria` section, choose the fields that would form the basis for predicting field values. For our example, choose the "Type" and the "Indicators" field.
You can also assign weights to the selected fields based on which the recommended similar records will be ranked as described in the Setting up Record Similarity section.



e. (Optional) To filter the field value suggestions, in the `Filter Suggestions` section, add the filter criteria. For example, if you only want to show similar records that have been created in the last 15 days, then you can add that as a filter criterion.

f. To save the suggestion settings, click **Save**.

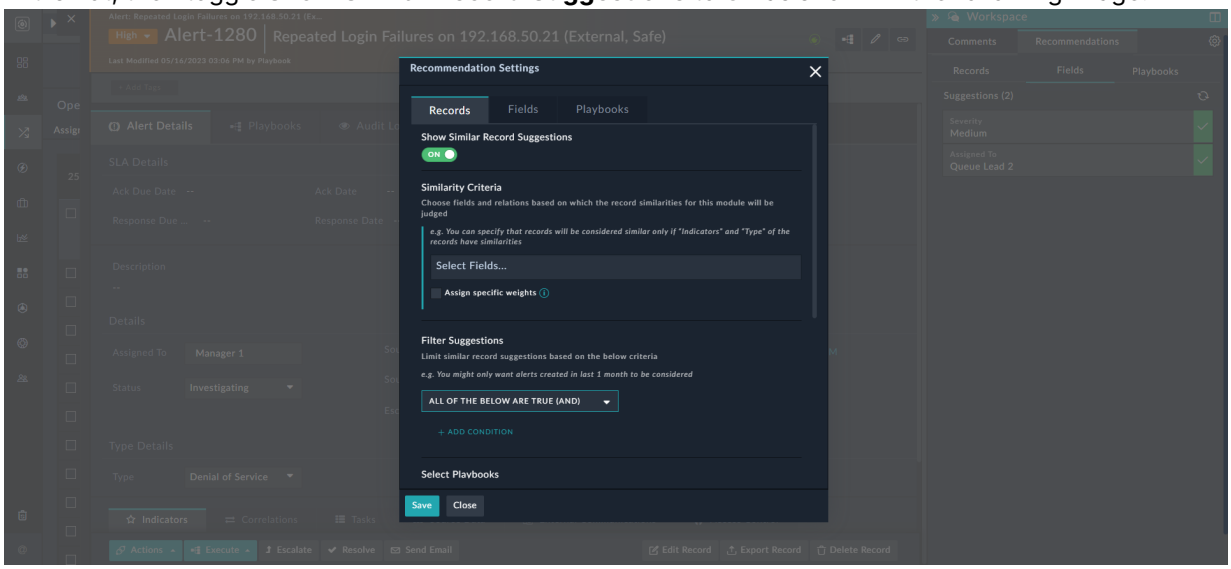
Based on the suggestion criteria that has been defined, the **Recommendations** pane > **Fields** tab will display field value suggestions as follows:



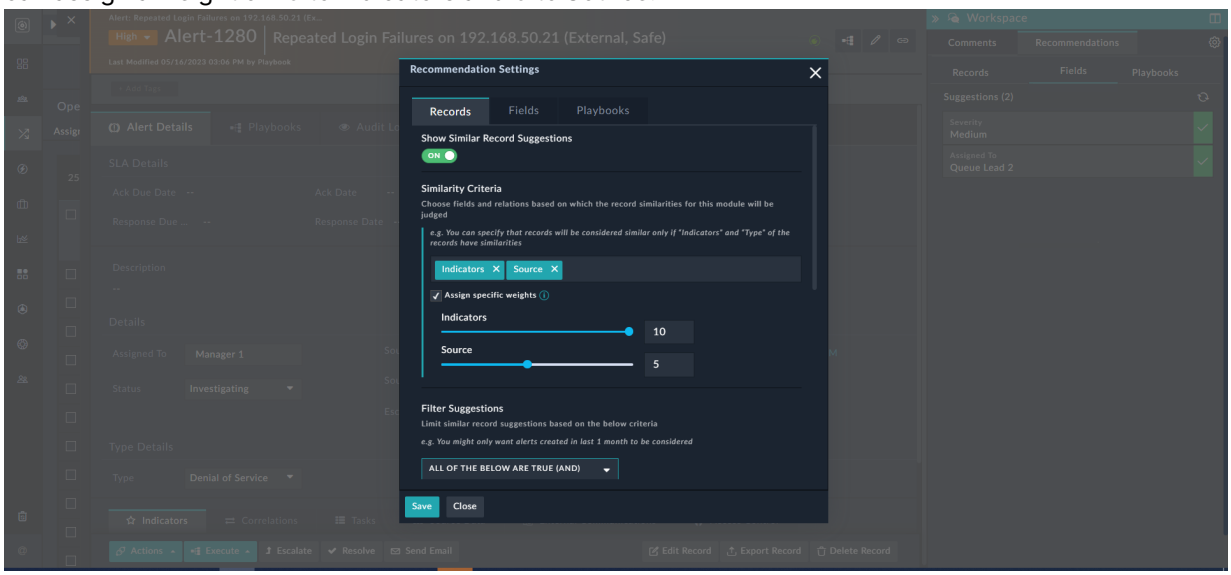
Using the field suggestions, users can choose to set the value of fields such as severity or assigned to, across all the similar records. For more information, see the *Working with Modules - Alerts & Incidents* and the *Dashboards, Templates, and Widgets* chapters in the "User Guide."

Setting Up Record Similarity

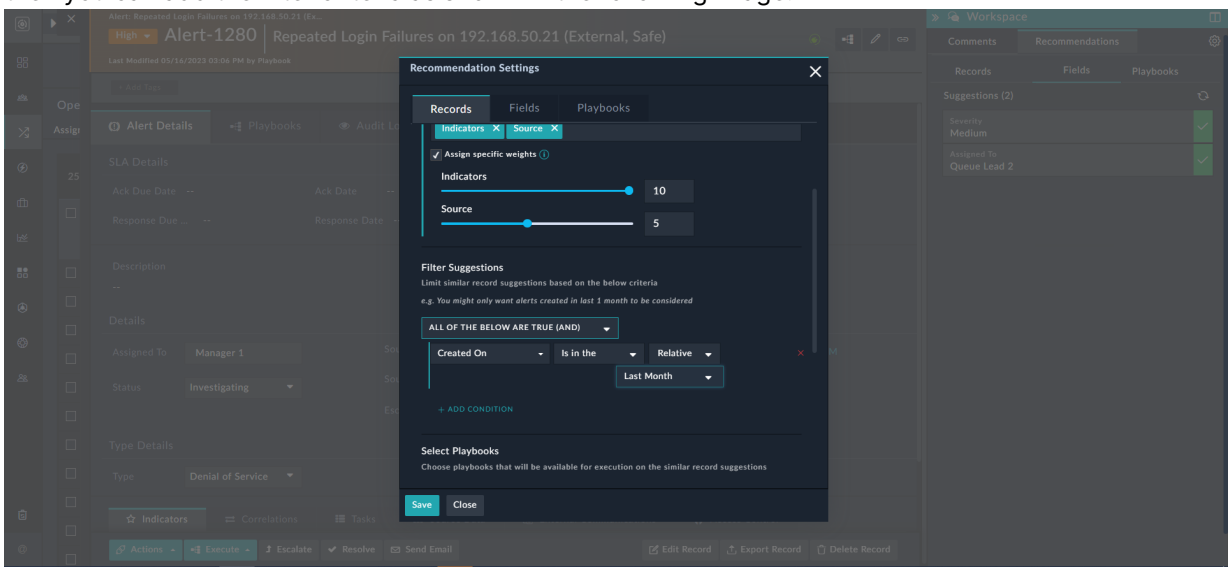
1. Open the Detail view of the alert record.
2. Click the **Workspace** icon, and then click the **Recommendations** tab.
3. To display similar records and specify the similarity criteria do the following:
   a. Click the **Settings** icon ( ) on the **Recommendations** tab.
   b. In the `Recommendations Settings` dialog, on the **Records** tab, ensure that the **Show Similar Record Suggestions** is toggled to **On**, which is the default.
   If it is not, then toggle **Show Similar Record Suggestions** to **On** as shown in the following image:



c. In the `Similarity Criteria` section, choose the fields and relations to create the criteria based on which records will be displayed.
For example, if you want to display the alerts whose indicators, such as domains, IP addresses, URLs, etc match the indicators of the alert record on which you are working and we also want to match the source of the alerts. Therefore, you will choose **Indicators** and **Source** from the **Select Field** drop-down list.
You can also assign weights to the selected fields based on which the recommended similar records will be ranked. To assign ranks, select the **Assign specific weights** checkbox, then use the slider to assign weights for each of the selected fields from 1 to 10, with 10 being the highest value. For example, if you want to give higher weightage to similar Indicators as compared to Source, then you can assign a weight of 10 to Indicators and 5 to Source:



d. (Optional) To filter the similar record suggestions, in the `Filter Suggestions` section, add the filter criteria. For example, if you only want to show similar records that have been created in the last month, then you can add the filter criteria as shown in the following image:

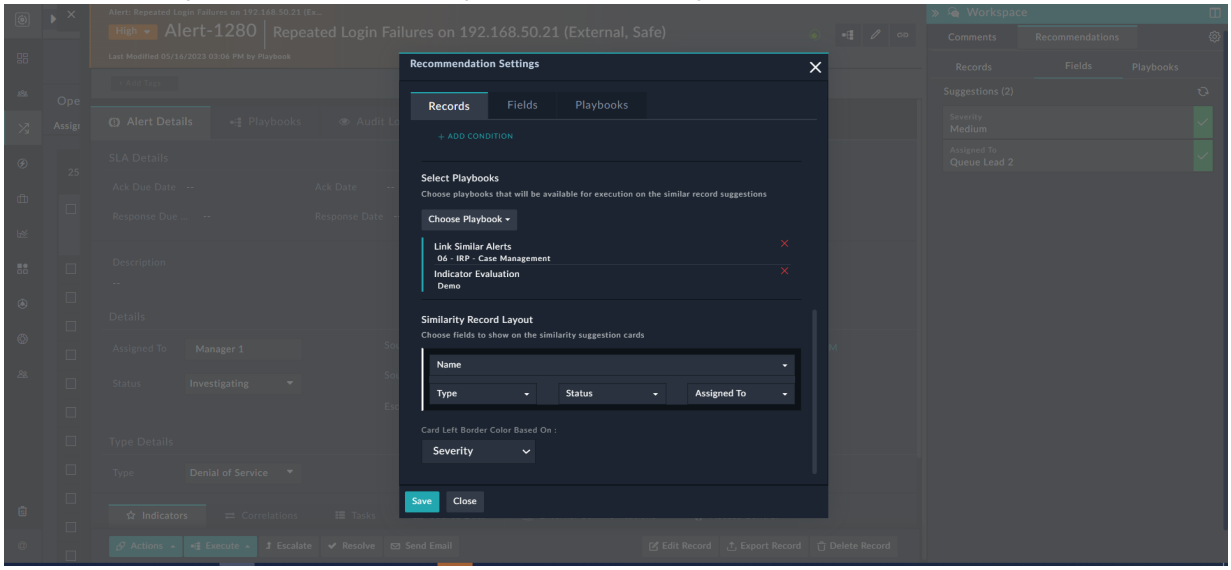


Adding filters narrows the records down to a smaller set, which in turn returns the results faster. For example, searching for similar records in the last one month will return results faster than searching for

similar records in the last one year.
Note: If you assign a "Custom" filter to a DateTime field, such as Assigned Date, then the date considered will be in the "UTC" time and not your system time.

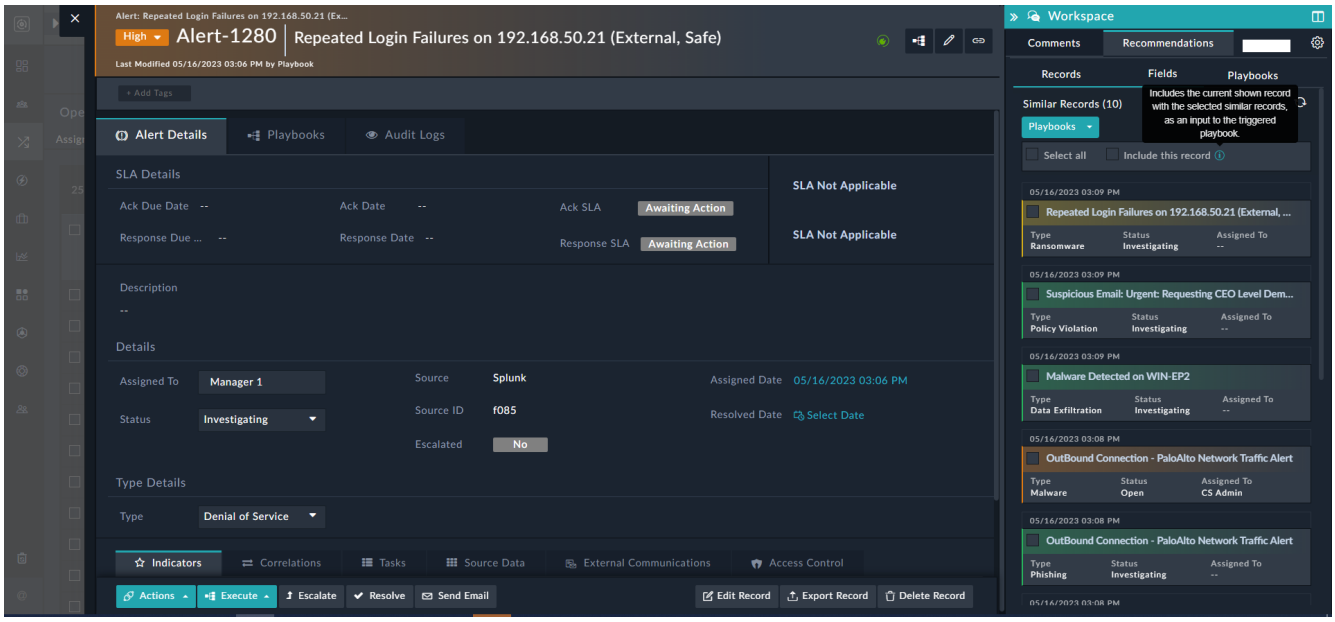
e. (Optional) In the `Select Playbooks` section, from the **Choose Playbook** list, search and select the playbooks that will be displayed on the **Recommendations** panel and which you can execute on similar records. For example, you can choose to evaluate indicators and therefore choose to run the `Indicator Evaluation` and `Link Similiar Alerts` playbooks on similar records.

f. To define the layout of the similar records, in the **Similarity Record Layout** section, you can specify the fields of the similar records that you want to include. For example, you can choose **Name**, **Type**, **Assigned To**, and **Status**, as the fields of the similar records that should be displayed.
You can also define the color of the left border of the card based on a specific picklist or field in the **Card Left Border Color Based On** field. The color of the card will depend on the colors that you have defined for the picklist items. For example, choose **Severity**:



g. To save the similarity settings, click **Save**.

Based on the similarity criteria that has been defined, the **Recommendations** pane > **Records** tab will display similar alerts as follows:



Using the record similarity criteria set, users can view the list of records that are similar to the record that they are working on and can quickly perform various actions across all the similar records such as evaluating similar indicators across the module, marking all the records as '*Resolved*', etc. For more information, see the *Working with Modules - Alerts & Incidents* and the *Dashboards, Templates, and Widgets* chapters in the "User Guide."

Setting Up Playbook Suggestions

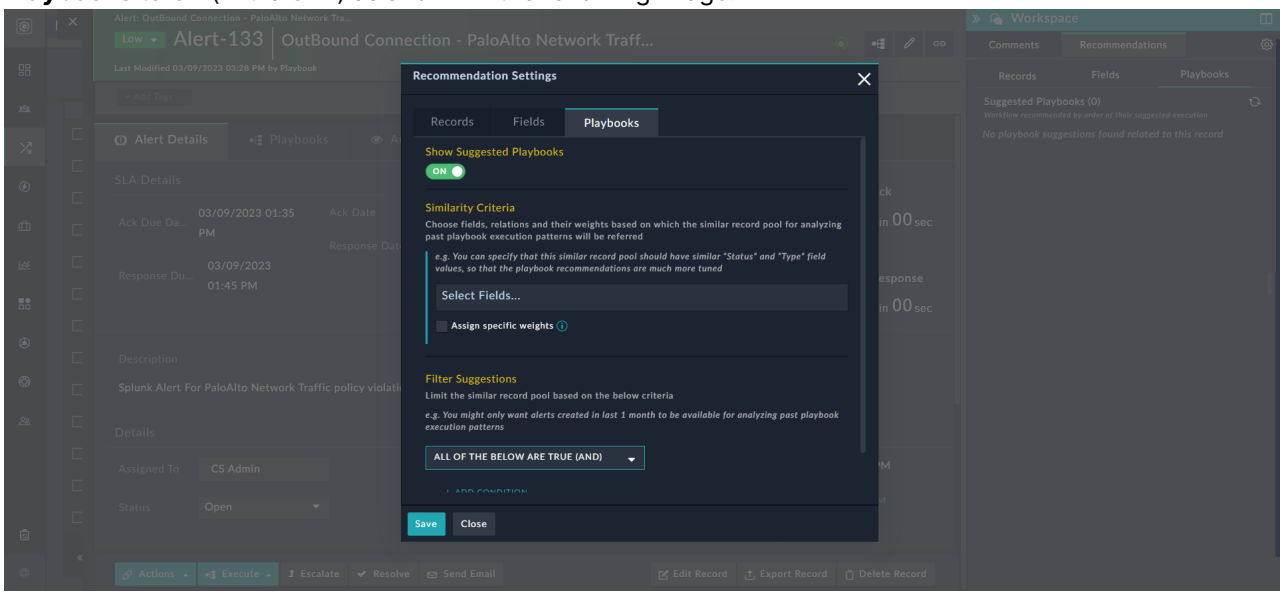
From release 7.4.0 onwards, you can setup workflow suggestions that enable analysts to view the workflow that is most likely to be executed on a given record in a particular order based on the previous actions on related records. This helps new analysts quickly become familiar with the many workflow processes that the organization uses for a particular record, for example, an alert record.

Systems where playbook suggestions have been setup and which have a fresh installation of FortiSOAR 7.4.0 or later will start displaying playbook suggestions over time, based on the generated data. For upgraded systems to 7.4.0 (or later), playbook suggestions will be displayed after playbooks are triggered on records after the upgrade to 7.4.0.

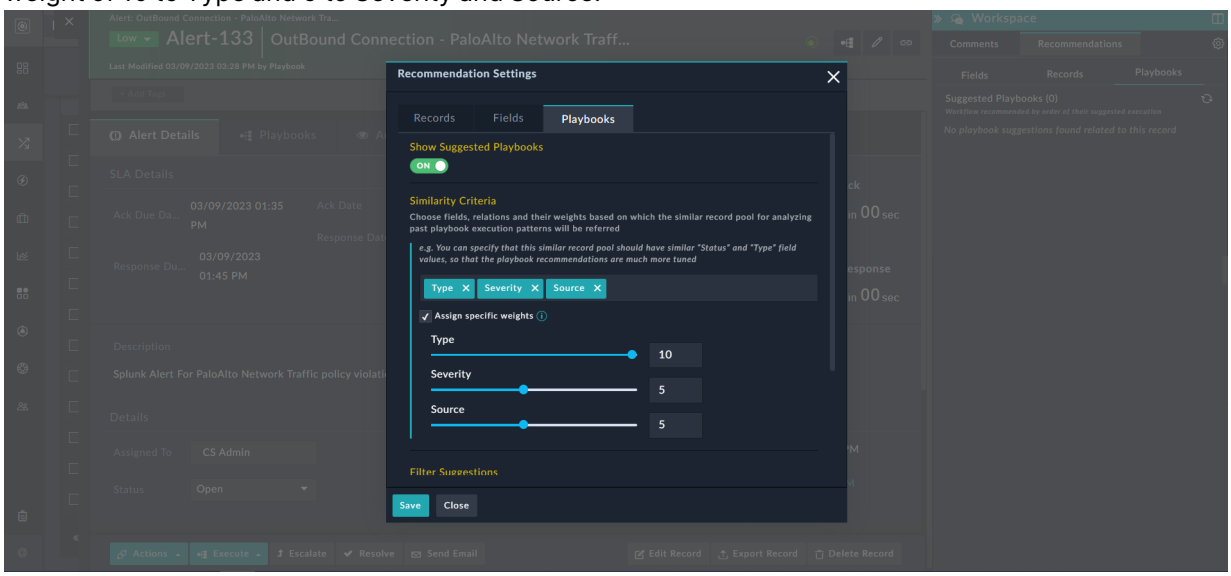
To set up playbook suggestions, do the following:

1. Open the Detail view of the alert record.
2. Click the **Workspace** icon, and then click the **Recommendations** tab.
3. Click the **Settings** icon ( ) on the **Recommendations** tab.

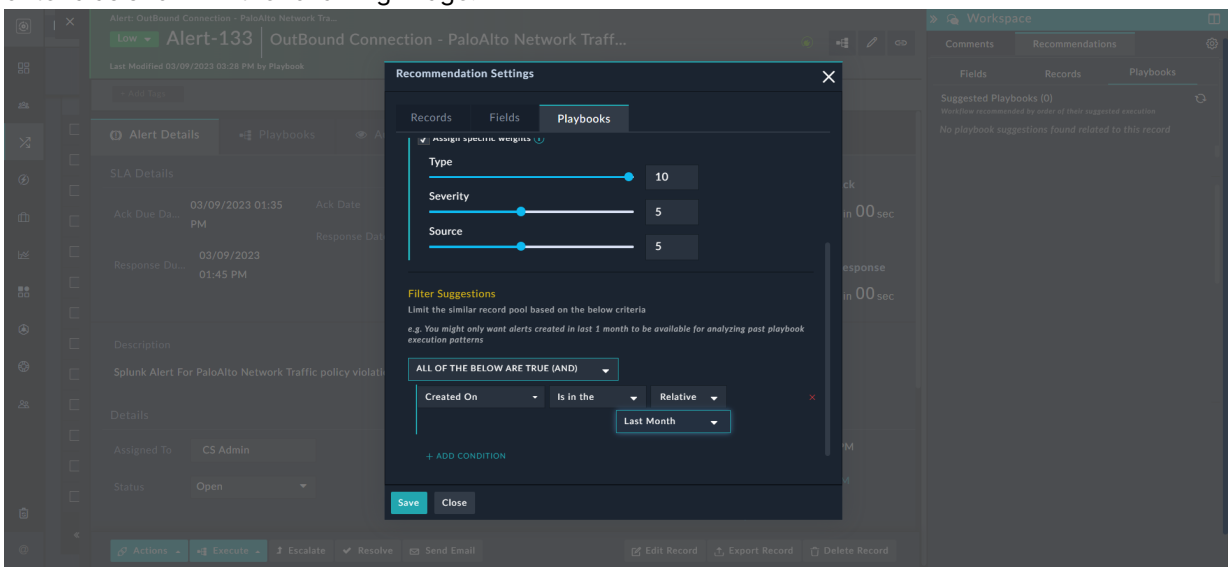
4. In the `Recommendations Settings` dialog, click the **Playbooks** tab, and then toggle **Show Suggested Playbooks** to **On** (if it is **OFF**) as shown in the following image:



5. To display the suggested playbooks and specify the criteria for the same, do the following:
   a. In the `Similarity Criteria` section, choose the fields and relations to create the criteria to determine the pool of similar records. Playbook suggestions are displayed considering previously executed playbooks pattern on the similar records' pool.
   For example, you can determine the similarity criteria based on the type, severity, and source. Therefore, you will choose **Type**, **Severity**, and **Source** from the **Select Field** drop-down list. Based on this criteria, similar records will be identified, and based on the playbooks run on the similar records, playbook suggestions are displayed.
   You can also assign weights to the selected fields based on which the suggested playbooks are ranked. To assign ranks, select the **Assign specific weights** checkbox, then use the slider to assign weights for each of the selected fields from 1 to 10, with 10 being the highest value. For example, if you want to give higher weightage to similar types as compared to severity or source, then you can assign a weight of 10 to Type and 5 to Severity and Source:



b. (Optional) To filter the suggestions from the pool of similar record, in the `Filter Suggestions` section, add the filter criteria. For example, if you only want to consider those similar records that have been created in the last month for analyzing past playbooks execution patterns, then you can add the filter criteria as shown in the following image:



Note: If you assign a "Custom" filter to a DateTime field, such as Assigned Date, then the date considered will be in "UTC" time and not your system time.

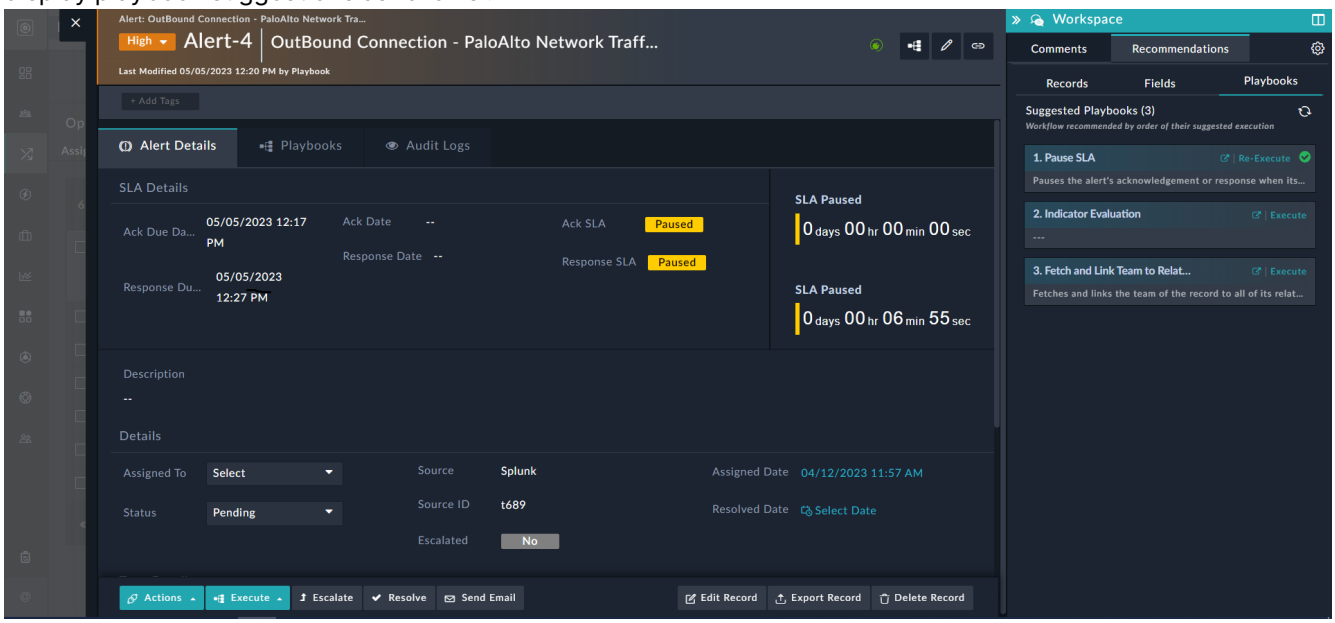
c. To save the similarity settings that would determine the suggested playbook list, click **Save**.

Playbook suggestions honor the conditions defined on the manual triggers. For example, defining a visibility condition on a playbook where that playbook is visible only for those alerts whose 'Type' is set as 'Phishing'. In this case, this particular playbook would only be shown in the playbook suggestions (if other criteria are met) of alert records with 'Type set to Phishing'.

Also, note that for viewing and executing playbooks suggestions, users must have a minimum of **Read** and **Execute** permissions on the `Playbooks` module (apart from other required permissions). Users must also have requisite permissions on similar records otherwise, playbooks executed on similar records (on which users do not have permissions) will not be suggested.

Add the `excludefromsuggestion` tag to playbooks that you want to exclude from appearing as suggestions in the 'Suggested Playbooks' list. For example, add the `excludefromsuggestion` tag to exclude the 'Pause SLA' playbook from the 'Suggested Playbooks' list. After this tag is added to the playbook, further runs of the playbook are not considered for suggestions; however, previous runs are still considered.

Based on the suggestion criteria that have been defined, the **Recommendations** pane > **Playbooks** tab will display playbook suggestions as follows :



Using the playbook suggestions, users can choose to run the playbook on the current record. For more information, see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide."

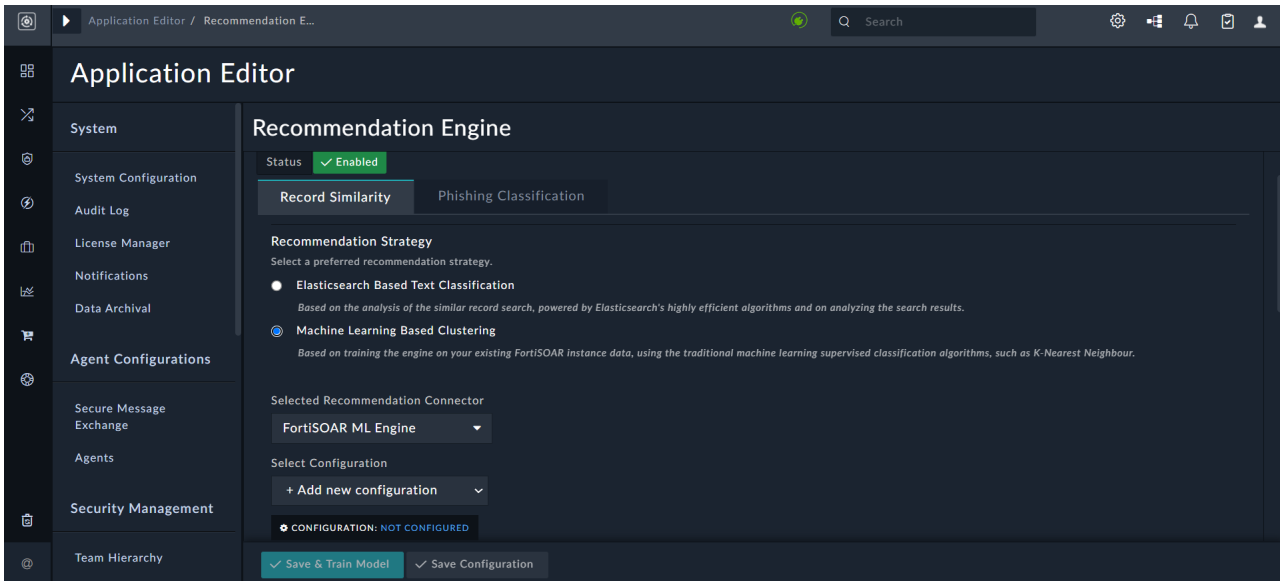
Machine Learning Based Clustering Text Classification

On the **Recommendation Engine** > **Record Similarity** page, select **Machine Learning Based Clustering** as the recommendation strategy. For **Machine Learning Based Clustering**, you need to train the ML engine using the data existing on your FortiSOAR instance. AI/ML technology can leverage past learning and similar patterns to smart predict values of record fields such as 'Assigned To' and 'Severity'. For example, for an incoming alert of type, Malware, your FortiSOAR system can fall back to similar Malware alerts that already existed in your system, and based on the similarity in patterns suggest values to the 'Assigned To' and 'Severity' fields in the new record. This saves time in a SOC as the task of sifting through records and assigning them is now done automatically.

After you have upgraded your system to release 7.2.0 or later from a release prior to 7.2.0, and you observe that errors are being displayed for field suggestions and record similarity, then you must retrain your Machine Learning model.

To configure and train the ML engine, do the following

1. On the '**Recommendation Engine**' > '**Record Similarity**' page, ensure that the Status of the Recommendation Engine is set to 'Enabled' and the **Machine Learning Based Clustering** option is selected. You will observe the FortiSOAR **ML Engine** is selected in the **Selected Recommendation Connector** drop-down list.

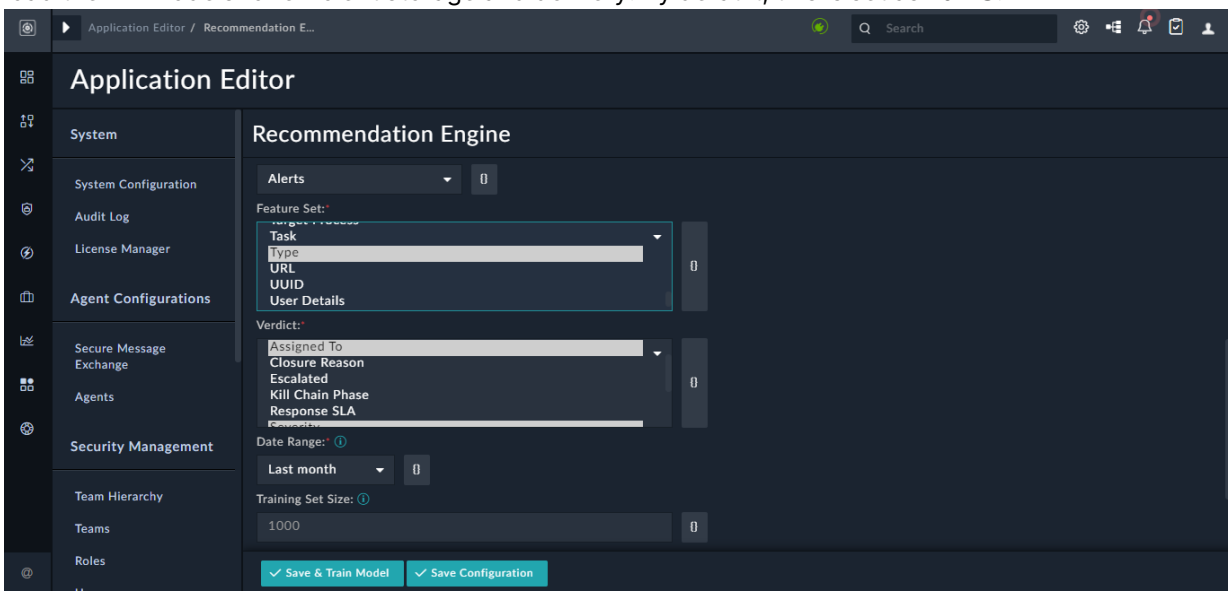


NOTE: Version 1.2.1 or later of the ML Engine connector is supported on FortiSOAR release 7.3.0 or later. Therefore, it is recommended not to install or upgrade the ML Engine connector v1.2.1 on FortiSOAR releases earlier than 7.3.0, such as FortiSOAR release 7.2.2, 7.2.1, etc. For further information on the ML Engine connector, see the ML Engine connector documentation in the FortiSOAR Content Hub.

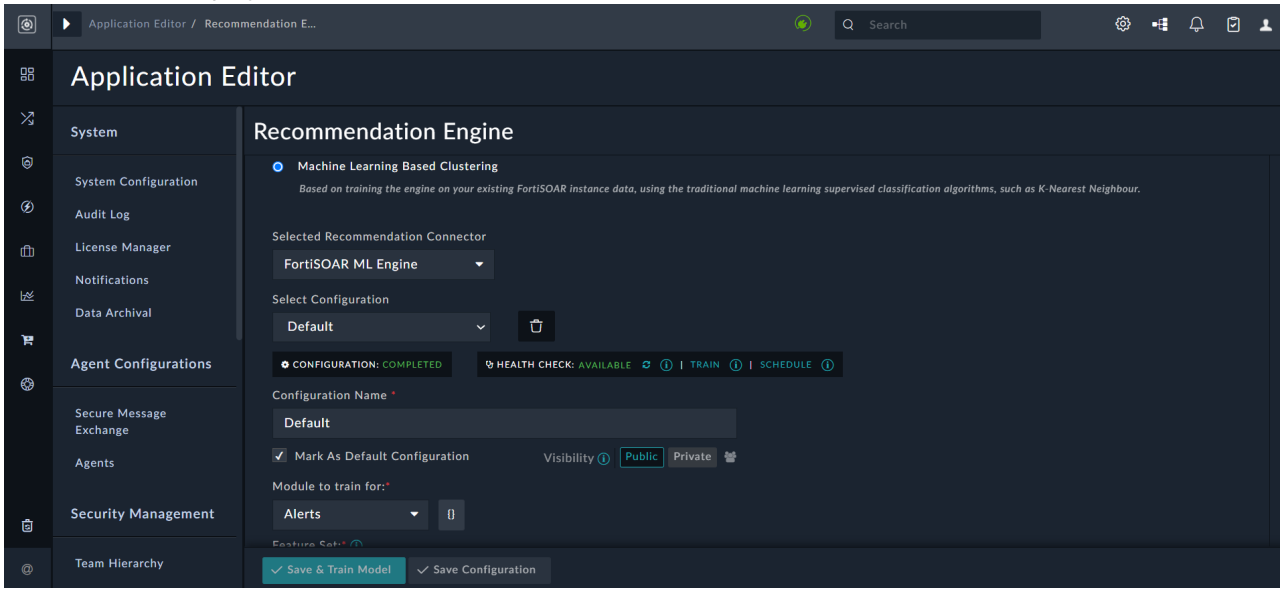
2. To configure FortiSOAR **ML Engine**, in the **Configuration Name** field, add a *unique name* for the configuration. The configuration name needs to be unique since you can have multiple configurations. Select the **Mark As Default Configuration** checkbox, if you want this particular configuration to be the default configuration of this connector, on this particular FortiSOAR instance.
   Note: You must select one configuration to be the default configuration of the FortiSOAR **ML Engine** connector.
   To add a new configuration, click the **Select Configuration** drop-down list and click **+ Add new configuration**. You can specify different training datasets (modules) for each configuration and can also create different training schedules for the datasets for each configuration. However, as a *best practice and for consistent results, you should have a single configuration per module*.
3. To train the FortiSOAR **ML Engine**, do the following:
   a. **From the Module to train for** drop-down list, select the module from which you want to select the fields for training and the fields that you want to predict. By default, the **Alert** module is selected.
   b. From the **Feature Set** list, select the field(s) using which you want to predict the field values. To select multiple fields, press `Ctrl` and select the field.
      For our example, where we want to predict the 'Assigned To' and 'Severity' fields based on the 'Type' of alert, select the **Type** field.
   c. From the **Verdict** list, select the field(s) that you want to predict. To select multiple fields, press `Ctrl` and select the field.
      For our example, we want to predict the 'Assigned To' and 'Severity' fields, therefore select the **Assigned To** and **Severity** fields.
   d. From the **Date Range** drop-down list, select the time range of records based on which you want to populate the training set.
      You can select from options such as Last Month, Last 6 months, Last year, etc. You can also select **Custom** and then specify the last X days to populate the training set.
   e. The **Training Set Size** specifies the number of records that make up the training set. It is set as 1000 records.
      Note: The value that you select from the **Date Range** drop-down list overrides this parameter

f. From the **Algorithm** drop-down list, select the ML supervised classification algorithm using which you want to predict the fields. You can choose between **K-Nearest Neighbors** (default) or **Decision Tree**.

g. In the **Listener Port** field, specify the port number of the socket where the ML Engine connector will load the ML models for efficient storage and delivery. By default, this is set as `10443`.

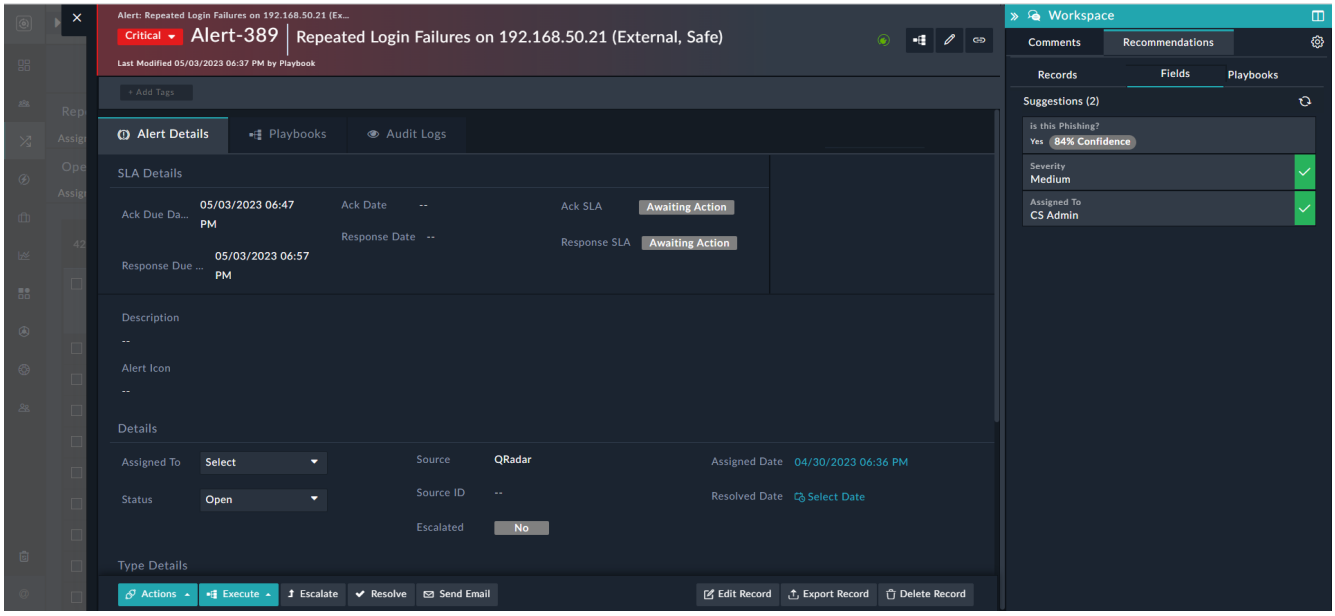


4. Once you have specified the dataset to be used for training the FortiSOAR **ML Engine**, click **Save & Train Model**. Clicking **Save & Train Model** saves the specified parameters and trains the model, 'Alerts' in our example, based on these parameters.
You will observe that the Configuration of the connector displays as Completed and the **Heath Check** of the connector displays as **Available**.

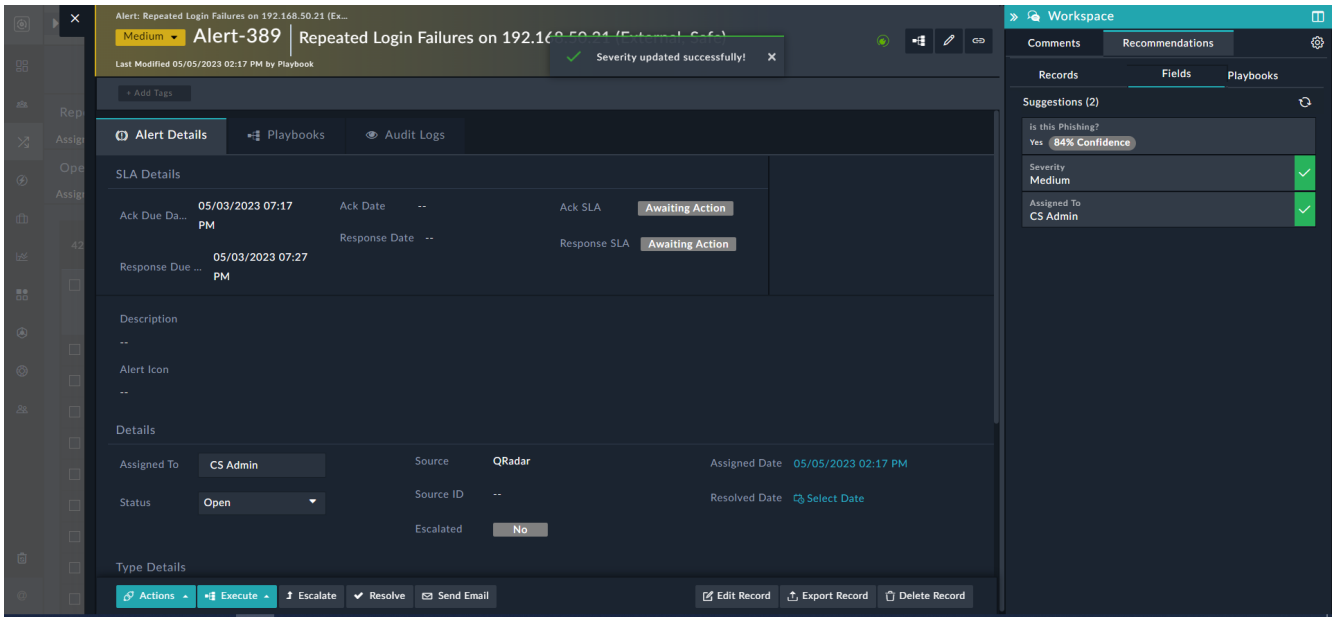


If you make any changes to the training dataset, such as adding or removing a field from the **Feature Set** or **Verdict**, click **Train** to update the dataset. To ensure that the dataset is trained on new incoming data regularly, you can also choose to train your dataset at regular intervals by scheduling training of the dataset by clicking **Schedule**. Clicking **Schedule** opens the `Schedule Details` dialog using which you can create a schedule for training your dataset. For more information on schedules, see the *Schedules* chapter in the "User Guide."

Once you have trained your dataset, FortiSOAR starts to analyze your dataset and based on the analysis displays records that are similar to the record you are working on, as well, predicts the values of field records that you have added to the **Verdict** field in the **Recommendations** pane > **Fields** tab. Since we have trained the dataset, in our example, to predict the 'Assigned To' and 'Severity' fields based on the 'Type' field, FortiSOAR provides suggestions for those fields as shown in the following image:



If you agree to the recommendations, then click the green check box beside the field, and that will populate that field in the record. For example, clicking the **Severity** green checkbox assigns 'Medium' as the record severity.



Similarly, you can view the list of records that are similar to the record that you are working on enabling you to quickly take remedial action. For more information on using record similarity and predicting values of field values in a record see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide."

Phishing Classification

Phishing is probably the most common form of cyber-attack, largely because it is easy to accomplish, and surprisingly effective. It is a type of social engineering attack wherein an attacker impersonates to be a trusted contact and sends the victim fake mails. Therefore, in release 7.2.0, FortiSOAR introduces the 'Phishing Classification' feature. Phishing Classifier is a Machine Learning-based classifier that helps to predict emails that can be 'Phishing' emails, which helps speed up the triage and overall investigation process.

FortiSOAR provides you with two methods for training the 'Phishing Classifier' connector to predict phishing emails

- **Pre-trained Model**: This model is trained using thousands of real-world phishing emails from Fortinet's security team. It is a quick-start way to understand the classification process.
- FortiSOAR **Module**: This model is a new machine-learning (ML) model that you create by training your local dataset using an existing FortiSOAR module and its records.

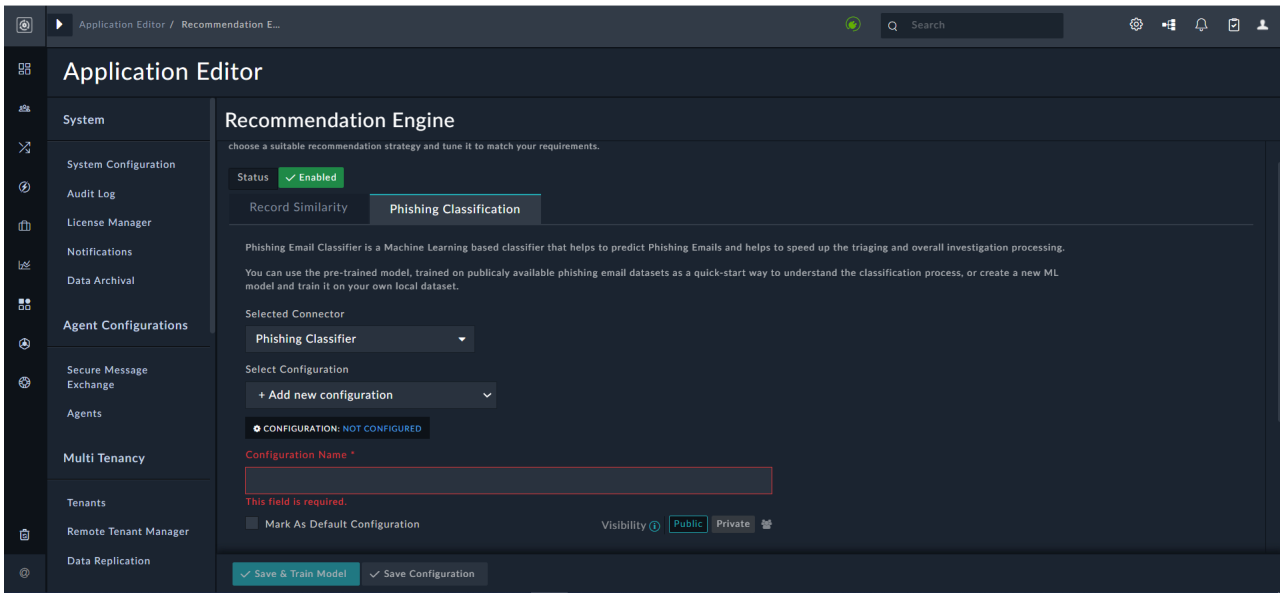
In the case of HA environments, all the training and suggestion operations take place on the primary node.

You should configure the 'Phishing Classifier' on either your FortiSOAR node or your FSR Agent node, but not simultaneously on your FortiSOAR node and your FSR Agent node.

Administrators can see the Advanced Settings topic if they want to make some advanced changes such as changing the port used by the ML engine, or changing the normalization technique used by the phishing classifier.

Configuring Phishing Classification based on the Pre-trained Model

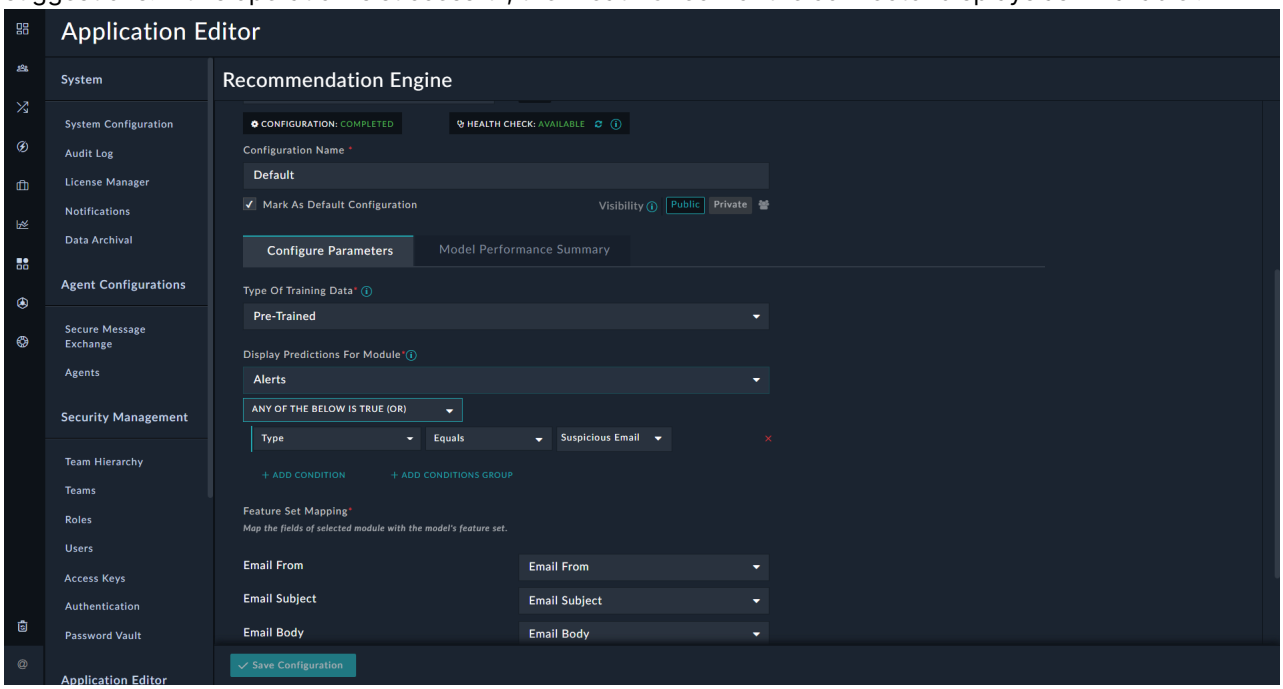
1. On the 'Recommendation Engine' > 'Phishing Classification' page, ensure that the status of Recommendation Engine is set to Enabled. You will observe the **Phishing Classifier** is selected in the **Selected Connector** drop-down list.



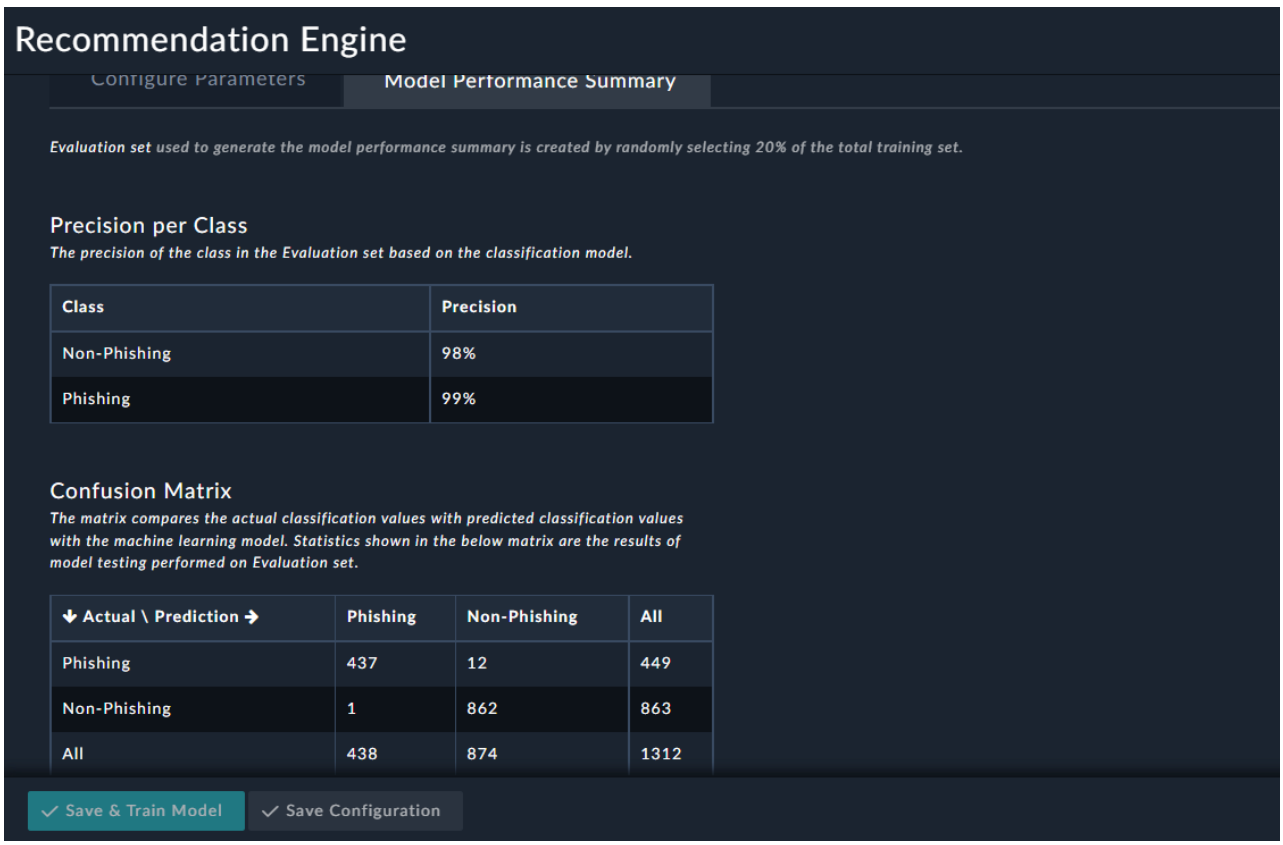
NOTE: Version 1.0.1 or later of the Phishing Classifier connector is supported on release 7.3.0 or later. Therefore, it is recommended not to install or upgrade the Phishing Classifier connector v1.0.1 on releases earlier than 7.3.0, such as release 7.2.2, 7.2.1, etc. For further information on the Phishing Classifier connector, see the Phishing Classifierconnector documentation in the Content Hub.

2. To configure the Phishing Classifier connector, in the **Configuration Name** field, add a *unique name* for the configuration. The configuration name needs to be unique since you can have multiple configurations. Select the **Mark As Default Configuration** checkbox, if you want this particular configuration to be the default configuration of this connector, on this particular FortiSOAR instance.
   Note: You must select one configuration to be the default configuration of the Phishing Classifier connector.
   If you have an existing configuration, then to add a new configuration, click the **Select Configuration** drop-down list and click **+ Add new configuration**. You can specify different training datasets for each configuration and can also create different training schedules for the datasets for each configuration. However, as a *best practice and for consistent results, you should have a single configuration per module*.
3. On the **Configure Parameters** tab, configure the following parameters:
   a. From the **Type of Training Data** drop-down list, select **Pre-Trained**.
      Choosing Pre-Trained means that you want to use the pre-trained dataset to predict 'Phishing Emails'.
   b. From the **Display Predictions For Module** drop-down list, select the module for whose records you want to display the suggestions. For example, if you select **Alerts**, it means that you want to suggestions for alert records.
   c. (Optional) To add further filtering on these records, click **Add Data Filters** and add additional filters. In this case, suggestion will be shown for only those records that satisfy the filters. For example, add `Type Equals Suspicious Email`, to classify only those alerts whose type is suspicious email.
   d. In `Feature Set Mapping`, specify which fields of the module selected for suggestion ('Alerts' in our example) maps to the ML model fields. You have to map the following fields: Email From, Email Subject, and Email Body.
4. Once you have specified the configurations for training the Phishing Classifier connector, click **Save Configuration**, which saves the specified parameters and begins training the model, 'Alerts', in our example, based on these parameters. At this step, the pre-trained model is loaded into memory for

suggestions. If this operation is successful, the 'Heath Check' of the connector displays as 'Available':

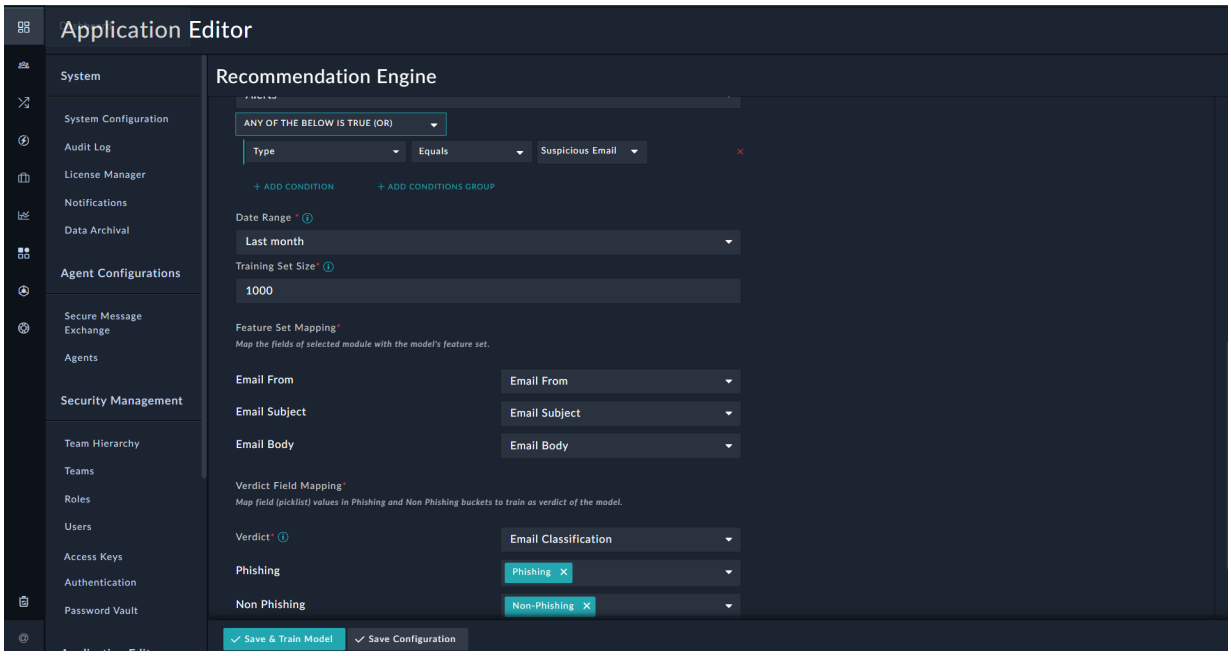


5. Click the **Model Performance Summary** tab to view the performance of the model in terms of precision per class:

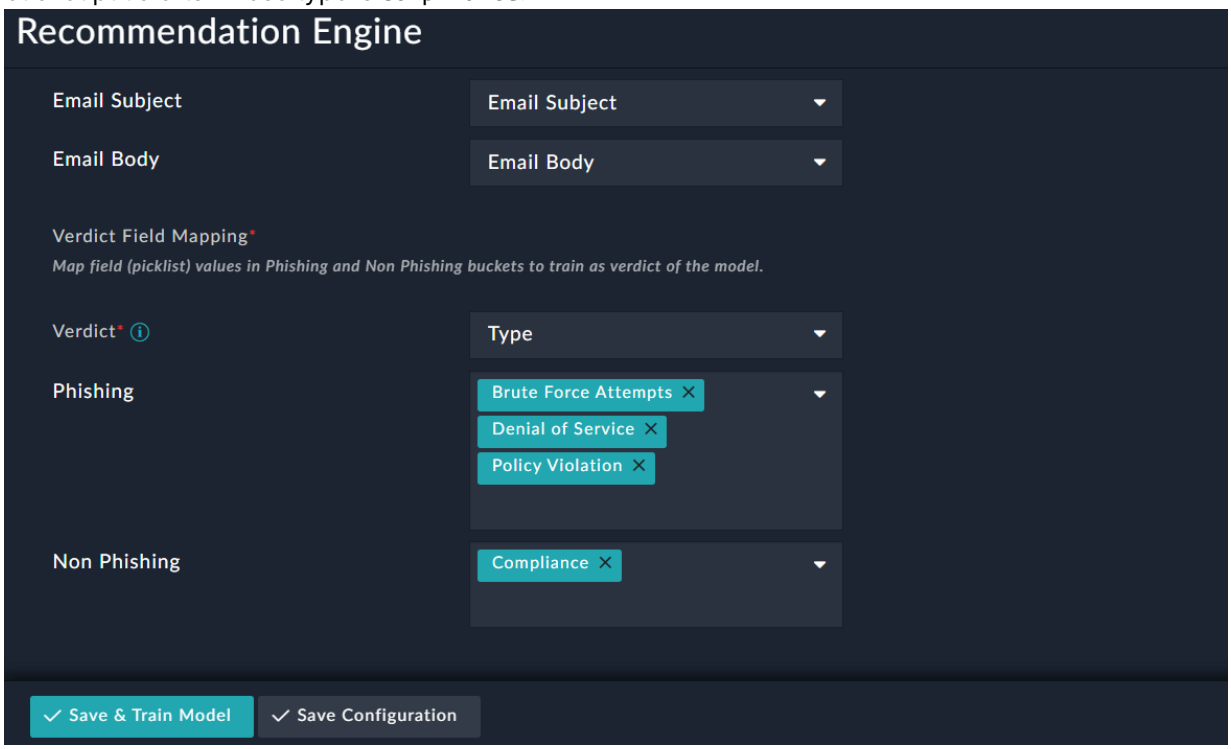


Configuring Phishing Classification based on a FortiSOAR Model

1. Configure the Phishing Classifier connector as mentioned in steps 1 and 2 of the Configuring Phishing Classification based on the Pre-trained Model section.
2. After you have configured the Phishing Classifier connector you need to specify its training dataset:
   a. From the **Type of Training Data** drop-down list, select **FortiSOAR Module**.
   Choosing FortiSOAR Module means that you want to use the data from your FortiSOAR instance to predict 'Phishing Emails'.
   b. From the **Module to Train For** drop-down list, select the module for which you want to train the data. For example, if you select **Alerts**, then alert records existing in your system are used to train the model, and then subsequently suggestions will be displayed on the alert records.
   c. (Optional) To add further filtering on these records, click **Add Data Filters** and add additional filters. In this case, suggestion will be shown for only those records that satisfy the filters. For example, add `Type Equals Suspicious Email`, to classify only those alerts whose type is suspicious email.
   d. From the **Date Range** drop-down list, select the time range of records based on which you want to populate the training set.
   You can select from options such as Last Month, Last 6 months, Last year, etc. You can also select **Custom** and then specify the last X days to populate the training set.
   e. The **Training Set Size** specifies the number of records that make up the training set. It is set as 1000 records.
   Note: The value that you select from the **Date Range** drop-down list overrides this parameter.
   f. In `Feature Set Mapping`, specify which fields of the module selected for suggestion ('Alerts' in our example) maps to the ML model fields. You have to map the following fields: Email From, Email Subject, and Email Body.
   g. In this case, since you are using an existing module for training, the model needs to know the classification of each email before the training. For this, you must specify which values of the field constitute as phishing and which constitute as non-phishing. Therefore, in `Verdict Field Mapping`, map the values of the picklist (field) in Phishing and Non-Phishing buckets that you want to train as the verdict of the model.
   By default, for the Alerts module a picklist named 'Email Classification' is added that has two items, Phishing and Non Phishing, which can be used by the users to classify the record:

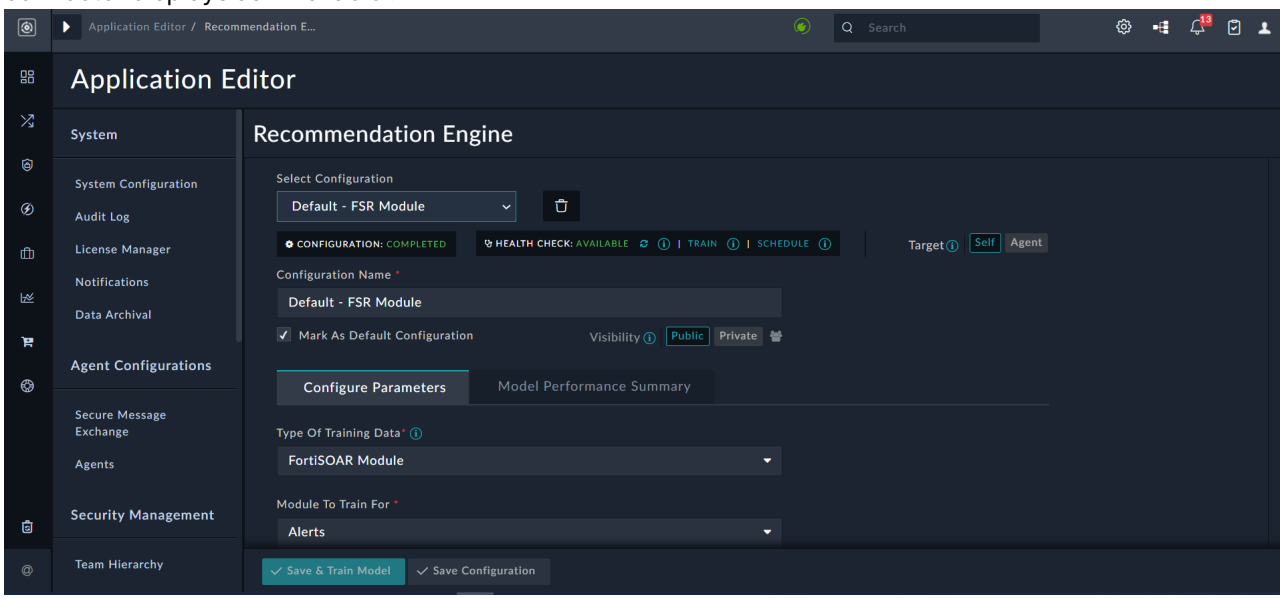


However, you can choose any other picklist (field) based on which you want to classify emails. For example, you can choose the verdict field as 'Type' and then in the 'Phishing' bucket put alerts whose type is `Brute Force Attempt`, `Denial of Service`, or `Policy Violation`, and in the 'Non Phishing' bucket put alerts whose type is `Compliance`.



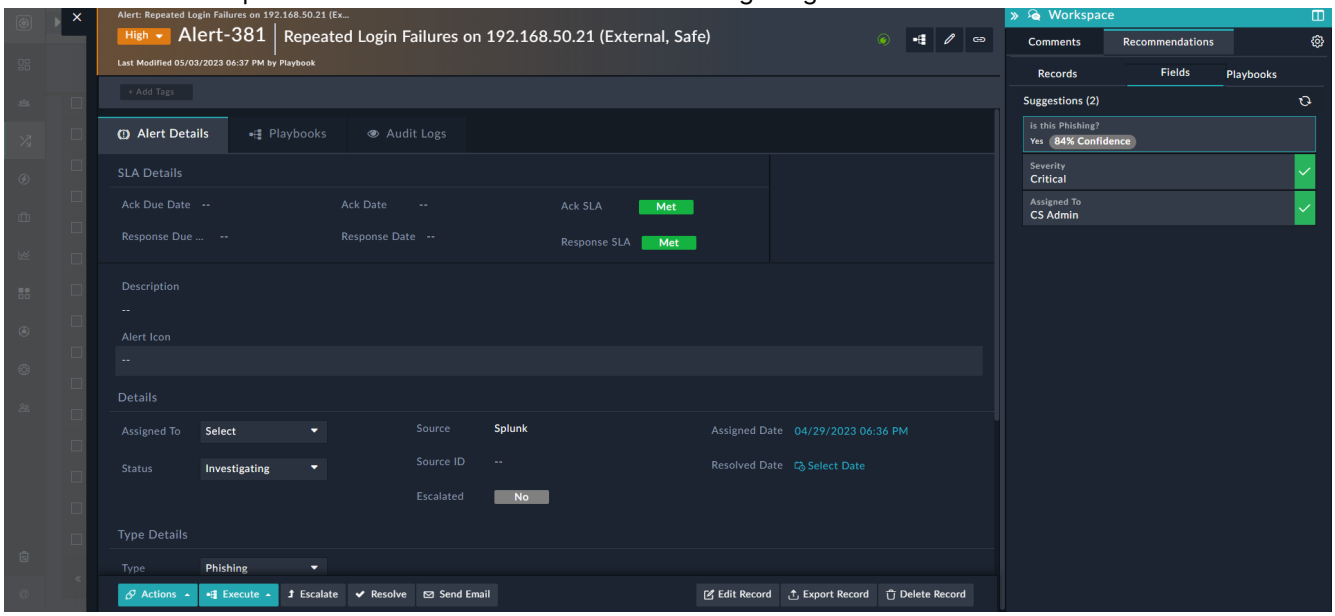
3. Once you have specified the dataset for training the Phishing Classifier connector, you can click **Save & Train Model** or **Save Configuration**.
   Clicking **Save Configuration** saves the specified parameters.
   Clicking **Save & Train Model** saves the specified parameters and also begins to train the model, 'Alerts', in our example, based to these parameters. Once the training is completed, the 'Heath Check' of the

connector displays as 'Available':



If you make any changes to the training dataset, such as adding or removing a field from the Verdict Field Mapping, click **Train** to update the dataset. To ensure that the dataset is trained on new incoming data regularly, you can also choose to train your dataset at regular intervals by scheduling training of the dataset by clicking **Schedule**. Clicking **Schedule** opens the `Schedule Details` dialog using which you can create a schedule for training your dataset. For more information on schedules, see the *Schedules* chapter in the "User Guide."

Once you have trained your dataset, FortiSOAR starts to analyze your dataset and based on the analysis displays suggestions on whether or not an alert record can be classified as a 'Phishing' record in the **Recommendations** pane > **Fields** tab as shown in the following image:



Open the detail view of an alert record, and click the **Recommendations** tab on the `Workspace` panel, to see a `Suggestion` section that contains the `Is this Phishing?` question followed by the answer to that question and the corresponding confidence level. For example, in the above image, the answer to the `Is this Phishing?` question is `Yes`, with `99%` confidence that it is a phishing email (record). Using this suggestion and its

corresponding confidence value it becomes easy for analysts to classify records into 'Phishing' and 'Non Phishing' and accordingly proceed with the investigation process.

In the case of HA environments in which a 'Takeover' operation has been performed; post-takeover, you have to retrain data on the new primary node to use the machine learning services to predict phishing suggestions.

Notes with respect to FSR agents:

- If you have installed and configured the Phishing Classifier connector on an agent node, and not on the FortiSOAR (base) node, then suggestions are not displayed in the **Recommendations** tab on the `Workspace` pane; however, you can use the agent configuration in connector actions in playbooks to get the suggestions.
- As FSR agents need to interact with modules to display suggestions, you require to modify the `FortiSOAR Agent` role to include access to the modules for which suggestions are configured. For example, if you have set up suggestions for the 'Alerts' module, then you must update the `FortiSOAR Agent` role with the minimum of `Read` permissions on the 'Alerts' module.

Advanced Settings

- By default, the ML engine runs on port 10449. If the same port is occupied by some other process in the system, then the administrator can change the port number in the `SERVER` section of the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/config/config.ini` file, and then restart the uwsgi service.
- The `TFIDF` section in the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/config/config.ini` file enables the administrator to provision controlling the frequency of a word in the corpus. A term is excluded from the feature set if it does not satisfy the `min_df` (minimum document frequency) or `max_df` (maximum document frequency). If you update the `TFIDF` section, then you must restart the uwsgi service.
  For more information, see https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
- Words that are included in the `ignore_words` field in the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/config/config.ini` file are excluded from the feature set. This allows administrators to exclude organization-specific words that should not be part of the feature set. If you make any changes to the `ignore_words` field, then you must restart the uwsgi service.
- Words that are included in the `function_words` field in the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/config/config.ini` file contain words that are generally found in phishing emails. Administrators can update words that are part of the `function_words` field. If you make any changes to the `function_words` field, then you must restart the uwsgi service.
- As part of pre-processing, unimportant words are removed from the data and the words are converted to their base forms to avoid redundancy. There are two methods to achieve this normalization: 'stem' and 'lemmatize'. By default, the Phishing Classifier uses the 'stem' method. You might want to change the normalization technique to lemmatize if you think that it might improve the suggestion accuracy. To change the normalization technique to lemmatize, do the following:
  a. Download the nltk data file on your FortiSOAR instance from https://repo.fortisoar.fortinet.com/downloads/scripts/nltk_data.tar.

b. Copy the downloaded .tar file to the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/resources/` folder.
c. Untar the nltk data file using the following command:
`tar -xvf nltk_data.tar`
d. Provide appropriate permissions to the nltk data file using the following commands:
`chmod -R 654 /opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/resources/nltk_data`
`chown -R nginx:nginx /opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/resources/nltk_data`
e. In the `/opt/cyops-integrations/integrations/connectors/phishing-classifier/ml_service/config/config.ini` file, update the value of the `word_normalization_technique` parameter from stem to lemmatize.
f. Restart the uwsgi service and retrain the model.

Export and Import Wizards

FortiSOAR provides you with a wizard-based export and import of record data, configuration information, dashboards, application settings, rules, etc., which enhances the user experience and improved architecture. These enhancements also allow for scheduled, API-based export and imports. For information about the export and import APIs, see the *API Methods* chapter in the "API Guide."

From version 7.0.2 onwards, the export wizard creates a `zip` file for all the exported content. Prior to version 7.0.2 content was exported in the `JSON` format. You can use both the zip and JSON (for content exported from a FortiSOAR system prior to 7.0.2) to import content using the Import Wizard. The zip file contains a folder for each entity that is exported, along with a json file (export.metadata.json) that contains metadata information such as the FortiSOAR version from which the content was exported, user who exported the content, datetime of when the content was exported, etc.

For example, if you have exported the alerts and incidents modules configurations, record data for the alerts and incidents modules, some dashboards, and some roles, the folder structure will be as follows:

FortiSOAR Export DatetTme folder
```
--+ modules
---+ alerts
------+ detail-layout.json
------+ form-layout.json
------+ list-layout.json
------+ mmd-layout.json
---+ incidents
------+ detail-layout.json
------+ form-layout.json
------+ list-layout.json
------+ mmd-layout.json
---+ records
------+ alerts
----------+ alerts0001.json
----+ incidents
----------+ incidents0001.json
---+ dashboards
```

```
------+ System Dashboard.json
------+ T1 Analyst.json
---+ roles
------+ T1 Analyst-<UUID>.json
------+ Security Administrator-<UUID>.json
```

Important: The record set file will have maximum 100 records.

Permissions required

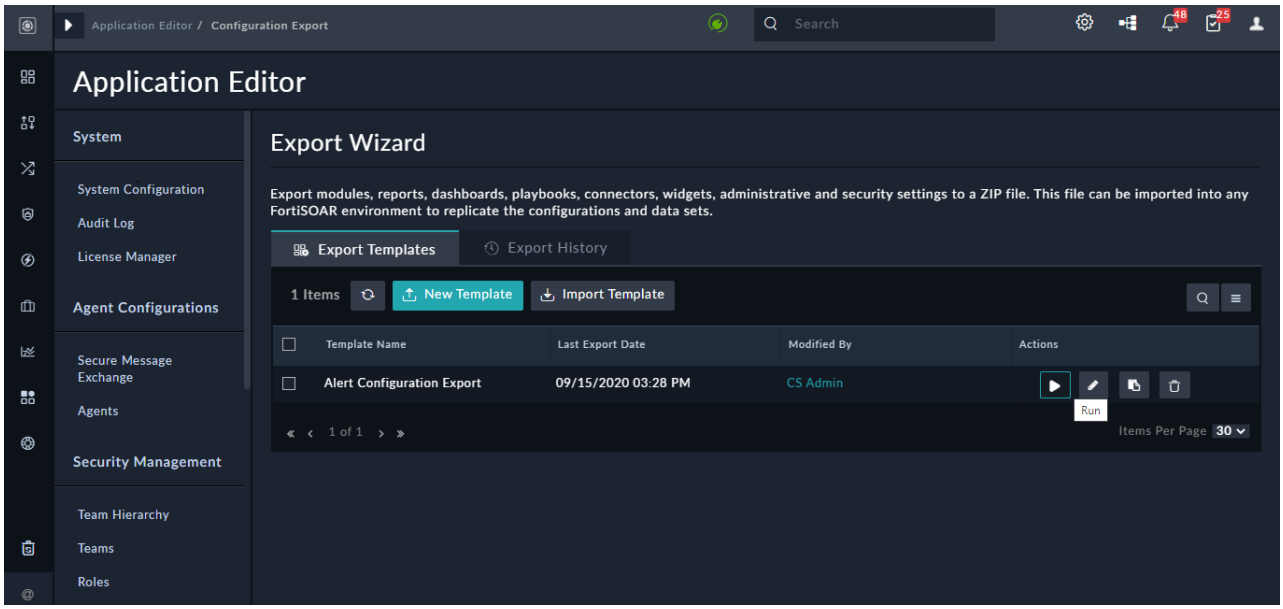
- To export and import record data and configurations using the Wizards or APIs, users who will be performing the import/export operations must be assigned a role that has `Create`, `Read` and `Update` permissions on the `Application`, `Security`, and `Playbook` modules.
  - Users who require to import files must additionally be assigned a role that has `Create` and `Read` permissions on the `Files` module.
  - Users who require to import connectors must additionally be assigned a role that has `Create`, `Read`, and `Update` permissions on the `Connectors` module.
  - Users who require to export connectors must additionally be assigned a role that has `Read` permissions on the `Connectors` module.
- If a playbook is running the import and export using the API, your playbook appliance also requires the same permissions.

Export Wizard

You can use the Export Wizard to export modules information such as record data, module metadata, field definitions, picklists, view templates, etc. of your modules. You can also export playbook collections, dashboards, reports, and administrative settings such as, application configuration, system views, export templates, etc.

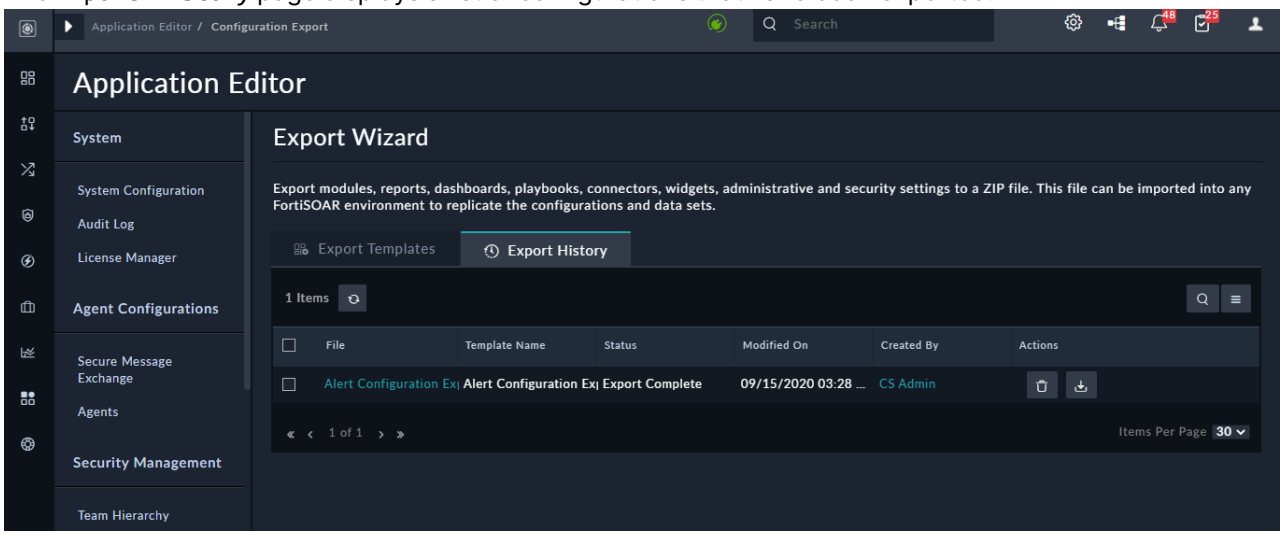
To export configurations, do the following:

1. Click **Settings** and in the `Application Editor` section, click **Export Wizard**.
   This displays the `Export Wizard` page.



To import an exported template, click the **Import Template** button

2. To begin a new export for configurations and create an export template, click the **Export Templates** tab, and then click **New Template**.
   This displays the `Choose Entities` page in the Export Wizard, in which you can choose all or any of the entities such as modules, playbooks, dashboards, etc. that you want to export.
   To run an existing configuration again, click the **Run** icon in the **Actions** column, which displays the "Run Export' screen of the Export Wizard using which you can rerun an existing configuration.
   To edit an existing configuration, click the **Edit** icon in the **Actions** column, which displays the "Choose Entities" screen of the Export Wizard using which you can edit the configurations you want to export as per your requirements. To delete an existing configuration template, click the **Delete** icon in the **Actions** column.
   If you want to use a playbook to schedule exporting configurations using an existing export template, you will require to add the UUID of the export template in the playbook. You can get the UUID of the export template by click the **Copy UUID to Clipboard** icon in the **Actions** column.
   The `Export History` page displays a list of configurations that have been exported:



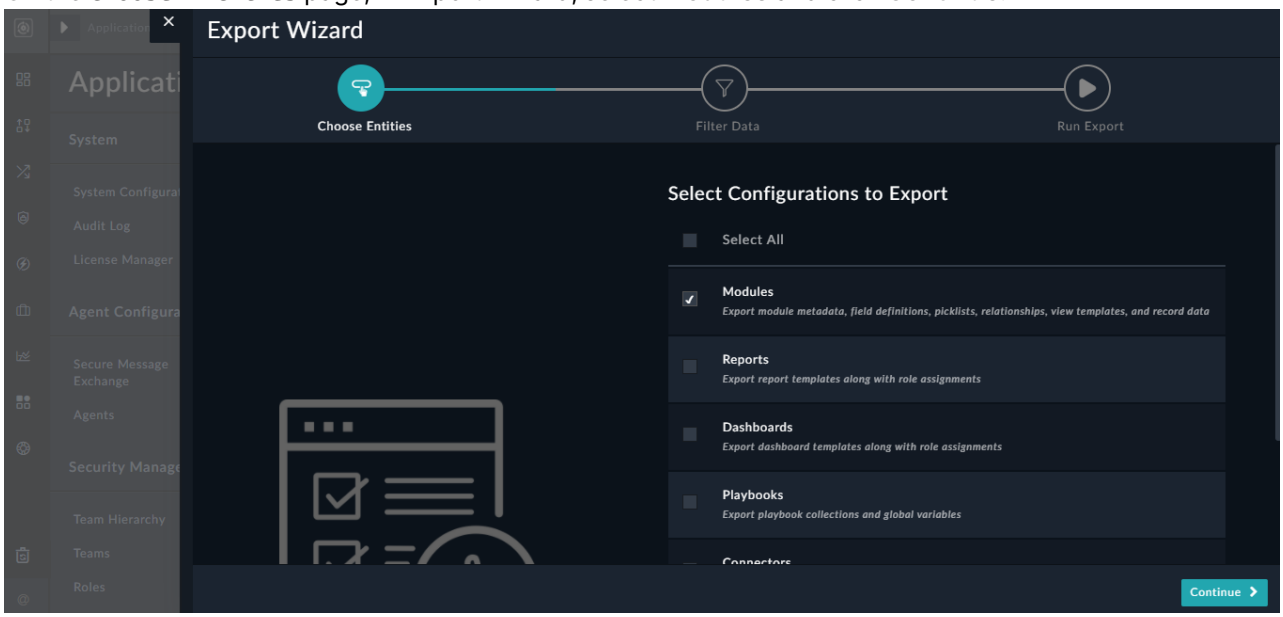
To download the exported file in the zip format, click the **Download** icon in the **Actions** column, and to delete a configuration file, click the **Delete** icon in the **Actions** column.

Exporting Modules and/or picklists

From version 7.0.2 onwards, export and import of record data is supported. The ability to export and import all data and configurations enables creating a near-exact replica of a FortiSOAR environment.

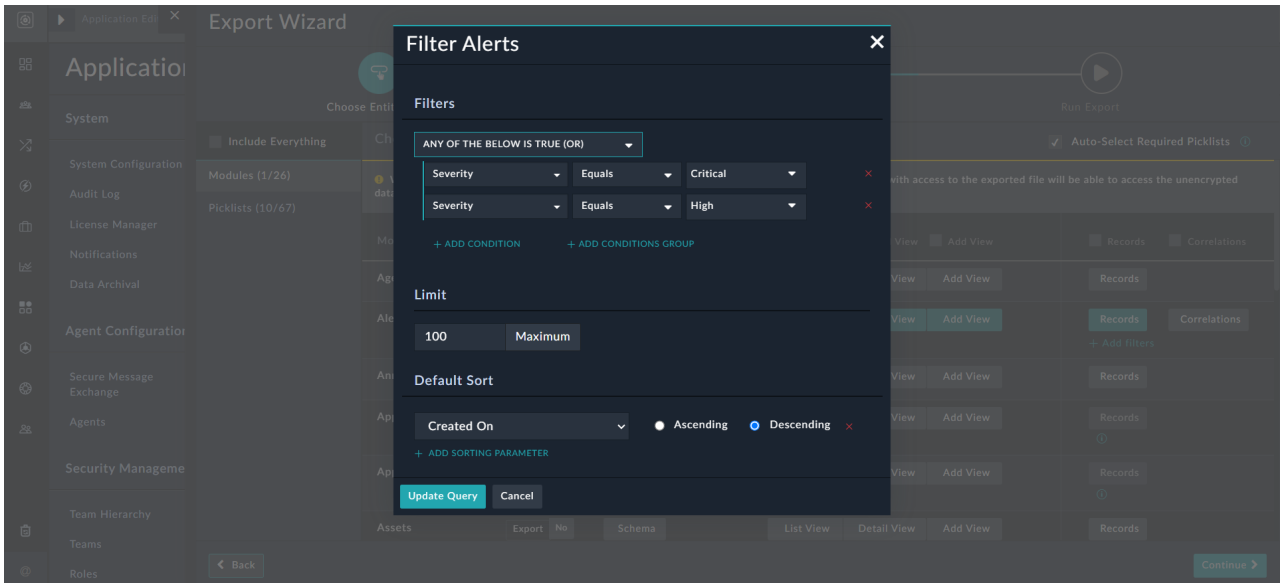
The maximum number of records that can be exported is 100000 per module. Also, note that importing a large number of records can greatly increase the duration of the import, and before you start exporting records, ensure that there is sufficient space in the `/tmp/` folder.

1. On the `Choose Entities` page, in Export Wizard, select **Modules** and click **Continue**.



   Note: You can choose to export one, all, or multiple entities.

2. On the `Filter Data` page, click the **Modules** option, and select the modules that you want to export. You can choose to export one, all, or multiple modules. You can also choose to export record data and all or any of the configuration information associated with a module, i.e., the module's schema, listing view, record view, and add views.
   NOTE: When exporting and importing data between FortiSOAR instances, it is recommended to export both the records data and its underlying schema . This is important because there could be schema differences, such as mmd modifications for mandatory fields, between instances, especially if the data is exported from an older release of FortiSOAR and imported into a newer release. If you only export the records without the schema and there are schema variations between the instances, some records may not be imported successfully.
   The **List View**, **Detail View**, and **Add View** exports the configuration information of the templates that you have created for the selected module(s). The **Records** exports the record data of the selected module(s).
   To selectively export records based on specific criteria, click the **Add Filters** link to open the `Filter <Module Name>` dialog. In this dialog, specify the conditions for filtering the records to be exported. For example, you can set a condition to export '`Alerts with High or Critical severity`' only:



From release 7.6.0 onwards, you can also filter records based on their UUID, allowing you to export specific records. You can include multiple UUIDs in a list or CSV format.

In addition to the filter criteria, you can choose to limit the number of records to be exported (with 100 as the default) and define the sorting criteria, such as '`Created On`', for the exported records. Once you have finished adding the filtering conditions, click **Update Query** to save your changes.

You can choose whether you want to export the correlation data along with the module's data. If you want to export the correlation data, then click **Correlations**.

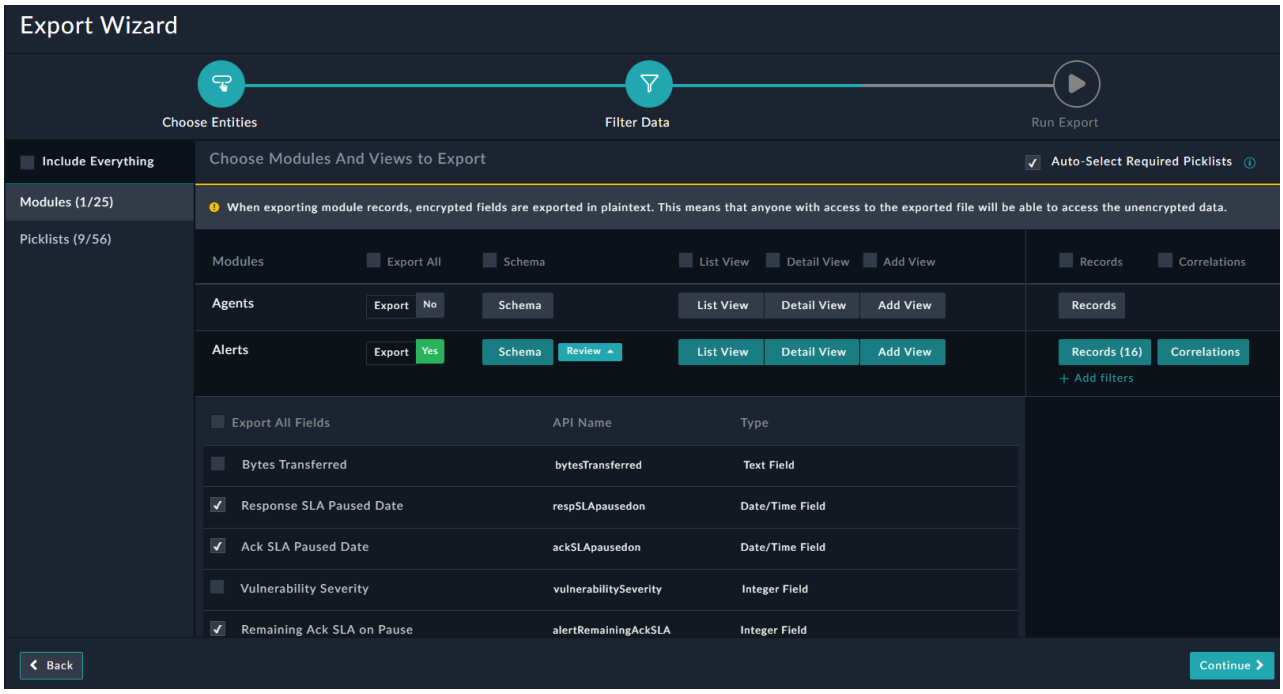
Important: When you are exporting (or importing) Queues or Shifts, you must select the Queues and Shifts module as well as their records. Queues and Shifts are exported as records and the configuration of the Queue Management page is exported using the System View Template. For more information on Queue and Shifts, see the *Queue, Shift, and Leave Management* chapter in the "User Guide."

To include all the selected entities, click the **Include Everything** checkbox. In this case, it exports the record data for all the modules and all its associated configuration information including the module's schema, views, and all the picklists.

Note: You cannot export or import record data for the system modules, i.e., the `People`, `Appliances`, and `Approvals` modules.

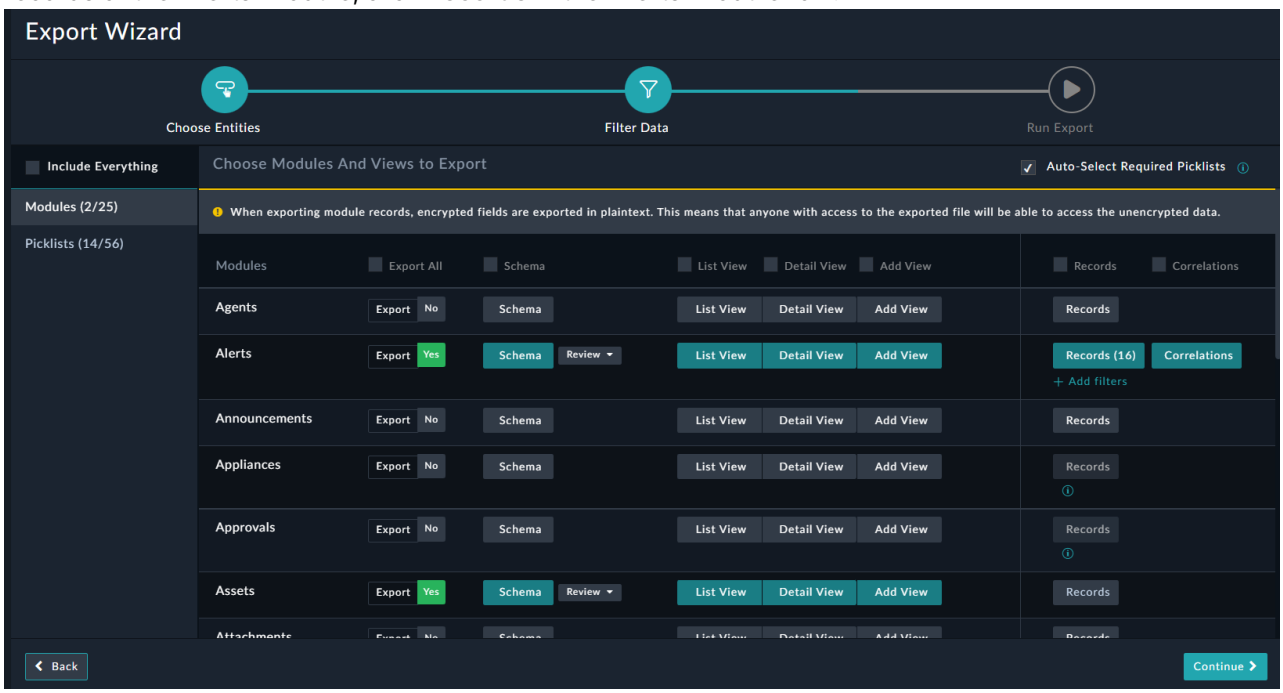
To export module configurations, choose the modules and their related configurations you want to export.

From release 7.2.0 onwards, you can choose to export selective fields from a module. Click the **Review** button to select the fields that you want to export; by default, all fields are exported. For example, if you do not want to export the 'Bytes Transferred' and 'Vulnerability Severity' fields, clear those check boxes:



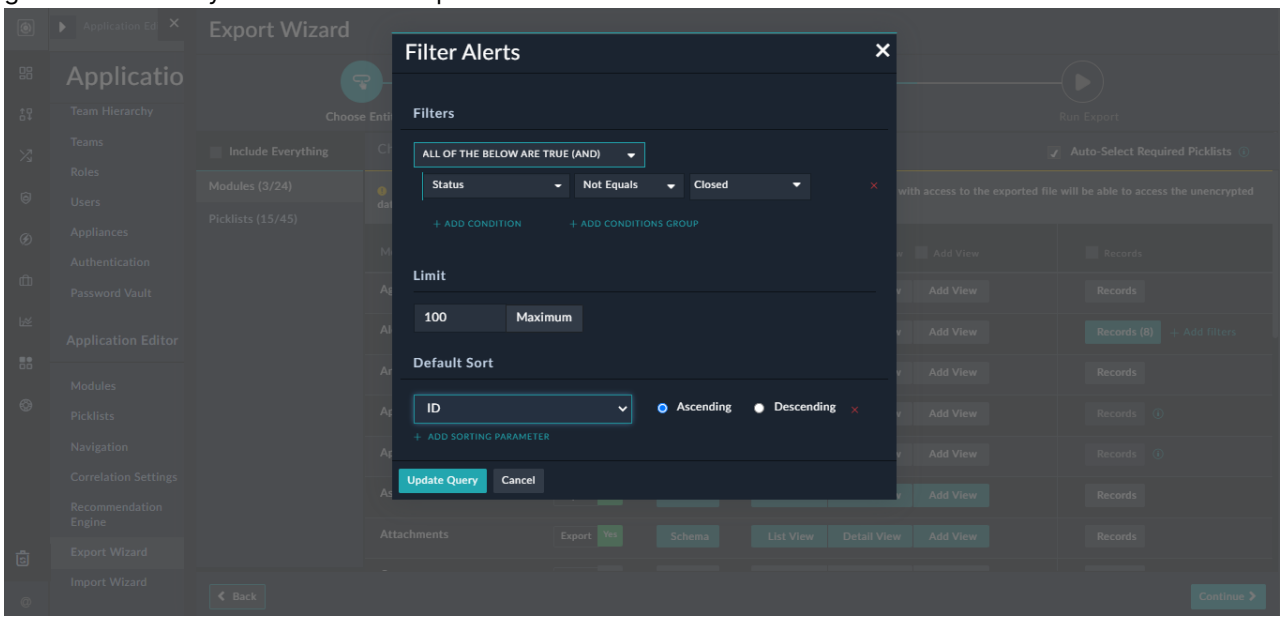
You will also observe that the **Auto-Select Required Picklists** checkbox is selected by default since the picklists associated with the module must also be exported when you are exporting the configuration information for the modules to ensure there are no issues when you import the configuration to another environment. Therefore, for example, if you select **Schema** for the "Alerts" module, you will observe that "9" picklists that are required for the "Alerts" module are automatically selected.

To export record data, click the **Modules** menu item, and in the `Choose Modules and Views to Export` table choose the modules whose record data you want to export. For example, if you want to export records of the 'Alerts' module, click **Records** in the 'Alerts' module row:



Once you click **Records**, an **Add Filters** link and a **Correlations** button are displayed. If you want to export

the records' correlations such as the records' related incidents, assets, etc., then click the **Correlations** button. You can also choose to filter the records you want to export, allowing customization of which records to export. To filter records, click the **Add Filters** link. For example, if you do not want to export alert records that are 'Closed', you can add that filter in the **Filter Alerts** dialog. You can also update the maximum number of records you want to export in the **Limit** field, the default being 100 records (maximum 100000),and also choose how you want to sort the exported data, by selecting a field from the **Default Sort** drop-down list. Once you have competed filtering the records, click **Update Query**.
Note: If a record set is included in the export, then the module schema for that record set is required and gets automatically included in the export.



Click the check boxes in the header row to perform bulk actions. For example, clicking the **Export All** checkbox selects all the modules their associated configurations, but not their record data. Similarly, clicking the **Schemas** checkbox in the header row changes **Export** to **Yes** for all modules and selects the schema for all the modules.
To enable export of configurations and record data for a particular module, in that module's row, toggle the **Export** button to **Yes**, which selects all the configuration information associated with a module, i.e., the module's schema, listing view, record view, and add views. If you do not want to export some configuration information, for example, add view, toggle the **Add View** button to disable exporting the add view configuration.
Note: The export and import wizards do not take care of data replication settings for modules. For example, in the case of an MSSP system, on a tenant node, if you have set up data replicatio, by selecting the **Enable Multi-Tenancy** checkbox, for the 'Comment' Module, and if you export this module and import it to another system, you will observe that the data replication flag is not set for the 'Comment' module.
If you want to export only picklists, click the **Picklists** menu item, and select the picklists you want to export. Using this menu item, you can export the picklists that are not associated with any module.
Note: When you import a picklist, by clicking **Import Wizard** and if that picklist already exists on your system, then the "Import Wizard" replaces the existing picklist.
Once you have completed choosing the modules and picklists that you want to export, click **Continue**.

3. On the `Review Export` page, you can review the record data and configuration information that you are exporting and can also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the specified

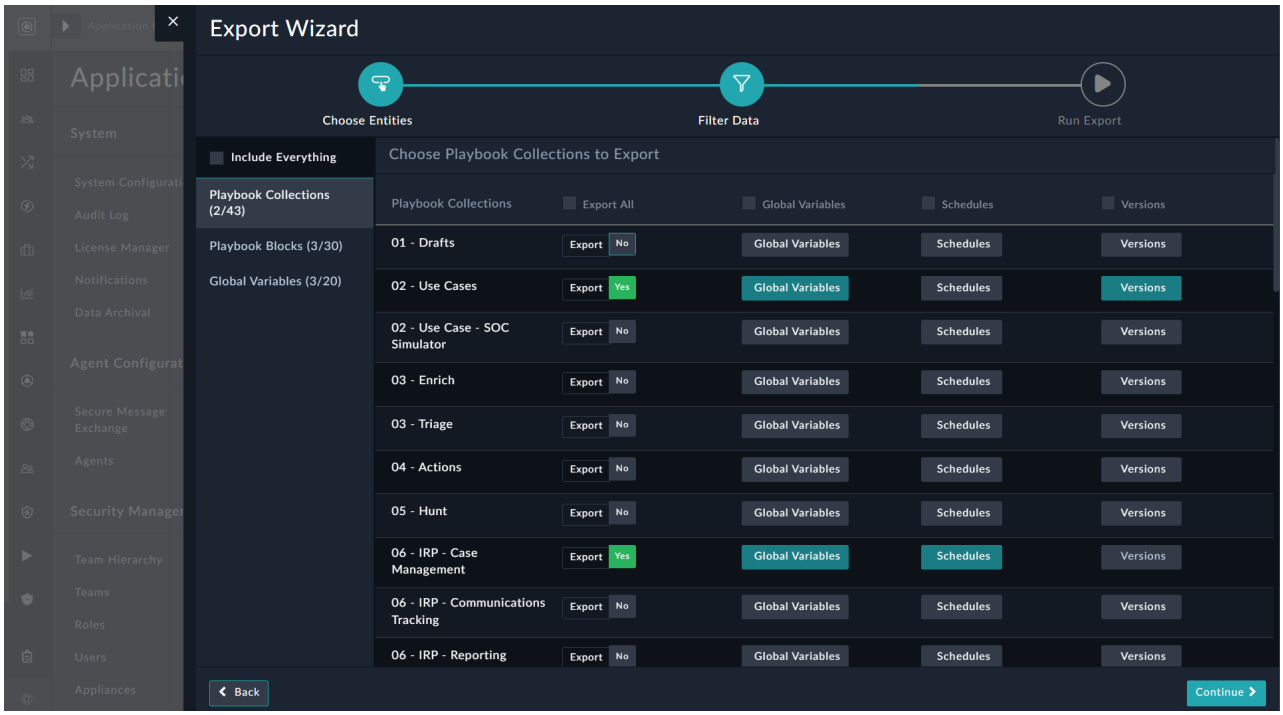
configuration information in a `zip` file that you can download and use in another environment, or click **Save** to save the configuration information.
FortiSOAR also displays warnings if there are any inconsistencies in the data, such as templates not found, to be exported. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the specified configuration information as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard in which you can edit the configurations you want to export as per your requirements.

Exporting Playbooks, Schedules, Global Variables, and/or Playbook Blocks

You can export playbook collections, global variables, and playbook blocks (Reference Blocks). FortiSOAR release 7.4.0 introduces 'Reference Blocks', which provide a useful set of references for users to ease their playbook building experience. For more information on Reference Blocks, see the "Playbooks Guide." Currently, you have to export the complete playbook collection and cannot select specific playbooks to be exported from within a playbook collection. It should be noted that child reference playbooks are exported along with their parent playbook collection, even if they are present in a playbook collection other than the parent playbook collection.

From release 7.4.2 onwards, when you are exporting playbook collections, you can choose whether schedules associated with playbooks should also be exported. This enhancement ensures that users know which schedules are being imported and installed on their system. Previously, if an exported playbook was included in a schedule, the schedule would be implicitly exported without any indication of the same on the FortiSOAR UI, which could cause issues in users' systems, especially for solution packs, where schedules would become a part of the pack unintentionally and get installed on systems installing the pack.

1. On the `Choose Entities` page, in the Export Wizard, select **Playbooks** and click **Continue**.
2. On the `Filter Data` page, select the playbook collections, playbook blocks, and global variables that you want to export.
   On the `Choose Playbook Collections to Export` page, to export specific playbook collections, in the specific `Playbook Collections` row, toggle **Export** to **Yes** to select that playbook collection. This includes all the global variables, schedules, and versions associated with that particular playbook collection in the export. You can choose not to export global variables, schedules, and versions associated with that particular playbook collection by deselecting global variables, schedules and versions in the row of the playbook collection. For example, as displayed in the following image, the 02 - Use Cases and 06 - IRP - Case Management playbook collections are being exported, where 'Schedules' associated with the 02 - Use Cases playbook collection and the 'Versions' associated with the 06 - IRP - Case Management playbook collection are not being exported:



To export all the playbook collections, including all associated global variables, schedules, and versions, click the **Export All** checkbox.

To include all global variables used in the playbook collections as part of the export, while exporting playbook collections, click the **Global Variables** checkbox. If you want to export only some of the global variables used in the selected playbooks, then clear the **Global Variables** checkbox, and then individually select the required global variables from the **Global Variables** menu. You might not want to import all global variables in the following cases:

- When you do not want the global variables' values to be reset. For example, you do not want to reset the `server_fqhn` that is used in all notification emails, etc. Do note that from release 7.6.0 onwards, if the global variable already exists on your system, then it is not imported on that system.
- When you want the global variables values to be different in development and production environments in the case of CI-CD.

To include all schedules associated with your playbooks while exporting playbook collections, click the **Schedules** checkbox.

To include all versions of your playbooks while exporting playbook collections, click the **Versions** checkbox.

To include all the selected entities, click the **Include Everything** checkbox. In this case, it exports all the playbook collections, global variables, playbook blocks, and schedules.

Click the **Playbook Blocks** menu, and click the **Playbook Blocks Name** checkbox to select or deselect all the playbook blocks to be exported. To export specific playbook blocks, select those playbook blocks. To include all global variables used in the selected block as part of the export, click the **Include Global Variables** checkbox. However, if you want to export only specific global variables, then go to the `Global Variables` section and select those specific variables.

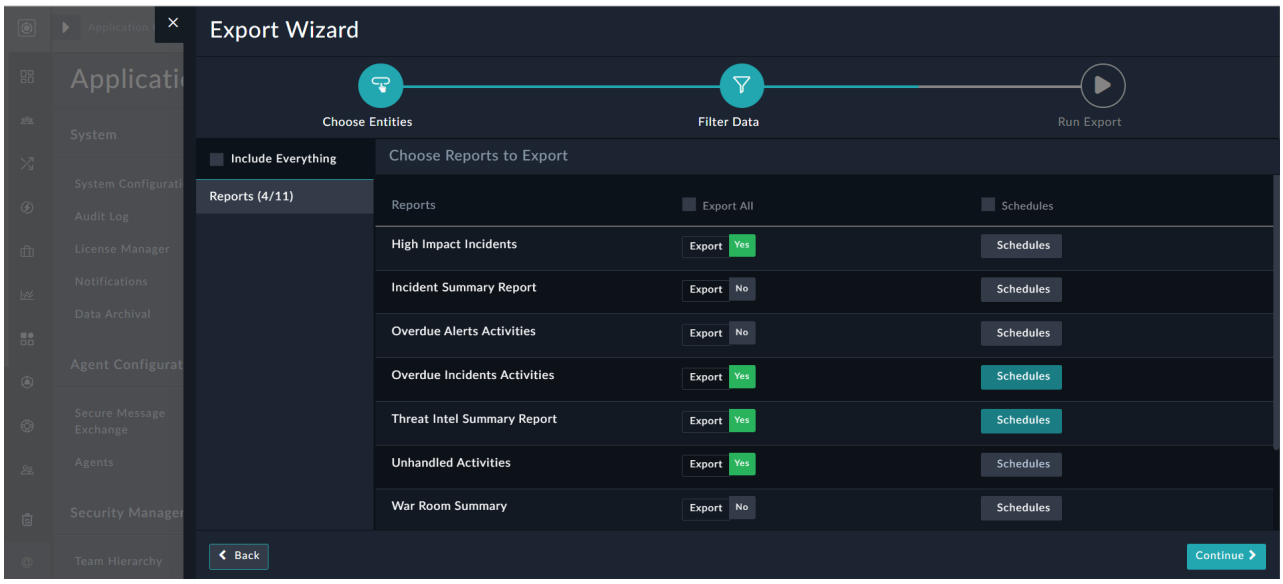
Likewise, click the **Global Variables** menu, and click the **Global Variable Name** checkbox to select or deselect all the global variables to be exported. To export specific global variables, select those global variables.

Once you have completed choosing the playbook collections, schedules, global variables, and/or playbook blocks that you want to export, click **Continue**.

3. On the `Review Export` page, review the playbook collections, global variables, and playbook blocks that you are exporting, and also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the playbook collections, schedules, global variables, and playbook blocks in a `zip` file that you can download and use in another environment, or click **Save** to save the playbook collections, global variables, schedules, and playbook blocks. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the playbook collections, global variables, and playbook blocks configuration as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.
   Note: When you import a playbook collection by clicking **Import Wizard**, and if that playbook collection exists, you can choose to either merge collections (skip existing playbooks) (default), merge collections (replace existing playbooks), rename the existing playbook collections, or rename the existing playbook collections by appending the original playbook collection name with a number. For more information, see Importing Configurations.
   Important: When you import a global variable by clicking **Import Wizard**, and if that global variable already exists on your system, then by default, the "Import Wizard" does not replace the existing global variable. When you import a playbook block, by clicking **Import Wizard** and if that playbook block already exists on your system, then by default, the "Import Wizard" does not replace the existing playbook block, i.e., the **Skip Current Block** is selected. You can choose **Replace Existing Block**, which will replace the playbook block existing on your system with the new playbook block.

Exporting Reports and associated Schedules

1. On the `Choose Entities` page, in the Export Wizard, select **Reports**, and then click **Continue**.
2. On the `Filter Data` page, select the reports that you want to export.
   In the `Choose Reports to Export` page, to export specific reports, in the specific `Report` row, toggle **Export** to **Yes** to select that report. This includes all the schedules associated with that particular report in the export. You can choose not to export schedules and versions of a particular report by deselecting Schedules and Versions in the row of the report. For example, in the following image, schedules for the High Impact Incidents report and the Unhandled Activities report will not be included:



To export all the reports and their associated schedules, click the **Export All** checkbox. To export the schedules, click the **Schedules** checkbox.
Once you have completed choosing the reports that you want to export, click **Continue**.

3. On the `Review Export` page, you can review the reports that you are exporting, and can also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the dashboards, reports, rules, and rule channels in a `zip` file that you can download and use in another environment, or click **Save** to save the dashboards, reports, rules, and rule channels. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the report template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.
   Note: When you import dashboards, reports, rules, or rule channels, by clicking **Import Wizard**, and if that dashboard or report already exists on your system, then the "Import Wizard" replaces the existing dashboard or report.

Exporting Dashboards, Delivery Rules, Rule Channels, and Pre-Processing Rules

1. On the `Choose Entities` page, in the Export Wizard, select **Dashboards**, **Rules & Channels**, and **Pre-processing Rules**, and then click **Continue**.
2. On the `Filter Data` page, select the dashboards, delivery rules, rule channels, and pre-processing rules that you want to export.
   Click the **Dashboards** menu item, and in the `Choose Dashboards To Export` table, click the **Dashboard Name** checkbox to select or deselect all the dashboards. To export specific dashboards, select those dashboards.
   Click the **Delivery Rules** menu item, and in the `Choose Delivery Rules To Export` table, click the **Rule Name** checkbox to select or deselect all the delivery rules. To export specific delivery rules, select those

delivery rules.
Similarly, click the **Rule Channels** menu item, and in the `Choose Rule Channels To Export` table, click the **Rule Channel Name** checkbox to select or deselect all the rule channels. To export specific rule channels, select those rule channels.
Click the **Pre-processing rules** menu item, and in the `Choose Pre-processing Rules To Export` table, click the **Rule Name** checkbox to select or deselect all the pre-processing rules. To export specific pre-processing rules, select those pre-processing rules.
To include all the selected entities, click the **Include Everything** checkbox. In this case, it exports all the dashboards, delivery rules, rule channels and pre-processing rules.
Once you have completed choosing the dashboards, delivery rules, rule channels, and/or pre-processing rules that you want to export, click **Continue**.

3. On the `Review Export` page, you can review the dashboards, delivery rules, rule channels, and pre-processing rules that you are exporting, and can also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
Once you have completed reviewing the information, click **Save & Run Export** to export the dashboards, delivery rules, rule channels, and pre-processing rules in a `zip` file that you can download and use in another environment, or click **Save** to save the dashboards, delivery rules, rule channels, and pre-processing rules. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the dashboards, delivery rules, rule channels, and pre-processing rules templates as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.
NOTE: When you import dashboards, delivery rules, rule channels, or pre-processing rules, by clicking **Import Wizard**, and if those dashboards, delivery rules, rule channels, or pre-processing rules already exist on your system, then the "Import Wizard" replaces the existing dashboards, delivery rules, rule channels, or pre-processing rules.

Exporting Connectors

You can export connectors that are installed on your system. You can export connector installation and their configurations. From version 7.0.2 onwards, you can also export and import `.tgz` files of widgets and connectors. If a connector version is not found in the global connector repository, then the export wizard will export the `.tgz` file of the connector instead of the 'rpm' name. Similarly, if a widget version is not found in the widgets repository, the export wizard will export the `.tgz` file for the widget.

Password and API keys are NOT encrypted during export, which means that anyone who has access to the exported file will be able to access the connectors. Therefore, you must be careful while exporting the connector configurations.

1. On the `Choose Entities` page, in Export Wizard, select **Connectors** and click **Continue**.
2. On the `Filter Data` page, select the connectors that you want to export. You can choose to export one, all, or multiple connectors. You can also choose to export the configuration information associated with a connector.
Click the **Connectors** menu item, and in the `Choose Connector To Export` table, select the connectors that you want to export. To export both the installation and the configuration for a particular connector, in that connector's row, toggle the **Export** button to **Yes**. If you want to export only the configuration for a

connector, then toggle the **Installation** button to disable exporting the installation for that connector, or toggle the **Configurations** button to disable exporting the configurations.
Click the checkboxes in the header row to perform bulk actions. For example, clicking the **Export All** checkbox, selects all the connectors their associated configurations. Similarly, clicking the **Configuration** checkbox in the header row, changes **Export** to **Yes** for all connectors and selects the configurations for all the connectors.
To include all the selected entities, click the **Include Everything** checkbox. In this case it exports the installations and configurations for all the connectors.
Once you have completed choosing the connectors that you want to export, click **Continue**.

3. On the `Review Export` page, you can review the connectors that you are exporting, and can also specify the name of the template that you are exporting, as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the connectors in a `zip` file that you can download and use in another environment, or click **Save** to save the connectors.
   If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the connector template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.

Exporting Widgets

You can export widgets that are installed on your system. From version 7.0.2 onwards, users can also export and import `.tgz` files of widgets and connectors. If a widget version is not found in the widget repository, then the export wizard will export the `.tgz` file for the widget.

1. On the `Choose Entities` page, in Export Wizard, select **Widgets** and click **Continue**.
2. On the `Filter Data` page, select the widgets that you want to export. You can choose to export one, all, or multiple widgets.
   Click the **Widgets** menu item, and in the `Choose Widgets To Export` table, select the widgets that you want to export, and click **Continue**.
3. On the `Review Export` page, you can review the widgets that you are exporting, and can also specify the name of the template that you are exporting, as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the widgets in a `zip` file that you can download and use in another environment, or click **Save** to save the widgets.
   If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the widget template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.

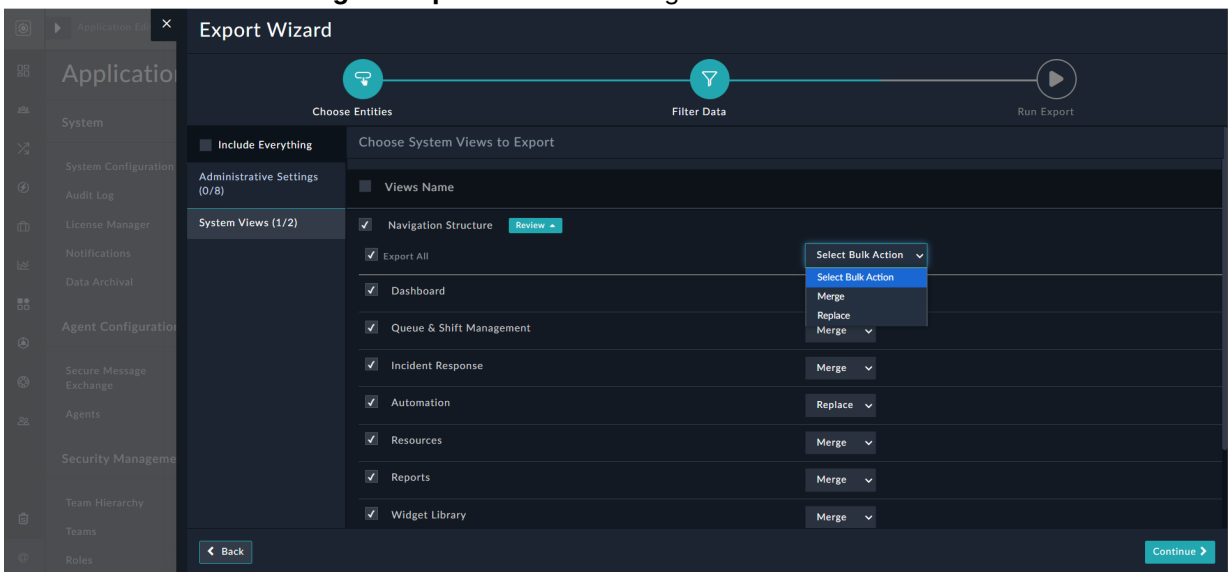
Exporting Administrative Settings and System Views

You can export system views and administrative settings, including customizations applied across your FortiSOAR instance. This includes application settings such as branding and notifications, SSO, LDAP, Radius

configurations, proxy settings, environment variables, etc.

Passwords are write-only fields and therefore they cannot be exported using Configuration Manager. Therefore, if you export your LDAP configurations and import that into another FortiSOAR system, passwords are not copied, and so you must manually enter the passwords for all users to perform any user-related activities, such as searching for users or updating user details.

1. On the `Choose Entities` page, in the Export Wizard, select **Administrative Settings** and click **Continue**.
   a. On the `Filter Data` page, select the roles and/or settings that you want to export.
   Click the **Administrative Settings** menu item. In the `Choose Administrative Settings To Export` table, in the `Administrative Settings` section, click the **Settings Name** checkbox to select or deselect all administrative settings. To export specific administrative settings, select those individually. Similarly, in the **System Views** menu item and click the **Views Name** checkbox to select or deselect all the system views. To export specific system views, select those individually.
   You can choose to customize the **Navigation Structure** you want to export. Click the **Review** button to display the items included in the navigation. By default, all navigation items are selected for export. Deselect the **Export All** checkbox to choose items individually. Additionally, from release 7.6.1 onwards, you can choose to either **Merge** (Default) or **Replace** the navigation items. Merge appends changes, while Replace overwrites all the navigation items.
   You can also choose to **Merge** or **Replace** selected navigation items:



   To include all entities in the export, click the **Include Everything** checkbox. In this case it exports all the system views and administrative settings.
   Once you have completed choosing the settings, and administrative settings that you want to export, click **Continue**.
2. On the `Review Export` page, you can review the settings that you are exporting and can also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the settings/views in a `zip` file that you can download and use in another environment, or click **Save** to save the settings. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as

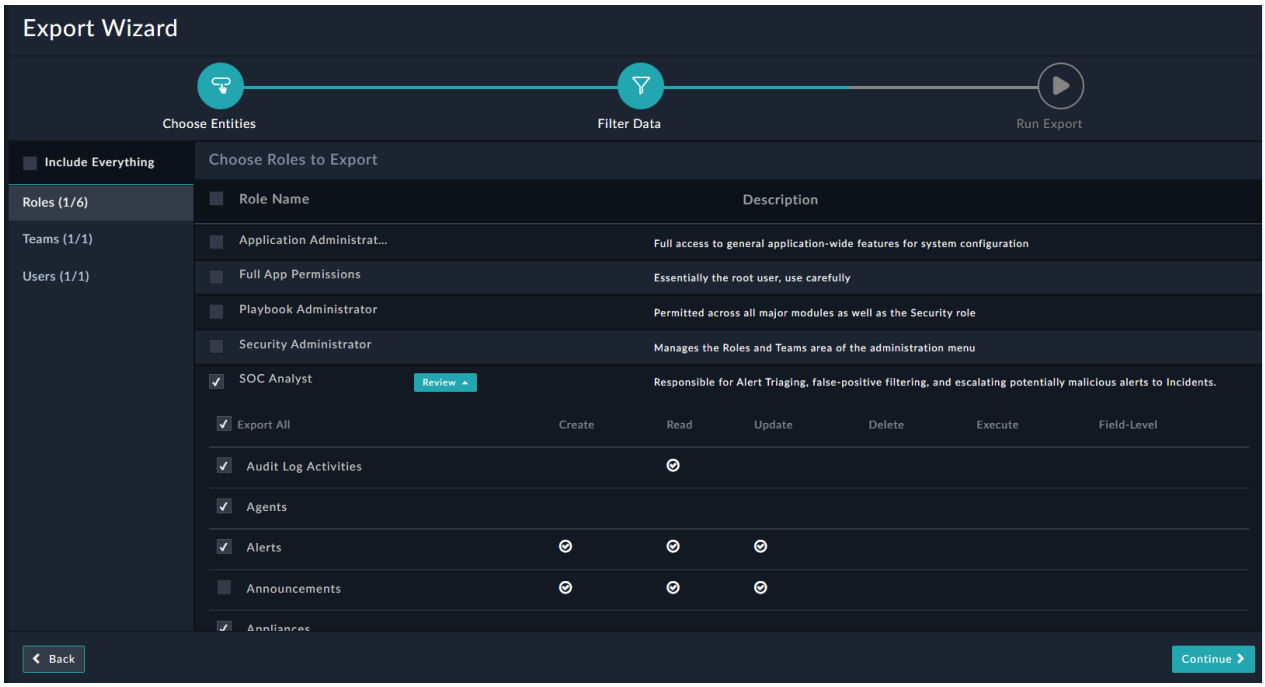
an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the settings template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.
When you import the exported configurations into a system, all the application settings that were applied on the system from which the application settings were exported get applied on the system where you import and install the settings. For example, if the system from which the application settings were exported had its "Audit Log Purge" enabled with the logs to be retained for the last month, the same Audit Log policy will apply on the system in which you import and install the application settings.
Important: If you have exported your SSO configuration and imported the SSO (SAML) configurations into a different FortiSOAR system, you require to make certain updates before SAML users can log into FortiSOAR. For more information, see Updates required to be done after importing SSO configurations.
Note: When you import the queue management configuration system view or any administration setting using the **Import Wizard**, if the configuration or setting already exists on your system, the Import Wizard will overwrite the same. For the navigation structure, you can choose to either merge or replace the existing structure. Additionally, at the individual navigation item level, you can selectively choose to Skip, Merge, or Replace the navigation item.

Exporting Security Settings

You can export security settings, which includes users, teams, and roles, that are present in your FortiSOAR instance.

From version 7.0.1 onwards, access type information, i.e., named or concurrent, is also exported along with the other user details, and the same are accordingly imported. If users are being imported from a version earlier than 7.0.1, i.e., 7.0.0, then those users do not have any access type information, and in this case, these users are imported as 'named' users.

1. On the `Choose Entities` page, in the Export Wizard, select **Security Settings** and click **Continue**.
2. On the `Filter Data` page, select the roles, teams, or users that you want to export.
   To export roles, click the **Roles** menu item, and in the `Choose Roles To Export` table, click the **Role Name** checkbox to select or deselect all the roles.
   To export specific roles, select those roles. You can export roles such as **Full App Permissions**, **Application Administrator**, **T1 Analyst**, **Security Administrator**, etc.
   From version 7.0.2 onward, you can choose to export specific module access for a particular role. For example, if you want to export the 'SOC Analyst' role, without having access to the 'Announcements' module, select the **SOC Analyst** role, then click the **Review** button, to display all the module permissions associated with this role. Clear the **Announcements** checkbox to remove the permissions associated with the Announcements module. When you import this role to another FortiSOAR system the 'Announcements' permissions for the SOC Analyst role will be removed:



To export teams, click the **Teams** menu item, and in the `Choose Teams To Export` table, click the **Team Name** checkbox to select or deselect all the teams. To export a specific team, select that team.
Similarly, to export users, click the **Users** menu item, and in the `Choose Users To Export` table, click the **Name** checkbox to select or deselect all the users. To export a specific user, select that user.
To include all the selected entities, click the **Include Everything** checkbox. In this case it exports all the roles, teams, and users.
Once you have completed choosing the roles, teams, and users that you want to export, click **Continue**.

3. On the `Review Export` page, you can review the roles, teams, and users that you are exporting, and can also specify the name of the template that you are exporting as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the roles, teams, and users in a `zip` file that you can download and use in another environment, or click **Save** to save the settings/roles. If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the roles, teams, and users template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.
   Note: When you import a role, user, or team by clicking **Import Wizard**, and if that role, user, or team already exists on your system, the Import Wizard will overwrite the existing role, user, or team.

Exporting Export Templates

The Export Wizard in release 7.6.0 has been improved to facilitate the export and import of 'Export Templates' such as those provided by the Continuous Delivery Solution Pack.

1. On the `Choose Entities` page, in Export Wizard, select **Export Template** and click **Continue**.
2. On the `Filter Data` page, select the export templates that you want to export. You can choose to export one, all, or multiple export templates. Click **Include Everything** to export all the export templates.
   Click the **Export Template** menu item, and in the `Choose Export Template` table, select the export templates that you want to export, and click **Continue**.
3. On the `Review Export` page, you can review the export templates that you are exporting, and can also specify the name of the template that you are exporting, as well as the name of the `zip` file that you want to export. If you change the template name, the file name automatically gets updated as per the template name specified.
   Once you have completed reviewing the information, click **Save & Run Export** to export the export templates in a `zip` file that you can download and use in another environment, or click **Save** to save the widgets.
   If you have clicked **Save & Run Export**, then the record of the export configuration that has been run is added as an entry in both the **Export Templates** and **Export History** pages. If you have clicked **Save**, then FortiSOAR saves the widget template as a record entry only in the **Export Templates** page. You can edit this configuration at any time by clicking the **Edit** icon in the **Actions** column, which again displays the Export Wizard, using which you can edit the configurations you want to export as per your requirements.

Import Wizard

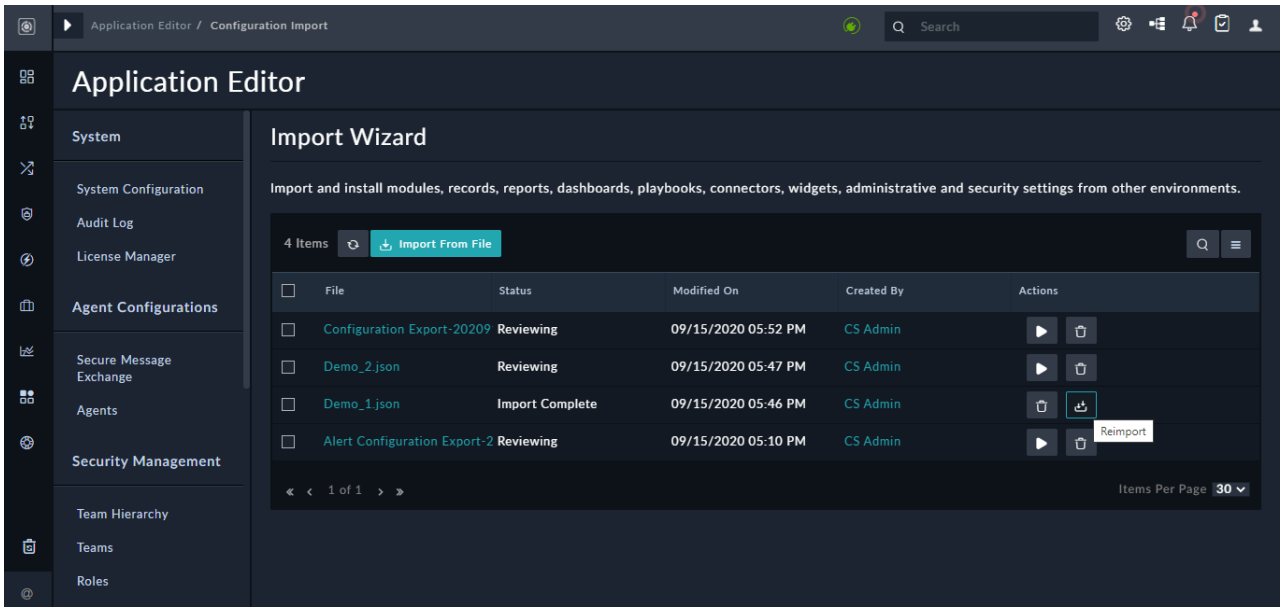
You can use the import wizard to import record data, configurations or metadata information for modules, playbook collections, dashboards, etc. from other environments into FortiSOAR. Using the import wizard, you can move model metadata, picklists, system view templates, dashboards, reports, roles, rules, playbooks, and application settings across environments.

Importing configurations

The following section provides an example of importing modules, dashboards, reports, system views, playbook collections, etc.

To import configurations into FortiSOAR the configurations file must be in the `JSON/ZIP` format. FortiSOAR ensures that you either revert or publish staged changes prior to importing configurations so that there are no issues during the import process.

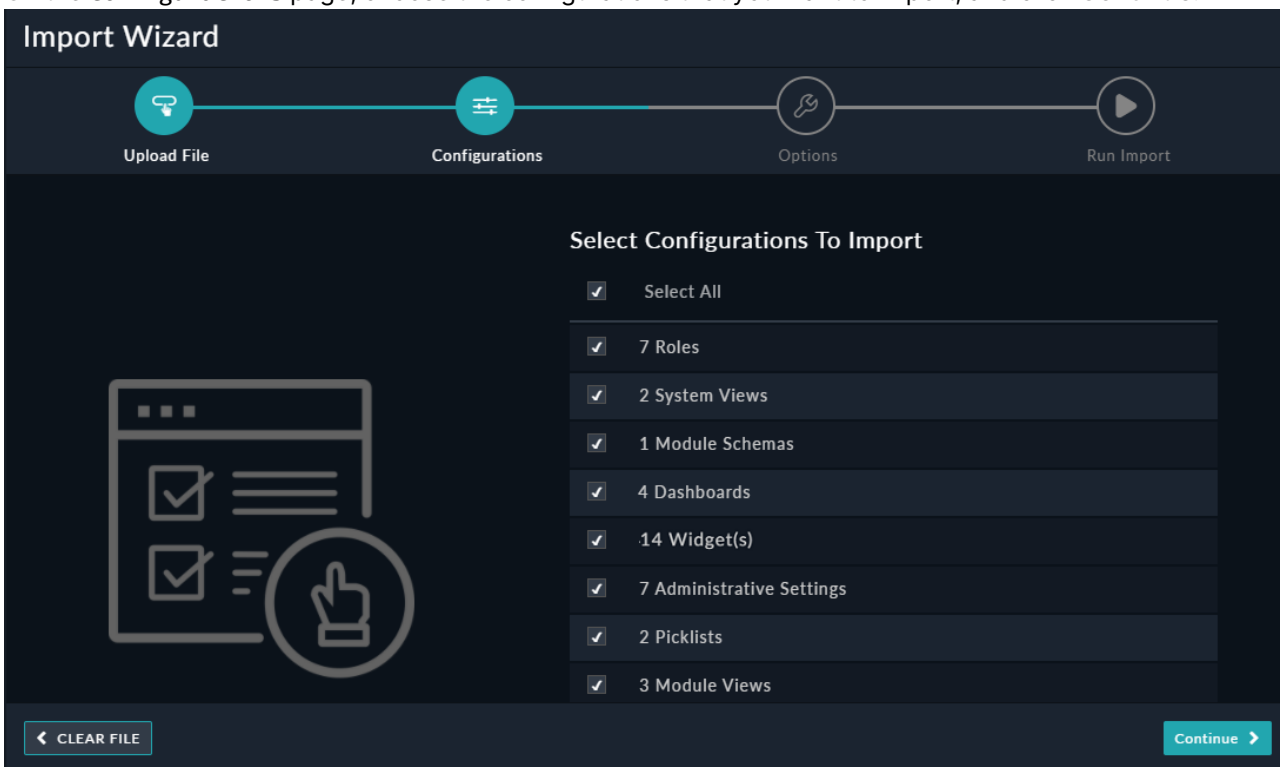
1. Click **Settings** and in the `Application Editor` section, click **Import Wizard**.
   This displays the `Import Wizard` page.



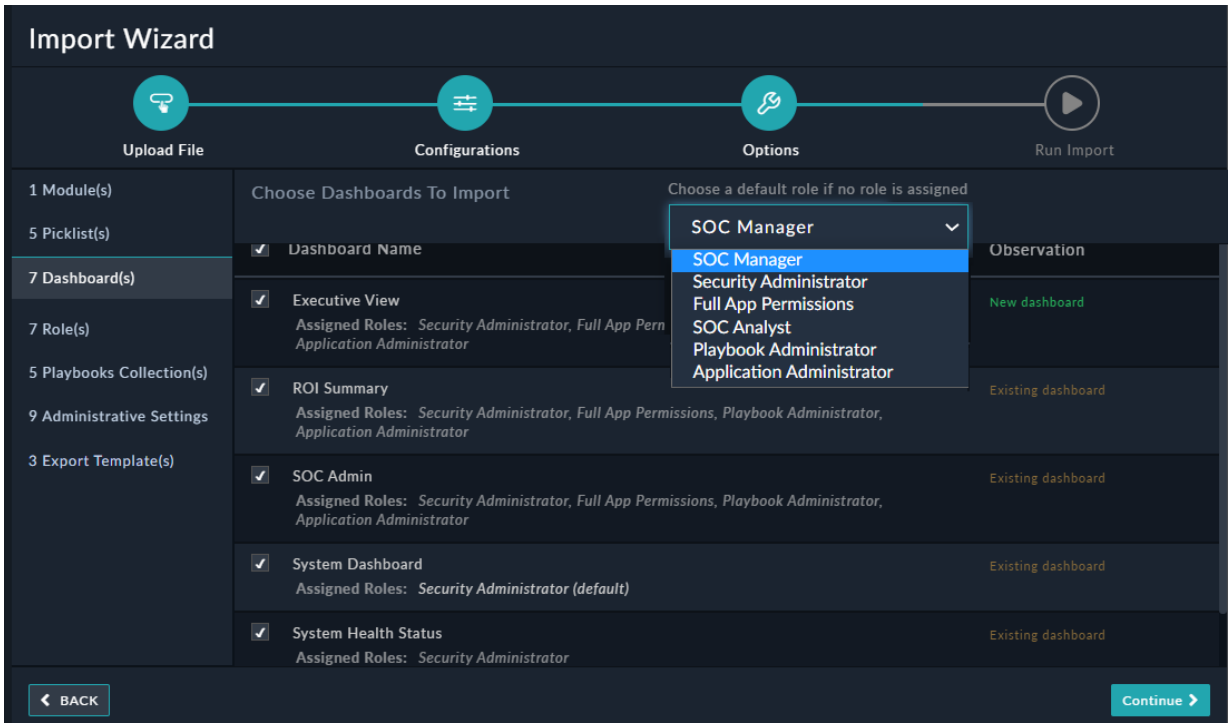
If you close the wizard without clicking **Run Import**, then the status of your import will display as "Reviewing", and you can click the **Continue** icon in the **Actions** column to display the "Configurations" screen of the Import Wizard, and you can continue review of the import configurations. If you have clicked **Run Import**, and the import process is completed, then the status of your import will display as "Import Complete". You can also the configuration at any time by clicking the **Reimport** icon in the **Actions** column to display the "Configurations" screen of the Import Wizard.

2. Click **Import From File**.
   This displays the `Upload File` page in **Import Wizard**. On this page, drag and drop the JSON or ZIP file, or click the **Download** icon and browse to the JSON or ZIP file to import configurations into FortiSOAR. If the JSON format is incorrect, FortiSOAR displays an error message and not import the file. If the JSON format is correct, FortiSOAR imports the configurations and displays details of what is being imported on the `Configurations` page.

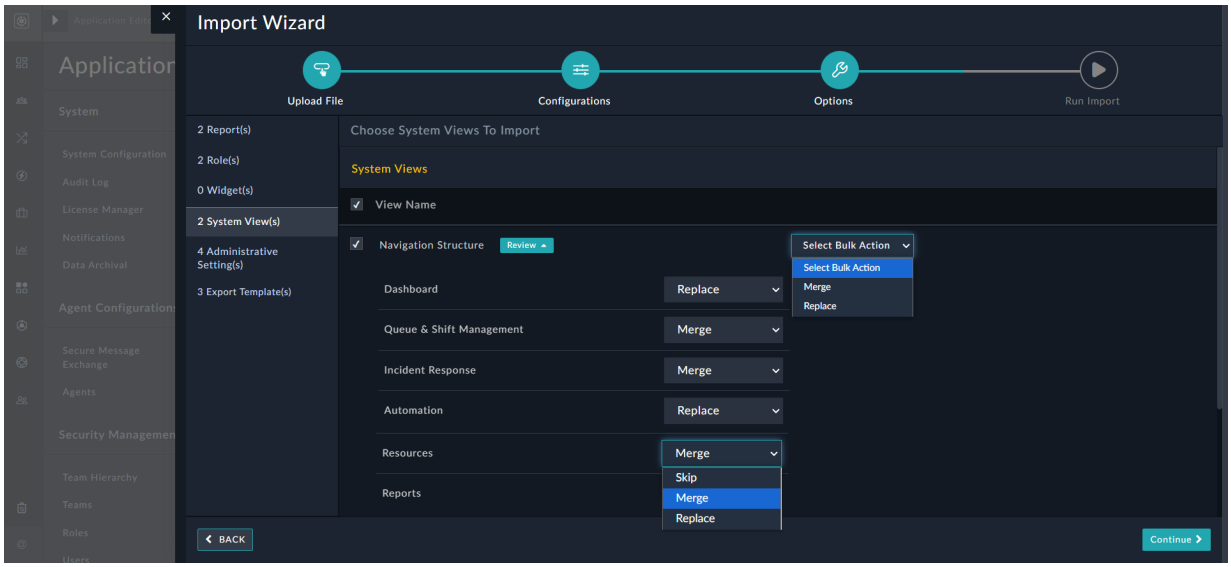
3. On the `Configurations` page, choose the configurations that you want to import, and click **Continue**.



   a. **Importing Dashboards, Widgets, Delivery Rules, Rule Channels, Pre-processing Rules, System Views, Administrative and Security (User, Team, Role) Settings, and Export Templates**:
   To import any entity (Dashboards, Delivery Rules, Rule Channels etc) on the `Options` page, click the respective menu item. The "Observation" column displays whether the entity that you are importing is "New" or "Existing". Click the **Dashboard Name**checkbox to select or deselect all the dashboards or click the checkbox alongside the individual dashboard to import particular dashboard or report.
   If you are importing Dashboards, then apart from displaying whether it is an existing or new dashboard, you can assign a default role to the dashboard using the **Choose a default role if no role is assigned** drop-down list:



The `Options` page for Roles, Users, and Teams contains the name and description of the entity and its "Observation" column displays whether the entity that you are importing is "New" or "Existing".
The `Options` page for Delivery Rules, Rule Channels, and Pre-processing Rules contains the name of the entity, and its "Observation" column displays whether the entity that you are importing is "New" or "Existing".
The `Options` page for Administrative Settings lists the names of the administrative settings.
The `Options` page for Widgets, lists the title, name, version, and install mode of the widgets. Its "Observation" column displays whether the widgets that you are importing are "New" or "Existing".
The `Options` page for System Views, click the **View Name** checkbox to select or deselect both the system views, i.e., Navigation Structure and Queue Management Configuration.
In the case of **Navigation Structure**, you can customize the imported navigation items by can clicking the **Review** button. You can choose to **Merge** (Default) or **Replace** the navigation items existing in your system. **Merge** appends the extra navigation items available in the import configuration to the navigation items existing in your system. **Replace** replaces your existing navigation items with the items specified in the import configuration. Additionally, at the individual navigation item level, you can also selectively choose to **Skip**, **Merge**, or **Replace** that navigation items:

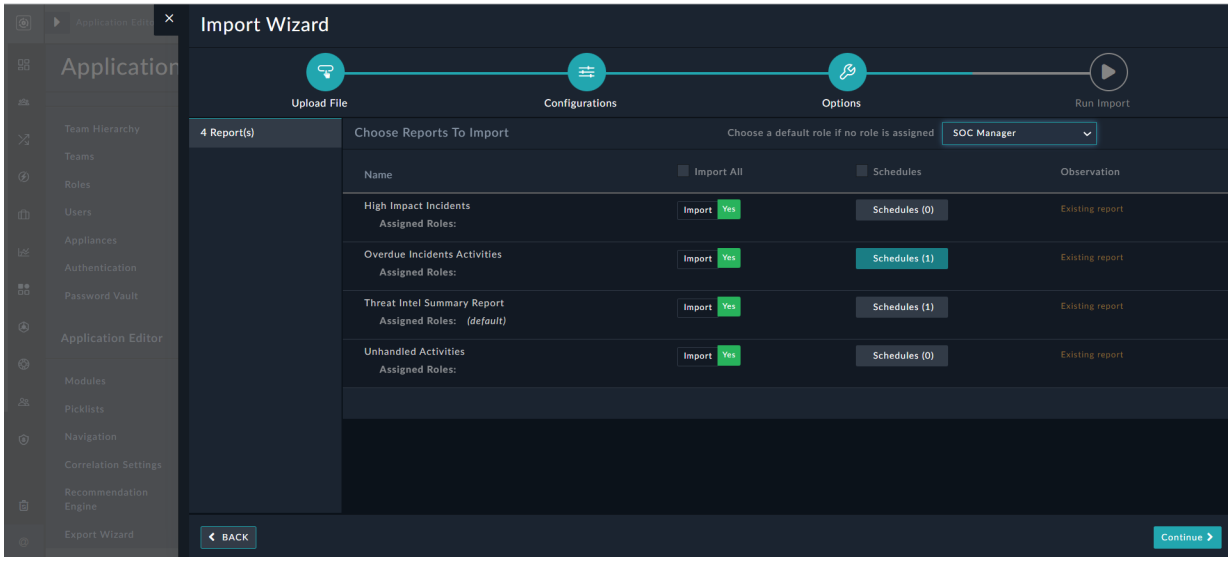


Click the `<Entity Name>` checkbox at the top of the corresponding `Options` pages to select or deselect all configurations of a specific type. To choose individual configurations, deselect this global checkbox. For example, to import all delivery rules, click the **Delivery Rules** menu item. Then, in the `Delivery Rules` section, use the **Rule Name** checkbox to select or deselect all the delivery rules.
The `Options` page for Export Templates lists the names of the export templates.
IMPORTANT: If dashboards, widgets, roles, uses, teams, system views (apart from navigation items), delivery rules, rule channels, pre-processing rules, or administrative settings that you are importing already exist in your system, then the Import Wizard overwrites the configurations of these entities in your system. In the case of widgets, when you try to import a specific version of a widget using the import configuration file, and that widget is not present in the FortiSOAR repository, then the latest version of that widget gets installed in your FortiSOAR system.
Notes on importing users:

- Users that are marked as 'Super Admin' are always imported in the 'Active' state.
- Concurrent users are imported as per their original state, i.e., 'Active' or 'Inactive'.
- Named users are also imported as per their original state, i.e., 'Active' or 'Inactive', if the current license permits. Named users are imported as 'Active' on a 'FIFO' basis, for example, if there are 5 users being imported in the 'Active state' to an instance where only 4 active user licenses are available, then the first 4 named users will be imported as Active and the remaining one as 'Inactive'. Administrators can change the state of users manually to 'Active' as required.
  Also, from version 7.0.1 onwards, access type information, i.e., named or concurrent, is also imported along with the other user details. If users are being imported from a version earlier than 7.0.1, i.e., 7.0.0, then those users do not have any access type information, and in this case, when these users are imported, they are imported as 'named' users.
- If new users are imported, then FortiSOAR will send those users an email to reset their passwords.

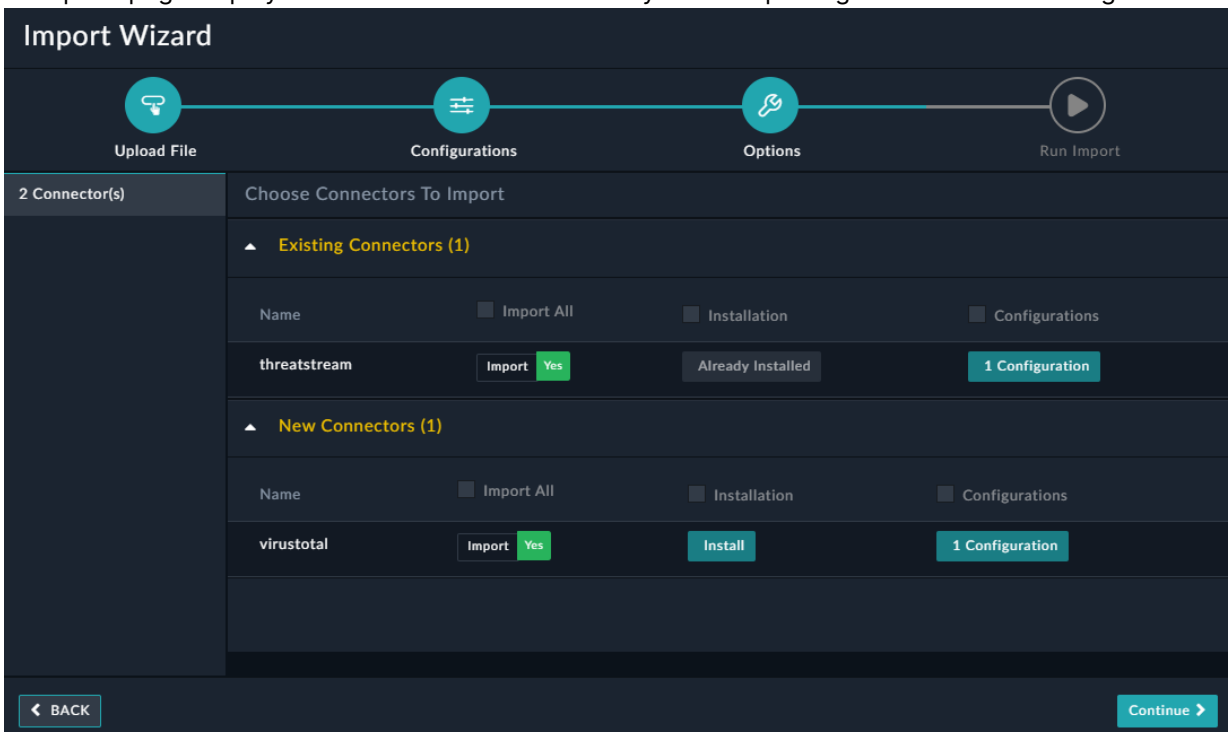
Importing Reports and associated Schedules:
To import reports, on the `Options` page, click the **Reports** menu item. The "Choose Reports to Import" page displays whether the reports that you are importing are "New" or "Existing". If the reports that you are importing already exist in your system, then the Import Wizard overwrites the configurations of these reports in your system. Use the **Choose a default role if no role is assigned** drop-down list to assign a default role to the reports you are importing. Schedules associated with imported reports are displayed in the row of that report; you can choose not to import the schedule by deselecting

schedules in **Schedules** in the row of that report. For example, the schedule associated with the Threat Intel Summary Report is deselected, so it will not be imported:



To import all the reports and their associated schedules, click the **Import All** checkbox. To export the all the schedules, click the **Schedules** checkbox.

Importing Connectors:
To import connectors, on the `Options` page, click the **Connectors** menu item. The "Choose Connectors to Import" page displays whether the connectors that you are importing are "New" or "Existing":



- If the connectors are new, then the connector import installs and configures the connector on your system.
- If the connectors are existing, and if the version of the installed connector on your system is the same or higher, then the connector import replaces only the connector configuration on your system.

- If the connectors are existing, and if the version of the installed connector on your system is the lower, then the connector import upgrades the connector on your system and replaces its configuration with the imported configuration.
- If the connectors are existing, and if the version of the installed connector on your system is the same or higher, and you are importing a connector with no configuration information, then nothing is replaced on your system.

Click the **Import All** checkbox in their respective sections to import all the connectors and their configurations in the respective 'Existing Connectors' or 'New Connectors' section. Similarly, click the **Installation** and **Configuration** checkboxes in the header to import all the installations and all the configurations respectively.

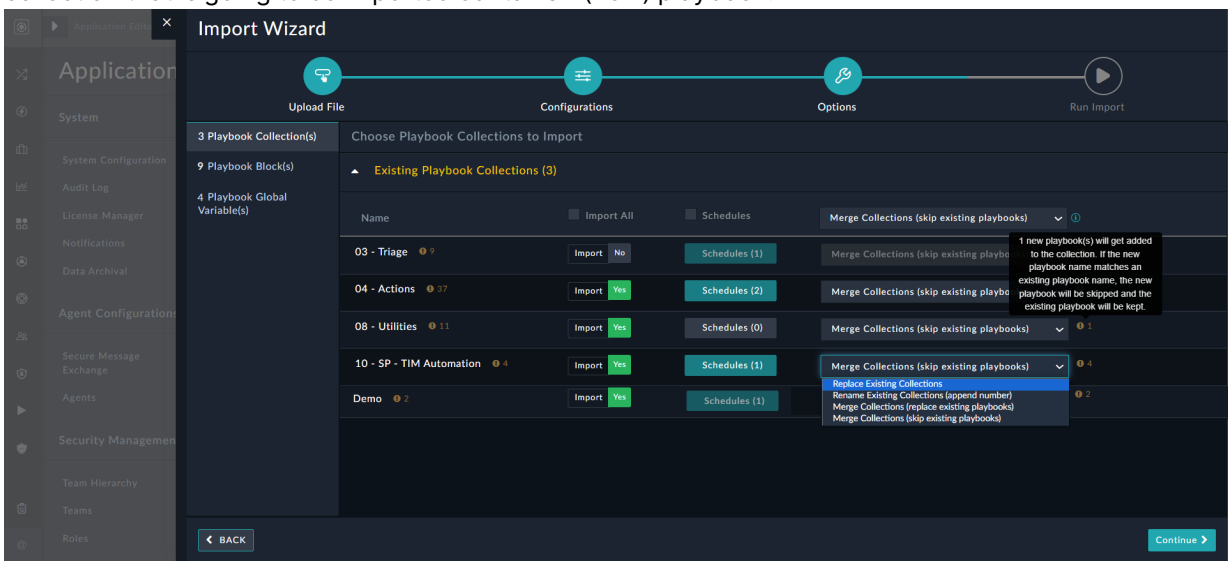
Toggle **Import** to **Yes** in a connector row to import that connector's installation and configuration and similarly toggling off the **Install** or **Configuration** buttons does not import the said installation or configuration.

Importing Playbook Collections, Schedules, Global Variables, and Playbook Blocks:

To import playbook collections, on the `Options` page, click the **Playbook Collections** menu item. Currently, you have to import the complete playbook collection, and cannot select specific playbooks to be imported from within a playbook collection. The playbook collections, global variables, and playbook blocks page displays the list of New and Existing options, for example, "New Playbook Collections", "Existing Playbook Collections" etc.

In the case of new playbook collections, global variables, and playbook blocks, the **Import All** checkbox in their respective sections is selected by default, to import all the new playbooks, global variables, and playbook blocks. You can choose not to import all the new playbooks, global variables, or playbook blocks by clearing the **Import All** checkbox, and then toggle **Import** to **Yes** alongside the individual playbook collections, schedules, global variables, or playbook blocks. For playbook collections, you can also choose to import schedules associated with the playbook collections.

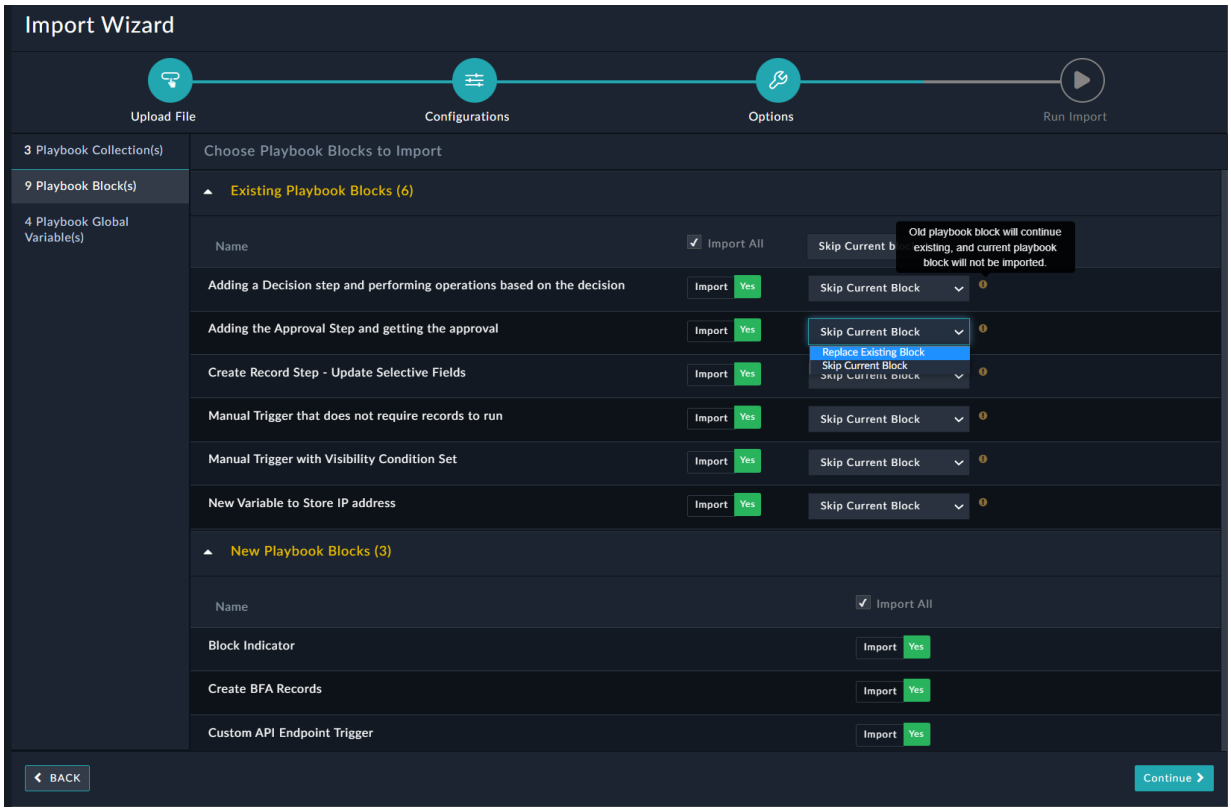
In the case of existing playbook collections, the count of playbooks both within the collection that is being imported, and the collection that exists on your system is also displayed, so that it becomes easier for users to know whether the correct playbook collection is being imported. For example, in the following image, the '08 - Utilities' playbook collection, contains 11 playbooks, whereas the playbook collection that is going to be imported contains 1 (new) playbook:



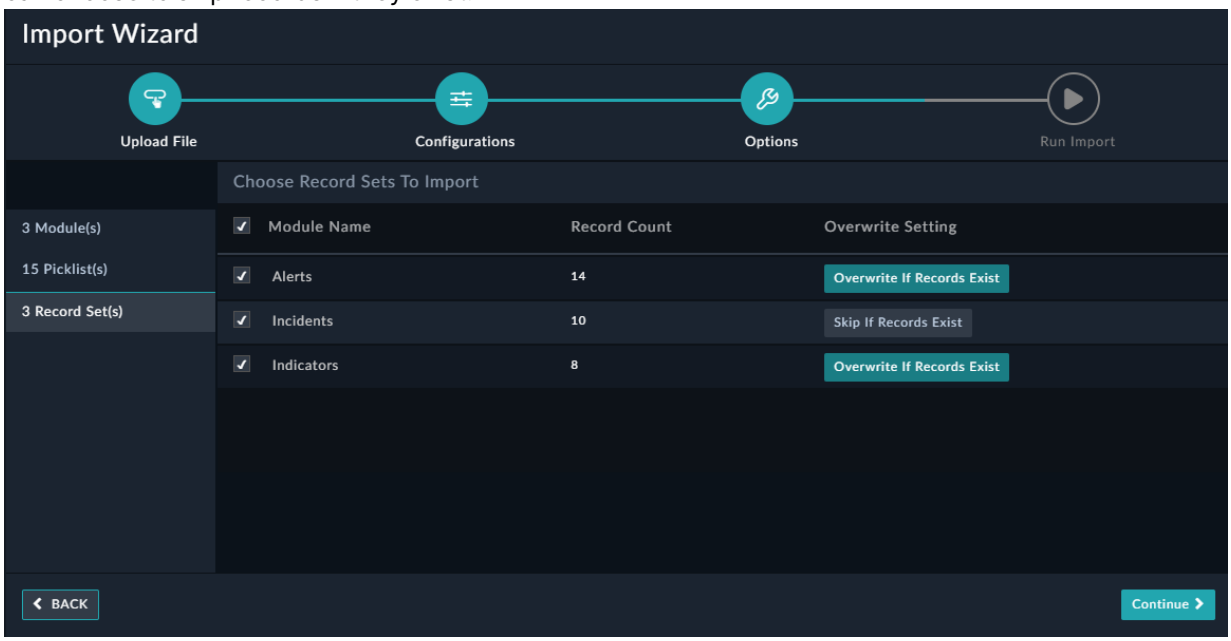
To import all schedules associated with the playbook collections you are importing, select the **Schedules** checkbox in the header. If you do not want to import all the schedules associated with the

playbook collections, clear the **Schedules** checkbox, and select or deselect the **Schedules** button in the respective row of the playbook collection that you are importing. The count of schedules associated with the playbook collections is also displayed.
In the case of existing playbook collections, apart from the **Import All** and **Schedules** checkbox in the header, the following "Bulk Actions" that you can take for existing playbook collections are also displayed: **Replace Existing Collections**, **Replace Existing Collection (append number)**, **Merge Collections (replace existing playbooks)**, or **Merge Collections (skip existing playbooks)**, which is the default option. You can choose to apply this action across all the playbook collections you are importing, or you can choose the action to be performed for each playbook collection that you are importing:
- If you retain the **Merge Collections (skip existing playbooks)** action, then the Import Wizard merges the playbook collection by skipping the existing playbooks. For example, if you have exported a 'Demo' playbook collection that has 2 playbooks, 'Create Demo Records' and 'Test Manual Input', and you are importing this into a system that has the 'Demo' playbook collection with the 'Create Demo Records' playbook, then the Import wizard merges the 'Demo' playbook collection such that it will not overwrite the 'Create Demo Records' playbook; but it will add the 'Test Manual Input' playbook.
- If you choose the **Merge Collections (replace existing playbooks)** action, then the Import Wizard merges the playbook collection by replacing the existing playbooks. For example, if you have exported a 'Demo' playbook collection, that has 2 playbooks, 'Create Demo Records' and 'Test Manual Input', and you are importing this into a system that has the 'Demo' playbook collection with the 'Create Demo Records' playbook, then the Import wizard merges the 'Demo' playbook collection by overwriting the existing 'Create Demo Records' playbook and adding the 'Test Manual Input' playbook.
- If you choose the **Replace Existing Collections** action, then the Import Wizard overwrites the playbook collections in your system.
- If you select **Replace Existing Collections (append number)**, then the Import Wizard creates a new playbook collection and appends a number to the original playbook collection name. For example, if you have exported a playbook collection named 'Demo' and you are importing the same playbook collection with **Replace Existing Collections (append number)** selected, then the imported collection will automatically be created as a new playbook collection named as 'Demo (1)'.

Importing Global Variables or Playbook Blocks: By default, global variables and playbook blocks that exist on your system are not imported, and new global variables and playbook blocks are imported. In the case of Playbook Blocks, in the case of existing blocks, you can choose to **Skip Current Block** (default) or **Replace Existing Block**:



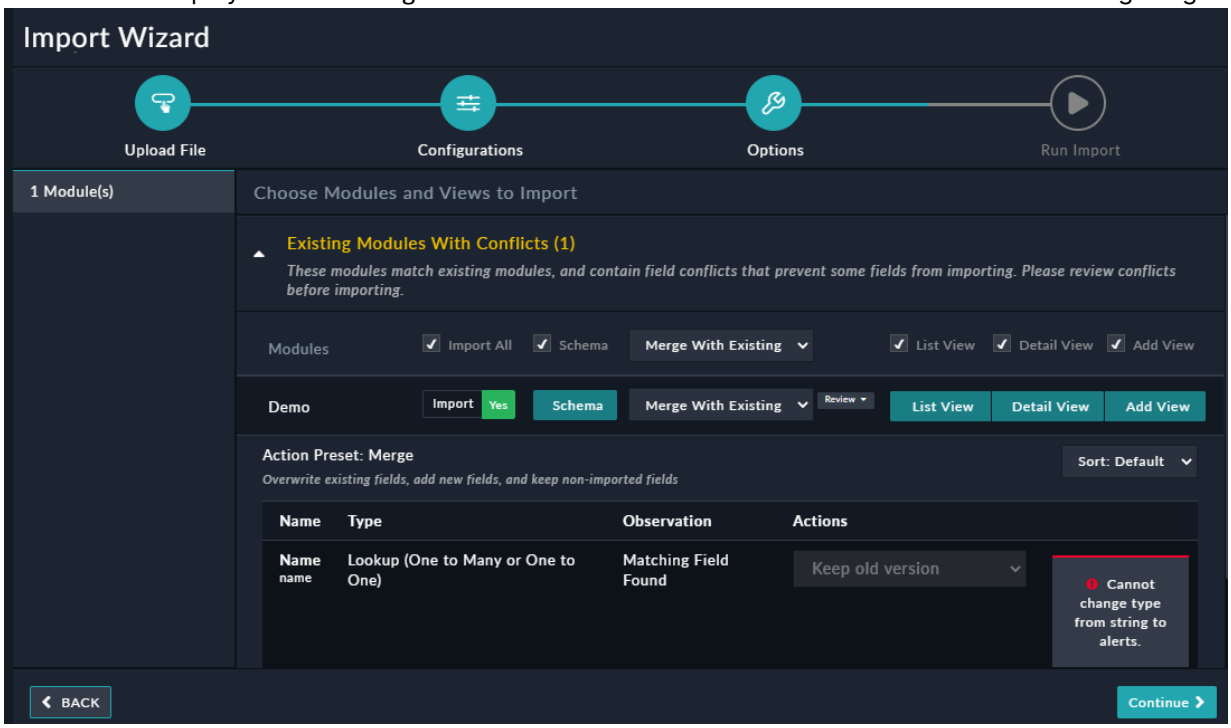
Importing Record Data:
To import record data, on the Options page, click the Record Set(s) menu item. The Options page displays the module name for which the records are being imported, the count of records to be imported, and the overwrite settings, i.e., you can choose to either Overwrite records if they exist or can choose to skip records if they exist.



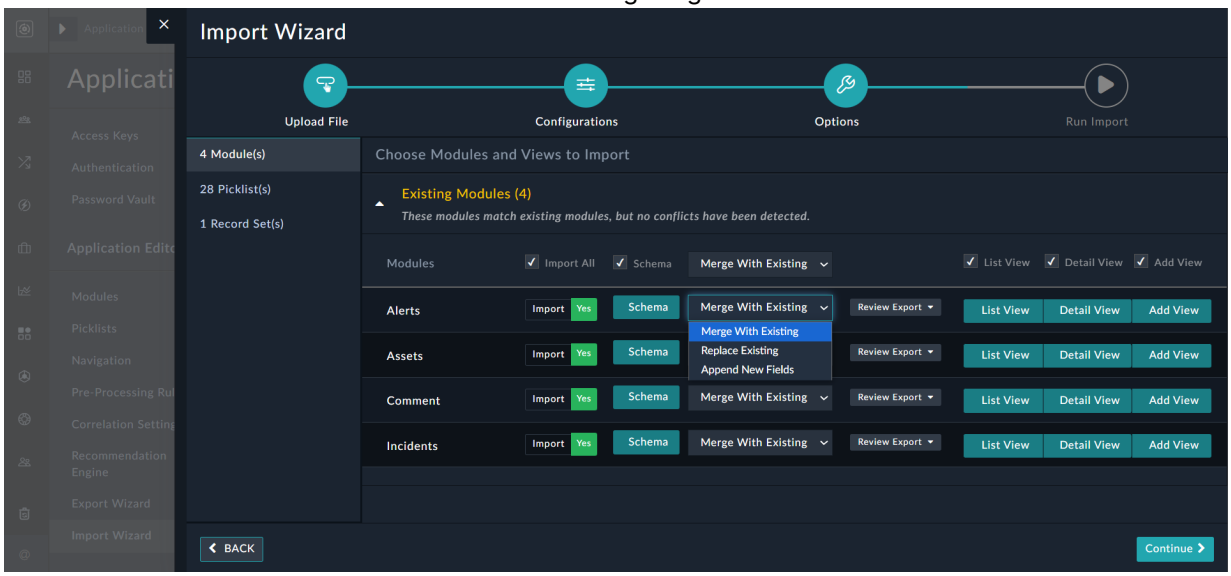
Note: If a record set is included in the import, then the module schema for that record set is required

and gets automatically included in the import.

Importing Modules Configurations and Picklists:
When you import configurations for existing modules, and if the modules that you are importing contain fields that conflicts with the existing fields that prevent some fields from being imported, then those modules are displayed in `Existing Modules With Conflicts` section as shown in the following image:



Modules whose fields have no conflicts with existing fields are displayed in the `Existing Modules Without Conflicts` section as shown in the following image:



Choose the options in the header row to perform bulk actions. For example, if you want to import all the modules, click the **Import All** checkbox, etc.
When importing module schemas, you can select one of the following options for existing

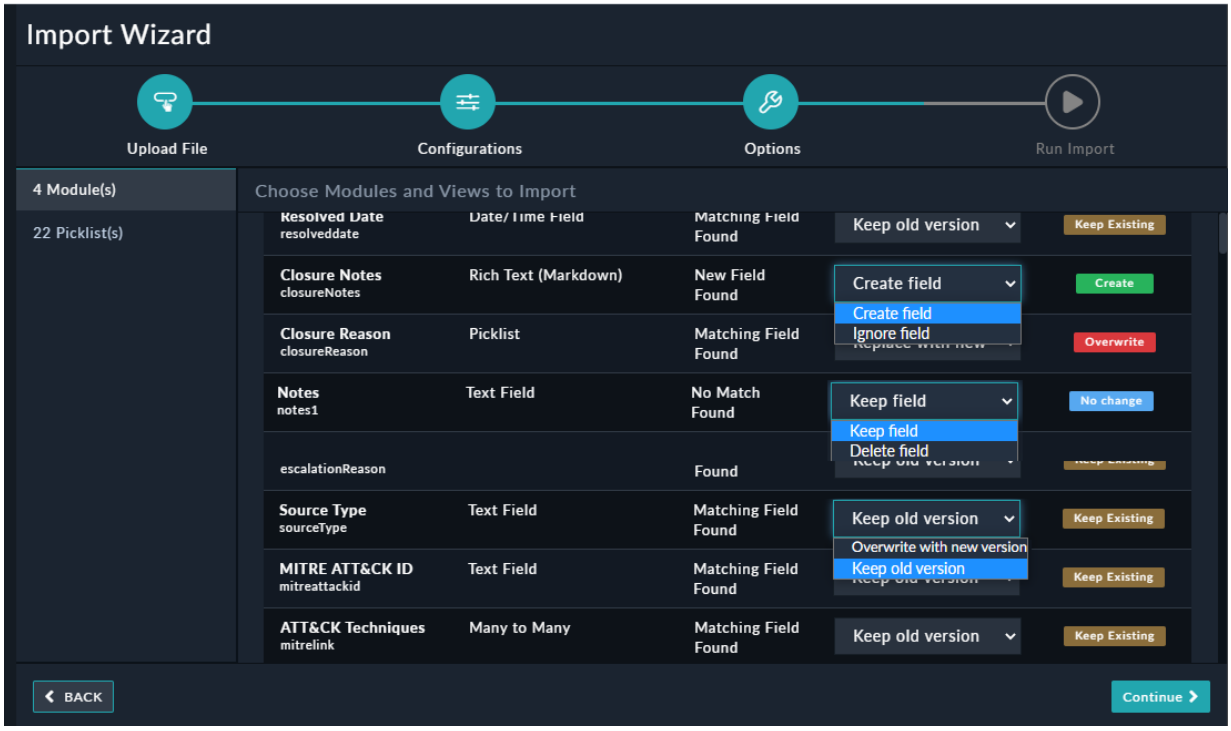
configurations:

- **Merge With Existing**, merges then the configurations, i.e., for example if you are importing an existing module, say Alerts, which has 3 new fields in the configuration that you are importing and 10 existing fields and you choose Merge, then post-import the Alerts modules will have 13 fields. Therefore, merge overwrites existing fields, adds new fields, and keeps non-imported fields.
- **Replace Existing**, replaces the existing configuration with the imported configuration, i.e., it overwrites existing fields, adds new fields, and deletes non-imported fields.
- **Appends New Fields**, keeps the existing fields as well as adds new fields, i.e., it keeps existing fields, adds new fields, and keeps non-imported fields.

Important Considerations when importing System View Templates (SVTs) :

- If the name of the SVT for a module matches an existing SVT name in the target system (system where the configuration is being imported), the existing SVT for that module will be replaced.
- If an SVT being imported already exists in the target system, and it is identified by its UUID, the existing SVT will be replaced.
- If the SVT in the imported configuration is marked as the default template, but the target system already has a default view template for the module, the imported SVT will be added as a non-default template.
- If you export the SVT from an FortiSOAR version earlier than 7.6.2 using the Export Wizard and import it into FortiSOAR version 7.6.2 or later using the Import Wizard, the SVT name will be displayed as 'Custom Template' and it will be editable.

Click the **Review Export** button to view the detailed schema of the module you are importing, which includes information about fields such as, which fields are replaced, which fields are retained, which fields are going to be created, and which fields are going to be ignored. You can therefore selectively decide what they want to do with fields that are different in the existing modules and in the configurations that they are importing. Select between various options such as **Create field**, **Ignore field**, **Keep old version**, or **Delete field**, or **Overwrite with new version**, etc., which are present in the **Actions** column of the respective fields and decide which fields are going to be imported.
You can choose to sort how the fields are displayed in the grid by clicking the **Sort** drop-down list. The **Sort** drop-down list has the **Default**, **A-Z**, or **Z-A** options.



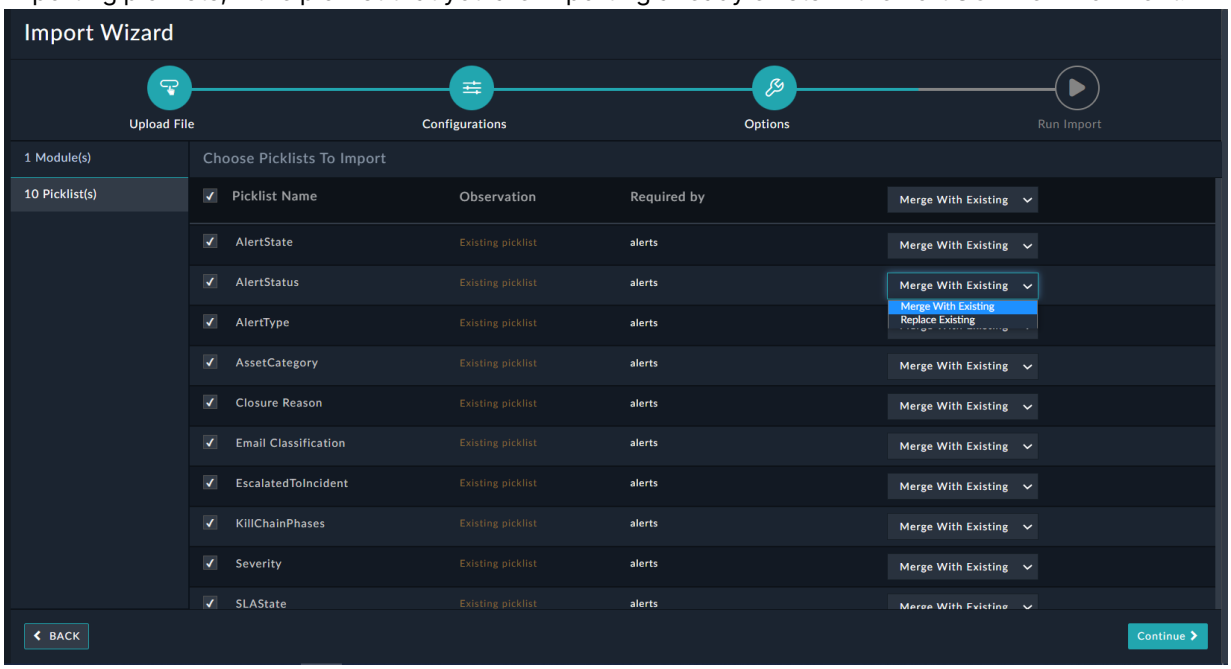
Observations displayed for various fields:

- **New Field Found**: Fields that are present only in the configuration that you want to import, i.e., fields that are newly added to the configuration. Available user actions are **Create field** or **Ignore field**.
  Note: If you select **Ignore field** then the newly added field is not included in the mmd when you import the configuration.
- **No Match Found**: Fields that are present only in the existing module and not in the configuration that you want to import, i.e., fields that are deleted from the configuration. Available user actions are **Keep field** or **Delete field**.
  Note: Delete field will delete the field from the mmd file.
- **Matching Field Found**: Fields that are present in both the configuration that you want to import and in the existing module, but which have *different properties* in the configuration that you want to import and in the existing module. These are fields that a user should replace with the newer version of the field. However, ensure that you review all the fields before choosing the import option since replacing a field with its newer version should not result in the publish failing due to for example, conversion of the field to an Unsupported type. Available user actions are **Replace with new version** or **Keep old version**.
- **System Field**: Fields, whose properties cannot be changed by users. An example of a system field would be the **First Name** field in the `People` module, which cannot be changed by users. For more information on system fields, see Module Editor.
  Note: The name and properties of the Lookup (One to Many or One to One), Many to Many, and Many to One field must not be changed once they have been defined. For example, the Alerts module contains a `Many to Many` with the **Indicators** field, and if in the configuration that you are importing the name of this field is changed to Indicator1 then the new field Indicator1 will not be imported.
  Once you have competed reviewing the import options, click **Continue**.

Importing Picklists

From release 7.2.0, you can choose to merge or replace picklists. These options are visible while

importing picklists, if the picklist that you are importing already exists in the FortiSOAR environment:



If you choose **Merge With Existing** (default), the picklists items that are being imported get added in the existing picklist. If you choose **Replace Existing**, then the imported picklists replace the existing picklists.

4. On the `Review Import` page, you can review the import details that you are importing, including details of which entities, views, etc., you are importing, the number of records being imported from modules, etc. Once you have reviewed the import details displayed by the Import Wizard, click **Run Import** to begin the import process or you can close the wizard. Clicking **Run Import** displays a configuration dialog, where you can click, **I have reviewed the changes - Publish** to import and publish the configuration into your FortiSOAR environment. Once the configuration publish begins, FortiSOAR displays the list of configurations being imported along with progress of the import. For example, `Publishing Modules (36%)` or `Importing Connectors`, etc., and once the process is completed the `Import Process Completed Successfully` message is displayed. If there are any issues with the configuration that you are trying to import then "Publish" operation fails and the wizard displays a message containing information about which configuration has failed such as `Error while Importing Reports`, and also the details of the error that caused the failure.

While importing connector configurations, the system does not perform health checks to ensure that the connector configurations are accessible. Therefore, the import will show successful even if a connector's health check returns “Disconnected”. It is your responsibility to review the configurations of imported connectors to ensure they are active.

Points to be considered while importing modules

- If a Tenant or Agent is imported then their status will be inactive you will need to re-configure the Master node on the tenant or agent.
- The Secure Message Exchange is imported as configured, if the secure message exchange is reachable from the FortiSOAR system and there is no change to its certificates or credentials.

- If you have edited a picklist on an environment (Env)1 and you import the Env1 configuration into Env2, in this case, the edited picklist items will be replaced.
- If you have added a field, say `test1`, to Env1 and added a field, say `test2`, to Env2, to the `Alerts` module in both environments. Now, if you export the `Alerts` module from Env1 and import the `Alerts` module to Env2, then the `Alerts` module in Env2 gets completely overridden, i.e., the `Alerts` module in Env2 will now only contain the `test1` field, and the `test2` field gets overridden.
  You can also select the **Merge** option to retain fields that were present in an existing module but which are not present in the exported (new) module.

Updates required to be done after importing SSO configurations

If you have exported your SSO configuration and imported the SSO (SAML) configurations into a different FortiSOAR system, you require to make the following updates to the service provider portal, before SAML users can log into FortiSOAR:

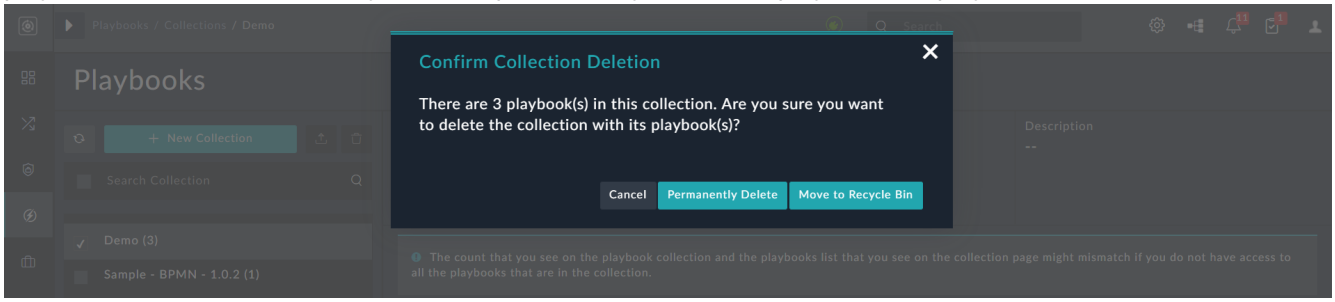
1. Update the "Single Sign On URL" to the URL of the system that is importing the SSO configuration.
2. Update any other field in the service provider's portal that mentions the FortiSOAR system URL.
3. Generate the X509 certificate for the FortiSOAR system that is importing the SSO configuration.

Once you have generated the X509 certificate, you must update the newly generated X509 certificate details on the `SSO Configuration` page in the FortiSOAR system that is importing the SSO configuration. To open the `SSO Configuration` page, click **Settings** > **Authentication** > **SSO Configuration**. In the `Identity Provider Configuration` section, in the **X509 Certificate** field update the details of the newly generated X509 certificate.

Recycle Bin

FortiSOAR version 7.2.0 adds a 'Recycle Bin' to support soft delete of workflow and module records; so that in the case of accidental deletion of playbook collections, playbooks or module records these records can be restored.

In the case of Playbook Collections and Playbooks, soft deletion is enabled by default, since an accidental deletion of playbook collections, or playbooks can result in lot of effort being lost and complete stoppage of automation you have configured. Therefore, when users want to delete playbooks or playbook collections, FortiSOAR displays a confirmation dialog where users can choose whether they want to move the playbooks or playbook collections to the recycle bin or permanently delete the playbooks or playbook collections:



Clicking **Move to Recycle Bin** moves all the playbooks of that collection to the recycle bin.

Uniqueness constraint is applied when records are in recycle bin. Therefore, for example, you cannot create a playbook collection or playbook, with the same name as a playbook collection or playbook that have been sent to the recycle bin (soft deleted).

In the case of modules, by default, the records are deleted (permanently deleted); however, you can configure modules to send records to the recycle bin instead of getting permanently deleted by selecting the **Enable Recycle Bin** option on the `Modules` page and then clicking **Save** and **Publish**. For more information, the Application Editor chapter.

You cannot configure the Recycle Bin for system modules, i.e., for the 'People', 'Appliances', 'Agents', 'Approvals', 'Tenants', 'Routers', 'Comments', and 'Saved Reports' modules. Therefore, records of these modules are always permanently deleted.

You can also schedule the purging of recycle bin records to periodically clear the records present in the recycle bin. For more information, see the System Configuration chapter.

Permissions Required

- To view Recycle Bin records, you must be assigned a role that has a minimum of `Read` permission on the 'Application' and 'Playbooks' modules. You also need the `Read` permission on modules whose recycle bin records you want to view.

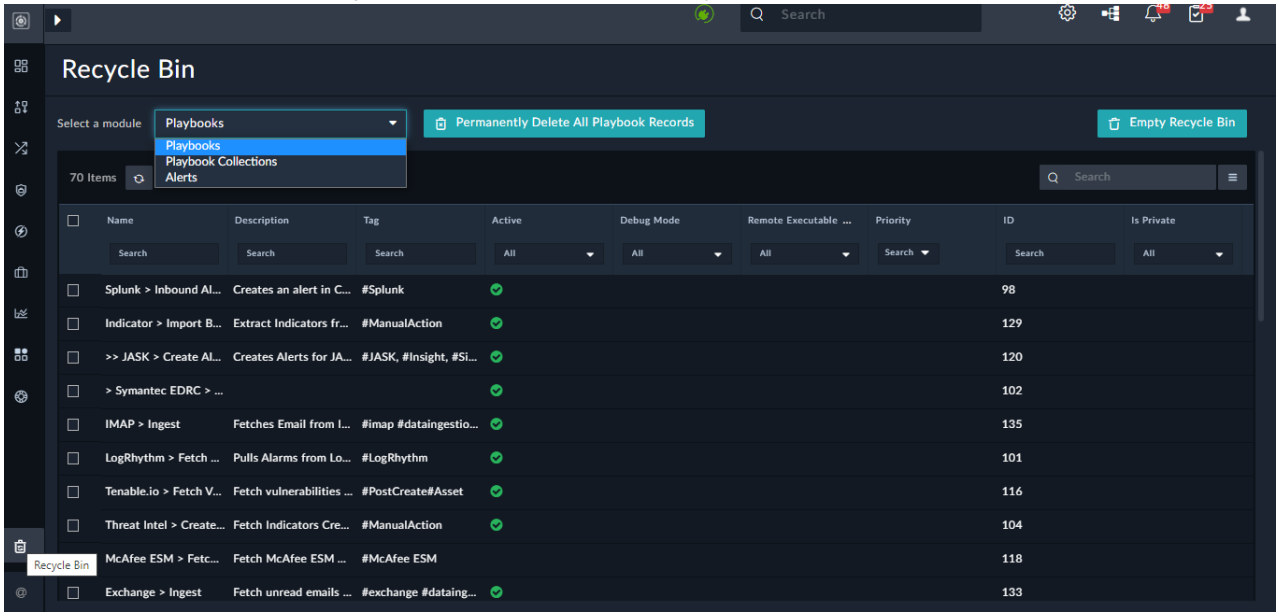
- To permanently delete Recycle Bin records, you must be assigned a role that has a minimum of `Read` and `Delete` permission on the 'Application' module, `Delete` permissions for the module such as 'Playbooks', 'Alerts', etc., whose records you want to permanently delete, and `Read` permission on the 'Playbooks' module.
  Note: The `Delete` permission on the 'Application' module is required only if a setting on the `System Configuration` page has been enabled to restrict access for permanently deleting records to only users with the 'Administrator' role. For more information, see the System Configuration chapter.
- To restore Recycle Bin records, you must be assigned a role that has a minimum of `Read` permission on the 'Application' and 'Playbooks' modules and `Update` permission for the module whose records you want to restore.

Using the Recycle Bin

You can use the Recycle Bin to view the soft-deleted records, playbooks, and playbook collections. You can also permanently delete items from the recycle bin, restore selected records, or empty the recycle bin.

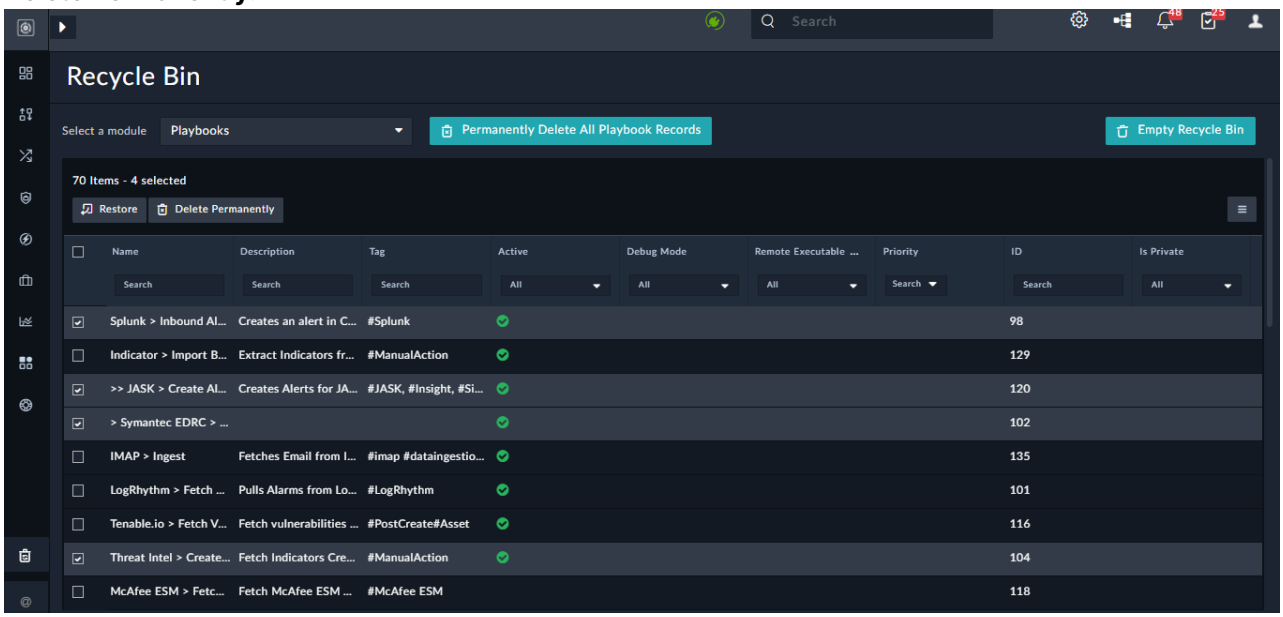
Click the **Recycle Bin** icon in the left navigation pane to display the `Recycle Bin` page with all the soft-deleted records. You can perform the following operations in the recycle bin:

- To search for specific records, click the **Search** icon and type the keywords in the search box. You can also filter records by typing the filter term, tag, or selecting the option in the first row of the record listing.
- To permanently delete all records that belong to the same module, from the **Select a module** drop-down list, select the module whose records you want to delete, and click **Permanently Delete All <Module Name> Records**. In our example, we have selected the 'Playbooks' module.



Note: The **Select a module** drop-down list displays the 'Playbooks' and 'Playbook Collections' options since by default the soft deletion of records for these modules is enabled. This drop-down list also displays all the other modules for which soft deletion has been enabled as is the case with the 'Alerts' module in our example.

- To permanently delete one record or more specific records, select the module to which the record belongs from the **Select a module** drop-down list. Then, select the record or records you want to delete and click **Delete Permanently**.



- Similarly, to restore one or more specific records, select the module to which the record belongs from the **Select a module** drop-down list. Then, select the record or records you want to restore and click **Restore**.
  - Records are restored with its existing relations. For example, if you have moved an alert that was related to an indicator to the recycle bin, then that alert will no longer be linked to that indicator record. If you then choose to restore that alert, then the alert gets linked back to the same indicator record.
  - In the case of playbooks, if you want to restore a playbook from a playbook collection, then the collection containing those playbooks is restored. For example, if you have a 'Demo' collection containing 3 playbooks, A, B, and C that you move to the recycle bin, and then restore B from the recycle bin, the Demo collection containing the B playbook gets restored.
  - In the case of MSSP environments, if both the master and tenant systems have enabled recycle bin on a specific module, then any record that is moved to the recycle bin on the master node also gets moved to the recycle bin on the tenant nodes. However, if only the master or any tenant enables the recycle bin and not vice-versa, then if a record is deleted from the master who has enabled the recycle bin but not the tenant, then the record gets permanently deleted from the tenant. If the same record is restored on the master, then the record also gets replicated back on the tenant node, however at this time on the record will be created as a new record on the tenant side
- To clear the Recycle Bin, i.e., permanently delete all records in the recycle bin, click **Empty Recycle Bin**.

Behavior of Recycle Bin in the case of MSSP environments

In the case of MSSP environments, if any module is enabled for the Recycle Bin, then it is recommended that it should be enabled on both the master and tenant systems.

In the case of MSSP environments, if both the master and tenant systems have enabled recycle bin on a specific module, then any record that is moved to the recycle bin on the master node also gets moved to the recycle bin on the tenant nodes. However, if only the master or any tenant enables the recycle bin and not vice-versa, and if a record is deleted from the master (who has enabled the recycle bin) then that record gets permanently deleted from the tenants (who have not enabled the recycle bin. Similarly, if a record is restored on the master, then the record also gets replicated back on the tenant nodes; however, the record gets created as a new record on the tenant side.

Also, if on the master node a module is marked to be enabled for the recycle bin, and the master pushes this module to the tenants, then that module gets enabled for the recycle bin also at the tenants' end.

Segmented Network Support

Overview

FortiSOAR supports segmented networks, which facilitates investigation in a multi-segmented network by allowing secure remote execution of connector actions. If your requirement is to be able to remotely run connector actions, then you can use the "FSR Agent".

Automated ingestion, enrichment, or triage actions using a SOAR platform require network connectivity to various endpoints on which you want to run connector actions. These devices or endpoints, can at times, be in a different network segment than the one where the FortiSOAR node is deployed. To connect to such endpoints in segmented networks, FortiSOAR provides a lightweight component, called the "FSR Agent". An FSR Agent can be deployed in a network segment and configured to receive and execute connector actions from a FortiSOAR node using its secure message exchange. The FSR Agent only needs an outbound network connectivity to the secure message exchange server on its TCP port. It does not need a VPN setup or an inbound network connectivity.

An FSR Agent is small, lightweight, consumes minimal system resources and it is very easy to deploy and maintain. An FSR Agent can represent a standalone agent that can be deployed at a specific endpoint, or an FSR Agent can represent a dedicated tenant. If your requirement is action execution in a segmented network, then an FSR Agent is good enough. However, if you need incident management capabilities, or require to ingest large volumes of data from a source in a remote network, then you must have a dedicated FSR tenant instance.

In cases such as a multi-segmented network where deploying a dedicated FortiSOAR node per segment is not feasible or required you can use the lightweight FSR Agent. You can install multiple agents on your FortiSOAR enterprise node using which you can run remote connector actions on various segments of your network.

You do not require any additional licensing for the FortiSOAR secure message exchange.

For the process of deploying FortiSOAR Agents, see the *Deploying FortiSOAR* chapter in the "Deployment Guide."

FortiSOAR Agent CLI

Use the FortiSOAR Agent CLI (`csagent`) to perform various administrative functions on the agent such as, importing a connector, setting configurations for a connector, starting or stopping services, listing connectors, checking the health of the agent, etc. For more details on the options available with `csagent`, use the help command: `csagent --help` and also see the Running connector actions on an FSR agent section.

Invoke connector actions using FSR agents in segmented networks

Once you have installed the FSR agent, you can install and configure connectors on the agent and then run remote actions on the FSR agent using connectors.

Minimal permissions required

To use FSR agents to run connector actions:

- `Execute` permissions on `Connectors`.
- `Read` permissions on `Application` and `Agents`.

Installing a connector on an FSR agent

Connectors, including custom connectors or older versions of connectors, which do not have their rpms available on the FortiSOAR server, and which are installed on the FortiSOAR instance can be installed on FSR agents using the FortiSOAR UI. Also, connectors that were created and published using the "Create New Connector" wizard can be installed on FSR agents using the FortiSOAR UI. For more information on the "Create New Connector" wizard, see the "Connectors Guide"
You can also choose to install some pre-defined connectors when you are installing an FSR agent. For information on installing FSR agents see the *Deploying FortiSOAR* chapter in the "Deployment Guide."

For releases prior to 7.2.0, you can deploy connectors that do not have their rpm available on the FortiSOAR server using the CLI on the agent system details of which are included in the Installing and configuring a custom connector on an FSR agent prior to release 7.2.0 section.

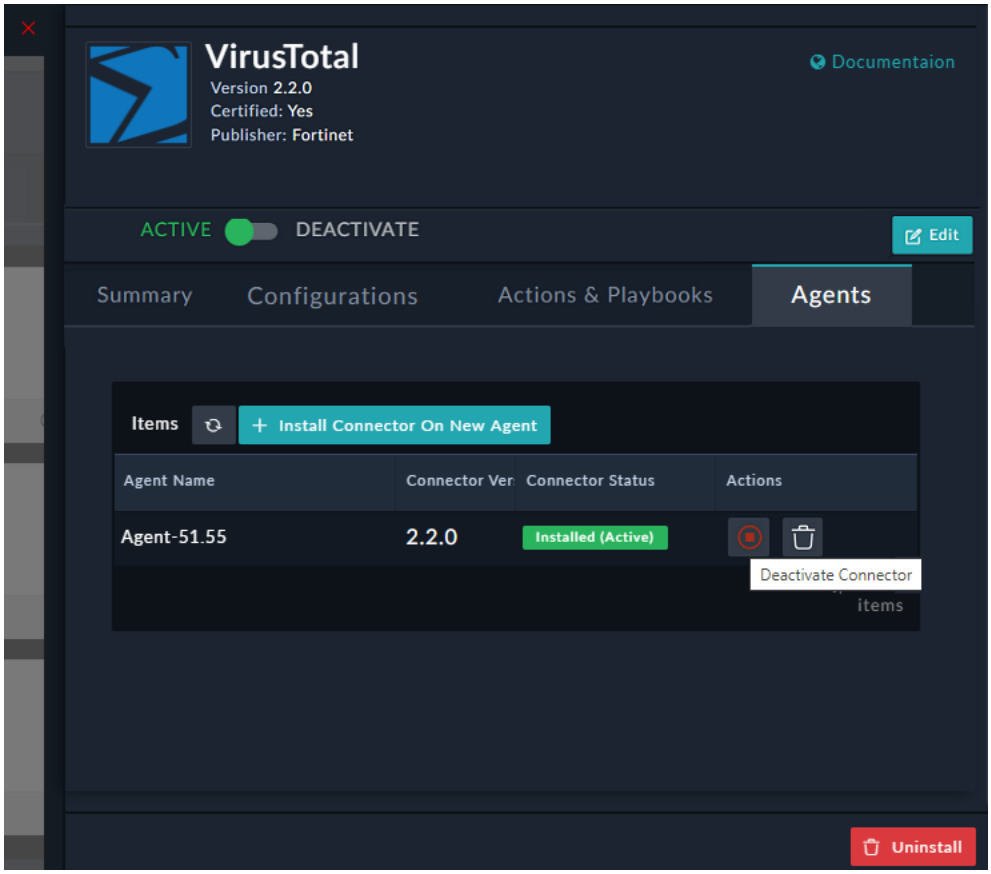
To install connectors on the FSR agent, do the following on the FortiSOAR node:

1. Log on to FortiSOAR.
2. On the left navigation pane, click **Content Hub > Manage** or **Automation** > **Connectors** > **Manage**. For more information on Content Hub, see the *Content Hub* chapter in the "User Guide."
   Important: Only connectors that are installed on the FortiSOAR node can be installed on FSR agents.
3. Click the connector that you want to install on the FSR agent, which opens the `Connector Configuration` pop-up.
4. On the `Connector Configuration` pop-up click the **Agent** tab.
   To install the connector on a new FSR agent, click **Install Connector on New Agent**, and from the `Agents` dialog, which contains a list of installed FSR agents, select the FSR agent on which you want to install the connector and click **Install**.
   Prior to version 7.0.1 the connector rpm, by default, used to get downloaded to the `/tmp` directory, which could be restricted for writing into it by other users causing the installation to fail. Therefore, from version 7.0.1 onwards, the connector rpm, by default, is downloaded to the `/opt/cyops-integrations/cyops_`

`connector_rpm` directory. Also, you can configure the directory to which you want the connector rpm to the downloaded. For more information see the Configuring the directory in which to download the connector rpm section.
Note: The `Agents` dialog lists only those FSR agents whose **Status** is "Remote Node Connected."

If you have updated a custom connector on the base node and you want to apply these edits and publish a connector with the same version on the FSR agent, then you must uninstall the connector from the FSR agent. Once the connector is uninstalled, install the updated connector. You must restart the agent service (`cyops-integrations-agent service`) once the updated connector is installed on the FSR agent to ensure the changes take effect.

The **Agent** tab displays the names of the FSR agents on which the connector is installed, the version of the connector installed, and the connector status. You can also use this tab to install that connector on a new FSR agent, and activate, deactivate, delete, or upgrade the connector on the FSR agent:



You can activate, deactivate, or uninstall the connector for a particular FSR agent by clicking the **Activate Connector**, **Deactivate Connector**, or **Uninstall Connector** buttons respectively. If the version of the connector on the FortiSOAR instance and the FSR agent is the same, then the **Upgrade Connector** button will not be visible in the FSR agent row. However, the version of the connector on the FortiSOAR instance is higher than the version of the connector installed on the FSR agent, then you can upgrade the connector by clicking the **Upgrade Connector** button.

Activating, deactivating, uninstalling, or upgrading the connector for an FSR agent only activates, deactivates, uninstalls, or upgrades that connector for that particular FSR agent and not for any other FSR agent or the FortiSOAR instance.

To configure connectors, see the Configuring Connectors section.

Configuring the directory in which to download the connector rpm

The connector rpm, by default, is downloaded to the `/opt/cyops-integrations/cyops_connector_rpm` directory. To configure the directory in which you want the connector rpm to be downloaded, follow these steps:

1. On the FSR agent:
   a. Create the new directory in which you want the connector rpm to be downloaded.
   Important: Ensure that this newly created directory is given `nginx` user permission.
   b. Open the `/opt/cyops-integrations/integrations/configs/config.ini` file and set the `conn_rpm_temp_dir` parameter value as the directory in which you want the connector rpm to be downloaded.
   c. Save the file and restart the `uwsgi` service.
2. On the FortiSOAR (base) node:
   a. Open the `/opt/cyops-integrations/integrations/configs/config.ini` file and set the `conn_rpm_temp_dir` parameter value as the directory in which you want the connector rpm to be downloaded.
   b. Save the file and restart the `uwsgi` service.

Installing and configuring a custom connector on an FSR agent prior to release 7.2.0

From release 7.2.0 onwards, you can install connectors whose rpms are unavailable on the FortiSOAR server on an FSR agent, and also connectors that were created and published using the "Create New Connector" wizard, as mentioned in the Installing a connector on an FSR agent section.
Prior to release 7.2.0, to install and configure a custom connector on an FSR agent, follow these steps:

1. Import the custom connector on your FortiSOAR (base) node.
2. Create an FSR agent installer but do not select the custom connector. For information on creating an FSR agent installer see the 'Adding an FSR agent' section in the *Deploying FortiSOAR* chapter of the "Deployment Guide."
3. Run the FSR agent installer on the FSR agent and confirm that the FSR agent is successfully installed. For information on running the FSR agent installer see the 'Installing an FSR agent' section in the *Deploying FortiSOAR* chapter of the "Deployment Guide."
4. Copy the custom connectors' `.tgz` file to the FSR agent's system.
5. Run the following commands to install and configure custom connectors:
   a. To import a custom connector, use the following command:
   `csagent --option import`
   With the `import` command, you can add the following parameters:
   `[--name Name]`: Name of the connector that you want to import.
   `[--version Version Number]`: Version number of the connector that you want to import.
   `[--path Path]`: Path of the folder that contains the connector you want to import.
   `[--path Bundle]`: Path of the archive that contains the connector you want to import.

For example, to import version 1.0.0 of the SNS connector, use the following command:
`csagent --option import --name sns_con --version 1.0.0 --bundle /root/sns_con.tgz`

b. To configure a custom connector, use the following command:
`csagent --option configure --name <connector_name> --version <connector_version>`
Once you run the above command you require to specify the values for the name and parameters of the configuration:
```
Name of the configuration: ....
Field 1: ....
Field 1: ....
```
To set the default configuration for the connector, use the `csagent --option configure --default` command.

Running connector operations on an FSR agent using `csagent`

Apart from installing and configuring a custom connector (prior to release 7.2.0), you can also perform the following connector operations using `csagent`:

- `check_health`: Checks the health, i.e., the status of the connector, i.e., if the connector is 'Disconnected' or 'Available'. To check the health of a connector, you need to provide the connector name, version, and configuration name with this command:
  `csagent --option check_health --name <connector_name> --version <connector_version> --config <configuration_name>`
- `execute`: Executes a connector operation, i.e., runs a particular action for the specified connector. To execute a connector operation, you need to provide the connector name, version, and configuration name, and also specify the name of the operation that you want to run with this command:
  `csagent --option execute --name <connector_name> --version <connector_version> --config <configuration_name> -operation <operation_name>`
  After you enter the above command, you will have to specify the parameters (if required) for the operation.
- `remove`: Deletes a particular version of the specified connector. To remove a connector, you need to provide the connector name and version with this command:
  `csagent --option remove --name <connector_name> --version <connector_version>`
- `list_operations`: Lists the supported operations for a particular version of the specified connector. To list the operations of a connector, you need to provide the connector name, version, and configuration name with this command:
  `csagent --option list_operations --name <connector_name> --version <connector_version> --config <configuration_name>`
- `list_configs`: Lists the configurations available for a particular version of the specified connector. To list the configurations of a connector, you need to provide the connector name and version with this command:
  `csagent --option list_configs --name <connector_name> --version <connector_version>`
- `list_connectors`: Lists all the connectors that are installed on the system, along with their version numbers and status:
  `csagent --option list_connectors`
- `services`: Starts, stops, or restarts the services on the system:
  `csagent --option services --action <start|stop|restart>`

Configuring connectors

You can configure connectors for the current FortiSOAR node, the FSR agent, or both. You can configure the connector on the current node (Self) or on a remote FSR Agent node (Agent) by clicking the **Self**, which is the default, or **Agent** buttons besides `Target`. You can configure multiple configurations for a connector on both the current node and the FSR agent node.

Configuration details, such as passwords, credentials, or other sensitive data can be stored by your administrator using "Password Vault". However, you will not be able to use these stored credentials to configure a connector on an FSR agent since vaults do not work on agents.

To configure connector, on the `Connectors` page, click the connector that you want to configure to open the `Connector Configuration` pop-up in which you can add connector configurations. To configure and execute connector actions, on the "Self" node, click **Self**, which is the default, if you are configuring the connector for the first time or if you want to add a new configuration, then click **Add new configuration** from the **Select Configuration** drop-down list and then add the name of the configuration and specify the configuration parameters. If you want to update an existing configuration, then select the configuration from the **Select Configuration** drop-down list and update the configuration parameters. If there is only one configuration, then that configuration will be selected automatically.

To configure and execute connector actions on the "FSR Agent" node, click **Agent**, and from the **Select Agent** drop-down list select the FSR agent on which you want to run the connector actions. If there is only one FSR agent installed, then that FSR agent will be selected automatically.

The **Select Agent** drop-down lists only those FSR agents whose Status is "Remote Node Connected."

If you are configuring the connector for the first time or if you want to add a new configuration, then click **Add new configuration** from the **Select Configuration** drop-down list and then add the name of the configuration and specify the configuration parameters. If you want to update an existing configuration, then select the configuration from the **Select Configuration** drop-down list and update the configuration parameters. If there is only one configuration, then that configuration will be selected automatically. For more information on configuring connectors, see the *Introduction to Connectors* chapter in the "Connectors" Guide.

Running remote actions

Once you have completed configuring the connectors, you can run remote actions on the FSR agent by using playbooks, or by running connector actions directly on records.

In case you are running remote actions using playbooks, you can add the connector as a step and then choose whether to execute that step on the current FortiSOAR node or remotely on the FSR agent node by clicking the **Self** or **Agent** buttons besides `Target`. By default, **Self** is selected, which means that the action will run on the current FortiSOAR node, then you must select the configuration using which you want to run the action since the FortiSOAR node can have multiple configurations, from the **Configuration** drop-down list.

Listener-based connector configuration on the FSR agent is supported. Listener-based connectors listen for live events on a server, and then these events are notified to FortiSOAR by triggering a playbook. For example, the Exchange connector, which has enabled its listener-based configuration starts a live listener for the specified email account. Therefore, if any new emails are received in the configured account or folder, the connector fetches that emails and triggers playbooks, such as the ingestion playbook, which is specified in the configuration.

Important: For Listener-based connectors to work and to trigger playbooks such as triggering the data ingestion playbook, the FSR agent appliance must have "Execute" permissions on 'Playbooks'.

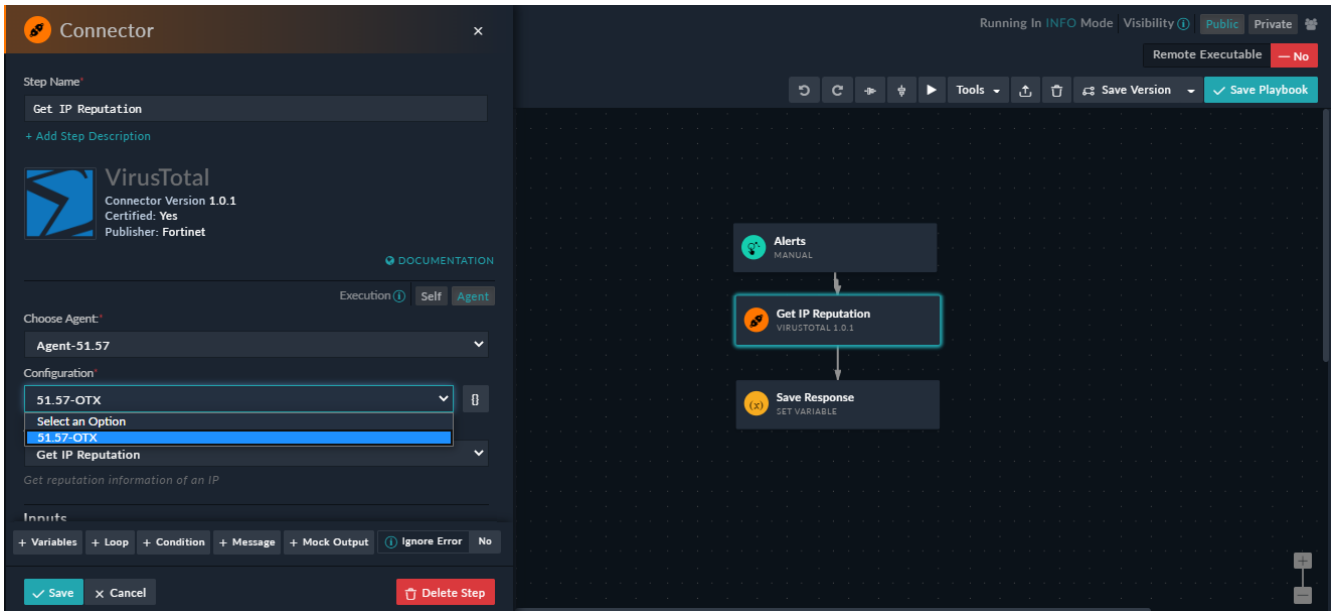
File-based operations are supported on the FSR agent, enabling you to perform connector actions that require file resources from FortiSOAR. For example, the "Get Unread Emails" action of the Exchange connector, has an option where you can choose to upload an attachment received in an email to FortiSOAR. You can perform the following operations:

- Upload a file to FortiSOAR
- Download a file from FortiSOAR.
- Support of any generic API request for GET, PUT, POST, and DELETE methods
  Note: To perform the operations, roles having appropriate permissions to the required modules must be assigned to the agent. For example, to download an attachment, the agent would require a minimum of 'Read' permission on the "Attachment" module.

However, note that any file that is downloaded on the agent using the download file action of the Utilities connector will not be available to any of the next steps in the playbook. For example, if you create a playbook add then add the Utilities connector operation "File: Download File From URL" step for an FSR Agent configuration, add the download URL, and save the step. Next, you add the "File: Create Attachment From File" step in which provide the file reference from "Download" step and save and run the playbook. The playbook will fail with an error such as `"Connector step is failing with error 'Invalid input :: Given filename/filepath /tmp/f68ab00fb7da4dfd9db4bb95abb1471e doesn't exists'"`. This is expected behavior since when a file download operation is performed on an FSR agent, the operation cleans the file when the response is returned to the base FortiSOAR node. Therefore, if any following step expects the downloaded file to be present at the agent will cause that step to fail.

You can click **Agent** and then from the **Choose Agent** drop-down list, you can select the FSR agent on which you want to run the action and you must also select the configuration using which you want to run the action since FSR agents can have multiple configurations. If you want the FSR agent configuration to be resolved dynamically, you can click **{}** in the **Configuration** field and then specify the connector configuration by either typing the connector configuration name or ID or specifying a Jinja variable that contains the connector configuration name or ID.

If you select an agent dynamically based on the configuration name in the playbook connector step, and the configuration name is incorrect, the connector step will be executed using the connector's default configuration on the system instead of the agent.



In case of a multi-tenant configuration, on the master node, if you click **Agent**, then you can also select the **Pick From Record's Ownership** option in the **Choose Agent** drop-down list, which would then read the record of the tenant and accordingly select the FSR agent to be run the action.

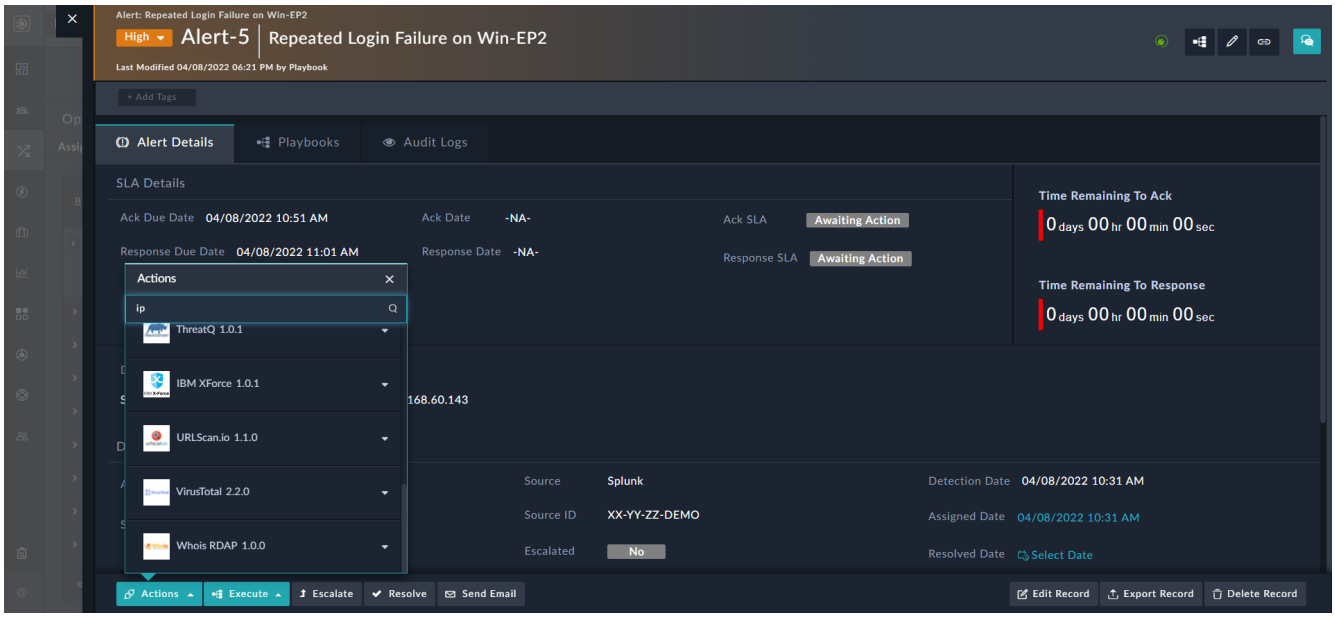
If you have only one configuration for the connector or have specified a default configuration, then that configuration automatically gets selected.

The **Configuration** drop-down lists only those FSR agents whose Status is "Remote Node Connected."

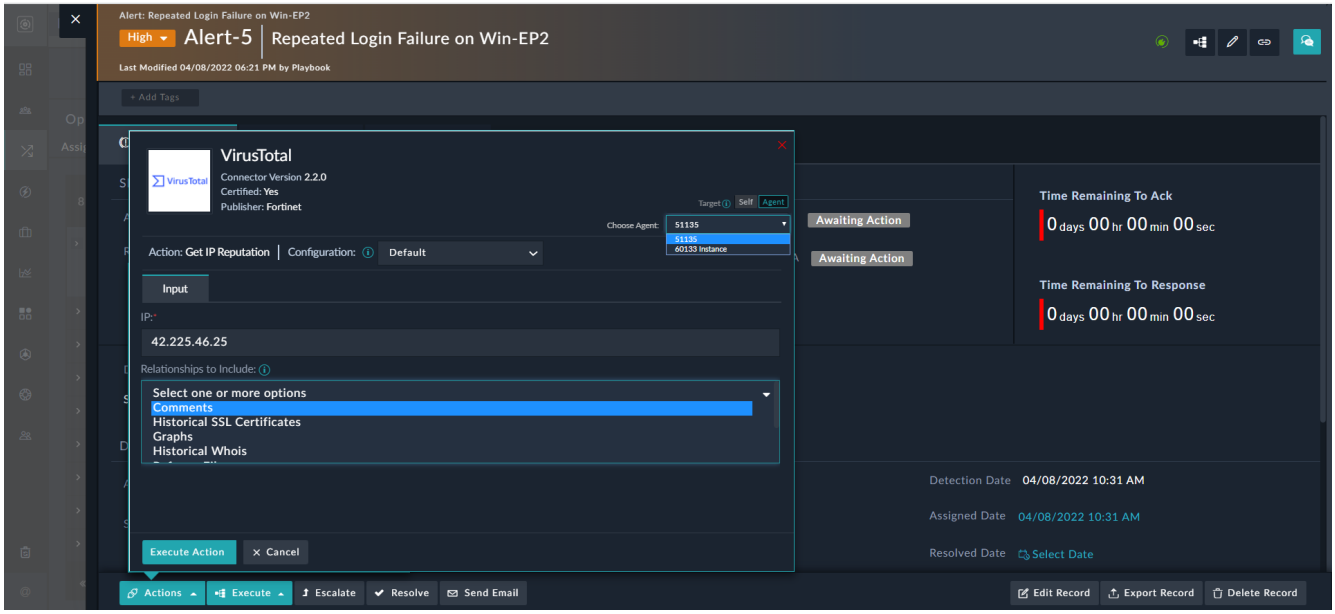
Next, you can select the action that you want to run, for example "Get IP Reputation", and provide the necessary input parameters for this action and save this step. Then you can continue to build the playbook as per your requirements.

Once you have developed the playbook that you want to execute using a remote FSR agent, and when you execute the same, you will observe in the 'Executed Playbook Log', that the playbook goes into the "Awaiting" state when it reaches the Connector step. This happens because this step is being executed on the remote FSR agent and then the playbook is consuming the result received back from the remote FSR agent for further processing.

Similarly, you can run direct connector actions on FSR agent records. To run direct connector actions on records, open the detail view of a record, for example, the detail view of an alert record and click the **Actions** button. The Actions list will display the active connectors. Search for actions that you want to perform using the **Search Action** box. For example, if you want to search for a reputation of an IP address, then you can type `IP` in the **Search Action** box, and the connectors that have any action related to an IP address will be displayed:



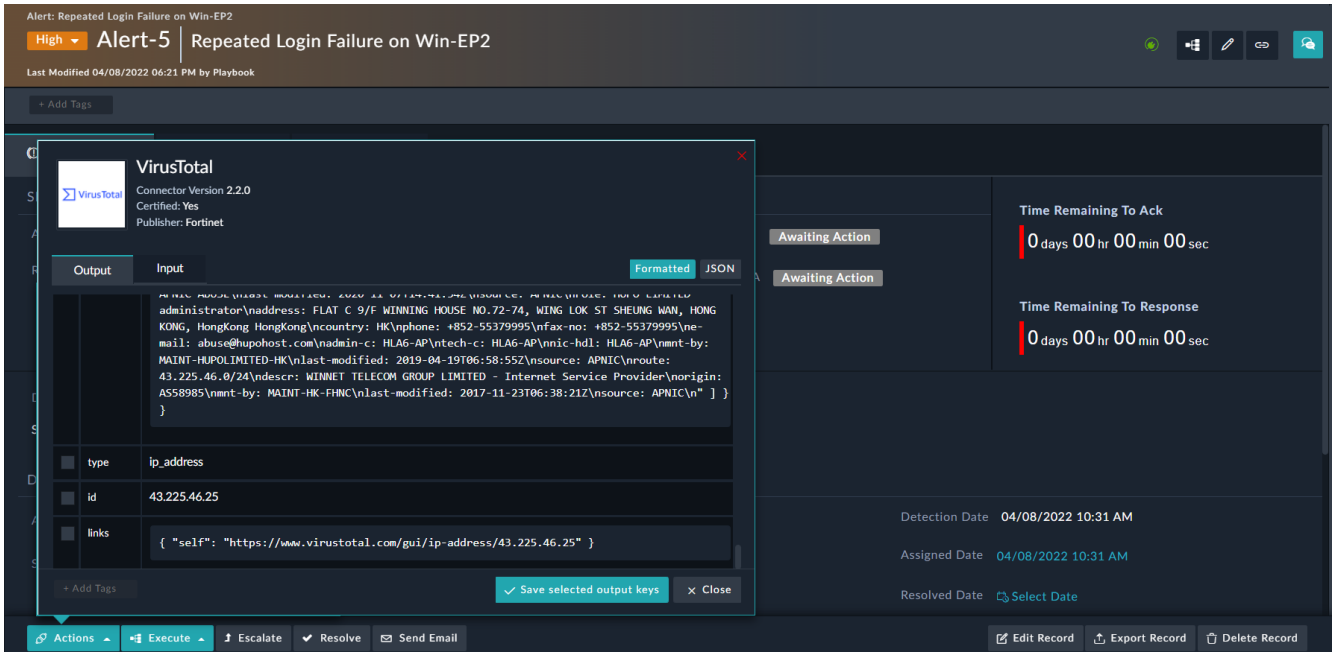
As shown in the above image, VirusTotal, IBM XForce, URL Scan.io, etc connectors have "IP" in their actions. Click the down arrow to view the actions associated with IP for each connector. For example, if you click VirusTotal, you will see the "Get IP Reputation" action, which for example we want to perform on the FSR agent. Select the **Get IP Reputation** action, which displays the VirusTotal dialog. From version 6.4.1 onwards, you are required to specify whether you want to run the action on the current FortiSOAR node or remotely on the FSR agent node by clicking the **Self** or **Agent** buttons besides `Target`. By default, **Self** is selected, which means that the action will run on the current FortiSOAR node, then you must select the configuration using which you want to run the action since the FortiSOAR node can have multiple configurations. If you click **Agent**, then you can select the FSR agent on which you want to run the action and you must also select the configuration using which you want to run the action since FSR agents can have multiple configurations.



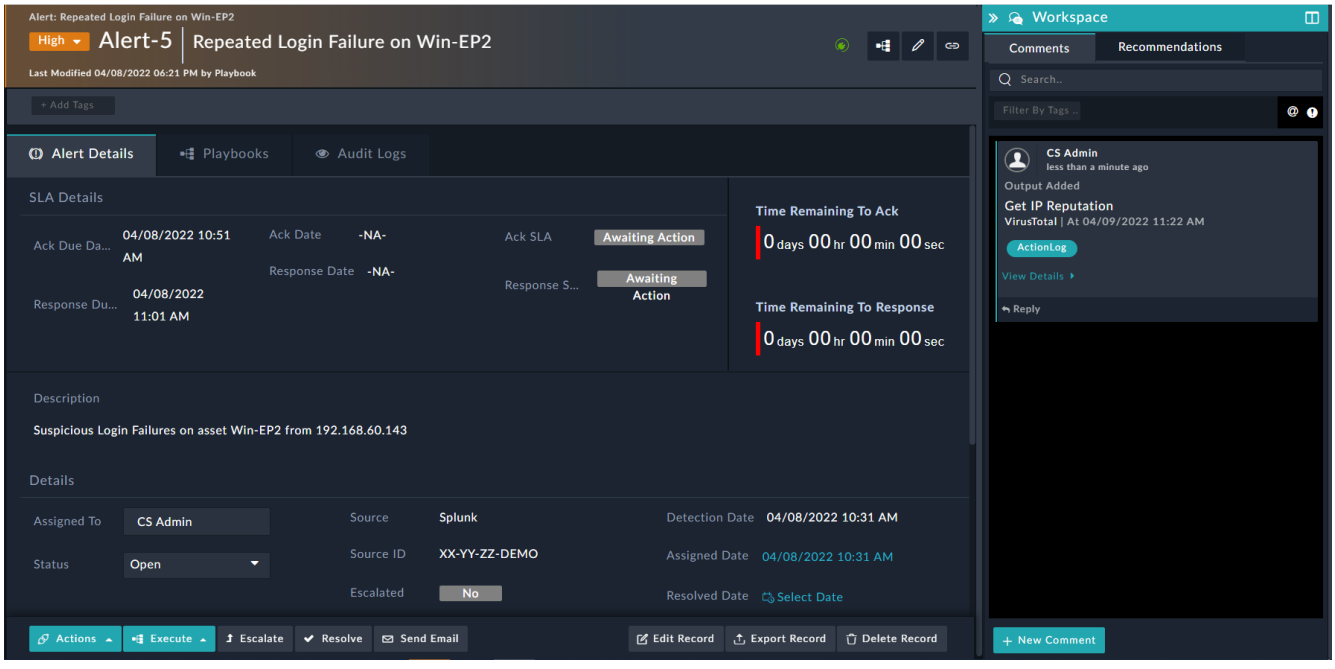
For our example, select **Agent**, then in the **Input** tab either enter an IP address or use the value of a record field. From the **Relationships to Include** list, you can also select the relationships such as Comments, Graphs, etc

that you want to include in the output, and then click **Execute Action**. The action will then be executed on the remote FSR agent, and you will be able to see the output of the action in the **Output** tab.

Note: RBAC is applicable for FSR Agents, i.e., if you do not have appropriate permissions for FSRAgents, then FSR agents will not be visible for running connector action on records.



Once you run the connector action, and if you have save the action output, then the action log is displayed in the collaboration panel as shown in the following image:



From version 6.4.3 onwards, if you run the connector action on the FSR agent, then collaboration panel also displays the name of the FSR agent after the connector name. For example, "`VirusTotal on <agent name> | At 08/07/2020 12:15 PM`"

For more information on `Running connector actions on a record`, see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide."

Upgrading an FSR Agent

You cannot directly upgrade the FSR Agent to release 7.5.0. However, you can directly upgrade an FSR agent from release 7.5.0 onwards, i.e, for example from 7.5.0 to 7.6.0.

If you have upgraded your base FortiSOAR node, then updating your FSR agent to the version of your base FortiSOAR node is highly recommended to ensure compatibility with the base product version and to benefit from the latest enhancements. Also, note that you cannot upgrade your FSR agent node to the latest FortiSOAR version until you have upgraded your base FortiSOAR node.

In the case of FortiSOAR release 7.5.0, direct upgrade to the FSR Agent to release 7.5.0 is not supported, since FortiSOAR release 7.5.0 has upgraded its OS platform to Rocky Linux or RHEL, and therefore the **Update** link on the `Agents` page is disabled. The agent installer is only supported for Linux kernel versions 4.18.0-372.13.1 and higher. To upgrade the FSR Agent, you must deploy a new VM with RHEL or Rocky Linux as the base operating system. Then, download the FSR agent installer and select the **Include pre-existing connectors on agent** option. Selecting this option bundles up the connectors and their configurations of those connectors that were previously installed and configured on the FSR agent.

Before you upgrade an agent ensure the following:

- Ensure that repo.fortisoar.fortinet.com is reachable or resolvable from the VM on which you want to install the agent.
- Ensure that the agent status is "Remote Node Connected".
  NOTE: If after inplace upgrade the FSR Agent does not get connected, i.e., its status displays as "Remote Node Unreachable", then restart the service on the agent using the following command:
  `# systemctl restart cyops-integrations-agent.service`

You can upgrade an agent using the following two methods:

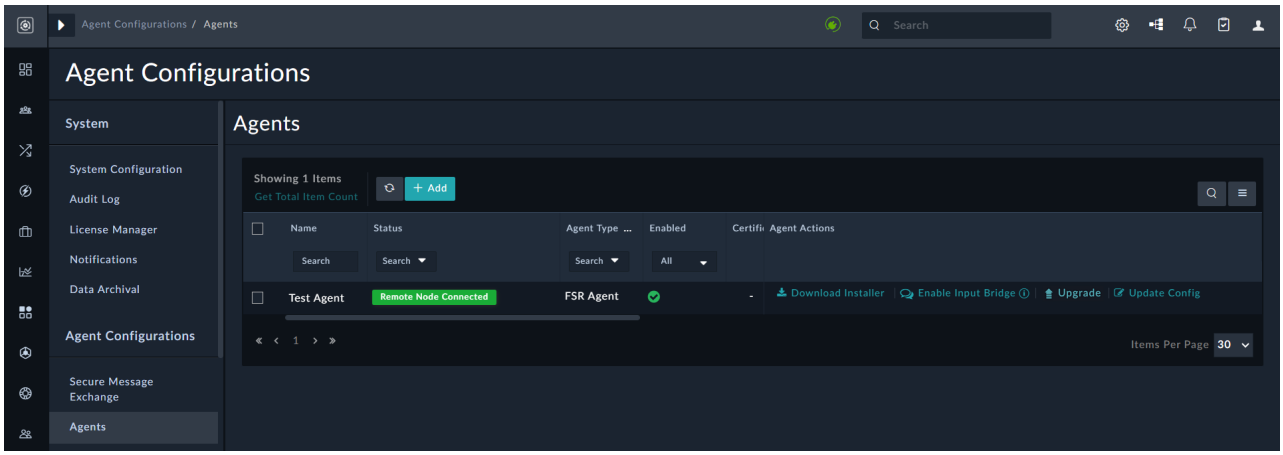
- Automatic Upgrade - Minimum applicable FSR Agent and FortiSOAR Node version for automatic upgrade is 6.4.3.
- Manual Upgrade - Minimum applicable FSR Agent version for manual upgrade is 6.4.1.

Performing an Automatic Upgrade

To perform an automatic upgrade, follow these steps:

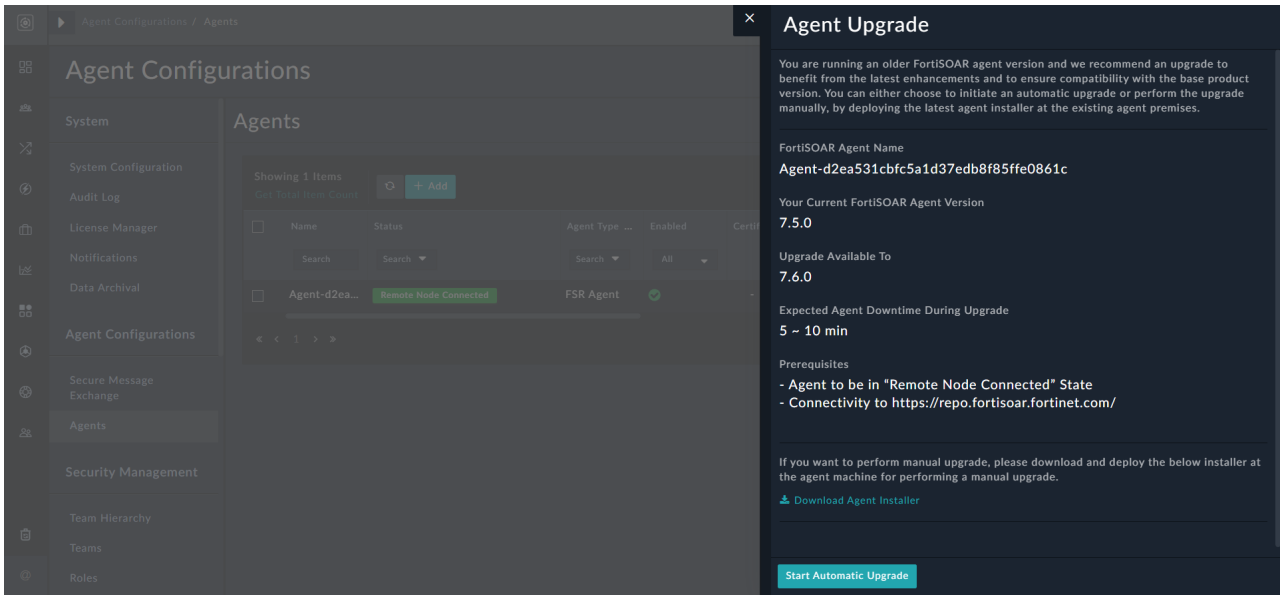
1. Log on to your base FortiSOAR node as an administrator and click the **Settings** icon to open the `System` page.
2. In the `Agent Configurations` section, click **Agents** in the left menu.

3. On the `Agents` page, in the `Agent Actions` column of the FSR Agent you want to upgrade, click the **Upgrade** link:



Clicking the **Upgrade** link opens the `Agent Upgrade` dialog.

4. The `Agent Upgrade` dialog contains the current version information of your FSR Agent and the version it to which it will be upgraded, as well as other information such as prerequisites to the upgrade and expected downtime:



Click **Start Automatic Upgrade** to display a confirmation dialog, and once you click **Confirm**, the FSR Agent begins to get automatically upgraded.

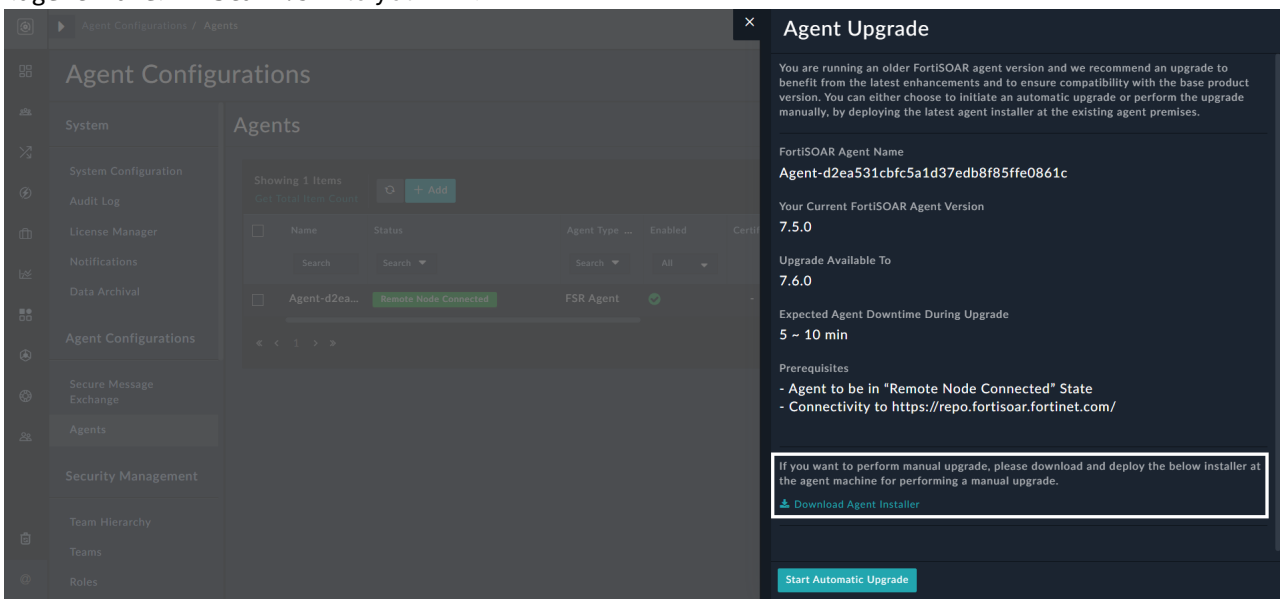
NOTE: You can choose to upgrade the configuration manually by downloading the agent installer and running it on the agent machine.

You can see messages related to the upgrade on the same screen and once the FSR Agent is successfully upgraded to the same version as your base FortiSOAR node, you will no longer see the **Upgrade** link in the `Agent Actions` column.

Performing a Manual Upgrade

If automatic upgrade of your FSR Agent fails for any reason, then you can try to manually upgrade your FSR Agent using the following steps:

1. Log on to your base FortiSOAR node as an administrator and click the **Settings** icon to open the `System` page.
2. In the `Agent Configurations` section, click **Agents** in the left menu.
3. On the `Agents` page, in the `Agent Actions` column of the FSR Agent you want to upgrade, click the **Upgrade** link to open the `Agent Upgrade` dialog.
4. On the `Agent Upgrade` dialog, click **Download Agent Installer** to download the upgrade file named as `<agent-name>-install.bin` to your VM:

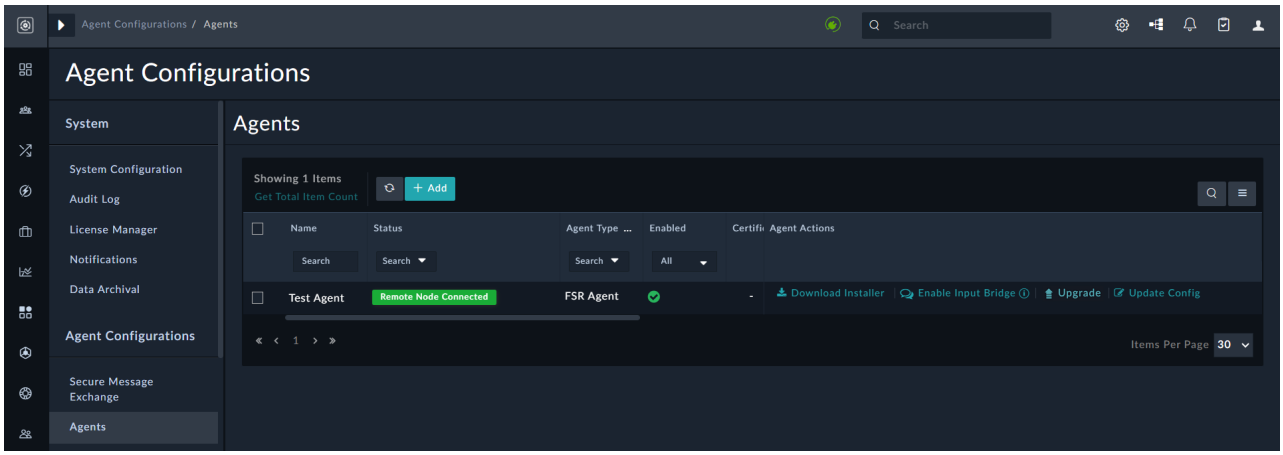


5. Copy and paste the `<agent-name>-install.bin` to your FSR agent's VM (FSR Agent Node).
6. SSH to the FSR agent's VM as a *root* user and run the following command:
   `# sh install.bin` or `# ./install.bin`
   Now, your FSR agent is upgraded to the same version as your base FortiSOAR node, and the **Upgrade** link is no longer displayed in the `Agent Actions` column.

Upgrading the configuration on an FSR Agent machine

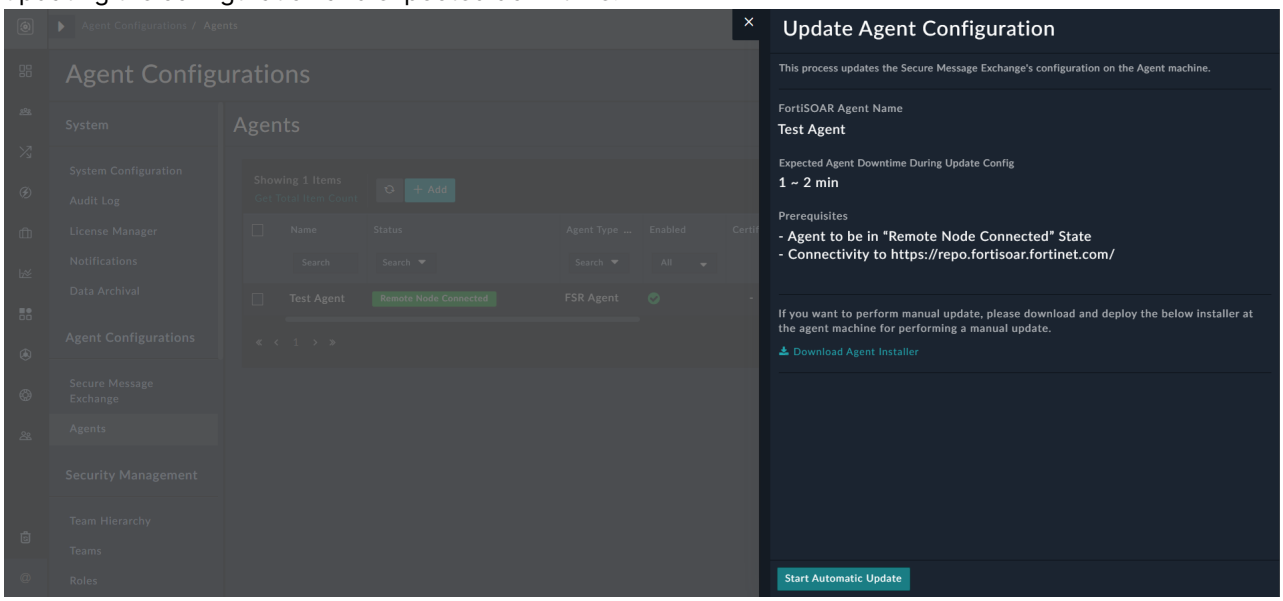
To update the configuration on the FSR Agent machine with the latest settings, such as changes in the hostname, SNI, address, etc., follow these steps:

1. Log on to your base FortiSOAR node as an administrator and click the **Settings** icon to open the `System` page.
2. In the `Agent Configurations` section, click **Agents** in the left menu.
3. On the `Agents` page, in the `Agent Actions` column of the FSR Agent you want to upgrade, click the **Update Config** link:



Clicking the **Update Config** link opens the `Update Agent Configuration` dialog.

4. The `Updated Agent Configuration` dialog contains the name of the FSR Agent, prerequisites to the updating the configuration and expected downtime:



Click **Start Automatic Update** to display a confirmation dialog, and once you click **Confirm**, the configuration on an FSR Agent machine to get automatically updated.
NOTE: Use this process when you are pushing changes for the same physical secure message exchange. However, if you are changing the secure message exchange, you must update the configuration manually by downloading the agent installer and running it on the agent machine.

Running unauthenticated manual inputs in segmented networks

From FortiSOAR release 7.3.0 onwards, non-FortiSOAR users can provide unauthenticated manual inputs using an FSR Agent. To provide inputs requested by a FortiSOAR playbook, you must configure the FSR Agent Communication Bridge by clicking the **Enable Input Bridge** option in the 'Agent Actions' column. For more information, on how to configure the agent communication bridge, see the FSR Agent Communication Bridge

connector document, and for manual input information, see the *Triggers &Steps* chapter in the "Playbooks Guide."

The **Enable Input Bridge** option will be visible on FortiSOAR release 7.3.0 and later.

Troubleshooting

Files to be used for troubleshooting

Use the "connectors.log" to troubleshoot FortiSOAR connectors. This log is located at: `/var/log/cyops/cyops-integrations/connectors.log`.

Use the "`config.ini`" file to update the connector settings, such as changing the logging level of connectors. This file is located at: `/opt/cyops-integrations/integrations/configs/config.ini`. Once you complete updating the "`config.ini`" file, you must restart the `cyops-integrations-agent` service.

Use the "`agent_config.yml`" file on the FSR agent instance for details such as the FortiSOAR node to which the FSR agent requires to be connected, the secure message exchange to be used, etc. This file is located at: `/opt/cyops-integrations/integrations/configs/agent_config.yml`.

The FSR agent remains stuck in the "Remote Node Unreachable" state after users update the certificates or if they change the authentication strategy

The FSR agent remains stuck in the "Remote Node Unreachable" state after users update the certificates or if users change the authentication strategy from 'Basic Auth' to 'Certificate Auth' or vice-versa because the communication channel between the agent and the secure message exchange does not get created.

Resolution

To resolve this issue, users have to restart the 'integration' service on their agent using the following command: `systemctl restart cyops-integrations-agent`

Deactivated FSR agent does not come back to the connected state even after activating the FSR agent

Activating a deactivated FSR agent does not change its connection state from "Remote Node Unreachable" to "Remote Node Connected" if the live connection mechanism does not correctly reflect the status.

Resolution

Restart the `cyops-integrations-agent` service on the FSR agent.

FSR Agents configuration page displays a "`Agent <Agent UUID> does not exist`" error when you click the Export Config link

This issue occurs if you have removed the default admin team, i.e., "SOC Team" and have not assigned the new admin team to the agent and playbook appliance.

Resolution

Recommended not to remove the default "SOC Team". However, if you have removed the SOC Team, then you must create a new admin team and assign that team to the agent and playbook appliance. To assign the new admin team to the agent and playbook appliance, click **Settings** > **Appliances** to open the `Appliances` page, and then click the **Agent** row to edit the agent appliance page. On the `Edit Appliance` page, in the `Team and Role` section, in the `Teams` table, select the new admin team to assign the same to the agent appliance. Perform the same steps for the playbook appliance.

Post restore to release 7.3.0, switching the SME of any Tenant or FSR Agent, the status of the tenant or agent is displayed as "Remote Node Unreachable"

If you have ugraded your FortiSOAR system from a release prior to 7.3.0 to release 7.3.0 or later using the 'migrate' script, i.e., backup and restore method, and you have switched the SME of any tenant or FSR agent, i.e., for example, configured an external SME and switched the SME of an agent or tenant to this external SME, then the status of the FSR agent or tenant displays as "Remote Node Unreachable".

Resolution

If you switch the SME of any tenant or agent, then you need to reconfigure the new SME for tenant and run installer again for agent.

If you toggle the SME of any tenant or agent to the older SME, then you we need to restart the `cyops-integrations-agent` service on the agent or tenant.

FortiSOAR Admin CLI

An administrator can use FortiSOAR Admin CLI (`csadm`) to perform various functions such as backing up and restoring data and run various FortiSOAR commands such as starting and stopping services and collecting logs.

Prerequisites

To run `csadm` you must login as `root` or have `sudo` permissions.

FortiSOAR Admin CLI - Usage

Once you type `# csadm` on the command prompt, the usage and subcommands of the FortiSOAR Admin CLI are displayed as shown in the following image:

```
[root@fortisoar csadmin]# csadm
usage: csadm [<subcommand> <options>]     Run subcommand
             [<subcommand> --help]        Show detailed help of subcommand
             [--help]                     Show this message

csadm subcommands are:
    certs                       - Generate and deploy certificates
    db                          - Manage database
    hostname                    - Change hostname
    license                     - Manage license
    user                        - Manage users
    log                         - Manage log
    mq                          - Manage message queue
    secure-message-exchange     - Manage Default (Embedded) Secure Message Exchange
    network                     - Manage network
    services                    - Manage services
    ha                          - Manage HA cluster
    system                      - Manage system settings
    package                     - Manage package
    upgrade                     - Manage FSR upgrade
[root@fortisoar csadmin]#
```

To perform a particular task in FortiSOAR using csadm, you must type `# csadm` and then its subcommand and the subcommand's arguments (if any). For example, to change a hostname use the following command:

```
# csadm hostname --set [<hostname to be set>]
```

You can get help for a particular subcommand by running following command:

```
# csadm <subcommand>
```

OR

```
# csadm <subcommand> --help
```

`csadm` supports the following subcommands:

Subcommand	Description

certs	Generates and deploys your certificates. You can use the following arguments with this subcommand: • `--deploy`: Deploys SSL certificates. For more information, see the `Updating the SSL certificates` section in the *Additional configuration settings for FortiSOAR* chapter in the "Deployment Guide." • `--generate <host name>`: Generates and deploys self-signed certificates.

db	Performs operations related to database.
	You can use the following arguments with this subcommand:
	• `--archival-externalize [ARCHIVAL_DB]`: Externalizes your data archival database.
	• `--backup [<backup_dir_path>]`: Performs a backup of your FortiSOAR system, including backup of both data and configuration files in the directory you have specified.
	Important: If you have externalized your PostgreSQL database, it is recommended to use the `csadm db --backup-config [<backup_dir_path>]` command for taking periodic backups of the configuration. Using the `--backup` argument is not recommended for an externalized PostgreSQL database.
	Optionally, you can use following options with `--backup` argument:
	  • `--exclude-workflow` option to exclude all the "Executed Playbook Logs" from the backup.
	  • `--exclude-audit` option to exclude all the "Executed Audit Logs" from the backup.
	  For more information, see the Backing up and Restoring FortiSOAR chapter.
	• `--backup-config [<backup_dir_path>]`: Performs a backup of only your configuration files in the directory you have specified.
	• `--change-passwd`: Changes the password of your PostgreSQL database.
	Once you run this command, you will be prompted to enter the password of your choice and confirm the password, which will then update your PostgreSQL database password to the new password.
	• `--check-connection`: Checks the database connection that is mentioned in the `db_external_config.yml` file.
	• `--restore [<backup_file_path>]`: Performs data restore from a locally stored file, whose path you have specified. The default location of the backup file is (`/home/csadmin/db_backup/DR_BACKUP_<yyyymmdd_hhmmss>.tgz`). For more information, see the Backing up and Restoring FortiSOAR chapter.
	• `-encrypt`: Generates an encrypted version of the text that you have specified on the prompt. Use this command to generate an encrypted version of the password that you have set for your PostgreSQL database.
	• `--externalize`: Performs externalization of your FortiSOAR PostgreSQL data. You must provide the path in which you want to save your database backup file. For more information, see the Externalization of your FortiSOAR PostgreSQL database chapter.
	`--check-connection`: Checks the connection between FortiSOAR and the external PostgreSQL database.
	• `--getsize`: Displays the size of the Primary Data, Audit Logs, and Workflow Logs in your database. From release 7.4.2 onwards, this command also displays the size of the archived data. Using this command, you can view information about your current usage and compute your data usage over time according to your purging policy.

	From release 7.0.0 onwards, you can also backup and restore the data of your external Secure Message Exchange (SME) system, by using the following arguments with the `db` subcommand: • `--backup [<backup_dir_path>]`: Performs a backup of your external SME system. • `--restore [<backup_file_path>`: Performs data restore for your external SME system from a locally stored file, whose path you have specified. The default location of the backup file is (`/home/csadmin/db_backup/DR_BACKUP_<yyyymmdd_hhmmss>.tgz`). For more information, see the Backing up and Restoring FortiSOAR chapter. **Note**: All other options of the `db` option are not applicable to the external SME.
ha	Manages your FortiSOAR High Availability cluster. For more information about HA and its commands, see the High Availability and Disaster Recovery support in FortiSOAR chapter.
hostname	Changes the name of the host and Fully Qualified Domain Name (FQDN) based on the parameters you have specified. You can use the following arguments with this subcommand: • `--set [<hostname>]`: If you specify a new hostname, then this changes your current hostname to the new hostname that you have specified, sets up the message broker, regenerates certificates, and restarts FortiSOAR services. If you do not specify a hostname, then this sets up the message broker, regenerates certificates using the existing hostname, and restarts FortiSOAR services. **Note**: Before you run this subcommand, you must ensure that the specified hostname is resolvable. • `--dns-name <DNS_SERVER_IP>`: Adds the DNS server entry to the `/etc/resolv.conf` file.

license	Manages your FortiSOAR license. You can use the following arguments with this subcommand:
	• `--get-device-uuid` : Displays the Device UUID of your FortiSOAR instance.
	• `--deploy-license <License File Path>`: Deploys your FortiSOAR license. For example, `csadm license --deploy-license temp/<Serial_No>.lic`. Your license file specifies the FortiSOAR edition: 'Enterprise', 'HA' or 'Multi-Tenant'.
	• `--show-details`: Displays details of the installed license, including type, edition, Device UUID, total users, expiry date, etc.
	  • Add the `[License File Path]` parameter to this argument, for example, `--show-details /home/<Serial_No>.lic`, to view the contents of the license file.
	  • Add the `--debug` parameter to this argument for example, `--show-details --debug`, or after the `[License File Path]` parameter, for example, `--show-details /home/<Serial_No>.lic --debug`, to view the FDN response in addition to the license details. This provides more details about the license and helps in troubleshooting licensing issues.
	• `--get-device-sn`: Displays the Serial Number of your FortiSOAR instance.
	• `--refresh-device-uuid`: Refreshes the device UUID if it has changed. For example, after a hardware change, you can run the `csadm license --refresh-device-uuid` command on the specific node of the HA cluster to continue using the old license.

user	Manages your FortiSOAR users. You can use the following options with this subcommand:

	• `show-logged-in-users`: Displays a list of currently logged in users whose access type is 'Concurrent'.
	The following arguments can be used with this option:
	  • `--access-type {Named,Concurrent}`: Access type, i.e., Named or Concurrent, of the users that you want to include in the list of currently logged in users. For example, if you specify `csadm user show-logged-in-users --access-type Named`, then the list of 'named' users currently logged into FortiSOAR will be displayed. By default, the access type is set as Concurrent.
	  • `--limit [1-30]`: Last *n* users who have logged into FortiSOAR. Use this argument to limit the number of users that you want to display in the list of currently logged in users. For example, if you specify `csadm user show-logged-in-users --limit 5`, then the list will display the last 5 logged in users. By default, the limit is set to 10. You can specify any value between 1 to 30.
	• `logout-user --username USERNAME`: Forcefully logs out a specific 'Concurrent' user from FortiSOAR; 'Named' users cannot be logged out. You must specify the username argument with this option, i.e., you must include the username of the user you want to log out of FortiSOAR. For example, to log out testuser1, specify `csadm user logout-user --username testuser1`

mq	FortiSOAR message queue controller (RabbitMQ) functions. You can use the following options with this subcommand:
	• `certs`: Manages the TLS certificates of the RabbitMQ server.
	You can perform the following actions with this option:
	  • `generate`: Generates the self-signed certificate that will be used for authenticate connections between the secure message exchange (RabbitMQ server) and any client or agent or distributed tenant.
	This generates the signed certificates for the secure message exchange with the CA name as '`FSRDefaultCA`'. If you have already deployed your own organization's signed certificate for the secure message exchange, then this subcommand will not overwrite your certs, instead a proper message will be displayed and the subcommand will exit.
	Note: If there are any FSR agents connected to this secure message exchange then you must download and run the agent installer again for the new certificates to be available to the agents.
	  • `deploy`: Deploys the server certificates. You must use the following arguments with this action:
	  • `--ca-cert MQ_CA_CERT_PATH`: Location of the CA certificates. By default, the CA certificate for the FortiSOAR self-signed certificate is present at the following location:
	`/opt/cyops/configs/rabbitmq/ssl/cyopsca/cacert.pem`
	Important: The input `.pem` files must be in the 'Unix' format. You can use `dos2unix` CLI to convert it to the Unix format.
	  • `--server-cert MQ_SERVER_CERT_PATH`: Location of the server certificates.
	  • `--server-key MQ_SERVER_KEY_PATH`: Location of the server key.
	• `db`: Manages the RabbitMQ database. You can perform the following action with this option:
	  • `flush`: Deletes and recreates the RabbitMQ database.

	• `client-certs`: Manages the client certificates. You can perform the following actions with this option:
	  • `generate`: Generates a client certificate using 'FSRDefaultCA'. Note that if you have deployed your own organization's signed certificates, then this subcommand will not overwrite your certs, instead a proper message will be displayed and the subcommand will exit.
	  You must use the following arguments with this action:
	  • `--common-name MQ_CLIENT_CERT_COMMON_NAME [--target-dir MQ_CLIENT_CERT_TARGET_DIR]`: Common name is used as the username when agent or tenant tries to connect with the secure message exchange. It is recommended that you specify the common name as the name of your agent or tenant. You can optionally also provide the target directory in which you want to store the generated client certificates. By default, the client certificates are generated in the current working directory.
	  • `mtls`: Enables or disables mTLS (mutual TSL). You can perform the following actions using this option:
	  • `enable`: Enables mTLS. If you have enabled mTLS, then you must reconfigure your secure message exchange and provide a pair of exchange event listener client certificates and exchange event listener client keys. For more information, see the *Deploying FortiSOAR* chapter in the "Deployment Guide." You must also reconfigure all the agents including the tenants' agents by updating the client certificate and client key which are required to connect to the secure message exchange. To reconfigure agents, download and run the agent installer again for the new certificates to be available to the agents.
	  • `disable`: Disables mTLS.
	  • `status`: Retrieves the current status of RabbitMQ mTLS, i.e., it displays either "`enabled`" or "`disabled`".
	  • `truststore`: Manages the RabbitMQ truststore. If your client CA certificates are different from the CA certificates of the secure message exchange, then you must add the client CA certificates to the truststore of the secure message exchange for authentication to work.
	  You can perform the following actions using this option:
	  • `add --ca-cert CA_CERT_PATH`: Adds a CA certificate to the truststore based on the CA certificate path you have specified.
	  • `remove --ca-cert-name`: Removes a CA certificate from the truststore. You must specify the name of the CA cert to remove it from the truststore. You can use the `csadm mq truststore list` command to get the name of the CA certificates present in the truststore.
	  • `list`: Lists the CA certificates that are present in the truststore and provides information about them such as name of the CA certificates, expiry of the CA certificates, etc.
	  • `refresh`: Refreshes the truststore. You need refresh the truststore when new certificates are deployed on the secure message exchange.

log	Performs log collection and forwarding of syslogs. You can use the following option and arguments with this subcommand:

	• `forward`: Forwards FortiSOAR logs to your central log management server (syslog server) that supports a Rsyslog client. For the options that you can use with this subcommand see the CLI commands used for forwarding FortiSOAR logs section. You can also configure forwarding of FortiSOAR logs using the FortiSOAR UI, details of which are in the System Configuration chapter. • `--collect [LOG_PATH]`: Collects logs and bundles them up into a `fortisoar-logs.tar.gz`file. You must specify the path where the logs should be collected. If you do not specify a path, then the logs will be collected in the current working directory. • `--password LOG_FILE_PASSWORD`: Password-protects the log file, i.e., the password would be required to extract the log file contents. The collected logs are bundled into `fortisoar-logs.tar.gz.gpg`. Therefore, to collect logs and to password-protect the logs, use the following command: `csadm log --collect [LOG_PATH][--password LOG_FILE_PASSWORD]`
secure-message-exchange	Manages the default secure message exchange server available with a FortiSOAR node. A secure message exchange establishes a secure channel that is used to relay information to the agents or tenant nodes. **Note**: For a production setup, it is recommended that you add and configure a separate secure message exchange for handling scale and high availability. You can use the following options with this subcommand: • `enable`: Enables the secure message exchange on your FortiSOAR instance if you want to use localhost, i.e., the Default (Embedded) secure message exchange to connect to an external agent or in case of a dedicated tenant. You must specify the `password`, which is the admin password that is used for setting up a communication channel for every tenant or agent node that will connect to this FortiSOAR instance using this local secure message exchange. All the other parameters are optional and if they are not specified, then the default values are set. If you do specify the values for any parameter, then the default values are replaced by the user-specified values. The following arguments are used with this option: ◦ `--name`: Name that you want to set for the secure message exchange. By default, this is set to `Default (Embedded)`. ◦ `--user`: Admin username that will be used to login to the secure message exchange management console and perform tasks such as configuring tenants and agents on the secure message exchange. Default value is *admin*. ◦ `--password`: Admin password that will be used to login to the secure message exchange management console. ◦ `--api-port`: RabbitMQ API port that should be enabled for configuring tenants and agents on the secure message exchange. Default value is *15671*. ◦ `--tcp-port`: RabbitMQ TCP port that should be enabled for data exchange with tenants and agents. Default value is *5671*. ◦ `--intra-tcp-port`: Intra TCP port that will be used for a localhost connection. ◦ `--no-interaction`: To be used if you do not want to be asked for any confirmations during the operation. • `disable`: Disables the embedded secure message exchange that you had enabled on your FortiSOAR instance for using localhost to connect to an external agent. • `show-config`: Displays the configuration details of your embedded secure message

	exchange, such as the name of the secure message exchange, username used to login to the secure message exchange, the TCP port and API port that is configured for your secure message exchange, etc. • `update-sni`: Updates the Server Name Indication (SNI) address for your embedded secure message exchange. The following arguments are used with this option: • `--sni <SNI>`: Specify the value of that SNI that you want to update for your secure message exchange • `show-status`: Displays the status of the your embedded secure message exchange. Options are not 'enabled', 'Configured', or 'Configuration Failed'.
services	FortiSOAR services controller (RabbitMQ) functions. You can use the following arguments with this subcommand: • `--start`: Starts all FortiSOAR services in their respective order. • `--stop`: Stops all FortiSOAR services in their respective order. • `--restart`: Restarts all FortiSOAR services in their respective order. • `--start-service <service-name>`: Starts the specified FortiSOAR service on your instance. For example, using `--start-service uwsgi` will start the 'uwsgi' service. • `--stop-service <service-name>`: Stops the specified FortiSOAR service on your instance. For example, using `--stop-service uwsgi` will stop the 'uwsgi' service. • `--restart-service <service-name>`: Restarts the specified FortiSOAR service on your instance. For example, using `--restart-service uwsgi` will restart the 'uwsgi' service. • `--status`: Displays the status, i.e., Running or Not Running of all FortiSOAR services. From release 7.4.1 onwards, the `--status` argument includes information about how long the services have been active. Knowing the last active time of a service can assist with troubleshooting when a service is restarting repeatedly due to an issue.
network	Manages network operations. You can use the following options with this subcommand: • `ipv6 --enable` : Configures FortiSOAR to use the IPv6 protocol, i.e., FortiSOAR can be accessed from a IPv6 address. The system will reboot as part of the execution. **IMPORTANT**: Before you run this command, ensure that IPv6 is assigned to the FortiSOAR VM. • `set-https-proxy --host<proxy_hostname> --port<proxy_port> --user<proxy_username> --password<proxy_password>`: Configures an https proxy server to serve all https requests from FortiSOAR. To configure an https proxy, you must specify the hostname and the port number of the HTTPS proxy server. You can also optionally specify the username and password used to access the HTTPS proxy server. **NOTE**: From release 7.6.0 onwards, the proxy username and password can include special characters such as '@', '.', '$', etc. **IMPORTANT**: By default, the 'HTTP' protocol is used to communicate with the proxy server. User the `--protocol` argument, if you want to set the communication protocol to 'HTTPS'. • `set-http-proxy --host<proxy_hostname> --port<proxy_port> --user<proxy_username> --password<proxy_password>`: Configures an http proxy server to serve all http requests from FortiSOAR. To configure an http proxy, you must specify the hostname and the port number of the HTTP proxy server. You can also optionally specify the username and password used to access the HTTP proxy server. • `list-proxy`: Lists the proxies that are configured.

	• `set-no-proxy --host<hostname>`: Configures a comma-separated list of hostnames that do not require to be routed through a proxy server. **Note**: Review the existing no-proxy list using the `list-proxy` option. You can add or remove proxies from the existing list by specifying a *complete comma-separated list of proxies* that you want to configure using the `set-no-proxy` option. For example, if you have added hostname1 to the no-proxy list and you want to add hostname2 to the no-proxy list, then you must run the command as: `csadm network set-no-proxy --host "hostname1, hostname2"` • `remove-proxy`: Removes all the configured proxies, i.e., `remove-proxy` will remove both the http and https proxies that have been configured.
system	Manages system settings. You can use the following options with this subcommand: • `config`: Configures your system to optimally use available resources, such as vCPU and RAM. The resource requirements for FortiSOAR are specified in the Deploying FortiSOAR chapter in the "Deployment Guide." The following option can be used with this subcommand: • `--mode`: Sets the configuration mode for your system. The available modes are: standard or optimal. • `standard`: Use this option if you have modified the system configuration and wish to restore it to the original settings. **NOTE**: The standard configuration is automatically applied to your system when the FortiSOAR Configuration Wizard is run. • `optimal`: Configures your system to use available resources optimally, especially when resources exceed the recommended specifications. To use this mode, first install the 'FortiSOAR Health Assessment Solution Pack' and generate the optimal configuration. By default, this option picks the optimal configuration generated by the FortiSOAR Health Assessment Solution Pack. Alternatively, you can specify a custom configuration file path using the `--path [PATH]` parameter. • `disk`: Provides Disk management and helps you address disk space issues. You can use this subcommand to extend a logical volume to occupy space that is available in its own volume group. Or, if a new disk is attached, then a single partition is created and the logical volume is expanded to occupy this partition according to the specified size in GB. It is important to note that the '`csadm system disk`' subcommand supports only one volume group per disk. You can perform the following actions using this option: • `expand-lv`: Expands the specified logical volume. The following arguments can be used with this action: • `--logical-volume`: Specify the name of the logical volume that you want to expand. Running `csadm system disk expand-lv --help` automatically lists the logical volumes that are available for expansion in the help message. **Note**: You cannot expand 'swap' and 'root' logical volumes using the `csadm system disk` option. • `--disk`: Name of the disk that you want to use to expand the logical volume. Running `csadm system disk expand-lv --help` automatically lists the disks that are attached to the system. Example of using the `--disk` argument: The command for expanding the

pgsql logical volume to use 10GB of a newly attached disk named 'sdf':
`# csadm system disk expand-lv --logical-volume relations --disk sdf --size 10`

- `--use-vg`: Specify a value for this argument if you want to extend a logical volume, by the size specified in GBs, to occupy available free space that is available in its own volume group.
  Example of using the `--use-vg` argument: The command for expanding the pgsql logical volume to consume 100% disk space of the volume group:
  `# csadm system disk expand-lv --logical-volume relations --use-vg`
  Important: Note the following points with respect to running `csadm system disk expand-lv`:
  - You must use either the `--disk` or the `--use-vg` argument with the `expand-lv` option.
  - For expansion to take place atleast 1GB free space must be available on the target entry (disk or logical volume). If there is less than 1GB of space available, then `csadm system disk expand-lv` will exit after displaying an appropriate message.
  - The `--disk` argument will not operate on a disk that has more than one partition. In this case `csadm system disk expand-lv` will exit after displaying an appropriate message such as "`...This subcommand does not support the automation of handling of multiple partitions due to complications involved....Exiting now`"
- `--size`: Specify the size in gigabytes (GBs) that will be consumed from the specified disk or volume group that contains the logical volume that you want to expand. You must specify a positive integer for this argument.
  Note: If you do not specify the `--size` argument, then 100% of the space available on the specified disk or volume group will be used.
  Running this subcommand displays information of the steps that are being performed and also provides information of the sizes of the logical volume and the disk or volume group before and after the expansion.
- `--validate`: Validates the inputs passed for the `csadm system disk expand-lv` command and provides a summary of changes that will be made after running this command. This summary displays the current lvm size, the current free space on the disk, and the expected LVM size following the execution of the command. The user will see an error if the requested disk space for expansion is less than the free space that is available.
  An example of this command:
  `# csadm system disk expand-lv --validate --logical-volume home --use-vg`
  Note: It is advised to use the `--validate` argument before executing the `csadm system disk expand-lv` command so that users know the details of the available space on the partition, the new expanded size of the disks, etc. Users will also be aware of any issues that could prevent them from expanding the partition.

- `fortimonitor`: (Introduced in 7.2.0) Manages FortiSOAR integration with FortiMonitor, i.e., FortiMonitor can be used to monitor your FortiSOAR instance. For more information, see the FortiSOAR integration with FortiMonitor chapter.

	The following options can be used with this subcommand: • `agent`: Manages the FortiMonitor agent. The following actions can be performed using this option: • `install`: Installs the FortiMonitor agent on the FortiSOAR instance you want monitored. You must specify the following argument with this option: • `--customer-key CUSTOMER_KEY`: Specify the customer key of your FortiMonitor account. • `uninstall`: Uninstalls the FortiMonitor agent from the monitored FortiSOAR instance. • `rebuild-metadata`: Rebuilds the metadata for a FortiMonitor agent. If you have made any changes to FortiSOAR components to be monitored by FortiMonitor such as adding connector monitoring, then you can run rebuild-metadata to enable the changes to be reflected immediately. • `show-details`: Displays the details such as the agent's uid, server group, tags, version, and other attributes of the FortiMonitor agent. • `env`: Displays and sets environment modes for your system. The following options can be used with this subcommand: • `show`: Displays the current mode set for your system. • `--mode`: Sets the mode for your system. Following modes can be set: upgrade or operational. In High Availability (HA) environments, configuring modes for nodes in the HA cluster is required during the upgrade process. The following modes can be set: • `operational`: The normal working mode for a node, where regular operations are performed. • `upgrade`: Prepares a node for an upgrade. For more information on upgrading HA clusters, see the *Upgrading a FortiSOAR High Availability Cluster* chapter in the "Upgrade Guide." For information about HA and its commands, see the High Availability and Disaster Recovery support in FortiSOAR chapter.
package	Installs, updates, or removes connectors (RPM packages) from your FortiSOAR system. You must specify the following options with this subcommand: • `install`: Installs an RPM package on your FortiSOAR system. You can use this command to install a connector from the FortiSOAR connector repository. You can use the following arguments with this option: • `--type {connector}` / `-t {connector}`: Type of package that you want to install, i.e., connector. • `--name NAME` / `-n Name`: Name of the connector that you want to install. **Note**: The connector name of the connector must begin with cyops-connector. This command installs the latest version of the connector that is currently present in the FortiSOAR connector repository For example, to install the Fortinet FortiSIEM connector, run the following command: `csadm package install -t connector -n cyops-connector-fortinet-fortisiem`. • `update`: Updates an RPM package on your FortiSOAR system. You can use this command to update a connector from the FortiSOAR connector repository. This

	command also requires the same arguments as the `install` option, i.e., `--type` and `--name`.
	• `remove`: Removes an RPM package from your FortiSOAR system. This command also requires the same arguments as the `install` option, i.e., `--type` and `--name`.
	• `content-hub`: Performs synchronization of Content Hub with your FortiSOAR system. The following actions can be performed using this option:
	  • `sync`: Synchronizes Content Hub with your FortiSOAR system. Only a single `sync` action can be executed at a time, i.e., while the `sync` action is running in the background, you cannot re-execute the same.
	  You can use the following arguments with this action:
	  • `--force`: Forces synchronization of Content Hub with your FortiSOAR system. If you use the `--force` argument with the `sync` action, then the `sync` process that is running in the background is ignored and a fresh sync process gets started.

upgrade	Manages FortiSOAR upgrades from release 7.5.0 to a release later than 7.5.0, for example, from 7.5.0 to 7.6.0, etc. **NOTE**: The `csadm upgrade` command cannot be used to upgrade FortiSOAR on Docker. For more information on upgrades, see the "Upgrade Guide".

CLI commands used for forwarding FortiSOAR logs

Use the `csadm log forward` command to forward FortiSOAR logs to your central log management server (syslog server) that supports a Rsyslog client. You can use the following options with this subcommand:

- `add-config` (`csadm log forward add config`): Adds configuration details for the syslog server to which you want to forward the FortiSOAR. You can use the following arguments with this option:
  - `--server`: Hostname of the syslog server to which you want to forward the FortiSOAR logs.
  - `--port`: Port number that you want to use to communicate with the syslog server.
  - `--protocol`: Protocol that you want to use to communicate with the syslog server. You can specify `tcp`, `udp`, or `relp`.
  - `--tls`: To securely communicate with the syslog server, set `-tls` to **true**.
    If you enable TLS, then in the `--ca-cert` argument, you must specify the path to the CA certificate PEM file which contains the complete chain of CA certificates including the filename.
    If you have a client certificate for your FortiSOAR client, then in the `--client-cert` argument, you must specify the path to the client certificate PEM file including the filename, and in the `--client-key` argument, you must specify the path to the client key PEM file including the filename.
  - `--filter`: Comma-separated list of filters to specify the type of logs that you want to forward to your syslog server. Valid values are `application, audit, none`, and by default, all the logs, i.e., application and audit logs are forwarded. If for example, if you want to forward audit logs only then specify `--filter=audit`.
    If you specify `--filter=none`, then no logs are forwarded, i.e., log forwarding is temporarily disabled. To enable the log forwarding again, use the `update-config` option with the `--filter` argument. For example, `csadm log forward update-config –uuid < UUID of configuration > --filter <audit,application>`.
    Note: You can define the rules to forward audit logs using the FortiSOAR UI. For more information, see the System Configuration chapter.

    - `--config-name`: Name of the configuration in which you want to store the log forwarding configuration details.
      Note: Validation checks such as, whether the syslog server is reachable on the specified port etc. are run before adding the syslog server, and the syslog server is added only if the configuration details entered are valid.
- `show-config` (`csadm log forward show-config`): Displays configuration details of the syslog server such as the server's IP address, protocol, TLS information, UUID of the configuration, etc.
- `remove-config` (`csadm log forward remove-config –uuid <UUID of configuration>`): Removes the syslog configuration based on the configuration UUID you have specified. To know the UUID of your configuration use the `show-config` option.
- `update-config` (`csadm log forward update-config –uuid < UUID of configuration>`): Updates the syslog configuration based on the configuration UUID you have specified. To know the UUID of your configuration use the `show-config` option. You can update any or all of the options as mention in the `add-config` subcommand.
  Use the `update-config` option with the `--filter` argument, to enable temporarily disabled log forwarding.

You can configure only a single syslog server. If you have already configured a syslog server and you try to add a new one, then FortiSOAR displays appropriate warning messages informing you that a syslog server is already configured, and adding a new syslog server will remove already configured one. Further processing is done based on your response (*yes*/*no*) to the messages.

High Availability and Disaster Recovery support in FortiSOAR

Overview

High Availability (HA) can be achieved by using **HA clusters**. FortiSOAR provides a clustering solution with more than one FortiSOAR node joined to form an HA cluster. When you deploy FortiSOAR instance, the FortiSOAR Configuration Wizard configures the instance as a single node cluster, and it is created as an active primary node. You can join more nodes as secondary nodes to this node to form a multi-node cluster. This method is explained in detail in this chapter.

FortiSOAR implements HA Clustering with the use of PostgreSQL database clustering. It supports Active/Active and Active/Passive configurations with both internal and external PostgreSQL databases.

FortiSOAR HA clusters can be used to fulfill the following two use cases:

- **Disaster Recovery (DR)**: By configuring an Active/Passive cluster that has the passive node located in a remote datacenter.
- **Scaling**: By using co-located Active/Active cluster nodes to achieve workflow execution across multiple nodes.

Replication slots are used to set up your HA cluster. Using replication slots to set up HA clusters, adds support for differential synchronization between the primary node and the secondary nodes when the secondary nodes get out of sync with the primary node (streaming replication without slots required full synchronization). Differential sync helps enhance the performance of various HA operations such as restoring the secondary nodes after a firedrill, forming a HA cluster after upgrading a secondary node, etc. For more information, see the Usage of the csadm ha command topic.

High Availability Types supported with FortiSOAR

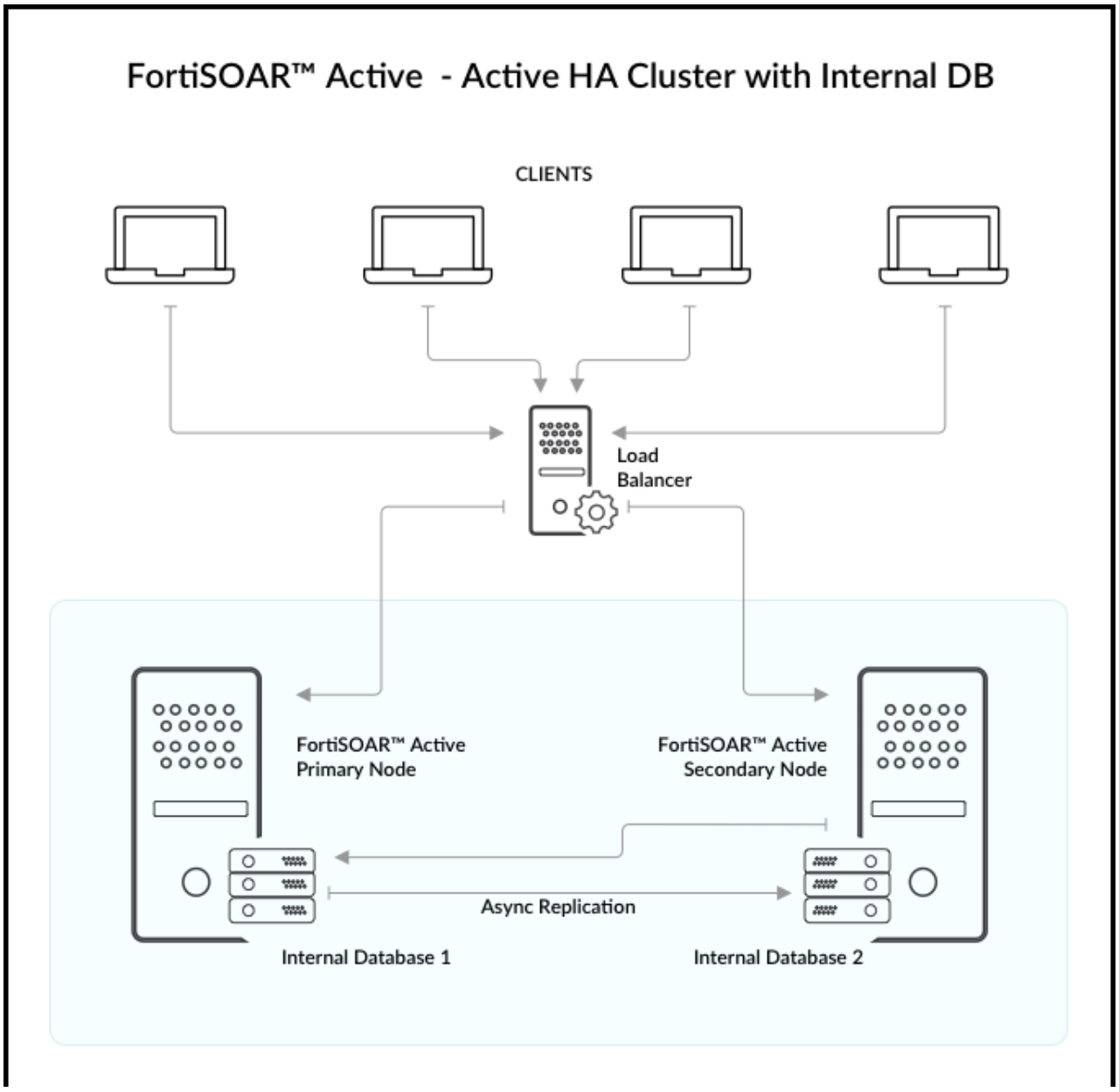
You can configure FortiSOAR with either an externalized PostgreSQL database or an internal PostgreSQL database. For both cases you can configure Active-Active or Active-Passive high availability clusters.

High Availability with an internal PostgreSQL database

FortiSOAR HA/DR is based on internal clustering that takes care of replicating data (PostgreSQL) to all cluster nodes, and provides an administration CLI (`csadm`) to manage the cluster and perform the "Takeover" operation, when necessary. FortiSOAR uses PostgreSQL streaming replication, which is asynchronous in nature. For more information, see PostgreSQL: Documentation.

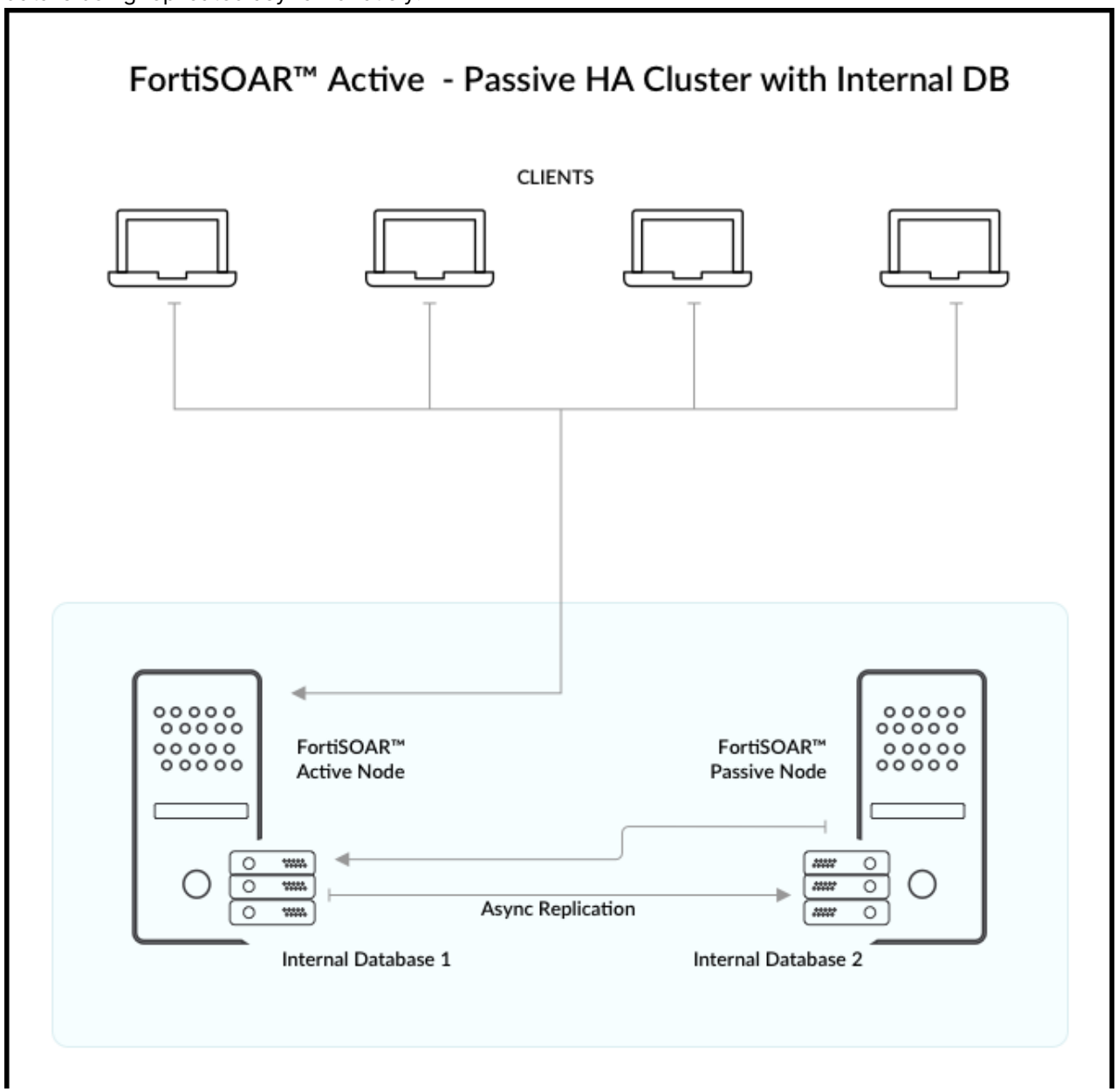
You can configure FortiSOAR for high availability (HA) with an internal PostgreSQL database in the following two ways:

- In an Active-Active HA cluster configuration, at least two nodes are actively running the same kind of service simultaneously. The main aim of the active-active cluster is to achieve load balancing and horizontal scaling, while data is being replicated asynchronously. You should front multiple active nodes with a proxy or a load balancer to effectively direct requests to all nodes. For more information about load balancers, see the Load Balancer section.



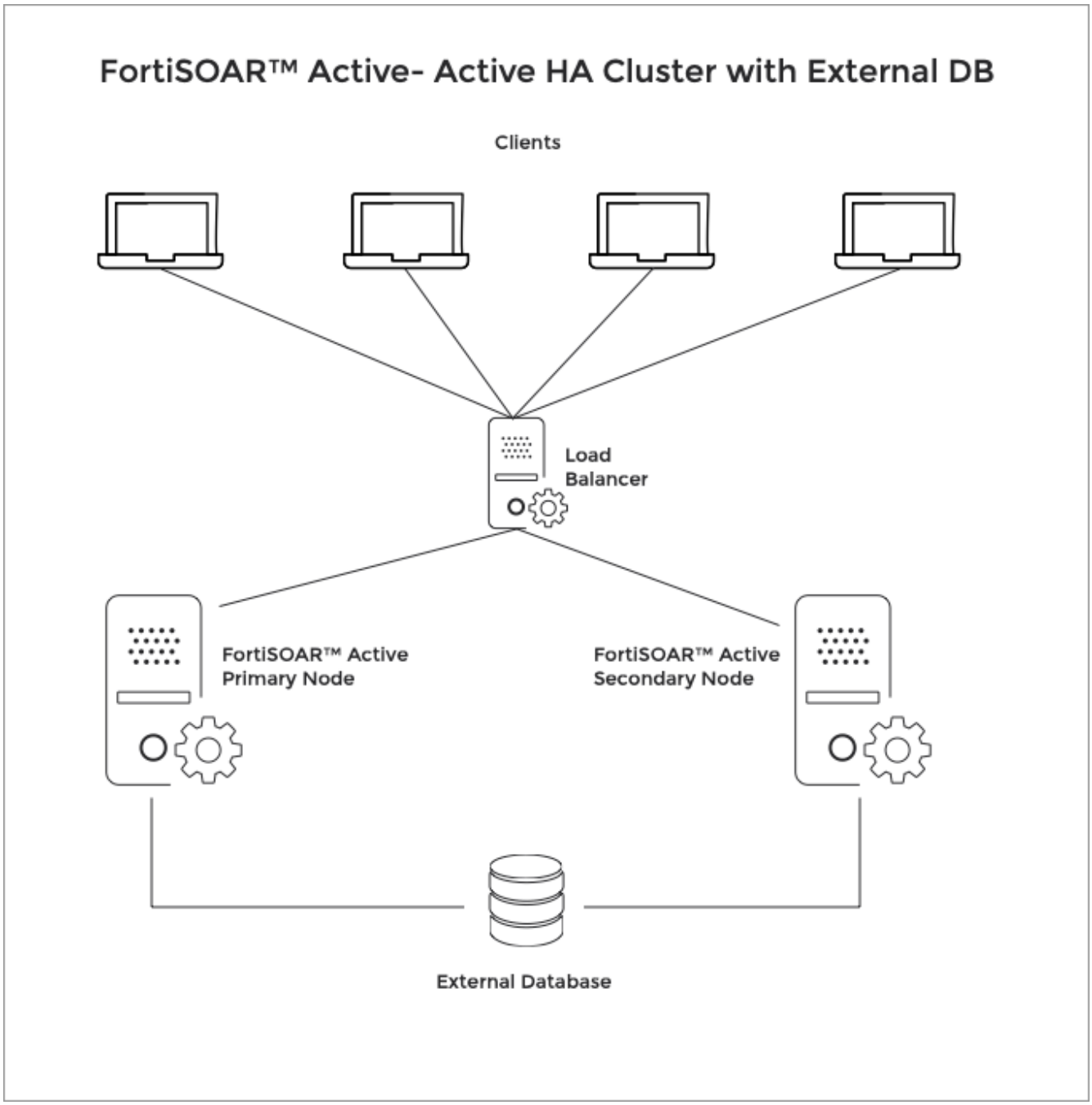
- In an Active-Passive HA cluster configuration, one or more passive or standby nodes are available to take over if the primary node fails. Processing is done only by the primary node. However, when the primary node fails, then a standby node can be promoted as the primary node. In this configuration, you can have one active node and one or more passive nodes configured in a cluster, which provides redundancy, while

data is being replicated asynchronously.



High Availability with an externalized PostgreSQL database

In case of an externalized database, the user will use their own database's HA solution. FortiSOAR ensures that changes done in the file system of any of the cluster nodes arising from the connector install/uninstall or any changes in the module definitions are synced across every node so a secondary or passive node can takeover in the least time in case of a failure of the primary node.



Cluster Licensing

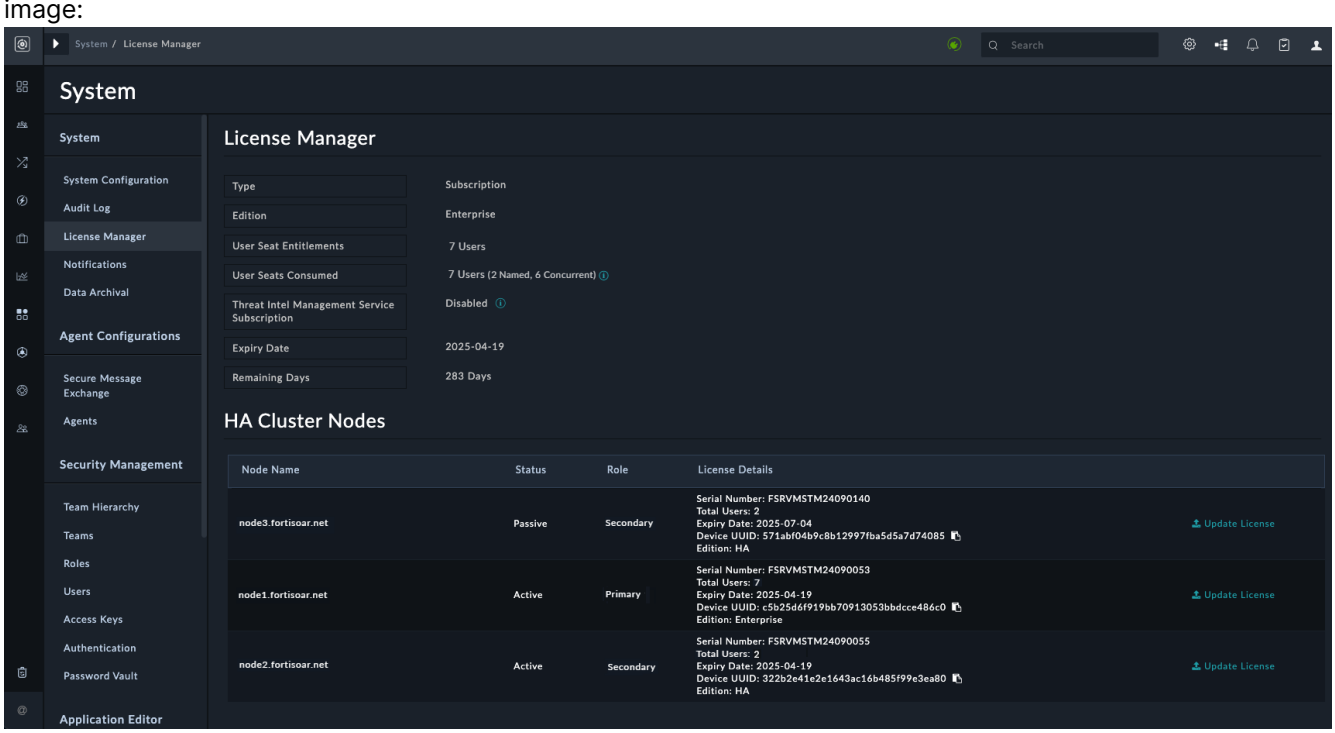
'Additional Users' entitlement is not required to be the same across all cluster nodes, i.e., you do not require to buy additional user licenses for clustered nodes. User count entitlement is validated from the primary node. The secondary nodes can have the basic two-user entitlement or alternatively use an "HA" license. The HA cluster shares the user count details from primary node of the cluster. Hence, all 'Concurrent Users' count restrictions

apply as per the primary node. If a node leaves the cluster, the restriction will apply as per its own original license. For more information about FortiSOAR licensing, see the *Licensing FortiSOAR* chapter in the "Deployment Guide."

In the case of an HA environment, you only need to buy one Threat Intelligence Management (TIM) subscription that can be used across your HA cluster. The primary node subscription gets cascaded to the secondary nodes.

Viewing and updating the license of an HA cluster

In case your FortiSOAR instance is part of a High Availability cluster, the `License Manager` page also displays the information about the nodes in the cluster, if you have added secondary node(s) as shown in the following image:

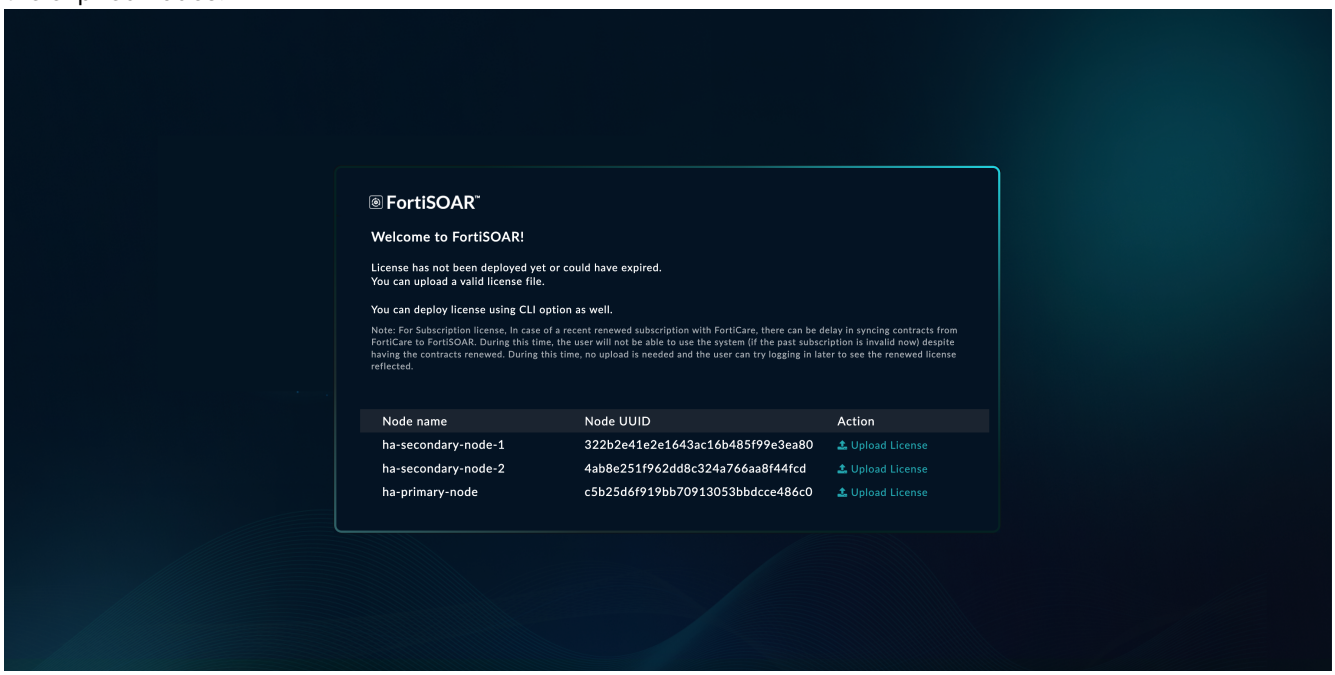


As shown in the above image, the primary node is Node 1 and that node is licensed with 7 users, therefore the User Seat Entitlements count displays as 7 users.
To update the license for each node, click **Update License** and upload the license for that node.

If you update a license that does not match with the system UUID, then you will get a warning on UI while updating the license. If you update the same license in more than one environment then the license is detected duplicate and you require to correct the license, else your FortiSOAR UI will be blocked in 2 hours.

If a license on one node of an HA cluster expires, you will not be able to access any nodes of that HA cluster. All nodes in that HA cluster will display the same FortiSOAR UI page, asking you to deploy a new valid license for

the expired nodes:



Prerequisites to configuring High Availability

- All nodes of a cluster must be on the same FortiSOAR version.
- All nodes of a cluster should DNS resolvable from each other (Both forward and reverse lookup).
- Execute the tmux command so your upgrade is not affected if your session times out.
- If you have a security group (AWS) or an external firewall between the HA nodes, then you must open the following ports between HA nodes on AWS or the external firewall:

Service (TCP)	Port
SSH	22
HTTPS	443
PostgreSQL	5432
ElasticSearch	9200
MQ traffic	5671
MQ Peer Discovery	4369
MQ inter-node	25672
Erlang distribution client ports	35672-35682 (range)

- Fronting and accessing the FortiSOAR HA Cluster with a Load Balancer such as HAProxy, FortiADC, Gobetween, or a Reverse Proxy is recommended so that the address remains unchanged on takeover. For more information about load balancers, see the Load Balancer section.

In the case of a FortiSOAR High Availability cluster environment, if a proxy is configured on a node in the cluster, then it is advisable to add other nodes in the 'no_proxy' list or ensure that the other nodes in the cluster are accessible to that node over the proxy.

Process for configuring High Availability

In order for playbooks and replications to function properly in an HA environment, '**Playbook Appliance**' must be assigned the following permissions:

- All permissions included in the default "Full app permission" role
- Security CRUD Permissions or the Security Admin Role
- Playbooks CRUD or the Playbooks Admin Role
- Application CRUD or the Application Admin role

Steps to configure FortiSOAR HA cluster with an internal PostgreSQL database

If you are configuring HA with an internal PostgreSQL database, ensure that you have met all the Prerequisites criteria (see the Prerequisites to configuring High Availability section) and then perform the following steps:

Important: You must join nodes to an HA cluster in a sequential order.

1. Use the FortiSOAR Admin CLI (`csadm`) to configure HA for your FortiSOAR instances. For more information, see the FortiSOAR Admin CLI chapter. Connect to your VM as a *root* user and run the following command:
   `# csadm ha`
   This will display the options available to configure HA:

```
[csadmin@fortisoar ~]$ sudo su
[root@fortisoar csadmin]# csadm ha
usage: csadm ha [-h]
                {join-cluster,export-conf,allowlist,list-nodes,leave-cluster,forget-node,takeover,firedrill,restore,get-replication-stat,show-health}
                ...

subcommand options are:

  list-nodes              List HA cluster details
  show-health             Show the current node health
  utils                   Utilities
  join-cluster            Join the HA cluster
  export-conf             Export the configuration file
  allowlist               Add secondary/passive server in the allowlist
  leave-cluster           Leave HA cluster
  forget-node             Remove a node from the HA cluster
  get-replication-stat    Get the replication statistics
  takeover                Perform takeover
  firedrill               Test DR
  restore                 Restore the server to either passive/secondary after firedrill
  clone-db                Clone a database from the cluster's primary node
  suspend-cluster         Suspend cluster for an upgrade
  resume-cluster          Join back to primary after an upgrade

subcommand options are:
  -h, --help            show this help message and exit
[root@fortisoar csadmin]#
```

2. To configure a node as a secondary node, ensure that all HA nodes are resolvable through DNS and then SSH to the server that you want to configure as a secondary node and run the following command:
   `# csadm ha join-cluster --status <active, passive> --role secondary --primary-node <DNS_Resolvable_Primary_Node_Name>`
   Once you enter this command, you will be prompted to enter the SSH password to access your primary node.
   In case of a cloud environment, where authentication is key-based, you require to run the following command:
   `# csadm ha join-cluster --status <active, passive> --role <primary, secondary> --primary-node <DNS_Resolvable_Primary_Node_Name> --primary-node-ssh-key <Path_To_Pem_File>`
   This will add the node as a secondary node in the cluster.
   Note: When you join a node to an HA cluster, the `list-nodes` subcommand does not display that a node is in the process of joining the cluster. The newly added node will be displayed in the `list-nodes` subcommand only after it has been added to the HA cluster.

Steps to configure FortiSOAR HA cluster with an external PostgreSQL database

If you are configuring HA with an external PostgreSQL database, perform the following steps:

1. Externalize the PostgreSQL database for the primary node of your HA configuration. For the procedure for externalizing PostgreSQL databases, see the Externalization of your FortiSOAR PostgreSQL database chapter.
2. Add the hostnames of the secondary nodes to the allowlist in the external database.
3. Add the hostnames of the secondary notes to the `pg_hba.conf` (`/var/lib/pgsql/16/data/pg_hba.conf`) file in the external database. This ensures that the external database trusts the FortiSOAR server for incoming connections.
4. Ensure that you have met all the Prerequisites criteria (see the Prerequisites to configuring High Availability section).
5. Create the HA cluster by following the steps mentioned in the Steps to configure FortiSOAR HA cluster with an internal PostgreSQL database section.

Takeover

Use the `csadm ha takeover` command to perform a takeover when the active primary node is down. Run this command on the secondary node that you want to promote to active primary.

In a FortiSOAR High Availability cluster environment, if a proxy is configured on a node, it is advisable to add other nodes to the 'no_proxy' list or ensure that all nodes are accessible to that node over the proxy.

If during `takeover` you specify **no** to the `Do you want to invoke 'join-cluster' on other cluster nodes?` prompt, or if any nodes are unreachable, you will have to reconfigure all the nodes (or the unreachable nodes) in the cluster to point to the new active primary node using the `csadm ha join-cluster` command.

During the takeover operation, FortiSOAR reindexes your complete Elasticsearch data, limited to the latest 50,000 records by default. The maximum number of records to be reindexed is a configurable parameter. See the `Elasticsearch data reindexing optimizations` topic in the Elasticsearch Configuration chapter.

During the takeover operation, licenses are swapped between the new primary node (Node B) and the old primary node (Node A). This leads to the following scenarios:

- **If Node A is alive during takeover**: It synchronizes with the Fortinet Licensing Portal, regardless of whether it rejoins the HA cluster, using the license previously associated with Node B.
- **If Node A is not alive during takeover**: It synchronizes with FDN with its old license, which is also being used by Node B, potentially causing a node lockout.
  To resolve this, run the `csadm license --decouple-license` command on the old primary node to remove its existing license. Then, run the `csadm license --sync-license` command on both the nodes, i.e., Node A and Node B to synchronize licenses with FDN properly.
  Note, that FortiSOAR allows a grace period of two hours when FDN reports a duplicate license.

After performing a takeover and configuring a secondary node as the active primary node, you will notice that log forwarder configurations are missing on the new primary node. This occurs because Syslog settings are not replicated to the passive node, which could be in a remote data center with potential network latencies. Also, the same Syslog server might not be the ideal choice for log forwarding from the DR node. If you want to forward logs from the passive node, you must enable it manually using the `csadm log forward` command. For more information, see the FortiSOAR Admin CLI chapter.

RabbitMQ clustering across all active HA nodes

In release 7.6.2, FortiSOAR enhances its High Availability (HA) architecture by introducing RabbitMQ clustering across all active HA nodes. This enhancement ensures that all queues are mirrored throughout the cluster, enabling seamless message processing and increasing system resilience.

Previously, the HA configuration only included PostgreSQL clustering for database synchronization. With this release, RabbitMQ clustering complements the existing setup by providing robust message handling and ensuring service continuity during node transitions.

RabbitMQ clusters are created across active HA nodes for both embedded (default) and external Secure Message Exchange (SME) when using the `csadm ha join-cluster` command. For details on setting up external SME cluster, see the Setting up High Availability of the Secure Message Exchange topic in the *Distributed Tenancy Support* chapter of the "Multi-Tenancy Support Guide."

Usage of the csadm ha command

Certain operations, such as takeover, join cluster, etc. might take a longer period of time to run, therefore you must execute the `tmux` command to ensure your operations are not affected if your session times out.

You can get help for the `csadm ha` command and subcommands using the `--help` argument.

It is recommended that you perform operations such as join-cluster, leave-cluster, etc sequentially. For example, when you are adding nodes to a cluster, it is recommended that you add the nodes in a sequence, i.e., one after the other rather than adding them in parallel.

The following table lists all the subcommands that you can use with the `csadm ha` command:

Subcommand	Brief Description

list-nodes	Lists all the nodes that are available in the cluster with their respective node names and ID, status, role, and a comment that contains information about which nodes have joined the specific HA cluster and the primary server.
	`[root@ha-primary-node csadmin]# csadm ha list-nodes`
	`nodeId nodeName status role comment`
	`* c5b25d6f919bb70913053bbdcce486c0 ha-primary-node active primary primary server`
	`322b2e41e2e1643ac16b485f99e3ea80 ha-secondary-node-1 active secondary Joined cluster with ha-primary-`
	`4ab8e251f962dd8c324a766aa8f44fcd ha-secondary-node-2 passive secondary Joined cluster with ha-primary-`
	You can filter nodes for specific status, role, etc.
	For example, if you want to retrieve only those nodes that are active use the following command: `csadm ha list-nodes --active`, or if you want to retrieve secondary active nodes, then use the following command: `csadm ha list-nodes --active --secondary`.
	Note: The `list-nodes` subcommand will not display a node that is in the process of joining the cluster, i.e., it will display the newly added node only after it has been added to the HA cluster.

export-conf	Exports the configuration of details of the active primary node to a configuration file named `ha.conf` located by default in the `/home/csadmin` directory, i.e., `/home/csadmin/ha.conf`.
allowlist	Adds the hostnames of the secondary nodes in the HA cluster to the allowlist on the active primary node. **Important**: Ensure that incoming TCP traffic from the IP address(es) [xxx.xxx.xx.xxx] of your FortiSOAR instance(s) on port(s) **22**, **443, 5432**, **9200**, and **5671** is not blocked by your organization's firewall. You can use the following argument with this sub-command: • `--nodes NODES`: Specify a comma-separated list of hostnames to be added to the allowlist on the active primary node.
join-cluster	Adds a node to the cluster with the role and status you have specified. For more details on `join-cluster`, see the Process for configuring HA section. You can use the following arguments with this sub-command: • `--status {active, passive}`: Specify the status of the node that you want to add

	to the HA cluster as either active or passive. • `--role {primary, secondary}`: Specify the role of the node that you want to add to the HA cluster as either primary or secondary. Requires to be specified when the status of the node is set to 'active.' • `--node-name [NODE_NAME]`: Specify the hostname of the node that you want to add to the HA cluster. By default this is set to the current hostname. • `--conf CONFIG_FILE_PATH`: Specify the name and path of the configuration file copied from the primary server • `--primary-node [PRIMARY_NODE_HOSTNAME]`: Specify the DNS name of the primary node in the HA cluster. • `--primary-node-ssh-key [PRIMARY_NODE_SSH_KEY_PATH]`: Specify the name and path of the SSH private key of the primary node in the HA cluster. • `--fetch-fresh-backup`: Takes a fresh backup from the primary node of the HA cluster. • `--skip-local-backup`: Skips taking the backup of the local database. Useful in cases where the disk space is low on the current node.
forget-node	Removes the entry of a non-reachable node from the HA cluster. **Important**: You must run this command from the primary node of the HA cluster, and this command cannot be run for 'self'. You can use the following arguments with this sub-command: • `--nodeId NODEID`: Specify the ID of the node you want to remove from the HA cluster. Use the `csadm ha list-nodes` command to get the Node IDs. • `--no-interaction`: Use this argument if you do not want to be asked for any confirmations during this operation.
get-replication-stat	Displays the replication statistics, i.e., the replication lag and status between cluster nodes. **Important**: The `get-replication-stat` sub-command is applicable only on the primary node. This sub-command displays information about sending lag, receiving lag, relaying lag, and total lag. **Note**: If you have configured FortiSOAR with an externalized PostgreSQL database, then replication statistics will not be displayed for the cluster nodes.
show-health	Displays the health information for the current node. You can use the following arguments with this sub-command: `--all nodes`: Displays the health information for all the nodes in an HA cluster. This information is also available for a single node, and can be used to setup monitoring and sending health statistics of a FortiSOAR instance to external monitoring applications. `--json`: Displays the health information in the JSON format.
firedrill	Tests your disaster recovery configuration. You can perform a firedrill on a secondary (active or passive) node only. Running the firedrill suspends the replication to the node's database and sets it up as a standalone node pointing to its local database. Since the firedrill is primarily performed to ensure that the database replication is set up correctly, hence it is not applicable when the database is externalized. Once you have completed the firedrill, ensure that you perform restore, to get the nodes

	back to replication mode. **Note**: If you intend to run the firedrill for extended periods of time, such as 5 or 6 hours, it is recommended that you delete replication slots. Deleting replication slots prevents disk space from getting full before the HA cluster is restored. Use the following command to delete the replication slots: `csadm ha utils replication-slots remove --slot-name <nameOfSlot>` **Licenses on a firedrilled node**: - If the node license had a user license entitlement matching the primary node user entitlement, all users can login to the firedrilled node. - If the node license had a basic user entitlement and the HA cluster had more active users, then only the `csadmin` user can login to the UI of the firedrilled node. The `csadmin` user can then activate two users who need to test the firedrill and make the rest of the users inactive. **Note**: This does not cause any impact to the primary node or other nodes in the HA cluster. Post-restore, the firedrilled node will join the cluster back and maximum active users as per the entitlement will be honored. **Schedules on a firedrilled node:** The node on which a firedrill is being performed will have their schedules and playbooks stopped, i.e., `celerybeatd` will be disabled on this node. This is done intentionally as any configured schedules or playbooks should not run when the node is in the firedrill mode.
restore	Restores the node back to its original state in the cluster after you have performed a firedrill. That is, `csadm ha restore` restores the node that was converted to the active primary node after the firedrill back to its original state of a secondary node. The restore command discards all activities such as record creation, that is done during the firedrill since that data is assumed to be test data. This command will restore the database from the content backed up prior to firedrill.
takeover	Performs a takeover when your active primary node is down. Therefore, you must run the `csadm ha takeover` command on the secondary node that you want to configure as your active primary node. You can use the following argument with this sub-command: • `--no-interaction`: Use this argument if you do not want to be asked for any confirmations during this operation.
leave-cluster	Removes a node from the cluster and the node goes back to the state it was in before joining the cluster. You can use the following argument with this sub-command: • `--no-interaction`: Use this argument if you do not want to be asked for any confirmations during this operation.
clone-db	Clones the database from the HA cluster's primary server. This is required when the database of a secondary node goes out of sync with the database of the primary node. The following arguments are used with this sub-command: • `--conf CONFIG_FILE_PATH`: Use this argument to specify the name and path of the configuration file copied from the primary server. This is required if you have configured FortiSOAR HA clusters on separate Docker host instances. In this case, SSH will not work as password changed is lost after the container is recreated, and clone-db currently only support SSH input. Therefore, this argument allowing you to add the file path of the config file and use the same to clone the database.

	• `--primary node PRIMARY_NODE_HOSTNAME`: Use this argument to specify the DNS hostname of the primary server in the HA cluster whose database you want to clone. • `--primary node-ssh-key PRIMARY_NODE_SSH_KEY_PATH`: Use this argument to specify the SSH private key of the primary server in the HA cluster whose database you want to clone. • `--prepare-for-immediate-join-cluster`: Use this argument to create temporary PostgreSQL physical replication slots to preserve WAL files for differential syncing. Delaying the join-cluster process after using this argument could cause increased disk space on the primary and cloned nodes, as it creates temporary replication slots on both the primary and cloned nodes.
suspend-cluster	Temporarily suspends the cluster for upgrading FortiSOAR. **Note**: You can run `suspend-cluster` only on a secondary (active/passive) node. The `suspend-cluster` sub-command does not remove the associated replication slot on the primary node, which was removed by `leave-cluster`. Since the associated replication slot is not removed from the primary node, the primary server does not remove the WALs that are required for the suspended server when that node joins back the cluster. Since all the required WALs are retained, when the `resume cluster` sub-command is run it performs a differential sync to join the node back to the HA cluster. You can use the following argument with this sub-command: • `--no-interaction`: Use this argument if you do not want to be asked for any confirmations during this operation.
resume cluster	Resumes the cluster once the upgrade is successfully completed on the secondary node. The resume cluster sub-command automatically joins the secondary node to the node that was the primary when `suspend cluster` was run on this node. **Note**: You can run resume cluster only on a suspended node. You should run the `resume-cluster` sub-command on all the secondary nodes to form the ha cluster. You can run the `resume-cluster` sub-command in **parallel** on all the secondary nodes that need to join back to the cluster.
utils	Utilities that manage the replication slots. You can use the following options with this sub-command: • `replication-slots`: Manages the physical replication slots. • `replication-slots list`: Displays a list of the physical replication slots. • `replication-slots remove --slot-name <>`: Removes a physical replication slot based on the name you have specified. • `checkpoint`: Runs PostgreSQL `CHECKPOINT` to clean up the WAL files, and which should be run after a replication slot is removed. This helps users quickly check if the WAL file storage is getting reduced after deleting a replication slot. When a replication slot is removed, PostgreSQL cleans up the unnecessary WAL files on `CHECKPOINT`. By default, PostgreSQL runs `CHECKPOINT` every 5 minutes. However, you can run this option if you do not want to wait the default 5 minutes.

Overview of nodes in a FortiSOAR HA cluster

- A FortiSOAR HA cluster can have only one active primary node, all the other nodes are either active secondary nodes or passive nodes. Uniqueness of the primary node is due to the following:
  - In case of an internal database, all active nodes talk to the database of the primary node for all reads/writes. The database of all other nodes is in the read-only mode and setup for replication from the primary node.
  - Although the queued workflows are distributed amongst all active nodes, the Workflow scheduler runs only on the primary node.
  - All active nodes index the data for quick search into ElasticSearch at the primary node.
  - All integrations or connectors that have a listener configured for notifications, such as IMAP, Exchange, Syslog, etc run the listeners only on the primary node.
    Therefore, if the primary node goes down, one of the other nodes in the cluster must be promoted as the new primary node and the other nodes should rejoin the cluster connecting to the new primary.
- Active secondary nodes connect to the database of the active primary node and serve FortiSOAR requests. However, passive nodes are used only for disaster recovery and they do not serve any FortiSOAR requests.

Checking replication between nodes in an active-passive configuration

When using an active-passive configuration with internal databases, ensure that replication between the nodes is working correctly using the following steps:

- Perform the `firedrill` operation at regular intervals to ensure that the passive node can takeover successfully, when required.
- Schedule full nightly backups at the active primary node using the FortiSOAR backup and restore scripts. For more information on backup and restore, see the Backing up and Restoring FortiSOAR chapter.

Installation of connectors on nodes in a HA cluster

Once you install connectors on a FortiSOAR node that is part of a HA cluster, the installation process automatically installs these connectors seamlessly on the other nodes of the HA cluster.

Once you have installed a connector dependency on a FortiSOAR node using the **Install** link on the connector's Configurations dialog, then that dependency is installed seamlessly on the other nodes of the HA cluster.

When you install a custom connector or older versions of connectors that did not have their rpm available on the FortiSOAR server, or connectors that were created and published using the "Create New Connector" wizard, on a FortiSOAR node that is part of a HA cluster, the installation process automatically installs these connectors

seamlessly on the other nodes of the HA cluster, as is the case with a FortiSOAR-published connector, i.e., connectors that have rpms.

Changing the hostname of primary and secondary nodes in an HA cluster

Perform the following steps if you want to change the hostname of the primary and secondary nodes in an existing HA cluster.

After the hostname has been reset, when users execute playbooks with an external manual input link, it is observed that the link that is generated in the email contains the original FQDN (hostname) rather than the one that has been updated. Therefore, users who are required to provide the input, have to manually update the FQDN (hostname) in the manual input link present in the email.

Changing the hostname of the primary node

1. Ensure the new hostname of the primary node is DNS resolvable from all the cluster nodes.
   If you are using `/etc/hosts`, ensure you update `/etc/hosts` correctly on all the cluster nodes with the new hostname of the primary node.
2. SSH to the secondary/passive node and stop all services using the following command:
   `csadm services --stop`
3. SSH to the primary node and change the hostname using the following command:
   `csadm hostname --set <new-hostname>`
4. SSH to the secondary/passive node and perform the following steps:
   a. Update the new hostname of the primary node in the following files:
      - `/opt/cyops/configs/database/db_config.yml`
      - If your database is not externalized, then update the '`primary_conninfo`' attribute in the `/var/lib/pgsql` file with the new hostname of the primary node.
        Important: You must update the new hostname of the primary node in the above-mentioned files on all the secondary/passive nodes in the HA cluster.
   b. Start all the services using the following command:
      `csadm services --start`
   c. Clear the crudhub cache using the following command:
      `sudo -u nginx php /opt/cyops-api/bin/console cache:clear --no-interaction`
   d. Restart all the services again using the following command:
      `csadm services --restart`

Changing the hostname of a secondary/passive node

1. Ensure the new hostname of a secondary/passive node is DNS resolvable from all the cluster nodes. If you are using `/etc/hosts`, ensure you update `/etc/hosts` correctly on all the cluster nodes with the new hostname of the secondary/passive node.
2. SSH to the primary node and add the new hostname of the secondary/passive node to the 'allowlist' in the primary node. Use the following command to add the new hostnames to the 'allowlist':
   csadm ha allowlist --nodes <new-hostname-of-secondary/passive>
3. SSH to the secondary/passive node and change the hostname using the following command:
   `csadm hostname --set <new-hostname>`

You must follow the above steps on each secondary/passive node for which you are changing the hostname.

Upgrading an HA cluster

For the procedure on how to upgrade a FortiSOAR High Availability Cluster see the *Upgrading a FortiSOAR High Availability Cluster* chapter in the "Upgrade Guide." Starting with release 7.6.1, FortiSOAR has optimized the process of upgrading HA clusters to significantly reducing downtime to just one or two minutes. This aims at enhancing user experience and meeting standards for uninterrupted operation. Prior to release 7.6.1, upgrading an HA cluster required approximately 30 minutes of downtime.

Health Checks in FortiSOAR HA nodes

FortiSOAR HA nodes are categorized as either Active (A) or Passive (P). Active nodes serve incoming requests, while passive nodes are used for replication.

The following endpoints can be used in the load balancer's health check section to redirect requests based on the node's status:

Endpoint	Expected Status
`https://{host}/auth/node?param=active`	The endpoint returns a 400 status if the node is passive (OR) is suspended (OR) in firedrill. Returns a 200 staus for active nodes.
`https://{host}/auth/node?param=primary`	The endpoint returns a 400 status if the node is secondary node (OR) is suspended (OR) in firedrill. Returns a 200 status for the primary node.

Incorporating these endpoints into the load balancer's health check eliminates the need for manual server additions or removals during HA operations, such as 'Leave Cluster'. 'Leave Cluster' requires manual intervention in the load balancer configuration because these endpoints return a 200 status when the node leaves the HA cluster and begins functioning independently..

For example, consider an A-A-A-P deployment. Without these endpoints, if takeover is performed and a passive node takes over the role of an active node, manual changes to the load balancer are required. Using these endpoints allows the load balancer to automatically detect whether a node is active or passive, eliminating the need for manual updates. The `https://{host}/auth/node?param=primary` endpoint is needed when an embedded SME is used in a FortiSOAR HA cluster. Since the SME operates only on the primary node, all the requests to the SME must be redirected to the primary node of the cluster.

Load Balancer

The clustered instances should be fronted by a TCP Load Balancer such as HAProxy, FortiADC, or Gobetween, and clients should connect to the cluster using the address of the proxy.

Once you have configured the Load Balancer of your choice to front the FortiSOAR HA cluster, the audit logs display the IP address of the Load Balancer. If you want the audit logs to display the IP address of nodes in the HA cluster, then you must enable the load balancer's `preserve_client_ip` setting.

Setting up HAProxy as a TCP load balancer fronting the two clustered nodes

The following steps list out the steps to install "HAProxy" as a load balancer on a Virtual Machine:

1. `# yum install haproxy`
2. Edit the `/etc/haproxy/haproxy.cfg` file to configure your HAProxy as follows:

```
defaults common_defaults
  log global
  timeout connect 5s
  timeout client 60s
  timeout server 60s
  option forwardfor
  option httpchk

defaults https_defaults from common_defaults
  mode http
  http-check connect port 443 ssl
  http-check send meth GET uri /auth/node?param=active
  http-check expect status 200

defaults tcp_defaults from common_defaults
  mode tcp
  http-check connect port 443 ssl
  http-check send meth GET uri /auth/node?param=primary
  http-check expect status 200
```

```
# ---------------------------------------------------------------------
# Front end for 443 i.e., nginx requests
# ---------------------------------------------------------------------
frontend main from common_defaults
  bind *:443 ssl crt <path-to-certificate>
  use_backend ha_cluster_443

# ---------------------------------------------------------------------
# Optional front end for 5671 i.e., RabbitMQ requests (for embedded SME)
# ---------------------------------------------------------------------
frontend mq from common_defaults
  mode tcp
  bind *:5671
  tcp-request inspect-delay 5s
  tcp-request content accept if { req_ssl_hello_type 1 }
  use_backend ha_cluster_5671

# ---------------------------------------------------------------------
# Backend for front end main
# ---------------------------------------------------------------------
backend ha_cluster_443 from https_defaults
  server <server-1-hostname> <server-1-ip-address>:443 check inter 30s ssl verify none
  server <server-2-hostname> <server-2-ip-address>:443 check inter 30s ssl verify none

# ---------------------------------------------------------------------
# Backend for front end mq
# ---------------------------------------------------------------------
backend ha_cluster_5671 from tcp_defaults
  server <server-1-hostname> <server-1-ip-address>:5671 check inter 30s verify none
  server <server-2-hostname> <server-2-ip-address>:5671 check inter 30s verify none
```

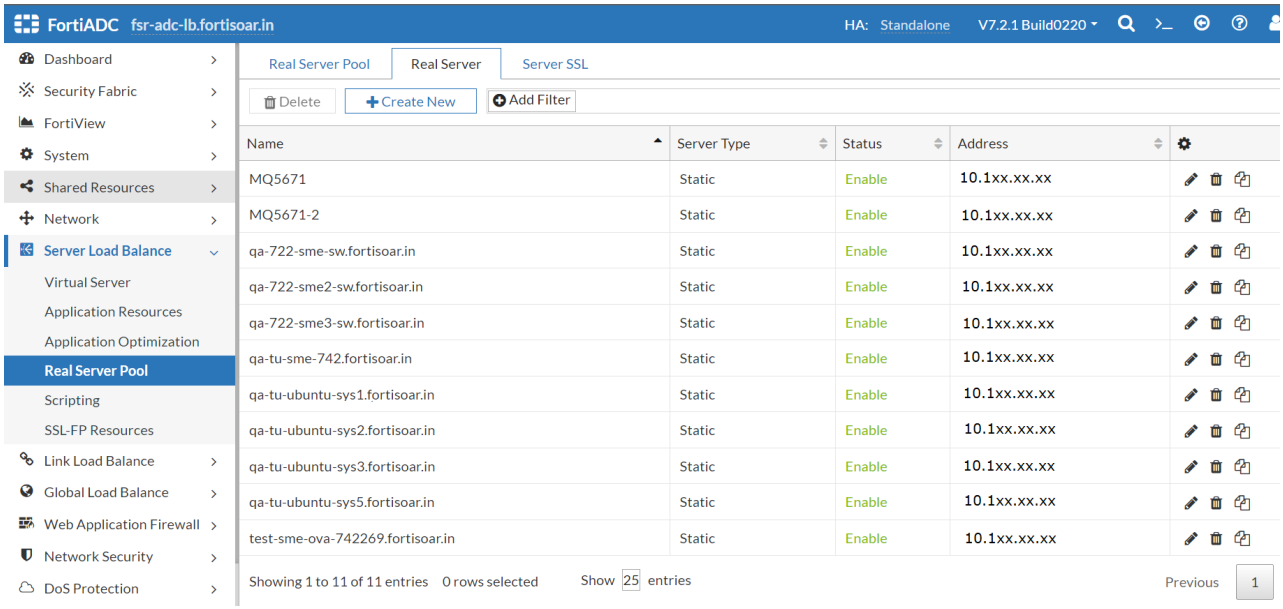
3. To reload the firewall, run the following commands:
   `$ sudo firewall-cmd --zone=public --add-port=<portspecifiedwhilebindingHAProxy>/tcp --permanent`
   `$ sudo firewall-cmd --reload`
4. Restart haproxy using the following command:
   `# systemctl restart haproxy`
5. Use the bind address (instead of the IP address of the node in the cluster) for accessing the FortiSOAR UI.

Configuring FortiSOAR in FortiADC

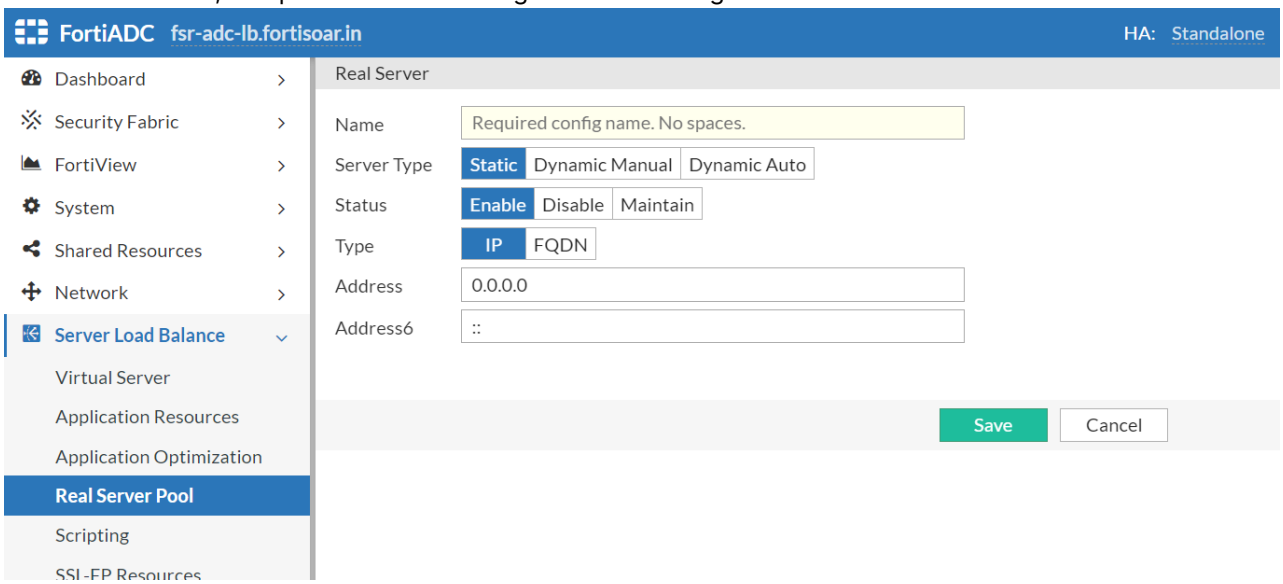
FortiADC is an advanced application delivery controller (ADC) that routes traffic to available destination servers based on health checks and load-balancing algorithms. It also improves application performance by assuming some of the server task load. Server tasks that can be handled by the FortiADC appliance include SSL encryption/decryption, WAF protection, Gzip compression, and routing processes, such as NAT.

Configuring FortiSOAR Active/Active HA cluster with FortiADC

1. Log in to FortiADC and navigate to **Server Load Balance** > **Real Server Pool**, and then click the **Real Server** tab.

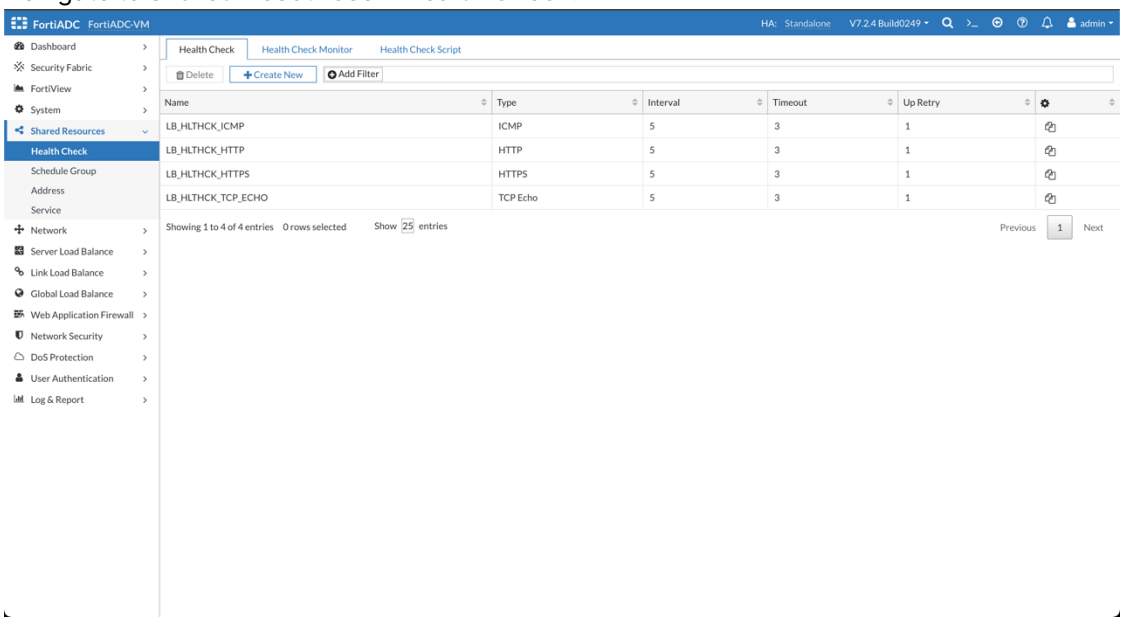


2. Click **Create New**, and provide the following details to configure a new real server:

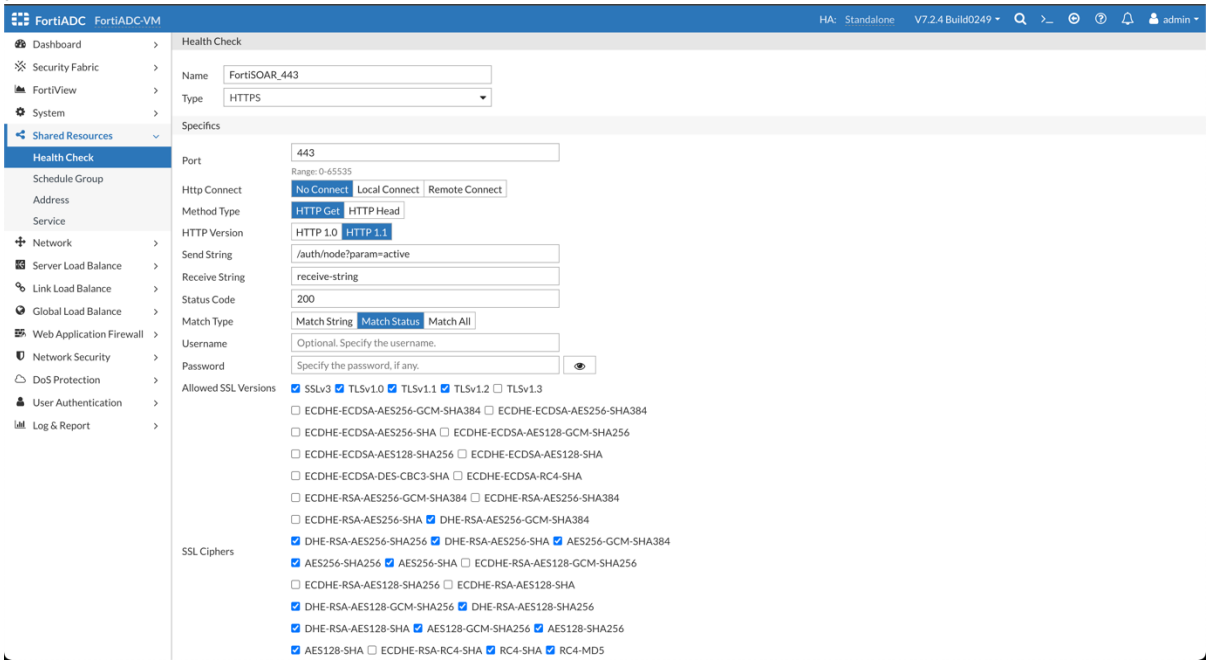


   a. In the **Name** field, specify a name of the new real server.
   b. Select the appropriate values for the **Server Type**, **Status**, and **Type** fields.
   In this example we have kept Server Type as Static, Status as Enable and Type as IP.
   c. In the **Address** field, enter the IP address of your system.
   d. Click **Save** to create a new real server.

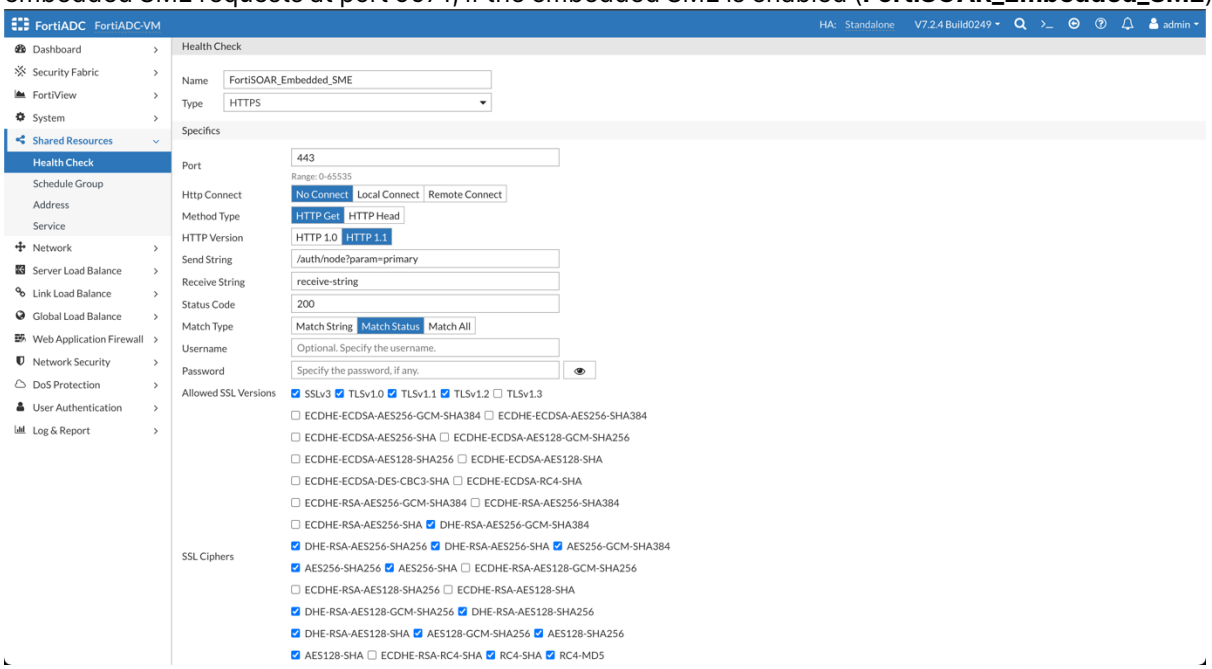
3. Navigate to **Shared Resources** > **Health Check**:



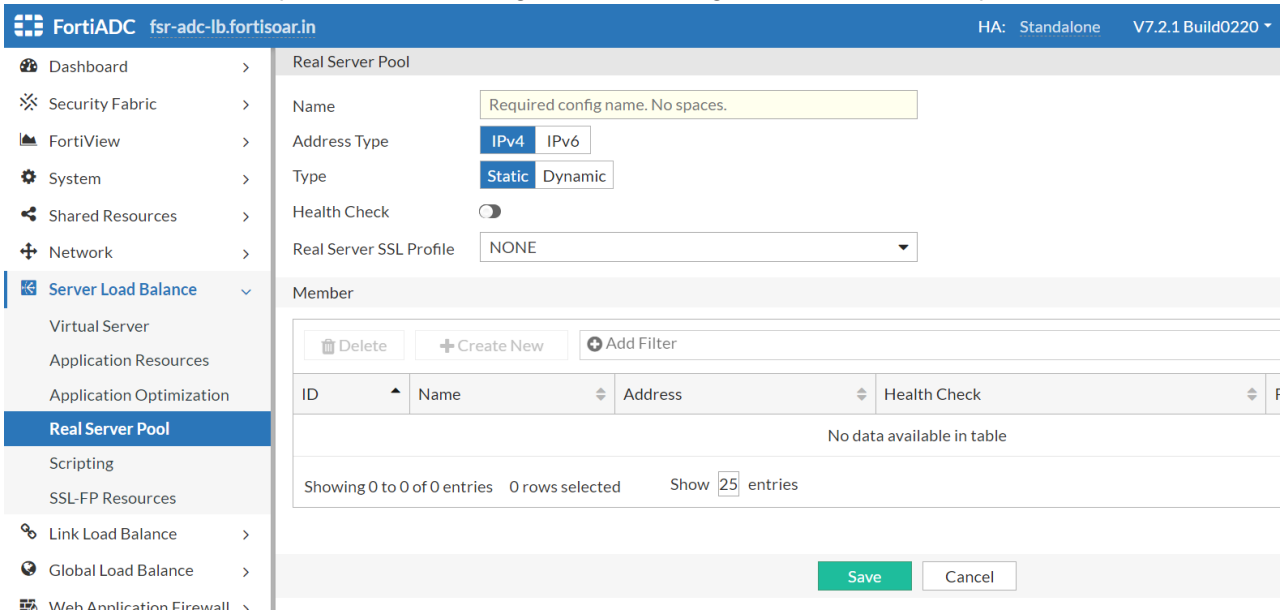
4. Click **Create New**, and provide the following details to configure health check for FortiSOAR requests at port 443 (**FortiSOAR_443**):



5. (Recommended) Click **Create New**, and provide the following details to configure health check for embedded SME requests at port 5671, if the embedded SME is enabled (**FortiSOAR_Embedded_SME**):



6. Click the **Real Server Pool** tab.
7. Click **Create New**, and provide the following details to configure a new real server pool:

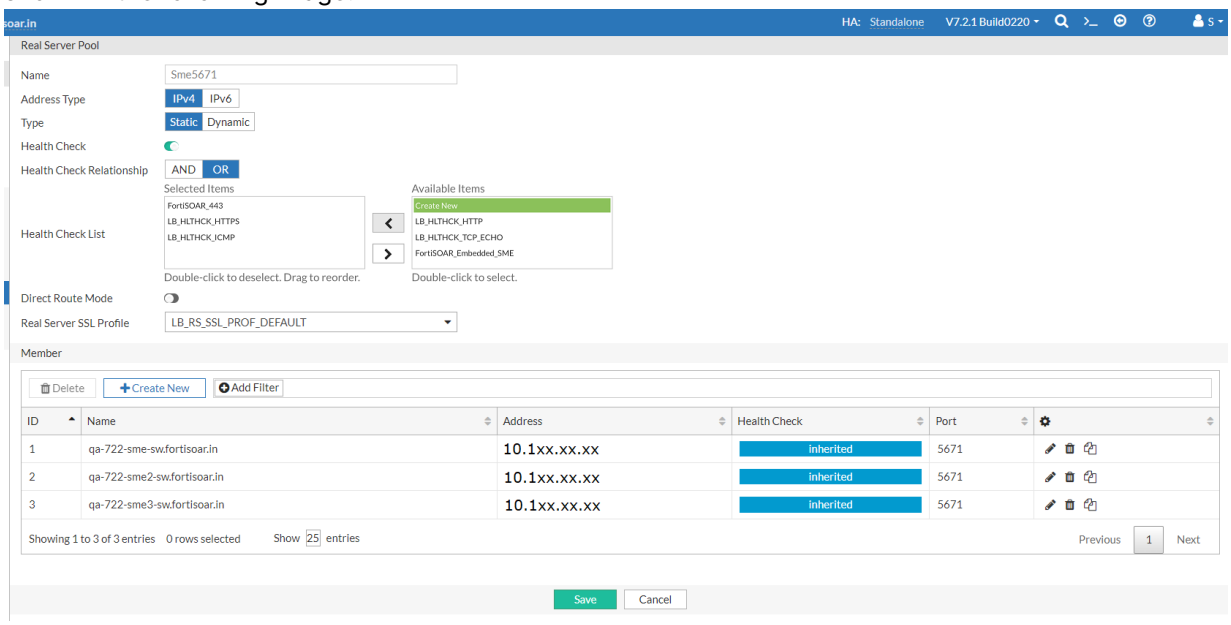


   a. In the **Name** field, specify a name of the new real server pool.
   b. Select the appropriate values for the **Address Type** and **Type** fields.
      In this example we have kept Address Type as IPv4 and Type as Static.
   c. From the **Real Server SSL Profile** drop-down, select the `LB_RS_SSL_PROF_DEFAULT` profile and click **Save**.

d. Enable **Health Check** and then do the following to configure health check:
   i. Toggle the **Health Check Relationship** switch to **OR**.
   ii. From the **Health Check List**, select and add the **LB_HLTHCK_ICMP**, **LB_HLTHCK_HTTPS** , and **FortiSOAR_443** profiles.
   iii. Click **Save**.

e. For Secure Message Exchange (SME), you must add two separate version pools, one for API and another for the TCP port. From the **Health Check List**, add **FortiSOAR_Embedded_SME**.

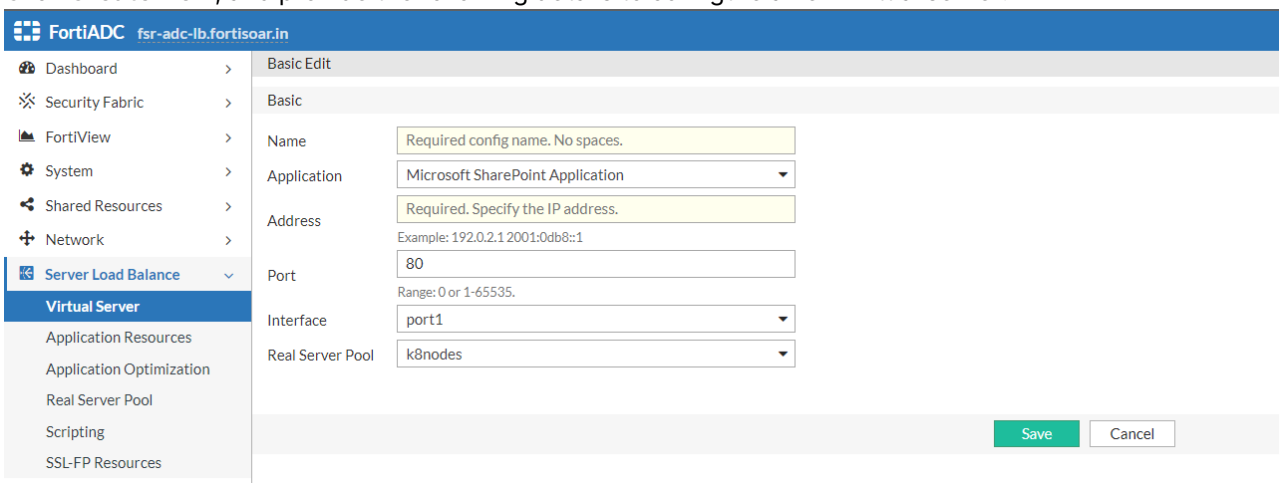
8. To add real server(s) to the newly created server pool, do the following:
   a. Navigate to **Server Load Balance** > **Real Server Pool**, and then click the **Real Server Pool** tab.
   b. Edit the server pool that you have created.
   c. Scroll down to the `Member` section and click **Create New** to add FortiSOAR servers in the server pool you have created.
   d. Select the appropriate real server pool and specify the port number for communication with FortiSOAR servers.
   e. Retain the remaining parameters as per their default values, and click **Save** to add FortiSOAR to the selected server pool.
   Once you have added all the servers to the selected server pool, the real server pool page appears as shown in the following image:



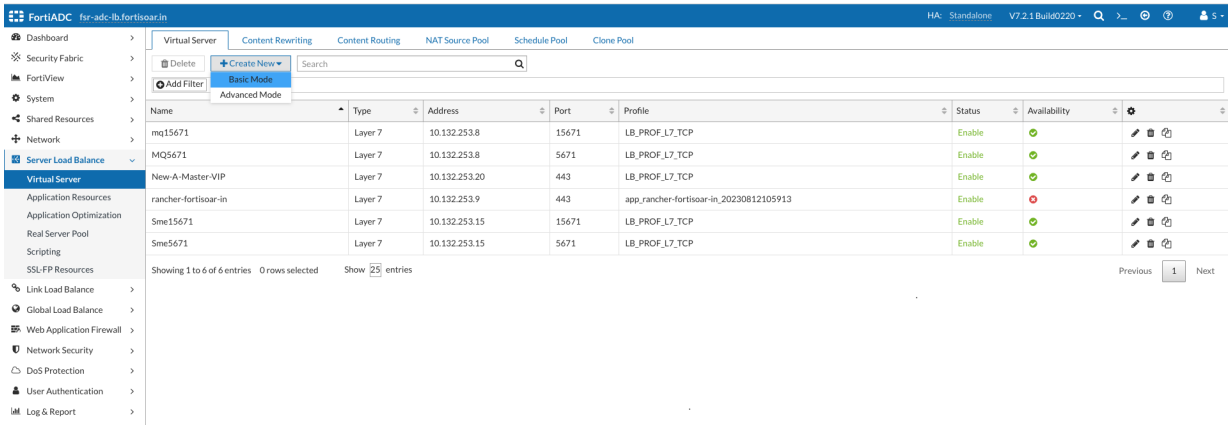
   f. For SME, you must add all real servers with API ports in the real server pool created for API ports. Perform the same steps to add real servers with TCP ports in the real server pool created for TCP ports.

9. Navigate to **Server Load Balance** > **Virtual Server**, and then click the **Virtual Server** tab.

10. Click **Create New**, and provide the following details to configure a new virtual server:



a. On the **Virtual Server** page, select **Basic Mode**:



b. In the **Name** field, specify a name of the new virtual server.

c. In the **Address** field, specify any IP address that can be used as a virtual IP address.

d. In the **Port** field, specify any valid TCP port to be used for communication. It is recommended that to you add the same port as specified for the real server pool.

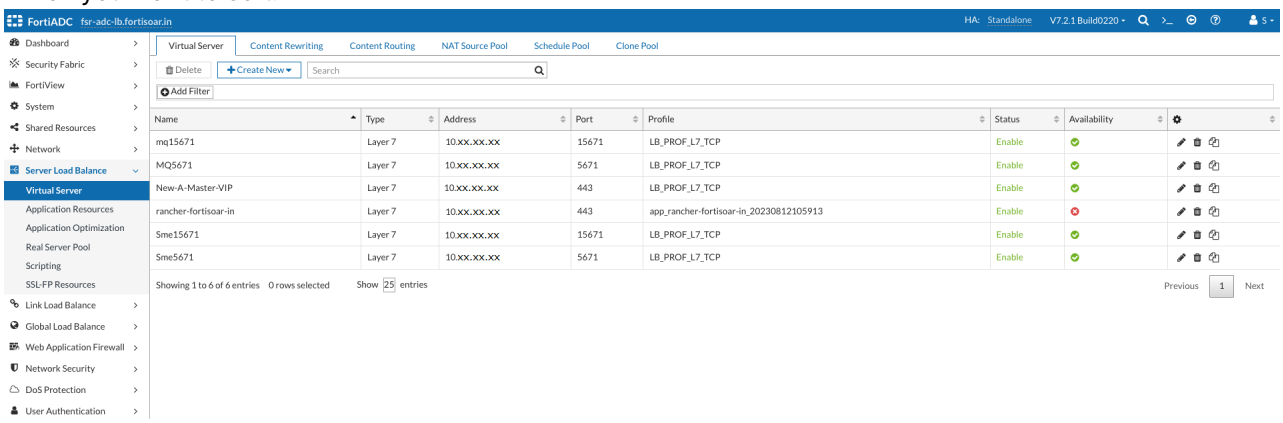
e. Set the **Application** field as '**HTTPS**' and select the appropriate value from the **Interface** drop-down list.

f. From the **Real Server Pool** drop-down list, select a real server pool that you have created.

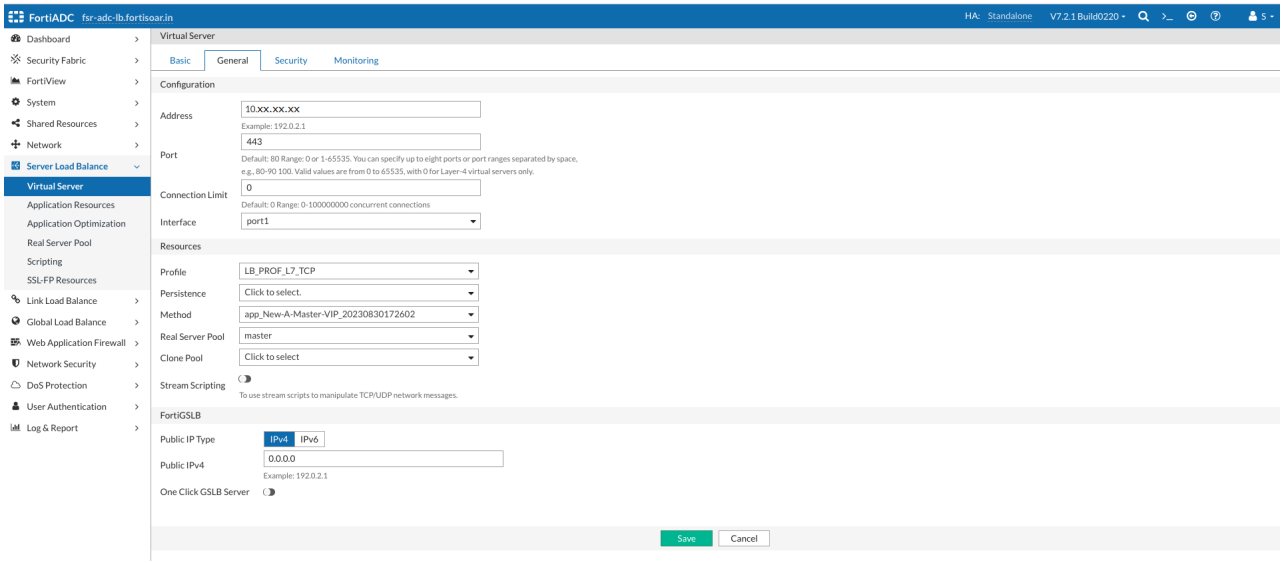
g. Click **Save**to save the configurations for your new virtual server.

h. For SME, you must add two virtual servers one for API ports and another one for TCP ports by selecting the corresponding real server pools created for API and TCP ports.

11. On the **Virtual Server** page, click the **Edit** icon in the row of the virtual server that you have added and which you want to edit:



12. On the virtual server page for that server, click the **General** tab:



On the **General** tab, edit the virtual server which has added as follows:

a. From the **Profile** drop-down, in the `Resource` section, select the `LB_PROF_L7_TCP` profile.
b. From the **Method** drop-down, select the `LB_METHOD_ROUND_ROBIN` method.
c. From the **Server Pool** drop-down, select **Persistence**.
d. Click **Save** to save the configuration details

Using the Gobetween load balancer

Gobetween is a minimalistic yet powerful high-performance L4 TCP, TLS, and UDP based load balancer.

It works on multiple platforms like Windows, Linux, Docker, Darwin, etc., and you can build your own load balancer using from source code. Balancing is done based on the following algorithms that you can choose in the configuration:

- IP hash
- World famous - Round Robin

- Least bandwidth
- Least connection
- Weight

Configuring Gobetween for FortiSOAR Active/Active HA Cluster

Installation:

Gobetween can be installed either on the Linux platform or on Windows. For details on installing gobetween, see 'Installation' section of the gobetween documentation.

Configuration:

Edit the `gobetween.toml` configuration file and then restart the `gobetween` service for the changes to take effect. A sample configuration follows:

The configuration has three sections,

- The first one describes the protocol to be used and defines the port to which the load balancer will be bound:
```
[servers.fsr]
protocol = "tcp"
bind = "0.0.0.0:3000"
```
- The second describes how the FortiSOAR nodes are discovered:
```
[servers.fsr.discovery]
kind = "static"
static_list = [
     "<YourIPAddressForHANode1>:443 weight=25 priority=1",
     "<YourIPAddressForHANode2>:443 weight=25 priority=1",
     "<YourIPAddressForHANode3>:443 weight=25 priority=1",
     "<YourIPAddressForHANode4>:443 weight=25 priority=1"
]
```
In the node discovery section, you need to add FortiSOAR nodes and provide their weight and priority to determine how requests to the load balancer will be addressed.
- The last one checks the 'health' status of each node:
```
[servers.fsr.healthcheck]
fails = 1
passes = 1
interval = "2s"
timeout="1s"
kind = "ping"
ping_timeout_duration = "500ms"
```

After you change the configuration file, you need to open the port and load the firewall:

1. Reload the firewall using the following commands:
```
$ sudo firewall-cmd --zone=public --add-port=<portspecifiedwhilebindingHAProxy>/tcp --permanent
$ sudo firewall-cmd --reload
```
2. Run the 'gobetween' service in `tmux` using the following command:
```
# sudo ./gobetween -c config/gobetween.toml
```

For more details about configuration, see the gobetween documentation.

Configuring Gobetween for a MQ Cluster

Initial procedure for setting up a RabbitMQ cluster, such as setting up the hosts file, installing the RabbitMQ server, etc, should already have been completed. Once the initial setup is completed, do the following:

1. Set up the RabbitMQ cluster: To setup the RabbitMQ cluster, ensure that the `.erlang.cookie` file is the same on all nodes. To achieve this, copy the '`.erlang.cookie`' file from the `/var/lib/rabbitmq` directory of the primary node to the other nodes. For our example, let us assume the primary node is 'node1' and secondary nodes are 'node2' and 'node3'. To copy the '`.erlang.cookie`' file use the `scp` command from the primary node ('node1'). For example:
   ```
   scp /var/lib/rabbitmq/.erlang.cookie root@node2:/var/lib/rabbitmq/
   scp /var/lib/rabbitmq/.erlang.cookie root@node3:/var/lib/rabbitmq/
   ```
   Ensure that there are no errors on both the servers, then join the node2 and node3 to node1, using the `join-cluster` command, to create a RabbitMQ cluster. For more information, see the Process for configuring High Availability section.
2. Configure RabbitMQ Setup Queue Mirroring: You must configure the 'ha policy' cluster for queue mirroring and replication to all cluster nodes. If the node that hosts queue master fails, the oldest mirror will be promoted to the new master as long as it synchronized, depending on the 'ha-mode' and 'ha-params' policies.
   Following are some examples of the RabbitMQ ha policies:
   Setup an ha policy named 'ha-all' with all queues on the RabbitMQ cluster that will be mirrored to all nodes on the cluster:
   ```
   sudo rabbitmqctl set_policy ha-all ".*" '{"ha-mode":"all"}'
   ```
   Setup ha policy named 'ha-nodes' with all queue names that start with 'nodes' and that will be mirrored to two specific nodes 'node02' and 'node03' on the cluster:
   ```
   sudo rabbitmqctl set_policy ha-nodes "^nodes\." \
        '{"ha-mode":"nodes","ha-params":["rabbit@node02", "rabbit@node03"]}'
   ```
   You can check all the available policies using the following command:
   ```
   sudo rabbitmqctl list_policies;
   ```
   If you want to remove a specific policy, use the following command:
   ```
   sudo rabbitmqctl clear_policy <name_of_policy>
   ```
3. Ensure that the SSL certificates that you specify while configuring the secure message exchange must be the same on all the nodes and should have the secure message exchange's CN name or should be a wildcard.
   For information on adding a secure message exchange, see the *Deploying FortiSOAR* chapter in the "Deployment Guide." When you are adding or configuring the secure message exchange, in the `Add New Secure Message Exchange` dialog ensure the following:
   - In the **Server Name Indication** field, ensure that you enter the Server Name Indication (SNI) address for the Secure Message Exchange. You must specify the SNI address when the Secure Message Exchange is behind a reverse proxy or in a cluster behind a load balancer.
   - In the **TCP Port** field ensure that you enter the same TCP port that you have specified while configuring the secure message exchange. Also, ensure that the FortiSOARnode has outbound connectivity to the secure message exchange at this port.
   - In the **Certificate** field, you must copy-paste the certificate text of the Certificate Authority (CA) that has signed the secure message exchange certificate in the `pem` format. If it is a chain, then the complete chain must be provided. By default, the CA certificate for the FortiSOAR self-signed certificate is present at the following location: `/opt/cyops/configs/rabbitmq/ssl/cyopsca/cacert.pem`
   - Enter the required details in the other fields and save the secure message exchange configuration.

4. Edit the `gobetween.toml` configuration file on each of the nodes in the MQ cluster and then restart the `gobetween` service for the changes to take effect. A sample configuration follows:
   The configuration has three sections,
   - The first one describes the protocol to be used and defines the ports to which the load balancer will be bound on various nodes of the MQ cluster. Ensure that you the enter the same TCP port that you have specified while configuring the secure message exchange and added in the `Add New Secure Message Exchange` dialog.
     For example, on node 1 it could be:
     ```
     [servers.routerapi]
     protocol = "tcp"
     bind = "0.0.0.0:3000"
     ```
     For example, on node 2 it could be:
     ```
     [servers.routertcp]
     protocol = "tcp"
     bind = "0.0.0.0:3000"
     ```
   - The second describes how the MQ cluster nodes are discovered:
     For example, on node 1 it could be:
     ```
     [servers.routerapi.discovery]
     kind = "static"
     static_list = [
          "router-node1.fortisoar.in:15671 weight=25 priority=1",
          "router-node2.fortisoar.in:54549 weight=25 priority=1",
          "router-node3.fortisoar.in:54549 weight=25 priority=2"
     ]
     ```
     For example, on node 2 it could be:
     ```
     [servers.routertcp.discovery]
     kind = "static"
     static_list = [
          "router-node1.fortisoar.in:5671 weight=25 priority=1",
          "router-node2.fortisoar.in:54558 weight=25 priority=1",
          "router-node3.fortisoar.in:54559 weight=25 priority=1"
     ]
     ```
     In the node discovery section, you need to add the secure message exchange for the nodes and provide their weight and priority to determine how requests to the load balancer will be addressed.
   - The last one checks the '`health`' status of the MQ cluster:
     For example, on node 1 it could be:
     ```
     [servers.routerapi.healthcheck]
     fails = 1
     passes = 1
     interval = "2s"
     timeout="1s"
     kind = "ping"
     ping_timeout_duration = "500ms"
     ```
     For example, on node 2 it could be:
     ```
     [servers.routertcp.healthcheck]
     fails = 1
     passes = 1
     interval = "2s"
     timeout="1s"
     kind = "ping"
     ```

```
ping_timeout_duration = "500ms"
```
For details, see the GoBetween HealthChecks document.

5. After you change the configuration file, you need to open the port and load the firewall:
   a. Reload the firewall using the following commands:
   ```
   $ sudo firewall-cmd --zone=public --add-port=<portspecifiedwhilebindingHAProxy>/tcp --permanent
   $ sudo firewall-cmd --reload
   ```
   b. Run the 'gobetween' service in `tmux` using the following command:
   ```
   # sudo ./gobetween -c config/gobetween.toml
   ```
6. Test your RabbitMQ cluster by opening your web browser and typing the IP address of a node, for example, node 1, whose port is set as '5671'.
   `http://<node1IP>:5671/`
   Type in the username and password you have configured. If everything is setup correctly, you will see the RabbitMQ admin **Dashboard** with the status of all the members of the cluster, i.e., node1, node2, and node3, displaying as up and running. You can click the **Admin** tab and click the **Users** menu to view the list of active users and the **Policies** menu to view the list of created policies.

Behavior that might be observed while publishing modules when you are accessing HA clusters using a load balancer

When you have initiated a publish for any module management activity and you are accessing your HA cluster with one or more active secondary nodes using a load balancer such as "HAProxy", then you might observe the following behaviors:

- While the Publish operation is in progress, you might see many publish status messages on the UI.
- If you have added a new field to the module, or you have removed a field from the module, then you might observe that these changes are not reflected on the UI. In such cases, you must log out of FortiSOAR and log back into FortiSOAR.
- After a successful publish of the module(s), you might observe that the **Publish** button is yet enabled and the modules yet have the asterisk (*) sign. In such cases, you must log out of FortiSOAR and log back into FortiSOAR to view the correct state of the Publish operation.

Extending support for two NICs on a FortiSOAR appliance for controlled traffic routing

Multihoming is a practice of connecting a host or a computer network to more than one network, which helps in segregating the network traffic for better performance and security. This section talks about multihoming FortiSOAR with two NICs. The first NIC works as a service interface and the second one as a management interface. The service interface is considered to be the default route having outbound internet connectivity. The management interface is considered to be protected from public space attacks and is connected to the intranet subnet. Data replication is done using the management interface.

If you have already set up an HA cluster, then you require to break that cluster by running the following command on each of the secondary nodes:

```
# csadm ha leave-cluster
```

The process of multihoming is divided into two sections:

- Section 1: Rocky Linux or RHEL changes for multihoming (MultiNIC)
- Section 2: FortiSOAR changes for multihoming

Section 1: Rocky Linux or RHEL changes for multihoming (MultiNIC)

1. Add a new NIC to the VM. Depending on your hypervisor, steps to add a new NIC might differ. Follow the hypervisor-specific document for the steps to add a new NIC.
2. Configure policy-based routing on Rocky Linux or RHEL, via network scripts when the NetworkManager is running, using the following steps:
   ```
   # yum install NetworkManager-config-routing-rules
   # systemctl enable NetworkManager-dispatcher.service
   # systemctl start NetworkManager-dispatcher.service
   ```
3. Determine the name of the network interface for the newly added NIC using the following command:
   ```
   # ip -o link show | awk -F': ' '{print $2}'| grep -v 'lo'
   ```
   Note: This command displays all the network interface names excluding loopback. If the newly-adding NIC is not visible, you need to reboot the VM using the `reboot` command. After the reboot, you can re-run the command and determine the name of the new NIC. The output of the command appears as follows:
   ```
   # ip -o link show | awk -F': ' '{print $2}'| grep -v 'lo'
   ens160
   ens192
   #
   ```
   Note: The output displayed is for a FortiSOAR VM on a VMWare hypervisor, the output will vary depending on your hypervisor.
   Also, from this step onwards till the end of the procedure, 'ens160' is considered as the 'service interface' (default route having outbound internet connectivity), and 'ens192' is considered as the 'management interface' (protected from public space attacks, private NIC).
4. To set the default route for the 'service interface', use the following command:
   `nmcli connection modify <con_name> ipv4.never-default no`
   For example, `nmcli connection modify ens160 ipv4.never-default no`
5. To ensure that the default route is not set for the 'management interface', use the following command:
   `nmcli connection modify <con_name> ipv4.never-default yes`
   For example, `nmcli connection modify ens192 ipv4.never-default yes`
6. Open the `/etc/iproute2/rt_tables` file and add the following lines to the file for the routing table information of the interfaces:
   ```
   200 ens160-rt
   201 ens192-rt
   ```
7. Add the rule for the service interface, using the following command:
   ```
   ip rule add from <ip-of-NIC>/32 table <table-name-from-routing-table>
   ip rule add to <ip-of-NIC>/32 table <table-name-from-routing-table>
   ```
   For example,
   ```
   ip rule add from 10.132.255.237/32 table ens160-rt
   ip rule add to 10.132.255.237/32 table ens160-rt
   ```
8. Add the rule for the management interface, using the following command:
   ```
   ip rule add from <ip-of-NIC>/32 table <table-name-from-routing-table>
   ip rule add to <ip-of-NIC>/32 table <table-name-from-routing-table>
   ```
   For example,
   ```
   ip rule add from 10.132.255.237/32 table ens192-rt
   ip rule add to 10.132.255.237/32 table ens192-rt
   ```
9. Add the route for the service interface, using the following command:
   ```
   sudo ip route add <subnet of NIC>/24 dev <NIC name> table <table-name-from-routing-table>
   sudo ip route add default via <subnet gateway> dev <NIC name> table <table-name-from-
   ```

routing-table>
For example,
```
sudo ip route add 10.132.255.0/24 dev ens160 table ens160-rt
sudo ip route add default via 10.132.255.1 dev ens160 table ens160-rt
```

10. Add the route for the management interface, using the following command:
```
sudo ip route add <subnet of NIC>/24 dev <NIC name> table <table-name-from-routing-table>
sudo ip route add default via <subnet gateway> dev <NIC name> table <table-name-from-routing-table>
```
For example,
```
sudo ip route add 10.132.255.0/24 dev ens192 table ens192-rt
sudo ip route add default via 10.132.255.1 dev ens192 table ens192-rt
```

11. Reload and notify changes to the NetworkManager using the following commands:
```
# nmcli connection reload
# nmcli connection up '<connection_name>'
# nmcli connection up '<connection name>'
```
You can get the connection names using the `# nmcli connection show` command, and then run the commands to reload and notify changes, for example:
```
# nmcli connection reload
# nmcli connection up 'System ens160'
# nmcli connection up 'System ens192'
```

Section 2: FortiSOAR changes for Multihoming

FortiSOAR Enterprise changes for Multihoming

1. Configure PostgreSQL to listen on the management NIC:
   Open the `/var/lib/pgsql/16/data/postgresql.conf` file
   Update the following entry and save the file:
   `listen_addresses = 'localhost,192.168.10.22'`
   Note: 192.168.10.22 is the sample IP address value of the management NIC.
2. Configure ElasticSearch to listen on the management NIC:
   Open the `/etc/elasticsearch/elasticsearch.yml` file
   Update the following entry and save the file:
   `network.host: [ _ens192_ , _lo_ ]`
   Note: 'ens192' is the network interface name of the management NIC and 'lo' means loopback.
3. From this step onwards, assume the service interface DNS name to be 'fortisoar.myorgdomain' and the management interface DNS name to be 'fortisoar-management.myorgdomain'.
   Add the service and management interface DNS names in `alt_names` section in the **`/opt/cyops/certs/leaf.openssl.conf`** file.
   For example,
   The original `alt_names` section in the `leaf.openssl.conf` file:
   ```
   [alt_names]
   DNS.1 = fortisoar.myorgdomain
   DNS.2 = localhost
   IP.1 = 127.0.0.1
   ```

   After adding the service and management interface DNS names:
   ```
   [alt_names]
   DNS.1 = fortisoar-management.myorgdomain
   DNS.2 = localhost
   DNS.3 = fortisoar.myorgdomain
   IP.1 = 127.0.0.1
   ```
4. Add the service and management interface DNS names in `alt_names` section in the **`/opt/cyops-rabbitmq/configs/ssl/openssl.cnf`** file.
   For example,
   The original `alt_names` section in the `openssl.cnf` file:
   ```
   [alt_names]
   DNS.1 = fortisoar.myorgdomain
   ```

   After adding the service and management interface DNS names:
   ```
   [alt_names]
   DNS.1 = fortisoar-management.myorgdomain
   DNS.2 = fortisoar.myorgdomain
   ```
   Note: If you use signed certificates, ensure that the certificate resolves both the service and management interface names.
5. Set the hostname to the management interface DNS name using the following command:
   `# csadm hostname --set fortisoar-management.myorgdomain`

6. Set the service interface DNS name in 'workflow' and 'crudhub' using the following commands:
   a. Update the value of `Server_fqhn` in the Playbook Designer by opening any playbook, and clicking **Tools** > **Global Variables**. In the Global Variables pane, set the value of the `Server_fqhn` variable as the service interface DNS name.
   b. Update the service interface DNS name in 'crudhub' using the following commands:
   ```
   # service_interface_dns_name=”fortisoar.myorgdomain”

   # /opt/cyops/scripts/api_caller.py --endpoint "https://localhost/api/3/system_
   settings/845c05cc-05b3-450e-9afb-df6b6e436321" --method PUT --payload "
   {\"globalValues\": { \"hostname\": \"$service_interface_dns_name\"}}" >/dev/null
   ```
7. Form the HA cluster again using the management interface DNS name. Use the `csadm ha join-cluster` command to reform the HA cluster. For more information on forming an HA cluster, see the Process for configuring High Availability topic.

FortiSOAR Secure Message Exchange changes for Multihoming

1. If you are using a signed certificate for secure message exchange, and if the master and tenant connect using different interfaces, ensure that the certificate resolves both the service and management interface names through Subject Alternative Names.
   If you are using a self-signed certificate for secure message exchange, then do the following:
   a. Add the service and management interface DNS names in `alt_names` section in the `/opt/cyops-rabbitmq/configs/ssl/openssl.cnf` file.
   For example, the original `alt_names` section in the `openssl.cnf` file appears as follows:
   ```
   [alt_names]
   DNS.1 = fortisoar.myorgdomain
   ```
   The `openssl.cnf` file appears as follows after you have added the service and management interface DNS names in the `alt_names` section:
   ```
   [alt_names]
   DNS.1 = fortisoar-management.myorgdomain
   DNS.2 = fortisoar.myorgdomain
   ```
   b. Regenerate the self-signed certificates using the FortiSOAR CLI:
   `csadm mq certs generate`
   c. If you have already configured master and tenant nodes, then do the following:
      i. Update the configuration of the secure message exchange with the new certificate on the master.
      ii. Remove the master configuration from the tenant node.
      iii. Restart the service using the `systemctl restart cyops-postman` command on both the master and tenant nodes.
   d. If the tenant nodes connect using a different hostname than the master node, then you will have to update the name in the configuration file downloaded from the master node for the 'sni' and 'address' keys before applying the configuration on tenant nodes.

Setting up a High Availability FortiSOAR cluster in the AWS Cloud with Aurora as the external database

This topic describes the procedure and sample test runs for setting up a highly scalable FortiSOAR cluster with Amazon Aurora as the database backend.

This topic covers the following tests:

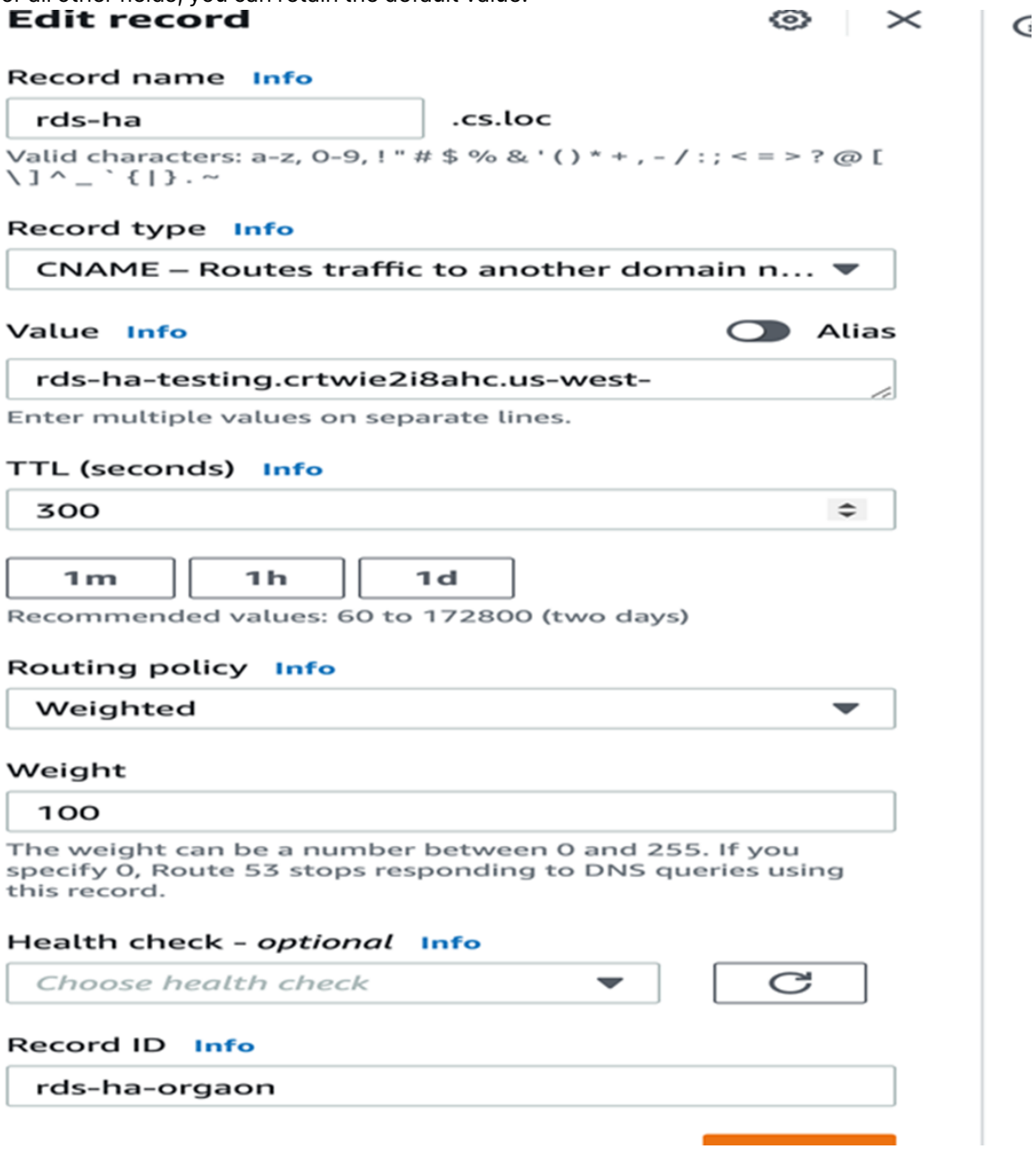
1. Verifying FortiSOAR functionality with the Aurora external database
2. Verifying FortiSOAR cluster failover to another region
3. FortiSOAR nodes Hydration
4. Upgrading Hydrated nodes in a FortiSOAR cluster

Configuration Details

The DR setup, for our example, has been configured as follows:

- Three FortiSOAR nodes in different Availability Zones located in the AWS Mumbai Region.
- Three FortiSOAR nodes in different Availability Zones located in the AWS Oregon Region.
- AWS Aurora Cluster with One Reader Instance and One Writer Instance with Global Database. For details see, https://aws.amazon.com/blogs/database/cross-region-disaster-recovery-using-amazon-aurora-global-database-for-amazon-aurora-postgresql/.
- Setup the Amazon Route53 service as the load balancer for Aurora database endpoint as follows:
  - Create a Route53 record set in the domain in which you want to configure the Route53 service. In the `Edit Record` form, enter the following details:
    - In the **Value** field, specify the global database cluster identifier name.
    - In the **Weight** field, specify 100.

- For all other fields, you can retain the default value:



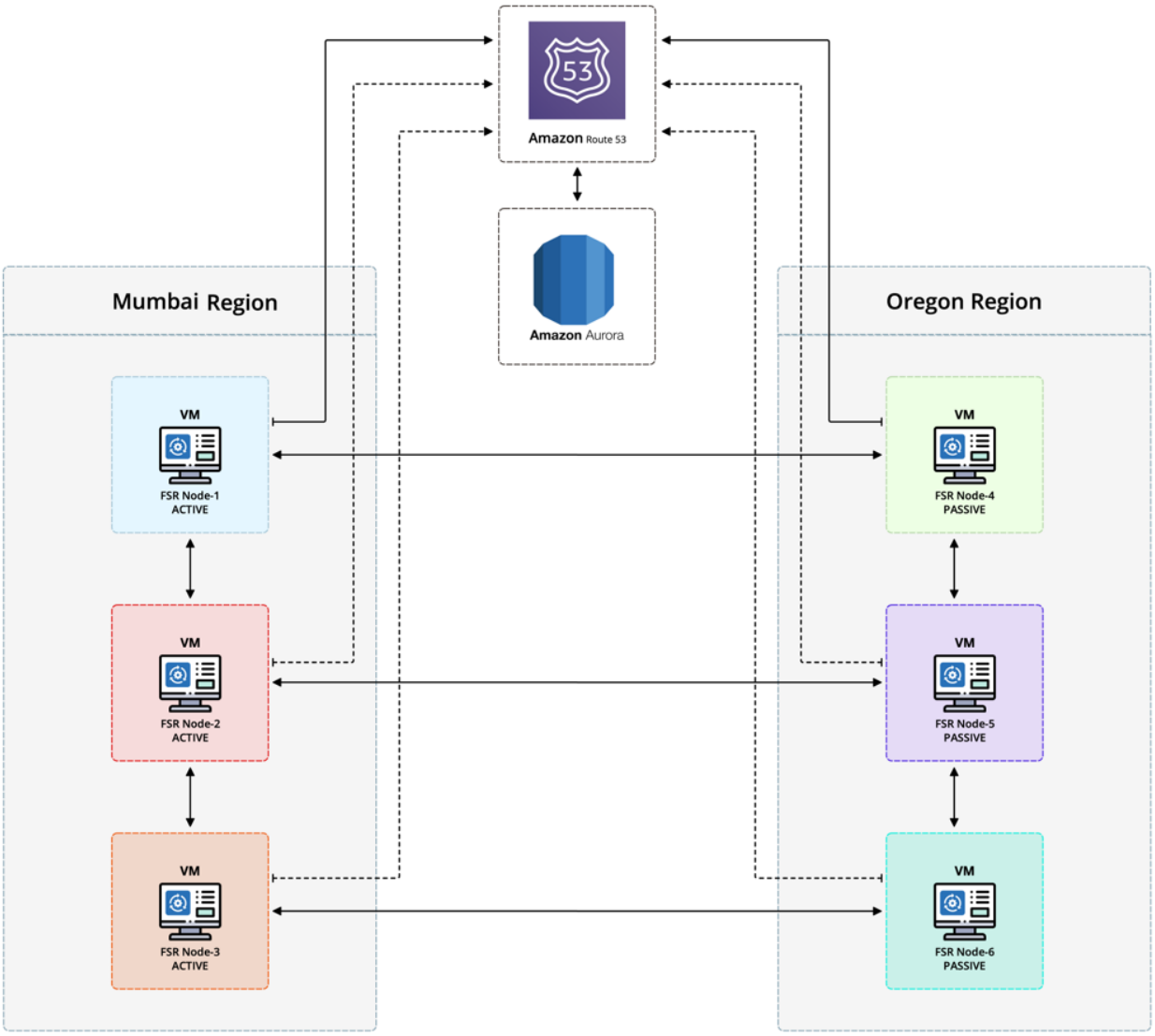
Setting up an external Aurora database on FortiSOAR

For the process of setting up an external Aurora database on FortiSOAR, see the Externalization of your FortiSOAR PostgreSQL database chapter.

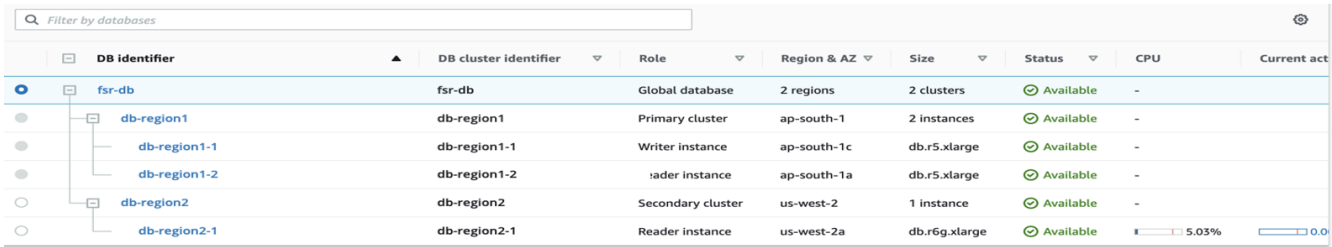
For more information on ingestion and playbook environment details, see the "Sizing Guide" on https://docs.fortinet.com/product/fortisoar/.

Configuration for the Aurora Database is 4vCPU, 16GB RAM, and for FortiSOAR it will be the standard configuration as mentioned in the "Deployment Guide".

Network Diagram



Structure of the Aurora Database



Verifying FortiSOAR functionality with the Aurora external database

1. Create a six-node FortiSOAR cluster where three nodes are in one region but in different availability zones and the other three nodes are in different regions and different availability zones. See the Network Diagram for more information.
2. Create an RDS Aurora Cluster with global database instances that span across regions. See the Structure of the Aurora Database diagram for more information.
3. Create an Amazon Route 53 setup in which you must add the database identifier of the primary database cluster.
4. Externalize the database of your FortiSOAR primary node. For more information on database externalization, see the Externalization of your FortiSOAR PostgreSQL database chapter.
5. Create a FortiSOAR HA cluster by adding the other FortiSOAR nodes to the primary node using the join-cluster command. In this HA cluster, three of the nodes will be active and the other three that are in different regions will be passive. For more information on creating an HA cluster, see the Process for configuring High Availability topic.
6. For generating data samples and verifying playbook execution as part of the test, the FortiSOAR Incident Response Solution Pack was used.
   Note: It is not mandatory for the configuration to install the Incident Response Solution Pack.
   For more information, see FortiSOAR Documentation in FUSE.
7. Create Demo Alerts with the following two schedules and properties:
   - Create one alert per minute
   - Create a burst of 150 alerts every hour
   - Each created alert has a size of 1 MB
   - Twelve playbooks run per alert
     Note: Keep this schedule running for a minimum of one week.

Outcomes

Confirmed the FortiSOAR functionality with the Aurora external database by performing operations such as:

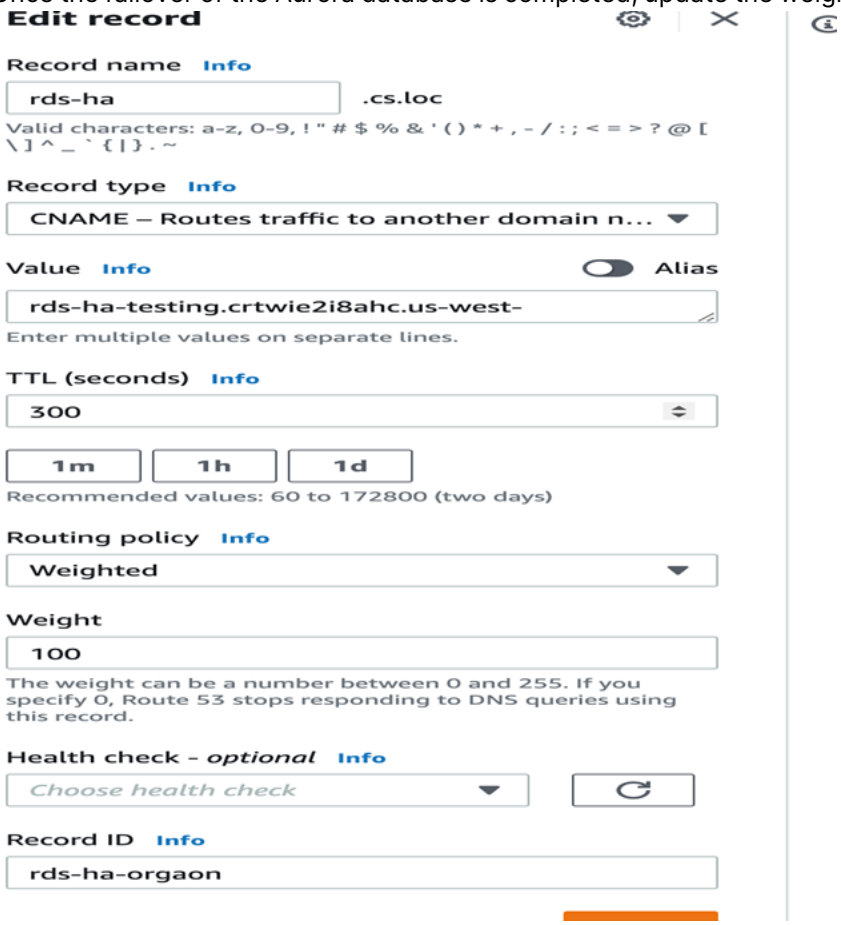
- Installation and configuration of connectors and solution packs.
- Set up of data ingestion.
- Tested the SLA workflow and widgets.
- Scheduled and triggered playbook executions.

Verifying FortiSOAR cluster failover to another region

The following sequence of steps were executed to verify the cluster failover:

1. Create or use the FortiSOAR HA cluster you have created in the Verifying FortiSOAR functionality with the Aurora external database topic.
2. Shut down the primary Site and initiate failover of the Aurora RDS Cluster to another region.

3. Once the failover of the Aurora database is completed, update the weightage in Route 53:



4. Initiate the Takeover operation on one of the passive FortiSOAR nodes and make it active. For information on the Takeover process, see the Takeover topic.
   Note: It took three minutes to complete the takeover operation.
5. After you have completed the takeover process and created the HA cluster using the `join-cluster` command, verify the data by logging onto a FortiSOAR active node.
   Note: It took six minutes to complete `join-cluster` operation for the remaining three nodes and the data size on the primary node is 52 GB.

Outcomes

Confirmed that the FortiSOAR cluster could successfully failover to another region, and it took five minutes to complete the `failover` operation. For the setup containing 52 GB of data, it was observed that the `join-cluster` operation for the remaining two nodes took 6 minutes.

FortiSOAR Nodes Hydration

Create a configuration that is similar to your FortiSOAR configuration VMs to test hydration.

1. Provision six new FortiSOAR VMs.
   These nodes serve as hydration replacements for the six original node, and hence there would be one in each of the availability zones as the original system.
2. Take the configuration backup from the original FortiSOAR nodes. For more information on backup and restore, see the Backing up and Restoring FortiSOAR chapter.
3. Restore that backup on each of the hydrated nodes. For more information on backup and restore, see the Backing up and Restoring FortiSOAR chapter.
4. On the primary hydrated node, do the following:
   a. Run the `sudo -u nginx php /opt/cyops-api/bin/console cache:clear` command.
   b. Run the `sudo -u nginx php /opt/cyops-api/bin/console app:system:update` command.
   c. Update the Device UUID in `db_config.yml` for the RabbitMQ password.
   d. Run the `csadm license –refresh-device-uuid` command.
   e. Restart all the services using `csadm services --restart` command.
5. Run the `join-cluster` command on the secondary nodes to join these nodes with the primary node and create the HA cluster.
6. Perform a sanity check on the newly-hydrated FortiSOAR VM cluster.
7. Delete the old FortiSOAR VMs.

Outcomes

The hydration process was successfully completed and a maintenance window of approximately two hours is needed to complete the hydration process.

Upgrading Hydrated FortiSOAR Nodes

The hydrated FortiSOAR Nodes were upgraded from FortiSOAR release 7.0.2-664 to 7.2.0 - Interim Build.

1. Run the leave-cluster command on each of the nodes to remove them from the FortiSOAR VM cluster.
2. Download the FortiSOAR Upgrade script on each node.
3. Upgrade each of the nodes. For more information on upgrading FortiSOAR, see the "Upgrade Guide."
4. Once all the nodes are upgraded, run the `join-cluster` command again on the nodes, to re-create the HA cluster.

Tunables

You can tune the following configurations:

- `max_wal_senders` = 10
  This attribute defines the maximum number of walsender processes. By default, this is set as `10`.
- `max_replication_slots` = 10
  This attribute defines the maximum number of replication slots. By default, this is set as `10`.

Every secondary/passive node needs one wal sender process and one replication slot on the primary node, which means that the above setting can configure a maximum of 10 secondary/passive nodes.

If you have more than 10 secondary/passive nodes, then you need to edit the value of the `max_wal_senders` and `max_replication_slots` attributes in the `/var/lib/pgsql/16/data/postgresql.conf` file on the primary node and restart the PostgreSQL service using the following command: `systemctl restart postgresql-16`
Note: You might find multiple occurrences of `max_wal_senders` and `max_replication_slots` attributes in the `postgresql.conf` file. You always need to edit last occurrence of the `max_wal_senders` attribute in the `postgresql.conf` file.

Also note that Settings changes that are done in any configuration file on an instance, such as changing the log level, etc., apply only to that instance. Therefore, if you want to apply the changed setting to all the node, you have to make those changes across all the cluster nodes.

Best practices

- Fronting and accessing the FortiSOAR HA cluster with a Load Balancer or a Reverse Proxy is recommended so that the address remains unchanged on takeover.
- You must ensure that the SIEM and other endpoints that FortiSOAR connects to are reachable on the virtualized host name (DNS) that would remain intact even after a failover (local or geo wide).
- The FortiSOAR node connects outbound to the SIEM, to periodically pull the "Alerts" (Terminology for this would differ for each SIEM, Eg, 'Offense', 'Corelated Event', 'Notable'). The "pull" model also ensures resiliency. In the case of downtime, once the FortiSOAR node comes back up, it would pull the alerts from last pulled time, ensuring there is no data loss even during down time.
- If you are planning to configure high availability in case of a multi-tenancy environment, i.e., for your master or tenant nodes, you must first configure high availability then configure MSSP. For more information on MSSP, see the "Multi-Tenancy support in FortiSOAR Guide".

Best practices for changing the shared memory size on HA nodes after setting up an HA cluster

The settings of some parameters determine the size of shared memory for tracking transaction IDs, locks, and prepared transactions. These shared memory structures should ideally be the same size on both the primary and standby nodes. This is needed to ensure that the standby nodes do not run out of shared memory during recovery. For example, if the primary had used a prepared transaction but the standby had not allocated any shared memory for tracking prepared transactions, then the recovery would pause until you change the configuration of the standby nodes. The parameters used for tracking transaction IDs, locks, and prepared transactions are as follows:

- `max_connections`
- `max_prepared_transactions`
- `max_locks_per_transaction`
- `max_wal_senders`
- `max_worker_processes`

The easiest way to ensure that the size of shared memory does not become a problem is to set the value of these parameters on the standby nodes such that they are equal to or greater than those on the primary node. Therefore, if you want to increase the values of these parameters, you should increase the parameter values first on all the standby nodes, before applying these changes to the primary node. Conversely, if you want to decrease these values; they should be decreased first on the primary node, before applying these changes to all the standby nodes. Also, note that when a standby node is promoted, it becomes the new reference for the required parameter settings for all the other standby nodes. Therefore, to avoid the size of shared memory becoming a problem during a switchover or failover, it is recommended to keep these settings the same on all the standby nodes.

Monitoring health of HA clusters

All secondary nodes in the cluster exchange HA heartbeat packets with the primary node so that the primary node can monitor and verify the status of all the secondary nodes and the secondary nodes can verify the status of the primary node.

Your system administrator can configure the monitoring of heartbeats on the **System Configuration** > **General** > **System & Cluster Health Monitoring** > **Cluster Health** section. Once you have configured monitoring of heartbeats and if any node in the HA cluster is unreachable, then the other active nodes in the cluster, which are operational, send email notifications and write log messages to alert the system administrator that a failure has occurred. For more information, see the `Configuring System and Cluster Health Monitoring` topic in the System Administration chapter.

Understanding HA Cluster Health Notifications

HA cluster health notification checks on the primary node

On every scheduled monitoring interval, which defaults to 5 minutes on the primary node for every secondary/passive node, the HA cluster health notifications checks:

- If there is a heartbeat miss from the secondary/passive node(s) in the last 15 minutes by taking the default values of monitoring interval (5 minutes) * missed heartbeat count (3). If there is a missed heartbeat, then the health notification check sends a "heartbeat failure" notification and exits.
- If the data replication from the primary node is broken. If yes, then the health notification check sends a notification containing the replication lag with respect to the last known `replay_lsn` of secondary node and exits.
  Following is a sample notification:

```
Following secondary FortiSOAR node(s) seem to have failed -
Node: hasecondary.myorgdomain
Current lag with primary node is 97 kB

Failure reason:
       1. The postgres database replication to the secondary node is not working due to data log
rotation at the primary node.
```

```
    2. The secondary node has been shutdown/halted.
    3. PostgreSQL not running on node(s).
    4. nodeName from 'csadm ha list-nodes' differs from actual FQDN used during join-cluster.

If node is up and running,
       1. Check the status of PostgreSQL service using 'systemctl status postgresql-<postgresql_
version_here> -l' on node to get more details.
       2. If you see 'FATAL: could not receive data from WAL stream: requested WAL segment has
already been removed' in the PostgreSQL service status, you need to re-join the cluster using
'csadm ha join-cluster --fetch-fresh-backup'
```

- If the replication lag reaches or crosses the set threshold specified, then the health notification check sends a notification containing the replication lag as shown in the following sample notification:

```
Replication lag threshold is reached on following node(s):
Node: hasecondary.myorgdomain
Current lag with primary node : 3431 MB
Configured Threshold : 3 GB
```

- If any service is not running, then the health notification check sends a "service failure" notification and exits.
- If a firedrill is in progress on a secondary/passive node. If yes, then the health notification check sends the following notification and exits.
  `Firedrill is in progress on following node(s):`
  `Node: hasecondary.myorgdomain`
  `Current lag with primary node : 52 kB`
  You can ignore the lag that is displayed in this case since this lag indicates the amount of data the firedrill node needs to sync when `csadm ha restore` is performed.
  You can also check the lag using the `get-replication-stat` command on the primary node.

HA cluster health notification checks on the secondary node

On every scheduled monitoring interval, which defaults to 5 minutes on the secondary node, the HA cluster health notifications checks:

- If there is a heartbeat miss from the primary node in the last 15 minutes by taking the default values of health beat interval (5minutes) * missed heartbeat count (3). If there is a missed heartbeat, then the health notification check sends a "heartbeat failure" notification and exits.
- If there is no heartbeat failure but there is a service failure, then the health notification check sends a "service failure" notification and exits.

HA cluster health notification checks when the HA cluster is set up with an external PostgreSQL database

If the PostgreSQL database is externalized, the email notifications generated by the primary node are different from when the PostgreSQL database is not externalized. On the primary node for every secondary/passive node, the HA cluster health notifications checks:

- If there is a heartbeat miss from the secondary/passive node(s) in the last 15 minutes by taking the default values of health beat interval (5 minutes) * missed heartbeat count (3). If there is a missed heartbeat, then the health notification check sends a "heartbeat failure" notification as follow and exits:
  ```
  Following secondary FortiSOAR node(s) seem to have failed -
  Node: hasecondary.myorgdomain
  Failure reason: Heartbeat failure. Check if the 'cyops-ha' service is running or not using
  'systemctl status cyops-ha'.
  ```
- If any service is not running, then the health notification check sends a "service failure" notification as follows and exits:
  ```
  Following secondary FortiSOAR node(s) seem to have failed -
  Node: hasecondary.myorgdomain
  Failure reason: cyops-auth service(s) not running.
  ```

HA cluster health notification checks when a secondary node is firedrilled

When a firedrill is in progress on a secondary/passive node, then you do not receive any 'out of sync' notification, instead the health notification check sends the following email notification and exits.

```
Firedrill is in progress on following node(s):
Node: hasecondary.myorgdomain
Current lag with primary node : 52 kB
```

You can ignore the lag that is displayed in this case since this lag indicates the amount of data the firedrill node needs to sync when `csadm ha restore` is performed.

You can also check the lag using the `get-replication-stat` command on the primary node. If a firedrill is in progress on a secondary/passive node, then you can ignore the lag displayed. This is because the 'total_lag' that gets shown in the `get-replication-stat` messages indicates the amount of data the secondary/passive node will need to sync when the `csadm ha restore` operation is performed on the node once the firedrill completes.

HA cluster health notification checks during takeover

1. When takeover is in progress, the previous primary node might send 'out of sync' email notifications for the node that is taken over, because the previous primary sees it as not replicating data anymore. These can be ignored. After the takeover is completed, we mark the previous primary node as `faulted`. Therefore, you will not see any replication statistics on the old primary node.
2. After the takeover is performed, you can ignore the messages of the `get-replication-stat` command on the new primary node. You can also ignore the 'out of sync' email notification that is generated by the new primary node since when we perform the takeover, the entries of all the nodes in the cluster are yet included in `csadm ha list-nodes`, and because the remaining nodes yet require to join the new primary node, this new primary node keeps generating the notification for all those nodes.
3. When all the other nodes of a HA cluster join back to the new primary node, then the health notification check starts to work and there will not be any ignorable notification.

Troubleshooting issues based on the notifications

The following section provides details on how to check and fix the possible reasons of failures that are listed in the email notifications sent by the HA cluster check.

To troubleshoot HA issues, you can use the HA log located at: `/var/log/cyops/cyops-auth/ha.log`.

WAL files size exceeds the configured threshold

The default threshold for WAL files size is 20 GB. The WAL files are stored at `/var/lib/pgsql/<postgresql_version>/data/pg_wal`.

Resolution:

1. Check if you have a higher value set for '`wal_keep_size`' in the `/var/lib/pgsql/<postgresql_version>/data/postgresql.conf` file. The default value is 1024 that denotes 1024MB, which means that the `pg_wal` directory will have 1024MB WALs.
   Reduce this value and restart the `postgresql-<postgresql_version>` service. For example:
   `systemctl restart postgresql-16`
2. If the 'wal_keep_size' value is lower, then do the following:
   a. List all replication slots using the following command:
   `csadm ha utils replication-slots list`
   The CLI will display the output as follows:
   ```
   # csadm ha utils replication-slots list
   slot_name                                        active
   ------------------------------------------- --------
   slot_phy_fsr_a82d365c74172096671a54741e074191 False
   slot_phy_fsr_46a5e8c98e7c77e4d16f5d29f28d2bc3 True
   #
   ```
   b. The format for `slot_name` is '`slot_phy_fsr_<nodeId>`'. The 'nodeId' is similar to the 'nodeId' shown in '`csadm ha list-nodes`'. The second column can have True/False. False indicates that the slot is inactive. You need to find all the inactive slots and compare their nodeIds against the output of the '**csadm ha list-nodes**' sub-command. Using this method, you can find out the nodes that have inactive slots present on the current server, and if that node is not part of the cluster, then you can remove its replication slot.
   Note: If a node is temporary down, then also the second column will display as False, i.e., indicative of an inactive slot. If you know that this node will not come online soon, then you can remove its replication slot.
   c. Remove the inactive replication slots that you have identified in the previous step using the following command:
   `casdm ha utils replication-slots remove --slot-name <>`
   For example:
   `casdm ha utils replication-slots remove --slot-name slot_phy_fsr_a82d365c74172096671a54741e074191`

Heartbeat Failure

Resolution:

When you get a heartbeat failure notification for a secondary node, then do the following:

1. Check if the `cyops-ha` service is running on that node, using the `systemctl status cyops-ha` command.
2. If it is not running, then you must restart the `cyops-ha` service.

Node name differs from actual FQDN

Resolution:

Correct the notification such as `nodeName from 'csadm ha list-nodes' differs from actual FQDN used during join-cluster.` using the following steps:

1. Login on the node for which you are receiving the above notification using SSH.
2. Use the following command to correct the FQDN of the node:
   `csadm ha set-node-name <enter-correct-FQDN-here>`

Secondary/Passive node is out of sync with the Primary node

This issue could occur due to the following reasons:

- PostgreSQL service status shows `requested WAL segment <some-number-here> has already been removed`
  OR
  The `csadm ha get-replication-stat` command shows a higher time lapsed from the last sync when compared to the general time lapsed.
  Resolution:
  In these cases, since the secondary/passive node is completely out of sync with the primary node, you need to perform the following steps on the secondary/passive node:
  1. Run the `touch /home/csadmin/.joincluster_in_progress` command to create the `.joincluster_in_progress` file. 2. Rejoin the cluster as follows:
  `csadm ha join-cluster --status active/passive --role secondary --primary-node <Primary-node-FQDN> --fetch-fresh-backup`
- When there is heavy write on the primary node and the secondary node has not yet copied the data before the data has rolled over, it will be 'out of sync' and a full synchronization is needed, which can cause the above failures.
  Resolution:
  Increase the `wal_keep_size` setting in the `/var/lib/pgsql/16/data/postgresql.conf` file.

PostgreSQL service is down on the primary node or PostgreSQL service is down on the externalized database host

If PostgreSQL service is down on the primary node, then the `cyops-ha service` on all the nodes will be down and there will be no notifications generated since the whole cluster is down; due to this you will also not be able to login to the FortiSOAR UI.

Resolution:

1. Check the reason for the failure using the `systemctl status postgresql-<postgresql_version_here> -l` on the primary node or the externalized database host.

2. Fix the issue based on the reason for failure.

Sample scale test that were done in the lab to understand the behavior of 'csadm ha get-replication-stat'

What was done before observing the behavior:

First we stopped the PostgreSQL service on the secondary/passive node.

Next, generated data on the primary node using the following script. You need to kill the script after some time when enough data is generated on the primary node.

```
[root@cybersponse csadmin]# cat data_load.sh
#!/bin/sh

psql -U cyberpgsql -d das -c "CREATE TABLE scale_data (
   section NUMERIC NOT NULL,
   id1     NUMERIC NOT NULL,
   id2     NUMERIC NOT NULL
);"


psql -U cyberpgsql -d das -c "
INSERT INTO scale_data
SELECT sections.*, gen.*
     , CEIL(RANDOM()*100)
  FROM GENERATE_SERIES(1, 300)     sections,
       GENERATE_SERIES(1, 900000) gen
 WHERE gen <= sections * 3000;"
[root@cybersponse csadmin]#
```

During the data generation process, we ran the `csadm ha get-replication-stat` on the primary node and you can observe that the secondary node is lagging with 4702 MB.

```
get-replication-stat on the primary node:

[root@cybersponse csadmin]# csadm ha get-replication-stat
------------------------------------------------
Warning:
Following could be the issues with the nodes:
1. The postgres database replication to the secondary node is not working due to data log rotation
at the primarynode.
2. The secondary node has been shutdown/halted.
3. PostgreSQL not running on node(s).
4. nodeName from 'csadm ha list-nodes' differs from actual FQDN used during join-cluster.
5. If a firedrill is in progress on the node, no action is required. The 'lag' that is displayed
indicates the amount of data the firedrill node needs to sync when 'csadm ha restore' will be
performed.

If node is up and running,
1.  Check the status of PostgreSQL service using 'systemctl status postgresql-16 -l' on node to
```

```
get more details.
2. If you see 'FATAL: could not receive data from WAL stream: requested WAL segment has already
been removed' in the PostgreSQL service status, you need to re-join the cluster using 'csadm ha
join-cluster --fetch-fresh-backup' for this node.

------------------------------------------------

nodeId                            nodeName                 status    role       comment
                          total_lag
-------------------------------  ----------------------  --------  ---------  ------------------
-------------------------  -----------
469c6330613a332c30dd4d8e3a607cf2  hasecondary.myorgdomain  active    secondary  Joined cluster
with haprimary.myorgdomain  4702 MB
[root@cybersponse csadmin]#
```

Next, start PostgreSQL on the secondary node and observed the 'replication-stat' on the primary node:

```
On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:27:31 2020

------------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
------------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  4458 MB        11 MB            213 MB           4683 MB

On the Primary node:

------------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
------------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  3600 MB        3456 kB          727 MB           4330 MB


On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:28:05 2020

------------------------------------------------
```

```
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  2774 MB        5632 kB          1273 MB          4052 MB


On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:28:28 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  1910 MB        6784 kB          1803 MB          3719 MB


On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:28:44 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  1153 MB        1408 kB          2278 MB          3433 MB

On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:29:00 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
```

```
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  452 MB         3200 kB          2726 MB          3181 MB
On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:29:25 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  0 bytes        0 bytes          2658 MB          2658 MB

On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:29:46 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
hasecondary.mydomain  0 bytes        154 kB           2172 MB          2172 MB


On the Primary node:

Every 2.0s: csadm ha get-replication-stat
Tue May 12 05:30:06 2020

-----------------------------------------------
Note:
'sending_lag' indicates load on the primary node
'receiving_lag' indicates network delay or load on the passive/secondary node
'replaying_lag' indicates load on the passive/secondary node
-----------------------------------------------

node_hostname            sending_lag    receiving_lag    replaying_lag    total_lag
-----------------------  -------------  ---------------  ---------------  -----------
```

```
hasecondary.mydomain  0 bytes        0 bytes          1687 MB          1687 MB
```

Disaster Recovery

- **Nightly database backups and incremental VM snapshots**: FortiSOAR provides backup scripts that are scheduled to run at pre-defined intervals and take full database backup on a shared or backed up drive. The full backups have to be supplemented with incremental Virtual Machine (VM) snapshots whenever there are changes made to the file system, such as connector installation changes, config file changes, upgrades, schedule changes, etc. For more information, see the Backing up and Restoring FortiSOAR chapter.
- **HA provided by the underlying virtualization platform**: Your Virtualization platform also provides HA, such as VMware HA and AWS EBS snapshots. This method relies on your expertise and infrastructure.
- **Externalized Database**: This method allows you to externalize your PostgreSQL database and uses your own database's HA solution. VM snapshots have to be taken when there are changes made to the file system, such as connector installation changes, config file changes, upgrades, schedule changes, etc. For more information on externalizing PostgreSQL database, see the Externalization of your FortiSOAR PostgreSQL database chapter.

FAQs

Q: How are scheduled playbooks distributed among two or more HA Active/Active nodes? Would these scheduled playbooks become duplicates?

A: No. The scheduled playbooks do not become duplicates, as the playbooks get scheduled across the active nodes in the HA cluster.

Q: In an HA Active/Active cluster that contains 3 or more nodes, which node becomes the primary node when the original primary node is down? Is there any election of primary role, or an in-built logic to auto choose a primary node, when the original primary goes down?

A: No. FortiSOAR does not have any election or in-built logic to auto choose a primary node when the original primary goes down. Users must manually initiate the 'Takeover' operation from the node that they want to make the new primary node.

Q. Does the original primary pre-empt the secondary primary node, when it comes online? If it does, then what happens to the DB changes made in the secondary node while the primary node was down?

A. When the original primary node comes up, it remains outside of the HA cluster, Users must manually initiate the 'Join Cluster' operation to add it back to the HA cluster with its respective role.

Q. What detection method is used for determining the status of the nodes in an HA cluster?

A. All secondary nodes in the cluster exchange HA heartbeat packets with the primary node so that the primary node can monitor and verify the status of all the secondary nodes and the secondary nodes can verify the status of the primary node.

Q. Can the load balancer determine that a node is down, and if yes, does it stop sending traffic to the node that is down?

A. This depends on how you configure your load balancer. In the case of an active-active HA cluster, if the load balancer is configured to move to backend nodes (secondary active in the cluster) if the primary fails, then the load balancer will pass requests to backend nodes. However, users need to make the backend node (secondary active) the primary node in the cluster when the primary fails, for a request to succeed.

Q. How does the fall back mechanism work for a 2-node HA (Active-Active) cluster when one node is down? How much delay is expected for recovery? What are the effects of a failover event to the SOC workflows for both nodes?

A. If any node in an HA cluster is down, it does not execute any workflow tasks, and these tasks are consumed by the nodes that are available. SOC workflows run on the queue-based distributed task execution model: all playbooks triggered are put in a queue at the primary node that is monitored by all other active nodes, and the active nodes pick tasks from this queue.

Q. How are HTTP requests distributed across active-active HA nodes, with or without stickiness of the application session?

A. HTTP requests are split between active-active HA nodes in accordance with load balancer settings. All requests can be routed through the primary node by the load balancer, or the policy can be configured for requests to be distributed round-robin among nodes.

Q. How do Active/Active HA nodes avoid duplication of SOC workflows?

A. SOC workflows run on the queue-based distributed task execution model: all playbooks triggered are put in a queue at the primary node that is monitored by all other active nodes, and the active nodes pick tasks from this queue.

Q. How are schedule playbooks configured in an HA Active/Active mode? Should the scheduled playbooks be configured in both the Active/Active nodes?

A. The schedule playbook can be configured from any node in the HA cluster or in the load balancer since there is only one master database that handles write operations, and all the others are standby nodes.

Troubleshooting

To troubleshoot HA issues, you can use the HA log located at: `/var/log/cyops/cyops-auth/ha.log`. To understand and troubleshoot the HA cluster health notifications, see the Monitoring health of HA clusters section.

Failure to create an HA cluster

If the process to configure HA using the automated `join cluster` fails, and the HA cluster is not created due to reasons such as, proxies set up etc, you can perform the steps mention in the following procedure and

configure HA:

1. Connect to your VM as a *root* user and run the following command:
   `# csadm ha`
   This will display the options available to configure HA.
2. To configure a node as a secondary node, perform the following steps:
   a. SSH to the active primary node and run the `csadm ha export-conf`command to export the configuration details of the active primary node to a configuration file named `ha.conf`.
   You must copy the `ha.conf` file from the active primary node to the node that you want to configure as a secondary node.
   b. On the active primary server, add the hostnames of the secondary nodes to the allowlist, using the following command:
   `# csadm ha allowlist --nodes`
   Add the comma-separated list of hostnames of the cluster nodes that you want to the add to the allowlist after the `--nodes` argument.
   Important: In case of an externalized database, you need to add all the nodes in a cluster to the allowlist in the `pg_hba.conf` file.
   c. Ensure that all HA nodes are resolvable through DNS and then SSH to the server that you want to configure as a secondary node and run the following command:
   `# csadm ha join-cluster --status <active, passive> --role <primary, secondary> --conf <location of the ha.conf file>`
   For example, `# csadm ha join-cluster --status passive --role secondary --conf tmp/ha.conf`
   This will add the node as a secondary node in the cluster.
   Note: If you run the `csadm ha join-cluster` command without adding the hostnames of the secondary nodes to the allowlist, then you will get an error such as, `Failed to verify....`
   Also, when you join a node to an HA cluster, the `list-nodes` command does not display that a node is in the process of joining the cluster. The newly added node will be displayed in the `list-nodes` command only after it has been added to the HA cluster.

Timeout failure while creating an HA cluster

The join-cluster operation fails with the "`Error: command timedout`". Currently, the timeout value set is 21600 seconds, i.e., 6 hours. However, if the primary node has a huge amount of data, 200 GB or more, then it is possible for the join-cluster operation to timeout.

Resolution

If you get a timeout error while running the join-cluster operation, then you can increase the value of the timeout key in the `config.yaml` file as follows:

1. Open the `/opt/cyops/configs/ha/config.yaml` file.
2. Change the value of the `timeout` key. The value of the `timeout` key is stored in seconds, and currently, it is set to 21600 seconds (6 hours).
3. Restart the `cyops-ha` service using the `systemctl restart cyops-ha` command.
   Note: You need to perform these steps on each HA cluster node.

Unable to add a node to an HA cluster using join-cluster, and the node gets stuck at a service restart

This issue occurs when you are performing `join-cluster` of any node and that node sticks at service restart, specifically at PostgreSQL restart.

Resolution

Terminate the `join-cluster` process and retry `join-cluster` using the `--fetch-fresh-backup` argument.

Fixing an HA cluster whose Primary node is halted and then resumed

If the primary node of your HA cluster is halted due to a system crash or similar issue, and a new cluster is formed using the other nodes, running the `list-nodes` command on the remaining nodes will show the primary node as being in the `Faulted` state. Since the administrator has triggered a takeover on the other cluster nodes, they will be aware of the faulted primary node. Even after the primary node is resumed, it will still be the primary node of its original cluster. As a result, when the node resumes, the `list-nodes` command on the primary node will show this node as `Primary Active`, and services such as `rabbitmq-server` and `fsr-api-consumer` may not be in the 'running' state.

Resolution

To ensure that only one node is marked as the 'Primary Active' node in the HA cluster, follow these steps:

1. On the resumed primary node, run the following commands:
   a. Reconfigure the 'rabbitmq' service using the following command:
   `csadm mq db flush`
   b. Run the `leave-cluster` command, which will remove this node from the HA cluster.
   c. Run the `join-cluster` command to add the node back to the HA cluster with the new primary node.

Unable to join a node to an HA cluster when a proxy is enabled

You are unable to join a node to an HA cluster using the `join-cluster` command when you have enabled a proxy using which clients should connect to the HA cluster.

Resolution

Run the following commands on your primary node:

```
$ sudo firewall-cmd --zone=trusted --add-source=<CIDR> --permanent
$ sudo firewall-cmd --zone=trusted --add-port=<ElasticSearchPort>/tcp --permanent
```

```
$ sudo firewall-cmd --reload
```

For example,

```
$ sudo firewall-cmd --zone=work--add-source=192.168.0.0/20 --permanent
$ sudo firewall-cmd --zone=work --add-port=8080-8090/tcp --permanent

$ sudo firewall-cmd --reload
```

Changes made in nodes in an active-active cluster fronted with a load balancer take some time to reflect

In the case of a FortiSOAR active-active cluster that is fronted with a load balancer or reverse proxy such as an HAProxy, changes such as, adding a module to the FortiSOAR navigation, updating or adding the permissions of the logged-in user, or updates done to the logged-in user's parents, child, and sibling hierarchy, do not get reflected immediately.

These issues occur due to local caching of these settings at the individual cluster nodes.

Resolution

Log off and log back into the FortiSOAR user interface after ten minutes to see the recent updates.

OR

If you want the settings to reflect immediately, run the following command on the "active" nodes in the cluster:
`php /opt/cyops-api/bin/console --env=prod app:cache:clear --env=prod`
Important: You do not require to run the above command on the "passive" nodes of the cluster.

Post Takeover the nodes in an HA cluster do not point to the new active primary node

This issue occurs when during the takeover process either the previous primary node is down or automatic `join-cluster` fails. In case of an internal database cluster, when the failed primary node comes online after the takeover, it still thinks of itself as the active primary node with all its services running. In case of an external database cluster, when the failed primary node comes online after the takeover, it detects its status as "Faulted" and disables all its services.

Resolution

Run the `csadm ha join-cluster` command to point all the nodes to the new active primary node. For details on `join-cluster`, see Process for configuring HA.

After performing the leave-cluster operation, the license is not found on a secondary node

In case of an internal DB, after you have performed the leave-cluster operation on a secondary node, if for example, you are upgrading the node, and when you are rejoining the node to the cluster, once the upgrade is done, you might see the following error: "`License not found on the system`". You might also see this error while trying to perform the 'restore' operation on a secondary node after completing the 'firedrill' operation.

Resolution

Run the following script as a *root* user on the secondary node on which you are getting the "`License not found on the system`" error:

```
#!/bin/bash

init(){
    current_pg_version=$(/bin/psql --version | egrep -o '[0-9]{1,}\.' | cut -d'.' -f1)
    re_join_cluster_sql="/opt/cyops-auth/.re-join-cluster.sql"
    db_config="/opt/cyops/configs/database/db_config.yml"
}

re_join_cluster(){
    if [ ! -f "$re_join_cluster_sql" ]; then
        echo "File [$re_join_cluster_sql] does not exist. Contact the Fortinet support team for
further assistance"
        exit 1
    fi
    csadm services --stop
    if [ ! -d "/var/lib/pgsql/${current_pg_version}.bkp" ]; then
        mv /var/lib/pgsql/$current_pg_version /var/lib/pgsql/${current_pg_version}.bkp
    fi
    # Below rm is required if in case user re-run the script again.
    rm -rf /var/lib/pgsql/${current_pg_version}
    rm -f /opt/cyops/configs/cyops_pg_${current_pg_version}_configured
    mkdir -p /var/lib/pgsql/${current_pg_version}/data
    chown -R postgres:postgres /var/lib/pgsql/${current_pg_version}
    chmod -R 700 /var/lib/pgsql/${current_pg_version}
    /opt/cyops-postgresql/config/config.sh ${current_pg_version}
    local hkey=$(csadm license --get-device-uuid)
    sudo -Hiu postgres psql -U postgres -c "ALTER USER cyberpgsql WITH ENCRYPTED PASSWORD
'$hkey';"
    createdb -U cyberpgsql -e -w --no-password das -O cyberpgsql -E UTF8
    psql -U cyberpgsql -d das < $re_join_cluster_sql
    touch /home/csadmin/.joincluster_in_progress
    if [ ! -f "${db_config}.bkp" ]; then
        yes| cp ${db_config} ${db_config}.bkp
    fi
    local db_pass_encrypted=$(python3 /opt/cyops/scripts/manage_passwords.py --encrypt $hkey)
    /opt/cyops/scripts/confUtil.py -f $db_config -k 'pg_password' -v "$db_pass_encrypted"
    systemctl start postgresql-${current_pg_version}
}

echo_manual_steps(){
echo "Perform below steps manually"
echo "
    1. If node is passive, then run below command, else skip it.
       csadm ha join-cluster --status passive --primary-node <primary-node> --fetch-fresh-backup
    2. If node is active/secondary.
       csadm ha join-cluster --status active --role secondary --primary-node <primary-node> --
fetch-fresh-backup
    3. rm -rf /var/lib/pgsql/${current_pg_version}.bkp
```

```
    4. rm -f /opt/cyops/configs/database/db_config.yml.bkp
    5. rm -rf /var/lib/pgsql/test_dr_backups"
}
####################
# Main/main/MAIN starts here
####################

# Stop right after any command failure
set -e
# Debug mode
set -x
init
re_join_cluster
####################
# You need to perform the below step manually.
########
# Turn off the dubug mode to see the steps to perform manually clearly
set +x
echo_manual_steps
exit 0
```

The leave-cluster operation fails at the "Starting PostgreSQL Service" step when a node in the cluster is faulted

This issue occurs in case of an active-active-passive cluster that has an internal db and whose HA cluster contains a node whose status is 'Faulted'. In this case, when you run the `leave-cluster` operation it fails at the "`Starting service postgresql-16`" step.

Resolution

To resolve this issue, run the following commands:

```
systemctl stop postgresql-16
rm -f /var/lib/pgsql/16/data/standby.signal
systemctl start postgresql-16
```

Once you have completed performing the above steps, run the `csadm ha leave-cluster` command.

Resetting the password for an instance that is part of active/active cluster causes the other instances of that cluster to not able to log in to FortiSOAR

If you reset the password of an instance that is part of an active/active cluster, then the FortiSOAR login page is not displayed for the other instances of this cluster. You will also observe data login failure errors in the `ha.log`

and the `prod.log` in case of both active/active and active/passive clusters.

Resolution

On the other instances that are part of the cluster, do the following:

1. Copy the encrypted password from the `db_config.yml` file located at `/opt/cyops/configs/database/db_config.yml` on the active node and then update the new password in the `db_config.yml` file on the secondary nodes.
2. Run the `cache:clear` command:
   `$ sudo -u nginx php /opt/cyops-api/bin/console cache:clear`
3. Restart FortiSOAR services:
   `# csadm services --restart`

The database of a secondary node in the HA cluster is out of synch with the database of the primary node

If the database of a secondary node in the HA cluster is completely out of sync, then you can should rerun the `join-cluster` command. However, if rejoining the cluster also does not resolve the secondary node issue, then perform the steps mentioned in the resolution.

Resolution

Run the `csadm ha clone-db` command to clone the database from the cluster's primary node. For information on the `csadm ha clone-db` command, see the Usage of the csadm ha command section.

The `csadm ha clone-db` command just copies the database of the primary node; therefore, this is not an alternative to backup and restore, which involves many other processes and should be continued to be performed. For information on backup and restore see the Backing up and Restoring FortiSOAR chapter.

Failure of the 'leave-cluster' command on a secondary node that had its database externalized before joining the HA cluster, and that is currently using an internal database

The '`leave-cluster`' command fails with the '`"FATAL: password authentication failed for user "cyberpgsql"`' error on a secondary node that had its database externalized before joining the HA cluster, and that is currently using an internal database.

Resolution

To resolve this issue, perform the following steps on the node where the `leave-cluster` command has failed:

1. Edit the `/var/lib/pgsql/16/data/pg_hba.conf` file to update the following:
   a. Update the '`local all all md5`' line to '`local all all trust`'.
   b. Update the '`host all all 127.0.0.1/32 md5`' line to '`host all all 127.0.0.1/32 trust`'.
2. Restart the PostgreSQL service using the '`systemctl restart postgresql-16`' command.
3. Edit the `/opt/cyops/configs/database/db_config.yml.default` file to update the following parameters:
   a. Update '`pg_external`' and '`pg_archival_external`' to `localhost`.
   b. Update '`pg_host`' and '`pg_archival_host`' to `localhost`.
   c. Update '`pg_port`' and '`pg_archival_port`' to `5432`.
   d. Update '`pg_ssl_cert_path`' and '`pg_archival_ssl_cert_path`' to `''`.
   e. Update '`pg_use_ssl`' and '`pg_archival_use_ssl`' to `false`.
   f. Update '`pg_user`' and '`pg_archival_user`' to `cyberpgsql`.
4. Run the '`csadm ha leave-cluster`' command.
5. Once '`leave-cluster`' is completed, revert the changes made in **step 1** to the `pg_hba.conf` file, and then restart the PostgreSQL service using the '`systemctl restart the postgresql-<postgresql_version> service. For example:`
   `systemctl restart postgresql-16stgresql-16`' command.

Common failures observed during the join-cluster operation

The `ha.log` is located at `/var/log/cyops/cyops-auth/ha.log`.

Failure to verify rabbitmq credentials

- The `ha.log` contains an error such as `hostname '<abc>' doesn't match <xyz>'`. This issue can occur when MQ certificates are regenerated with a different hostname.
  Resolution
  To resolve this issue, regenerate MQ certificates on the primary node using the `csadm mq certs generate` command.
  It is important to exercise caution while running this command, especially if embedded secure message exchange (SME) is configured and FSR Agents are running as in the case of FortiSOARCloud.
- The `ha.log` contains an error such as `Hostname mismatch, certificate is not valid for abc@xyz.com. (_ssl.c:1129)`. This issue can occur when a previously exported `ha.conf` is being used even after a hostname change.
  Resolution
  To resolve this issue, use the latest exported `ha.conf`.
  OR
  Run `join-cluster` using SSH credentials.
- The `ha.log` contains an error such as `Failed while fetching ha.primary.fortisoar.in version. Error: HTTPSConnectionPool(host='ha.primary.fortisoar.in', port=443): Max retries exceeded with url: /version.json (Caused by ProxyError('Cannot connect to proxy.', OSError('Tunnel connection failed: 403 Forbidden')))`. This issue can occur when a proxy is enabled but 'no-proxy' is not added for the primary host.
  Resolution
  Add all nodes in 'no-proxy'.

Failure to verify elasticsearch credentials

The `ha.log` contains an error such as `ConnectionError(HTTPSConnectionPool (host='ha.primary.fortisoar.in', port=9200): Max retries exceeded with url: / (Caused by NewConnectionError('<urllib3.connection.HTTPSConnection object at 0x7fe91e710220>: Failed to establish a new connection: [Errno 111] Connection refused'))) caused by: ConnectionError (HTTPSConnectionPool(host='ha.primary.fortisoar.in', port=9200): Max retries exceeded with url: / (Caused by NewConnectionError('<urllib3.connection.HTTPSConnection object at 0x7fe91e710220>: Failed to establish a new connection: [Errno 111] Connection refused')))`. This issue can occur when elasticsearch is down on the primary node or port 9200 is not open.
Resolution
To resolve this issue, check the elasticsearch service on the primary node. Also, check if port 9200 is opened in the firewall.

Failure to verify PostgreSQL credentials

This failure can occur due to hostname resolution issues. The hostname must be correct for both forward and reverse lookup to validate the IP and hostname bidirectionally. This means that the hostname used in `pg_hba.conf` should be the one that address-to-name resolution of the client's IP address returns, otherwise the line won't be matched. In some cases, even if the primary node is resolvable from secondary nodes, the `join-cluster` operation might fail due to this problem. For more information, see auth-pg-hba-conf.

Join-cluster fails on the differential sync step

This issue commonly occurs in re-joining cluster scenarios where there is a significant data difference between the primary node and secondary nodes, causing `pg_rewind` to fail. This typically happens in two cases:

- Adding a node back to the cluster after a long absence.
- Following a FortiSOAR upgrade, re-joining a node to the cluster after significant data has been ingested.

Resolutions:

Depending on the situation, you can resolve the issue by using any of the following resolutions:

- If the PostgreSQL service on the secondary node is up and running then terminate the `join-cluster` process and retry `join-cluster` using the `--fetch-fresh-backup` argument.
- If the PostgreSQL service is not starting, and the `join-cluster` process is stuck at “Starting postgresql server”. This could possibly occur due to the following error: “could not receive data from WAL stream: ERROR: requested WAL segment 000000010000000000000020 has already been removed”.
  To resolve this issue, follow these steps:
  a. Run the `clone-db` command:
  ```
  csadm ha clone-db --conf ha.conf
  ```
  OR
  ```
  csadm ha clone-db --primary-node --prepare-for-immediate-join-cluster
  ```
  b. Connect to your local PostgreSQL (password is of the primary node).
  c. Truncate the 'envc' and 'cluster' tables.
  d. Run the `csadm ha join-cluster --status active --role primary` command.
  e. Deploy your license.
  f. Rerun the `join-cluster` command without the '--fetch-fresh-backup' option.

Elasticsearch Configuration

FortiSOAR leverages the fast search capability of Elasticsearch for quick text search across all records and files in the FortiSOAR database. FortiSOAR supports externalization of Elasticsearch data. Externalization is indexing of data to an Elasticsearch instance that has the same or higher version of Elasticsearch outside of the FortiSOAR virtual appliance; the steps for which are covered in this chapter.

The minimum version of your Elasticsearch cluster must be 7.0.2, if you want to externalize your ElasticticSearch data.

If you want to externalize your other FortiSOAR PostgreSQL database, see the Externalization of your FortiSOAR PostgreSQL database chapter.

Externalization and Authentication of Elasticsearch

If you require to change the location of your Elasticsearch instance from your local instance to a remote machine, run the following steps on your externalized Elasticseach machine:

1. Use the Elasticsearch documentation to install elasticsearch.
   Important: Ensure that you install the same version of elasticsearch that is currently installed on your FortiSOAR instance on the externalized Elasticseach machine. For example, if you have version elasticsearch v7.2.0 currently on your FortiSOAR instance, then u must install elasticsearch v7.2.0 on the remote machine on which you want to externalize elasticseach.
2. Configure elasticsearch to accept connections from outside 'localhost' by updating the '`network.host`' and '`discovery.type`' variables to `network.host: 0.0.0.0`' and '`discovery.type: single-node`' in the `/etc/elasticsearch/elasticsearch.yml` file.
   You must comment the `cluster.initial_master_nodes` line in the `elasticsearch.yml` file. For example, `#cluster.initial_master_nodes: ["example_hostname"]`
3. Run the following commands:
   ```
   # mkdir -p /opt/cyops-search
   # chmod 755 /opt/cyops-search
   ```
4. Copy the `/opt/cyops-search/exclude.list` file from your FortiSOAR machine to the `/opt/cyops-search/exclude.list` file on your externalized elasticsearch host.
5. Copy the `/etc/elasticsearch/security.policy` file from your FortiSOAR machine to the `/etc/elasticsearch/security.policy` file on your externalized elasticsearch host.
6. Run the following commands:
   ```
   # chmod 644 /opt/cyops-search/exclude.list
   # chmod 660 /etc/elasticsearch/security.policy
   # chown root:elasticsearch /etc/elasticsearch/security.policy
   ```

7. Append the '`-Djava.security.policy=/etc/elasticsearch/security.policy`' line to the `/etc/elasticsearch/jvm.options` file.
8. Start elasticsearch using the following command:
   `# systemctl start elasticsearch`
9. Open the elasticsearch port in your firewall, using the following command:
   ```
   # firewall-cmd --permanent --add-port=9200/tcp
   # firewall-cmd --reload
   ```
10. (Optional) To enable SSL for elasticsearch, see the https://www.elastic.co/blog/configuring-ssl-tls-and-https-to-secure-elasticsearch-kibana-beats-and-logstash blog.
11. Once elasticsearch is externalized, it is recommended that you stop and mask 'localhost' elasticsearch on your FortiSOAR machine, using the following commands:
    ```
    # systemctl stop elasticsearch
    # systemctl disable elasticsearch
    # systemctl mask elasticsearch
    ```

Once elasticsearch is externalized, update the `db_config.yml` file, which is located at: `/opt/cyops/configs/database/db_config.yml`

In the `db_config.yml` file, you require to update the host and port (if needed) in the `elasticsearch` section that appears as follows:

```
elasticsearch:
  es_external: false
  es_host: localhost
  es_port: 9200
  es_user: elastic
  initial_backoff: 60
  max_backoff: 6000
  secret: None
  ssl_cert_path: ""
  use_ssl: false
```

To change the location of your Elasticsearch instance from your local instance to a remote machine:

- `es_external: false` > Set the value of `es_external` to `true` to externalization of elasticsearch.
- `es_host: localhost` > Update host value with the hostname or IP address of the remote Elasticsearch machine.
- `es_port: 9200` > Update the port required to access the remote Elasticsearch machine, if required.

For authentication of Elasticsearch:

- `es_user: elastic` > Update the username that is used to access the remote Elasticsearch machine, if Authentication is enabled on the remote Elasticsearch machine.
- `secret: None` > Update the secret (password) that is used to access the remote Elasticsearch machine, if Authentication is enabled on the remote Elasticsearch machine.

You also require to assign `ngnix` permission to the SSL certificate that you have specified in the `db_config.yml` file using the following command:

```
chown nginx:nginx filename.pem
```

Externalized elasticseach must have SSL enabled for use in the FortiSOAR high availability cluster. Also, ensure that you set the `use_ssl` flag to 'true' and specify the `ssl_cert_path` as the path of your external elasticsearch CA certificate.

Migration of Elasticsearch data

Once you complete the externalization of Elasticsearch, you will require to migrate your data from your local instance to the remote Elasticsearch machine.

To migrate the remote Elasticsearch machine run the following command on your FortiSOAR instance as a `root` user after changing the directory to `/opt/cyops-api/`:

```
$ sudo -u nginx php /opt/cyops-api/bin/console app:elastic:create --env="prod"
```

Troubleshooting Tips

FortiSOAR Search Errors

FortiSOAR Search performs indexing in an asynchronous fashion in the backend. Users could be faced with certain scenarios that could lead to a restart of services, which can cause indexing to stop. In this case, FortiSOAR might display any of the following errors when users are performing a search operation on FortiSOAR:

- `Search indexing is in progress. Partial results are returned.`
- `Search indexing has stopped. You must manually rerun indexing (see product documentation for instructions) or raise a support ticket for the same.`
- `We are sorry, but the server encountered an error while handling your search request. Please contact your administrator for assistance.`

In this case, use the `/var/log/cyops/cyops-search/falcon.log` log file to check which modules are published and indexed and which modules are yet to be published (pending).

For example, the `/var/log/cyops/cyops-search/falcon.log` log file will display results as follows:

```
2019-02-13,11:00:44 INFO blocking_connection: _dispatch_events(): 1445: Module Currently Getting
Published: ['attachments']
2019-02-13,11:00:44 INFO blocking_connection: _dispatch_events(): 1445: Indexing for Module:
'attachments' started Total Records to be indexed: '1'
2019-02-13,11:00:49 INFO blocking_connection: _dispatch_events(): 1445: Module: 'attachments'
Successful Total Records indexed: '1'
2019-02-13,11:00:49 INFO blocking_connection: _dispatch_events(): 1445: on_publish_message called
2019-02-13,11:00:53 INFO blocking_connection: _dispatch_events(): 1445: creating index with
mapping
2019-02-13,11:01:00 INFO blocking_connection: _dispatch_events(): 1445: Module Currently Getting
Published: ['emails']
2019-02-13,11:01:02 INFO blocking_connection: _dispatch_events(): 1445: Indexing for Module:
'emails' started Total Records to be indexed: '1'
2019-02-13,11:01:04 INFO blocking_connection: _dispatch_events(): 1445: Module: 'emails'
Successful Total Records indexed: '1'
```

The above example shows the `attachments` and `emails` modules currently being indexed and its total number of records. Any failure in indexing any modules will be logged here. You can monitor the progress of this file while the indexing is in progress.

If any module(s) are missing from the published list or if any module has the `Publish Module: '<name of module>' Unsuccessful` listed in the `/var/log/cyops/cyops-search/falcon.log` log file; the `indicators` and `tasks` modules in our example, then you must manually run the indexing for those module(s) using the following command:

```
$ sudo -u nginx php bin/console app:elastic:create --env="prod" --index='{"type":["<list of comma-seperated module names that require to be indexed>"]}
```

For our example, run the following command:

```
$ sudo -u nginx php bin/console app:elastic:create --env="prod" --index='{"type":["indicators","tasks"]}'
```

Optimizing the reindexing of Elasticsearch data

If reindexing is required in your environment for syncing that latest Elasticsearch data, FortiSOAR reindexes only the latest 50,000 (maximum) records. This is because reindexing the entire set of records makes the system sluggish and results in high CPU utilization.

You can customize the maximum number of records to be reindexed by editing the `/opt/cyops/configs/cyops-search/config.yml` file and changing the value of the following parameter: `max_indexing_limit: 50000`.

You need to explicitly add the "`max_indexing_limit`" parameter to the `/opt/cyops/configs/cyops-search/config.yml` file if you have upgraded your instance to FortiSOAR release 7.4.0 or later.

Externalization of your FortiSOAR PostgreSQL database

This chapter explains the steps required to externalize your FortiSOAR PostgreSQL database. For information about ElasticSearch configuration, including ElasticSearch externalization, see the ElasticSearch Configuration chapter.

Externalization is migration of data from your local database instance to a remote database instance that has same version of PostgreSQL, outside of the FortiSOAR virtual appliance.

When using an external database, ensure that it meets the same specifications as a FortiSOAR node. These requirements are detailed in the Planning topic of the Deploying FortiSOAR chapter in the "Deployment Guide."
Additionally, customers are responsible for managing external databases, including performing regular backups and upgrading the PostgreSQL to a version compatible with the installed FortiSOAR release, etc.

To externalize your FortiSOAR PostgreSQL database you must have *root access* on your FortiSOAR system and you must use the FortiSOAR Admin CLI (`csadm`). For more information on `csadm`, see the FortiSOAR Admin CLI chapter.

From release 7.6.0 onwards, FortiSOAR uses advanced PostgreSQL utilities, 'pg_squeeze' and 'pg_repack' to optimize the process of reclaiming unused disk space. For more information, see the *Debugging and Optimizing Playbooks* chapter in the "Playbooks Guide". If you want to use these utilities in the case of an externalized FortiSOAR PostgreSQL database, you must install the utilities using the steps mentioned in the Installing the pg_squeeze and pg_repack extensions topic.

Prerequisites

- Prepare your Remote instance:
  - Provision a Rocky Linux 9.3/9.4/9.5 or RHEL 9.3/9.4/9.5 system as your remote instance.
  - Remote instance must allow inbound communication from your FortiSOAR local Virtual Machine.
  - Remote instance must have PostgreSQL version 16.x. The default Rocky Linux repository does not have PostgreSQL 16 version available. Therefore, to use the PostgreSQL 16 server package from the FortiSOAR repo, do the following:
    i. Disable PostgreSQL using the following command:
    ```
    #sudo yum module disable postgresql
    ```
    ii. Create a repo file named, for example, `fsr-pg.repo` in `/etc/yum.repos.d/`
    iii. Enter the following details in the repo file, i.e., in the `fsr-pg.repo` file, and then save the file:
    ```
    [fsr-pgdg]
    name=FortiSOAR PostgreSQL repository
    baseurl=https://repo.fortisoar.fortinet.com/7.6.2/third-party/postgres/pgdg16/
    ```

```
enabled=1
gpgcheck=1
gpgkey=https://repo.fortisoar.fortinet.com/7.6.2/third-party/postgres/RPM-GPG-KEY-PGDG-16
```

iv. Install PostgreSQL 16 using the following command:
`#sudo yum install postgresql16-server postgresql16`

- Ensure that you when install and configure your PostgreSQL server, you also install the following package:
`postgresql<<version>>-contrib-<postgresql_version>-1PGDG.rhel9.x86_64`
NOTE: The rpm versions of your 'postgresql-server' and 'postgresql-contrib' must be the same.
Example of rpms:
postgresql-server: `postgresql16-server-16-1PGDG.rhel9.x86_64`
`postgresql-contrib: postgresql16-contrib-16.0-1PGDG.rhel9.x86_64`

- Create the `postgresql.conf` files in the `/var/lib/pgsql/16` directory using the following command:
`/usr/pgsql-16/bin/postgresql-16-setup initdb`
- Prepare your Local FortiSOAR instance:
  - Ensure that port 5432 is opened for PostgreSQL to allow inbound and outbound communication with the remote instance.
- Ensure that the connectivity between your FortiSOAR local instance and remote PostgreSQL instance is established.
- If the FortiSOAR instance was connected previously to the same instance of the database that is being externalized, it could lead to a stale connection being present to the FortiSOAR database on the external PostgreSQL server, and then the `csadm db --externalize` command could fail with a "Failed to drop database `<name of database>`" error. To resolve this issue and release all stale connections, restart the postgres service using the following command:
`systemctl start postgresql-<postgresql_version>`
For example, `systemctl start postgresql-16`
- Ensure that you have stopped all your schedules and that you have no playbooks in the running state.

Ensure that you have enough disk space available to perform DB externalization tasks. It is recommended that you have available disk space of around 3X of the data size, for example, if your data size is 2GB, then you should have around 6GB of available disk space, to ensure that the processes do not stop or fail.

Externalizing FortiSOAR databases

1. On the externalized PostgreSQL database run the following commands:
   a. To ensure that the PostgreSQL server allows connections, open the firewall port:
   ```
   # firewall-cmd --add-service=postgresql --permanent
   # firewall-cmd --reload
   ```
   b. To ensure that the `pg_hba.conf` file, trusts the FortiSOAR server for incoming connections:
   Add the following entries to the file `/var/lib/pgsql/16/data/pg_hba.conf` file:
   ```
   local all all md5
   host all all ip/subnetmask md5
   ```
   For example, if the ip/subnetmask of your FortiSOAR instance is xxx.xxx.xxx.xxx/xx then add the

following to the `pg_hba.conf` file:
```
local all all md5
host all all xxx.xxx.xxx.xxx/xx md5
```

c. To ensure that the `postgresql.conf` file, trusts the FortiSOAR server for incoming connections:
Make the following changes to the `/var/lib/pgsql/16/data/postgresql.conf` file:
```
listen_addresses = '*'
port = 5432
```

d. Restart PostgreSQL using the following command:
```
systemctl start postgresql-<postgresql version>
```

e. Create a `cyberpgsql` user using the following command:
```
sudo -u postgres psql -U postgres -c "CREATE USER cyberpgsql WITH SUPERUSER PASSWORD 'changeme'";
```
Important: The password of the `cyberpgsql` must not contain the following characters: ' ^$.

f. Enable the `hstore` and `uuid-osop` extensions (this is a *recommended* step) using the following commands:
```
sudo -u postgres psql -U postgres -d postgres -c "CREATE EXTENSION IF NOT EXISTS hstore;"
sudo -u postgres psql -U postgres -d postgres -c 'CREATE EXTENSION IF NOT EXISTS "uuid-ossp";'
```

2. SSH to your FortiSOAR VM and login as a *root* user.
3. Create the `db_external_config.yml` file at the following location `/opt/cyops/configs/database/`
Use the following command to create the `db_external_config.yml` file:
```
# cp /opt/cyops/configs/database/db_config.yml /opt/cyops/configs/database/db_external_config.yml
```
4. Update the newly created `db_external_config.yml` file for PostgreSQL as follows:
In the `postgres` section:
   a. Set the `pg_external` parameter to "true".
   This parameter determines whether or not the postgres database needs to be externalized. If it is set to "true", then the postgres database is externalized, and if set to "false" (default), then the postgres database is not externalized.
   b. Update the value of the postgres host (`pg_host`) and postgres port (`pg_port`) (if needed) parameters.
   Important: In the case of the Aurora database, in the `pg_host` parameter, specify the global database cluster identifier name. For more information on setting up a High Availability FortiSOAR cluster in AWS Cloud with Aurora as the external database, see the High Availability chapter.
   c. Add the encrypted password that you have set on your remote PostgreSQL server in the `pg_password` parameter.
   You can encrypt your PostgreSQL passwords by running the `csadm db --encrypt` command as a *root* user. For more information on `csadm`, see the FortiSOAR Admin CLI chapter.
5. Check the connectivity between the FortiSOAR local instance and remote PostgreSQL database using the `csadm db --check-connection` command.
6. To externalize the PostgreSQL database, type the following command:
```
# csadm db --externalize
```
Once you run the above command, you will be asked to provide the path in which you want to save your database backup file.
Note: If you run the `# csadm db --externalize` option more than once (i.e., you are running the option again after the first time), then `csadm` will display a message such as:
`The databases already exist in postgresql, do you want to delete these databases (y/n):` If you want to externalize your PostgreSQL database again you must type `y`.
7. After you have completed externalizing your PostgreSQL database, you should restart your schedules.

You can choose to externalize both the main PostgreSQL database and your archival database to the same external database or a different external database. However, if you have externalized your main PostgreSQL database, then you must externalize your archival database, i.e., the external database for data archival cannot be set to 'localhost'.
If you are upgrading your FortiSOAR instance from a release prior to 7.2.0 to a 7.2.0 or later release, then ensure that your externalized PostgreSQL database does not contain a database with the name 'data archival', else there might be conflicts with the data archival feature.
For more information, see the "Data Archival" topic in the System Configuration chapter.

You must keep the `max_connections` value for your external PostgreSQL to 300. By default, the `max_connections` value is 300. You can set the `max_connections` value in the `postgresql.conf` file, which is generally located at `/var/lib/pgsql/16/data/postgresql.conf`.

Setting up an externalized database on the cloud

If you want to set up your externalized database on the cloud, for example AWS, do the following:

1. Set up your AWS RDS with its default port, i.e., **5432** and **postgresql-<postgresql_version>**, for example, **postgresql-16**
2. Setup the master password for the PostgreSQL user.
3. Ensure that your FortiSOAR system is able to access the RDS system using RDS endpoint, for which you might have to update security groups in AWS RDS. Check the connectivity between FortiSOAR and RDS systems using the `telnet` command and using the following psql command:
   `psql -h <pg_hostname> -U <pg_username> -p <port_no> -l postgres`
4. Once the connectivity is verified, create the `db_external_config.yml` file at the following location:
   `/opt/cyops/configs/database/`
5. Navigate to `/opt/cyops/configs/database/` and then run the following command:
   `cp db_config.yml db_external_config.yml`
6. Connect to the RDS system to create a user:
   `psql -h <pg_hostname> -U <pg_username> -p <port_no> -d postgres`
7. Create the `cyberpsql` user in the RDS system using the following command:
   `CREATE USER cyberpgsql WITH PASSWORD '<your password>' CREATEROLE CREATEDB;`
8. Grant appropriate access to the `cyberpsql` user using the following command:
   `GRANT cyberpgsql TO postgres;`
9. Exit the psql shell.
10. Edit the `db_external_config.yml` file and update the `postgres` section as mentioned in the Externalizing FortiSOAR databases section.
    You need to encrypt the password using the following command:
    `csadm db --encrypt`

11. To externalize the PostgreSQL database, type the following command:
    `# csadm db --externalize`
    This command takes some time for completion.

Performance differences might be seen in cases where there is high network latency between the FortiSOAR and RDS systems.

Setting up an externalized database without data migration

Maintaining an externalized database with 100% uptime enables seamless recovery in case of unforeseen issues with the primary FortiSOAR instance, for example, ('A'). If the FortiSOAR instance 'A' becomes non-operational, the recovery process involves deploying a new FortiSOAR instance ('B') and connecting it directly to the same externalized database to ensure operational continuity.

Follow these steps to establish a direct connection between FortiSOAR instance ('B') and the existing database:

1. Deploy a new FortiSOAR instance ('B').
2. SSH to the FortiSOAR VM ('B') and log in as a *root* user.
3. Ensure that the connectivity between your new FortiSOAR instance 'B' and the remote PostgreSQL instance is established.
4. To ensure that the `pg_hba.conf` file on the externalized database instance, trusts the new FortiSOAR instance 'B' for incoming connections:
   a. Add the following entries to the `/var/lib/pgsql/16/data/pg_hba.conf` file:
      `host all all <new ip of 'B' instance>/subnetmask md5`
      For example, if the ip/subnetmask of your FortiSOAR instance is xxx.xxx.xxx.xxx/xx then add the following to the `pg_hba.conf` file:
      `host all all xxx.xxx.xxx.xxx/xx md5`
   b. Restart the PostgreSQL service on the externalized database instance using the following command:
      `systemctl restart postgresql-<postgresql version>`
5. Update the `db_config.yml` file located at `/opt/cyops/configs/database/` in the new FortiSOAR instance ('B') as follows:
   a. Set the `pg_external` parameter to "true" to externalize the postgres database.
   b. Update the value of the postgres host (`pg_host`) and postgres port (`pg_port`) (if needed) parameters.
   c. Add the encrypted password that you have set on your remote PostgreSQL server in the `pg_password` parameter.
6. To archive the database, edit the '`postgres_archival`' section in the `db_config.yml` file as follows:
   a. Set the `pg_archival_external` parameter to "true" to externalize the postgres archival database.
   b. Update the value of the data archival host (`pg_archival_host`) and data archival port (`pg_archival_port`) (if needed) parameters.
   c. Add the encrypted password that you have set on your remote data archival server in the `pg_archival_password` parameter.

7. After updating the `db_config.yml` file, run the following commands on FortiSOAR instance ('B') to apply the changes:
   a. `csadm services --restart`
   b. `systemctl restart php-fpm && sudo -u nginx php /opt/cyops-api/bin/console cache:clear && systemctl restart php-fpm`
   c. Copy the `.Defuse.key` located at `/opt/cyops/configs/cyops-api` from previous 'A' instance and paste it into the same location in the new FortiSOAR instance 'B'.
   d. `#sudo -u nginx php /opt/cyops-api/bin/console app:system:update -laf`
   e. Update the "`nodeId`" value in the `/opt/cyops-auth/utilities/das.ini` file on the new FortiSOAR instance 'B' using the following steps:
      i. Copy the "`nodeId`" of the faulted node using the following command:
      `# csadm ha list-nodes`
      ii. Open the `/opt/cyops-auth/utilities/das.ini` file on the 'B' instance, and in the `[CLUSTER]` section, replace the "`nodeid`" field value with previously copied "`nodeId`" value .Save the changes and close the `das.ini` file.
      iii. Restart the `cyops-auth service` using the following command:
      `#systemctl restart cyops-auth`
   f. Refresh the device UUID of the 'B' instance using the following command:
   `#csadm license --refresh-device-uuid`
   This command must display the "`Device UUID refreshed successfully.`" message.
   g. Open your FortiSOAR instance in a browser and click on "Continue to Login>" to access the FortiSOAR UI.
   h. Regenerate appliance and app keys using the following commands:
   ```
   # csadm certs --app-key
   # systemctl restart php-fpm && sudo -u nginx php /opt/cyops-api/bin/console cache:clear && systemctl restart php-fpm fsr-api-consumer
   # csadm services --restart
   ```
8. Stop and mask the PostgreSQL service on FortiSOAR instance ('B'):
   a. `systemctl stop postgresql-<postgresql_version>`
   b. `systemctl disable postgresql-<postgresql_version>`
   c. `systemctl mask postgresql-<postgresql_version>`

Any custom configuration changes made to the environment must be re-applied on the new instance.

Installing the pg_squeeze and pg_repack extensions

'`pg_squeeze`' and '`pg_repack`' are advanced PostgreSQL utilities used to optimize the process of reclaiming unused disk space. These utilities address issues such as database bloat and performance slowdowns in operations such as playbook searches, caused by extensive data retention and the accumulation of a large

number of records over time. This is often a result of PostgreSQL space quickly filling up when numerous playbooks are executed, even in 'Info' mode. The process reclaims disk space from PostgreSQL by optimally managing data.

Reference: https://github.com/cybertec-postgresql/pg_squeeze

For a cloud-managed database like AWS RDS, only enable the 'pg_squeeze' and 'pg_repack' extensions if they are supported on the database.
When externalizing your database, if you encounter an error like "permission denied to create extension 'pg_repack'," it indicates that the specified extension is not present in your database. In such cases, you can update the values of the 'USE_PG_SQUEEZE' and 'USE_PG_REPACK' keys in the /opt/cyops-workflow/sealab/sealab/config.ini file to 'FALSE'. For more information, see the *Debugging and Optimizing Playbooks* chapter in the "Playbooks Guide".

Installing pg_squeeze

Prerequisites

- Only 'Superusers' can use the utility.
- Check that postgres16 is already installed, if not then install the following:
  a. Check that the crb repository is enabled, if not then enable it using the following command:
  `sudo dnf config-manager --set-enabled crb`
  b. postgresql16
  c. postgresql16-server
  d. postgresql16-devel

Installation of the cybertec binary

1. Update 'PATH' to include PGSQL BIN:
   `export PATH=$PATH:/usr/pgsql-16/bin/`
2. Run the following commands:
   a. `wget https://repo.fortisoar.fortinet.com/7.6.2/third-party/postgres/pg_squeeze/pg_squeeze-master.zip`
   b. `unzip pg_squeeze-master.zip`
   c. `yum install redhat-rpm-config`
   d. `yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm`
   e. `yum install ccache`
   f. `cd pg_squeeze-master`
   g. Verify that "make" utility is installed, if not then install it using the following command:
   `yum install make`
   h. `make`
   i. `make install`

j. Create an extension in the psql shell by running the following command:
`CREATE EXTENSION pg_squeeze`

Configuration

1. Edit the `/var/lib/pgsql/<postgresql_version>/data/postgresql.conf` file, for example, the `/var/lib/pgsql/16/data/postgresql.conf` file and add the following entries:
```
wal_level = logical
max_replication_slots = 10 # ... or add 1 to the current value.
shared_preload_libraries = 'pg_squeeze' # ... or add the library to the existing ones.
```
2. Restart the PostgreSQL service using the following command:
`systemctl restart postgresql-16`

Tuning

You can add parameters for making 'auto vaccum' work more efficiently with or without `pg_squeeze`. Edit the `/var/lib/pgsql/<postgresql_version>/data/postgresql.conf` file, for example, the `/var/lib/pgsql/16/data/postgresql.conf` file and add the following parameters for 'auto vaccum' tuning:

- `autovacuum_work_mem = -1`
- `log_autovacuum_min_duration = -1`
- `autovacuum_max_workers = 3 - > 4`
- `autovacuum_naptime = 1min`
- `autovacuum_vacuum_threshold = 35`
- `autovacuum_analyze_threshold = 35`
- `autovacuum_vacuum_scale_factor = 0.1`
- `autovacuum_analyze_scale_factor = 0.1`
- `autovacuum_freeze_max_age = 200000000`
- `autovacuum_multixact_freeze_max_age = 400000000`
- `autovacuum_vacuum_cost_delay = 0.5`
- `autovacuum_vacuum_cost_limit = -1`
- `vacuum_cost_limit = 200`
- `maintenance_work_mem = 1920MB`

You can also change the timeout values for the `pg_squeeze` tasks by editing the default values of the following keys in the `/opt/cyops-workflow/sealab/sealab/config.ini` file:
```
SQUEEZE_SOFT_TASK_TIMEOUT: 3600
SQUEEZE_TASK_TIMEOUT: 3800
```

After editing the `config.ini` file and updating the values, restart the services by running the following command:
`systemctl restart uwsgi celeryd fsr-workflow celerybeatd:`

Installing pg_repack

Prerequisites

- Only 'Superusers' can use the utility.
- Check that postgres16 is already installed, if not then install the following:
  a. postgresql16
  b. postgresql16-server
  c. postgresql16-devel
- Check that `pg_conf` is already installed, if not then do the following:
  a. `dnf install postgresql16-devel`
  b. Add the following into the `PATH` variable:
  `export PATH="$PATH:/usr/pgsql-16/bin"`
- Check that the following required libraries are available, if not then install the following:
  - `dnf install libzstd-devel lz4-devel`
  - `dnf install readline-devel`
  - `sudo dnf install openssl-devel zlib-devel`

Installation

1. Download pg_repack:
   a. `wget https://api.pgxn.org/dist/pg_repack/1.5.0/pg_repack-1.5.0.zip`
   b. `unzip pg_repack-1.5.0.zip`
2. `cd pg_repack-1.5.0`
3. `make`
4. `make install`
5. Create an extension in the psql shell for the 'Sealab' database by running the following command:
   `CREATE EXTENSION pg_repack`

Troubleshooting

Unable to edit connector configuration on new externalized FortiSOAR instances

Users will be unable to edit connector configurations on a new FortiSOAR instance 'B', that uses an existing externalized database. This issue occurs because the externalized database retains entries for connectors that were installed on the previous instance (e,g, 'A'). As a result, when the new FortiSOAR instance ('B') connects to the same externalized database, the corresponding filesystem content for those connectors does not exist on instance 'B'.

Resolution

To resolve this issue, follow these steps:

1. SSH into the new FortiSOAR instance 'B' using 'root' access.
2. Download the `restore-connectors-widgets-from-externaldb.bin` utility using either `wget` or `curl`, to FortiSOAR instance 'B' from the following URL:
   `# wget https://repo.fortisoar.fortinet.com/fortisoar/scripts/restore-connectors-widgets-from-externaldb.bin`
3. Make the downloaded `restore-connectors-widgets-from-externaldb.bin` utility executable using the following command:
   `# chmod +x restore-connectors-widgets-from-externaldb.bin`
4. Execute the `tmux` command as it allows re-connection to terminal session and ensures that the script execution process is uninterrupted. If `tmux` is not installed on your instance, install it using the following command:
   `# yum install -y tmux`
5. Run the downloaded script using `tmux`:
   `# tmux`
   `# sh restore-connectors-widgets-from-externaldb.bin`
   OR
   `# ./restore-connectors-widgets-from-externaldb.bin`

Backing up and Restoring FortiSOAR

This chapter describes the process of backing up and restoring FortiSOAR, whether or not you have not externalized your PostgreSQL database. The FortiSOAR backup and restore process, is optimized and its performance is enhanced to make it more time-efficient and effective. The backup process now includes backups for files such as JWT keys, and the Elasticsearch database.

Data cannot be restored on any FortiSOAR instance with an externalized database. You can back up either the complete data or just the configuration files of an externalized FortiSOAR instance that is externalized. However, this backup can only be restored on an instance that does not have its database externalized.

Prerequisites

You must have the `root` or `sudo` permissions to perform backup and restore.

Ensure that you have enough disk space available to perform backup and restore tasks. It is recommended that you have available disk space of around 3X of the data size, for example, if your data size is 2GB, then you should have around 6GB of available disk space, to ensure that the processes do not stop or fail.

Backup Process

Use the FortiSOAR Admin CLI (`csadm`) `db` subcommand to regularly perform backups and restore, which restores the data seamlessly to a new FortiSOAR environment. To perform backup and restore, you must have *root access* on your FortiSOAR system. For more information on `csadm`, see the FortiSOAR Admin CLI chapter.

The FortiSOAR Admin CLI performs a full database backup of your FortiSOAR server each time. There is no provision of incremental backups.

The FortiSOAR version of the system on which the database restore will be performed must be the same as the version of the system from where the database backup is performed.

The FortiSOAR Admin CLI backs up the latest three backup files every time it creates a new backup. Any backups older than the latest three backups are deleted.

Data that is backed up during the backup process

The FortiSOAR Admin CLI backs up the following files, configurations, and data during the backup process:

- site-packages
- connectors
- playbooks
- solution packs
- application.conf
- db_config.yml
- pg_hba.conf
- JWT keys (from release 7.3.0 onwards) to ensure that manual input emails sent to users are still valid post restore
- Syslog forwarding configuration
- All major configuration files such as das.ini, postgresql.conf
- system configuration that is present in Settings > System Configuration
- Elasticsearch database backups as per requirements
  Note: From release 7.3.0 onwards, elasticsearch indexes are also backed up as part of the backup process. However, elasticsearch indexes require space equal to the current data present in the elasticsearch partition. Therefore, the elasticsearch indexes will be backed up only if there is sufficient space in your FortiSOAR system to accommodate the indexes; otherwise, the backup of elasticsearch indexes is skipped.
- PostgreSQL database backups as per requirements
- User-defined custom expressions

Backup of the configuration files are taken only in case of localized databases.

Prerequisites to running the backup process

You must have the NFS or local backup storage path.

Performing a backup

To perform a backup run the `csadm` command on any FortiSOAR machine using any terminal. A user who has `root` or `sudo` permissions can run the `csadm` command.

1. SSH to your FortiSOAR VM and login as a *root* user.
2. To perform a backup, type the following command:
   `# csadm db --backup [<backup_dir_path>]`
   `[<backup_dir_path>]` is the directory where backup files will be created. If you do not specify any path, then by default, the backup file is stored in the current working directory.
   IMPORTANT: If you have externalized your PostgreSQL database, it is recommended to use the `csadm db`

`--backup-config [<backup_dir_path>]` command for taking periodic backups of the configuration. Using the `--backup` argument is not recommended for an externalized PostgreSQL database.
Optionally, you can specify the `--exclude-workflow` option to exclude all the "Executed Playbook Logs" and the `--exclude-audit` option to exclude all the "Executed Audit Logs" from the backup. Executed playbook and audit logs are primarily meant for debugging so they are not a very critical component to be backed up. However, they constitute a major part of the database size, so excluding them from the backup reduces time and space needed for the backup. For example, to exclude all the "Executed Playbook Logs" from the backup, type the command as follows:
`# csadm db --backup [<backup_dir_path>] --exclude-workflow`
Important: FortiSOAR backs up the latest three backup files every time it creates a new backup. Any backups older than the latest three backups are deleted.

3. (Optional) If you want to backup only your configuration files, then type the following command:
`# csadm db --backup-config [<backup_dir_path>]`
Once you run the above command, you will be asked to provide the path of the configuration backup file. If you do not specify any path, then by default, the backup file is stored in the current working directory.

Running a backup as a scheduled job

Following is an example of running a backup as a scheduled cron job, on your FortiSOAR system or external Secure Message Exchange, that will run at 12:30 am every day. You can schedule the backup process based on your requirements.

Add the cron job to run at 12:30 am every day as follows:

```
$ sudo crontab –e
30 00 * * * csadm db --backup <backup_dir_path>
```

Once the backup process is successfully completed, the final `DR_BACKUP_<FortiSOAR_version>_timestamp.tgz` file is located in the directory where the backup files are created. It would be the same directory that you have specified when you ran the `csadm db --backup <backup_dir_path>` command. The `DR_BACKUP_<FortiSOAR_version>_timestamp.tgz` file includes the timestamp on when the backup is created.

The `DR_BACKUP_<FortiSOAR_version>_timestamp.tgz` file includes all the backup files. You can run the following command to check the contents of the `DR_BACKUP_<FortiSOAR_version>_timestamp.tgz` file :
`# tar -tvf <DR_BACKUP_<FortiSOAR_version>_timestamp.tgz>`

Restore Process

To restore the data on a new FortiSOAR server run the `csadm` command on any FortiSOAR machine using any terminal. A user who has `root` or `sudo` permissions can run the `csadm`.

It is recommended to use the home directory for the restore process. For the root user, this is typically /root; for the csadmin user, it is /home/csadmin. Alternatively, a dedicated directory can be used to stage the backup file. If the available space in the home directory is insufficient, users should extend the directory's storage capacity. Avoid using system directories (/etc, /bin, /usr, /var, or /lib) to store the backup file during the restore process.

Restoring data

1. Move the backup file to the new FortiSOAR VM.
2. SSH to the new FortiSOAR VM and login as a *root* user.
3. To restore the data, type the following command:
   `# csadm db --restore <backup_file_path>`
   `[<backup_file_path>]` is the directory where you have saved the backed up files. Note that the backup process, by default stores the backup in a locally saved file: `/home/csadmin/db_backup/DR_BACKUP_<yyyymmdd_hhmmss>.tgz`

When restoring a backup, it is recommended to maintain the same hostname as the original machine if you were using embedded Secure Message Exchange (SME) to connect to a dedicated tenant/agent. This will eliminate the need to reconfigure the agent or tenant for use. This will ensure that you do not require to reconfigure agent or tenant for use. If the hostnames are different, you might need to reconfigure agent or tenant connected using an embedded SME.

Post-Restore Steps

Once you have restored FortiSOAR, you need to perform the following steps:

- Re-deploy your existing license on the restored FortiSOAR instance. For information on deploying the license, see the *Licensing FortiSOAR* chapter in the "Deployment Guide."
- Update the value of the `Server_fqhn` global variable to the hostname of the machine on which the data is restored. This is necessary because, following a restore, the `Server_fqhn` global variable is set to the hostname of the system that was backed up, whereas it needs to be changed to the hostname of the machine that had been restored.

Notes for the Restore process

- If you have backed up a FortiSOAR instance that has a Secure Message Exchange enabled and is using a signed certificate, then you would need to re-apply the signed certificate on the new instance. For steps on how to replace certificates, see the *Replacing the self-signed certificates on the secure message exchange with signed certificates* topic in the "Multi-tenancy support in FortiSOAR Guide."
  If the restore is done on a machine that has a different FQHN, you will need to update the master FQHN at any tenant nodes or agents, which are connected through this secure message exchange. You can either update the tenant nodes or agents manually on the respective remote nodes or download the installer file again from the master and apply it on the remote nodes. For more information, see the Segmented Network support in FortiSOAR chapter.

Restoring a backup of an FortiSOAR instance with an externalized database to an instance that is not externalized

For information on database externalization, see the Externalization of your FortiSOAR PostgreSQL database chapter.

Before restoring the backup to the new FortiSOAR VM, do the following:

1. In order for the `pg_hba.conf` file to trust the new FortiSOAR server for incoming connections, add the following entries to the `/var/lib/pgsql/16/data/pg_hba.conf` file :
   `local all all md5`
   `host all all xxx.xxx.xxx.xxx/xx md5`
   where the ip/subnetmask of your new FortiSOAR instance is xxx.xxx.xxx.xxx/xx.
2. In order for the `postgresql.conf` file to trust the new FortiSOAR server for incoming connections, update the `/var/lib/pgsql/<postgresql version>/data/postgresql.conf` file, for example, the `/var/lib/pgsql/16/data/postgresql.conf` file as follows:
   `listen_addresses = '*'`
3. Restart PostgreSQL using the following command:
   `systemctl restart postgresql-<postgresql version>`

To restore the backup on the new FortiSOAR VM, perform the same steps as mentioned in the Restoring data topic.

If you have backed up only the configuration, using the `csadm db --backup-config` command, of your FortiSOAR instance that has an externalized database, then once you have restored FortiSOAR on another instance, you need to run the following commands to point FortiSOAR to the external database:

```
sudo -u nginx php /opt/cyops-api/bin/console cache:clear
systemctl restart php-fpm
```

Backup and Restore process for FortiSOAR High Availability systems

You can backup and restore your High Availability (HA) systems, using the following steps:

1. Configure a backup for your primary database that can be used to recover from any corruption or loss of your database. Steps for which are mentioned in the Backup Process section.
   IMPORTANT: Only take the backup of the primary node of your HA cluster.
2. Remove the secondary nodes from the HA cluster using the `csadm ha leave-cluster` command. For more information on the HA and the `csadm ha` CLI, see the High Availability and Disaster Recovery support in FortiSOAR chapter.
3. Restore the primary database. Steps for which are mentioned in the Restore Process section.
4. Add the secondary nodes back to the HA cluster using the `csadm ha join-cluster` command.

Regenerating RabbitMQ certificates when you are creating an HA cluster using a restored node

In addition to the above steps, if you are creating an HA cluster after you have taken a backup of an enterprise node, say T1 and restored this backup on another enterprise node, say T2, you also need to regenerate RabbitMQ certificates on the restored node. This is required as FortiSOAR restores the TLS certificate of T1 on the T2 node, causing hostname validation failure when you try to join another node to the T2 node in a HA cluster. Therefore, once you have restored your backup of the T1 node to the T2 node, then you must regenerate MQ certificates on the T2 node using the following command:
`csadm mq certs --generate`

Backup and Restore process for the External Secure Message Exchange systems

From version 7.0.0 onwards, you can also backup and restore the data of your external Secure Message Exchange (SME) system, by using the following commands:

- `csadm db --backup [<backup_dir_path>]`: Performs a backup of your external SME system. Steps for which are mentioned in the Backup Process section.
  Note: The `--exclude-workflow` and `--exclude-audtit` options are not applicable to the external SME.
- `csadm db --restore [<backup_file_path>`: Performs data restore for your external SME system from a locally stored file, whose path you have specified. The default location of the backup file is (`/home/csadmin/db_backup/DR_BACKUP_<yyyymmdd_hhmmss>.tgz`). Steps for which are mentioned in the Restore Process section.

In case of your configured MSSP setup, if you have taken a backup of your external SME instance 'A' and restored that backup on instance 'B', then in case of environments that use signed certificates you should not have any issues since the CLI copies over the certificates from the backed up to the restored node. However, if the certificates are the default self-signed certificates that are auto-generated by the platform, then these are not restored as they are tied to the hostname, and the hostname of the node 'B' might be different from node 'A'. So if you have a development environment with self-signed certs, after the SME is restored to the new node 'B', you have to perform different set of steps when the restored SME hostname is same and when it is different. Steps for both these scenarios follow:

Steps to be run on the external SME after it is restored when the external SME hostname is the same post restore:

1. Replace the certificate on the external SME with the certificate copied from `<Extracted Backup Folder Name>/secure_message_exchange/ssl to cp -rp ssl/ /opt/cyops/configs/rabbitmq/ssl`
2. Restart the services on the external SME using the following command: `csadm services --restart`
3. Restart the `cyops-postman` service on the tenant node.
4. Restart the `csagent` service on SNS Agents using the following command: `csagent --option services --action restart`

Steps to be run on the external SME after it is restored when the external SME hostname is changed post restore:

1. Update the certificate content on the master node's external SME. Path from where you need to take the certificate's content is `/opt/cyops/configs/rabbitmq/ssl/cyopsca/cacert.pem`
2. On the master node, export and download the configuration of the tenant node.
3. Import the same configuration (json) file on each tenant node to setup replication.

Troubleshooting Backup and Restore Issues

Post-restore the FSR agent status is not updated as "Remote Node Unreachable"

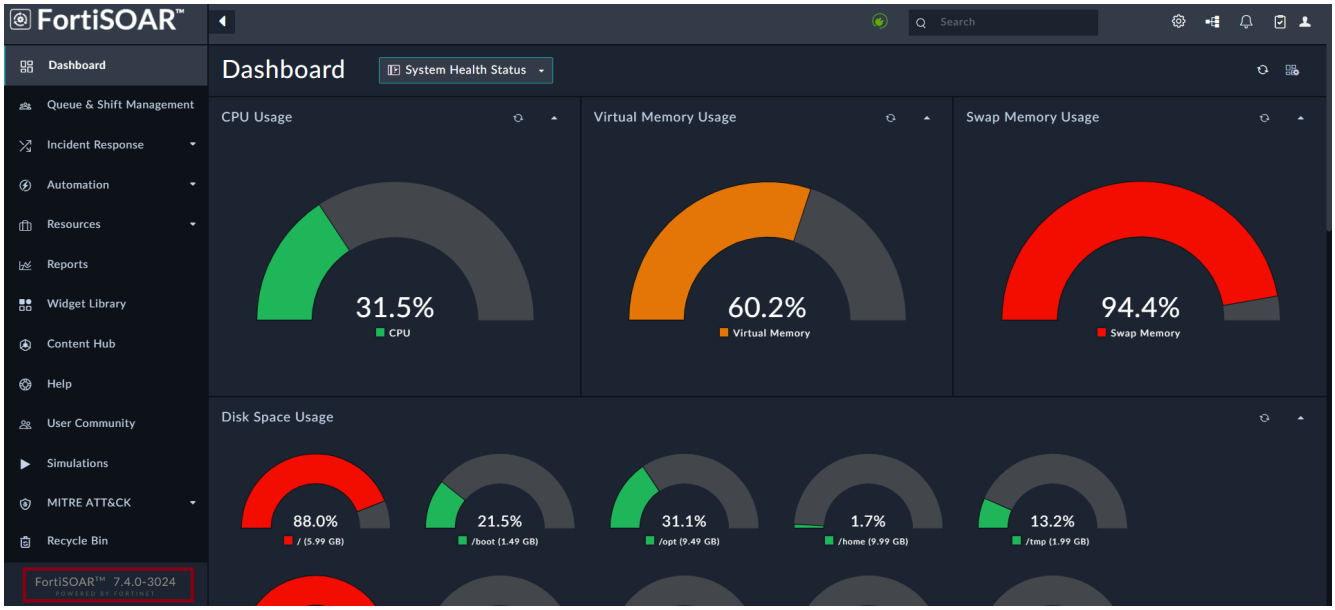
When you have restored FSR agents using a backup with agents and self-SME, the agent status does not get updated as "Remote Node Unreachable".

Resolution

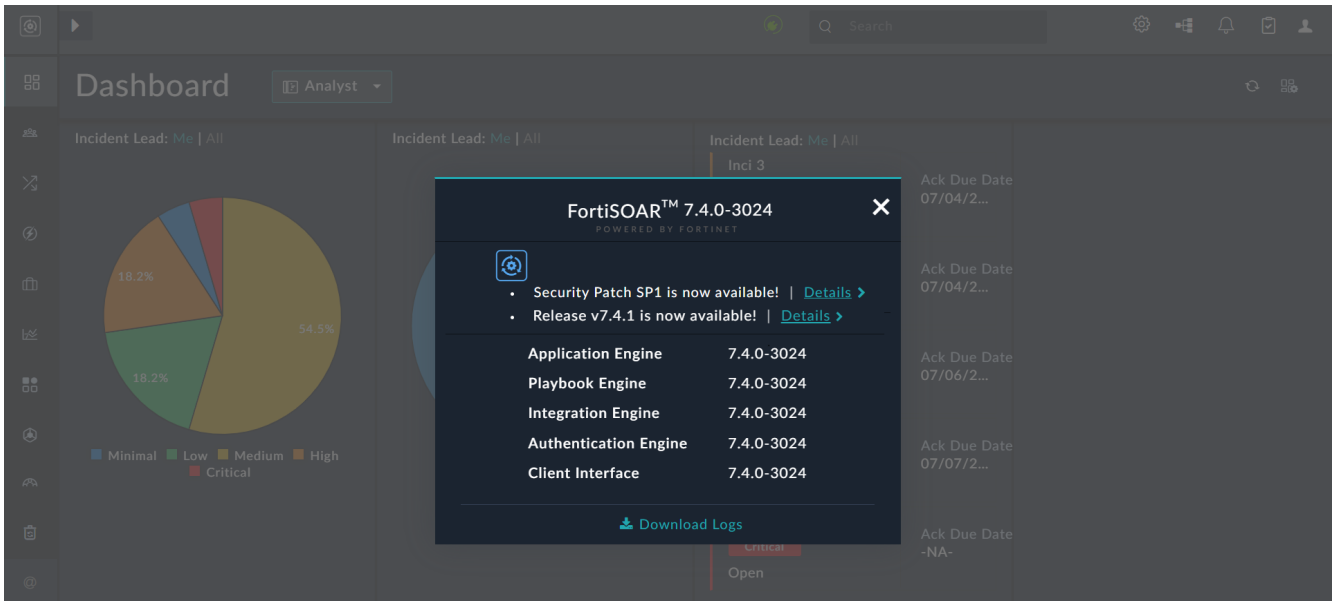
1. Navigate to the FSR agent and disable the FSR agent and then enable it again to reflect the correct connectivity status of the FSR agent, which is "Remote Node Unreachable".
2. Ensure that the restored node is resolvable from the FSR agent system.
3. Click the **Download Installer** link for each FSR agent with the **Include Pre-existing Connectors on Agent** checkbox selected.
4. Install the downloaded bin file, for example `<agent-name>-install.bin`, on the FSR agent.
5. Restart the `cyops-integrations-agent service` on the FSR agent.
   Now check the FSR agent status; it should display as "Available".

About FortiSOAR

The left navigation panel contains a link that includes the version and build number of FortiSOAR that is installed in your environment. For example, in the following image, the version of FortiSOAR installed is 7.4.0, and the build number is 3024:



Clicking on the **FortiSOAR Version Number-Build Number** link displays the version information of the major components of FortiSOAR, which are: Application Engine, Playbook Engine, Integration Engine, Authentication Engine, and Client Interface.



You can use the information presented in the **Version** pop-up, in the following cases:

- If you require some issue resolution or feature enhancement, then you might need to know the exact version of FortiSOAR installed in your environment, since the fix or enhancement might vary based on the version.
- There can be instances where you require only a component, for example, the Client Interface, within FortiSOAR to be updated. In such cases, you might need to know the versions of all the components in your FortiSOAR system.

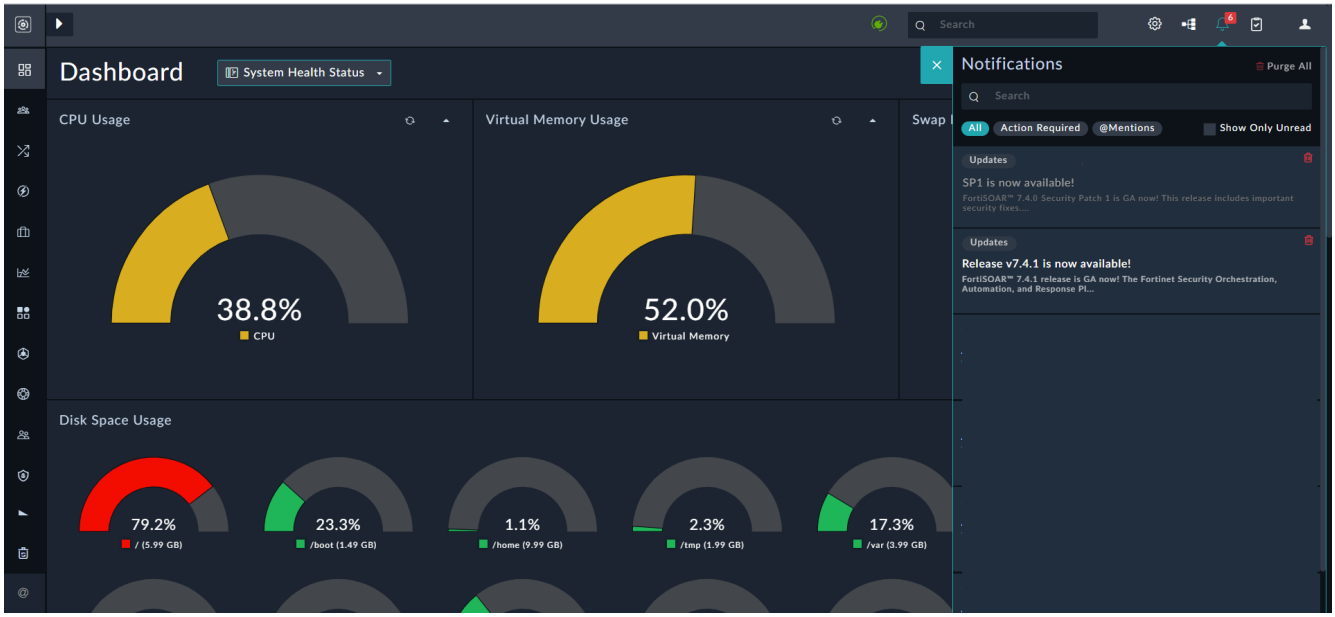
The 'Version' pop-up also displays notifications when a new Release (always the latest) and/or the latest Security Patch that is released for the version that is installed on your system is available. For example, if you have release 7.4.0 installed, then this dialog displays the following notifications:

- Availability of the latest release, which in the case of our example is 7.4.1.
- Availability of a new Security Patch on release 7.4.0, which in the case of our example is Security Patch 1. This notification always contains the latest Security Patch that is released for the version installed on your system, for example, if you had FortiSOAR release 7.2.2 installed on your system, then the notification would state Security Patch 7, as there are seven security patches released for FortiSOAR release 7.2.2.

The Release notification contains a **Details** link to that version's release notes so that users can get details about the latest available release. This keeps users informed about the latest releases and helps them in making informed decisions about upgrading to the latest available version. Similarly, the Security Patch notification also contains a **Details** link to the security patch documentation, using which users can get details about the latest security patch and make informed decisions on applying the security patch to their systems.

The FortiSOAR login page displays the number of the latest security patch (SP), if any SP is available for the FortiSOAR release installed on your system. For example, if you have FortiSOAR release 7.4.0 installed on your system, which has SP1 available, then the login page displays 7.4.0-3024 (SP1).

You can also view similar notifications when you click the **Notifications** icon on the top-right corner of the FortiSOAR screen. To view the upgrade notification, click the **Notifications** icon and look for the **Updates** tag:

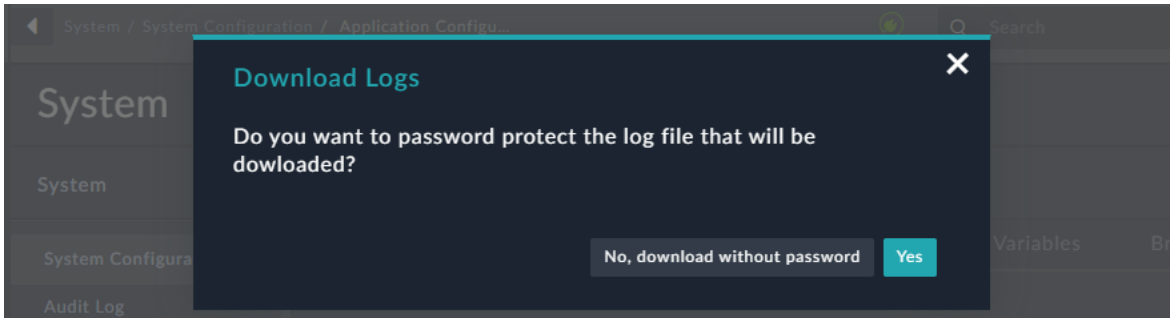


To view the version's release notes, click the release notification row, and to view the security patch documentation, click the security patch notification row. To dismiss the notifications, click the **Delete** icon in the notification row.

Downloading FortiSOAR logs

The 'Version' pop-up also displays a **Download Logs** link using which you can collect logs directly from UI. Application logs are important and are often required to troubleshoot issues, and during upgrade and installation operations. Prior to release 7.0.0, log collection was only possible using CLI commands, and there could be some SOC environments where SSH access to systems are very restricted and required to go through various approvals. Therefore, in such cases, collecting logs would become a tedious task. To ease the process of log collection, you can directly collect logs from the Version pop-up and share them with support team for further troubleshooting.

Clicking the **Download Logs** link displays a **Download Logs** dialog that provides you with an option to either download the log files without a password or to password-protect the downloaded log files.



By default, the **Yes** option is selected, i.e., you must add a password to protect the downloaded log files, so that the log files get an added security and can be opened only by users who have the password and not by everyone who has access to the system. Clicking **Yes** opens the **Download logs with password** dialog where you can enter the password for the log files and then click **Download**. If you click **No, download without password**, then the process of collecting and downloading the logs starts immediately.

The following log files are downloaded:

```
/var/log/cyops
/var/log/nginx
/var/log/elasticsearch
/var/log/messages*
/var/log/audit
/var/log/rabbitmq
/var/log/php-fpm
```

Monitoring FortiSOAR

Administrators can monitor various important aspects of their FortiSOAR system such as uptime (availability) of FortiSOAR, monitoring databases, services, disk space utilization, CPU and Memory utilization, etc.

The "System Health Status" Dashboard, monitoring playbooks, and the High Availability (HA) notifications that FortiSOAR sends already monitor various elements of the FortiSOAR system and also send appropriate notifications to users.

You can also use the `csadm ha show-health` command to get the heath information for FortiSOAR nodes. For more information, see the High Availability and Disaster Recovery support in FortiSOAR chapter.

This chapter intends to provide more details on what are the various aspects that can be monitored in case you want to fine tune the monitoring and/or setup monitoring using custom tools.

You can set up the system monitoring and purging of audit and playbook logs as part of your initial deployment and configuration process. For more information, see the `Setting up monitoring for your FortiSOAR system` topic in the *Additional configuration settings for FortiSOAR* chapter of the "Deployment Guide."

You can also set up system monitoring for FortiSOAR, both in case of a single node system and High Availability (HA) clusters. You can also receive monitor the system and get email notifications for failures of any FortiSOAR service, or if any monitored thresholds exceed the set threshold. In case of HA clusters, in addition, you can also monitor and get notified in case of heartbeat failures and high replication lags between nodes of your HA cluster. For more information, see the `Configuring System and Cluster Health Monitoring` topic in the System Configuration chapter.

For the list of logs that you can use for troubleshooting FortiSOAR, see the Debugging, Troubleshooting, and optimizing FortiSOAR chapter.

For information on monitoring the secure message exchange, see the *Monitoring the connectivity of the different nodes at the secure message exchange* topic in the "Multi-tenancy support in FortiSOAR Guide."

Benefits of monitoring

Implementing effective application monitoring offers the following benefits:

- Increased server, services, and application availability.
- Faster detection of network outages and protocol failures.
- Faster detection of failed services, processes, and scheduled jobs.

Manually setting up monitoring for each FortiSOAR component

To monitor various components of your FortiSOAR system, you need to SSH to your FortiSOAR VM and login as a user who has `root` or `sudo` permissions and then run the commands mentioned in the following sections.

Monitoring uptime of FortiSOAR

To monitor availability of FortiSOAR, run the `uptime` command.

You can also get the latest health check details for your FortiSOAR system. By default, the FortiSOAR health check runs every 5 minutes. For more information, see the System Configuration chapter.

Monitoring FortiSOAR services

To know the status of all FortiSOAR services run the `# csadm services --status` command.

To view the status of individual FortiSOAR services, run the `# systemctl status <service_name>` command. For example, to see the status of the `nginx` service, use the `# systemctl status nginx` command.

When you run `# csadm services --status` command the status of FortiSOAR services are displayed with a background color so that you can quickly and easily identify which services are running and which are not running. The status of services that are running are displayed in a Green background, and the status of services that are not running are displayed in a Red background.

The following image displays how the statuses of FortiSOAR services are displayed when some services are running, and some are not running, including information about how long the services have been active. Knowing the last active time of a service can assist with troubleshooting when a service is restarting repeatedly due to an issue:

```
[root@plain-host-rockylinux csadmin]# csadm services --status
rabbitmq-server..........[Running]      since Wed 2023-05-03 13:49:31 IST
elasticsearch............[Running]      since Wed 2023-05-03 13:49:38 IST
postgresql-14............[Running]      since Wed 2023-05-03 13:49:26 IST
nginx....................[Not Running]  since Wed 2023-05-03 14:01:09 IST
cyops-auth...............[Running]      since Wed 2023-05-03 13:49:38 IST
php-fpm..................[Running]      since Wed 2023-05-03 13:49:26 IST
uwsgi....................[Running]      since Wed 2023-05-03 13:50:18 IST
celeryd..................[Not Running]  since Wed 2023-05-03 14:01:35 IST
celerybeatd..............[Running]      since Wed 2023-05-03 13:50:07 IST
cyops-tomcat.............[Running]      since Wed 2023-05-03 13:49:38 IST
cyops-search.............[Running]      since Wed 2023-05-03 13:49:38 IST
fsr-workflow.............[Running]      since Wed 2023-05-03 13:50:28 IST
fsr-api-consumer.........[Running]      since Wed 2023-05-03 13:49:38 IST
cyops-ha.................[Running]      since Wed 2023-05-03 13:49:41 IST
```

You can also use the "System Monitoring" widget to monitor various FortiSOAR system resources such as CPU, Disk Space and memory utilization and status of various FortiSOAR services. FortiSOAR includes a default system monitoring dashboard, the "System Health Status" Dashboard, that displays the usage and health of various components in your FortiSOAR system. For more information on the System Monitoring widget, see the *Dashboards, Templates, and Widgets* chapter in the "User Guide."

Monitoring databases

To know the status of your PostgreSQL database run the `systemctl status postgresql-$(psql --version |egrep -o '[0-9]{1,}\.'| cut -d'.' -f1) -l` command.

To know the status of your Elasticsearch database run the `systemctl status elasticsearch` command.

Monitoring Disk Space Utilization

```
$ sudo df -H | grep -vE '^Filesystem|tmpfs|cdrom' | awk '{ print $5 " " $1 }'
43% /dev/mapper/vgos-root
13% /dev/sda1
2% /dev/mapper/vgos-tmp
10% /dev/mapper/vgos-var
1% /dev/mapper/vgos-rabbitmq
1% /dev/mapper/vgos-home
1% /dev/mapper/vgdata-relations
1% /dev/mapper/vgsearch-search
3% /dev/mapper/vgos-log
1% /dev/mapper/vgcoredump-coredump
```

```
                    7% /dev/mapper/vgos-audit
                    22% /dev/mapper/vgapp-csapps
            $
```

You can also use the System Monitoring widget and the "System Health Status" Dashboard to monitor the disk space utilization.

Monitoring CPU and Memory Utilization

```
# Top 50 process with memory and CPU usage
       $sudo ps -eo pid,cmd,%mem,%cpu --sort=-%mem | head -50
       # RAM and Swap memory usage command
       free -m
```

You can also use the System Monitoring widget and the "System Health Status" Dashboard to monitor the CPU and memory utilization.

Monitoring connectors

Use the "Connector Health" widget to track the health of all the configurations of all your configured connectors. You can view the status of your connector configurations using the "System Health Status" Dashboard.

You can also retrieve the health status of any connector configuration using the following API call: `POST /api/integration/connectors/healthcheck/<name>/<version>/?config=<config_id>`
For example, `POST /api/integration/connectors/healthcheck/smtp/2.3.3/?config=88c3d39c-2fa9-4731-b00d-29815008f17c`

Following are additional APIs around connectors and configurations:

- `POST /api/integration/connectors`: Use this API to list all connectors installed on a FortiSOAR instance.
- `POST /api/integration/connectors/<name>/<version>`: Use this API to list all configurations for a connector.

For information on authenticating and invoking FortiSOAR APIs, see the "API Guide."

Monitoring workflows

To know the number of workflows that are queued and not yet picked up for execution use the following command:

```
rabbitmqctl list_queues -p fsr-cluster --no-table-headers --silent | grep -E "^\s*celery\s+" | awk '{print $2}'
```

The number returned by this command should be 0, or at the maximum within two digits. If this number remains high for long, it means that the workflow engine is not able to cope up with the requests and requires to be tuned or you need to scale horizontally. From version 7.0.0 onwards, you can set the threshold for the workflow queue (default is 100) and if the threshold set is reached or crossed for this parameter, an email notification is sent to the specified email addresses, see the System Configuration chapter for more information. For information on tuning workflows, see the Debugging, Troubleshooting, and optimizing FortiSOAR chapter and *Debugging and Optimizing Playbooks* in the "Playbooks Guide."

FortiSOAR integration with FortiMonitor

From release 7.2.0 onwards, FortiSOAR has been integrated with FortiMonitor to enable monitoring including CPU, RAM, Disk monitoring, network card bandwidth, Nginx, PostgreSQL monitoring etc., of your FortiSOAR instances using FortiMonitor. FortiMonitor is used for managing large-scale infrastructure monitoring from single pane of glass, regardless of infrastructure deployment, and it empowers monitoring, incident management, and automated remediation.

Setting up a FortiSOAR instance to be monitored using FortiMonitor

To monitor FortiSOAR using FortiMonitor, you need to install an agent on the instance that needs to be monitored.

You can set up monitoring of an external secure message exchange instance by FortiMonitor; however, monitoring in the case of an air gapped environment is not supported.

If you are existing FortiMonitor customer, then log onto your FortiMonitor instance and click **Monitoring** > **Monitoring Policies**. Next, click **Default Monitoring** and select the FortiSOAR checkbox in the drawer that appears before installing the FortiMonitor agent.

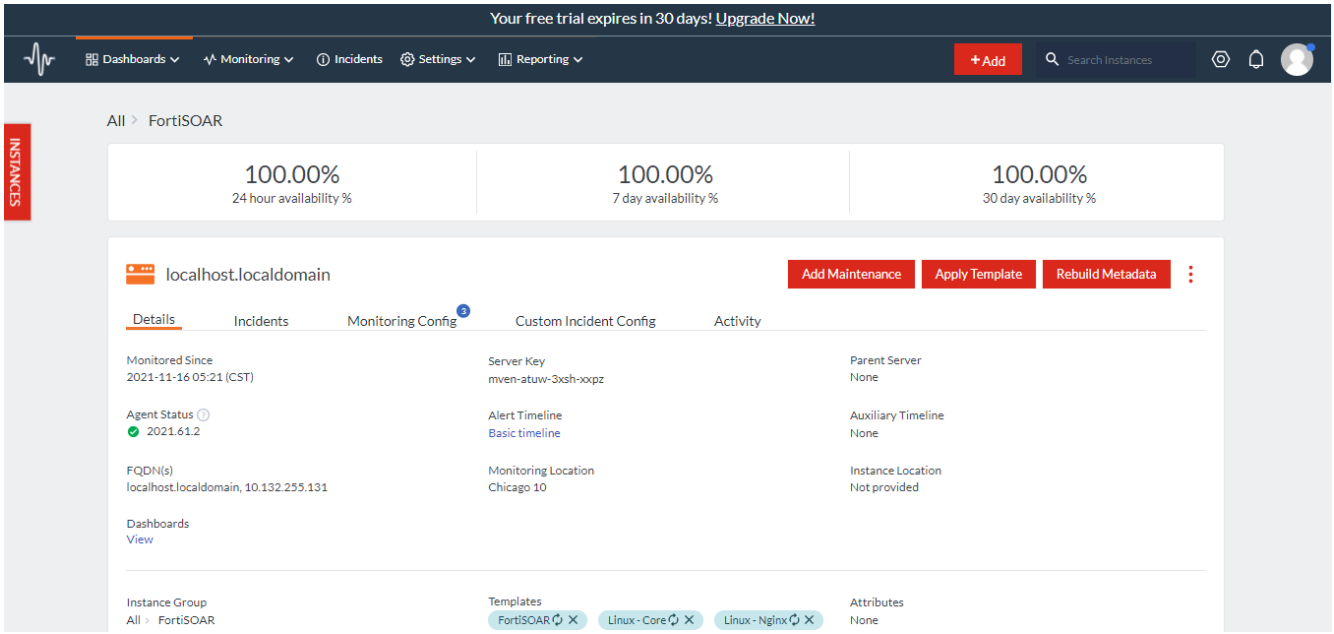
Steps to install the agent is as follows:

1. You must have a FortiMonitor account. If you do not have a FortiMonitor account, you can create an account with a 30-day free trial on https://www.fortinet.com/offers/fortimonitor-free-trial.
2. Once you have created your account, you need to get your customer key. You can get your customer key by logging into FortiMonitor (https://fortimonitor.forticloud.com/login) using your credentials. Then, click on your profile and select the **My Account** option. In the `Edit Account` dialog, from the **Customer Key** field, copy your customer key.
3. Ensure that the following URLs are reachable from your instance:
   - `repo.fortimonitor.com`
   - `rx.us01.fortimonitor.com`
4. SSH to the FortiSOAR instance that you want to FortiMonitor to monitor and run the `csadm system fortimonitor agent install --customer-key <your customer key>` command.
   This installs the agent on the instance that requires to be monitored using FortiMonitor.
   For details on managing the FortiMonitor agent using `csadm`, see the FortiSOAR Admin CLI chapter.

If FortiMonitor (`fm-agent`) is set up on your FortiSOAR system before you upgrade to FortiSOAR release 7.3.0 or later from a release prior to 7.3.0, then post-upgrade, FortiMonitor is automatically reinstalled and configured irrespective of the method used for upgrading FortiSOAR. For more information on upgrading FortiSOAR, see the "Upgrade Guide."

Monitoring FortiSOAR using FortiMonitor

Log into FortiMonitor (https://fortimonitor.forticloud.com/login) and you will see the FortiSOAR dashboard as displayed in the following sample image:



Click the **Instances** button and then click **FortiSOAR** to view all the FortiSOAR instances that are being monitored by FortiMonitor.

Components that are monitored out-of-the box by FortiMonitor on FortiSOAR

- CPU (Usage in %)
- Disk (Usage in % for every logical volume and also for /boot partition)
- I/O (read and write requests/sec for disks)
- Network card bandwidth including loopback(lo) interface (kb/sec)
- RAM usage (%)
- NTP (Difference between NTP and machine in sec)

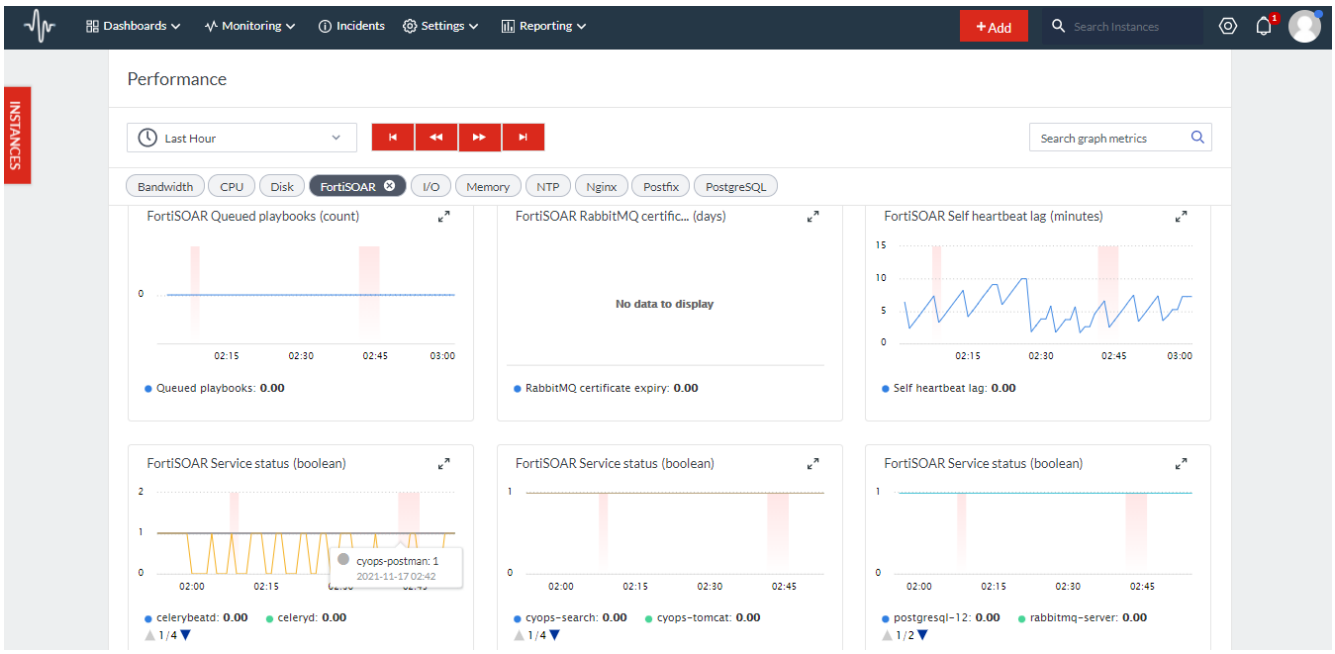
- Nginx (dropped connections, requests per second, handled connections, etc)
- PostgreSQL for every database (connectors, das, gateway, notifier, postman, sealab, venom) includes active connections, blocks read from disk(blocks/min), buffer cache hit rate(%), total transactions(tx/min), etc.
- Postfix (number of requests, postfix queue size)

For information on FortiMonitor, see the FortiMonitor Documentation.

FortiSOAR-specific components that are monitored by FortiMonitor

- Services status
- License Expiry
- Root Certificate Expiry
- Audit logs database size
- Cluster data replication lag (Applicable only for HA)
- Cluster heartbeat lag (Applicable only for HA)
- Connector health
- Nginx certificate expiry
- Playbook logs database size
- Primary database size
- Queued playbooks
- RabbitMQ certificate expiry
- Self heartbeat lag
- Tenancy data replication lag (Applicable only for MSSP)

To filter FortiSOAR metrics, click the **FortiSOAR** metric:



High Availability clusters are not monitored by default, and require some additional steps to be performed, which is mentioned in the following topic.

Enabling High Availability Cluster Monitoring

To monitor an HA cluster, you must perform the steps mentioned in the Setting up a FortiSOAR instance to be monitored using FortiMonitor topic for each node in the cluster. You also have to set up attributes for the nodes on FortiMonitor for the takeover operation as described in the following topic.

Performing Takeover

In an HA cluster all secondary nodes monitor the heartbeat of the primary node. If the primary node is down for more than 20 minutes, then FortiMonitor creates an incident for takeover.

If there is more than one secondary node in the HA cluster, then each of the nodes creates separate incidents asking for approval for this countermeasure (takeover). However, you should approve this countermeasure only for the node that you want to create as the new primary node of the cluster, and ignore the approval requests from the remaining secondary nodes.

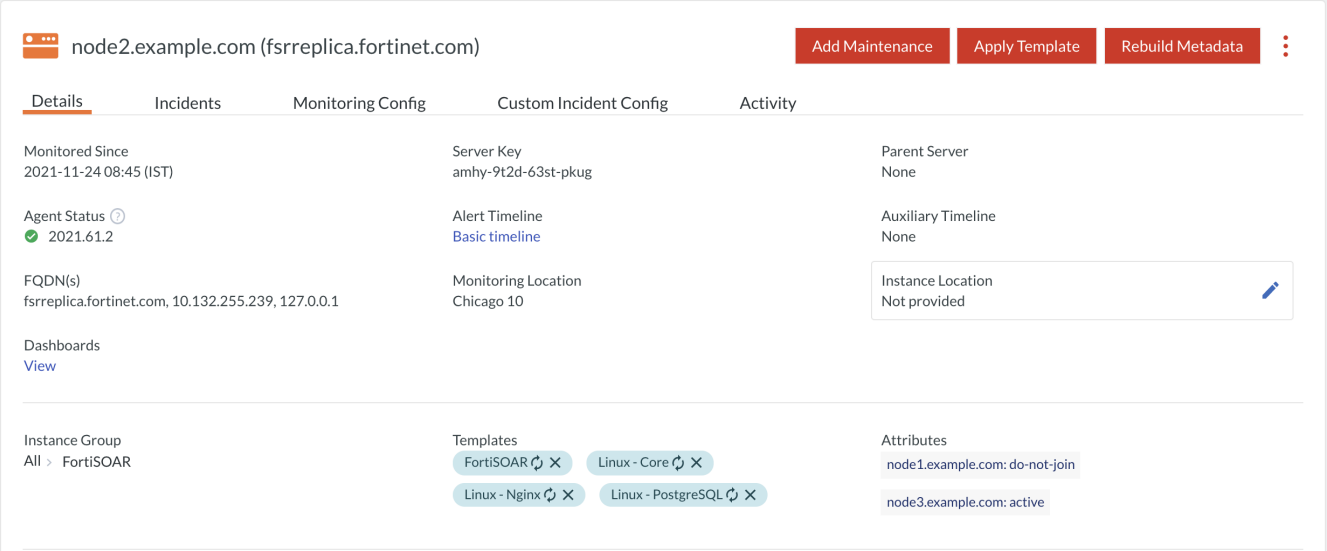
To perform takeover to make a secondary node as the primary cluster node, click the **Notifications** icon and view the active incidents. Click the incident from the secondary node that you want to create as the new primary node of the cluster, then click on the **countermeasure** icon and then **Approve** the takeover only for that node. Ignore the approval requests from the remaining secondary nodes.

By default, all other nodes join the new primary node as secondary passive nodes. However, you can choose to change this by adding attributes for the nodes on the server page of the new primary node.

Example of adding attributes on the server page of the new primary node:
node1.example.com --old primary
node2.example.com --new primary
node3.example.com --this will join the new primary node (node2) by default, as a secondary passive node. However, if we want to make node3 as active secondary, then on the server page of the new primary node (node2), in the `Instance Group` section, click **Attributes**. In the `Instance Configuration` dialog, in the `Attributes` section, click **Add Attribute** to add a Key and Value pair. For example, in the **Key** field, enter `node3.example.com`, and in the **Value** field, enter `active` and click **Save**, and then click **Save Changes**. In the **Value** field, you can add 'active' that means that the specified node will join the HA cluster as an "active secondary node", or 'do-not-join' that means that the specified node will not join the HA cluster.

It is recommended that attributes for the nodes that are part of an HA cluster should be set when the HA cluster is created.

An example screenshot of a FortiMonitor page containing a FortiSOAR group whose nodes that are part of an HA cluster have their attributes set:



Skipping monitoring of connectors

All the configurations of all the connectors that are configured on the FortiSOAR node (not agent nodes) are monitored. However, If you want to skip monitoring for a connector, then perform the following steps:

1. SSH to the FortiSOAR instance that is being monitored by FortiMonitor.
2. Open the `fm_agent.cfg` file:
   `vi /etc/fm-agent/fm_agent.cfg`.
3. In the `fm_agent.cfg` file, in the `[fortisoar]` section, add the following:
   `fortisoar_connector_exclusion_list = <comma-separated-connector-list>`
   For example, if you want to skip monitoring for the IMAP and SMTP connectors, add the following line to the `fm_agent.cfg` file:
   `fortisoar_connector_exclusion_list = smtp,imap`
4. Save and quit the `fm_agent.cfg` file.
5. Rebuilt the metadata for the FortiMonitor agent using the following command:
   `csadm system fortimonitor agent rebuild-metadata`
   Running this command, enables the connector monitoring changes to be updated in the agent, and the Connector Health metrics get immediately reflected.
   If you do not run this command, then the connector monitoring changes get reflected after one hour.

OnSight Collector

The OnSight collector is a lightweight virtual appliance that sits on your internal private network and allows you to securely monitor infrastructure, which is protected behind your firewall. The collector has many capabilities, including: network level probing (ICMP, jitter, packet loss, port level), uptime checks, synthetic monitoring checks, network device metric collection, and many other telemetry integrations.

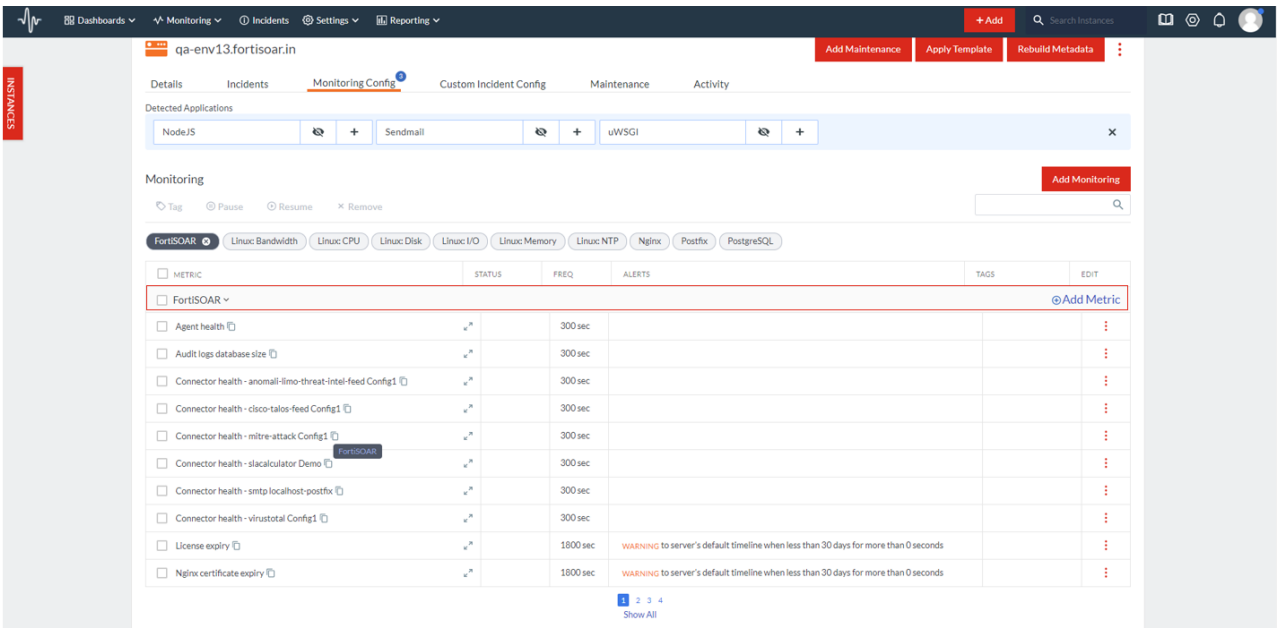
More information about the OnSight vCollector can be found in "FortiMonitor User Guide > OnSight vCollector" at https://docs.fortinet.com/fortimonitor.

Frequently Asked Questions

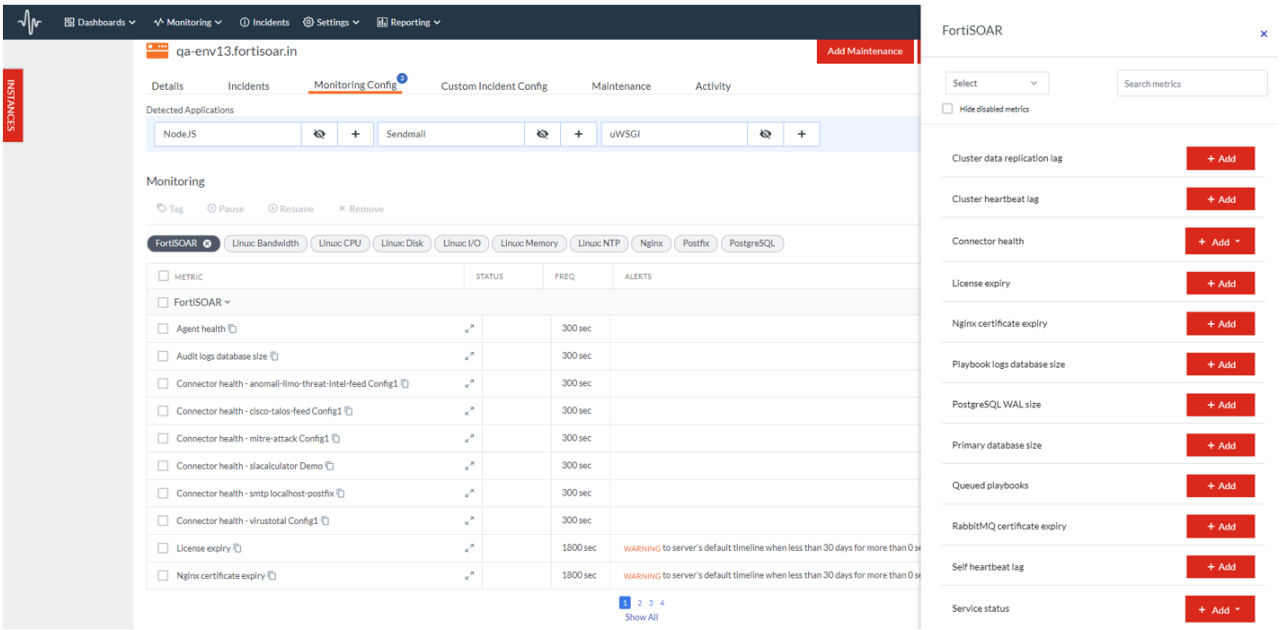
Q. How to add queued playbooks as a metric for monitoring using the FortiSOAR template

A. To add a queued playbook as a metric, do the following:

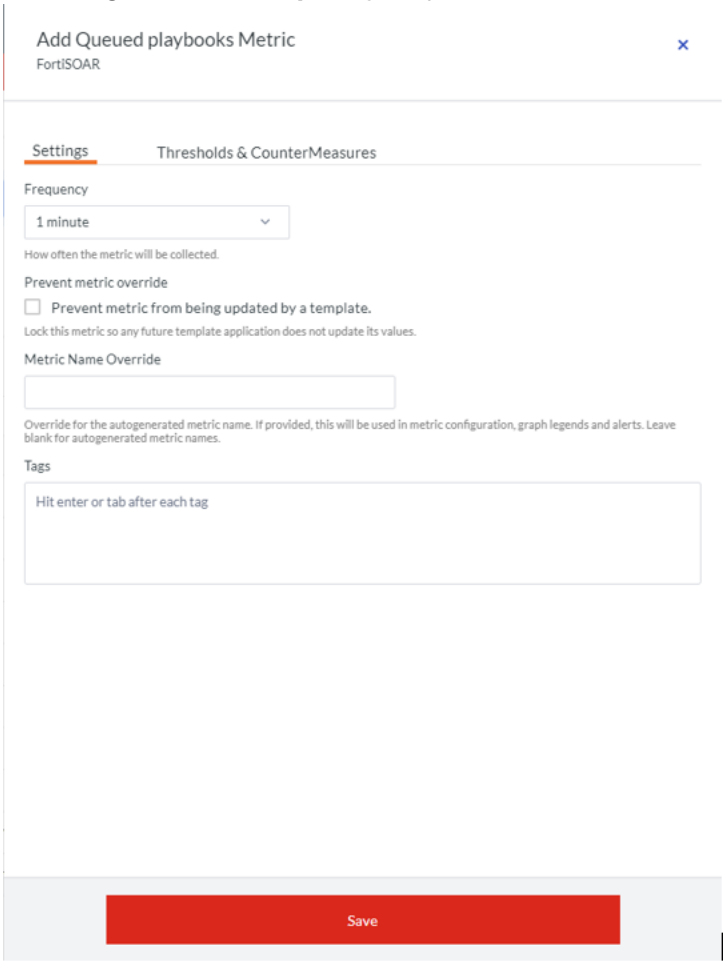
1. Log into FortiMonitor.
2. Navigate to the **Monitoring Config** tab.
3. Select the **FortiSOAR** tag.
4. Click on the **Add Metric** link:



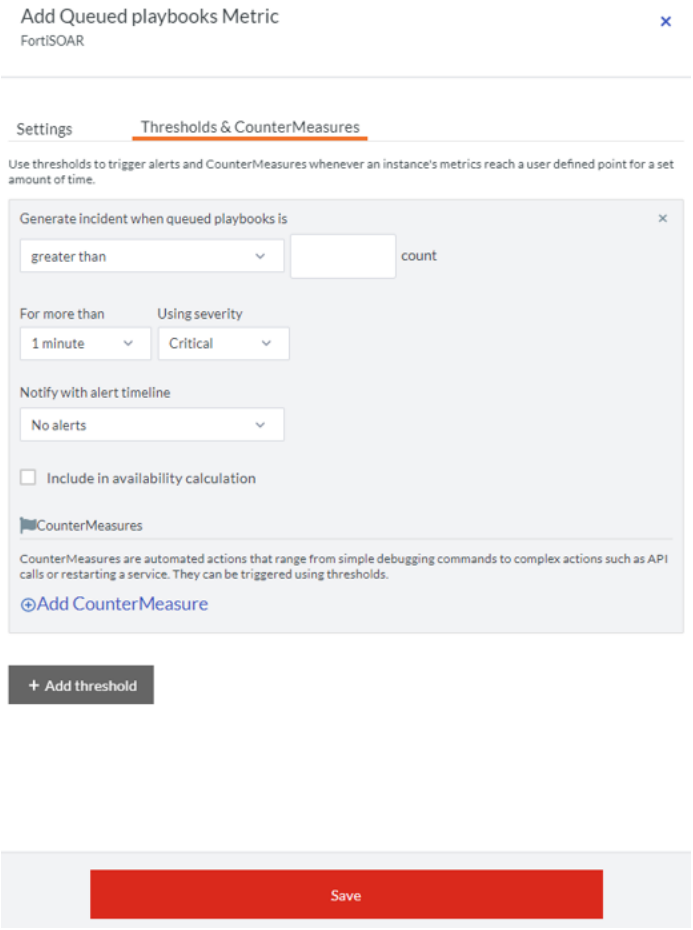
5. Select the **Queued playbooks** metric:



6. In **Settings** from the **Frequency** drop-down list select the frequency of monitoring the queued playbook:



7. Click the **Thresholds & CounterMeasures** tab and add the threshold counts as per your requirements:



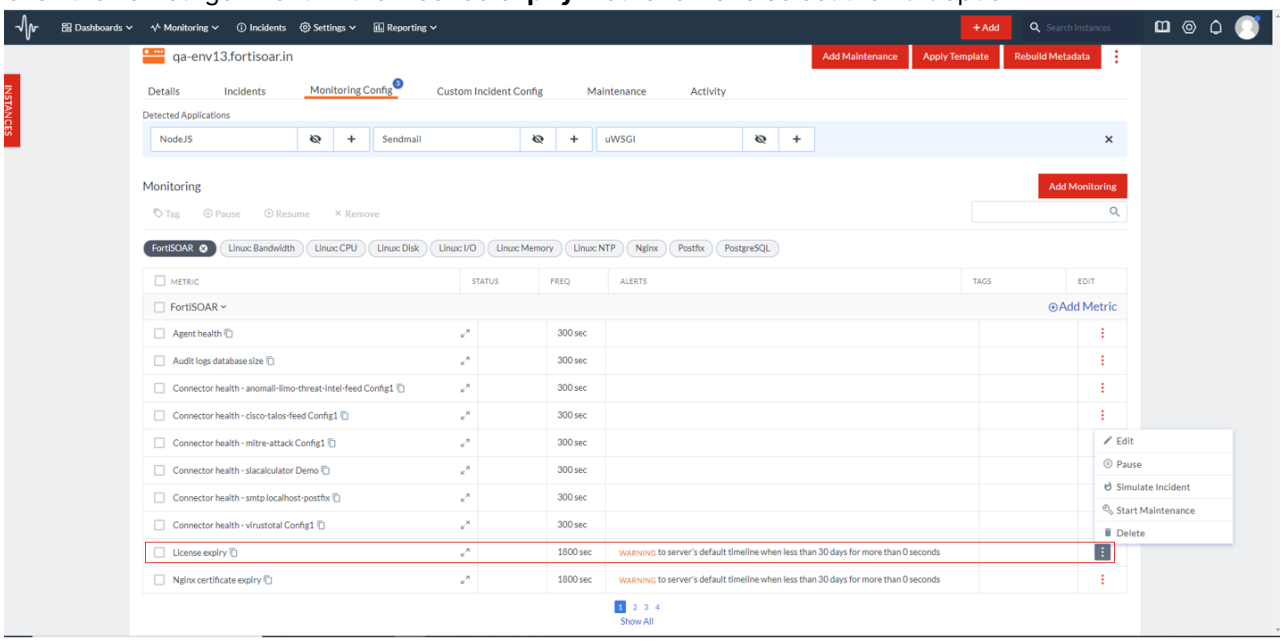
8. Click **Save** to save the queued playbooks as a metric for monitoring.

Q. How to change the monitoring interval for the License expiry metric

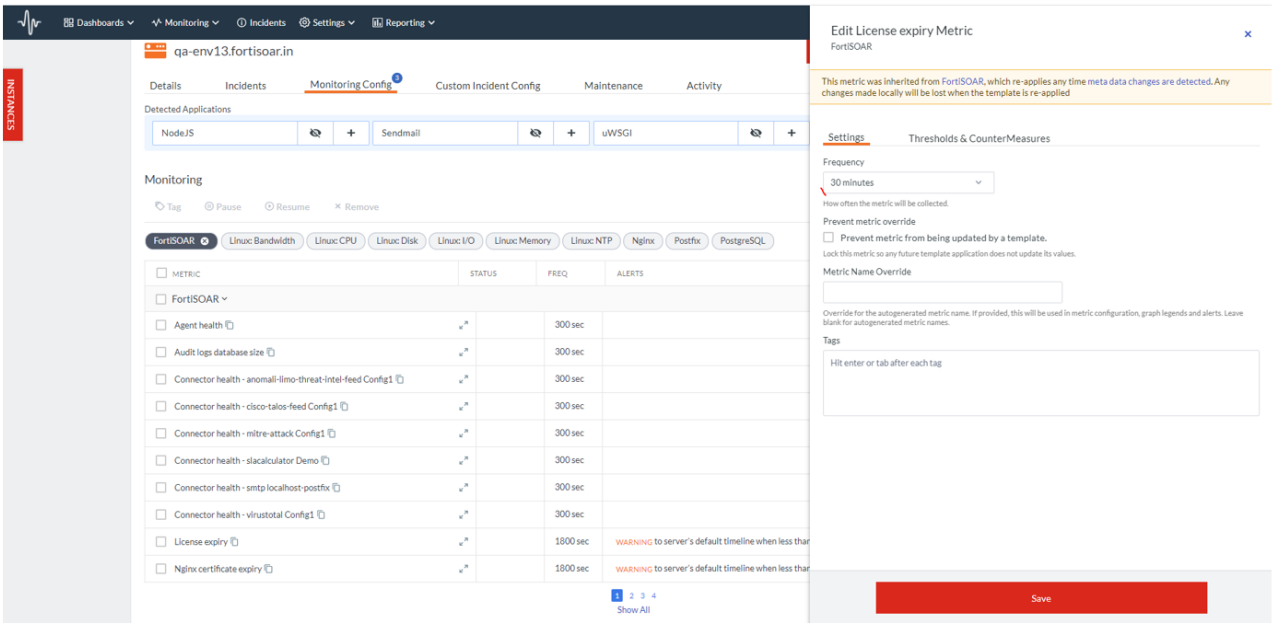
A. To change the monitoring interval for the License expiry metric, do the following:

1. Log into FortiMonitor.
2. Navigate to the **Monitoring Config** tab.
3. Select the **FortiSOAR** tag.

4. Click the hamburger menu in the **License expiry** metric row and select the **Edit** option:



5. In **Settings** from the **Frequency** drop-down list, select the frequency of monitoring the license expiry:

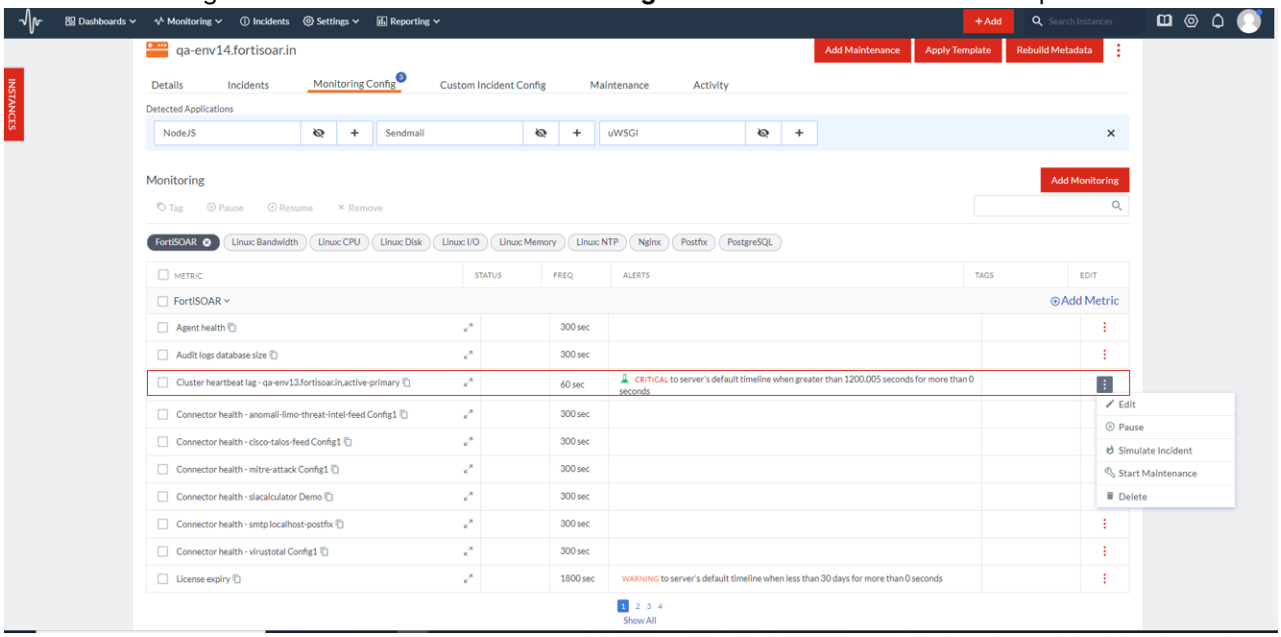


6. Click **Save** to update the monitoring interval for the License expiry metric.

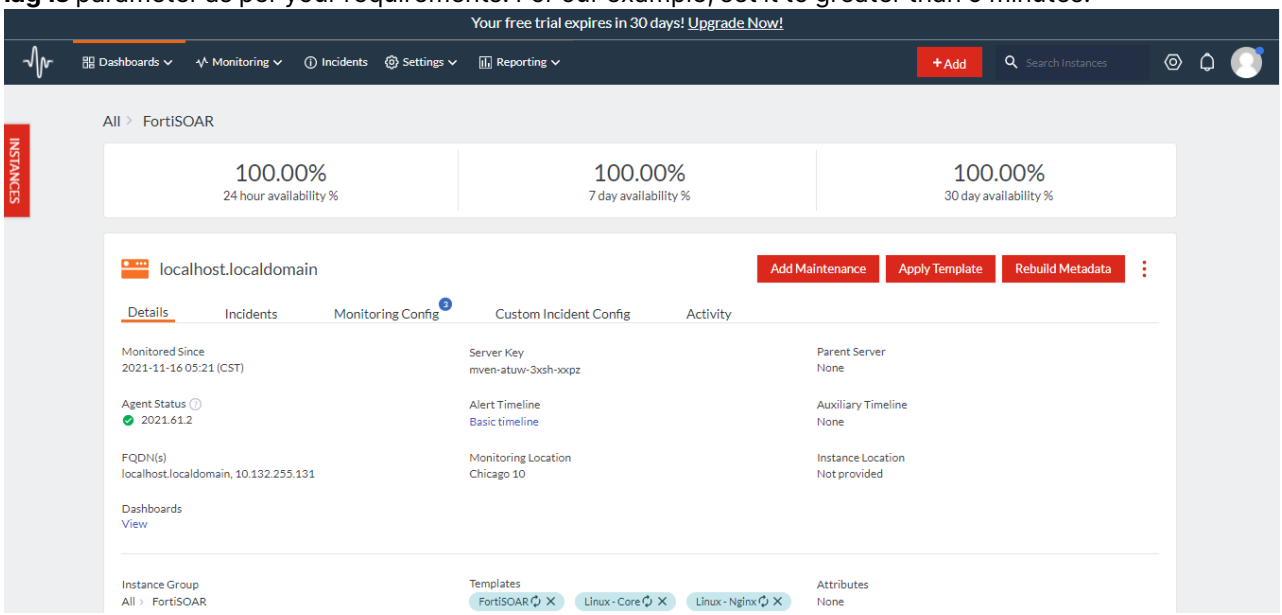
Q. How to modify the timings of generating a Takeover Incident

A. To change the timings of generating a Takeover incident from the default of 20 minutes too, for example, 5 minutes, i.e., to generate a Takeover if the cluster heartbeat lag is greater than 5 minutes, do the following:

1. Log into FortiMonitor.
2. Click on a secondary node.
3. Navigate to the **Monitoring Config** tab.
4. Select the **FortiSOAR** tag.
5. Click the hamburger menu in the **Cluster heartbeat lag** metric row and select the **Edit** option:



6. Click the **Thresholds & CounterMeasures** tab and modify the **Generate Incident when cluster heartbeat lag is** parameter as per your requirements. For our example, set it to greater than 5 minutes:



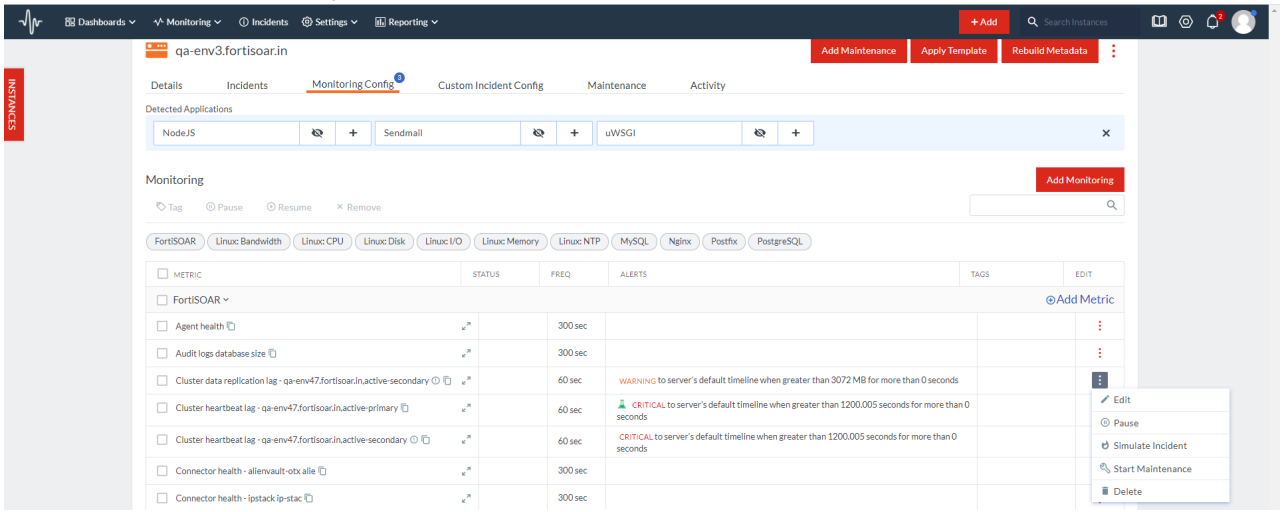
7. Click **Save** to update to the timing of generating a Takeover Incident.

Q. How to remove stale entries on FortiMonitor for FortiSOAR metrics after a takeover

A. Post-takeover, you might observe stale entries for FortiSOAR metrics on Fortimontior as FortiMonitor does not remove the stale entries on its own when the metadata is rebuilt. The stale metrics require to be removed manually as follows:

For the Primary Node:

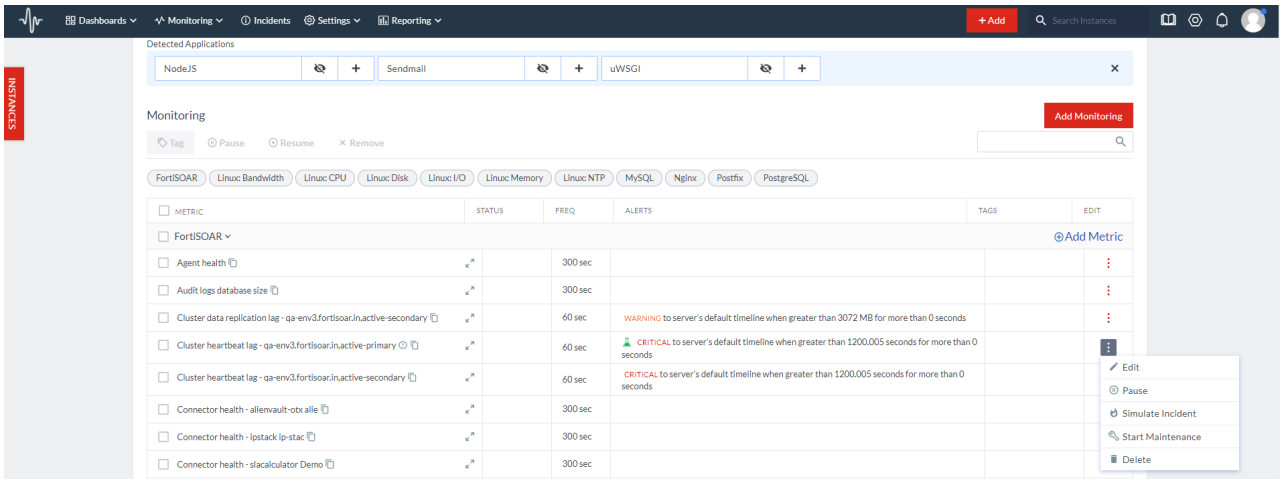
1. Log into FortiMonitor.
2. Click on a primary node.
3. Navigate to the **Monitoring Config** tab.
4. Select the **FortiSOAR** tag.
5. Click the hamburger menu in a FortiSOAR metric row, for example, the **Cluster data replication lag** metric row and select the **Delete** option:



This deletes the stale entry for this particular FortiSOAR metric. You can do the same for the other FortiSOAR metrics that display stale entries.

For the Secondary Node:

1. Log into FortiMonitor.
2. Click on a secondary node.
3. Navigate to the **Monitoring Config** tab.
4. Select the **FortiSOAR** tag.
5. Click the hamburger menu in a FortiSOAR metric row, for example, the **Cluster heartbeat lag** metric row and select the **Delete** option:



This deletes the stale entry for this particular FortiSOAR metric on this secondary node. You can do the same for the other FortiSOAR metrics that display stale entries on other secondary nodes.

Debugging, Troubleshooting, and Optimizing FortiSOAR

Administrators can use various logs that FortiSOAR generates to troubleshoot FortiSOAR issues. This chapter lists the key FortiSOAR services and processes and also provides some troubleshooting tips. This chapter also provides some additional configuration settings so that you can tune the results that get displayed by FortiSOAR for record similarity and field prediction. For more information on record similarity and field prediction, see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide."

If you face any issues while deploying or upgrading FortiSOAR, see the *Troubleshooting FortiSOAR* chapter in the "Deployment Guide." If you face deployment or upgrade failures due to insufficient space, or if you face issues while using FortiSOAR that might be caused due to insufficient space, like you are unable to log into FortiSOAR or FortiSOAR services stop working, then see the `Issues occurring in FortiSOAR due to insufficient space` section in the *Troubleshooting FortiSOAR* chapter in the "Deployment Guide."

For debugging and optimizing playbooks and workflows, see the *Debugging and Optimizing Playbooks* chapter in the "Playbooks Guide."

For information about monitoring various important aspects of your FortiSOAR system, see the Monitoring FortiSOAR chapter.

List of logs used for troubleshooting FortiSOAR

FortiSOAR log files are stored in the following location: `/var/log/cyops`. You will find the following directories in the `/var/log/cyops` location:

Log Name	Purpose
`cyops-api/ssl_cyops_api_access.log`	Used for troubleshooting web (nginx) UI or API access issues.
`cyops-api/ssl_cyops_api_error.log`	Used for troubleshooting API errors.
`cyops-api/prod.log`	Used for troubleshooting FortiSOAR PHP related issues.
`cyops-api/last_system_publish.log`	Used for troubleshooting publishing issues. It captures the output when a publish is fired from the UI after MMD changes.
`cyops-auth/das.log`	Used for troubleshooting FortiSOAR authentication issues.
`cyops-auth/fdn.log`	Used for troubleshooting FortiSOAR license synchronization issues with FortiGuard Distribution

	Network (FDN).
cyops-auth/ha.log	Used for troubleshooting FortiSOAR High Availability issues.
cyops-gateway/auditlog.log	Used for troubleshooting FortiSOAR audit log issues.
cyops-gateway/saml.log	Used for troubleshooting FortiSOAR SAML issues.
cyops-search/falcon.log	Used for troubleshooting FortiSOAR Search issues. If you get any error when you are indexing or searching for a record in FortiSOAR, you can use the `falcon.log` file to troubleshoot ElasticSearch issues. This log is also used for checking the status of ElasticSearch indexing.
cyops-gateway/gateway.log	Used for troubleshooting FortiSOAR Gateway issues such as, audit log page failing to load or the SSO configuration page failing to load.
cyops-notifier/notifier.log	Used for troubleshooting FortiSOAR Web Socket issues.
cyops-workflow/beat.log	Used for troubleshooting issues of the FortiSOAR Scheduler.
cyops-workflow/celeryd.log	Used for troubleshooting FortiSOAR playbook runtime issues.
cyops-workflow/sealab.log	Used for troubleshooting FortiSOAR playbook framework issues.
cyops-workflow/ssl_cyops_workflow_access.log	Used for troubleshooting playbook access issues.
cyops-workflow/ssl_cyops_workflow_error.log	Used for troubleshooting playbook errors.
cyops-workflow/uwsgi.log	Used for troubleshooting FortiSOAR playbook and connector issues.
cyops-integrations/connectors.log	Used for troubleshooting FortiSOAR connector related issues.
cyops-integrations/integrations.log	Used for troubleshooting FortiSOAR connector framework issues.
cyops-integrations/integrations/imap/listener.log	Used for troubleshooting the IMAP connector issues.
cyops-integrations/ssl_cyops_integrations_access.log	Used for troubleshooting connector access issues.
cyops-integrations/ssl_cyops_integrations_error.log	Used for troubleshooting connector errors.
csadm/db.log	Used for troubleshooting database externalization errors.

`install` For example, `6.4.1-2133.log`.	Used for troubleshooting FortiSOAR installation issues. Install logs are named according to the FortiSOAR version and build number.
`install/connectors.log`	Used for troubleshooting connector installation issues.
`upgrade_fortisoar_<version_number>-<timestamp>.log` For example, `upgrade_fortisoar_6.0.0-2020-02-18-1558604373.log`	Stores upgrade console log and you can use it to troubleshoot FortiSOAR upgrade issues.
`install/config-vm-<timestamp>.log` For example, `install/config-vm-27_Nov_2018_18h_14m_34s.log`	Used for troubleshooting FortiSOAR Configuration Wizard issues.

For troubleshooting FortiSOAR audit log issues use the `auditlog.log` located at `/var/log/cyops/cyops-gateway/auditlog.log`.

For OS level errors, use the `Messages` logs located at `/var/log/messages`.

For troubleshooting issues related to FortiSOAR containers, use the following logs:

- `/var/lib/docker/volumes/fortisoar_fortisoar_log/_data/cyops/extension/boot.log`
- `/var/lib/docker/volumes/fortisoar_fortisoar_log/_data/cyops/extension/extension.log`

For troubleshooting issues related to dedicated tenant nodes or FortiSOAR agents, see the following logs:

- `var/log/cyops/cyops-routing-agent/postman.log`
- `var/log/cyops/cyops-routing-agent/uwsgi.log`
- `var/log/cyops/cyops-routing-agent/ssl_cyops_routing_agent_error.log`
- `var/log/cyops/cyops-routing-agent/ssl_cyops_routing_agent_access.log`.

For troubleshooting issues related to FortiMonitor, see the following logs:

- `var/log/fm-agent/fm-agent.log`: For troubleshooting issues related to the FortiMonitor Agent
- `var/log/cyops/fmonitor/agent.log`: For troubleshooting issues related to the csadm CLI for FortiMonitor.

For troubleshooting issues related to the separate runtime user for integrations:

- `var/log/cyops/cyops-integrations/integration.log`
- `var/log/cyops/cyops-integrations/connectors.log`
- `getfacl var/log/cyops/cyops-integrations/`
- `getfacl opt/cyops-integrations/integrations`
- `getfacl var/run/uwsgi`
- `getfacl opt/cyops-integrations/.env`

Logging Levels

You can set the following logging levels in the log files:

- **DEBUG**: Low-level system information for debugging purposes.
- **INFO**: General system information.

- **WARNING**: Information describing a minor problem that has occurred.
- **ERROR**: Information describing a major problem that has occurred.
- **CRITICAL**: Information describing a critical problem that has occurred.

Changing the logging levels

- For **sealab** or **workflow**:
  a. Open the `/opt/cyops-workflow/sealab/sealab/config.ini` file and set the `WORKFLOW_LOG_LEVEL` parameter to the required logging level. For example, `WORKFLOW_LOG_LEVEL = 'INFO'`
  b. Restart the `uwsgi` service.
- For **integrations**:
  a. Open the `/opt/cyops-integrations/integrations/configs/config.ini` file and set the `connector_logger_level` parameter to the required logging level. For example, `connector_logger_level= 'INFO'`
  b. Restart the `uwsgi` service.
- For **celeryd**:
  a. Open the `/etc/celery/celeryd.conf` file and set the `CELERYD_LOG_LEVEL` parameter to the required logging level. For example, `CELERYD_LOG_LEVEL = 'INFO'`
  b. Restart the `celeryd` service.
- For **nginx (UI)**, **API**, or **php**:
  a. Open the `/opt/cyops-api/config/packages/prod/monolog.yaml` file and set the `level` parameter to the required logging level. For example, `level = 'INFO'`
  b. Run the `# systemctl restart php-fpm nginx` command.
- To enable all console errors in your browser, add `DEBUG = true` in local storage as follows:
  a. Open your browser and click select **More Tools** > **Developer Tools**.
  b. Click the **Application** tab and then click **Storage**.
  c. In Local Storage, type `DEBUG` as the key and `true` as the value.

List of key FortiSOAR services and processes

Name of Services/Processes	Description
postgresql-16	Service for all application data stored in postgresql DB. To know the status of this service, use the `# systemctl status postgresql-<postgresql_version>` command. For example, `# systemctl status postgresql-16`
elasticsearch	Service to bring up the elasticsearch service. To know the status of this service, use the `# systemctl status elasticsearch` command.
php-fpm	Service for PHP FastCGI implementation. To know the status of this service, use the `# systemctl status php-fpm`

	command.
uwsgi	Software application that aims at developing a full stack for building hosting services. uWSGI is named after the Web Server Gateway Interface. We host our playbook execution engine application and connector integrations applications on a uWSGI server. To know the status of uwsgi use the `# systemctl status uwsgi` command.
celeryd	celeryd is used to run the playbooks asynchronously in the FortiSOAR playbook execution engine. To know the status of celeryd use the `# systemctl status celeryd` command.
celerybeatd	celerybeatd is a playbook scheduler; used to kick off tasks at regular intervals, that are then executed by available worker nodes in the cluster. To know the status of use the `# systemctl status celerybeatd` command.
cyops-auth	Service used for FortiSOAR authentications. To know the status of this services, use the `# systemctl status cyops-auth` command.
cyops-tomcat	Service used for SSO, auditing, and websocket. To know the status of this services, use the `# systemctl status tomcat` command.
cyops-search	Service used for full-text searching, finding similar records and predicting the value of fields based on similarity. The cyops-search process contains sub processes for the 'Data', 'Status', 'System', and 'Search' queues. Sub processes enable the parallel execution of operations such as indexing, searching, etc., which improves performance and user experience.
cyops-ha	Service is responsible for setting up High Availability in the FortiSOAR environment
cyops-postman	Service is responsible for setting up and managing FortiSOAR's distributed multi-tenant setup.
cyops-integrations-agent	Service is responsible for running actions on remote FortiSOAR agents.
rabbitmq-server	Service is responsible to send audit and live sync notifications. This service is also responsible for data transfer in a distributed environment.
nginx	Service used for Web UI. To know the status of this services, use the `# systemctl status nginx` command.
fsr-workflow	Service is used for execution of the rule engine. To troubleshoot issues with this service, use the `/var/log/cyops/cyops-workflow/fsr-workflow.log`.
fsr-api-consumer	Service is used for installation of the Content Hub. To troubleshoot issues with this service, use the `/var/log/cyops/cyops-api/prod.log`.

If you want to restart, start, or stop all the services together, use FortiSOAR Admin CLI (`csadm`). For more information on `csadm`, see the FortiSOAR Admin CLI chapter.

You can run the `csadm` command on any FortiSOAR machine using any terminal. Any user who has `root` or `sudo` permissions can run the `csadm` command.

To restart FortiSOAR services, type: `# csadm services --restart`

To start FortiSOAR services, type: `# csadm services --start`

To stop FortiSOAR services, type: `# csadm services --stop`

To know the status of all FortiSOAR services type: `# csadm services --status`

To view the workflows queue, run the following command on your FortiSOAR system:
`rabbitmqctl list_queues -p fsr-cluster`

To know more about monitoring services, see the Monitoring FortiSOAR chapter.

Configurations required for exporting of records with unsupported character sets in the PDF format

You can export records to PDF, by clicking the 'Hamburger' menu on the grid view and selecting the **Export All Visible Columns to PDF** option, or you can open the record and in the detail view of the record, and click the **Export Record** > **Export as PDF** option. The FortiSOAR UI uses the 'pdfmake' library, to export the record data from the record's grid and view panel to the PDF format. The 'pdfmake' library by default comes with the ROBOTO font that does not support Korean, Chinese, and other character sets. Therefore, to export the record data that contains unsupported character sets from the record's grid and view panel in FortiSOAR to the PDF format, you can use one of the following options:

Option 1:

Use `SSH` and update the `/opt/cyops-ui/vendor/config.json` to provide the CDN URLs for a specific font:

```
"pdfmake": {
    "font": {
         "normal": "https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.66/fonts/Roboto/Roboto-
Regular.ttf",
         "bold": "https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.66/fonts/Roboto/Roboto-
Medium.ttf",
         "italics": "https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.66/fonts/Roboto/Roboto-
Italic.ttf",
         "bolditalics":
"https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.66/fonts/Roboto/Roboto-MediumItalic.ttf"
    }
 }
```

Option 2

Build and use the custom font for 'pdfmake' as follows:

1. Copy font files into the `/opt/cyops-ui/vendor/pdfmake` directory.
2. Build the `vfs_fonts.js` file with new fonts. For example to build the `vfs_fonts.js` file with the 'Roboto' font family use the following command:
   `# sh build.sh Roboto-Regular.ttf Roboto-Medium.ttf Roboto-Italic.ttf Roboto-MediumItalic.ttf`
3. Update the `/opt/cyops-ui/vendor/config.json` file with the font references:
```
"pdfmake": {
   "font": {
     "normal": "Roboto-Regular.ttf",
     "bold": "Roboto-Medium.ttf"
     "italics": "Roboto-Italics.ttf"
     "bolditalics": "Roboto-MediumItalic.ttf"
   }
}
```
   You can also refer to the `/opt/cyops-ui/vendor/pdfmake/README.txt` file for instructions.

Regenerating Self-Signed certificates

For FortiSOAR enterprise systems:

If your self-signed certificates are expired or about to expire, regenerate them by running the following command as a *root* user (using '`sudo su`' and using the `csadmin` password) in an SSH session:
`csadm certs --generate 'hostname'`
This command regenerates all internal certificates.

If your deployed nginx certificate has expired, re-deploy the updated certificate using the `csadm certs --deploy` command.

For FortiSOAR High Availability (HA) clustered environments:

1. Run the following command on both the primary and secondary nodes as a *root* user (using '`sudo su`' and the `csadmin` password) in an SSH session:
   `csadm certs --generate 'hostname'`
2. For systems running on FortiSOAR release 7.5.1 and later, run the following command ***only*** on secondary nodes to ensure they share the same CA certificate as the primary node:
   `csadm mq ca reset-with-primary`
3. Restart all FortiSOAR services using the following command:
   `#csadm services --restart`

Notification alerts

If users have enabled FortiSOAR health notifications, they will receive email alerts when certificates are close to expiry. Alternatively, they can check the messages logs for warnings such as:

```
Mar 6 15:12:58 <hostname> python3[2732752]: [2025-03-06 15:12:58] WARNING thread: run(): 58:
Postgres certificate is going to expire on 27 Mar, 2025
Mar 6 15:12:58 <hostname> python3[2732752]: [2025-03-06 15:12:58] WARNING thread: run(): 58:
Nginx certificate is going to expire on 27 Mar, 2025
```

Additional settings for record similarity and field predictions

FortiSOAR supports "Record Similarity" i.e., FortiSOAR displays records that are similar to the record on which you are working. FortiSOAR also supports "Record Field Value Prediction" i.e., FortiSOAR predicts values of fields of your choice within a record from the values of fields of existing records based on the criteria you have defined, making it easier for analysts to make informed decisions. For more information, see the *Working with Modules - Alerts & Incidents* chapter in the "User Guide."

This section provides information on how you can tune the results that are displayed by FortiSOAR for record similarity and field predictions using the following parameters in the `/opt/cyops/config/cyops-search/config.yml` file:

- `minimum_should_match: <percentageValue>`: This setting defines that a record will be considered similar only if there is a match of at least the percentage value that you have specified on the related fields. This is especially true for similarity based on related records. For example, if you set this parameter as `minimum_should_match: 10%` (default), then if you have defined similarity for alerts based on related indicators, then FortiSOAR will display only those records as similar that match a minimum of 10% of the indicators. Therefore, for an alert that has 10 related indicators, FortiSOAR similarity results will display alerts that even have one common indicator; but if an alert has 20 related indicators, then FortiSOAR similarity results will display only those alerts that have at least 2 indicators in common.
- `max_query_terms: <numberOfItems>`: This setting defines how many terms of the parent record will be looked up for similarity in other records. Continuing the same example as above, if you set this parameter as `max_query_terms: 25` (default), then if an alert has more than 25 indicators, only 25 of them will be checked for similarity in other records. Note that increasing the value of this setting will increase the time FortiSOAR takes to return similarity and suggestion results.

For more information on the above parameters and other parameters, refer to the ElastiSearch reference at: https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-mlt-query.html

Configure the maximum number of records to be fetched in one call

To optimize performance, the maximum number of records to be fetched in single call should be limited to 100 for optimal performance.

Starting from release 7.5.0, the default limit of fetching of records with many to many relationships, such as Alerts to Indicators, is set to 100. This change aims to improve performance and prevent out of memory exception during response serialization.

You can configure this limit in case of user-defined module such as 'Alerts', 'Incidents' etc. You cannot configure limits for system-defined modules such as 'People', 'Appliances', 'Agents', etc. The limit can be configured as the following:

- A request parameter: `$fsr_max_relation_count`.
  An example of this is when using the 'Find Records' step in a playbook, you can specify the maximum number of correlated records to be fetched.
- A parameter: `response_max_relation_count` in the `/opt/cyops-api/config/parameters_prod.yaml` file.
  NOTE: Setting the `response_max_relation_count` to '1' or '-1' will include the entire result set in the response. It's also important to note that the value set in `$fsr_max_relation_count` takes precedence over the value set in `response_max_relation_count`. For example if you have set 500 in `$fsr_max_relation_count` and kept the default value of 100 in `response_max_relation_count`, the number of records fetched will be limited to 500.

Additionally, the limit for fetching audit log records is also set to 100. To modify this limit, change the value of the `max_relation_count` parameter in the `/opt/cyops-api/config/parameters_prod.yaml` file:

```
# max number of relations to normalize
   max_relation_count: 100
```

After updating the value of either the `response_max_relation_count` or the `max_relation_count` parameter, you need to run the following command:

```
# systemctl restart php-fpm && sudo -u nginx php /opt/cyops-api/bin/console cache:clear && systemctl restart php-fpm
```

Change the default value of some of the user profile parameters

An administrator with CRUD permissions on the `Security` module can change the default value of the following user-profile related parameters:

Parameter Name	Description	Default Value
`max_reset_attempts`	Maximum number of times users' can click the **Reset Password** link before actually resetting their password. If the user exceeds the value set in this parameter, then users' will not get a new link to reset their password based on the number of hours specified in the `reset_locktime`parameter. By default, the `max_reset_attempts` is set to 10 times and the `reset_locktime` is set to 12 hours, therefore, if a user clicks the **Reset Password** 11 times without actually resetting their password, then the user will not get a new link to reset their password for 12 hours.	10 times
`reset_locktime`	Number of hours that users' will not get a new link to reset their password if they exceed the value set in the `max_reset_attempts` parameter.	12 hours

`max_failed`	Number of times that users' can enter an incorrect password, while logging into FortiSOAR, before their account gets locked. If the user exceeds the value set in this parameter, then the user will get locked out based on the number of minutes specified in the `lock_minutes`parameter. By default, the `max_failed` is set to 5 times and the `lock_minutes` is set to 30 mins, therefore, if a user enters an incorrect password 6 times, then their account gets locked for 30 mins.	5 times
`lock_ minutes`	Number of minutes that users' account gets locked if they exceed the value set in the `max_failed` parameter.	30 mins

To change the value of the `max_reset_attempts` parameter, the administrator should run the following curl command on their FortiSOAR system:

```
curl -X PUT \
  https://<FORTISOAR HOSTNAME/IP>/api/auth/config \
  -H 'Authorization: <Bearer Token>' \
  -d '{
       "option":"max_reset_attempts",
       "value":5
}'
```

The above command changes the number of times users' can click the **Reset Password** link to 5 times, i.e., a user can click the **Reset Password** link 5 times without actually setting the new password. However, if the user clicks the **Reset Password** link for the 6th time, the user will be blocked.

Similarly, to change the value of the `reset_locktime`, `max_failed`, and `lock_minutes` parameters, the administrator should run the same curl command on their FortiSOAR system, but after changing the option and value parameter values:

```
curl -X PUT \
  https://<FORTISOAR HOSTNAME/IP>/api/auth/config \
  -H 'Authorization: <Bearer Token>' \
  -d '{
       "option":"reset_locktime",
       "value":2
}'
```

The above command changes the number of hours that users' will not get a new link to reset their password to 2 hours if they have exceeded the value set in the `max_reset_attempts` parameter.

```
curl -X PUT \
  https://<FORTISOAR HOSTNAME/IP>/api/auth/config \
  -H 'Authorization: <Bearer Token>' \
  -d '{
       "option":"max_failed",
       "value":3
}'
```

The above command changes the number of times that users' can enter an incorrect password while logging into FortiSOAR before their account gets locked to 3, i.e., users' account will be locked if they enter an incorrect password 4 times while logging into FortiSOAR.

```
curl -X PUT \
  https://<FORTISOAR HOSTNAME/IP>/api/auth/config \
  -H 'Authorization: <Bearer Token>' \
  -d '{
       "option":"lock_minutes",
       "value":15
}'
```

The above command changes the number of minutes that users' account will be locked to 15 minutes if they have exceeded the value set in the `max_reset_attempts` parameter.

Security considerations for Websockets

The Websocket server allows connection from any hostname URL UI. To restrict connections to the Websocket server, you can implement Cross-Origin Resource Sharing (CORS) restrictions in WebSocket.

To implement CORS in WebSocket, you have to add a comma-separated list of allowed hosts (IP address or domain name) in the `/etc/cyops/config.yml` file and the hostname should start with "`http://`" or "`https://`". Dummy entries have already been added in the `/etc/cyops/config.yml` file to help you in this process. If you have added domain names to the `/etc/cyops/config.yml` file, then you must also restart the 'tomcat' service, using the `systemctl restart cyops-tomcat` command.

Once the hostname entries have been added, the WebSocket will only establish a connection with those UIs whose URLs match the specified hostnames.

Backup of the most recent System View Template (SVT) before saving the edited SVT

Since SVT modifications are challenging to undo, the current SVT is backed up in the database before saving the edited SVT, meaning that both the current SVT and the previous to the most recent SVT are saved.
If users need to restore the previous to the most recent SVT, then they can do the following:

1. SSH to your FortiSOAR instance with 'root' access.
2. Run the following command to restore the 'detail' view SVT of a module:
   `sudo -u nginx php /opt/cyops-api/bin/console app:restore:viewtemplate --restore --system-view-template modules-<moduleName>-detail`
   For example, if you want to restore the 'detail' view SVT of the 'alerts' module, run the following command:
   `sudo -u nginx php /opt/cyops-api/bin/console app:restore:viewtemplate --restore --system-view-template modules-alerts-detail`
   To restore the 'form' view SVT of a module run the following command:
   `sudo -u nginx php /opt/cyops-api/bin/console app:restore:viewtemplate --restore --system-view-template modules-<moduleName>-form`
   Similarly, to restore the 'list' view SVT of a module run the following command:

```
sudo -u nginx php /opt/cyops-api/bin/console app:restore:viewtemplate --restore --system-
view-template modules-<moduleName>-list
```

Automation added to fetch Google Chrome updates

From release 7.6.0 onwards, FortiSOAR includes automation to fetch the latest updates for Google Chrome. If a vulnerability scan detects any issues with the Google Chrome browser, users can upgrade Chrome using the following command without the need to install a security patch:
`yum update google-chrome-stable -y`

Troubleshooting Tips

Your Workflow data size has increased

Increase in your Workflow data can cause performance bottlenecks.

Resolution:

You can purge Executed Playbook Logs using the **Purge Logs** button on the top-right of the `Executed Playbook Logs` dialog. For more information on purging, see the *Debugging and Optimizing Playbooks* chapter in the "Playbooks Guide."

Error displayed while performing a search operation in FortiSOAR

Resolution:

If you get any error while performing a global search in FortiSOAR, check that the `elasticsearch.service` and the `cyops-search.service` are running.

If these are not running, then start these services using the following commands:

```
# systemctl start elasticsearch
```

```
# systemctl start cyops-search
```

For more information, see the `FortiSOAR Search Errors` topic in the Elasticsearch Configuration chapter.

Reindexing FortiSOAR modules for search

Partial indexing of a module, or when a module does not get indexed, can lead to errors in FortiSOAR search. You can manually reindex any skipped or unsuccessfully indexed modules. For more information, see the `FortiSOAR Search Errors` topic in the Elasticsearch Configuration chapter.

Resolution:

To reindex all the FortiSOAR modules, run the following command:

```
$ sudo -u nginx php /opt/cyops-api/bin/console app:elastic:create --env="prod"
```

To reindex specific FortiSOAR modules, run the following command:

```
$ sudo -u nginx php /opt/cyops-api/bin/console app:elastic:create --env="prod" --index='
{"type":"type of the module(s)"}'
```

For example:

```
$ sudo -u nginx php /opt/cyops-api/bin/console app:elastic:create --env="prod" --index='
{"type":"indicators", "tasks"}
```

FortiSOAR crashing with "out of memory" errors

By default, FortiSOAR configures Elaticsearch to use 4 GB of RAM. If there are too many records or any very heavy records (such as large files uploaded) created per day on the system, it might crash with "out of memory" errors. To fix this, you must increase the memory allocated to Elasticsearch.

Resolution

1. Update the value of the following parameters in the `/etc/elasticsearch/jvm.options` file to a higher value based on memory available on your server:
   `-Xms4g`
   `-Xmx4g`
2. Restart the Elasticsearch service using the following command:
   `systemctl restart elasticsearch`

Changing Postgres worker memory

When the primary data in the system becomes large (eg, over million alerts) and you notice that the system is slow to respond. The slowness could be caused due to database queries taking longer with the increased database size.

Resolution

You can fine tune this behavior by increasing the following Postgres settings based on the available free memory on the system:

1. Increase the shared buffer and worker memory in the `/var/lib/pgsql/16/data/postgresql.conf` file:
   `shared_buffers = 7680MB`
   `work_mem = 8MB`

2. Restart Postgres using the following command:
   `systemctl restart postgresql-16`

Crashing of the Tomcat server when the 'Detailed' mode for Log Forwarding is enabled

If you have enabled the 'Detailed' mode for Log Forwarding in **Settings** > **Log Forwarding** > **Audit Logs**, then your Tomcat server might crash and the 'Live Sync' might continuously fail.

Resolution

To avoid crashing of your Tomcat server and you do not want to increase the memory assigned to the Tomcat service, you need to disable log forwarding and then set log forwarding to the 'Basic' mode as follows:

1. SSH to your FortiSOAR server and login as a *root* user.
2. List the current log forward settings using the following command:
   `# csadm log forward show-config`
3. Copy the UUID from the output of the `show-config` command.
4. Disable forwarding of the audit logs to the syslog server using the following command:
   `# csadm log forward update-config --uuid <uuid> --filter application`
   or
   `# csadm log forward update-config --uuid <uuid> --filter none`
5. Restart the Tomcat service using the following command:
   `# systemctl restart cyops-tomcat`
6. Log on to FortiSOAR, and click **Settings** > **System Configuration**.
7. Click the **Log Forwarding** tab.
8. From the **Specify Audit Log Detail Level** drop-down list, select **Basic**, and click **Save**.

If you for some reason you require to use the 'Detailed' mode for log forwarding, you can increase the memory assigned to the Tomcat service running the log forwarding process using the following steps:

1. In `/usr/lib/systemd/system/cyops-tomcat.service` change the *Xms* value as follows:
   `Environment="CATALINA_OPTS=-Xms4096M -Xmx4096M -server -XX:+UseParallelGC"`
2. Run `systemctl daemon-reload`
3. Run `systemctl restart cyops-tomcat`

Recommendation Engine does not work and displays an "ML Service not running" error

If you have upgraded FortiSOAR or due to some other reason, you might observe that the Recommendation Engine is not working and not displaying any suggestions. You can also get an error such as "ML Service is not running." This issue can occur if the `ml` service fails to start on port 10449.

Resolution

Restart the `uwsgi` service and you will observe that the `ml` service is running on port 10449. Once the `ml` service starts, you will observe that the recommendation engine begins to work and displays appropriate suggestions like phishing classification.

Enabling PostgreSQL logging to identify API requests that are taking longer than expected

1. Open the `/var/lib/pgsql/16/data/postgresql.conf` file.
2. Ensure that the "`logging_collector`" parameter is set to 'on'.
3. Find the "`log_min_duration_statement`" parameter, and update its value based on your requirements, then save the `postgresql.conf` file:
   - Setting the value of this parameter to '-1' disables time-based logging.
   - Setting the value of this parameter to a positive integer logs the duration of any request that lasts longer than 'X' milliseconds, where 'X' is the value that you have assigned to this parameter.
   - Setting the value of this parameter to '300' helps identify the API requests that are taking longer than usual.
4. Restart the 'postgresql' service using the following command:
   `systemctl restart postgresql-16`
5. Monitor the `.log` files in the `/var/lib/pgsql/16/data/pg_log/` directory to know which API requests are taking longer than usual.

www.fortinet.com

Copyright© 2025 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, FortiCare® and FortiGuard®, and certain other marks are registered trademarks of Fortinet, Inc., and other Fortinet names herein may also be registered and/or common law trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance and other metrics contained herein were attained in internal lab tests under ideal conditions, and actual performance and other results may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's Chief Legal Officer, with a purchaser that expressly warrants that the identified product will perform according to certain expressly-identified performance metrics and, in such event, only the specific performance metrics expressly identified in such binding written contract shall be binding on Fortinet. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. Fortinet disclaims in full any covenants, representations, and guarantees pursuant hereto, whether express or implied. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.