Kiteworks

Kiteworks Secure MFT Server User Guide

February 2024

Version 8.3.0-01

Copyright

Corporate Headquarters

Kiteworks USA, LLC
1510 Fashion Island Blvd., Suite 100
San Mateo, CA 94404, USA

Contact Us

p. 1.650.485.4300 | f. 1.650.485.4308
www.kiteworks.com

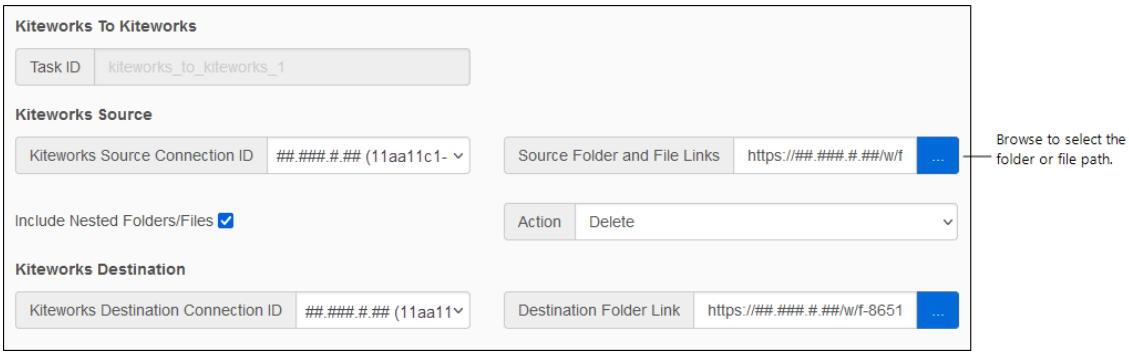
Table of Contents

What's new

New features in the Kiteworks Secure MFT Server

Select folder paths when creating DAGs

When creating DAGs, browse to select folder and file source and destination paths instead of manually copying the paths from applications. First select the connection ID of the source or destination server. Then in the source or destination field, click the browse button to select the folder or file path.



See Create and edit DAGs.

Hash characters in DAG schedule cron expressions

When configuring a DAG run schedule, hash (#) characters are supported in cron expressions. For example, “0 9 * * Mon#2” will run the DAG at 9 AM on every second Monday of the month.

See Create and edit DAGs.

Create rules for moving files

When configuring DAGs, you can now define additional rules for file moves in the following JSON format:

```
{ "rules": [
  {   "id" : "String",
      "source_folder" : "String",
      "destination_folder" : "String",
      "regex_type" : "String",
      "source_regex" : "String",
      "destination_regex" : "String"    }
  ] }
```

The rules are executed sequentially and each rule resembles a separate DAG run. If a rule doesn't match any files, it is skipped.

- **id** - Any self-assigned ID which is used in the log.
- **source_folder** - Must contain the path from the root.
- **destination_folder** - Must be inside the destination defined at the DAG level.
- **regex_type, source_regex, destination_regex -** Optional, and if not provided the DAG values will be

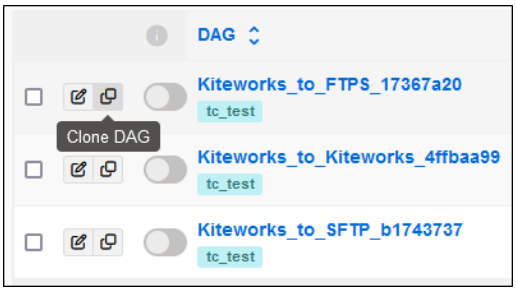
used special characters such as \d \w. These characters need to be escaped with 4 backslashes (for example \\\\d \\\\w).

Change file names using regular expressions

Previously, regular expressions could be used to select files in the source system that should be transferred. Later the function was extended so that the target folders could be assembled from parts of the source folder using regular expressions. In this version it is now possible to rename the file names on the target system with parts from the source folders and source file names.

Copy DAGs

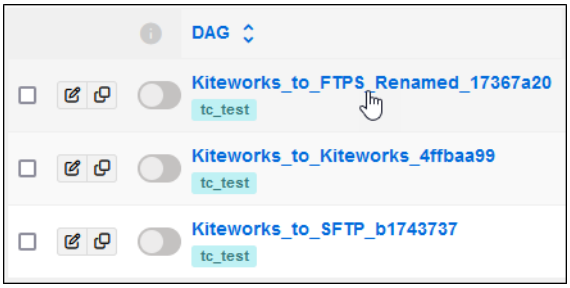
When creating similar DAGs, save time by copying and then editing the new DAGs. Change the name of the DAG (DAG ID), edit its form and operator properties, and then customize the workflow as needed.



See Copy DAGs.

Rename DAGs

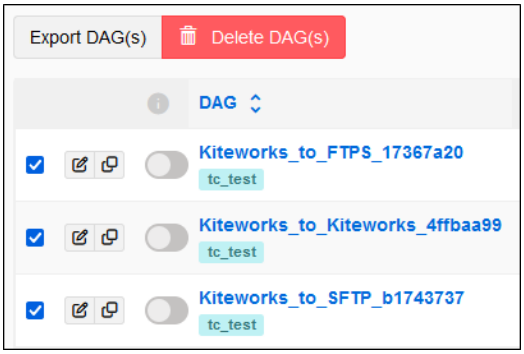
When editing DAGs, now you can also change the DAG name (DAG ID). A new number is appended to the name to ensure that the name is unique.



See Create and edit DAGs.

Delete DAGs

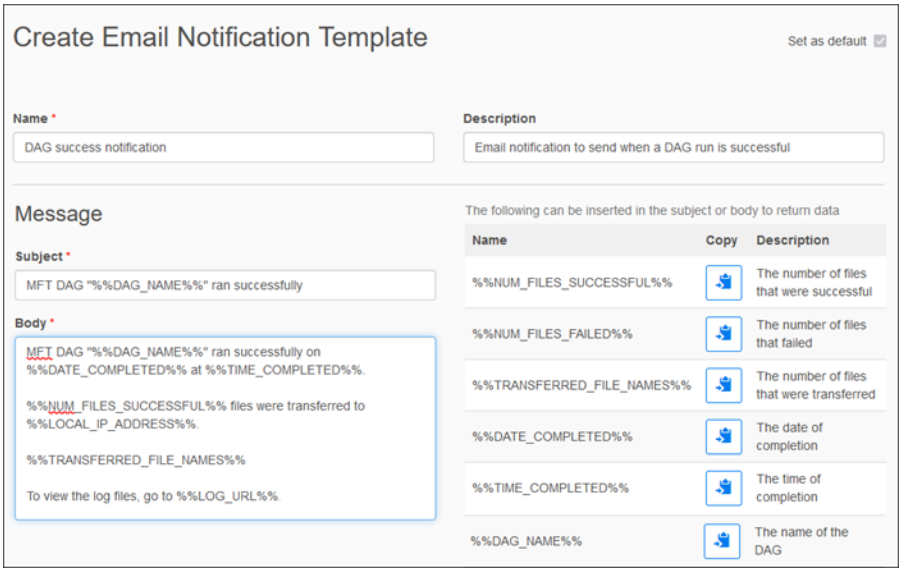
Delete a single DAG or multiple DAGs at the same time. On the DAGs view page, select the DAGs you want to delete, and then click Delete DAG(s).



See Delete DAGs.

Create email notifications

Create custom email notification templates to get notified when DAG runs are successful, partially successful, or when an error occurs while processing a DAG run. When configuring a DAG, you select the notification template you want to use to notify recipients of the DAG run status. An email notification is sent after each attempt to perform a task, so there may be multiple email notifications until the task has successfully completed.



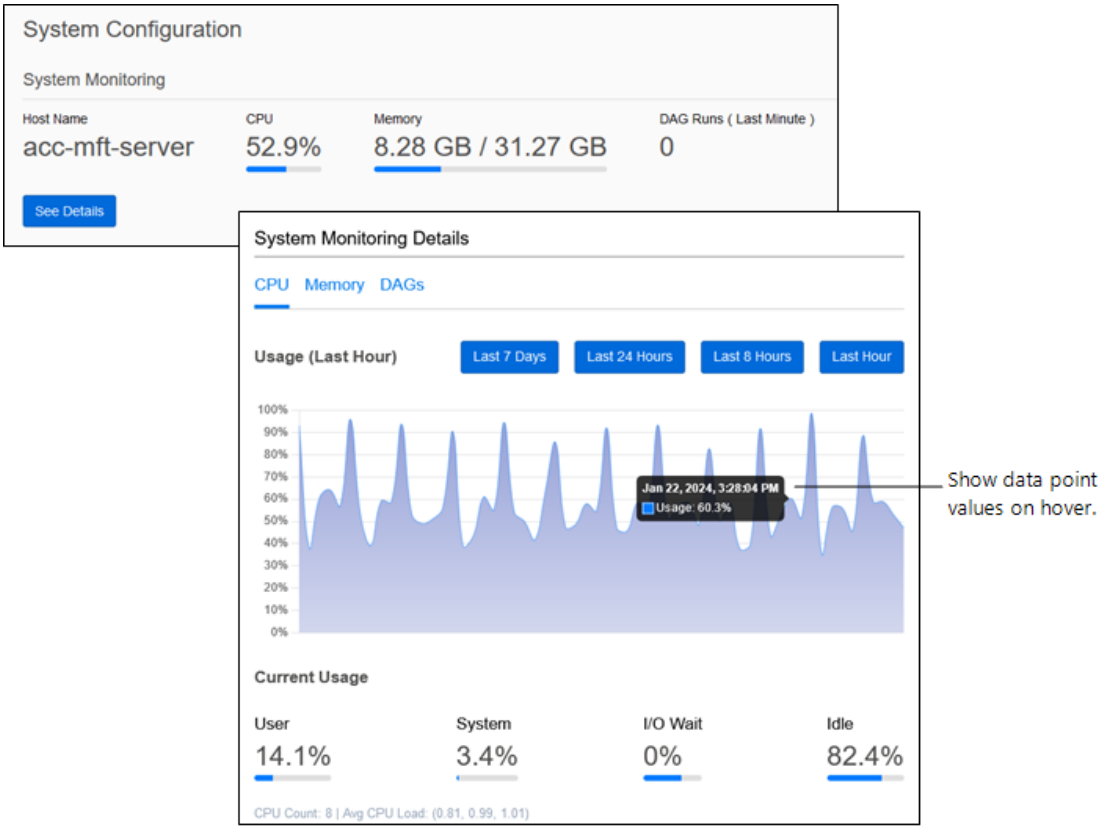
See Create email notifications.

File action timestamps added to logs

When viewing the task logs, a timestamp corresponding to each file action has been added to the log table.

Monitor system health

Monitor CPU, memory, and DAG runs for each server in the cluster. When viewing details, you can display the value of a data point when you rest the pointer over it. On the Secure MFT Server navigation bar, click Admin > Setup > System Configuration.



For information about managing the MFT server, see the *Kiteworks Secure MFT Server Deployment Guide*.

Operators renamed reflect Kiteworks branding

The following DAG operators were renamed to reflect the latest Kiteworks branding.

For a description of each operator, see the DAG form operators section in Create and edit DAGs.

Operator type	Original operator name	New operator name
Source	Accellion File List	Kiteworks File List
Source	Accellion Download	Kiteworks Download
Source	Accellion Privileged Download	Kiteworks Privileged Download
Source	Accellion Archived Mail List	Kiteworks Archived Mail List
Source	Accellion Archived Mail Domain	Kiteworks Archived Mail Domain
Destination	Accellion Upload	Kiteworks Upload
Destination	Accellion Sanitized Upload	Kiteworks Sanitized Upload
Scan	Accellion Scan Status	Kiteworks Scan Status
Action	Accellion Action	Kiteworks Action
Action	Accellion Lock	Kiteworks Lock
Action	Accellion Unlock	Kiteworks Unlock
Action	Accellion Archived Mail Send	Kiteworks Archived Mail Send
Transfer	Accellion to SFTP	Kiteworks to SFTP
Transfer	Accellion to Accellion	Kiteworks to Kiteworks
Transfer	Accellion to SMB	Kiteworks to SMB
Transfer	Accellion to FTPS	Kiteworks to FTPS
Transfer	FTPS to Accellion	FTPS to Kiteworks
Transfer	SMB to Accellion	SMB to Kiteworks
Transfer	SFTP to Accellion	SFTP to Kiteworks

Get started

Sign in to the Kiteworks Secure MFT Server

To sign in:

1. In a web browser, enter the address of the Secure MFT Server.
   Example: https://<host name or ip address>:8080.
2. On the sign in screen, type your user name and password, and then click Sign In.

To sign out:

In the upper right corner of the screen, click your user name, and then click Logout.

Change your password

If someone created a password for you, you should change it immediately so that your account is secure.

To change your password:

1. In the upper right corner of the screen, click your user name, and then click Profile.
2. On the user information page, click Reset My Password.
3. On the Reset Password Form page, enter and confirm a new password, and then click Save.
   Note: Passwords are encrypted for security. You won't see them on the page.
4. To use your new password, sign out and then back in again.

Using Kiteworks Secure MFT Server

The Kiteworks Secure MFT Server is a customized Airflow web interface that makes it easy for you to visualize workflows running in production, monitor their progress, and troubleshoot issues. You can view the status of a job, its underlying code, and perform actions on the job.

The navigation bar contains menus for accessing functions. The role you've been assigned determines which menus and options you can access. Refer to About roles.

A variety of views are provided for you to monitor workflows in detail.

DAGs view

Airflow defines a Directed Acyclic Graph (DAG) as "a collection of all the tasks you want to run, organized in a way that reflects their relationships and dependencies." DAGs are acyclic, meaning there are no circular dependencies or loops that can cause infinite execution loops. DAGs are essentially "workflows" and you'll see the terms "DAG" and "workflow" used interchangeably throughout this document.

Use the DAGs view to see the complete list and status of DAGs created by you and other users, and to perform actions on the workflows. Your role determines whether you can see all DAGs created by other users or only a subset in the list.

At a glance, for each DAG you can see how many tasks are running, have succeeded, or have failed. If a DAG contains tags, you can filter the list to view a subset of DAGs.

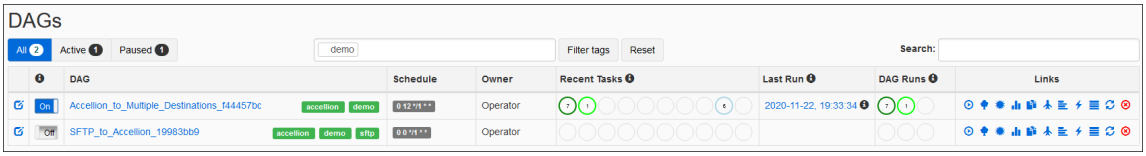


Table 1. Columns in the DAGs View

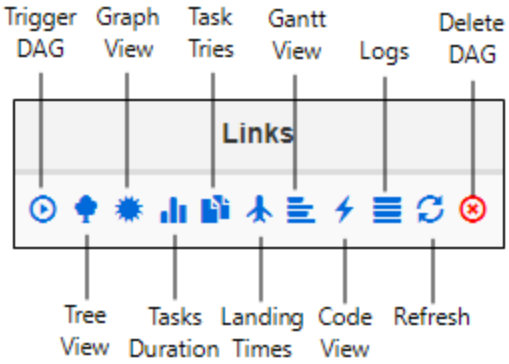
Name	Description (Optional)
Edit	Opens the DAG in the editor for editing.
Info (On/Off)	Turn the DAG on or off. By default, DAGs are off. A DAG won't run according to its schedule until you turn it on. If you turn off a DAG while it's running, the run will complete and then the DAG will turn off.
DAG	The name of the DAG along with unique ID (GUID) that was automatically appended to the DAG name. If the DAG includes filter tags, the tags are also listed in the column. Click the DAG name to open the Tree View to see a summary of the DAG's run across time.

Table 1. Columns in the DAGs View (continued)

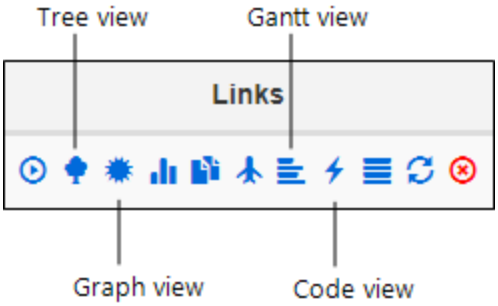
Name	Description (Optional)
Schedule	The amount of time after which the scheduler will trigger the DAG. You set the schedule when you create a DAG. Click the value to see the schedule, along with the status of any prior DAG runs. In the Run ID column, if the ID starts with "manual", this means that the DAG run was triggered manually. If it starts with "scheduled", the DAG was triggered according to its schedule.
Owner	The user name that was provided when the DAG was created.
Recent Tasks	The status of the tasks in the most recent DAG run. If the DAG is running, this column shows the live status of tasks in process. Each circle represents a possible status of tasks in the latest run. Hover over a circle to see its status indicator. Tasks that haven't been scheduled or queued have no status and are shown in the second to last "none" circle.
Last Run	The most recent date and time the DAG was run.
DAG Runs	The status of entire DAG runs. Each circle represents a possible status. When all tasks in a workflow are completed successfully, the first circle shows the status "success". Numbers in these circles increase each time the DAG is run.
Links	Shortcuts for accessing different views and selected pages in the application.

Tree, Graph, Gantt, and Code views

The Links column contains shortcuts for accessing different views and selected pages in the application.



The Tree, Graph, Gantt, and Code views are the most commonly used views. From each view, you can click on a task instance to see detailed metadata and perform some actions.

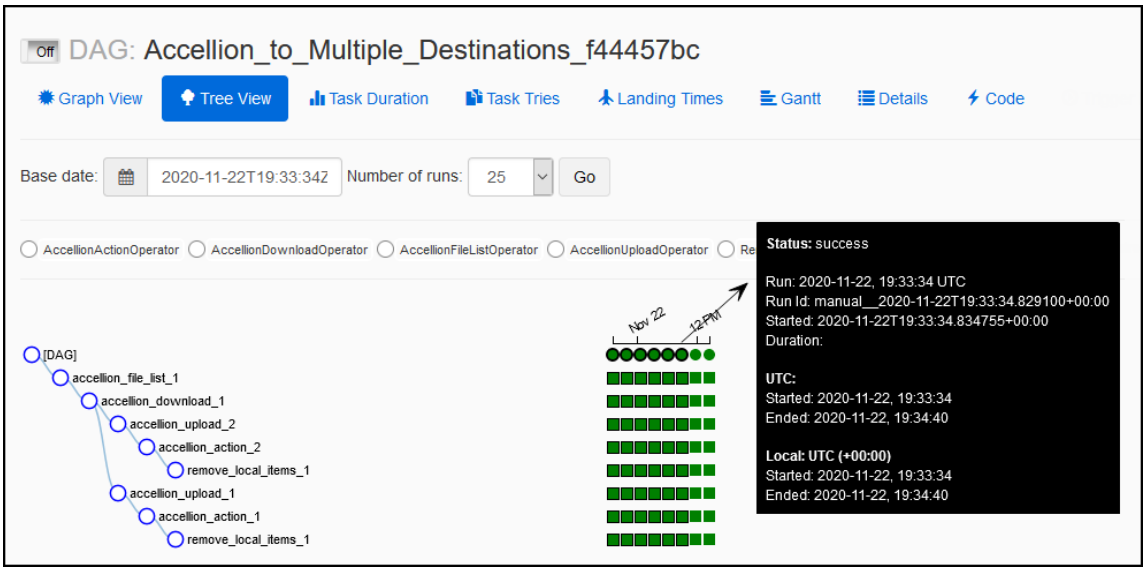


Tree view

Use the Tree view to get a summary of a DAG's run across time. If a task in the workflow is late, you can quickly see where the different steps are and identify the blocking ones.

Each column represents a DAG run. Each box in a column corresponds to the task shown on the left. DAG run columns are shown in chronological order.

The circle at the top of each column represents the overall DAG run and the corresponding boxes represent individual tasks in the run. Hover over a circle or box to see details about a run or task. You can also click on circles or boxes to perform actions at the DAG or task level.



Graph view

Use the Graph view to get a comprehensive view of a DAG at its task level, along with its dependencies and the status of each DAG run. Each task instance is surrounded by a border with a color representing the state of the task.

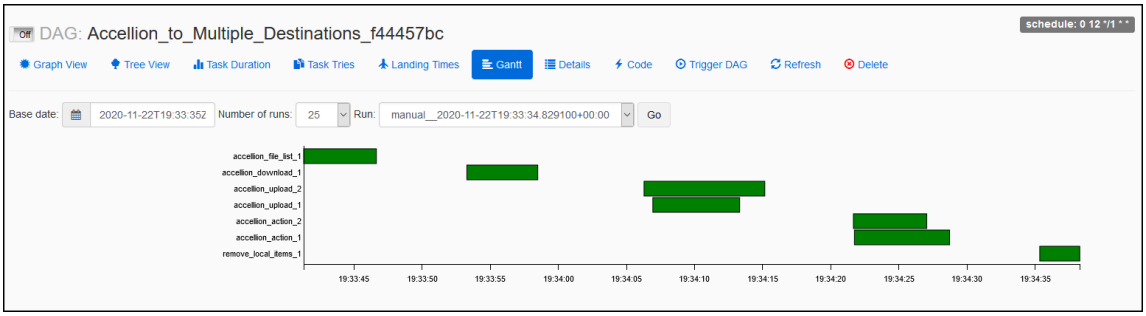
Above the pane is a list of operators used in the DAG. Above that are filters and layouts. From the Run list, you can select a prior run to see its status. The Layout list enables you to change the graphical representation of the DAG.

Hover over a task to see details such as when the task started, how long it took to complete, and so on. Click a task to perform actions on it. You can also view the log for that particular task to see which files transferred successfully, troubleshoot any failures, and more.

Gantt view

The Gantt view lets you analyze task duration and overlap. Use it to quickly identify bottlenecks and where the bulk of the time is being spent in a specific DAG run. For example, you can see if a task is taking too long to run, which tasks are overlapping or running in parallel, and so on.

The Gantt view also shows time gaps between tasks. The latency time of the scheduler can depend on factors such as the number of threads available, scheduler hardware time, complexity of jobs, available resources, and so on.



Code view

The Code view shows the underlying code in the DAG.

Additional ways to monitor DAGs

The Links column includes additional shortcuts to other pages useful for monitoring DAGs.

- Task Duration

  The Task Duration page shows the duration of different tasks in a DAG over past runs. Use it to identify outliers and quickly understand where time is being spent in your DAG over the course of its runs. The y-axis shows the task duration and the x-axis shows the time it took for a task to be run. Hover over lines to view task data.

- Task Tries

  If tasks in a DAG are failing, use the Task Tries page to see the number of times the scheduler tried to run the tasks. Each dot in a line represents a complete DAG run. Hover over a dot to see the number of tries for each task.

- Landing Times

  Use the Landing Time page to see whether a DAG run completed on schedule, including any retries. The landing time is the time when a job completed minus the time it should have started.

- Logs

  Use the Logs page to view individual logs associated with a DAG.

Manage roles and users

About roles

Roles are a collection of permissions. Each user account requires a role to access menus, views, and features.

Role	Description
Admin	Access to all pages, menus and permissions. Admins can also create custom roles.
DAG Manager	Access to the DAGs view page and the Browse menu for monitoring DAG runs, jobs, logs, and more. Access to the Admin menu for modifying pools, variables, and XComs. Access to the Docs and About menus.
Operator	Access to the Admin menu to create and edit connections. Access to the Docs and About menus.
Public	For security purposes, the Public role is assigned to LDAP user accounts by default. The Public role has no permissions, and none can be assigned. Users assigned the Public role will not be able to view the UI or perform any functions until an administrator assigns them higher roles.

To see which role you've been assigned, in the upper right corner of the screen, click the arrow next to your name and view your user account profile.

Also, if your role includes access to the Security menu, you can go to the user list page to view your role and the roles assigned to other users. From the Security menu, click List Users.

For a list of permissions assigned to each role, refer to Create and edit roles.

Create and edit roles

Roles are a collection of permissions. Each user account requires a role to access menus, views, and features. Once you create a role, you can assign it to a user account.

You can create a role based on a default one or create a role from scratch. Then you can assign it to a user account.

Recommendation: Although your role may allow you to edit default roles, use them as templates for creating custom roles. Once you edit a default role, you can't automatically restore its default permissions.

To create a role from an existing role:

1. On the navigation bar, click Security > List Roles.
2. Select the role you want to use as a template, and then click Actions > Copy Role.
3. Next to the copied role, click Edit Record.
4. Edit the role name and permissions.
   - To add a permission, click next to the last permission in the list, and then select the permission.
   - To find a permission, start typing the name of the permission.
5. When finished, click Save.

To create a role from scratch:

1. On the navigation bar, click Security > List Roles.
2. On the toolbar, click Add a New Record.
3. Type a name for the role.
4. Assign permissions to the role, and then click Save.

To edit a role:

1. In the row you want to edit, click Edit Record.
2. Edit the record as needed, and then click Save.

To delete a role:

In the row you want to delete, click Delete Record.

Role and permissions

The following permissions are in the order they appear when you go to add them to a role.

Permission	Admin	DAG Manager	Operator	Public
can read on Connections	✓	✓	✓	–
can read on DAGs	✓	✓	–	–
can read on DAG Runs	✓	✓	–	–
can read on Task Instances	✓	✓	–	–
can read on Audit Logs	✓	✓	–	–

Permission	Admin	DAG Manager	Operator	Public
can read on ImportError	✓	✓	–	–
can read on Pools	✓	✓	–	–
can read on Providers	✓	–	–	–
can read on Variables	✓	✓	–	–
can read on XComs	✓	✓	–	–
can read on DAG Code	✓	✓	–	–
can read on Configurations	✓	–	–	–
can read on Plugins	✓	–	–	–
can read on Roles	✓	–	–	–
can read on Permissions	✓	–	–	–
can read on Users	✓	–	–	–
can read on Website	✓	✓	✓	–
can read on My Password	✓	✓	✓	–
can read on My Profile	✓	✓	✓	–
can read on Jobs	✓	✓	–	–
can read on SLA Misses	✓	✓	–	–
can read on Task Logs	✓	✓	–	–
can read on Task Reschedules	✓	✓	–	–
can read on Event Listener	✓	✓	–	–
can read on Triggers	✓	–	–	–
can read on Passwords	✓	–	–	–
can read on User Stats Chart	✓	–	–	–
can read on View Menus	✓	–	–	–
can read on Permission Views	✓	–	–	–
can edit on Connections	✓	✓	✓	–
can edit on DAGs	✓	✓	–	–
can edit on DAG Runs	✓	✓	–	–
can edit on Task Instances	✓	✓	–	–

Permission	Admin	DAG Manager	Operator	Public
can edit on Pools	✓	✓	–	–
can edit on Variables	✓	✓	–	–
can edit on Roles	✓	–	–	–
can edit on Users	✓	–	–	–
can edit on My Password	✓	✓	✓	–
can edit on My Profile	✓	✓	✓	–
can edit on Event Listener	✓	✓	–	–
can edit on AccellionDagEditorView	✓	✓	–	–
can edit on Passwords	✓	–	–	–
can delete on Connections	✓	✓	✓	–
can delete on DAGs	✓	✓	–	–
can delete on DAG Runs	✓	✓	–	–
can delete on Task Instances	✓	✓	–	–
can delete on Pools	✓	✓	–	–
can delete on Variables	✓	✓	–	–
can delete on XComs	✓	✓	–	–
can delete on Roles	✓	–	–	–
can delete on Users	✓	–	–	–
can delete on Event Listener	✓	✓	–	–
can create on Connections	✓	✓	✓	–
can create on DAGs	✓	✓	–	–
can create on DAG Runs	✓	✓	–	–
can create on Task Instances	✓	–	–	–
can create on Pools	✓	–	–	–
can create on Variables	✓	–	–	–
can create on XComs	✓	–	–	–
can create on Roles	✓	–	–	–
can create on Users	✓	–	–	–

Permission	Admin	DAG Manager	Operator	Public
can create on Event Listener	✓	✓	–	–
menu access on Connections	✓	✓	✓	–
menu access on DAG Runs	✓	✓	–	–
menu access on Task Instances	✓	✓	–	–
menu access on Audit Logs	✓	✓	–	–
menu access on Pools	✓	✓	–	–
menu access on Variables	✓	✓	–	–
menu access on XComs	✓	✓	–	–
menu access on Docs	✓	✓	✓	–
menu access on Kiteworks Secure MFT Server User Guide	✓	✓	✓	–
menu access on Documentation	✓	–	–	–
menu access on About	✓	✓	✓	–
menu access on Version	✓	✓	✓	–
menu access on Admin	✓	✓	✓	–
menu access on API Templates	✓	✓	✓	–
menu access on Jobs	✓	✓	–	–
menu access on SLA Misses	✓	✓	–	–
menu access on Task Reschedules	✓	✓	–	–
menu access on Browse	✓	✓	–	–
menu access on Create	✓	✓	–	–
menu access on DAG	✓	✓	–	–
menu access on Event Listener	✓	✓	–	–
menu access on Triggers	✓	–	–	–
menu access on Setup	✓	–	–	–
menu access on List Users	✓	–	–	–
menu access on Security	✓	–	–	–
menu access on List Roles	✓	–	–	–

Permission	Admin	DAG Manager	Operator	Public
menu access on User's Statistics	✓	–	–	–
menu access on Base Permissions	✓	–	–	–
menu access on Views/Menus	✓	–	–	–
menu access on Permission on Views/Menus	✓	–	–	–
menu access on Third Party Licenses	✓	–	–	–
can list on ApiTemplate	✓	✓	✓	–
can list on Connection	✓	✓	✓	–
can list on DiskManagementView	✓	–	–	–
can list on SslUploadView	✓	–	–	–
can list on AuthenticationSettingView	✓	–	–	–
can list on SmtpSettingView	✓	–	–	–
can list on SoftwareUpdateView	✓	–	–	–
can list on ActivateView	✓	–	–	–
can list on RemoteManagementView	✓	–	–	–
can list on LogsView	✓	–	–	–
can list on SystemRebootSettingView	✓	–	–	–
can list on DagsCreateView	✓	–	–	–
can list on NetworkSettingView	✓	–	–	–
can list on DevelopmentView	✓	–	–	–
can list on ThirdPartyLicensesView	✓	–	–	–
can download on ActivateView	✓	–	–	–
can add on ApiTemplate	✓	✓	✓	–
can add on AccellionDagEditorView	✓	✓	–	–
can version on VersionView	✓	✓	✓	–
can template access on AccellionAPI	✓	✓	✓	–
can import connections on AccellionAPI	✓	✓	✓	–
can access on AccellionAPI	✓	✓	–	–
can add disk on DiskManagementView	✓	–	–	–

Permission	Admin	DAG Manager	Operator	Public
can verify image on SoftwareUpdateView	✓	–	–	–
can run offline update on SoftwareUpdateView	✓	–	–	–
can get software update status on SoftwareUpdateView	✓	–	–	–
can upload image on SoftwareUpdateView	✓	–	–	–
can get release notes on SoftwareUpdateView	✓	–	–	–
can delete offline image on SoftwareUpdateView	✓	–	–	–
can admin access on AccellionAPI	✓	–	–	–
can get on MenuApi	✓	–	–	–
can cacertupload on SslUploadView	✓	–	–	–
can sslupload on SslUploadView	✓	–	–	–
can run post update on SoftwareUpdateView	✓	–	–	–
can check available storage on SoftwareUpdateView	✓	–	–	–
set failed on Task Instances	✓	✓	–	–
set retry on Task Instances	✓	✓	–	–
set running on Task Instances	✓	✓	–	–
set skipped on Task Instances	✓	✓	–	–
set success on Task Instances	✓	✓	–	–
clear on Task Instances	✓	✓	–	–
menu access on Notification Templates	✓	–	✓	–
can create on Notification Templates	✓	–	✓	–
can read on Notification Templates	✓	–	✓	–
can edit on Notification Templates	✓	–	✓	–
can delete on Notification Templates	✓	–	✓	–
can read on Connection Browsing	✓	–	–	–
can create on Patches	✓	–	–	–
can read on Patches	✓	–	–	–

Permission	Admin	DAG Manager	Operator	Public
can upload on Patches	✓	–	–	–
can delete on Patches	✓	–	–	–
can apply on Patches	✓	–	–	–
can revert on Patches	✓	–	–	–

Create and edit user accounts

When you create a user account, you also assign one or more roles to the account. For a list of list default roles and their permissions, refer to Create and edit roles.

To create a user account:

1. On the navigation bar, click Security > List Users.
2. On the toolbar, click Add a New Record.
3. Fill in the fields, and then click Save.
4. If you created an account for another user, send them the password and ask them to change it when the sign in to the server.

To edit a user account:

1. On the navigation bar, click Security > List Users.
2. In the row you want to edit, click Edit Record.
3. Edit the record as needed, and then click Save.

If you want to edit your account, but don't have access to the Security menu, you can still change your first and last name in your user profile settings. In the upper right corner of the screen, click your user name, and then click Profile. On the user information page, click Edit User. To make other changes, such as to your email address, contact a user who has administrative permission to edit all user accounts.

To delete a role:

In the row you want to delete, click Delete Record.

Manage connections

Create and edit connections

In your DAG you can define your job to perform any task with a data source, but first you need a connection to that data source.

To create a connection:

1. On the navigation bar, click Admin > Connections.
2. On the toolbar, click Add a New Record .
3. From the Conn Type list, select the connection you want to create, and then fill in the fields. If you select the Test Connection check box, after you save your settings a message will appear if there is a problem connecting to the server.

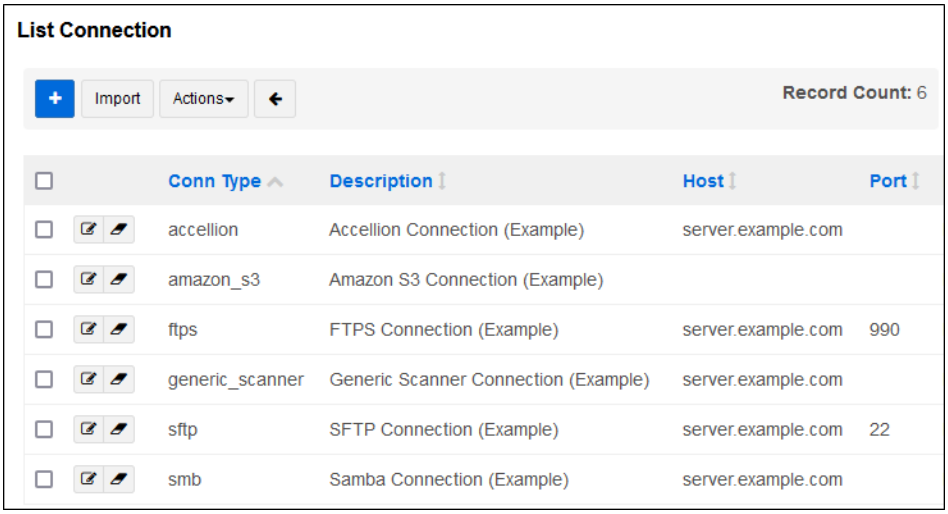
   Refer to Default connection types.
4. Click Save.

   Result: A unique ID (GUID) is generated to ensure the connection is unique.

To edit a connection:

1. In the row you want to edit, click Edit Record .

   Note: Passwords are encrypted for security. You won't see them on the page.
2. Edit the record as needed, and then click Save.



Default connection types

The following default connections are provided for you to customize.

- Kiteworks connection
- Amazon S3 connection
- Box connection
- FTPS connection
- Generic Scanner connection
- PGP connection
- SMB connection

Kiteworks connection

Table 1. Kiteworks connection properties

Name	Description (Optional)
Conn type	Kiteworks
Description (Optional)	A description of how you want to identify the connection.
Host/URL	The host name (domain name) or IP address of the server. **Example:** server.example.com
Username	The email address for connecting to the server.
Password/Passphrase	The password for connecting to the server.
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.

Amazon S3 connection

Table 2. Amazon S3 connection properties

Name	Description (Optional)
Conn type	Amazon S3
Description (Optional)	A description of how you want to identify the connection.
Access key ID	The Amazon S3 access key provided for programmatically authenticating to the server. Access keys consist of an access key ID and secret access key, which are used to sign programmatic requests that you make to AWS. You create access key pairs from the AWS Management Console.
Secret access key	The secret key that was created with the access key.
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
Delimiter	The character you want to use to group keys. For S3 buckets that mimic normal file systems, the delimiter is the / character. For more information, refer to Listing Keys Hierarchically Using a Prefix and Delimiter in the Amazon S3 documentation.
Amazon S3 region	The geographical region where Amazon S3 stores the buckets created on the AWS server. **Example:** ap-southeast-1
Server-side encryption	Protect data uploads using server-side encryption with Amazon S3-managed encryption keys (SSE-S3). Amazon S3 server-side encryption uses one of the strongest block ciphers available to encrypt your data -- 256-bit Advanced Encryption Standard (AES-256) To enable server-side encryption using an Amazon S3-managed key, enter AES256. For more information, refer to Protecting data using server-side encryption with Amazon S3-managed encryption keys (SSE-S3) in the Amazon S3 documentation.

Box connection

To configure Box connections, you need to provide access to the Box files through a custom application developed in the Box Developer Console.

Table 3. Box connection properties

Name	Description (Optional)
Conn type	Box
Description (Optional)	A description of how you want to identify the connection.
Client ID	The client ID of the application that is requesting to authenticate the user. To get the client ID for your application, sign in to your Box developer and open the app. In the app console, click the Edit Application link, and then click the Configuration link. In the Credentials section, the client ID is listed along with its associated client secret.
Client secret	The client secret associated with the client ID.
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
App User ID	To get the user ID for your application, sign in to your Box developer and open the app. In the app console, click the General Settings link. In the App Info section, the User ID is the app user ID.

FTPS connection

Table 4. FTPS connection properties

Name	Description (Optional)
Conn type	FTPS
Description (Optional)	A description of how you want to identify the connection.
Host/URL	The host name (domain name) or IP address of the server. **Example:** server.example.com
Username	The email address for connecting to the server.
Password/Passphrase	The password for connecting to the server. Alternatively, you can use a private key to authenticate. If using a private key to authenticate, enter the passphrase used to protect the private key.
Port	The default port for FTPS is 990. FTPS connections use implicit FTPS. An SSL connection is immediately established via port 990 before login or file transfer can begin. If the recipient fails to comply with the security request, the server immediately drops the connection.
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
Passive/Active	Select one of the following options: • Passive: The client connects to the host, and the host replies with a random port that the client should connect to and to initiate the data transfer. The client initiates the data transfer connection. • Active: The client connects to the host and tells the host to connect to the client on a random specified port to initiate the data transfer. The host initiates the data transfer connection. For assistance choosing the appropriate setting, contact the FTPS server administrator for the configuration.

Generic Scanner connection

The generic scanner connection enables you to connect to any third-party Content Disarm and Reconstruction (CDR) scanner using an API template. You configure general connection settings and then select an API template to process the requests of the corresponding CDR scanner.

CDR scanners perform CDR file processing on files being uploaded to the destination server. Incoming files are treated as suspicious and undergo deep threat scans to transform them into safe, neutralized, and harmless copies. Files that can't be transformed are perceived as threats and removed before upload to the destination server.

Table 5. Generic Scanner connection properties

Name	Description (Optional)
Conn type	Generic Scanner
Description (Optional)	A description of how you want to identify the connection.
Host/URL	The host name (domain name) or IP address of the scan server. For example, server.example.com.
Login	The email address for connecting to the scan server.
Password/API key	The password or API key provided by your administrator for authenticating to the server
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
API template	Select a template to use or download one and use it to create a new, custom template. You create API templates using Postman. Refer to Create API templates using Postman.

PGP connection

Each PGP connection requires its own public and private key pair. A DAG operator can contain up to two PGP connections. For example, to encrypt and decrypt files you will need one connection to enter into the "PGP encryption connection ID" field when configuring the DAG operator. To sign and then verify files, you will need to create a second connection to enter into the "PGP signature connection ID" field for the DAG operator.

Table 6. PGP connection properties

Name	Description (Optional)
Conn type	PGP
Description (Optional)	A description of how you want to identify the connection.
Password/Passphrase	The password or passphrase used for unlocking the private key.
Test credentials	Verify that the private key, public key, and password are valid for signing and decrypting files.
Private key	Enter or paste the private key you want to use to decrypt files. The private key will be encrypted and require the passphrase to unlock it.
Public key	Enter or paste the public key you want to use to encrypt files.

SFTP connection

Table 7. SFTP connection properties

Name	Description (Optional)
Conn type	SFTP
Description (Optional)	A description of how you want to identify the connection.
Host/URL	The host name (domain name) or IP address of the server. **Example:** server.example.com
Username	The email address for connecting to the server.
Password/Passphrase	The password for connecting to the server. Alternatively, you can use a private key to authenticate. If using a private key to authenticate, enter the passphrase used to protect the private key.
Port	The default port for SFTP is 22.

Table 7. SFTP connection properties (continued)

Name	Description (Optional)
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
Private key	Instead of using a password, use a private key to connect to the server.

SMB connection

Table 8. SMB connection properties

Name	Description (Optional)
Conn type	SMB
Description (Optional)	A description of how you want to identify the connection.
Host/URL	The host name (domain name) or IP address of the server. **Example:** server.example.com
Schema	The parent directory for accessing files on the server.
Login	The email address for connecting to the server.
Password/Passphrase	The password for connecting to the server.
Port	The default port for SMB, either 139 or 445.
Test connection	Test the connection to the server. **Note:** When you select this check box, the connection is tested after you save the connection settings. Fill in the connection settings, select the Test Connection check box, and then click Save. A message only appears if there is a problem connecting to the server.
Without IP parameter	If there is an issue connecting to the SMB server, connect to the server without passing the "-I <ip_address>" parameter.

Import connections

If you need to create a lot of connections, you can save time by creating them externally and then importing them to the server. You create a JSON file containing the connection parameters, run a script to encrypt the file, and then import the file to the MFT server. Imported connections are assigned unique IDs (GUIDs) to ensure that they are unique.

The following files contain the information you need to get started. To get the zip file, click Admin > Connections > Import, and then click the Download Client Side Script button.

- example_connection.json - A JSON file containing example parameters for all supported connection types.
- requirements.txt - A file listing Python requirements.
- connection_encryption.md - A file containing documentation for running the "encrypt_ connections.py" script used to encrypt the JSON file prior to import.
- encrypt_connections.py - The Python script for encrypting the JSON file.

When you import your encrypted JSON file, it's stored temporarily on the server. Once the connections are created, the file is deleted from the server.

Create the JSON file containing connection parameters

When creating the JSON file, refer to the example_connection.json file. It contains examples of all supported connection types. Here's a sample of the code you'll see in the file.

Examples from the JSON file

```
{
    "connections": [
      {
        "desc": "Kiteworks Test Conn",
        "conn_type": "kiteworks",
        "host": "https://example.kiteworks.com",
        "login": "abc@example.com",
        "password": "ExamplePassword",
        "schema": null,
        "port": null,
        "extra" : null
      },
      {
        "desc": "SFTP Test",
        "conn_type": "sftp",
        "host": "10.254.12.90",
        "login": "abc@acc.com",
        "password": "TestPassword",
        "schema": null,
        "port": 22,
        "extra" : null
      },
      {
        "desc": "string",
        "conn_type": "smb",
        "host": "qnap2.sd.dev",
        "login": "test",
        "password": "testPassword",
        "schema": "smb",
```

```
        "port": null,
        "extra" : null
      }
    ]
  }
```

Encrypt the JSON file using the Python script

For prerequisites and instructions on using the Python script, refer to the "requirements.txt" and "connection_encryption.md" files.

Run the "encrypt_connections.py" script to encrypt the JSON file. Provide a secure passphrase consisting of 8-16 characters, with at least one lowercase, one uppercase, and one numeric character. You'll need to enter this passphrase when importing the file.

Import the encrypted JSON file to the server

Once you've encrypted the JSON file containing the connection parameters, you're ready to import the connections to the server.

To import the encrypted JSON file:

1. On the navigation bar, click Admin > Connections.
2. On the toolbar, click Import.
3. Select the encrypted JSON file, and then click Open.
4. Enter the passphrase for decrypting the file.
5. Click Import.

   Result: The imported connections are added to the connections list.

Create API templates using Postman

API templates are provided for interacting with the following third-party CDR scanners:

- odix
- OPSWAT MetaDefender
- ReSec
- Sasa GateScanner

Each template contains preconfigured requests that make it possible for you to use it without having to manually configure the requests. You can also add support for other scanners by creating new, custom API templates.

To create a template, download an existing template on which to base the new one, edit the template using Postman, and then upload the new template to the templates list on the MFT server. You can download the Postman app here.

Download API templates from the MFT server

To download an API template from the MFT server:

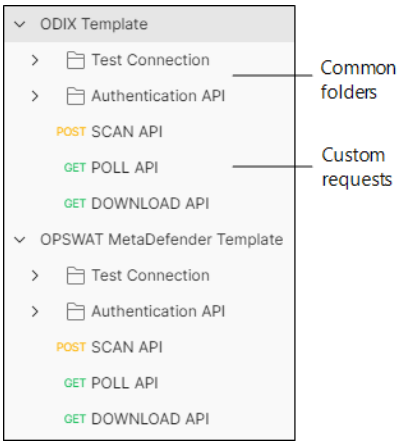
1. On the navigation bar, click Admin > API Templates.
2. In the template row, click Download .

Structure of API templates in Postman

Each template consists of a Postman Collection of predefined requests. The templates are intended to work with asynchronous scanning, although synchronous scanning should work as long as the request does not time out.

Each collection contains Test Connection and Authentication API folders, followed by a set of top-level requests based on scanner functionality. Files are scanned in parallel, with requests being called sequentially for each file.

You can add, rename, and customize requests as needed to reflect the scanner's service structure. You must have at least one top-level request in a collection.



Variable examples

Each template contains variables that will be defined when you create connections and configuring operators in DAGs. In custom templates, you can also hard-code parameters.

Table 1. Examples of variables used in Generic Scanner connections

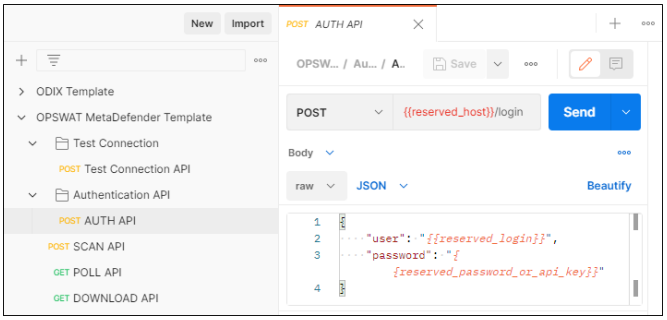
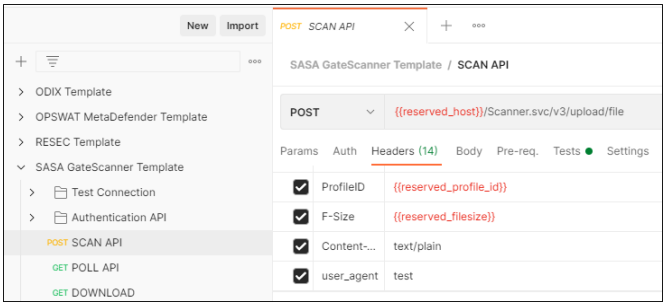
Variable name	Generic Scanner connection source	Example
{{reserved_host}}	Host	
{{reserved_login}}	Login	
{{reserved_password_or_api_key}}	Password/API key	

Table 2. Example of variable used in Generic Scanner operators in DAGs

Variable name	Generic Scanner operator source	Example
{{reserved_profile_id}}	Scan profile	

Variable definitions

In custom templates, you can set parameters for headers, query, path, and body (form-data, url-encoded-form-data, raw, binary file).

- The key for these parameters must be plain text, but the value can be plain text or a variable denoted as {{variable}}.
- Variables can have any name as long as they are wrapped in double curly braces and use the namespace of the API template. When scanning a file, the variables will be substituted with their values before running a request.
- You can set the value of the variable in a connection or parse it from the response of a previous request.

The following reserved variables are exceptions to this behavior.

Table 3. Variable definitions

Variable name	Variable type	Rules
{{reserved_file}}	File	Can: • Use these variables in the template. Cannot: • Parse these variables from a response. • Define the values of these variables in connections.
{{reserved_ filename}}	Filename	
{{reserved_ filesize}}	Filesize (bytes)	
{{reserved_ attempt}}	Attempt number (per request)	
{{reserved_raw_ response}}	Raw response	Can: • Must use these exact variable to parse their corresponding values. Cannot: • Define the values of these variables in connections.
{{reserved_state}}	Scanning state	
{{reserved_result}}	Scan result	
{{reserved_ profile_id}}	Profile ID	Can: • Use this variable in the template. • Define the value a DAG operator. Cannot: • Define the value of this variable in a connections.

Variable response parsing

In custom templates, you must define how to set the value of a variable from a previous request, using the predefined result names. This is useful when you need to pass values from one request to the next, such as a job ID.

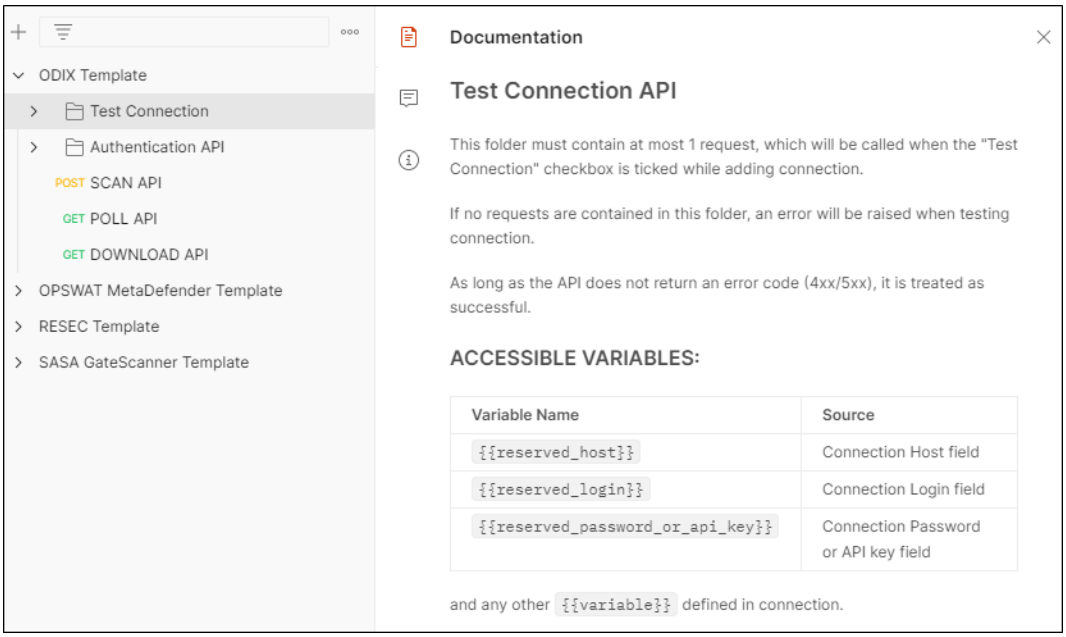
Example of how to parse a response

```
//Response structure of result API:
{
  "metadata": {
    "job_id": 12345
  },
  files: [{"result": "scanning"}]
}

//Response parser (JSON format):
{
    "{{job_id}}": "metadata.job_id",
    "{{reserved_scan_state}}": "files.[0].result"
}
```

API documentation

For each collection, documentation is provided to help you configure requests. . It contains more detailed information on response parsing, flow control using variables, and definitions of reserved variables.



Import API templates to Postman

To import an API template into Postman:

1. In Postman, click File > Import.
2. Upload the file, and then click Import.

Edit API templates in Postman

To edit an API template in Postman:

1. Create or configure requests as needed. When creating a request, you can set the HTTP verb, the request URL, query or path parameters, headers and the body. In the Postman documentation, refer to Building Requests.
2. Save the request. Once you are finished editing the collection, export it as a new API template.

Export API templates from Postman

To export an API template from Postman:

1. Right-click the template name, and then click Export.
2. Select Collection v2.1, and then click Export.

Import API templates to the MFT server

To import an API template to the MFT server:

1. On the navigation bar, click Admin > API Templates.
2. On the toolbar, click Add a New Record .
3. Select the template file, and then click Upload API Template.

   Result: The template is added to the template list and is available for you to select when creating a connection.

Manage workflows (DAGs)

Create and edit event listeners

You can create event listeners to monitor Kiteworks folders and start any workflow based on when events occur in those folders. For example, when files are added to an Kiteworks folder, trigger a DAG that downloads files from an SFTP server.

Examples of events are when files are added, moved, viewed, or deleted from Kiteworks folders.

Once you create an event listener, you can select it for use in a DAG that runs a workflow.

Create event listeners

To create an event listener:

1. On the navigation bar, click Create > Event Listener.
2. On the Create Event Listener page, fill in the properties. For a description of each form property, refer to Event listener properties
3. When finished, click Submit.

   Result: The new event listener is added to the Event Listeners view page. It may take a moment for it to be added. You can refresh the browser page to update the Event Listener list.

Edit event listeners

To edit an event listener:

1. On the navigation bar, click Event Listeners.
2. In the row of the Event Listener you want to edit, click Edit .
3. Edit the properties, and then click Save.

Event listener properties

Table 1. Event listener properties

Property	Description (Optional)
Event Listener ID	Enter a name for the event listener. A number will be automatically appended to the name to ensure that the name is unique. **Requirement:** The name must start with a character and contain only letters, numbers, and underscores.
Enable Event Listener	Activates the event listener.
Owner	By default, your user name is used. You can enter any name in this metadata field, such as the name of the person you want to be responsible for the item.
Tags	You can use tags to filter the list of event listeners on the Event Listeners view page. Enter one or more tag values separated by commas.
Connection ID	Select the connection ID of the Kiteworks server where the event will occur. If you need to create a connection ID, refer to Create and edit connections.
Kiteworks folder link	The permalink to the Kiteworks source folder you want the listener to watch. Once the event occurs in this folder, the action will trigger any DAGs using this listener. To get a folder or file permalink, click the browse button and select the folder or file path.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Events	Select the events you want to trigger a DAG. Some events, such as "file changes" are groups that add a series of events related to the group. For example, add file, copy file, move file, and so on. After adding a group, you can delete the events you don't want.
DAGs	Select the DAGs you want the event listener to trigger.

Table 1. Event listener properties (continued)

Property	Description (Optional)
Event Listener filter	Use a regular expression (also known as regex or regexp) to get a subset of folder paths that match only those matching the pattern. The regex applies to the entire file path, without any trailing slashes. so, with an example the file /a/b/c/file.txt, the regex filter will be applied to /a/b/c only. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w*$ - Match all folder paths that start with "test" and containing only alphanumeric characters. • ^\w*(test)$ - Match all folder names that end with "test" and containing only alphanumeric characters.
Delay between checks	The number of seconds you want the event listener to wait before triggering a DAG. For example, if you specify 20 seconds as the time buffer, and five upload events occur during that 20 second period, the event listener will trigger the associated DAG only one time because all events occurred within the 20 second period.

Create and edit DAGs

You can create workflows (DAGs) to transfer files between Kiteworks servers, or to and from file shares on other servers such as SFTP and SMB servers.

When building a workflow, keep the following information in mind.

- To save time, you can arrange and connect operators (tasks) before configuring them. Based on their dependencies, some operator properties will get filled in automatically for you.
- When you connect operators, connection lines in red indicate that there are additional properties you need to configure to successfully connect the operators.
- At a minimum add one download operator and one upload operator. Additional operators are optional. Follow this sequence and refer to Example workflows.
  - File List operator (optional)

    Filter a list of files before downloading them from a server. Use this operator when you want to filter the list to download only specific files or file types.

    Only one File List operator can be used in a workflow.
  - Download operator

    Download files from a server. Files are downloaded to an encrypted temporary folder on the server to be picked up and uploaded to the destination folder. Once the workflow completes, the files are deleted from the temporary folder.

    Only one Download operator can be used in a workflow.
  - Scan operator (optional)

    Use CDR to scan files before uploading them to a server.
  - Upload operator

    Upload files to a server. You can add as many upload operators as you want.
  - Define Action on Source operator (optional)

    Used with the Action operator, indicate that you want to perform an action on the source files. For example, you want to delete the files from the source server after uploading them.

    Only one Define Action on Source operator can be used in a workflow.

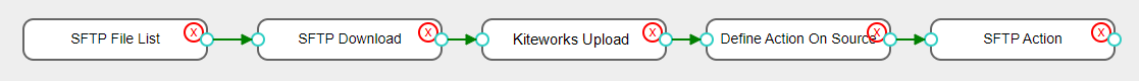
    Exception: This operator can't be used with Box operators.
  - Action operator (optional)

    Depending on the operator you placed before it, does the following:
    - If the previous operator is a Define Action on Source operator, performs an action you specify on the files on the source server.
    - If the previous operator is an upload operator, performs the action on the files being uploaded to the destination server.

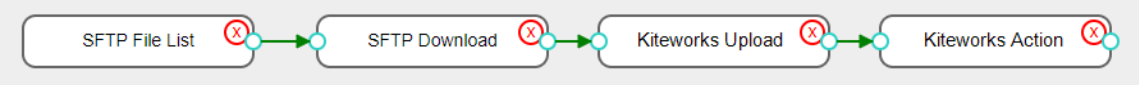
    Exception: Action operators can't be used with Box operators.

Example workflows

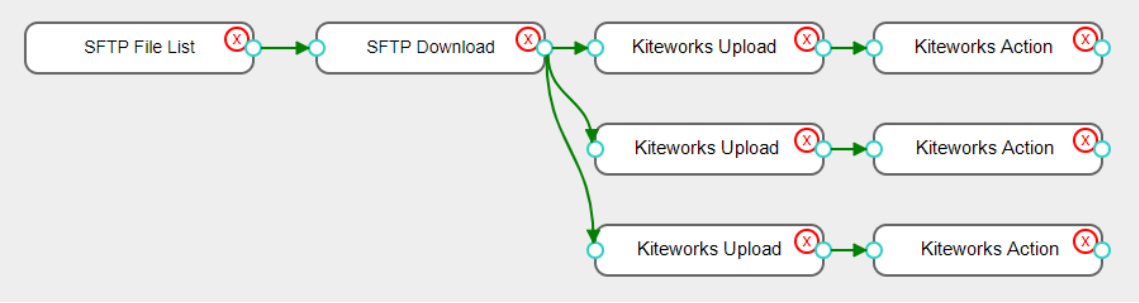
Example 1: Get a subset of files from a folder on an SFTP server, download the files, upload the files to a folder on an Kiteworks server, and then perform an action on the source files (for example, delete the files from the SFTP server).



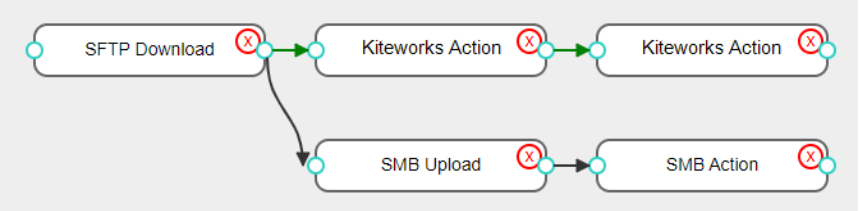
Example 2: Get a subset of files from a folder on an SFTP server, download the files, upload the files to a folder on an Kiteworks server, and then perform an action on the uploaded files (for example, rename the files on the Kiteworks server).



Example 3: Get a subset of files from a folder on an SFTP server, download the files, upload the files to folders on multiple Kiteworks servers, and then perform an action on the files that were uploaded to those servers (for example, rename the files on the Kiteworks servers).



Example 4: Get all files from a folder on an SFTP server, download the files, upload the files to folders on Kiteworks and SMB servers, and then perform an action on the files that were uploaded to those servers (for example, rename the files on the Kiteworks and SMB servers).



Example 5: Get a list of email messages designated for archival and download the messages as zip files. Upload the zip files to an SFTP server and, in a parallel process, extract the .eml messages and attachments and send them to one or more destination addresses.

Create DAGs

Prerequisites:

- Create connection IDs for the source and destination servers. You'll need to select them when you create the workflow. Refer to Create and edit connections.
- If you want an event to trigger the workflow, create an event listener for the event. You'll need to specify the event listener when you create the workflow. Refer to Create and edit event listeners.

To create a DAG:

1. On the navigation bar, click Create DAG.
2. On the Form tab, fill in the form properties. For a description of each form property, refer to DAG form properties
3. In the editor, drag the operators you want from the palette to the editor to build a workflow. For a description of each operator and its properties, refer to DAG form operators.
4. When finished, click Submit.

   Result: The new DAG is added to the DAGs view page. It may take a moment for the scheduler to create the DAG. You can refresh the browser page to update the DAG list.

Edit DAGs

You can edit a DAG to rename it, update its form and operator properties, and also update the structure of the tasks in the workflow.

Caution: If you rearrange or delete tasks in a workflow after running the DAG, the historical information of those tasks disappears from Airflow. If you already ran the workflow and want to preserve the history of original tasks, create a new workflow instead.

To edit a DAG:

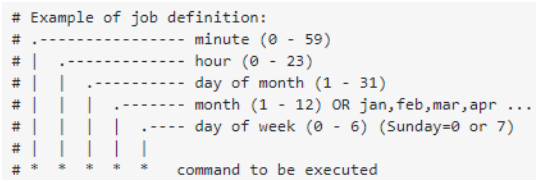
1. On the navigation bar, click DAGs to open the DAGs view page.
2. In the row of the DAG you want to edit, click Edit .
3. Edit the DAG as needed, and then click Submit.

DAG form properties

Table 1. DAG form properties

Property	Description (Optional)
DAG ID	Enter a name for the DAG. A number will be appended to the name to ensure that the name is unique. **Requirement:** The name must start with a character and contain only letters, numbers, and underscores.
Owner	By default, your user name is used. You can enter any name in this metadata field, such as the name of the person you want to be responsible for the item.
Access control	Select the roles you want to give access to the DAG. By default, users with the Admin and DAG Manager roles can access all DAGs. If your system includes custom roles with permission to access DAGs, you'll see them listed here as well. **Requirement:** To select from a list of roles, your role needs the following permissions: "DagModelView" and "can list on DagRunModelView". You can ask your administrator to assign these permissions to your role.
Tags	You can use tags to filter the list of DAGs on the DAGs view page. Enter one or more tag values separated by commas.
Max. Active Runs	The maximum number of running task instances the DAG can have before queuing tasks.
DAG run callback	The DAG you want to run based on the success or failure of the current DAG. For example, if the current workflow is successful you may want to automatically run the next DAG in the business operation workflow. If the flow fails, you may want to run a DAG to remedy that failure. • On Success The DAG ID (name of the DAG) you want to run when the current DAG is successful. • On Failure The DAG ID (name of the DAG) you want to run if the current DAG fails. **Tip:** To get the DAG IDs you want without closing the DAG creation page, right-click the DAGs menu and open the link in another tab or window. From the DAGs view page, copy each ID you want and paste it into the DAG creation form.
Event trigger	If you want an event to trigger the workflow, select the event listener for the event. Refer to Create and edit event listeners.

Table 1. DAG form properties (continued)

<table>
<tr><th>Property</th><th>Description (Optional)</th></tr>
<tr><td>Schedule</td><td>Enter the date, interval, and time you want the scheduler to trigger the DAG. If you enter a past date, when the DAG runs it will backfill runs to the default date.
<table>
<tr><th>Interval</th><th>Description</th></tr>
<tr><td>None</td><td>Don't schedule. For example, you haven't determined the schedule yet or you plan to trigger the DAG manually.</td></tr>
<tr><td>Once</td><td>Run only once on the date you specify.
If you schedule a past or current date, the DAG triggers moment you turn it on.
If you schedule a future date and turn on the DAG, the DAG triggers on that date at midnight.</td></tr>
<tr><td>Minutes</td><td>Run every number of minutes, up to 60 minutes.</td></tr>
<tr><td>Hourly</td><td>Run once an hour at the number of minutes you specify, up to 60 minutes.</td></tr>
<tr><td>Daily</td><td>Run once a day at the time you specify.</td></tr>
<tr><td>Weekly</td><td>Run one or more days each week at the time you specify.</td></tr>
<tr><td>Monthly</td><td>Run on the day, month, and time you specify.</td></tr>
<tr><td>Advanced</td><td>Use a cron expression to schedule the DAG runs.
Cron jobs are scheduled at recurring intervals. You can schedule a job to run multiple times a day, or on specific days and months.
A schedule is described using unix-cron string format (* * * * *)—a set of five values in a line indicating when the job should be executed.
 

Example expressions:
Run every minute: * * * * *
Run every Saturday at 23:45 (11:45 PM): 45 23 * * 6
Run every Monday at 09:00: 0 9 * * 1
Additionally, the # (hash) character is supported in cron expressions. For example, the expression “0 9 * * Mon#2” will run the DAG at 9 AM on every second Monday of the month.</td></tr>
</table></td></tr>
</table>

Table 1. DAG form properties (continued)

Property	Description (Optional)
Apply timeout	Prevent DAGs from getting stuck in the running process. DAGs that timeout are skipped. The timeout interval is the same as the schedule interval. The log associated with the task lists files that were skipped and the reason for the action.
Retry	Retry running an operator task if a previous run fails. • Amount The number of times you want the scheduler to retry running the task. • Delay How much time should pass between retries.
Email notifications	Select the templates you want to use for reporting the success, partial success, and any error details associated with the DAG run. To create email notification templates, see Create email notifications.
Email recipients	Enter the email address of the recipients you want to receive the email notifications.

DAG form operators

Source operators

Use source operators to get a list of files and download them from servers.

Kiteworks File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Select source folder	The permalinks to folders and files you want to get before downloading them from the Kiteworks server. You can enter multiple folder and file links. **Example:** <folder link 1>, <folder link 2>, <file link 1> To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression (file name)	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.

Property	Description
Regular expression (folder path)	Use a regular expression (also known as regex or regexp) to get a subset of folder paths that match only those matching the pattern. The regex applies to the entire file path, without any trailing slashes. so, with an example the file /a/b/c/file.txt, the regex filter will be applied to /a/b/c only. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w*$ - Match all folder paths that start with "test" and containing only alphanumeric characters. • ^\w*(test)$ - Match all folder names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

Kiteworks Download

Download files from a server to a temporary folder on the server.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Source link of folder/file	The permalink to the Kiteworks folder from which to download items. To get a folder or file permalink, click the browse button and select the folder or file path.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

Kiteworks Privileged Download

Download files from a Kiteworks folder with privileged access, allowing download of files marked "pending scanning".

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Source link of folder/file	The permalink to the Kiteworks folder from which to download items. To get a folder or file permalink, click the browse button and select the folder or file path.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

Kiteworks Archived Mail List

Used in conjunction with the Kiteworks mail connector feature, this operator provides a list of email messages flagged for archival. The list includes only messages sent since the previous DAG run.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	The connection ID of the source Kiteworks server.

Kiteworks Archived Mail Download

Used in conjunction with the Kiteworks mail connector feature and the Kiteworks Archived Mail List operator, this operator will download email messages and attachments designated for archive.

Messages are downloaded as zip files containing the .eml file and any attachments.

Rule: Only one Mail Download operator can be used in a workflow. This operator must be preceded by the Kiteworks Archived Mail List operator.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Include attachments	Download email attachments.
Allow partial success	When downloading messages returned by the Kiteworks Archived Mail List operator, ignore messages that have encountered issues, such as exceeding file size limits. This allows the task to proceed to success. The log associated with the task lists messages that were skipped and the reason for the action. If you don't select this check box, no messages will be skipped. If an issue occurs with any messages, no messages will be downloaded and the task will fail.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

SFTP File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source folder	The path to the SFTP remote directory from which to download items. Click the browse button to access the source and select the folder path.
Regular expression filter	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

SFTP Download

Download files from a server to a temporary folder on the server.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source path	The path to the SFTP remote directory from which to download items. Click the browse button to access the source and select the folder path.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

SMB File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source folder	The path to the SMB (Server Message Block) shared folder from which to download items. Click the browse button to access the source and select the folder path.
Regular expression (file name)	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.
Regular expression (folder path)	Use a regular expression (also known as regex or regexp) to get a subset of folder paths that match only those matching the pattern. The regex applies to the entire file path, without any trailing slashes. so, with an example the file /a/b/c/file.txt, the regex filter will be applied to /a/b/c only. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w*$ - Match all folder paths that start with "test" and containing only alphanumeric characters. • ^\w*(test)$ - Match all folder names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

SMB Download

Download files from a server to a temporary folder on the server.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source path	The path to the SMB (Server Message Block) shared folder from which to download items. Click the browse button to access the source and select the folder path.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

S3 File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Prefix	The path to the files you want to download. **Examples:** You have files in the following places: 1. folder1/file1.txt. 2. folder2/file2.txt 3. folder1/file3.txt To get all files, you can use an empty prefix ("") or a longer common one such as "folder". To get files in locations 1 and 3, use the prefix "folder1/" or "folder1, and also select the "Include nested folders/files" check box. To get files in only location 1, use the prefix "folder1/file1".
S3 bucket	The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
Regular expression filter	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

S3 Download

Download files from a server to a temporary folder on the server.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Prefix	The prefix to the files you want to download. **Examples:** You have files in the following places: 1. folder1/file1.txt. 2. folder2/file2.txt 3. folder1/file3.txt To get all files, you can use an empty prefix ("") or a longer common one such as "folder". To get files in locations 1 and 3, use the prefix "folder1/" or "folder1, and also select the "Include nested folders/files" check box. To get files in only location 1, use the prefix "folder1/file1".
S3 bucket	The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

FTPS File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

FTPS operators transfer files in binary mode only.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source folder	The absolute path to the FTPS directory from which to download items. Click the browse button to access the source and select the folder path. **Example:** /Folder/Test123
Regular expression filter	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

FTPS Download

Download files from a server to a temporary folder on the server.

FTPS operators transfer files in binary mode only.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Source path	The absolute path to the FTPS directory from which to download items. Click the browse button to access the source and select the folder path. **Example:** /Folder/Test123 **Tip:** To fill in this value automatically, connect this node to the preceding node.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder. This option is not available if an FTPS File List operator precedes the FTPS Download operator and you've connected the nodes. This is because you set the nest option at the root of the workflow and the FTPS Download operator inherits the setting from the preceding FTPS File List operator.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.

Box File List

Get a list of files before downloading them from a server. Use this operator when you want to filter the file list to download only specified files or file types.

Rule: Only one File List operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Folder ID	The ID of the Box folder from which to download items. Click the browse button to access the source and select the folder ID.
Regular expression filter	Use a regular expression (also known as regex or regexp) to get a subset of file names that match only those matching the pattern. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html. **Examples:** • ^(test)\w* - Match all file names that start with "test" and containing only alphanumeric characters. • \w*(test)$ - Match all file names that end with "test" and containing only alphanumeric characters.
File extension filter	Download only file types you specify. To download all file types, leave this field blank. When entering file extensions, dot characters are not required. For example, you can enter ".docx, .pptx, .xslx" or "docx, pptx, xslx". When using regular expression and file extension filters together, you filter only those files that match both rules. For example, match all files that start with "test" and have the ".pdf" file extension.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.

Box Download

Download files from a server to a temporary folder on the server.

Automatically generates a SHA-1 hash code with each downloaded file and lists the code in the corresponding log.

Rule: Only one Download operator can be used in a workflow.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Folder ID	The ID of the Box folder from which to download items. Click the browse button to access the source and select the folder ID. **Alternative:** If you preceded this node with a Box File List node, you can fill this value in automatically. Just connect this node to the preceding node.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.

Destination operators

Use destination operators to upload files to servers. For example, you want to get an updated list of files daily and upload them to multiple destinations.

Kiteworks Upload

Upload files to a server.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Destination folder link	The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.

Kiteworks Sanitized Upload

Use this operator to upload scanned files to Kiteworks if the file has been modified since downloading, for example, by a CDR scan.

Rule: Must be preceded by the Kiteworks Scan Status operator.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Scan status task ID	The type of scanner configured. Select AV, DLP, or Both.

SFTP Upload

Upload files to a server.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Select destination folder	The path to the SFTP remote directory where you want to upload items. Click the browse button to access the source and select the folder path.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

SMB Upload

Upload files to a server.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Select destination folder	The path to the SMB (Server Message Block) shared folder where you want to upload items. Click the browse button to access the source and select the folder path.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

S3 Upload

Upload files to a server.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Destination prefix	The prefix you want to add to uploaded file names. For example, if a file is named "file1/txt", adding the prefix "folder1" results in the file being renamed to "folder1file1.txt". Additional examples: • Prefix = "folder1/". File renamed to "folder1/file1.txt". • Prefix = "folder1/folder2". File renamed to "folder1/folder2file1.txt". • Prefix = "anyotherprefix". File renamed to "anyotherprefixfile1.txt". • Any delimiters in the prefix (/) are interpreted as folder breaks.
S3 bucket	The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
On name conflict	Specify what to do if a file with the same name exists in the destination folder. • Keep both - If versioning is enabled for the files on the S3 bucket, creates a new version in the files. Otherwise, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx". • Overwrite - Replaces the file. • Ignore - Excludes the file from the transfer process.

FTPS Upload

Upload files to a server.

FTPS operators transfer files in binary mode only.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Select destination folder	The path to the FTPS directory where you want to upload items. **Example:** folder1/folder2
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

Box Upload

Upload files to a server.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File download/scan task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Destination Folder ID	The ID of the Box folder from which to download items. The ID for any folder can be determined by visiting this folder in the web application and copying the ID from the URL. For example, for the URL https://*.app.box.com/folder/123 the folder_id is 123. **Tip:** If you preceded this node with a Box File List node, you can fill this value in automatically. Just connect this node to the preceding node.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.

Scan operators

The generic scanner connection enables you to connect to any third-party Content Disarm and Reconstruction (CDR) scanner using an API template. You configure general connection settings and then select an API template to process the requests of the corresponding CDR scanner.

CDR scanners perform CDR file processing on files being uploaded to the destination server. Incoming files are treated as suspicious and undergo deep threat scans to transform them into safe, neutralized, and harmless copies. Files that can't be transformed are perceived as threats and removed before upload to the destination server.

Generic Scanner

Scan files using third-party CDR scanners such as odix, OPSWAT MetaDefender, ReSec, and Sasa GateScanner.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Previous download task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Include checksum	Generate a highly secure SHA3-256 hash code with each downloaded file and list the code in the corresponding log.
Allow partial success	When downloading files returned by the File List operator, ignore files that are being scanned or have encountered issues such as having been placed in quarantine. This allows the task to proceed to success. The log associated with the task lists files that were skipped and the reason for the action. If you don't select this check box, no files will be skipped. If an issue occurs with any files, no files will be downloaded and the task will fail.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.

Kiteworks Scan Status

Submits the scan status results from generic scanners to Kiteworks.

Rule: This must be preceded by the Generic Scanner operator.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the source server.
Scanner type	The type of scanner configured. Select AV, DLP, or Both.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

Action operators

Use action operators to perform actions on files uploaded to server.

Action operators can also be used with the "Define On Source" operator to perform the action on the source server. In this scenario, you insert the Define On Action operator before the action operator.

Exception: Action operators can't be used with Box operators to perform actions on Box source and destination files.

Kiteworks Action

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Action	
-- Copy	Copy the files to a different folder on the server. • Destination folder link The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
-- Move	Move the files to a folder on the same or different server. • Destination folder/file link The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Delete	Delete the files from the folder on the server.

Property	Description
-- Rename	Rename the files before they get uploaded to the destination server. • Prefix Add a prefix to each file name. • Suffix Add a suffix to each file name. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

SFTP Action

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Action	
-- Copy	Copy the files to a different folder on the server. • Select destination folder The path to the folder where you want to copy the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Move	Move the files to a folder on the same or different server. • Select destination folder The path to the folder where you want to move the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Delete	Delete the files from the folder on the server.

Property	Description
-- Rename	Rename the files before they get uploaded to the destination server. • Prefix Add a prefix to each file name. • Suffix Add a suffix to each file name. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

Define Action on Source

Use this operator to perform an action on the files residing on the source server.

Exception: This operator can't be used with Box operators.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Source Task ID	Select the connection ID of the source server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

SMB Action

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Action	
-- Copy	Copy the files to a different folder on the server. • Select destination folder The path to the folder where you want to copy the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Move	Move the files to a folder on the same or different server. • Select destination folder The path to the folder where you want to move the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Delete	Delete the files from the folder on the server.

Property	Description
-- Rename	Rename the files before they get uploaded to the destination server. • Prefix Add a prefix to each file name. • Suffix Add a suffix to each file name. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

S3 Action

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Action	
-- Copy	Copy the files to a different folder on the server. • Destination prefix The prefix to where you want to copy the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx". • S3 bucket The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket

Property	Description
-- Move	Move the files to a folder on the same or different server. • Destination prefix The prefix to where you want to move the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx". • S3 bucket The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
-- Delete	Delete the files from the folder on the server.
-- Rename	Rename the files before they get uploaded to the destination server. • Prefix Add a prefix to each file name. • Suffix Add a suffix to each file name. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

FTPS Action

FTPS operators transfer files in binary mode only.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Action	
-- Move	Move the files to a folder on the same or different server. • Select destination folder The path to the folder where you want to move the items. • Flatten When uploading files in subfolders, upload the files without keeping the subfolder structure. This is useful if you needed to download files in nested folders, but don't want to recreate the folder structure in the destination. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".
-- Delete	Delete the files from the folder on the server.
-- Rename	Rename the files before they get uploaded to the destination server. • Prefix Add a prefix to each file name. • Suffix Add a suffix to each file name. • On name conflict Specify what to do if a file with the same name exists in the destination folder. You can create a copy of the file, overwrite the file, or exclude it from the transfer process. If you create a file copy, a number is appended to the file name to make it unique. For example, "Annual Report (1).docx".

S3 Unlock

Unlock specific versions of S3 objects by removing a legal hold (if present). Subsequent operators will still refer to this specific version.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
S3 bucket	The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

S3 Lock

Protect the latest version of S3 objects by applying a legal hold (if present), preventing that version from being deleted. Additionally, subsequent operators will refer to that version, even if newer versions are added.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
S3 bucket	The name of the S3 bucket where your data objects (files) are stored. Buckets are the fundamental containers in Amazon S3 for data storage. **Example:** se-s3-bucket
Previous task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

Kiteworks Lock

Protect Kiteworks files by locking them. Locked files cannot be moved or deleted.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

Kiteworks Unlock

Remove the lock from Kiteworks files.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Connection ID	Select the connection ID of the destination server.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.

Kiteworks Archived Mail Send

Used in conjunction with the Kiteworks mail connector feature, the Kiteworks Archived Mail List, and Archive Mail Download operators, this operator will send downloaded messages to the designated email destination.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Allow partial success	When downloading messages returned by the Kiteworks Archived Mail Download operator, ignore messages that have encountered issues, such as exceeding file size limits. This allows the task to proceed to success. The log associated with the task lists messages that were skipped and the reason for the action. If you don't select this check box, no messages will be skipped. If an issue occurs with any messages, no messages will be downloaded and the task will fail.
Email destination	The destination address to which messages should be sent. **Rule:** Only one address is allowed per node. To send to multiple addresses, add an Kiteworks Archived Mail Send node, in parallel, for each address.
Email from	The address to use as the sending address. The default value is the Mail From address configured in Admin > Setup > SMTP.
Include attachments	Download email attachments.

Name Parser

Append a timestamp to downloaded files before uploading them to a destination. This prevents files from overwriting each other. You can append the timestamp to a file name as the prefix or suffix.

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
File list task ID	The task ID that was automatically assigned to the preceding node. To fill in this value automatically, connect this node to the preceding node.
Append file name with	The current date and time.
Append in	Append the timestamp as a prefix or suffix.
Append operator	The task ID that was automatically assigned to the preceding node.

Transfer operators

Use transfer operators to streamline the creation of workflows for transferring files between Kiteworks servers, and between and FTP, SFTP, and SFTP servers.

Use a single operator to perform all tasks in a workflow -- download, scan, upload, rename, and more. In addition to providing improved performance, these new operators support using regular expressions to get source and destination folders, and the ability to detect whether downloaded files were modified during the workflow run and delete only the unmodified files from the originating server.

Kiteworks to SFTP

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Kiteworks source	
Connection ID	Select the connection ID of the source server.
Select source folder	The permalinks to folders and files you want to get before downloading them from the Kiteworks server. You can enter multiple folder and file links. **Example:** <folder link 1>, <folder link 2>, <file link 1> To get a folder or file permalink, click the browse button and select the folder or file path.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Action	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
SFTP destination	
SFTP Connection ID	Select the connection ID of the destination server.

Property	Description
Select destination folder	The path to the SFTP remote directory where you want to upload items. Click the browse button to access the source and select the folder path.
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

Kiteworks to Kiteworks

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Kiteworks source	
Kiteworks Source Connection ID	Select the connection ID of the source server.
Select source folder	The permalinks to folders and files you want to get before downloading them from the Kiteworks server. You can enter multiple folder and file links. **Example:** <folder link 1>, <folder link 2>, <file link 1> To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Action	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
Kiteworks destination	
Kiteworks Destination Connection ID	Select the connection ID of the destination server.
Destination folder link	The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path.

Property	Description
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Timestamp on destination	
Add timestamp for files on destination	Append a timestamp to downloaded files before uploading them to a destination folder. This prevents files from overwriting each other. You can append the timestamp to a file name as the prefix or suffix.
Timestamp format	Specify the format for the appended timestamp.
Stamp placement	Insert the timestamp before or after the file name. **Prefix example:** 2023-02-21 01:01:02_<filename>.txt **Suffix example:** <filename>_2023-02-21 01:01:02.txt
Add delimiter between the file name and the timestamp	Insert a delimiter between the file name and timestamp. Use a symbol or space to separate the file name and extension. Examples of common delimiters include comma, semicolon, space, and underscore. **Example:** <filename>_2023-02-21 01:01:02.txt
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	

Property	Description
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

Kiteworks to SMB

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
SMB destination	
SMB Connection ID	Select the connection ID of the source server.
Select destination folder	The path to the SMB (Server Message Block) shared folder where you want to upload items. Click the browse button to access the source and select the folder path.
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Kiteworks source	
Kiteworks Connection ID	Select the connection ID of the destination server.
Source folder link	The permalink to the Kiteworks folder where you want to download items. To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Action on source	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.

Property	Description
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

Kiteworks to FTPS

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Kiteworks source	
Kiteworks Connection ID	Select the connection ID of the source server.
Source folder link	The permalink to the Kiteworks folder where you want to download items. To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Action	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
FTPS destination	
FTPS Connection ID	Select the connection ID of the destination server.
Select destination folder	The path to the FTPS directory where you want to upload items. **Example:** folder1/folder2
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.

Property	Description
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

FTPS to Kiteworks

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
Source folder	The absolute path to the FTPS directory from which to download items. Click the browse button to access the source and select the folder path. **Example:** /Folder/Test123
FTPS source	
FTPS Connection ID	Select the connection ID of the source server.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Action	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
Kiteworks destination	
Kiteworks Connection ID	Select the connection ID of the destination server.
Destination folder link	The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.

Property	Description
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Timestamp on destination	
Add timestamp for files on destination	Append a timestamp to downloaded files before uploading them to a destination folder. This prevents files from overwriting each other. You can append the timestamp to a file name as the prefix or suffix.
Timestamp format	Specify the format for the appended timestamp.
Stamp placement	Insert the timestamp before or after the file name. **Prefix example:** 2023-02-21 01:01:02_<filename>.txt **Suffix example:** <filename>_2023-02-21 01:01:02.txt
Add delimiter between the file name and the timestamp	Insert a delimiter between the file name and timestamp. Use a symbol or space to separate the file name and extension. Examples of common delimiters include comma, semicolon, space, and underscore. **Example:** <filename>_2023-02-21 01:01:02.txt
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

SMB to Kiteworks

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
SMB source	
SMB Connection ID	Select the connection ID of the source server.
Source path	The path to the SMB (Server Message Block) shared folder from which to download items. Click the browse button to access the source and select the folder path.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Action on source	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Validate file content changes	If the Action is to set to delete the source files that were downloaded from the source server, only delete the files if they remained unmodified during the workflow process. If the source files were modified, don't delete them and log this information in the log file.
Kiteworks destination	
Kiteworks Connection ID	Select the connection ID of the destination server.
Destination folder link	The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.

Property	Description
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Timestamp on destination	
Add timestamp for files on destination	Append a timestamp to downloaded files before uploading them to a destination folder. This prevents files from overwriting each other. You can append the timestamp to a file name as the prefix or suffix.
Timestamp format	Specify the format for the appended timestamp.
Stamp placement	Insert the timestamp before or after the file name. **Prefix example:** 2023-02-21 01:01:02_<filename>.txt **Suffix example:** <filename>_2023-02-21 01:01:02.txt
Add delimiter between the file name and the timestamp	Insert a delimiter between the file name and timestamp. Use a symbol or space to separate the file name and extension. Examples of common delimiters include comma, semicolon, space, and underscore. **Example:** <filename>_2023-02-21 01:01:02.txt
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

SFTP to Kiteworks

Property	Description
Task ID	The unique name that gets automatically assigned to the task.
SFTP source	
SFTP Connection ID	Select the connection ID of the source server.
Source path	The path to the SFTP remote directory from which to download items. Click the browse button to access the source and select the folder path.
Include nested folders/files	If the folder contains subfolders, include files from subfolders. When uploading the files, the subfolder structure is created in the destination folder.
Regular expression of source folder	Use a regular expression (also known as regex or regexp) to get the source folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
Action on source	
-- Delete	Delete the downloaded files from the source server.
-- Do nothing	Don't delete the downloaded files from the source server.
Kiteworks destination	
Kiteworks Connection ID	Select the connection ID of the destination server.
Destination folder link	The permalink to the Kiteworks folder where you want to upload items. To get a folder or file permalink, click the browse button and select the folder or file path.
Regular expression of destination folder	Use a regular expression (also known as regex or regexp) to get the destination folder matching the pattern. The regex applies to the entire file path, without any trailing slashes. Regular expressions from the standard Python library "re" module are supported. Refer to https://docs.python.org/3/library/re.html.
On name conflict	Specify what to do if a file with the same name exists in the destination folder. You can create a new version (Keep Both), overwrite the file, or exclude (Ignore) it from the transfer process.
Timestamp on destination	
Add timestamp for files on destination	Append a timestamp to downloaded files before uploading them to a destination folder. This prevents files from overwriting each other. You can append the timestamp to a file name as the prefix or suffix.

Property	Description
Timestamp format	Specify the format for the appended timestamp.
Stamp placement	Insert the timestamp before or after the file name. **Prefix example:** 2023-02-21 01:01:02_<filename>.txt **Suffix example:** <filename>_2023-02-21 01:01:02.txt
Add delimiter between the file name and the timestamp	Insert a delimiter between the file name and timestamp. Use a symbol or space to separate the file name and extension. Examples of common delimiters include comma, semicolon, space, and underscore. **Example:** <filename>_2023-02-21 01:01:02.txt
Security scan	
Scanner connection ID	The connection ID of the generic scanner associated with the CDR scanner.
Scan profile	The profile ID of the scan engine. For example, if using Sasa GateScanner, provide the GateScanner engine profile ID.
Polling interval	The frequency (in seconds) by which to query for the scanning result of a file sent to the scanning server. If left blank, polling continues until the scan request is completed.
Max polling attempts	The number of times to query for the scan result of a file sent to the scanning server.
Share TCP session	If the scanning server is deployed on-premise and is multi-node, clear this checkbox to ensure that the HTTP request sent by the MFT server is not rejected by the scanning server due to an invalid TCP session.
Enable SSL	If the certificate is self-signed or the scanning server does not support HTTP, clear this checkbox.
PGP	
PGP encryption connection ID	The connection ID used to encrypt and decrypt files.

Property	Description
PGP action	Define the action you want to perform on the files. • Encrypt only - Use the key pair provided by the "PGP encryption connection ID" to encrypt all files. This adds the .pgp extension to each file. • Decrypt only - Use the key pair provided by the "PGP encryption connection ID" to decrypt all files. This removes the .pgp file extension from each file. Only files with the .gpg, .pgp, and .asc file extensions can be decrypted. • Sign then encrypt - Use the key pair provided by the "PGP signature connection ID" to check the digital signature of the file and verify that the file was actually signed by the person or entity claimed to be the sender. • Decrypt then verify - Use the key pair provided by the "PGP signature connection ID" to check the file integrity to verify that the file was not altered. When running a DAG, if the option to encrypt the files is specified, the action will be executed before uploading the files. If the option to decrypt the files is specified, files will be decrypted after downloading the files.
PGP signature connection ID	The connection ID used to sign or verify signatures.
Priority weight	You can assign a priority level to a DAG. DAGs with the highest priorities will be run first and according to their scheduled intervals. The following values can be assigned: 0 = low priority (the default priority), 1,000 = high priority, and 10,000 = highest priority.

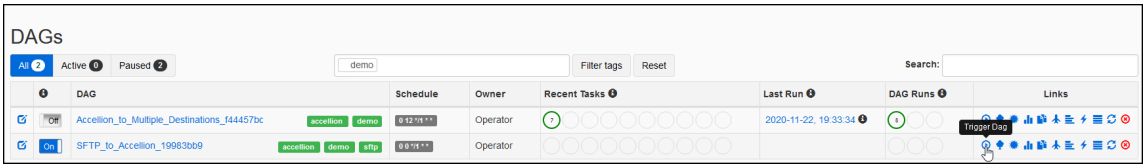
Run DAGs

To run a DAG:

1. On the DAGs view page, turn on the DAG.
2. Perform one of the following actions:
   - Wait for the scheduler to trigger the run according to the DAG's scheduled interval.
   - If the DAG file doesn't contain a set schedule (schedule = None) or you just want to start the initial run immediately, you can trigger the DAG manually. In the Links column, click Trigger DAG.

     When you manually trigger a DAG that has a defined schedule interval, subsequent DAG runs will be created by the scheduler process based on that schedule.

   Result: The DAGs view page gives you a high-level view of the run process. You may need to periodically refresh the browser to see status changes in action.



Monitor DAG runs

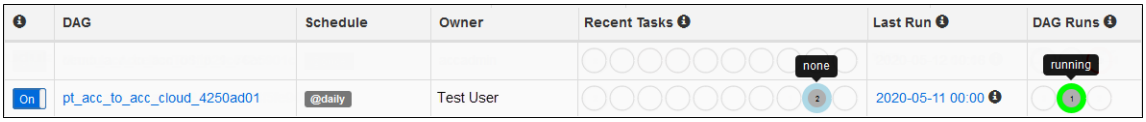
In a successful DAG run, each task in the DAG goes through these stages:

- None - No status. The scheduler creates an empty task instance.
- Scheduled - The scheduler determines the task instance needs to run.
- Queued - The scheduler sends the task to the executor to run in the queue.
- Running - A worker picks up the task and starts running it.
- Success - The task is completed successfully.

The following examples show these stages when running a DAG. The example DAG downloads files from a server, and then uploads them to a different server. . The DAG contains six tasks that run sequentially:

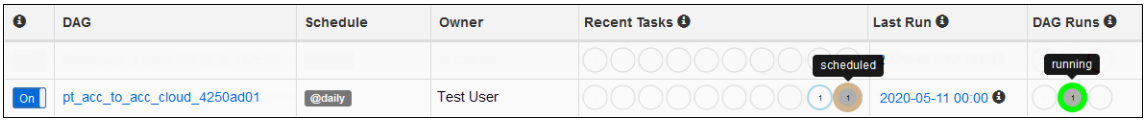
Example 1 - No status

The DAG Runs column shows the DAG is running. The Recent Tasks column shows the tasks in the DAG are waiting to be scheduled.



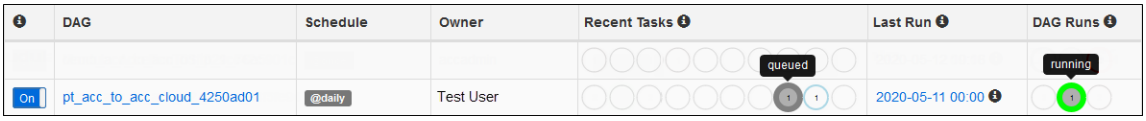
Example 2 - Task scheduled

Task 1 has been scheduled.



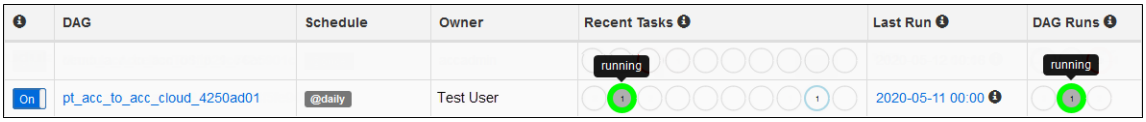
Example 3 - Task queued

Task 1 is queued and ready to run.



Example 4 - Task running

Task 1 is running.



Example 5 - Task successful

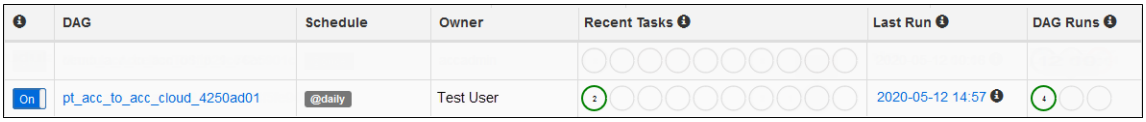
Task 1 ran successfully. Subsequent tasks are queued and ready to run.

Example 6 - DAG run successful

With a DAG run completed, the Recent Tasks columns shows that the two tasks ran successfully. The DAG Runs column shows that the entire DAG run was successful.

Example 7 - DAG status after multiple runs

If this DAG runs successfully multiple times, the DAG Runs column will ultimately show the cumulative number of successful runs. Each time the DAG runs, the Recent Tasks column will show results that include both the old and new tasks. Once the final run completes, it will show only the state of the tasks in the most recent run.



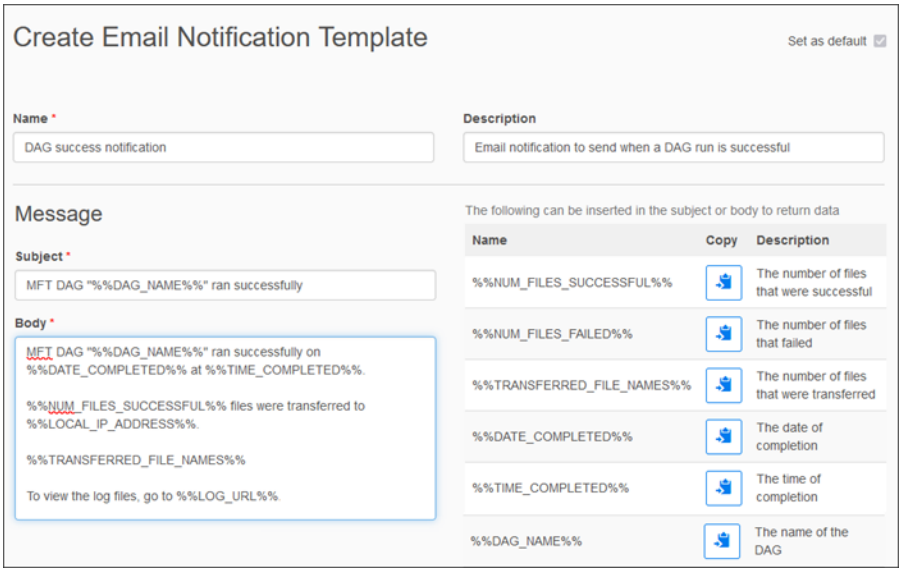
Create email notifications

You can create custom email notifications to send when DAG runs are successful, partially successful, or when an error occurs while processing the DAG run. When configuring a DAG, you select the template you want to use to notify recipients of the DAG run status. An email notification is sent after each attempt to perform a task, so there may be multiple email notifications until the task has successfully completed.

To create an email notification template:

1. Go to Admin > Notification Templates.
2. On the template creation page, click Create New.
3. Enter a name and description for the template.
4. In the Subject line, enter a combination of text and parameters you want to use to report on the DAG run. To enter a parameter, click the clipboard icon next to the parameter to copy it, and then paste the parameter into the subject line.

   For example, entering the %%DAG_NAME%% parameter will extract the name of the DAG when sending the notification.
5. In the Body, enter a combination of text and parameters to communicate the information you want to report.
   - For the %%TIME_COMPLETED%% parameter, the time format will be HH-MM-SS (for example, 03:49:54 +00:00 UTC).
   - For the %%DATE_COMPLETED%% parameter, date format will be YYYY-MM-DD.
6. If you want to make the template the default template, select the "Set as default" checkbox at the top right corner of the page. The default template is the template that will be selected automatically when configuring DAG.
7. Click Save.

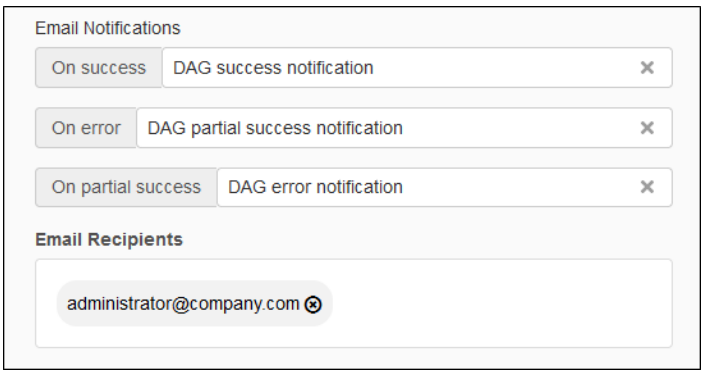


To add an email notification to a DAG:

1. On the navigation bar, click DAGs to open the DAGs view page.
2. In the row of the DAG you want to edit, click Edit .

3 In the Email Notification section, Select the templates you want to use for reporting the success, partial success, and any error details associated with the DAG run.

4 In the Email Recipients section, enter the email address of the recipients you want to receive the email notifications.

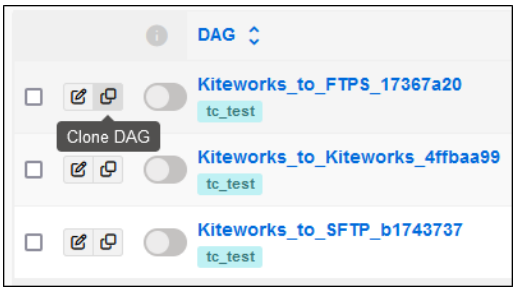


Copy DAGs

Copying a DAG enables you to duplicate the DAG. Then you can change the DAG ID (the DAG name), edit its form and operator properties, and the structure of the tasks in the workflow as needed.

To copy a DAG:

1. On the navigation bar, click DAGs to open the DAGs view page.
2. Select the DAG you want to copy, and then click Clone DAG .
3. In the DAG editor, change the DAG ID (DAG name) and then edit the form properties and operators as needed.
4. To save your changes, click Submit.



Export and import DAGs

Export DAGs

You can export DAGs for backup or for importing to a different Secure MFT Server.

DAGs and their connection parameters get exported to a zip file. When you import them to a server, the connections are also created on that server.

The export file contains the following items:

- connection_enc - An encrypted file containing the connections that were exported with the DAGs.
- "dags" folder - The exported DAGs, each in editable YAML format.

To export one or more DAGs:

1. On the navigation bar, click DAGs to open the DAGs view page.
2. Perform one of the following actions:
   - To export individual DAGs, select the DAGs you want to export, and then click Export DAGs.
   - To export all DAGs, deselect all DAGs, and then click Export All.
3. Enter a password to encrypt the connections being exported with the DAGs, and then click Export.
4. If a different user will be importing the DAGs to a server, send them the zip file along with the password for decrypting the file during import.

Import DAGs

When you import DAGs, connections exported in the DAGs are created on the server. If you import DAGs that already exist on the server, the existing DAGs will be replaced with the imported ones.

Prerequisite: To import DAGs, you'll need the zip file containing the DAGs and the password for decrypting the file during import.

To import a DAG:

1. On the navigation bar, click Create DAG, and then click the File Upload tab.
2. In the Import DAG section, select the zip file containing the DAGs, and then click Import DAG.
3. Enter the password for decrypting the file.
4. Select the role you want to have access to the DAG. To assign multiple roles, select and add each role, one at a time.
5. Click Import.

   Result: The imported DAGs are added to the DAGs view page. The imported connections are added to the connections list. It may take a moment for the scheduler to create these items. You can refresh the browser page to update the DAG list.

Upload YAML files to create DAGs

To upload a YAML file:

1. Write the DAG code to a YAML file.
2. On the navigation bar, click Create DAG, and then click the File Upload tab.
3. Upload the YAML file, and then click Submit.

   Result: The new DAG is added to the DAGs view page. It may take a moment for the scheduler to create the DAG. You can refresh the browser page to update the DAG list.

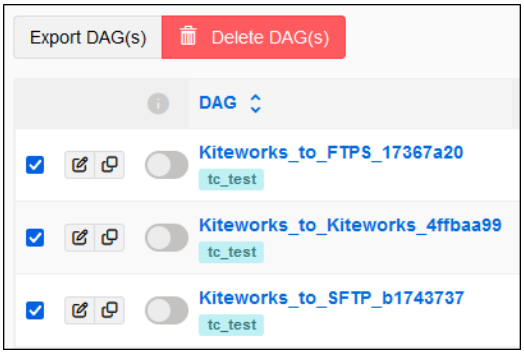
Delete DAGs

If you delete a DAG that's running, the run will complete in the background and then the DAG will be deleted.

Deleting a DAG deletes the file, while retaining the logs from prior DAG runs. To view the logs, go to the navigation menu, and then click Browse > Audit Logs. In the log list, click the corresponding DAG ID.

To delete a DAG:

1. On the navigation bar, click DAGs to open the DAGs view page.
2. Select the DAGs you want to delete, and then click Delete DAG(s).



Custom operators

Create custom operators in Development Mode

To further extend the flexibility of Secure MFT Server workflows, you can create custom operators that combine with existing Kiteworks operators to perform operations specific to your business needs.

Custom operators are defined by two files – a Python file that provides the operator logic, and a YAML file that describes how the custom operator interacts with other operators in your DAG. With your MFT Server in development mode, you can import these files to create your operator and test within your DAG without impacting your production environment.

Development mode is permission-based and typically available only to administrators. This feature is also controlled by the Kiteworks license. If you don't have access to development mode, contact your Kiteworks server administrator. They can check your permissions and also to contact their Kiteworks account representative to enable the feature in the Kiteworks license.

Enable development mode

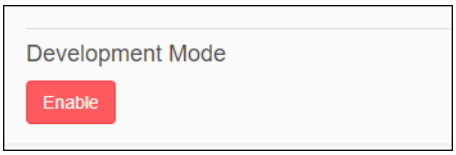
The MFT Server includes a development mode that can be enabled to provide a safe environment in which to create and test custom operators.

For improved security, MFT servers in development mode will not run DAGs based on event listeners or configured schedules. DAG runs must be manually started in development mode.

Important: Once development mode is enabled, the MFT Server cannot be returned to production. Enabling development mode disables DAG scheduling and event listener triggers, although you can still run DAGs manually. Enabling development mode will restart the web services, which will cause the MFT server to be offline until the restart completes.

To enable development mode:

1. On the navigation bar, click Admin > Setup > Development.
2. On the Development page, click Enable.



3. You will be prompted to confirm. Read the prompt and click OK to continue.

Result: A confirmation will be displayed when development mode is enabled and the web services have been restarted.

Create custom operators

With the MFT server in development mode, you can create and upload custom operators to include in your DAGs. Custom operators are defined by two files – a Python file that provides the operator logic, and a YAML file that defines how the custom operator can interact with other operators.

Using the custom operator templates

A template Python and YAML file are provided to assist you in creating your custom operators. The templates include the basic structure required, but do not include any custom code. You can edit these templates using a standard text editor such as Notepad.

Create the custom operator Python file

The supplied Python template is preconfigured to import the required operators and Kiteworks utilities. Additionally, the template is pre-populated with the "init" and "execute" methods required to define an operator. These methods should be further defined based your specific needs.

To create the custom operator Python file:

1 Using the information below as a guide, update the template to define the name for your operator.

Item	Description (Optional)
Class SampleCustomOperator	Change the class name (SampleCustomOperator) to something descriptive for your purposes. This will be used by the "operator_class" parameter in the associated YAML file.
name	Choose a name for your operator. This name must be unique among all operators on your MFT server. This will be used by the "name" parameter in the associated YAML file to pair the operator files.

2 Configure the "init" method to define the allowed parameters for your operator. Refer to DAG form operators for suggestions on the type of parameters used.

3 Modify the "execute" method to include your custom code.

Create the custom operator YAML file

An operator's YAML file defines the inputs for an operator and what sort of validation should be done for those inputs.

To create the custom operator YAML file:

1 Using the information below as a guide, update the operator YAML file to include:

Item	Description (Optional)
name	The name of your operator, defined in the operator's Python file.
operator_class	This is the name of the operator class you defined in the Python file.

2 Define the required properties for your operator. The provided template includes the properties and parameters below. The properties you define must match what you've defined in the "init" method in your operator's Python file.

Item	Description (Optional)
task_id	This property does not need to be changed. For reference, it includes the keywords below: • type: string • label: "Task ID" • placeholder: "Task ID" • description: "Unique identifier of this task" • form: "static"
conn_id	This property defines the connection used by the operator. The template uses the keywords below: • type: string • label: "API Connection ID" • placeholder: "Kiteworks Connection ID" • description: "Kiteworks connection ID that are created through Admin/Connections page" • form: "show" • pattern: ^[\da-f]{8}-([\da-f]{4}-){3}[\da-f]{12}$
Previous_task_id	• type: string • label: "File Download/Scan Task ID" • placeholder: "Task ID" • description: "Id of the previous file download/scan task" • form: "static" • previous_task: "previous"
required	This defines the properties required when configuring the operator in the DAG.

Item	Description (Optional)
previous_operators	This parameter defines the operators that are allowed to precede the custom operator in the DAG. Use "null" to disable validation.
next_operators	This parameter defines the operators that are allowed to follow the custom operator in the DAG. Use "null" to disable validation.

Upload the operator files

Once your operator files are complete, you can upload them to your MFT server running in development mode.

To upload the operator files:

1. On the navigation bar, click Admin > Setup.
2. Click the Development tab.
3. In the Upload Testing Operator Files section, click Choose File to select your Python and YAML files.
4. Click Submit.

   Result: The files will be uploaded to the MFT Server. An automated consistency check will be performed to verify syntax and basic parameters. If the consistency check succeeds, the operator will be listed in the Testing Operator section.
5. Click Restart Web Service to add the operators to the Custom section of the DAG operators.

Additional resources

Resources

Get support

Need assistance with your Kiteworks product? Contact us with requests for assistance or to submit product enhancements.

Contact technical support at support@kiteworks.com.

Learn more about Kiteworks products

Visit the Kiteworks Support Portal to read technical articles, view training videos, search the knowledge base, and download additional user documentation.