MOVEit Automation Web Admin Help

This document was published on Friday, October 27, 2017 at 15:33.

Contents

Introduction

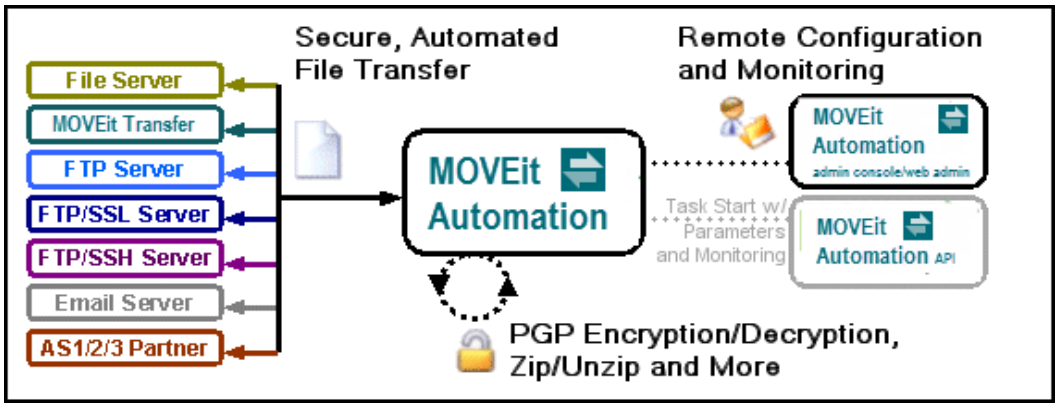
Overview

MOVEit Automation is an enterprise-level, Windows-based, automated managed file transfer (MFT) workflow engine that pulls, processes and pushes files on a scheduled, event-driven or on-demand basis between internal and external systems, including MOVEit Transfer servers.

MOVEit Automation does this using easily-created tasks (no programming required) that can exchange files between multiple systems using multiple protocols, and process files with many built-in functions (including OpenPGP encryption) and custom VBScript scripts.

MOVEit Automation is provided in multiple editions to match the needs of a wide variety of customers. The license determines the features that you can access.

- You can review the features that are included with each license and determine which license meets your business needs here: ***Edition Information https://www.ipswitch.com/secure-information-and-file-transfer/moveit-automation***.
- To review the functions and features available in your current license, run the Config Utillity. For more information, see ***GetMICConfig Utility***.



Note: Some functionality is only available to certain product editions.

Supported Servers and Protocols

MOVEit Automation editions securely and automatically transfer files to and from:

- FTP servers
- FTP over SSL (FTPS) servers
- FTP over SSH (SFTP) servers
- the local filesystem
- network folders
- email servers
- MOVEit Transfer servers
- AS1, AS2 or AS3 servers.

Supported File Processing

File and folder synchronization/replication is available between any two selected folders on these servers.

PGP encryption/decryption, zip operations, rename, find and replace, command-line applications and anti-virus integration are also built in and require no additional software. *(Native PGP must be enabled in your MOVEit Automation license code. PGP availability is dependant on your* MOVEit Automation edition.*)*

Automated server-to-server file transfers require no knowledge of any script language because MOVEit Automation provides an operator-friendly user interface to schedule tasks and monitor their progress.

In cases where custom scripts are necessary, certain MOVEit Automation editions fully support VBScript, with many custom functions to make integration with MOVEit Automation tasks seamless. External applications and schedulers may also be used to start tasks, pass task parameters, monitor running tasks, and retrieve transfer and audit logs through MOVEit Automation API, available in component and command-line versions for Windows and Java.

Security

Based on a robust scheduling facility, MOVEit Automation avoids becoming a security target by protecting sensitive access information with powerful encryption, local files with NIST 800-88-compliant data erasure and configuration channels with SSL.

Remote access to MOVEit Automation is restricted to specific Windows users in local or domain groups. With certain MOVEit Automation editions, group access to transfer tasks and related elements can be fine-tuned to delegate management in a number of commonly requested configurations. The basic MOVEit Automation edition does not allow the configuration of permissions.

All transactions are logged to a tamper-evident ODBC database usable by the MOVEit Automation built-in reporting feature and by custom billing and tracking systems.

Local or remote configuration, control and monitoring of the MOVEit Automation service is performed through the Web Admin application. Unlimited copies of this management client can be run at any specified time, and different permissions to different task groups can be allocated to different users.

Notice: The Web Admin application is available. It requires a separate installation process. For more information, see ***MOVEit 2017 Plus Installation Guide http://docs.ipswitch.com/MOVEit/Transfer2017Plus/Manuals/en/index.htm#44112.htm***.

Cryptographic services, including complete encryption of all configuration files, are provided by MOVEit Crypto. Also available as a separately licensed commercial product for Windows or Linux developers, MOVEit Crypto has been validated under FIPS-140-2 by the United States and Canadian governments.

Key Features

MOVEit Automation simplifies the creation of standardized workflows and makes it easier to ensure reliability, security and compliance. The key features associated with MOVEit Automation are outlined below.

Licensing

MOVEit Automation is provided in multiple editions to match the needs of a wide variety of customers. The license determines the features that you can access.

- You can review the features that are included with each license and determine which license meets your business needs here: ***Edition Information https://www.ipswitch.com/secure-information-and-file-transfer/moveit-automation***.
- To review the functions and features available in your current license, run the Config Utillity. For more information, see ***GetMICConfig Utility***.

Exchange Files with:

- FTP servers
  - Insecure FTP.
  - FTP over SSL, Explicit and Implicit Connect Modes, plus "CCC" support.
  - Client Certificates.
  - Active and Passive Data Transfer Modes.
  - Supports Custom "Quote" Commands, Upon Connection and Per-File.
  - Supports Unusual Directory Listings, for example IBM.
  - Supports Synchronization/Replication.
  - Automatic Retry of Failed Transfers.
  - "Partial File" Protection, Avoids Pulling Files in Use, Rename After Upload.
  - Client NAT.
  - Download Integrity Verification, MD5.
- SSH Servers
  - Password and Public Key Authentication.
  - Supports Unusual Directory Listings.

- Supports Synchronization/Replication.
- Automatic Retry of Failed Transfers.
- "Partial File" Protection, Avoids Pulling Files in Use, Rename After Upload.
- Download Integrity Verification, MD5.

- MOVEit Transfer servers (HTTPS interface)

Note: Prior to the MOVEit Automation 2017 release, the name of the MOVEit DMZ product was changed to MOVEit Transfer. In MOVEit Automation 2017, both MOVEit Transfer servers and supported versions of MOVEit DMZ servers can be added as MOVEit Transfer hosts. In this help system, the term *MOVEit Transfer host* refers to a host based on a MOVEit DMZ server or a MOVEit Transfer server, unless otherwise indicated. For more information on supported versions, see the *Release Notes https://docs.ipswitch.com/moveit/Automation2017Plus/ReleaseNotes/en/index.htm*.

  - Event-Driven,"Transfer Immediately Upon Complete Receipt"
  - Complete Guaranteed Delivery
    - Cryptographic-Quality Integrity Check, SHA1.
    - Automatic Restart of Partial Transfers.
    - Automatic Retry of Failed Transfers.
  - "Partial File" Protection, Avoids Pulling Files in Use, Rename After Upload.
  - Session Reuse, for Performance.
  - Supports Synchronization/Replication.
- Windows servers
  - Local Filesystem
    - Event-Driven, "Transfer Immediately Upon Complete Receipt"
  - Windows-based network shares , including "Mapped Drive Letters"
    - Event-Driven, "Transfer Immediately Upon Complete Receipt"
  - Novell and other NAS resources
  - All
    - Automatic Retry of Failed Transfers.
    - "Partial File" Protection, Avoids Pulling Files in Use, Rename After Upload.
  - Supports Synchronization/Replication.
- Email servers
  - Outbound Protocol: SMTP.
  - Inbound Protocol: POP3.
  - Password Authentication.
  - SMIME Encrypt/Decrypt/Sign Scripts included.
  - Automatic Retry of Failed Transfers.
- AS1, AS2 and AS3 Partners
  - Drummond "eBusiness" certified software.
  - AS1: POP3/S and SMTP/S with email MDN support.
  - AS2: HTTP/S with asynchronous, synchronous and email asynchronous MDN support.
  - AS3: FTP/S with MDN support.

-  HTTP (web) servers
  - HTTP or HTTPS protocol.
  - HTTP authentication.
  - Upload via PUT or POST.
  - Download via GET.
  - Special support for Microsoft SharePoint Server.

Schedule Tasks to run:

- When a target file is complete (*MOVEit Transfer and FileSystem only*).
- On specific days of the week, for example Monday and Wednesday.
- On specific dates in the month, for example the 15th and 20th.
- At a specific time, for example at 17:00pm, once or repeatedly.
- Several times, within a specified time range, for example from 13.00 to 15.00.
- Until certain files are transmitted successfully
- In response to other successful or failed transfers
- When certain files are found on disk, for example rigger files.
- When commanded to do so by an external application through the MOVEit Automation API interface

Configure Tasks to:

- Zip/unzip files
- Retrieve new files, for example files that are newer than 1 days or since the last time MOVEit Automation looked.
- Retrieve old files, for example files that are older than 7 days.
- Retrieve small or large files
- Transfer available files in batches, up to X files or just more than Y bytes per task run.
- Never overwrite existing files.
- Append to existing files.
- Create outbound folders if they do not already exist.
- Run simultaneously with many other tasks, even to different hosts.
- Ignore specified files or folders.
- Delete, rename, or move source files after a successful transfer.
- Run command-line applications against files.
- Use macros to rename outbound files or select source files, for example [mm][yyyy].
- Use macros to look for inbound files
- Kick off other tasks or send email notifications in response to responses and failures
- React to or parse, or both, trigger files.
- PGP Encrypt/Decrypt Files, using native PGP and license.
- SMIME Encrypt/Decrypt Files, using Included SMIME Control.
- Handle large files, for example, over 10GB.
- Synchronize/replicate files and folders between two selected folders.
- Record the path, size and date of source and destination files, the speed of the transfer, and any errors encountered or processes run in a tamper-evident transfer database.
- Execute custom VBScript against any file.
- Use Advanced Tasks to apply conditional processing and consolidate "chained" tasks.

Configure Integrated Antivirus to:

- Work with Symantec AntiVirus, McAfee VirusScan, and Trend Micro OfficeScan.
- Either ignore infected source files or actively delete them.

Configure Logs and Events to:

- Write to a text file.

- Write to the Windows Application Event Log or a Windows "MOVEit" Event Log. Either of these sources may be sent to SysLog or SNMP servers.
- Send email notifications if tasks succeed, fail or do no run.

Monitor Tasks through:

- Running "debug log" with extensive debugging information, for example FTP "200" messages.
- Display of active and inactive tasks.
- "Reports" dialog with historical task run and audit events.
- Drill-down views of specific task runs and file transfers.
- Your own custom log display using MOVEit Automation API (or ODBC) access to real-time log database.

Ensure Security with:

- Encrypted file transport using FTP over SSL, FTP over SSH and HTTPS.
- Encrypted Admin/Central communications (requires installation of valid SSL certificate on MOVEit Automation platform).
- Encrypted configuration files (256-bit AES).
- Encryption of individual files with PGP or SMIME.
- AS2 and the other ASx protocols (AS1 and AS3).
- Auditing of administrator actions, task runs, and file transfers.
- Tamper detection of audit and activity logs, via cryptographic hashing.
- Overwrites of temporary cache files with cryptographic-quality random data (NIST 800-88-compliant data erasure).
- NIST-validated cryptography.

Run as:

- A service on Windows 2008 and 2012.
- A desktop application with the scheduler disabled and/or a special test configuration file in test and development situations.

Delegate to Operators the Ability to:

- Monitor, run, make minor or major changes to, add and delete certain tasks.
- Monitor/reference or make changes to, add and delete certain hosts, SSL certs, SSH certs, PGP keys, and scripts.
- Start/restart/stop certain tasks.

Conventions used in this documentation

Cueing graphics

The documentation for features that are only available in MOVEit Automation 2017 Plus Service Pack 1 are identified by the following cueing graphic.

Icon updates

Some User Interface improvements were implemented in Service Pack 1. The following table provides a list of updated icons.

2017 Plus	2017 Plus Service Pack 1
Gears icon	More Options icon
Actions icon	More Options icon
Import icon	Import icon
Export icon	Export icon

Web Admin - Overview

In the MOVEit Automation 2017 Plus release, the Web Admin interface has been expanded to include most of the functionality in the MOVEit Automation Admin Console.

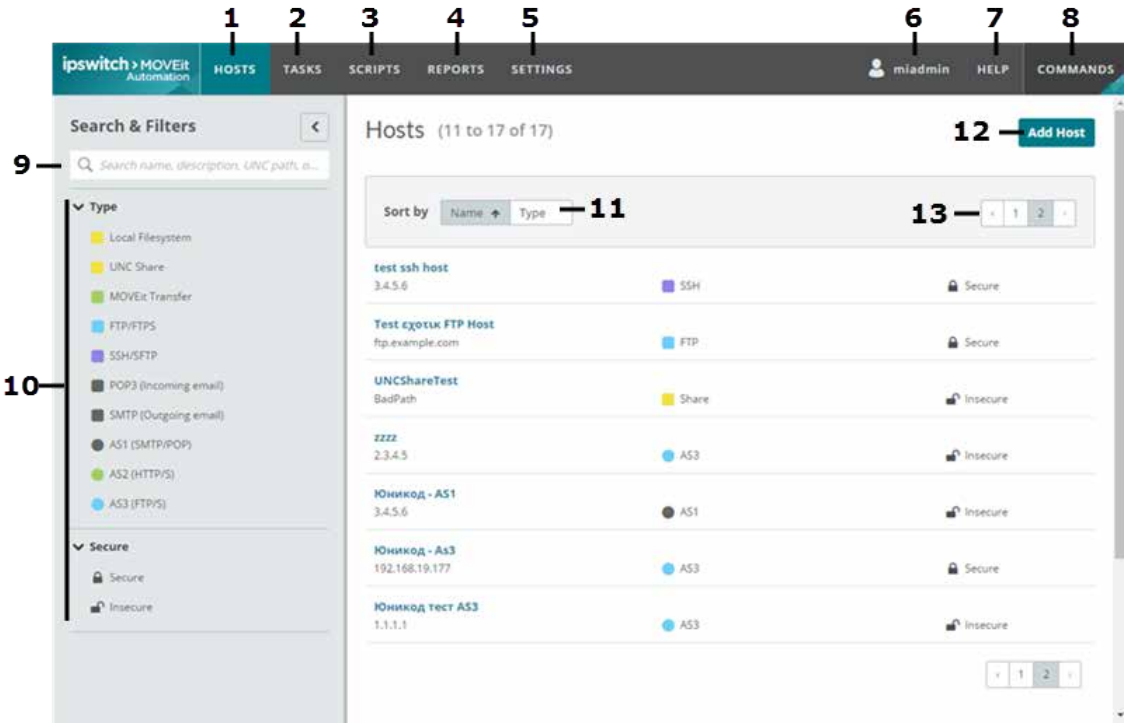
If you are signed in to Web Admin, the MOVEit Automation software has already been installed, and the administrator has used the Admin Console to set up permissions and optionally, MOVEit user groups.

This section describes the areas of the Web Admin interface.

Web Admin Tour

The Web Admin navigation bar contains a link for each functional area of MOVEit Automation.

The table describes the areas indicated by the red numbers in the figure. For more information about specific items, click the links in the table.



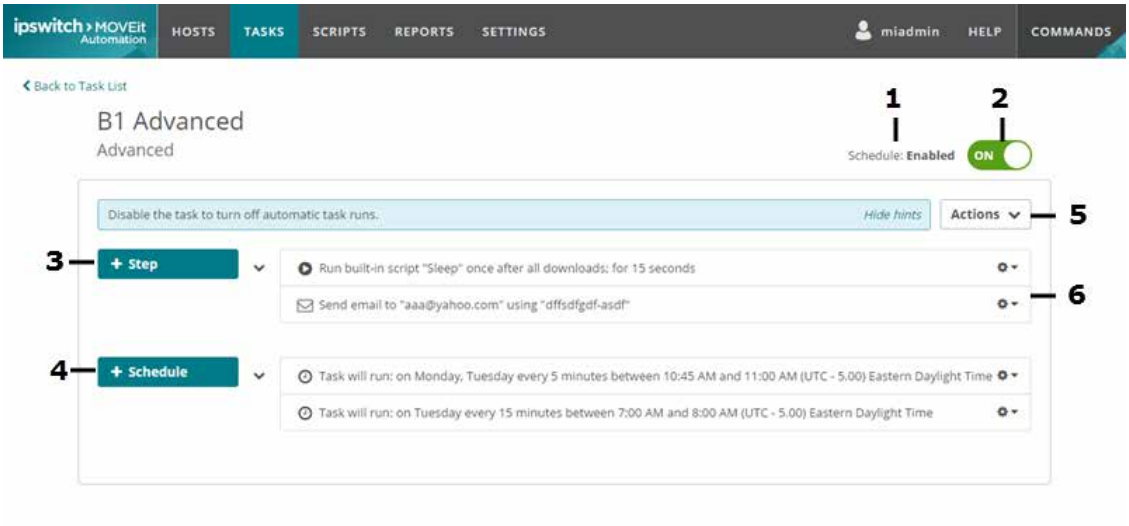
Number	Item	Description/Actions
1	HOSTS	Add and manage Hosts
2	TASKS	Task List § View list of all tasks, showing the task name and ***schedule*** (on page 30) of each task, and the start time and duration of the last run of the task. § Result, Files, and Size columns reflect information from the last (most recent) run of the task. § Add and manage tasks. § Perform bulk actions on all listed tasks. § Use filters to limit the tasks listed. For more information, see Bulk Actions. See also: § Individual Task Page - Quick Tour.

3	SCRIPTS	Add a custom script. Manage built in scripts. For more information, see ***Scripts*** (on page 103).
4	REPORTS	***Task Run*** (on page 149) § View details about each task run. ***File Activity*** (on page 151) § View details about activity of individual files that are processed during task runs. ***Audit*** (on page 152) § View the audit details for the system.
5	SETTINGS	§ ***System Settings*** (on page 154) - Logs, tasks, state file, tamper detection. § ***Task Groups*** (on page 155) - Add, manage task groups § ***Date Lists*** (on page 157) - Add, manage, import date lists § ***Keys and Certs*** (on page 157) - Import, create keys and certs § ***Global Parameters*** (on page 172) - Create, manage global parameters
6	username	***User Preferences*** (on page 19): § Time display format § Items per page
7	HELP	Help for the page or item currently displayed.
8	COMMANDS	Commands that execute immediately: refresh, import, or export config, stop or start task scheduler, shut down service, test antivirus, reset tamper detection. For more information, see Commands.
9	Search	Searches the current page. For more information, see ***Searching, Filtering and Sorting*** (on page 12).
10	Filters	Lists selections in categories that pertain to the current page. For example, on the SETTINGS > Keys and Certs page, categories include Type (for key type), Key Group, and Expired. For more information, see ***Searching, Filtering and Sorting*** (on page 12).
11	Sort selections	Sorts (orders) the list of items. Sort types are based on the page/tab displayed or selected.
12	Add item	Add the item that pertains to the displayed page. For example, on the TASKS page, you can Add Task.
13	Pages	Subsequent pages when the list of items spans multiple pages. Set the count per page in ***username*** > User preferences. See table row 6 - username, above.

Individual Task - Quick Tour

On the TASKS tab, clicking an individual task name opens a page that shows the components of the task.

The figure shows an Advanced task. The table describes the areas indicated by the red numbers in the figure. For more information, click the links in the table.



Number	Item	Description/Actions
1	Task schedule status	For more information, see ***Task schedule status*** (on page 30).
2	Enable or Disable	Click to toggle ON (enable) or OFF (disable). For more information, see ***Task schedule status*** (on page 30).
3	+ Step	Add a step to the task. Options depend on the task type. See What comprises a task.
4	+ Schedule	Add and configure a schedule. For more information, see ***Schedule*** (on page 85).
5	Actions drop-down	Actions to perform on the displayed task. Options depend on the task type. For more information, see ***Actions*** (on page 95).
6	Gears icons	Options for the specific step or item. All items include options to edit or delete the individual step or item. Additional options on the dropdown list depend on the type of item or step. For more information, see ***Task Actions*** (on page 95).

Searching, Filtering, and Sorting

Searching, filtering, and sorting options are available across the HOSTS, TASKS, SCRIPTS, REPORTS, and SETTINGS tabs. You can search for items, apply filters, or both to limit the number of items shown in the list. You can sort the task list and results lists.

Searching

You can search for items to limit the number of items shown in the list.

The Search field is available at page level on the HOSTS, TASKS, SCRIPTS, REPORTS, and SETTINGS tabs. The following list details the search options available on each page.

- HOSTS: Name, Description, UNC Path, Host Address.
- TASKS: Task Name, Description.
- SCRIPTS: Name, Description.
- REPORTS > Task Run: Task Name
- REPORTS > File Activity: Source Path, Destination Path.
- REPORTS > Audit: Target Name, Username, IP Address.
- SETTINGS > Task Groups: Name.
- SETTINGS > Date Lists: Name, Description.
- SETTINGS > Keys and Certs: Name, Key ID.
- SETTINGS > Global Parameters: Name, Value.

Task List Searching

A search applied on the Task List page is automatically applied to the Task Run, File Activity, and Audit tabs as a filter. You can remove or edit the applied filters. For more information about removing or editing the filter, see ***Filtering*** (on page 13).

Ø *To show/hide the Search & Filters panel:*

- Click the Show or Hide Search & Filters icon .

Search terms applied to a collapsed Search & Filters panel are indicated with a green check mark on the search icon  .

Filters applied to a collapsed Search & Filters panel are indicated with a green check mark on the filters icon  .

Filtering

The Search & Filters panel contains categories that pertain to the type of item shown on the page.

The following list details the filter category options available on each page:

- HOSTS: Type, Secure
- TASKS: Last Run Result, Schedule, Task Group, Folders and Files, Uses Host, Uses Script, Type
- SCRIPTS: Type
- REPORTS > Task run: Result, Time/Date
- REPORTS > File Activity: Result, Activity, Time/Date
- REPORTS > Audit: Result, Action, Time/Date, Log ID
- SETTINGS > Keys and Certs: Type, Key Group, Expired

Ø *To select a filter:*

- Choose from one of the following options. Your filter selection is displayed below the search field in the Search & Filter panel. Newly added filters are temporarily highlighted.
  - To create a single filter condition, select one filter in one category. For example, selecting Running in the Last Run Result category displays the results for Running.
  - To create a multi-filter condition in one category, select two or more filters in one category. The 'OR' condition is applied. For example, selecting Running or Success in the Last Run Result category displays the results for Running or Success.
  - To create a complex filter condition, select one or more filters in two or more categories. Within categories, where two or more filters are selected, the 'OR' condition is applied. Where filters are selected in two or more categories, the 'AND' condition is applied. For example, selecting Running or Success in the Last Run Result category, and Traditional in the Type category displays the results for Running or Success, and Traditional.

Ø *To clear filters:*

- Choose from one of the following options.
  - All filters: Click **Clear Filters**.
  - Individual filters, a single filter is selected in a category: Click the X on single filter selection. Alternatively, under the category, to de-select the filter click the selection.
  - Category filters, multiple filters are selected in a category: Click the X on the category filter selection. Alternatively, under the category, to de-select the filters click each selection.

Task List Filtering

Filters applied on the Task List page are automatically applied to the Task Run, File Activity, and Audit pages. You can remove or edit the applied filters.

- To remove the automatically applied filters, clear the **Apply Task List filter(s)** check box above the applied filters in the **Search & Filters** panel. To reapply the task list filters select the **Apply Task List filter(s)** check box.
- To edit the automatically applied filters, click **Edit** the **Apply Task List filter(s)** in the **Search & Filters** panel. You are redirected to the Task List page. Edit the selected filters on the Task List page. For more information, see ***Filtering and Sorting*** (on page 12).

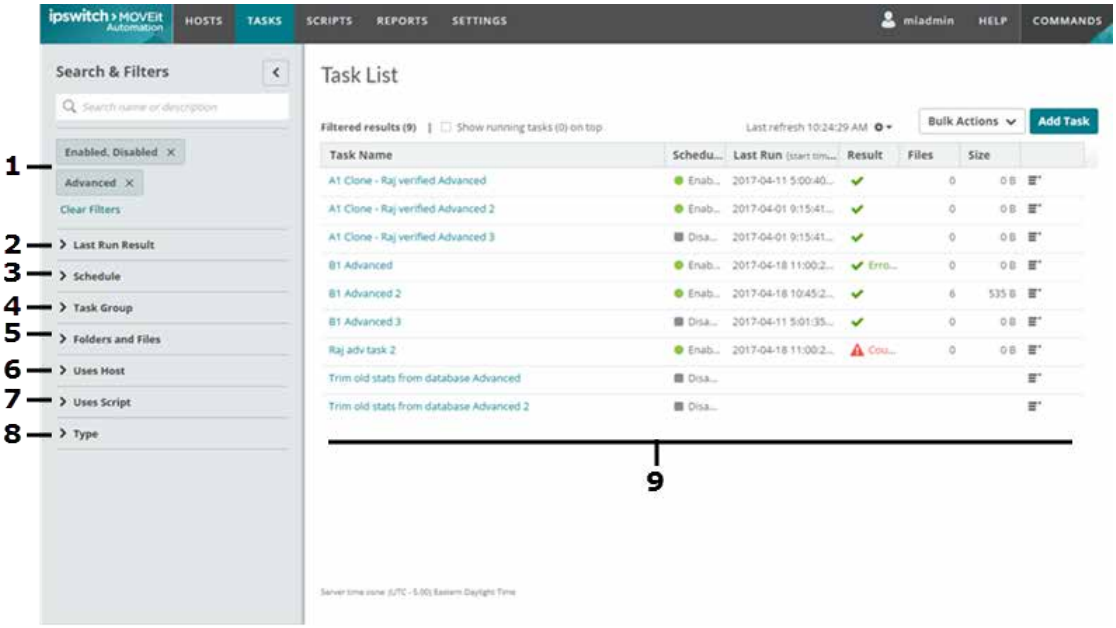
Ø *To show/hide the Search & Filters panel:*

- Click the **Show** or **Hide Search & Filters** icon .

Search terms applied to a collapsed Search & Filters panel are indicated with a green check mark on the search icon .

Filters applied to a collapsed Search & Filters panel are indicated with a green check mark on the filters icon .

The figure shows an example of filters on the TASKS page. The task list shows the tasks that meet the filter criteria. The table describes the areas indicated by the red numbers in the figure.



Number	Item	Description/Actions
1	Filters in use	Displays the selected filters.
2	Last Run Result	Options: Running, Success, Failure, No Transfers
3	Schedule	Options: Enabled, Disabled, Unscheduled, Config Incomplete

4	Task Group	Lists the existing task groups.
5	Folders and Files	Specify a folder path and optional filemask.
6	Uses Host	Lists the existing hosts.
7	Uses Script	Lists built-in scripts and custom scripts
8	Type	Options: Traditional, Advanced, Sync
9	Results	All items that meet the filter criteria. To sort by column, click a column head.

Note: The TASKS page Bulk Actions button applies the action you select to all of the items in the list, even if the list spans multiple pages. Use the filters to limit the list of items.

Sorting

You can organize the data by sorting the page lists. The Sort option is available at page level on the HOSTS, TASKS, REPORTS, SCRIPTS, and SETTINGS tabs.

Ø *To sort a page list, select one of the following options, depending on availability:*

1 Sort by column header:

a To sort the column ascending, click the column header once. The ascending icon , is displayed in the header.

b To sort the column descending, click the column header twice. The descending icon , is displayed in the header.

2. Placeholder

a To sort the list ascending, click the Sort by option once. The ascending icon , is displayed in the Sort by field.

b To sort the list descending, click the Sort by option twice. The descending icon , is displayed in the Sort by field.

You can only select one column or Sort by option at a time.

The following list details the sort options available on each page.

- HOSTS: Name, Type
- TASKS: Task Name, Schedule, Last Run, Result, Files, Size
- SCRIPTS: Name, Type
- REPORTS > Task Run: Start Time, Task Name, Duration, Result, Files Sent, Total Bytes Sent, Started by
- REPORTS > File Activity: Log Time, Task Name, Source File Name, Activity, Result, Size, Source Path, Destination Path
- REPORTS > Audit: Log Time, Action, Target Type, Target Name, Result, Username, IP Address
- SETTINGS > Task Groups: Name
- SETTINGS > Date Lists: Name
- SETTINGS > Keys and Certs: Name, Type, Expire Date
- SETTINGS > Global Parameters: Name, Value

Common Activities

The table lists common activities and where to find them in Web Admin.

To...	Do this...	For more information, see
Create a task	**1** Click TASKS. **2** Click Add Task. Provide a friendly name and select the task type. Click Add Task.	§ Configure a Traditional Task § ***Configure an Advanced Task*** (on page 33) § ***Configure a Synchronization Task*** (on page 35)
Edit a task	**1** Click TASKS. Click the task name. **2** Add Steps, Next Actions, or a Schedule.	***Task Elements*** (on page 53)
Export a task	**1** Click TASKS. Click the task name. **2** Select Actions > Export.	***Task Actions*** (on page 95)
Create a task group	§ Click SETTINGS > Task Groups. Click Add Task Group. - or - § Click TASKS. Filter the list to show only the tasks you want in the group. Click Bulk Actions > Create Task Group.	***Task Groups*** (on page 155)
Add a task or other member to a task group	**1** From the top menu, select SETTINGS > Task Groups. Click the name of the task group. **2** Click Edit. The Edit task group dialog box opens. **3** In the Members section, select the tab for the member type and click +Add ***member type***. **4** The Browse ***member type*** dialog box opens. Make selections. To select multiple items, hold the Ctrl key while selecting. Click OK.	***Task Groups*** (on page 155)
Add multiple tasks to a task group	**1** Click TASKS. Use filters to limit the tasks that appear in the list. **2** Click Bulk Actions > Add to task group. **3** Select the task group and click OK. **4** In the Members section, select tabs and delete or promote as needed.	Bulk Actions
Enable or disable a task	A disabled task cannot be run by the scheduler, even if the task has a schedule. **1** Click TASKS. Click the task name. **2** Click the Enable Task button to toggle ON (Enabled) or OFF (Disabled)	***Task Schedule Status*** (on page 30)

To...	Do this...	For more information, see
Delete a task	Click TASKS, Click a task name. Click Actions > Delete.	***Task Actions*** (on page 95)
Clone a task	**1** Click TASKS and click the task name. **2** Select Actions > Clone. The clone is named ***OriginalTaskName*** Clone.	***Task Actions*** (on page 95)
Clone a source, destination, process, schedule, or next action	**1** Click TASKS and click the task name. **2** In the row for the element that you want to clone, click the more options icon ⋯ and select Clone.	***Task Actions*** (on page 95)

Features Included in Web Admin

Both the MOVEit Automation Admin Console and the MOVEit Automation Web Admin interfaces are supported in Version 2017 Plus, so if you are accessing the same MOVEit Automation server, configurations made in one interface are automatically accessible through the other interface. Both interfaces observe the user permissions configured in the Admin Console.

Features and Functions Implemented in MOVEit Automation 2017 Plus Web Admin

- Hosts
- Tasks
- Task run and file activity history and reporting
- Single view for tasks and task state
- Task Groups
- Sync Tasks
- Advanced Tasks
- Scripts
- Commands
- Settings: System Settings, Task Groups, Date Lists, Keys & Certs, Global Parameters
- Expired keys list
- List sorting and filtering
- Context-sensitive Help (dialog box and page)
- Report access from a single location
- Config import/export
- Task export
- Audit Log reporting*
- Edit Source Timestamps*

*New in Web Admin

Features and Functions Not Implemented in MOVEit Automation 2017 Plus Web Admin

Use the Admin Console for these features and functions:

- Debug Log

Edit User Preferences

- Task Import
- User Permissions Configuration

To access this dialog box: In the top right of the Web Admin screen, click your **username** and select **Preferences**.

Make selections for the following items, and click **Save**.

- **Time Display Format:** 12 or 24 hour
- **Items per page:** Number of items that appear on the HOSTS, TASKS, and SCRIPTS pages. When the total number of items exceeds the Items per page, the list continues on another page. Numbers near the bottom of the screen indicate subsequent pages.

Controlling Access to MOVEit Automation

Note: The **Bulk Actions** button on the **TASKS** page performs the selected action on all the tasks that appear in the list, including those that appear on subsequent pages if the count exceeds the specified Items per page. Use the filters to limit which tasks appear. For more information, see Bulk Actions.

HOSTS

Permissions features, including adding users and MOVEit-User groups, are not supported in MOVEit Automation Web Admin.

Use the Admin Console for these features.

Hosts define endpoint servers where MOVEit Automation can get or put files. Sources and destinations depend on host definitions. You must define a host before you can create and configure tasks that use the host in a Source or Destination.

Changes made to a host definition immediately affect all sources and destinations that depend on that definition. This behavior is useful when you are configuring many tasks that use the same host. For example, if the host changes its IP address, the source or destination reflects the change.

The Hosts page displays the following columns of information for each host.

Host Name - Specifies the unique host name.

Host Type - Specifies the host type. There are multiple host types. Each host is a single type. For more information, see ***Supported Host Types*** (on page 23).

Security - Specifies whether the host is *Secure* or *Insecure*.

Note: Prior to the MOVEit Automation 2017 release, the name of the MOVEit DMZ product was changed to MOVEit Transfer. In MOVEit Automation 2017, both MOVEit Transfer servers and supported versions of MOVEit DMZ servers can be added as MOVEit Transfer hosts. In this help system, the term *MOVEit Transfer host* refers to a host based on a MOVEit DMZ server or a MOVEit Transfer server, unless otherwise indicated. For more information on supported versions, see the *Release Notes https://docs.ipswitch.com/moveit/Automation2017Plus/ReleaseNotes/en/index.htm*.

Hosts key features

The Hosts page displays a list of existing Hosts. Use this page to view, search, sort, edit, or create new and existing hosts.

View existing Hosts

To view a list of existing Hosts in MOVEit Automation Web Admin,

1 Click the HOSTS link in the Web Admin navigation bar.

A list of existing Hosts is displayed. The list can span several pages.

Search existing Hosts

To limit the number of items that are displayed in a list, you can search the list. The Search field is available at the Hosts page level.

The Search field searches the following host properties: Name, Description, UNC Path, Host Address.

---

1 Click the HOSTS link in the Web Admin navigation bar.
2 Input the search term in the Search field in the Search & Filters panel.

An updated list is generated as the search term is input in to the Search field.

Filtering existing Hosts

To narrow the lit of Hosts, or your search results, you can filter the list of hosts.

The following filter category options are available on the Hosts page: Type, Secure.

---

1 Click the HOSTS link in the Web Admin navigation bar.
2 Expand Type and Secure in the Search & Filters panel.
3 Select the host type or types, the security options, or both, for the filtering that you want to apply.

An updated list is generated as the filters are applied to the Hosts page.

Editing an existing Host

To edit an existing host.

1. Click the HOSTS link in the Web Admin navigation bar.
2. To select a host that you want to edit, click the host name.
3. Click Edit on one or more of the following host categories. The available categories differ for different Host types.

- General
- Uploads
- Limits
- Timeouts
- Timeouts & Transfers
- Security
- Proxy Server
- Advanced
- Directory List Parsing
- Additional Commands
- Firewall
- Decryption
- SMTP
- Email MDN
- Retry
- Miscellaneous

1. Edit the properties as required.
2. To save your changes, click Save.

Creating a new Host

To create a new Host.

1. Click the HOSTS link in the Web Admin navigation bar.
2. Click Add Host and select the Host type from the list.
3. Input property values based on the host type that you selected. The available categories and properties differ depending on your Host type.
4. To save your changes, click Add Host.

The new Host is added to the list of Hosts.

Supported Host Types

To add a host: In the top menu bar, click HOSTS. On the right side of the screen, click Add Host and select a host type.

Host type	Description
Local filesystem	The local computer on which MOVEit Automation is running. This host appears automatically in the list. You do not need to explicitly add it. ***Field descriptions - General Properties*** (on page 305). ***Field descriptions - Additional Properties*** (on page 306)
UNC Share	Any remote Windows shares that MOVEit Automation is configured to access. ***Field descriptions - General Properties*** (on page 309) ***Field descriptions - Additional Settings*** (on page 310)
MOVEit Transfer Server	A MOVEit Transfer server accessible via HTTPS, usually over TCP port 443. ***Field descriptions - General Properties*** (on page 313). ***Field descriptions - Additional Settings*** (on page 314)
FTP/FTPS Server	A plain FTP server or FTP over SSL server, usually accessible over TCP port 21. The most common TCP port used for FTP over SSL in implicit mode is 990. ***Field descriptions - General Properties*** (on page 316) ***Field descriptions - Additional Properties*** (on page 319)
SSH/SFTP Server	An FTP over SSH server, usually accessible over TCP port 22. ***Field descriptions - General Properties*** (on page 325). ***Field descriptions - Additional Properties*** (on page 326)
POP3 Server	An inbound email server usually accessible over TCP port 110. ***Field descriptions - General Properties*** (on page 331). ***Field descriptions - Additional Properties*** (on page 332)
SMTP Server	An outbound email server usually accessible over TCP port 25. ***Field descriptions - General Properties*** (on page 333) ***Field descriptions - Additional Properties*** (on page 334)

Host type	Description
AS1 (SMTP/POP)	An AS1 trading partner relationship. An AS1 host defines the parameters for transferring files to and from a partner via the AS1 protocol. AS1 uses email (SMTP and POP3) transports, often with SSL transport security. ***Field descriptions - General Properties*** (on page 335). ***Field descriptions - Additional Properties*** (on page 336)
AS2 (HTTP/S)	An AS2 trading partner relationship. AS2 uses mostly web (HTTP) transport, often with SSL transport security. ***Field descriptions - General Properties*** (on page 338). ***Field descriptions - Additional Properties*** (on page 340)
AS3 (FTP/S)	An AS3 trading partner relationship. AS3 uses FTP transport, often with SSL transport security. ***Field descriptions - General Properties*** (on page 341). ***Field descriptions - Additional Properties*** (on page 343)

Configure a Proxy Server

Configure a proxy server for an SSH/SFTP host.

1. Create an SSH/SFTP host. For more information, see Hosts.
2. On the HOSTS page, select the host that you created. The properties page for the host opens.
3. On the Proxy Server row, click Edit.
4. Fill in the fields. For more information, see ***SSH/SFTP Host - Additional Properties*** (on page 326).

Tests Performed on Hosts

When you add a host, a Test button on the dialog box checks the configuration information that you provided. The following tests are executed, based on the type of host.

Host Type	Tests
MOVEit Transfer Server	Signs on to the configured host and port using the configured security settings and username and password. If successful, attempts to execute a directory listing on the root directory.

FTP Server	Signs on to the configured host and port using the configured security settings, transfer mode, username, password, and account. Uses any advanced options configured, such as client certificate and/or NAT settings. If successful, attempts to execute a directory listing on the current directory returned by the signon transaction.
SSH Server	Signs on to the configured host and port using the configured host key setting and username and password. If a client key is configured in the advanced options, it is used. If successful, attempts to execute a directory listing on the current directory that is returned by the signon transaction.
SMTP Server	Prompts for an email address to which to send a test email message. Attempts to sign on to the configured host and port and send an email message using the configured sender address and the provided recipient address. Note: A successful test here indicates only that MOVEit Automation was able to connect to the SMTP server and request that it send an email message to the provided recipient address. To confirm that the test was successful, you must check the provided recipient address to confirm that the test message was accepted and delivered.
POP3 Server	Signs on to the configured host and port using the configured username and password. If successful, attempts to get a count of waiting messages.

Enabling File Notifications

This feature is available on MOVEit Transfer and filesystem (Local Filesystem and UNC Share) hosts.

1. Select HOSTS. Click the name of the host.
2. In the General section, click Edit. The Edit *host-type* dialog box opens.
3. Select the Use File Notifications checkbox.
4. Click Save.

Note: On remote filesystems, file notifications work only for remote servers running Windows operating systems. Remote servers running Windows 9x or a non-Microsoft operating system do not usually provide file notifications even though they might be able to share files.

See also:

- ***Set the Notification Polling Interval*** (on page 26)
- ***File Notifications and Schedules*** (on page 27)
- ***How File Notifications Work*** (on page 26)

How File Notifications Work

File notifications allow MOVEit Automation to start a task based on the arrival of a file in a directory. The task runs only when the files that are associated with the task are available. This feature is also known as *event driven transfer*. It is available for MOVEit Transfer and filesystem sources.

For MOVEit Transfer Servers: MOVEit Automation software polls MOVEit Transfer servers for a list of recently-arrived files. One request is made per polling interval, regardless of the number of tasks affected. Because MOVEit Automation does not sign on and sign off for each request, this process is more efficient than having many tasks that each run and check for recently arrived files.

The MOVEit Automation notifier uses the username and password that are associated directly with the host. Tasks with sources that override this username might not be notified promptly.

If the username cannot sign on: MOVEit Automation waits for a period of 60 times the notification polling interval, up to a maximum of 3600 seconds, and then tries again. This extra delay prevents MOVEit Transfer from locking out the user or IP address

For filesystem notifications: The notifier uses Windows directory change notifications, using `ReadDirectoryChangesW`. When a file arrives, MOVEit Automation waits until the file is no longer locked by another process, and then runs the corresponding tasks.

To set the notification polling interval:

1. In the top menu bar, click SETTINGS > System Settings.
2. In the Tasks row, click Edit. The Edit Tasks dialog box opens.
3. Set the Notification Polling Interval. For more information, see ***System-wide Task Settings*** (on page 347).

File Notifications and Task Schedules

Like the scheduler, File Notifications require a task to have a schedule in order to run that task. As with the scheduler, the notifier runs the task only during the scheduled intervals.

However, tasks for which all sources use notifications are run much less frequently than when file notifications are not used. Those tasks will not be run periodically as specified by the task interval (for example, every 15 minutes). Instead, with a few exceptions, the task is run only when files that match one of that task's sources arrive. If all sources use File Notifications, the Repeat only until first success option on a schedule is ignored.

Exceptions:

- To catch files that arrive outside the scheduled hours, the task is run at the beginning of each schedule associated with that task, even when no notifications have arrived.
- At startup, MOVEit Automation runs all tasks that are subject to notifications and whose schedule spans the time that MOVEit Automation is starting. This is to catch files that arrived during the time that MOVEit Automation was not running.
- If a schedule spans midnight (for example, 20:00 to 04:00), then the task is always run at midnight.
- If a schedule is marked Log failure if no files found during scheduled run and no files have been found yet during the schedule, the task is run at the end of the schedule, so that it may fail if necessary.
- Schedules marked Run even if notifications are enabled for the host cause the task to run even if notifications are enabled. Note that a task may have multiple schedules, and each schedule has its own setting for this option..

  For more information, see ***Add/Edit Task Schedule*** (on page 85).

For example, if a task is scheduled to run every 15 minutes between 08:00 and 17:00, and all of its sources are marked to use notifications, then the "every 15 minutes" portion of the scheduling information is ignored. Instead, the task is run at 08:00 (in order to catch any files that may have arrived overnight), and thereafter the task is run only when a file arrives.

If only some of a task's sources correspond to hosts for which notifications have been enabled, then the task will be run both by the normal scheduler and by the notifier.

The arrival of a given file might cause several tasks to run, if those tasks are all watching the same directory. The notifier respects the Include Subdirectories option.

TASKS

The simultaneous arrival of multiple files that are being looked for by a task do not necessarily cause that task to be run multiple times. However, if a task is already running when files for that task arrive, the notifier queues a request to run the task again when the task completes.

A task is used to exchange files between multiple systems using multiple protocols, and process files with built-in functions and custom scripts. Tasks define how, where, and when data is transmitted or manipulated.

Types of tasks:

- Traditional tasks pull files from sources and push them to destinations according to schedules, and can use processes to run scripts. For more information, see Configuring a Traditional Task.
- Advanced tasks pull files from sources and push them to destinations according to schedules, and can use processes to run scripts. They can also provide conditional processing (IF), and include task loops (FOR). For more information, see ***Configuring an Advanced Task*** (on page 33).
- Synchronization tasks replicate the contents of two folders, in one-way or two-way directions. Options control whether deletions and extra files are permitted. For more information, see ***Configuring a Synchronization Task*** (on page 35).

What comprises a task?

In the table, an asterisk (*****) indicates that the element can be used in that task type. For more information, click the links in the table.

Task Elements	Traditional	Advanced	Sync
Steps			
Source (on page 53)	*	*	
Process (on page 71)	*	*	
Destination (on page 72)	*	*	
File Loop (on page 80)		*	
Conditional Branch (if) (on page 81)		*	
Update Original (rename or delete) (on page 90)		*	
Send Email (on page 82)		*	
Run Task (on page 83)		*	
Next Actions			
Next Action - Send Email (on page 83)	*		*
Next Action - Run Task (on page 84)	*		*
Schedule			
Add/Edit Task Schedule (on page 85)	*	*	*
Define Sync Folders			*
Define Folder A (on page 36), ***Define Folder B*** (on page 44)			*

The TASKS page lists the tasks and provides access to the task-related functions.

§ Bulk Actions

Bulk actions occur on *all tasks that are listed on the TASKS page* (including multiple pages). Before you perform a bulk action, ***use the filters*** (on page 12) to limit the list of tasks to those on which you want to perform the bulk action. For more information, see Bulk Actions.

§ Add Task

Add a new traditional, advanced, or synchronized task.

The ***individual taskname*** page details the task elements and provides access to task editing options.

§ Actions dropdown for actions to perform on the individual task, such as Run Now, Delete the task, Export. Available options correspond to the type of task and the task schedule status. For more information, see Task Actions.

§ Add or edit one or more of the following elements; Step, Schedule, Next Action.

See also:

§ ***Navigating Web Admin*** (on page 9)

§ ***Filtering and Sorting*** (on page 12)

Task List

TASKS

Each row details a single task including the task name, schedule status, and the start time of its last run. Other information in the row (Result, and Files Sent) pertains to the last run.

The list can span multiple pages.

Ø *To limit the number of tasks shown in the list*

§ In the left panel, ***select filters*** (on page 12).

Ø *To view the result details of a task's previous runs:*

§ In the task row, Actions column, select View Run History. The Task Run tab opens, showing the runs for the task.

Ø *To pause, resume, or refresh the current page:*

1 Click the more options icon ⋯ above the table, and select one of the following options.

§ To pause the page refresh, click Pause Page Refresh.

§ To resume the page refresh, click Refresh Page Now.

§ To refresh the page, click Refresh Page Now.

More Options

The more options menu for each task provides options to explore specific information for this task. Options available depend on the task schedule status and whether it is running.

- Run Task Now. (Only available for inactive tasks. Not available for ***incomplete tasks*** (on page 30).)
- Stop Running Task. (Only available for running tasks. Not available for ***incomplete tasks*** (on page 30).)
- Enable/Disable Schedule. (Only available for scheduled tasks.) For more information, see ***Task Schedule Status*** (on page 30).
- View Task Config. Opens the individual task page.
- View Run History. Opens the Task Run tab for the task.
- View File Activity. Opens the File Activity tab for the task.
- View Audit Records. Opens the Audit tab for the task.

See also:

- ***Task Schedule Status*** (on page 30)
- ***Bulk Actions*** (on page 100)
- ***About Tasks*** (on page 27)
- ***Common Activities*** (on page 17)

Task Schedule Status

The Schedule Status of each task is listed on the TASKS page. You can ***filter the task list*** (on page 12) by task Schedule.

Each task is in one of the following task schedules, which indicates whether and how the task can be run.

Schedule	Description
Enabled	Schedule Enabled § Have either a minimum of one source and one destination, or one process. § Have a schedule. § Are run by the MOVEit Automation scheduler or started by file notification events. § Can be started by Run Now commands, or by Next Actions or scripts that start tasks. A task must be explicitly enabled after it has all the required components.
Disabled	Schedule Disabled § Have either a minimum one source and one destination, or a minimum of one process. § Have a schedule. § Have been marked Disabled by an administrator or a task clone operation. § Can be started by Run Now commands, or by Next Actions or scripts that start tasks. § Will not be run by the scheduler or by file notification events.

Schedule	Description
Unscheduled	Unscheduled tasks § Have either a minimum one source and one destination, or a minimum of one process. § *Do not* have a schedule. § Can be started by Run Now commands, or by Next Actions or scripts that start tasks. § Will not be run by the scheduler or by file notification events.
Config Incomplete	Incomplete tasks § Are missing a key task element: either one source and one destination, or one process. § *Cannot* be run by the scheduler or by file notification events § *Cannot* be started with Run Now, or by Next Actions or scripts that start tasks. Attempts to do so will fail.

See also:

§ ***Run Now*** (on page 95)
§ ***Schedules*** (on page 85)
§ ***Next Action - Run Task*** (on page 84)
§ ***File Notifications*** (on page 25)

Add Task

A task consists of one or more steps, to include a source and a destination, and a schedule. Further task options include a process that runs a script, and advanced functions such as loops and conditional blocks.

You can add new tasks and import tasks in MOVEit Automation. Task import is only available to Service Pack 1 users.

For more information, see :

§ ***Add New Task*** (on page 31)
§ ***Import Task*** (on page 50)

Add a New Task

You can use MOVEit Automation to add a new task.

To add a new task, complete the following steps.

1 Open MOVEit Automation and select the TASKS tab.
2 In the Add Task list, click + Add Task.
3 Input a name in the Friendly Name field.
4 Select one of the following task types, and click Add Task.

- Traditional
  - Task that includes a source, destination, and schedule, and optionally a process that runs a script.
- Advanced
  - Task that includes advanced functions such as loops and conditional blocks in addition to sources, destinations, and processes.
- Synchronization
  - Task that replicates the contents of two folders to keep the contents in sync, with options to control whether deletions or extra files are permitted.

For more information, see :

- ***Configuring a Traditional Task*** (on page 32)
- ***Configuring an Advanced Task*** (on page 33)
- ***Configuring a Synchronization Task*** (on page 35)

Configuring a Traditional Task

Traditional tasks pull files from sources and push them to destinations according to schedules, and can use processes to run scripts.

Prerequisite: Hosts must have already been added to MOVEit Automation.

To create a Traditional task:

1. In the top menu, click TASKS. At the upper right, click Add Task. Specify a name, select Traditional, and click Add Task.

   The new task is created as an empty task.
2. Click Step and select Source. The Add Source dialog box opens. Select how to load the source file, select a host, and click Next.
3. The properties page for the source opens. Make your selections. For more information, see ***Add a Task Source*** (on page 53).
4. Click Step and select additional elements. At a miminum, the task must have a Source and a Destination, or a Source and a Process.
5. Click Schedule. Task without schedules can be run only with Run Now.
6. Optionally, select a Next Action. See below for details.

Elements in Traditional Tasks

Typically, you define the elements in the order shown.

- Source (on page 53): A source defines a single location from which files are obtained for use in a task. Each source is a reference to a host. A task can have an unlimited number of sources.
- Process (on page 71): In a task, a process runs a single built-in script or custom script. The script must exist before you can add it to a process. For more information, see ***Scripts*** (on page 103). In a traditional task, the process is run on all the files that you defined in the source.
- Destination (on page 72): A destination defines a single location to which files are sent when a task runs. A task can have any number of destinations, or no destinations. If a task has no destinations, it must have at least one process to be eligible to run.
- Schedule (on page 85): A task can have more than one schedule. To run automatically, a task must have a minimum of one schedule. Tasks with or without schedules can be run manually by the operator.
- Next Action - Send Email (on page 83): Sends an email based on the results of the task.
- Next Action - Run Task (on page 84): Runs another task based on the results of the task you are configuring.

Examples of What Traditional Tasks Can Do

- Run processes.
- Delete/rename/move files on sources after copying them to destinations.
- Rename downloaded files and folders before writing or creating them on destinations.
- Pull from multiple destinations in a single task.
- Push to multiple destinations in a single task.
- Work with AS1, AS2, AS3 and SMTP/POP3 (email) sources and destinations.
- Select source files based on specific date criteria (such as "older than 60 days").
- Handle blind downloads, typically through FTP servers that do not provide directory listings.
- Zip or unzip files.
- Issue per-file FTP commands.

Configuring an Advanced Task

Advanced tasks pull files from sources and push them to destinations according to schedules, and can use processes to run scripts. They can also provide conditional processing (IF), and include task loops (FOR).

Prerequisite: Hosts must have already been added to MOVEit Automation.

To create an Advanced task:

1. In the top menu, click TASKS. On the TASKS page, at the upper right, click Add Task. Specify a name, select Advanced, and click Add Task.

   The new task is created as an empty task.
2. Click Step and add an element.

You can also create an Advanced task by creating it from a Traditional task. For more information, see ***Create Advanced Task from Task*** (on page 99).

Elements in Advanced Tasks

Advanced tasks can contain these elements: ***source*** (on page 53)s, ***destination*** (on page 72)s, ***schedule*** (on page 85)s, and ***process*** (on page 71)es. They can also include the following advanced elements:

- File Loop (FOR): A file loop causes a set of steps to be performed one time for each file that has been downloaded or created so far in the task. An unlimited number of steps can appear inside a loop. Any task element, except a schedule, can be inside a loop. For more information, see ***Add File Loop*** (on page 80).
- Conditional branch (IF): A conditional branch (IF block) defines a condition, and the set of actions to be performed if that condition is met or not met. You can configure a conditional branch anywhere in an Advanced task, including inside a file loop. Any task element except a schedule can appear inside a conditional branch. For more information, see ***Add Conditional Branch*** (on page 81).
- Update Original File: Use this step after a transfer is successful to delete or rename original files. Typically used in a file loop. For more information, see ***Update Original*** (on page 90).
- Send Email: Adds a Send Email step to an Advanced task. The email is sent regardless of the task results. The step can be placed anywhere in the task. For more information, see ***Send Email*** (on page 82).
- Run Task: Runs another task as a step in an Advanced task. The step can be placed anywhere in the task. For example, you can use Run Task inside a conditional branch to process only those files that meet a certain condition. For more information, see ***Run Task*** (on page 83).

Advanced Task Processing

The position of elements in an Advanced task determines the processing flow of the task:

- Elements are processed from top to bottom.
- Elements that are inline (left aligned, rather than indented) are considered independent elements, which means processing is completed for one before moving on to the next.
- An element that is indented under a File Loop is processed for each file in the list of source files.
- An element that is indented under a Conditional branch (If Block) is executed only if the specified conditions are met.
- You can move elements within an Advanced Task without needing to create or delete elements.

Configuring a Synchronization Task

Synchronization tasks replicate the contents of two folders, in one-way or two-way directions. Options control whether deletions and extra files are permitted.

Prerequisite: Hosts must have already been added to MOVEit Automation.

Sync tasks can be performed on files and folders on the following host types: Local file, mapped drive or UNC path, MOVEit Transfer servers, FTP and FTP/S servers, and SFTP servers.

Sync tasks can be unidirectional or bidirectional. Newly-created tasks are unidirectional. Before you can change the sync direction, you must define Folder A and Folder B.

- Unidirectional tasks replicate content from Folder A to Folder B, indicated by a single arrow 
  A unidirectional task copies items from Folder A into Folder B, without changing any existing non-matching items in Folder B. For example, a unidirectional task can be used for backup purposes. To further specify how to handle files whose name and size match, see ***Initial Run Options*** (on page 46).
- Multidirectional tasks replicate content from Folder A to Folder B, and from Folder B to Folder A, indicated by two arrows.  .

To create a sync task:

1. Click TASKS. On the Tasks page, click Add Task. Provide a Friendly Name, select Synchronization, and click Add Task.
   The task page opens. When first created, the sync task is unidirectional, shown by a one-way arrow.
2. Click Folder A. The Add new sync folder A dialog box opens.
3. Select the type of host to sync from, and select a host. Click Next.
4. Provide General settings. For more information, see ***General settings - Sync folders*** (on page 37),
5. Provide Task Override settings for Folder A. The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.
   - ***Local Folder*** (on page 60)
   - ***Mapped Drive or UNC path*** (on page 60)
   - ***MOVEit Transfer server*** (on page 61)
   - ***FTP or FTP/S server*** (on page 62)
   - ***SFTP server*** (on page 64)
6. If the source is an FTP or FTP/S server, click the ***Additional Commands*** (on page 44) tab and optionally specify commands to execute per file before and after the synchronization occurs.
7. Click Folder B and make your selections. For field description information, see the links in Steps 4, 5, and 6, above.
8. Specify the sync type (direction) and how to handle deleted files: In the upper right of the sync task page, click Actions > Task Settings. The settings page opens. In the Sync row, click Edit. For more information, see ***Task Settings - Sync Info*** (on page 93).
9. Click Schedule and ***add a schedule*** (on page 85).

10 Optionally, click Next Action and add a ***Send Email*** (on page 83) and/or ***Run Task*** (on page 84) action.

11 Preview the task. Click Actions > View Sync Preview. The ***Sync Preview*** (on page 45) dialog box opens.

12 In the Sync Preview dialog box: To determine how the task handles files on Folders A and B whose name and size match, click ***Change Initial Run Options*** (on page 46) and make a selection.

13 To run the task, click Actions > Run Now.

Note: If you include a special filter when you configure Synchronization task's ***General settings*** (on page 37), the fields accept only those items that match the filter. For example, if the filter is >100MB, the only things that the sync considers are those larger than 100MB.

In this section:

Add/Edit Sync Folder A

Settings in Folder A define the folders and files used in the synchronization task, and control what happens during the task.

To access these settings:

§ New sync tasks: Click TASKS > Add Task and create a Synchronization task. In the task, click inside Folder A. Select where to sync from.
§ Existing Folder A: Do either of the following:
  § Click the folder's gear icon and select Edit. -or-
  § Click the gear icon and select Change Host. Select where to sync from, and select a host.

General settings options are the same for all host types. For more information, see ***Sync Folders - General Settings*** (on page 37).

Host Override settings differ depending on the host type. See host override settings for:

§ ***Local Folder*** (on page 60)
§ ***Mapped Drive or UNC path*** (on page 60)
§ ***MOVEit Transfer server*** (on page 61)
§ ***FTP or FTP/S server*** (on page 62)
§ ***SFTP server*** (on page 64)

Additional Commands (on page 44) settings are used in FTP and FTP/S hosts.

Sync Folders - General Settings

These settings descriptions pertain to all source types used in Folder A and Folder B.

Sync Folder - General tab	Description
Folder(s)	Folder name or path where files are located on the host. Browse button is not available if the source uses an FTP or SSH host with Blind Downloads enabled.
For all host types	***Macros*** (on page 176) are allowed in this field. Wildcards (*) are permitted.

For Filesystem, MOVEit Transfer, FTP, and SSH hosts	Note: Do not use a wildcard (* or **) in the first position of the path.
	Example of *
	`/home/jal/data*/reports` matches any folder named **reports** that is a direct subfolder of a folder whose name starts with **data** and is a subfolder of `/home/jal`.
	§ The following folders match:
	`/home/jal/data/reports`
	`/home/jal/dataaccounts/reports`.
	§ The following folder does not match:
	`/home/jal/reports`
	MOVEit Automation specific wildcard operator ** matches any number of intermediate subfolders. A trailing ** means the same as checking **Include Subdirectories**
	Example of **
	`/home/jal/**/reports` matches any folder named **reports** that is a direct or indirect subfolder of `jal`.
	§ The following folders match: `/home/jal/**/reports`:
	`/home/jal/data1/2005/07/reports`
	`/home/jal/2005/reports`
	`/home/jal/reports`
	§ The following folders *do not match* `/home/jal/**/reports`:
	`/home/jal/2005/report1`
	`/home/jal/2005/reports/mydata`
	Note: The folder `/home/jal/2005/reports/mydata` would match if **Search Subdirectories** was checked.

Folders checkboxes	

§ Ignore folders	If checked, optionally specify a folder mask. Folder masks must reflect folder names, not full or relative folder paths. The folder masks match against the names of subfolders in the current folder being searched. To specify multiple folders, use a semicolon (;) separated list with no spaces. ***Macros*** (on page 176) are allowed in this field.
§ Search subdirectories	Searches subdirectories of the specified folder (as well as the folder itself) according to the specified file mask. If not checked, the sync task copies subfolders that are located one level below the specified folders, but *does not* pick up any files or subfolders in those subfolders.
File(s)	The filename or filemask used to select files on the remote host. ***Macros*** (on page 176) are allowed in this field. Note: When Collect only new files is selected, if you change the filemask on an existing source, doing so could cause old files to be downloaded from the source host. File collection timestamps are stored by filemask internally in MOVEit Automation. Changing the filemask negates existing timestamps.
Files checkboxes	
§ Ignore file(s)	Select and specify the files to ignore. ***Macros*** (on page 176) are allowed in this field.
§ Special filters	Specifies files based on size and date/time of last modification. Select, then click Edit to specify filter criteria. Choose to match any or all criteria. Note: If you set a special filter, the synchronization considers only those items that match the filter. For example, if the filter specifies > 100MB, the only things that the sync considers are those larger than 100 MB.

Sync Folders - Host Override settings

The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.

Source override settings depend on the type of host that the sync folder references. In the following list, click the type of host used by the sync folder.

§ ***Local folder*** (on page 39)
§ ***Mapped drive or UNC path*** (on page 39)
§ ***MOVEit Transfer server*** (on page 40)
§ ***FTP or FTP/S server*** (on page 41)
§ ***SFTP server*** (on page 43)

Sync Folders - Local Folder, Mapped Drive, UNC Source Override

The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.

Settings in this topic pertain to sync folders that use Local File System hosts or UNC hosts. For information about source overrides for other host types, see ***Sync folders - Host Override settings*** (on page 38).

Local File System or UNC - Host Override tab	Description
Timeouts tab	
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
Override Default Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.

Sync Folders - MOVEit Transfer Source Override

The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.

Settings in this topic pertain to sync folders that use MOVEit Transfer hosts. For information about source overrides for other host types, see ***Sync folders - Host Override settings*** (on page 38).

MOVEit Transfer - Host Override tab	Description
Authentication tab	
§ Override Default Credentials (MOVEit Transfer, FTP, SSH servers)	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Certificate (MOVEit Transfer and FTP servers)	This option is available for MOVEit Transfer and FTP sources only. Click Set Cert and select a certificate
File Sorting tab	
§ Override Default File Sorting	Select to specify how to sort files.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Sync Folders - FTP, FTP/S Source Override

The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.

Settings in this topic pertain to sync folders that use FTP or FTP/S hosts. For information about source overrides for other host types, see ***Sync folders - Host Override settings*** (on page 38).

FTP - Host Override tab	Description
Authentication tab	
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Certificate	Click Set Cert and select a certificate.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Data timeout	Number of seconds to wait when sending data to or receiving data from the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
§ Override Default Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Transfer tab	

FTP - Host Override tab	Description
§ Override Default Transfer Type	ASCII or Binary mode
§ Override Default Transfer Mode	§ Active: normal mode of operation for FTP transfers § Passive: typically used for FTP clients that are located behind a firewall.
§ Override Default XSHA1 Setting	The XSHA1 command is used if the FTP server supports it. The file that is received or transmitted is compared with the copy of the file that is on the server, by comparing SHA1 hashes of the file.
§ Override Default Blind Download	When blind downloads are enabled, MOVEit Automation does not use any directory listing commands. The blind download option is rarely used, and is intended primarily to accommodate unusual FTP servers.
§ Override Default Resume	For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
§ Override Default Reuse SSL Session	Forces data connections to use the same SSL session as the existing control connection. Use to override the setting for a given source or destination task element configuration, so that you comply with partner server settings that require reuse of an SSL session for data connections.
Download Limits tab	
§ Override default file count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, the task is automatically rerun.
§ Override default file size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, the task is automatically rerun.
Security tab	
§ Override default Use MD5	For more information, see ***FTP Host - Additional Properties: Use MD5*** (on page 319)
§ Use MD5	Make your selections.
§ Override Default MD5 Filename	Default is `MD5SUM`
§ MD5 Filename	Name of the MD5 file to look for. Default is MD5SUM.

Sync Folders - SFTP Source Override

The sync folder's Host Override settings override the corresponding settings that are configured in the host. Your selections affect the sync folder as it is defined in the context of the task.

Settings in this topic pertain to sync folders that use SFTP hosts. For information about source overrides for other host types, see ***Sync folders - Host Override settings*** (on page 38).

SFTP - Host Override	Description
Authentication tab	
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Key	Click Set Key and select a key.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
§ Override Default Transfer Rescan time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Transfer tab	
§ Override Default Blind Download	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
§ Override Default Resume	For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
Security tab	
§ Override default Use MD5	For more information, see ***FTP Host - Additional Properties: Use MD5*** (on page 319)
§ Use MD5	Make a selection.
§ Override Default MD5 Filename	Default is `MD5SUM`
§ MD5 Filename	Name of the MD5 file to look for. Default is MD5SUM.

Sync folder - FTP - Additional Commands

These settings pertain to Synchronization tasks, when defining Folder A or Folder B as an FTP or FTP/S host.

Folder A Commands

Folder A	Description
Commands to execute (per file) before transfer	Quote commands to execute on files in Folder A before the sync occurs. ***Macros*** (on page 176) are allowed in this field.
Commands to execute (per file) after transfer	Quote commands to execute on files in Folder A after the sync occurs. ***Macros*** (on page 176) are allowed in this field.

Folder B Commands

Folder B	Description
Commands to execute (per file) before transfer	Quote commands to execute on files in Folder B before the sync occurs. ***Macros*** (on page 176) are allowed in this field.
Commands to execute (per file) after transfer	Quote commands to execute on files in Folder B after the sync occurs. ***Macros*** (on page 176) are allowed in this field.

Add/Edit Sync Folder B

Settings in Folder B determine where the folders and files from Folder A are synced to. You can also specify Host Override settings.

To access these settings:

- New sync tasks: Click TASKS > Add Task and create a Synchronization task. In the task, click inside Folder B. Select where to sync to.
- Existing Folder B: Do either of the following:
  - Click the folder's gear icon and select Edit. -or-
  - Click the gear icon and select Change Host. Select where to sync to, and select a host.

General settings

Folder B General tab	Description
Folder(s)	Folder name or path where files are located on the host. ***Macros*** (on page 176) are allowed in this field. Wildcards (*) are permitted.

Host Override settings differ depending on the host type. See host override settings for:

- ***Local folder*** (on page 39)
- ***Mapped drive or UNC path*** (on page 39)
- ***MOVEit Transfer server*** (on page 40)
- ***FTP or FTP/S server*** (on page 41)
- ***SFTP server*** (on page 43)

Additional Commands (on page 44) settings are used in FTP and FTP/S Hosts

View Sync Preview

This topic applies to Sync tasks.

The sync preview dialog box shows the actions that the sync will perform the next time the task runs.

Note: If you View Sync Preview immediately after the task runs, and no changes have occurred in Folders A or B since the task ran, the Sync Preview shows no actions.

Prerequisite: The Sync task must have a Folder A and Folder B defined.

To access this dialog box: TASKS > *Click a Sync task name* > Actions > View Sync Preview.

Note: Depending on the number of files and folders in the sources, it can take several minutes or longer for MOVEit Automation to compile a list of changes that the sync task will make. For performance reasons, the Preview lists a maximum of 1,000 entries. If more than 1,000 entries are received, an alert message indicates the number of total entries.

Sync Preview Field	Description
Path column (Folders A and B)	The path relative to the location defined in Folder A or Folder B. Lists the items to be acted upon.
Action	Shows an icon for the action to take place, and a direction arrow. § Copy  § Add  To specify how to handle subfolders and their contents during a sync task, see ***Sync Folders - General Settings*** (on page 37). § Delete  . To specify how to handle items that are deleted from Folder A and or Folder B, see Sync Delete Options in ***Task Settings - Sync Info.*** (on page 93)
Timestamp	Timestamp of the item.

Sync Preview Field	Description
Change Initial Run Options	Determines how to handle items whose name and size match in Folder A and Folder B. For more information, see ***Run Options*** (on page 46).

Change Initial Run Options - Sync Task

Initial Run options determine how to handle files whose name and size match in Folder A and Folder B.

To access this dialog box:

1. Select TASKS > *Click a Sync task name.* On the right side of the page, click Actions > View Sync Preview.
2. In the Sync Preview dialog box, click Change Initial Run Options.

Initial Run Option	Description
Consider files whose name match to be identical.	Files that meet these criteria are not copied.
Consider files whose name and size match to be identical.	Default. Files that meet these criteria are not copied.
Consider files whose name and size match to be identical only if their timestamps are within the following timespan.	Enforces a time limit to determine whether files match. Specify an interval. Files whose name and size match, and have timestamps within the interval you select, are not copied.
Copy all files from Folder A to Folder B.	Copies all files, regardless of name, size, and timestamp. Note: Copy actions might take some time.

Sync Tasks - Actions

The following actions pertain to Synchronization tasks. For actions that pertain to all tasks, see ***Task Actions*** (on page 95).

To access this menu: Click TASKS and click a synchronization task name. At the upper right of the page, click Actions.

Actions - Sync tasks	Description
Swap Sync Folders	Folder A and Folder B are swapped; that is, Folder A's definition becomes Folder B's definition, and Folder B's definition becomes Folder A's definition. The saved directory listings that define which files and folders are ignored, moved, created, etc. are also swapped.

Actions - Sync tasks	Description
Clear Sync Listings	Causes the sync task to be treated as a new sync task. Might cause MOVEit Automation to re-transfer files and folders that have already been moved. The sync task is marked Disabled. Before you can re-enable the task, you must run a Sync Preview.
View Sync Preview	Opens the Sync Preview dialog box. For more information, see ***View Sync Preview*** (on page 45).

To change sync direction:

§ Click TASKS and click a sync task. Click Actions > Task Settings. In the Sync row, click Edit. Select a Sync Type. (This option is not on the Actions menu.)

Note: Before you can change the sync direction, you must ***Add Folder A*** (on page 36) and ***Add Folder B*** (on page 44).

Macro - Sending Sync Actions in Next Action Email

Synchronization tasks can use the same types of Next Actions as traditional tasks, which include:

§ ***Add a Send Email step*** (on page 83)
§ ***Run Task*** (on page 84)

Special Macro(s) for Synchronization Tasks

You can use the macro [SyncReport()] macro to generate a report of all synchronization actions and send the report in a Next Action email notification. For more information, see ***Macro Keywords*** (on page 178).

Format of the Sync Report:

§ Folder A: (folder path and host of Folder A)
  § X files were copied from Folder B
    · (Specific list of files, if any)
  § X empty folders were created in Folder A
    · (Specific list of folders, if any)
  § X files were deleted from Folder A
    · (Specific list of files, if any)
  § X folders were deleted from Folder A
    · (Specific list of folders, if any)
§ Folder B: (folder path and host of Folder B)
  § X files were copied from Folder A
    · (Specific list of files, if any)
  § X empty folders were created in Folder B
    · (Specific list of folders, if any)
  § X files were deleted from Folder B

- (Specific list of files, if any)

§ X folders were deleted from Folder B

- (Specific list of folders, if any)

Individual file listings are indented four spaces, start with a path, and include last modified date/time and size in parenthesis. For example:

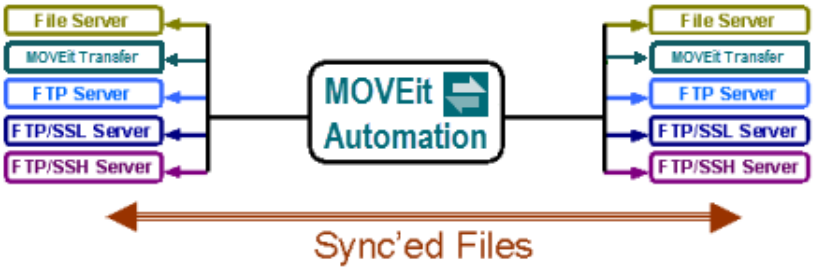
```
Reports/C13543/urlscan_unattend.txt  (2007-03-28 12:55:45,  32 bytes)
```

Individual folder listings are indented four spaces and only include a path. For example:

```
Reports/D20130
```

More About Synchronization

MOVEit Automation can replicate the contents of two folders to ensure that the files and folder structures remain in sync. Any two folders on the MOVEit Automation local hard drive, other Windows servers/shares, FTP servers, FTPS servers, SFTP servers and/or MOVEit Transfer servers can be involved in a single synchronization task.



Synchronization tasks consist of Folder A, Folder B, and a sync direction, which can be unidirectional (from Folder A to Folder B) or bidirectional (Folder A to Folder B, and Folder B to Folder A). Additional options control how to handle deletions and extra files.

Synchronization tasks:

§ Have a ***schedule*** (on page 85).

§ Can be event driven if hosts permit ***file notifications*** (on page 25).

§ Can be configured to transfer and/or exclude folders and files based on name, extension, or size. For more information, see ***Sync Folders - General Settings*** (on page 37).

§ ***Next Action - Send Email*** (on page 83) and ***Next action -Run Task*** (on page 84) actions can be included.

§ A special ***[SyncReport()] macro*** (on page 178) can be used to summarize actions taken by sync tasks.

Synchronization tasks can:

- Delete files and folders from one folder if MOVEit Automation notices they have been deleted from another folder.
- Replicate (add and trim) empty folder structures. (Traditional tasks only create folder structures, and do so only if files are present in them.)
- Respond to file and folder create, delete and rename events. (Traditional tasks only react to file create or rename events.)
- Be configured to automatically handle new files and create destination subfolders as necessary.

Synchronization tasks cannot:

- Run processes.
- Delete/rename/move files on sources after copying them to destinations.
- Rename downloaded files and folders before writing or creating them on destinations.
- Pull from multiple destinations in a single task.
- Push to multiple destinations in a single task.
- Work with AS1, AS2, AS3 and SMTP/POP3 (email) sources and destinations.
- Select source files based on date criteria (such as "older than 60 days").
- Handle "blind" downloads (typically through FTP servers that do not provide directory listings).
- Zip or unzip files.
- Issue per-file FTP commands.

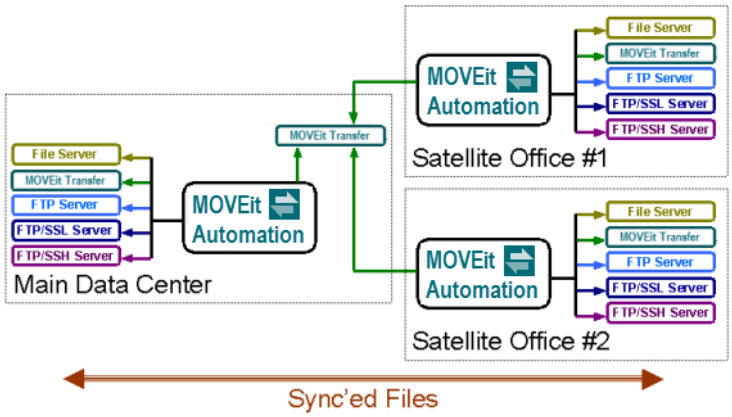
- Permissions and Settings

In most cases, the account MOVEit Automation uses to authenticate to folders used in sync tasks requires permissions to:

- Read, write, list and delete files
- Overwrite existing files (this is a separate folder-level option on MOVEit Transfer folders)
- Create and remove subfolders files (this is Subs permission in MOVEit Transfer)

Synchronizing Content To or Between Multiple Remote Sites

MOVEit Automation can synchronize content to or between multiple remote sites across the Internet, even if the remote sites have no secure server. To achieve this, install MOVEit Automation software at each remote site and use one MOVEit Transfer server as a shared repository.



MOVEit Transfer Configuration

If each remote site contains unique content, set up a different folder for each remote site. Otherwise, set up a single folder containing shared content for all remote sites. Set up all folders to ALLOW overwrites and grant any related user-folder permissions Read, List, Write, Delete and Subs access.

Each remote site requires its own end user account or FileAdmin account. The main data center also requires a FileAdmin account.

Main Data Center MOVEit Automation Configuration

If each remote site contains unique content, set up a different sync task for each remote site's folder on MOVEit Transfer. Otherwise, set up a single sync task to move shared content for all remote sites. Set up every related sync task to have a MOVEit Transfer folder as Folder B, regardless of synchronization direction or other options.)

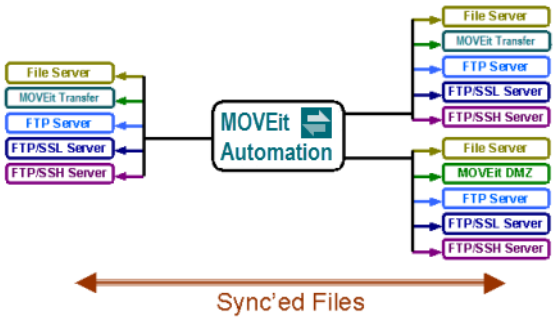
Remote Site MOVEit Automation Configuration

Regardless of whether each remote site has its own unique content, each remote site needs only one sync task to sync from either the remote site's unique content MOVEit Transfer folder or a shared MOVEit Transfer folder. Set up every related task to have a MOVEit Transfer folder as Folder A, regardless of synchronization direction or other options.

Synchronizing Content To or Between Multiple Folders

MOVEit Automation can synchronize content to or between more than two folders. To achieve this,

- Choose one older to serve as a *master* folder.
- Set up one task for each remote folder with which you want to synchronize the master folder. Every related task has the master folder as Folder A, regardless of synchronization direction or other options.



Import Task

Service Pack 1

You can import one or more tasks, which will add the imported tasks to the task list and allow you to use and run the task.

You can use MOVEit Automation to import a task from an XML task export file. The task's properties, actions, triggers, and settings are in an XML file.

To import a task, you must have administrative access.

To import a task, complete the following steps.

1 Open MOVEit Automation and select the TASKS tab.

2 In the Add Task list, click Import Task.

3 Browse to and select the XML task export file or files that you want to import. If the task contains password protected keys or certs, complete step 4, otherwise skip to step 5.

4 If the selected task contains password protected keys or certs, you must enter the password in the Enter Password dialog box and click OK.

Note: You cannot proceed without entering a password. If you input an incorrect password, the task is imported without the password protected keys or certs.

5 Review the file elements that are displayed in the Import Task(s) dialog box.

6 If an element to be imported matches an existing element in the config, the duplicate task element is accompanied by the overwrite icon (Overwrite). The new element will overwrite the existing element in the config if the task is imported without removing or renaming the duplicate element. For more information, see ***Resolve Duplicate Task Elements*** (on page 52)

7 Before the task or tasks are imported, you can rename or remove task elements. For more information, see ***Rename Task Element*** (on page 51) and ***Remove Task Element*** (on page 51).

8 The Disable the schedule(s) of imported task(s) and Filter to display the imported task(s) only check boxes are selected by default. For more information, see ***Import Options*** (on page 52)

9 Click Import.

The task is imported and the The task has been imported message displays.

Rename Task Element

To rename an element, complete the following steps.

1 Click the element to select it and enable the Rename button.

2 Click the Rename button.

3 Input a new name in the Rename Imported Item dialog box and click Save.

The word new in parentheses (New) is appended to the renamed task element in the Import Task(s) dialog box.

Remove Task Element

To remove an element, complete the following steps.

1 Click the element to select it and enable the Remove button.

2 Click the Remove button.

3 Click Remove in the Remove Imported Item dialog box.

Resolve Duplicate Task Elements

Duplicate task elements are accompanied by the overwrite icon (Overwrite) in the **Import Tasks** dialog box.

Choose one of the following options to resolve the duplicate element. The default option is **Overwrite**.

§ **Overwrite**

To overwrite the element, complete the following steps.

1. Click **Import**. The task is imported and overwrites the existing task in the application.

§ **Rename**

Renaming a duplicate task element creates a new element. This does not affect the export file.

To rename an element, complete the following steps.

1. Click the element to select it and enable the **Rename** button.
2. Click the **Rename** button.
3. Input a new name in the **Rename Imported Item** dialog box and click **Save**.

The word new in parentheses (New) is appended to the renamed task element in the **Import Task(s)** dialog box.

§ **Remove**

Removing a duplicate task element does not affect the export file.

To remove an element, complete the following steps.

1. Click the element to select it and enable the **Remove** button.
2. Click the **Remove** button.
3. Click **Remove** in the **Remove Imported Item** dialog box.

Import Options

There are two scheduling and filtering options available on the **Import Task(s)** dialog box. These options control the schedule of imported tasks and limit the number of items listed following the task import.

The check boxes are selected by default. To deselect the options, clear the check boxes.

Disable the schedule(s) of imported task(s): The imported task element schedules are disabled. This prevents an imported task from running before it is reviewed. To run an imported task, you must enable the task schedule. For more information, see ***Add/Edit Task Schedule*** (on page 30).

Filter to display the imported task(s) only: The task list filter is modified to list only imported tasks. This limits the number of tasks listed, to make it easier to find the imported tasks. To change the listed tasks, modify. For more information, see ***Searching, Filtering and Sorting*** (on page 12).

Task Elements

Add a Task Source

A source defines a single location from which files are obtained for use in a task. Each source is a reference to a host. A task can have an unlimited number of sources.

Sources can be added to traditional tasks and advanced tasks.

Prerequisite: Before you add a source to a task, you must have already added the host that the source references. For example, to add a source that is a folder on a MOVEit Transfer server, you must have previously added a MOVEit Transfer host for that server. For more information, see Add a Host.

To add a source:

1. Select TASKS. Add a new traditional or advanced task, or click an existing one. The task properties page opens.
2. Click Step > Source. If this is an advanced task, specify the location for the source step.
3. In the Add Source dialog box, select how you want to load the source file, and select a host, Click Next.
4. Click the tabs and make selections to specify the behavior of the source files.

For field descriptions, click the link in the following table for the source's host type

Host Type of Source	Click for Source Settings Descriptions:
Local filesystem, UNC, MOVEit Transfer, FTP, FTP/S, SFTP	§ ***General settings*** (on page 54) § ***After Transfer settings*** (on page 58) § ***Host Override settings*** (on page 59)
FTP Servers	§ ***Additional Commands*** (on page 59) settings
POP3	§ ***General settings POP3*** (on page 66) § ***After Transfer settings POP 3*** (on page 67) § ***Host Override settings POP3*** (on page 68)
AS1	***Settings AS1*** (on page 69)
AS2	***Settings AS2*** (on page 69)
AS3	***Settings AS3*** (on page 70)

Add/Edit Source - General settings

These settings apply to the following source types in traditional and advanced tasks:

- Local folder
- UNC host/mapped drive
- MOVEit Transfer server
- FTP or FTP/S server
- SFTP server

To access these settings:

- **For an existing task:** Select **TASKS**. Click the task name. Click **Step > Source**. Or if the source is already defined, click its more options icon and select **Edit**.
- **For a new task:** Click TASKS and add a traditional or advance task. Click **Step > Source**.

Add Source - General tab	Description
Folder(s)	Folder name or path where files are located on the host. Browse button is not available if the source uses an FTP or SSH host with Blind Downloads enabled.
For all host types	***Macros*** (on page 176) are allowed in this field.
For local filesystem hosts	Do not use full UNC paths.

For Filesystem, MOVEit Transfer, FTP, and SSH hosts	**Note:** Do not use a wildcard (* or **) in the first position of the path.
	Example of *
	`/home/jal/data*/reports` matches any folder named **reports** that is a direct subfolder of a folder whose name starts with **data** and is a subfolder of `/home/jal`.
	§ The following folders match:
	`/home/jal/data/reports`
	`/home/jal/dataaccounts/reports.`
	§ The following folder does not match:
	`/home/jal/reports`
	MOVEit Automation specific wildcard operator ** ** matches any number of intermediate subfolders. A trailing ** means the same as checking **Include Subdirectories
	Example of **
	`/home/jal/**/reports` matches any folder named **reports** that is a direct or indirect subfolder of `jal`.
	§ The following folders match: `/home/jal/**/reports`:
	`/home/jal/data1/2005/07/reports`
	`/home/jal/2005/reports`
	`/home/jal/reports`
	§ The following folders *do not match* `/home/jal/**/reports`:
	`/home/jal/2005/report1`
	`/home/jal/2005/reports/mydata`
	Note: The folder `/home/jal/2005/reports/mydata` would match if **Search Subdirectories** was checked.

For Remote filesystems by UNC or mounted drive letter	First add the host as a Share. If you use a UNC, MOVEit Automation attempts to find a matching Share host and prompts you to use it.
Folders checkboxes	

§ Ignore folders	If checked, optionally specify a folder mask. Folder masks must reflect folder names, not full or relative folder paths. The folder masks match against the names of subfolders in the current folder being searched. To specify multiple folders, use a semicolon (`;`).separated list with no spaces. ***Macros*** (on page 176) are allowed in this field.
§ Search subdirectories	Searches subdirectories of the specified folder (as well as the folder itself) according to the specified file mask.
File(s)	The filename or filemask used to select files on the remote host. ***Macros*** (on page 176) are allowed in this field. Special wildcard characters: § Asterisk (*) matches zero or more characters in that position in the filename. § Question mark (?) matches one character at that position in the filename. You can use multiple wildcard characters in a single mask. Examples: § `x*.rpt` matches `x.rpt`, x1`.rpt`, and `xyz.rpt`. It does not match `xyz.rp` or `zz.rpt` § `a?rpt` matches `a1.rpt` and `aQ.rpt` It does not match `a.rpt` or `a1.rp` If you want the task to wait until all filetypes in the mask are available from the source, also select Retry if No Files Found. Note: When Collect only new files is selected, if you change the filemask on an existing source, doing so could cause old files to be downloaded from the source host. File collection timestamps are stored by filemask internally in MOVEit Automation. Changing the filemask negates existing timestamps.
Files checkboxes	

§	Ignore file(s)	Select and specify the files to ignore. ***Macros*** (on page 176) are allowed in this field. See File(s) description, above, for details. Note: ampersand (&) is always treated as a literal in this field.
§	Special filters	Specifies files based on size and date/time of last modification. Select, then click Edit to specify filter criteria. Choose to match any or all criteria.
§	Collect only new files	Selects only new files from the remote server. For filesystem, FTP, and SSH hosts, MOVEit Automation maintains a database of most recent timestamps by host, directory, and filemask. For information on how to edit this database, see Editing Source Timestamps. MOVEit Automation relies on the MOVEit Transfer definition of newness only until the first new file is downloaded for any given folder. After that, MOVEit Automation uses the same new file definition used by the other source types. For filesystem files, MOVEit Automation uses the Last Modified stamp to determine newness. Note: State information for this option is saved on a per-task basis, not on a per-host basis. If the task has run previously and sent files, the files are resent initially after this option is selected, if the files still exist on the source.
§	Expand zip file	Uncompresses any files that end in the extension `.zip`.
§	Retry if no files found	Selected: If the previous directory listing was successful but resulted in no matching files, the task times out and retries the directory listing. The retry count and retry timeout are used. You can set this feature to wait for a file to appear during the running of a task, If the maximum number of retries is reached and no matching files are found, the task continues with the next step, with no error flagged. Not selected: A directory listing that returns no files is not retried, and no files are downloaded from the source. Notes: § Defaults for Retry Count and Retry Timeout are set on a host's Limits settings. To access on an existing host, click HOSTS and click the host name. Expand the Limits category. § On a Source, overrides for Retry Count and Retry Timeout are set on the Create/Edit Source dialog box ***Host Override*** (on page 59) tab.

See also:

Add/Edit Source - After Transfer Settings (on page 58)
Add/Edit Source - Host Override Settings (on page 59)

After Transfer settings: UNC, MOVEit Transfer, FTP, SFTP

These settings apply to the following source types in traditional and advanced tasks. The indicate how to handle original source files after the task completes successfully.

- Local folder
- UNC host/mapped drive
- MOVEit Transfer server
- FTP or FTP/S server
- SFTP server

For settings of other source types, see ***POP3*** (on page 66) sources, ***AS1*** (on page 69) sources, ***AS2*** (on page 69) sources, ***AS3*** (on page 70) sources.

After Transfer settings

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source.

After Transfer tab

Source - After Transfer tab	Description
After successful transfer:	
§ Do nothing	Default
§ Delete original files	Deletes the original files from the source. Tip: This is an effective way to prevent double posts. Note: For the special case of deleting POP3 messages, see ***POP3 Sources*** (on page 255).
§ Rename original files	Renames the original files to the name you specify. The name can contain macros, such as [OrigName]. For more information, see ***Macros*** (on page 176). Some hosts, such as FTP hosts, permit you to move source files across folders using this option and properly formatted paths.
§ Renamed files may overwrite existing files	Available if Rename original files is selected.

See also:

- ***Add/Edit Source - General Settings*** (on page 54)
- ***Add/Edit Source - Host Override Settings*** (on page 61)

Source - FTP Additional Commands

This topic pertains to FTP and FTP/S sources. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

About commands on an FTP host

FTP hosts can be configured to execute commands in any of the following situations:

- Upon signon
- Per file, before files are transferred
- Per file, after files are transferred

For an FTP Source, you can configure *additional* commands to execute on a per-file basis, before and after each transfer. The commands that are configured at the Host level are also executed. For more information, see ***FTP Host - Additional Properties*** (on page 319).

To access Additional Commands settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an FTP source.

FTP source - Additional Commands tab	Description
Commands to execute (per file) before transfer	Quote commands to send to the server before each file is downloaded. ***Macros*** (on page 176) are allowed in this field.
Commands to execute (per file) after transfer	Quote commands to send to the server after each file is downloaded. ***Macros*** (on page 176) are allowed in this field.

Host Override Settings on a Source

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

Source Override fields that are available on the source depend on the type of host the source references.

For source override settings descriptions: In the following list, click the type of host used by the source:

- ***Local Folder*** (on page 60)
- ***Mapped Drive or UNC path*** (on page 60)
- ***MOVEit Transfer server*** (on page 61)
- ***FTP or FTP/S server*** (on page 62)
- ***SFTP server*** (on page 64)
- ***POP3 (email) server*** (on page 68)

ASx sources do not have host override settings.

Source - Local File System or UNC - Host Override Settings

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

Settings in this topic pertain to sources that use Local File System hosts or UNC hosts. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access the Host Override settings for Local File System or UNC sources:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add a Local File System or UNC source.

Local File System or UNC source - Host Override tab	Description
Timeouts tab	
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
Override Default Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Download Limits tab	
Override Default File Count Download Limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Override Default File Size Download Limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.

Source - DMZ Host Override Settings

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

Settings in this topic pertain to MOVEit Transfer source types. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add a MOVEit Transfer source.

MOVEit Transfer source - Host Override tab	Description
Authentication tab	
§ Override Default Credentials (MOVEit Transfer, FTP, SSH servers)	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Certificate (MOVEit Transfer and FTP servers)	This option is available for MOVEit Transfer and FTP sources only. Click Set Cert and select a certificate.
File Sorting tab	
§ Override Default File Sorting	Select to specify how to sort files.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
Download Limits tab	

§ Override Default File Count Download Limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
§ Override Default File Size Download Limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.

See also:

- ***General settings*** (on page 54) for Local filesystem, UNC, MOVEit Transfer, FTP, FTP/S, SFTP sources
- ***After Transfer settings*** (on page 58) for Local filesystem, UNC, MOVEit Transfer, FTP, FTP/S, SFTP sources
- Settings for ***POP3*** (on page 66) sources, ***AS1*** (on page 69) sources, ***AS2*** (on page 69) sources, ***AS3*** (on page 70) sources
- ***Supported Host Types*** (on page 23)

Source - FTP Host Override Settings

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

The settings in this topic pertain to sources that use FTP or FTP/S hosts. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an FTP source.

FTP source - Host Override tab	Description
Authentication tab	

FTP source - Host Override tab	Description
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host. Specify an account.
§ Override Default Client Certificate	Click **Set Cert** and select a certificate.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Data timeout	Number of seconds to wait when sending data to or receiving data from the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
§ Override Default Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Transfer tab	
§ Override Default Transfer Type	ASCII or Binary mode
§ Override Default Transfer Mode	§ **Active:** normal mode of operation for FTP transfers § **Passive:** typically used for FTP clients that are located behind a firewall.
§ Override Default XSHA1 Setting	The XSHA1 command is used if the FTP server supports it. The file that is received or transmitted is compared with the copy of the file that is on the server, by comparing SHA1 hashes of the file.
§ Override Default Blind Download	When blind downloads are enabled, MOVEit Automation does not use any directory listing commands. The blind download option is rarely used, and is intended primarily to accommodate unusual FTP servers.
§ Override Default Resume	For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
§ Override Default Reuse SSL Session	Forces data connections to use the same SSL session as the existing control connection. Use to override the setting for a given source or destination task element configuration, so that you comply with partner server settings that require reuse of an SSL session for data connections.
Download Limits tab	

FTP source - Host Override tab	Description
§ Override default file count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
§ Override default file size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Security tab	
§ Override default Use MD5	For more information, see ***FTP Host - Additional Properties: Use MD5*** (on page 319)
§ Use MD5	Make your selections.
§ Override Default MD5 Filename	Default is `MD5SUM`
§ MD5 Filename	Name of the MD5 file to look for. Default is `MD5SUM`.

Source - SFTP Host Override Settings

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

The settings in this topic pertain to SFTP source types. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an SFTP source.

SFTP Source - Host Override	Description
Authentication tab	

SFTP Source - Host Override	Description
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Key	Click **Set Key** and select a key.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
§ Override Default Transfer Rescan time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Transfer tab	
§ Override Default Blind Download	When blind downloads are enabled, MOVEit Automation does not use any directory listing commands. The blind download option is rarely used, and is intended primarily to accommodate unusual FTP servers.
§ Override Default Resume	For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
Download Limits tab	
§ Override default file count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
§ Override default file size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Security tab	

SFTP Source - Host Override	Description
§ Override default Use MD5	For more information, see ***FTP Host - Additional Properties: Use MD5*** (on page 319)
§ Use MD5	Make a selection.
§ Override Default MD5 Filename	Default is `MD5SUM`
§ MD5 Filename	Name of the MD5 file to look for. Default is `MD5SUM`.

Source - POP3 General Settings

These settings apply to the POP3 source type. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

About POP3 Sources

POP3 sources download email messages from a POP3 server. Each attachment in a message is considered a separate file. The body of a POP3 email message is not considered a file, so an email message that has no attachments is considered to have no files. For more information, see ***POP3 Sources*** (on page 255).

To access POP3 General Settings:

§ For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon and select Edit.
§ For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add a POP3 source.

POP3 Source - General	Description
Ignore file(s)	Select and specify the files to ignore. ***Macros*** (on page 176) are allowed in this field. Special wildcard characters: § Asterisk (*) matches zero or more characters in that position in the filename. § Question mark (?) matches one character at that position in the filename. You can use multiple wildcard characters in a single mask. Examples: § `x*.rpt` matches `x.rpt`, x1`.rpt`, and `xyz.rpt`. It does not match `xyz.rp` or `zz.rpt` § `a?rpt` matches `a1.rpt` and `aQ.rpt` It does not match `a.rpt` or `a1.rp` If you want the task to wait until all filetypes in the mask are available from the source, also select Retry if No Files Found.
Collect only new files	Selects only new files from the remote server.
Expand zip files	Uncompresses any files that end in the extension `.zip`.

POP3 Source - General	Description
Retry if no files found	Selected: If the previous directory listing was successful but resulted in no matching files, the task times out and retries the directory listing. The retry count and retry timeout are used. You can set this feature to wait for a file to appear during the running of a task, If the maximum number of retries is reached and no matching files are found, the task continues with the next step, with no error flagged. Not selected: A directory listing that returns no files is not retried, and no files are downloaded from the source.

See also

§ ***POP3 Source - After Transfer Settings*** (on page 67)
§ ***POP3 Source - Host Override Settings*** (on page 68)

Source - POP3 After Transfer Settings

These settings apply to POP3 sources. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access POP3 After Transfer settings:

§ For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
§ For a new task: Click TASKS and add a traditional or advance task. Click Step > Source.

POP3 source - After Transfer tab	Description
Do nothing	Default
Delete original files	Deletes the original file from the source. Note: POP3 servers consider each attachment to be a separate file. For more information, see ***POP3 Sources*** (on page 255).

See also

§ ***POP3 Source - General Settings*** (on page 66)
§ ***POP3 Source - Host Override Settings*** (on page 68)

Source - POP3 Host Override Settings

A source's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the source as it is defined in the context of the task.

These settings apply to POP3 sources. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon ⋯ and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source.

POP3 source - Host Override	Description
Authentication tab	
Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
Timeouts tab	
Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

See also

- ***POP3 Source - General Settings*** (on page 66)
- ***POP3 Source - After Transfer Settings*** (on page 67)

Source - AS1 Settings

These settings apply to the AS1 source type. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an AS1 source.

AS1 Source - Field	Description
Subject Match	Indicates which messages to download from the AS1 POP3 server. ***Macros*** (on page 176) and the wildcard characters asterisk (*) and question mark (?) are allowed.
Override Default Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Source - AS2 Settings

These settings apply to the AS2 source type. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

- For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon and select Edit.
- For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an AS2 source.

AS2 Source - Field	Description
File(s)	The files to download from the AS2 MOVEit Transfer server. You can use Macros in this field, as well as * and ? wildcard characters. Multiple masks can be entered, separated by semicolons (;).
Ignore file(s)	Select and enter filenames or filemasks. Multiple masks can be entered, separated by semicolons (;). ***Macros*** (on page 176) are allowed in this field.
Override Default Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3.

AS2 Source - Field	Description
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Source - AS3 Settings

These settings apply to the AS3 source type. For information about other source types, see the table in ***Add a Task Source*** (on page 53).

To access these settings:

§ For an existing task: Select TASKS. Click the task name. Click Step > Source. Or if the source is already defined, click its more options icon and select Edit.

§ For a new task: Click TASKS and add a traditional or advance task. Click Step > Source. Add an AS3 source.

AS3 Source - Field	Description
Folder(s)	Folder on the AS3 FTP server to search for EDI data message files. ***Macros*** (on page 176) are allowed in this field.
File(s)	Files to be downloaded from the AS3 FTP server. ***Macros*** (on page 176) are allowed in this field. Special wildcard characters: § Asterisk (*) matches zero or more characters in that position in the filename. § Question mark (?) matches one character at that position in the filename. You can use multiple wildcard characters in a single mask. Examples: § `x*.rpt` matches `x.rpt`, x1`.rpt`, and `xyz.rpt`. It does not match `xyz.rp` or `zz.rpt` § `a?rpt` matches `a1.rpt` and `aQ.rpt` It does not match `a.rpt` or `a1.rp` If you want the task to wait until all filetypes in the mask are available from the source, also select Retry if No Files Found.
Ignore file(s)	Select and specify the files to ignore. ***Macros*** (on page 176) are allowed in this field. See File(s) above for details.
Upload MDN to the same folder	Default is checked. MDNs are written to the same FTP folder in which the original EDI data message was located.
MDN Path	Available if Upload MDN to the same folder is not checked. ***Macros*** (on page 176) are allowed in this field.
MDN Filename	The name of the file to which MOVEit Automation writes the MDN if an MDN is requested by the sending partner. ***Macros*** (on page 176) are allowed in this field. ***Macros*** (on page 176) are allowed in this field.

AS3 Source - Field	Description
Override Default Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Add a Task Process

In a task, a process runs a single built-in script or custom script. The script must exist before you can add it to a process. For more information, see ***Scripts*** (on page 103). A task can have more than one process.

To access this dialog box:

1. Select TASKS. Create a new Traditional or Advanced task, or select an existing one. You can ***filter the list*** (on page 12).
2. On the task properties page, click Step > Process.

   If the task has existing steps:
   - In traditional tasks: The new process is added immediately after any existing processes.
   - In advanced tasks: At the location for the new process, click Insert here.
3. Make your selections in the Add Process dialog box.

Add Process field	Description
Script	Select a script.
Description	Optional.
Run script	§ Per File: Process runs one time for each file that is downloaded from all sources, before the file is sent to any destination hosts. If there are no sources, the process is run one time. This is the typical setting for most processes. § Once After All Downloads: Process runs only one time after all source files have been downloaded, and before any files are uploaded to any destination hosts. This setting is useful for tasks such as zipping multiple files into a single archive. The process runs even if no files have been downloaded. Such a process can determine whether any files were downloaded by checking to see whether the MICacheFiles() function returns an empty string.
Use process as a destination	Sets the process as a destination so that even If a previous run of the task has already sent files to all destinations, or if a task has no destinations, the task will run. For more information, see ***Use Process as a Destination*** (on page 89).

Edit task parameters	Click and provide parameter values. If the script has required parameters, the values you set here apply to the task you are configuring. If a different task uses the same script, you set parameter values for that task when adding a process to that task. You can also specify a global parameter.

Add a Task Destination

A destination defines a single location to which files are sent when a task runs. A task can have any number of destinations, or no destinations. If a task has no destinations, it must have at least one process to be eligible to run. For more information, see ***Process*** (on page 71).

Prerequisite: Before you add a destination to a task, you must have already added the host that the destination references. For example, to add a destination that is a folder on a MOVEit Transfer server, you must have previously added a MOVEit Transfer host for that server. For more information, see Add a Host.

To add a destination:

1. Select TASKS. Add a new traditional or advanced task, or click an existing one. The task properties page opens.
2. Click Step > Destination. If this is an advanced task, specify the location for the destination step.
3. In the Add Destination dialog box, select how you want to load the destination file, and select a host, Click Next.
4. Click the tabs and make selections to specify the behavior of the destination files.

For field descriptions, click the link in the following table for the type of destination.

Type of Destination	Click for Destination Settings Descriptions:
Local filesystem UNC MOVEit Transfer FTP FTP/S, SFTP	§ ***General settings*** (on page 73) § ***Host Override settings*** (on page 75)
Email attachment	***Send as Email Attachment Settings*** (on page 74)
AS1, AS2, AS3 servers	***AS1, AS2, AS3 Settings*** (on page 79)

Destination - General Settings

These settings pertain to the following Destination types:

- Local folder
- Mapped drive or UNC
- MOVEit Transfer server
- FTP or FTP/S server
- SFTP server

To access these settings:

- For an existing task: Select TASKS. Click the task name, Click Step > Destination. Or if the destination is already defined, click its gear icon and select Edit.
- For a new task: Click TASKS and add a traditional or advanced task. Click Step > Destination..

Destination - General tab	Description
Folder(s)	Folder name or path to save files on the remote host. ***Macros*** (on page 176) are allowed in this field. For local filesystem hosts: Do not use full UNC paths. To use a remote filesystem by UNC or mounted drive letter, first add the host as a Share. If you use a UNC, MOVEit Automation attempts to find a matching Share host and prompts you to use it.
Use relative subdirectories	If one or more source elements have the Include Subdirectories option selected, files found in subdirectories of the source folder are uploaded to the same subdirectory they were found in on the destination host.
Create directories if necessary	Selected: If the directory named in the path does not exist, MOVEit Automation creates the directory (or the relative subdirectory if Use relative subdirectories is checked). Not selected: If the server directory does not exist, the transfer fails.
Use original filenames	The name under which the file was saved on the source. If not checked, uses the name you define. The name can contain ***macros*** (on page 176).
If a file of the same name already exists:	§ Do not overwrite § Overwrite (default) § Append to
Compress each file (zip format)	Compresses each file individually. Uses the original filename appended with .zip. For example, `MyFile.txt` is compressed to a file named `MyFile.zip`. To zip multiple files together, use the built-in script ***Zip Advanced*** (on page 136).

See also:

- ***Destination - Send as Email Attachment*** (on page 74)
- ***Destination - AS1, AS2, and AS3 Settings*** (on page 79)

Destination - Send as Email Attachment

Each file is sent as an attachment in a separate email. See the Compress each file field in the table below.

Prerequisite: For the Send as Email Attachment option to be available, you must have an SMPT host configured. For more information, see ***SMTP Hosts*** (on page 333).

To access these settings:

§ For an existing task: Select TASKS. Click the task name, Click Step > Destination. Or if the destination is already defined, click its gear icon and select Edit.

§ For a new task: Click TASKS and add a traditional or advanced task. Click Step > Destination.. Select Send as an email attachment.

Email Attachment - General tab	Description
To Address	Required.
Subject	Email subject.
Body	Email message body.
Use original filenames	The name under which the file was saved on the source. If not checked, uses the name you define. The name can contain ***macros*** (on page 176).
Compress each file (zip format)	Compresses each file individually. Uses the original filename appended with .zip. For example, `MyFile.txt` is compressed to a file named `MyFile.zip`. To zip multiple files together, use the built-in script ***Zip Advanced*** (on page 136).

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

Email - Host Override tab	Description
Override Defatult Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Host Override Settings on a Destination

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

Host Override settings differ depending on the Destination's host type.For more information, see make a selection from the following list:

Host Override settings for:

- ***Local filesystem Host*** (on page 75)
- ***UNC path or mapped drive*** (on page 75)
- ***MOVEit Transfer server*** (on page 76)
- ***FTP or FTP/S server*** (on page 76)
- ***SFTP server*** (on page 78)
- ***Email attachment*** (on page 74)

Destinations that use AS1, AS2, and AS3 servers do not have Host Override settings.

Destination - Local System and UNC Settings

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

To access these settings:

- For an existing task: Select TASKS. Click the task name, Click Step > Destination. Or if the destination is already defined, click its gear icon  and select Edit.
- For a new task: Click TASKS and add a traditional or advanced task. Click Step > Destination..

Local File System and UNC Sources - Host Override tab	Description
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. Specify a retry count.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. Specify a retry timeout.
Transfer tab (UNC destinations only)	
Override Default Use Windows CopyFile API	
Use Windows Copyfile API for UNC transfers	Default is checked. Available if Override Default Windows CopyFile API is checked.

Destination - DMZ Host Override Settings

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

To access these settings:

- **For an existing task:** Select **TASKS**. Click the task name, Click **Step > Destination**. Or if the destination is already defined, click its gear icon and select **Edit**.
- **For a new task:** Click **TASKS** and add a traditional or advanced task. Click **Step > Destination**..

MOVEit Transfer destination - Host Override tab	Description
Authentication tab	
Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
Override Default Client Certificate	Click **Set Cert** and select a certificate
Timeouts tab	
Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Destination - FTP Host Override and Commands Settings

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

To access these settings:

- **For an existing task:** Select **TASKS**. Click the task name, Click **Step > Destination**. Or if the destination is already defined, click its gear icon and select **Edit**.
- **For a new task:** Click **TASKS** and add a traditional or advanced task. Click **Step > Destination**..

FTP destination - Host Override tab	Description
Authentication tab	

FTP destination - Host Override tab	Description
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Certificate	Click **Set Cert** and select a certificate
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
Transfer tab	
§ Override Default Transfer Type	ASCII or Binary mode
§ Override Defaultj Transfer Mode	§ **Active:** normal mode of operation for FTP transfers § **Passive:** typically used for FTP clients that are located behind a firewall.
§ Override Default XSHA1 Setting	If selected, the option **Use XSHA1 command, if available** becomes available.
§ Override Default Blind Upload	When blind uploads are enabled, MOVEit Automation does not use any directory listing commands.
§ Override Default Resume	If selected, the option **Resume partial transfers (if possible)** becomes available. For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
§ Override Default Reuse SSL Session	If selected, the option **Reuse SSL session for data connections** becomes available. Forces data connections to use the same SSL session as the existing control connection. Use to override the setting for a given source or destination task element configuration, so that you comply with partner server settings that require reuse of an SSL session for data connections.

FTP destination - Additional Commands tab	Description
Commands to execute (per file) before transfer	Quote commands to send to the server before each file is uploaded. ***Macros*** (on page 176) are allowed in this field.
Commands to execute (per file) after transfer	Quote commands to send to the server after each file is uploaded. ***Macros*** (on page 176) are allowed in this field.

Destination - SSH/SFTP Host Override Settings

A destination's Host Override settings override the corresponding setting that is configured in the host. Your selections affect the destination as it is defined in the context of the task.

To access these settings:

- For an existing task: Select TASKS. Click the task name, Click Step > Destination. Or if the destination is already defined, click its gear icon and select Edit.
- For a new task: Click TASKS and add a traditional or advanced task. Click Step > Destination.. Add an SFTP destination.

SSH/SFTP Destination - Host Override tab	Description
Authentication tab	
§ Override Default Credentials	Specify a username and password for MOVEit Automation to use to sign on to the host.
§ Override Default Client Key	Click Set Key and select a key.
Timeouts tab	
§ Override Default Connect Timeout	Number of seconds to wait when attempting to connect to the host.
§ Override Default Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure.
§ Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
§ Override Default Transfer Rescan time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Transfer tab	

SSH/SFTP Destination - Host Override tab	Description
§ Override Default Resume	If selected, the option **Resume partial transfers (if possible)** becomes available. For more information, see ***FTP Host - Additional Properties: Resume partial transfers (if possible)*** (on page 319)
File Attributes tab	
§ Override Default File Attributes	Applies the specified UNIX-style file attributes to a file after a successful upload. This setting applies only to hosts that are based on UNIX-like file systems.
§ Set file permisisons after upload	Available if Override Default File Attributes is selected. Choose permissions for User, Group, and Owner.

Destination - AS1, AS2, AS3 Settings

To access these settings:

§ **For an existing task:** Select **TASKS**. Click the task name, Click **Step > Destination**. Or if the destination is already defined, click its gear icon  and select **Edit**.

§ **For a new task:** Click **TASKS** and add a traditional or advanced task. Click **Step > Destination**.. Add an AS1 destination.

ASx Destination	Description
Use original filenames	The name under which the file was saved on the source. If not checked, uses the name you define. The name can contain ***macros*** (on page 176).
Send all source files in a single message	If not checked, sends individual ASx messages for each source file.
Request MDN	Requests an MDN from the destination partner to verify that the data arrived. The task is considered to be incomplete until the MDN arrives.
Request Signed MDN	Requests that the MDN sent by the destnation partner be signed by the partner's SSL certificate, to verify the origin of the MDN message.
Use partner certificate for signature validations	The Partner Certificate configured in the referenced AS1 host is used to validate the signature on the MDN that is received form the partner. If not checked, click **Set Cert** and select a different SSL certificate to use for signature validation

ASx Destination	Description
(AS2 only) Request Asynchronous MDN	Request that the MDN be sent after the file transfer connection has been closed. Can be sent to an HTTP server or by email.
(AS3 only) Look for MDN in same path as files	MOVEit Automation looks for MDNs from the destination partner in the same FTP folder to which the EDI data message was uploaded.
(AS3 only) MDN Filemask	Used to search for MDNs from the destination partner. ***Macros*** (on page 176) and the wildcards asterisk (*) and question mark (?) are allowed.
(AS1 only) MDN Email Address	Destination to which the destination partner sends MDNs. Default is the `[HostOrgEmail]` macro, which represents the My Organization - Email Address field value of the referenced AS1 host.
(AS2 only) Request Email MDN	Available if Request Asynchronous MDN is selected. The AS2 host must have an AS1 host configured to receive email MDNs. This can be configured in the Email MDN settings of the AS2 host. See ***AS2 Host - Additional Properties*** (on page 340).
(AS2 and AS3 only) MDN URL	URL to which HTTP MDNs are sent by the destination partner.
Override Default Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3.
Override Default Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.

Add a File Loop (FOR)

This topic pertains to Advanced tasks.

A file loop causes a set of steps to be performed one time for each file that has been downloaded or created so far in the task. An unlimited number of steps can appear inside a loop. Any task element, except a schedule, can be inside a loop.

Prerequisite: A file loop must be preceded by at least one source element, or by at least one process that adds to the list of files.

To add a file loop:

1. Click TASKS, click an Advanced task or create a new one.
2. If the task does not have a source, add a source.
   - If this is the first source to be added to the task, you are prompted to add a file loop. Click Yes. The file loop is added immediately after the source in the list of task elements.

- If a source already exists, click **Step > File Loop**. You are prompted for the location of the file loop. At the location, click **Insert here**.

A **For each file...** row is added to the list of task elements.

3. For each element to include in the file loop:
   a) Click **Step** and select the element. You are prompted for a location.
   b) Click an indented **Insert here** that is located under the **For each file...** row that designates the start of the file loop.
   c) Fill out any required fields for the element you are adding. For example, if you are adding a process, select the script and any required parameters.
4. To reposition any element in the list, click the gear icon and select **Move**. In the new position in the list, click **Move here**.

See also:

- ***Conditional Branch (IF block)*** (on page 81)

Add Conditional Branch (IF block)

This topic applies to Advanced tasks.

A conditional branch (IF block) defines a condition, and the set of actions to be performed if that condition is met or not met. You can configure a conditional branch anywhere in an Advanced task, including inside a file loop. Any task element except a schedule can appear inside a conditional branch.

A conditional branch takes an action one time if the condition is met. To take an action more than one time, for example, on each file, place the conditional branch inside a ***File Loop*** (on page 80).

Requirements: The conditional branch (IF block) requires the following elements:

- The item to be acted upon (for example file name matches *.txt). The condition can act on source files or on the values of task parameters
- The action to take if the file meets the condition. Actions include pulling files from a specified source, pushing files to a specified destination, or running the process.

To access this dialog box: Click **TASKS**. Create a new Advanced task, or select an existing one. Click **Step > Conditional (If)**. Specify a location for the conditional branch by clicking **Insert Here**.

Conditional branch (IF)	Description
Item to evaluate	The item to evaluate with the condition. Options: File name, file size, file timestamp, file error code, task error code, custom macro. For task parameter values, use a custom macro. For more information, see ***Macro Keywords*** (on page 178).
Condition Operator	Selections correspond to the type of item to evaluate. For example, for File Size, operators are = (equal to) != (not equal to), > (greater than), >= (greater than or equal to), < (less than), <= (less than or equal to).

Conditional branch (IF)	Description
Value	Value to match/not match. Selections correspond to the type of item to evaluate. For example, for File Timestamp, selections include a number, and type of unit (hours, days, months, or years ago).
Add condition	Click to add another condition. With multiple conditions, the options Match any / Match all become available.

To Add Default (Else) Branch:

1 Click Step > Conditional (if) and specify a location by clicking Insert Here.

2 In the row for the IF branch, click the gear icon and select Add Default (Else) Branch. The Else branch is added directly after the conditional branch, and has the label Otherwise.

Add a Send Email Step (advanced task)

Adds a Send Email step to an Advanced task. The email is sent regardless of the task results. The step can be placed anywhere in the task.

To add a send email to a Traditional or Sync task, see ***Add Action - Send Email*** (on page 83).

To access this feature: TASKS > *Click an advanced task* > Step > Send Email. Indicate the location of the Send Email step.

Step - Send Email	Description
Select a host	SMPT Host to use for sending email. The host must be already configured in MOVEit Automation in order to appear in the list.
To address	Email address or comma separated list of email addresses. ***Macros*** (on page 176) are allowed in this field.
Subject	Email subject.
Body	Email body.

Add a Run Task Step

Runs another task as a step in an Advanced task.

To add a Run Task to a traditional or sync task, see ***Next Action - Run Task*** (on page 84).

To access this dialog box: TASKS > *Click an advanced task* > Step > Run Task.

Step - Run Task	Description
Task to Run	The task must already exist in MOVEit Automation to appear in the drop-down list.
Wait until this task is complete before continuing	Stops the Advanced Task processing until the Task to Run has completed.
Parameters	Parameters that can be accessed by scripts and macros. Custom parameter names can contain upper-and lowercase alpha characters, period (.), underscore (_), and colon (:).

Next Action - Send Email

This topic applies to Traditional and Synchronization tasks. For Advanced tasks, see ***Add a Send Email step*** (on page 82).

To access this dialog box: TASKS > *Click traditional or sync task* > Next Action > Send Email

Add Action - Send Email	Description
Execute on	Any combination of results of the task § Success - The task found files to operate on and successfully operated on them. § Failure - The task found files to operate on but failed to operate on them. § No Action - The task did not find any files to operate on, or the files it found were ignored by a process.
Execute after	§ Tasks - after the entire task completes. § Each file - after each file is processed. (This option is not available for Sync tasks.)
SMPT Host	SMPT Host to use for sending email. The host must be already configured in MOVEit Automation in order to appear in the list.
To Address	Email address, or comma-separated list of email addresses. ***Macros*** (on page 176) are allowed in this field.
Subject	Email subject. ***Macros*** (on page 176) are allowed in this field.
Body	Email body. ***Macros*** (on page 176) are allowed in this field.

Next Action - Run Task

Adds a Run Task action to a Traditional or Synchronization task. For advanced tasks, see ***Add a Run Task Step*** (on page 83).

After the task you are configuring completes, runs another task that you specify.

If the task you specify is already running when MOVEit Automation tries to start it via Next Action, MOVEit Automation waits until the running instance of the task is completed before it starts the task.

To access this dialog box: TASK > *Click a traditional or sync task* > Next Action > Run Task .

Add Action - Run Task	Description
Execute On	Any combination of results of the task § Success - The task found files to operate on and successfully operated on them. § Failure - The task found files to operate on but failed to operate on them. § No Action - The task did not find any files to operate on, or the files it found were ignored by a process.
Execute After	§ Tasks - after the entire task completes. § Each file - after each file is processed. (This option is not available for Sync tasks.)
Task to Run	Task to run after the current task (that is, the task in which you are adding this Next Action) runs.
Parameters	Parameters for Task to Run.

Using Error Macros in Next Action - Run Task

Use the built-in `[Error...` macros to send the error code and description of any error that occurred in the subject or body of a Next Action notification.

For example, a Next Action message body of:

```
At [yyyy]-[mm]-[dd] [hh]:[tt]:[ss], task '[TaskName]'
encountered error #[ErrorCodeFile] - [ErrorDescriptionFile] - while transporting
'[OrigName]' (FYI, the current
task error is #[ErrorCodeTask] - [ErrorDescriptionTask])
```

...will be interpreted as:

```
At 2005-01-07 12:37:26, task 'Test Error Macros'
encountered error #2234 - CopyFile returned
Access is denied. - while transporting 'readme.txt'
(FYI, the current task error is #2234 - CopyFile
returned Access is denied.)
```

If you settle on a preferred combination of messages using error macros, you can propagate their use by setting them up as ***Global Parameters*** (on page 172) and configuring your tasks' Next Actions to work off global parameters rather than a different message for each task.

Next Action Tips

Next Actions can generate failures. For example, if a task successfully transfers 5 files but cannot send the email to notify the operator, task is marked as failed. To prevent this problem for Send Email next actions, point your mail host to a reliable external server or to the localhost SMTP server.

See also ***Macros*** (on page 176).

Add/Edit Task Schedule

A task can have more than one schedule. To run automatically, a task must have a minimum of one schedule. Tasks with or without schedules can be run manually by the operator.

In a task with more than one schedule, each schedule runs independently of the other schedules.

To access this dialog box: Click TASKS and click a task name. and

- To create a new scedule: Click Schedule.
- To edit an existing schedule: In the schedule row, click the gear icon and select Edit.

Add/Edit Schedule Field Descriptions

Add/Edit Schedule Field	Description
Days tab	

Week Month	**Week** and **Month** selections - The schedule runs on all of the days you select. Example: In Week, Monday and Thursday are selected; and in Month, 10 and 21 are selected. The schedule runs on every Monday and Thursday every week, *and* on the 10th and 21st of every month.
Custom	Adds a date list to the schedule. Date lists must have been previously created and added to MOVEit Automation to be included in the drop-down list. For more information, see ***Date Lists*** (on page 157). Select the date list and specify how to reference it: § **Include**: Run the tasks on the dates specified by the date list § **Exclude**: Do not run the task on any date that is specified in the date list. This behavior overrules (takes precedence over) any other days or dates in the schedule. **Note**: Do not use a date list and an overnight interval (for example, start time 10 pm and end time 3 am) in the same schedule. Instead, use two schedules in the task, one for the date list and the other for the overnight interval. Each schedule runs independently of the other schedule.
Times tab	
Start time	Scheduled start time. If **Repeat** is not selected, the task runs at the time you specify, on the days of the week and month that you selected on the Days tab. If no days are specified on the Days tab, the schedule runs once at the first occurence of the start time you specify. If the source host has file notifications enabled, polling occurs during the times specified by the schedule, at intervals defined by the polling interval. If files are available, the task runs.
Interval End time	Available if **Repeat** is selected. Specify the repeat interval and an end time. If the end time falls at the repeat interval, the task runs at the end time. For example, with a 2 pm start time, 3 pm end time, and repeat interval of 30 minutes, the task runs at 2 pm, 2:30 pm, and 3 pm. **Note**: Do not use a date list and an overnight interval (for example, start time 10 pm and end time 3 am) in the same schedule. Instead, use two schedules in the task, one for the date list and the other for the overnight interval. Each schedule runs independently of the other schedule.
Another time	Specify a start time to run the task, and an optional repeat interval and end time. Details appear on a separate row. The schedule runs on all the times and intervals that are defined on all the rows.

Options tab	
Repeat only until first success	Available if Repeat is selected. This option is ignored if all sources have file notifications enabled. Runs the task at the intervals specified, as long as no previous run in the schedule has successfully downloaded and processed files. After a task successfully retrieves and processes one file, all further runs in the schedule are suppressed until the next day. If you know that only one batch of files will appear within a given time range, this feature can be useful in reducing the load on remote servers and on MOVEit Automation.
Log failure if no files found during scheduled run	During the last run of a task during the schedule, the run fails if no runs during that schedule have succeeded in downloading and processing a minnimum of one file. Example: You expect a customer to place a file on a server every Friday between 1 am and 5 am. You schedule a task to run every 10 minutes during that time. If no file has appeared by the last run in that period, the run is logged as failed. Note: If you select this option, it is recommended that you designate a ***Next Action/Failure*** (on page 84) setting (for traditional tasks) or ***Conditional*** (on page 81) step (for advanced tasks).
Run even if notifications are enabled for the host	Available if the host associated with the source has File Notifications enabled. The task is run according to this schedule even if all associated source hosts have file notifications enabled.

See also:

- ***How Schedules Work*** (on page 88)
- ***Enabling File Notifications*** (on page 25)
- ***How File Notifications Work*** (on page 26)

How Schedules Work

MOVEit Automation checks for new tasks to run once a minute. Each time a check occurs, the scheduler scans all tasks to see which ones are eligible to run this minute. A task is eligible to run if ANY of its schedules lists the current minute as a valid time to run. The same task is not run multiple times in a minute, even if more than one of its schedules matches the current minute.

If the ***Maximum Running Tasks*** (on page 347) is set to 0, all eligible tasks are started simultaneously. Otherwise, the scheduler starts as many eligible tasks as it can without exceeding that setting. The other tasks are queued, and are run after currently-running tasks complete. Each task runs in its own thread..

A task is not run if a previous copy of the task is still running. Missed runs are run the next time that the scheduler runs and sees that another copy of the task is not running. However, no more than one missed run will be made up. For example, if a task is scheduled to be run at 9:00, 9:05, 9:10, 9:15, and 9:20, and the 9:00 task takes 16.5 minutes to run, then the 9:05 and 9:10 runs of the task will be skipped. At 9:17, the scheduler will run the 9:15 task two minutes late.

The scheduler does not run tasks that were scheduled to run prior to the time that MOVEit Automation was started. In the example above, if MOVEit Automation is started at 9:17, it does not run the task until 9:20, skipping the 9:00, 9:05, 9:10, and 9:15 runs.

Schedules and Event-Driven Tasks

Schedules also control when event-driven tasks are run. Events that arrive for an task are ignored unless they arrive during the scheduled time window. Missed files are handled by an automatic task run performed automatically when the task enters its next scheduled window.

For example,

- A task listens for events every day from 3am to 7am.
- A file arrives at 2am. This file is ignored and the task is not started at that time.
- At 3am MOVEit Automation automatically runs the task to look for missed files, finds the 2am file and downloads it.
- A second file arrives at 4am. As soon as this file is complete, MOVEit Automation runs the task to download it.
- A third file arrives at 8am. This file is ignored. Unless the task is started manually, this file will be picked up by the task the following day at 3am.

To schedule tasks on a periodic and event-driven basis, use the option **Run even if notifications are enabled for the host**.

To schedule a task to ignore file events, set it up to run once at specific times.

Schedules and Failover

At startup, and when a secondary failover node becomes primary, MOVEit Automation runs all event-driven tasks whose schedule covers the current date and time.

This Script Behaves As A Destination

Note: In MOVEit Automation releases prior to 9.0, this feature was known as Script Behaves as a Destination.

MOVEit Automation does not download a file if a previous run has already sent it to all destinations. As a result, if there are no destinations, the file is considered as having been sent to all destinations. If a task has no destinations, but has processes, and one of the processes fails, downloaded files are not downloaded again the next time the task runs.

Use this feature to force a process step to behave as a destination when you have a ***transfer exception*** (on page 98).

To access this feature: ***Add a process*** (on page 71) to a traditional or advanced task. On the Add Process dialog box, select Use process as a destination.

Add/Edit a Date List

A date list is a list of dates that is created separately from tasks. You can create a date list in Web Admin, and you can import a text file that contains a date list.

To use the date list, you select it in a schedule inside a task. For more information, see ***Schedules*** (on page 85).

To access this dialog box: Click SETTINGS > Date Lists.

- Create new date list: Click Add Date List.
- Edit a date list: Click a date list name. In the Properties row, click Edit.

Date List Field	Description
Friendly Name	Name that appears in the list of date lists
Description	Optional.
Date Entries	Make selections in Year, Month, Day to specify a single date. Click Add Entry.
Optional comment	In the date list, the comment precedes the date you specify.
Import Entries	Click and specify whether to Append or Overwrite entries in the date list. Browse for a text document that contains a date list. Syntax: § Each entry is a single line that uses the format YYYY-MM-DD § Asterisk (*) wildcard is allowed in the year and month places. § Hash (#) character designates a comment.

See also:

- ***Add/Edit a Task Schedule*** (on page 85)

Move a Task Element

Change the location of an element in a task

1. Click TASKS and click a task name. The task page opens, showing the list of task elements.
2. Locate the element that you want to move. At the right side of the row, click the gear icon and select Move.
3. At the position where you want to put the element, click Move here.

Update Original (rename or delete)

This topic applies to Advanced tasks.

Deletes or renames original source files.

Recommended: Before adding this step to a task, check to confirm that the transfer was successful by adding a ***Conditional branch*** (on page 81) and checking for a file error. If there are no errors, add the Update Original step.

To access this dialog box: Click TASKS > *Advanced task name* > Step > Update Original. To specify a location, click Insert here.

Field	Description
Update original file	What to do with the original source files. Delete or Rename.
Rename to	Available if Rename is selected. Indicates how to rename the source files. Specify a filename or file mask ***Macros*** (on page 176) are allowed in this field.
Renamed files may overwrite original files	Checked: Overwrites any source file whose name matches the name you specify in Rename to. Not checked: Maintains any source file whose name matches the name specified in Rename to.

See also

§ ***Add a Conditional Branch (IF block)*** (on page 81)

Multiple Sources, Destinations, Processes, and Schedules

Tasks can have more than one of the following elements:

If a task has more than one...	What happens when the task runs:
source host	Files are retrieved from all sources before any files are processed or sent to their final destinations.
process	The process is run against all fies that have been retrieved
destination host	All files that have been retrieved are sent to all hosts
schedule	The task runs at all of the times specified by all of the schedules

Task Settings and Parameters

To access task information:

1. Select TASKS and click a task name.
2. Click Actions > Task Settings. The <taskname> page lists settings in categories.
3. In the category row, click Edit.

For more information, see :

- ***Task Settings - General*** (on page 91)
- ***Task Settings - Advanced*** (on page 92)
- ***Task Setttings - Parameters*** (on page 94)
- ***Task Settings - Sync*** (on page 93)

Edit Task Info - General

To access this dialog box: TASKS > *Click-a-task-name* > Actions > Task Settings > (General row) Edit.

Task Info Field - General	Description
Friendly Name	Name that appears in the list of tasks, and at the top of the task properties page.
ID	System-generated identifier. Read-only. This field appears on the task settings page, when you expand the General settings row. It does not appear on the Edit Task Info dialog box.

Task Info Field - General	Description
Enabled	Yes - task can be started by the scheduler or started by notification events No - task cannot be run by the scheduler or by file notification events. This field appears on the task settings page, when you expand the General settings row. It does not appear on the Edit Task Info dialog box. For more information, see ***Task Schedules*** (on page 85).
Description	Optional
Operator Notes	Optional

See also:

- ***Task Settings - Advanced*** (on page 92)
- ***Task Setttings - Parameters*** (on page 94)
- ***Task Settings - Sync*** (on page 93)

Task Settings - Advanced Info

To access this dialog box: TASKS > *Click a task name* > Actions > Task Settings > (Advanced row) Edit.

Task Info Field - Advanced	Description
Cache files	Determines names of files when they are stored in the MOVEit Automation cache directory for processing. Options: § Use random names § Use original names - If one or more of the task sources use Include Subdirectories, files are stored with relative folder path names. Tip: This selection is useful when creating zip files of an arbitrary collection of files.
Use default state cache settings	Task uses the system's ***default State Cache settings*** (on page 348).
Keep this task's state information cached	Available if Use default state cache settings is not selected Select length of time.
Automatically re-run task if source download limits are encountered	The task is re-run, which allows MOVEit Automation to pick up the remaining source files without waiting for the next scheduled run of the task. If this option is enabled, and ***Collect Only New Files*** (on page 54) and ***Delete/Rename After Successful Download*** (on page 58) are not enabled, the task could loop indefinitely. Source download limits are set on the ***Host Override settings on the source*** (on page 59). Note: Source download limits can also be set at the host level (and not overridden at the source level). For more information, see File count download limit and File size download limit in the Additional Properties of ***specific host types*** (on page 23).

See also:

- ***Task Settings - General*** (on page 91)
- ***Task Setttings - Parameters*** (on page 94)
- ***Task Settings - Sync*** (on page 93)

Task Settings - Sync Info

These settings apply only to Synchronization tasks.

To access this dialog box: TASKS > *Click a sync task name* > Actions > Task Settings > (Sync row) Edit.

Sync Task - Task Info	Description
Sync Type	**Unidirectional** - replicates content from Folder A to Folder B **Bidirectional** - replicates content from Folder A to Folder B and from Folder B to Folder A
Sync Delete options	
§ Ignore Deletes	**For all tasks:** Replicates new files, updated files, and renamed files and folders. Additionally: § **for one-way task:** Ignores any files or folders that are deleted from Folder A § **for two-way task:** Ignores any files or folders deleted from either Folder A or Folder B.
§ Sync Deletes (default)	**For all tasks:** Replicates new files, updated files, and renamed files and folders. Additionally: § **for one-way tasks:** Deletes from Folder B any files or folders that were deleted from Folder A § **for two-way tasks:** Deletes from the other folder any files or folders that were deleted from either Folder A or Folder B This option is the default for new two-way sync tasks.
§ Sync deletes and delete extra files on Folder B	Available only to one-way tasks. This option is the default for new one-way sync tasks. Deletes from Folder B any files or folders that were deleted from Folder A. Deletes any extra files or folders that were created or renamed on Folder B by anything other than this sync task.
Case sensitive comparisons	Considers case in names when identifying files and folders.

See also:

- ***Task Settings - General*** (on page 91)
- ***Task Setttings - Parameters*** (on page 94)
- ***Task Settings - Advanced*** (on page 92)

Task Settings - Parameters Info

To access this dialog box:

- For existing tasks: TASKS > *Click a task-name* > Actions > Task Settings > (Parameters row) Edit
- When adding a process to a task: TASKS > Click a task name > Step > Process > Edit task parameters.

In the Edit Task Parameters dialog box, required parameters appear in red text. Click the plus sign (+) and specify a value.

See also:

- ***Add a Task Parameter*** (on page 94)
- ***Add a Task Process*** (on page 71)
- ***Global Parameters*** (on page 172)

Add Task Parameter

To access this dialog box: TASKS > *Click a task name* > Actions > Task Settings > Parameters > Edit > Add Parameter.

Specify a parameter name and value.

See also:

- ***Add a Task Process*** (on page 71)
- ***Global Parameters*** (on page 172)

Task Actions

The Actions button appears at the top right of an individual task page, after you click a task name from the list.

Note: The Bulk Actions button appears on the TASKS page, at the top right of the list of tasks. For more information, see Bulk Actions.

Actions Available on an Individual Task Page

When you click a task name to open the task's properties, the Actions drop-down list is available on the right side of the page. In the following table, an asterisk (*) indicates that the action is available for that task type.

Actions	Description	Trad	Adv	Sync
Run Now	Task is run immediately.	*	*	*
Clone	Task is cloned immediately, and named *Original-Task-Name* Clone. If you clone the original task again, it is named *Original-Task-Name* Clone 2.	*	*	*
Delete	You must confirm the deletion. For more information, see ***Delete Task***.	*	*	*
Export	Task is exported as an XML file to your default download location.	*	*	*
Edit Transfer Exceptions	Controls what a task does when it runs after a previous failure. See ***Edit Task Transfer Exceptions*** (on page 98).	*	*	*
Create Advanced Task from a Traditional Task	Immediately creates a task named *Original-Task-Name* Advanced. The new Advanced task is disabled. See ***Create Advanced Task from a Traditional Task*** (on page 99).	*		
Task Settings	View and edit Task Settings. For more information, see ***Task Settings and Parameters*** (on page 91).	*	*	*
Swap Sync Folders	Folder A and Folder B are swapped; that is, Folder A's definition becomes Folder B's definition, and Folder B's definition becomes Folder A's definition. The saved directory listings that define which files and folders are ignored, moved, created, etc. are also swapped. See ***Configuring a Synchronization Task*** (on page 35).			*

Actions	Description	Trad	Adv	Sync
Clear Sync Listings	Causes the sync task to be treated as a new sync task. Might cause MOVEit Automation to re-transfer files and folders that have already been moved.			*
View Sync Preview	Opens the ***Sync Preview dialog box*** (on page 45).			*

Run Task

You can prompt a scheduled or unscheduled task to run at any time. The task configuration must be complete to run a task.

To run a task, complete the following steps.

1. Open the TASKS tab, and select the task that you want to run from the Tasks List. The task properties page displays.
2. Click Actions > Run Now. The Started task <*task name*> message displays.
3. To confirm that the task was run, open the TASKS tab and sort the Task List by Last Run (start time) to locate the task.

The selected task runs without modifying the the task schedule.

Clone Task

Rather than creating a task set from scratch, you can clone an existing task and make any required edits.

To clone a task, complete the following steps.

1. Open the TASKS tab, and select the task that you want to clone from the Tasks List. The task properties page displays.
2. Click Actions > Clone. The Clone Task Info dialog box displays. The default clone name is <*original_task_name*> *Clone*. You can edit the name and description.
3. To clone the task, click Add Task. The task properties page of the new task displays and the Added task <*task name*> message displays.
4. You can edit the new task as required. For more information, see ***Task Elements***.

Delete Task

If you no longer require a task, you can delete it from MOVEit Automation.

To delete a task, complete the following steps.

1. Open the TASKS tab, and select the task that you want to delete from the Tasks List. The task properties page is displayed.
2. Click Actions > Delete. The Confirm delete task dialog box displays.
3. To delete the task, click Yes. The Deleted task <*task name*> message displays.

The task is deleted, and removed from the task list.

Export Task

You can export a task so that the task is stored in an XML file that can be imported if required. The task's properties, actions, triggers, and settings are in an XML file.

To export a task that does not contain keys or certificates, complete the following steps.

1. Open the TASKS tab, and select the task that you want to export from the Tasks List. The task properties page is displayed.
2. Click Actions > Export.

The task is exported. The default name of the exported task is <*original_task_name*> *Export*.

To export a task that contains keys or certificates, complete the following steps.

1. Complete steps 1 and 2, above. The Export certificate and keys dialog box displays.
2. The Export Certificates and Keys check box is automatically selected. To export the task without the keys or certificates, clear the check box and click OK.
3. To export the task with the keys and certificates, input and confirm a password, to password protect the keys and certificates. Click OK.

The task is exported. The default name of the exported task is <*original_task_name*> *Export*.

Edit Task Transfer Exceptions

What is a transfer exception?

Task transfer exceptions let you control what a Traditional or Advanced task does when it runs after a previous failure.

When a task fails after transferring some, but not all, of its files, an entry is created in the task's ***state file*** (on page 348), listing the files that have been completely or partially processed. These files are the *task transfer exceptions*. The next time that the task runs, MOVEit Automation does not perform duplicate processing on the files that were processed successfully during the previous processing. This prevents duplicate posting of files. When a task succeeds, any task failure history is removed from the state file.

You can remove entries that were processed successfully by a previous run of a task, so that the next time the task runs, the files that you have removed from the exceptions list will be processed again.

To edit transfer exceptions for a Traditional or Advanced task:

1. Click TASKS and select the task. (You can ***filter the list*** (on page 12).) The task properties page opens.
2. Click Actions > Edit Transfer Exceptions.

   The Edit Transfer Exceptions dialog box opens. If a previous run of the task was unsuccessful, the files that were successfully transferred appear in the list. These files *are not* transferred during subsequent task runs.
3. Select files to remove from the list, and click Remove selected.

   Files that you remove are transferred during the next run of the task.

   Files that remain in the list are not transferred again during the next run of the task.

Example 1

A task has two destinations. During a task run, it downloads files A and B and sends them to Destination 1, but cannot send them to Destination 2. MOVEit Automation marks both files as having been sent to Destination 1 but not Destination 2. The next time that the task runs, it downloads the files again, but sends them only to Destination 2.

Example 2

A task has one destination that is an unreliable FTP server. On a task run, it downloads files A and B and sends A successfully to the FTP server, but the FTP server crashes while B is being sent. MOVEit Automation marks file A completely processed. Then the next time the task runs, file A is not downloaded. File B is downloaded and sent to its destination.

Create Advanced Task from a Traditional Task

A *Traditional* task tracks whether a process succeeds or fails. If the process fails, the task does not complete the transfer. In an *Advanced* task, you use a conditional branch to track the success or failure of a process. The conditional branch is typically located in a file loop.

When you convert a traditional task to an advanced task, the following occurs:

- A new advanced task is created with the name <original-task-name> Advanced.
- The new task is disabled.
- MOVEit Automation adds File Loops and Conditional Branches where appropriate.
- If the traditional task included a Next Action Email, it is converted to a Send Email step. A Next Action Run Task is converted to a Run Task step.

To convert a traditional task to an advanced task:

1. Click TASKS and click a traditional task.
2. At the upper right side of the page, click Actions > Create Advanced Task from Task.

   The new task page opens. The process and destination are now in a File Loop. If a Next Action Email was part of the traditional task, it is converted to an Email element inside the conditional branch.
3. To add elements, click Step and make a selection.
4. *Recommended:* Using test data, run the task. Check the logs to see if the task is working as expected.

Editing Source Timestamps

Editing source timestamps allows you to determine which files MOVEit Automation considers new.

For each combination of task, source host, pathname, and filemask, MOVEit Automation stores a date/time stamp of the most recent file in that directory that matches that mask and that has been successfully processed, to determine which files are new. To re-run the file transfer as new, edit the file source timestamp.

For example: A task has a source that scans FTP host "XYZ FTP Server", looking for files in the "/uploads" path, matching the mask "*.txt", and the "Collect Only New Files" option is selected. The task was run successfully, and transferred the new file report.txt with a timestamp of 12 March 2014 at 2:15:00 pm. MOVEit Automation created an entry for the host "XYZ FTP Server", path "/uploads", mask "*.txt", with a stamp of 2014-03-12 14:15:00. To re-run the file transfer as new, edit the timestamps for this source to a value before the timestamp of the file, for example 2014-03-12 13:15:00.

You can use timestamps to define the behavior of a task before it is run. For example, to define when files are pulled from a host for a specific task, you can add a timestamp value for the proper path and mask.

Ø *To edit a source timestamp*

1. Select the TASKS tab. Click the task name. Click the source's more options icon ⋯ and select Edit.
2. In the Edit Source dialog box, select the Collect only new files check box, and click Save.
3. Click the source's more options icon ⋯ and select Edit source timestamps.
4. In the Task Source Timestamps dialog box, click the Edit timestamp entry icon ✎.
5. Edit the timestamp, and click Save.

Bulk Actions

Bulk actions occur on *all tasks that are listed on the TASKS page* (including multiple pages). Before you perform a bulk action, ***use the filters*** (on page 12) to limit the list of tasks to those on which you want to perform the bulk action.

For example, if the list of tasks includes 45 tasks and spans 6 pages, clicking **Bulk Actions > Create task group** creates a task group that contains all 45 tasks.

Bulk Action	Description
Create task group	Opens the ***Add new task group*** (on page 156) dialog box. Members are all tasks in the list on the TASKS page. For more information, see ***Create Task Group*** (on page 101).
Add to task group	Select the task group and click **OK**. Opens the ***Edit task group*** (on page 156) dialog box, All tasks currently listed on the TASKS page are added to the group. For more information, see ***Add to Task Group*** (on page 101).
Enable	Enables all eligible tasks currently listed on the TASKS page. Tasks must have a schedule in order to be enabled. For more information, see ***Enable Tasks*** (on page 102).
Disable	Disables all tasks currently listed on the TASKS list. Disabled tasks cannot be started by a schedule. For more information, see ***Disable Tasks*** (on page 102).
Export	Immediately exports all tasks in a XML file. You *are not asked* to confirm the export. The files are placed in your default download location. For more information, see ***Export Tasks*** (on page 103).
Delete	Deletes all listed tasks. You must confirm the deletion. For more information, see ***Delete Tasks*** (on page 103).
View Run History SP 1	Opens the Task Run tab, which displays the run details for all of the tasks listed in the current Tasks List. For more information, see ***Task Run Reports*** (on page 149).
View File Activity SP 1	Opens the File Activity tab, which displays the file activity details for all of the tasks listed in the current Tasks List. For more information, see ***File Activity Reports*** (on page 151).
View Audit Records SP 1	Opens the Audit tab, which displays the audit report details for all of the tasks listed in the current Tasks List. For more information, see ***Audit Reports*** (on page 152).

Create a Task Group

You can create a task group to group multiple tasks together. For more information, see ***Task Groups*** (on page 155).

To create a task group, complete the following steps.

1. Open the TASKS tab, and use the filters to list the tasks that you want to group.
2. Click Bulk Actions > Create Task Group. The Add new task group dialog displays.
3. Input the the following task group details:

Task Group Field	Description
Friendly Name	Name that appears in the list of task groups
Description	Optional
Members - Tasks tab	If you created the task group from the SETTINGS > Task Group page, initially the group has no members. Click Add Tasks, select tasks, and click OK. If you created the task group using Bulk Actions, the group contains all the tasks in the list at that time. To remove a task from the group, click the red X.
Members - Tabs for member types: § Hosts § Scripts § SSH Keys § SSL Certs § PGP Keys	In each tab: *Assigned members* - You can add members of that type to the task group. Click the tab and click Add <member type>. *Referenced members*: In each tab, items of that type that are associated with the tasks in the group are listed in dimmed font. For example, if a task has a source or destination that uses a specific host, the host is a referenced member. Referenced members provide read-only access to task elements. To promote a referenced member to an assigned member, click .

4. Click Add Task Group. The Added Task Group <group name> message displays.

The task group is created.

Edit a Task Group

You can add tasks to an existing task group to group specific tasks together. For more information, see ***Task Groups*** (on page 155).

To add tasks to an existing task group, complete the following steps.

1. Open the TASKS tab, and use the filters to list the tasks that you want to add to an exiting task group.
2. Click Bulk Actions > Add to Task Group. The Browse Task Groups dialog displays.
3. Select the Task Group to which you want to add the tasks. Click OK.
4. The Edit task group dialog displays.
5. Input or edit the the following task group details, if required:

Task Group Field	Description
Friendly Name	Name that appears in the list of task groups
Description	Optional
Members - Tasks tab	If you created the task group from the SETTINGS > Task Group page, initially the group has no members. Click Add Tasks, select tasks, and click OK. If you created the task group using Bulk Actions, the group contains all the tasks in the list at that time. To remove a task from the group, click the red X.
Members - Tabs for member types: § Hosts § Scripts § SSH Keys § SSL Certs § PGP Keys	In each tab: *Assigned members* - You can add members of that type to the task group. Click the tab and click Add <member type>. *Referenced members*: In each tab, items of that type that are associated with the tasks in the group are listed in dimmed font. For example, if a task has a source or destination that uses a specific host, the host is a referenced member. Referenced members provide read-only access to task elements. To promote a referenced member to an assigned member, click .

4. Click Save. The Updated Task Group <group name> message displays.

The selected tasks are added to the task group.

Enable Tasks

You can enable multiple tasks so that they can be run on demand or when they are scheduled to run.

Note: Tasks must have a schedule in order to be enabled.

To enable multiple tasks, complete the following steps.

1 Open the TASKS tab, and use the filters to list the tasks that you want to enable.
2 Click Bulk Actions > Enable. The Updated selected tasks message displays.

The tasks are enabled.

Disable Tasks

You can disable multiple tasks to prevent the task running. You can re-enable the tasks at a later date, if required. For more information, see ***Enable Tasks*** (on page 102).

Note: Disabled tasks cannot be started by a schedule.

To disable multiple tasks, complete the following steps.

1 Open the TASKS tab, and use the filters to list the tasks that you want to disable.
2 Click Bulk Actions > Disable. The Updated selected tasks message displays.

The tasks are disabled.

Export Tasks

You can export tasks so that the tasks are stored in an XML file that can be imported if required. The task's properties, actions, triggers, and settings are in an XML file.

To export multiple tasks, complete the following steps.

1. Open the TASKS tab, and use the filters to list the tasks that you want to export.
2. Click Bulk Actions > Export.

The tasks are exported to a single XML file. The default name of the exported task is *Tasks Export*.

Delete Tasks

If you no longer require multiple tasks, you can delete them from MOVEit Automation.

To delete multiple tasks simultaneously, complete the following steps.

1. Open the TASKS tab, and use the filters to list the tasks that you want to delete.
2. Click Bulk Actions > Delete. The Confirm delete tasks dialog box displays.
3. To delete the tasks, click Yes. The Deleted selected tasks message displays.

SCRIPTS

The tasks are deleted, and removed from the task list.

About scripts

To use a script, you add a process to a traditional or advanced task, select a script for the process to run, set parameter values as required, and select options for how to run the script. For more information, see ***Add a Task Process*** (on page 71).

The SCRIPTS page lists all available scripts. You can ***filter the list*** (on page 12).

On the SCRIPTS page you can:

- View built-in scripts

  MOVEit Automation contains built-in scripts that you use as-is. You can view the description and parameters of built-in scripts, but you cannot view or edit the source code. You set the parameters for the script after you associate it with a process.

  For more information, see

  - ***Types of Built-in Scripts*** (on page 104)
  - ***Built-in Scripts by Script Name*** (on page 106)
- Add and manage custom scripts

  This feature is available in MOVEit Automation Enterprise.

Custom scripts are written in VBScript. You can write scripts, or import them from the directory of sample scripts provided.

Sample scripts are installed with the MOVEit Automation Admin Console. If you are using MOVEit Automation Web Admin, sample scripts are available from the ***Customer Portal*** (***https://ipswitchft.secure.force.com/cp/***). Log in to the portal and click My Products.

For more information, see

- ***Add, Edit, or Import Custom Scripts*** (on page 137)
- ***Examples of Custom Scripts*** (on page 147)

Types of Built-in Scripts

MOVEit Automation provides built-in scripts for purposes such as PGP encryption/decryption, zipping/unzipping files in archives with parameters, find-and-replace operations, and invocation of command-line applications. These scripts are available after MOVEit Automation is installed. You cannot view or change the source code of built-in scripts.

§ To view built-in scripts in Web Admin, click SCRIPTS. You can ***filter the list*** (on page 12).

§ To see a script's parameters, click the script name.

You can also create your own custom scripts. For more information, see ***Custom Scripts*** (on page 137).

To use a script, you create a task, add a process to the task, and associate the script with that process. The process defines details of how to run the script. For more information, see ***Add a Task Process*** (on page 71).

Types of built-in process scripts

Process scripts can be run as processes in tasks

§ ***Process Scripts provided with MOVEit Automation*** (on page 104).

§ ***Process Scripts provided when PGP license is enabled*** (on page 106).

See also:

§ ***Built-in Scripts by Script Name*** (on page 106)

Process Scripts

Built-in Process scripts provided with MOVEit Automation. For details, click the script name.

Process Script	Description
Certs Backup (on page 107)	Extracts client certificates from Windows and saves them to two files for backup purposes.
Certs Restore (on page 107)	Accepts as input PFX files created by the Certs Backup script, and imports them into Windows

Process Script	Description
Command Line App (on page 108)	Runs a command line application. Parameter placeholders to indicate input file and output file/folder allow for seamless integration into MOVEit Automation tasks
Find Or Replace (on page 111)	Finds (that is, counts) or replaces particular words or phrases in files. Also can strip or replace special characters such as tabs and line feeds
Header ID (on page 113)	Reads the first (Header) line of a file, determines the ID from a set character position and passes this as a new task parameter. Also can strip the first line.
HTTP Get (on page 114)	Downloads a file from a webserver, using GET.
HTTP Post (on page 114)	Uploads a file to a webserver, using the POST verb.
HTTP Put (on page 115)	Uploads a file to a webserver, using the PUT verb.
HTTP SharePoint Get (on page 116)	Downloads a file from a Microsoft SharePoint server.
HTTP SharePoint Put (on page 116)	Uploads a file to a Microsoft SharePoint server.
Ignore All Files (on page 117)	Causes all files already downloaded or added via a script to be ignored in subsequent processing steps.
Look Up (on page 117)	Looks up one or more values from a table given a key such as a file name, uploading username or source path.
MessageWay Translation (on page 123)	Sends a file to a MessageWay server for translation or data format conversion.
No Op (on page 125)	"No Operation". Generally used to force MOVEit Automation to delete files from a source without transferring them.
Prepend Lines (on page 127)	Inserts up to four lines at the beginning of a file.
Report Long Running Tasks (on page 128)	Queries the micstats database and reports back information about any tasks that have been running longer than the specified time interval.
Set Destination (on page 129)	Changes the host definition (and optionally, the path) of the destination used in this task run.
Sleep (on page 130)	Pauses task operation for a specified number of seconds or milliseconds.
SMIME Receive (on page 131)	Retrieves and decrypts an S/MIME encoded email attachment.
SMIME Send (on page 132)	Encrypts a file and sends it as an S/MIME email attachment.
Tamper Detect (on page 132)	Checks for tampering in the audit and activity history tables.
Trim Statistics DB (on page 133)	Archives old MOVEit Automation log records into a flat file every few days.
Unzip Advanced (on page 134)	Unzips ZIP archives with an optional ZIP password.
XSL Transform (on page 135)	Transforms XML documents with XSL stylesheets/templates
Zip Advanced (on page 136)	Creates ZIP archives with one or multiple files, configurable level of compression and an optional ZIP password.

Process Scripts for PGP Files

The following built-in Process scripts are available when a MOVEit Automation PGP license has been enabled.

For information about how to use MOVEit Automation to encrypt and decrypt PGP files, see the ***About PGP Keys*** (on page 163).

Process Scripts for PGP files	Description
PGP Decrypt (on page 126)	Decrypts PGP files.
PGP Encrypt and Sign (on page 126)	Encrypts and signs PGP files.
PGP Encrypt Only (on page 127)	Encrypts PGP files, does not sign them.

Built-in Scripts by Script Name

For description, input parameters, and usage notes, click the script name.

Certs Backup

Certs Backup extracts the MOVEit Automation client certificates from its Windows Certificate Store and saves them to two files so they can be saved/sent to a destination for backup purposes.

To restore the certificates in these files, use the built-in script ***Certs Restore*** (on page 107).

To use Certs Backup, first configure a source-less task with a per-task process that runs this script. Then configure one or more destinations to save the files to their final locations.

For an example of a complete task and a recommendation on its use, see ***Central Service - Backup - Automated Configuration Replication*** (on page 370).

Input Parameters

- Certs_Password (*Required*) - The password that will encrypt the output PFX files.
- Certs_Filename_Personal - The filename (no path) of the output PFX file containing the certificates (with private keys) from the My Certs/Personal store. Default value is "CertsPersonal.pfx". Date macros are allowed.
- Certs_Filename_OtherPeople - The filename (no path) of the output PFX file containing the certificates (no private keys) from the Other Certs/Other People store. (Currently used only for SMIME recipients and some AS1/AS2/AS3 certificates.) Default value is "CertsOtherPeople.pfx". Date macros are allowed.

Notes

For diagnostic purposes only, the MOVEit Automation service can be run on the desktop in the foreground, under the user who is signed in on the PC. In this situation, if the signed-in user is different from the user who is configured to run the service, all certificates are inaccessible because certificates are stored by user. To run in the foreground and do certs backup, log in to the desktop with the same service user credentials.

Certs Restore

Certs Restore imports client certificates from input PFX files created by ***Certs Backup*** (on page 107) into the Windows Certificate Store used by MOVEit Automation.

To use Certs Restore, configure a task with a source that downloads the two files created by ***Certs Backup*** (on page 107). Then configure a per-file process to run this script to restore the certificates in those two files.

An example of a complete task and a recommendation on its use can be found in the ***Central Service - Backup - Automated Configuration Replication*** (on page 370) documentation.

Input Parameters

- Certs_Password (*Required*) - The password that the built-in ***Certs Backup*** (on page 107) script used to encrypt the PFX files.

Notes

When MOVEit Automation is running as a foreground application, this script will generally not work properly because a different user's Windows Certificate Store will generally be used in foreground mode than when MOVEit Automation is running as a service.

Command Line App

"Command Line App" runs a command line application. It contains parameter placeholders to indicate input and output files.

Input Parameters

- CommandLineApp_AppPath (*Required*) - Specifies the full path of the command line application to run.
- CommandLineApp_AppParms - Specifies the parameters passed into the command line application. Use the special macro [InputFile] to indicate the name of the file against which the command line application should execute. If you expect the command line application to write new output use either [OutputFile] for a single output file or [OutputFolder] to indicate an output folder (which can include multiple files and subfolders). Pipe output to [StdOut] and [StdErr] macros to have this information displayed in the debug log and in parameters for later use.
- CommandLineApp_AltReturnCodes - By default, a return code of "0" indicates success. You can use this parameter to add or change the return code or codes that indicate success. Each values must be separated by a comma. For example, if the values "3,27,52" are applied to this parameter, each of the three return codes signal success. If the return code of "0" is removed from the parameter, it no longer indicates success.

Notes

The "[InputFile]", "[OutputFile]", "[OutputFolder]", "[StdOut]" and "[StdErr]" macros used with this built-in process are not available for general use in other MOVEit Automation sources, destinations, processes or next actions.

When the "[OutputFolder]" macro is used, any "[InputFile]" cache files are ignored in the final output. For example, if CommandLineApp was used to run "unzip.exe [InputFile] -d [OutputFolder]", the Zip file indicated by the "[InputFile]" parameter is not included in the set of files sent to the destination.

To record any "standard" or "error" output written by your command line application, append the following phrase to your usual CommandLineApp_AppParms value.
> [StdOut] 2> [StdErr]
For example, if your original CommandLineApp_AppParms value is "a c:\windows\system32\eula.txt", you can record output with a revised CommandLineApp_AppParms value of "a c:\windows\system32\eula.txt > [StdOut] 2> [StdErr]". Any "StdErr" output appears in the debug log at the "Warnings" level and higher and any "StdOut" output will appear in the debug log at the "Some Debug" level and higher. To use the first 8192 characters of each type of output in your Next Actions, use the related "[Parm:CommandLineApp_StdOut]" and "[Parm:CommandLineApp_StdErr]" output parameters.

This built-in script can be run per-file or once-after-all-files. This built-in script can be run as the first step of a task.

Date macros (on page 182) are frequently used with command line arguments. Operators (such as the minus sign) normally apply to all times and dates in a macro phrase. To apply operators to only part of a macro phrase, use double-quotes to delimit phrases.

For example, if today is currently July 5, 2007, a macro of:

- [dd][mm-][yyyy] [dd][mm][yyyy] yields 05062007 05062007
- "[dd][mm-][yyyy]" [dd][mm][yyyy] yields "05062007" 05072007

Example #1

Joe wants to run a command line application to read files passing through MOVEit Automation and make sure they contain valid data. His application will NOT change the contents of the files. The syntax used by his application is "checkapp.exe -verify *(input file)*" and his application is installed into his "C:\Program Files\VerifyIt" folder.

To integrate this application with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Select "Command Line App" as the process
3. Set process parameters:
   - CommandLineApp_AppPath = "C:\Program Files\VerifyIt\checkapp.exe"
   - *(Add)* CommandLineApp_AppParms = "-verify [InputFile]"

To see any standard or error output generated by the command line application as it ran, add the phrase " > [StdOut] 2> [StdErr]" to the end of the CommandLineApp_AppPath value. This shows this output in his debug log: command line errors at the "Warning" level or higher and other output at the "Some Debug" level or higher.

To see this kind of output in an email or send it to another task, use the output macros "[Parm:CommandLineApp_StdOut]" and "[Parm:CommandLineApp_StdErr]".

Example #2

Nancy wants to run a command line application to process files passing through MOVEit Automation. Her application will change the contents of the files if they are valid and return a non-zero error code if the files are not valid. The syntax used by her application is "alterapp.exe -x207 -i=*(input file)* -o=*(output file)*" and her application is installed into her "D:\AlterProg" folder.

To integrate this application with MOVEit Automation:

1. Create a new task with a source, process, destination, and schedule.
2. Select "Command Line App" as the process.
3. Set process parameters:
   - CommandLineApp_AppPath = "D:\AlterProg\alterapp.exe"
   - *(Add)* CommandLineApp_AppParms = "-x207 -i=[InputFile] -o=[OutputFile]"

Example #3

Pedro wants MOVEit Automation to transfer reports created by a stand-alone command-line application. His application does not need a "source file" and returns a non-zero error code if it cannot create its reports. The syntax used by his application is "makereports.exe -repcode=76 *(output file)*" and his application is installed into his "C:\Program Files\DBExtracts" folder.

To integrate this application with MOVEit Automation:

1. Create a new task with a process, destination and schedule. (No source).
2. Select "Command Line App" as the process.
3. Set process parameters:

- CommandLineApp_AppPath = "C:\Program Files\DBExtracts\makereports.exe"
- *(Add)* CommandLineApp_AppParms = "-repcode=76 [OutputFile]"

Example #4

Paul wants to run a command line "unzip with odd encryption" application on files passing through MOVEit Automation. Each original archive file can contain one or more files, and might include files in archived subfolders. The syntax used by his application is "oddzip.exe -enc=codfish -ifil=*(input file)* -ofol=*(output folder)*" and the application is installed into the "D:\OddZip" folder.

To integrate this application with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Select "Command Line App" as the process.
3. Set process parameters:
   - CommandLineApp_AppPath = "D:\OddZip\oddzip.exe"
   - (Add) CommandLineApp_AppParms = "-enc=codfish -ifil=[InputFile] -ofol=[OutputFolder]"

To control whether to respect the subfolders in which the archive file's members were stored in the Destination element, check or uncheck the **Use Relative Subdirectories** option.

Find Or Replace

Note: A Replace operation can be applied only to text files. If a binary file is encountered, the script might fail with a macro error.

Find Or Replace Finds (that is, counts) or replaces particular words or phrases in files. It can also handle special characters such as tabs and line feeds.

Input Parameters

- FindOrReplace_Find (*Required*) - Specifies a value to find.
- FindOrReplace_Replace - Specifies a value to replace. (Ignored if Action is "Find".)
- FindOrReplace_Action (*Required*) - Specifies whether to just find (and count) instances or replace them as well.
- FindOrReplace_CaseSensitive (*Required*) - Specifies whether comparisons are case sensitive or not.
- FindOrReplace_SkipFileSizeMB - If a file is larger than this (MB) and if special characters are being used, then skip that file. The default is 100; maximum value is 500 (500MB).

Output Parameters

- FindOrReplace_Count - Returns the number of times the Find value appeared in this file.
- FindOrReplace_CountTask - Returns the number of times the Find value appeared in all files encountered during this task run.

Special Characters

FindOrReplace can handle special character operations such replacing all tabs with three spaces or stripping carriage returns from a file. To work with special characters, represent those characters by using special character macros in your FindOrReplace_Find or FindOrReplace_Replace parameters as appropriate.

The following special character macros are supported.

- [char_cr] - Carriage return (ASCII 13)
- [char_lf] - Line feed (ASCII 10)
- [char_crlf] - Carriage return followed by a line feed (ASCII 13+10)
- [char_tab] - Tab (ASCII 9)
- [char_ff] - Form feed (ASCII 12)
- [char_null] - Null (ASCII 0)
- [char_###] - ASCII code ###

For example:

- To replace tabs with three spaces, configure:
  - FindOrReplace_Find = "[char_tab]"
  - FindOrReplace_Replace = " "
- To strip line feeds, configure:
  - FindOrReplace_Find = "[char_lf]"
  - FindOrReplace_Replace = ""
- To add a carriage return to the end of each line feed, configure:
  - FindOrReplace_Find = "[char_lf]"

- FindOrReplace_Replace = "[char_crlf]"

- To replace "doAg" with "BonE", configure:
  - FindOrReplace_Find = "do[char_65]g"
  - FindOrReplace_Replace = "[char_66]on[char_69]"

  or, noting that [char_65] is a capital A, configure:

  - FindOrReplace_Find = "doAg"
  - FindOrReplace_Replace = "BonE"

Notes

Use the **replace** operation only against text files. Do not use to replace text in a binary file; this could have unforeseen consequences.

You can run this built-in script only one time per file.

This script cannot be run as the first step in a task.

Example #1

Ed wants email that lists the number of times that the word "- PAGE" appears in files that pass through MOVEit Automation.

To perform this operation with MOVEit Automation:

1. Create a new task with a source, process, destination, schedule and next action.
2. Select "Find Or Replace" as the process
3. Set process parameters:
   - FindOrReplace_Action = "Find"
   - FindOrReplace_Find = "- PAGE"
   - FindOrReplace_CaseSensitive = "Yes"
4. Configure the next action to:
   - Send an email with the macro "[Parm:FindOrReplace_Count]" in the message body.
   - Run per-file (not per-task).

Example #2

Jane wants to replace all instances of "bubbler" with "drinking fountain" in files that pass through MOVEit Automation.

To perform this operation with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Select "Find Or Replace" as the process
3. Set process parameters:
   - FindOrReplace_Action = "Replace"
   - FindOrReplace_Find = "bubbler"
   - FindOrReplace_CaseSensitive = "No"
   - *(Add)* FindOrReplace_Replace = "drinking fountain"

Header ID

Header ID reads a text string on the first (Header) line of a file from a set character position. A new task parameter called HeaderID_Value can be used for integration into MOVEit Automation tasks.

Input Parameters

§ HeaderID_Length (*Required*) - Specifies the number of characters (length) to read from the first line of the file.

§ HeaderID_Start (*Required*) - Specifies the starting character position (column) from which to read the text string value.

§ HeaderID_Strip - Set this optional parameter to "Yes" if you want to remove the entire first line after capturing the Header ID_Value.

Output Parameter

§ HeaderID_Value - Returns the text string found on the first line, beginning at HeaderID_Start and continuing for HeaderID_Length characters.

Example #1

Bill wants to read a value from the first line of a file starting in column 1 and continuing 8 characters, and then use this text string to rename the file on the task's destination with a date stamp extension.

To integrate this application with MOVEit Automation, Bill does the following:

1. Creates a new task with a source, process, destination and schedule.
2. Selects "Header ID" as the process.
3. Sets process parameters:
   - HeaderID_Start = 1
   - HeaderID_Length = 8
4. Edits the Destination to set FileName to [Parm:HeaderID_Value].[YYYY][MM][DD]

Example #2

Cheryl wants to read a value from the first line starting in column 5 and continuing 6 characters. She wants to discard this first line and use the HeaderID_Value in conjunction with a LookUp process to find the folder on a FTP server where each respective file should go.

To integrate this application with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Select "Header ID" as the 1st process
3. Set the Header ID process parameters:
   - HeaderID_Start = 5
   - HeaderID_Length = 6
4. Add a second process, choosing the built in "Look Up" script
5. Set the Look Up process parameters:

- LookUp_Key = [Parm:HeaderID_Value]
- LookUp_FilePath = C:\LookUp\List1.txt (Note: This is an example)
- LookUp_ActionIfKeyNotMatched = Throw_Error
- LookUp_MatchType = Require_Exact_Match

6. Edit the Destination to set Folder to [Parm:LookUp_Value]

HTTP Get

HTTP Get downloads a file from a website, using HTTP[S] with the GET verb. This script is not suitable for files larger than a few dozen megabytes, because it loads the entire file into memory before saving it.

Use this script like a source. Then configure one or destinations to save the files to their final locations.

Input Parameters

- HTTP_URL *(Required)* - The full URL of the file; for example, https://myserver/reports/Daily.txt. Date and parameter macros are permitted.
- HTTP_IgnoreCertProblems - Whether to ignore problems with the remote server certificate (such as signer not trusted). Default is False.
- HTTP_User - The username, for HTTP authentication.
- HTTP_Password - The password, for HTTP authentication.
- HTTP_DestFilename - The destination filename. If not specified, the filename from the URL is used. Date and parameter macros are permitted.

Notes

HTTP Get works only when the website requires either no authentication, or HTTP authentication.

HTTP Get *does not work* if the website requires the user to sign on via a web form, and/or if the user must navigate through the site in order to access the download page.

HTTP Post

HTTP Post uploads a file to a website, using HTTP[S] with the POST verb. POST is the mechanism used by most websites that accept browser-based uploads.

Use this script like a destination, as a per-file process. Typically you would use this in a task with one or more sources.

Input Parameters

- HTTP_URL (*Required*) - The URL to post to; for example, https://myserver/cust/upload.aspx. May include macros.
- HTTP_IgnoreCertProblems - Whether to ignore problems with the remote server certificate (such as signer not trusted). Default is False.
- HTTP_User - The username, for HTTP authentication.

Note: If the website uses form-based authentication rather than HTTP authentication, this script does not work.

- HTTP_Password - The password, for HTTP authentication.
- HTTP_DestFilename - The destination filename. If not specified, the original filename is used.
- HTTP_FileFormField (*Required*) - The name of the form field for the file contents. Can include macros.
- HTTP_ExtraFields - The names and values of optional extra form fields to be provided along with the POSTed file. This is very application-specific. The format is a string like fldname1=value1|fldname2=value2|... Can include macros.
- HTTP_MaxFileSizeMB - If a file is larger than this (MB), then skip that file and signal an error. Default 100; max value 500.

Notes

HTTP Post works only when the website requires either no authentication, or HTTP authentication.

HTTP Post does not work for websites that require the user to sign on via a web form to authenticate.

This script is not suitable for files larger than a few dozen megabytes, because it loads the entire file into memory before saving it.

HTTP Put

HTTP Put uploads a file to a website, using HTTP[S] with the PUT verb.

Note: PUT is not the usual way to upload files, and is not accepted by many websites. The ***HTTP Post*** (on page 114) script might work as an alternative for these sites. However, when available, the HTTP PUT technique is preferable because it is simpler.

Use this script like a destination, as a per-file process. Typically you would use this in a task with one or more sources.

Input Parameters

- HTTP_URL (*Required*) - The URL of the folder to upload to; for example, https://myserver/reports. Can include macros.
- HTTP_IgnoreCertProblems - Whether to ignore problems with the remote server certificate (such as signer not trusted). Default is False.
- HTTP_User - The username, for HTTP authentication.
- HTTP_Password - The password, for HTTP authentication.
- HTTP_DestFilename - The destination filename. If not specified, the original filename is used.
- HTTP_MaxFileSizeMB - If a file is larger than the specified number of (MB), then skip that file and signal an error. Default 100; max value 500.

Notes

HTTP Put works only when the website requires either no authentication, or HTTP authentication.

HTTP Put *does not work* for websites that require the user to sign on via a web form to authenticate.

To enable PUT on a remote webserver running Microsoft IIS 6.0, use IIS Manager on that webserver to enable the WebDAV Web Service Extension, and turn on Write access to the virtual directory.

This script is not suitable for files larger than a few dozen megabytes, because it loads the entire file into memory before saving it.

HTTP SharePoint Get

HTTP SharePoint Get downloads files from a Windows SharePoint Server website, using HTTP[S]. This script is not suitable for files larger than a few dozen megabytes, because it loads the entire file into memory before saving it.

Use this script like a source. Then configure one or destinations to save the files to their final locations.

Input Parameters

- SharePoint_BaseURL (*Required*) - The base URL of the website.
  For SharePoint 2003, this will be something like https://server.
  For SharePoint 2007, this will be something like https://server/Docs.
- SharePoint_DeleteAfterDownload - Whether to delete the source file from SharePoint after a successful download. Default is False.
- SharePoint_IgnoreCertProblems - Whether to ignore problems with the remote server certificate (such as signer not trusted). Default is False.
- SharePoint_HTTPUser - The username, for HTTP authentication.
- SharePoint_HTTPPassword - The password, for HTTP authentication.
- SharePoint_Folder (*Required*) - The name of the SharePoint folder, including parent folders if applicable. Date and parameter macros are allowed.
  For SharePoint 2003, use "Shared Documents" for the default library.
  For SharePoint 2007, use "Documents" for the default library.
- SharePoint_FileMask (*Required*) - The filemask specifying which files to download. For example, `*.txt`. Date and parameter macros are permitted.

HTTP SharePoint Put

HTTP SharePoint Put uploads a file to a Windows SharePoint Server website, using HTTP[S]. This script is not suitable for files larger than a few dozen megabytes, because it loads the entire file into memory before sending it.

Use this script like a destination, as a per-file process. Typically you would use this in a task with one or more sources.

Input Parameters

- SharePoint_BaseURL (*Required*) - The base URL of the website. For SharePoint 2003, this will be something like https://server. For SharePoint 2007, this will be something like https://server/Docs.
- SharePoint_IgnoreCertProblems - Whether to ignore problems with the remote server certificate (such as signer not trusted). Default is False.
- SharePoint_HTTPUser - The username, for HTTP authentication.
- SharePoint_HTTPPassword - The password, for HTTP authentication.
- SharePoint_Folder (*Required*) - The name of the SharePoint folder, including parent folders if applicable. Macros are permitted.
  For SharePoint 2003, use "Shared Documents" for the default library.
  For SharePoint 2007, use "Documents" for the default library.

- SharePoint_Filename - The name to be given to the file. If not specified, the original filename is used. Macros are permitted.
- SharePoint_MaxFileSizeMB - If a file is larger than the specified number of (MB), then skip that file and signal an error. Default 100; max value 500.

Ignore All Files

Ignore All Files causes all files already downloaded or added via a script to be ignored in subsequent processing steps.

Input Parameters

- IgnoreAllFiles_Error5010 (*Required*): Affects which Next Action is executed at the end of the task, and allows you to select how subsequent processing occurs. Values: True - The script returns error 5010 to indicate No Files. False (default) - Returns 0 to indicate Success.

  When a task checks multiple sources for files, but finds no files, the IgnoreAllFiles script returns a Success message, which could be confusing. When you set this parameter to *True*, it will always return error 5010, whether files were found or not.

Example #1

You use trigger files to determine which source files need to be transferred, and you want to exclude the trigger files from the source list.

To perform this operation with MOVEit Automation:

1. Download X trigger files from a source.
2. For each trigger file, run custom "Process Trigger Files" script, which reads each trigger file and builds a list of files to download into a custom task parameter.
3. Run Ignore All Files to clear the trigger files from the file processing list.
4. Download all files specified by the custom task parameter built from the trigger files.
5. Upload files to the target destination.

Look Up

Look Up looks up a key, usually expressed with a macro, against a text file that is filled with one column of keys and up to five columns of values.

For example, given a file name of "report56.txt" (in the Lookup_Key parameter) and the path to a file containing the following keys and values (in the Lookup_FilePath parameter), the Look Up built-in script returns a value of "customer13.rpt" (in the Lookup_Value parameter). To obtain this Lookup_Value, the Look Up built-in script compares Lookup_Key to the "file keys" and returns the related value from the first matching line.

report44.txt,customer01.ttk

report56.txt,customer13.rpt

report66.txt,customer87.lml

Input Parameters

- LookUp_Key (*Required*) - The value to look up.
- LookUp_FilePath (*Required*) - Specifies the path where the lookup file containing delimited keys and values is located.
- LookUp_NumValues - The number of value columns that the script looks for within the lookup table. Default is 5 value columns.
- LookUp_CaseSensitive (*Required*) - Specifies whether key comparisons are case-sensitive. Default is No. This parameter applies for all values value of LookUp_MatchType.
- LookUp_Delimiter - Specifies an alternate column delimiter. Default value is a comma (,). For example, if a single line of keys and values is "753|blue|large", set this parameter to a vertical bar (|).
- LookUp_CommentChar - The alternate character to indicate that a line is a comment line, not a data line. Default is a single quote ('). For example, if a single line of comment is "# This is a comment", set this parameter to a pound (hash) sign (#).
- LookUp_ActionIfKeyNotMatched (*Required*) - Specifies the action to take if LookUp_Key does not match any values in the table. Values:
  - Throw_Error (*Default*) - End this task run and/or file transfer with an error (#120).
  - Ignore_File - End this task run and/or file transfer with an "ignore this file" status code. Normally, use this option only if your Lookup script has been set to run "per file" so individual files can be ignored.
  - Return_Key_in_Values - Continue this task run and/or file transfer without error, and copy the value of LookUp_Key into all the LookUp_Value* parameters.
  - Return_Blank_Values - Continue this task run and/or file transfer without error and set the LookUp_Value* parameters to a blank value.
- LookUp_MatchType (*Required*) - Specifies how LookUp_Key should be looked up against keys in the file table. The term "file key" refers to the values in column 1 in the lookup file. Values:
  - Require_Exact_Match (*Default*) - LookUp_Key must exactly match keys in the file. Examples:

    LookUp_Key "dog.txt" matches file key "dog.txt", but does not match the file key "do".

    LookUp_Key "do" does not match file key "dog.txt".
  - Allow_Partial_Match_of_LookUp_Key - LookUp_Key is treated as a "partial key" and matches file keys if they appear anywhere in the file keys. Wildcards are permitted (see list of wildcards below).

    Example: A LookUp_Key of "dog.txt" or "dog" matches file key "dog.txt", but does not match the file key "do".
  - Allow_Partial_Match_of_File_Keys - File keys are treated as "partial keys" and match LookUp_Key if they appear anywhere in the lookup key. Wildcards are permitted (see list of wildcards below).

    Example: LookUp_Key "dog" matches file key "do", but not file key "dog.txt".
- LookUp_ReturnAs - Optional, alternate name for LookUp_Value return value.

  Example: If LookUp_ReturnAs="AltFolder" and LookUp_Value="\another\fol", then the macro "[Parm:AltFolder]" is interpreted as "\another\fol" during the rest of the task run.

Wildcards

Lookup keys can contain wildcards, except when "Require_Exact_Match" is in effect. The wildcard syntax is similar to Windows filename masks, with some enhancements:

Wildcard char	Meaning
*	Match zero or more of any character.
?	Match exactly one of any character.
@	Match exactly one alphabetic character.
#	Match exactly one digit.
anything else	Match exactly that character.

Examples:

dog#	Matches "dog3" and "Mydog8acat". Does not match "dog" or "dogs3"
dog*7	Matches "dog7", "dog37" and ")dog/7PP". Does not match "dog" or "do7".

Output Parameters

- **LookUp_Value** - Value found by looking up LookUp_Key. If there are multiple data columns in the lookup table, this is the value from the first column. If LookUp_Key was not matched, this value is controlled by the LookUp_MatchType parameter.
- **LookUp_Value2** - Similar to LookUp_Value, except from data column 2, if available.
- **LookUp_Value3** - Similar to LookUp_Value, except from data column 3, if available.
- **LookUp_Value4** - Similar to LookUp_Value, except from data column 4, if available.
- **LookUp_Value5** - Similar to LookUp_Value, except from data column 5, if available.
- **LookUp_ValueN** - Similar to LookUp_Value, except from data column N, if available.
- **(Value of LookUp_ReturnAs)** - If a value has been provided in the LookUp_ReturnAs parameter, the value of LookUp_Value is also be returned as a parameter that has that name. For example, if LookUp_ReturnAs="AltFolder" and LookUp_Value="\another\fol", then the macro "[Parm:AltFolder]" is interpreted as "\another\fol" during the rest of the task run.

Notes

In most production cases, you will probably use macros such as "[OrigName]", "[OnlyName]", "[RelativePath]" or "[MID([OrigName], 2, 3)]" in your "LookUp_Key" parameter.

This built-in script can be run wherever processes are allowed, including alone in its own task.

Example #1

Fred wants to look up a specific internal folder based on the username that uploaded a file. If a record for a particular username cannot be found, log an error.

To perform this operation with MOVEit Automation:

1. Create a "lookup table" file containing content similar to the following and save it as "d:\customer2folder\fred.txt"

   ' Format is username,folder
   jack,D:\blue\2134
   diane,D:\red\3734
   american,D:\blue\3357
   kids,D:\red\1651
   heartland,D:\red\2162
2. Create a new task with a source, process, destination, and schedule.
3. Select "Look Up" as the process.
4. Set process parameters:
   - LookUp_Key = "[OrigUser]"
     (Username of user who uploaded the file; only works on MOVEit Transfer sources.)
   - LookUp_FilePath = "d:\customer2folder\fred.txt"
   - LookUp_CaseSensitive = "No"
   - LookUp_MatchType = "Require_Exact_Match"
   - LookUp_ActionIfKeyNotMatched = "Throw_Error"
5. Configure the destination to:
   - use the macro "[Parm:LookUp_Value]" in the "Path" field.

Given the contents of "fred.txt" shown above, the following uploader usernames causes the following values to be placed into the LookUp_Value parameter.

Uploader Username	LookUp_Value
diane	D:\red\3734
heartland	D:\red\2162
congress	*(NONE - ERROR)*

Example #2

Nancy wants to change the names that several of her files are saved as, but many file names already have the correct names.

To perform this operation with MOVEit Automation:

1. Create a "lookup table" file containing content similar to the following and save it as "d:\correctoddfiles\nancy.txt"

   ```
   ' Format is incoming filename, corrected filename
   JHJ45KK,nice_J45.dat
   JTE_KTR_67,nice_K67.dat
   K_P0KX_R89,nice_L89.dat
   ```
2. Create a new task with a source, process, destination, and schedule.
3. Select "Look Up" as the process.
4. Set process parameters:

- LookUp_Key = "[OrigName]"
- LookUp_FilePath = "d:\correctoddfiles\nancy.txt"
- LookUp_CaseSensitive = "Yes"
- LookUp_MatchType = "Require_Exact_Match"
- LookUp_ActionIfKeyNotMatched = "Return_Key_in_Values"
  (If there is no match, let the original file name "fall through".)

5. Configure the destination to:
   - use the macro "[Parm:LookUp_Value]" in the "Filename" field.

Given the contents of "nancy.txt" displayed above, the following filenames cause the following values to be placed into the LookUp_Value parameter.

Source File Name	LookUp_Value
JHJ45KK	nice_J45.dat
hello.txt	hello.txt
K_POKX_R89	nice_L89.dat
K_pokx_R89	K_pokx_R89

Example #3

Ed wants to set FTP mainframe parameters based on the names of files MOVEit Automation has just downloaded. He wants to be able to handle both specific filenames (for example, "four.txt"), general extension (for example ".txt") and provide a "catch-all" value.

To perform this operation with MOVEit Automation:

1. Create a "lookup table" file containing content similar to the following and save it as "d:\blocking\ed.txt"

```
' List specific filenames first
four.txt,80,4
' Next, list specific extensions
.txt,80,10
.dat,133,5
' Finally, provide a catch-all case
' (this assumes all incoming filenames will contain a period)
.,80,10
```

2. Create a new task with a source, process, destination, and schedule.
3. Select "Look Up" as the process.
4. Set process parameters:
   - LookUp_Key = "[OrigName]"
   - LookUp_FilePath = "d:\blocking\ed.txt"
   - LookUp_CaseSensitive = "No"
   - LookUp_MatchType = "Allow_Partial_Match_of_File_Keys"
     (This allows "five.txt" to match the ".txt" file record.)
   - LookUp_ActionIfKeyNotMatched = "Return_Blank_Values"
5. Configure the FTP destination to:

- use the macros "[Parm:LookUp_Value]" and "[Parm:LookUp_Value]" in the "additional commands to execute before transfer" field.

Given the contents of "ed.txt" displayed above, the following source file names cause the following values to be placed into the LookUp_Value and LookUp_Value2 parameters.

Source File Name	LookUp_Value	LookUp_Value2
four.txt	80	4
five.txt	80	10
six.dat	133	5
seven.rpt	80	10

Example #4

Ralph wants to scan an entire folder structure and transfer only the files that match names in his lookup table file. Files that do match should usually be renamed.

To perform this operation with MOVEit Automation:

1. Create a "lookup table" file containing content similar to the following and save it as "d:\onlysome\ralph.txt"

```
 ' List specific filenames first
ur.txt,rrr[OnlyName].xtx
' Next, list specific extensions
.txt,[OrigName]
.dat,[OnlyName].tad
```

2. Create a new task with a source, process, destination, and schedule.
3. Select "Look Up" as the process.
4. Set process parameters:
   - LookUp_Key = "[OrigName]"
   - LookUp_FilePath = "d:\onlysome\ralph.txt"
   - LookUp_CaseSensitive = "No"
   - LookUp_MatchType = "Allow_Partial_Match_of_File_Keys"
     (This will allow "five.txt" to match the ".txt" file record.)
   - LookUp_ActionIfKeyNotMatched = "Ignore_Files"
5. Configure the destination to:
   - use the macro "[Parm:LookUp_Value]" in the destination "filename" field.

Given the contents of "ralph.txt" displayed above, the following source file names cause the following files to appear on the destination with the following names.

Source File Name	Destination File Name
four.txt	rrrfour.xtx
five.txt	five.txt
six.dat	six.tad
seven.rpt	*(file ignored)*

MessageWay Translation

MessageWay Translation sends a file to a MessageWay server for translation or data format conversion.

The resulting output files are added to the list of files to process, which typically means they are sent to one or more destinations, depending on the task's configuration. The original input file is ignored, which means that the task will do no further processing on it, and it will not be sent to task destinations.

Depending on the input file and the translation rules configured on the MessageWay server, a single input file might result in multiple output files. If no processing rules match the input file, or if errors occur, no output files might be created.

The MessageWay translation engine is very powerful, and the various EDI (Electronic Data Interchange) standards governing data format are complex. For detailed information on how to set up a MessageWay server for translation, see the translation-related MessageWay manuals, such as *MessageWay Translator Workbench User's Guide and Reference*

For more information about the script itself, see the topic Common Applications - MessageWay Translation.

Input Parameters

- MWayConn_Host (*Required*) - Hostname of the MessageWay server.
- MWayConn_SSL - Whether to use SSL. Default is True.

> Note: if you use SSL, you must specify the certificate fingerprint of the MessageWay server in MWayConn_SSLFingerprint. If the server is on the same computer as MOVEit Automation, you can safely specify False here and avoid having to know the certificate's fingerprint.

- MWayConn_Port - TCP port number. Default: 6280 if not SSL, or 6243 if SSL. You generally do not need to specify this parameter.
- MWayConn_SSLFingerprint - Hexadecimal fingerprint (MD5 or SHA1) of the server's certificate. Required if MWayConn_SSL is True. (There is no way to specify that any certificate should be accepted.) The string consists of groups of 2 hex characters separated by spaces.
- MWayConn_User (*Required*) - MessageWay username.
- MWayConn_Password (*Required*) - MessageWay user's password.
- MWayConn_Recipient - Destination of translated files. For example, the destination "translate:moveit" means that there must be a translation location named "translate" and a mailbox named "moveit" configured in MessageWay. The specified user must have sufficient access to these locations.
- MWayConn_Sender - Arbitrary sender's name. Can include macros. The MessageWay translation engine can base its translation rules partly on the sender's name.
- MWayConn_MIMEType - Arbitrary MIME type string. Can include macros.
- MWayConn_RetryCount - Maximum number of retries to connect to MessageWay. Default is 0. The script retries only if certain types of errors occur, such as an inability to connect to the MessageWay server.
- MWayConn_RetrySeconds - Number of seconds between attempts to connect to MessageWay. Default is 30.
- MWayConn_MaxSeconds - The maximum number of seconds to wait for MessageWay to process the file. A value of 0 means no limit. May include macros. Default is 7200 (two hours).
- MWayConn_PollIntervalSeconds - The number of seconds to wait between queries to MessageWay to determine whether processing has completed. Can include macros. Default is 5 seconds.
- MWayConn_ExceptionsInsteadOfData - Tells the script how to behave when any exception occurs (meaning poorly formatted data).

  True: No data files are returned. Instead, the exception report files are returned.

  False (default) Only data files are returned. If False and exceptions do occur, you must look up the exception reports directly through MessageWay.
- MWayConn_TraceFilename - The full path to a file that receives a detailed trace log of the script's communications with the MessageWay server. For example, C:\tmp\MWTrace.txt. Use this parameter only to debug problems interacting with the MessageWay server.

Debugging Parameters

The following parameters are rarely used, and are primarily used during product development.

- MWayConn_ResultsDebugFilename - The full path to a filename that receives a copy of the results file from the MessageWay helper utility. Can include macros. By default, no debug copy of the results file is made.
- MWayConn_ForceAtLeastSeconds - The minimum number of seconds to wait after downloading results from MessageWay, prior to processing the files. This parameter was implemented to allow testing features like progress bars. By default, no additional waiting is done.

Output Parameters

- MWayConn_Report - Contains the last translation exception report file returned from MessageWay. (Typically, only one report file is returned.) If no report files are returned, this parameter is empty if no report files are returned.

  Note: if the report is too long to fit in a task parameter, it might be truncated.

Error handling

This script returns error 5300 if the translation takes place, but returns one of the following error statuses:

- Partially Accepted
- Accepted with Errors
- Rejected

When running a Traditional task, MOVEit Automation regards a script returning error 5300 as a successful process, which, depending on circumstances, might not be the desired behavior.

For maximum flexibility, the best practice is to use this script with Advanced rather than Traditional tasks. This is especially recommended if you set MWayConn_ExceptionsInsteadOfData to True, because in this case, if translation exceptions occur, the output from the task consists of report files rather than data files.

If you want to send report files to a different destination, using an Advanced Task and checking for a process error code of 5300 is recommended.

See also ***Advanced Topics - MessageWay CLI*** (on page 278)

No Op

No Op means "No Operation". No Op is a dummy task that always completes successfully. It is most often used in a task whose role is to delete files. Files that are successfully downloaded by a task with a NoOp process are deleted if the "delete after successful transfer" option has been checked on the source.

No Op does not take any parameters or output any parameters.

Notes

This built-in script can be run per-file or once-after-all-files.

This built-in script can be run as the first step of a task.

Example #1

Jack wants MOVEit Automation to delete any files from a remote directory that end with an `*.tmp` extension.

To perform this operation with MOVEit Automation:

1. Create a new task with a source, process and schedule. (No destination)
2. Check the "Delete Originals After Success Transfer" option and specify `*.tmp` as the file mask on the source.
3. Select "No Op" as the process.

When this task executes, all `*.tmp` files are downloaded from Jack's source into the MOVEit Automation cache. The NoOp process executes, and MOVEit Automation then deletes all the files it just downloaded from the source. No files are retained permanently or transferred to any destination.

PGP Decrypt

PGP Decrypt decrypts a message using PGP. This script is available only if a valid ***MOVEit Automation OpenPGP license*** (on page 170) is available.

This script does not require any PGP key parameters. An encrypted PGP file "self-describes" itself to the point where MOVEit Automation can determine what settings it needs in order to decrypt the file if the appropriate PGP keys have been set up in MOVEit Automation.

For more information, see ***About PGP Keys*** (on page 163).

Input Parameters

- PGPPreserveName - True: The name of the unencrypted file that results from the PGP Decrypt process is based on the preserved name that is also passed in the encrypted PGP file.

  Default is False, because not all PGP packages preserve the original file names.
- PGPPreferredDecryptionKey - If there are a large number of private keys available to MOVEit Automation, set this parameter to the preferred key to use for decrypting files. This key is used first to decrypt files handed to the script. If decryption fails, the rest of the available keys are used in turn to attempt to decrypt the file.
- PGPCheckSignature - Set this to "True" to check the signature (requires that the message be signed). Default is False, because not all PGP encrypted files are also "signed,"

PGP Encrypt and Sign

PGP Encrypt and Sign encrypts and signs a file using PGP. This built-in script is active only if a valid ***MOVEit Automation OpenPGP license*** (on page 170) is available.

Normally, this script should be run as a per-file process. If run as a per-task process, it encrypts the last file added to the cache, which means that this script can be chained with a script like ZipAdvanced.

For more information, see ***About PGP Keys*** (on page 163).

Input Parameters

- PGPRecipientKey (*Required*) - The key of the recipient. A PGP key selection pop-up opens for you to provide this value. Multiple recipients are allowed.
- PGPSignerKey (*Required*) - The private/public key pair (a.k.a., "secret key") to be used for signing. A PGP key selection pop-up opens for you to provide this value.
- PGPASCIIArmor - True for the output file to be ASCII-armored, False for the output file to be binary. Default is False.
- PGPTextMode - True to encode the output file for automatic text mode conversion. Default is False.
- PGPForceV3Sigs - True if the output file should be signed using the old version 3 signatures. This is often required by McAfee E-Business Server. Default is False.
- PGPSigningHash - The hash algorithm to be used for signing. This is often useful for compatibility with out of date PGP clients. Default is SHA1.

For the symmetric encryption algorithm (for example, AES256 or 3DES), MOVEit Automation uses the algorithm that is associated with the first recipient's key.

PGP Encrypt Only

PGP Encrypt Only encrypts (but does not sign) a file using PGP. This built-in script is active only if a valid ***MOVEit Automation OpenPGP license*** (on page 170) is available.

Normally, this script should be run as a per-file process. If run as a per-task process, it encrypts the last file added to the cache, which means that this script can be chained with a script like ZipAdvanced.

For more information, see ***About PGP Keys*** (on page 163).

Input Parameters

- PGPRecipientKey (*Required*) - The key of the recipient. A PGP key selection pop-up opens for you to provide this value. Multiple recipients are allowed.
- PGPASCIIArmor - True if the output file should be ASCII-armored; False for binary. Default is False.
- PGPTextMode - True if the output file should be encoded for automatic text mode conversion. Default is False.

For the symmetric encryption algorithm (for example, AES256 or 3DES), MOVEit Automation uses the algorithm that is associated with the first recipient's key.

Prepend Lines

"Prepend Lines" inserts lines at the beginning of a file. Up to four different lines can be inserted, including blank lines.

Input Parameters

- PrependLines_Line1 (*Required*) - Specifies the first line to insert into the file. This field can contain macros. To specify a blank line, use the string "(blank)".
- PrependLines_Line2 - Specifies the second line to insert into the file. This field can contain macros. To specify a blank line, use the string "(blank)".
- PrependLines_Line3 - Specifies the third line to insert into the file. This field can contain macros. To specify a blank line, use the string "(blank)".
- PrependLines_Line4 - Specifies the fourth line to insert into the file. This field can contain macros. To specify a blank line, use the string "(blank)".

For more information, see about macros, see ***Macros*** (on page 176).

Output Parameters

- PrependLines_LineCount - Returns the number of lines added by this run of PrependLines. (Could be zero or blank if an error occurs.)

Notes

Use this prepend operation only against text files. Do not use it in a binary file, because this could have unforeseen consequences.

This built-in script can be run only one time per file.

This built-in script cannot be run as the first step of a task.

Example #1

Ed wants to insert a note with today's date followed by a blank line into archived log files that are moved from one server to another.

To perform this operation with MOVEit Automation:

1. Create a new task with a source, process, destination, and schedule.
2. Select "Prepend Lines" as the process.
3. Set process parameters:
   - PrependLines_Line1 = "Archived log file processed on [yyyy]-[mm]-[dd]"
   - PrependLines_Line2 = "(blank)"

Report Long Running Tasks

Report Long Running Tasks queries the micstats database and report back information about any tasks that have be running longer than the specified time interval.

Input Parameters

- ReportLongTasks_Time (*Required*) - The number of hours or days a task must have been running to be included in the report.
- ReportLongTasks_Unit (*Required*) - Specifies whether the ReportLongTasks_Time parameter is in hours or days.
- ReportLongTasks_CommandTimeout - The timeout in seconds to use when performing database queries. Default is 180 seconds (3 minutes).

Output Parameters

- ReportLongTasks_Results
  - If one of more tasks are found, a list of tasks that have been running longer than the specified length of time is returned.
  - If no tasks are founds, a summary line stating that "0" tasks were found is returned.
  - If errors occur, a blank string is returned.

Notes

The most common use for the output parameter ReportLongTasks_Results is in a message body for a Send Email step that occurs after the Run Script step for Report Long Running Tasks.

Set Destination

Set Destination changes the host definition, and optionally, the path, of the destination that is used in this task run.

This built-in script is typically used with macros. For more information, see ***Macros*** (on page 176). Three common scenarios in which it may play a role are briefly described below.

- Partial Paths or Filenames - If you have hosts named "FTP1" and "FTP2", you can use a macro like "[LEFT([OrigName], 4)]" in the "SetDestination_Host" parameter against a filename like "FTP2_report025245.dat" to select different hosts at runtime.
- After Look Up - If you have hosts named "Remote Company FTP" and "SSH for Brooklyn", you can use the built-in Look Up script first to determine which host to use based on incoming filenames, file sizes, file paths, etc. and then use a macro like "[Parm:LookUp_Value]" in the "SetDestination_Host" parameter to select different hosts at runtime.
- With MOVEit Automation API - Because MOVEit Automation can pass parameters to any task it runs, you can use a macro like "[Parm:HostFromAPI]" in the "SetDestination_Host" parameter so that the MOVEit Automation API application selects different hosts at runtime.

Input Parameters

- SetDestination_Host (*Required*) - The friendly name of the destination host. The host you select does not need to be currently defined in the task, nor does it need to be of the same type. For example, you can use this parameter to switch the destination to a MOVEit Transfer server even if the task's destination is currently a single FTP server.

  However, if you use this parameter to switch between hosts of different types (such as between FTPS and MOVEit Transfer servers) you MUST also use the "SetDestination_Path" parameter to ensure that destination paths are properly parsed. To indicate a Windows file system host, use a value of "(default)"; in this case begin the "SetDestination_Path" value with a drive letter (`C:\`) or a UNC (`\\server\share\`).
- SetDestination_Path (*Optional*) - Sets the Destination path. Ignored if blank. Macros are allowed and often used in this field.
- SetDestination_IgnoreError (*Required*) - Specifies whether to ignore errors in setting host and path.

  Set the value to ON only if you have a fallback destination configured.

Notes

This built-in script can be run per-file or once-after-all-files.

This built-in script can be run as the first step of a task.

Sleep

Sleep pauses the current task between file transfers for a specified number of seconds or milliseconds.

Input Parameters

- SleepScript_Time (*Required*) - The number of seconds or milliseconds for the script to sleep. Maximum is 16000.
- SleepScript_Unit (*Required*) - Specifies whether to count off time in milliseconds or seconds. Default is seconds.

Notes

This built-in script can be run per-file or once-after-all-files.

This built-in script can be run as the first step of a task.

Example #1

Kara has noticed that a particular application on a remote server has performance issues when MOVEit Automation uploads files to it as fast as MOVEit Automation can. Kara wants to introduce an artificial pause into the MOVEit Automation file transfer task that points to the destination so that the remote application has time to process each file. After some trial-and-error, Kara has decided that a 5 second pause usually clears up the issues with the remote application.

To make this happen in the MOVEit Automation task:

1. Select the existing task, which already has a source, destination and schedule.
2. Add a new process: "Sleep", and make sure the process is set to run "per-file".
3. Set process parameters:
   - SleepScript_Unit = "seconds"
   - SleepScript_Time = "5"

SMIME Receive

SMIME Receive retrieves messages from a POP3 server, decrypts them if necessary using a certificate in the Windows certificate store, and adds any attachments to the list of files to be processed.

Input Parameters

- SMIME_POPAddress (*Required*) - The address of the POP3 server. Example: mail.mycompany.com.
- SMIME_Username (*Required*) - The username for accessing the above POP3 server. Example: joeuser.
- SMIME_Password - The password for POP3 user account.
- SMIME_DelWhenDone - Whether to delete messages when finished processing them. Default is True.
- SMIME_DelMsgWOAttach - Whether to delete messages found on the POP3 account that do not contain attachments. If SMIME_DelWhenDone is True, messages will be deleted anyway. Default is False

If the script fails to run and returns an error, it is typically due to a configuration error. See the list of error codes to diagnose the problem.

Error Code	Meaning
500	A required task parameter was not found.
501	Could not access POP3 server.
502	Error while retrieving a message.
503	Error decrypting a message.
504	Error verifying a message signature.
505	Error saving attachment file.

SMIME Send

SMIME Send encrypts and/or signs files, and sends them as email using an SMTP server.

Input Parameters

- SMIME_SMTPAddress (*Required*) - The address of the SMTP server. Example: mail.mycompany.com.
- SMIME_Sender (*Required*) - The email address of the sender. Example: joe@mycompany.com.
- SMIME_Recipient - The email address of the recipient. Example: mary@shinythings.com.
- SMIME_Subject - Email message subject text. Default: "File from MOVEit Automation is attached".
- SMIME_Body - Email message body text. Default: "This message should have come with a file attached by MOVEit Automation.".
- SMIME_RecipientCert - The subject of the recipient's certificate. Default: recipient email address.
- SMIME_SenderCert - The subject of the sender's certificate. Default: sender email address.
- SMIME_Sign - Whether the message should be cryptographically signed. Default: True.
- SMIME_Encrypt - Whether the message should be encrypted. Default: True.

If the script fails to run and returns an error, it is typically caused by a configuration error. See the list of error codes to diagnose the problem.

Error Code	Meaning
500	A required task parameter was not found.
501	File attachment failure.
502	Could not find a certificate matching the provided subject(s).
503	Error sending the email message to the specified SMTP server.

Tamper Detect

Tamper Detect performs ***tamper detection*** (on page 301) on the three main audit tables in MOVEit Automation. It is used by the built-in task "Tamper Detect". This script is used only for the Tamper Detect task.

Tamper Detect sends email to administrators if it suspects tampering. A daily report of its activity is in the most current "Audit" view as well as in the "Task Runs" and "File Activity" views.

Input Parameters

- TamperCheck_EmailOperator - Specifies when the script emails a tamper detection report to the email address configured in Errors tab of the MOVEit Automation configuration utility. Values:
  - Never - Never email the report.
  - OnError (default) - Email the report only if errors are detected.
  - Always - Always email the report.

Notes

Run this built-in script as the first step of a task.

The report generated by this script is an ASCII file or message that is typically about 30 lines long. The first line of the message indicates whether any errors were found.

Trim Statistics DB

Trim Statistics DB deletes all MOVEit Automation audit records older than a certain number of days, and optionally saves them to tab-separated files.

By default, MOVEit Automation has a pre-installed task that uses this script with parameters set to specific values. If you want to create your own task using this script, note the required parameters.

Input Parameters

- DirLog (*Required*) - The folder into which to write the debug and error logs. In the pre-installed task, this parameter is set to `\Program Files\MOVEit\MICStats\Logs`. If you do not want debug and error logs, set this value to blank.
- DirArchive - The optional folder into which to write the tab-delimited archives. In the pre-installed task, this parameter is set to `\Program Files\MOVEit\MICStats`. If you do not want deleted records archived to a text file, set this to an empty value.
- TrimStatsDB_DaysToRetainArchive - The number of days to retain the tab-delimited archive files. Log files older than this number of days that are found in "`\Program Files\MOVEit\MICStats\Logs`" (or the value of DirLog) are deleted. Use this setting carefully, because deleted files *are not* put into the Recycle Bin.
- DSNArchive - The optional ODBC DSN (data source name) into which to save a copy of the records being deleted. In the pre-installed task, this parameter is set to an empty string, which means that the delete database records are not written to another database.
- DaysToKeep (*Required*) - The number of days to retain records. In the pre-installed task, this parameter is set to 40.

  This parameter applies to the Task Runs and File/Folder Activity tables in the database,
- DaysToKeepNoXfers - The number of days to retain TaskRuns records marked "No Xfers". If not supplied, no special action is taken. If you have many tasks that run frequently, you can set this parameter to a small number of days in order to reduce the size of the database and increase its performance. The default is empty, which means no special processing is done for these records.
- TrimStatsDB_ArchiveNoXfers - Specifies whether to archive or discard TaskRuns records that are marked "No Xfers". Default is No, meaning that records are discarded.
- TrimStatsDB_DaysToKeepAudit - The number of days to retain Audit records. This parameter applies only to the Audit table in the database. If no value is specified, the value that was applied to DaysToKeep is used.
- TrimStatsDB_DateFormat - The date format that is in use. Usually only used on Windows systems outside the United States. The legal values are MonthDayYear (default) and DayMonthYear.
- TrimStatsDB_CommandTimeout - The number of seconds a client connection to the database remain actives without timing out. Default is 180 seconds (3 minutes).

Notes

Use this built-in script as the only step of a task. Process-level per-file or once-after-all-files settings will be ignored.

Over time, the statistics database can accumulate many records, slowing performance and using excessive amounts of disk space. To prevent this, MOVEit Automation is typically configured to run this built-in script periodically, optionally saving them to a file or another database before deletion.

Example #1

Bob has noticed that the various task history dialogs in MOVEit Automation have been sluggish lately and asks Ipswitch for some advice. Ipswitch notices that Bob has more than two hundred tasks that poll FTP servers every five minutes, all day long. Most (98%) of these polls do not yield files, but MOVEit Automation continues to log this information during each poll. Over the course of a single day, this means that more than 55,000 "no xfer" actions are logged to the database. Bob knows that "no xfer" information can be of value in debugging, but Ipswitch would prefer that he remove them after a few days and only retain "success" or "failure" audit entries for long periods of time. Bob and Ipswitch decide to retain one week of "no xfer" records, and to keep 60 days of "success" and "failure" logs in the database. The "no xfer" records will not be saved to the longer term archive logs.

To accomplish this with MOVEit Automation:

1. Select the existing "Trim Stats DB" task, which already has a process and schedule.
2. Set process parameters:
   - DaysToKeep = "60"
   - *(Add)* DaysToKeepNoXfers = "7"

Unzip Advanced

Unzip Advanced unzips an single archive containing one or more files, with an optional ZIP password. It can work with archives that contain nested folders and can also handle the "bzip2" encryption type used by the WinZip 9.0 file compression utility.

If you need to decrypt entries in the zip file, use the Unzip Advanced script, instead of the setting **Expand zip files** (in Web Admin) or **Expand compressed (zip) files** (Admin Console) when defining a Source.

Input Parameters

§ **UnzipAdvanced_Password** - Specifies an optional password to apply to the Zip file. Default value is blank, which indicates no decryption is to be done.

CAUTION: Using a password to protect a Zip archive is not usually enough protection to thwart a determined hacker.

Output Parameters

§ **ZipFileSize** - Number of bytes in the zip file. This and other size parameters are populated whether or not the Zip file is successfully unzipped. If more than one zip file is encountered, this parameter contains only the size of the last zip file processed by this task run.
§ **ZipFileSizeKB** - Number of kilobytes (KB) in the zip file.
§ **ZipFileSizeMB** - Number of megabytes (MB) in the zip file.
§ **ZipFileSizeGB** - Number of gigabytes (GB) in the zip file.

Notes

This built-in script must be run as a once-after-all-files process.

This built-in script cannot be run as the first step of a task.

The task's "Cache Files" option must be set to "Use Random Names".

XSL Transform

XSL Transform transforms XML documents downloaded from a source using a specific XSL stylesheet.

Input Parameters

- XSLTransform_XSLPath (*Required*) - Specifies the path that MOVEit Automation uses to obtain the XSL stylesheet.
- XSLTransform_XSLFile (*Required*) - Specifies the name of the XSL stylesheet file.

Notes

This built-in script must be run per-file (not once-after-all-files).

This built-in script cannot be run as the first step of a task.

Example #1

Xavier wants to automate the XSL transformation of several XML files. In this instance, he wants to transform incoming XML files using an XSL template called "example.xsl". This XSL template is located in the "d:\projects\templates" folder on MOVEit Automation.

To integrate this application with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Select "XSL Transform" as the process.
3. Set process parameters:
   - XSLTransform_XSLPath = "d:\projects\templates"
   - XSLTransform_XSLFile = "example.xsl"

To use MOVEit Automation to perform other XSL transformations, consider setting "XSLTransform_XSLPath" as a global parameter so that it does not need to be set on every task.

Zip Advanced

Zip Advanced zips one or more files into a single archive with a configurable level of compression and an optional ZIP password.

Use Zip Advanced instead of the destination-level "Zip" checkbox if any of the following zip features are needed.

- Password - Although not very secure, Zip Advanced can apply a password to a zip archive.
- Multiple Files - The destination-level Zip checkbox will zip each file into its own individual zip archive. If you need to zip multiple files together instead, use Zip Advanced as a "run once after all downloads" process instead.
- Specific Zip Filename - To determine the zip filename, the destination-level Zip checkbox strips any given file extension and appends `.zip` to the filename. If you need to preserve the entire original filename in the name of the zip archive or want to substitute it with something else, use the Zip Advanced process and apply the name you want in the appropriate destination configuration.

Input Parameters

- ZipAdvanced_Password - An optional password to apply to the Zip file. Default is blank, which indicates no password is to set on the Zip archive.

  CAUTION: Using a password to protect a Zip archive is not usually enough protection to thwart a determined hacker.

- ZipAdvanced_Compression - Specifies the level of compression. More compression takes more time. Values: Normal (default) None, Low and High. All levels involve vendor-neutral Zip compression standards; you are unlikely to encounter incompatibilities with any level.

Output Parameters

- ZipFileSize - Number of bytes in the zip file. This and other size parameters are populated only if the Zip file is successfully created.
- ZipFileSizeKB - Number of kilobytes (KB) in the zip file.
- ZipFileSizeMB - Number of megabytes (MB) in the zip file.
- ZipFileSizeGB - Number of gigabytes (GB) in the zip file.

Notes

Zip Advanced recursively includes subfolders in its archives if the "Include Subfolders" option has been set on any task sources.

Zip Advanced requires that the task-level "Cache Files" option be set to "Use Original Names". To set this option:

- In the Admin Console right-click a task name and select Edit Task Info...
- In Web Admin, click TASKS, click a task name, expand the Advanced properties and click Edit.

This built-in script can be run per-file or once-after-all-files.

This built-in script cannot be run as the first step of a task.

Example #1

Al wants to download several `*.rpt` files from an FTP server, include a `readme.txt` from his local computer, and zip the entire package together in a highly compressed Zip archive.

To accomplish this with MOVEit Automation:

1. Create a new task with a source, process, destination and schedule.
2. Point the new source to the FTP server and use `*.rpt` as the file mask.
3. Select "Zip Advanced" as the process.
4. Set process parameters:
   - ZipAdvanced_Compression = "High"
   - Run = "Once After All Downloads"
5. Add a second source to the task and point it at the local "readme.txt" file.

The final task has:

§ Two sources: one FTP server and one local,
§ The Zip Advanced process (set to run after material is downloaded from all sources), and
§ Any destination that you specify.

Custom Scripts

The custom scripts feature is available with MOVEit Automation Enterprise.

A custom script can call any function defined by VBScript. It can also instantiate and invoke COM objects, such as `Scripting.FileSystemObject` and command-line applications using the `MIRunCommand` function.

After you save a custom script in MOVEit Automation, the script can be used in a process. Custom scripts are stored encrypted in the MOVEit Automation configuration file.

To...	Do this:
Add (create) a new custom script	**1** Click SCRIPTS > Add Script. Provide a name and optional description. **2** Select a script template. In the Script Source box, make changes as needed. Save. For more information, see ***Add New Script*** (on page 139).
Import a `.vbs` script	**1** Click SCRIPTS > Add Script. In the Add New Script dialog box, click Import Script File. Locate the script and click Open. **2** In the Script Source box, make changes as needed. Save. The original (external) script is not affected by changes you make to the script in Web Admin. NOTE: If you have a custom script open in Web Admin, and you import an additional script, the source code of the new script overwrites the source code in the Script Source box. You do not receive a warning message.

Export a custom script	**1** Add a new script or click an existing custom script name. **2** On the script properties page, click Export Script File. The script is exported as a `.vbs` file using the friendly name as the filename. Example: `myscript.vbs`
Edit a custom script	**1** Click SCRIPTS and click the script name. The properties page for the script opens. **2** Click Edit. Make changes and Save. NOTE: If you change the friendly name of a script and save, that same script is saved with the new friendly name. To use an existing script as the starting point for a new script, see Use an existing custom script to create a new script in the last row of this table.
Edit a custom script in an external editor	**1** Click SCRIPTS and click the script name. On the script properties page, click Export Script File. **2** Using the external editor, make changes in the exported file and save. **3** In Web Admin, click the script name and click Edit. Click Import Script File and select the file. The imported file overwrites the Script Source. Save.
Use an existing custom script to create a new script	Use this procedure to retain an existing script in MOVEit Automation and create a new script based on it. **1** Add a custom script, save it, and then export it. **2** Add a new custom script and give it a Friendly Name. **3** Import the script that you exported (in Step 1). Make changes as needed, and save.

See also:

§ MOVEit-specific Functions and Subroutines
§ ***Custom Script Samples*** (on page 147)

Add New Script

This custom scripts feature is available only in MOVEit Automation Enterprise. For more information about editing, importing, and exporting custom scripts, see ***Custom Scripts*** (on page 137).

To access this dialog box: Select SCRIPTS > Add Script.

You can add/write a new script or import an existing VBScript (`*.vbs`) into MOVEit Automation.

Add/Edit Script Field	Description
Friendly Name	Name that appears in the list of scripts on the SCRIPTS page. NOTE: If you edit an existing script and change its Friendly Name, the edited script is saved with the new Friendly Name.
Description	Optional description
For Each Line Template	Click to open a template in the Script Source box for a script that runs after each file in the associated task is processed.
All At Once Template	Click to open a template in the Script Source box for a script that runs after all the files in the associated task are processed.
Script Source	Script source.
Import Script File	Click to import an existing VBScript file.(`*.vbs`) NOTE: Any code in the Script Source field is overwritten by the source code of the script that you import. You do not receive a warning message.
Export Script File	Click to export the script you created in this dialog box. The script is downloaded as a `.vbs` with the name you provided in the Friendly Name field. For example, if you download a script that has a friendly name of My Test Script, the file `My Test Script.vbs` is downloaded.

Syntax - Custom Scripts

MOVEit Automation custom scripts use Microsoft VBScript. In addition to the basic functions provided by that environment, MOVEit Automation also makes available several application-specific functions.

Functions and Subroutines

The functions and subroutines, and associated details are listed in alphabetical order in the following table.

The functions and subroutines are described as follows:

- The function name is displayed in **bold** text.
- The return value of the function, if applicable, precedes the function name.
- The function arguments follow the function name. Function calls with a return value require the function arguments to be surrounded by parenthesis, while function calls without a return value require the parenthesis to be omitted. Optional arguments are surrounded by brackets.

For example, `errcode = MISetDestHost`(ConfiguredHostName [,idest])

`Where, MISetDestHost` is the function name

`errcode` is the return value

`ConfiguredHostName` is the required argument

`idest` is an optional argument.

Functions and Subroutines	Description
MIAddFile CacheFilename, AssignedFilename	Adds a file to the list of files to be sent to destinations. Note: Before you use MIAddFile, call MINewCacheFilename() to create the temporary file to send. CacheFilename - The name of the temporary file to send AssignedFilename - The name to be given to the file on the destination server. The filename can be specified with forward slash (/) or backslash (\) characters, in which case the name is considered a folder path and filename relative to the destination path. To prevent infinite loops, scripts are not currently run on files added via MIAddFile.
DirName = **MICacheDir**()	Returns the full path of the directory where MOVEit Automation keeps its temporary files
FileName = **MICacheFilename**()	Returns the name of the temporary copy of the file. This file can be overwritten if the custom script needs to change the contents of the file before it is transferred.

Functions and Subroutines	Description
FilesString = MICacheFiles()	Returns a string containing the subdirectories and filenames for all active files in the cache. These names are relative to MICacheDir and reflect the actual names on disk in the cache directory, not the original names from the sources. The names are separated by the pipe (|) character. The last file does not have a | at the end. This list is empty immediately after a call to MIIgnoreFiles. The list does not include, for instance, the names of zip files that have been downloaded from sources marked "Uncompress Archives".
bOK = MIDeleteFileSecure(Filename)	Overwrites the specified file with random bytes, and then deletes the file. Returns True if successful.
MIDirAddEntry FilenameToMatch, Date, Size, bIsDir, FilenameForGet, FilenameOriginal	Adds an entry to the FTP or SSH directory listing that is being parsed. This should be called only from custom directory parsing scripts. For more information, see ***Custom Directory Parsing*** (on page 245)
sDirListing = MIDirGetListing()	Returns the entire verbatim listing from the FTP or SSH server. Call this function only from custom directory parsing scripts. For more information, see ***Custom Directory Parsing*** (on page 245).
DbgLevel = MIGetDebugLevel()	Returns the debug level currently set in MOVEit Automation. Values: Level — Meaning 0 — Internal errors 10 — Task/File Errors 20 — Task/File Warnings 30 — Task Completions 40 — File Completions 50 — Some Debug 60 — More Debug 70 — All Debug
OriginalFilename=MIGetOriginalFilename()	Returns the original filename of the file.
Result = MIGetTaskInfo(InfoString)	Returns the specified information about the current task. Values: Info String — Value returned by MIGetTaskInfo "CacheUsesOriginalNames" — True if the task is configured to use original filenames in the cache directory False if the task is configured to use random filenames. "NSources" — The number of sources in the task. "ProcessIsPerFile" — True if the current process is run for each file False if the current process is run one time after all downloads. "ShouldStop" — True if an operator has requested that the task be stopped.

Functions and Subroutines	Description
ParamValue = MIGetTaskParam(ParamName)	Returns the value of the specified task parameter. If the current task has no such parameter, the empty string is returned.
MIIgnoreFiles [bDeleteOrigIfCfg [,bKeepAsNew]]	Causes all files already downloaded or added via MIAddFile to be ignored in subsequent processing steps. For example, use this if your custom script creates a zip file containing the downloaded files, and you do not want the downloaded files to be sent. **Parameter** — **Meaning** bDeleteOrigIfCfg — Whether to delete the original files if the Delete Original File(s) After Successful Transfer option has been set in the source. Default is True. bKeepAsNew — Whether to continue to regard these files as new in the next task run even if the task succeeds. This is meaningful only if Collect Only New Files is set in the source. Default is False.
MIIgnoreThisFile [bDeleteOrigIfCfg [,bKeepAsNew]]	Causes the current file to be ignored in subsequent processing steps. For example, call this method if you want your custom script to ignore zero-length files. When called without parameters, this method is equivalent to calling MISetErrorCode 5000. Call this method only in per-file processes. Parameter — Meaning bDeleteOrigIfCfg — Whether to delete the original file if the Delete Original File(s) After Successful Transfer option has been set in the source. Default is False. bKeepAsNew — Whether to continue to regard this file as new in the next task run even if the task succeeds. This is meaningful only if Collect Only New Files is set in the source. Default is False.
MILogMsg Message	Logs a message to the debug window and debug log file. The message is preceded by the task name.
MyString = MIMacro(MacroText)	Evaluates the macro MacroText and returns the resulting text. MacroText can contain any combination of ***macros*** (on page 176).
FileName = MINewCacheFilename()	Returns a new, unique temporary filename. The filename will be a full path and will be in the cache folder.

Functions and Subroutines	Description
bOK = **MIReplaceCacheFile**(Filename)	Securely deletes the current temporary cache file, and replaces it with the contents of Filename. For example, use this to send a different file, such as a file that you just created, instead of the downloaded file. Typically you first call MINewCacheFilename() to get a filename, use the FileSystemObject to create the file based on the contents of the file in the cache, and then call MIReplaceCacheFile() to tell MOVEit Automation to use the new file
retval = **MIRunCommand**(Command)	Runs a Windows system command through the Windows system command interpreter. Command is a command, such as "Notepad MyFile.txt". This can be the name of a command, such as an .exe or .bat file, or the name of a file with an associated extension, such as .vbs, or a built-in CMD.EXE command such as DIR. MIRunCommand waits until the command is complete. If you want the custom script to continue running while the command runs, use the Windows START command to launch the program. MIRunCommand returns either: § A long integer: -1 if the program could not be found, or § The return code from the program. Programs typically return 0 upon success, or >0 upon failure. MIRunCommand does not work well with pathnames that contain spaces. To avoid these situations, use the Scripting.FileSystemObject to look up safe versions of paths before passing them to MIRunCommand. For example: Set fso = CreateObject("Scripting.FileSystemObject") AppPath = "C:\Program Files\My App" AppExe = AppPath & "\" & "runit.exe" SafeAppPath = fso.GetFolder(AppPath).ShortPath SafeAppExe = fso.GetFile(AppExe).ShortPath `ErrorCode = MIRunCommand(SafeAppExe & " " & " -o " & SafeAppPath & "\out.txt")`

Functions and Subroutines	Description
errcode = MISetDestHost(ConfiguredHostName [,idest])	Changes the current file's view of one of the task's destinations so that it points to the given configured host. ConfiguredHostName is the name of one of the configured hosts such as "XYZ Corp FTP Server", not an Internet host domain name. The new host does not need to be the same type as the destination being changed. For example, the destination can be of type FileSystem, but you can change it to point to an FTP server. Only the current file being processed (if any), and any files added via MIAddFile during this run of the process, are affected. If you are using this command to switch between hosts of different types (for example, between FTPS and MOVEit Transfer servers) use the "MISetDestPath" command to ensure that destination paths are properly parsed. Use a value of "`(default)`" to indicate a Windows file system host; in this case your path value should either begin with a drive letter (for example, `C:\`) or a UNC (for example, `\\server\share\`). idest is the ordinal number of the destination to change. Default is 1, which means the first destination in the task. (Most tasks have only one destination.) Returns an error code: 0 indicates success 2540 indicates that the named host does not exist `2850 indicates that idest is out of range.`
errcode = MISetDestPath(NewPath [,idest])	Changes the current file's view of one of the task's destinations so that it points to a different path. NewPath is the new path or directory name. idest has the same meaning that it does for MISetDestHost. Only the current file being processed (if any), and any files added via MIAddFile during this run of the process, are affected. Returns an error code: 0 indicates success `2850 indicates that idest is out of range`

Functions and Subroutines	Description
MISetErrorCode NumericErrorCode	Sets the error code and error description for this process. If the custom script determines that the file should not be transferred, it calls MISetErrorCode with a non-zero numeric error code and MISetErrorDescription with a textual description of the error. This information is recorded by MOVEit Automation, and the file is not sent. If the special value 5000 is set by MISetErrorCode, MOVEit Automation ignores the file. That is, it does not send the file, does not delete the original file, and does not flag an error that would cause the task to fail. Returning error code 5000 can be used to ignore unwanted files without alarming operators by having the task marked as unsuccessful. See also **MIIgnoreThisFile** If the special value 5010 is set by MISetErrorCode, MOVEit Automation does not count this process as having been run. Ordinarily, MOVEit Automation marks a task has having completed successfully (as opposed to "No actions taken") if any process runs, even if no files were downloaded or uploaded. However, if all processes in a task call MISetErrorCode 5010, and no files are transferred, then that task run is considered to have completed with "No actions taken". This affects which Next Actions are executed at the end of the task
MISetErrorDescription ErrorDescription	Sets the error code and error description for this process. If the custom script determines that the file should not be transferred, it calls MISetErrorCode with a non-zero numeric error code and MISetErrorDescription with a textual description of the error. This information is recorded by MOVEit Automation, and the file is not sent. If the special value 5000 is set by MISetErrorCode, MOVEit Automation ignores the file. That is, it does not send the file, does not delete the original file, and does not flag an error that would cause the task to fail. Returning error code 5000 can be used to ignore unwanted files without alarming operators by having the task marked as unsuccessful. See also **MIIgnoreThisFile** If the special value 5010 is set by MISetErrorCode, MOVEit Automation does not count this process as having been run. Ordinarily, MOVEit Automation marks a task has having completed successfully (as opposed to "No actions taken") if any process runs, even if no files were downloaded or uploaded. However, if all processes in a task call MISetErrorCode 5010, and no files are transferred, then that task run is considered to have completed with "No actions taken". This affects which Next Actions are executed at the end of the task
MISetFilename NewFilename	Changes the name under which the file should be stored at the destination. Do not include a path in NewFilename.
MISetStatus StatusText	Set the status text to be displayed by MOVEit Automation Admin while the script is running. This can be used by long-running scripts to inform the operator the status of the process. If MISetStatus is not called, the message "Running script *scriptname*" is displayed.

Functions and Subroutines	Description
MISetTaskParam ParamName,ParamValue	Sets the value of the specified task parameter ParamName. If the current task has no such parameter, a parameter of that name is created. Parameter names are not case-sensitive.
MISleep Milliseconds	Suspends the custom script for the specified number of milliseconds. Very little processor time is consumed during the pause.

ErrCode = MIStartTask(TaskNameOrID [,TaskParams])	Starts the specified task. TaskNameOrID is either a task name or a task ID. (MOVEit Automation checks first for a task name match.)
	TaskParams allows you to specify values for task parameters. If a parameter name matches the name of a parameter configured in the task definition, the new value overwrites the configured value.
	The format of TaskParams is `ParamName1=ParamVal1|ParamName2=ParamVal2|...` The trailing | at the end of the last task parameter is optional.
	Example 1: Run a task called ShowMyID that shows the value of the ID stored in variable `strID`.
	`Result = MIStartTask("ShowMyID", "ID is: "&strID)`
	Example 2: A task called SendSummaryFile is started with two task parameters computed from within the script: `ID` and `CheckNum`.
	`blnResult = MIStartTask("SendSummaryFile", "ID=" & strID & "|CheckNum=" & intCheckNum)`
	Returns 0 if the task was started. Note: the task will probably still be running when this function returns; a 0 return code does not mean that the task will successfully run to completion

TaskGroups = MITaskGroups()	Returns the name of the task that is running the custom script.
TaskName = MITaskname()	Returns the names of the task groups to which the task belongs. This is a string containing the names of all applicable groups, separated by |. For example, the result might be "Daily|For Argus Bank|By Fred", or just "". You can process this in VBScript by creating an array from it using the Split function.

Custom Script Samples

Some sample scripts are for example only, others are generalized versions of production scripts that are used by several MOVEit Automation data center customers.

For more information, see ***Sample Scripts*** (on page 147) and ***Production Scripts*** (on page 147).

Where to obtain sample scripts

- MOVEit Automation Admin Console - The installation program for MOVEit Automation Admin installs sample scripts into a sample scripts subfolder.
- Web Admin - If you are using Web Admin, scripts are available from the ***Customer Portal*** (***https://ipswitchft.secure.force.com/cp/***). Log in to the portal and click My Products.

For basic examples of scripts that read in and process files, in Web Admin, select SCRIPTS > Add Script, and create a simple All at Once or For Each Line example.

Each script runs in a separate thread, in the context of the task with which it is associated, so errors in one script will not affect MOVEit Automation or the other tasks MOVEit Automation is working on.

Example Scripts

- CvtCatalog.vbs - Performs simple XML processing against a file read line-by-line.
- Custom Errors.vbs - Example of how to generate custom error codes and messages from a script.
- FixLen Record In.vbs - Reads in a file that has fixed-length lines, parses it, totals a few columns of numbers, and generates an exception report. Built to work with the files created by `FixLen Record Out.vbs`
- FixLen Record Out.vbs - Randomly generates fixed length data sets to be consumed by `FixLen Record Out.vbs`
- Reverse File.vbs - Reads in a file and writes out the complete contents backwards. (for example, `bluefox` becomes `xofeulb`)
- StripLF.vbs - Removes newlines from a file.
- Task Groups.vbs - Overwrites a file with the name of the task that is running, and the list of task groups the task belongs to.

Production Scripts

- Clean IIS Web Logs.vbs - Strips local addresses and non-interesting files out of IIS web server logs. This script contains a list of the file types (.gif, .jpg, etc) to strip from the log. Ipswitch uses this script to pre-process the logs from its marketing sites.
- CleanupFolder.vbs - This script can be used only on the local Windows file system.

  Searches through a folder and deletes any files that were last modified more than the number of days you specify with fileAgeDays, and any empty folders, regardless of age.

  To delete remote files, use a No Op task and configure a filter to download only files older than n days. (For an example, see ***Configuring Tasks - Processes/Scripts - Built-In - No Op*** (see "***No Op***" on page 125).

  Parameters used by this script:

  - folderPath - The path to the folder to cleanup.
  - fileAgeDays - Files older than (this value) days are deleted.

  - § includeSubfolders - If true, recurse through subfolders. Default is false.
  - § deleteEmptyFolders - If true, deletes any folders that are empty. Default is false.
  - § logToFile - If true, a file named CleanupFolder_yyyy-mm-dd.log is created in the `folderPath` directory. This log file records each file considered and whether it was deleted. Default is false.
- § FileSize.vbs - Obtains the file size from a single transferred file. This can be used in conjunction with Next Actions. Works with only one file or the last file transferred.
- § IgnoreSmallFile.vbs - Checks the size of each file processed against a minimum file size specified by the `MinSize` parameter. Any files smaller than the minimum are ignored.
- § Kick Off Task.vbs - Reads in from a file a "shared secret", a list of task names, and runs all the named tasks. Used by a few large companies who like to start tasks from a mainframe process. In these cases the source is a file on the mainframe's FTP server marked to be deleted after successful transfer. There is no destination.
- § OrigNames.vbs - Compiles a complete list of all the files used by task into a comma-delimited list. Used by people who want a complete list of all files processed by a task that is sent to them in Next Action emails. (The Next Action message body contains the macro `[Parm:OrigNames]`)
- § Ping.vbs - Pings several remote hosts to make sure they are still responding. Generates an error message if one or more of these hosts goes down. This script is used in a task with a single process and one or more Next Actions.
- § PKZipWithPass.vbs - Uses PKZip to compress a file and secure the compressed file with a password. Requires the PKZip application from PKWare.
- § PurgeStats.vbs - Purges all entries that are older than a specified limit from the statistics database. (Note: this sample script was reimplemented as the built-in Trim Statistics DB script.)
- § Run DOS Command.vbs - Executes a command specified by a task parameter, with arguments also specified by task parameters. The command would normally be a task that is run at the command line, such as `zip.exe` or `copy`. (Note: The functions of this script are also available in the built-in Command Line App script.)
- § TrimStatsDB.vbs - Goes through the MOVEit Automation statistics database and purges entries older than X days. Purged entries can be deleted, copied to text files, or sent to another database. This script is run daily by MOVEit Automation users who process large numbers of files, including most datacenters. (Note: this sample script was reimplemented as the built-in Trim Statistics DB script.) For more information, see ***Trimming the database*** (see "***Trimming***" on page 301).
- § WordCount.vbs - Counts the number of times a particular word appears in source files. The word to search for is configured in a task parameter. The word count, a brief report, and other information is written back into other task parameters. Note: The functions of this script are also available in the built-in script Find Or Replace.
- § ZipAllFiles.vbs - Calls a command-line zip utility to zip up a collection of downloaded files and replace them with the zip file, so that only the zip file is sent to the task destinations. You can use this script with the Run Process Once and Use Original Names for Cache Files features to zip an arbitrary collection of files and folders in one task, compared with two or more tasks for previous methods. Note: The functions of this script are also available in the built-in script ZipAdvanced.
- § ZipDir.vbs - Calls a command-line zip utility to zip a local folder and its contents. Adds the zip file to the list of sources to be processed to destinations. Note: The functions of this script are also available in the built-in script ZipAdvanced.
- § ZipExe.vbs - Uses the MOVEit Automation command-line facility to demonstrate how to zip multiple source files into a single ZIP archive. Note: The functions of this script are also available in the built-in script ZipAdvanced.

REPORTS

Reports are used to provide access to current status and recorded statistics in a convenient report format. Report data is refreshed periodically. The following three different report display types are available.

- Task Run: For each task, the Task Run report displays the start time, the duration in the format minutes:seconds, result, numbers of files and bytes sent, and how task was started.
- File Activity: The File Activity report lists file activities with information such as log time, task name, activity type, source and destination information, and results. Each activity of a task is listed on a separate row.
- Audit: The Audit report includes information about commands and configuration changes in MOVEit Automation, and displays the Log Time, Action, Target Type, Target Name, Result, Username and IP Address.

See also:

- ***Navigating Web Admin*** (on page 9)
- ***Filtering and Sorting*** (on page 12)

Task Run

REPORTS > Task Run

For each task, the Task Run report details the start time, task name, duration in the format minutes:seconds, result, numbers of files and bytes sent, and how task was started.

Items that appear in the list depend on how you accessed the Task Run page.

How did you access the Task Run page?	The Task Run page contains
From the Task List page, for a single task, selected the action View Run History.	Information for each run of the selected task.
From the Task List page, set optional filters. Clicked REPORTS > Task Run.	Information for each run of every task listed in the Task List, after filters are applied.

Task List Filtering

Filters applied on the Task List page are automatically applied to the Task Run, File Activity, and Audit pages. You can remove or edit the applied filters.

- To remove the automatically applied filters, clear the **Apply Task List filter(s)** check box above the applied filters in the **Search & Filters** panel. To reapply the task list filters select the **Apply Task List filter(s)** check box.
- To edit the automatically applied filters, click **Edit** the **Apply Task List filter(s)** in the **Search & Filters** panel. You are redirected to the Task List page. Edit the selected filters on the Task List page. For more information, see ***Filtering and Sorting*** (on page 12).

More Options

The more options menu for each task provides options to explore specific information for this run or this task.

- *For this run*
  - View File Activity. Opens and filters the File Activity page for the selected run.
  - View Run Details. Opens the Task Run Detail pane for the selected run.
- *For this task*
  - View Task Config. Opens the individual task configuration
  - View Run History. Filters the Task Run page for the selected task.
  - View File Activity. Opens and filters the file activity for the individual task.

Ø *To view Task Run information for other tasks:*

Go back to the Task List page and do either of the following:

- Set different filters, then click **REPORTS** > **Task Run**.
- For a different task, select the action **View Run History**.

Ø *To pause, resume, or refresh the current page:*

1. Click the more options icon above the table, and select one of the following options.
   - To pause the page refresh, click **Pause Page Refresh**.
   - To resume the page refresh, click **Refresh Page Now**.
   - To refresh the page, click **Refresh Page Now**.

See also:

- Task List
- File Activity
- About Tasks

File Activity

REPORTS > File Activity

The File Activity report details information such as log time, task name, activity type, source and destination information, and results. Each activity of a task is listed on a separate row.

Items that appear in the list depend on how you accessed the File Activity page. To further narrow the list, make filter selections in the left pane.

How did you access the File Activity page?	The File Activity page contains
From the Task List page, selected a single task and selected the action **View File Activity**.	Information for each file that was processed by the single task.
From the Task List page, set optional filters. Clicked **REPORTS > File Activity**.	Information for each run of every task listed in the Task List, after filters are applied.

Task List Filtering

Filters applied on the Task List page are automatically applied to the Task Run, File Activity, and Audit pages. You can remove or edit the applied filters.

- To remove the automatically applied filters, clear the **Apply Task List filter(s)** check box above the applied filters in the **Search & Filters** panel. To reapply the task list filters select the **Apply Task List filter(s)** check box.
- To edit the automatically applied filters, click **Edit** the **Apply Task List filter(s)** in the **Search & Filters** panel. You are redirected to the Task List page. Edit the selected filters on the Task List page. For more information, see ***Filtering and Sorting*** (on page 12).

More Options

The more options menu for each task provides options to explore specific information for this file activity or this task.

- *For this file activity*
  - View Activity Details
- *For this task*
  - View Task Config
  - View Run History
  - View FIle Activity

Ø *To view File Activity information for other tasks:*

Go back to the Task List page and do either of the following:

- Set different filters, then click **REPORTS > File Activity**.
- For a different task, select the action **View File Activity**.

Ø *To pause, resume, or refresh the current page:*

1 Click the more options icon above the table, and select one of the following options.

- To pause the page refresh, click **Pause Page Refresh**.

- To resume the page refresh, click **Refresh Page Now**.
- To refresh the page, click **Refresh Page Now**.

See also:

- Task List
- Task Run
- About Tasks

Custom Duration

Search for a task or activity that occurred within a custom timeframe.

To access this dialog box: Click **REPORTS >Task Run** or **REPORTS > File Activity**. In the left Search pane, expand **Time/Date**. From the dropdown list, select **Custom**.

Ø *To establish the time frame in which to search:*

- Make selections in the **From** and **To** fields.

Audit

REPORTS > Audit

Audit reports include information about commands and configuration changes in MOVEit Automation.

For each entry, the Audit report displays the Log Time, Action, Target Type, Target Name, Result, Username and IP Address.

Items that appear in the list depend on how you accessed the Audit page.

How did you access the Audit page?	The Audit page contains
From the Hosts page, selected a single host, clicked **Actions > View Audit Records**.	List of the audited events against the selected host.
From the Task List page, for a single task, selected the action **View Audit Records**.	List of the audited events against the selected task.
From the Task List page, set optional filters. Clicked **REPORTS > Audit**.	List of the audited events against every task listed in the Task List, after filters are applied.
From the ***Scripts page*** (on page 103), selected a single custom script, clicked **Actions > View Audit Records**.	List of the audited events against the selected custom script.

Task List Filtering

Filters applied on the Task List page are automatically applied to the Task Run, File Activity, and Audit pages. You can remove or edit the applied filters.

- To remove the automatically applied filters, clear the **Apply Task List filter(s)** check box above the applied filters in the **Search & Filters** panel. To reapply the task list filters select the **Apply Task List filter(s)** check box.
- To edit the automatically applied filters, click **Edit** the **Apply Task List filter(s)** in the **Search & Filters** panel. You are redirected to the Task List page. Edit the selected filters on the Task List page. For more information, see ***Filtering and Sorting*** (on page 12).

Action filter

The Action filter limits report data to entries whose action matches the selected action options. The following table details the available Action filters.

Name	Description
Host config	Matches host configuration changes.
Task config	Matches task configuration changes.
Global settings config	Matches global settings changes.
Key and cert config	Matches SSH key, SSL cert, and PGP key configuration changes.
Script config	Matches script configuration changes.
Config import/export	Matches configuration import and export actions.
User group config	Matches user group configuration changes.
Task group config	Matches task group configuration changes.
Task control	Matches task control commands.
Schedule control	Matches scheduler thread control commands.
Debug level changes	Matches debug level changes.
Tamper check/reset	Matches tamper check actions.
Authentication	Matches signon/signoff audit entries.
Other actions	Matches any audit entry not covered by the above selections.

Time/Date filter

The Time/Date filter limits report data to entries within a predefined or custom time.

Log ID filter

The Log ID filter limits report data to entries greater than the specified value.

To view Audit information for other Hosts, Tasks, or Scripts.

Ø *Hosts*

Go back to the Hosts page, select a different host, and click Actions > View Audit Records.

Ø *Tasks*

Go back to the Task List page and do one of the following:

§ Set different filters, and click REPORTS > Audit.

§ For a different task, select the View Audit Records action.

Ø *Scripts*

Go back to the ***Scripts page*** (on page 103), select a different custom script, and click Actions > View Audit Records.

Ø *To pause, resume, or refresh the current page:*

1 Click the more options icon ⋯ above the table, and select one of the following options.

§ To pause the page refresh, click Pause Page Refresh.

§ To resume the page refresh, click Refresh Page Now.

§ To refresh the page, click Refresh Page Now.

See also:

§ Task List

§ File Activity

SETTINGS

§ About Tasks

System Settings

To access this page: Select SETTINGS > System Settings.

System Setting	Description
Debug Log	Specifies the amount of debug information that is saved, the filename, and maximum log file size. ***Field descriptions*** (on page 345).
Audit Log	Determines how to record issues that are encountered during file transfers. ***Field descriptions*** (on page 346).

Windows Event Log	Controls which Windows Event log is used. ***Field descriptions*** (on page 346).
ASx Logging	Directory for log files for each ASx transmission. ***Field descriptions*** (on page 347).
Tasks	Settings that apply to all tasks, such as maximum number of running tasks, polling intervals, multiple same task runs. ***Field descriptions*** (on page 347).
State File	Controls the caching of state information. ***Field descriptions*** (on page 348).
Tamper Detection	Activates the ability of MOVEit Automation to detect attempts by an intruder to alter database tables that contain audit information and activity history. ***Field Descriptions*** (on page 349).

Task Groups

Use task groups to

- Organize large numbers of tasks, hosts, scripts, and other elements. In the task filter, selecting a task group lists only the tasks that belong to the group.
- Filter the tasks that are listed on the TASKS page.
- Group together tasks, hosts, and keys/certs that are related to the same customer or work unit. For example, one task group to include the tasks/hosts/keys/certs for the Finance Department, and another task group to include those elements for the Sales department.
- Control access to MOVEit Automation. You can set permissions for a task group. Individual users are granted these permissions based on their membership in the group.

  The tasks and elements to which a user in a MOVEit Users group has access is determined by:

  - The task groups that are associated with that MOVEit Users group, and
  - The permissions that are are assigned to those task groups.
- Use task groups in association with MOVEit Users groups

  You can associate a task group with a MOVEit Users group. Members of the MOVEit Users group have access to the elements of the task group.

Note: To add members to a MOVEit Users group, associate a task group with a MOVEit Users group, or assign permissions to a task group, use the Admin Console Settings > Permissions menu. This feature is not supported in Web Admin.

However, any changes you make to a task group via Web Admin update the permissions for any users that have access to that task group.

See ***Create or Edit a Task Group*** (on page 156)

Add/Edit Task Group

Tasks must exist in MOVEit Automation before you can add them to a task group.

From the TASKS page, using Bulk Actions:

All of the tasks that are listed on the TASKS page (including multiple pages) are included in the group. Before you create a group or add to an existing group, use the filters to choose the tasks to include or add to the group.

- **Create a task group**: On the TASKS page, use filters to select the tasks. Click **Bulk Actions > Create task group**. The Add new task group dialog box opens.
- **Add tasks to a task group**: On the TASKS page, use filters to select tasks. Click **Bulk Actions > Add to task group**. Select the group and click **OK**. The Edit task group dialog box opens.

From the SETTINGS > Task Groups page

- **Create a task group**: Click **Add Task Group**. The Add new task group dialog box opens. Click **Add Tasks**, select the tasks, and click **OK**.
- **Add tasks to a task group**: Click a task group name. The properties page opens. Click **Edit**. Make changes in the Edit task group dialog box.

Task Group Field	Description
Friendly Name	Name that appears in the list of task groups
Description	Optional
Members - Tasks tab	If you created the task group from the SETTINGS > Task Group page, initially the group has no members. Click **Add Tasks**, select tasks, and click **OK**. If you created the task group using Bulk Actions, the group contains all the tasks in the list at that time. To remove a task from the group, click the red X.
Members - Tabs for member types: § Hosts § Scripts § SSH Keys § SSL Certs § PGP Keys	In each tab: *Assigned members* - You can add members of that type to the task group. Click the tab and click **Add <member type>**. *Referenced members*: In each tab, items of that type that are associated with the tasks in the group are listed in dimmed font. For example, if a task has a source or destination that uses a specific host, the host is a referenced member. Referenced members provide read-only access to task elements. To promote a referenced member to an assigned member, click  .

To remove a task group:

- Click **SETTINGS > Task Groups**. Select a task group and click **Remove**.

Date Lists

A date list contains a list of dates. You create date lists in Web Admin, or you can import a date list from a `*.txt` document.

To use a date list in a task, you add a schedule to the task and select the date list in the schedule. Settings in the schedule also determine how to handle the dates: include (run the task on the dates in the date list) or exclude (do not run the task on the dates). For more information, see ***Schedules*** (on page 85).

- To create a date list: Click SETTINGS > Date Lists. Click Add Date List.
- To edit a date list: Click SETTINGS > Date Lists. Click a date list name. In the Properties row, click Edit.
  See ***Add/Edit Date List field descriptions*** (on page 89).

Date Lists Created Outside Web Admin

You can create a date list in a text document. Use the following syntax:

- Each entry is a single line that uses the format `YYYY-MM-DD`
- Asterisk (*) wildcard is allowed in the year and month places.
- Hash (#) character designates a comment.

To add the date list to MOVEit Automation: Click SETTINGS > Date Lists. On the Date Lists page, click Add Date List > Import Date List.

See also ***Schedule*** (on page 85).

Keys and Certs

The SETTINGS > Keys and Certs page lists all keys and certs that have been created and/or imported into MOVEit Automation. You can ***filter the list*** (on page 12).

Certificate or Key Type	Information
SSL Client Certificates	MOVEit Automation uses client certificates for FTP/S and MOVEit Transfer authentication, S/MIME signing/encryption and AS1/AS2/AS3 authentication/signing/encryption. § ***About SSL Client Certificates*** (on page 158) § ***Obtaining and importing SSL Client Certificates*** (on page 159) § ***Creating an SSL Certificate*** (on page 160)

SSH Client Keys	SSH is typically used to access remote computers securely and to control web servers remotely. § ***Create an SSL Client Key*** (on page 161) § ***Importing SSH Client Keys*** (on page 161)
PGP Keys	PGP keys are typically associated with email messages and to encrypt files for transfer over public networks. To use PGP Keys in MOVEit Automation, you must purchase a special license key. For more information, see ***The OpenPGP Module in MOVEit Automation*** (on page 170). § ***About PGP Keys*** (on page 163) § ***Creating PGP Keys*** (on page 165) § ***Using PGP Keys from other PGP applications*** (on page 168)

About SSL Client Certificates

Authentication

MOVEit Automation uses client certificates for FTP/S and MOVEit Transfer authentication, S/MIME signing/encryption and AS1/AS2/AS3 authentication/signing/encryption.

An X.509 digital certificate is a document that verifies the identity of the holder of the certificate. Digital certificates are often issued by and vouched for by Certification Authorities (CAs), but may also be "self-signed". Every certificate contains two keys used by public/private key cryptography.

A certificate that is used for client authentication consists of the following components:

§ A public component that contains the name of the client and the public key.
§ A private component that contains an encrypted version of the private key.

Although it is possible to have a certificate without the private component, such a certificate cannot be used as a client certificate.

§ A password that protects the private key.

Certificate Stores on a Windows system

On a Windows system, certificates are registered with the operating system, usually in one of two locations: the Local Machine store, or the Current User store. (This store is also known as the "Personal" or "My" store.) The Local Machine store contains certificates that can be accessed by anyone on the local computer. Only administrators can add or modify certificates in this store. The Current User store contains certificates that can be accessed only by the currently signed on user. The current user has full access to the store and may add or modify certificates in the store. MOVEit Automation accesses the Current User store when looking for certificates, and can install (*import)* a certificate into this store.

To use a client certificate with MOVEit Automation:

1 Obtain a certificate, convert it (if necessary) to a form understood by Microsoft software, and import it into MOVEit Automation. For more information, see ***Obtaining and Importing SSL Client Certificates*** (on page 159).

2 Configure a MOVEit Automation host to use the certificate when communicating with a particular FTP server, MOVEit Transfer server, AS2 partner, etc. For more information, see the ***field descriptions for the type of host*** (on page 23).

Obtaining and Importing SSL Client Certificates

See also: ***About SSL Client Certificates*** (on page 158).

Ø *Obtaining a Certificate*

Certificates are typically delivered in one of the following forms:

§ Two ASCII files with extensions `.crt` and `.key` (or `.cer` and `.key`).

§ One binary file with the extension `.p12` or `.pfx`.

1 From the server administrator of your FTP server, obtain the following:

§ A certificate that has been registered with the FTP server.

§ The password for that certificate.

2 Put the certificate files on the computer that is running MOVEit Automation. If you are using a network file transfer mechanism to transmit the certificate files, use the proper ASCII vs binary transfer method.

Ø *Converting the Certificate*

Microsoft software imports client certificates from `.p12` (also known as `.pfx`) files. If you received `.crt` and `.key` files instead of a `.p12` file, you must convert them to `.p12` format. You can do this with the free program OpenSSL.exe from the ***OpenSSL Project*** (see http://www.openssl.org - ***http://www.openssl.org***).

Example

You receive the files `fred.crt` and `fred.key`. To convert them to a single `fred.p12` file, use the following command:

```
openssl pkcs12 -inkey fred.key -in fred.crt -export -out fred.p12
```

The command prompts for the password to the `fred.key` file before it writes the `fred.p12` file.

Ø *Importing the Certificate*

MOVEit Automation accesses the Current User store when looking for certificates, and can install (*import*) a certificate into this store.

1 In Web Admin, select SETTINGS > Keys and Certs > Import > SSL Client Cert. Select the `.p12` file and click Open.

2 Provide the password and click OK.

Create SSL Certificate

To access this dialog box: Click SETTINGS > Keys and Certs > Create > SSL Client Cert.

SSL Certificates that you create in MOVEit Automation with this dialog box are self-signed and are not automatically trusted by other sites. Self-signed certificates are suitable for the following purposes:

§ Testing.
§ Securing communications between MOVEit Automation and MOVEit Automation Admin.
§ In some cases, for production use in applications such as AS2.

The use of self-signed certificates is *not recommended* for securing web servers such as IIS, because they cause trust errors to occur with end users who visit a site configured with a self-signed certificate.

For certificates that are issued by well-known certificating authorities, see ***Obtaining and Importing SSL Client Certificates*** (on page 159).

NOTE: After you click Create Certificate on this dialog box, you cannot make any changes to the properties.

SSL Certificate Field	Description
Friendly Name	The name that appears in the list of keys and certificates
Email Address	Optional.
Country	Optional.
State	Optional.
Organization	Optional.
Organizational Unit	Optional.
Expires after	Select a number and unit (Days or Years)
Key Size	Length of the key in bits. Longer keys are more secure but require more processing time for cryptographic operations.

About SSH Client Certificates

SSH (Secure Shell) is a security protocol that is used tomake a secure connection to a server that has the SSH and SFTP (Secure File Transfer Protocol) protocols installed. SSH encrypts all communications to and from the client and server. When MOVEit Automation makes an SSH connection, it uses the SFTP protocol to transfer files to and from the server over the secure connection.

Using MOVEit Automation, you can:

§ ***Create new private/public SSH Key pairs*** (on page 161).
§ ***Import SSH Keys*** (on page 161) to use for SFTP connections. These are typically public only keys.
§ ***Connect to SFTP hosts using an SSH Key*** (on page 325).

Create SSH Client Key

To access this dialog box: Click SETTINGS > Keys and Certs > Create > SSH Client Key

SSH Key Field	Description
Friendly Name	The name that appears in the list of keys and certificates.
Key Type	Options: § RSA (default) § DSA
Key Size	Options: § 1024 (default) § 2048 § 3072 § 4096

See also: ***SSH/SFTP Hosts*** (on page 325)

Edit SSH Client Key

To access this dialog box: Click SETTINGS > Keys and Certs. In the list, click the name of the SSH Client Key. You can ***filter the list*** (on page 12).

On the Properties row, click Edit.

§ You can change the Friendly Name, which appears in the list of keys and certs.

Importing SSH Client Keys

MOVEit Automation can import existing keys that have been obtained from remote servers. The most common SSH implementation, OpenSSH, generates its keys via ssh-keygen and stores the keys in files named `$HOME/.ssh/id_dsa` or `$HOME/.ssh/id_rsa`, where `$HOME` is the home directory of the user and `user` is the user name.

Example of a Linux session that generates a key:

```
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
```

```
The key fingerprint is:
18:37:c3:bc:10:f0:c0:38:19:3e:80:7b:73:79:15:9c user@linuxsrv1
$
```

To import an SSH Client Key into MOVEit Automation:

1. Transfer the key (the file that does not end in `.pub`) to a location that can be accessed from the computer where you are running MOVEit Automation Web Admin.
2. In Web Admin, click SETTINGS > Keys and Certs. Click Import > SSH Client Key.
3. Browse to and select the key. The Add Imported SSH Client Key dialog box opens
4. Provide a Friendly Name and optional Password, and click Add Key.

Configuring the key on the SSH server

After a key has been created, the SSH server must be configured to authorize the key for logon. The procedure for this depends on the type of SSH software running on the server.

- OpenSSH:, Append the OpenSSH version of the public key (one very long line of text) to the file `~/.ssh/authorized_keys` on the user's UNIX machine.
- SSH.com: Append a line like `Key mykey.pub` to the file `~/.ssh2/authorization,` and create the file `~/.ssh2/mykey.pub` with the contents of the SSH format of the key.
- SSH.com Tectia Server: Upload the SSH version of the public key to the user's `authorized_keys` directory on the server, with an arbitrary filename. This is typically
  - On UNIX: `$HOME/.ssh2/authorized_keys`
  - On Windows: `%USERPROFILE%\.ssh2\authorized_keys`

For other server types, see the documentation for that server.

PuTTY Key Generator

By default, the PuTTY Key Generator exports two files; one for a private key and one for a public key. To generate a file format for use in the SSH client, export the PuTTY key as an OpenSSH Key (using the Conversions menu, if available). The passphrase you designate will also be used for the exported OpenSSH key..

ssh.com Key Import

You can import only ssh.com keys that *are not* password-protected.

Note: Blank password OpenSSH and Password Protected OpenSSH keys can be imported.

About PGP Keys

Note: To use PGP Keys in MOVEit Automation, you must purchase a special license key. For more information, see ***The OpenPGP Module in MOVEit Automation*** (on page 170).

MOVEit Automation stores PGP keys in files called *keyrings*, which contain public keys, and private/public keypairs. For more information, see ***Public and Private Keys*** (on page 164).

To view PGP Keys in MOVEit Automation

§ Click SETTINGS > Keys and Certs. The page lists all keys and certs. You can ***filter the list*** (on page 12).

In this section:

Public and Private Keys

PGP Public Keys

Public keys are non-secret keys that are often widely distributed to other users. To encrypt a file to send to someone, you must have a copy of their public key. If you sign the file, the recipient must have a copy of your public key in order to check the signature.

Typically you import the public keys of several other users into your keyring, and export your own public key to send to other users. There is usually little security risk associated with distributing your public key.

Public PGP keys are those for which you lack a private key.

In Web Admin you can ***import*** (on page 168) and ***export*** (on page 169) keys

PGP Private/Public Keypairs

Private/public keypairs (also known as *secret keys* or *private keys*) are keys that are generated by you and that contain information that must not be given to other users. A secret key also contains a copy of an associated public key. You must export the public component of your private/public keypair to allow others to encrypt files to be sent to you.

Although a password is not required, private/public keypairs are typically encrypted with a password.

As a best practice, minimize the number of different secret keys, even though it is possible to have multiple private/public keypairs.

In MOVEit Automation Admin, private/public keypairs are listed as **Private** because you have the private keys.

Setting Up PGP Keys

When you first install/configure PGP software, you typically do the following, in the order shown:

1. ***Create a new PGP key pair*** (on page 165).

   This is your key pair (or your company's). It consists of one private key and one public key.

   The private key is password-protected and stored securely on your machine.

   The public key is meant to be distributed to anyone else who needs to exchange PGP-encrypted files with you. Typically this key is exported to an ASCII file and emailed as an attachment.
2. ***Import the keys*** (on page 168) of partners and customers with whom you want to exchange PGP-encrypted files. Keys imported this way are put on your *keyring*.
3. Before going into production, test the exchanging files with PGP.

   To run a test:

   - Make sure that both sides (the recipient and the sender) have each other's public key.
   - The sender encrypts the file with the recipient's private key. Optionally, the sender can sign the file with the sender's private key; doing so provides a method to authenticate the sender.
   - The recipient receives the file and decrypts it using their (the recipient's) private key. If the sender also signed with the sender's private key, the recipient can verify the authenticity of the contents by using the sender's public key.

Create PGP Key

Note: To use PGP Keys in MOVEit Automation, you must purchase a special license key. For more information, see ***The OpenPGP Module in MOVEit Automation*** (on page 170).

Use this dialog box to create a private/public keypair in your Private keys collection.

To access this dialog box: Click SETTINGS > Keys and Certs > Create > PGP Key.

PGP Key Field	Description
Friendly Name	Name that appears in the list of Keys and Certificates.
Email Address	Optional. If provided, is included in the friendly name of the key. The address is not usually used to address PGP-encrypted email, but provides contact information for technical issues regarding the PGP key.
Format	Supported formats: § RSA § DSS/DH (Digital Signature Standard / Diffie-Hellman). § RSA Legacy - might be necessary if you are exchanging encrypted files with someone who is using a very old version of PGP. NOTE: The previous PGP module for MOVEit Automation, Authora EDGE PGP Library, has been replaced by Didisoft OpenPGP Library for .NET in order to address various limitations. Didisoft does not support generating DSS or RSA Legacy keys, which are options that EDGE SDK does support. For backward compatibility, these options are still available. However, if you attempt to generate a DSS or RSA Legacy key using the new IPSP/Didisoft components, you receive the error message "This version of MOVEit Automation doesn't support generating xxxxx keys".
Length	Length of the key in bits. Longer keys are more secure but require more processing time for cryptographic operations. Options: § 2048 bits § 2048 bits (usually preferred) § 4096 bits - generating a key of this length might take over 10 minutes
Signing Algorithm	Hash algorithm that is used for signing the key. Cannot be selected for RSA legacy keys. Options: § SHA1 - Default for DSS keys. Some older PGP applications support only SHA1 for DSS keys. § SHA256 - More secure. Default for RSA keys. For compatibility with older applications, you might need to select SHA1 for RSA keys. § SHA512 - Provides the best security.

Expires After	Specify a number and unit (Days or Years). Shorter expiration times are: § More secure - Reduces the time available for an opponent who gains access. § Less convenient - When the key approaches its expiration date, you must generate a new key and send its public component to your correspondents.
Password and password confirmation	Used to encrypt the secret key. The password is recorded in the MOVEit Automation encrypted settings file, so that you do not need to reenter it when signing or decrypting files.

See also:

§ ***About PGP Keys*** (on page 163)
§ ***Selecting the Algorithm for the Public Key*** (on page 166)
§ ***Importing PGP Keys*** (on page 168)
§ ***Exporting keys*** (on page 169)

Edit PGP Key

To access this dialog box: Select SETTINGS > Keys and Certs. In the list, click the name of the PGP key. You can ***filter the list*** (on page 12).

On the Properties row, click Edit.

Field	Description
Key Name	This name you provided when you created the key. You cannot edit this field.

Encryption Algorithm	When encrypting, MOVEit Automation uses the symmetric encryption algorithm associated with the public key of the first recipient. Make your choice on the basis of compatibility with the recipient's software. Options: § Default - uses the default preferred algorithm that is specified in the PGP public key of the recipient. All other choices override this algorithm. NOTE: This is the safest choice. § 3DES - Triple DES. Three rounds of the 56 bit DES algorithm. § AES128 - 128-bit AES. AES is the Advanced Encryption Standard approved by the US National Institute of Standards and Technology. § AES192 - 192-bit AES. § AES256 - 256-bit AES. § CAST5 - 128-bit CAST5. Rarely used outside the context of PGP. § IDEA - 128-bit IDEA. Rarely used outside the context of PGP. § TWOFISH - Similar in security to AES.

See also

About PGP Key (on page 163)s

Obtaining PGP Keys from Other PGP Applications

If you have been using another PGP application, you have already established a keyring.

You can use these keys in MOVEit Automation by exporting them from the other PGP application and then importing them into MOVEit Automation. Exporting the keys does not remove them from the original application. You can continue to use them with the old application.

This topic includes instructions for:

§ Exporting from GNU Privacy Guard
§ Exporting from PGP™ Command Line from Symantec®

See also: ***Importing PGP Keys into MOVEit Automation*** (on page 168)

Exporting from GNU Privacy Guard

To export a single public key from GnuPG, use a command line like:

```
gpg -a --export "Fred Smith" >fredsmith-public-key.asc
```

To export all public keys from GnuPG, use a command line like:

```
gpg -a --export >all-public-keys.asc
```

To export a single private key from GnuPG, use a command line like:

```
gpg -a --export-secret-keys "Fred Smith" >fredsmith-private-key.asc
```

To export all private keys from GnuPG, use a command line like:

```
gpg -a --export-secret-keys >all-private-keys.asc
```

Note: unlike some other applications, GnuPG does not export the public key when it exports the private key. To export both the private key and the public key for a user, use a sequence like:

```
gpg -a --export "Fred Smith" >fredsmith-both.asc gpg -a --export-secret-keys "Fred Smith" >>fredsmith-both.asc
```

Exporting from PGP® Command LIne from Symantec®

To export a single public key from PGP, use a command line like:

```
pgp --export "Mary Jones" --output maryjones-public-key.asc
```

To export a single public/private keypair from PGP, use a command line like:

```
pgp --export-key-pair "Mary Jones" --output maryjones-both.asc
```

There is not a single PGP Command Line command that will export all keys.

Importing PGP Keys

Note: To use PGP Keys in MOVEit Automation, you must purchase a special license key. For more information, see ***The OpenPGP Module in MOVEit Automation*** (on page 170).

Import PGP keys that were ***obtained (exported) from other PGP applications*** (on page 168).

1 Click SETTINGS > Keys and Certs. Click Import > PGP Key. A browse box opens.
2 Select the file that was exported by the other PGP application. The Enter Password dialog box opens.
3 If the PGP key has a password, enter it. If it does not, leave the field blank. Click OK.

Exporting PGP Keys

Note: To use PGP Keys in MOVEit Automation, you must purchase a special license key. For more information, see ***The OpenPGP Module in MOVEit Automation*** (on page 170).

1. Click SETTINGS > Keys and Certs. In the list, select the key. You can ***filter the list*** (on page 12). The properties page for the key opens.
2. In the upper right corner of the page, click Export. The Export PGP key dialog box opens.
3. Optionally select Include Private Key. Click Export Key. The file is downloaded as an `.asc` file.

Deleting Keys

Caution: Do not delete a key listed under Private keys unless you have a backup copy of it. Without the key, you cannot decrypt messages that were encrypted by the sender with the public component of that key.

To remove a key from your keyring:

1. Click SETTINGS > Keys and Certs.
2. In the list of keys and certs, select the key. You can ***filter the list*** (on page 12). The properties page for the key opens.
3. In the upper right corner of the page, click Delete. Confirm the deletion.

OpenPGP Module in MOVEit Automation

PGP encryption is a form of public key file encryption that is used with email messages and to encrypt files for transfer over public networks.

MOVEit Automation contains a built-in, fully integrated, OpenPGP software module with comprehensive encryption and key management capabilities. Use of the MOVEit Automation OpenPGP capabilities is optional. Activation requires a special license key, and commercial use requires payment of a one-time license fee and an annual maintenance fee. For more information, contact ***MOVEit support https://www.ipswitch.com/support/***.

The OpenPGP module in MOVEit Automation supports:

- Creating and deleting public and private keys.
- Importing and exporting private keys with other OpenPGP applications.
- Automatically encrypting, encrypting and signing, decrypting, and signature-checking by new and existing >MICEN> tasks.

The OpenPGP software in MOVEit Automation has been commercially licensed from ***Didisoft http://www.didisoft.com/net-openpgp/***, which warrants that it is fully interoperable with all other OpenPGP applications, including PGP Command-Line[TM] by PGP Corporation.

Note: Symantec Corporation holds a registered trademark on the term PGP and sells OpenPGP products under the name PGP®. All uses of the term PGP in MOVEit Automation products and documentation are to be treated as the common abbreviation of OpenPGP and not as references to Symantec Corporation software, except where noted.

MOVEit Automation is also used to automate other third-party command-line PGP clients. See the document *PGPOtherVendors* for complete documentation and a library of pre-tested scripts to automate command-line utilities from GnuPG, Network Associates Command Line and PGP Corp. (Similar command-line clients have also been configured to work with MOVEit Automation.) For more information, contact ***MOVEit support https://www.ipswitch.com/support/***.

Using Global Parameters to Sign Outgoing Files

It is common for a given site to sign most or all of its outgoing files with the same key.

To avoid having to set the PGPSignerKey in the Process step of each task that does PGP encryption and signing, you can set a global parameter named PGPSignerKey.

You can override the global signer key by specifying it in an individual process.

For more information, see Tasks.

Using a Process to Encrypt Files

To encrypt a file before sending it to its destination: In a task, add a process step that uses the PGP Encrypt and Sign or the PGP Encrypt Only script.

Prerequisites: You must already have added PGP keys to MOVEit Automation. For more information, see ***Create PGP Key*** (on page 165).

- PGP Encrypt and Sign requires a recipient key and a signer key. Only private keys can be used to sign files. These keys are in your Private group
- PGP Encrypt Only requires a recipient key, which can be selected from Private Keys (if you want to send a file to yourself) or from Public Keys.

Details:

1. Click TASKS. Create a new traditional or advanced task, or edit an existing task.
2. Click Step > Process. If the task already contains steps, select a location for the new step.
   The Add Process dialog box opens.
3. Select one of the following built-in scripts:
   - PGP Encrypt and Sign (Best practice, because it allows the recipient to confirm that you are the sender of the message.)
   - PGP Encrypt Only
4. Click Edit task parameters. To provide a parameter value, click the plus sign (+) next to the parameter.
   Parameter values for these scripts are PGP keys.
5. Click Set Key. The Browse PGP Keys dialog box opens, listing the keys of the appropriate type from your Private Keys or Public Keys key group.
6. Select a key and click OK. Click Save. For additional parameter values, click the plus sign and repeat steps 5 and 6.
7. In the Edit Task Parameters dialog box, click Save. Click Add Process.

You can indicate to your recipient that the file you are sending is PGP-encrypted. For more information, see ***Mark File as PGP-Encrypted*** (on page 171).

Mark File as PGP-encrypted

By default, MOVEit Automation sends files under their original name. You can indicate to your recipient that the file you are sending is PGP-encrypted by appending .pgp to the filename.

To do this, edit the task's destination.

Details:

1. Click TASKS. Create a new traditional or advanced task, or edit an existing task, so that it contains a ***process to encrypt files*** (on page 171).
2. Do one of the following:
   - Add a destination: Click Step > Destination, specify how to save the destination file, and click Next. The next Edit Destination dialog box opens.

     - or -
   - Edit an existing destination: In the destination row, click  and select Edit. The Edit Destination dialog box opens.

3 Uncheck the Use original filename(s) checkbox. Change the filename to [OrigName].pgp and click Save.

Decrypting Files

To decrypt files: In a task, add a process step that references the built-in script ***PGP Decrypt*** (on page 126).

If the original (encrypted) had an extension indicating that it was encrypted (for example, .pgp), edit the task's destination to save the decrypted version without the extension.

Prerequisites: To decrypt a file, you must have a private key that corresponds to the public key that was used to encrypt the file. MOVEit Automation searches for a matching key in your PGP keyring.

Details:

1 Click TASKS. Create a new traditional or advanced task, or edit an existing task.
2 Click Step > Process. If the task already contains steps, select a location for the new step.
The Add process dialog box opens.
3 Select the built-in script PGP Decrypt. Click Add Process.
4 If the original (encrypted) file used an extension such as .pgp to indicate that they were encrypted, edit the destination to remove the extension. Do one of the following:
- Add a destination: Click Step > Destination, specify how to save the destination file, and click Next. The next Edit Destination dialog box opens.
- or -
- Edit an existing destination: In the destination row, click ⚙▾ and select Edit. The Edit Destination dialog box opens.
5 Uncheck the Use original filename(s) checkbox. Change the filename to [OnlyName] and click Save.

Global Parameters

Global parameters apply to all tasks.

What happens when a task references a parameter?

A task can reference a parameter via the [Parm:ParameterName] syntax, or a script call to MIGetTaskParam(). The following occurs, in the order listed:

1 MOVEit Automation looks in the information for that task for a parameter with that name. If it is found, that task parameter is used.
2 If the parameter is not found in the task, MOVEit Automation looks for a global task parameter with that name and uses it if available.
3 If no parameter with that name is available, MOVEit Automation uses an empty string.

Uses for Global Parameters

Global task parameters can be useful when there are multiple tasks or Next Actions that use the same parameter. You can use built-in parameters to define default or common settings across multiple tasks that use built-in MOVEit Automation scripts. For example:

- PGP public/private key pairs that are used to sign/encrypt PGP files. For more information, see Using Global Parameters for Keys.
- ZIP process compression options.
- Common "email errors to" email addresses

Types of global parameters

- A **built-in global parameter** is the global setting for any parameter that is used by a built-in script.

  If a specific built-in script parameter is validated or selected using a drop-down menu, the built-in global parameter is validated or selected in the same way.
- A **custom global parameter** is one that you create, that is not a parameter in any built-in script.

  You must always type the value of a custom global parameter. The value is never validated.

Ø *To configure a global parameter:*

1. Select **SETTINGS > Global Parameters.**
2. Click **Add Parameter** and provide a **Parameter Name** and **Value.**

Add/Edit Global Parameter

Global parameters apply to all tasks.

To access this dialog box: Click **SETTINGS > Global Parameters**

- **To Add a global parameter**

  Click **Add Parameter** and provide a Parameter Name and Value.
- **To Edit a global parameter:**

  Click a parameter name. In the Edit Parameter dialog box, make changes.

For more information, see ***About Global Parameters*** (on page 172).

Global Parameters and Error Reporting using Next Action

To use global task parameters to make error reporting through Next Action email messages easier, set up the following name/value pairs as global task parameters.

ERR_EMAIL = single email address or a comma-separated list of email addresses)

ERR_SUBJECT = ERROR - [Taskname] - [hh]:[tt]:[ss]

ERR_MESSAGE = At [yyyy]-[mm]-[dd] [hh]:[tt]:[ss], task '[TaskName]' encountered error #[ErrorCodeFile] - [ErrorDescriptionFile] - while transporting '[OrigName]'

On each task you want to send this error, set up a per-file, on errors Next Action that sends email to "[PARM:ERR_EMAIL]" with subject "[PARM:ERR_SUBJECT]" and a message "[PARM:ERR_MESSAGE]".

See also:

COMMANDS

§ ***About Global Parameters*** (on page 172)
§ ***Add or Edit a Global Parameter*** (on page 173)

To access these commands:

§ At the top right of the Web Admin window, click COMMANDS and make a selection.
The command is executed immediately.

The commands that appear in your menu depend on your permissions and configuration.

Command	Description
Refresh config	Updates the user's configuration to the most recent configuration that is stored in and being used by MOVEit Automation. Keyboard shortcut: F5 Use of this command is rare. MOVEit Automation Admin automatically performs this activity as needed, as part of its normal operation.
Import config	Takes an exported XML config file and reads it back into the system. Can be used to reinstall a backed-up configuration.
Export config	Writes a copy of the XML config file that is currently loaded into MOVEit Automation Admin. This command can be used to back up configurations.

Stop task scheduler	Disables the MOVEit Automation task scheduler. Results: § No tasks are automatically started. § Tasks that are already running are run to completion. § Tasks can be manually started by MOVEit Automation Admin or API. If MOVEit Automation is restarted, the Scheduler is automatically restarted. To start the MOVEit Automation service with the scheduler disabled, start MOVEit Automation from the command line, using the -k option. For more information, see ***Running MOVEit Automation in the Foreground, Not As a Service*** (on page 353).
Start task scheduler	Enables the task scheduler.
Shut down service	Disables the MOVEit Automation task scheduler. Waits until no tasks are running, and then stops the MOVEit Automation service. A Host Disconnected message appears. *Recommended:* Shutdown service is the preferred method to cleanly shut down MOVEit Automation. Stopping the service through the control panel or command line interface abruptly terminates any tasks that are running. Note: Tasks that are launched ("looped") using Next Actions might have to be stopped individually, because such tasks are not launched from the scheduler. On the TASKS page, locate the task and select the action Stop Running Task.
Test antivirus	Causes MOVEit Automation to deposit a file that has a `*.tmp` extension with a test virus signature (the EICAR test string) into the MOVEit Automation cache directory. The test file is harmless, but it should be treated by any real-time antivirus package as an actual virus. If the test is successful, a message appears in MOVEit Automation Admin. For example: `Success - AntiVirus test successful. Detected antivirus package 'Symantec AntiVirus'.`

Reset tamper detection	Resets the MOVEit Automation tamper detection mechanisms so that it begins tamper checking of audit and statistics entries. Any previous entries are no longer covered by tamper detection.

Macros

Macros are configuration snippets used in source, destination and next action elements to represent dates, times, filenames and task parameters. For example, if the year is 2015, a task using a macro of `data[yyyy].log` runs against a value of `data2015.log`.

Macro keywords:

- Enclose macro keywords in square brackets (`[somekeyword]`).
- Place macro arguments (usually a DateSpec or an interger) after the keyword and a colon (`:`). For example:

  `[someKeyword]:someArguement`.

Macros in task parameters or global parameters are interpreted as macros as usual. For example:

A global parameter named `Error_Subject` is set with a value of `ERROR in '[TaskName]' at [hh]:[mm]:[ss]`

A Next Action subject is set to `[Parm:Error_Subject]`

When the task is run, this is interpreted as `ERROR in 'Get TPS Reports' at 12:34`

Where macros can be used:

- Source Folder Name
- Source Filemask
- Destination Path
- Destination Filename
- Email Message Address To
- Email Message Subject
- Email Message Body
- FTP QUOTE Fields
- Parameters consumed by built-in or custom scripts

See also:

- List of Macro Keywords
- ***Macro Date and Time Syntax*** (on page 182)
- ***Macro Functions*** (on page 184)

Macro Keywords

Attribute	Description	Applicable Host Types
DestFileName	The filename of the most recent destination file, not including any directory names. For example: "report12.txt". This is used primarily in per-file Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
DestFolderPath	The folder path, including all folder components but not the filename, of the most recent destination file. For example: `/pub/reports`. This is used primarily in per-file Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
ErrorCodeFile	The last error code encountered for the current file, or 0 if no error. For an example of how to use this macro to report errors, see Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
ErrorDescriptionFile	The last error description encountered for the current file, or empty if no error. For an example of how to use this macro to report errors, see Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
ErrorCodeTask	The last error code encountered for this task, or 0 if no error. For an example of how to use this macro to report errors, see Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
ErrorDescriptionTask	The last error description encountered for this task, or empty if no error. For an example of how to use this macro to report errors, see Next Actions-***Send Email*** (on page 83) or ***Run Task*** (on page 84).	All
FileDateStamp	The date stamp of the file as recorded by the source, in the form YYYY-MM-DD HH:MM:SS. Not all sources provide date stamps. When using this macro in a destination filename or folder name, you usually combine it with macro string functions. For example, `[LEFT([FileDateStamp],10)]` yields the `YYYY-MM-DD` date part and `[MID([FileDateStamp],12,2)]` will yields the hour from the date and time information of the original source file..	Filesystem, FTP (most), MOVEit, SSH

Attribute	Description	Applicable Host Types
FileSize	The size of the file, in bytes, as recorded by the source. Note: some unusual FTP servers do not provide the file size; in these cases, a size of 0 is used.	Filesystem, FTP (most), MOVEit, SSH
FolderID	Unique number identifying a MOVEit folder. For example: `1236518`.	MOVEit
FolderName	Name of remote folder. This macro returns only the last part of the path. Example: Given a full remote path of `frog\dog\cat`, this macro returns `cat`.	Filesystem, FTP, MOVEit, SSH
FullPath	The full path of the file as it was on the source, including all directories and the filename.	Filesystem, FTP, MOVEit, SSH
ID	Unique number identifying a MOVEit file. For example: `251660214`.	MOVEit
NominalStart	The time this task was officially started in YYYY-MM-DD HH:MM:SS format (for example, 2006-07-18 11:33:16). This value combined with the TaskID yields a key that uniquely identifies a single task run, for example, in the "stats" database table.	All
OrigComment	The upload comment that was specified when the file was uploaded. (This is often blank.)	MOVEit
OrigName	Original name of this file. Example: `Example.txt`	Filesystem, FTP, MOVEit, SSH
OrigUser	The username of the user who uploaded the file.	MOVEit
OrigUserEmail	The email address of the user who uploaded the file.	MOVEit, POP3
OrigUserFull	The full name of the user who uploaded the file.	MOVEit
OrigUserID	The UserID of the user who uploaded the file, if the MOVEit Transfer host is version 5.5 or later. If the MOVEit Transfer host is an earlier version, this will be the empty string. (The UserID is typically a long string starting with the username; a typical UserID might be "fred9zyupmuxa6dk".)	MOVEit

Attribute	Description	Applicable Host Types
OnlyName	The original filename minus the extension and the period. Example: Given "frog.txt", this macro returns "frog".	Filesystem, FTP, MOVEit, SSH
OnlyExt	The original filename extension. Example: Given "frog.txt", this macro returns "txt".	Filesystem, FTP, MOVEit, SSH
Parm:*ParmName*	Returns the task parameter named ParmName. If there is no such parameter, the empty string is returned.	All
RelativePath	The pathname of the directory for this file, relative to the originally specified source path. This applies only when Include Subdirectories is selected, and cannot be used for source paths. For example, if the source path is `C:\outgoing` and the file in question is `C:\outgoing\reports\Fred.txt`, then when used in a destination path, [RelativePath] is `reports`.	Filesystem, FTP, MOVEit, SSH
Rnd	A random decimal number. Use the format `[Rnd:len]` where len is the desired number of digits. The random number generator is of cryptographic quality.	All
SyncReport()	Synchronization tasks can use a `[SyncReport()]` macro that allows a complete report of all synchronization actions to be sent in a Next Action email notification. For more information, see Synchronization - Next Actions. documentation.	Next Actions Following Sync Tasks
TaskID	The ID of the task that is running. This number is used internally in the configuration files and the database to identify tasks. A typical value would be a nine-digit number such as `618116254`.	All
TaskName	The name of the task that is running	All
TaskStatus	The status of the task that is running. Values: § `Success` - the task encountered no errors and processed at least one file or ran at least one script § `Failure` - the task encountered one or more errors § `No xfers` - the task encountered no errors but did not process any files or run any scripts.	All
Date specification (see below)	Current date and time. Example: Given a time of `10:06` and a macro of `[HH][TT]`, the macro returns `1006`.	Filesystem, FTP, MOVEit, SSH

Attribute	Description	Applicable Host Types
Macro function (see below)	Returns the results of a string operation	All

Examples: (Given January 3, 2002 13:11:01, original filename=myfile.txt)

- `[WW][AAA].[OnlyExt]` Returns `00Thu.txt`
- `gotit[JJJ]-[YYYY].log` Returns `gotit002-2002.log`
- `[OrigName]` Returns `myfile.txt`
- `[OnlyName]_[MM]-[DD]-[YY].[OnlyExt]` Returns `myfile_01-03-02.txt`
- `ReallyBigText [AAA], [MM]-[DD]-[YY]` Returns `ReallyBigText Tue, 01-03-02`

Macro Date and Time Syntax

The Macro Date and Time Syntax specifies how date and time elements are represented in filenames and messages. Elements supported include day-of-week, day-of-month, day-of-year (Julian), hour, minute, second, month, week-of-year and year.

Operators such as the minus sign normally apply to all times and dates in a macro phrase. To apply operators to only part of a macro phrase, use double-quotes to delimit phrases. For example, if today is currently July 5, 2007, a macro of:

- `[dd][mm-][yyyy] [dd][mm][yyyy]` yields 05062007 05062007
- `"[dd][mm-][yyyy]" [dd][mm][yyyy]` yields "05062007" 05072007

Attribute	Description
A	DAY-OF-WEEK; minimal numeric. Example: "2" (Sunday=0)
AAA	DAY-OF-WEEK; three-letter abbreviation. Example: "Tue"
AAAA	DAY-OF-WEEK; full. Example: "Tuesday"
B	DAY-OF-WEEK; minimal numeric. Example: "2" (Sunday=1)
D	DAY-OF-MONTH; minimal representation. Example: "7"
DD	DAY-OF-MONTH; two-digit representation. Example: "07"
H	HOUR; minimal representation, 24-hour clock. Example: "7"
HH	HOUR; two-digit representation, 24-hour clock. Example: "07"
HHH	HOUR; minimal representation, 12-hour clock. Example: "7"
HHHH	HOUR; two-digit representation, 12-hour clock. Example: "07"
II	am/pm; two-digit representation of "am" or "pm" designation. Example: "pm"
J	JULIAN DATE; minimal representation. Example: "7" First day of year is 0.
JJJ	JULIAN DATE; three-digit representation. Example: "007" First day of year is 0.
K	JULIAN DATE; minimal representation. Example: "7" First day of year is 1.
KKK	JULIAN DATE; three-digit representation. Example: "007" First day of year is 1.
M	MONTH; minimal numeric representation. Example: "7" First month is 1.

Attribute	Description
MM	MONTH; two-digit numeric representation. Example: "07" First month is 1.
MMM	MONTH; short representation. Example: "Jan"
MMMM	MONTH; full representation. Example: "January"
S	SECOND; minimal representation. Example: "7"
SS	SECOND; two-digit representation. Example: "07"
T	MINUTE; minimal representation. Example: "7"
TT	MINUTE; two-digit representation. Example: "07"
W	WEEK-OF-YEAR; minimal representation. First week is numbered 0. Example: "2"
WW	WEEK-OF-YEAR; two-digit representation. First week is numbered 0. Example: "02"
X	WEEK-OF-YEAR; minimal representation. First week is numbered 1. Example: "2"
XX	WEEK-OF-YEAR; two-digit representation. First week is numbered 1. Example: "02"
YY	YEAR; two-digit representation. Example: "02"
YYYY	YEAR; four-digit representation. Example: "2002"
+	The + symbol following any date designation increments the current date by one unit of time, namely amount of time implied by the designation. The entire date is affected, with rollover from day-to-day, month-to-month, year-to-year, etc. as required. A + after a day designation increments the date by one day; a + after a month designation increments the date by one month, etc. For example, on June 30, 2003, "[MMM] [DD], [YYYY]" is rendered "Jun 30, 2003", but "[MMM] [DD+], [YYYY]" is rendered "Jul 1, 2003".
-	The - symbol following any date designation will decrement the current date by one unit of time, namely the amount of time implied by the designation. The entire date is affected, as with a + symbol. Hence, [MMM-] decrements the date by one month, etc.

Macro Functions

Macro functions are built-in functions that perform string operations on their arguments.

These functions are patterned after the identically-named functions in Basic.

Macro function references must start with a [, then the name of the function, then a (.

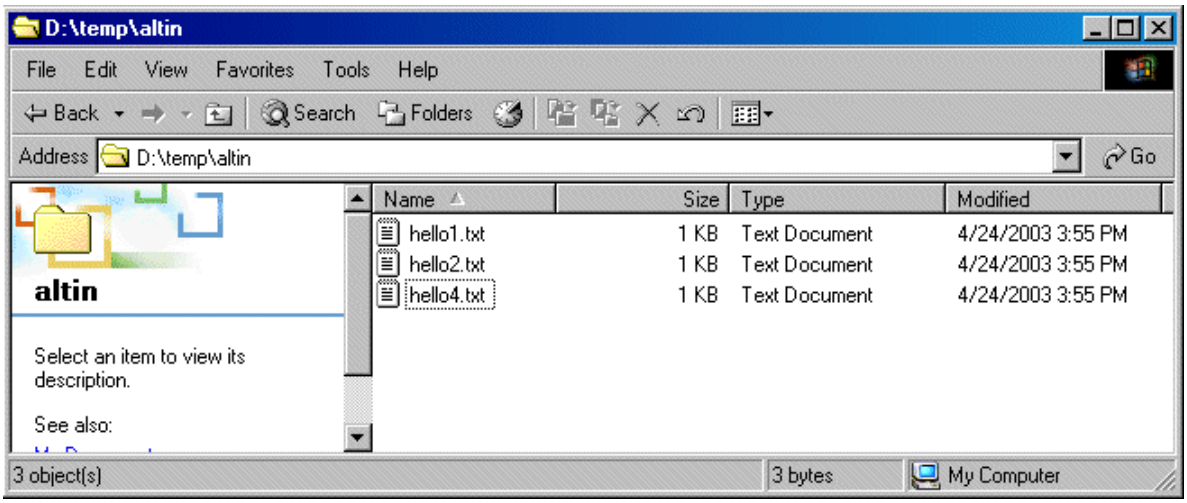
Function name	Description
LEFT(arg, count)	Returns the leftmost "count" characters of "arg". If arg is less than count characters long, the entire string is returned.
LEN(arg)	Returns the number of characters in "arg", as a decimal number.
MID(arg, start, count)	Returns "count" characters from "arg", starting at position "start" (where 1 is the first character). If ", count" is omitted, then the function returns the characters starting at "start" and going through the end of "arg". For example, if the original filename is ABCDE.TXT, then the value of the macro [MID([OrigName], 2, 3)] is BCD and the value of [MID([OrigName],2)] is BCDE.TXT.
RIGHT(arg, count)	Returns the rightmost "count" characters of "arg". If arg is less than count characters long, the entire string is returned.

Common Applications

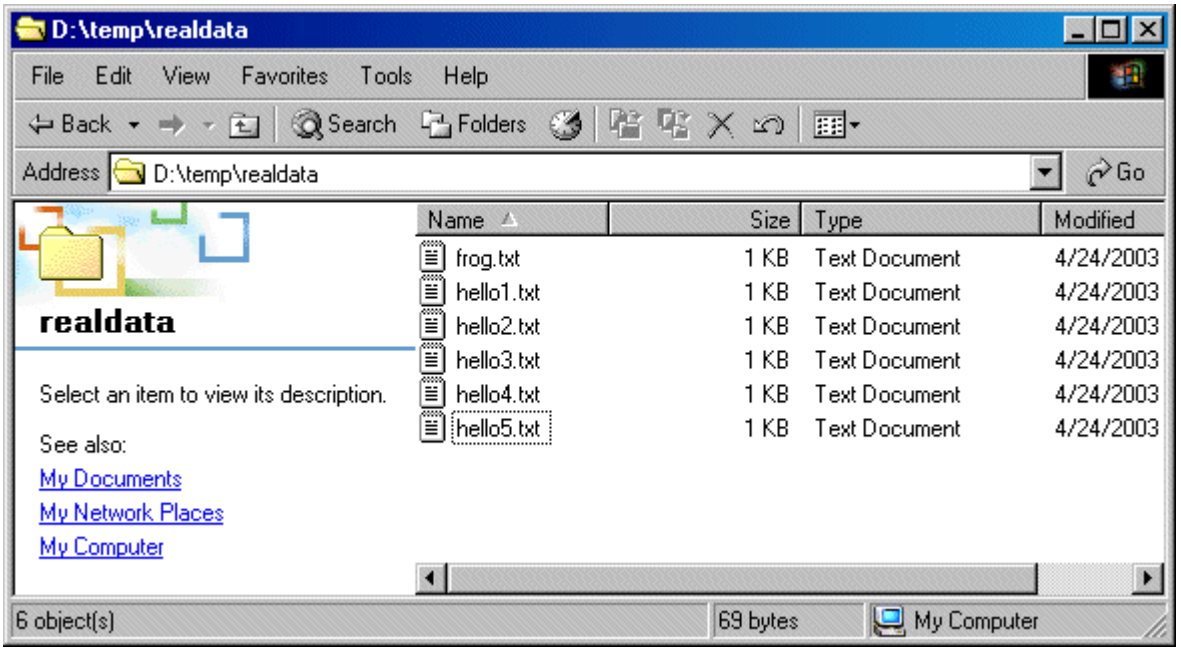
Trigger Files

Using its scripting abilities, MOVEit Automation can be configured to use a "trigger" file to force the upload of a second "data" file of the same name from a local folder. This is a situation often encountered with mainframe file transfers, where a mainframe determines which files are to be transferred by using trigger files such as these.

In this example, we have a "trigger" folder named "d:\temp\altin". This folder contains three very small text files. (The contents of these files can be anything, but it is the names of the files which are important.)



We also have a "data" folder named "d:\temp\realdata". This folder contains 6 data files, but we only want to move a few of these files up to our selected destination.

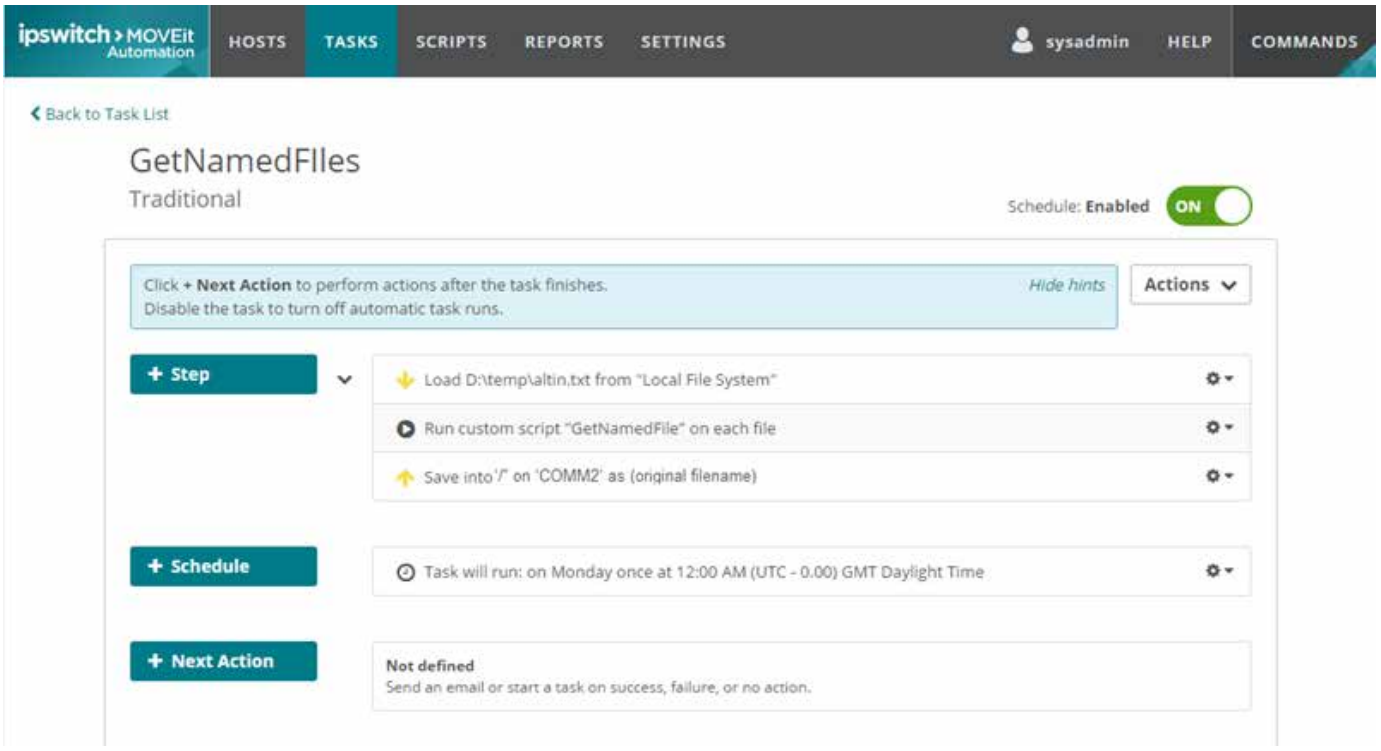


We create a new script called "GetNamedFile.vbs" and import it into MOVEit Automation as a process named "GetNamedFile". This process does three things:

§ Builds up the full path of the real data file given the name of a trigger file and a path specified as a task parameter.
§ Replaces the contents of the working cache with the contents of the data file.
§ Deletes (and scrubs) the original data file.

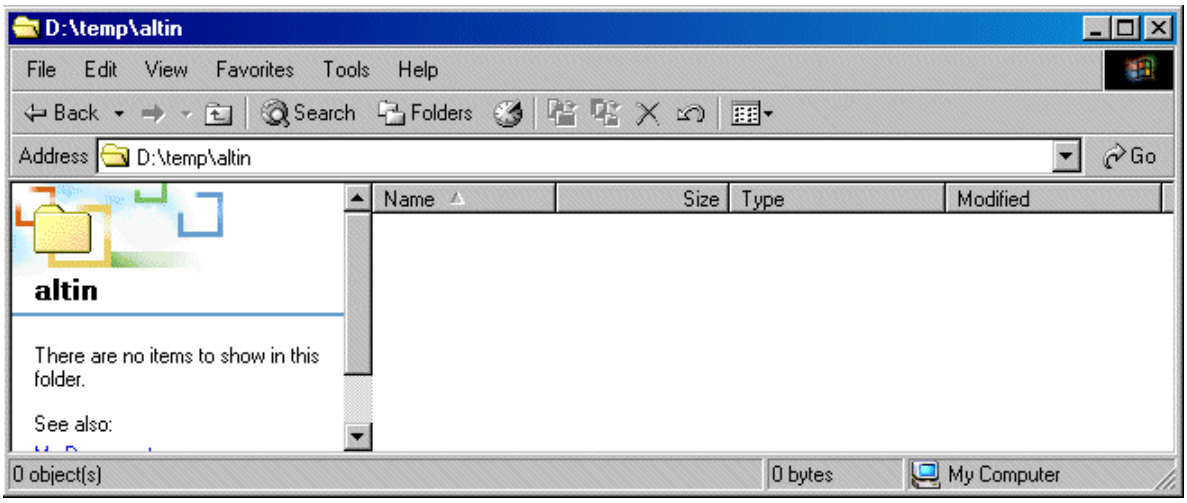
Next, we build a new task to download the trigger file, run the process and upload the data file.

§ Either the source of the trigger file or destination of the data file can be a remote machine (FTP, SSH, MOVEit)
§ However, the original data file must be on a local drive.
§ Select the "Delete original file(s)" check box on the source. Trigger files should be destroyed as soon as they force an upload.

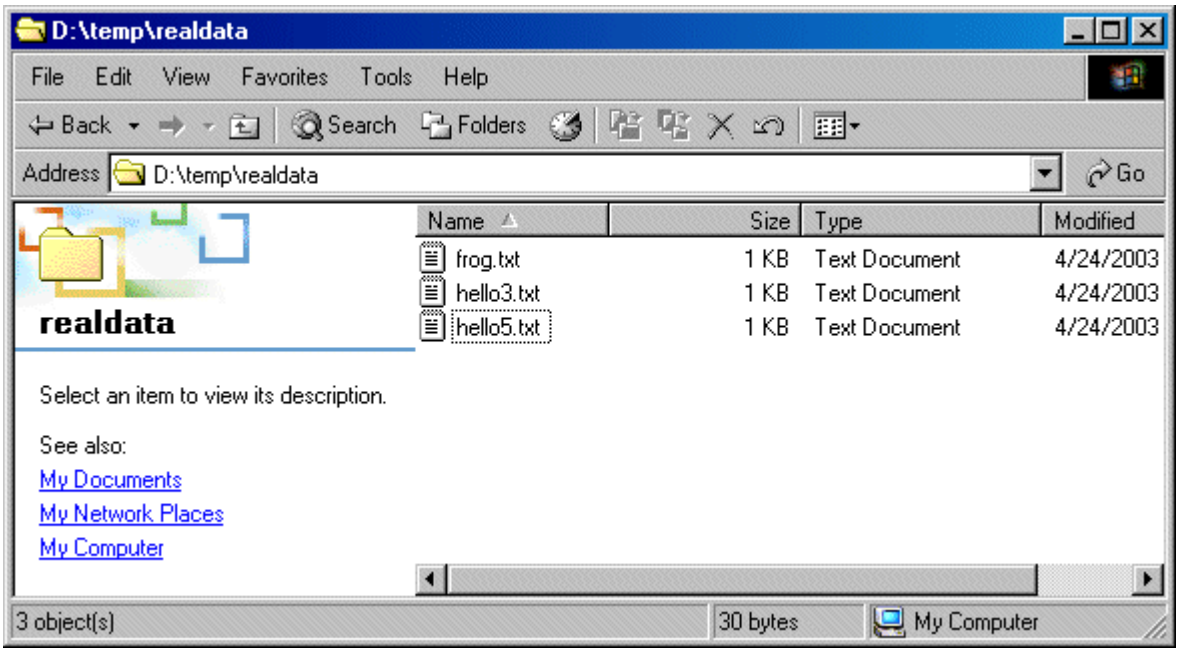


A critical addition to the task is the use of a Task Parameter to indicate where to find the data files. Be sure to add a Task Parameter called "DataPath" to your task with the full path of your data folder.

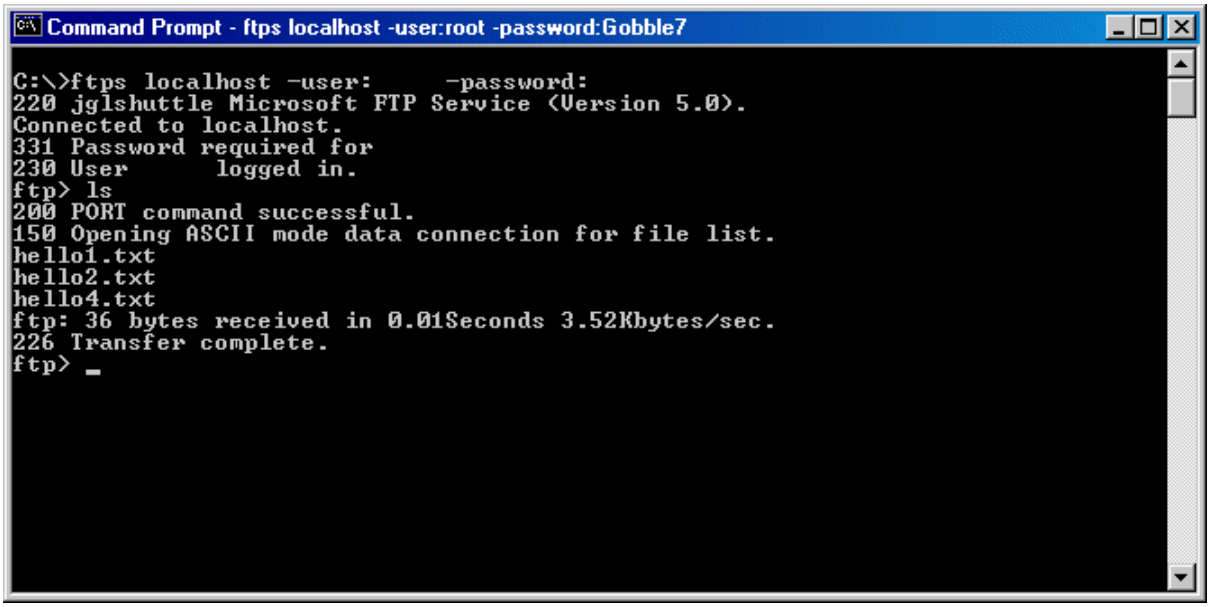
Run the task. The trigger files disappear.



Also notice that only the three selected data files are gone.



Finally, notice that the data files are where they are supposed to be on the destination. (Download one or two manually to make sure!)



GetNamedFile.vbs

```
Dim FileName, FilePath, FullPath

FileName = MIMacro("[OrigName]")
DataPath = MIGetTaskParam("DataPath")
FullPath = DataPath & "\" & FileName

MIReplaceCacheFile(FullPath) MIDeleteFileSecure(FullPath)
```

"Last Day Of Month" Schedules

While MOVEit Automation does not directly support scheduling task runs on the last day of the month, the Date Lists feature can be used to provide equivalent functionality. A date list can be set up containing all "end-of-the-month" dates for a period of years, and used as the schedule date for a task. A list of all "end-of-the-month" days for the period of June, 2017 through December, 2020, is included below. To use it, create a new date list in MOVEit Automation Admin Console, and then copy the list below and paste it into the Entries text box.

Another way to provide end-of-month date lists is to use the wildcard character *, instead of specifying dates for each year. This will allow the use of a 12-line date list, instead of 12 lines for each year. The drawback to this method, however, is that the date list will need to be modified for leap years. An example wildcard datelist is supplied below.

For information about creating and managing date lists, see the Date Lists page in this manual. For information about using date lists, see the Schedules page in this manual.

"End-Of-The-Month" Days for June, 2017 through December, 2020

2017-06-30
2017-07-31
2017-08-31
2017-09-30
2017-10-31
2017-11-30
2017-12-31
2018-01-31
2018-02-28
2018-03-31
2018-04-30
2018-05-31
2018-06-30
2018-07-31
2018-08-31
2018-09-30
2018-10-31
2018-11-30
2018-12-31
2019-01-31
2019-02-28
2019-03-31
2019-04-30
2019-05-31
2019-06-30
2019-07-31
2019-08-31
2019-09-30
2019-10-31
2019-11-30
2019-12-31
2020-01-31
2020-02-29
2020-03-31
2020-04-30
2020-05-31
2020-06-30
2020-07-31
2020-08-31
2020-09-30
2020-10-31
2020-11-30
2020-12-31

Wildcard "End-Of-The-Month" Days for Non-Leap Years

*-01-31
*-02-28
*-03-31
*-04-30
*-05-31
*-06-30
*-07-31
*-08-31
*-09-30
*-10-31
*-11-30
*-12-31

Wildcard "End-Of-The-Month" Days for Leap Years

*-01-31
*-02-29
*-03-31
*-04-30
*-05-31
*-06-30
*-07-31
*-08-31
*-09-30
*-10-31
*-11-30
*-12-31

Converting EBCDIC Text to ASCII Text

Mainframes often use a 256-bit character set called EBCDIC rather than the 128-bit ASCII character set most often used on Windows, Mac and UNIX platforms. MOVEit Automation can be used to convert text files from ASCII to EBCDIC or from EBCDIC to ASCII using the "CommandLineApp" built-in script and a command-line utility called "ebc2asc".

To convert EBCDIC text files to ASCII (or ASCII to EBCDIC) using MOVEit Automation:

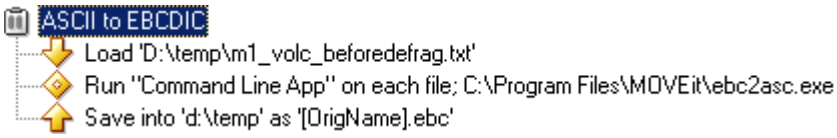
1. Sign on to the ***Ipswitch Community https://community.ipswitch.com/s/*** and download "ebc2asc.exe" from ***Converting EBCDIC text to ASCII text https://community.ipswitch.com/s/article/ka036000000kQR2AAM/Converting-EBCDIC-Text-to-ASCII-Text*** into your "C:\Program Files\MOVEit" folder *(or other appropriate local folder)*.
2. Set up a file transfer task (one source and one destination) to download and save our original text file without converting it. If your source is a mainframe accessible via FTP, you may need to specify "BINARY" mode and you may also need to specify additional "QUOTE" commands to complete the transfer.
3. After you have the basic file transfer task working, add a "Command Line App" built-in process.
4. Configure "Command Line App" with the following parameters:
   - CommandLineApp_AppPath = C:\Program Files\MOVEit\ebc2asc.exe (or the actual location of this file)

- § CommandLineApp_OutputFile = Yes
- § EBCDIC to ASCII only: CommandLineApp_AppParms = <"[InputFile]" >"[OutputFile]"
- § ASCII to EBCDIC only: CommandLineApp_AppParms = -a <"[InputFile]" >"[OutputFile]"

5 Save these parameters and test. When checked, EBCDIC files should be unreadable using Notepad.exe, but ASCII files should be readable using Notepad.exe.

Sample Task - ASCII to EBCDIC

This task loads a local ASCII file (output from a defrag check) and converts it to an EBCDIC file.

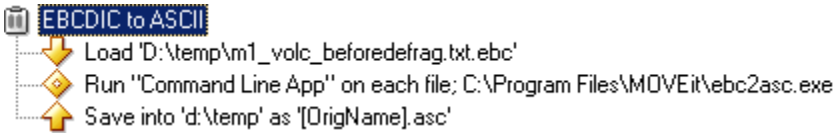


These are the CommandLineApp parameters. Notice the "-a" argument.

CommandLineApp_AppParms	-a <"[InputFile]" >"[OutputFile]"
CommandLineApp_AppPath	C:\Program Files\MOVEit\ebc2asc.exe
CommandLineApp_OutputFile	Yes

Sample Task - EBCDIC to ASCII

This task loads a local EBCDIC file (the EBCDIC file created in the previous step) and converts it back to an ASCII file.



These are the CommandLineApp parameters. Notice there is no "-a" argument.

CommandLineApp_AppParms	<"[InputFile]" >"[OutputFile]"
CommandLineApp_AppPath	C:\Program Files\MOVEit\ebc2asc.exe
CommandLineApp_OutputFile	Yes

HTTP Uploads and Downloads

MOVEit Automation supports uploading to and downloading from webservers using the HTTP and HTTPS protocol. This is in addition to the support for MOVEit Transfer hosts, which also use the HTTP and HTTPS protocols, but require a specific upload/download format.

Support for this capability comes in the form of built-in scripts:

- ***HTTP Get*** (on page 114) - downloads a file from a webserver using the HTTP GET verb.
- ***HTTP Post*** (on page 114) - uploads a file to a webserver using the HTTP POST verb.
- ***HTTP Put*** (on page 115) - uploads a file to a webserver using the HTTP PUT verb. This mechanism is not available on many websites, but it is preferable to POST because it is easier to configure.
- ***HTTP SharePoint Get*** (on page 116) - downloads a file from a Microsoft SharePoint Server webserver.
- ***HTTP SharePoint Put*** (on page 116) - Uploads a file to a Microsoft SharePoint Server webserver.

When configuring a task to access a webserver, use one of the above scripts as a Process step instead of adding a Source or Destination.

Not all webservers can be successfully accessed via these scripts. Some websites require complex, human-oriented navigation which is beyond the scope of these scripts.

MessageWay Translation

If your organization sends documents in EDI (Electronic Data Interchange) format, you may also need to transform the data in these documents to match a format used by a trading partner. In EDI terms, this is known as a translation. If you use MessageWay and the MessageWay Translator from Ipswitch to do these translations, then you can incorporate the translation workflow into a MOVEit Automation task.

Using the built-in script `MessageWay Translation`, you can set up a task that gets files from a source location, sends them to the MessageWay Translator, and puts the translated files in a destination. This topic provides an example of how you set up a task using the built-in script.

- For more information about the script, see ***MessageWay Translation*** (on page 123).
- For more information about configuring MessageWay, see *MessageWay User's Guide and Reference http://docs.ipswitch.com/MessageWay/MessageWay61/Manuals/MessageWayUserGuideandReference.pdf*. For more information about configuring and testing a translation, see *MW Translator Workbench User's Guide and Reference* and *MW Translator Workbench Tutorial*.

Task that Runs a Translation

This example shows how to set up a task that runs a translation. It translates a test document X12 850 (Purchase Order) to a proprietary fixed format document and generates an acknowledgement (X12 997). This example uses files installed with the MessageWay installation.

The example assumes that a MessageWay environment, including the MessageWay Translator, is already configured and has the following user and location information:

- user name and password: micentral
- location name for the MWTranslator service: translate
- location name for pickup mailbox: moveit

To translate an EDI document using MOVEit Automation and MessageWay, in this example, do the following in MOVEit Automation:

1. Set up a file transfer Advanced Task (one source and one destination) to download and save the X850TEST file without translating it.
2. Once you have the basic file transfer task working, add a "MessageWay Translation" built-in process.
3. Configure "MessageWay Translation" with the following parameters:
   - MWayConn_Host: the IP address or host name of the MessageWay server
   - MWayConn_User: micentral (a MessageWay user)
   - MWayConn_Password: password associated with the micentral MessageWay user
   - MWayConn_Recipient: translate:moveit (a compound address that represents the default MessageWay service location plus the MessageWay pickup mailbox)
   - MWayConn_Sender: X850Test
4. Run the task to test the translation.

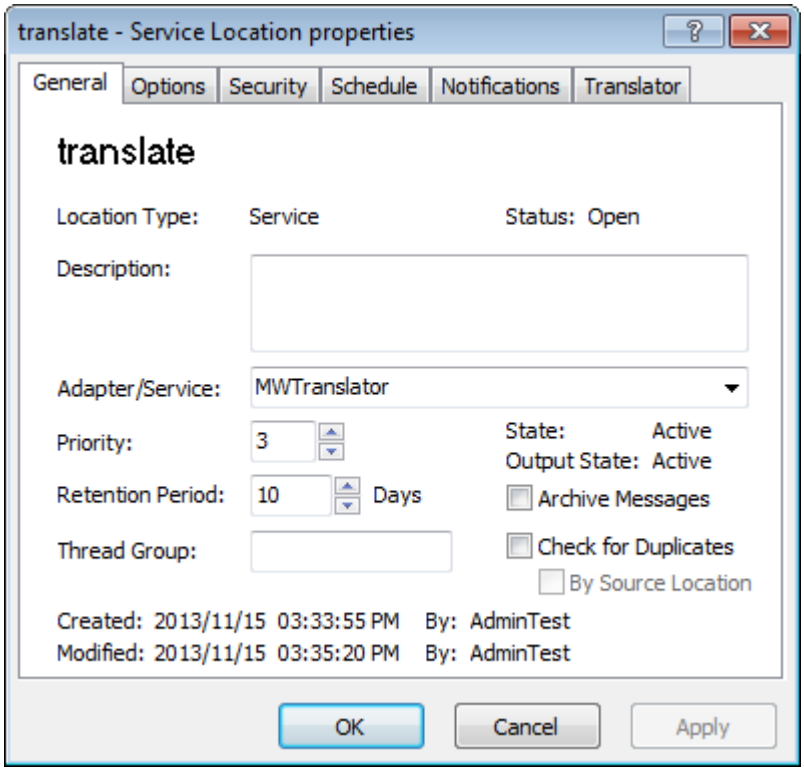
Sample Task

This sample task shows our example using the X850TEST files that are installed with MessageWay, and are described in the "MessageWay Installation Guide." Translation of EDI documents require an associated trading partner and map to be configured in the MessageWay Translation Workbench. This configuration is already present for the X850TEST, but needs to be done for each type of document to be translated. See the "MessageWay Translator Workbench User's Guide and Reference" for more information about configuring the environment.

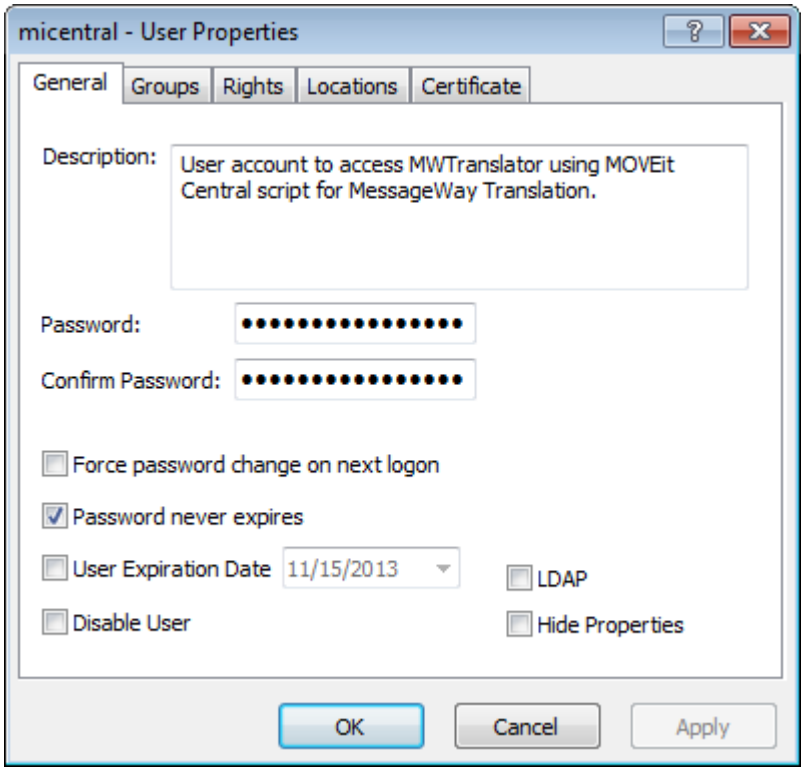
MessageWay Setup

Before you create the task in MOVEit Automation, configure the following items in the MessageWay environment:

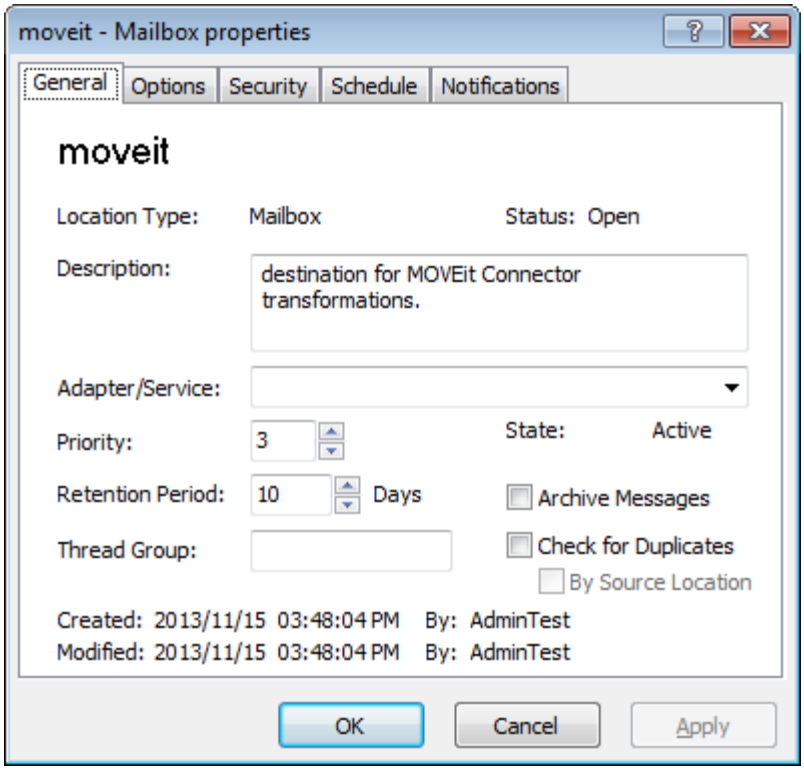
§ A default Service location for the MWTranslator service named translate. The Security tab must show the Administrator group.



§ A MessageWay user named **micentral** and associated password, which the MOVEit Automation script uses to log in to MessageWay. The user must have appropriate rights. For information about assigning rights to remote users, see *MessageWay User's Guide and Reference*. The Locations tab for this user must have an entry for **translate**, the default Service location.

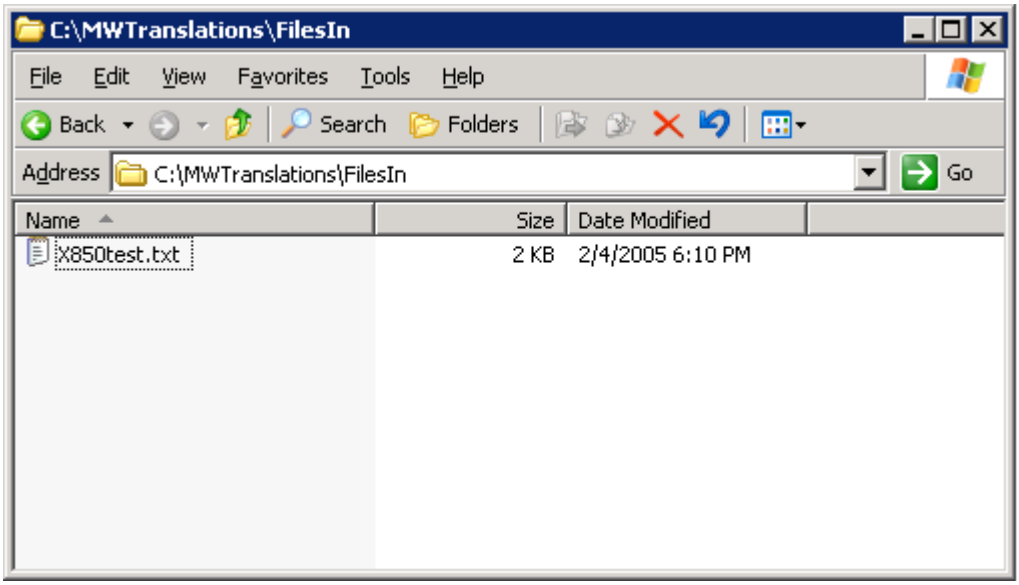


§ A pickup mailbox named **moveit**. The Security tab must show the Administrator group.



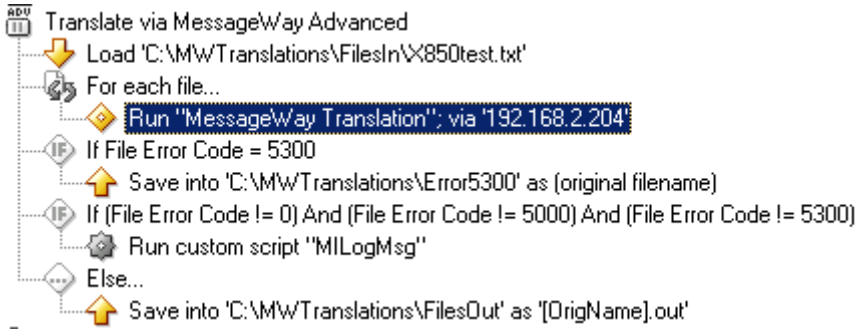
MOVEit Central Setup

Next, create an Advanced Task in MOVEit Automation to send the X850TEST.txt file for translation. The source location is: `c:\MWTranslations\FilesIn\X850Test.txt`



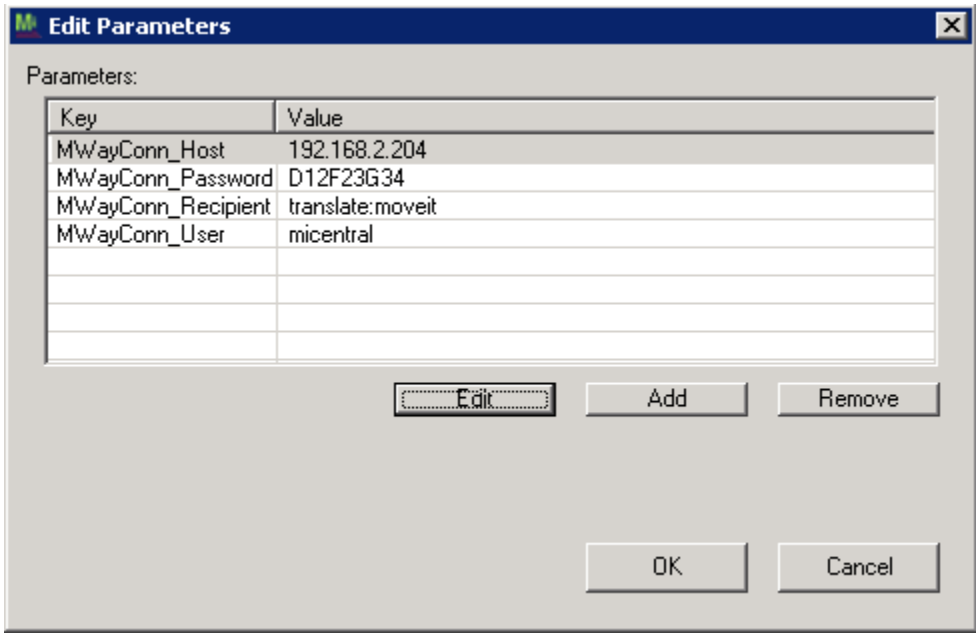
Note: The X850TEST.txt file is located the host where you installed MessageWay, in the folder: C:\MessageWay\Server\MWTranslator

Create an Advanced Task named Translate Via MessageWay. The complete task is shown here:

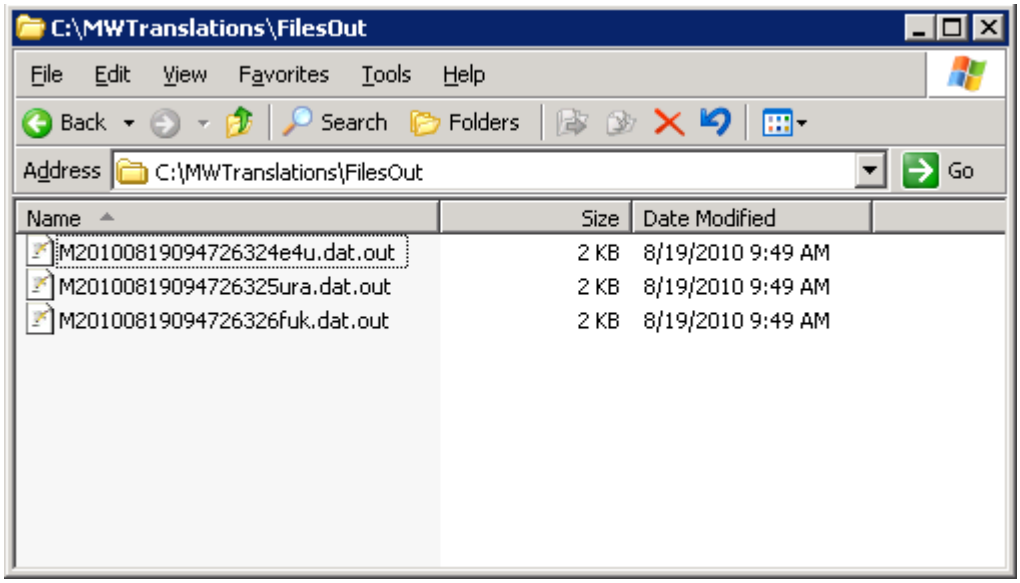


There is a source file named C:\MWTranslations\FilesIn\X850Test.txt. In the Advanced Task, this file is defined as the source.

The FOR file loop runs the MessageWay Translation process, selected from the list of built-in scripts. The parameters needed for MessageWay are defined in the built-in script. The example shows the required parameters. For a description of all parameters, see the topic ***Configuring Tasks - Processes/Scripts - Built-In - MessageWay Translation*** (see "***MessageWay Translation***" on page 123).



The MessageWay Translation process sends the source file(s) to the MessageWay Translator. If an error does not occur, the translated files are received back and placed in the destination: C:\MWTranslations\FilesOut



If a translation error (error code 5300) occurs, this means that some of the files were successfully translated. In this case, the resulting files are sent to the destination: c:\MWTranslations\Error5300. Note that the MessageWay Translation script differs from other tasks in allowing a partially successful outcome. You can set up the task to accept partially successful translation, or to reject the translation and receive a report. For more information, see the description of the MWayConn_ExceptionsInsteadofData parameter in the MessageWay Translation built-in script.

If an error other than code 5000 or 5003 occurs, it is logged in the location specified by the custom script MlLogMsg.

S/MIME Email

See also ***Advanced Topics - MessageWay CLI*** (on page 278)

Overview

S/MIME Email is a standards-based method for sending and receiving secure, verified email messages. It involves using public/private-key based certificates to encrypt and/or sign an email message, so that only the recipient of the email can open it (if encrypted), and the recipient knows with a high degree of certainty who sent the message (if signed).

MOVEit Automation ships with S/MIME support, but the actual S/MIME operations occur in one or two pre-qualified scripts. Likewise, S/MIME parameters are expressed as task-level parameters which control qualified scripts rather than source or destination options.

How Does S/MIME Work?

Encrypting, signing, and decrypting S/MIME email messages requires the use of certificates. Certificates are simply public and/or private keys wrapped up in a specific format, so that they can be used together and understood by various programs. The Email Architect tools rely on Microsoft Windows Certificate Stores to contain and manage the various certificates that may be used to create and receive S/MIME emails.

An S/MIME email message can be signed, encrypted, or both. Encrypting a message is done using the public key certificate of the recipient of the message. This ensures that only the recipient can decrypt the message, as the encryption is done so that only the recipient's private key certificate can reverse the encryption. Signing the message is done with the sender's private key certificate, and ensures to the recipient that the sender of the message is who they say they are. A hash of the message is also created by the signing process so that the recipient of the message knows that the message has not been changed since it was written.

As with PGP encryption/decryption, some amount of key exchange is required. In order to encrypt a message to a given recipient, the sender must have a copy of the recipient's public key certificate. This is generally accomplished by having the recipient send the sender a signed S/MIME email message. S/MIME signatures are done in such a way that the sender's public key certificate can be extracted and stored for later use. Most modern email clients, including Microsoft Outlook Express and Mozilla Email Client will automatically recognize a signed message, extract the public key certificate, and store it for examination and later use.

Configuring Certificates

Before MOVEit Automation can be used to exchange S/MIME email, the certificates of both the sender and the recipient must be obtained and stored correctly.

Personal Certificate

The personal certificate is used to sign outgoing messages and decrypt incoming messages. Personal certificates can be obtained from most Certificate Authority companies, such as ***Thawte*** (***https://www.thawte.com/assets/documents/guides/pdf/print/personal_cert.pdf***). The personal certificate contains both a private and public key, and is generally given out based on email address.

A personal certificate should be obtained in a PKCS12 format, which allows both the public and private keys to reside in the same password-protected file. This file will usually have an extension of either .pfx, or .p12.

The certificate should be stored in the Personal certificate store of the local user that MOVEit Automation is running under. To import the certificate, use the SSL Client Certificates menu option in MOVEit Automation Admin.

Other Certificates

Other certificates are used to encrypt outgoing messages to a specific user. Other certificates are the public-key half of the other party's personal certificate. They are available from the other party. Usually, the other party send a signed S/MIME message to the current user to provide the other certificate.

The public-key certificate should be obtained as an X.509 certificate, which may be encoded in a binary format (DER) or a text format (base-64). Both encoding formats are usually stored in a file with an extension of .cer. The certificate should then be stored in the Other People certificate store of the local user that MOVEit Automation is running under.

To import the certificate into Windows, use the SSL Client Certificates menu option in MOVEit Automation Admin.

Sending and Receiving

See the ***SMIME Receive*** (on page 131) script for details. You can use the built-in S/MIME scripts to send or receive encrypted and/or signed S/MIME email messages with source files as attachments.

Before you begin, the following prerequisites are in place:

- The current user's personal certificate and all other required party certificates are configured.
- MOVEit Automation is configured to run under the proper user code that has access to the required certificates.

Ø *To send an encrypted and/or signed S/MIME email messages with source files as attachments, complete the following steps.*

1. Create a new task and add a source to load the file(s) that you want to send in S/MIME signed and/or encrypted emails.
2. Add a process to the task that references the built-in SMIME Send script, and configure the task parameters accordingly. See the ***SMIME Send*** (on page 132) script for details.

Ø *To receive an encrypted and/or signed S/MIME email messages with source files as attachments, complete the following steps.*

1 Create a new task and add a destination to process the file(s) that will be received as attachments to incoming S/MIME emails.

AS1, AS2, AS3

2 Add a process to the task that references the built-in SMIME Receive script, and configure the task parameters accordingly. See the ***SMIME Receive*** (on page 131) script for details.

Overview

Approaching the AS1, AS2 and AS3 protocols (collectively referred to as "ASx" herein) can be a daunting experience for someone without any file transfer experience. On the other hand, someone with an understanding of the transport protocols (SMTP, POP3, HTTP, FTP and SSL/TLS) and/or public-key/private-key encryption (such as SMIME or PGP) should find some familiar ground.

ASx Protocols Provide Similar File Encryption to PGP, SMIME or Other Encrypted File Methods

The ASx protocols use a subset of SMIME ("Secure MIME") to sign and encrypt files. SMIME is a public/private-key encryption technology based on SSL certificates. Many email clients implement SMIME to encrypt and decrypt email messages and attachments, but the ASx implementation of SMIME is specific enough to consider ASx clients and SMIME-enabled email clients incompatible.

Similar public/private-key technology can be found in OpenPGP (which uses simple keys rather than SSL certs to sign and encrypt) and many lesser-known technologies such as "strongly authenticated, encrypted zip files". However, any public/private-key implementation which does not carry an AS1, AS2 or AS3 mark should be considered incompatible with ASx technology because the ASx protocols are quite strict.

ASx Protocols Provide Superior Receipts to PGP, SMIME or Other Encrypted File Methods

The primary advantage of ASx over other public/private key file encryption schemes is that the ASx protocol includes an "MDN" receipt mechanism that proves to the sender that the designated recipient of an ASx message actually received and decrypted the message and verified the identity of the sender.

An MDN ("message disposition notification") receipt is a direct extension of the "delivery receipts" you may have seen or used in your favorite email client. Under AS1 MDNs may be returned via email like any other delivery receipt, but under AS2 and AS3, MDNs may be returned via the HTTP and FTP protocols, respectively.

Regardless of the actual ASx protocol used, each MDN can be cryptographically signed by the recipient of an ASx message and lets the original ASx message sender know the recipient performed three important actions:

- Received the message
- Decrypted the message (if the ASx message was encrypted)
- Validated the sender's signature (if the ASx message was signed)

MDNs also provide cryptographic hash information about the file sent so the sender can verify that the recipient actually received the file the sender thought they sent.

ASx Protocols Provide Standards-Based "Non-Repudiation", "Pedigrees", etc.

In the file transfer arena, "non-repudiation" is a term that means someone can prove who sent a file, who received a file and that the contents of the file were not changed between sender and recipient. In practice, "who sent" and "who received" are questions answered by authentication credentials ranging from usernames and passwords to certificates and keys. The "not changed" question is almost always answered by a cryptographic-quality hash.

However, there is really a fourth piece to the "non-repudiation" puzzle: the record of the "who sent", "who received" and "not changed" itself. When this information is retained in traditional logs, those logs must be made tamper-evident through cryptographic technology (such as that included in MOVEit products). ASx provides an alternate "non-repudiation" technology through standards-based MDNs: each MDN is a "non-repudiation" receipt for a single file.

In MOVEit Automation, MDNs are retained in the MOVEit Automation tamper-evident audit database from which they may be examined and exported at any time.

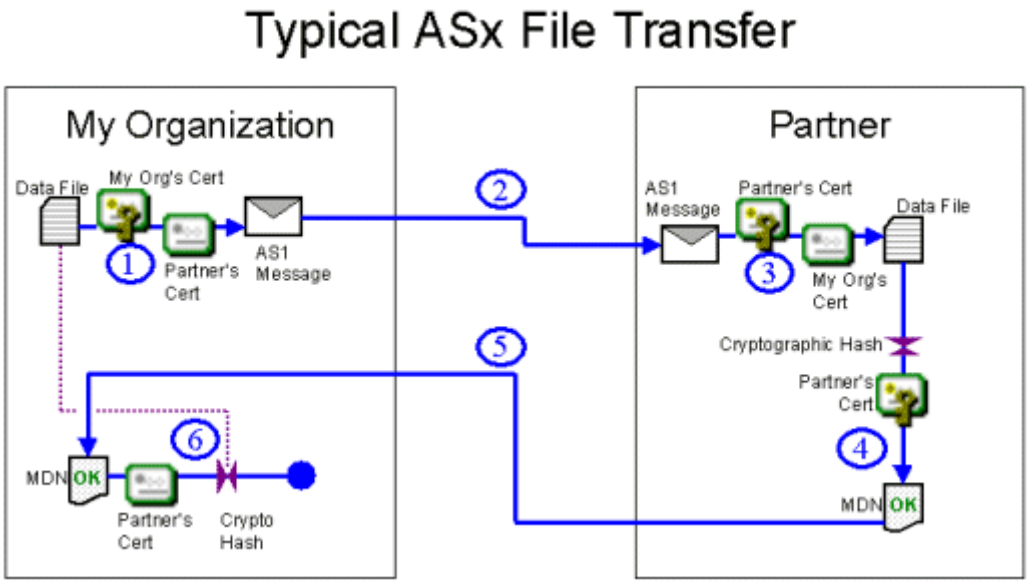
Again, in the file transfer arena, if a series of file transfers all had characteristics of non-repudiation, it used to be common to refer to this situation as "end-to-end file non-repudiation" or an "unbroken chain of non-repudiation." More recently (and thanks largely to new regulatory requirements) this same concept has been described as a "file pedigree".

In either case the end result is the same: if you can prove who sent a file over each hop, who received that file over each hop and that the contents of the file were not changed between original sender and final recipient, you have all the necessary elements to satisfy "end-to-end file non-repudiation", "file pedigree", etc.

Proper implementation of the MOVEit ASx protocols will give you "end-to-end file non-repudiation", "file pedigree", etc.

(Notice that "encryption" is NOT an element of file non-repudiation. Many operational requirements ask for encryption on only part of a file delivery chain because some internal process needs to examine and possible modify a file along the way; MOVEit Automation is often involved in this kind of processing step.)

AS1, AS2 and AS3



All of the ASx protocols can:

1. Encrypt a file using a recipient's public SSL certificate and sign the file using the sender's private SSL certificate
2. Specify the type and manner of MDN that the recipient should return
3. Deliver the file to a partner
4. Decrypt a file using a recipient's private SSL certificate and confirm the signature of the sender using the sender's public SSL certificate
5. Create an MDN delivery receipt signed with the recipient's private SSL certificate and containing a cryptographic hash of the file contents in order to prove that the recipient got the unaltered file

6 Return the MDN to the sender

7 Verify the MDN (against the recipient's public SSL certificate and the cryptographic hash) to absolutely prove that the recipient got the file

See also (on their respective "AS1, AS2 and AS3 - The ASx Protocol" pages):

§ ***How an AS1 File Transfer Works*** (on page 209)
§ ***How an AS2 File Transfer Works*** (on page 215)
§ ***How an AS3 File Transfer Works*** (on page 223)

In addition, all of the ASx protocols treat a single file as a single message. Unless you explicitly zip or otherwise bundle multiple files together before an ASx operation, each file will be sent individually and each will be paired up with its own MDN later.

The difference between the AS1, AS2 and AS3 protocols is really the different TRANSPORT protocol each one uses to send messages and receive MDNs.

§ AS1: Email
§ AS2: HTTP(S)
§ AS3: FTP(S)

The MDNs for AS2 transfers come in three varieties. In the synchronous HTTP(S) type, MDNs are transmitted using the same connection as the original file upload. In the asynchronous HTTP(S) type, MDNs are transmitted using a different web upload session. In the asynchronous email type, MDNs are transmitted via email, just as in AS1 MDN transmissions.

AS1 was developed first, followed by AS2 and AS3. AS2 did not completely detach itself from AS1 in one respect: asynchronous email MDNs. Remember that the file sender always controls how its MDN is returned from the recipient.

Some of the primary attributes, advantages and disadvantages of the three ASx protocols are summarized in the table below.

Comparison of AS1, AS2 and AS3

	AS1	AS2			AS3
File Transport	Email	HTTP / HTTPS			FTP / FTPS
MDN Transport	Email	HTTP/HTTPS Synchronous	HTTP/HTTPS Asynchronous	Email Asynchronous	FTP / FTPS
Requires special ASx server (receiver)	NO	yes	yes	yes	NO
Requires special ASx server (sender)	NO	NO	yes	NO	NO
Non-repudiation for large files	YES	no	YES	YES	YES
Firewall friendly	YES	YES	YES	YES	no
Widely supported	no	YES	YES	no	YES
MDN is available immediately	no	YES	no	no	no
Two-factor transport authentication	YES	no*	no*	no*	YES
"Desktop" clients can ___ files	SEND, RECEIVE AND VERIFY	Send and Verify Only	Send Only	Send and Verify Only	SEND, RECEIVE AND VERIFY

* some AS2 servers offer "basic" username/password authentication, but most AS2 clients do not support it

See also (on their respective "AS1, AS2 and AS3 - The ASx Protocol" pages):

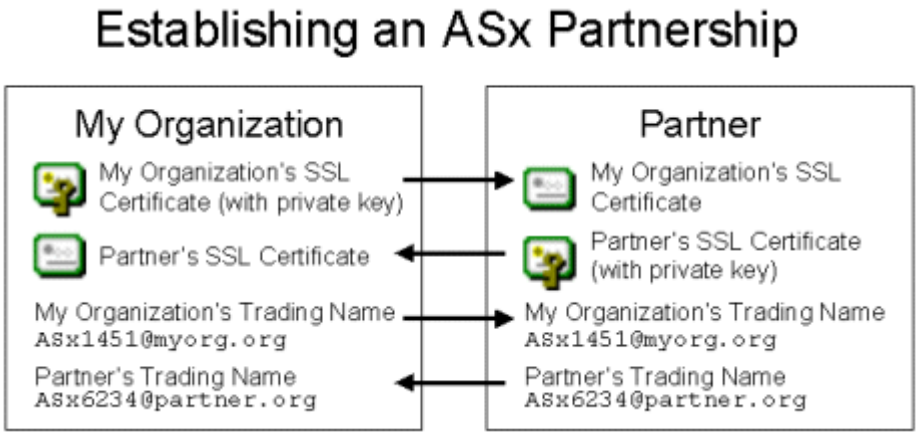
§ ***Advantages/Disadvantages of AS1*** (on page 213)
§ ***Advantages/Disadvantages of AS2*** (on page 222)
§ ***Advantages/Disadvantages of AS3*** (on page 227)

Identifying My Organization and a Partner

ASx transfers are "1-sender, 1-recipient" affairs. Although support for multiple recipients is common in similar encryption schemes such as PGP and SMIME, ASx transfers do not support this concept.

Most ASx products define the two sides in a file exchanges as "my organization" (i.e., you) and "your partner" (i.e., anyone else). Both sides are responsible for coming up with at least two pieces of information: an SSL certificate and their trading partner name. (In practice, one side may provide all of this information.)

The two sides must exchange their information (minus the private keys on their own SSL certificates) and agree on which certificates and trading names before any ASx file transfers may take place.



Each AS1, AS2 or AS3 protocol will also require each side to agree on additional protocol-specific items such as which server to send files too, what credentials to use to authenticate to the server and what flavors of MDNs are supported. Additional information about these specifics is covered in each ASx protocol's discussion.

Optional Elements

If everyone always signed and encrypted their ASx messages, requested signed MDNs and used SSL encryption when transporting ASx messages there would still be plenty of options to configure. However, almost every element discussed so far is really an optional element. Specifically, the following configuration items are among those considered optional in the ASx specifications. (You can probably guess - hint: "Y" - what the best practice values are.)

- Sign message? (Y/N)
- Encrypt message? (Y/N)
- Request MDN? (Y/N)
- If requested, request a signed MDN? (Y/N)
- Use SSL transport encryption while sending the message? (Y/N)
- Use SSL transport encryption while sending the MDN? (Y/N)

Limitations of the ASx Protocols

When used properly, ASx protocols solve a number of traditionally vexing secure file transfer issues, but they do not solve all problems. Some of the cases that require additional thought and planning are described below.

ASx's "Two-Way Handshake" Does Not Let Receiver Know Sender Got MDN

As described above, properly configured MDNs provide a high degree of non-repudiation. The sender knows that the recipient got his/her file, and the recipient knows that he/she is looking at an exact copy of the original content. However, the recipient never knows for sure whether the sender received or verified a requested MDN.

TCP networking uses a "three-way" handshake to avoid a similar problem. The three handshakes in TCP are:

1. Client sends a "SYN" to the server to ask for a connection.
2. Server sends an "ACK" packet back to the client to confirm the connection and also sends an "ACK" to the client to confirm opening the connection.
3. Client sends an "ACK" back to the server to confirm that the client knows the connection is open.

The ASx protocols specify only two of three possible "handshakes": an ASx file recipient never finds out what the file sender thinks of the MDN the file recipient created. This limitation can lead to several issues:

- ASx file recipients must retain MDNs of any ASx message that requested one unless the recipient can absolutely not deliver the MDN.
- "Duplicate posts" are possible if an ASx sender is set to resend files until a valid MDN is received and an ASx recipient believes that it has successfully posted a valid MDN back to a sender's server.

ASx MDNs Represent Handoffs of Responsibility, Not Fitness

The ASx protocols require that MDNs get sent as soon as an ASx message recipient can decrypt, validate the signature of and verify the contents of a data file.

In other words, after an MDN has been successfully sent, it is now the recipient's sole responsibility to not lose the decrypted file (or at least retain and be able to decrypt the original file at will). If internal processing or delivery errors crop up, they are the file recipient's sole responsibility and MDN technology can not be used to notify the sender about any data file format or content problems.

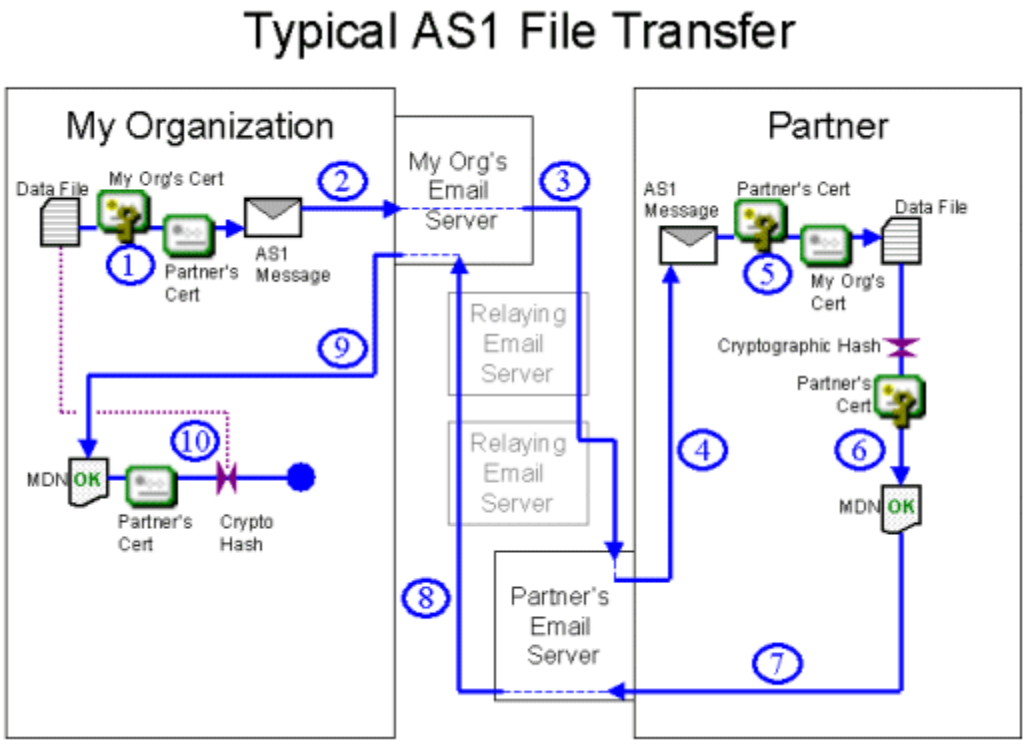
The AS1 Protocol

The AS1 protocol relies on email. It was the first ASx protocol developed and established the signing, encryption and MDN conventions used in later AS2 and AS3 protocols. It is probably the easiest ASx protocol to set up and work with, but it is rarely used.

- ***How an AS1 File Transfer Works*** (on page 210)
- ***MOVEit Automation Implementation of AS1*** (on page 212)
- ***Advantages/Disadvantages of AS1*** (on page 213)

How an AS1 File Transfer Works

Like any other ASx file transfer, AS1 file transfers typically require both sides of the exchange to trade SSL certificates and specific "trading partner" names before any transfers can take place. AS1 trading partner names must really be email addresses. (AS1 is the only ASx protocol that contains this requirement.)



1. You encrypt a data file with the public key on your partner's SSL certificate and sign it with the private key of your organization's SSL certificate as you bundle everything into an AS1 message. *(Both the encryption and signing steps are optional, but should be used when possible.)*
2. You send the AS1 message to an email server via SMTP. Often, this will be your local email server. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*
3. If the AS1 message was sent to your local email server, it will now deliver it to your partner's email server using the SMTP protocol. Along the way your AS1 message may traverse several intermediate email servers as it is relayed across the Internet or corporate email infrastructure. *(Cleartext message headers are rarely protected with SSL transport if relay servers are involved.)* This step will be skipped if the AS1 message was delivered directly to your partner's email server in step #2.
4. Your partner will retrieve your AS1 message off your partner's local email server using the POP3 protocol. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*
5. If the message is encrypted, your partner will decrypt it using the private key on his/her SSL certificate. If the message is signed, your partner will validate your signature using the public key on your SSL certificate. Your partner will also use the contents of the AS1 message to verify that the data file they now have is identical to the data file you sent them.

6 If you requested an MDN delivery receipt for your data file, your partner will calculate a cryptographic hash from the data file they received, sign the hash (and some other information) with the private key on their SSL certificate and create an MDN delivery receipt message. *(The signing step is optional and controlled by the original message sender.)*

7 Your partner will send his/her MDN delivery receipt message to an email server via SMTP. Often, this will be your partner's local email server. *(Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)*

8 If the MDN delivery receipt message were sent to your partner's local email server, it will now deliver it to your email server using the SMTP protocol. Along the way your partner's MDN delivery receipt message may traverse several intermediate email servers as it is relayed across the Internet or corporate email infrastructure. *(The cleartext MDN delivery receipt message is rarely protected with SSL transport if relay servers are involved.)* This step will be skipped if the MDN delivery receipt message was delivered directly to your email server in step #7.

9 You will retrieve your partner's MDN delivery receipt message off your local email server using the POP3 protocol. *(Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)*

10 You will inspect your partner's MDN delivery receipt message, making sure that you can verify his/her signature using the public key on your partner's SSL certificate and that the cryptographic hash calculated from your partner's copy of your data file matches the same hash calculation from your original data file.

Variations

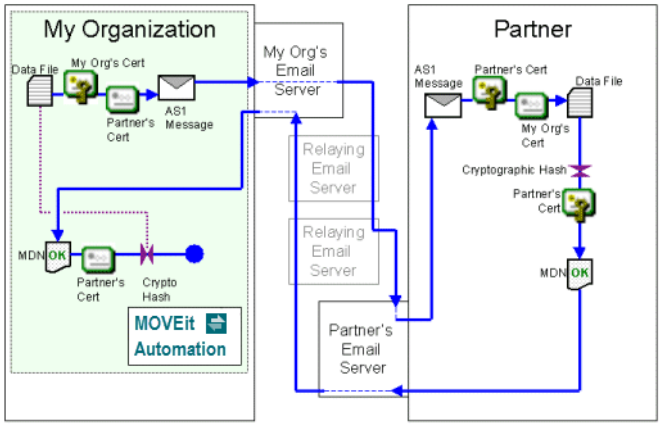
Direct connections to each mail server - AS1 clients can usually be configured to connect directly to your partner's email server rather than your local email server. Doing so avoids multiple "relay hops" but usually involves some additional firewall configuration.

Shared mail server - Your organization and your partner could agree to use different email accounts on the same physical email server, hosted at your data center, your partner's data center, or any other email server in the world. Doing so avoids multiple "relay hops" but usually involves some additional firewall configuration.

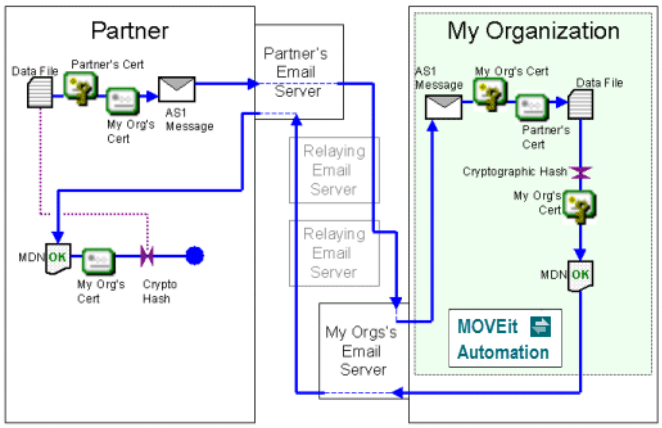
MOVEit Implementation of AS1

MOVEit Automation is the only MOVEit product required to send or receive files using AS1. In either case files and MDNs are sent through email servers. Because virtually all organizations already have access to an email server, there are no email servers bundled with MOVEit products.

AS1 File Send with MOVEit



AS1 File Receive with MOVEit



See also:

- AS1, AS2 and AS3 - Hosts - AS1
- AS1, AS2 and AS3 - Tasks - AS1 - Source
- AS1, AS2 and AS3 - Tasks - AS1 - Destination

Advantages/Disadvantages of AS1 (Compared to AS2 and AS3)

AS1 is the original ASx protocol. All of the file encryption and signing elements of ASx are present in this protocol, so the following discussion really concentrates on the SMTP/POP3 email protocol used to transport AS1 messages and MDNs receipts.

Advantages

Advantages of AS1	More information
If you have an AS1 client and access to a email server, you can send and receive AS1 transmissions	Nearly everyone connected to the Internet these days has access to an email server. (A user does not need to control or host the email server participating in an AS1 transmission). AS1 is the easiest of the ASx protocols to install and configure.
Conceptually, "SMIME messages" and "MDN receipts" fit well with AS1's email-based model.	If you have previously sent encrypted messages (with SMIME or PGP) and/or used delivery receipts, you already are familiar with the way AS1 works.
AS1 is firewall-friendly.	If you can send and receive email messages to and from the Internet, you can perform AS1 transfers (even if your only access is to a local email server). However, firewall issues will likely appear if you perform "direct-to-remote-server" AS1 transmissions, because most modern firewall rule sets permit only designated email servers to send messages to and from the internal network.

Disadvantages

Disadvantages of AS1	More information
Very few people use AS1.	The ASx protocols did not gain wide acceptance until AS2 was introduced; most people today use AS2 or AS3 instead of AS1.
Loss of control over email relay hops.	Typically, to send email, you send a message to a local email server. This server sends your message to another email server. Eventually, your message arrives at the receiver's email server, from which the message receiver can pull your message down and read it. Three common problems with this system of multiple email hops are § Transmission time is increased § SSL enforcement is possible only on the first (usually internal) hop § Your AS1 encrypted messages and signed MDNs can be copied and retained by any intermediate server. To avoid these problems some people have implemented direct-to-remote-server AS1 transmissions, but these configurations usually require firewall setups that lead them to consider other ASx protocols.

AS1 messages are lumped in with regular email.	In most situations AS1 messages are passed through traditional email servers, which means they are subject to attachment filters, size limits, spam filters, anti-virus filters, server downtime, message queues, spam surges and other email issues that people often do not want to involve in file transfers with their partners. ("Getting our file transmissions off the mail server" is why many companies set up a dedicated secure file transfer infrastructure in the first place.)

==========

- Advantage: If you have an AS1 client and access to a email server, you can send and receive AS1 transmissions. Nearly everyone connected to the Internet these days has access to an email server (you don't need to control or host the email server participating in an AS1 transmission), so AS1 is arguably the easiest of the ASx protocols to install and configure.
- Advantage: Conceptually, "SMIME messages" and "MDN receipts" fit well with AS1's email-based model. If you have previously sent encrypted messages (with SMIME or PGP) and/or used delivery receipts, you already have a pretty good feel for the way AS1 works.
- Advantage: AS1 is firewall-friendly. If you can send and receive email messages to and from the Internet , you can perform AS1 transfers (even if your only access is to a local email server). However, firewall issues will likely appear if you decide to perform "direct-to-remote-server" AS1 transmissions because most modern firewall rule sets only permit designated email servers to send messages to and from the internal network.
- Disadvantage: Very few people use AS1. The ASx protocols really did not gain wide acceptance until AS2 was introduced; most people today use AS2 or AS3 instead of AS1.
- Disadvantage: Loss of control over email relay hops. Typically, to send email, you send a message to a local email server. This server turns around and sends your message to another email server. Eventually, your message arrives at the receiver's email server, from which the message receiver can pull your message down and read it. Three common problems with this system of multiple email hops are 1) that transmission time is increased, 2) SSL enforcement is only possible on the first (usually internal) hop and 3) your AS1 encrypted messages and signed MDNs can be copied and retained by any intermediate server. To avoid these problems some people have implemented direct-to-remote-server AS1 transmissions, but these configurations usually require firewall setups that lead them to consider other ASx protocols.
- Disadvantage: AS1 messages are lumped in with regular email. In most situations AS1 messages are passed through traditional email servers, which means they are subject to attachment filters, size limits, spam filters, anti-virus filters, server downtime, message queues, spam surges and other email issues that people often do not want to involve in file transfers with their partners. ("Getting our file transmissions off the mail server" is why many companies set up a dedicated secure file transfer infrastructure in the first place.)

See also: ***Comparison of AS1, AS2 and AS3*** (on page 204) on the "AS1, AS2 and AS3 - Overview" page.

The AS2 Protocol

The AS2 protocol is based on HTTP. It was the second ASx protocol developed and uses the same signing, encryption and MDN conventions used in the original AS1 protocol. AS2 is the most popular of the ASx protocols but usually requires more work to set up than AS1 or AS3.

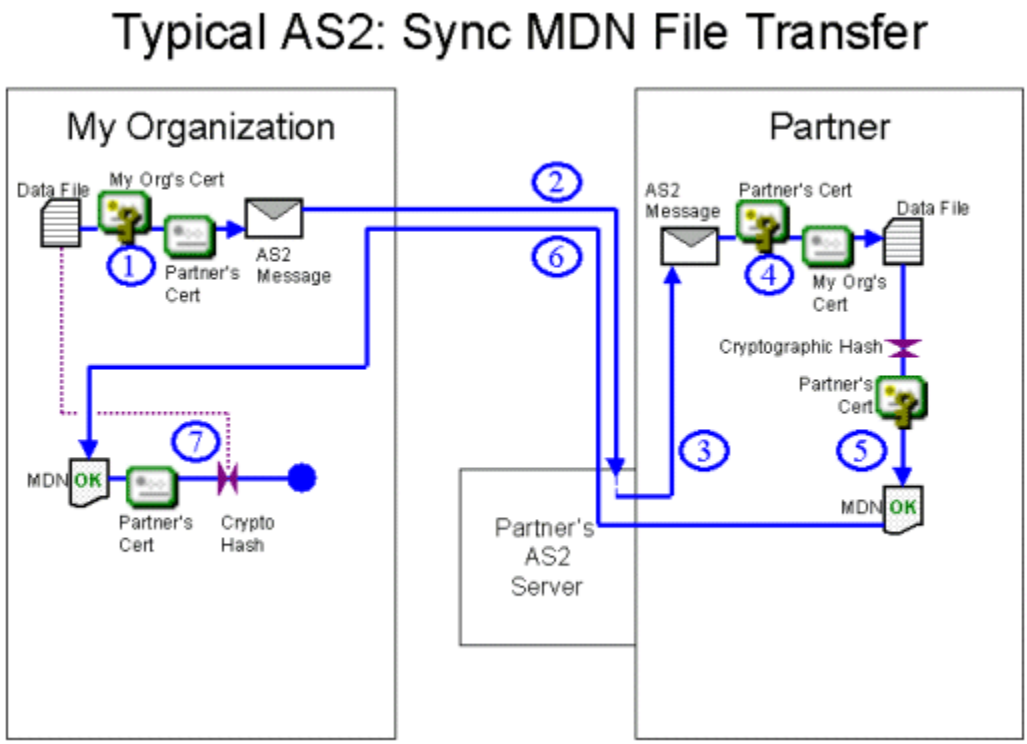
How an AS2 File Transfer Works

Like any other ASx file transfer, AS2 file transfers typically require both sides of the exchange to trade SSL certificates and specific "trading partner" names before any transfers can take place. AS2 trading partner names can be any valid phrase.

Unlike any other ASx file transfer, AS2 file transfers offer several "MDN return" options instead of the traditional options of "yes" or "no". Specifically, the choices are:

- Return Synchronous MDN via HTTP(S) (a.k.a. "AS2 Sync") - This popular option allows AS2 MDNs to be returned to AS2 message sender clients over the same HTTP connection they used to send the original message. This "MDN while you wait" capability makes "AS2 Sync" transfers the fastest of any type of ASx file transfer, but it also keeps this flavor of MDN request from being used with large files (which may time out in low bandwidth situations).
- Return Asynchronous MDN via HTTP(S) (a.k.a. "AS2 Async") - This popular option allows AS2 MDNs to be returned to the AS2 message sender's server later over a different HTTP connection. This flavor of MDN request is usually used if large files are involved.
- Return (Asynchronous) MDN via Email - This rarely-used option allows AS2 MDNs to be returned to AS2 message senders via email rather than HTTP. Otherwise, it is similar to "AS2 Async (HTTP)".
- Do not return MDN - This option works like it does in any other ASx protocol: the receiver of an AS2 message with this option set simply does not try to return an MDN to the AS2 message sender.

AS2 with Synchronous MDN via HTTP(S)



1 You encrypt a data file with the public key on your partner's SSL certificate and sign it with the private key of your organization's SSL certificate as you bundle everything into an AS2 message. (Both the encryption and signing steps are optional, but should be used when possible.)

2 You send the AS2 message to your partner's AS2 server AND WAIT UNTIL YOUR PARTNER RETURNS AN "MDN" RESPONSE. (Credentials and cleartext message headers may be protected with SSL transport in this step.)

3 Your partner will retrieve your AS2 message off his/her AS2 server. (Credentials and cleartext message headers may be protected with SSL transport in this step.)

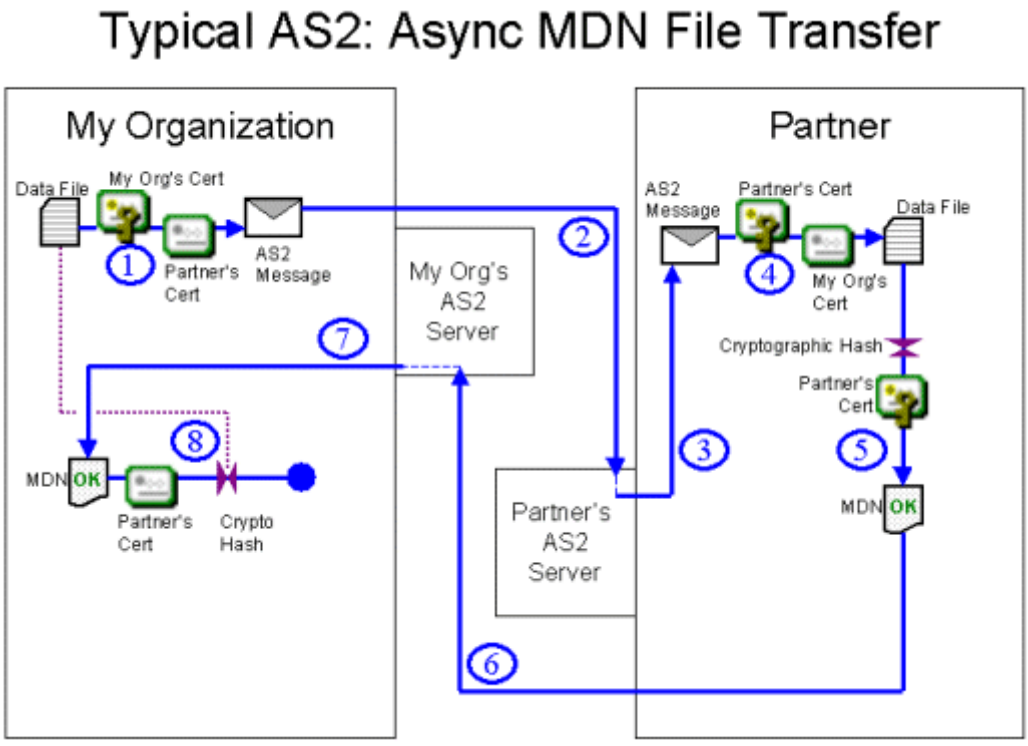
4 If the message is encrypted, your partner will decrypt it using the private key on his/her SSL certificate. If the message is signed, your partner will validate your signature using the public key on your SSL certificate. Your partner will also use the contents of the AS2 message to verify that the data file they now have is identical to the data file you sent them.

5 If you requested an MDN delivery receipt for your data file, your partner will calculate a cryptographic hash from the data file they received, sign the hash (and some other information) with the private key on their SSL certificate and create an MDN delivery receipt message. (The signing step is optional and controlled by the original message sender.)

6 Your partner will send his/her MDN delivery receipt message back as a response to your still-waiting AS2 "send message and get MDN" request. Once you receive this response (or time out while waiting) you will close your connection. (Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)

7 You will inspect your partner's MDN delivery receipt message, making sure that you can verify his/her signature using the public key on your partner's SSL certificate and that the cryptographic hash calculated from your partner's copy of your data file matches the same hash calculation from your original data file.

AS2 with Asynchronous MDN via HTTP(S)



1 You encrypt a data file with the public key on your partner's SSL certificate and sign it with the private key of your organization's SSL certificate as you bundle everything into an AS2 message. (Both the encryption and signing steps are optional, but should be used when possible.)

2 You send the AS2 message to your partner's AS2 server and close the connection. (Credentials and cleartext message headers may be protected with SSL transport in this step.)

3 Your partner will retrieve your AS2 message off his/her AS2 server. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*

4 If the message is encrypted, your partner will decrypt it using the private key on his/her SSL certificate. If the message is signed, your partner will validate your signature using the public key on your SSL certificate. Your partner will also use the contents of the AS2 message to verify that the data file they now have is identical to the data file you sent them.

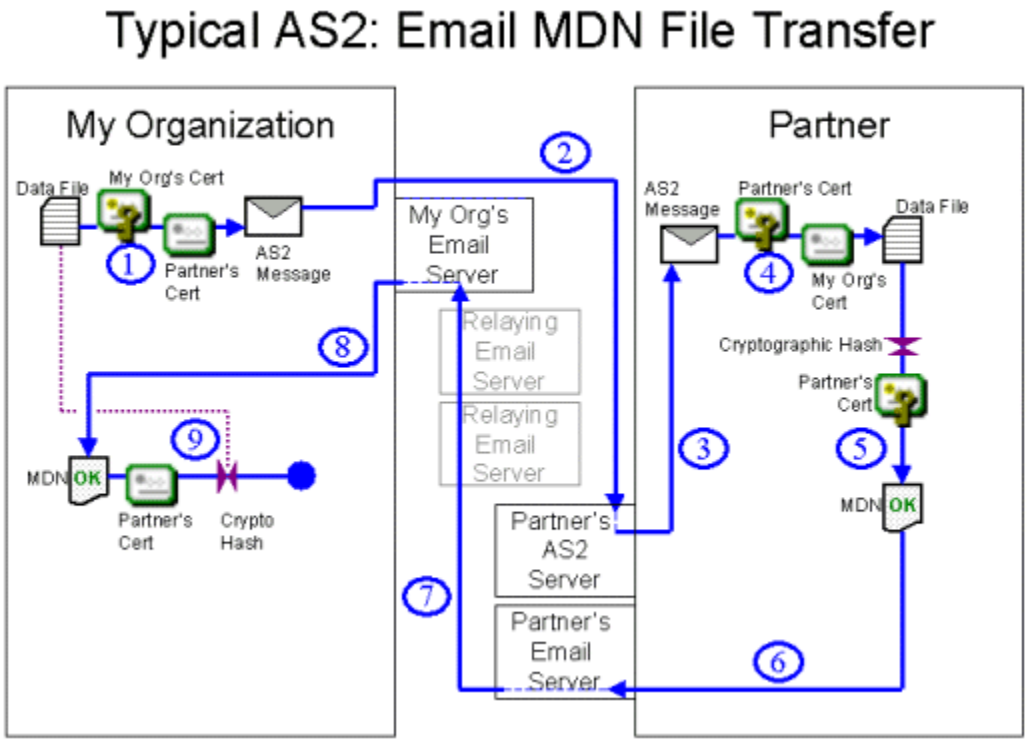
5 If you requested an MDN delivery receipt for your data file, your partner will calculate a cryptographic hash from the data file they received, sign the hash (and some other information) with the private key on their SSL certificate and create an MDN delivery receipt message. (The signing step is optional and controlled by the original message sender.)

6 Your partner will send his/her MDN delivery receipt message back by posting it to your AS2 server. (Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)

7 You will retrieve your partner's MDN delivery receipt message off your AS2 server. (Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)

8 You will inspect your partner's MDN delivery receipt message, making sure that you can verify his/her signature using the public key on your partner's SSL certificate and that the cryptographic hash calculated from your partner's copy of your data file matches the same hash calculation from your original data file.

AS2 with (Asynchronous) MDN via Email



1 You encrypt a data file with the public key on your partner's SSL certificate and sign it with the private key of your organization's SSL certificate as you bundle everything into an AS2 message. (Both the encryption and signing steps are optional, but should be used when possible.)

2 You send the AS2 message to your partner's AS2 server and close the connection. (Credentials and cleartext message headers may be protected with SSL transport in this step.)

3 Your partner will retrieve your AS2 message off his/her AS2 server. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*

4 If the message is encrypted, your partner will decrypt it using the private key on his/her SSL certificate. If the message is signed, your partner will validate your signature using the public key on your SSL certificate. Your partner will also use the contents of the AS2 message to verify that the data file they now have is identical to the data file you sent them.

5 If you requested an MDN delivery receipt for your data file, your partner will calculate a cryptographic hash from the data file they received, sign the hash (and some other information) with the private key on their SSL certificate and create an MDN delivery receipt message. *(The signing step is optional and controlled by the original message sender.)*

6 Your partner will send his/her MDN delivery receipt message to an email server via SMTP. Often, this will be your partner's local email server. (Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)

7 If the MDN delivery receipt message were sent to your partner's local email server, it will now deliver it to your email server using the SMTP protocol. Along the way your partner's MDN delivery receipt message may traverse several intermediate email servers as it is relayed across the Internet or corporate email infrastructure. (The cleartext MDN delivery receipt message is rarely protected with SSL transport if relay servers are involved.) This step will be skipped if the MDN delivery receipt message was delivered directly to your email server in step #6.

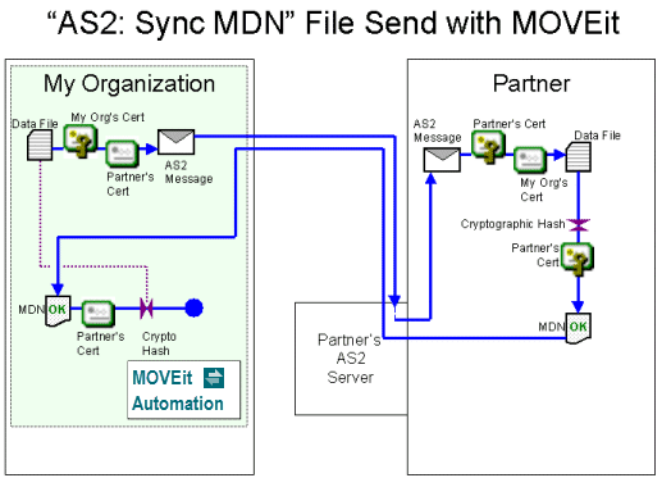
8 You will retrieve your partner's MDN delivery receipt message off your local email server using the POP3 protocol. (Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)

9 You will inspect your partner's MDN delivery receipt message, making sure that you can verify his/her signature using the public key on your partner's SSL certificate and that the cryptographic hash calculated from your partner's copy of your data file matches the same hash calculation from your original data file.

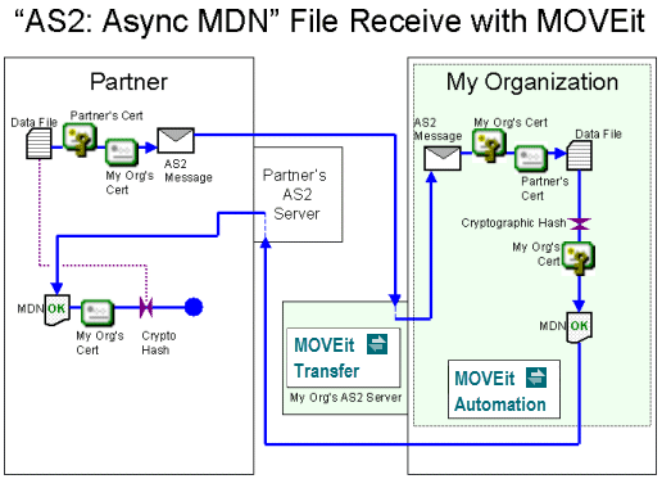
MOVEit Implementation of AS2

To support AS2 most sites will typically need to deploy both MOVEit Automation and MOVEit Transfer. MOVEit Automation performs all AS2 encryption, decryption, signing, verification and sending steps, but MOVEit Transfer is required to receive AS2 files or AS2 HTTP-based asynchronous MDNs.

AS2 with Synchronous MDN via HTTP(S)

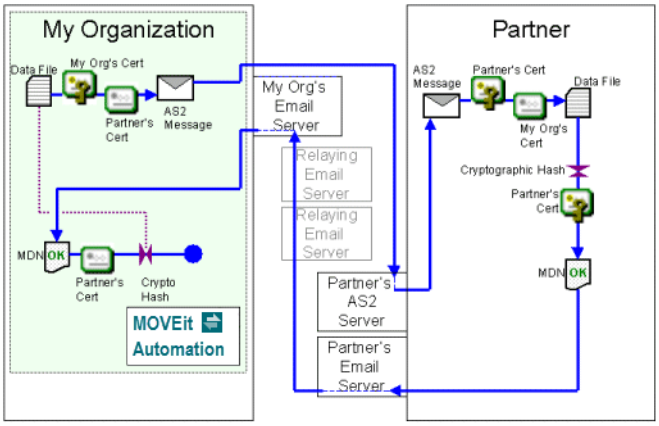


AS2 with Asynchronous MDN via HTTP(S)

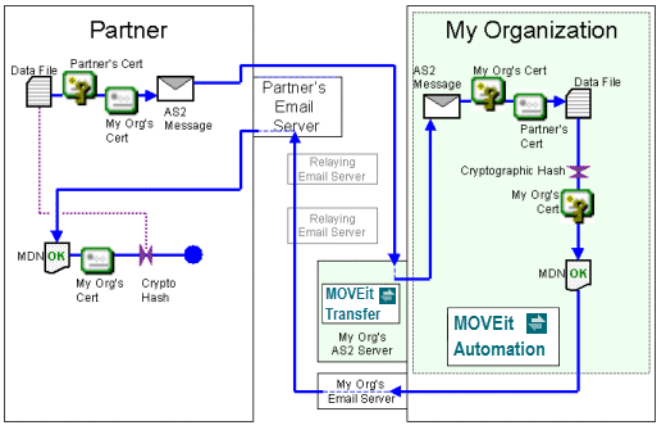


AS2 with (Asynchronous) MDN via Email

"AS2: Email MDN" File Send with MOVEit



"AS2: Email MDN" File Receive with MOVEit



See also:

§ AS1, AS2 and AS3 - Hosts - AS2
§ AS1, AS2 and AS3 - Tasks - AS2 - Source
§ AS1, AS2 and AS3 - Tasks - AS2 - Destination

Advantages/Disadvantages of AS2 (Compared to AS1 and AS3)

The "sync MDNs" make the AS2 protocol the fastest and most discrete of any of the ASx protocols, because only one "external" connection between your organization and your partner is required to complete these transactions. (Any other ASx "transfer file and return MDN" operation requires at least two external connections.)

- Advantage: AS2 is the most popular of the ASx protocols. Most people who support any one or two of the ASx protocols support AS2. It is a de facto standard in many industries, and an explicit standard in others. For example, some pharmaceutical file transfer "pedigrees".
- Advantage: AS2 is firewall-friendly when sending files. If you can connect to Internet-based web sites from your desktop, you can probably send AS2 files of any size and receive (synchronous-mode) AS2 MDNs for small to medium sized files.
- Advantage: AS2 is the only ASx protocol that allows the sender to ask for an "immediate" (synchronous) MDN response. AS2 allows senders to request immediate, "synchronous" MDN's as part of their HTTP file transmission. Synchronous MDN responses are calculated on the fly as soon as the entire file is received and returned as the response to the file transmission over the same HTTP connection. All AS1, AS3 and "asynchronous" AS2 MDNs are expressed as files to be picked up after the original transmission is complete and closed rather than as an immediate response to the current transmission.
- Disadvantage: Synchronous ("immediate") MDN responses are only appropriate for small files. Both sides must usually set up an AS2 server if large files are to transferred. AS2 transmissions involving large files can "time out" (with no "I'll get back to you" recourse) if the files sent are large. (The actual value of "large" depends on available bandwidth and timers in AS2 client, AS2 server and any intervening HTTP proxy server.) To handle large files, AS2 asynchronous MDNs may be requested instead, but these may only be requested if the file sender also owns an AS2 server on which the file receiver can post asynchronous MDNs. (In other words, "desktop AS2 clients" can 1) either send and verify small files with a synchronous MDN or 2) send large files without any MDN or verification.)
- Disadvantage: AS2 requires firewall configuration and deployment of a designated AS2 server when receiving files. To receive AS2 files, you must set up your own AS2 server (usually in a DMZ network segment) and open up firewall rules that allow remote AS2 clients to connect to your AS2 server. (MOVEit Transfer fills the role of an AS2 server in MOVEit.) Neither AS1 nor AS3 require you to host your own server to receive ASx files.
- Disadvantage: AS2 messages may be subject to HTTP proxy rules. In most situations AS2 messages are passed through traditional HTTP proxies, which means they are subject to content filters, size limits, server downtime, banned sites, "header" restrictions and other HTTP proxy issues that people may do not want to involve in file transfers with their partners. (In practice, AS2 HTTP proxy issues tend to be less of a hassle than AS1 email server issues, but "header" restrictions are probably an area to keep an eye on because AS2 depends heavily on special headers.)
- Disadvantage: AS2 has no standard concept of "username" or "password" when posting files. ("Two-factor authentication" is not standard.) Both AS1 and AS3 support the traditional file transfer concept of providing a username and password before you are allowed to upload files to the server. AS2 does not. (Some AS2 servers may have implemented "basic authentication", but it is not supported in most AS2 clients.)

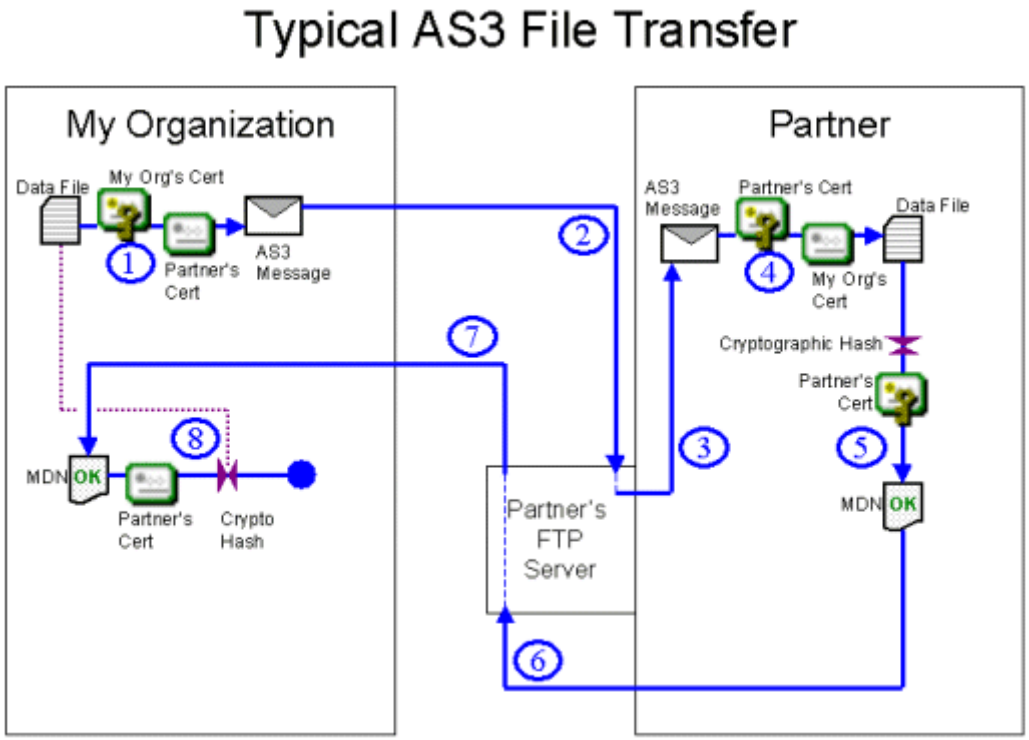
See also: ***Comparison of AS1, AS2 and AS3*** (on page 204) on the "AS1, AS2 and AS3 - Overview" page.

The AS3 Protocol

The AS3 protocol is based on FTP. It is the latest ASx protocol developed and uses the same signing, encryption and MDN conventions used in the original AS1 protocol. After AS1, AS3 is probably the easiest ASx protocol to set up and work with (if firewall issues do not crop up), but AS3 still trails AS2 in terms of general acceptance.

How an AS3 File Transfer Works

Like any other ASx file transfer, AS3 file transfers typically require both sides of the exchange to trade SSL certificates and specific "trading partner" names before any transfers can take place. AS3 trading partner names can be any valid phrase.



1 You encrypt a data file with the public key on your partner's SSL certificate and sign it with the private key of your organization's SSL certificate as you bundle everything into an AS3 message. *(Both the encryption and signing steps are optional, but should be used when possible.)*

2 You send the AS3 message to an FTP server. This could be your FTP server, your partner's FTP server or a hosted FTP server somewhere else. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*

3. Your partner will retrieve your AS3 message off the same FTP server. *(Credentials and cleartext message headers may be protected with SSL transport in this step.)*
4. If the message is encrypted, your partner will decrypt it using the private key on his/her SSL certificate. If the message is signed, your partner will validate your signature using the public key on your SSL certificate. Your partner will also use the contents of the AS3 message to verify that the data file they now have is identical to the data file you sent them.
5. If you requested an MDN delivery receipt for your data file, your partner will calculate a cryptographic hash from the data file they received, sign the hash (and some other information) with the private key on their SSL certificate and create an MDN delivery receipt message. *(The signing step is optional and controlled by the original message sender.)*
6. Your partner will send his/her MDN delivery receipt message back to the same FTP server the original AS3 message traveled through, though perhaps in a different folder or bearing a different file name. *(Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)*
7. You will retrieve your partner's MDN delivery receipt message off the same FTP server. *(Credentials and the cleartext MDN delivery receipt message may be protected with SSL transport in this step.)*
8. You will inspect your partner's MDN delivery receipt message, making sure that you can verify his/her signature using the public key on your partner's SSL certificate and that the cryptographic hash calculated from your partner's copy of your data file matches the same hash calculation from your original data file.

Variations

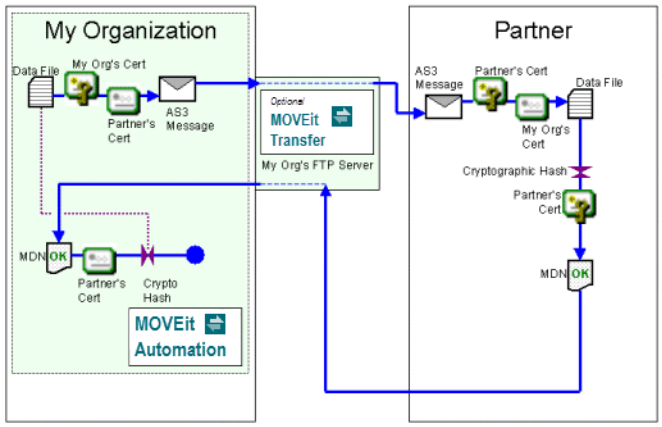
FTP Server Location - The FTP server used in an AS3 transfer could be your FTP server, your partner's FTP server or a hosted FTP server somewhere else. If you have control over the FTP server, we recommend deploying/using a ***MOVEit Transfer FTP server*** (on page 228).

MOVEit Implementation of AS3

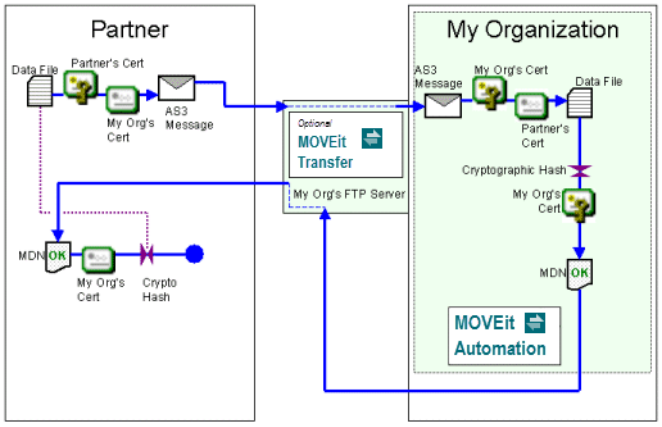
MOVEit Automation is the only MOVEit product required to send or receive files using AS1. In either case files and MDNs are sent through FTP servers, and we recommend deploying/using a ***MOVEit Transfer FTP server*** (on page 228) when possible.

...Using Your FTP Server

AS3 File Send with MOVEit

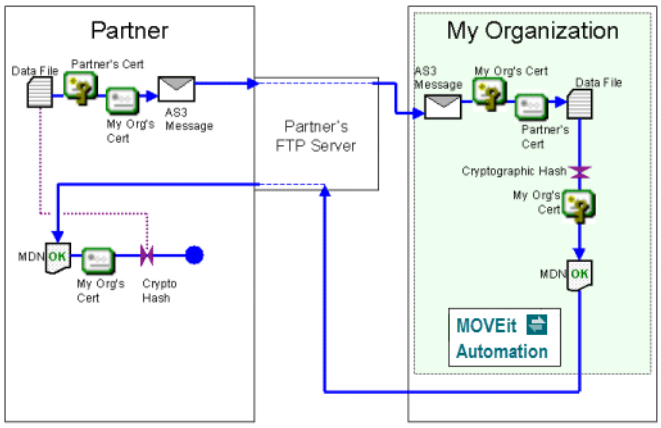


AS3 File Receive with MOVEit

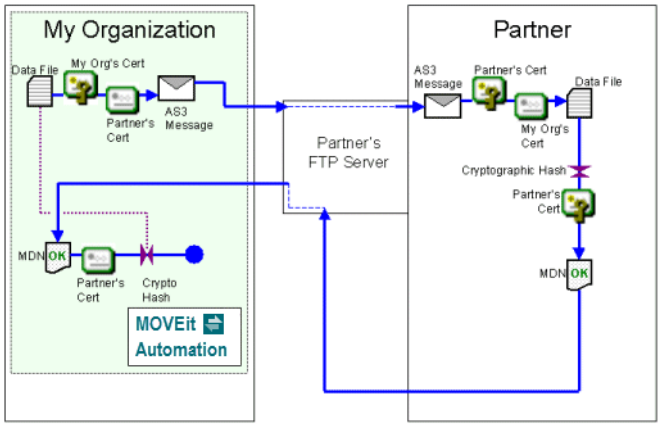


...Using Your Partner's FTP Server (Or a Hosted FTP Server)

AS3 File Receive with MOVEit



AS3 File Send with MOVEit



See also:

§ AS1, AS2 and AS3 - Hosts - AS3
§ AS1, AS2 and AS3 - Tasks - AS3 - Source
§ AS1, AS2 and AS3 - Tasks - AS3 - Destination

Advantages/Disadvantages of AS3 (Compared to AS1 and AS2)

AS3 was developed to add ASx file transfer capabilities to the well-established FTP/SSL ("FTPS") protocol. Also, using FTP as a transport rather than HTTP seemed to address the "no standard regarding username/password" limitation of AS2: most FTP servers already require username/password. The AS3 protocol is generally recognized by various industries as the "next" ASx protocol, but movement toward AS3 from established AS2 users has not been rapid.

- Advantage: If you have an AS3 client and access to an FTP server, you can send and receive AS3 transmissions. You do not need to control or host the FTP server participating in an AS3 transmission, so AS3 ranks just behind AS1 in terms of easiest ASx protocols to install and configure as long as firewall issues are not much of a concern.
- Disadvantage: AS3 has frequent firewall issues. AS3 is built on the FTP/SSL protocol, one of the most firewall-unfriendly protocols in use today. Some of the common issues involving FTP/SSL involve NAT translation, multiple data ports and improper translation of FTP commands by intervening firewalls. (Some people look for SSH and/or HTTP file transfer solutions specifically to avoid reoccurring FTP/SSL firewall issues; MOVEit products offer several tactical solutions for various FTP/SSL issues in terms of features and support.)
- Disadvantage: No AS3 transmission mode is as fast as AS2 "synchronous MDN" transfers. This is likely the issue that keeps established many ASx players from moving from AS2 to AS3. When AS2 senders request on-the-fly "synchronous MDNs" for their small file transfers (such as part orders), AS2 is by far the fastest ASx protocol. The AS3 protocol does not support similar MDN-on-the-fly capabilities.

See also: ***Comparison of AS1, AS2 and AS3*** (on page 204) on the "AS1, AS2 and AS3 - Overview" page.

MOVEit Implementation

The MOVEit product family implements a complete AS1, AS2 and AS3 file transfer solution. MOVEit products can be used to send and receive files using any of these three protocols.

Different combinations of MOVEit products are required to implement the different protocols:

- AS1: MOVEit Automation is the only MOVEit product required to implement AS1. (However, access to an email server must also be available.)
- AS2: MOVEit Automation and MOVEit Transfer are both required to implement AS2. (MOVEit Transfer is the MOVEit Automation AS2 server for purposes of receiving AS2 messages and MDNs.)
- AS3: MOVEit Automation is the only MOVEit product required to implement AS3. (Access to an FTP(S) server must also be available; for security or operational reasons you may want to make this server a MOVEit Transfer FTPS server.)

There are a few unusual exceptions to these rules:

- You can use MOVEit Transfer without MOVEit Automation to house AS3 messages and AS3 MDNs. However, MOVEit Transfer cannot encrypt, sign or verify MDN files; it simply stores, protects access to and logs access to them securely.
- You can use MOVEit Automation without MOVEit Transfer to send files as AS2 messages if synchronous MDNs and email MDNs are the only types of MDNs ever requested. (MOVEit Automation cannot receive AS2 files or HTTP-based asynchronous MDNs without MOVEit Transfer.)

The following table summarizes the roles MOVEit products play in providing AS1, AS2 and AS3 services.

MOVEit Implementation of AS1, AS2, and AS3

	AS1	AS2			AS3
File Transport	email	HTTP/HTTPS			FTP/FTPS
MDN Transport	email	HTTP/HTTPS synchronous	HTTP/HTTPS asynchronous	email asynchronous	FTP/FTPS
Server required to receive files	Any email server	MOVEit Transfer	MOVEit Transfer	MOVEit Transfer	Any FTP(S) server (such as MOVEit Transfer
Server required to send files	Any email server	n/a	MOVEit Transfer	Any email server	Any FTP(S) server (such as MOVEit Transfer
Client required to receive files	MOVEit Automation	MOVEit Automation	MOVEit Automation	MOVEit Automation	MOVEit Automation
Client required to send files	MOVEit Automation	MOVEit Automation	MOVEit Automation	MOVEit Automation	MOVEit Automation

See also (on their respective "AS1, AS2 and AS3 - The ASx Protocol" pages):

§ ***MOVEit Implementation of AS1*** (on page 212)
§ ***MOVEit Implementation of AS2*** (on page 220)
§ ***MOVEit Implementation of AS3*** (on page 225)

Drummond "eBusinessReady" Certification

MOVEit Transfer supports any AS2 client that has been Drummond or eBusinessReady certified. The software that MOVEit Transfer uses to handle incoming AS2 files and MDNs has been certified eBusinessReady under a program now managed by Drummond.

Why MOVEit Transfer is best choice for AS3

MOVEit Transfer has been able to participate in AS3 transmissions as a secure FTP server for years. Traditionally, people have thought that any FTP server with basic security features such as SSL with client certificate authentication could be used in AS3 transmissions. However, operational experience and security best practices have led many to higher expectations of their AS3 FTP server.

The MDN response files returned to AS3 file senders and used for non-repudiation can be signed, but are never encrypted. To protect these important files from tampering or unauthorized view, MOVEit Transfer offers its own built-in FIPS-validated encryption and cryptographic file integrity checks while at rest and in transit.

The FTP protocol can be tricky to implement across firewalls and NAT when SSL is introduced. To deal with these challenges, MOVEit Transfer offers comprehensive, remote-readable protocol logs and features that handle almost every possible FTP over SSL or NAT configuration. Three of the technologies MOVEit Transfer uses to avoid FTP firewall problems include a configuration of limited passive server port ranges (that has been widely copied in the industry since it was introduced in MOVEit Transfer), explicit configuration of NAT and a recent technology called "Clear Command Channel" (CCC).

Finally, the auditing facility in MOVEit Transfer can be used to help complete AS3 non-repudiation chains. In order for both sides in an AS3 exchange to agree that both parties have the same file, both sides must possess the same MDN. However, if the MDN is downloaded by the original file sender but there is a later dispute about whether or not this action actually took place, MOVEit Transfer tamper-evident audit logs can be used to show that the original file sender's MDN was made available and downloaded at a specific time by a specific user connected from a specific IP address.

Advantages of a MOVEit Implementation

MOVEit Automation calculates and stores an MDN for every ASx message it processes whether or not an MDN was requested. This feature allow operations to temporarily disable automatic MDNs and send them later using another channel if the MDN delivery channel has been temporarily disrupted. This will not work for synchronous MDNs, for obvious reasons.

All file transfers, including ASx message operations (and their MDNs), are logged in a tamper-evident audit log.

There are configurable, automatic retries on file transfers and MDN transfers.

Time-saving configuration prevents you from having to completely redefine each new file transfer with the same partner and to save steps with new partners.

Limitations of the MOVEit Implementation

MOVEit does not support the RC4 encryption algorithm, although this algorithm could be supported in a future version if necessary. (Contact Ipswitch if you need to support this non-FIPS algorithm.)

MOVEit does not support GET-method AS2 messages. Some AS2 clients support these type of messages, but POST-method submissions are the industry standard (and generally regarded as more secure and less work for operations).

About Certificates

ASx protocols use certificates to sign and encrypt files. Digitl certificates are also known as X.509 certificates, web certifictes, and client certificates.

What are certificates?

All digital certificates are made up a public key, a private key and additional information such as *common name* (CN). Certificates can be distributed with or without their private key, but in most situations you should NOT distribute certs containing your private key.

Many digital certificates are signed by other certificate authority (CA) certificates. This allows people and computers that trust the certificate authorities to trust, use and allow certificates signed by the certificate authorities.

For more information, see ***SSL Client Certificates*** (on page 158).

Where do you get a certificate?

Certificates without private keys are usually delivered to you by your trading partners. You can ***import these certificates into MOVEit Automation*** (on page 159).

A certificate with a private key can be obtained by any of the following methods:

- Purchase a commercial client certificate from Thawte, Verisign or one of the many other commercial CA vendors. This option is useful if your AS partners requiretrusted CAs as well as specific certificates in AS transactions. These certificates are also known as *email certificates* because they can also be used with SMIME-encrypted email.)
- Get a new certificate from your corporate CA. If your company is already issuing client certificates and acting as its own CA, your certificate group can provide certificate and instructions for using it.
- Obtain a certificate (with private key) from your partner. Some partners will deliver a `*.pfx` or other format certificate-with-private-key file before you start trading. You ***import the certificate into MOVEit Automation*** (on page 159). To do so, you need the password.
- Create your own certificate (on page 160).

Where to Configure Certificates

The two most common uses of certificates in AS transfers are to sign/verify messages and encrypt/decrypt messages. MOVEit Automation requires these two different certificates for any AS transport: one is defined in the Partner Organization section of each AS host and the other is defined in the My Organization section of each AS host. For more information, see General properties for ***AS1*** (on page 335), ***AS2*** (on page 338), and ***AS3*** (on page 341) hosts

However, there may be as many as 8 different certificates (plus any number of CA certificates) involved in any AS2 transfer. The following list describes possible certificate uses and where they are configured in MOVEit Automation. Certificate uses #1-5 use the ***Create SSL Certificate*** (on page 160) feature. Items #6, 7, and 8 require importing and configuring certificates through other means.

1. Cert (w/ private key) you use to sign messages for your partner and your partner normally uses to encrypt files for you. (REQUIRED) - This is configured on the main page of your AS Host definition in the "My Organization" pane.
2. Cert (w/ private key) you use to decrypt messages and MDNs from your partner - Normally this is the same certificate used to sign messages for your partner (see item #1), but an alternate certificate can be used for this purpose. To define an alternate decryption certificate, see the ASx Host's Additional Properties. (***AS1*** (on page 336), ***AS2*** (on page 340), ***AS3*** (on page 343))
3. Partner's cert (no private key) you use to encrypt messages for your partner and your partner will use to sign his/her messages (REQUIRED) - This is configured on the main page of your AS Host definition in the "Partner" pane.
4. Partner's cert (no private key) your partner will use to sign MDNs - Normally this is the same cert you use to encrypt messages for your partner, but an alternate certificate can be used for this purpose. To define an alternate signature verification certificate, use the related option on AS destinations. (This is a task-level, not a host-level option.)
5. Optional SSL client cert (w/ private key) you use to authenticate to your partners AS server - This is an optional authentication credential you may need to provide before your partner's AS server will permit you to post a message or MDN. To designate this kind of certificate, use the SSL Client Cert option in the Partner pane on the main page of your AS Host definition.
6. Optional SSL client cert (no private key) you require your partners to provide to your AS server - This is an optional authentication credential your partner may need to provide before your partner will be permitted to post a message or MDN to your AS server. To set this up on your MOVEit Transfer server when acting as an AS2 server, you may need to set up an additional IIS site and enable IIS certificate mapping after requiring certificates through IIS. (The MOVEit Transfer AS2 facility does not currently allow you to require client certificates through the software.) To set this up on your MOVEit Transfer server when acting as an AS3 server, perform the same actions as you would to require client certificates on the MOVEit Transfer FTP interface.
7. Optional SSL server cert (w/ private key) you use to provide SSL transport security on your AS server - All SSL-protected servers require a digital certificate, and SSL-protected AS servers protected by SSL are no exception. Although this use of digital certificates is technically optional under the AS protocol standards, SSL server certificates are commonly found protecting AS2 and AS3 servers, including MOVEit Transfer servers (which act as your AS2 servers and can also be AS3 servers). If you are running a MOVEit Transfer server, an SSL server certificate will automatically have been set up for you during installation, and existing procedures to renew/replace your SSL server certificates through IIS and/or MOVEit Transfer FTP are all that are required.

8 Optional SSL server cert (w/ private key) your partner uses to provide SSL transport security on their AS server - All SSL-protected servers require a digital certificate, and AS servers protected by SSL are no exception. Although this certificate is optional, it is commonly found protecting AS2 and AS3 servers. If your partner's server SSL cert is not signed by a trusted CA, you may use the Ignore Cert Errors option to avoid the need to import your partner's SSL server certificate.

Configuring AS1, AS2, and AS3 Hosts

To configure ASx hosts, click HOSTS > Add Host and select the host type.

§ ***AS1 Host Field Descriptions*** (on page 335)
§ ***AS2 Host Field Descriptions*** (on page 338)
§ ***AS3 Host Field Descriptions*** (on page 341)

The Role of MOVEit Transfer in in AS2 File Transfers

MOVEit Transfer can accept and store AS2 messages and asynchronous AS2 MDNs that will be processed later (and often immediately) by MOVEit Automation. MOVEit Transfer, rather than MOVEit Automation, is used in the role of an AS2 server because MOVEit Transfer already serves the function of a secure, Internet-exposed HTTP(S) server and MOVEit Automation already has an interface to MOVEit Transfer.

No additional license is required to accept and store AS2 messages and asynchronous AS2 MDNs on MOVEit Transfer because this feature is only useful when a separate AS1, AS2 and AS3 license has been purchased for MOVEit Automation.

AS2 messages and asynchronous AS2 MDNs are uploaded and downloaded through HTTP(S) but are not part of the normal MOVEit Transfer file system. All AS2 messages and AS2 MDNs are located in special `/AS/[partner-name]` folders, created as needed (where `[partner-name]` is your partner's official trading name.) For example, if your partner `John Smith` sends you an AS2 message, it will be found in the `/AS2/John Smith` folder.

MOVEit Transfer administrators can view and delete AS2 message files through their usual web interface.

AS2 Server URL and MOVEit Transfer File Specifics

MOVEit Transfer receives AS2 messages and asynchronous AS2 MDNs though its built-in `as2receiver.aspx` component. When your AS2 trading partners ask for the URL they should use to post AS2 messages for you, give them a URL containing `as2receiver.aspx` and the name of your host. For example, `https://as2.moveittransfer.com/as2receiver.aspx`.

The same URL value is also used when requesting AS2 asynchronous MDNs as an AS2 destination step in MOVEit Automation, but MOVEit Automation lets you specify a macro of `[AS2ReceiverURL]` (in the `MDN URL` field) and determines the exact URL at run time, because each AS2 Host can be linked to a specific MOVEit Transfer Host.

Storage and tagging of AS2 messages

AS2 messages are typically stored as files named `AS2Data`. If you want different MOVEit Automation tasks to process different AS2 messages from the same partner, you can tag each type of AS2 message transmission separately so that MOVEit Automation tasks can distinguish between them.

To tag different types of AS2 transmissions, include a `?Tag=[some-as2-filename]` argument on the URLs you provide to your partners. For example, the URL `https://as2.moveittransfer.com/as2receiver.aspx?Tag=Blue` would force MOVEit Transfer to save AS2 messages from partners using that URL as files named `Blue` rather than `AS2Data`.

Storage and tagging of AS2 MDNs

Asynchronous AS2 MDNs are stored as files named `MDN`=[AS2-ID] where `[AS2-ID]` is the ID of the original AS2 message.

For example: an AS2MDN filename is `MDN=373c55dc-f4b6-4c1b-81a1-e39f3a1c22d7@9b751ee7-d32e-4138-8124-1c107f2cd5d2`.

AS2 MDNs are stored in folders named after the partners who sent them. MOVEit Automation looks for the MDNs by using the values configured for partner name in the AS2 Host definition.

If your MOVEit Transfer hosts multiple Organizations, and you want each organization to use its own store of AS2 messages and MDNs, include an `OrgID=[OrgID]` tag (for example, `OrgID=8011`)in the URLs you give to your partners and configure in your requests for asynchronous HTTP MDNs.

For example, to have related AS2 messages and MDNs to go to a particular organization in a multiorganization configuration, do the following:

- Give partners URLs such as `https://as2.moveittransfer.com/as2receiver.aspx?OrgID=8011` or `https://as2.moveittransfer.com/as2receiver.aspx?Tag=Blue&OrgID=8011` and
- Configure a URL of `[AS2ReceiverURL]?OrgID=8011` in your asynchronous HTTP MDN field.

What happens to AS2 messages and asynchronoous AS2 MDNs?

AS2 messages and asynchronous AS2 MDNs are deleted from MOVEit Transfer as soon as any of the following occur:

- MOVEit Automation successfully decrypts and/or validates them
- MOVEit Automationdetermines that they are unfit
- MOVEit Automation gives up after (re)trying to deliver any requested MDNs.

AS2 messages that have requested synchronous MDNs are automatically deleted from MOVEit Transfer folders if MOVEit Transfer cannot deliver their respective MDNs.

To apply additional automated clean up rules to AS2 folders and files, use the Folder Settings web interface in MOVEit Transfer.

Troubleshooting

Troubleshooting AS2 transmissions can be challenging because of all the different elements involved in a single AS2 transfer. However, the following methodologies should help you tackle transfer issues.

Troubleshooting Tasks with AS2 Destinations

Tasks with AS2 destinations are used to ***send files*** (on page 220) to your partners.

1. Double-check that you and your partner agree on the following items and that they are configured identically on both sides of the transmission.
   - The URL of your partner's remote AS2 server
   - Your organization's name and your partner's name
   - Your organization's client certificate and your partner's client certificate
   - The type of encryption to be used
   - What sort of MDN you should receive from your partner (usually "none", "synchronous" or "asynchronous"; your partner doesn't need to configure this but should probably know about your choice or will have an opinion of their own)
2. Make sure MOVEit Automation can connect to your partner's AS2 server. You test this when you run your transfer task - pay attention to "host not defined", "cannot connect", "404" errors and the like. If you are having problems here, your partner's URL is likely incorrect or inaccessible. (It's generally worth asking if you are the first one to try this particular connection.)
3. Make sure MOVEit Automation thinks it has sent the file successfully. You will know this is the case if MOVEit Automation shows a working status of "X bytes sent" for your AS2 task and X is both "large" (sometimes larger than the original file size) and constant. If this is as far as MOVEit Automation gets (because it it waiting for an MDN), the task will usually fail with an "AS2 Post Error: Timeout" error after one minute.
4. Make sure the remote AS2 server thinks it has sent the MDN successfully. If MOVEit Automation is getting past this step successfully, the task will simply complete successfully. If the task does not complete successfully, failure could be due to a number of things:
   - Remote AS2 server told MOVEit Automation it received the file but then never processed it or failed to process it and silently through it away. You will need an administrator on the remote AS2 server to help you if this is the case.
   - Remote AS2 server does not support the requested MDN and takes the file anyway - another type of "silent" failure. You may want to switch your MDN type between sync/async, but you may need to get the remote AS2 server administrator involved in here too.

- § Remote AS2 server processes your file but fails to get you a synchronous MDN back in time. If this is the case, the remote AS2 server may log that it created an MDN for your file, but it should also log the fact that you never got it.
- § Remote AS2 server processes your file but cannot send you an asynchronous MDN. As long as you have taken care to leave a value of "[AS2ReceiverURL]" in your Destination's "MDN URL" this error is likely due to an unresolvable DNS, proxy server or other connection problem on the remote AS2 server's side.

Troubleshooting Tasks with AS2 Sources

Tasks with AS2 sources are used to ***receive files*** (on page 220) from your partners.

1. **Double-check that you and your partner agree on the following items** and that they are configured identically on both sides of the transmission.
   - § The URL of your (MOVEit Transfer) AS2 server. This will be something like "https://myserver.moveittransfer.com/as2receiver.aspx"
   - § Your organization's name and your partner's name
   - § Your organization's client certificate and your partner's client certificate
   - § The type of encryption to be used
   - § (You can ask about what sort of MDN your partner expects, but there is nothing to configure in MOVEit Automation regarding this information because AS2 file senders configure this value and AS2 file receivers - MOVEit Automation in this case - are expected to pull it off incoming AS2 messages.)
2. **Make sure your partner's AS2 client can connect to your MOVEit Transfer server.** You can start with basic connectivity and DNS tests by simply asking your partner to connect to your MOVEit Transfer using the URL you use for normal, interactive web access. Then have your partner try to send an AS2 file with the client and look/listen for "cannot connect", "404" and other errors that suggest that the remote AS2 client cannot connect to the AS2 interface of your MOVEit Transfer server.
3. **Make sure your partner is successfully posting files to MOVEit Transfer.** sign on to your MOVEit Transfer server as an Admin or FileAdmin to see if you suspect your partner is not posting AS2 files successfully. If your partner is posting files successfully, you will see a folder named "/AS2/[PartnerName]" where "[PartnerName]" is the exact name of your partner (as configured in your AS2 host configuration). As your partner posts AS2 files, you will also see files named "AS2Data" (or something else if URLs with the "Tag=" attribute are used) show up in this folder and in the audit log.
4. **Make sure MOVEit Automation is automatically kicking off the task associated with this transfer correctly.** There are several reasons why this could not be happening - see the "Tasks Configured to Receive AS2 Files Do Not Run Automatically" section below for details.
5. **Make sure your MOVEit Automation task is correctly processing your partner's AS2 file and returning a valid MDN.** Fortunately, this is mostly internal processing at this point: MOVEit Automation will provide you information about any problems occurring here. If your partner has requested an asynchronous MDN for its AS2 file, it is possible that the URL he/she provided in the AS2 message is invalid or unreachable, but this is almost the only error caused by external conditions that could be encountered at this stage.

Error Messages Encountered During AS2 File Transfer

cannot connect to MIAS2: Access is denied

This message usually indicates that the MOVEit Automation "MIAS2.exe" AS2 helper application has not been started. This application should be started and have its own "Task Manager - Processes" entry when the MOVEit Automation service starts. First try restarting the MOVEit Automation service. If this does not fix the problem, use the "Run MOVEit Automation manually" option from the "Start | Programs | MOVEit Automation" program group to run MOVEit Automation in the foreground and watch for other clues from the MOVEit Automation or MIAS2 windows in the foreground.

Host default partner cert not found

This message often means that a partner's client certificate was imported and selected in an AS2 Host configuration, but that the underlying certificate has since been deleted. The best way to correct this situation is to reimport your partner's client certificate and reselect it in the AS2 Host configuration.

405 Method Not Supported

This message means you got to a web server (all AS2 servers are web servers) but that the web server doesn't understand or allow your request. If you copied an "Outgoing HTTP URL" from an AS2 Host configuration into a web browser, this message is perfectly normal (especially if your partner's server is an MOVEit Transfer AS2 server). However, if you see this message during an AS2 file transfer it more likely indicates one or more of the following problems:

- The "Outgoing HTTP URL" you typed in is incorrect.
- A proxy server between your MOVEit Automation and your partner's AS2 server does not allow AS2 traffic.
- URLScan or some other host-based intrusion engine does not allow AS2 transactions.

The requested name is valid, but no data of the requested type was found

This error typically indicates that a DNS entry for a configured hostname could not found. If you see this error you should recheck any hostnames configured as part of this transfer. In a specific case, if this error starts with a "AS2SendMDN error: " prefix then the value of the "SMTP Server to be used for sending email MDNs" field in your AS2 host's Advanced settings ("Email MDN" tab) is probably not correct or not reachable.

304 Could not write to file

This message may mean that the transfer has exceeded the file size limit for AS2 Receive. The limit for a single file is 1 GB. If you are attaching files to a message (sent via ASx), the limit for a single message and attached files is 200 MB.

Tasks Configured to Receive AS2 Files Do Not Run Automatically

If you are receiving AS2 files from partners, you must set up tasks with AS2 Sources for each partner that will be sending you AS2 files. Partners post AS2 files to a MOVEit Transfer server and MOVEit Automation normally learns about posted files and acts on them within seconds of their completion.

Reasons why tasks configured to receive AS2 files do not start automatically include:

- Your partner isn't really posting AS2 files successfully - Your partner will post AS2 files to your MOVEit Transfer server so you must sign on to your MOVEit Transfer server as an Admin or FileAdmin to see if you suspect your partner is not posting AS2 files successfully. If your partner is posting files successfully, you will see a folder named "/AS2/[PartnerName]" where "[PartnerName]" is the exact name of your partner (as configured in your AS2 host configuration). As your partner posts AS2 files, you will also see files named "AS2Data" (or something else if URLs with the "Tag=" attribute are used) show up in this folder and in the audit log. If AS2 file posts are not making it this far, please consult the "405 Method Not Supported" advice above.
- AS2 poller is not running - If you watch the MOVEit Automation debug log at the All Debug level with no task filter set, you should see orange messages like "AS poller found X files on..." and "AS2 poller polled X hosts, saw Y files, started Z tasks" scroll by every few seconds. If you do not see these messages, the AS2 poller (that looks for AS2 file postings on MOVEit Transfer) is probably not running. Normally, restarting the MOVEit Automation service will fix this.
- AS2 poller is finding files, but your task isn't scheduled to run at the time the files are found - At the All Debug level, orange messages like "Considering new file AS2/.../... for task X" will scroll by whenever new AS2 files are posted to your MOVEit Transfer server. If the task you would expect to act on the posted files is not one of the ones listed, it is probably because your task is missing a schedule that would allow it to run when files arrive during a particular window of time. The easiest way to correct this situation is to add a "always on" schedule to your task that runs on "All Days", "Repeated" between "00:00" and "23:59".
- AS2 poller is finding files and your task is scheduled to run when the files are found but the related "receive" task still isn't getting called. - If this is your situation, make sure your AS2 source's "File Tag(s)" match (or include) the filenames of AS2 files being posted to your MOVEit Transfer server. When in doubt, use a wildcard File Tag of "*" to download everything from that particular partner.

The Role of MOVEit Transfer in AS3 File Transfers

MOVEit Transfer can accept and store AS3 messages and AS3 MDNs that will be processed later by MOVEit Automation or any other AS3 client. MOVEit Transfer, rather than MOVEit Automation, is used in the role of an AS3 server because MOVEit Transfer already serves the function of a secure, Internet-exposed FTP(S) server.

No additional license is required to accept and store AS3 messages and AS3 MDNs on MOVEit Transfer because, according to the AS3 specification, any FTP server can function as an AS3 server. This means that if you have licensed a MOVEit Transfer server, you already have an AS3 server.

AS3 messages and AS3 MDNs are uploaded and downloaded through FTP and therefore are part of the normal MOVEit Transfer file system. All AS3 messages and AS3 MDNs are located in the `/Home/...` or `/Distribution/...` folders and are otherwise treated as normal files.

ASx Source and Desitnation Options

Source and Destination options for ASx hosts

- ***AS1 - Source and Destinatio*** (on page 239)n
- ***AS2 - Source and Destination*** (on page 240)
- ***AS3 - Source and Destination*** (on page 242)

AS1 - Source and Destination

AS1 Source

An AS1 source is a reference to an AS1 host that defines a single ruleset for a file from a partner's EDI data message for use in a task. Partner credentials, encryption method and other partnership-level details are configured at the Host level.

AS1 Source Options:

§ Subject Match - Indicates which messages to download from the AS1 POP3 server. ***Macros*** (on page 176) and the wildcard characters * and ? are allowed.

AS1 Destination

An AS1 destination is a reference to an AS1 host that defines a single ruleset for sending a file as an EDI data message to a partner and requesting an MDN to confirm receipt of the file. Partner credentials, encryption method and other partnership-level details are configured at the Host level.

AS1 Destination Options:

§ Subject - The subject of the email message that is sent to the partner. You can use ***Macros*** (on page 176) in this field.

§ Use Original File Name(s) - If checked, MOVEit Automation sends the file with the name under which it was saved on the source.

If not checked, MOVEit Automation uses the name defined in the Filename field. This name can contain macros.

§ Send all source files in a single ASx message – If checked, the destination sends a single ASx message with multiple attachments.

If not checked, the destination sends individual ASx messages for each source file.

§ Request MDN - If checked, MOVEit Automation requests an MDN from the destination partner to verify that the data arrived, and does not consider the task complete until it has received one.

§ Request Signed MDN - If checked, MOVEit Automation requests that the MDN sent by the destination partner be signed by the partner's SSL certificate, to verify the origin of the MDN message.

§ Use Partner Encryption Certificate for Signature Validation - If checked, MOVEit Automation uses the Partner Certificate configured in the referenced AS1 host to validate the signature on the MDN received from the partner. If not checked, a different SSL certificate can be configured for signature validation in the Validation Certificate field.

§ MDN Email Address - The email address to which the destination partner sends MDNs. Default is the `[HostOrgEmail]` macro, which represents by the My Organization - Email Address field value of the referenced AS1 host.

AS2 - Source and Destination

AS2 Source

An AS2 source is a reference to an AS2 host that defines a single ruleset for a file from a partner's EDI data message for use in a task. Partner credentials, encryption method and other partnership-level details are configured at the Host level.

AS2 Source Options:

- File Tag(s) - Specifies the files to download from the AS2 MOVEit Transfer server. You can use ***Macros*** (on page 176) in this field, as well as * and ? wildcard characters. Multiple masks can be entered, separated by semicolons (;).
- Ignore File(s) - Ifchecked, one or more file masks can be entered. MOVEit Automation ignores any files that match one of the entered masks. You can use Macros in this field. Multiple masks can be entered, separated by semicolons (;).

Note: For an AS2 Receive, the file size limit for a single file is 1 GB. If you are attaching files to a message (sent via ASx), the limit for a single message and attached files is 200 MB.

AS2 Destination

An AS2 destination is a reference to an AS2 host that defines a single ruleset for sending a file as an EDI data message to a partner and requesting an MDN to confirm receipt of the file.

- Partner credentials, encryption method and other partnership-level details are configured at the Host level. For more information, see ***Configuring ASx Hosts*** (on page 233).
- For asynchronous HTTP(S) MDN requests, you must have an incoming MOVEit Transfer host already defined.
- For asynchronous email MDN requests, you must have an AS1 host already defined.

AS2 Destination Options:

- Use Original File Name(s) - The file is sent with the name under which it was saved on the source. If not checked, the file is sent with name defined in the Filename field. This name can contain ***Macros*** (on page 176).
- Request MDN -MOVEit Automation requests an MDN from the destination partner to verify that the data arrived, and does not consider the task complete until it has received one.
- Send all source files in a single ASx message –Tthis destination sends a single ASx message with multiple attachments, instead of an individual ASx message for each of the source files,.
- Request Signed MDN - MOVEit Automation requests that the MDN sent by the destination partner is signed by the partner's SSL certificate, to verify the origin of the MDN message.
- Use Partner Encryption Certificate for Signature Validation - MOVEit Automation uses the Partner Certificate configured in the referenced AS1 host to validate the signature on the MDN received from the partner.

  If not checked, a different SSL certificate can be configured for signature validation in the Validation Certificate field.

- **Request Asynchronous MDN** - MOVEit Automation requests that the MDN be sent after the file transfer connection has been closed, instead of at the end of the file transfer before the connection closes. An asynchronous MDN can be sent either to an HTTP server, or by email.

  If not checked, a synchronous MDN will be requested.
- **Request Email MDN** - MOVEit Automation requests that the MDN be sent via email to the email address indicated by the **MDN Email Address** field.
- **MDN URL** - The URL to which HTTP MDNs are sent by the destination partner. This field is available only if the **Request Asynchronous MDN** and **Request Email MDN** checkboxes are *not checked*.
  - For synchronous MDN requests, this value (while required by the AS2 specification) is typically ignored because the MDN is sent in response to the file transfer via the same connection.
  - For asynchronous HTTP(S) MDN requests, this value is necessary to tell the destination partner where to send the MDN.
  - Default is the `[AS2ReceiverURL]` macro, which represents a properly formatted URL based on the linked MOVEit Transfer host of the referenced AS2 host. Note that this field will not be available if the Request Asynchronous MDN and Request Email MDN options are checked.
- **MDN Email Address** - The email address to which the destination partner should send MDNs. By default, this is set to the "[AS1OrgEmail]" macro, which represents the My Organization - Email Address field value of the AS1 host configured in the referenced AS2 host. Note that this field will only be available if the Request Asynchronous MDN and Request Email MDN options are checked.

AS3 - Source and Destination

AS3 Source

An AS3 source is a reference to an AS3 host that defines a single ruleset for obtaining a file from a partner's EDI data message for use in a task. Partner credentials, encryption method and other partnership-level details are configured at the Host level.

AS3 Source Options:

- Folder - The folder on the AS3 FTP server to search for EDI data message files. You can use ***Macros*** (on page 176) in this field.
- FTP File(s) - The files to be downloaded from the AS3 FTP server. You canuse Macros in this field, as well as * and ? wildcard characters. Multiple masks may be entered, separated by semicolons (;).
- Ignore File(s) - If checked, one or more file masks can be entered. Files whose names match one of the entered masks are ignored. You can use Macros in this field. Multiple masks can be entered, separated by semicolons (;).
- Upload MDN(s) to Same Folder as Files - If checked, MDNs are written to the same FTP folder in which the original EDI data message was located. If not checked, you can specify an MDN Folder. You can use Macros in the MDN Folder field.
- MDN Filename - The name of the file where MOVEit Automation writes the MDN to if an MDN is requested by the sending partner. You can use Macros in this field.

AS3 Destination

An AS3 destination is a reference to an AS3 host that defines a single ruleset for sending a file as an EDI data message to a partner and requesting an MDN to confirm receipt of the file. Partner credentials, encryption method and other partnership-level details are configured at the Host level.

AS3 Destination Options:

- Folder - The folder on the AS3 FTP server where EDI data message are written. You can use ***Macros*** (on page 176) in this field.
- Use Original File Name(s) - If checked. the file is sent with the name under which it was saved on the source. If not checked, the file is sent with the name defined in the Filename field. This name can contain macros.
- Send all source files in a single ASx message – If checked, this destination sends a single ASx message with multiple attachments.

  If not checked, the destination sends an individual ASx message for each source file.
- Request MDN - If checked, MOVEit Automation requests an MDN from the destination partner to verify that the data arrived, and does consider the task complete until it has received one.
- Request Signed MDN - If checked, MOVEit Automation requests that the MDN sent by the destination partner is igned by the partner's SSL certificate, to verify the origin of the MDN message.
- Use Partner Encryption Certificate for Signature Validation - If checked, MOVEit Automation uses the Partner Certificate configured in the referenced AS1 host to validate the signature on the MDN received from the partner.

  If not checked, a different SSL certificate can be configured for signature validation in the Validation Certificate field.

- **Look for MDN(s) in Same Folder as Files** - If checked, MOVEit Automation looks for MDNs from the destination partner in the same FTP folder that we uploaded the EDI data message file to.

  If not checked, a different MDN folder can be entered in the **MDN Folder** field. You can use Macros in the **MDN Folder** field.
- **MDN Filemask** - The filemask to use to search for MDNs from the destination partner. Macros and the wildcard characters * and ? are allowed.

Advanced Topics

- **MDN URL** - The FTP URL to which MDNs are sent by the destination partner.. This value, while required by the AS3 specification, is typically ignored. By default, it is set to a properly formatted URL based on the referenced AS3 hostname or IP address.

FTP Source Integrity

Verifying a downloaded file's integrity requires a means of assuring that the local copy of the file is identical to the remote copy of the file. When downloading a file from a MOVEit Transfer host, MOVEit Automation automatically sends a "hash" of the downloaded file to MOVEit Transfer, to make sure the file it received is identical to the file on MOVEit Transfer. This "hash" is essentially a fingerprint of the file, and is constructed so that the likelihood of two different files having the same hash is very remote.

FTP servers do not support a standard method for providing such an assurance for clients who download files from them. However, many FTP server operators opt to use a method which provides a partial assurance of the downloaded file's integrity. This method involves making available a file on the FTP server which contains a list of hashes of the other available files. The client can then check the hashes listed in that file against the files it downloaded.

Any hash system can be used for this method, but the most frequently used is the MD5 hash. MOVEit Automation now supports using an MD5 file on a source FTP server to check downloaded files against. The option is available on FTP and SSH hosts, and is overridable on sources using FTP or SSH hosts. When properly configured, MOVEit Automation can check for an MD5 hash file (normally these files are named MD5SUM, but MOVEit Automation does support supplying a different name), and verify its downloaded files against the hashes contained in it. If a file does not match the hash listed for it, or if the file does not have a hash listed for it (only if MD5 checking is set to Required), MOVEit Automation will generate an error and discontinue processing of that file.

Note that because this method relies on downloading a list of file hashes, it cannot provide complete integrity verification, since there is no way for MOVEit Automation to make sure that the MD5 hash file was not altered in transit. It does, however, provide more verification than a normal FTP transfer, and under normal circumstances, will provide a defense against files that are somehow corrupted during transport.

Setting Up MD5 Hash Files on an FTP Server

The majority of FTP server operators that provide MD5 files use a program called "md5sum" to generate those files. This program takes a list of files and generates a list of MD5 hashes for those files. This output is then redirected to a file, normally called MD5SUM, which resides in the directory along with the files it contains hashes for. FTP operators wishing to provide MD5 hash files for a MOVEit Automation client should use this program. It is widely available on the internet, as well as being included in most UNIX distributions today. Use your favorite search engine to find a copy for your specific system, if you don't have it installed already.

To generate an MD5 hash file with md5sum, simply execute it with the list of files you wish to create hashes for, and redirect its output to a file called MD5SUM. For example, to create a hash file of all the files in the /ftproot/products directory on a server, issue the following command:

```
cd /ftproot/products
md5sum * > MD5SUM
```

Issuing a command like this for all important FTP content directories on a frequent basis will help provide added assurance that the files downloaded by clients such as MOVEit Automation are identical to the files on your FTP server.

Custom Directory Parsing

When downloading from an FTP server, an automated FTP client must request a directory listing and parse the results to determine the names, date stamps, sizes, and other information about the files on the server. There is no official standard format for this information; each FTP server vendor uses its own format. As a result, FTP clients must be able to handle different types of directory listings. SSH clients face a similar situation.

MOVEit Automation provides the following options to handle this situation:

- **Automatic recognition of common servers**. By default, MOVEit Automation automatically recognizes and parses directory listings from the most common types of FTP and SSH servers, including Microsoft Windows®, UNIX®, Novell®, MVS®, Unisys®, and Van Dyke®.
- **Blind downloads**. If the name of the file is known in advance, select the Blind Downloads option when defining an FTP or SSH host, so that MOVEit Automation does not use any directory listing commands when downloading from the host. For more information, see ***Blind downloads***
- **Column-based custom parsing** (on page 246). If the directory listing has a simple tabular format that can be described by rows and columns, you can specify these parameters in the host definition.
- **Directory parsing script** (on page 247). You can write a script in VBScript to parse a directory listing. Any format can be handled this way.

Column-based Custom Parsing

With column-based directory parsing, directory listings are assumed to contain one file per line, with optional header and trailer information that is ignored.

To specify custom directory list parsing parameters in the MOVEit Automation Web Admin interface, see ***FTP Host - Additional Properties*** (on page 319) or ***SSH Host - Additional Properties*** (on page 326).

The following parameters are configured for the host:

- Filename start column. The column number (with 1 being the first column) in which the filename starts. The filename ends at the first space, or end-of-line.
- Date start column. If non-zero, specifies the column at which the file's date stamp starts. The date/time format must match one of the formats described in Date and Time Formats below. If zero, the `Only New Files` option is not available.
- Skip Lines Top. The number of lines of header information to be skipped at the beginning of the listing. Typically this is 0.
- Skip Lines Bottom. The number of lines of trailer information to be skipped at the end of the listing. Typically this is 0.

For the following example of a 5-line FTP directory listing:

```
XYZZY FTP Server Directory Listing, prepared at 14:10:55

         267 2006-02-15 15:27:39 trnreport.txt
      537401 2005-11-29 21:12:47 xyz2005.rpt
*** END OF FILE LIST.
```

the settings would be:

Filename column	3 4
Date column	1 4
Skip Lines Top	2
Skip Lines Bottom	1

Directory Parsing Script

With this option, you write a script (or use a vendor-supplied script) to parse a directory listing. Directory parsing scripts are written in VBScript and are configured and edited the same as other MOVEit Automation scripts. Two directory-parsing-specific functions are available:

§ sDirListing = MlDirGetListing()

Returns the entire verbatim listing from the FTP or SSH server. This typically contains lines separated by CR and LF (ASCII 13 and 10).

§ MlDirAddEntry FilenameToMatch, Date, Size, bIsDir, FilenameForGet, FilenameOriginal

Adds an entry to the FTP or SSH directory listing being parsed.

FilenameToMatch	The filename against which MOVEit Automation matches when doing filename wildcard matches.
Date	A string that uses one of the accepted date formats.
Size	File size in bytes. Set this to 0 if the size is unknown.
bIsDir	Boolean variable. True - the entry is a directory. False - the entry is a file.
FilenameForGet	Filename to send to the server when requesting a download of the file. Typically the same as FilenameToMatch. Used for FTP servers that run on operating systems with unusual filesystems.
FilenameOriginal	Filename to return as the original filename, in contexts such as the [OrigName] macro. It is usually the same as FilenameToMatch. Used for FTP servers that run on operating systems with unusual filesystems.

Note: The last two parameters, FilenameForGet and FilenameOriginal, are for use with FTP servers running on operating systems with unusual filesystems. Normally all three filenames should be set the same.

Most other MOVEit Automation MIxxx functions are not available in a directory parsing script.

To explain the differences between the three versions of the filename, consider a hypothetical FTP server that allows multiple numbered versions of a file, with the version following the filename in a directory listing. Suppose the directory listing looks like this:

```
MYFILE.DAT;22 45321 2006-05-06 08:11:56
MYFILE.DAT;21 44090 2006-05-05 17:20:40
README.TXT;3 8192 2005-12-30 21:38:27
```

In this directory listing, there are two versions of MYFILE.DAT, with version 22 being the more recent.

Ordinarily, the user does not know in advance which numeric version is desired; the user knows only that they want the most recent version, or the next-to-most-recent version, etc. Therefore, it is not recommended to configure a MOVEit Automation source with a filemask that refers to a specific version number.

For the purposes of this FTP server, you can invent a filemask syntax in which the most recent version is referred to as MYFILE.DAT(0), the next most recent version as MYFILE.DAT(-1), etc. However, when transferring the file to the destination, the version number is not relevant, because most destination servers do not recognize file versions. Name the file MYFILE.DAT.

There are three versions of the name:

- The name used in the file mask in the source to select the file. For example, MYFILE.DAT(0)
- The name that MOVEit Automation uses to retreive the file. For example, MYFILE.DAT;22
- The name used in the destination as the Original Name. For example, MYFILE.DAT

So, a script parsing this directory listing would do the equivalent of:

```
FilenameToMatch = "MYFILE.DAT(0)"
MyDate = "2006-05-06 08:11:56"
MySize = 45321
bIsDir = False
FilenameForGet = "MYFILE.DAT;22"
FilenameOriginal = "MYFILE.DAT"
MIDirAddEntry FilenameToMatch, MyDate, MySize, bIsDir, FilenameForGet,
FilenameOriginal
```

See the ***Sample Script*** (on page 250).

Date and Time Formats

Both of the MOVEit Automation custom directory parsing options recognize several datestamp formats. If the date format used by an FTP server does not match one of the following formats, you cannot use column-based parsing. Instead, use a script to massage the date before sending it to MIDirAddEntry.

A file date/time stamp is assumed to consist of a date, followed by one or more spaces, followed by a time. If the time is not recognized, it is treated as midnight (00:00:00).

Date formats:

- YYYY-MM-DD - The preferred format.
- YY-MM-DD - This uses a Y2K-like "pivot year" of 1970. Values from 0 to 70 are assumed to be 2000 through 2070. Values 71-99 are assumed to be 1971-1999.
- MM/DD/YY - Uses the same pivot year.
- MM/DD - Assumes that the date is within the last 12 months. If the date is today or prior to the current day of the year, it is assumed to be in this year, else it's assumed to be in the previous year.

Time formats:

- hh:mm - Time in 24-hour format.
- hh:mm:ss - Time in 24-hour format.
- hh:mm***AMPM*** - Time in 12-hour format. ***AMPM*** must be AM, am, PM, or pm.
- hh:mm:ss***AMPM*** - Time in 12-hour format, as above

Sample Script

The following script demonstrates custom directory parsing.

```
' This script parses a directory listing from a Windows FTP server.
' In real life, this script would not be necessary, because
' MOVEit Central is able to natively recognize and parse directory
' listings on Windows FTP servers.
'
' A Windows FTP server returns a directory list like: ' 05-02-06 04:22PM 734
ModsNotes.txt
' 10-13-05 09:13AM <DIR> Incoming
' 123456789a123456789b123456789c123456789d123456789
'
Option Explicit
Sub Main()
  Dim sDirListing, aryLines, MyLine, TestForDir, MyName, j
  Dim MyYear, MyMonthDay, MyTime, MyDate, MySize, bIsDir, NFiles, NDirs
sDirListing = MIDirGetListing()
   ' Break apart the listing into an array of lines.
   aryLines = Split(sDirListing, vbCrLf)
   NFiles = 0
   NDirs = 0
  For j = LBound(aryLines) To UBound(aryLines)
      MyLine = aryLines(j)
      ' Heuristic to ignore any unreasonably short lines.
    If Len(MyLine) > 39 Then
        ' MOVEit Central doesn't understand mm-dd-yy format,
        ' so change a date like 10-13-05 to 05-10-13.
        MyMonthDay = Mid(MyLine, 1, 5)
        MyYear = Mid(MyLine, 7, 2)
        MyTime = Mid(MyLine, 11, 7)
        MyDate = MyYear & "-" & MyMonthDay & "" & MyTime
        MyName = Mid(MyLine, 40)
        TestForDir = Mid(MyLine, 25, 5)
      If TestForDir = "<DIR>" Then
            bIsDir = True
            MySize = 0
            NDirs = NDirs + 1
      Else
            bIsDir = False
            MySize = Mid(MyLine, 19, 38-19+1)
            NFiles = NFiles + 1
      End If
        MIDirAddEntry MyName, MyDate, MySize, bIsDir, MyName, MyName
    End If
  Next
  If MIGetDebugLevel() >= 50 Then
      MILogMsg "Found" & NFiles & "files and" & NDirs & "dirs"
  End If
End Sub

Main
```

SysLog and SNMP

MOVEit Automation logs to the Windows Event Log. It does not directly log events to SysLog or SNMP management consoles,

This section briefly describes several utilities that send MOVEit Automation entries from the Windows Event Log to a Syslog or SNMP management console. As a best practice, if you want to use these utilities, log events into the Windows MOVEit Event Log instead of the Windows Application Event Log. This avoids having to screen for particular event log entry sources.

Syslog Utilities

SysLog is based on UDP (usually port 514). SysLog is an "unreliable" protocol in the sense that neither the client nor the server have any information about whether if SysLog messages are dropped by the network.

Event Reporter

A commercial client named ***Event Reporter*** (see http://www.eventreporter.com/en - ***http://www.eventreporter.com/en***) performs filtering on event logs before sending them to a SysLog.

Snare

A freeware client named ***Snare*** (see http://www.intersectalliance.com/projects/SnareWindows - ***http://www.intersectalliance.com/projects/SnareWindows***) performs filtering on event logs before sending.

WinAgents Event Log Translation Service

A commercial client named ***WinAgents Event Log Translation Service*** (see http://www.winagents.com/en/products/eventlog-syslog/ - ***http://www.winagents.com/en/products/eventlog-syslog/***) performs some filtering on event logs before sending them to a SysLog server and/or an SNMP management console.

winlogd

A freeware utility named ***winlogd*** (see http://www.edoceo.com/creo/winlogd/ - ***http://www.edoceo.com/creo/winlogd/***) can be used to scoot all events from all event logs to a designated SysLog server.

```
D:\temp>winlogd -i
Installation successful, say `net start winlogd`

D:\temp>winlogd --show
Server:   192.168.101.1
Port:     514
Facility: LOCAL3
Monitor:  6000
Flush:    6000

D:\temp>net start winlogd
The winlogd service is starting.
The winlogd service was started successfully.
```

This program does not have a lot of options (Server, Port and Facility), but it is a quick and effective way to get MOVEit Transfer events and other messages into a designated SysLog server.

SNMP

The SNMP protocol uses the concepts of "community"; typically events are fired off into a community and an SNMP management console collects, logs and perhaps acts upon them. Ipswitch makes no suggestion regarding SNMP management consoles; our customers usually either have or do not have one, and selection of this type of server goes well beyond this documentation. However, Ipswitch does suggest a couple of clients which would likely work as an SNMP "client" in most SNMP situations.

Like SysLog, SNMP is based on UDP (usually port 161). As such, SNMP is not the most reliable protocol out there.

Unlike SysLog clients, SNMP "clients" tend to be purchased in bulk. If you own an SNMP management console, you probably already also own an SNMP client you can use. (Ask the group in charge of your SNMP management console.) There are vendors who will offer you a compatible, standalone SNMP client.

WinAgents Event Log Translation Service

A commercial client named ***WinAgents Event Log Translation Service*** (see http://www.winagents.com/en/products/eventlog-syslog/ - ***http://www.winagents.com/en/products/eventlog-syslog/***) performs some filtering on event logs before sending them to a SysLog server and/or an SNMP management console.

Antivirus

Using Real-Time Scanners

MOVEit Automation can be used to scan downloaded files via its interface to third-party real-time antivirus utilities. These utilities work by immediately deleting infected files as they are written to or read from the MOVEit Automation cache directory. MOVEit Automation will notice that the file is no longer available and will obtain the infection information from the antivirus logs. It will then take the action that you have configured on the configuration program's ***Virus tab*** (on page 361).

MOVEit Automation will consider any individual file transfer that failed because a virus was detected to be a "normal" failure in the sense that it will log a specific "virus found" message in the file failure record and will initiate any configured "failure" next actions (including email alerts) configured for the task. Furthermore, MOVEit Automation will consider any task that finds a virus in one of its files to have partially failed, although it will normally continue to transfer all files that did not contain viruses in the same task run.

MOVEit Automation currently interfaces with the following antivirus programs:

- Symantec AntiVirus
- McAfee VirusScan
- Trend Micro OfficeScan

MOVEit Automation will notice and handle infections detected by other real-time antivirus programs, but it will not be able to report the name of the specific virus that was detected.

After connecting to MOVEit Automation, use the `Command > Test Antivirus` command from MOVEit Automation Admin to test whether MOVEit Automation and your local antivirus package are successfully communicating.

Notes on Trend Micro OfficeScan. If you are using Trend Micro OfficeScan, you should be aware that the default installation options enable scanning for only a few file extensions. This will cause the scanner to miss most infections, since by default, MOVEit Automation uses random temporary filenames in its cache, not the original filenames. To instruct OfficeScan to scan all filenames, point your browser at its web interface and choose the following links: Clients, Scan Options, Real-Time Scan Settings, Scan Target, All scannable files.

Using Processes to Scan Files On Demand

Less commonly, MOVEit Automation can be used to individually scan files in its cache using a third-party antivirus program. To actively scan each file passing through MOVEit Automation, you would probably use the included `Run DOS Command.vbs` script or a derivation start the command-line utility provided by your antivirus client. This script runs a single command and errors out if a command-line antivirus client returns a code other than 0.

Alternatively, you could compose a script to invoke a COM interface of an antivirus client. This approach is more work, but could also supply MOVEit Automation with more information. If you use this approach, you must configure your real-time antivirus client to ignore the MOVEit Automation cache folder to avoid interference between the two scanning mechanisms.

Note: When setting files to scan in your Antivirus program, you must exclude `mic*.xml` `config/state/hash` files to improve the performance.

POP3 Sources

POP3 sources download email messages from a POP3 server. Each attachment in a message is considered a distinct file. The body of a POP3 email message is not considered a file, so an email message with no attachments is considered to have no files.

The fact that an email message may have multiple attachments makes POP3 sources unique from the point of view of Collect Only New Files and Delete Original processing. For example, if an email message has two attachments:

- report1.txt
- fig1.gif

and a task specifies that only *.txt attachments be downloaded, then only report1.txt will be processed. MOVEit Automation will note (in the POP3 host's corresponding State File) the fact that this report1.txt has been processed, and that fig1.gif has not been processed. Subsequent attempts to download *.txt files from this source will not collect report1.txt if Collect Only New Files is checked.

If Delete Original is checked, this message will not be deleted, because some other task may wish to download *.gif files. If some other task does download and successfully process fig1.gif, and Delete Original is marked for this task, then the task will see that all attachments for this message have now been processed, and the task will delete the message.

The ramifications of this processing are:

- Messages containing attachments with filenames that do not match any masks from any POP3 sources will never be deleted, even if Delete Original is checked in some or all sources.
- Messages containing no attachments will never be deleted, because they will never be processed.

It is therefore a good idea to periodically manually check the list of messages waiting on the server for this user, especially if the server is Internet accessible and therefore likely to receive unwanted bulk email.

GetMICConfig Utility

Included with MOVEit Automation is a command-line utility, GetMICConfig.exe, to retrieve the current configuration from a running copy of MOVEit Automation on the local machine. The ability to retrieve the current configuration useful for certain advanced tasks, such as generating custom reports.

Although the MOVEit Automation configuration is already stored in a disk file named miccfg.xml, this file is encrypted and is therefore not usable by external applications. And although a similar config export capability is available via MOVEit Automation Admin, MOVEit Automation Admin requires human interaction. By contrast, GetMICConfig is suitable for running from a script or batch file.

GetMICConfig is installed in the same directory as MOVEit Automation, typically \Program Files\MOVEit.

GetMICConfig syntax:

```
GetMICConfig -o outfile [-k]
```

where:

outfile The name of the desired output file. This file will contain the entire MOVEit Automation configuration in plaintext XML format. (This does not include the small number of settings, such as the license key, that are maintained in the registry and administered by the ***MICentralCfg*** (see "***MOVEit Automation Config Utility***" on page 354) program.)

-k Specifies that the old copy of *outfile*, if any, should be kept if GetMICConfig is unable to retrieve the settings. By default, GetMICConfig will delete any old *outfile* before connecting to MOVEit Automation.

The program exit code is:

0 if all OK
1 if command-line error
2 if communication error with MOVEit Automation
3 if bad response from MOVEit Automation
4 if could not write output file

This can be checked from a batch file using IF ERRORLEVEL.

Port Numbers

This technical document describes the IP port numbers used by MOVEit Automation. This information is provided to allow network administrators to configure firewalls appropriately.

Most protocols (marked with * below) allow non-standard ports to be used, so at some sites, some additional ports not mentioned here might need to be opened up on the firewall.

MOVEit Automation (MICentral.exe)

Port	Direction	Description
21 *	Outgoing	Typical port number for traditional, or "explicit" secure, FTP servers.
22 *	Outgoing	Typical port number for SSH servers.
25 *	Outgoing	Typical port number used for outbound email using SMTP (Simple Mail Transfer Protocol).
80 *	Outgoing	Typical port number for insecure HTTP servers, including MOVEit Transfer.
110 *	Outgoing	Typical port number used to fetch incoming email using POP3 (Post Office Protocol).
139	Outgoing	Port used for Windows filesystem shares. The protocol is known as SMB (Server Message Block).
443 *	Outgoing	Typical port number for secure HTTPS servers, including MOVEit Transfer.
990 *	Outgoing	Typical port number implicit secure FTP servers.
1433	Outgoing	MOVEit Automation generally connects to Microsoft SQL Server on this port, if SQL Server is the database engine.
3306	Outgoing	MOVEit Automation connects to MySQL on this port, if MySQL is the database engine.
3471-3473	Incoming	MOVEit Automation Admin connects to MOVEit Automation on these ports.
3472	Outgoing	On a failover system, MOVEit Automation connects to the other MOVEit Automation on this port.
3478-3479	Outgoing	MOVEit Automation connects to the AS/2 module, MIAS2.exe, on these ports.
Various	Incoming	Unpredictable port numbers > 1023 used for active mode FTP data transfers.

Port	Direction	Description
Various	Outgoing	Generally unpredictable port numbers > 1023 used for passive mode FTP data transfers.

MOVEit Automation Admin (miadmin.exe)

Port	Direction	Description
3471-3473	Outgoing	MOVEit Automation Admin connects to MOVEit Automation on these ports.

MOVEit Automation AS/2 helper (MIAS2.exe)

Port	Direction	Description
21 *	Outgoing	Typical port number for traditional, or "explicit" secure, FTP servers.
25 *	Outgoing	Typical port number used for outbound email using SMTP (Simple Mail Transfer Protocol).
80 *	Outgoing	Typical port number for insecure HTTP servers, including MOVEit Transfer.
110 *	Outgoing	Typical port number used to fetch incoming email using POP3 (Post Office Protocol).
443 *	Outgoing	Typical port number for secure HTTPS servers, including MOVEit Transfer.
990 *	Outgoing	Typical port number implicit secure FTP servers.
3478-3479	Incoming	MOVEit Automation connects to the AS/2 module, MIAS2.exe, on these ports.
Various	Incoming	Unpredictable port numbers > 1023 used for active mode FTP data transfers.
Various	Outgoing	Generally unpredictable port numbers > 1023 used for passive mode FTP data transfers.

MySQL (mysqld-nt.exe)

MySQL is one of the database engines that is supported by MOVEit Automation.

Port	Direction	Description
3306	Incoming	Database server listens on this port number. Only local (127.0.0.1) connections are needed.

* indicates a default port number that can be overridden.

System Internals

This technical document describes the registry settings used by MOVEit Automation. This information is rarely needed, as most of the settings here are managed by the MOVEit Automation Config program. However, there are a few rarely-used settings, documented in this style, that can be configured only by direct manipulation of the registry by a program such as RegEdit. These values normally do not appear in the registry at all.

Hidden Configuration File Settings

The following are hidden settings that can be set from within the MOVEit Automation XML configuration file (miccfg.xml):

XPath	Description
Settings/Globals/MaxMaxSimulTasks	The maximum allowed value for the MaxSimulTasks setting. If this does not exist in the config, then it defaults to 100.
Settings/Globals/ProcessFilesAddedInLoop	Whether an Advanced Task For loop should process files added in that loop. If this does not exist in the config, then it defaults to 0, which means that a file added by a script within a For loop will not be processed by further iterations of that loop. (However, it will be processed by other For loops.) A value of 1 causes script-added files to be processed by future iterations of the loop, likely causing an undesirable infinite loop. MOVEit Automation 7.1 and previous versions defaulted this value to 1; 7.1.1 and subsequent versions default it to the less dangerous 0 value.
Settings/Globals/KeepAliveIntervalSecs	The KeepAliveIntervalSecs setting is used to prevent dropped connections between the MOVEit Automation Service and the Admin console when MOVEit Automation is used in a network environment that monitors idle connections. When the KeepAliveIntervalSecs setting is set to the default value of 540 seconds (9 minutes), the Admin console sends a keep alive command (at a 9 minute interval) to the MOVEit Automation service. A value of -1 for KeepAliveIntervalSecs will disable keep alive commands completely.

XPath	Description
Settings/Globals/KeepAliveTimeoutSecs	When set to the default value of 1800 seconds (30 minutes), the KeepAliveTimeoutSecs setting ignores a socket timeout of up to 30 minutes. A value of -1 for KeepAliveTimeoutSecs will disable keep alive socket timeouts, but keep alive commands will continue to be sent.

Registry Settings

The following values appear under HKEY_LOCAL_MACHINE \ Software \ Standard Networks \ MOVEitCentral:

Name	Type	Description
CertIssuer	String	The name on the SSL certificate that MOVEit Automation uses to encrypt communications with MOVEit Automation Admin.
CertSerial	String	The serial number (typically "00") of the SSL certificate that MOVEit Automation uses to encrypt communications with MOVEit Automation Admin.
DeleteCacheInsecurely	DWORD	If 0, then at the end of a task, MOVEit Automation will overwrite temporary files with random bytes before deleting them. If 1, MOVEit Automation will delete the files.
DMZBigBufSize	DWORD	The size, in bytes, of the large buffer used by the MOVEit Transfer client. The default is rather large at 104857600 (100 MB), to accommodate very large responses from MOVEit Transfer. You may wish to decrease this value, perhaps to 1000000, in order to have MOVEit Automation use less memory, and start up and shut down faster.
EmailFrom	String	The "From:" email address used in error emails.
EmailServer	String	The host name or IP address of the SMTP server used for error emails.
ErrorEmail	String	The "To:" email address(es) used in error emails.
FlushLogAlways	DWORD	If 1, MOVEit Automation will flush the debug log to disk after every write. This greatly slows performance, and should be used only if you want to be certain to have the entire log file available if MOVEit Automation crashes.

Name	Type	Description
HashKey	String	This is the encrypted value used as the basis of an encryption key for hash-chaining the records in the database, in order to be able to detect tampering. If you are performing a MOVEit Automation Failover or a migration operation, do not copy the "HashKey" value from the registry into any of the fields on the "Tamper" tab in the MOVEit Automation Config utility. Instead copy "HashKey" registry values from one registry to another to avoid reencrypting an encrypted value.
LicenseKey	String	The license key that enables this copy of MOVEit Automation.
MaxHTTPSessionsPerServer	DWORD	The maximum number of HTTP connections that can be maintained simultaneously to a given webserver. The default is 100, and there is rarely a need to change this.
MinAdminVersion	String	The minimum version of MOVEit Automation Admin required, in a format like 4.5.0.0. If present, this non-standard setting overrides the value coded into MOVEit Automation.
MySQLDir	String	The directory in which MySQL is installed; typically C:\MySQL.
MySQLRootPW	String	The password to the MySQL database user named root; strongly encrypted.
RequireSSL	DWORD	1 if connections from MOVEit Automation Admin are required to be encrypted with SSL; else 0 if they are not encrypted.
SchedDisabled	DWORD	1 if the task scheduler should be disabled when MOVEit Automation starts; else 0 if the scheduler should run normally.
SSHBufferSize	DWORD	The size, in bytes, of the buffer used by the SSH client. The default will work well for almost all cases. This value should be set only if needed for compatibility with unusual SSH servers, or if a very large value is needed to enhance WAN performance.
StatsDSN	String	The ODBC Data Set Name of the database used by MOVEit Automation, if StatsUseConnStr is missing or 0. This is used when MySQL is the database engine. This is nearly always "DSN=micstats;".

Name	Type	Description
StatsConnStr	String	The database connection string, used if StatsUseConnStr is 1. At runtime, a reference to the macro [DBPassword] is filled in with the actual password, decrypted from StatsConnStrPW.
StatsConnStrPW	String	The encrypted password to the database user, used if StatsUseConnStr is 1 and if the macro [DBPassword] appears in StatsConnStr.
StatsUseConnStr	DWORD	1 if StatsConnStr should be used for database connection settings. This is used for Microsoft SQL Server. If the setting is 0 or is missing, StatsDSN is used instead.
StoreLocation	DWORD	A numeric value used to find the certificate that MOVEit Automation uses to encrypt communications with MOVEit Automation Admin; nearly always 0x00020000.
StoreName	String	The name of the certificate store containing the certificate that MOVEit Automation uses to encrypt communications with MOVEit Automation Admin; nearly always "My".
SuppressLowFragHeap	DWORD	If 1, this non-standard setting prevents MOVEit Automation from using the "low fragmentation heap" for memory management. Use this only if you have a specific reason for doing so.
TempDir	String	The name of the parent temporary folder used for cache files; typically C:\TEMP\MIC.
Update	DWORD	An arbitrary value changed by the configuration program to alert MOVEit Automation that the registry values have changed.
VirusHandlingIDed	DWORD	A numeric code indicating how MOVEit Automation should react to files that appear to be infected with a specific, identifiable virus.
VirusHandlingNotIDed	DWORD	A numeric code indicating how MOVEit Automation should react to files that appear to be infected with a virus, but we don't know which virus.

The following values appear under HKEY_LOCAL_MACHINE \ Software \ Standard Networks \ MOVEitCentral \ Install. They are used only by the install program:

Name	Type	Description
MICAdminUserName	String	The username of the MOVEit Automation Admin user created during the install.

Name	Type	Description
MICAdminUserWhetherCreated	DWORD	1 if the install program created the above user, else 0 if an existing user was selected.
ServiceUserName	String	The username of the user under which the MOVEit Automation service is running.
ServiceUserWhetherCreated	DWORD	1 if the install program created the above user, else 0 if an existing user was selected.

The following values appear under HKEY_LOCAL_MACHINE \ Software \ Standard Networks \ MOVEitCentral \ Resil. They apply only to a failover installation:

Name	Type	Description
AdminPassword	String	The password to the user on the remote MOVEit Automation system (strongly encrypted).
AdminUser	String	The username of the MOVEit Automation user on the remote system.
HostsToPing	String	A comma-separated list of hosts to ping before a secondary assumes the primary role.
Node	DWORD	The number of this node: 0 for non-failover; else 1 or 2.
OtherHost	String	The hostname or IP address of the other MOVEit Automation.
StartupRole	DWORD	The failover role that MOVEit Automation should assume at startup; 1 means primary and 2 means secondary. This is ignored if the node number is 0.
SuppressDBRep	DWORD	Whether replication of the database should be suppressed. The default is 0, which means that in failover mode, the database will be replicated. 1 means to not replicate the database in failover mode, and should be used only by advanced users. 1 should be specified if, for instance, both nodes are connected to the same clustered database, where the cluster is providing the high availability normally provided by MOVEit Automation itself.

MOVEit Automation Error Codes

The following table lists the error codes that are generated by MOVEit Automation.

Code	Error Description
0	SIL_ERROR_NONE
100	SIL_ERROR_INTERNAL
200	SIL_ERROR_OTHER
2010	SIL_ERROR_INVALID_USERNAME
2012	SIL_ERROR_NOT_SIGNED_ON
2020	SIL_ERROR_NOPERM
2025	SIL_ERROR_INVALID_CREDENTIALS
2040	SIL_ERROR_DATABASE_UPDATE
2042	SIL_ERROR_DATABASE_OPEN
2043	SIL_ERROR_DATABASE_MISC
2044	SIL_ERROR_DATABASE_CONN
2046	SIL_ERROR_DATABASE_READ
2050	SIL_ERROR_INVALIDUSER
2056	SIL_ERROR_INVALID_FOLDERID
2058	SIL_ERROR_INVALID_FILEID
2060	SIL_ERROR_INVALID_FILETYPE
2068	SIL_ERROR_INVALID_PARENTID
2069	SIL_ERROR_INVALID_PARENTINHERITRIGHTS
2070	SIL_ERROR_INVALID_FOLDERTYPE
2071	SIL_ERROR_INVALID_FOLDERNOTETYPE
2072	SIL_ERROR_INVALID_INSTID
2074	SIL_ERROR_INVALID_USERPERM
2075	SIL_ERROR_INVALID_FOLDERNOTETIME
2076	SIL_ERROR_INVALID_EMAILADDRESS
2080	SIL_ERROR_INVALID_FOLDERSYSTEMTYPE
2085	SIL_ERROR_INVALID_FOLDERCLEANTIME
2086	SIL_ERROR_INVALID_FOLDERCLEANTYPE
2090	SIL_ERROR_DUPLICATE_FOLDERNAME

2092	SIL_ERROR_DUPLICATE_FILENAME
2100	SIL_ERROR_INVALID_FOLDERNAME
2102	SIL_ERROR_INVALID_FILENAME
2210	SIL_ERROR_FILESYSTEM_FOLDERCREATE
2220	SIL_ERROR_FILESYSTEM_FOLDERDELETE
2230	SIL_ERROR_FILESYSTEM_FILECREATE
2234	SIL_ERROR_FILESYSTEM_FILECOPY
2240	SIL_ERROR_FILESYSTEM_FILEDELETE
2244	SIL_ERROR_FILESYSTEM_FILEOPEN
2310	SIL_ERROR_INVALID_XMLREQUEST
2320	SIL_ERROR_INVALID_TRANSACTION
2410	SIL_ERROR_INVALID_PASSWORD
2412	SIL_ERROR_INVALID_USERREALNAME
2414	SIL_ERROR_INVALID_CLIENT_CERT
2415	SIL_ERROR_NO_CLIENT_CERT
2416	SIL_ERROR_CLIENT_CERT_REQUIRED
2430	SIL_ERROR_UPLOAD_INVALIDPARMS
2500	SIL_ERROR_INVALID_INSTFORMRESP
2510	SIL_ERROR_INVALID_INSTHISTTIME
2520	SIL_ERROR_INVALID_HOSTPERMIT
2522	SIL_ERROR_INVALID_HPPRIORITY
2524	SIL_ERROR_INVALID_HPPERMITID
2526	SIL_ERROR_INVALID_HPHOST
2528	SIL_ERROR_INVALID_HPRULE
2540	SIL_ERROR_INVALID_HOST
2670	SIL_ERROR_INVALID_COMMENT
2674	SIL_ERROR_DUPLICATE_FOLDERUSER
2678	SIL_ERROR_INVALID_FOLDERUSER
2680	SIL_ERROR_INVALID_PASSPHRASE
2684	SIL_ERROR_DUPLICATE_INSTNAME
2688	SIL_ERROR_REGISTRY_KEYCREATE
2692	SIL_ERROR_INVALID_INSTNAME
2696	SIL_ERROR_REGISTRY_KEYDESTROY
2700	SIL_ERROR_UPLOAD_FILECREATE

2800	SIL_ERROR_CANT_ACCESS_SERVER
2801	SIL_ERROR_SERVER_APP
2850	SIL_ERROR_INVALID_PARAMETER
2860	SIL_ERROR_SETTINGS_OUT_OF_DATE
2904	SIL_ERROR_ILLEGAL_TRANSLATION
2908	SIL_ERROR_FAILED_TRANSLATION
2954	SIL_ERROR_BROWSER_FILEPUSH
2958	SIL_ERROR_INVALID_FILEBUNDLETYPE
2964	SIL_ERROR_BUNDLE_EMPTYFILES
2968	SIL_ERROR_INVALID_DEBUGLEVEL
2972	SIL_ERROR_INVALID_URL
2974	SIL_ERROR_MULTI_SIGNON_PROHIBITED
2976	SIL_ERROR_ILLEGAL_USERATHOST
2978	SIL_ERROR_UNAUTHORIZED_USER
2980	SIL_ERROR_UPLOAD_EMPTYFILE
2984	SIL_ERROR_INVALID_FOLDERACCESS
2988	SIL_ERROR_INVALID_NOTE
3108	SIL_ERROR_INVALID_GROUPNAME
3112	SIL_ERROR_DUPLICATE_GROUPNAME
3116	SIL_ERROR_INVALID_GROUPID
3200	SIL_ERROR_EXPIRED_SESSION
3201	SIL_ERROR_FAILED_WEB_REQUEST
3202	SIL_ERROR_PROXY_NOT_AUTHORIZED
3300	SIL_ERROR_ENCRYPTION
3400	SIL_ERROR_LICENSING
3500	SIL_ERROR_EMAIL
3600	SIL_ERROR_DOWNLOAD
3601	SIL_ERROR_HASH_CHECK_FAILED
3610	SIL_ERROR_DELETE_FILE
3611	SIL_ERROR_RENAME_FILE
3700	SIL_ERROR_SCRIPT
3800	SIL_ERROR_UPLOAD
3801	SIL_ERROR_CHUNKED
3810	SIL_ERROR_RESUME_UPLOAD

3820	SIL_ERROR_RESUME_DOWNLOAD
3850	SIL_ERROR_CANCELLED
3900	SIL_ERROR_LICENSE
3960	SIL_ERROR_TAMPERED
4000	SIL_ERROR_ALREADY_RUNNING
4001	SIL_ERROR_NOT_RUNNING
4002	SIL_ERROR_COULD_NOT_START
4010	SIL_ERROR_ALREADY_CHECKED_OUT
4011	SIL_ERROR_NOT_CHECKED_OUT
4015	SIL_ERROR_CANNOT_CHECK_IN
4020	SIL_ERROR_DOES_NOT_EXIST
4021	SIL_ERROR_CANNOT_SAVE
4022	SIL_ERROR_NOT_AVAILABLE
4023	SIL_ERROR_USE_ALTERNATE
4030	SIL_ERROR_STOPPED_BY_USER
4040	SIL_ERROR_PROXY_PROBLEMS
4050	SIL_ERROR_BAD_CMD_SECONDARY
4060	SIL_ERROR_TOO_MANY_FAILOVER
4070	SIL_ERROR_BAD_CMD_RESULT
4080	SIL_ERROR_MISSING_CFG_ITEM
4090	SIL_ERROR_DUPLICATE_ITEM
4100	SIL_ERROR_SSL_WEAK
4110	SIL_ERROR_INCOMPATIBLE_UNICODE_CHAR
4204	SIL_ERROR_NO_FILE_MATCH
5000	SIL_ERROR_IGNORE_FILE
5010	SIL_ERROR_NO_ACTION
5100	SIL_ERROR_MISSING_PARAM
5110	SIL_ERROR_MACRO_ERROR
5120	SIL_ERROR_MAX_EXCEEDED
5130	SIL_ERROR_ALL_HOSTS_DOWN
5140	SIL_ERROR_CANT_START_TASK
5200	SIL_ERROR_TAMPER_INTERNAL
5300	SIL_ERROR_TRANSFORM_EXCEPTIONS
5500	SIL_ERROR_CENTRALCLT_TIMEOUT

5510	SIL_ERROR_KEEPALIVE_TIMEOUT
5600	SIL_ERROR_UNC_SHARES_NOT_INITIALIZED
5601	SIL_ERROR_UNC_HAS_ACTIVE_TASKS
5602	SIL_ERROR_UNC_ALREADY_EXISTS
6000	SIL_ERROR_TRY_ANOTHER_HOST
6100	SIL_ERROR_AV_VIRUS_DETECTED
6101	SIL_ERROR_AV_CANNOT_START
6102	SIL_ERROR_AV_CANNOT_SCAN
6103	SIL_ERROR_AV_CANNOT_FINISH
6110	SIL_ERROR_AV_LOST_CONN
6111	SIL_ERROR_AV_BAD_RESP
6120	SIL_ERROR_AV_MAYBE_VIRUS
6121	SIL_ERROR_AV_NO_RT_SCAN
6122	SIL_ERROR_AV_UNKNOWN_PKG
6123	SIL_ERROR_AV_MISIDENTIFIED
6124	SIL_ERROR_AV_CANT_FIND_REC
6125	SIL_ERROR_AV_YES_VIRUS
6150	SIL_ERROR_DLP_DETECTED
6200	SIL_ERROR_CANT_FIND_KEY
6210	SIL_ERROR_CERTIFICATE
6300	SIL_ERROR_INVALID_TASK
6310	SIL_ERROR_CANT_SYNC_COMPARE
6320	SIL_ERROR_DELETE_FOLDER
6321	SIL_ERROR_CREATE_FOLDER
6330	SIL_ERROR_WRONG_VERSION

Local Mail Relay

Consider using Windows Server's IIS SMTP server as a local mail relay on your MOVEit Automation system any of the following conditions apply.

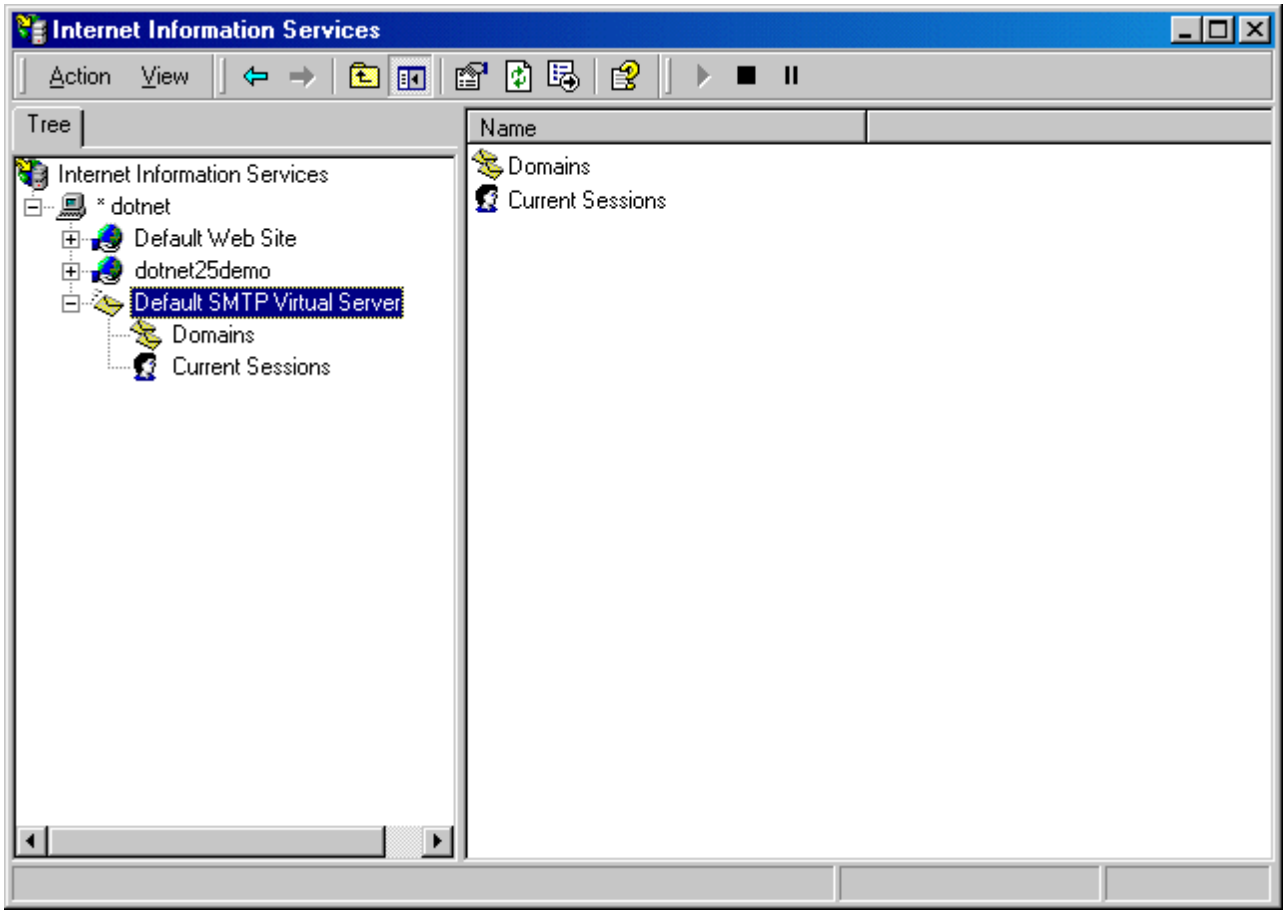
- You have many tasks that send email notifications
- You have a slow or busy mail server
- Your mail server requires an advanced method of authentication for relaying mail

MOVEit Automation can send files by email, or send email alerts on completion of a task. These notifications happen as part of the task run, so MOVEit Automation must wait for each message to be sent before the task is completed. When MOVEit Automation is dealing with busy email servers or large numbers of recipients, tasks may run significantly longer than normal. To keep MOVEit Automation from having to wait for each message to be sent, we can instead spool these messages to a local SMTP server which will queue up the messages and send them out to "real" email servers when they are better able to accept the traffic.

Instructions

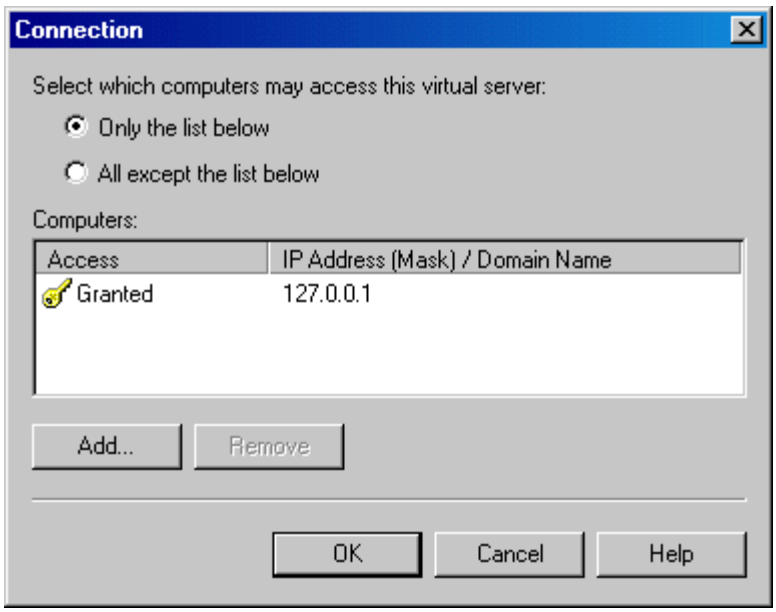
Step 1

Ensure that you have the SMTP component installed in your local IIS server. When installed correctly, you should see a Default SMTP Virtual Server node in your IIS administration window under the local machine. If you do not have the SMTP component installed, or do not have IIS installed, you will need to install them through the Add/Remove Windows Components option of the Add/Remove Programs window, which can be found in the Control Panel.



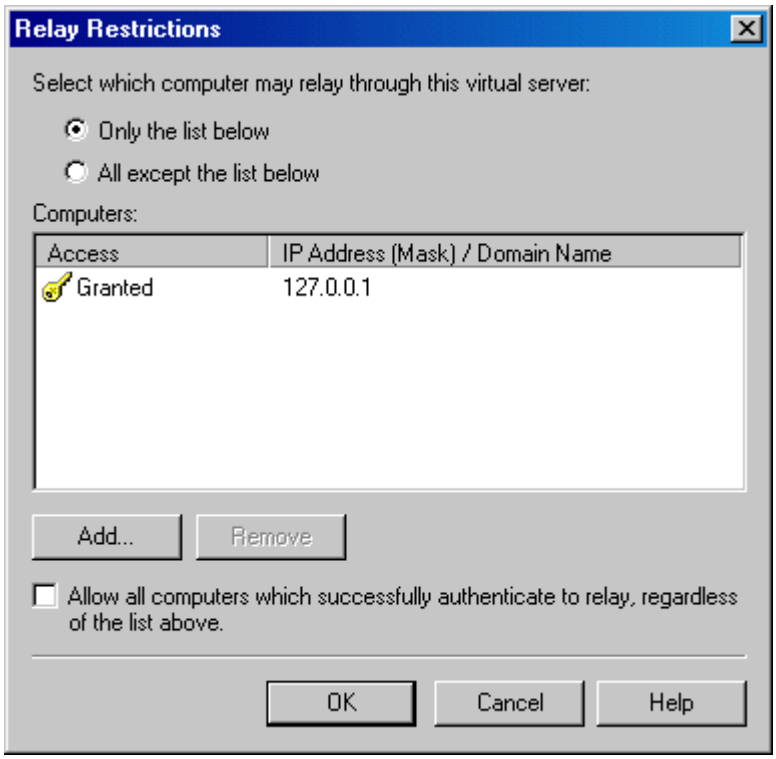
Step 2

Open up the properties window of the SMTP service by right-clicking on the SMTP service node and selecting Properties. In the properties window, select the Access tab. In the Access tab, open the Connection Control window by clicking on the Connection button in the Connection Control section. Restrict access to the SMTP server by selecting the Only The List Below option and adding the localhost IP address 127.0.0.1 to the access list. Click OK to exit the window.



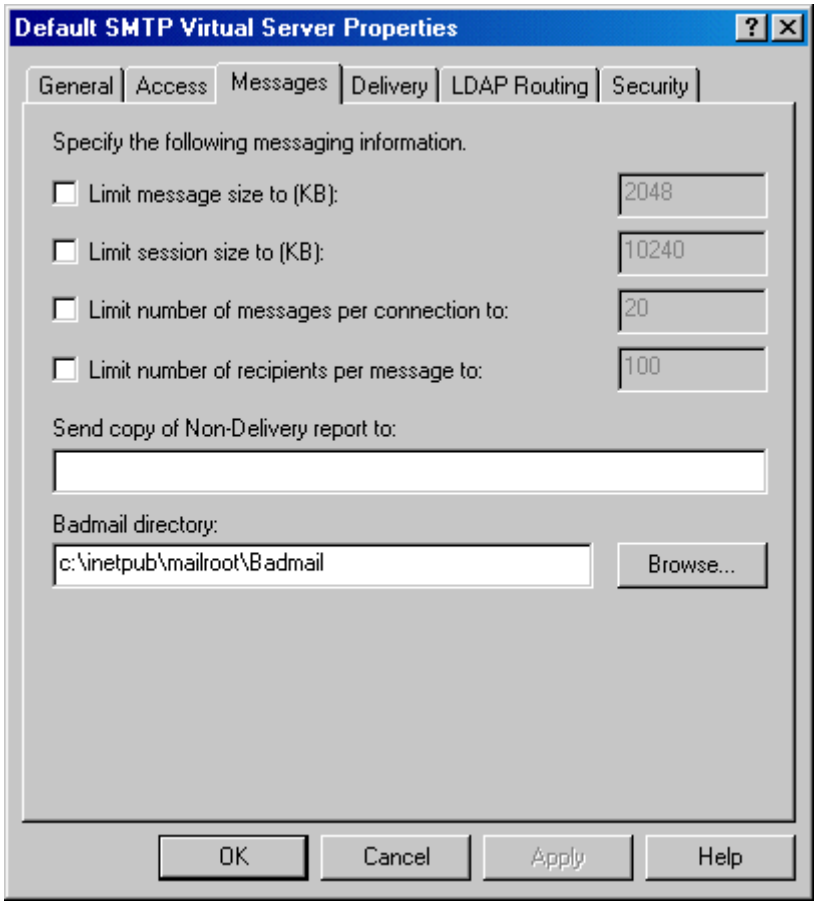
Step 3

In the Access tab, open the Relay Restrictions window by clicking the Relay button in the Relay Restrictions section. Restrict relay access to the SMTP server by selecting the Only The List Below option and adding the localhost IP address 127.0.0.1 to the access list. Make sure the Successful Authentication Relay option is turned off. Click OK to exit the window.



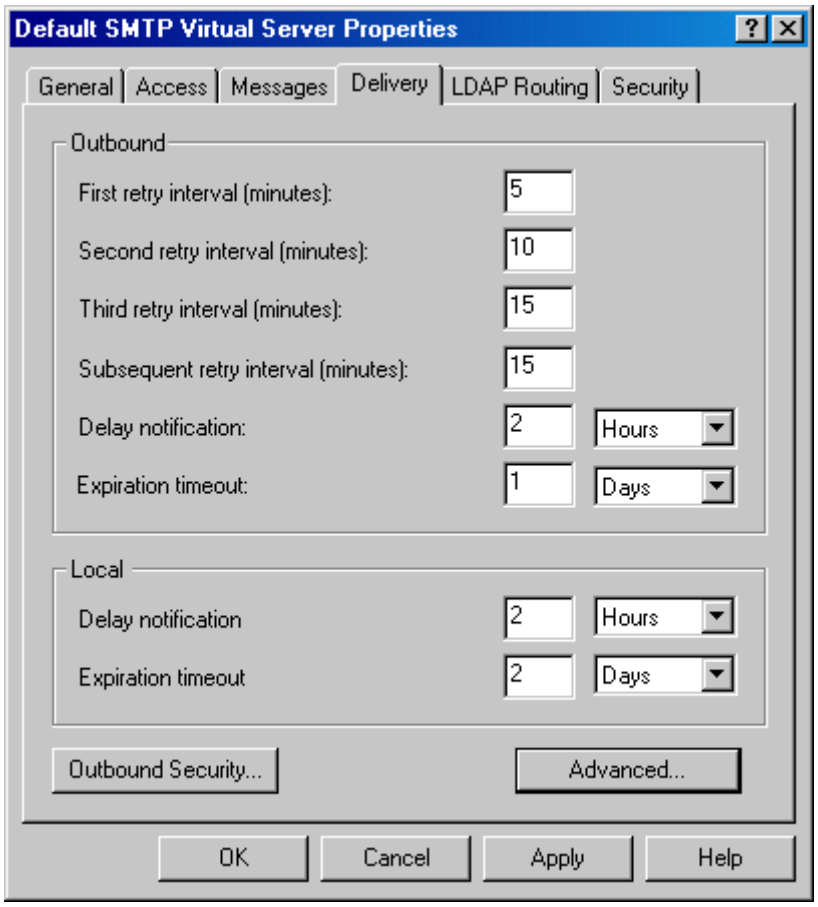
Step 4

In the Properties window, switch to the Messages tab. In the Messages tab, turn off all the message limits.



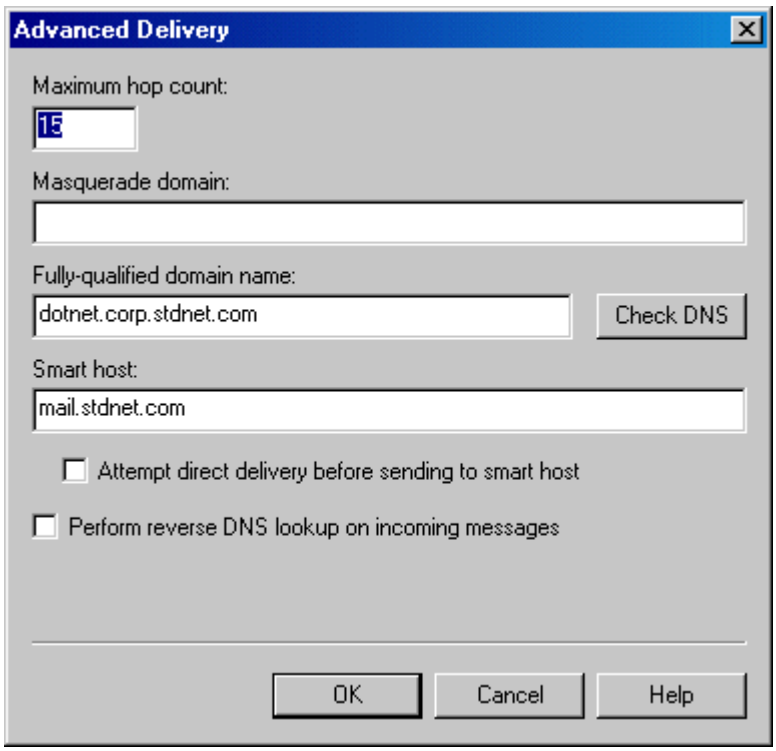
Step 5

In the Properties window, switch to the Delivery tab. In the Delivery tab, change the default delivery intervals and timeouts to smaller values. Recommended values are shown in the image below.



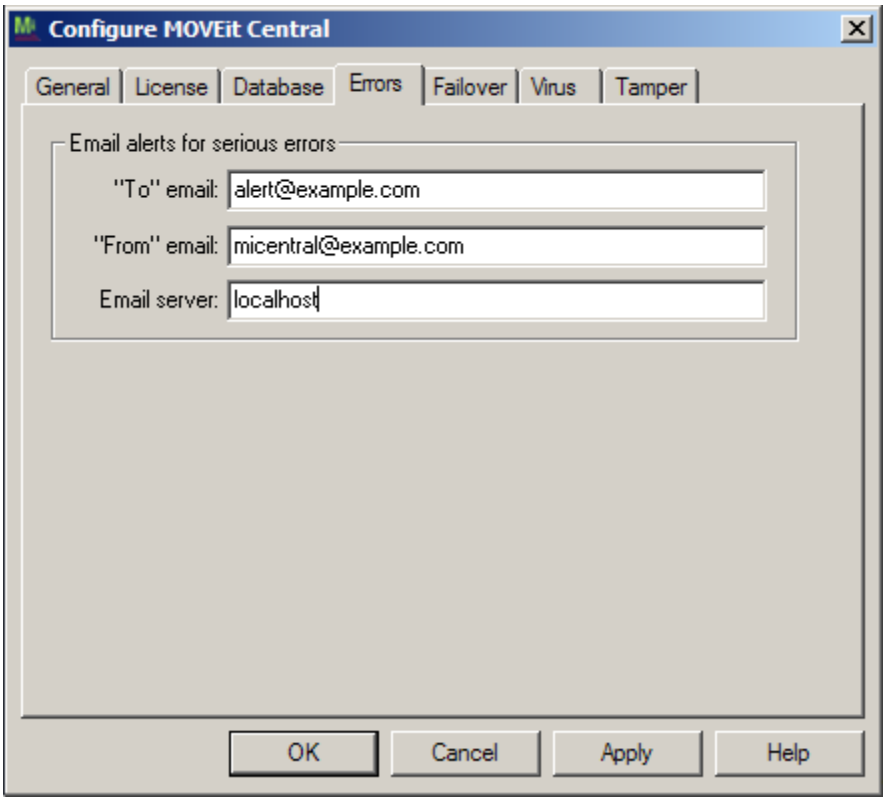
Step 6

In the Delivery tab, open the Advanced Delivery Options window by clicking the Advanced button. Set the Fully Qualified Domain Name setting to the name of your MOVEit Automation server. Set the Smart Host setting to the name of your main SMTP server. Click OK to exit the window. Configuration of the SMTP server is now complete. Click OK in the Properties window and make sure the SMTP service is started.



Step 7

The final step is configuring your Central server to use the new local SMTP service. First, open the MOVEit Automation Config program (Start -> Programs -> MOVEit Automation) and switch to the Errors tab. Enter localhost as the "Email server" value. Click OK to exit the Config program. The change should happen immediately; no restart is required.



Next, sign on to the Central server using MOVEit Automation Admin and switch to the Hosts tab. If you have an SMTP host configured for your outgoing email server, open it by double-clicking the host entry and change the Host value to localhost. Remove any authentication values as they will not be needed.

Tuning

You will probably want to tinker with the "outgoing connection limit" (default is 1000) if one of your goals is to keep MOVEit Automation from overloading your "real" mail server. (Typical "throttled" values are from 1-5.) To alter this setting, open the SMTP properties, go to the "General" tab and open the "connection" dialog.

Finished

Your local SMTP relay server should now be set up, and your MOVEit Automation server configured to use it. If you have any questions about or problems with this process, please contact the MOVEit ***technical support*** (***http://www.ipswitchft.com/company/contactsupport.aspx***) department for additional assistance.

Repeat on Each Node if Running MOVEit Automation Failover

If you are running MOVEit Automation Failover, you must repeat this procedure on each node in the cluster.

Troubleshooting

Problem: Cannot connect to local mail relay.

- Solution 1: Open the "Services" from "Start | Program Files | Administrative Tools". Make sure the "Simple Mail Transport Protocol" service is started and that it is set up to start "Automatically."
- Solution 2: Open the "Internet Services Manager" from "Start | Program Files | Administrative Tools". Make sure the "Default SMTP Virtual Server" is NOT "stopped".
- Solution 3: Open the "Internet Services Manager" from "Start | Programs | Administrative Tools". Right-click on "Default SMTP Virtual Server" and select Properties. In the General tab make sure the IP Address is set for "All Unassigned".
- Solution 4: Go to the command line and type "netstat -a -n". Look for any TCP entries with a local address ENDING with ":25". If there are none, the SMTP server failed to bind to its listening port; reboot the server.

If MOVEit Automation reports that it is sending email OK, but the mail messages are not actually reaching their destination, open the local SMTP server queue folder and look for messages there which correspond with your MOVEit Automation messages. (The queue folder is usually named something like "c:\inetpub\mailroot\queue".)

Problem: Mail is being queued on the local SMTP server and is not being delivered.

- Solution 1: Make sure your SMART HOST contains the value which used to be the Email Server field in your MOVEit Automation configuration.
- Solution 2: Make sure the "Attempt Direct Delivery" box (near the Smart Host setting) is NOT CHECKED.
- Solution 3: Look for entries in your SYSTEM event log from SMTP or SMTPSVC which complain about "DNS" problem. If you see events like these, change the SMART HOST (described above) to an IP address surrounded by square brackets, for example [66.170.5.142].

MessageWay CLI

In MOVEit Automation, the built-in MessageWay Translator script uses a command-line program, `xformviamway`, to communicate with a MessageWay server. In ordinary use, MOVEit Automation administrators can rely upon the built-in script to handle all interaction with the command-line program.

Advanced MOVEit Automation administrators might want to use `xformviamway` to handle unusual or complex scenarios. This section describes how to use the program. It is located in \Program Files\MOVEit.

See also:

- Common Applications - MessageWay Translation
- ***Configuring Tasks - Processes/Scripts - Built-In - MessageWay Translation*** (see "***MessageWay Translation***" on page 123)."

The program `xformviamway.exe` sends a single file to MessageWay and, after MessageWay has completed processing the file, receives the resulting output files, writes two status files, and terminates.

Program Arguments

The program is invoked as follows:

`xformviamway.exe -i` *infile* `-o` *outfile* `-c` *completionfile*

infile is the name of an input file giving such information as how to connect to the MessageWay server, where the input file is, and where the output files should be created. See below for a description of the format of this file.

outfile is the name of an output file to be created by xformviamway, giving the status of the completed translation, and listing the names of the data and report files returned from MessageWay. See below for a description of the format of this file.

completionfile is the name of a completion file to be created by xformviamway when it finishes. The contents of this file are unimportant; the creation of this file indicates that xformviamway has completed writing *outfile*.

Input File

The input file is a text file in `.INI` file format, with one input value per line.

Sample file

```
[Params]
Host=172.16.23.204
SSL=False
Port=6280
SSLFingerprint=
User=micentral
Password=G7z3fN9wP
Recipient=translate:moveit
Sender=X850TEST
MIMEType=
ExceptionsInsteadOfData=True
FilenameToProcess=C:\TEMP\MIC\c88-0002\atc10001.tmp
OriginalFilename=X850test-MultOut.txt OutputDir=C:\TEMP\MIC\c88-0002
MaxSeconds=300
PollIntervalSeconds=5
TraceFilename=
ForceAtLeastSeconds=20
```

Input Values

All values are required except where marked optional.

Name	Description
Host	The hostname or IP address of the MessageWay server.
SSL	True if SSL should be used, else False. Note: if you use SSL, you must specify the MessageWay server's certificate's fingerprint in SSLFingerprint. If the server is on the same computer as MOVEit Automation, you can safely specify False here and avoid having to know the certificate's fingerprint.
Port	The TCP port to which the program should connect. Typically 6280 if not SSL, or 6243 if SSL.
SSLFingerprint	(Optional) Hexadecimal fingerprint (MD5 or SHA1) of the server's certificate. Required if SSL is True. (There is no way to specify that any certificate should be accepted.) The string consists of groups of 2 hex characters separated by spaces.
User	The MessageWay username. This user must have sufficient permission to access the recipient location.
Password	The password of the MessageWay user.

Name	Description
Recipient	Destination (in MessageWay terminology) of the translated files. Typically this will be something like `translate:moveit`, which means that there must be a translation location named `translate` and a mailbox named `moveit` configured in MessageWay. The specified user must have sufficient access to these locations.
Sender	An arbitrary sender's name. The MessageWay translation engine may base its translation partly on the sender's name.
MIMEType	(Optional) An arbitrary MIME type string.
ExceptionsInsteadOfData	Indicates how the script behaves when any exception occurs (meaning poorly formatted data). Values: True - no data files should be returned if an exception occurs; instead, the exception report files are returned. False (default) - only data files are returned. If False and exceptions do occur, you will have to look up the exception reports directly through MessageWay.
FilenameToProcess	The full path to the input data file to process. Note: MessageWay is not informed of this filename; instead, MessageWay is told that the filename is the value specified in OriginalFilename.
OriginalFilename	The filename to provide to MessageWay. Do not include a path in this filename. Depending upon how MessageWay is configured, the type of processing done may be partially determined by this filename.
OutputDir	The full path of the directory into which xformviamway should write its data and report files. For example, `C:\data\xlate`.
MaxSeconds	(Optional) The maximum number of seconds to wait for MessageWay to process the file. A value of 0 means no limit. Can include macros. Defaults to 7200 (two hours).
PollIntervalSeconds	(Optional) The number of seconds to wait between queries to MessageWay to determine whether processing has completed. May include macros. Defaults to 5 seconds.
TraceFilename	(Optional) The full path to a file which will receive a detailed trace log of xformviamway's communications with the MessageWay server. For example, C:\tmp\MWTrace.txt. Use this parameter only to debug problems interacting with the MessageWay server.

Name	Description
ForceAtLeastSeconds	(Optional) A rarely-used parameter which specifies the minimum amount of time that xformviamway should take before responding. Specified as an integer number of seconds. If specified, xformviamway will wait until at least ForceAtLeastSeconds seconds have passed before returning, even if the processing was complete before that amount of time. This parameter was implemented to allow testing of progress bars.

Output File

The xformviamway `outfile` is an XML file describing the results of the translation. The actual data and report files resulting from the MessageWay processing are separate files which are pointed to by `outfile`. Here is a sample file:

```
<Output>
  <InputProcessingStatus>Accepted</InputProcessingStatus>
  <RFiles>
    <RFile>
      <MessageID>2010080413120504btdb</MessageID>

<CacheFilename>C:\TEMP\MIC\c88-0002/2010080413120504btdb.tmp</CacheFilename>
      <FileType>Output</FileType>
      <Filename>M2010080413120504btdb.dat</Filename>
    </RFile>
    <RFile>
      <MessageID>2010080413120504c55l</MessageID>

<CacheFilename>C:\TEMP\MIC\c88-0002/2010080413120504c55l.tmp</CacheFilename>
      <FileType>Output</FileType>
      <Filename>M2010080413120504c55l.dat</Filename>
    </RFile>
    <RFile>
      <MessageID>2010080413120504deg9</MessageID>

<CacheFilename>C:\TEMP\MIC\c88-0002/2010080413120504deg9.tmp</CacheFilename>
      <FileType>Output</FileType>
      <Filename>M2010080413120504deg9.dat</Filename>
    </RFile>
    <RFile>
      <MessageID>2010080413120504egd8</MessageID>

<CacheFilename>C:\TEMP\MIC\c88-0002/2010080413120504egd8.tmp</CacheFilename>
      <FileType>Report</FileType>
      <Filename>2010080413120504ac2j.txt</Filename>
    </RFile>
  </RFiles>
  <ErrorCode>0</ErrorCode>
  <ErrorDescription></ErrorDescription>
  <ShouldRetry>false</ShouldRetry>
  <OriginalMessageID>2010080413120504ac2j</OriginalMessageID>
</Output>
```

The XML tags are described below.

Name	Description

Name	Description
InputProcessingStatus	The processing status as returned by MessageWay. The possible values are: § Accepted § Rejected § Accepted with errors § Partially accepted § *(empty)* An empty value usually indicates an error.
RFile	Each RFile node represents a single file returned from MessageWay. The subnodes are: § MessageID - The Message ID of the file inside MessageWay. This is typically useful only if you are going to look up the message from the MessageWay Dashboard. § CacheFilename - The filename of a file created by MessageWay. This file will be in the OutputDir specified in the input file. § FileType - Either Data or Report. MessageWay Data files contain the translated output. A single input file might result in multiple data output files. MessageWay Report files contain descriptions of the processing of the input file, including detailed error messages if the input file contained incorrect data. § Filename - The name of the file (message) as it is known inside MessageWay.
ErrorCode	An error code giving the overall success of the process. 0 means success. Non-zero values mean failure. Generally, translation runs that result in exceptions due, for instance, to invalid input data result in an error code of 0. Non-zero errors are normally returned only if xformviamway cannot connect to or login to the MessageWay error, or if the MessageWay server returns an ill-formatted response.
ErrorDescription	An error description elaborating on the error that causes a non-zero ErrorCode. If ErrorCode is 0, ErrorDescription will be empty.
ShouldRetry	true if the xformviamway run failed, but in a way that implies that a subsequent run may succeed. For instance, if xformviamway cannot not connect to the MessageWay server, it returns a ShouldRetry of true, because the MessageWay server may be down only temporarily. ShouldRetry is false if the xformviamway run succeeded, or if it failed in a way that implies that a retry is unlikely to succeed. ShouldRetry is intended to be used by retry logic of the process invoking xformviamway.

Name	Description
OriginalMessageID	The message ID assigned by MessageWay to the input file. This can be useful for analysingThe message ID assigned by MessageWay to the input file. This can be useful for analysing message flow via MessageWay Dashboard. OriginalMessageID will be empty if MessageWay did not accept the input file - for example, if the user could not login.

Return Code

The program returns 0 if processing succeeded; this includes translation that resulted in exceptions. The program returns a non-zero error code upon failure.

Database

In this section:

Schema

This topic describes how MOVEit Automation keeps track of histories of task runs and file transfers. MOVEit Automation manages its database automatically, so very few sites will need the information contained in this topic. Both MySQL and Microsoft SQL Server are supported.

MOVEit Automation uses an ODBC-compliant database to store statistics and status information on task runs and file transfers. To configure the database, see ***MOVEit Automation Config Utility*** (on page 357). See also ***Trimming the database*** (see "***Trimming***" on page 301).

The database contains the following tables:

Stats

Stats contains one record for each attempt to send a file.

Field Name	Type	Description
ID	bigint	A record ID which is incremented automatically for each record added to the table. Primary key.
LogStamp	char	Date/time of record
TaskID	int	The ID of the task.
Node	int	The failover node number of the copy of MOVEit Automation that logged this record. This is 0 if failover is not being used.
NominalStart	char	The time when the task started or was scheduled to start. There is a key that consists of TaskID and NominalStart.
Action	char	The action being logged. Values: § `get` - logged only if there is an error retrieving a file. Successful gets are not logged; instead, we wait until sending or processing the file before logging it. § `send` - logged after trying to send a file. Most statistics records are of this type. § `process` - logged after running a script. § `delete` - logged if there is an error deleting a file. § `internal` - logged if there is an internal error.
SourceHost	char	The source host of the file. This is the "friendly" name configured in MOVEit Automation Admin, not the IP domain name. For process records, the name of the host if per-file, or blank if per-task.
SourceFilename	char	The filename of the source file. On send records, for process-created files, the script-assigned name. For unzipped files, reflects the original file. On process records, the name of the source file if per-file, or blank if per-task.

Field Name	Type	Description
SourceFilenameOnly	char	The filename of the source file, without the pathname.
SourceFileID	char	The ID of the source file if it originated from MOVEit Transfer. If not, value is empty.
SourceNBytes	double	The number of bytes in the source file. Always 0 for process-created files. For unzipped files, reflects the original file's size.
SourceDuration	double	The number of seconds the download took. Always 0 for process-created files. For unzipped files, reflects the original file's transfer.
DestHost	char	The destination host of the file. This is the "friendly" name, not the IP domain name. For process records, the name of the script.
DestFilename	char	The filename of the destination file. For process records, empty.
DestFilenameOnly	char	The filename of the destination file, without the pathname.
DestFileID	char	The ID of the destination file if it was sent to MOVEit Transfer, else empty.
NBytes	double	Number of bytes transferred. 0 if there is any error. For records of successful sends, always populated, even for process-created files. In case of upload error: 0. For process records: always 0.
DestDuration	double	The number of seconds the upload took. For process records, the time taken by the script. (Note inconsistency with NBytes.) In case of upload error, this is 0, to be consistent with NBytes.
ErrCode	int	An error code, where 0 indicates success.
Message	char	An error message, or other text describing this action. Usually empty if success.
Hash	char	Cryptographic hash of this record and the previous record's hash. This is used to implement tamper detection.

Task Runs

The table TaskRuns contains one record for each run of a task.

Field Name	Type	Description
ID	bigint	A record ID that is incremented automatically for each record added to the table. Primary key.

Field Name	Type	Description
LogStamp	datetime	Date/time of record
TaskID	int	The ID of the task.
NominalStart	char	The time when the task started or was scheduled to start. TaskID and NominalStart together uniquely identify a record. There is a key that consists of these two fields.
TaskName	char	Name of task.
TimeStarted	char	The date and time the task started.
TimeEnded	char	The date and time the task ended.
StartedBy	char	Who started the task. If a remote logged-in user started the task (via Run Now), this is the username. If a user from localhost started the task, this is `Local`. If the scheduler started the task, this is `Scheduler`.
Success	char	Values: Failure, Success, No xfers. No xfers is logged when no matching files could be found.
FilesSent	int	The number of files successfully sent.
TotalBytesSent	double	The number of bytes successfully sent. This is a double because not all databases support huge integers, and the number may exceed the capacity of a 4-byte integer.
HasBeenRead	int	A flag used by MOVEit Automation Admin to keep track of whether the operator has said "don't show me this task run again". The default value is 0. Value is set to 1 to mean the task should not be shown anymore.
LastErrorType	int	The type of the last error. Values: 0 = no error; 5 = warning; 6 = error; 7 = internal error
LastErrorText	char	The text of the last error, if any.
Hash	char	Cryptographic hash of this record and the previous record's hash. This is used to implement tamper detection.

Task Groups

The table TaskGroups contains the task groups and their members. This table duplicates the information in the configuration file, and is provided only as a convenience for organizations doing custom report generation from the database. The task groups table is updated every time the user edits the task groups with MOVEit Automation Admin.

FieldName	Type	Description
GroupName	char	The name of a task group. There might be several records with the same GroupName, one for each task belonging to this group.
TaskID	int	The ID of a task belonging to this group. A task can belong to several groups, so there might be several records with the same TaskID.

Audittable

The audit table (introduced in MOVEit Automation 4.0) includes one record for each configuration change made by an administrator:

Field Name	Type	Description
ID	bigint	A record ID that is incremented automatically for each record added to the table. Primary key.
LogTime	char	The date/time stamp of when the change was made.
Node	int	The failover node number, or 0 if failover is not being used.
Action	char	The action being performed, such as `cfgsec_update`.
TargetType	char	The type of entity being changed, such as `task`.
TargetID	int	The ID of the entity being changed, such as 239634085.
TargetName	char	The name of the entity being changed, such as `Detroit Monthly Summary`.
CentralVersion	char	The version of MOVEit Automation that was in use, such as `3.5.6.0`.
ClientIP	char	The IP address of the client application that performed the action. This address might be different from the IP address when using the Web Admin Client. Example: `192.168.1.45`
AgentBrand	char	The name of the client program, such as `MOVEit Automation Admin`.
AgentVersion	char	The version of the client program, such as `3.5.6.1`.
Username	char	The username of the user who performed the action, such as `lukey`.
IPAddress	char	The IP address of the user who performed the action, such as `129.168.1.45`.
Error	int	An error code; 0 if no error.

Field Name	Type	Description
ErrorText	char	An error message, if Error was not 0.
Message	char	Optional details of the change.
Hash	char	Cryptographic hash of this record and the previous record's hash. This is used to implement tamper detection.

MySQL

This topic describes MySQL-specific details of how MOVEit Automation interfaces to its database. When MySQL has been selected as the database engine (and this is the default), MOVEit Automation installs, upgrades, manages, and updates its database automatically, so very few sites will need the information contained in this topic.

In MySQL, the tables can be created with SQL statements like:

```
CREATE TABLE `stats` (
  `ID` bigint(20) NOT NULL auto_increment,
  `LogStamp` varchar(24) default NULL,
  `TaskID` int(11) NOT NULL default '0',
  `Node` smallint(6) NOT NULL default '0',
  `NominalStart` varchar(24) NOT NULL default '',
  `Action` varchar(12) default NULL,
  `SourceHost` varchar(100) default NULL,
  `SourceFilename` varchar(255) default NULL,
  `SourceFilenameOnly` varchar(255) default NULL,
  `SourceFileID` varchar(24) NOT NULL default '',
  `SourceStamp` varchar(24) default NULL,
  `SourceNBytes` double default '-1',
  `SourceDuration` double default '-1',
  `SourceMsgID` text,
  `SourceMDN` text,
  `DestHost` varchar(100) default NULL,
  `DestFilename` varchar(255) default NULL,
  `DestFilenameOnly` varchar(255) default NULL,
  `DestFileID` varchar(24) NOT NULL default '',
  `NBytes` double default NULL,
  `DestDuration` double default '-1',
  `DestMsgID` text,
  `DestMDN` text,
  `ErrCode` int(11) default NULL,
  `Message` varchar(250) default NULL,
  `Hash` varchar(40) default NULL,
  PRIMARY KEY (`ID`),
  KEY `StatsUniqueRun` (`TaskID`,`NominalStart`),
  KEY `Action` (`Action`),
  KEY `ErrCode` (`ErrCode`),
  KEY `LogStampTaskIDIndex` (`LogStamp`,`TaskID`)
);

CREATE TABLE `taskruns` (
  `ID` bigint(20) NOT NULL auto_increment,
  `LogStamp` varchar(24) default NULL,
  `TaskID` int(11) NOT NULL default '0',
  `Node` smallint(6) NOT NULL default '0',
  `NominalStart` varchar(24) NOT NULL default '',
  `TaskName` varchar(200) default NULL,
  `RecType` varchar(8) default NULL,
  `TimeStarted` varchar(24) default NULL,
  `TimeEnded` varchar(24) default NULL,
  `StartedBy` varchar(32) default NULL,
  `Success` varchar(12) default NULL,
```

```
  `FilesSent` int(11) default NULL,
  `TotalBytesSent` double default NULL,
  `HasBeenRead` int(11) default '0',
  `LastErrorType` int(11) default NULL,
  `LastErrorText` varchar(250) default NULL,
  `Hash` varchar(40) default NULL,
  PRIMARY KEY (`ID`),
  KEY `TaskRunsUniqueRun` (`TaskID`,`NominalStart`),
  KEY `Success` (`Success`),
  KEY `HasBeenRead` (`HasBeenRead`),
  KEY `LogStampTaskIDIndex` (`LogStamp`,`TaskID`)
);

CREATE TABLE `taskgroups` (
  `GroupName` varchar(50) default NULL,
  `TaskID` int(11) NOT NULL default '0'
);

CREATE TABLE `audit` (
  `ID` bigint(20) NOT NULL auto_increment,
  `LogTime` varchar(24) default NULL,
  `Node` smallint(6) default NULL,
  `Action` varchar(24) default NULL,
  `TargetType` varchar(24) default NULL,
  `TargetID` varchar(50) default NULL,
  `TargetName` varchar(200) default NULL,
  `CentralVersion` varchar(12) default NULL,
  `AgentBrand` varchar(32) default NULL,
  `AgentVersion` varchar(12) default NULL,
  `Username` varchar(80) default NULL,
  `IPAddress` varchar(16) default NULL,
  `Error` int(11) default NULL,
  `ErrorText` text,
  `Message` text,
  `Hash` varchar(40) default NULL,
  `ClientIP` varchar(16) default NULL,
  PRIMARY KEY (`ID`),
  KEY `LogTime` (`LogTime`)
);

CREATE TABLE `tmplastruns` (
  `TaskIDOfMax` int(11) default NULL,
  `IDOfMax` bigint(20) default NULL
);
```

You must also grant access to the MySQL database with a statement like:

```
GRANT ALL ON MICStats.* TO MICentral@localhost IDENTIFIED BY 'mypassword123';
```

This example creates a user named MICentral with a password of mypassword123.

Then create an ODBC DSN and specify the username and password you gave above. Be sure to check the "Change BIGINT columns to INT" option.

The DSN associated with this database is provided to MOVEit Automation by configuring the DSN field in the MOVEit Automation Config program.

MSSQL

This topic describes Microsoft SQL Server-specific details of how MOVEit Automation interfaces to its database.

When SQL Server has been selected as the database engine, MOVEit Automation manages its database automatically, so very few sites will need the information contained in this topic. However, unlike with MySQL, a MOVEit Automation installation does not install or update the SQL Server software itself. Therefore, a system administrator must make sure that periodic Microsoft updates are applied.

For the supported versions of Microsoft SQL Server, see MOVEit Automation Service Requirements.

To configure the MOVEit Automation connection to MS SQL Server, use the ***MOVEit Automation Config Utility*** (on page 357).

MOVEit Automation creates the database using T-SQL statements like this:

```
USE [master]
GO
IF EXISTS (SELECT name FROM sys.databases WHERE name = N'micstats') DROP DATABASE
[micstats];
GO
CREATE DATABASE [micstats]
GO
USE [micstats]
GO
CREATE TABLE
  [dbo].[audit]
(
  ID [bigint] NOT NULL IDENTITY(1,2) NOT FOR REPLICATION,
  LogTime [varchar](24) NULL,
  Node [smallint] NULL,
  Action [varchar](24) NULL,
  TargetType [varchar](24) NULL,
  TargetID [varchar](50) NULL,
  TargetName [varchar](200) NULL,
  CentralVersion [varchar](12) NULL,
  AgentBrand [varchar](32) NULL,
  AgentVersion [varchar](12) NULL,
  Username [varchar](80) NULL,
  IPAddress [varchar](16) NULL,
  Error [int] NULL,
  ErrorText text,
  Message text,
  Hash [varchar](40) NULL,
  ClientIP [varchar](16) NULL,
);
GO
ALTER TABLE
  [dbo].[audit]
ADD
  CONSTRAINT [DF_audit_LogTime] DEFAULT NULL FOR [LogTime],
  CONSTRAINT [DF_audit_Node] DEFAULT NULL FOR [Node],
  CONSTRAINT [DF_audit_Action] DEFAULT NULL FOR [Action],
  CONSTRAINT [DF_audit_TargetType] DEFAULT NULL FOR [TargetType],
  CONSTRAINT [DF_audit_TargetID] DEFAULT NULL FOR [TargetID],
```

```
  CONSTRAINT [DF_audit_TargetName] DEFAULT NULL FOR [TargetName],
  CONSTRAINT [DF_audit_CentralVersion] DEFAULT NULL FOR [CentralVersion],
  CONSTRAINT [DF_audit_AgentBrand] DEFAULT NULL FOR [AgentBrand],
  CONSTRAINT [DF_audit_AgentVersion] DEFAULT NULL FOR [AgentVersion],
  CONSTRAINT [DF_audit_Username] DEFAULT NULL FOR [Username],
  CONSTRAINT [DF_audit_IPAddress] DEFAULT NULL FOR [IPAddress],
  CONSTRAINT [DF_audit_Error] DEFAULT NULL FOR [Error],
  CONSTRAINT [DF_audit_Hash] DEFAULT NULL FOR [Hash],
  CONSTRAINT [PK_audit] PRIMARY KEY ([ID])
GO
CREATE NONCLUSTERED INDEX
        [IX_audit_LogTime]
ON
        [dbo].[audit]
(
        [LogTime] ASC
)
GO
CREATE TABLE
  [dbo].[stats]
(
  ID [bigint] NOT NULL IDENTITY (1,2) NOT FOR REPLICATION,
  LogStamp [varchar](24) NULL,
  TaskID [int] NOT NULL,
  Node [smallint] NOT NULL,
  NominalStart [varchar](24) NOT NULL,
  Action [varchar](12) NULL,
  SourceHost [varchar](100) NULL,
  SourceFilename [varchar](255) NULL,
  SourceFilenameOnly [varchar](255) NULL,
  SourceFileID [varchar](24) NOT NULL,
  SourceStamp [varchar](24) NULL,
  SourceNBytes float NULL,
  SourceDuration float NULL,
  SourceMsgID text,
  SourceMDN text,
  DestHost [varchar](100) NULL,
  DestFilename [varchar](255) NULL,
  DestFilenameOnly [varchar](255) NULL,
  DestFileID [varchar](24) NOT NULL,
  NBytes float NULL,
  DestDuration float NULL,
  DestMsgID text,
  DestMDN text,
  ErrCode [int] NULL,
  Message [varchar](250) NULL,
  Hash [varchar](40) NULL
);
GO
ALTER TABLE
  [dbo].[stats]
ADD
  CONSTRAINT [DF_stats_LogStamp] DEFAULT NULL FOR [LogStamp],
  CONSTRAINT [DF_stats_TaskID] DEFAULT 0 FOR [TaskID],
  CONSTRAINT [DF_stats_Node] DEFAULT 0 FOR [Node],
  CONSTRAINT [DF_stats_NominalStart] DEFAULT '' FOR [NominalStart],
```

```
  CONSTRAINT [DF_stats_Action] DEFAULT NULL FOR [Action],
  CONSTRAINT [DF_stats_SourceHost] DEFAULT NULL FOR [SourceHost],
  CONSTRAINT [DF_stats_SourceFilename] DEFAULT NULL FOR [SourceFilename],
  CONSTRAINT [DF_stats_SourceFilenameOnly] DEFAULT NULL FOR
[SourceFilenameOnly],
  CONSTRAINT [DF_stats_SourceFileID] DEFAULT '' FOR [SourceFileID],
  CONSTRAINT [DF_stats_SourceStamp] DEFAULT NULL FOR [SourceStamp],
  CONSTRAINT [DF_stats_SourceNBytes] DEFAULT '-1' FOR [SourceNBytes],
  CONSTRAINT [DF_stats_SourceDuration] DEFAULT '-1' FOR [SourceDuration],
  CONSTRAINT [DF_stats_DestHost] DEFAULT NULL FOR [DestHost],
  CONSTRAINT [DF_stats_DestFilename] DEFAULT NULL FOR [DestFilename],
  CONSTRAINT [DF_stats_DestFilenameOnly] DEFAULT NULL FOR [DestFilenameOnly],
  CONSTRAINT [DF_stats_DestFileID] DEFAULT '' FOR [DestFileID],
  CONSTRAINT [DF_stats_NBytes] DEFAULT NULL FOR [NBytes],
  CONSTRAINT [DF_stats_DestDuration] DEFAULT '-1' FOR [DestDuration],
  CONSTRAINT [DF_stats_ErrCode] DEFAULT NULL FOR [ErrCode],
  CONSTRAINT [DF_stats_Message] DEFAULT NULL FOR [Message],
  CONSTRAINT [DF_stats_Hash] DEFAULT NULL FOR [Hash],
  CONSTRAINT [PK_stats] PRIMARY KEY ([ID])
GO
CREATE NONCLUSTERED INDEX
        [IX_stats_StatsUniqueRun]
ON
        [dbo].[stats]
(
        TaskID ASC,NominalStart ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_stats_Action]
ON
        [dbo].[stats]
(
        [Action] ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_stats_ErrCode]
ON
        [dbo].[stats]
(
        [ErrCode] ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_stats_LogStampTaskIDIndex]
ON
        [dbo].[stats]
(
        [LogStamp] ASC, [TaskID] ASC
)
GO
CREATE TABLE
  [dbo].[taskgroups]
(
  GroupName [varchar](50) NULL,
```

```
  TaskID [int] NOT NULL
);
GO
ALTER TABLE
  [dbo].[taskgroups]
ADD
  CONSTRAINT [DF_taskgroups_GroupName] DEFAULT NULL FOR [GroupName],
  CONSTRAINT [DF_taskgroups_TaskID] DEFAULT 0 FOR [TaskID]
GO
CREATE TABLE
  [dbo].[taskruns]
(
  ID [bigint] NOT NULL IDENTITY (1,2) NOT FOR REPLICATION,
  LogStamp [varchar](24) NULL,
  TaskID [int] NOT NULL,
  Node [smallint] NOT NULL,
  NominalStart [varchar](24) NOT NULL,
  TaskName [varchar](200) NULL,
  RecType [varchar](8) NULL,
  TimeStarted [varchar](24) NULL,
  TimeEnded [varchar](24) NULL,
  StartedBy [varchar](32) NULL,
  Success [varchar](12) NULL,
  FilesSent [int] NULL,
  TotalBytesSent float NULL,
  HasBeenRead [int] NULL,
  LastErrorType [int] NULL,
  LastErrorText [varchar](250) NULL,
  Hash [varchar](40) NULL,
);
GO
ALTER TABLE
  [dbo].[taskruns]
ADD
  CONSTRAINT [DF_taskruns_LogStamp] DEFAULT NULL FOR [LogStamp],
  CONSTRAINT [DF_taskruns_TaskID] DEFAULT 0 FOR [TaskID],
  CONSTRAINT [DF_taskruns_Node] DEFAULT 0 FOR [Node],
  CONSTRAINT [DF_taskruns_NominalStart] DEFAULT '' FOR [NominalStart],
  CONSTRAINT [DF_taskruns_TaskName] DEFAULT NULL FOR [TaskName],
  CONSTRAINT [DF_taskruns_RecType] DEFAULT NULL FOR [RecType],
  CONSTRAINT [DF_taskruns_TimeStarted] DEFAULT NULL FOR [TimeStarted],
  CONSTRAINT [DF_taskruns_TimeEnded] DEFAULT NULL FOR [TimeEnded],
  CONSTRAINT [DF_taskruns_StartedBy] DEFAULT NULL FOR [StartedBy],
  CONSTRAINT [DF_taskruns_Success] DEFAULT NULL FOR [Success],
  CONSTRAINT [DF_taskruns_FilesSent] DEFAULT NULL FOR [FilesSent],
  CONSTRAINT [DF_taskruns_TotalBytesSent] DEFAULT NULL FOR [TotalBytesSent],
  CONSTRAINT [DF_taskruns_HasBeenRead] DEFAULT 0 FOR [HasBeenRead],
  CONSTRAINT [DF_taskruns_LastErrorType] DEFAULT NULL FOR [LastErrorType],
  CONSTRAINT [DF_taskruns_LastErrorText] DEFAULT NULL FOR [LastErrorText],
  CONSTRAINT [DF_taskruns_Hash] DEFAULT NULL FOR [Hash],
  CONSTRAINT [PK_taskruns] PRIMARY KEY ([ID])
GO
CREATE NONCLUSTERED INDEX
        [IX_taskruns_TaskRunsUniqueRun]
ON
        [dbo].[taskruns]
```

```
(
        [TaskID] ASC, [NominalStart] ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_taskruns_Success]
ON
        [dbo].[taskruns]
(
        [Success] ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_taskruns_HasBeenRead]
ON
        [dbo].[taskruns]
(
        [HasBeenRead] ASC
)
GO
CREATE NONCLUSTERED INDEX
        [IX_taskruns_LogStampTaskIDIndex]
ON
        [dbo].[taskruns]
(
        [LogStamp] ASC, [TaskID] ASC
)
GO
CREATE TABLE tmplastruns
(
  TaskIDOfMax [int] NULL,
  IDOfMax [bigint] NULL
);
GO
ALTER TABLE
  [dbo].[tmplastruns]
ADD
  CONSTRAINT [DF_tmplastruns_TaskIDOfMax] DEFAULT NULL FOR [TaskIDOfMax],
  CONSTRAINT [DF_tmplastruns_IDOfMax] DEFAULT NULL FOR [IDOfMax]
GO
```

Converter

The Convert to MS SQL Server utility creates an empty MOVEit Automation database on a Microsoft SQL Server and configures MOVEit Automation to use that database.This utility is run only one time to create the initial database. It provides an option to copy an existing MySQL database to the newly created SQL Server database. After the copying of the existing MySQL database is complete, the registry settings are automatically changed to use the MS SQL database. If the existing database is not to be copied, the ***Configure MOVEit Automation*** (on page 357) program can be used to alter the settings.

Convert to MS SQL Server is not included in the MOVEit Automation installation. You can download it from the ***MOVEit support site https://www.ipswitch.com/support/***. For information about prerequisites and where to find the download, see ***How do I get MOVEit Automation to work with Microsoft SQL Server?*** (***https://community.ipswitch.com/s/article/ka036000000kNN8AAM/How-do-I-get-MOVEit-Central-to-work-with-Microsoft-SQL-Server-1307565983335***)

Note: DO NOT run this utility on a MOVEit Automation version that older than version 6.3. Data loss could occur. To convert a version of MOVEit Automation older than 6.3, first upgrade to version 6.3 or later, and then run the conversion.

Ø *To use the Convert to MS SQL Server Utility*

1 Select Programs > MOVEit Automation > Convert to MS SQL Server. The utility runs as a wizard.

The PREPARING TO INSTALL window appears. The utility checks to see whether the Microsoft SQL Native Client database driver is already installed. If it is not installed, the program installs it automatically.

2 On the IDENTIFY DATABASE SERVER dialog box, provide the credentials that are used to create the database.

These credentials must already exist, and are typically not the credentials used by MOVEit Automation to access the database after it is running.

Fields:

- SQL Server host. Hostname or IP address of the database server.
- Instance, if any. Name of the instance. Typically this field is empty, meaning the default instance. In some configurations of Microsoft SQL Server Express, the value must be SQL Express.
- Credentials for initial creation of database. The credentials for the option you select must already exist.
  - Use Windows Authentication. Uses credentials of the Windows user that is running the utility. The computer on which the SQL Server is running must have an identical Windows user, and the SQL Server must be configured to recognize that user.
  - Use SQL Server Authentication. Sets the user to the specified SQL Server login and password. For example, the login sa. The values you specify must already exist.

3 Click Next.

If the MOVEit Automation service is running, the conversion utility will offer to stop it. If you do not stop the service, the new database will not go into use until you restart the service.

4 On the TEST DATABASE CONNECTION dialog box, the utility connects to SQL Server to test the credentials.

The utility also checks to see whether a **`micstats`** database already exists on this server. If it exists, you are prompted to confirm whether it is OK to delete the database. Click OK.

5 When the TEST DATABASE CONNECTION dialog box displays the message **Connected to database successfully**, click **Next**.

Note: If you cannot connect, click **Back** and change your connection settings.

6 On the CREATE DATABASE dialog box, the message **Empty database created successfully** appears. Provide the following information:

Specify how MOVEit Automation should authenticate to the database:

§ **Windows Authentication**. Sets the credentials to the WIndows user that runs the MOVEit Automation service. The computer on which the SQL Server is running must have an identical user. The conversion program creates the SQL Server login, and the corresponding user within the database.

§ **SQL Server Authentication**. Sets the user to the specified SQL login and password when authenticating to the database. The conversion utility creates the specified SQL login and database user on the SQL Server. The password is stored encrypted in the registry.

7 Click **Next**.

Note: If you are using SQL Server authentication and the following error message appears:

The user is not associated with a trusted SQL Server connection

the most likely cause of this error is that the database is not configured to accept SQL Server logins. Do the following:

a) Open SQL Server Management Studio.

b) Right-click on the server name and choose Properties.

c) Choose Security, and then choose SQL Server and Windows Authentication mode.

You must stop and start the MSSQLSERVER service for the changes to take effect.

8 After the SQL login is created and MOVEit Automation is set to use it, the SQL DATABASE CREATION COMPLETED dialog box appears.

Perform full data integrity verification checkbox.

- **Selected**. After the copy is done, the program scans the two databases and compares all records. This process adds approximately 30% to the total conversion time.
- **Not selected**. After the copy is done, the program checks only that the correct number of records exist in the destination database.

9 Click **Next** to start copying the existing MySQL database into the newly-created MS SQL Server database. Or, click **Cancel** to leave the new database unpopulated

The COPYING DATABASE page shows a status bar. If the database is very large, the copy process might take a long time, such as several hours for multi-gigabyte databases.

10 When the **Database copied successfully** message appears, click **Finish**.

11 If you previously stopped the MOVEit Automation service, you are prompted to start it.

Log File

The conversion program places a log file in the user's temporary directory, as specified by the environment variable TMP. The name of this directory is similar to **`C:\Users\username\Local Settings\Temp`**. The filename is **`MICMyToMS.log`**. If you need to contact technical support, have this log file available.

Tamper Detection

MOVEit Automation can detect attempts by an intruder to alter the database tables containing audit information and activity history. An intruder might change these records to erase evidence of unauthorized use of the system, or to falsify file transfer histories.

MOVEit Automation implements tamper detection by populating a field named Hash on each record of its three major database tables. This Hash field contains the cryptographic hash of the current record and the previous record's Hash value, and is therefore part of a "hash chain". MOVEit Automation uses its built-in FIPS 140-2 validated SHA1-HMAC keyed hash algorithm. The key to each hash chain is derived from a tamper detection key that is entered during installation of the product, and stored (in a cryptographically altered form) in the registry.

Current tamper detection information is stored, in encrypted form, in a file named `michash.xml`, in the same directory as the MOVEit Automation configuration and state files.

Detecting tampering

MOVEit Automation contains the built-in task Tamper Detect that checks for tampering. This task runs the built-in Tamper Detect script. By default, it runs nightly. It can also be run upon demand.

Recovering from problems

In the event of a system crash or certain other problems, the `michash.xml` file might become corrupted or out-of-date. MOVEit Automation continues to run, but the Tamper Detect task begins sending alerts of possible tampering. Normally, tamper detection resets itself after sending an email alert, but you can reset tamper detection by using the MOVEit Automation Admin Reset Tamper Detection command. Subsequent to using this command, MOVEit Automation can detect future tampering, but ignores any tampering that has already occurred.

Trimming

Over time, the statistics database can accumulate many records, slowing performance and using megabytes of disk space. To prevent this, MOVEit Automation run a built-in script called Trim Statistics DB periodically to delete old records from the database, optionally saving them to a file or another database before deletion.

For more information about the built-in Trim Statistics DB script, see the ***Scripts - Built-In - Trim Statistics DB documentation*** (see "***Trim Statistics DB***" on page 133).

Troubleshooting

Normally, the statistics database operates silently, behind the scenes of MOVEit Automation, and requires no active maintenance on the part of the administrator to operate. Rarely, however, one or more database tables can become corrupted, which might prevent MOVEit Automation from successfully logging task run information. These corruptions are often caused by unexpected reboots, such as during a power failure. They can also occur when backup programs make copies of database table files while the database server is running. When a database table is corrupted, it can no longer be accessed by the database server until it has been repaired.

If you think you have had, or may be having a database corruption problem, first check the log output. MOVEit Automation accesses the database several times during a typical task run, and when serious database problems occur, they are always written to the running log, which is accessible from MOVEit Automation Admin and from the local system. Here is an example of a table corruption error that would be found in the log:

```
Task "My Task": Could not log task end: [TCX][MyODBC]Can't open file: 'stats.MYD'.
(errno: 145)
```

MySQL Database

If database errors have occurred and you are using MySQL as the database engine, there are steps you can take to repair the database.

Automatic Repair

Recent versions of MOVEit Automation have enabled a database option that automatically repairs tables that it finds corrupted. This means that most of these occurrences happen with little notice by end users. Although no action on the part of the administrators is required in these cases, administrators might want to keep informed of any such happenings. Information is logged by the database server when such corruptions occur, and when they are automatically repaired.

This log information is located in the `\mysql\data` directory of your MOVEit Automation server. It is stored in a file named HOSTNAME.err, where HOSTNAME is the name of the server. A typical corruption detection and repair event is logged like this:

```
041122 1:13:58 read_const: Got error 134 when reading table ./micstats/stats
041122 1:14:00 read_const: Got error 134 when reading table ./micstats/stats
041122 1:41:46 Warning: Checking table: './micstats/stats'
041122 1:41:46 Warning: Recovering table: './micstats/stats'
```

Manual Repair

In the very rare case that the automatic table repair functionality fails, you must repair the table manually. It is not necessary to stop the MOVEit Automation service during the manual repair process.

The MySQL service MUST be running for this sequence of commands to succeed.

To manually repair a database table, open a command-prompt on your Central system and log in to the MySQL server using the `root` account created during the Central installation. To log onto the MySQL server using root, `cd` to your `\mysql\bin` directory and issue this command:

```
mysql --user=root --password=YOUR_ROOT_PASSWORD micstats
```

After you are logged in, execute the following CHECK TABLE command against the table you think has been corrupted:

```
CHECK TABLE stats;
```

This command typically generates several lines of information. The last line indicates the status of the table. If the CHECK response indicates the table needs to be repaired, issue the following repair command:

```
REPAIR TABLE stats;
```

This might take several minutes, depending on the size of the table, and generate several lines of output. If the repair was successful, the last line of output contain a status message of `OK`.

If the manual repair process is unsuccessful after several tries, contact ***MOVEit support https://www.ipswitch.com/support/*** for assistance.

Microsoft SQL Server

SQL Server generally does not require manual repair of database tables. If you encounter database problems with SQL Server as your database, contact your database administrator, or ***MOVEit support https://www.ipswitch.com/support/***.

Reference

Supported Host Types

To add a host: In the top menu bar, click HOSTS. On the right side of the screen, click Add Host and select a host type.

Host type	Description
Local filesystem	The local computer on which MOVEit Automation is running. This host appears automatically in the list. You do not need to explicitly add it. ***Field descriptions - General Properties*** (on page 305). ***Field descriptions - Additional Properties*** (on page 306)
UNC Share	Any remote Windows shares that MOVEit Automation is configured to access. ***Field descriptions - General Properties*** (on page 309) ***Field descriptions - Additional Settings*** (on page 310)
MOVEit Transfer Server	A MOVEit Transfer server accessible via HTTPS, usually over TCP port 443. ***Field descriptions - General Properties*** (on page 313). ***Field descriptions - Additional Settings*** (on page 314)
FTP/FTPS Server	A plain FTP server or FTP over SSL server, usually accessible over TCP port 21. The most common TCP port used for FTP over SSL in implicit mode is 990. ***Field descriptions - General Properties*** (on page 316) ***Field descriptions - Additional Properties*** (on page 319)
SSH/SFTP Server	An FTP over SSH server, usually accessible over TCP port 22. ***Field descriptions - General Properties*** (on page 325). ***Field descriptions - Additional Properties*** (on page 326)
POP3 Server	An inbound email server usually accessible over TCP port 110. ***Field descriptions - General Properties*** (on page 331). ***Field descriptions - Additional Properties*** (on page 332)

Host type	Description
SMTP Server	An outbound email server usually accessible over TCP port 25. ***Field descriptions - General Properties*** (on page 333) ***Field descriptions - Additional Properties*** (on page 334)
AS1 (SMTP/POP)	An AS1 trading partner relationship. An AS1 host defines the parameters for transferring files to and from a partner via the AS1 protocol. AS1 uses email (SMTP and POP3) transports, often with SSL transport security. ***Field descriptions - General Properties*** (on page 335). ***Field descriptions - Additional Properties*** (on page 336)
AS2 (HTTP/S)	An AS2 trading partner relationship. AS2 uses mostly web (HTTP) transport, often with SSL transport security. ***Field descriptions - General Properties*** (on page 338). ***Field descriptions - Additional Properties*** (on page 340)
AS3 (FTP/S)	An AS3 trading partner relationship. AS3 uses FTP transport, often with SSL transport security. ***Field descriptions - General Properties*** (on page 341). ***Field descriptions - Additional Properties*** (on page 343)

Local File System Host Field Descriptions

To access this dialog box:

1. In the top menu bar, click HOSTS. In the left panel, under Type, click Local Filesystem.
2. In the Hosts area, click Local Filesystem. The properties page for the local filesystem opens.

The local filesystem is the system where MOVEit Automation is installed. It is listed automatically on the HOSTS page.

The following table shows the General properties. To edit Uploads, Limits, or Advanced properties, see ***Local File System - Additional Properties*** (on page 306).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

Local File System	Description
General	
ID	System-generated identifier, You cannot edit this field
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.

Local File System	Description
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0. The value can be overridden by settings in individual tasks. NOTE: In the Admin Console, this property is known as Default Xfer Rescan.
Use File Notifications	If selected: Tasks that access this host are run when files arrive, rather than periodically as defined by the scheduler. *Recommended*: Select this checkbox. For more information, see File Notifications.

Local File System - Additional Properties

To access Uploads, Limits, and Advanced properties:

1. In the top menu bar, click HOSTS. In the left panel, under Type, click Local Filesystem.
2. In the Hosts area, click Local Filesystem. The properties page for the local filesystem opens.
3. In the row for the property area you want to edit, click Edit.

Note: For General properties, see ***Local Filesystem Host Field Descriptions*** (on page 305).

Local File System	Description
Uploads	

Local File System	Description
Upload as temp file then rename	Default is No (not selected). If selected: File is uploaded under a temporary filename, and then the just-uploaded file is renamed to a different name. This option can be used to avoid triggering another automation system that depends on the existence of a certain filename, but which cannot detect the difference between open/closed or started/finished files. Temp file upload name: Default: `CTMP[Rnd:4]`. You can provide a specific pattern for naming temporary files. The default yields values like `CTMP9243` and `CTMP2495`. If you use a different value, make sure to avoid duplicate temporary filenames. Renames occur on a file-by-file basis as soon as each file has been uploaded.
Limits	
File count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
File size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Advanced	
Delete State for Host Path entries after	Number of days after which State information for this host is deleted. Notes: When this host is assigned as a task source that uses Collect Only New Files, MOVEit Automation saves file stamp information that is specific to this host and the task source's folder path and file mask. If a source's folder path and/or file mask contains a non-static macro (for example, [DD] or other date/time macros) the actual source folder path/file mask combination can potentially be different every time the task runs. This can cause the saved state information for this host to grow excessively large and without limits.

Local File System	Description
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.
Account for Daylight Savings	This host automatically changes the apparent time of existing files when Daylight Savings Time comes into effect, or reverts to Standard Time. This setting changes the following: § How the Collect Only New Files source option is processed. § The way that synchronization works. In these situations, MOVEit Automation compensates for changes made by the host by adjusting the apparent timestamps of files by one hour. This prevents the unnecessary transfer of files that appear to be new because their apparent times have changed since they were last observed. For example, a file is modified in January (during Standard Time) at 8:00 a.m. When Daylight Savings Time comes into effect, that file's apparent modification time changes to 9:00 a.m. If the Host adjusts timestamps is selected, MOVEit Automation internally adjusts its view of the file's time back to 8:00 a.m. for comparison purposes, and does not consider the time to be new.

.

Windows UNC Share Field Descriptions

To access this dialog box: Select HOSTS > Add Host > UNC Share.

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***UNC Host - Additional Properties*** (on page 310).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

UNC Share Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Resource UNC	The path using a UNC. Example: `\\server\share`
Drive letter	The mapped drive.
Use File Notifications	If selected: Tasks that access this host are run when files arrive, rather than periodically as defined by the scheduler. *Recommended*: Select this checkbox. For more information, see File Notifications.
Use MOVEit Automation Run-As Credentials	For authentication, uses the permissions of the current user that MOVEit Automation is running as, instead of a specified username and password.
Use these Credentials: Username/Password	Authentication credentials. Values that you enter are stored encrypted on a disk. Both username and password can contain macro references. For Share hosts, these fields are not available if Use MOVEit Automation Run-As credentials is selected.

UNC Host - Additional Properties

To access Limits, Timeouts, and Advanced properties: Click HOSTS and click the name of the UNC host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***Windows UNC Share Field Descriptions*** (on page 309).

UNC Host Field	Description
Limits	
File count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
File size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Timeouts	
Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0. The value can be overridden by settings in individual tasks. NOTE: In the Admin Console, this property is known as Default Xfer Rescan.
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Advanced	

UNC Host Field	Description
Use Windows CopyFile API	Default is selected. MOVEit Automation uses this function to transfer files to and from UNC hosts (does not apply to synchronization tasks). Under some conditions, performance is affected, especially when transferring large files. If you experience performance problems when transferring files to/from this host, deselect this option.
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.
Delete State for Host path entries after	Number of days after which State information for this host is deleted. Notes: When this host is assigned as a task source that uses Collect Only New Files, MOVEit Automation saves file stamp information that is specific to this host and the task source's folder path and file mask. If a source's folder path and/or file mask contains a non-static macro (for example, [DD] or other date/time macros) the actual source folder path/file mask combination can potentially be different every time the task runs. This can cause the saved state information for this host to grow excessively large and without limits.

UNC Host Field	Description
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. § Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.
Account for Daylight Savings	This host automatically changes the apparent time of existing files when Daylight Savings Time comes into effect, or reverts to Standard Time. This setting changes the following: § How the Collect Only New Files source option is processed. § The way that synchronization works. In these situations, MOVEit Automation compensates for changes made by the host by adjusting the apparent timestamps of files by one hour. This prevents the unnecessary transfer of files that appear to be new because their apparent times have changed since they were last observed. For example, a file is modified in January (during Standard Time) at 8:00 a.m. When Daylight Savings Time comes into effect, that file's apparent modification time changes to 9:00 a.m. If the Host adjusts timestamps is selected, MOVEit Automation internally adjusts its view of the file's time back to 8:00 a.m. for comparison purposes, and does not consider the time to be new.

MOVEit Transfer Host

To access this dialog box: Select HOSTS > Add Host > MOVEit Transfer.

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***MOVEit Transfer Host - Additional Properties*** (on page 314).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

MOVEit Transfer Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Hostname/IP Address	The full hostname (example: MOVEitstdnet.com) or IP address (example: 192.168.12.14) of the remote server.
Port	Port number of the remote server on which to connect.
Username / Password	Authentication credentials. Values that you enter are stored encrypted on a disk. Both username and password can contain macro references.
IIS Virtual Directory	Defines any alternate locations where MOVEit Transfer is installed. For example, an installation can host another secure web site on the same machine. Use this field to define any alternate locations used. Example: `/mysubdir/`. If you are unsure, set this field to a single slash: `/`.
Sort Files by	Specifies how to sort file listings when they are retrieved form a MOVEit Transfer server. Files are downloaded in the order they are listed, so your selections define which files are download first.
Use File Notifications	If selected: Tasks that access this host are run when files arrive, rather than periodically as defined by the scheduler. *Recommended*: Select this checkbox. For more information, see File Notifications.
Secure Connection with SSL	If selected: Enables secure communication between MOVEit Automation and the remote host. Secure MOVEit Transfer connections are typically initiated on port 443, rather than on insecure port 80.

MOVEit Transfer Host Field	Description
Ignore SSL Certificate errors	Available only if Secure Connection is set. Ignores SSL certificate problems. Problems include: expired certificate, wrong hostname, certificate issued by an untrusted authority. This option is used primarily during testing, when you are using a temporary test certificate. If you do not select this checkbox, and MOVEit Automation detects a questionable certificate, MOVEit Automation will not connect to the host, and an error is logged.
Client Certificate	The SSL client certificate to use when establishing connections. Click Set Cert and choose a certificate.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

MOVEit Transfer Host - Additional Properties

To access Limits, Timeouts, and Advanced properties: Click HOSTS and click the name of the MOVEit Transfer host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***MOVEit Transfer Host Field Descriptions*** (on page 313).

MOVEit Transfer Host Field	Description
Limits	
File Count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.

MOVEit Transfer Host Field	Description
File size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Timeouts	
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds.
Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Advanced	
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.

MOVEit Transfer Host Field	Description
Delete State for Host path entries after	Number of days after which State information for this host is deleted. Notes: When this host is assigned as a task source that uses Collect Only New Files, MOVEit Automation saves file stamp information that is specific to this host and the task source's folder path and file mask. If a source's folder path and/or file mask contains a non-static macro (for example, [DD] or other date/time macros) the actual source folder path/file mask combination can potentially be different every time the task runs. This can cause the saved state information for this host to grow excessively large and without limits.
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. § Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.

FTP Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > FTP/FTPS.

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***FTP Host - Additional Properties.*** (on page 319)

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

FTP Host Field	Description
General	

FTP Host Field	Description
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Hostname/IP Address	The full hostname (example: MOVEitstdnet.com) or IP address (example: 192.168.12.14) of the remote server.
Port	Port number of the remote server on which to connect.
Transfer Type	Options: ASCII or Binary mode. This setting is overridden if the transfer type is specified in any source or destination that is related to this host.
Transfer Mode	§ Active: normal mode of operation for FTP transfers § Passive: typically used for FTP clients that are located behind a firewall. This setting is overridden if the transfer mode is specified in any source or destination that is related to this host.
Username and Password	Authentication credentials. Values that you enter are stored encrypted on a disk. Both username and password can contain macro references.
Account	The FTP account for this server. Can contain macro references. Note: In most cases, leave this field empty. A small number of FTP servers require an account to be entered during log in, after the username and password.
Blind downloads	When downloading from this host, MOVEit Automation does not use any directory listing commands. For FTP servers, this includes change directory (CWD) and list directory (LIST) commands. The FileMask specified in the source is a single filename (not a mask) and the program downloads the file without first checking to see if it exists. This option is rarely used, and is intended primarily to accommodate unusual FTP servers.
Blind uploads	When uploading to this host, MOVEit Automation interprets all destination paths as absolute. MOVEit Automation issues a CWD to the path specified, and does a PUT to save the file in that location. This option is rarely used, and is intended primarily to accommodate unusual FTP servers.

FTP Host Field	Description
Secure Connection	Style of encryption to use when connecting to the FTP server. Your selection depends on the styles that are offered by the particular FTP server. § None: No encryption § Explicit (Ctrl/Data): Both the control connection and any data connections are encrypted. Note: MOVEit Automation does an AUTH TLS, followed by an explicit PROT P). § Explicit (Ctrl Only): Only the control connection is encrypted. § Implicit (Ctrl/Data): Both the control connection and any data connections are encrypted. This type of connection is usually done to port 990. This encryption style is considered obsolete.
Ignore SSL Certificate errors	Available only if Secure Connection is set. Ignores SSL certificate problems. Problems include: expired certificate, wrong hostname, certificate issued by an untrusted authority. This option is used primarily during testing, when you are using a temporary test certificate. If you do not select this checkbox, and MOVEit Automation detects a questionable certificate, MOVEit Automation will not connect to the host, and an error is logged.
Cleartext after signon (CCC)	If selected: After connecting and signing on securely, switches to unencrypted mode for the control connection. This option is less secure because it allows an opponent to see the names of the files that you transfer. However, if the FTP server is behind a firewall that does NATing, this option allows the firewall to rewrite the responses to PASV commands, which allows MOVEit Automation to connect transparently to the correct IP address. Note: This option is rarely needed, because the Ignore PASV IP in passive mode option, which is on by default, accomplishes the same thing.
Client Certificate	The SSL client certificate to use when establishing connections. Click Set Cert and choose a certificate.
Test button	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

FTP Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the FTP or FTPS host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***FTP Host Field Descriptions*** (on page 316).

FTP/FTPS Host Field	Description
Uploads	
Upload as temp file then rename	Default is No (not selected). If selected: File is uploaded under a temporary filename, and then the just-uploaded file is renamed to a different name. This option can be used to avoid triggering another automation system that depends on the existence of a certain filename, but which cannot detect the difference between open/closed or started/finished files. Temp file upload name: Default: `CTMP[Rnd:4]`. You can provide a specific pattern for naming temporary files. The default yields values like `CTMP9243` and `CTMP2495`. If you use a different value, make sure to avoid duplicate temporary filenames. Renames occur on a file-by-file basis as soon as each file has been uploaded.
Limits	
File count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
File size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Timeouts and Transfers	

FTP/FTPS Host Field	Description
Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0. The value can be overridden by settings in individual tasks. NOTE: In the Admin Console, this property is known as Default Xfer Rescan.
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Resume partial transfers (if possible)	Applies to task runs in which the file is a binary file and the Retry Count is greater than zero. During the task run, if an upload or download fails, MOVEit Automation attempts to resume the transfer. The number of resume attempts is set in Retry Count. The transfer starts where the previous transfer failed. This setting can be overridden in a task by settings in Source and/or Destination.
Use XSHA1 command, if available	The XSHA1 command is used if the FTP server supports it. The file that is received or transmitted is compared with the copy of the file that is on the server, by comparing SHA1 hashes of the file.
Reuse SSL session for data connections	Forces data connections to use the same SSL session as the existing control connection. Use to override the setting for a given source or destination task element configuration, so that you comply with partner server settings that require reuse of an SSL session for data connections.
Security	

FTP/FTPS Host Field	Description
Use MD5	Specifies whether MOVEit Automation looks for an MD5 file. MD5 files contain MD5 hashes of source files on the server. Options: Never: (default) MOVEit Automation does not look for an MD5 file. If Present: MOVEit Automation looks for the specified MD5 file. If the file contains a hash for a source file, the file is checked against the hash. § If the file matches the hash, the file continues. § If the file does not match, an error is generated. § If no hash is found for the source file, the file continues. Always (required): MOVEit Automation looks for the specified MD5 file. § If file is not found, an error is generated. § If a file is found, all downloaded files are checked against it. § If any file does not match its hash, or a hash does not exist for the file, an error is generated.
MD5 Filename	Name of the MD5 file to look for. Default is `MD5SUM`.
Advanced	
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.

FTP/FTPS Host Field	Description
Delete State for Host path entries after	Number of days after which State information for this host is deleted. Notes: When this host is assigned as a task source that uses Collect Only New Files, MOVEit Automation saves file stamp information that is specific to this host and the task source's folder path and file mask. If a source's folder path and/or file mask contains a non-static macro (for example, [DD] or other date/time macros) the actual source folder path/file mask combination can potentially be different every time the task runs. This can cause the saved state information for this host to grow excessively large and without limits.
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.

FTP/FTPS Host Field	Description
Host adjusts timestamps for Daylight Savings	This host automatically changes the apparent time of existing files when Daylight Savings Time comes into effect, or reverts to Standard Time. This setting changes the following: § How the Collect Only New Files source option is processed. § The way that synchronization works. In these situations, MOVEit Automation compensates for changes made by the host by adjusting the apparent timestamps of files by one hour. This prevents the unnecessary transfer of files that appear to be new because their apparent times have changed since they were last observed. For example, a file is modified in January (during Standard Time) at 8:00 a.m. When Daylight Savings Time comes into effect, that file's apparent modification time changes to 9:00 a.m. If the Host adjusts timestamps is selected, MOVEit Automation internally adjusts its view of the file's time back to 8:00 a.m. for comparison purposes, and does not consider the time to be new.
Client External IP (active NAT mode)	This field is used only in certain unusual network configurations, such as when there is a router between the MOVEit Automation server and the FTP server that is doing Network Address Translation (NAT). Specify the IP address or Hostname for MOVEit Automation to send to the FTP server when in active mode, instead of the actual IP address/Hostname of the MOVEit Automation server.
Ignore PASV IP in passive mode	If selected: In passive mode, MOVEit Automation uses the IP address associated with the host configuration, and ignores the IP address given by the FTP server.
Directory List Parsing	
Use built-in automatic directory list parsing.	Selected (default): MOVEit Automation automatically recognizes and parses directory listings from the most common types of FTP and SSH servers.

FTP/FTPS Host Field	Description
Use custom directory list parsing parameters	This option is needed only for unusual brands of FTP and SSH servers. Directory listings are assumed to contain one file per line, with optional header and trailer information that is ignored. Skip lines: § Top: Number of lines of header information to be skipped at the beginning of the listing. Typically 0. § Bottom: Number of lines of trailer information to be skipped at the end of the listing. Typically 0. Start columns: § Filename: Column number where the filename starts. (First column is 1.) Filename ends at the first space, or end-of-line. § Date: If non-zero, the column at which the file's date stamp starts. For more information, see ***Column-based Custom Parsing*** (on page 246).
Use directory parsing script	Select the script. You must have created and imported the script for it to appear in the list. For more information, see ***Directory Parsing Script*** (on page 247).--
Additional Commands	
Commands to execute upon signon	Some FTP servers (especially those in front of enterprise servers or legacy equipment) work best if given quote block formatting and file type commands before transfers are performed. Commands that you list here are executed in addition to any commands that are specified in any source/destination related to this host. Macros are supported in this field. Tip: For AS/400 (iSeries) FTP Servers - enter a value of `SITE LISTFMT 1` to request that the AS/400 use a standard (UNIX-like) listing format, which is automatically recognized by MOVEit Automation.
Commands to execute (per file) before transfer	The quote commands to execute. Commands that you list here are executed in addition to any commands that are specified in any source/destination related to this host. Macros are supported in this field.

FTP/FTPS Host Field	Description
Commands to execute (per file) after transfer	The quote commands to execute. Commands that you list here are executed in addition to any commands that are specified in any source/destination related to this host. Macros are supported in this field.

SSH/SFTP Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > SSH/SFTP.

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***SSH/SFTP Host - Additional Properties*** (on page 326).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

SSH/SFTP Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Hostname/IP Address	The full hostname (example: MOVEitstdnet.com) or IP address (example: 192.168.12.14) of the remote server.
Port	Port number of the remote server on which to connect.
Username/Password	Authentication credentials. Values that you enter are stored encrypted on a disk. Both username and password can contain macro references.
Blind downloads	When downloading from this host, MOVEit Automation does not use any directory listing commands. For FTP servers, this includes change directory (CWD) and list directory (LIST) commands. The FileMask specified in the source is a single filename (not a mask) and the program downloads the file without first checking to see if it exists. This option is rarely used, and is intended primarily to accommodate unusual FTP servers.
Disable compression	If selected: Turns off compression for all communications with the SSH server. Recommended only for a few, rare SSH servers that do not support compression.

SSH/SFTP Host Field	Description
Client Key	Optional SSH client key that is associated with this user. SSH servers require either a username and password, or a username and client key. To use client key authentication, click Set Key and make a selection. The Browse SSH client keys dialog box lists SSH keys that you have created or imported
Host Key	Click Retrieve key from to obtain the host key from the Hostname/IP address and Port you specified. § If a key is specified, and the public key that is presented by the SSH host does not match the approved key, MOVEit Automation will not connect to the host because it suspects a security problem.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

SSH/SFTP Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the SSH/SFTP host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***SSH/SFTP Host Field Descriptions*** (on page 325).

SSH/SFTP Host Field	Description
Uploads	
Upload as temp file then rename	Default is No (not selected). If selected: File is uploaded under a temporary filename, and then the just-uploaded file is renamed to a different name. This option can be used to avoid triggering another automation system that depends on the existence of a certain filename, but which cannot detect the difference between open/closed or started/finished files. Temp file upload name: Default: CTMP[Rnd:4]. You can provide a specific pattern for naming temporary files. The default yields values like CTMP9243 and CTMP2495. If you use a different value, make sure to avoid duplicate temporary filenames. Renames occur on a file-by-file basis as soon as each file has been uploaded.

SSH/SFTP Host Field	Description
Set file permissions after upload	This setting affects only SSH hosts that are based on a UNIX-like file system. Default is not selected. If selected: You can set UNIX-style file attributes on the file after it uploads successfully. Enter a 3-digit octal numeric representation, or select checkboxes to set permissions.
Limits	
File count download limit	Maximum number of files that are downloaded from a source in a single task run against this host. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
File size download limit	Maximum number of bytes that are downloaded from a source in a single task run against this host. Value of 0 means no limit. Downloading stops after the first file that causes the number of downloaded bytes to exceed the limit. If the limit is exceeded, and the Advanced task setting ***Automatically re-run task if source download limits are encountered*** (on page 92) is set, the task is automatically rerun.
Timeouts	
Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0. The value can be overridden by settings in individual tasks. NOTE: In the Admin Console, this property is known as Default Xfer Rescan.
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Connect Timeout	Number of seconds to wait when attempting to connect to the host.

SSH/SFTP Host Field	Description
Resume partial transfers (if possible)	Applies to task runs in which the file is a binary file and the Retry Count is greater than zero. During the task run, if an upload or download fails, MOVEit Automation attempts to resume the transfer. The number of resume attempts is set in Retry Count. The transfer starts where the previous transfer failed. This setting can be overridden in a task by settings in Source and/or Destination.
Transfer Rescan Time	Number of seconds between retries of a transfer (get or put) before MOVEit Automation stops attempting the transfer. Default is 0.
Advanced	
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.
Delete State for Host path entries after	Number of days after which State information for this host is deleted. Notes: When this host is assigned as a task source that uses Collect Only New Files, MOVEit Automation saves file stamp information that is specific to this host and the task source's folder path and file mask. If a source's folder path and/or file mask contains a non-static macro (for example, [DD] or other date/time macros) the actual source folder path/file mask combination can potentially be different every time the task runs. This can cause the saved state information for this host to grow excessively large and without limits.

SSH/SFTP Host Field	Description
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. § Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.
Account for Daylight Savings	This host automatically changes the apparent time of existing files when Daylight Savings Time comes into effect, or reverts to Standard Time. This setting changes the following: § How the Collect Only New Files source option is processed. § The way that synchronization works. In these situations, MOVEit Automation compensates for changes made by the host by adjusting the apparent timestamps of files by one hour. This prevents the unnecessary transfer of files that appear to be new because their apparent times have changed since they were last observed. For example, a file is modified in January (during Standard Time) at 8:00 a.m. When Daylight Savings Time comes into effect, that file's apparent modification time changes to 9:00 a.m. If the Host adjusts timestamps is selected, MOVEit Automation internally adjusts its view of the file's time back to 8:00 a.m. for comparison purposes, and does not consider the time to be new.
Optimize SSH transfer buffer	This setting is useful if you are experiencing issues related to partial file downloads. Disabling this option might potentially resolve these issues. Default is selected.
Security	
Encryption Type	The encryption algorithm that is allowed when attempting to connect to this server. Default is (in this order) AES192, 3DES, Blowfish, AES128, AES256.

SSH/SFTP Host Field	Description
Use MD5	Specifies whether MOVEit Automation looks for an MD5 file. MD5 files contain MD5 hashes of source files on the server. Options: Never: (default) MOVEit Automation does not look for an MD5 file. If Present: MOVEit Automation looks for the specified MD5 file. If the file contains a hash for a source file, the file is checked against the hash. § If the file matches the hash, the file continues. § If the file does not match, an error is generated. § If no hash is found for the source file, the file continues. Always (required): MOVEit Automation looks for the specified MD5 file. § If file is not found, an error is generated. § If a file is found, all downloaded files are checked against it. If any file does not match its hash, or a hash does not exist for the file, an error is generated.
MD5 Filename	Name of the MD5 file to look for. Default is `MD5SUM`.
Proxy Server	
Proxy Type	Options: SOCKS4, SOCKS5, WebStandard
Hostname/IP Address	Hostname/IP Address
Port	§ SOCKS4 and SOCKS5 default port is 1080 § WebStandard default port is 808
Username, Password	Credentials. SOCKS4 protocol does not support password authentication.
	For more information, see ***Add a Proxy Server.*** (on page 24)
Directory List Parsing	
Use built-in automatic directory list parsing	Selected (default): MOVEit Automation automatically recognizes and parses directory listings from the most common types of FTP and SSH servers.

SSH/SFTP Host Field	Description
Use custom directory list parsing parameters	This option is needed only for unusual brands of FTP and SSH servers. Directory listings are assumed to contain one file per line, with optional header and trailer information that is ignored. Skip lines: § Top: Number of lines of header information to be skipped at the beginning of the listing. Typically 0. § Bottom: Number of lines of trailer information to be skipped at the end of the listing. Typically 0. Start columns: § Filename: Column number where the filename starts. (First column is 1.) Filename ends at the first space, or end-of-line. § Date: If non-zero, the column at which the file's date stamp starts. For more information, see ***Column-based Custom Parsing*** (on page 246).
Use directory parsing script	Select the script. You must have created and imported the script for it to appear in the list. For more information, see ***Directory Parsing Script*** (on page 247).

POP3 (Incoming Email) Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > POP3 (Incoming email).

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***POP3 Host - Additional Properties*** (on page 332).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

POP3 Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Hostname/IP address	The full hostname (example: MOVEitstdnet.com) or IP address (example: 192.168.12.14) of the remote server.
Port	Port number of the remote server on which to connect.

POP3 Host Field	Description
Username/Password	Authentication credentials. Values that you enter are stored encrypted on a disk. Both username and password can contain macro references.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

POP3 Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the POP3 host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***POP3 Host Field Descriptions*** (on page 331).

POP3 Host Field	Description
Timeouts	
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Advanced	
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.

POP3 Host Field	Description
Use default state cache settings	§ Selected (default). Uses the default state cache settings that are specified in SETTINGS > System Settings > State File. For more information, see ***State File Settings*** (on page 348). § Not selected: Specify how long to cache state information for this host. § Notes: By default, MOVEit Automation keeps state file information cached in memory to achieve maximum efficiency. In some environments, this can become very memory intensive. Use the State Cache settings to remove state file information from memory after a task run, or after a specified amount of time.

SMTP (Outgoing Email) Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > SMTP (Outgoing email).

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***SMTP (Outgoing Email) - Additional Properties*** (on page 334).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

SMTP Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Hostname/IP Address	The full hostname (example: MOVEitstdnet.com) or IP address (example: 192.168.12.14) of the remote server.
Port	Port number of the remote server on which to connect.
From Address	Email address from which messages and attachments are sent through this server.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

SMTP (Outgoing Email) Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the SMTP host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***SMTP (Outgoing Email) Host Field Descriptions*** (on page 333).

SMTP Host Field	Description
Timeouts	
Retry Count	Number of extra times that a transfer (get or put) is retried before MOVEit Automation no longer attempts the transfer. Default is 0 - transfer is not retried after a failure. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Connect Timeout	Number of seconds to wait when attempting to connect to the host.
Data Timeout	Number of seconds to wait when sending data to or receiving data from the host.
Advanced	
Alternate Host	Secondary host to which to rollover when the primary host is not available. For example, if after the designated Retry Count, problems are encountered connecting or logging in to the primary host. Rollover *does not* occur if other problems exist, such as: the directory does not exist, or there are insufficient permissions to access a file. For a host to appear as an option in this field, you must have previously created it. The alternate host must be the same type of host as the primary host. This field applies to the individual host. This is different from the failover feature that applies to nodes.

AS1 Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > AS1 (SMTP/POP).

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***AS1 Host - Additional Properties*** (on page 336).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

AS1 Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Show only FIPS approved signing and encryption algorithms	Determines which options are listed in the Signing Algorithm and Encryption Algorithm fields in the My Organization and Partner sections.
My Organization	
Email Address	Identifies your organization within the file transfer process. The address by which files are sent and received.
Signing Certificate	The SSL certificate used by your organization for message signing and decryption. Must be a full public/private certificate. The public portion of the certificate is given to the file transfer partner so that they can encrypt files to your organization and verify messages signed by your organization.
Signature Algorithm	The hash that is used for signing files. Supported algorithms: SHA1, MD5, SHA-224, SHA-256, SHA-384, SHA-512.
Partner Organization	
Email Address	Identifies the partner within the file transfer process. The address to which files are sent.
Certificate	The public portion of the SSL certificate used by the partner. This certificate is used to encrypt files to the partner and verify messages signed by the partner.
Encryption Algorithm	The symmetric encryption algorithm used for encrypting files. The algorithm must be agreed upon by both sides of the file transfer process. Supported algorithms are 3DES, DES, AES, AESCBC192, AESCBC256, and RC2. None disables encryption of files.
Compression Algorithm	The format used to automatically compress files. The format must be agreed upon by both sides of the file transfer process. The only supported format is ZLib. None disables the compression of files.

AS1 Host Field	Description
EDI Data Type	Tag that describes the format of the data. The tag is placed in the outbound message and applies to sending only. It does not affect the actual data bytes that are sent. Options: § application/edi-x12 (default) Used in most cases. § application/octet-stream: More generic, preferred by some recipients.
Transport Settings	
POP3 Hostname/IP Address	The email server from which AS1 messages are retrieved.
Port	Port number of the POP3 server.
Secure Connection Type	SSL connection type to use when connecting to the POP3 server. Options: § None: MOVEit Automation connects insecurely to the server § Explicit: MOVEit Automation connects insecurely to the server and then requests that a secure connection be negotiated before continuing. § Implicit: MOVEit Automation connects securely to the server.
Ignore SSL Certificate errors	If selected: Problems with the POP3 server's SSL certificate are ignored. Examples: lack of trust, a name that does not match the host name.
Username/Password	Credentials for MOVEit Automation to use when authenticating to the POP3 server.
Client Certificate	SSL client certificate to use when establishing a secure connection to the POP3 server.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

AS1 Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the AS1 host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***AS1 Host Field Descriptions*** (on page 335).

AS1 Host Field	Description
Firewall	

AS1 Host Field	Description
Type	Type of firewall with which MOVEit Automation communicates. Available firewall types: Tunnel, SOCKS4, SOCKS5. Default is None. For the firewall you select, provide the following: § **Hostname/IP Address** of the firewall. § **Port** - TCP port of the firewall. § **Username/Password** for MOVEit Automation to authenticate to the firewall.
Decryption	
Use Signing Certificate for Decryption	**Selected** (default): The signing certificate configured in the My Organization section of the host properties is used to decrypt files that are received from the partner. **Not selected:** Click **Set Cert** and select a certificate to be used to decrypt files received from the partner. Choose a certificate that is different from the signing certificate.
SMTP	
Use POP3 Server for SMTP	**Selected** (default). **Not selected:** Specify the following: § Hostname/IP address and Port of the server to use for SMTP. § Authentication Method. § Username/Password to authenticate to the server.
Retry	
Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Miscellaneous	
Pause before rerunning successful task	Number of seconds before the task is rerun. Default is 10 seconds. ASx file transfers operate one file at a time. After a successful transfer, the task is repeated to process any additional files.
MDN Poll Count	The number of times MOVEit Automation polls the POP3 server for an MDN message from the partner, after an EDI data message has been sent to the partner. Default is 10.
MDN Poll Timeout	Number of seconds between MDN polls. Default is 30 seconds.

AS1 Host Field	Description
Delete Messages Older Than	MOVEit Automation deletes messages on the POP3 server that are older than the specified number of days, Default is 7 days.

AS2 Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > AS2 (HTTP/S).

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***AS2 Host - Additional Properties*** (on page 340).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

AS2 Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Show only FIPS approved signing and encryption algorithms	Determines which options are listed in the Signing Algorithm and Encryption Algorithm fields in the My Organization and Partner sections.
My Organization	
Organization Name	Name by which your organization is known, and identified within the file transfer process.
Signing Certificate	The SSL certificate used by your organization for message signing and decryption. Must be a full public/private certificate. The public portion of the certificate is given to the file transfer partner so that they can encrypt files to your organization and verify messages signed by your organization.
Signature Algorithm	The hash that is used for signing files. Supported algorithms: SHA1, MD5, SHA-224, SHA-256, SHA-384, SHA-512.
MOVEit Transfer Host	The MOVEit Transfer host that receives AS2 messages from the partner.
Partner Organization	
Partner Name	Name by which the partner is known, and identified within the file transfer process.
Certificate	The public portion of the SSL certificate used by the partner. This certificate is used to encrypt files to the partner and verify messages signed by the partner.

AS2 Host Field	Description
Encryption Algorithm	The symmetric encryption algorithm used for encrypting files. The algorithm must be agreed upon by both sides of the file transfer process. Supported algorithms are 3DES, DES, AES, AESCBC192, AESCBC256, and RC2. None disables encryption of files.
Compression Algorithm	The format used to automatically compress files. The format must be agreed upon by both sides of the file transfer process. The only supported format is ZLib. None disables the compression of files.
EDI Data Type	Tag that describes the format of the data. The tag is placed in the outbound message and applies to sending only. It does not affect the actual data bytes that are sent. Options: § application/edi-x12 (default) Used in most cases. § application/octet-stream: More generic, preferred by some recipients.
Transport Settings	
Partner URL	The partner's HTTP server URL to which MOVEit Automation posts AS2 messages.
Ignore SSL Certificate Errors	If selected: Problems with the partner HTTP server's SSL certificate are ignored. Examples: lack of trust, a name that does not match the host name.
Use HTTP Authentication	If selected: Use HTTP authentication with the specified username/password when sending files via this AS2 host.
Username/Password	Credentials for MOVEit Automation to use when attempting HTTP authentication.
Client Certificate	SSL client certificate to use when establishing a secure connection to the partner's HTTP server.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

AS2 Host - Additional Properties

To access these additional fields: Select HOSTS and click the name of the AS2 host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***AS2 Host Field Descriptions*** (on page 338).

AS2 Host Field	Description
Firewall	
Type	Type of firewall with which MOVEit Automation communicates. Available firewall types: Tunnel, SOCKS4, SOCKS5. Default is None. For the firewall you select, provide the following: § Hostname/IP Address of the firewall. § Port - TCP port of the firewall. § Username/Password for MOVEit Automation to authenticate to the firewall.
Decryption	
Use Signing Certificate for Decryption	Selected (default): The signing certificate configured in the My Organization section of the host properties is used to decrypt files that are received from the partner. Not selected: Click Set Cert and select a certificate to be used to decrypt files received from the partner. Choose a certificate that is different from the signing certificate.
Proxy Server	
Proxy Type	Type of proxy server with which MOVEit Automation communicates. Available proxy types: Default, Specific. Default is None.
Hostname/IP Address, Port	Hostname/IP address and TCP Port of the proxy server.
Username/Password	Username/password for MOVEit Automation to use to authenticate to the proxy server.
SSL Mode	Whether and how MOVEit Automation uses SSL to communicate with the proxy server. Options: Auto (default), Always, Never, Tunnel.
Email MDN	
AS1 Host to Receive Email MDNs	To request asynchronous email MDNs, specify an existing AS1 host. The host's configuration options determine the parameters that are sent in the MDN request.

AS2 Host Field	Description
SMTP Server to Send Email MDNs	The SMTP server through which email MDNs are sent when requested by the partner.
Email Address to Send Email MDNs From	The email address that is listed as the From address in email MDNs that are sent to the partner.
Retry	
Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Miscellaneous	
MDN Poll Count	The number of times MOVEit Automation polls the POP3 server for an MDN message from the partner, after an EDI data message has been sent to the partner. Default is 10.
MDN Poll Timeout	Number of seconds between MDN polls. Default is 30 seconds.

AS3 Host Field Descriptions

To access this dialog box: Select HOSTS > Add Host > AS3 (FTP/S).

When you create the host, you can set the following General properties. To set additional properties, edit the host. For more information, see ***AS3 Host - Additional Properties*** (on page 343).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

AS3 Host Field	Description
General	
Friendly Name	A name for the host. The name you specify appears in the MOVEit Automation user interface.
Description	Description of the host. Optional. This field is informational only, and does not affect the operation of the host.
Show only FIPS approved signing and encryption algorithms	Determines which options are listed in the Signing Algorithm and Encryption Algorithm fields in the My Organization and Partner sections.

AS3 Host Field	Description
My Organization	
Organization Name	Name by which your organization is known, and identified within the file transfer process.
Signing Certificate	The SSL certificate used by your organization for message signing and decryption. Must be a full public/private certificate. The public portion of the certificate is given to the file transfer partner so that they can encrypt files to your organization and verify messages signed by your organization.
Signature Algorithm	The hash that is used for signing files. Supported algorithms: SHA1, MD5, SHA-224, SHA-256, SHA-384, SHA-512.
Partner Organization	
Partner Name	Name by which the partner is known, and identified within the file transfer process.
Certificate	The public portion of the SSL certificate used by the partner. This certificate is used to encrypt files to the partner and verify messages signed by the partner.
Encryption Algorithm	The symmetric encryption algorithm used for encrypting files. The algorithm must be agreed upon by both sides of the file transfer process. Supported algorithms are 3DES, DES, AES, AESCBC192, AESCBC256, and RC2. None disables encryption of files.
Compression Algorithm	The format used to automatically compress files. The format must be agreed upon by both sides of the file transfer process. The only supported format is ZLib. None disables the compression of files.
EDI Data Type	Tag that describes the format of the data. The tag is placed in the outbound message and applies to sending only. It does not affect the actual data bytes that are sent. Options: § application/edi-x12 (default) Used in most cases. § application/octet-stream: More generic, preferred by some recipients.
Transport Settings	
FTP Hostname/IP Address	Hostname or IP address of FTP server that is used for AS3 messages uploads or downloads.
Port	Port number of the specified FTP server.

AS3 Host Field	Description
Secure Connection	SSL connection type to use when connecting to the FTP server. Options: § None: MOVEit Automation connects insecurely to the server § Explicit: MOVEit Automation connects insecurely to the server and then requests that a secure connection be negotiated before continuing. § Implicit: MOVEit Automation connects securely to the server.
Ignore SSL Certificate errors	If selected: Problems with the partner FTP server's SSL certificate are ignored. Examples: lack of trust, a name that does not match the host name.
Username/Password	Credentials for MOVEit Automation to use when authenticating to the FTP server.
Client Certificate	SSL client certificate to use when establishing a secure connection to the FTP server.
Transfer Mode	FTP Transfer mode to use for uploading and downloading files. Options: § Active: The FTP server initiates a data channel connection to MOVEit Automation for data transfers. § Passive: (default). MOVEit Automation initiates a data channel connection to the FTP server for data transfers.
Test	Click to test your connection. For more information, see ***Tests Performed on Hosts*** (on page 24).

AS3 Host - Additional Properties

To access these additional fields: Click HOSTS and click the name of the AS3 host. You can ***filter the list*** (on page 12).

NOTE: The host's properties page shows the configured settings for each property area. To view all possible settings, click Edit.

For General properties, see ***AS3 Host Field Descriptions*** (on page 341).

AS3 Host Field	Description
Firewall	

AS3 Host Field	Description
Type	Type of firewall with which MOVEit Automation communicates. Available firewall types: Tunnel, SOCKS4, SOCKS5. Default is None. For the firewall you select, provide the following: § **Hostname/IP Address** of the firewall. § **Port** - TCP port of the firewall. § **Username/Password** for MOVEit Automation to authenticate to the firewall.
Decryption	
Use Signing Certificate for Decryption	**Selected** (default): The signing certificate configured in the My Organization section of the host properties is used to decrypt files that are received from the partner. **Not selected**: Click **Set Cert** and select a certificate to be used to decrypt files received from the partner. Choose a certificate that is different from the signing certificate.
Retry	
Retry Count	The number of times that a transfer (get or put) is retried before the MOVEit Automation software no longer attempts the transfer. Values: 0 - transfer is not retried after a failure. Default is 3. The value can be overridden by settings in individual tasks.
Retry Timeout	The number of seconds between retries of a transfer (get or put). Default is 30 seconds. The value can be overridden by settings in individual tasks.
Miscellaneous	
Pause before rerunning successful task	Number of seconds before the task is rerun. Default is 10 seconds. ASx file transfers operate one file at a time. After a successful transfer, the task is repeated to process any additional files.
MDN Poll Count	The number of times MOVEit Automation polls the FTP server for an MDN message from the partner, after an EDI data message has been sent to the partner. Default is 10.
MDN Poll Timeout	Number of seconds between MDN polls. Default is 30 seconds.

System Settings

To access this page: Select SETTINGS > System Settings.

System Setting	Description
Debug Log	Specifies the amount of debug information that is saved, the filename, and maximum log file size. ***Field descriptions*** (on page 345).
Audit Log	Determines how to record issues that are encountered during file transfers. ***Field descriptions*** (on page 346).
Windows Event Log	Controls which Windows Event log is used. ***Field descriptions*** (on page 346).
ASx Logging	Directory for log files for each ASx transmission. ***Field descriptions*** (on page 347).
Tasks	Settings that apply to all tasks, such as maximum number of running tasks, polling intervals, multiple same task runs. ***Field descriptions*** (on page 347).
State File	Controls the caching of state information. ***Field descriptions*** (on page 348).
Tamper Detection	Activates the ability of MOVEit Automation to detect attempts by an intruder to alter database tables that contain audit information and activity history. ***Field Descriptions*** (on page 349).

Debug Log Settings

To access these settings: Select SETTINGS > System Settings. On the Debug Log line, click Edit.

Debug Log Setting	Description
Global Debug Level	Specifies how much debug information is logged to disk. If no task filter is set, appears in the Log. Options: (listed here from low to high amounts of information) All debug, Internal errors, Task/File errors, Task/file warnings, Task completions, File Completions Some Debug, More Debug, All Debug
Log to Disk	Selected (default) MOVEit Automation writes its log file to disk. The scrolling display on the Log tab of MOVEit Automation Admin is not affected by this setting.

Log filename	Default name: `MoveITC.log`. Default location is the same folder where MOVEit Automation is installed. To save to a different name and/or location, provide a complete path including drive letter. For example, `D:\moveit\logs\central.log`
Maximum Log File size	Default is 10 MB. When the log file reaches the maximum size, it is automatically rolled. Old log files are retained for one generation as `.old` files

Audit Log Settings

Determines how to record issues that are encountered during file transfers.

To access these settings: Select **Settings > System Settings**. In the Audit Log row, click **Edit**.

Audit Log Setting	Description
Audit Transfer Retry Attempts	When issues are encountered during a file transfer: § **Yes** creates a separate audit log entry for each individual retry attempt to the database. § **No** creates a single audit log entry after all retry attempts have completed.

Windows Event Log Settings

To access these settings: Select **Settings > System Settings**. In the Windows Event Log row, click **Edit**.

Event Log Setting	Description
Log Name	The Windows Event log that is used **Options**: MOVEit, Application.
Log Level	Amount of information that is written to the local Windows Event Log. **Options** (listed here from low to high amounts of information) Internal Errors, Task/File Errors, Task File Warnings, Task Completions, File Completions.
Audit Log Level	Amount of audit log information is written to the local Windows Event Log. **Options** (listed here from low to high amounts of information) None, Only Errors (default), All.

ASx Log Settings

Directory for log files for each ASx transmission.

To access these settings: Select SETTINGS > System, and on the ASx Logging row, click Edit.

ASx Log Setting	Description
Directory for ASx Logs	Location for log files for each ASx transmission. The directory can be on a UNC path. To write logs to separate directories for each MessageID and Date, include the macros `%MessageID%` and `%Date%` in the directory path. The account that the MOVEit Automation service is running as (default is `micsv`) must have permissions for the directory you specify.

System-Wide Task Settings

To access these settings: Select SETTINGS > System, and on the Tasks row, click Edit.

These settings affect all tasks.

Tasks Setting	Description
Maximum Running Tasks	Maximum number of tasks that MOVEit Automation can run at any given time. Default is 20 tasks. Tasks prevented from running immediately are queued. Queued tasks are started from the queue in FIFO order when the number of currently-running tasks drops below the value you specify. This setting is useful on machines that attempt to launch hundreds of tasks at the same time. Note: This limit affects only the tasks that are run directly by the MOVEit Automation scheduler. Tasks that are started manually by a MOVEit Automation Admin or by the <ICEN> API ignore the running tasks limit, and are run before any queued scheduled tasks.

Allow Multiple Runs of Same Task	Controls whether multiple runs of a task are run at the same time Does not affect tasks that use AS2 hosts. Can be overridden when starting a task manually.
Suppress Queued Tasks on Startup	If checked, tasks that are subject to notification are not run automatically at startup. Also affects running of these tasks when a secondary fails over to being the primary. Selecting this option gives better startup performance, but might result in the delayed retrieval of new files.
Notification Polling Interval	Specifies how often MOVEit Automation software polls any target servers that use the File Notification option for new files.
AS2 Polling Interval	This setting is available only if the AS1/AS2/AS3 model is licensed. Specifies how often MOVEit Automation software polls an AS1\2-linked MOVEit Transfer server for incoming AS2 messages.

State File Settings

Controls the caching of state information.

To access these settings: Select SETTINGS > System Settings. In the State File row, click Edit.

This setting applies to all tasks.

Field	Description
Keep this task's state file information cached:	Options: § Always. § Until end of task run. § For a specified interval.

See also:

§ ***Task Transfer Exceptions*** (on page 98)

Tamper Detection Settings

Activates the ability of MOVEit Automation to detect attempts by an intruder to alter database tables that contain audit information and activity history.

Select SETTINGS > System Settings. In the Tamper Detection row, click Edit.

Field	Description
Tamper Detection	Selected: Enables tamper detection MOVEit Automation can detect attempts by an intruder to alter the database tables that contain audit information and activity history. An intruder would change these records to erase evidence of unauthorized use of the system, or to falsify transfer histories.

Tamper Detection Settings

Select SETTINGS > System Settings. In the Tamper Detection row, click Edit.

Field	Description
Tamper Detection	Selected: Enables tamper detection MOVEit Automation can detect attempts by an intruder to alter the database tables that contain audit information and activity history. An intruder would change these records to erase evidence of unauthorized use of the system, or to falsify transfer histories.

MOVEit Automation Service

Overview

The MOVEit Automation installation program installs these components:

§ MOVEit Automation service

MOVEit Automation is normally installed as a service. This is the program that maintains the configuration files and runs configured tasks.

§ MOVEit Automation Config Utility

Most MOVEit Automation settings, such as hostnames, directories, schedules, and logging, are managed by MOVEit Automation Admin. However, a few settings are managed by a separate program, MOVEit Automation Config. These are ordinarily set one time at installation; there is rarely a need to change them.

The MOVEit Automation service maintains the configuration file and runs configured tasks

To configure and control the transfer and manipulation of tasks, you can use either of the following programs. Each one is installed separately. For more information, see ***Installation Guide https://docs.ipswitch.com/MOVEit/Central91/Manuals/en/index.htm#31751.htm***.

§ MOVEit Automation Admin Console, installed separately.

§ MOVEit Automation Web Admin, a web-based program. It has a subset of the functions that are available in MOVEit Automation Admin. For more information, see see the Install <ICEN> Web Admin section of the ***Installation Guide https://docs.ipswitch.com/MOVEit/Central91/Manuals/en/index.htm#31751.htm***.

Starting and Stopping

The MOVEit Automation service normally starts automatically when the computer boots.

Starting the MOVEit Automation service manually:

If the MOVEit Automation service is not running, start it manually using either of the following methods:

§ Windows command-line console.

§ Windows Service control panel.

For more information, see ***Service - Running As...*** (see "***Running As...***" on page 351)

Stopping the MOVEit Automation service:

§ *Recommended:* Use the MOVEit Automation Admin Shut Down Service command. Using this command avoids killing any running tasks.

§ Alternative ways to stop the service: use Windows command-line console or Windows Service control panel.

Firewall Considerations

The type of "outbound" access that MOVEit Automation requires through a firewall depends on which remote hosts you want to access. For example, if you want to access a remote MOVEit Transfer server, you must allow MOVEit Automation to connect to that server using HTTPS.

It is not normally necessary for firewall rules to be configured to allow "inbound" access to a MOVEit Automation. Two exceptions:

§ If MOVEit Automation Admin needs to be allowed to connect from a remote IP addresses through a firewall. (In this case, configure MOVEit Automation to force SSL encryption when communicating with MOVEit Automation Admin). Ports 3471, 3472, and 3473 are used.

§ If an additional server (for example, Microsoft IIS FTP) has been installed on the same platform as MOVEit Automation.

  Although inbound firewall rules are not required to access MOVEit Automation in this situation, most firewall administrators take a "rules for machine" view rather than "rules for application" view. Therefore, it is a good practice to inform your firewall administrators if you plan on installing any "helper" services on the MOVEit Automation platform.)

For more information, see ***Port Numbers*** (on page 257).

Running As...

MOVEit Automation normally runs as a service. It is usually installed to run under a specific local Windows administrator account.

§ ***Recommended configuration: running as a service*** (on page 352).
§ ***Converting from MOVEit Automation as a local system service*** (on page 352)
§ ***Running MOVEit Automation in the foreground, not as a service*** (on page 353)
§ ***Additional Considerations*** (on page 354).

See also:

Starting and Stopping the MOVEit Automation Service, (on page 350)

Recommended Configuration: Running As a Service Under a Specific Windows Administrator

If you are running MOVEit Automation as a service, and you want to access remote (or mapped) Windows file systems, you must run that service as a specific Windows administrator, not as LocalSystem. New installations of MOVEit Automation are configured this way.

Note: Early installations of MOVEit Automation set up the MOVEit Automation service to run as LocalSystem. For instructions on how to convert from LocalSystem to a specific account, see ***Converting From MOVEit Automation as a LocalSystem Service*** (on page 352).

Authenticating to a Windows domain or Active Directory

If you want to authenticate MOVEit Automation Admin (and MOVEit Automation API) users to a Windows domain (or Active Directory), you might need to run MOVEit Automation under a domain user account that is also an administrator on the local MOVEit Automation machine. If you do not, you might see "RPC Server is unavailable" errors when your domain users attempt to authenticate to MOVEit Automation.

Preferred: Run MOVEit Automation as a domain user with local administrative permissions.

Not recommended: Run MOVEit Automation as a full domain administrator.

Converting From MOVEit Automation as a Local System Service

Windows does not allow services to access network shares if the services are running under the local system account.

If you are running the MOVEit Automation service as Local System, use this procedure to switch to a specific administrator account.

1. On the Windows machine, go to Administrative Tools > Services.
2. Right-click MOVEit Automation, and select Properties.
3. Click the Log On tab.

   The Local System account is selected.
4. Select This account and select a local administrator account for which you know the password.
5. Click OK to change the user under which MOVEit Automation runs.
6. The account you choose must have the *Act as part of the operating system* right. To configure this:
   a) Go to Administrative Tools / Local Security Policy.
   b) Expand the Local Policies tree, and choose User Rights Assignment.
   c) Double-click Act as part of the operating system.
   d) In the dialog box that opens, click Add, and specify the user under which the service will be running.
7. Make sure the account you choose has full file/folder permission rights to the MOVEit Automation "Program Files\MOVEit" folder and cache folder.

Running MOVEit Automation in the Foreground, Not As a Service

Running MOVEit Automation in the foreground as a "normal" application instead of as a service is useful in the following situations:

- You are attempting to replicate and diagnose unusual problems caused by permissions or policies recently applied to the administrative user under which MOVEit Automation runs
- You want to bring MOVEit Automation up with the scheduler disabled so you can examine an imported configuration without running any tasks.

To run MOVEit Automation as a normal foreground application:, select the "Run MOVEit Automation Service in Foreground" option from the Start menu "MOVEit Automation" program group.

Starting MOVEit Automation from the command line

To take advantage of other run options, you must start MOVEit Automation from the command line.

CAUTION: Some of these options can harm your existing MOVEit Automation implementation,. Read the entire description of each option before using it.

The following options are currently available:

- -? - Causes MOVEit Automation to display a very short help dialog and exit.
- -manual - Causes MOVEit Automation to run as a normal foreground application. (Using this option and this option only will achieve the same result as selecting the "Run MOVEit Automation Service in Foreground" option from the Start menu "MOVEit Automation" program group.)
- -config [config_file] - Causes MOVEit Automation to launch using a config file other than "miccfg.xml". When this option is used, "-manual" is also almost always used as well.
- -k - Causes MOVEit Automation to launch with the scheduler disabled. When this option is used, "-manual" is also almost always used as well.
- -remove - Causes MOVEit Automation to uninstall its Windows service entry. (This is much different than uninstalling the entire application.) Using this option could break your MOVEit Automation implementation.
- -install - Causes MOVEit Automation to install its Windows service entry. (This is much different than installing the entire application.) Using this option could break your MOVEit Automation implementation.

Ipswitch technicians use the following batch file (saved in the "/Program Files/MOVEit" folder) to safely launch customer configurations in the foreground for troubleshooting purposes. This batch file disables the scheduler and requires technicians to type in the explicit name of an alternate configuration.

```
@echo off SET CUSTOMERCONFIG=%1 if R%CUSTOMERCONFIG%R==RR GOTO NOCONFIG
"c:\program files\moveit\micentral.exe" -k -manual -config %1 GOTO THEEND
:NOCONFIG echo *** You MUST provide the path of a custom config file! :THEEND
```

Additional Considerations

Permissions and roles required:

- If you want to manage task permission groups from MOVEit Automation Admin, the user under which MOVEit Automation runs must be an administrator.
- The user under which MOVEit Automation runs must have FULL permissions to both of the following:
  - The local folder in which the MOVEit Automation program and configuration files are stored,
  - The MOVEit Automation cache folder.

Note: If the user does not have these permissions, any changes that the user makes to the Central configuration are not permanent. During start-up, MOVEit Automation checks permissions to the folder. If permissions are insufficient, an alert with instructions for resolving the issue is sent to the Email address set in the Config utility - Errors tab, and MOVEit Automation terminates the current session.

Permissions Not Required:

- The user under which MOVEit Automation runs does NOT need permission to all Windows file shares that you would like to access. Specific Windows file share credentials will be configured for each file share.

MOVEit Automation Config Utility

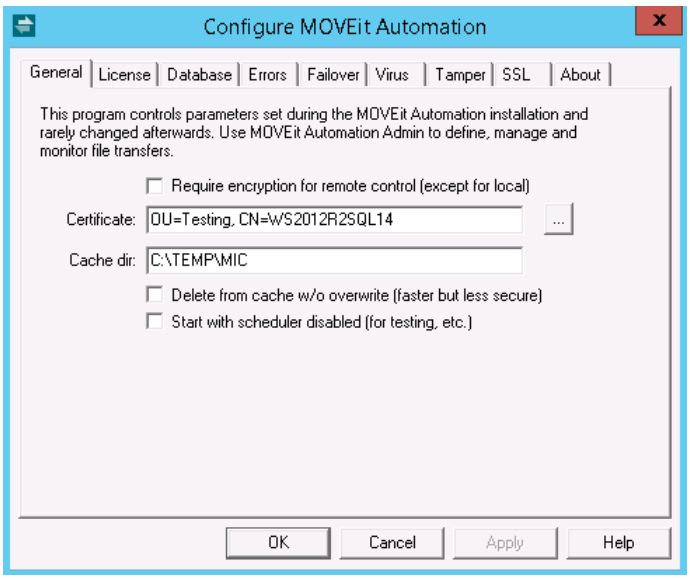
Most MOVEit Automation settings, such as hostnames, directories, schedules, and logging, are managed by MOVEit Automation Admin. However, a few settings are managed by a separate program, MOVEit Automation Config. These are ordinarily set once at installation; there is rarely a need to change them.

To run the configuration program, use the Start menu shortcut MOVEit Automation Config.

The MOVEit Automation Config utility dialog box contains the following tabs:

- ***General*** (on page 355)
- ***License*** (on page 356)
- ***Database*** (on page 357)
- ***Error*** (on page 358)
- ***Failover*** (on page 358) (Enterprise only)
- ***Virus*** (on page 361)
- ***Tamper*** (on page 362)
- ***SSL*** (on page 363)
- ***About*** (on page 366)

General Tab



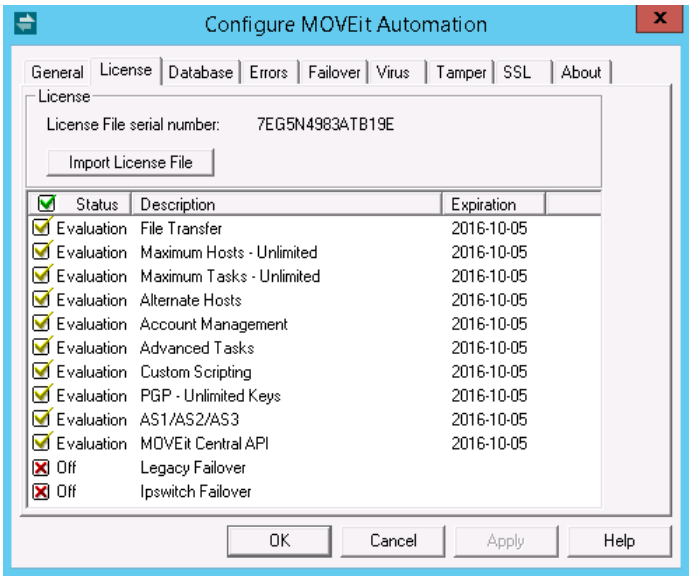
Settings on this tab::

- **Whether connections from MOVEit Automation Admin or API should be encrypted.** Normally, you should enable encryption, for best security. However, you can disable encryption if, for instance, you have not obtained a certificate. Note: encryption is always disabled when MOVEit Automation Admin or API is connecting from the same computer (localhost).
- **Encryption certificate.** This is used only for encrypted links from MOVEit Automation Admin or MOVEit Automation API. The installation program will by default install a "test" (self-signed) certificate for this purpose. If you have an existing SSL certificate on the MOVEit Automation system, you may select it here. Providing a secure connection for MOVEit Automation Admin is not necessary, but it is recommended if MOVEit Automation Admin sessions will be accessing the MOVEit Automation server from outside your private network. You can use the MOVEit Automation SSL Certificate Manager to create a certificate suitable for use here.
- **Cache dir.** This is the directory that will be used to store files while they are being processed by a task; files will automatically be deleted from here (with NIST 800-88-compliant cryptographic overwrite) when related tasks complete. Normally this directory should be on your largest available hard drive.

WARNING: You must manually create any alternate folder specified by this location. MOVEit Automation will not create this folder if it is missing. The user under which the MOVEit Automation service runs must have read/write/delete/subdirectory access to this folder.

- **Delete from cache w/o overwrite.** MOVEit Automation normally deletes and then overwrites its cache files with cryptographic-quality random data. If this option is enabled, file transfers might be faster (especially large, filesystem-to-filesystem transfers).
- **Start with scheduler disabled.** Normally, MOVEit Automation starts running tasks right after starting. Select this option if, for testing or operational purposes, you want to make sure that the MOVEit Automation scheduler is *not active* when MOVEit Automation starts. MOVEit Automation can still start tasks explicitly via MOVEit Automation Admin or MOVEit Automation API. If this option is selected, you can enable the scheduler after startup via MOVEit Automation Admin; however, the next time MOVEit Automation starts, the scheduler will again be disabled until this option is deselected in the configuration program.

License Tab



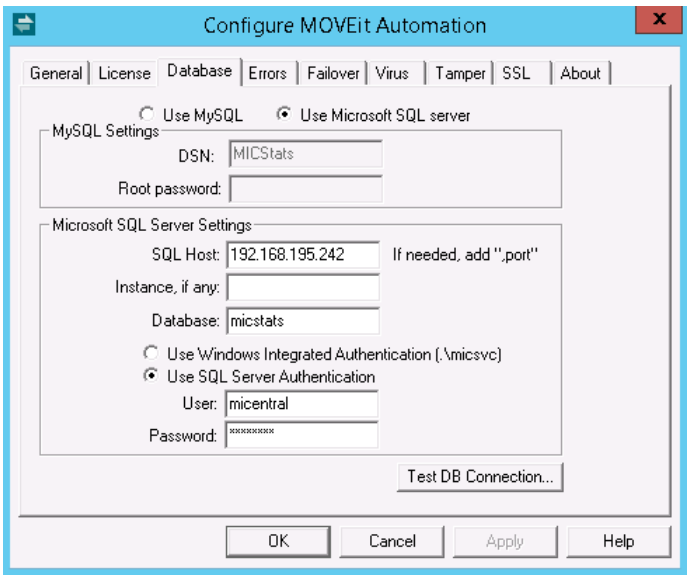
This tab shows the license file serial number and the features enabled by the license. A license file is required to activate the program. The file is provided to you when you evaluate or license the program. In addition to enabling the basic operation of MOVEit Automation, a license file can also activate optional features of the program. Each license file expires on a certain date; some license files have different features that expire on different dates.

The settings on this tab are:

- **License File serial number.** The serial number of the file you supplied during the install process.
- **Import License File.** To switch to a different license file, browse for your license file, and select to change it.

The license file is activated when you click **Apply** or **OK**.

Database Tab



This tab allows you to choose which database engine to use, and to select options specific to that database engine:

If you choose *Use MySQL*, you can change these settings:

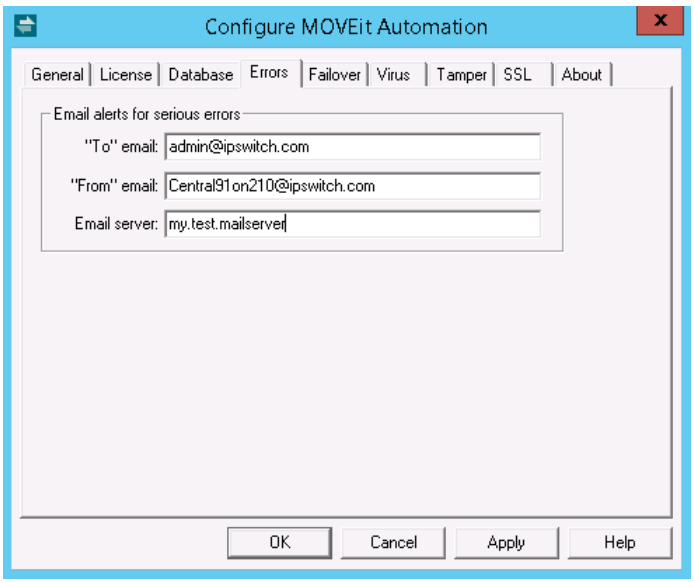
- DSN. The ODBC Data Source Name of the database. There is rarely a need to change this from the default of "micstats".
- MySQL root password. The password of the "root" user in the MySQL database. This is stored encrypted in the registry, and is used only by the install program. The previous value is not displayed in the dialog box when the program starts--not even masked by *'s--for security reasons.

If you choose *Use Microsoft SQL Server*, you can change these settings:

- Host: The hostname of the SQL Server.
- Instance: The name of the SQL Server "instance". This is usually empty, meaning the default instance.
- Database: The name of the database. This is nearly always "micstats".
- Use Windows Integrated Authentication. This causes MOVEit Automation to authenticate to SQL Server using the credentials associated with the MOVEit Automation service. These credentials are shown on the radio button; for instance, "(.\micsvc)" means the local Windows user micsvc (as opposed to a domain user). The SQL Server must have a login with the same name, associated with a Windows username of the same name.
- Use SQL Server Authentication. This causes MOVEit Automation to authenticate to SQL Server using the specified SQL login and password. The password is stored encrypted in the local registry. These credentials must exist on the SQL Server, and there must be a corresponding user in the micstats database on that server. Typically these are created during creation of the database, and usually do not need to be changed.

Test DB Connection. Click this button to test whether you can connect to the database using the specified credentials.

Errors Tab



The Errors tab is used to configure email messages that are sent when a serious error occurs. MOVEit Automation sends these emails primarily when in failover mode. These settings are independent of the Host and Task email settings that are used in normal running of tasks.

The settings on this tab are:

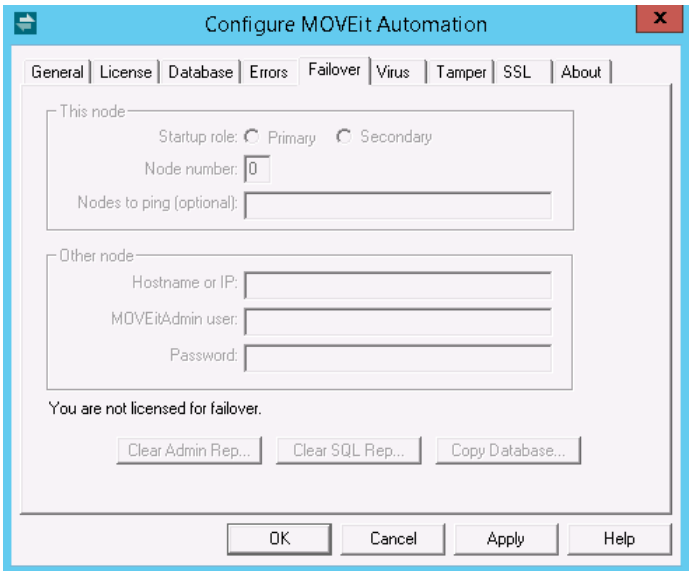
- "To" error email. An optional comma-separated list of email addresses to which a message is sent when a serious error occurs. Some of the situations in which MOVEit Automation sends messages to these addresses include pending and actual failover and tamper detection instances. If this field is empty, no email is sent.
- "From" email. The address that MOVEit Automation places in the "From:" line of emails that it sends as a result of a serious error.
- Email server. The hostname or IP address of the email server that is used for these messages.

Failover Tab (Enterprise Only)

MOVEit Automation has the capability of running in *failover mode*, in which one server automatically stands in for another if the first one fails.

NOTE: This information describes the Legacy Failover system, which has been replaced by the Ipswitch Failover Manager. The Ipswitch Failover Manager is available. It requires a separate installation and managed externally. For more information, go to the Ipswitch support site, and under Secure Information and File Transfer, in the dropdown box, select ***Ipswitch Failover https://www.ipswitch.com/support/documentation***.

Ipswitch will continue to support Legacy Failover installations for MOVEit Automation. The Legacy Failover system is built into MOVEit Automation. The Legacy Failover system is included in the MOVEit Automation documentation. The new Ipswitch Failover system is documented separately; see the above link.



The Failover tab is used to configure the ***Failover*** (on page 373) capability of MOVEit Automation. This tab is grayed out unless you have entered a license key that enables failover. Sites which have not licensed failover can ignore this tab.

The settings on this tab are:

This node

- Startup role. This is the role that this node will assume when MOVEit Automation starts. When installing failover for the first time, you should set one node to be Primary and the other node to be Secondary. Subsequently, MOVEit Automation itself will manage this value on the two nodes when a failover occurs.
- Node number. This is 0, 1, or 2. 0 disables the failover feature even if you have a license for failover. To enable failover, assign the number 1 to one node, and 2 to the other node. The numbers 1 and 2 have no special significance; however, by convention, the value 1 is typically given to the node that is initially assigned the primary role, and the value 2 is given to the node initially assigned the secondary role.
- Nodes to ping. This is an optional comma-separated list of nodes that MOVEit Automation should "ping" before assuming the primary role. If this list is empty, no ping test is done. If the list is not empty, at least one of the nodes must respond to a ping before MOVEit Automation will start running tasks. The purpose of this feature is to prevent two copies of MOVEit Automation from each thinking the other is down because the network between them is down. To take best advantage of this feature, you should enter the hostname of one or more computers that reside on the same network as the other node.
- It is recommended that the value of a "nearby" and trusted router be configured in this field. If you do not have a dedicated network device which fits the bill, it is probably best to leave this value blank.

Other node

- Hostname or IP. The hostname or IP address of the other node.
- MOVEit Admin user. The Windows user on the other node which MOVEit Automation should use to login to the other node. You must create this user on the other node, and make it a member of the "MOVEit Admin" group. It does not need to be a member of the "Users" group. You might want to follow the convention of using the username "micfailover" on both nodes.
- Password. The password of the above user. This password is stored in the registry, using 256-bit AES encryption.

The buttons on this tab are typically used only during a ***resynchronization*** (on page 384) operation after MOVEit Automation has been stopped:

- Clear Admin Rep... This erases any MOVEit Automation Admin commands that are scheduled to be replicated from this node to the other node. It does this by deleting the MICMisc.blg file. (A new blank file will be created automatically when MOVEit Automation is next started.)
- Clear SQL Rep... This erases any SQL statements that are scheduled to be replicated from this node to the other node. It does this by deleting the MICSQL.blg file. (A new blank file will be created automatically when MOVEit Automation is next started.)
- Copy Database... This allows you to copy the MICStats database from the other node to this node. This process overwrites the current node's statistics database with the contents of the other node's database. This operation should ordinarily be performed only on the secondary node.

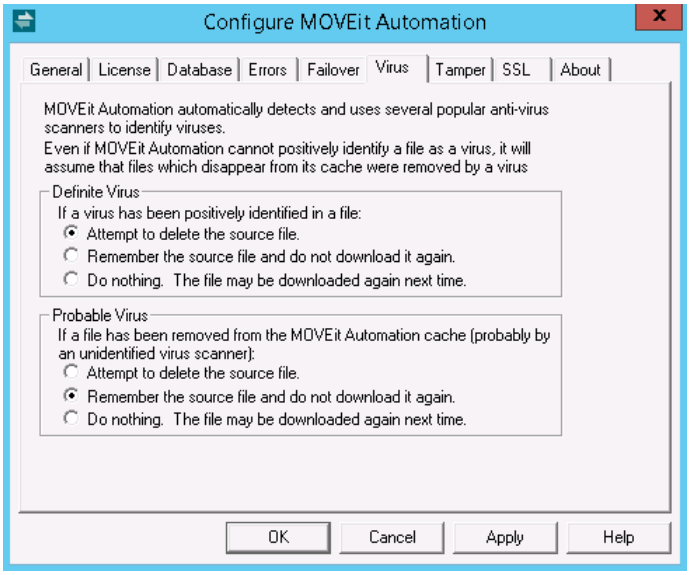
  You need to do a Copy Database on the secondary node when you first install the secondary node. You may also perform this operation subsequently if the database on the current node has gotten out-of-sync with the one on the other node.

When you choose Copy Database, you will be prompted for the remote directory from which to copy the database files, and the local directory to which you should copy them. The program's initial defaults assume that you have installed MySQL on C: and that you are using the default database name. Check the suggested paths and, if necessary, correct them for your installation before choosing OK to start the copy. The configuration program will remember the changed values the next time you choose Copy Database.

The Copy Database command requires that there be a Windows user on the remote node with the same username and password as the session under which you are running the configuration program. This user must have read access to the files in the MySQL\Data\micstats directory.

Please note that if you change the IP address of the other MOVEit Automation node in this dialog, the Copy Database parameters will not automatically pick up on the change. If, however, you are using hostnames to define your other host, you may not need to make a change here.

Virus Tab



The Virus tab configures how MOVEit Automation interacts with third-party real-time antivirus programs.

- **Definite Virus.** This setting configures what MOVEit Automation does if it determines that a specific antivirus scanner has identified a file as having a specific virus.
- **Probable Virus.** This setting configures what MOVEit Automation does if a file it is processing is suddenly deleted, but either:
  - MOVEit Automation cannot determine that the deletion was done by an antivirus program, or
  - MOVEit Automation cannot determine from the antivirus program which virus was detected.

  Typically, this situation arises when an antivirus program that is not supported by MOVEit Automation is running.

For each case, MOVEit Automation can take any of the following actions (in addition to marking the individual file transfer as failed and task as partially failed):

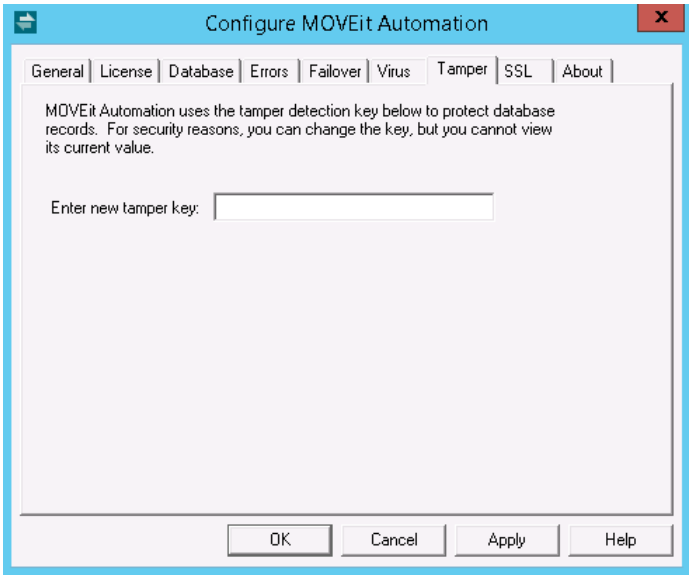
§ **Attempt to delete the source file.** If the source file cannot be deleted from the host (perhaps due to insufficient permissions), MOVEit Automation will "remember" the source file, as below.

§ **Remember the source file and do not download it again.** MOVEit Automation makes an entry in the "Task Transfer Exceptions" list for this task, which causes future runs of this task to not download the file. If you want MOVEit Automation to download the apparently infected file again, remove these entries using the MOVEit Automation Admin right-click option **Edit Task Transfer Exceptions**.

§ **Do nothing.** The file can be downloaded again next time. This option is rarely useful because the antivirus scanner is likely to intercept the file again, causing the same situation to occur again.

See also ***Advanced Topics - Antivirus*** (see "***Antivirus***" on page 254).

Tamper Tab



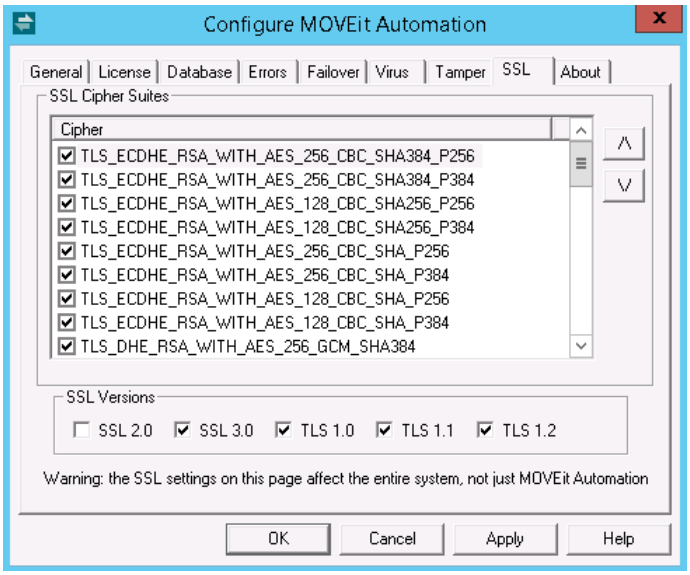
The Tamper tab is used to reset the tamper detection key used to protect database records. Any value typed here will be encrypted and then stored in the registry. (Do not copy the "HashKey" value from the registry into this field; instead copy "HashKey" registry values from one registry to another.) The derived tamper key is used to maintain a cryptographic "hash chain" of database records. For security reasons, the current tamper detection key is not shown; you can use this tab only to set a new tamper detection key. Because the tamper detection key is set during installation and should not be changed once set, there is rarely a need to use this tab.

SSL Tab

MOVEit uses Microsoft's built-in TLS/SSL security support provider (Schannel.dll). In all supported versions of Windows, there are several available protocols and cipher suite options enabled by default. Not all of them will meet your security and compliance needs. For example, the much older SSLv2 protocol is enabled by default on the server but is not allowed for PCI-compliant web applications. Be careful to choose the right mix of strong encryption methods and acceptable client support.

Warning: Changing the cipher suites or TLS/SSL versions can affect any applications that use TLS/SSL. Be sure you are aware of the requirements of other applications before making a change. For example, if you select SSL 2.0 only, MOVEit will not be able to connect to the Microsoft SQL database. The intent of this dialog is to allow you to avoid using a weak cipher where not allowed by PCI, FIPS, or other standards.

You can use the SSL tab to select the cipher suites and SSL versions that can be used when establishing an SSL session.



Selecting SSL Encryption Methods

The SSL Cipher Suites section allows you to choose which cipher suites are permissible, and their order of preference. A cipher suite is a named combination of authentication, encryption, and message authentication code (MAC) algorithms used to negotiate the security settings. By default, all ciphers suites enabled in the base Windows OS are enabled.

Note: Both the client's and the server's preferences are taken into consideration when choosing the actual cipher for a given session. Though the server's first choice won't always be chosen, the cipher that ends up being chosen will always be in the set of allowed algorithms on both sides.

Select the Enabled check box to disable a selected entry or to enable an unselected entry.

Entries closer to the top of the list are given preference over entries lower down. Use the arrow buttons to move entries up or down in the list. Even if you must permit weaker cipher suites, you should always put the stronger ones at the top of the list.

Selecting SSL Versions

SSL Versions are shown at the bottom of the SSL Tab. The default selections include SSL 2.0, SSL 3.0, and TLS 1.0. The versions selected determine the cipher suites that are available.

Select a check box to disable a selected version, or to enable an unselected version.

Note: After any SSL Version change, you need to reboot the system before the change takes effect.

Note: Be aware that the security policy setting System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing will restrict the available cipher suites and protocols. For example, TLS 1.0 will always be enforced.

Note: Be aware that the setting SSL cipher suite order via Group Policy will override any changes made to cipher suite order on this tab.

How to Test SSL Changes

To test SSL changes, first obtain a copy of OpenSSL. You can get OpenSSL.exe from the ***OpenSSL Project*** (***http://www.openssl.org***). Consult the following examples which show how to use this client and understand the information it provides.

(You need to type the commands in purple. Look for the results in red.)

Using OpenSSL to verify TLS1 is running on a remote server

This test was performed against our moveit.ipswitch.com support server. It shows that a connection using TLS1.

```
openssl s_client -connect WIN-TRL4JLD99D8:3471 -tls1
Loading 'screen' into random state - done
CONNECTED(000000FC)
depth=0 CN = WIN-TRL4JLD99D8, OU = Testing
verify error:num=18:self signed certificate
verify return:1
depth=0 CN = WIN-TRL4JLD99D8, OU = Testing
verify return:1
---
```

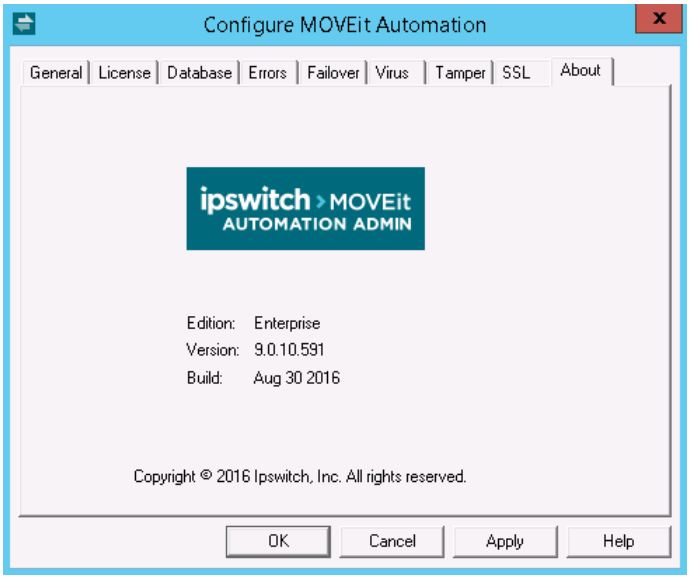
```
Certificate chain
 0 s:/CN=WIN-TRL4JLD99D8/OU=Testing
   i:/CN=WIN-TRL4JLD99D8/OU=Testing
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIC0TCCAbmgAwIBAgIBADANBgkqhkiG9w0BAQUFADAsMRgwFgYDVQQDEw9XSU4t
VFJMNEpMRDk5RDgxEDAOBgNVBAsTB1Rlc3RpbmcwHhcNMTQxMjA1MjE0MjUxWhcN
MjMwNDA1MjE0MjUxWjAsMRgwFgYDVQQDEw9XSU4tVFJMNEpMRDk5RDgxEDAOBgNV
BAsTB1Rlc3RpbmcwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC+Alcm
qM+OpzRA1Sk+qr8Ofsvnt52sVg/B+J9b5UsRWabrNAulXMmh6gHLOXGzZMR4ouha
IXRT91aOoJa4NBnoT/oymRnv4tSIDc77jG/fiXTvX5R+XXvpo83npfuuRIu+9S1S
F1C3elO/c6B2imeblee16HN+x6GNgRaLCX5fFKsdZaM9LEAxCTGcq2bBjOEQiAXH
uZ+S5yu7BUZcIJCoJh5WRDYS1GSlIonMBGgOHCgXOLHjh7mqQvMvugkF8ldz5piZ
mEnHdKnbxXqGhhh/kU7E6OuELc9E+Iscm2ZY2zJt9gylyoLmmm+l102G5bvzHEDT
3Trceu6GHj2mnd+dAgMBAAEwDQYJKoZIhvcNAQEFBQADggEBAC+wCLSVwfR8tkWH
tVh0Nd9pyMdMuyDNmR9M1041FBf94F+6SXXSL/R6pcfOVgYqGdzhZqEg/xATD4RW
wnAEg+pSTW0WrJBiSLRIzbV3ykESjHdk5s53wJfCRaOYAgqeWjxy2AHpFTEd7K4Y
OQ98z5sdWQ2ZERttlcTZi1PwTyEgOqpGJqGJ1Bi4sMUPqX8+Ob95zESW5A2dzaB0
F6SUZosCkC8+00QTOhnnGW/MJk02ox5AwOUeVs1HJk/U9IydDy3Knka2gDed9W04
b4f5EAeoGzihBWI3GEKW44Tbmtm17nQTILmtyaBOPJPaRzl5LdG86tDc1SOAcOPz
lHnv2mg=
-----END CERTIFICATE-----
subject=/CN=WIN-TRL4JLD99D8/OU=Testing
issuer=/CN=WIN-TRL4JLD99D8/OU=Testing
---
No client certificate CA names sent
---
SSL handshake has read 1215 bytes and written 349 bytes
---
New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES256-SHA
Server public key is 2048 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
     Protocol  : TLSv1
   Cipher    : ECDHE-RSA-AES256-SHA
    Session-ID:
3F2700008883D23306B46F8A63A138348A412268F08E35B9E016401FD4C98854

    Session-ID-ctx:
    Master-Key:
D54E13EE42BFF28224222EB489758CAA37A1F24D1073F98414EF0A379EE9249F
D4647C1BEC129F306714A5CD17780980
    Key-Arg   : None
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    Start Time: 1418072824
    Timeout   : 7200 (sec)
    Verify return code: 18 (self signed certificate)
```

Using OpenSSL to verify SSL 3 is NOT running on a remote server

(This test was performed against an internal IIS server after SSL3 was manually disabled.)

```
openssl s_client -connect WIN-TRL4JLD99D8:3471 -ssl3
Loading 'screen' into random state - done
CONNECTED(0000012C)
2980:error:1408F10B:SSL routines:SSL3_GET_RECORD:wrong version number:.\ssl\s3_pkt.c:338:
---
no peer certificate available
---
No client certificate CA names sent
---
SSL handshake has read 5 bytes and written 7 bytes
---
New, (NONE), Cipher is (NONE)
Secure Renegotiation IS NOT supported
Compression: NONE
Expansion: NONE
SSL-Session:
    Protocol  : SSLv3
    Cipher    : 0000
    Session-ID:
    Session-ID-ctx:
    Master-Key:
    Key-Arg   : None
    PSK identity: None
    PSK identity hint: None
    SRP username: None
    Start Time: 1418074464
    Timeout   : 7200 (sec)
    Verify return code: 0 (ok)
---
```

About Tab



The About tab shows the MOVEit Automation edition you have, the version number, and the software build date.

Proxy Servers

MOVEit Automation supports the following HTTP/S proxy server configurations when communicating with any MOVEit Transfer server:

- No Proxy Server
- Proxy Server, No Authentication Required
- Proxy Server with "Windows Integrated" NTLM Authentication

MOVEit Automation draws information about whether it should use a proxy server, and the host and port to which it should connect in order to access the proxy server from the Internet Explorer settings of the user under which MOVEit Automation runs as a service.

MOVEit Automation cannot make use of proxy servers to connect to MOVEit Transfer when it is running as "LocalSystem".

No Proxy Server

MOVEit Automation uses the hostname/IP address and port number configured in its service user's Internet Explorer settings for its proxy settings. If these values are empty, no proxy server is in use.

Proxy Server, No Authentication Required

MOVEit Automation uses the hostname/IP address and port number configured in its service user's Internet Explorer settings for its proxy settings. If these values are filled in, a proxy server is in use and MOVEit Automation attempts to connect through it.

Proxy Server with "Windows Integrated" NTLM Authentication

MOVEit Automation uses the hostname/IP address and port number configured in its service user's Internet Explorer settings for its proxy settings. If these values are filled in, MOVEit Automation knows that a proxy server is in use and will attempt to connect through it.

If the proxy server requires "Windows Integrated" NTLM authentication, MOVEit Automation will automatically present the proxy server with the same username and password credentials the MOVEit Automation service already provides to sign on the Windows server.

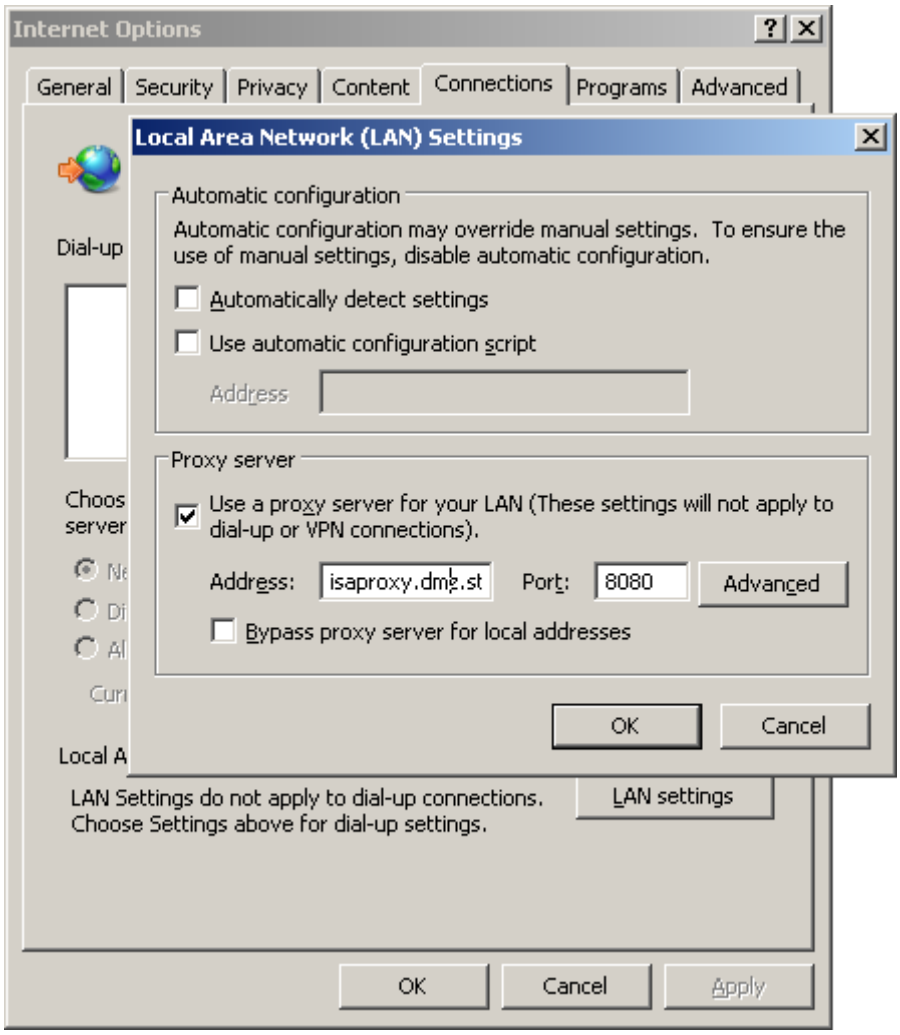
Internet Explorer's Proxy Settings

To access and change the MOVEit Automation proxy values, you will likely need to sign on interactively as a MOVEit Automation service user and launch Internet Explorer as that user.

To get to the Internet Explorer proxy settings:

1. Open Internet Explorer's multi-tab "Options" panel.
2. Go to the "Connections" tab.

3 Click the "LAN Settings" button.



Proxy Server Settings on Non-MOVEit Hosts

When MOVEit Automation is not connecting to a MOVEit Transfer server, proxy settings may be configurable items in related host entries. For example, AS2 hosts have their own proxy settings.

Backup

Information required to back up MOVEit Automation includes:

§ List and description of critical files for disaster recovery
§ Steps to use replication for a hot standby server

Disaster Recovery

An installation can use MOVEit Automation to copy critical files to a remote server for backup to tape, or to configure a hot standby.

Back up the following critical files and folders regularly. They are located in the Program Files\MOVEit directory.

Critical files.

- miccfg.xml - MOVEit Automation configuration file. Contains a complete list of tasks, hosts, certificates and other information.
- michash.xml - MOVEit Automation "hash" file. Contains tamper detection information.
- PGPPath\secring.pgp and PGPPath\pubring.pgp - PGP keyrings. Contains the PGP keys used by MOVEit Automation, if the optional PGP capability has been licensed. (These files are found in the PGPPath subdirectory of the MOVEit directory.)

Critical folders. Back up the folders and their contents.

- StateFiles – Contains MOVEit Automation state files for individual hosts/tasks. These files contain important information such as date/timestamps for folders and files, saved directory listings for synchronization tasks, and more.

Note: The configuration file and state files are encrypted. Before you can make changes to them, you must first use MOVEit Automation Admin to authenticate to MOVEit Automation..

Statistics database.

Back up the Statistics database regularly. This information is stored in a database.

- If you are using MySQL as your database engine, the database server is usually installed in the "C:\mysql\" or "D:\mysql\" directory. Back up alll files in the "..\mysql\data\MICStats" directory should be backed up to protect the statistics database.
- If you are using Microsoft SQL Server, contact your database administrator for information on backing up the database.

Client certificates to be backed up.

The Microsoft Windows certificate store contains the SSL server certificate used to secure communications with MOVEit Automation clients. If you are using client certificates , S/MIME or AS1/AS2/AS3, the Windows store also contains the related certificates.

To create a backup of the client certificates, run a MOVEit Automation task that uses the ***Certs Backup*** (on page 107) script.

It is less important to back up the MOVEit Automation SSL server certificate used to protect MOVEit Automation Admin and MOVEit Automation API connections because a new server certificate can be created by doing a fresh install.

Registry items to be backed up:

- The registry key HKEY_LOCAL_MACHINE\Software\Standard Networks\MOVEitCentral contains values such as:
  - A pointer to the SSL server certificate used to secure communications between MOVEit Automation and MOVEit Automation Admin/API
  - The ODBC DSN used for the statistics database

- The temporary cache directory
- The license key that enables MOVEit Automation

- The Windows user database. If you have created Windows users and/or groups for use by MOVEit Automation Admin users, the Windows user authentication database should also be backed up. This database is part of the Windows registry.

The registry is typically not changed often, so occasional backups should be sufficient.

Configuration Aging

Each time MOVEit Automation successfully saves a copy of its XML-based configuration files (miccfg.xml, etc.), it also ages up to four older versions in a simple grandfather scheme. For example, the "three saves ago" file is titled "miccfg.OL3".

If one of these files gets corrupted (often by an over-zealous anti-virus or backup utility), it is often possible to restore MOVEit Automation to a close-to-current state by copying one of the aged copies over the appropriate configuration file. For more information, contact ***MOVEit support https://www.ipswitch.com/support/***.

Automated Configuration Replication

To maintain a hot standby MOVEit Automation system, MOVEit Automation can be used to replicate its own configuration files to a standby MOVEit Automation server. The process involves setting up a locked-down FTP server on the standby server, and configuring the primary MOVEit Automation to upload its configuration files to the standby server using the FTP server.

NOTE: In order for replication of the configuration files to work properly, both systems must be running the same version of MOVEit Automation. Also, there must be a network connection between the two servers.

Please also see the ***MOVEit Automation Failover*** (on page 373) documentation.

Setting Up Replication

This procedure sets up configuration replication between a production Central (primary) and a hot standby MOVEit Automation (secondary).

1. Install MOVEit Automation on the secondary server. Stop the MOVEit Automation service and set the start method to MANUAL.
2. Install IIS FTP services on the secondary server. Configure IIS FTP in the following way:
   a) Add a new Windows user:
      1. On your desktop, right-click **My Computer** and select **Manage**.
      2. Open the Configuration \ Local Users and Groups \ Users tree.
      3. Right-click the **Users** folder and select **New User**. Provide the following information:
         User name: micftp
         Password: Type any password
         User must change password at next login: UNCHECK this box
         User cannot change password: CHECK this checkbox
         Password never expires: CHECK this checkbox
      4. Click **Create**.
   b) Assign permissions to the new Windows user:

1. Browse to the \Program Files or \Program Files (x86) folder, depending on your system architecture.
2. Select the MOVEit folder, right-click and select Properties.
3. Click the Security tab, click Edit. Click Add. then select the local computer list of users and select the new micftp user.

   Click Add, and then click OK.
4. Click the Security tab, select the micftp user and turn on the FULL CONTROL option.
5. Close this dialog.

c) Install the IIS FTP service if required.

1. If the IIS FTP service is not installed, right-click My Computer and select Manage. The Server Manager dialog box opens.
2. Click the Roles section. In the resulting display, under the "Role Services" section, click the "Add Role Services" option.
3. Find and select the "FTP Server" option (may be called "FTP Publishing Service" on some machines) from the list of available Role services and click the "Next" button. Click the "Install" button to complete the installation.

d) Configure the IIS FTP service.

1. Open the Internet Information Services manager console.
2. Right click the Sites subsection and select Add FTP Site.
3. Give the FTP site a name and select the \Program Files\MOVEit directory as the physical path for the content directory. Click Next.
4. Enter the desired IP Address, Port, and SSL settings. Click Next
5. Fill in the fields as follows:

   Authentication type: Basic

   Allow access to: Select Specified users and type micftp. Select Read and Write options.

   Click Finish. The FTP site is added.
6. Verify the FTP site is started and test the connection from the other MOVEit Automation node.

3 On the primary MOVEit Automation, create a Certs Backup task to back up client certificates.

a) Create a new task with a process, destination and schedule (no source).

b) Add a PER-TASK process that runs the Certs Backup built-in script.

   Use the default output filenames CertsPersonal.pfx and CertsOtherPeople.pfx. Specify a password for the output PFX files.

c) Add a destination that copies the file to \Program Files\MOVEit.

d) Add a schedule to run the task periodically every day.

4 On the primary MOVEit Automation. create a Certs Restore task to restore client certificates.

a) Create a new task with a source, process and destination (no schedule).

b) Add a source that loads Certs*.pfx from \Program Files\MOVEit.

c) Add a PER-FILE process that runs the Certs Restore built-in script. Specify the same password used by the above task.

d) DO NOT schedule the task. This task will not be run under normal circumstances; it will be run manually by operator after a failover, on the newly-promoted primary node.

5 Start broadcasting the MOVEit Automation configuration from the primary server to the secondary server.

a) Add a new FTP host that points to the secondary MOVEit Automation IIS FTP.

b) Create a new **Backup MOVEit Automation** task:

1. Source: Local File \Program Files\MOVEit\miccfg.xml
2. Source: Local File \Program Files\MOVEit\michash.xml
3. Source: Local File \Program Files\MOVEit\CertsPersonal.pfx
4. Source: Local File \Program Files\MOVEit\CertsOtherPeople.pfx
5. Source: All Local Files/Folders under \Program Files\MOVEit\StateFiles
6. Destination: FTP Host (secondary server); directory /; enable the Overwrite Files option

c) Schedule the task to run every X minutes (5 minutes, 30 minutes?).

d) Test the movement of the configuration files.

e) Create a second **Backup MOVEit Automation 2** task to handle the PGP keyrings.

1. **Source**: Local Files \Program Files\MOVEit\PGPPath*.pgp
2. **Destination**: FTP Host (secondary server); directory PGPPath; enable the Overwrite Files option

f) Schedule the task to run every X minutes (5 minutes, 30 minutes?).

g) Test the movement of the PGP keyrings.

6 Test the entire procedure:

a) On the primary server, stop the MOVEit Automation service. Use the MOVEit Automation Admin Shut Down Service command if tasks could be running).

b) On the secondary server, start the MOVEit Automation service, and then run the **Certs Restore** task.

c) Confirm that the secondary server's configuration is identical to the primary server's configuration.

Failover (Enterprise Only)

MOVEit Automation has the capability of running in *failover mode*, in which one server automatically stands in for another if the first one fails.

NOTE: This information describes the Legacy Failover system, which has been replaced by the Ipswitch Failover Manager. The Ipswitch Failover Manager is available. It requires a separate installation and managed externally. For more information, go to the Ipswitch support site, and under Secure Information and File Transfer, in the dropdown box, select ***Ipswitch Failover https://www.ipswitch.com/support/documentation***.

Ipswitch will continue to support Legacy Failover installations for MOVEit Automation. The Legacy Failover system is built into MOVEit Automation. The Legacy Failover system is included in the MOVEit Automation documentation. The new Ipswitch Failover system is documented separately; see the above link.

Overview

This overview describes:

Requirements

MOVEit Automation Failover requires:

§ Two computers running Windows Server 2008 or Windows Server 2012 and using the same type of database server(for supported operating systems and databases, see Requirements).

  Note: If you are migrating MOVEit Automation to a new server, see the following Knowledge Base articles:

  § ***How do I migrate my MOVEit Automation to another server?*** (***https://community.ipswitch.com/s/article/ka036000000kL0mAAE/How-do-I-migrate-my-MOVEit-Central-to-another-server-1307565958579***)

  § ***How do I get MOVEit Automation to work with Microsoft SQL Server?*** (***https://community.ipswitch.com/s/article/ka036000000kNN8AAM/How-do-I-get-MOVEit-Central-to-work-with-Microsoft-SQL-Server-1307565983335***)

§ MOVEit Automation. The failover capability is built into all versions of MOVEit Automation; however, it must be enabled by a special license.

§ A license that enables the failover capability. The same license key can be used on both computers.

§ A TCP connection between the two computers that allows access to ports 3472 and 3473. Also, if you use the MySQL database, NetBIOS over TCP must be enabled between the two machines to allow the "Copy Database From" action to occur.

See ***Failover Installation*** (see "***Installation***" on page 377) for how to install failover.

How Failover Works

In a failover configuration, there are two computers running MOVEit Automation. At any given time, one of them is the "primary node" and the other is the "secondary node". The primary node is responsible for running all tasks, and for accepting all connections from MOVEit Automation Admin users. The secondary node is passive: its only responsibility is to maintain an up-to-date copy of the primary node's settings, and to promote itself to the primary node if the other node fails. The secondary node does not run tasks or allow MOVEit Automation Admin to make changes to its configuration.

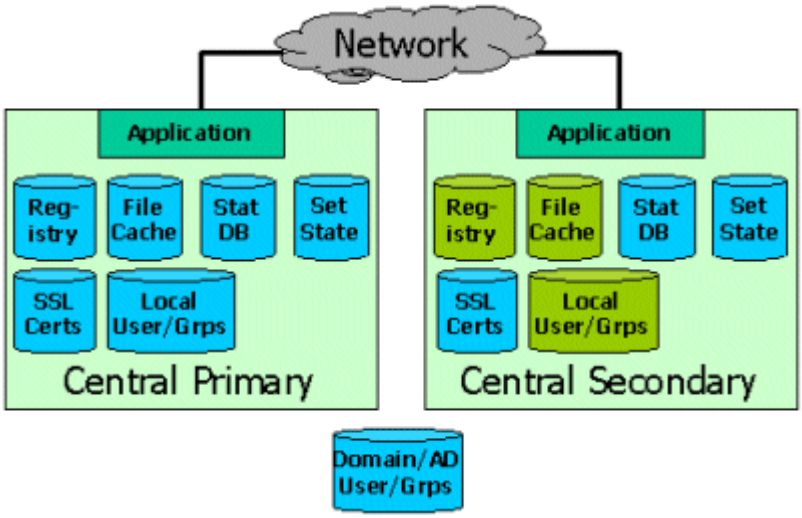
The secondary node connects to the primary node via the same TCP interface that is used by MOVEit Automation Admin. It uses this interface to determine the health of the primary node: if it cannot connect for a few minutes, it will assume that the primary is dead and will become the primary itself. The secondary node also uses this TCP interface to replicate changes on-the-fly from the primary node. Also, both nodes use this interface to ensure that there is exactly one primary node at all times. This prevents a situation in which both nodes are primary, and potentially transferring files twice.

Status information on the failover aspects of the system is available on the Failover tab of MOVEit Automation Admin.

What Failover Replicates

The following settings are automatically replicated from the primary node to the secondary node:

- The configuration file miccfg.xml, which contains most MOVEit Automation settings, including definitions of tasks, hosts, scripts, SSH keys, date lists, and so on.
- The StateFiles folder, which contains various XML configuration files specific to individual hosts/tasks. Each "state" file contains information such as date/timestamps for determining file newness, saved directory listings for synchronization tasks, and more.
- The tamper detection file michash.xml, which contains information used to detect tampering of the database.
- The PGP keyrings, PGPPath\secring.pgp and PGPPath\pubring.pgp. These files contain the PGP keys used by MOVEit Automation, if the optional PGP capability has been licensed.
- The MICSTATS database, a database which contains a record of tasks that were run, files that were transferred, and administrator actions that were performed.
- Creation, deletion, and other manipulation of local Windows users and groups used to access MOVEit Automation. (Domain users and groups don't need to be replicated as long as both failover nodes are members of the same domain.)
- SSL certificates, in the Microsoft Windows certificate store. This includes:
- Client certificates, with private keys, optionally used to identify MOVEit Automation when connecting to secure FTP and MOVEit Transfer servers.
- Server certificates, with private keys, used to secure communications with MOVEit Automation Admin.
- Other people's certificates, without private keys, used for sending S/MIME email (a rarely used capability).



The following are not replicated:

- The registry key, which can be found in one of the following locations:
  - HKEY_LOCAL_MACHINE\Software\Standard Networks\MOVEitCentral
  - HKEY_LOCAL_MACHINE\Software\Wow6432Node\Standard Networks\MOVEitCentral

This key contains infrequently-changed settings such as the license key, the directory used for temporary files, and so on. These settings are maintained by the MOVEit Automation configuration program. If you run this configuration program and make changes on one node, you should make those same changes to the other node.

§ The temporary "cache" directory. Any files stored temporarily by a running task are not replicated to the secondary node. The state of any tasks that were running are lost. Those tasks will be run at the next normally scheduled interval on the secondary node after it takes over.

Database Replication

Note: If you use Microsoft SQL Server, a single database is shared by Node 1 and Node 2 (and generally installed on a third node), so database replication is unnecessary.

If you use the MySQL database, the database is replicated by the primary node sending SQL statements to the secondary node, which runs them itself on its own copy of the database. (Replication features built in to the database are not used.) During the usually short time between the original update of the primary database and the corresponding update of the secondary database, the SQL statements are stored in an encrypted file named MICSQL.blg. This buffering of SQL statements allows the replication to be done at a later time if the secondary node is down.

After a Failover

When the secondary node becomes a primary node, it enables the task scheduler, allowing tasks to be run. It also begins accepting connections from MOVEit Automation Admin. Because the other node is dead, the new primary node will initially not be able to replicate changes to it.

When the dead node comes back to life, it checks with the other running node before deciding whether to be the primary or secondary node. Assuming that the other node is still running in primary node, the formerly dead node will become secondary and will catch up on any changes made while it was down.

Failover Alerts

If a MOVEit Automation failover occurs, the following message are sent from the following MOVEit Automation node to the email address configured on the "Errors" tab of the MOVEit Automation Configuration utility.

§ From the SECONDARY NODE: "I cannot contact the other MOVEit Automation node. I was the secondary node, but I'm becoming primary." Upon receipt of this message, an administrator should take whatever steps necessary to restore the MOVEit Automation service on the old primary node.

When the MOVEit Automation service is restored on the old primary node, the following messages from the following nodes are sent:

§ From the PRIMARY (old secondary) NODE: "I was finally able to login to the remote MOVEit Automation."

§ From the SECONDARY (old primary) NODE: "Other node is running as primary. Even though we were primary last time, we'll be secondary now."

Resynchronizing

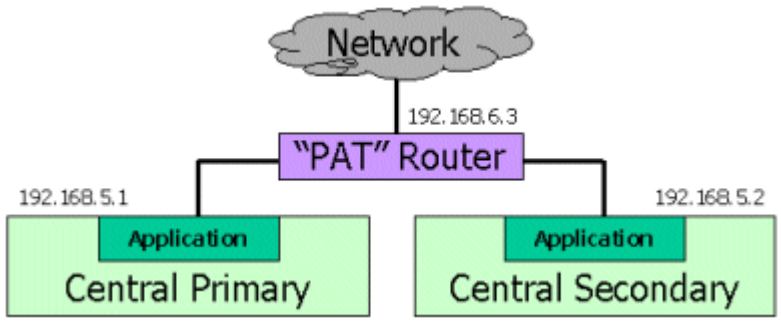
See the "***Failover - Common Procedures - Resynchronization*** (see "***Resynchronization***" on page 383)" documentation for this procedure. Also, see "***Central Service - Failover - Common Procedures - Node Swap*** (see "***Node Swap***" on page 383)" for instructions on switching the primary / secondary roles of two nodes.

MOVEit Automation Admin Considerations

The ***Failover Tab*** (on page 381) allows you to monitor failover status. Additionally, if you connect to a failover-enabled node that is not running in primary mode, all tabs other than Log and Failover will be empty.

How to Make Two MOVEit Automation Systems Appear As One

Firewalls and internal servers can be configured only to accept connections from a single IP address. In this case, it is suggested that installers set their network up in such a way that any non-MOVEit Automation machine sees the MOVEit Automation cluster as a single IP address. The best way to do this is to use a router that does network address translation and overloading of a single address with multiple sessions. (Cisco calls this "PAT" or "NAT overloading").



For example, if MOVEit Automation node #1 has IP address 192.168.5.1 and node #2 has IP address 192.168.5.2, a router can be configured to overload IP address 192.168.6.3. 192.168.6.3 would be the only IP address the rest of the world would know about; neither of the 192.168.5.* addresses would need to be configured in any IP-specific firewall or configuration.

Installation

Initial installation of a failover-capable MOVEit Automation node is the same as any MOVEit Automation install or upgrade, since the failover capabilities are built into all copies of MOVEit Central 3.2 or later. However, to make a MOVEit Automation system failover-capable, some additional steps need to be taken on each of the two nodes after the software have been installed on both nodes.

See also ***Failover Overview*** (on page 373).

Assumptions and Requirements

These instructions assume the following:

§ The same version of MOVEit Automation has been installed on both Node 1 and Node 2. The two nodes must use the same database type (MySQL or Microsoft SQL Server).

Note: If you use the MySQL database, MOVEit Automation maintains the databases on Node 1 and Node 2. If you use Microsoft SQL Server, a single database is shared by Node 1 and Node 2, and generally installed on a third node.

§ The same tamper detection key has been entered during installation on both nodes. If this is not the case, use RegEdit to manually copy the registry value for HashKey from Node 1 to Node 2. This registry value, which can be found in one of the following locations:

  § HKEY_LOCAL_MACHINE\Software\Standard Networks\MOVEitCentral\HashKey

  § HKEY_LOCAL_MACHINE\Software\Wow6432Node\Standard Networks\MOVEitCentral\HashKey

Note: Do not use the Tamper tab of the MOVEit Automation Config utility.

§ Node 1 is where the current MOVEit Automation configuration and database are located. If your configuration uses Microsoft SQL Server, the database can be on a third node.

§ Node 1 can connect to Node 2, and Node 2 can connect to Node 1, with Windows connectivity.

§ Any Windows users and groups used by MOVEit Automation are already present on both nodes. .After failover is installed and running, any changes made to Windows users and groups used by MOVEit Automation on one node are automatically replicated on the other node.

§ If using Microsoft SQL Server as the database, make sure that the SQL Server Utilities are installed on both nodes. MOVEit Automation uses these utilities to access the database to read and write the transaction records. When you install MOVEit Automation and select to use a SQL Server database, or if you ***convert a MySQL installation*** (see "***Converter***" on page 299) to SQL Server, MOVEit Automation installs the SQL Server Utilities. You can can also download the utilities from the Microsoft website.

Step-by-Step Instructions

Before you proceed, make sure that the previous ***assumptions*** (on page 378) are met.

1 Nodes 1 and 2: Using the Permissions dialog in MOVEit Automation Admin, create a Windows user (either a local or domain user) named centralfailover into the MOVEit Admin Windows group on both nodes. Use the same password on both nodes.

The centralfailover user will be used by the other node for replication. You can use an alternate username, but make sure it is the same username on both machines.

2 Nodes 1 and 2: Stop the MOVEit Automation service on both nodes. Use the MOVEit Automation Admin Shut Down Service command on Node 1 if tasks could be running. You can issue the command net stop moveitcentral or use the Windows control panel to stop the service on Node 2.

3 Nodes 1 and 2: If the ***MOVEit Automation Config*** (see "***MOVEit Automation Config Utility***" on page 354) program is not already running, start it via the shortcut located in Start > Programs > MOVEit Automation on both nodes.

4 Nodes 1 and 2: Enter the same failover-capable license key in the ***License tab*** (on page 356) on both nodes.

5 Nodes 1 and 2: Configure the email addresses of an administrator in the ***Errors tab*** (on page 358) on both nodes. *Recommended:* Provide these values so that you will be emailed when serious errors or failover events are encountered.

6 Node 1 ONLY: Go to the ***Failover tab*** (on page 358) and configure the following information:

- This Node - Startup Role: Primary
- This Node - Node Number: 1
- This Node - Nodes to ping: (*Optional: usually the IP address of trusted router, leave blank if unsure*)
- Other Node - Hostname or IP: [IP Address of Node #2]
- Other Node - MOVEit Admin user: centralfailover
- Other Node - Password: [centralfailover's Password]
- Click Apply.

7 Node 2 ONLY: Go to the ***Failover tab*** (on page 358) and configure the following information:

- This Node - Startup Role: Secondary
- This Node - Node Number: 2
- This Node - Nodes to ping: (*Optional: usually the IP address of trusted router, leave blank if unsure*)
- Other Node - Hostname or IP: [IP Address of Node #1]
- Other Node - MOVEit Admin user: centralfailover
- Other Node - Password: [centralfailover's Password]
- Click Apply.

8 Nodes 1 and 2: If you are using MySQL as your database, click the "Clear SQL Rep..." button on the ***Failover tab*** (on page 358) on both nodes.

Note: The first time that you click Clear SQL Rep..., if you receive error messages that state that there is nothing to replicate, you can ignore these errors.

9 Node 2 ONLY: If you are using MySQL as your database: On node 2 only, click the Copy Database button on the ***Failover tab*** (on page 358) to do a one-time replication of the database from the primary node to the secondary node. This might take several minutes if your existing database is large.

You might also need to change the Copy from value on the Copy statistics database dialog thatl appears. If you encounter 'From' directory does not exist or cannot be accessed errors, use the following command-line sequence to confirm the following:

- This MOVEit Automation can connect to the other MOVEit Automation via Windows File Sharing
- The centralfailover user can be authenticated to the other MOVEit Automation, and
- centralfailover has read access to the file share that is listed in the Copy from field.

```
C:\>net use H: \\192.168.3.172\micstats /user:centralfailover
The password or user name is invalid for \\192.168.3.172\micstats.
Enter the password for 'centralfailover' to connect to
'192.168.3.172': *******
The command completed successfully.
C:\>H:
H:\>dir stats.myi
09/22/2005 02:28 AM           53,248 stats.MYI
H:\>copy stats.myi c:\
1 file(s) copied.
```

10 Node 1 and 2: If you are using Microsoft SQL Server as your database: On both nodes, make sure that you set the registry entry DWORD named SuppressDBRep to a value of 1. This registry entry is found in one of these locations (you may need to add the registry entry):

- § HKEY_LOCAL_MACHINE\Software\Standard Networks\MOVEitCentral\Resil
- § HKEY_LOCAL_MACHINE\Software\Wow6432Node\Standard Networks\MOVEitCentral\Resil

You do not want the SQL Server database to be replicated because Node 1 and Node 2 share the same SQL Server database.

11 Node 1 ONLY: Start the MOVEit Automation service. (From a command line, you may issue the command "net start moveitcentral".)

12 Node 2 ONLY: Start the MOVEit Automation service. (From a command line, you may issue the command "net start moveitcentral".)

13 Node 1 ONLY: Open MOVEit Automation Admin and connect to "localhost". Within about three minutes, the ***MOVEit Automation Admin Failover tab*** (on page 381) shows that Node 1 is the primary MOVEit Automation node and Node 2 is the secondary MOVEit Automation node.

14 Node 1 ONLY: Use the MOVEit Automation Admin Command > Reset Tamper Detection menu item to reset tamper detection for the failover system.

Securing Connections Between Nodes

Recommended: On both nodes, assign an SSL certificate and choose "Require encryption for remote control" in the MOVEit Automation configuration program's ***General tab*** (on page 355), This allows MOVEit Automation to use SSL-encrypted TCP connections to replicate settings.

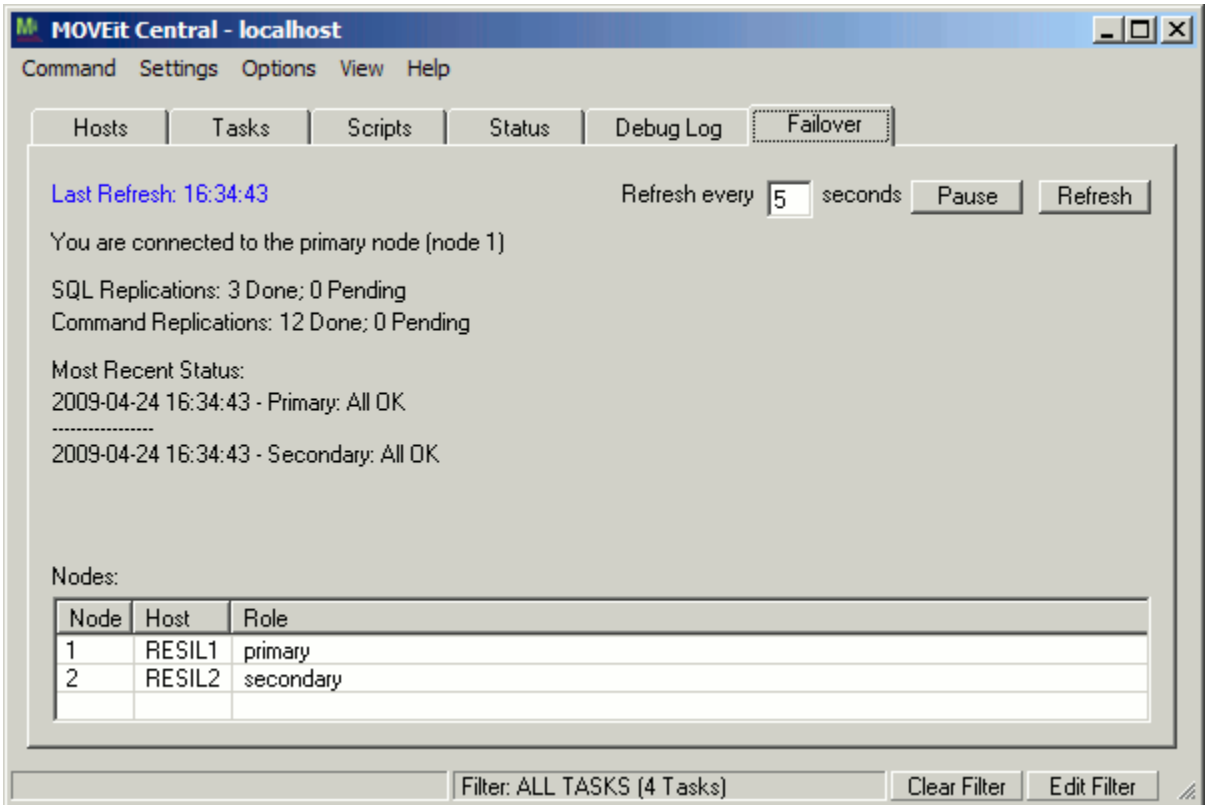
Updates

See the ***Failover - Common Procedures - Software Upgrade*** (see "***Software Update***" on page 382) documentation for information about upgrading MOVEit Automation when deployed in a Failover configuration.

Admin Failover Tab

The MOVEit Automation Admin Failover tab is used to monitor the failover status of MOVEit Automation. The Failover tab appears only when the failover feature is enabled. (MOVEit Automation must be licensed for failover, and its node number must be greater than 0.)

(See ***MOVEit Automation Config - Failover Tab documentation*** (on page 358) for information about the Failover tab in the MOVEit Automation Config utility.)



If you are connected to a primary node, the following information is displayed:

- The node number to which you are connected (1 or 2).
- The number of SQL statements that have successfully been replicated to the secondary node since MOVEit Automation has started.
- The number of pending SQL statements; that is, the number that are waiting to be replicated.

  If the number of pending statements is greater than zero, and the number of statements that have been replicated hasn't changed in 20 seconds, there is probably a problem.
- The failover status of MOVEit Automation, and its timestamp.
- A list of the nodes, with these items for each node:
  - The node number
  - The hostname
  - That node's current role

If the node you are connected to is not a primary, only the failover status is shown.

If the node changes roles, then the next time you are on the Failover tab, you will be asked to exit MOVEit Automation Admin and reconnect. This will allow MOVEit Automation Admin to reset its displays to correctly reflect the new role of the node.

See also ***Failover Overview*** (on page 373).

Common Procedures

You may perform some common procedures, such as:

- Backups
- Software upgrades
- Node swaps
- Resynchronization

Backup

Use normal backup procedures to back up your MOVEit Automation Failover primary node. Backup files created through these procedures can be restored on either standalone or failover MOVEit Automation systems. See the ***Central Service - Backup*** (on page 368) documentation for more details about recommended backup procedures.

Software Update

Before updating make sure both nodes have been ***synchronized*** (see "***Resynchronization***" on page 383). Then stop the MOVEit Automation service on both nodes (using the MOVEit Automation Admin Shut Down Service command on the primary node). After the services have been stopped on both nodes, run the update on the primary node first and make sure that the MOVEit Automation service is started on this node before you update the secondary node. (By upgrading the primary node first, no unnecessary failover will occur.)

After the update, you might receive an email from one or both MOVEit Automations reporting that they could not contact the other MOVEit Automation node. This is a result of having the service running on one node and not the other during the upgrade process. Youi can ignore this message.

Windows Updates

Windows updates can normally be applied without shutting the MOVEit Automation services down. However, if a reboot is required by a Windows update package, plan to take down the MOVEit Automation service on the secondary node, then boot the primary node and let it come back up before you reboot the secondary node. (Make sure that the secondary node also brings its MOVEit Automation service up after its reboot.)

Node Swap

You might want to switch the roles of a primary and secondary node back after a failover. For instance, if you have computers named CENTRALA and CENTRALB, you might find it convenient to have the CENTRALA computer normally be the primary node. Thus, after a failover from CENTRALA to CENTRALB, you may wish to force CENTRALA to be the primary node again.

To accomplish this:

1. Ensure that the new primary (CENTRALB in this example) actually does have the current task configuration.
2. Run briefly with both nodes up in the reversed role scenario, to allow any changed settings to replicate from recently-promoted CENTRALB to the recently-demoted CENTRALA.

   Important: Both nodes must be running to allow MOVEit Automation to sync the state file.
3. Follow the instructions for ***resyncing the database*** (see "***Resynchronization***" on page 383), but do not start the MOVEit Automation services. (Remember to always invoke the copy function from the current secondary node.)
4. Use the MOVEit Automation Configuration utility to set the appropriate values for "Startup Role" on each of the two nodes. (One will be set to "Primary" and the other will be set to "Secondary".)
5. Start the new primary MOVEit Automation, followed by the new secondary.

Resynchronization

You might need to resynchronize the two nodes under certain conditions. For example, after a failover, if the former secondary has become primary, you might want to force the original primary to be primary again. Follow these instructions to resynchronize the nodes:

1. Stop MOVEit Automation service on the current secondary node.
2. Stop MOVEit Automation service on the current primary node (using MOVEit Automation Admin's "Shut Down Service" command if tasks could be running).
3. Stop MySQL service on both nodes.
4. On the desired new primary node, run the ***MOVEit Automation configuration utility*** (on page 358) and:
   a) Set the startup role to primary.
   b) Choose the "Clear Admin Rep" button.
   c) Choose the "Clear SQL Rep" button.
5. On the desired new secondary node, run the MOVEit Automation configuration utility and:
   a) Set the startup role to secondary.
   b) Choose the "Clear Admin Rep" button.
   c) Choose the "Clear SQL Rep" button.
6. Determine which of the two nodes has the most current database. Generally, this is the last machine to have served as the primary node. Go to the OTHER node (the one with the less current database) and choose the "Copy Database" button to request a complete copy of the most current database. Wait for the copy procedure to complete.
7. Start MySQL service on both nodes.
8. Start MOVEit Automation service on the new primary node.

9 Start MOVEit Automation service on the new secondary node.

FTP Failover

This section discusses some additional failover-related configuration that can be done on a MOVEit Automation server that is also running an FTP server.

Overview

Some sites run an FTP server on the same computer as MOVEit Automation. Remote servers or mainframes send files to MOVEit Automation via FTP, and MOVEit Automation tasks process the files. For better performance and reliability, the local FTP directory is configured as a filesystem source rather than an FTP source.

In a failover scenario, the following types of problems can occur:

§ If MOVEit Automation server A fails after some files have been sent to MOVEit Automation, but before the related task runs, those files are not transferred when the secondary server B takes over, because they are on the failed computer. (If MOVEit Automation has been set to use filesystem notifications, this time window is short.) If the new primary B fails days, weeks, or months later, the old, unprocessed files on server A might be processed when it becomes the primary. Depending on how tasks are configured, this coulc cause obsolete files to be sent, confusing the recipient.

§ Remote processes that have been programmed to send files to FTP server A might not know to send those files to FTP server B if A is down.

These problems can be addressed with features built into Microsoft Windows which allow you to create a single system image from two FTP servers:

To address these problems, use the following Microsoft Windows features to create a single system system image from two FTP servers

§ Use Distributed File System to create a single storage area into which files sent via FTP to either of the MOVEit Automation computers are stored. See ***FTP Replication - DFS*** (see "***With DFS***" on page 385).

§ Use Network Load Balancing to assign a single IP address that can be used by remote computers to access either FTP server as if the two were a single computer. See ***FTP Replication - NLB*** (see "***With NLB***" on page 394).

See also ***Failover Overview*** (on page 373).

With DFS

Microsoft Distributed File System (DFS) can be used to create a single storage area into which filesthat are sent via FTP to either MOVEit Automation computer are stored. Files received by the FTP server on either computer are automatically copied to the corresponding directory on the other computer. The copying is done quickly, so the file appears on both computers nearly simultaneously (depending on the size of the file and other factors).

DFS is available on Windows Server. You must be a member of a domain to use DFS.

Overview

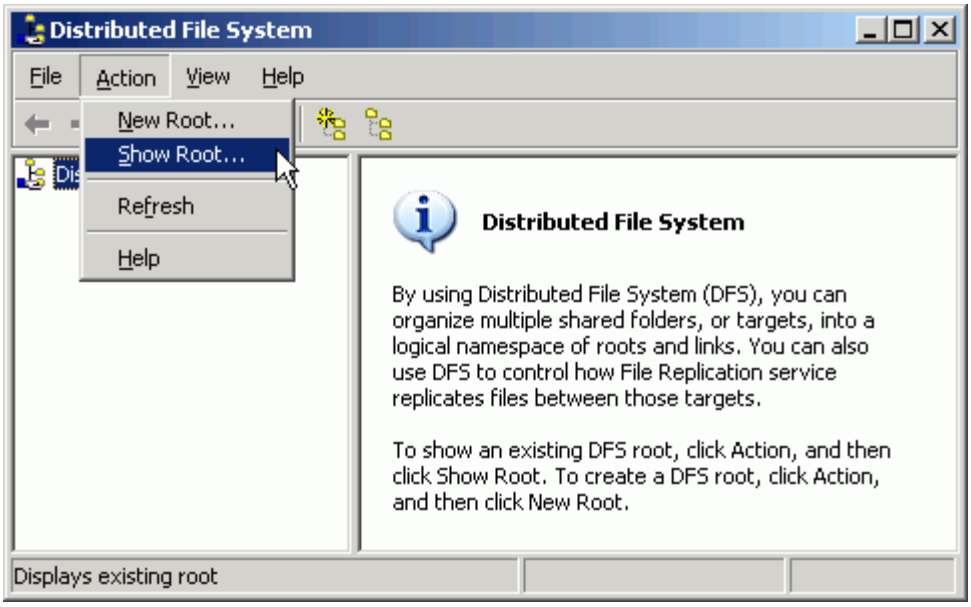
In brief, to use DFS, you take the following steps:

- Create a DFS root on your domain if none already exists
- On each FTP server, create a network share for the root FTP directory on that computer
- For each of these shares, add a link to that share to the DFS root
- Use the Replication Wizard to set up replication between these two links

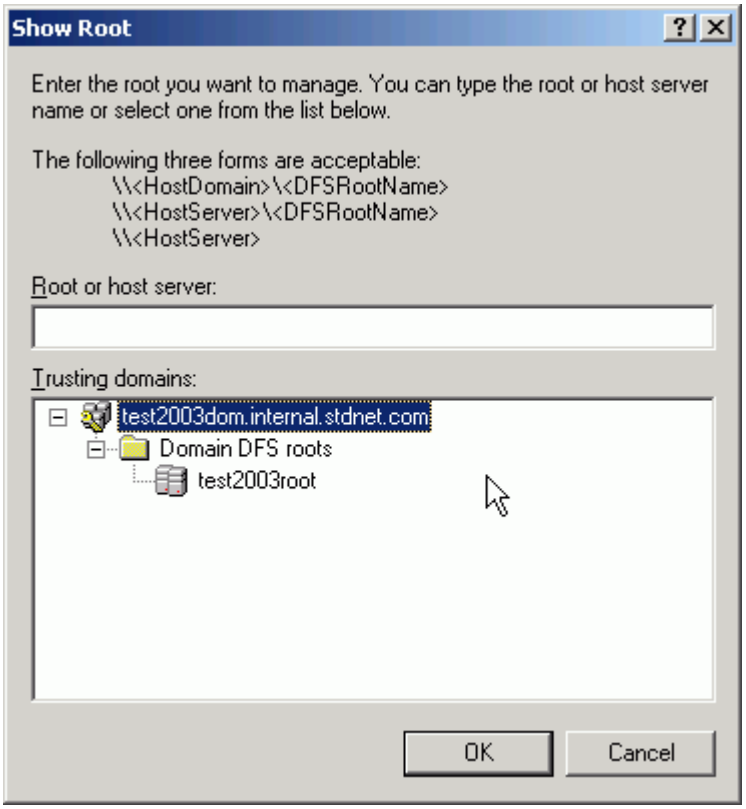
Details

1 Check to see whether a DFS root already exists on your domain:

a) Run Administrative Tools > Distributed File System and select Action > Show Root...

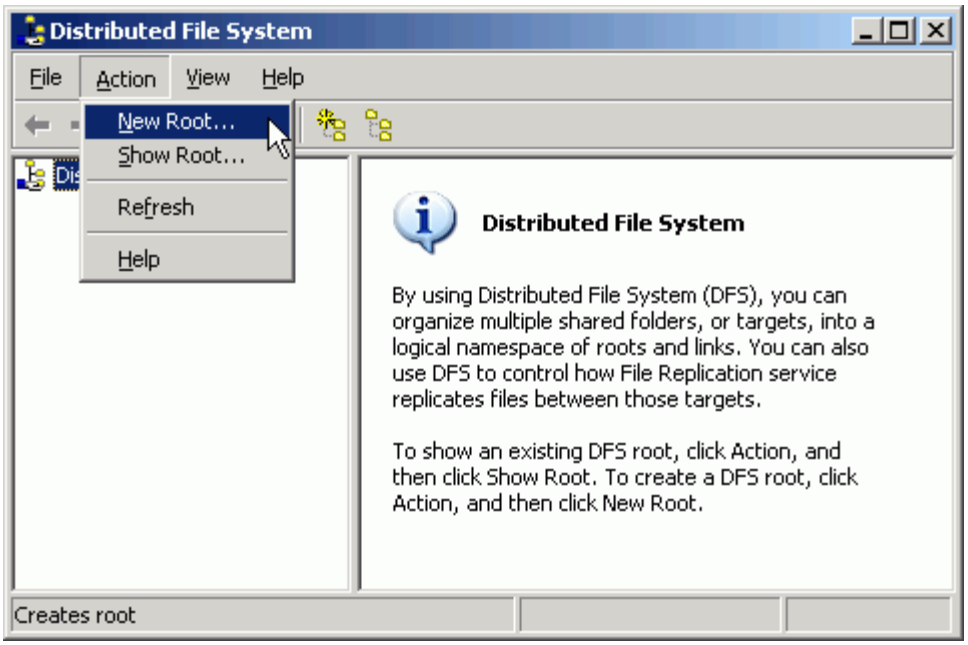


b) Expand Domain DFS roots and look to see if there are any entries beneath it. In this example, there is already a root named test2003root.

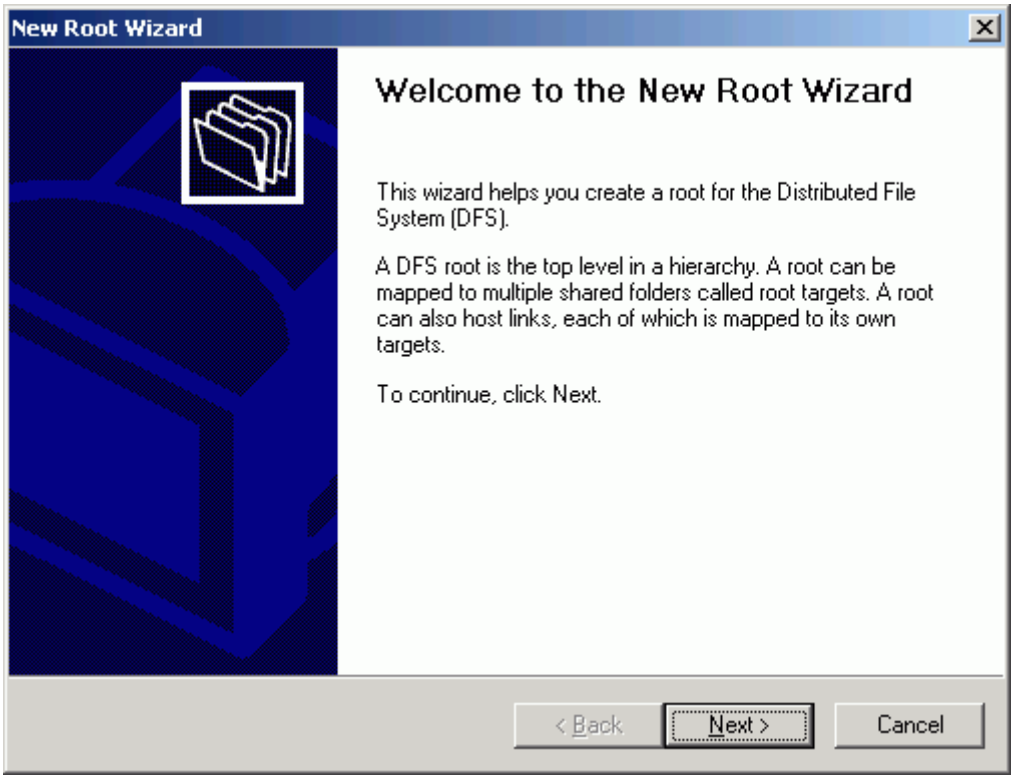


2 If there is not already a root, create one:

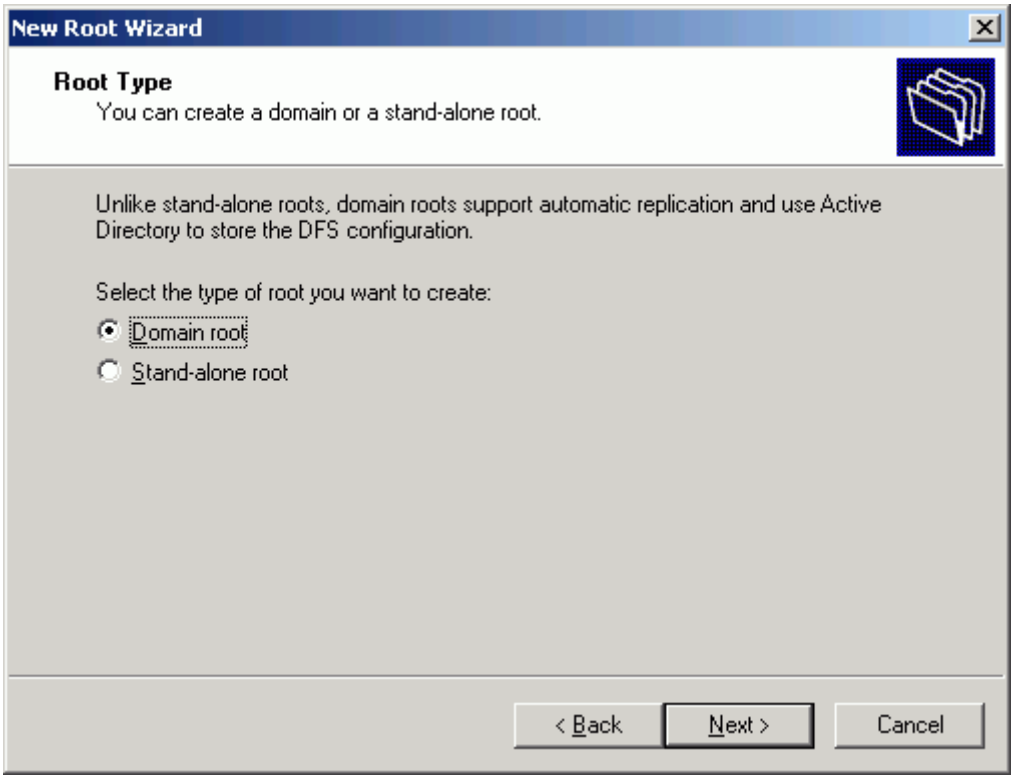
a) Select Action > New Root...



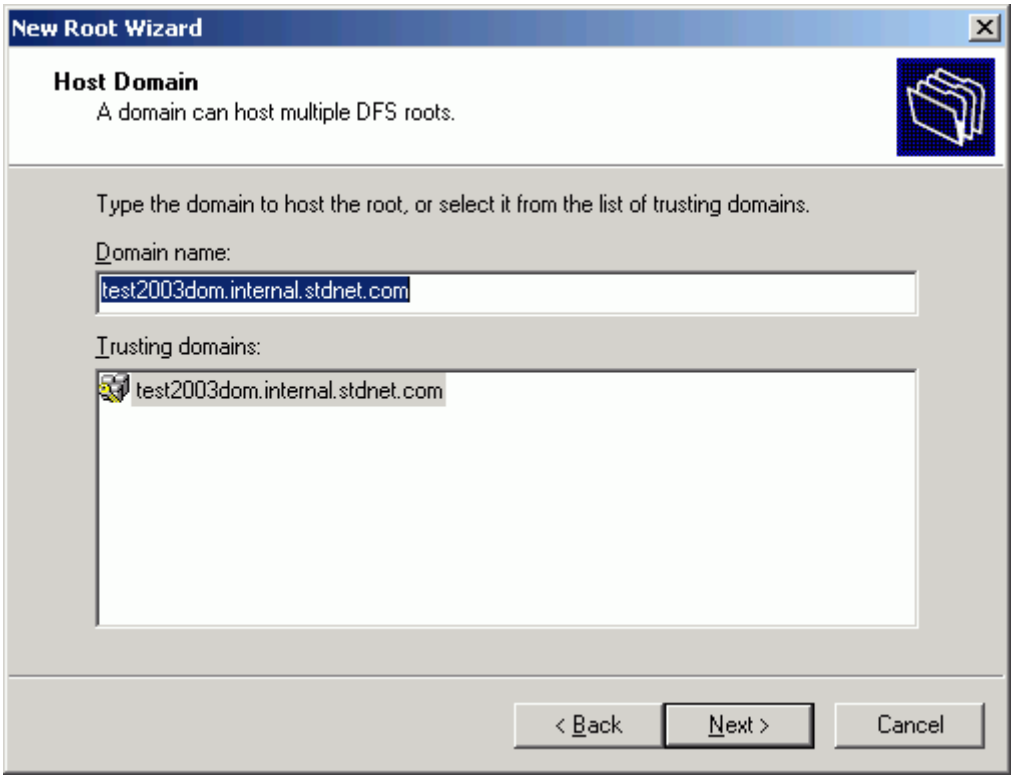
b) In the New Root Wizard, click **Next**.



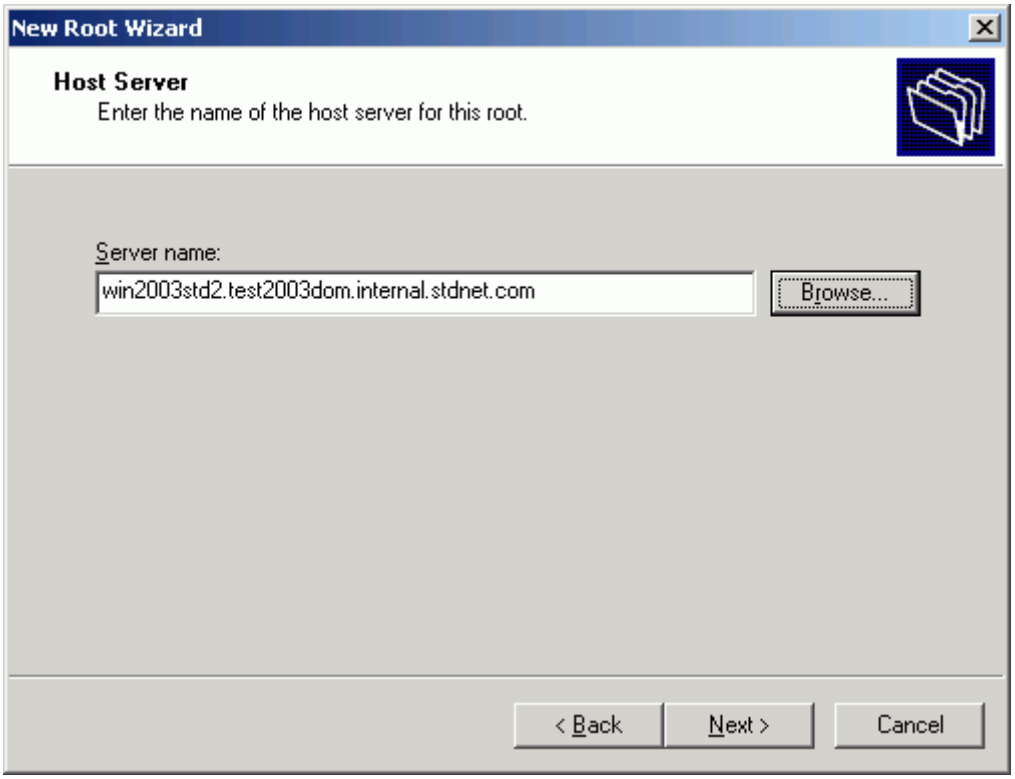
c) For Root Type, select **Domain root** and click **Next**. Replication requires the root to be of type Domain.



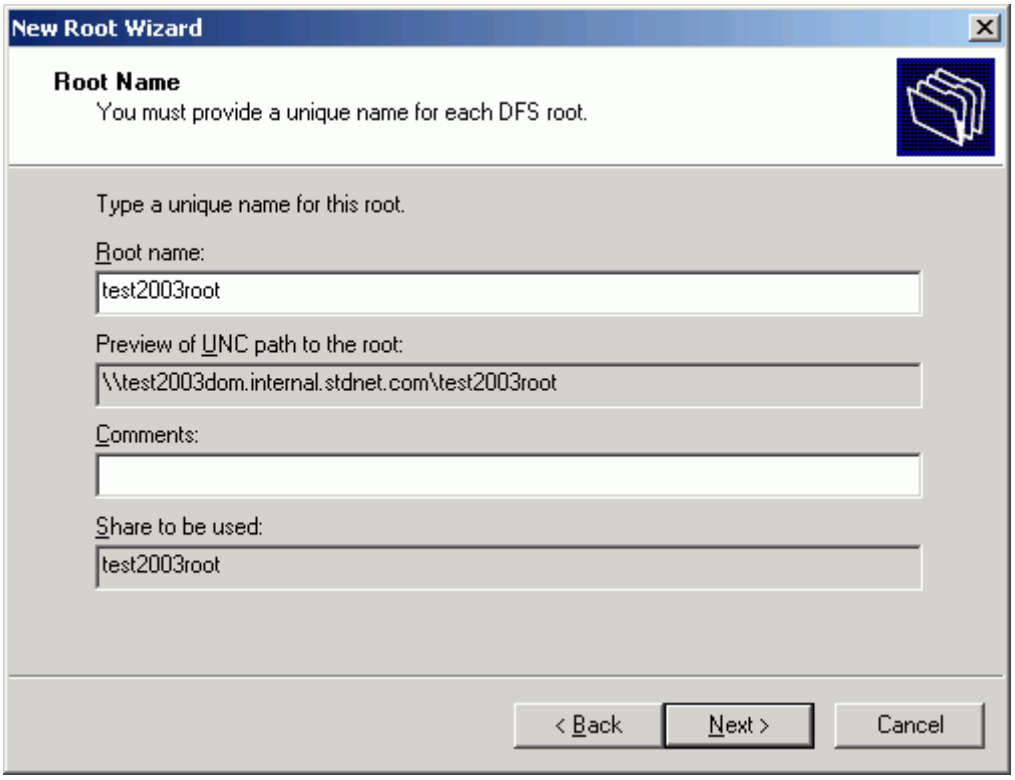
d) For Domain, select the domain and click **Next**.



e) For Host Server, click **Browse** and select the name of a MOVEit Automation server. (It does not matter which one you choose.) Click **OK** to close the dialog box. Click **Next**.



f) For Root Name, type a name of your choosing and click **Next**.



g) Click **Finish**.

3 Create filesystem shares on each of the MOVEit Automation computers, each pointing to the local root FTP directory. For example, if your two servers are named win2003srv1 and win2003srv2, and each has its IIS FTP root at c:\inetpub\ftproot, then you would create the shares:

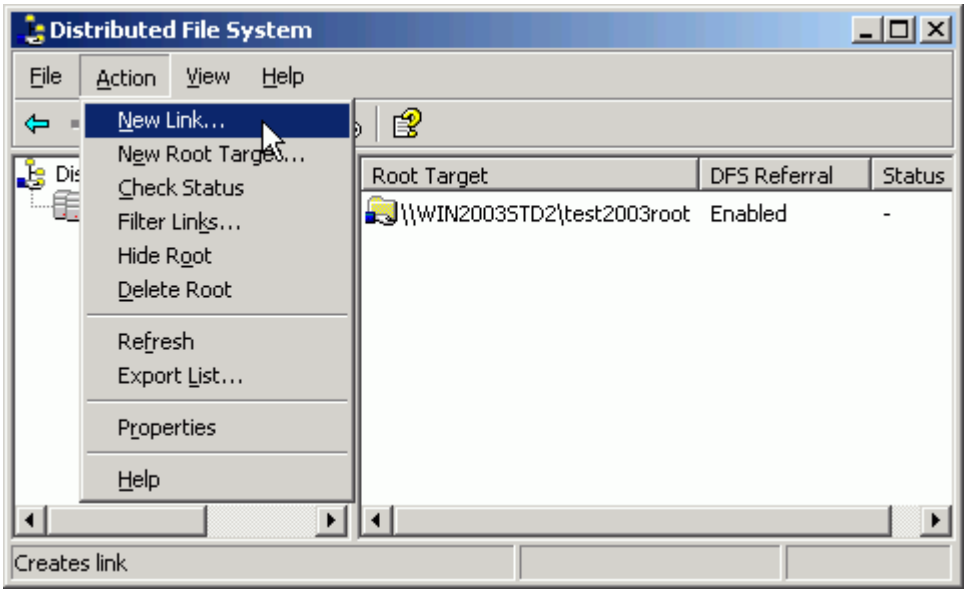
Sharename	Local directory pointed to
\\win2003srv1\ftpshare	c:\inetpub\ftproot
\\win2003srv2\ftpshare	c:\inetpub\ftproot

4 On one of the servers, create a link to the local ftpshare:

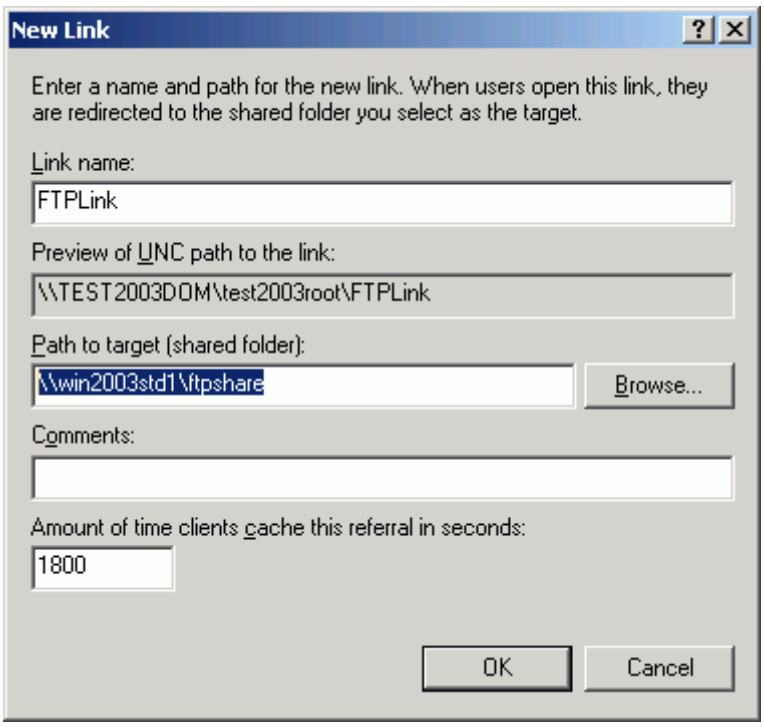
a) Run **Administrative Tools > Distributed File System** if it is not already running.

b) Select the DFS root you created. If the root is not visible in the tree, use Action > Show Root... as described in Step 1a.

c) Use **Action > New Link...** to open the New Link dialog.



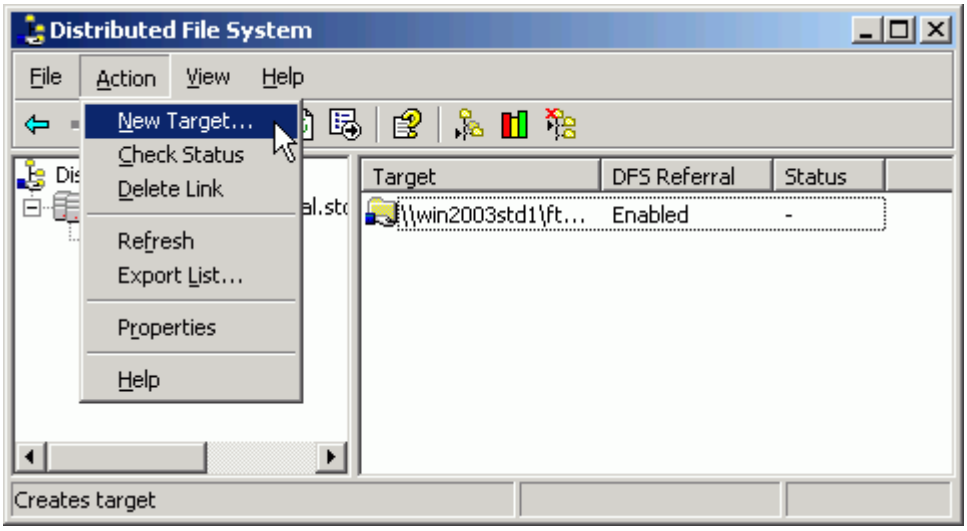
d) In the New Link dialog box, type a link name of your choice (for example, FTPLink) and type the full path to the share (for example, \\win2003std1\ftpshare) and choose OK.



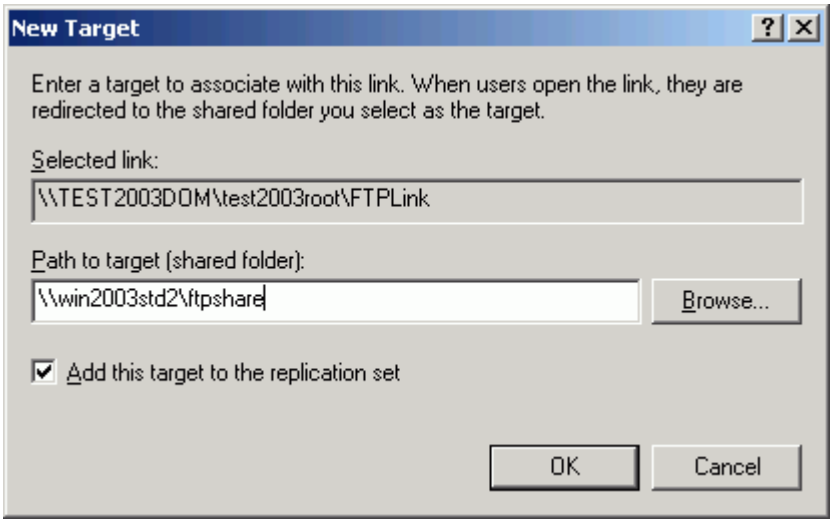
5 Create a second target to that link and enable replication:

a) On either server, run **Administrative Tools > Distributed File System** if it is not already running.

b) Select the newly-created link and select **Action > New Target...**



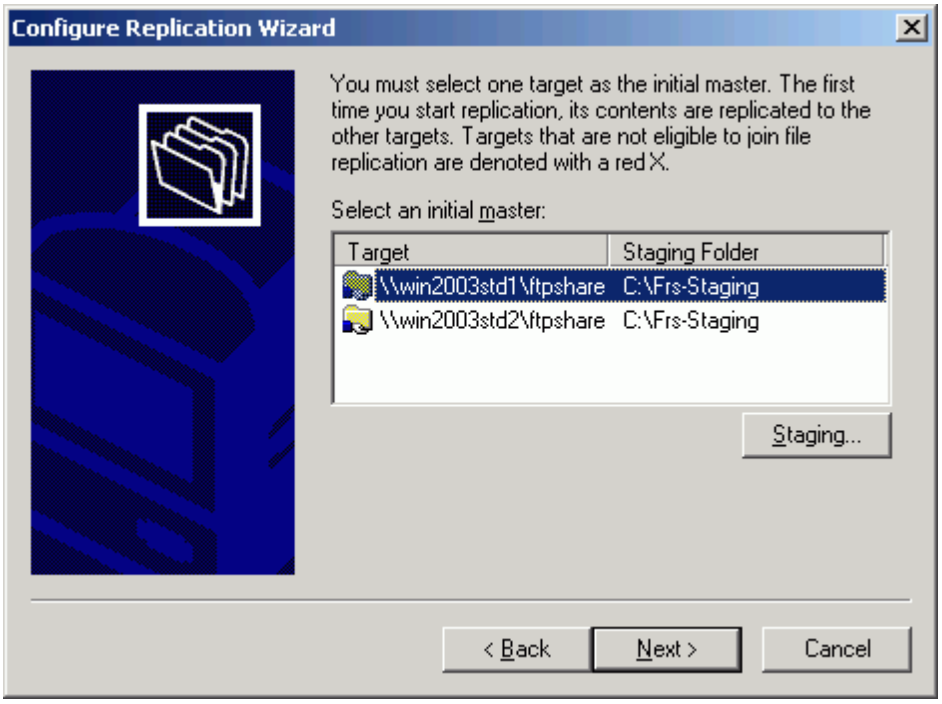
c) In the New Target dialog, enter the name of the other FTP share created above. Leave the "Add this target to the replication set" box checked.



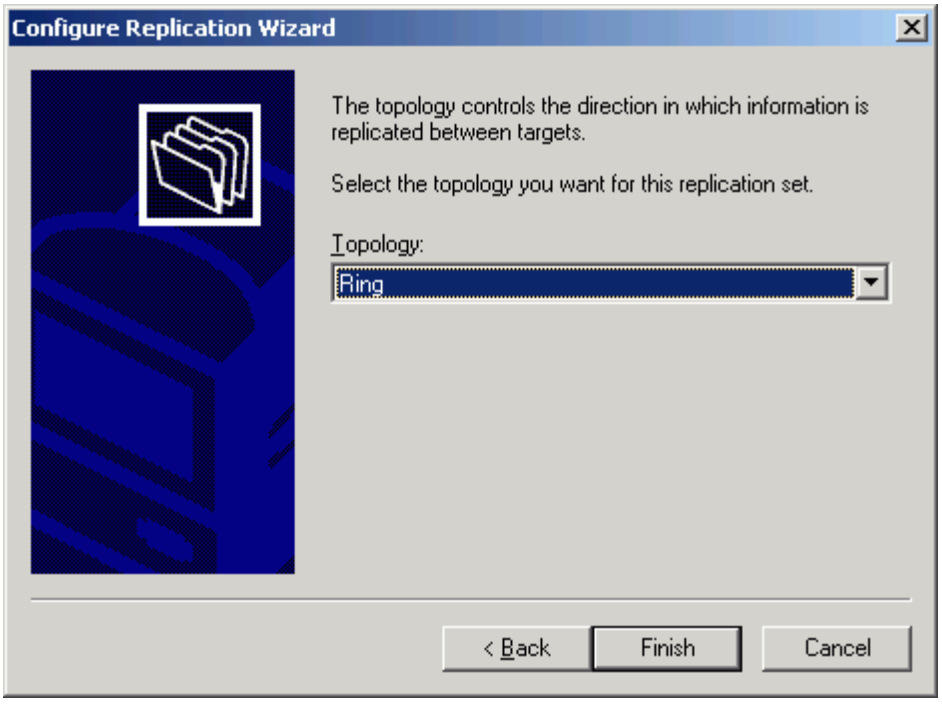
d) You receive the prompt: "The target cannot be replicated until replication is configured. Do you want to configure it now?" Click Yes.

e) At the Configure Replication Wizard welcome page, click Next.

f) You will be asked to select the initial master. If neither FTP server has received any files yet, it doesn't matter which one you select. Otherwise, select the FTP server which is more current. Then click Next.



g) For the replication topology, accept the default of Ring, and choose Finish.



6 Test file replication by sending a short file to one of the FTP servers. It should appear on the other server within seconds. You may have to stop and start the File Replication service on both computers to enable file replication.

After an outage, Windows may take a substantial amount of time--sometimes more than 10 minutes--to re-enable file replication.

You will probably also want to set up ***Network Load Balancing*** (see "***With NLB***" on page 394). See also ***Failover Overview*** (on page 373).

With NLB

Microsoft Network Load Balancing (NLB) can be used to share a single IP address between two servers. This allows incoming FTP sessions to connect to either of the two MOVEit Automation computers. If both are running, NLB assigns the incoming session to one of the computers, typically the one with higher priority. If one computer is down, NLB assigns the incoming session to the working computer.

Overview

NLB does not allow outbound connections from both computers to use the same IP address. Thus, you should ensure that either:

- Firewalls protecting hosts accessed by MOVEit Automation have rules allow incoming connections from either computer running MOVEit Automation

  – or –

- The firewall protecting MOVEit Automation uses Network Address Translation to make both computers look as if they have the same IP address

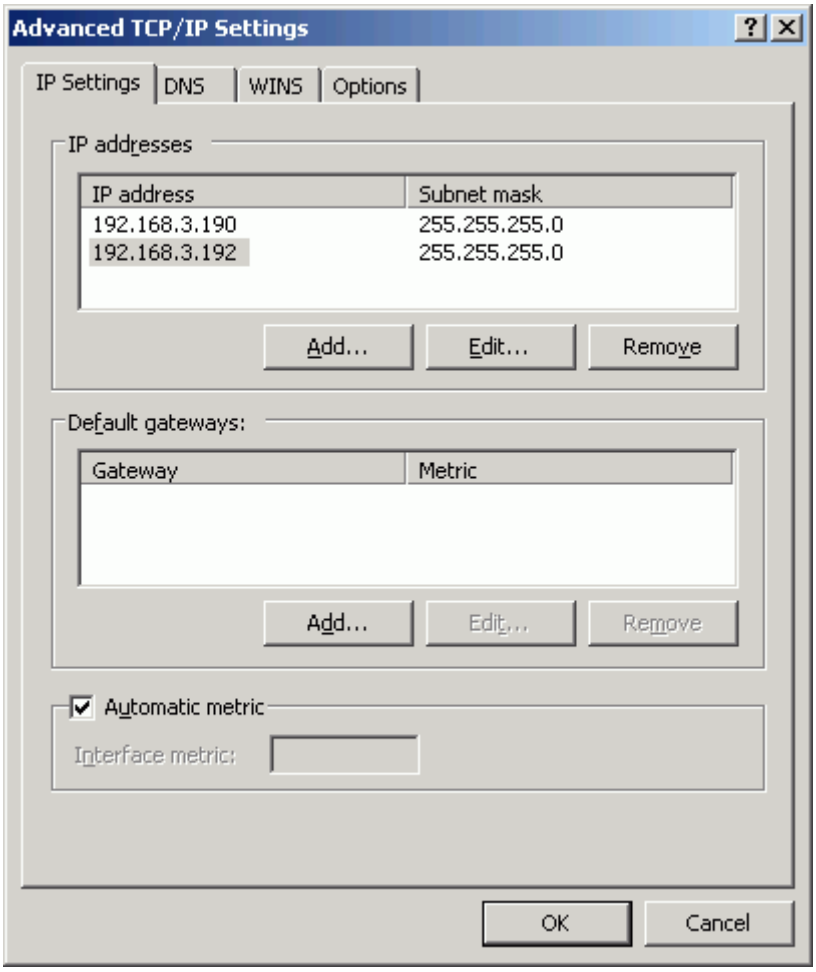
NLB is available on all editions of Windows 2008 and 2012.

Details

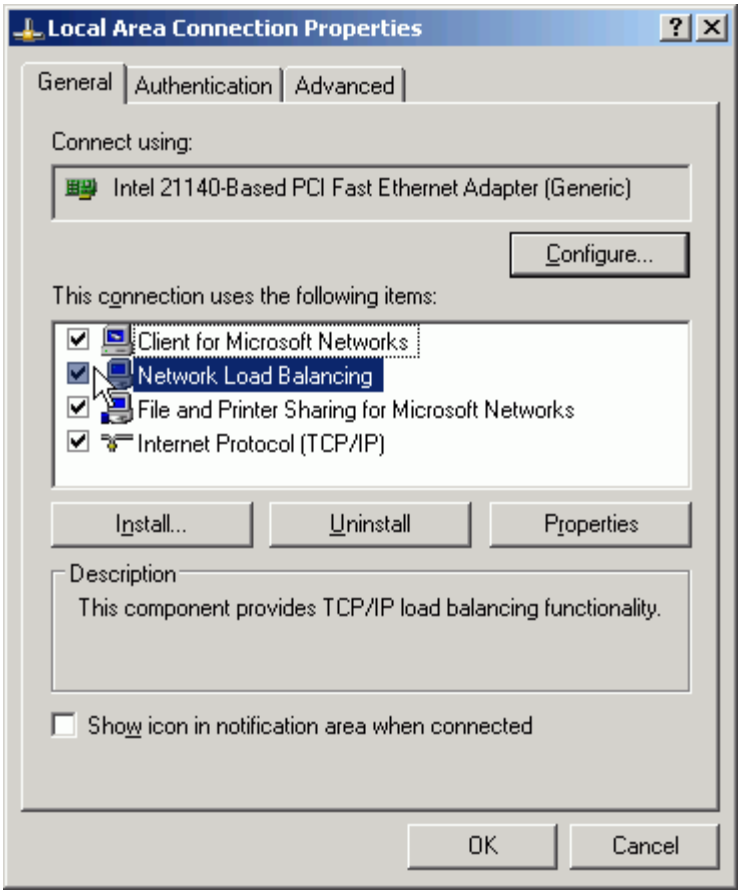
To use NLB, you take the following steps:

1. Decide on an IP address by which the cluster will be known. This should be different from the IP addresses of the individual MOVEit Automation computers in the cluster.
2. On one MOVEit Automation node, configure Windows Network Load Balancing as follows:

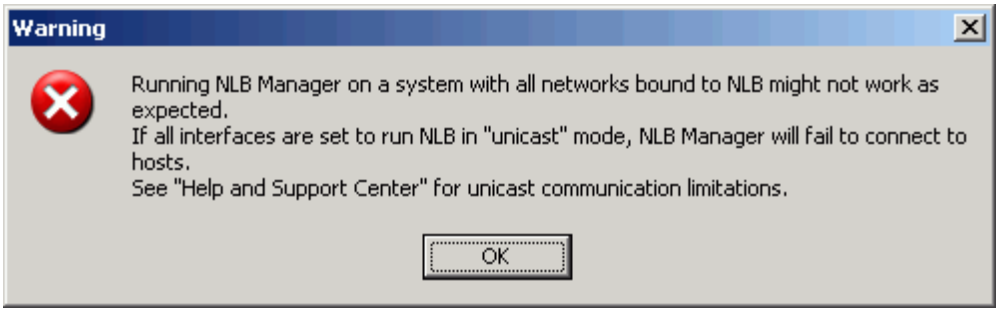
a) Use Network Connections | (connection name) | Properties | Internet Protocol (TCP/IP) | Properties | Advanced | Add... to add the cluster IP address to the list of addresses for your network adapter. If you have multiple adapters, use the one which will face the systems which you will be accessing.



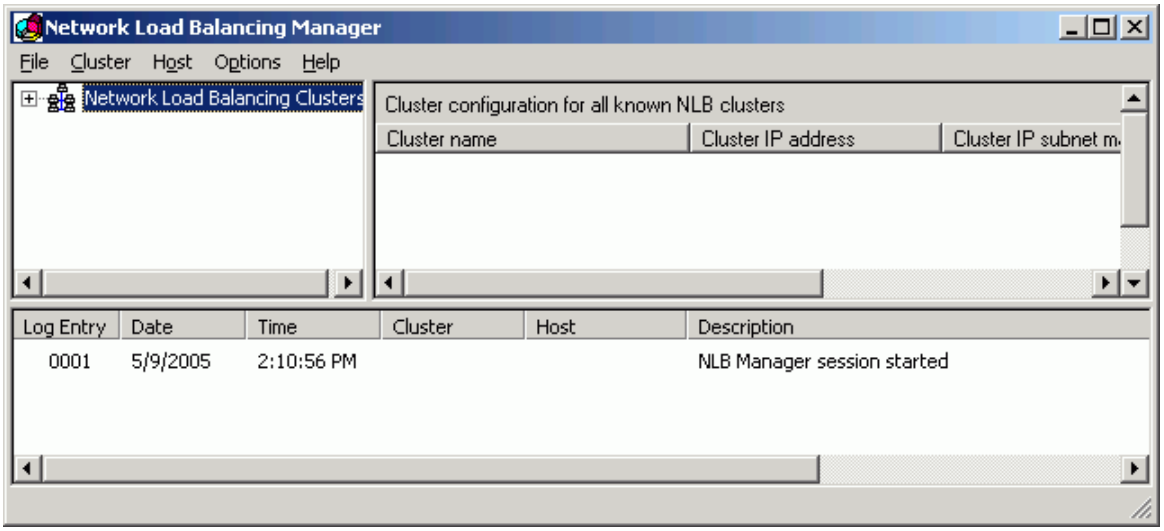
b) Enable Windows Network Load Balancing on each server from the Network Connections | Properties dialog.



c) Run the Network Load Balancing Manager from Administrative Tools. You can ignore the warning:

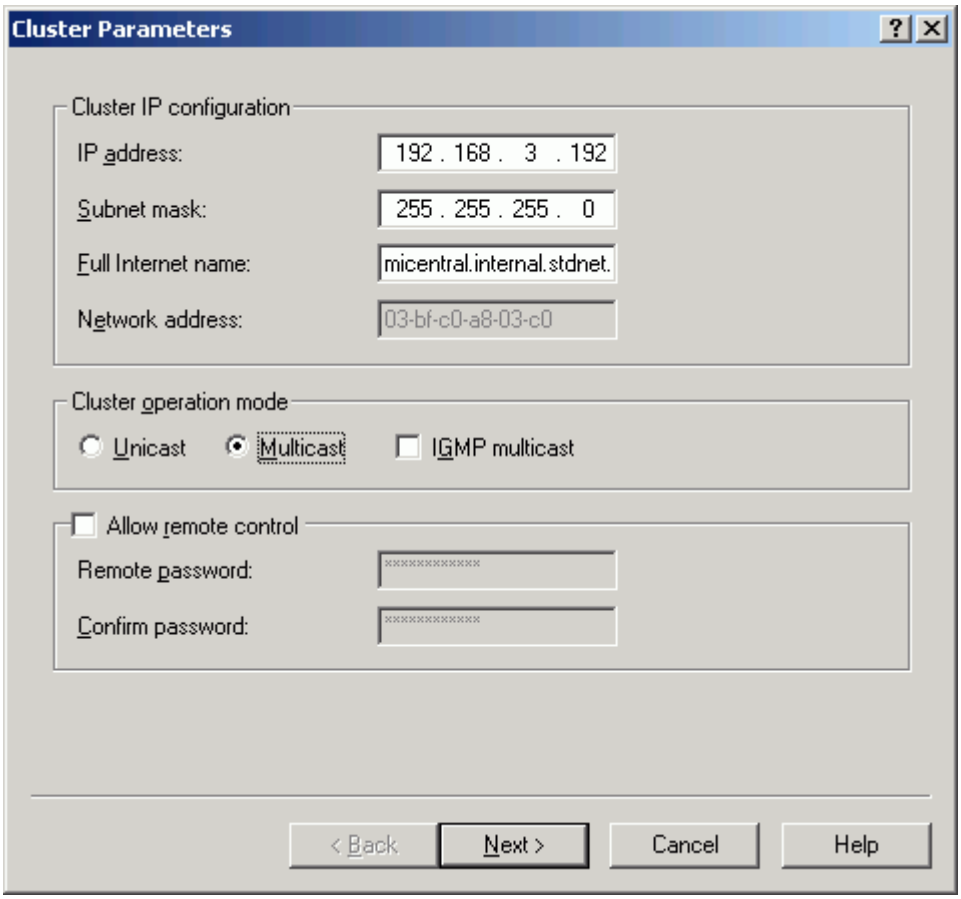


Then you'll get the main window:

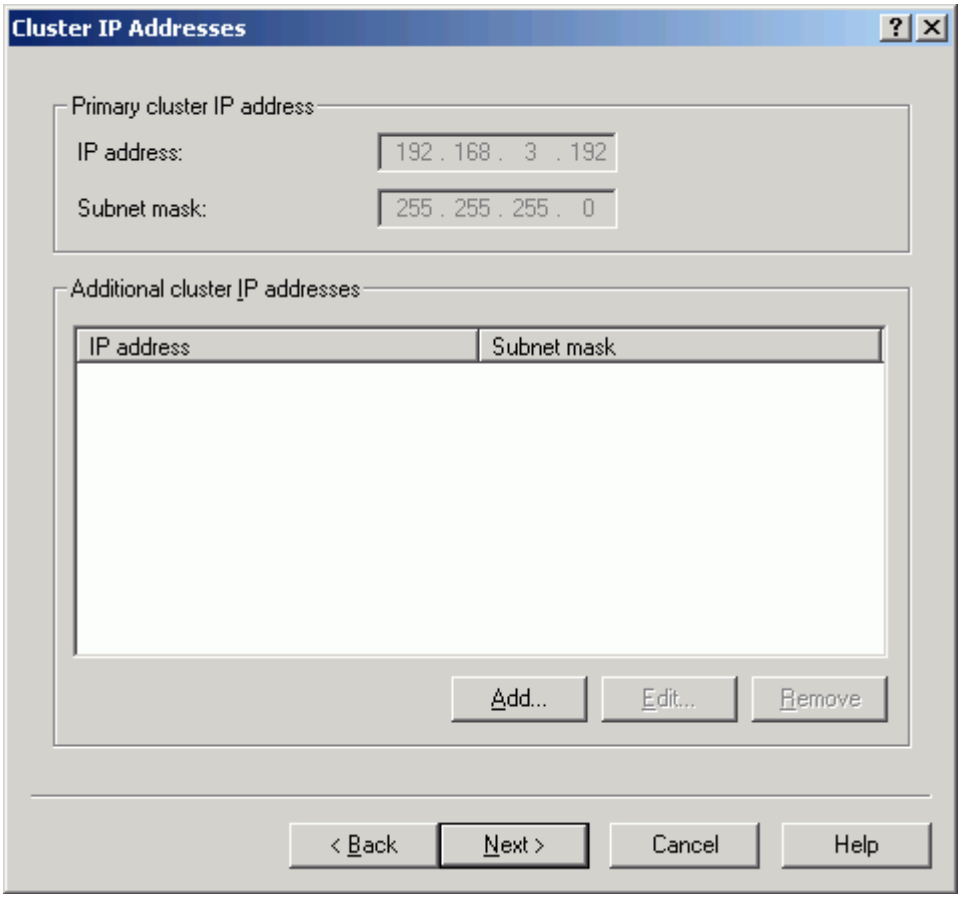


d) Choose **Cluster** | **New** to be prompted with the "Cluster Parameters" window.

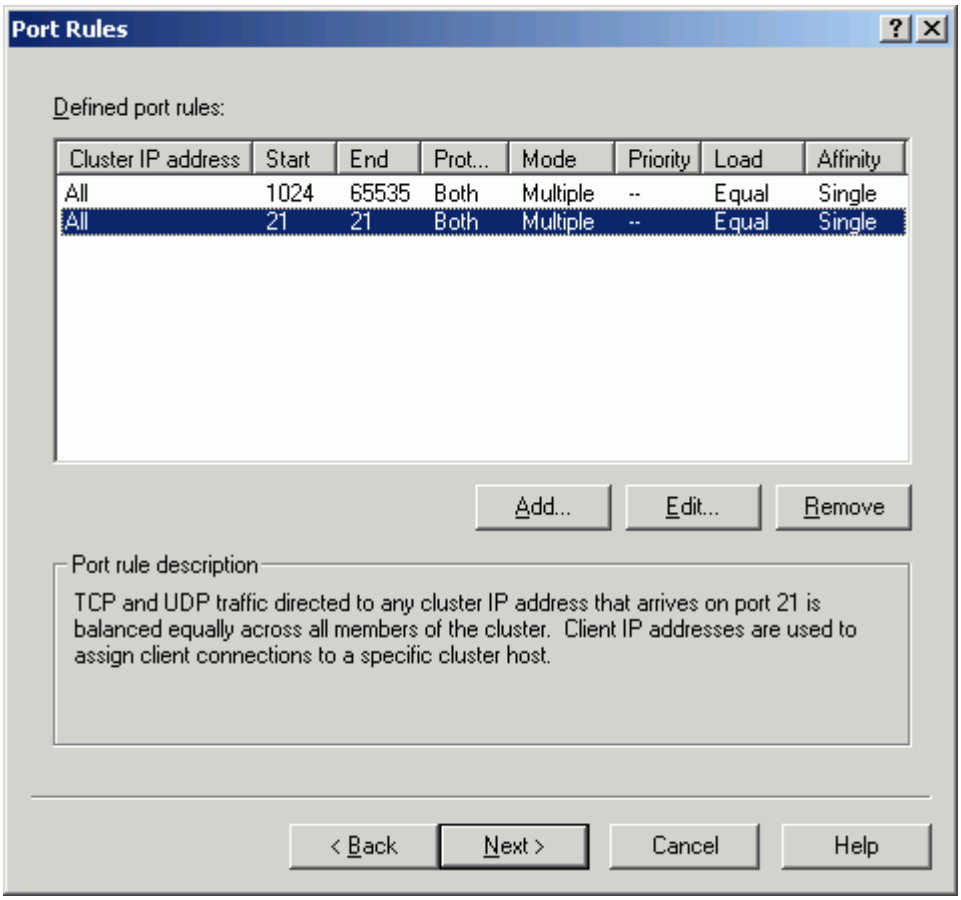
e) Enter the external IP address (configured in Advanced TCP/IP Settings above) which the MOVEit Automation computers will share and choose **Multicast** for operation mode. Enter the subnet mask and domain name. Then choose **Next**.



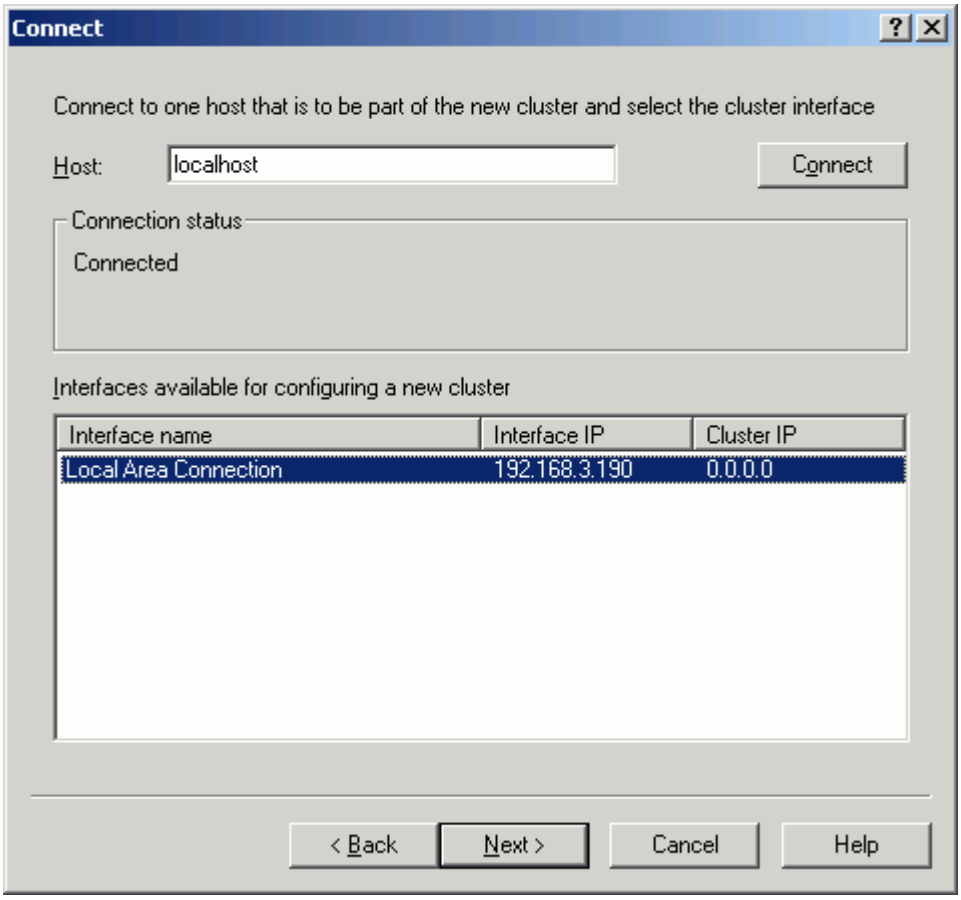
f) Do *not* enter any information into the Cluster IP Addresses dialog. Simply choose **Next**.



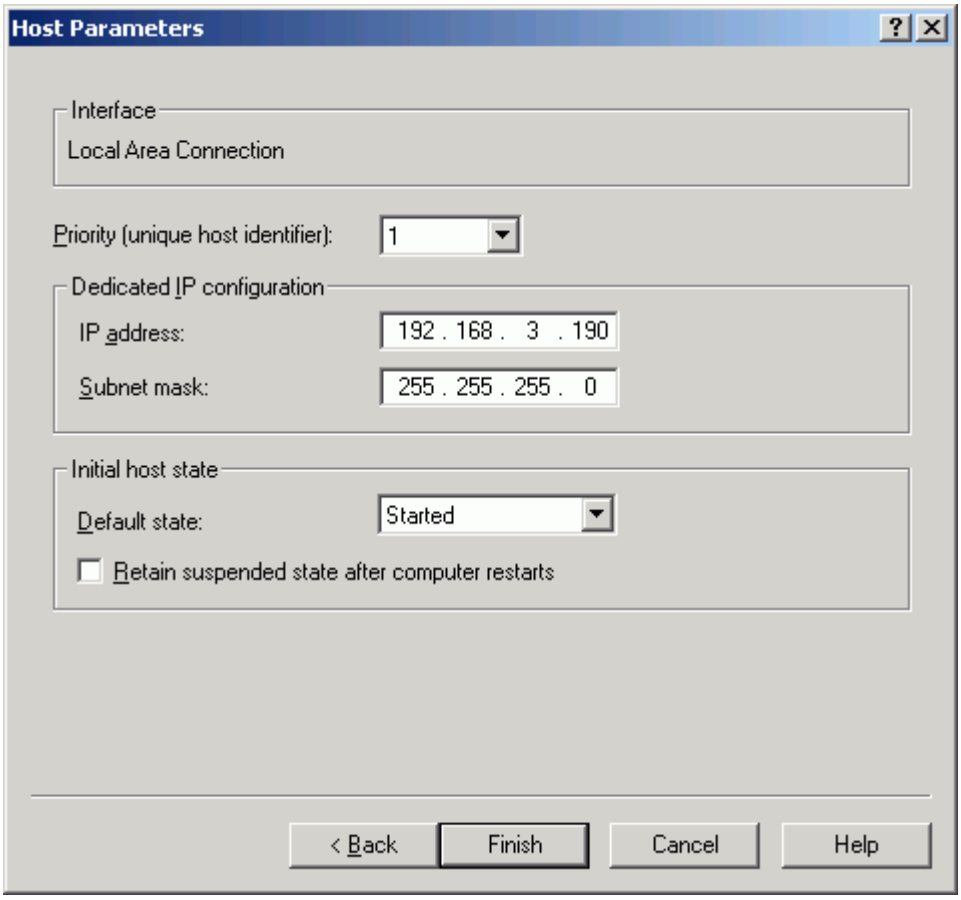
g) Configure your inbound port settings. In some situations, sticking with the default of "all ports" will be sufficient, but you will want to configure specific ports if you are using remote control software or a remote copy of MOVEit Automation Admin to access each node in the MOVEit Automation cluster. A more reliable setup is to only add the inbound ports needed by your FTP server. (For example: FTP control port 21 and the passive FTP data ports). Once you have the proper list, choose **Next**.



h) At the Connect dialog, enter **localhost** and choose **Connect**. This will cause a list of network interfaces to appear. Select the external network interface and choose **Next**. (In the example below, there is only one interface.)



i) In the Host Parameters dialog, set the dedicated IP address to the main IP address of this adapter. Set an appropriate subnet mask. Leave the other parameters at their default values. Choose Finish to complete the setup.

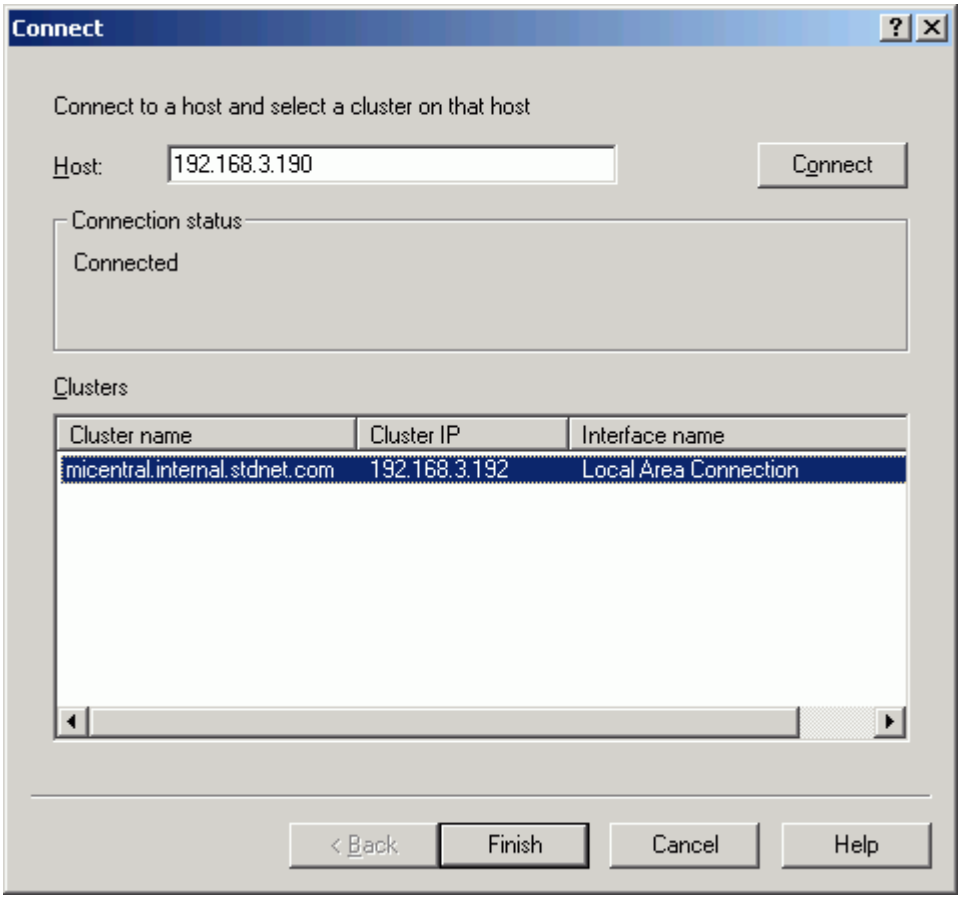


3 On the other MOVEit Automation box, perform the following steps:

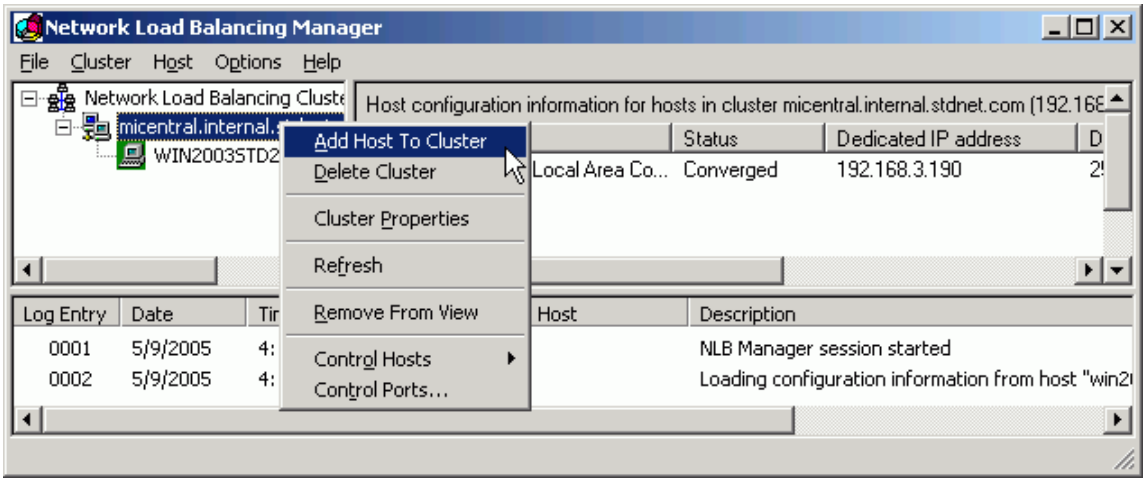
a) Enable Network Load Balancing, as above.

b) In Network Load Balancing Manager, use Cluster | Connect to Existing to bring up the Connect dialog.

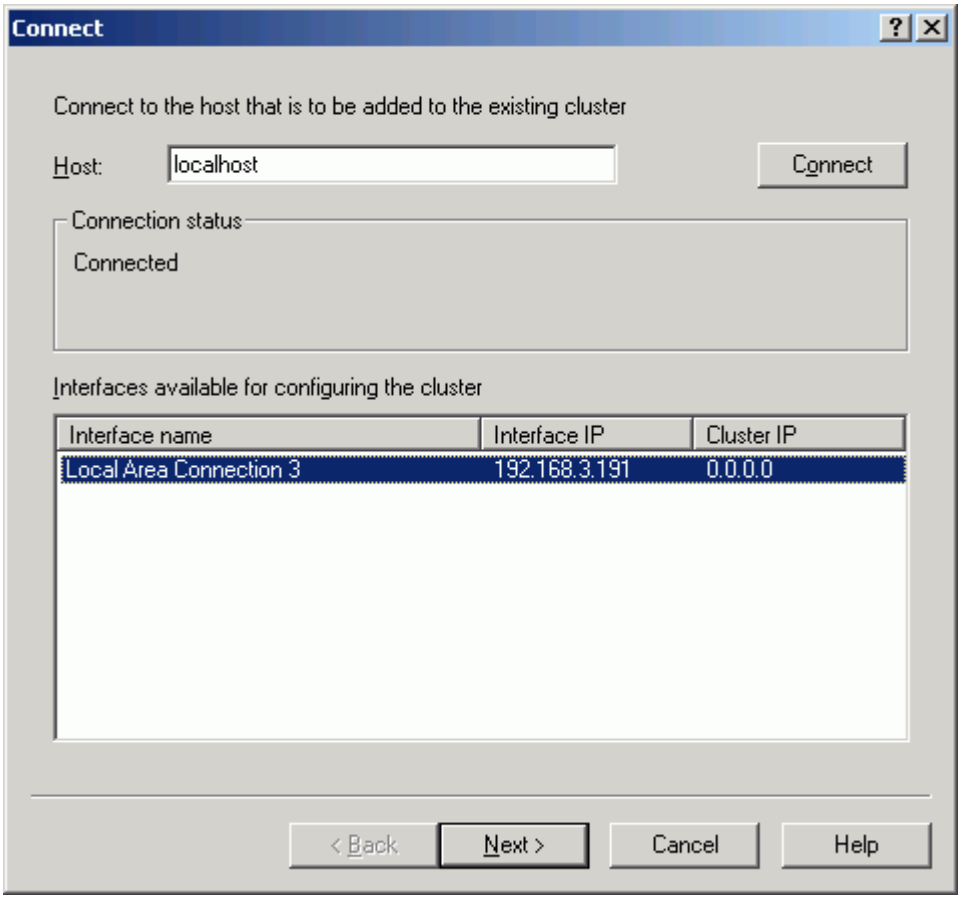
c) Enter the IP address of the other computer that was configured above, and choose **Connect**. Select the cluster name corresponding to the one you just configured on the other computer, and choose **Finish**.



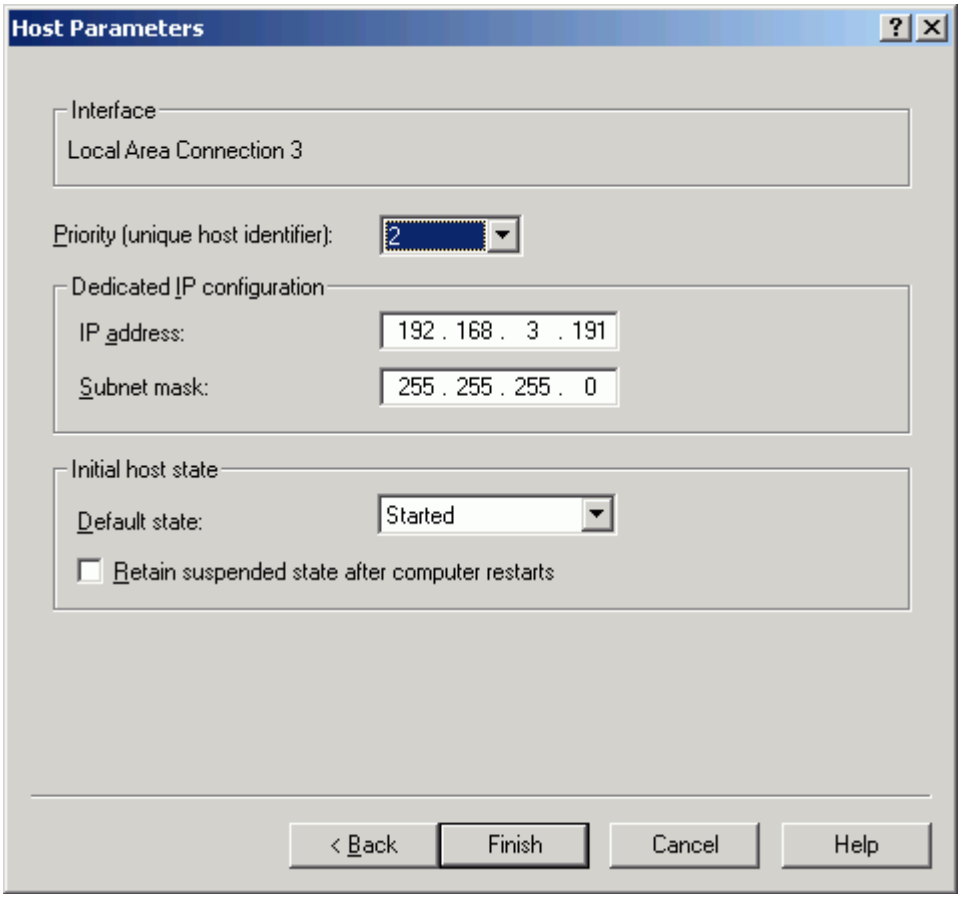
d) Right-click the name of the cluster, and choose **Add Host To Cluster**.



e) Enter "localhost" as the host name, and choose **Connect**. Select the name of the external interface, and choose **Next**.



f) Accept the defaults for the Host Parameters, and choose Finish.



g) Use Network Connections | (connection name) | Properties | Internet Protocol (TCP/IP) | Properties | Advanced | Add... to add the cluster IP address to the list of addresses for your network adapter.

4 You may need to reboot the systems to clear NLB errors due to temporary issues with recognizing the cluster IP address.

See also ***Failover Overview*** (on page 373).

Legal Information

Open Source Software used in MOVEit Automation Web Admin

Library	2017 Plus	License
angular (***https://angularjs.org/***)	1.5.11	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-animate (***https://github.com/angular/bower-angular-animate***)	1.5.11	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-bootstrap (***http://angular-ui.github.io/bootstrap/***)	2.5.0	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-cookies (***https://github.com/angular/bower-angular-cookies***)	1.5.11	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***)) (***https://github.com/angular/angular.js/blob/master/LICENSE***)
angular-growl-v2 (***http://janstevens.github.io/angular-growl-2/***)	0.7.5	***MIT (Jan Stevens)*** (***https://github.com/JanStevens/angular-growl-2/blob/master/LICENSE***)
angular-jquery-timepicker (***https://github.com/recras/angular-jquery-timepicker***)	1.12.0	***MIT (Recras)*** (***https://github.com/Recras/angular-jquery-timepicker/blob/master/LICENSE***)

Library	2017 Plus	License
angular-resource (***https://github.com/angular/bower-angular-resource***)	1.5.11	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-sanitize (***https://github.com/angular/bower-angular-sanitize***)	1.5.11	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-scroll-glue (***https://github.com/luegg/angularjs-scroll-glue***)	2.2.0	***MIT (Luegg)*** (***https://github.com/luegg/angularjs-scroll-glue***)
angular-translate (***https://github.com/angular-translate/angular-translate***)	2.15.1	***MIT (angularjs.org*** (***https://github.com/angular/angular.js/blob/master/LICENSE***))
angular-ui-router (***https://github.com/angular-ui/ui-router***)	0.4.2	***MIT (angular-ui)*** (***https://github.com/angular-ui/ui-utils/blob/master/LICENSE***)
angular-ui-grid http://ui-grid.info/	3.2.9	***MIT (angular-ui)*** (***https://github.com/angular-ui/ui-utils/blob/master/LICENSE***)
bootstrap (***http://getbootstrap.com/***)	3.3.7	***MIT (bootstrap)*** (***https://github.com/twbs/bootstrap/blob/master/LICENSE***)
es5-shim (***https://github.com/es-shims/es5-shim***)	4.5.9	***MIT (Kristopher Michael Kowal)*** (***https://github.com/es-shims/es5-shim/blob/master/LICENSE***)
font-awesome (***http://fontawesome.io/***)	4.7.0	***SIL OFL 1.1 (Fonts)*** (***http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=ofl***) ***MIT (Code)*** (***http://opensource.org/licenses/mit-license.html***)
jquery (***http://jquery.com/***)	3.2.1	***MIT (jquery foundation)*** (***https://jquery.org/license/***)
jquery-timepicker-jt (***https://github.com/jonthornton/jquery-timepicker***)	1.8.11	***MIT (Jon Thornton)*** (***https://github.com/jonthornton/jquery-timepicker***)
json3 (***https://bestiejs.github.io/json3/***)	3.3.2	***MIT (Kit Cambridge)*** (***http://kit.mit-license.org/***)

Library	2017 Plus	License
lodash (***http://lodash.com/***)	4.15.0	***MIT (The Dojo Foundation) https://raw.githubusercontent.com/lodash/lodash/4.5.1/LICENSE***
moment (***http://momentjs.com/***)	2.18.1	***MIT (Moment.js contributors)*** (***https://github.com/moment/moment/blob/develop/LICENSE***)
ngstorage (***https://github.com/gsklee/ngstorage***)	0.3.11	***MIT (Gias Kay Lee)*** (***https://github.com/gsklee/ngStorage/blob/master/LICENSE***)
Java Runtime Environment (JRE) (***http://www.oracle.com/technetwork/java/javase/overview/index.html***)	1.8.0_131	***Java Binary Code License*** (***http://www.oracle.com/technetwork/java/javase/terms/license/index.html***)
Tomcat (***http://tomcat.apache.org/***)	8.0.42	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
Spring Boot (***http://projects.spring.io/spring-boot/***)	1.5.2 RELEASE	***Apache License 2.0*** (***https://github.com/spring-projects/spring-boot/blob/master/LICENSE.txt***)
jackson-annotations (***https://github.com/fasterxml/jackson-annotations/***)	2.8.8	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
jackson-dataformat-xml (***https://github.com/fasterxml/jackson-dataformat-xml***)	2.8.8	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
jackson-module-jaxb-annotations (***https://github.com/fasterxml/jackson-module-jaxb-annotations***)	2.8.8	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
jackson-databind https://github.com/FasterXML/jackson-databind	2.8.8	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
jackson-core https://github.com/FasterXML/jackson-core	2.8.8	***Apache License 2.0*** (***http://www.apache.org/licenses/LICENSE-2.0***)
Saxon-HE (***http://www.saxonica.com/welcome/welcome.xml***)	9.7.0-18	***Mozilla Public License 2.0 https://www.mozilla.org/en-US/MPL/2.0/FAQ/***
json (***http://www.json.org/***)	20160810	***JSON License*** (***http://www.json.org/license.html***)

Library	2017 Plus	License
***jasypt* (*http://www.jasypt.org/*)**	1.9.2	***Apache License 2.0* (*http://www.jasypt.org/license.html*)**
***tomcat-embed-jasper* (*http://tomcat.apache.org/*)**	8.0.42	***Apache License 2.0* (*http://www.apache.org/licenses/LICENSE-2.0*)**
commons-io https://commons.apache.org/proper/commons-io/	2.5	***Apache License 2.0* (*http://www.apache.org/licenses/*)**
***hibernate-validator* (*http://hibernate.org/validator/*)**	54.1 Final	***Apache License 2.0 https://raw.githubusercontent.com/hibernate/hibernate-validator/master/license.txt***
***xercesImpl* (*http://xerces.apache.org/xerces2-j/*)**	2.11.0	***Apache License 2.0* (*http://www.apache.org/licenses/LICENSE-2.0*)**
***commons-lang3* (*http://commons.apache.org/proper/commons-lang/*)**	3.5	***Apache License 2.0* (*http://www.apache.org/licenses/*)**
spring-rest-exception-handler https://github.com/jirutka/spring-rest-exception-handler	1.2.0	***Apache License 2.0* (*http://www.apache.org/licenses/LICENSE-2.0*)**
ziplet https://github.com/ziplet/ziplet	2.2.0	***Apache License 2.0 https://github.com/ziplet/ziplet/blob/master/LICENSE***
rsql-parser https://github.com/jirutka/rsql-parser	2.1.0	***MIT* (*http://opensource.org/licenses/mit-license.html*)**

Software License

SOFTWARE LICENSE AND SUPPORT AGREEMENT APPLICABLE TO MESSAGEWAY AND MOVEIT SOFTWARE

This License and Support Agreement ("Agreement") is entered into as of today ("Effective Date"), by and between Ipswitch, Inc., with offices located at 83 Hartwell Avenue, Lexington, MA 02421 ("Licensor") and YOU ("Licensee").

WHEREAS, the parties hereto wish to provide the terms and conditions under which Licensor will supply Licensee Software (as defined below) for the term provided herein; and

WHEREAS, Licensee desires to obtain, and Licensor is willing to grant to Licensee a nonexclusive, royalty-free, perpetual, nontransferable license to use the Software subject to the terms and conditions set forth herein.

NOW, THEREFORE, in consideration of the foregoing, of the mutual covenants and undertakings contained herein and of other good and valuable consideration, the receipt and sufficiency of which are hereby acknowledged, the parties, intending to be legally bound, hereby agree as follows:

1. DEFINITION OF TERMS

1.1 "Software" means the Licensor’s standard, unmodified computer software programs in object code form for the MessageWay or MOVEit programs purchased by Licensee.

1.2 "Confidential Information" means any confidential information concerning the Software, Licensor’s and Licensee's business in general, all data pertaining to Licensor’s and Licensee's customers, and the terms and conditions of this Agreement.

1.3 "Total Fees" means the total sum for the Software, which includes license fees, first year support fee and any services.

1.4 "Taxes" means all sales, use, excise, value added, and other taxes and duties however designated levied by any taxing authority. Taxes shall not include any levies by any taxing authority based upon the net income of Licensor.

1.5 "Third Party" means any party other than Licensor or Licensee or their respective employees.

1.6. "Critical Problem" means a failure of the Software to perform in essential compliance with the material specifications set forth in its documentation, such failure being of the nature that Licensee is unable to utilize the Software for its operational purposes.

2. LICENSE

2.1 Licensor agrees to furnish the Software to Licensee and does hereby grant to Licensee a non-exclusive, royalty-free, non-transferable, perpetual license, without the right to sub-license, to use the Software, in its object code form only, on its premises, for the purpose of processing Licensee's own electronic file exchange with its customers (the "License").

2.2 For use as authorized, Licensee may copy reasonable quantities of any standard end user documentation; and may copy machine language code, in whole or in part, in reasonable quantities, in printed or electronic form, for use by Licensee for archive, back-up, disaster recovery, or emergency restart purposes, or to replace copies made on defective media. Licensee shall reproduce and include Licensor's proprietary rights and copyright notices on all such copies, in whole or in part, of the Software.

2.3 The License allows Software to be installed on one production server and also on one non-production server.

2.4 The License includes the unlimited right to distribute and use the Java Web browser plugins, Java and Windows command-line clients, the MOVEit Wizard ActiveX control, MOVEit Xfer, or other end-user components, as applicable.

2.5 Notwithstanding anything contained herein to the contrary, Licensee shall not allow any Software to be used on an external commercial (fee based) time-sharing basis or service bureau arrangement of any kind. As an exception to the preceding sentence, Licensee may use the Software to provide private cloud services to one (1) end user customer of Licensee specifically identified to Ipswitch, provided that Licensee completes and returns Schedule 1 to Ipswitch (available upon request from Ipswitch Sales Department) prior to providing such services.

2.6 Licensee assumes responsibility for selection of the Software to achieve Licensee's intended results and for the use and valid operation of the Software.

2.7 Licensee acknowledges that the Software (including any and all modifications, enhancements, or customizations thereof) consists of proprietary products of Licensor or its third party suppliers, and the proprietary rights that protect such property may include, but are not limited to, U.S. and international copyrights, trademarks, patents and trade secret laws of general applicability. All right, title and interest in and to the Software are and shall remain with Licensor or its third party suppliers, as applicable. This Agreement does not convey to Licensee any interest in or title to the Software, but only a limited right of use revocable in accordance with its terms.

2.8 Licensee shall not: decompile, disassemble, reverse engineer, extract, or otherwise produce any source code of the Software; disclose, divulge, communicate, or allow access to the Software to any person except Licensee's authorized agents, employees, or other parties expressly authorized hereunder, or as expressly permitted hereunder.

2.9 Licensee shall not isolate, extract, or otherwise utilize any components embedded in the Software for any purposes other than those supported by the core functions of the Software. Embedded Third Party components shall not be installed or configured, administered, customized, or directly accessed by way of component APIs independent from the APIs and functions of the Software. Embedded Third Party components shall not be independently upgraded or changed in any way except through officially released Ipswitch patches, updates or versions.

3. SUPPORT

3.1 Standard Support Coverage. If and for so long as Licensee purchases annual support, Licensor shall provide to Licensee unlimited telephone support and remote diagnostic assistance during Licensor's normal business hours. Licensor shall respond to support calls within one (1) hour of the initial call for such support by Licensee.

3.2. Extended Support Coverage. Extended Support Services ("Extended Support") provides Licensee with 24-hour, 7 day per week emergency assistance with guaranteed two (2) hour callback. Such service shall be restricted to Critical Problems. The Extended Support shall be available to Licensee either as specified and prepaid in Licensee's annual software support fee, or if not so specified, at the current fixed hourly rate for Extended Support provided by Licensor with a two (2) hour minimum charge per incident.

3.3 Licensor shall provide to Licensee at no charge Software updates and enhancements to licensed products when made available generally to Licensor's other customers, if and for so long as Licensee purchases annual support.

4. TERM

The term of this Agreement and the license grant shall begin on the Effective Date and continue until it is terminated under Section 10. Support may be renewed annually upon Licensee's payment of Licensor's then current fee for annual support for so long as Licensor offers support services.

5. PAYMENT

5.1 Licensee shall be responsible for and shall pay all applicable Taxes (including any interest and penalties thereon) if any, imposed by taxing authorities by reason of the sale and delivery of products herein provided. In no event will Licensee be obligated to pay taxes on Licensor's income.

5.2 Each payment to be made to Licensor under this Agreement shall be paid by Licensee.

6. WARRANTIES

6.1 Licensor warrants that the Software will perform in essential compliance with the material performance specifications set forth in its documentation for a period of one year following the Effective Date. In the event the Software does not so perform, Licensor shall resolve any such defect in a timely manner, or, at its option replace the defective portion thereof at no additional cost to Licensee, or refund the Software license fees paid, reduced by thirty-three per-cent (33%) per year from the Effective Date of this Agreement.

6.2 Licensor warrants that the services described in this Agreement shall be performed in a professional manner and with the standard of care and diligence in the industry, as well as industry standards of documentation, methodology, and control.

6.3 THERE ARE NO OTHER WARRANTIES EXPRESSED OR IMPLIED AND SELLER DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

7. INTELLECTUAL PROPERTY INDEMNITY

7.1 Licensor will defend Licensee and hold it harmless against any claim or action that alleges that the use of the Software infringes a patent, copyright, trade secret, or other proprietary right of a Third Party (a "Claim"), and Licensor will pay resulting costs, damages, and reasonable attorney fees awarded, provided that: (i) Licensee notifies Licensor in writing within thirty (30) days after learning that the Claim has been brought or might be asserted; (ii) allows Licensor sole control of the defense and all related settlement negotiations; and (iii) provides Licensor with the information, authority, and all assistance reasonably requested by the Licensor to provide the aforementioned defense. Licensee shall have the right to be represented in any such Claim by its own counsel, at its own expense.

7.2 In addition to Licensor's obligations under Section 7.1, if as a result of any such Claim, Licensee is enjoined from using the Software, Licensor will, at its sole option and expense (i) procure for Licensee the right to continue to use the Software; or (ii) replace or modify the Software so that it becomes non-infringing, which replacement or modification must be functionally equivalent, so as to settle such claim, or (iii) refund the Software fees paid, reduced by thirty-three per-cent (33%) per year or portion from the Effective Date of this Agreement and refund the annual software support fees paid for the current period. The indemnity hereunder shall not apply if and to the extent that the Claim results from (i) a correction or modification of the Software not provided by Licensor; (ii) a failure to promptly install and utilize an update; or (iii) the combination of the Software with any items not provided by Licensor.

8. LIMITATION OF LIABILITY

8.1 TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL LICENSOR BE LIABLE TO LICENSEE FOR INDIRECT, INCIDENTAL, SPECIAL, ECONOMIC, EXEMPLARY OR CONSEQUENTIAL DAMAGES, WHETHER IN TORT OR IN CONTRACT, INCLUDING LOSS OF PROFITS ARISING OUT OF THE USE OF OR THE INABILITY TO USE IPSWITCH PRODUCTS OR SERVICES, INCLUDING, WITHOUT LIMITATION, DAMAGES OR COSTS RELATING TO THE LOSS OF PROFITS, BUSINESS, GOODWILL, DATA, OR COMPUTER PROGRAMS, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO LICENSEE. THE FOREGOING LIMITATION OF LIABILITY SHALL NOT LIMIT LICENSOR'S OBLIGATIONS TO INDEMNIFY LICENSEE FOR ANY CLAIMS FOR DAMAGES AGAINST LICENSEE FOR INFRINGEMENT ON INTELLECTUAL PROPERTY.

8.2 Notwithstanding anything contained in this Agreement to the contrary, Licensor's cumulative liability to Licensee or any party resulting from any claims, demands, or actions arising out of or relating to this Agreement shall not exceed the license fees paid to Licensor for the applicable Software.

9. NON-DISCLOSURE

9.1 Each party agrees to hold as confidential all Confidential Information received by such party ("Recipient") from the other party ("Disclosing Party"). All Confidential Information shall remain the property of Disclosing Party. Confidential Information will be returned to Disclosing Party at the termination of this Agreement.

9.2 Recipient will use the same care and discretion to avoid disclosure of Confidential Information as it uses with its own similar information that it does not wish disclosed, but in no event less than a reasonable standard of care for the industry and materials in question. Recipient may use Confidential Information only in the furtherance of the purposes of this Agreement. Recipient may disclose Confidential Information to (i) its employees and employees of its affiliates who have a need to know; and (ii) any other party with Disclosing Party's written consent. Recipient may disclose Confidential Information to the extent required by law. However, Recipient agrees to give Disclosing Party prompt notice and make a reasonable effort to obtain a protective order. The provisions of this paragraph survive any termination of this Agreement.

9.3 No obligation of confidentiality applies to any Confidential Information that Recipient (i) already possesses without obligation of confidentiality; (ii) develops independently; (iii) rightfully receives without obligation of confidentiality from a third party. No obligation of confidentiality applies to any Confidential Information that is, or becomes, publicly available without breach of this Agreement.

9.4 The terms of this Section 9 shall survive termination of this Agreement or any Schedules.

10. TERMINATION

10.1 Licensee may terminate this Agreement at any time by returning the Software, documentation, and all copies thereof to Licensor or by certifying their destruction. Licensee shall receive no refund of any fees or other amount on termination unless this Agreement is terminated under Section 6.1 or 7.2(iii) above.

10.2 Licensor may terminate this Agreement if (i) Licensee fails to pay any license or other fees or any part thereof, or (ii) Licensee breaches any material term or condition of this Agreement and does not remedy such breach within thirty (30) days after receiving written notice thereof. Licensor may terminate this Agreement immediately on written notice, if (a) Licensee copies, distributes or discloses the Software in violation of this Agreement or otherwise breaches its duty of confidentiality, or (b) bankruptcy or insolvency proceedings are instituted by or against Licensee, or a receiver is appointed, or if the Software in Licensee's possession is the object of attachment, sequestration or other comparable action, and any such proceeding or action is not vacated or terminated within sixty (60) days after commencement or filing. Upon any termination of this Agreement, Licensee shall (x) immediately cease all use of the Software, (y) return the Software, documentation, and all copies thereof to Licensor or certify their destruction, and (z) notify all third parties using the Software through Licensee to do the same.

10.3 Exercise of the right of termination afforded to either party in this Agreement shall not prejudice the legal rights or remedies either party may have against the other in respect of any breach of the terms of this Agreement.

10.4 Upon the termination of this Agreement for any reason, both parties shall return to the other as appropriate all Software and Confidential Information in the other's possession or, with the other's approval, destroy such information with certification by an officer.

11. NOTICES

Any notice required or permitted to be given hereunder shall be given by: (i) Registered or Certified Mail, Return Receipt Requested, postage prepaid; (ii) by confirmed facsimile; or (iii) by nationally recognized courier service to the other party at the addresses set forth above or to such other address as a party may designate in writing. All such notices shall be effective upon receipt.

12. GOVERNING LAW

This Agreement will be governed by the substantive laws of the Commonwealth of Massachusetts, without reference to provisions relating to conflict of laws.

13. EXPORT LAW

The Software may not be downloaded or otherwise exported or re-exported to any country subject to U.S. trade sanctions governing the Software, sanctioned countries including those restricted under License Exception ENC under Sections 740.17 (A) and (B)(3) of the Export Administration Regulations set forth by the United States Department of Commerce, Bureau of Industry and Security, or by citizens or residents of such countries except citizens who are lawful permanent residents of countries not subject to such sanctions, or by anyone on the U.S. Treasury Department's list of Specially Designated Nationals and Blocked Persons or the U.S. Commerce Department's Table of Denial Orders.

14. GENERAL

14.1 Licensor and Licensee expressly agree they are acting as independent contractors and under no circumstances shall any of the employees of one party be deemed the employees of the other for any purpose. This Agreement shall not be construed as authority for either party to act for the other party in any agency or other capacity, or to make commitments of any kind for the account of or on behalf of the other except as expressly authorized herein.

14.2 The section headings used herein are inserted only as a matter of convenience and for reference and shall not affect the construction or interpretation of this Agreement.

14.3 If any provision of this Agreement is held to be unenforceable or invalid, the other provisions shall continue in full force and effect.

14.4 The failure of either party to insist on strict performance of any of the provisions hereunder shall not be construed as the waiver of any subsequent default of a similar nature.

14.5 This instrument constitutes the complete and exclusive statement of the Agreement between the parties on the subject matter hereof unless superseded by a written agreement executed by specifically identified and duly authorized representatives of each party.

ZIP.exe and UNZIP.exe utility license:

This is version 2003-May-08 of the Info-ZIP copyright and license. The definitive version of this document should be available at ftp://ftp.info-zip.org/pub/infozip/license.html indefinitely. Copyright (c) 1990-2003 Info-ZIP. All rights reserved. For the purposes of this copyright and license, "Info-ZIP" is defined as the following set of individuals: Mark Adler, John Bush, Karl Davis, Harald Denker, Jean-Michel Dubois, Jean-loup Gailly, Hunter Goatley, Ian Gorman, Chris Herborth, Dirk Haase, Greg Hartwig, Robert Heath, Jonathan Hudson, Paul Kienitz, David Kirschbaum, Johnny Lee, Onno van der Linden, Igor Mandrichenko, Steve P. Miller, Sergio Monesi, Keith Owens, George Petrov, Greg Roelofs, Kai Uwe Rommel, Steve Salisbury, Dave Smith, Christian Spieler, Antoine Verheijen, Paul von Behren, Rich Wales, Mike White This software is provided "as is," without warranty of any kind, express or implied. In no event shall Info-ZIP or its contributors be held liable for any direct, indirect, incidental, special or consequential damages arising out of the use of or inability to use this software. Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. Redistributions of source code must retain the above copyright notice, definition, disclaimer, and this list of conditions. 2. Redistributions in binary form (compiled executables) must reproduce the above copyright notice, definition, disclaimer, and this list of conditions in documentation and/or other materials provided with the distribution. The sole exception to this condition is redistribution of a standard UnZipSFX binary (including SFXWiz) as part of a self-extracting archive; that is permitted without inclusion of this license, as long as the normal SFX banner has not been removed from the binary or disabled. 3. Altered versions--including, but not limited to, ports to new operating systems, existing ports with new graphical interfaces, and dynamic, shared, or static library versions--must be plainly marked as such and must not be misrepresented as being the original source. Such altered versions also must not be misrepresented as being Info-ZIP releases--including, but not limited to, labeling of the altered versions with the names "Info-ZIP" (or any variation thereof, including, but not limited to, different capitalizations), "Pocket UnZip," "WiZ" or "MacZip" without the explicit permission of Info-ZIP. Such altered versions are further prohibited from misrepresentative use of the Zip-Bugs or Info-ZIP e-mail addresses or of the Info-ZIP URL(s). 4. Info-ZIP retains the right to use the names "Info-ZIP," "Zip," "UnZip," "UnZipSFX," "WiZ," "Pocket UnZip," "Pocket Zip," and "MacZip" for its own source and binary releases.

Legal Information - Americans with Disabilities Act (ADA) Compliance

MICEN> is in full compliance with the Americans with Disabilities Act. Specifically, MOVEit Automation complies with Section 1194.21 "Software Applications and Operating Systems". (This specification is online at this URL: "***http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/guide-to-the-section-508-standards*** (***http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/guide-to-the-section-508-standards***)")

- (a) Executing Function from Keyboard - Keyboard access to all functions is provided.
- (b) Accessibility Features - Applications do not interfere with existing accessibility features.
- (c) Input Focus - Focus is always clearly marked/displayed.
- (d) User Interface Element - Text is also provided for all images.
- (e) Bitmap Images - Bitmap images are consistent throughout application.
- (f) Textual Information - Text display occurs via OS functions.
- (g) User Selected Attributes - Applications accept user-selected color and contrast settings.
- (h) Animation - N/A. (Not used.)
- (i) Color Coding - Colors are never only method of displaying information.
- (j) Color and Contrast Settings - N/A. (OS color and contrast settings are used.)
- (k) Flashing or Blinking Text - Elements do not flash.
- (l) Electronic Forms - Consistent text/label presentation is used.

Legal Information - Export Restrictions

MOVEit Automation makes use of MOVEit Crypto for its cryptographic services. Therefore, MOVEit Automation is subject to the same export restrictions as MOVEit Crypto; these are described below.

MOVEit Crypto has undergone extensive review by an independent testing laboratory accredited under the Cryptographic Module Validation (CMV) Program run by NIST and the Communications Security Establishment (CSE) of the Canadian Government. As a result, MOVEit Crypto has received the following certifications.

- FIPS 140-2 "Security Requirements for Cryptographic Modules" validation (Certificate 310).
- FIPS 197 "Advanced Encryption Standard (AES) Algorithm" validation (Certificate 30).

MOVEit Crypto (and therefore, MOVEit Automation) is subject to U.S. Department of Commerce export controls, which prohibit it from being downloaded or otherwise exported or re-exported to Cuba, Iran, Iraq, Libya, North Korea, Sudan, Syria, or to a national or resident of any of these countries.

About Ipswitch

Contact...

The Ipswitch Support Center is an information and diagnostic center available for Ipswitch customers to:

- Obtain advice on proper product installation, configuration, and operation
- Report any product problems and receive timely resolutions
- Request software updates
- Inquire on software release contents and status

The support center provides support for all Ipswitch licensed products according to the terms of your Ipswitch Support Agreement. These support services are provided to Ipswitch's direct customers and resellers, while indirect customers are serviced by their own respective reseller. Support for customizations to Ipswitch software is the responsibility of the customer or their reseller's Integration Services group. For more information on support for customized software please contact your reseller's Integration Services manager.

To access the Support Center, you can use the following links:

- Visit our ***Web site https://www.ipswitch.com/support/*** for the latest contact information.
- Visit the ***customer portal*** (***https://ipswitchft.secure.force.com/cp/***).