Migration Manager for PSTs 1.2.1

User Guide

Dell Migration Manager for PSTs User Guide

Updated - February 2016
Software Version - 1.2.1

Dell Inc.
Attn: LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656

Refer to our web site (software.dell.com) for regional and international office information.

Legend

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

IMPORTANT NOTE, **NOTE**, **TIP**, **MOBILE**, or **VIDEO:** An information icon indicates supporting information.

Contents

1

Product Overview

Product components

Migration Manager for PSTs (MMP) lets you migrate PST files from computers within an Active Directory domain to Exchange 2016, Exchange 2013, Exchange 2010, or Microsoft Office 365 Exchange. This can be accomplished through the Migration Manager for PSTs web GUI, or the PowerShell command-line interface. Migration Manager for PSTs contains the following components.

- **Configuration Console:** This Windows utility allows you to configure the SQL connection website settings, agent settings and logging settings for MMP. You need to run this on every MMP server on which you install an MMP component. You may also use the Configuration Console later to change these settings.
- **Web Service and Scheduling Agent:** The Web UI and the service layer providing data to the MMP Web UI, Agents, PowerShell Cmdlets, and the windows service that manages scheduled tasks for processing by other MMP agents.
- **Discovery agent:** The windows service that performs discovery of users and computers within Active Directory domains, as well as file scanning of those computers for PST files.
- **Migration agent:** The windows service that handles migration of PST file data into Microsoft Office 365 or Exchange.
- **PowerShell Cmdlets:** The PowerShell command line interface for MMP.
- **Dell Log Viewer:** The Dell Log Viewer simplifies the viewing and interpretation of program log files, which document events and warnings in Dell programs.

System requirements and credentials

To view a list of the supported versions of software, hardware, and credentials required to run the Migration Manager for PSTs, see the *Migration Manager for PSTs Quick Start Guide*.

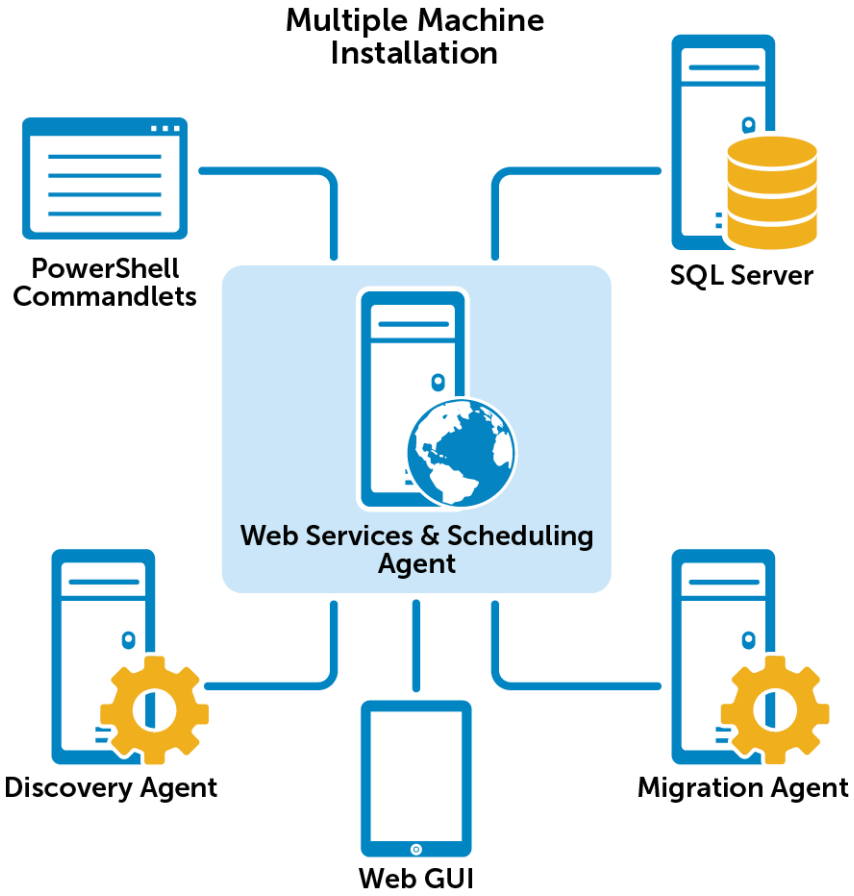
Advanced installation scenario

Instructions for a basic installation of Migration Manager for PSTs are located in the *Migration Manager for PSTs Quick Start Guide*. For many scenarios, all MMP components can be installed to a single server. In some cases, you may want to leverage existing IIS or SQL Server environments, or spread your agents among multiple computers in order to balance load or increase performance for large migration jobs. It is recommended that the **Web Services and Scheduling Agent** component be installed and is running prior to installing the other components. The other components can be installed but will not run until the Web Services and Scheduling Agent component is installed.

Instructions for installing Migration Manager for PSTs from the command line are described in the following section.

IMPORTANT: Installing more than one discovery is not recommended.

The migration scenario below assumes that you will install the Web Services and Scheduling Agent, the Discovery Agent, and Migration Agent on separate computers.



To install the Web Services and Scheduling Agent from the setup wizard:

1. Run the appropriate 32- or 64-bit MSI file and launch the setup wizard on the computer you have selected to use for the component(s) you are installing.
2. On the Welcome screen, click **Next** to start the setup process.
3. On the Software Transaction Agreement screen, read the license agreement and then select **I accept the terms of the license agreement**.
4. On the Custom Setup screen, select to install the **Web Services and Scheduling Agent** component. Disable the other components at this time.

5 On the Destination Folders screen, in the **Install Dell Migration Manager for PSTs to:** box, use the default directory **C:\Program Files\Dell\Migration Manager for PSTs\,** or click **Change** to search for a different directory.

6 On the Ready to install Dell Migration Manager for PSTs, click **Install** to begin the installation.

7 On the Installing Dell Migration Manager for PSTs screen, a status is displayed during the installation. Click **Cancel** to cancel the installation.

8 On the Completed the Dell Migration Manager for PSTs Setup Wizard screen, you have the option to launch the Configuration Console to configure MMP on your server. This is the default option. If you wish to finish configuring MMP on your server later, you can uncheck this option. Now click **Finish** to exit the Setup Wizard.

To install the Discovery Agent or Migration agent on different computers:

1 Repeat the process above used to install the Web Services and Scheduling Agent, excluding all other components but the Discovery Agent or Migration Agent.

2 Use the Configuration Console to configure the agents to use the Web Server that you configured when installing the Web Service and Scheduling Agent component.

Using multiple migration agents

Multiple migration agents can be installed to increase migration performance. Each agent must be on its own server. The migration load is distributed evenly among agent servers. Each migration agent must be configured through the Configuration Console.

While there is not a limit to the number of migration agents that can be added to your environment, the performance gained by adding agents is dependent upon multiple factors, including:

- Network throughput
    - MMP agents to client machines
    - MMP to Exchange or O365
- Exchange performance
- O365 throughput

If multiple migration agents are used with an O365 target, additional credentials need to be entered because connections to O365 may be limited to 10 per credential. See the *Office 365 target credentials* section of the *Migration Manager for PSTs Quick Start Guide*.

Migrating Locked PST Files

If a PST file is locked, MMP attempts to migrate it according to the following process:

- First, the MMP Migration Agent attempts to migrate PSTs based upon the Global Migration Agent Settings configured in the MMP Configuration Console.

  The default is 5 attempts with a 30 minute delay between each attempt.

- On the final attempt to migrate a PST, if an error indicates that the file is open in Outlook, the Migration Agent will run a remote executable using PAExec on the client computer and attempt to copy the file to a local temp directory on the client computer.

  This is an effort to reduce the time spent copying the PST while the file is in use.

- Next, the remote executable running on the client will attempt to copy the PST file to a temp directory on the MMP Migration Server. Finally, the MMP Migration Agent will perform the migration of the PST file from that location.

If the temp directory on the client computer is not accessible, the remote executable will attempt to copy the file directly to the MMP Migration Server temp directory. For example, this may happen if there is no local profile for the account under which the migration service is running.

Using the Configuration Console

After you have installed the MMP product components, you can use the Configuration Console to configure MMP. The Configuration Console is an MMP utility that lets you configure MMP with your SQL server and IIS server. You will need to run this at least once on your MMP server. You can run it again later if you need to make changes to your SQL and IIS connections.

You can launch the Configuration Console from the final screen of the installer, or from the Windows Start Menu. The first time you run the Configuration Console, it runs in a wizard mode where you must navigate through all of the configuration screens in order by clicking **Next** after each screen. After you have saved your configuration settings the first time, the Configuration Console runs in a free mode where you can navigate to any screen, configure your settings, and then click **Apply** on the Summary screen to save them.

If you run the Configuration Console again, it displays the previously saved settings.

Welcome

The Welcome screen lists information you need to gather prior to starting your configuration. Review the list and gather the necessary information.

MMP requires a valid DLV license for migration. If you have previously installed using an ASC license, the Welcome screen provides a link to http://license.quest.com/upgrade to obtain a new license.

Website

On the Website screen, enter the following web service connection information. If you are using SSL, make sure that your IIS server has been configured with the necessary certificates.

- **Website address:** The address for the Migration Manager for PSTs website. For example, **wspst01.sitraka.com**
- **Port:** The port for the Migration Manager for PSTs website. If the port number you enter is in use, it cannot be used for website installation. An error message prompts you to select a different port. For example, you may choose **443** if you will select SSL encryption, or **8080** if you want to use an alternate port.
- **Use SSL encryption (https) to connect:** This checkbox is selected by default. This option is recommended, though if you use SSL a valid certificate will need to be installed in IIS for secure communication.
- **Allow use of a self-signed certificate:** Allow use of a user-generated identify certificate.

Once you have entered the Web Site settings, the URL for your MMP website is displayed.

If you have selected the Use SSL option, then the URL to your MMP website will be in the format **https://<web server name>:<port>**. Using the examples above, the URL to the website would be **https://wspst01.sitraka.com:443** or just **https://wspst01.sitraka.com**.

If you have not selected the Use SSL option, then the URL to your MMP website will be in the format **http://<web server name>:<port>**. Using the examples above, the URL to the website would be **http://wspst01.sitraka.com:8080**.

If you wish to copy this URL to the Windows Clipboard, you can do so.

SQL Server

NOTE: If the MMP Web Server is installed on a separate machine from the one running Migration Manager for PSTs, the SQL Server screen will not be displayed in the Configuration Console on the Migration Manager for PSTs server. The SQL Server screen will be displayed when running the Configuration Console on the MMP Web Server machine.

On the SQL Server screen, enter the SQL Server Name and Authentication method for the Migration Manager for PSTs database. The user must have a valid login on the SQL Server and will be assigned correct roles to access the database. These SQL Server credentials do not require system administrator privileges.

- **Server name:** The name of the SQL server to be used for the connection.

  If you are using a local SQL Server Express installation, then your Server name is **.\SQLEXPRESS**. If you are connecting to the default instance of a regular SQL Server, then the name could be the fully qualified domain name of the SQL Server or the instance name that can be resolved from your MMP server to the SQL Server instance. For example, **SQLServer01** or **SQLServer01.sitraka.com**.

- **Authentication:** Select either SQL Server Authentication or Windows Authentication.

  Windows Authentication will use the Application Pool Identity credentials from the Migration Manager for PSTs web service.

- **Username:** Only used for SQL Server Authentication. The name of the user account that will be used to connect to the SQL server.

- **Password:** Only used for SQL Server Authentication. The password for the user account that will be used to connect to the SQL server.

Enter credentials that have sysadmin privileges in the section below. These credentials are used for creating, upgrading and assigning database roles during the configuration and are not saved.

- **Authentication:** Select either SQL Server Authentication or Windows Authentication.

Windows Authentication will use the credentials of the currently logged in user. This user must have a valid SQL login with sysadmin privileges on the SQL server.

To give a user SQL **dbcreate** and **sysadmin** privileges on the SQL server, complete the following steps:

1. Open SQL Management Studio.
2. Connect to the SQL Server instance hosting the MMP database.
3. Expand the Security folder in the tree view.
4. Expand the Logins folder in the tree view.
5. Select the user and open their properties.
6. Select the Server Roles option and verify that the **sysadmin** role is checked for the user account that will be used to connect to the SQL server.

NOTE: The user set in the AppPool must have a SQL login. A public role is sufficient. The system administrator credentials are used to add this user to the MMP database with db_owner privileges.

Web Server

Enter credentials for the web service application pool. These credentials will be used to access the database if **Windows Authentication (web service credentials)** was selected on the SQL Server screen.

NOTE: If the MMP Web Server is installed on a separate machine from the one running Migration Manager for PSTs, the Web Server screen will not be displayed in the Configuration Console on the Migration Manager for PSTs server. The Web Server screen will be displayed when running the Configuration Console on the MMP Web Server machine.

- **Username:** Name of user account that connects to the database if Windows Authentication is used.
- **Password:** Password for user account that connects to the database if Windows Authentication is used.

Agent

The Agent screen allows you to configure local agents and global migration settings. Enter the following information in the Local Agent Settings section:

- **Agent:** Discovery Agent, Migration Agent, or Schedule Agent.
- **Restore Defaults:** Restore the settings below to their defaults.
- **Poll time (seconds):** The frequency for the agents to look for tasks to run.
- **Task count:** The number of concurrent tasks the agent runs.

  This option is not displayed under the schedule agent.
- **Request Timeout for all agents (milliseconds):** The time before all agents time out.

Sometimes situations occur in which a PST file cannot be migrated at first. A common scenario that occurs is a locked PST file resulting from a user having a PST file open in Outlook. In this case, the PST file will be retried later.

Enter the following information in the Global Migration Agent Settings section:

- **Retry delay (minutes):** The minimum time to wait before the first attempt to retry the PST file migration. The wait time automatically doubles with each retry after the first until the number of retries set in the Retry attempts field is reached.
- **Retry attempts:** The number of retries to attempt before failing the migration.
- **Max Message Size:** Enter the maximum message size allowed for the migration. The default is 10MB.

If the PST file still fails to migrate, the owner of the failed PST file is placed into a failed PST migrations collection. See the Collections for failed PST migrations section.

License

MMP requires a valid DLV license for migration.

- **License File:** Browse to enter a valid license for migration.

If there is an existing license, adding a new license replaces the existing license. You can view the existing license by clicking the information icon at the top right-hand corner of the page and selecting the Licenses tab.

Logging

The Logging screen allows you to select a component and configure its logging settings. Only messages of the selected log level and higher will be logged.

- **Product Component:** From the drop-down list, select **Discovery Agent**, **Migration Agent**, **Schedule Agent**, **Web Service**, or **Windows PowerShell**.
- **Restore Defaults:** This button sets the currently-selected component's log settings to the default settings.
- **Apply to All:** This button applies the settings of the currently-selected component to the rest of the components.

In the Migration Agent Log Settings box, enter the following settings for the selected agent:

- **Log Level:** Select the desired log level from the drop-down list. Log levels include:

All	Info	Error
Verbose	Notice	Fatal
Debug	Warning	Off

- **Maximum size of each log file (MB):** Enter the maximum size for each log file. The default is 25 MB.

- **Number of files to keep:** The maximum number of log files. The default is 10.
- **Error Context Logging:** Select this checkbox to log the context of previous messages leading up to an error, including debug-level messages.

Summary screen

The Summary screen lists the configuration changes that will be made. Click **Finish** to save and apply your configuration settings if this is the first time you have run the Configuration Console and it is operating in wizard mode. Click **Apply** to save and apply your changes if you have run the Configuration Console before.

If there is a problem with your configuration settings, a warning will be displayed. On the Website tab of the Configuration Console, make sure that **Allow use of a self-signed certificate** is selected.

Command line setup

Product components can be installed all at once or individually from the command line. If you want to launch the setup Wizard from a command line, or to perform a silent install in a Windows Shell, you will use the **msiexec.exe** command to install the Migration Manager for PSTs components from the appropriate 32- or 64-bit MSI file. The possible command-line arguments for installing Migration Manager for PSTs are:

- **/I <path to MSI file>** Use this argument to set the file path to the appropriate MSI file for your 32- or 64-bit OS. For example, `/I "E:\Setup\DellMMPInstaller (x64).msi"`
- **/q** Optional argument to run quietly, without showing the setup Wizard GUI. Required for a silent install.
- **/r** Optional argument to run in reduced GUI mode, to see the progress bar, but with no user interaction required.
- **LICENSEACCEPTED=1** Optional argument to signify that you accept the product license agreement. Required for a successful silent install.
- **ADDLOCAL=<feature name>** Optional argument. If not included, all features will be installed. To include a subset of features, then include one ADDLOCAL argument for each of the features you wish to install.
    - Migration Manager for PSTs Web Service and Scheduling Agent:
      ADDLOCAL=WebFeature
    - Migration Manager for PSTs PowerShell Cmdlets:
      ADDLOCAL=PowershellFeature
    - Migration Manager for PSTs Discovery Agent Service:
      ADDLOCAL=DiscoveryAgentServiceFeature
    - Migration Manager for PSTs Migration Agent Service:
      ADDLOCAL=MigrationAgentServiceFeature
    - Dell Log Viewer:
      ADDLOCAL=DellLogViewerFeature

For example, to perform a silent install of all of the components, given the following sample data, issue the command line as shown below.

```
msiexec.exe /I "E:\Setup\DellMMPInstaller (x64).msi" LICENSEACCEPTED=1
ADDLOCAL=WebFeature ADDLOCAL=PowerShellFeature
ADDLOCAL=DiscoveryAgentServiceFeature ADDLOCAL=MigrationAgentServiceFeature
ADDLOCAL=DellLogViewerFeature /q
```

This installs all MMP components on your server.

Command line configuration

When running the configuration cmdlets for the first time after installing Migration Manager for PSTs, they must be run in the order listed in the examples below. After you have configured the product for the first time, the configuration cmdlets can be run in any order. The MMP services must be restarted any time you configure the product using the configuration cmdlets.

The examples below use SSL. However, the cmdlets will not set the certificate on the binding. Go to IIS and add the certificate on the binding of 443. On the Website tab of the Configuration Console, make sure that **Allow use of a self-signed certificate** is selected.

Common SQL variables for the MMP database

The name and location of the SQL server. For example, **".\SQLEXPRESS" or " SQLServer01.sitraka.com".**

```
$SqlServer = 'SQLServerName'
```

The SQL server administrative credentials used to set rights for the SqlUser on the MMP database. For example, **"sqladmin".**

```
$SqlAdmin = 'sqladmin'
```

The password for the SQL administrator.

```
$AdminPwd = "adminPassword"
```

Converts the administrator password into a secure string.

```
$SqlAdminPwd = ConvertTo-SecureString $AdminPwd-AsPlainText -Force # The password
for the Sql Admin
```

Converts the administrator account name and the secure string password into a PowerShell credential object.

```
$SqlAdminCreds = New-Object System.Management.Automation.PSCredential ($SqlAdmin,
$SqlAdminPwd)
```

A valid SQL user that will be given rights on the MMP database during setup. For example, **"mmpDbUser".**

```
$SqlUser = 'mmpDbUser'
```

The password for the SQL user.

```
$UserPwd = "userPassword"
```

Converts the user password into a secure string.

```
$SqlUserPwd = ConvertTo-SecureString $UserPwd-AsPlainText -Force # The password for
the Sql User
```

Converts the user account name and the secure string password into a PowerShell credential object.

```
$SqlUserCreds = New-Object System.Management.Automation.PSCredential ($SqlUser,
$SqlUserPwd)
```

Common IIS variables for the MMP website

The computer name and domain (or IP address) of the server to host the MMP Website. This should match the certificate if using https. For example, **wspst01.sitraka.com**.

```
$Website = "computer.domain.com"
```

The port number to configure the MMP Website. For example, **443**.

```
$Port = '443'
```

The protocol to be used by the MMP Website, either **http** or **https**.

```
$Protocol = 'Https'
```

The address of the MMP Website. For example, **http://www.sitraka.com:443**

```
$MMPWebsite = "$Protocol"+":\\"+"$Website"+":$Port"
```

Configuring MMP with SQL authentication

To configure MMP using SQL Authentication, use the following example. For information on each of the cmdlets used in this example, see the PowerShell Cmdlets chapter of this guide.

Import the MMP modules for use by the MMP cmdlets.

```
Import-Module PstMigratorModules
```

Configure MMP.

```
Install-MMPConfigFile
Install-MMPDatabase -SqlServer $SqlServer -SqlAuthenticationCredentials
$SqlAdminCreds
Set-MMPDatabaseRole -SqlLogin $SqlUser -SqlAuthenticationCredentials $SqlAdminCreds
-SqlServer $SqlServer
Set-MMPWebsite -ServerName $Website -Port $Port -Protocol $Protocol
Set-MMPSqlConnection -SqlServer $SqlServer -UseSqlAuthentication -
SqlAuthenticationCredentials $SqlUserCreds
Restart-MMPServices
```

Lists the address given to the MMP Website.

```
$MMPWebsite
```

Common variables for the MMP App Pool (Windows Authentication)

This user must be a member of the local administrators and have Sql access.

```
$AppPoolUser = 'domain\username'
```

Converts the password into a secure string.

```
$AppPoolPwd = ConvertTo-SecureString 'Password' -AsPlainText -Force
```

Converts the user account name and the secure string password into a PowerShell credential object.

```
$AppPoolCreds = New-Object System.Management.Automation.PSCredential ($AppPoolUser,
$AppPoolPwd)
```

Configuring MMP with Windows Authentication

To configure MMP using Windows Authentication, use the following example. For information on each of the cmdlets used in this example, see the PowerShell Cmdlets chapter of this guide.

Import the MMP modules for use by the MMP cmdlets.

```
Import-Module PstMigratorModules
```

Configure MMP.

```
Install-MMPConfigFile
Install-MMPDatabase -SqlServer $SqlServer
Set-MMPDatabaseRole -SqlLogin $AppPoolUser -SqlServer $SqlServer
Set-MMPWebsite -ServerName $Website -Port $Port -Protocol $Protocol
Set-MMPWebsiteAppPoolCredentials -Credentials $AppPoolCreds
Set-MMPSqlConnection -SqlServer $SqlServer
Restart-MMPServices # Restarts all the MMP services
```

Lists the address given to the MMP Website.

```
$MMPWebsite
```

Upgrading from a previous version

1 To configure your upgrade using PowerShell cmdlets, follow the full example scripts listed in the *Command Line Setup* section above.

2 Run the **Repair-MMPConfigurations cmdlet** to check and update the configuration file for the web server from previous versions, as described below.

3 Run the configuration cmdlets, as described in Command Line Configuration section above.

Complete the following steps to set the configuration logs to their default settings after an upgrade from 1.0.3.18 or a prior version only:

- Use the **Get-MMPLogConfig cmdlet**, **Set-MMPLogConfig**, and the example that follows to set the log configuration files to their default settings, as described below.

Repair-MMPConfigurations

Use the Repair-MMPConfigurations cmdlet to check and update the configuration file for the webserver from previous versions. This cmdlet displays an error if run on a machine without the web server installed.

Syntax

```
Repair-MMPConfigurations  [<CommonParameters>]
```

Example

To check and update the configuration file for the webserver from previous versions, use the following example:

```
Repair-MMPConfigurations
```

Get-MMPLogConfig

Use the Get-MMPLogConfig cmdlet to return log configuration file for the specified component.

Syntax

```
Get-MMPLogConfig -ConfigFileLocation <string> [<CommonParameters>]
Get-MMPLogConfig -ComponentType <ProductComponent> [-PipelineVariable <String>]
```

Parameters

PARAMETER	DESCRIPTION
ConfigFileLocation	This parameter specifies the location of the log file to display.
ComponentType	If this parameter is specified, the cmdlet looks for the MMP installation and returns the contents of the log configuration file specified. This parameter can have one of the following values: DiscoveryAgent, MigrationAgent, ScheduleAgent, Webservice or WindowsPowershell.

Example

To return the log configuration file for the Discovery Agent, use the following example:

```
Get-MMPLogConfig -ComponentType DiscoveryAgent
```

The following is an example of the returned values from the Get-MMPLogConfig cmdlet for the Discovery Agent with default values:

```
Component: DiscoveryAgent
Location: C:\Program Files\Dell\Migration Manager for PSTs\Discovery Agent
Service\log4net.config
LogLevelThreshold: Info
FileSize: 25
FileCoun: 10
BufferEnabled: True
BufferMultiplier: 5
BufferSize: 50
Valid: True
```

Set-MMPLogConfig

Use the Set-MMPLogConfig cmdlet to specify the settings for a particular log configuration file.

Syntax

```
Set-MMPLogConfig -MmpLogConfig <ILog42NetConfig> [-LogLevelThreshold <LogLevel>] [-
FileSize <Int32>] [-FileCount <Int32>] [-PipelineVariable <String>]
[<CommonParameters>]
```

```
Set-MMPLogConfig -MmpLogConfig <ILog42NetConfig> [-BufferSize <Int32>] [-
BufferMultiplier <Int32>] [-LogLevelThreshold <LogLevel>] [-FileSize <Int32>] [-
FileCount <Int32>] [-BufferEnabled [<Boolean>]] [-PipelineVariable <String>]
[<CommonParameters>]
```

The following returned values are not parameters and cannot be changed:

- The **Component** returned is the **ComponentType** specified in the Get-MMPLogConfig cmdlet.
- The **Location** returned is the path to the component specified by the **ConfigFileLocation** or **ComponentType** specified in the Get-MMPLogConfig cmdlet.
- **Valid** indicates whether all portions of the log configuration file were successfully read. It is updated when the log file is read. After saving changes to the configuration file, the file is read again and the output is sent to the command line.

Parameters

PARAMETER	DESCRIPTION
MMPLogConfig	This mandatory parameter is provided by the output of the Get-MMPLogConfig parameter. It can be passed on the pipeline, as shown in the example below.
FileSize	The maximum size that can be written to the log file before the logger writes to a new log file.
FileCount	The number of log files that will be retained.
BufferEnabled	Indicates whether the MultithreadedBufferingForwardingAppender is enabled.
BufferMultiplier	The internal queue of messages set to the buffer size times the multiplier.
BufferSize	The number of lines that will be retained in the buffer before the lines are written to the log file.

Example

To set each logConfig to the default settings, use the following example:

```
$logConfigs = @()
$configs = "DiscoveryAgent", "MigrationAgent", "ScheduleAgent", "WebService",
"WindowsPowershell"
foreach($x in $configs)
{
$logConfigs += Get-MMPLogConfig -ComponentType $x
}
$logConfigs | Set-MMPLogConfig -

SetConfigToDefault
```

Performance considerations

When scaling up your installation, keep in mind that Microsoft imposes some limits on the number of connections that an Migration Manager for PSTs agent can make to non-server versions of Windows. In addition, customers migrating to O365 have limits on the number of connections to O365 that can be made simultaneously.

Limits for individual agents:

- The number of connections allowed to a Windows workstation varies depending upon the operating system. For example, the limit for Windows 7 is 20 concurrent TCP connections.
- Unlimited connections to each Windows Server where PSTs reside.

Limits shared by all agents:

- 10 concurrent connections to O365 (additional accounts may be configured to increase concurrent connections by 10 per credential to O365).
- Exchange has connection limits as well. These can be configured by the Exchange administrator.

Changing the SQL server connection string

After installation, you many need to change the SQL server connection string for MMP. Earlier versions of this document included instructions for manual changes to change the connection string. You should now use the Configuration Console or the configuration cmdlets if you need to update the SQL Server connection settings.

2

The Migration Web UI

Home

The Migration Web UI Home page contains the following links for performing the steps of a migration. These links are explained in the sections below. The URL to the MMP website is defined by the choices you made during setup, with specific examples given in the previous section, such as **https://wspst01.sitraka.com**.

Home
Sources
Discover
    Users
    Computers
    PSTs
Collections
Targets
Migrate
Events

Home

The information icon at the top right-hand corner of the home page opens the About box. This box contains the About, Licenses, and Contact tabs.

About

The About contains information about the product version, and other legal information such as trademarks and the copyright.

Licenses

The licenses tab displays information about your migration license, including:

- **License:** The name of the product license.
- **Type:** The type of license purchased. For example, Beta, Term, etc.
- **Expires:** The expiration date for the license.
- **Seats licensed:** The number of seats purchased under the license agreement.
- **Seats used:** The number of seats used under the license agreement.

Licenses can be added or replaced in the Configuration Console. If there is an existing license, adding a new license replaces the existing license.

Contact

The Contact tab displays information for contacting Dell about this product.

Viewing help topics

Each web page displays a help menu that you can click to access help.

- **Help Topics:** This link takes you to Epic, an online system where you can view your product documentation. It provides access to documentation from anywhere, on any internet-enabled device.
- **Search Support Portal and Knowledge Base:** This link takes you to the support Knowledge Base. You can search the Knowledge Base by product or key word. The support portal also provides links to other useful information.
- **Visit the Community:** Communities provide information about specific products such as documents or discussions on particular topics.
- **Check for Updates:** This link takes you to a support page where you can download software updates.

Sources

The Sources link displays connections used to connect to Active Directory. By default, source connections are sorted alphabetically by name. You can change the sort order by clicking the column headings.

New Source

The New Source button contains the following fields:

- **Source name:** The source name for the connection.
- **Admin name:** The name of the administrator account.
- **Password:** The password for the administrator account.

Hiding and unhiding source connections.

Once a source connection is created it cannot be deleted. The source connections can be hidden and unhidden from view as needed.

To hide source connections:

- Click the **Hide** icon () on a source connection to hide it from view.

To view hidden source connections:

- Select the **View Hidden** checkbox above the source connection list.

To unhide hidden source connections:

1. Select the **View Hidden** checkbox above the source connection list to view the hidden source connections.
2. Click the **Unhide** icon () on the hidden source connections you want to unhide.

To view source connections after they are unhidden:

- Deselect the **View Hidden** checkbox to display the unhidden source connections.

Discover

The Discover link is used to create a new discovery task or manage existing discovery tasks. By default, discovery tasks are sorted by last run time. You can change the sort order by clicking the column headings. You can filter the results displayed in the Name, Status, and Last Ran columns.

Discovery results can be exported to a CSV file. This allows you to work with the results in a spreadsheet and create adhoc reports. While not all fields collected from the Active Directory scan are displayed in the Migration Web UI, every field that is collected from the Active Directory scan is output to the CSV file.

You can also export discovery results to a CSV file using PowerShell cmdlets.

The Discovery link contains the following buttons to create discovery tasks:

- New User Discovery
- New Computer Discovery
- New PST Discovery on Computers
- New PST Discovery on Network Shares

New User Discovery

The Settings tab contains the following fields:

- **Name:** The user discovery task name.
- **Source:** The source connection to use for the discovery task.

Advanced Discovery Filters

- **Containers:** Select containers to restrict the scope of discovery.

The Schedule tab contains the following fields:

- **Run Now:** Start the discovery now.
- **Run Once:** Start the discovery on the specified date and time.
- **Run Recurring:** Start the discovery on the specified days, time, and range of recurrence.

New Computer Discovery

The Settings tab contains the following fields:

- **Name:** The computer discovery task name.
- **Source:** The source connection to use for the discovery task.

Advanced Discovery Filters

- **Containers:** Select containers to restrict the scope of discovery.

The Schedule tab contains the following fields:

- **Run Now:** Start the discovery now.
- **Run Once:** Start the discovery on the specified date and time.
- **Run Recurring:** Start the discovery on the specified days, time, and range of recurrence.

New PST Discovery on Computers

The Settings tab contains the following fields:

- **Name:** The PST file discovery task name.
- **Source:** The source connection to use for the discovery task.

- **Relative path:** The comma-delimited list of paths to the PST files. Leave the field blank to search the entire computer for PST files. This can include drives.
- **Recurse:** Search all directories under the specified path(s) to the PST file.
- **Exclude system files:** Do not search system files.

The Computers tab contains the following fields:

- **Filter available computers:** Filter by name.
- **Filter by OU:** Filter by organization unit from the drop-down list.

The Schedule tab contains the following fields:

- **Run Now:** Start the discovery now.
- **Run Once:** Start the discovery on the specified date and time.
- **Run Recurring:** Start the discovery on the specified days, time, and range of recurrence.

New PST Discovery On Network Shares

The Settings tab contains the following fields:

- **Name:** The PST file discovery task name.
- **Source:** The source connection to use for the discovery task.
- **Relative path:** The comma-delimited list of paths from the network share root. Leave the field blank to search all relative paths.
- **Recurse:** Search all directories under the specified path to the PST file.
- **Exclude system files:** Do not search system files.

The Network Shares tab contains the following fields:

- **Filter available network shares:** Filter by name.

Discovered Items

The Discover link contains the following subpages:

- Users
- Computers
- PSTs

Users

Click this link to view a complete list of the discovered users. By default, users are sorted alphabetically by name. You can change the sort order by clicking the column headings. You can filter results displayed in the Name, Email, Mailbox Enabled, OU, and Discovered columns.

The following information is provided:

- **Name:** The name of the discovered user.
- **Email:** The target email address.
- **Mailbox Enabled:** Indicates whether the user is mailbox enabled.
- **OU:** The organizational unit of the discovered user.
- **Discovered:** The date and time stamp of the discovery.

MMP uses several attributes in Active Directory to determine if a user is mailbox enabled. MMP filters out non mailbox-enabled users from the Collection wizard in the website and in migration tasks. If any of the following Active Directory user attributes are set, MMP considers the user to be mailbox enabled:

- homeMTA
- homeMDB
- msExchHomeServerName
- msExchMailboxGuid

Computers

Click this link to view a complete list of the discovered computers. By default, computers are sorted alphabetically by name. You can change the sort order by clicking the column headings. You can filter the results displayed in the Name, Host, OU, Type, and Discovered columns.

The following information is provided:

- **Name:** The name of the discovered computer.
- **Host:** The host name of the discovered computer.
- **OU:** The organizational unit of the discovered computer.
- **Type:** The computer or network share.
- **Discovered:** The date and time stamp of the discovery.

Click **New Network Share** to add a network share.

- **Name:** The network share name.
- **Source:** The source connection to access the network share.
- **Path:** The path to the network share.

PSTs

Click this link to view a complete list of the discovered PSTs. By default, PSTs are sorted by discovered time. You can change the sort order by clicking the column headings. You can filter results displayed in the Name, Owner, Location, Path, Size, and Discovered columns.

The following information is provided:

- **Name:** The name of the PST file.
- **Owner:** The name of the person who owns the PST file.

  The owner of a PST file is determined by the Access Control List (ACL) on the source system. To change the owner, follow the instructions below.
- **Location:** The name of the computer or network share on which the PST file was discovered.
- **Path:** The relative path to the computer where the discovered PST is located.
- **Size:** The size of the PST file.
- **Discovered:** The date and time stamp of the discovery.

To change the owner of a PST file, complete the following steps:

1. On the PSTs page, hover over the owner of the PST in the Owner column to activate the Edit icon.
2. Click anywhere in the **Owner** box.
3. In the Owner box, type at least 2 characters of the owner name to display the drop-down list.
4. Select an owner for the PST file from the drop-down list.

5 Repeat this process for each PST owner you wish to change on the web page.

When the owner of a PST file has been changed, a red marker is displayed by the owner of the PST file.

6 If you have changed the owner of 1 or more PST files, the **Apply Changes** link at the top of the page is activated. Click **Apply Changes** to save your changes. Or, click **Cancel Changes** to remove any changes that you have made to the owners of PST files.

NOTE: Changes to the owner of a particular PST file on a web page must be saved before navigating to another page. If you attempt to navigate off the web page prior to saving your changes, the following message is displayed: You have unsaved changes. Do you want to save your changes before continuing?

You can also change the owner of the PST after discover with the **Set-MMPPstFileOwner** cmdlet.

Collections

The Collections link allows you to create collections of users or one or more PSTs for migration. By default, collections are sorted by alphabetically by name. You can change the sort order by clicking the column headings. You can filter results displayed in the Name, Type, and Label columns.

New User collection

On the Settings tab, enter the following information:

- **Name:** The collection name.
- **Label:** The collection label.

On the Users tab, select the users that you wish to include in the collection and move them to the table on the right.

New PST collection

On the Settings tab, enter the following information:

- **Name:** The collection name. Hover over the Name field to display the UNC path.
- **Label:** The collection label.

On the **PSTs** tab, select the PSTs you wish to include in the collection and move them to the table on the right.

Hiding and unhiding collections.

Once a collection is created it cannot be deleted. The collections can be hidden and unhidden from view as needed.

To hide collections:

- Click the **Hide** icon ( ) on a collection to hide it from view.

To view hidden collections:

- Select the **View Hidden** checkbox above the collection list.

To unhide hidden collections:

1 Select the **View Hidden** checkbox above the collection list to view the hidden user collections.

2 Click the **Unhide** icon ( ) on the hidden collections you want to unhide.

To view user collections after they are unhidden:

- Deselect the **View Hidden** checkbox to display the unhidden collections.

Collections for failed PST migrations

When situations occur in which a PST file cannot be migrated at first and it meets the retry criteria, the PST will be retried for migration. MMP will attempt to migrate the PST file again based on the number of retries entered in the **Retry attempts** box on the Migration Agent Configuration page in the Configuration Console.If the PST file still fails to migrate, the PST file is placed into a failed PST migrations collection.

To migrate a failed PST migrations collection:

1. Go to the migration task that contains the failed PST file and edit the task.
2. In the **Name** field, rename the task to a unique name. Renaming the original task instead of creating a new task allows you to keep the settings from the original task.
3. Under **Collection**, locate the collection with -failed appended to the name of the original collection that contained the failed PST file.
4. Make sure that the target is set correctly in the **Target** field.
5. Click **Next** and review the previous settings on the **Migration Settings** and **Schedule** tabs and proceed with the migration.

Targets

The Targets link allows you to enter the credentials that you use to connect to the target system. See additional details in the Migration Workflow section below. By default, target connections are sorted by alphabetically by name. You can change the sort order by clicking the column headings.

New Target

The New Target button contains the following fields:

- **Target Name:** The name to use for the target connection.
- **System:** The target system.
    - **Use auto-discovery** (checkbox): Select this checkbox if you want MMP to use the target system's auto-discover service to find the target server name (URL). If you leave this checkbox unmarked, you must specify the **Server name/URL** in the next field below.
- **Server name/URL:** The name —or— URL of the target server.
- **Use SSL** (checkbox): Select this checkbox if you want MMP to use SSL for its connections to the target server.
- **Admin name:** The name of the administrator account used to connect to the target.
- **Password:** The password for the administrator account.
- **Add Additional Credential:** Add an additional administrator account to use for the connection.

Hiding and unhiding target connections.

Once a target connection is created it cannot be deleted. The target connections can be hidden and unhidden from view as needed.

To hide target connections:

- Click the **Hide** icon () on a target connection to hide it from view.

To view hidden target connections:

- Select the **View Hidden** checkbox above the target connection list.

To unhide hidden target connections:

1. Select the **View Hidden** checkbox above the target connection list to view the hidden target connections.
2. Click the **Unhide** icon () on the hidden target connections you want to unhide.

To view target connections after they are unhidden:

- Deselect the **View Hidden** checkbox to display the unhidden target connections.

Migrate

The Migrate link allows you to create a new migration task or manage your existing migration tasks. By default, migration tasks are sorted by last run time. You can change the sort order by clicking the column headings. You can filter the results displayed for the Name, Status, Progress, and Last Ran columns.

New Migration

The New Migration button contains the following fields:

- **Name:** The migration task name.
- **Collection:** The collection to use for the migration task.
- **Target:** The target system to use for the migration.

Events

The Events link takes you to the Events page, which shows the complete list of events generated during discovery and migration. The top of the page displays the number of errors, warnings, and information messages. By default, events are sorted by timestamp. You can change the sort order by clicking the column headings. You can filter results displayed in the Type, Task, Message, Log Location, and Timestamp columns.

The following information is displayed:

- **Type:** The type of event message: error, warning, or information as indicated by the icons.
- **Task:** The task name for which the event is generated.
- **Message:** The event message content.
- **Log Location:** The path for the log file.
- **Timestamp:** The time the event message was generated.

Click on any event to display information about it in the Event details box. A brief synopsis of the event is shown in the box. Click **Details** to display the full message content of the event. The following information is also displayed:

- **Task:** The name of the task for which the event was generated. You can get the task ID by hovering over the task name.
- **Log Location:** Lists the location and name of the log file that contains the event.

- **Troubleshoot:** Click the link to query the Knowledge Base for informational articles about the event message.

Events can be exported to a CSV file from the Migration Web UI. This allows you to work with the results in a spreadsheet and create adhoc reports.

You can also export events to a CSV file using PowerShell cmdlets.

Dashboard widgets

The Migration Web UI home page contains widgets that display statistics for the most recent migration of each PST. Widgets are automatically refreshed every 10 minutes and can be manually refreshed by clicking the refresh button. Widgets include the following:

Migration Status

The Migration Status widget displays the status of all migration tasks run to date.

- **Pending:** Number of the most recent migrations, per PST, waiting to be processed.
- **In Progress:** Number of the most recent migrations, per PST, currently processing.
- **Completed:** Number of the most recent migrations, per PST, that have completed processing.
- **Completed with errors:** Number of the most recent migrations, per PST, that have completed processing, but had errors.
- **Failed:** Number of the most recent migrations, per PST, that could not be processed.
- **Canceled:** Number of the most recent migrations, per PST, that have been stopped.

Migration Progress

The Migration Progress widget shows a snapshot of the PST migration tasks each day for the last 7 days. Hover over the bars to see the migration task count for each date. These counts are not cumulative.

The Migration Progress widget shows the following information about the PST files selected for migration:

- Migrated PSTs
- Warnings
- Errors

Discovery Progress

The Discovery Progress widget shows a snapshot of the new PST files discovered each day for the last 7 days. Hover over the bars to see the discovered PST file count for each date. These counts are not cumulative.

The Discovery Progress widget shows the following information about the PST files:

- Discovered PSTs
- Warnings
- Errors

Events

The Events widget shows the number of events generated and the number of minutes since those events were generated. You can hover over any event to view the details of that event. To view all event messages, click **Show All**.

Migration Statistics

The Migration Statistics widget shows statistics about the migration or migrations to date, including:

- **First Migration Start:** The date and time that the first migration was started.
- **PSTs Migrated:** The number of PST migrations to date.
- **Messages Migrated:** The number of messages migrated to date.
- **Appointments Migrated:** The number of appointments migrated to date.
- **Tasks Migrated:** The number of tasks migrated to date.
- **Contacts Migrated:** The number of contacts migrated to date.
- **Errors:** The number of errors generated during the migration(s).
- **Data Migrated:** The amount of data migrated.
- **Current Rate:** The data migrated per hour for currently active tasks.

Migration History

The Migration History widget shows a series of migration task status snapshots for the last 7 days. Hover over the bars to see the status information for each date. To view the counts for each category, hover over the appropriate color for the category you want to view. Each snapshot is a cumulative status of all migration tasks run to that date.

The Migration History widget shows the following information about the PST files selected for migration:

- Pending: The number of migration tasks that are waiting to be processed.
- **In Progress:** The number of migration tasks currently processing.
- **Completed:** The number of migration tasks that have completed processing.
- **Completed with errors:** The number of migration tasks that have completed processing, but had errors.
- **Failed:** The number of migration tasks that could not be processed.
- **Canceled:** The number of migration tasks that have been stopped.

Other pages

The following pages display useful information about your migration.

Collections Dashboard

You can view the dashboard for a particular collection by completing the following steps:

1. Under **Collections**, find the name of the collection for which you want to view the dashboard.
2. Click **Details** for that collection.

   The widgets display the information for the collection you are viewing.

Targets Dashboard

You can view the dashboard for a particular collection by completing the following steps:

1. Under **Targets**, find the name of the target for which you want to view the dashboard.
2. Click **Details** for that target.

   The widgets display the information for the target you are viewing.

Metadata collection

Metadata is collected for discovered computers, users, and PSTs when filters are applied.

Metadata collected from discovered computers

The following metadata is collected from Active Directory for discovered computers when the following filter is applied: "(objectClass=computer)"

Displayed in MMP Database	Active Directory Property
Id	ObjectGuid
DnsHostName	DnsHostName
LastLogon	LastLogonTimeStamp
DistinguishedName	DistinguishedName
OrganizationalUnit	OrganizationalUnit
Name	Name

How the metadata from discovered computers is displayed in the UI

Displayed in MMP Database	Displayed As
Name	Name
DnsHostnName	Host
OrganizationalUnit	OU

Metadata collected from discovered users

The following metadata is collected from Active Directory for discovered users when the following filter is applied: "(&(&(objectCategory=person)(objectClass=user)))"

Displayed in MMP Database	Active Directory Property
Id	ObjectGuid
DisplayName	DisplayName
TargetEmailAddress	ProxyAddresses
Name	DisplayName
Category	objectCategory
SamAccountName	samAccountName
UserPrincipalName	userPrincipalName

Displayed in MMP Database	Active Directory Property
LegacyExchangeDn	LegacyExchangeDn
Ou	DistinguishedName
MailboxEnabled	homeMTA,homeDBB,msExchHomeServerName, msExchMailboxGuid
DomainName	Not taken from Active Directoyr, taken from source credentials
SID - Appears in the SID table in the database	objectSID
SID - Appears in the SID table in the database	SIDHistory

How metadata from discovered users is displayed in the UI

Displayed in MMP Database	Displayed As
Name	Name
TargetEmailAddress	Email
MailboxEnabled	MailboxEnabled
OU	OU

Metadata collected from discovered PST files.

The following metadata is collected from Active Directory for discovered PST files:

Displayed in MMP Database	File Properties
CreationTime	CreationTime
LastModifiedTime	LastModifiedTime
FileSize	Length
UncFilePath	FullName
FullPath	FullName
DiscoveredSidId	SecurityIdentifier
Name	Name

How metadata from discovered PSTs is displayed in the UI

Displayed in MMP Database	Displayed As
Name	Name
FullPath	Path
FileSize	Size

3

PowerShell Cmdlets

Using PowerShell Cmdlets

You may use these PowerShell cmdlets to configure and troubleshoot Migration Manager for PSTs.

To import the PowerShell module so you can use the MMP cmdlets, enter the following command in a PowerShell console:

```
Import-Module PSTMigratorModules
```

To view a list of the available MMP cmdlets, enter the following command:

```
Get-Command -Module PSTMigratorModules
```

Common Parameters

Many MMP cmdlets take the -InputObject and -Id parameters.

PARAMETER	DESCRIPTION
InputObject	You use the -InputObject parameter to specify a PowerShell object (or objects) as input to the cmdlet.
Id	You use the -Id parameter to specify a PowerShell object (or objects) as by the Id value of the object.

For example, you could add a collection to the MMP database using -InputObject as follows:

```
$Collection = New-MMPCollection -Name "My Name" -Label "My Label"
$Collection = Add-MMPCollection -InputObject $Collection
```

You could change the credentials associated with a Connection with the -Id parameter as follows:

```
$Credential = Get-Credential –ConnectionId $Connection.Id
```

```
$Id = $Credential.Id
Set-MMPCredential –Id $Id –Credentials (Get-Credential)
```

The MMP cmdlets support the common PowerShell parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -WarningAction, -WarningVariable, -OutBuffer and -OutVariable. For more information about these common parameters, enter the following command:

```
Get-Help about_commonparameters
```

Installation and configuration scenarios

The installation and configuration section provides scenarios for Configuring Migration Manager for PSTs with SQL Authentication and Configuring MMP with Windows Authentication, followed by detailed cmdlets.

Installing and configuring MMP with SQL Authentication

The following section lists the cmdlets required for installation and configuration with SQL Authentication. The cmdlets are listed in order of execution.

1. Create the **mmp.config** file if it does not exist:Start-MMPWebsite
2. Create the MMP database if it does not exist: Install-MMPDatabase
3. Check for a valid SQL login and create a database user in the MMP database if the user doesn't exist. Then assign the required database roles to the user: Set-MMPDatabaseRole
4. Configure website location for agents and PowerShell cmdlets: Set-MMPWebSite
5. Configure the connection to the SQL server: Set-MMPSqlConnection
6. Restart the MMP agents: Restart-MMPServices

For detailed cmdlets and examples and additional cmdlets, see the Installation cmdlets and Configuration cmdlets sections below.

Installing and configuring MMP with Windows Authentication

The following section lists the cmdlets required for installation and configuration with SQL Authentication. The cmdlets are listed in order of execution.

1. Create the **mmp.config** file if it does not exist: Start-MMPWebsite
2. Create the MMP database if it does not exist: Install-MMPDatabase
3. Check for a valid SQL login and create a database user in the MMP database if the user doesn't exist. Then assign the required database roles to the user: Set-MMPDatabaseRole
4. Configure website location for agents and PowerShell cmdlets: Set-MMPWebSite
5. Configure the credentials used for the application pool: Set-MMPWebSiteAppPoolCredentials
6. Configure the connection to the SQL server: Set-MMPSqlConnection
7. Restart the MMP agents: Restart-MMPServices

For detailed cmdlets and examples and additional cmdlets, see the Installation cmdlets and Configuration cmdlets sections below.

Installation cmdlets

Install-MMPConfigFile

Use the Install-MMPConfigFile cmdlet to create the **mmp.config** file if it does not exist.

Syntax

Install-MMPConfigFile [-PipelineVariable <String>] [<CommonParameters>]

Example

Install-MMPConfigFile

Install-MMPDatabase

Use the Install-MMPDatabase cmdlet to create the MMP database if it does not exist.

Syntax

```
Install-MMPDatabase -SqlServer <String> [<CommonParameters>]

Install-MMPDatabase [-Port <Nullable`1[Int32]>] [<CommonParameters>]

Install-MMPDatabase [-SqlAuthenticationCredentials <PSCredential>]
[<CommonParameters>]

Install-MMPDatabase [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
SqlServer	Required. The name and location of the SQL server. - For example, ".\SQLEXPRESS" or " SQLServer01.sitraka.com"
Port	The port for the SQL server.
SqlAuthenticationCredentials	Use this parameter to specify the PowerShell credential object.

Example

To create the database on SQLServer01.sitraka.com using SQL authentication, use the following example:

```
Install-MMPDatabase -SqlServer "SQLServer01.sitraka.com" -Port 443 -
SqlAuthenticationCredentials (Get-Credential)
```

Update-MMPDatabase

Use the Update-MMPDatabase cmdlet to upgrade the database to the current schema. If the database is current, this cmdlet will not change anything. If the database is out of date, this cmdlet triggers a database update.

Syntax

```
Update-MMPDatabase -SqlServer <String> [<CommonParameters>]

Update-MMPDatabase [-Port <Nullable`1[Int32]>] [<CommonParameters>]

Update-MMPDatabase [-SqlAuthenticationCredentials <PSCredential>]
[<CommonParameters>]

Update-MMPDatabase [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
SqlServer	Required. The name and location of the SQL server. For example, ".\SQLEXPRESS" or " SQLServer01.sitraka.com"
Port	The port for the SQL server.
SqlAuthenticationCredentials	Use this parameter to specify the PowerShell credential object.

Example

Update the database using Windows authentication.

```
Update-MMPDatabase -SqlServer ".\SQLSERVER"
```

Update the database on the specified machinename using SQL authentication.

```
Update-MMPDatabase -SqlServer "SQLServer01.sitraka.com" -Port 433 -
SqlAuthenticationCredentials (Get-Credential)
```

Set-MMPDatabaseRole

The Set-MMPDatabaseRole cmdlet checks for a valid SQL login and creates a database user in the MMP database if the user doesn't exist. It then assigns the required database roles to the user. This cmdlet requires administrator credentials.

Syntax

```
Set-MMPDatabaseRole -SqlLogin <String> [<CommonParameters>]

Set-MMPDatabaseRole -SqlServer <String> [<CommonParameters>]

Set-MMPDatabaseRole [-Port <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPDatabaseRole [-SqlAuthenticationCredentials <PSCredential>]
[<CommonParameters>]

Set-MMPDatabaseRole [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
SqlServer	Required. The name and location of the SQL server. - For example, ".\SQLEXPRESS" or "SQLServer01.sitraka.com"
Port	The port for the SQL server.

PARAMETER	DESCRIPTION
SqlAuthenticationCredentials	Use this parameter to specify the PowerShell credential object.
SqlLogin	A SQL login account.

Example

To assign the database role on the specified machinename using SQL authentication, use the following example:

```
Set-MMPDatabaseRole -SqlServer "SQLServer01.sitraka.com" -Port 433 -
SqlAuthenticationCredentials (Get-Credential) -SqlLogin sqllogin
```

Add-MMPLicense

Use the Add-MMPLicense cmdlet to register a new product license in the MMP database or to replace the existing license. You can view the current license info with the Get-MMPLicense cmdlet.

Running the Add-MMPLicense cmdlet requires administrator credentials.

Syntax

Add-MMPLicense -Filename <String> [<CommonParameters>]

Add-MMPLicense [-PipelineVariable <String>] [<CommonParameters>]

Parameters

PARAMETER	DESCRIPTION
Filename	Use the Filename parameter to specify the path to the license key file that you received from Dell.

Example

If the path to your new MMP license (.ASC) file was “C:\temp\MMP_license.asc”, then you could replace your old license with the following command:

```
Add-MMPLicense -Filename “C:\temp\MMP_license.asc”
```

Get-MMPLicense

Use the Get-MMPLicense cmdlet to see the currently configured license from the MMP server. You can update your license with the Add-MMPLicense cmdlet.

Syntax

```
Get-MMPLicense [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To retrieve the license usage information, use the following example:

```
Get-MMPLicense
```

Get-MMPLogConfig

Use the Get-MMPLogConfig cmdlet to return log configuration file for the specified component.

Syntax

```
Get-MMPLogConfig -ComponentType <ProductComponent> [<CommonParameters>]

Get-MMPLogConfig [-PipelineVariable <String>] [<CommonParameters>]

Get-MMPLogConfig -ConfigFileLocation <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConfigFileLocation	This parameter specifies the location of the log file to display.
ComponentType	If this parameter is specified, the cmdlet looks for the MMP installation and returns the contents of the log configuration file specified. This parameter can have one of the following values: DiscoveryAgent, MigrationAgent, ScheduleAgent, Webservice or WindowsPowershell.

Example

To return the log configuration file for the Discovery Agent, use the following example:

```
Get-MMPLogConfig -ComponentType DiscoveryAgent
```

The following is an example of the returned values from the Get-MMPLogConfig cmdlet for the Discovery Agent with default values:

```
Component: DiscoveryAgent
Location: C:\Program Files\Dell\Migration Manager for PSTs\Discovery Agent
Service\log4net.config
LogLevelThreshold: Info
FileSize: 25
FileCoun: 10
BufferEnabled: True
BufferMultiplier: 5
BufferSize: 50
Valid: True
```

Set-MMPLogConfig

Use the Set-MMPLogConfig cmdlet to specify the settings for a particular log configuration file.

Syntax

```
Set-MMPLogConfig -MmpLogConfig <ILog42NetConfig> [<CommonParameters>]

Set-MMPLogConfig [-LogLevelThreshold <Nullable`1[LogLevel]>] [<CommonParameters>]

Set-MMPLogConfig [-FileSize <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPLogConfig [-FileCount <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPLogConfig [-SetConfigToDefault [<SwitchParameter>]] [<CommonParameters>]

Set-MMPLogConfig [-PipelineVariable <String>] [<CommonParameters>]
```

```
Set-MMPLogConfig [-BufferSize <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPLogConfig [-BufferMultiplier <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPLogConfig [-BufferEnabled <Nullable`1[Boolean]>] [<CommonParameters>]
```

The following returned values are not parameters and cannot be changed:

- The **Component** returned is the **ComponentType** specified in the Get-MMPLogConfig cmdlet.
- The **Location** returned is the path to the component specified by the **ConfigFileLocation** or **ComponentType** specified in the Get-MMPLogConfig cmdlet.
- **Valid** indicates whether all portions of the log configuration file were successfully read. It is updated when the log file is read. After saving changes to the configuration file, the file is read again and the output is sent to the command line.

Parameters

PARAMETER	DESCRIPTION
MMPLogConfig	This mandatory parameter is provided by the output of the Get-MMPLogConfig parameter. It can be passed on the pipeline, as shown in the example below.
FileSize	The maximum size that can be written to the log file before the logger writes to a new log file.
FileCount	The number of log files that will be retained.
BufferEnabled	Indicates whether the MultithreadedBufferingForwardingAppender is enabled.
BufferMultiplier	The internal queue of messages set to the buffer size times the multiplier.
BufferSize	The number of lines that will be retained in the buffer before the lines are written to the log file.

Example

To set the file size for the Discovery Agent log file, use the following example:

```
$config = Get-MMPLogConfig -ComponentType DiscoveryAgent

$config | Set-MMPLogConfig -FileSize 20
```

Advanced example

To set each logConfig to the default settings, use the following example:

```
$logConfigs = @()
$configs = "DiscoveryAgent", "MigrationAgent", "ScheduleAgent", "WebService",
"WindowsPowershell"
foreach($x in $configs)
{
$logConfigs += Get-MMPLogConfig -ComponentType $x
}
$logConfigs | Set-MMPLogConfig -SetConfigToDefault
```

Advanced example with comments

To set each logConfig to the default settings, use the following example:

To initialize the variable the **logConfigs** from the Get-MMPLogConfig cmdlet will be saved in:

```
$logConfigs = @()
```

To list each options for the **Enum -ComponentType**:

```
$configs = "DiscoveryAgent", "MigrationAgent", "ScheduleAgent", "WebService",
"WindowsPowershell"
```

To retrieve each component's **logConfig** based on the list above, use the following example:

```
foreach($x in $configs)
{
$logConfigs += Get-MMPLogConfig -ComponentType $x
}
```

To set each retrieved **logConfig** to the default settings, use the following example:

```
$logConfigs | Set-MMPLogConfig -SetConfigToDefault
```

Restart-MMPServices

Use the Restart-MMPServices cmdlet to restart the MMP agents. The MMP agents include Migration, Discovery, and Schedule.

Syntax

```
Restart-MMPServices [-PipelineVariable <String>] [<CommonParameters>]
```

Example

```
Restart-MMPServices
```

Confirm-MMPSqlConnection

Use the Confirm-MMPSqlConnection cmdlet to validate SQL connection settings.

Syntax

```
Confirm-MMPSqlConnection [-InputObject <SqlConnectionSettings>]
[<CommonParameters>]

Confirm-MMPSqlConnection [-SqlServer <String>] [<CommonParameters>]

Confirm-MMPSqlConnection [-Port <Nullable`1[Int32]>] [<CommonParameters>]

Confirm-MMPSqlConnection [-PipelineVariable <String>] [<CommonParameters>]

Confirm-MMPSqlConnection [-SqlAuthenticationCredentials <PSCredential>]
[<CommonParameters>]

Confirm-MMPSqlConnection [-UseSqlAuthentication [<SwitchParameter>]]
[<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
SqlServer	The SQL server to connect to, also referred to as Data Source.
Port	Optionally use this parameter to set the port for the SQL server.
UseSqlAuthentication	Optionally use this parameter to specify using SQL authentication.
SqlAuthenticationCredentials	Required if SQL authentication is specified. Credentials to use for SQL authentication.

Example

To validate a change to the current SQL connection settings, use the following example:

```
Get-MMPSqlConnection | Confirm-MMPSqlConnection -SqlServer "SQLServer01.sitraka.com"
-Port 12345 -UseSqlAuthentication -SqlAuthenticationCredentials (Get-Credential)
```

To validate new SQL connection settings, follow this example:

```
Confirm-MMPSqlConnection -SqlServer "NewServer" -Port 12345 -UseSqlAuthentication -
SqlAuthenticationCredentials (Get-Credential)
```

Get-MMPSqlConnection

Use the Get-MMPSqlConnection cmdlet to get the configured SQL connection settings.

Syntax

```
Get-MMPSqlConnection [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To get the configured SQL connection settings, use the following example:

```
Get-MMPSqlConnection
```

Set-MMPSqlConnection

Use the Set-MMPSqlConnection cmdlet to configure the connection to the SQL server. If the UseSqlConnection parameter is not specified, Windows Authentication will be used.

Syntax

```
Set-MMPSqlConnection [-InputObject <SqlConnectionSettings>] [<CommonParameters>]

Set-MMPSqlConnection [-SqlServer <String>] [<CommonParameters>]

Set-MMPSqlConnection [-Port <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPSqlConnection [-PipelineVariable <String>] [<CommonParameters>]

Set-MMPSqlConnection [-SqlAuthenticationCredentials <PSCredential>]
[<CommonParameters>]

Set-MMPSqlConnection [-UseSqlAuthentication [<SwitchParameter>]]
[<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
SqlServer	The SQL server to connect to, also referred to as Data Source.
Port	Optionally use this parameter to set the port for the SQL server.
UseSqlAuthentication	Optionally use this parameter to specify using SQL authentication.
SqlAuthenticationCredentials	Required if SQL authentication is specified. Credentials to use for SQL authentication.

Example

To change the configured SQL connection settings, follow this example:

```
Set-MMPSqlConnection -SqlServer "sitraka\sqlexpress" -UseSqlAuthentication $true -
SqlAuthenticationCredentials (Get-Credential)
```

Set-MMPWebSite

Use the Set-MMPWebSite cmdlet to configure website location for agents and PowerShell cmdlets so the other PowerShell cmdlets don't require you to set the following:

- ServerName with the Fully Qualified Domain Name of the MMP website,
- Port,
- Protocol

The Set-MMPWebSite cmdlet sets these globally but they can be overridden. The Set-MMPWebiste cmdlet also changes the ports in IIS (binding) of the Dell Migration Manager for PSTs website and modifies the configuration files for agents and PowerShell cmdlets on this computer.

The mmp.config file is located in: Common ApplicationData (C:\ProgramData on windows7) {CommonAppData}\Dell\Migration Manager for PSTs\

Syntax

```
Set-MMPWebsite -ServerName <String> [<CommonParameters>]

Set-MMPWebsite -Port <Int32> [<CommonParameters>]

Set-MMPWebsite [-AllowSelfSignedCert [<SwitchParameter>]] [<CommonParameters>]

Set-MMPWebsite [-RequestTimeout <Nullable`1[Int32]>] [<CommonParameters>]

Set-MMPWebsite -Protocol <BindingProtocol> [<CommonParameters>]

Set-MMPWebsite [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Port	The port number used for the binding.
Protocol	Sets the security level. Valid values are: -- Http: Binding will use Http -- Https: Binding will use Https
ServerName	Use this parameter to specify the name of the server where your MMP web service is running. When setting the ServerName with HTTPS, the ServerName should match the name on the certificate assigned to the website.

Example

To configure website location for agents and PowerShell cmdlets, use the following example:

```
Set-MMPWebsite -ServerName "Sitraka.com" -Port 443 -Protocol https
```

Configuration cmdlets

Get-MMPAgent

Use the Get-MMPAgent cmdlet to list the your MMP agents. You can use the -Id parameter to get a single agent.

Syntax

```
Get-MMPAgent [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPAgent [-Id <String>] [<CommonParameters>]
Get-MMPAgent [-InputObject <Agent>] [<CommonParameters>]
```

Example

To get a list of all the currently idle agents, use the following example:

```
Get-MMPAgent | Where-Object {$_.AgentStatus -eq "Idle"}
```

Get-MMPAgentConfiguration

Use the Get-MMPAgentConfiguration cmdlet to return the configuration for the MMP agent.

Syntax

```
Get-MMPAgent [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPAgent [-Id <String>] [<CommonParameters>]
Get-MMPAgent [-InputObject <Agent>] [<CommonParameters>]
```

Example

To display the MMP configuration, use the following example:

```
Get-MMPAgentConfiguration
```

Set-MMPAgentConfiguration

Use the Set-MMPAgentConfiguration cmdlet to update the MMP agent configuration settings.

Syntax

```
Set-MMPAgentConfiguration [-RetryDelay <Nullable`1[Double]>] [<CommonParameters>]
Set-MMPAgentConfiguration [-RetryAttempts <Nullable`1[Int32]>] [<CommonParameters>]
Set-MMPAgentConfiguration [-MaxMessageSize <Nullable`1[Int64]>]
[<CommonParameters>]
Set-MMPAgentConfiguration [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
RetryDelay	The minimum time in minutes to wait before retrying a task. The default is 30.
RetryAttempts	The number of retries before the task fails and the status is marked as an error. The default is 5.
MaxMessageSize	Sets the maximum message size allowed for migration. The default is 10MB.

Example

To set the maximum message size to 25MB, use the following example:

```
Set-MMPAgentConfiguration -MaxMessageSize 26214400
```

Get-MMPAgentSettings

Use the Get-MMPAgentSettings cmdlet to return the configuration settings for the MMP agent.

Syntax

```
Get-MMPAgentSettings -Agent <AgentType> [<CommonParameters>]
Get-MMPAgentSettings [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Agent	The type of agent for which to return configuration settings. Agents include the following: • Schedule • Discovery • Migration

Example

To display the MMP agent settings, use the following example:

```
Get-MMPAgentSettings -Agent Discovery
```

Set-MMPAgentSettings

Use the Set-MMPAgentSettings cmdlet to update the settings for the MMP agent.

Syntax

```
Set-MMPAgentSettings -Agent <AgentType> [<CommonParameters>]
Set-MMPAgentSettings [-TaskCount <Nullable`1[Int32]>] [<CommonParameters>]
Set-MMPAgentSettings [-PollIntervalSeconds <Nullable`1[Int32]>][<CommonParameters>]
```

```
Set-MMPAgentSettings [-Force [<SwitchParameter>]] [<CommonParameters>]
Set-MMPAgentSettings [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Agent	The type of agent for which to return configuration settings. Agents include the following: • Schedule • Discovery • Migration
TaskCount	The number of tasks an agent can concurrently handle. The default is 10 for the Discovery Agent, and 5 for the Migration Agent.
PollIntervalSeconds	The amount of time in seconds between checking for tasks. The default is 10 seconds for all agents.

Example

To update MMP agent settings, use the following example:

```
Set<Nullable`1[Boolean]>]
[<CommonParameters>]
Set-MMPCommonAgentSettings [-UseSelfsignedCert <Nullable`1[Boolean]>]
[<CommonParameters>]
Set-MMPCommonAgentSettings [-RequestTimeout <Nullable`1[Int32]>]
[<CommonParameters>]
Set-MMPCommonAgentSettings [-PipelineVariable <String>]
[<CommonParameters>]
```

Get-MMPAppConfig

Use the Get-MMPAppConfig cmdlet to view the current application configuration from the MMP server.

Syntax

```
Get-MMPAppConfig [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To display the MMP configuration, use the following example:

```
Get-MMPAppConfig
```

Set-MMPCommonAgentSettings

Use the Set-MMPCommonAgentSettings cmdlet to update configuration settings shared by all MMP agents.

Syntax

```
Set-MMPCommonAgentSettings [-ServerName <String>] [<CommonParameters>]
Set-MMPCommonAgentSettings [-NonSsl Get-MMPAppConfig]
```

Syntax

```
Get-MMPAppConfig [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To display the MMP configuration, use the following example:

```
Get-MMPAppConfig
```

Repair-MMPConfigurations

Use the Repair-MMPConfigurations cmdlet to check and update the configuration file for the webserver from previous versions. This cmdlet displays an error if run on a machine without the web server installed.

Syntax

```
Repair-MMPConfigurations [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To check and update the configuration file for the webserver from previous versions, use the following example:

```
Repair-MMPConfigurations
```

Start-MMPWebsite

Use the Start-MMPWebsite cmdlet to start the MMPWebsite.

Syntax

```
Start-MMPWebSite  [<CommonParameters>]
```

Example

To start the MMPWebsite, use the following example:

```
Start-MMPWebSite
```

Stop-MMPWebsite

Use the Stop-MMPWebSite cmdlet to stop the MMPWebsite.

Syntax

```
Stop-MMPWebsite [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To stop the MMPWebsite, use the following example:

```
Stop-MMPWebSite
```

Start-MMPWebSiteAppPool

Use the Start-MMPAppPool cmdlet to start the MMP application pool.

Syntax

```
Start-MMPWebsiteAppPool [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To start the MMP application pool, use the following example:

```
Start-MMPWebSiteAppPool
```

Stop-MMPWebSiteAppPool

Use the Stop-MMPAppPool cmdlet to stop the MMP application pool.

Syntax

```
Stop-MMPWebsiteAppPool [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To stop the MMP application pool, use the following example:

```
Stop-MMPAppPool
```

Set-MMPWebSiteAppPoolCredentials

Use the Set-MMPWebSiteAppPoolCredentials cmdlet to configure the credentials used for the application pool.

Syntax

```
Set-MMPWebsiteAppPoolCredentials [-Credentials <PSCredential>] [<CommonParameters>]
Set-MMPWebsiteAppPoolCredentials [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To set the credentials used by the MMP application pool, use the following example:

```
Set-MMPWebSiteAppPoolCredentials -Credentials (Get-Credential)
```

Get-MMPWebSiteAppPoolState

Use the Get-MMPAppPoolState cmdlet to return the state of the MMP application pool.

Syntax

```
Get-MMPWebsiteAppPoolState [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To get the state of the MMP App Pool, use the following example:

```
Get-MMPWebSiteAppPoolState
```

Get-MMPWebSiteAppPoolUserName

Use the Get-MMPWebSiteAppPoolUserName cmdlet to return the name of the user whose credentials are being used for the AppPool.

Syntax

```
Get-MMPWebsiteAppPoolUserName [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To return the name of the user whose credentials are being used for the MMP application pool, use the following example:

```
Get-MMPWebSiteAppPoolUserName
```

Add-MMPWebsiteBinding

Use the Add-MMPWebsiteBinding cmdlet to add a binding to the MMP website that uses a specific port and sets the security level.

Syntax

```
Add-MMPWebsiteBinding -Port <Int32> [<CommonParameters>]
Add-MMPWebsiteBinding -Protocol <BindingProtocol> [<CommonParameters>]
Add-MMPWebsiteBinding [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Port	The port number used for the binding. Use this optional parameter to specify the protocol the binding will use.
Protocol	Sets the security level. Valid values are: -- Http: Binding will use Http -- Https: Binding will use Https

Example

To add a binding to the MMP website that uses a specific port and sets the security level to Http, follow this example:

```
Add-MMPWebsiteBinding -Port 80 -Protocol Http
```

To add a binding to the MMP website that uses a specific port and sets the security level to Https, follow this example:

```
Add-MMPWebsiteBinding -Port 443 -Protocol Https
```

Get-MMPWebsiteBinding

Use the Get-MMPWebsiteBinding cmdlet to retrieve all MMP website bindings.

Syntax

```
Get-MMPWebsiteBinding [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To retrieve all MMP website bindings, use the following example:

```
Get-MMPWebsiteBinding
```

Remove-MMPWebsiteBinding

Use the Remove-MMPWebsiteBinding cmdlet to remove a specific MMP website binding or all MMP website bindings.

Syntax

```
Remove-MMPWebsiteBinding [-Binding <Binding>] [<CommonParameters>]
Remove-MMPWebsiteBinding [-RemoveAll [<SwitchParameter>]] [<CommonParameters>]
Remove-MMPWebsiteBinding [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Binding	How IIS determines what requests a website responds to based upon the IP address, hostname, and port.
RemoveAll	Optionally use this parameter to remove all bindings from the MMP website.

Example

To assign a binding from port 443 to a variable for use in other cmdlets, such as Remove-MMPWebsiteBinding, follow this example:

```
$binding = Get-MMPWebsiteBinding | Where-Object -Property BindingInformation -like
"*443*"

Remove-MMPWebsiteBinding -Binding $binding
```

Get-MMPWebsiteState

Use the Get-MMPWebSiteState cmdlet to retrieve the state of the MMP website. States of the MMP website include: Starting; Started; Stopping; Stopped; and Unknown.

Syntax

```
Get-MMPWebsiteState [-PipelineVariable <String>] [<CommonParameters>]
```

Example

To retrieve the state of the MMP website, use the following example:

```
Get-MMPWebSiteState
```

Discovery scenario

The Discovery section provides a scenario for performing the discovery process, followed by detailed cmdlets.

Discovery

The following section lists the cmdlets required for discovery. The cmdlets are listed in order of execution for the discovery process:

1. Create a new connection: Add-MMPCredential
2. Add the connection to the MMP database: Add-MMPConnection
3. Get the connection from the MMP database: Get-MMPConnection
4. Create a credential for a connection: Remove-MMPCredential
5. Add a credential to the MMP database: Add-MMPCredential
6. Retrieve a list of credentials from the MMP server: Get-MMPCredential
7. Create a user discovery task object in MMP: Start-MMPUserDiscoveryTask
8. Get the user discovery task from the MMP database and verify it's status: Get-MMPTask

9 Create a computer discovery task object in MMP: Start-MMPComputerDiscoveryTask

10 Get the computer discovery task from the MMP database and verify it's status: Get-MMPTask

11 Retrieve a list of the computers discovered by MMP: Get-MMPDiscoveredComputer

12 Create a task object in MMP: Start-MMPPstFileDiscoveryTask

13 Get the task object from the MMP database and verify it's status: Get-MMPTask

14 Retrieve a list of events from the MMP server: Get-MMPEvent

For detailed cmdlets and examples and additional cmdlets, see the Discovery cmdlets section below.

Discovery cmdlets

Start-MMPComputerDiscoveryTask

Use the Start-MMPComputerDiscoveryTask cmdlet to queue a computer discovery task for processing by the MMP agents. The OuFilter parameter is used to provide an LDAP filter. Use the StartTime, RecurDaysOfWeek, RecurCount and RecurEndTime parameters to configure the task schedule.

Syntax

```
Start-MMPComputerDiscoveryTask -SourceConnectionId <Nullable`1[Guid]>
[<CommonParameters>]

Start-MMPComputerDiscoveryTask -Name <String> [<CommonParameters>]

Start-MMPComputerDiscoveryTask [-StartTimeUtc <DateTime>] [<CommonParameters>]

Start-MMPComputerDiscoveryTask [-RecurEndTimeUtc <Nullable`1[DateTime]>]
[<CommonParameters>]

Start-MMPComputerDiscoveryTask [-RecurCount <Nullable`1[Int32]>]
[<CommonParameters>]

Start-MMPComputerDiscoveryTask [-RecurDaysOfWeek <RecurDaysOfWeekEnum>]
[<CommonParameters>]

Start-MMPComputerDiscoveryTask [-PipelineVariable <String>] [<CommonParameters>]

Start-MMPComputerDiscoveryTask -TaskId <Nullable`1[Guid]> [<CommonParameters>]

Start-MMPComputerDiscoveryTask -SourceConnectionID <MigrationConnection> [-OuFilter
<string[]>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TaskId	Use the TaskId parameter to set the task by GUID for the computer discovery task.
SourceConnectionId	Use the SourceConnectionId GUID to specify the source connection for this task.
StartTimeUtc	Optionally use this parameter to set the UTC time of day when the agent will launch the computer discovery task.
RecurEndTimeUtc	Optionally use this parameter to set the UTC end time for this computer discovery task.
RecurCount	Optionally use this parameter to set the number of times this agent will launch the computer discovery task.

PARAMETER	DESCRIPTION
RecurDaysOfWeek	Optionally use this parameter to set which days of the week are on the schedule for the computer discovery task. Options include Everyday, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday.
Name	Use this parameter to set the name of the computer discovery task.
OuFilter	Use this parameter to filter by organizational unit.

Example

To create a Task object in MMP, use the following example. This syntax stores the Task in the "$ComputerDiscovery" variable.

```
$ComputerDiscovery = Start-MMPComputerDiscoveryTask -SourceConnectionId
$sourceConnection.Id -Name "Sitraka Active Directory Computer Discovery"
```

To get the task above from the MMP database and verify that it has a status of completed, use the following example:

```
Get-MMPTask -Id $ComputerDiscovery.Id
```

To get an organizational unit (OU) to filter by when performing a computer discovery task, use the following example:

```
$ou = Get-MMPOrganizationalUnit -Connection $connection | Select -Last 1
```

```
Start-MMPComputerDiscoveryTask -OuFilter $ou -SourceConnectionId $connection.Id -
Name "Computer Discovery with Ou Filtering"
```

New-MMPConnection

Use the New-MMPConnection cmdlet to create a source or target connection. You can use the Add-MMPConnection cmdlet to add a new connection to the MMP database.

Note that if no connection flag is specified (-SourceConnection, -TargetConnection), the cmdlet defaults to a source connection. If **-EwsUrl** is not specified, autodiscover is used on a target connection.

Syntax

```
New-MMPConnection -Name <String> [<CommonParameters>]
```

```
New-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
```

```
New-MMPConnection -SourceConnection [<SwitchParameter>] [<CommonParameters>]
```

```
New-MMPConnection -TargetConnection [<SwitchParameter>] [<CommonParameters>]
```

```
New-MMPConnection -ServerKind <ServerType> [<CommonParameters>]
```

```
New-MMPConnection [-NonSsl [<SwitchParameter>]] [<CommonParameters>]
```

```
New-MMPConnection -EwsUrl <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to set the name of your new connection.
NonSsl	Use this optional parameter if you do not want to use SSL to connect to the MMP web services over HTTP. By default, HTTPS is used.
TargetConnection	Use this switch if you want a new target connection.

PARAMETER	DESCRIPTION
SourceConnection	Use this switch if you want a new source connection.
ServerKind	Use the ServerKind parameter to set the type of target server. This parameter only applies to new target connections. Valid values are Exchange2010, Exchange2013, Exchange2016 and Office365.
EwsUrl	Use the EwsUrl parameter to set the URL of the Exchange Web Service on the target Exchange server..This parameter only applies to new target connections.

Example

To create a new target connection, use the following example:

```
$targetConnection = New-MMPConnection -Name "Sitraka Exchange Connection" -
TargetConnection -EwsUrl "exchange.sitraka.com" -ServerKind Exchange2010
```

```
$targetConnection = $targetConnection | Add-MMPConnection
```

Add-MMPConnection

Use the Add-MMPConnection cmdlet to add a source or target connection to the MMP database. You can create a connection configuration with the New-MMPConnection cmdlet.

Syntax

```
Add-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
```

```
Add-MMPConnection -InputObject <MigrationConnection> [<CommonParameters>]
```

Example

To add the connection to the MMP Database, use the following example.

```
$sourceConnection = New-MMPConnection -Name "Sitraka Source Connection"
-SourceConnection$sourceConnection = $sourceConnection | Add-MMPConnection
```

Get-MMPConnection

Use the Get-MMPConnection cmdlet to retrieve a list of connections from the MMP server. You can use the -Connection, -ConnectionId, -Task, -TaskId, -User, and -UserId parameters to get connections related to specific objects in the MMP database.

Syntax

```
Get-MMPConnection [-ReturnType <ReturnType>] [<CommonParameters>]
```

```
Get-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
```

```
Get-MMPConnection [-TargetConnection [<SwitchParameter>]] [<CommonParameters>]
```

```
Get-MMPConnection [-SourceConnection [<SwitchParameter>]] [<CommonParameters>]
```

```
Get-MMPConnection [-Collection <MigrationCollection>] [<CommonParameters>]
```

```
Get-MMPConnection [-CollectionId <Guid>] [<CommonParameters>]
```

```
Get-MMPConnection [-Task <MigrationTask>] [<CommonParameters>]
```

```
Get-MMPConnection [-TaskId <Guid>] [<CommonParameters>]

Get-MMPConnection [-Id <String>] [<CommonParameters>]

Get-MMPConnection [-InputObject <MigrationConnection>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TargetConnection	Use this switch if you want to retrieve the target connections.
SourceConnection	Use this switch if you want to retrieve the source connections.
Collection	Use the Collection parameter to retrieve the list of connections associated with that collection.
CollectionId	Use the Collection parameter to retrieve the list of connections associated with that collection GUID.
Task	Use the Task parameter to retrieve the list of connections associated with that task.
TaskId	Use the Task parameter to retrieve the list of connections associated with that task GUID.
ReturnType	Use the ReturnType parameter to retrieve hidden, not hidden, or all connections.

Example

To get the connection from the MMP Database, use the following example:

```
$sourceConnection = Get-MMPConnection -SourceConnection | Where-Object -Property
Name -eq "Sitraka Source Connection"
```

Set-MMPSourceConnection

Use the Set-MMPSourceConnection cmdlet to set a source connection on the MMP server.

Syntax

```
Set-MMPSourceConnection -Id <string> [-Name <string>] [-Hidden <bool>]
[<CommonParameters>]

Set-MMPSourceConnection -InputObject <MigrationConnection> [-Name <string>] [-Hidden
<bool>]  [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to update the name of your connection.
Hidden	Optionally use this parameter to update the hidden attribute of a connection.

Example

To set a source connection to hidden in the database, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Source Connection"

Set-MMPSourceConnection -InputObject $Connection -Hidden $True
```

Set-MMPTargetConnection

Use the Set-MMPTargetConnection cmdlet to set a target connection on the MMP server.

Syntax

```
Set-MMPTargetConnection -Id <string> [-ServerKind <ServerType> {Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-UseAutoDiscover] [-Name <string>] [-Hidden
    <bool>]  [<CommonParameters>]

Set-MMPTargetConnection -Id <string> [-ServerKind <ServerType> {Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-EwsUrl <string>] [-Name <string>] [-Hidden
    <bool>]  [<CommonParameters>]

Set-MMPTargetConnection -InputObject <MigrationConnection> [-ServerKind <ServerType> {Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-EwsUrl <string>] [-Name
    <string>] [-Hidden <bool>]  [<CommonParameters>]

Set-MMPTargetConnection -InputObject <MigrationConnection> [-ServerKind <ServerType> {Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-UseAutoDiscover] [-Name
    <string>] [-Hidden <bool>]  [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to update the name of your connection.
ServerKind	Use the ServerKind parameter to set the type of target server. This parameter only applies to new target connections. Valid values are Exchange2010, Exchange2013, Exchange2016 and Office365.
UseAutoDiscover	Use this switch to update the use of autodiscovery on the Exchange target server. This parameter only applies to new target connections.
EwsUrl	Use the EwsUrl parameter to update the URL of the Exchange Web Service on the target Exchange server.This parameter only applies to new target connections.
Hidden	Optionally use this parameter to update the hidden attribute of a connection.
UseSsl	Use a Secure Sockets Layer certificate.

Example

To change the server type of a connection from Exchange 2010 to Exchange 2013, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"

Set-MMPTargetConnection -Id $TargetConnection.Id -ServerKind Exchange2013
```

To update a connection to use Autodiscover, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"

Set-MMPTargetConnection -Id $TargetConnection.Id -UseAutoDiscover
```

To update a connection to use EWS, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
```

```
Set-MMPTargetConnection -Id $TargetConnection.Id -EwsUrl Exchange2010.sitraka.com
```

To set a target connection to hidden in the database, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
Set-MMPTargetConnection -InputObject $Connection -Hidden $True
```

Add-MMPCredential

Use the Add-MMPCredential cmdlet to add a credential to the MMP database. To get credentials from the MMP database, use the Get-MMPCredential cmdlet. You can create a new credential with the New-MMPCredential cmdlet.

Syntax

```
Add-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]
Add-MMPCredential -InputObject <Credential> [<CommonParameters>]
```

Example

To add the credential to the MMP database, use the following example:

```
$cred = New-MMPCredential -Credentials (Get-Credential) -ConnectionId
$targetConnection.Id
$cred = $cred | Add-MMPCredential
```

Get-MMPCredential

Use the Get-MMPCredential cmdlet to retrieve a list of credentials from the MMP server. You can use the -ConnectionId parameter to get the credentials for a specific connection.

Syntax

```
Get-MMPCredential [-ConnectionId <Guid>] [<CommonParameters>]
Get-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPCredential [-Id <String>] [<CommonParameters>]
Get-MMPCredential [-InputObject <Credential>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Specifying a connection with the ConnectionId GUID will retrieve the list of credentials associated with that connection.

Example

To retrieve the credential for the connection, use the following example:

```
$cred = Get-MMPCredential -ConnectionId $targetConnection.Id
```

Get-MMPDiscoveredComputer

Use the Get-MMPDiscoveredComputer cmdlet to retrieve a list of the computers discovered by MMP. You can use the -Id parameter to retrieve a specific computer.

Syntax

```
Get-MMPDiscoveredComputer [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPDiscoveredComputer [-Id <String>] [<CommonParameters>]
Get-MMPDiscoveredComputer [-InputObject <Computer>] [<CommonParameters>]
```

Example

To retrieve all of the discovered computers, use the following example:

```
$ComputersToSearch = Get-MMPDiscoveredComputer
```

To create a variable for all computers discovered in a specific OU, use the following example:

```
$computers = Get-MMPDiscoveredComputer | Where-Object OrganizationalUnit -like
"*Sales*"
```

Remove-MMPCredential

Use the Remove-MMPCredential cmdlet to mark a credential as deleted in the database.

Syntax

```
Remove-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]
Remove-MMPCredential -Id <String> [<CommonParameters>]
Remove-MMPCredential -InputObject <Credential> [<CommonParameters>]
```

Example

To mark a credential as deleted in the database, use the following example:

```
$Credential = Get-MMPCredential -Id
Remove-MMPCredential -Id $Credential.Id
```

Set-MMPCredential

Use the Set-MMPCredential cmdlet to update a credential on the MMP server.

Syntax

```
Set-MMPCredential [-ConnectionId <Guid>] [<CommonParameters>]
Set-MMPCredential [-Credentials <PSCredential>] [<CommonParameters>]
Set-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]
Set-MMPCredential -Id <String> [<CommonParameters>]
Set-MMPCredential -InputObject <Credential> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Use the ConnectionId GUID to update the connection for this credential.
Credentials	Use this parameter to update the PowerShell credential object.

Example

To update the credentials, for example, the password, for a credential stored in the MMP database, use the following example:

```
Set-MMPCredential -Id $Credential.Id -Credentials (Get-Credential)
```

To update a connection on a credential use the following command:

```
$connection = Get-MMPConnection | Where-Object Name -Like "My Connection"
$credential = Get-MMPCredential | Where-Object UserName -Like "Administrator"
Set-MMPCredential -InputObject $credential -ConnectionId $connection.id
```

Add-MMPNetworkShare

Use the Add-MmpNetworkShare cmdlet to create network share computer objects.

Syntax

```
Add-MMPNetworkShare -ConnectionId <Guid> [<CommonParameters>]
Add-MMPNetworkShare -Name <String> [<CommonParameters>]
Add-MMPNetworkShare -Path <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Specifying a connection with the ConnectionId GUID will retrieve the list of credentials associated with that connection.
Name	Use the Name parameter to set the name of your new connection.
Path	Use this parameter to list the paths to be searched for PST files on the computers. When a local path is used, all computers selected will be searched on that path. When a UNC path(s) is specified, only the computer in the UNC path(s) is searched.

Example

To create a network share computer object, use the following example:

```
$SourceConnection = Get-MMPConnection | Where-Object -Property Name -EQ "My
Connection"
Add-MMPNetworkShare -Name 'MyNetworkShare' -Path '\\MyShare' -ConnectionId
$sourceconnection.Id
```

Confirm-MMPNetworkShare

Use the Confirm-MMPNetworkShare cmdlet to validate that a network share path is accessible with a given connection. The cmdlet will output a message indicating whether the share was valid or not.

Syntax

```
Confirm-MMPNetworkShare -Connection <MigrationConnection> [<CommonParameters>]

Confirm-MMPNetworkShare -Path <String> [<CommonParameters>]

Confirm-MMPNetworkShare [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Specifying a connection with the ConnectionId GUID will retrieve the list of credentials associated with that connection.
Path	Use this parameter to list the paths to be searched for PST files on the computers. When a local path is used, all computers selected will be searched on that path. When a UNC path(s) is specified, only the computer in the UNC path(s) is searched.

Example

To validate that a network share path is accessible:

```
$SourceConnection = Get-MMPConnection | Where-Object -Property Name -EQ "My
Connection"

Confirm-MMPNetworkShare -Path '\\MySharePath' -ConnectionId $sourceconnection.Id -
verbose
```

Valid:

```
Network Share '\\MySharePath' successfully validated with user
'administrator@mydomain.com'.
```

Invalid:

```
Failed to validate Network Share '\\MySharePath' with user
'administrator@mydomain.com'. Details: 'Logon failure.
```

Get-MMPNetworkShare

Use the Get-MMPNetworkShare cmdlet to retrieve network share Computer objects.

Syntax

```
Get-MMPNetworkShare [-PipelineVariable <String>] [<CommonParameters>]

Get-MMPNetworkShare [-Id <String>] [<CommonParameters>]

Get-MMPNetworkShare [-InputObject <Computer>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Id	You use the -Id parameter to specify a PowerShell object (or objects) as by the Id value of the object.

Example

To get all network shares:

```
Get-MMPNetworkShare
```

To get a specific network share:

```
Get-MMPNetworkShare -Name 'MyNetworkShare' -Path '\\MyShare' -ConnectionId
$sourceconnection.Id
```

Set-MMPNetworkShare

Use the Set-MmpNetworkShare cmdlet to change the name or path of a network share computer object.

Syntax

```
Set-MMPNetworkShare [-Name <String>] [<CommonParameters>]

Set-MMPNetworkShare [-Path <String>] [<CommonParameters>]

Set-MMPNetworkShare [-PipelineVariable <String>] [<CommonParameters>]

Set-MMPNetworkShare -Id <String> [<CommonParameters>]

Set-MMPNetworkShare -InputObject <Computer> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to set the name of your new connection.
Path	Use this parameter to list the paths to be searched for PST files on the computers. When a local path is used, all computers selected will be searched on that path. When a UNC path(s) is specified, only the computer in the UNC path(s) is searched.
Id	You use the -Id parameter to specify a PowerShell object (or objects) as by the Id value of the object.

Example

To change the name or path of a network share computer object:

```
Set-MMPNetworkShare -Name 'MyNetworkShare' -Path '\\MyShare' -ConnectionId
$sourceconnection.Id
```

Import-MMPPstFile

Use the Import-MmpPstFile cmdlet to import a list of PST file objects from a CSV file (exported from the Website or through Powershell) to be used in MMP cmdlets. The cmdlet will output a list of PST file objects.

Syntax

```
Import-MMPPstFile -FilePath <string> [-ValidateExists] [-Delimiter <string>] [-
Encoding <Encoding>]  [<CommonParameters>]
```

Example

Importing PSTs from a CSV, while validating that they exist, into a new Collection:

Create a new Collection

```
$pstColl = New-MMPCollection -Name "New Psts Collection" -Label "From Exported Psts"
-CollectionType Pst | Add-MMPCollection
```

Import PSTs into the Collection

```
Import-MMPPstFile -FilePath C:\ExportedPsts.csv -ValidateExists |
Add-MMPCollectionMember -Collection $pstColl
```

Importing PSTs from a TSV, specifying that the file is Unicode encoded, into a new Collection:

```
$unicodeEncoding = [System.Text.Encoding]::Unicode
```

```
Import-MMPPstFile -FilePath C:\ExportedPsts.tsv -Encoding $unicodeEncoding -
Delimiter "`t" | Add-MMPCollectionMember -Collection $pstColl
```

Start-MMPPstFileDiscoveryTask

Use the Start-MMPPstFileDiscoveryTask cmdlet to create a task object in MMP.

Syntax

```
Start-MMPPstFileDiscoveryTask -SourceConnectionId <Nullable`1[Guid]>
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask -Name <String> [<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-StartTimeUtc <DateTime>] [<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-RecurEndTimeUtc <Nullable`1[DateTime]>]
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-RecurCount <Nullable`1[Int32]>]
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-RecurDaysOfWeek <RecurDaysOfWeekEnum>]
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-PipelineVariable <String>] [<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask -Computers <Computer[]> [<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-NonRecursive [<SwitchParameter>]]
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-Path <String>] [<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask [-IncludeSystemFiles [<SwitchParameter>]]
[<CommonParameters>]
```

```
Start-MMPPstFileDiscoveryTask -TaskId <Nullable`1[Guid]> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TaskId	Use the TaskId parameter to set the task by GUID for the PST filediscovery task.
SourceConnectionId	Use the SourceConnectionId GUID to specify the source connection for this task.
StartTimeUtc	Optionally use this parameter to set the UTC time of day when the agent will launch the computer discovery task.
RecurEndTimeUtc	Optionally use this parameter to set the UTC end time for this computer discovery task.
RecurCount	Optionally use this parameter to set the number of times this agent will launch the computer discovery task.
RecurDaysOfWeek	Optionally use this parameter to set which days of the week are on the schedule for the file discovery task. Options include Everyday, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday.
Name	Use this parameter to set the name of the file discovery task.
Computers	Use this parameter to specify the computers to be scanned. Network shares can also be entered. Network shares are computer objects for which the type is **NetworkShare**, and can be used as parameters to cmdlets that accept computer objects. The NetworkShare path becomes the location of a computer object.
Path	Use this parameter to list the local paths to be searched for PST files on the computers. A UNC path is not allowed.
NonRecursive	Use this switch to scan only the specified directory.
IncludeSystemFiles	Use this switch to specify if you want to scan system file folders for PST files.

Example

To create a task object in MMP and store the task in a variable, use the following example:

```
$pstFileDiscovery = Start-MMPPstFileDiscoveryTask -Computers $computer -
SourceConnectionId $sourceConnection.Id -Name "Sitraka Pst File Discovery"
```

To get the task above from the MMP database and verify that it has a status of completed, use the following example:

```
Get-MMPTask -id $pstFileDiscovery.id
```

Set-MMPPstFileOwner

Use the Set-MMPPstFileOwner cmdlet to associate a discovered user with a discovered PST file. This is useful for assigning owners to orphaned files, which you can identify with the Get-MMPDiscoveredPstFile when you use the -OrphanedFiles switch.

Syntax

```
Set-MMPPstFileOwner [-PipelineVariable <String>] [<CommonParameters>]

Set-MMPPstFileOwner -RevertToDiscoveredOwner [<SwitchParameter>]
[<CommonParameters>]

Set-MMPPstFileOwner -InputObject <PstFile> [<CommonParameters>]

Set-MMPPstFileOwner -Id <String> [<CommonParameters>]

Set-MMPPstFileOwner -UserId <Guid> [<CommonParameters>]

Set-MMPPstFileOwner -User <User> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
User	Use this parameter to update the user assigned as the owner of the PST file.
UserId	Use this parameter to update the user, by GUID, assigned as the owner of the PST file.
RevertToDiscoveredOwner	Use this switch to return the ownership of this PST file to the owner who was found during the PST file discovery phase.

Example

To update the owner of all the orphaned PST files to an administrative account, use the following example:

```
$User = Get-MMPDiscoveredUser -Email "administrator@sitraka.com"
$AllOphanedFiles = Get-MMPDiscoveredPstFiles -OrphanFiles
Set-MMPPstFileOwner -UserId $User.Id -InputObject $AllOrphanedFiles
```

To reassign a user's pst files to a new user, use the following example:

```
$User = Get-MMPDiscoveredUser -Email "administrator@sitraka.com"
$usersFiles = Get-MMPDiscoveredPstFile -Email $user.TargetEmailAddress
$usersFiles | Set-MMPPstFileOwner -User $User
```

Resolve-MMPSender

Use the Resolve-MMPSender cmdlet to identify the owner of a PST file based upon sent mail.

Syntax

```
Resolve-MMPSender -PstFile <PstFile> [<CommonParameters>]
Resolve-MMPSender [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
PstFile	The PST file for which you are trying to identify the owner.

Basic Example

Assign the PST file for which you are trying to identify the owner:

```
$PstFile = Get-MMPDiscoveredPstFile -OrphanFiles | Select-Object -First 1
```

Run the cmdlet:

```
Resolve-MMPSender -PstFile $pstFile
```

The output of the cmdlet may vary. The example above has 5 different senders. A sender is an email address used to send mail in the Sent Items folder of the PST, as shown in the output below. This cmdlet outputs the senders LegacyExchangeDN, number of items sent, and, if possible, a matching user from the MMP database. The output is displayed in descending order of items sent.

SenderLegacyExchangeDn	Sent Items	User
/O=SHIRE 2010/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=...	7	DellSoftware.PstMigration.DataModels.User
/O=SHIRE 2010/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=...	3	DellSoftware.PstMigration.DataModels.User
/O=SHIRE 2010/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=...	3	DellSoftware.PstMigration.DataModels.User
/O=SHIRE 2010/OU=EXCHANGE ADMINISTRATIVE GROUP (FYDIBOHF23SPDLT)/CN=RECIPIENTS/CN=...	2	DellSoftware.PstMigration.DataModels.Use
Test.Sender@Sitraka.COM	1	

Advanced Example

Pipe the output of the Resolve-MMPSender cmdlet to the Select-Object cmdlet to return the first sender:

```
$sender = Resolve-MMPSender -PstFile $pst | Select-Object -First 1
```

To use the Sender and PST variables to set the owner of the PST file:

```
Set-MMPPstFileOwner -InputObject $PstFile -User $sender.User
```

To verify the change in ownership of the PST file, use one of the following commands:

List the PST files owned by the sender:

```
Get-MMPDiscoveredPstFile -User $sender.User
```

Verify that the sender is the owner of the specified PST file:

```
Get-MMPDiscoveredUser -PstFile $pst
```

Import-MMPUser

Use the Import-MMPUser cmdlet to import a list of user objects from a CSV file (exported from the Website or through Powershell) to be used in MMP cmdlets. The cmdlet will output a list of user objects.

Syntax

```
Import-MMPUser -FilePath <string> [-ValidateExists] [-Delimiter <string>] [-Encoding <Encoding>]  [<CommonParameters>]
```

Example

Import PSTs from a CSV file, while validating that they exist, into a new collection:

Create a new collection:

```
$userColl = New-MMPCollection -Name "New Users Collection" -Label "From Exported Users" -CollectionType User | Add-MMPCollection
```

Import PSTs into the collection:

```
Import-MMPPstFile -FilePath C:\Users.csv -ValidateExists | Add-MMPCollectionMember -Collection $userColl
```

Import PSTs from a TSV, specifying that the file is Unicode encoded, into a new Collection:

```
$unicodeEncoding = [System.Text.Encoding]::Unicode
```

```
Import-MMPUser -FilePath C:\ExportedUsers.tsv -Encoding $unicodeEncoding -Delimiter
"`t" | Add-MMPCollectionMember -Collection $userColl
```

Start-MMPUserDiscoveryTask

Use the Start-MMPUserDiscoveryTask cmdlet to queue a user discovery task for processing by the MMP agents. The OuFilter parameter is used to provide an LDAP filter. Use the StartTime, RecurDaysOfWeek, RecurCount and RecurEndTime parameters to configure the task schedule.

Syntax

```
Start-MMPUserDiscoveryTask -SourceConnectionId <Nullable`1[Guid]>
[<CommonParameters>]

Start-MMPUserDiscoveryTask -Name <String> [<CommonParameters>]

Start-MMPUserDiscoveryTask [-StartTimeUtc <DateTime>] [<CommonParameters>]

Start-MMPUserDiscoveryTask [-RecurEndTimeUtc <Nullable`1[DateTime]>]
[<CommonParameters>]

Start-MMPUserDiscoveryTask [-RecurCount <Nullable`1[Int32]>] [<CommonParameters>]

Start-MMPUserDiscoveryTask [-RecurDaysOfWeek <RecurDaysOfWeekEnum>]
[<CommonParameters>]

Start-MMPUserDiscoveryTask [-PipelineVariable <String>]

[<CommonParameters>]

Start-MMPUserDiscoveryTask -TaskId <Nullable`1[Guid]> [<CommonParameters>]

Start-MMPUserDiscoveryTask -SourceConnectionID <MigrationConnection> [-OuFilter
<string[]>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TaskId	Use the TaskId parameter to set the task by GUID for the user discovery task.
SourceConnectionId	Use the SourceConnectionId GUID to specify the source connection for this task.
StartTimeUtc	Optionally use this parameter to set the UTC time of day when the agent will launch the user discovery task.
RecurEndTimeUtc	Optionally use this parameter to set the UTC end time for this user discovery task.
RecurCount	Optionally use this parameter to set the number of times this agent will launch the user discovery task.
RecurDaysOfWeek	Optionally use this parameter to set which days of the week are on the schedule for the user discovery task. Options include Everyday, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday.
Name	Use this parameter to set the name of the user discovery task.
OuFilter	Use this parameter to filter by organizational unit.

Example

To create a task object in MMP, use the following example. This syntax stores the Task in the "$UserDiscovery" variable.

```
$UserDiscovery = Start-MMPUserDiscoveryTask -SourceConnectionId $sourceConnection.Id
-Name "Sitraka Active Directory User Discovery"
Get-MMPTask -Id $UserDiscovery.Id
```

To get an organizational unit (OU) to filter by when performing a computer discovery task, use the following example:

```
$ou = Get-MMPOrganizationalUnit -Connection $connection | Select -Last 1

Start-MMPUserDiscoveryTask -OuFilter $ou -SourceConnectionId $connection.Id -Name
"User Discovery with Ou Filtering"
```

Get-MMPEvent

Use the Get-MMPEvent cmdlet to retrieve a list of events from the MMP server. You can use the -TaskId or -Task parameter to get events from a specific task.

Syntax

```
Get-MMPEvent [-PipelineVariable <String>] [<CommonParameters>]

Get-MMPEvent -Task <MigrationTask> [<CommonParameters>]

Get-MMPEvent -TaskId <Guid> [<CommonParameters>]

Get-MMPEvent [-Id <String>] [<CommonParameters>]

Get-MMPEvent [-InputObject <MigrationEvent>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Task	Optionally use the Task parameter to find the events associated with that Task.
TaskId	Optionally use TaskId parameter allows you to specify the task by its GUID.

Example

To get a list of all the error events from the database, use the following example:

```
Get-MMPEvent | Where-Object {$_.EventType -eq "ERROR"}
```

Get-MMPTask

Use the Get-MMPTask cmdlet to retrieve a list of tasks from the MMP server. You can use the -Id parameter to get a specific task. You can use the -Schedule, -ScheduleId, -Collection, -CollectionId, -Connection, -ConnectionId, -Event and -EventId parameters to get tasks related to specific objects in the MMP database. You can use the -Type parameter to restrict the list to a specific type of tasks. You can get more details about subtasks with the -Detailed switch.

Syntax

```
Get-MMPTask [-Detailed [<SwitchParameter>]] [<CommonParameters>]

Get-MMPTask [-Type <Nullable`1[TaskType]>] [<CommonParameters>]

Get-MMPTask [-PipelineVariable <String>] [<CommonParameters>]
```

```
Get-MMPTask [-Schedule <MigrationSchedule>] [<CommonParameters>]
Get-MMPTask [-ScheduleId <Guid>] [<CommonParameters>]
Get-MMPTask [-Collection <MigrationCollection>] [<CommonParameters>]
Get-MMPTask [-CollectionId <Guid>] [<CommonParameters>]
Get-MMPTask [-Connection <MigrationConnection>] [<CommonParameters>]
Get-MMPTask [-ConnectionId <Guid>] [<CommonParameters>]
Get-MMPTask [-Event <MigrationEvent>] [<CommonParameters>]
Get-MMPTask [-EventId <Guid>] [<CommonParameters>]
Get-MMPTask [-Id <String>] [<CommonParameters>]
Get-MMPTask [-InputObject <MigrationTask>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Event	Use the Event parameter to retrieve the list of tasks associated with that event.
EventId	Use the EventId parameter to retrieve the list of tasks associated with that event GUID.
Collection	Use the Collection parameter to retrieve the list of tasks associated with that collection.
CollectionId	Use the Collection parameter to retrieve the list of tasks associated with that collection GUID.
Connection	Use the Connection parameter to retrieve the list of tasks associated with that connection.
ConnectionId	Use the Connection parameter to retrieve the list of tasks associated with that connection GUID.
Detailed	Use the Detailed parameter when you have a task and get details on the task so that you can get associated subtasks.
Schedule	Use the Schedule parameter to retrieve the list of tasks associated with that schedule.
ScheduleId	Use the Schedule parameter to retrieve the list of tasks associated with that schedule GUID.
Type	Use the Type parameter to restrict your results to a specific type of task. Valid values are Migration, UserDiscovery, ComputerDiscovery, and PstFileDiscovery.

Example

To get a specific task from the database , use the following example:

```
Get-MMPTask -Id $TaskId
```

To get the detailed migration task information associated with a certain schedule, use the following example:

```
Get-MMPTask -ScheduleId $Schedule.Id -Detailed
```

Get-MMPOrganizationalUnit

Use the Get-MMPOrganizationalUnit cmdlet to retrieve a list of distinguished names (DN) from Active Directory to limit the results returned by the **Start-MMPComputerDiscoveryTask** and **Start-MMPUserDiscoveryTask** cmdlets.

Syntax

```
Get-MMPOrganizationalUnit  [<CommonParameters>]
Get-MMPOrganizationalUnit -Connection <MigrationConnection>  [<CommonParameters>]
Get-MMPOrganizationalUnit -ConnectionId <guid>  [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Connection	Use this switch if you want to retrieve the source connections.
OuFilter	Use this parameter to filter by organizational unit.

Example

To get an organizational unit (OU) to filter by when performing a computer discovery task, use the following example:

```
$ou = Get-MMPOrganizationalUnit -Connection $connection | Select -Last 1
Start-MMPComputerDiscoveryTask -OuFilter $ou -SourceConnectionId $connection.Id -
Name "Computer Discovery with Ou Filtering"
```

To get an organizational unit (OU) to filter by when performing a computer discovery task, use the following example:

```
$ou = Get-MMPOrganizationalUnit -Connection $connection | Select -Last 1
Start-MMPUserDiscoveryTask -OuFilter $ou -SourceConnectionId $connection.Id -Name
"User Discovery with Ou Filtering"
```

Migration

The Migration section provides a scenario for performing a migration, followed by detailed cmdlets.

Scenario

The following section lists the cmdlets required for performing a migration. The cmdlets are listed in order of execution for the migration process:

1. Retrieve a list of the users discovered by MMP: Get-MMPDiscoveredUser
2. Retrieve a list of the PST files discovered by MMP: Get-MMPDiscoveredPstFile
3. Create a new collection: New-MMPCollection
4. Add the collection to the MMP database: Add-MMPCollection
5. Retrieve the list of collections you created from the MMP database: Get-MMPCollection
6. Add a user as a member of the collection: Add-MMPCollectionMember
7. Create a new target connection: New-MMPConnection
8. Retrieve a list of connections from the MMP server: Get-MMPConnection
9. Create a new credential for the connection: New-MMPCredential
10. Add the credential to the MMP database: Add-MMPCredential
11. Retrieve a list of credentials from the MMP server: Get-MMPCredential

12 Queue a migration task for processing by the MMP agents: Start-MMPMigrationTask

13 Retrieve a task from the MMP server: Get-MMPTask

For detailed cmdlets and examples and additional cmdlets, see the Migration cmdlets section below.

Migration cmdlets

Get-MMPEvent

Use the Get-MMPEvent cmdlet to retrieve a list of events from the MMP server. You can use the -TaskId or -Task parameter to get events from a specific task.

Syntax

```
Get-MMPEvent [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPEvent -Task <MigrationTask> [<CommonParameters>]
Get-MMPEvent -TaskId <Guid> [<CommonParameters>]
Get-MMPEvent [-Id <String>] [<CommonParameters>]
Get-MMPEvent [-InputObject <MigrationEvent>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Task	Optionally use the Task parameter to find the events associated with that Task.
TaskId	Optionally use TaskId parameter allows you to specify the task by its GUID.

Example

To an event based on a specific task, use the following example:

```
$task = Get-MMPTask | Where-Object Name -like "Sitraka Computer Discovery Task"
$event = Get-MMPEvent -Task $Task
```

Remove-MMPEvent

Use the Remove-MMPEvent cmdlet to remove an event from the MMP database. You can use the -Id to remove a specifc event.

Syntax

```
Remove-MMPEvent [-PipelineVariable <String>] [<CommonParameters>]
Remove-MMPEvent -Id <String> [<CommonParameters>]
Remove-MMPEvent -InputObject <MigrationEvent> [<CommonParameters>]
```

Example

To remove events from the database, use the following example:

```
$Event = Get-MMPEvent | Where-Object -Property ObjectName -EQ "My Event Name"
Remove-MMPEvent -InputObject $Event
```

Set-MMPEvent

Use the Set-MMPEvent cmdlet to update an event on the MMP server.

Syntax

```
Set-MMPEvent [-EventState <String>] [<CommonParameters>]
Set-MMPEvent [-PipelineVariable <String>] [<CommonParameters>]
Set-MMPEvent -Id <String> [<CommonParameters>]
Set-MMPEvent -InputObject <MigrationEvent> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
EventState	Use the EventState parameter to specify the new state of the event.

Example

To promote all the warnings to errors, use the following example:

```
$events = Get-MMPEvent | Where-Object {$_.EventType -eq "WARNING"}
Set-MMPEvent -InputObject $events -EventState “ERROR”
```

New-MMPCollection

Use the New-MMPCollection cmdlet to create a new user collection. You can add new collections to the MMP database with the Add-MMPCollection cmdlet.

Syntax

```
New-MMPCollection -Name <String> [<CommonParameters>]
New-MMPCollection -Label <String> [<CommonParameters>]
New-MMPCollection [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to give your collection a name.
Label	Use the Label parameter to give your collection a descriptive label.

Example

To create a new collection, use the following example:

```
$collection = New-MMPCollection -Name "Sitraka PST Migration 1" -Label
"Administrators psts"
```

Add-MMPCollection

Use the Add-MMPCollection cmdlet to add a collection of users to the MMP database. You can create collections with the New-MMPCollection cmdlet.

Syntax

```
Add-MMPCollection [-PipelineVariable <String>] [<CommonParameters>]
Add-MMPCollection -InputObject <MigrationCollection> [<CommonParameters>]
```

Example

To add the collection to the MMP database, use the following example:

```
$collection = New-MMPCollection -Name "Sitraka PST Migration 1" -Label
"Administrators psts"
$Collection = $collection | Add-MMPCollection
```

Get-MMPCollection

Use the Get-MMPCollection cmdlet to get a list of collections from the MMP server. You can use the -Id parameter to get a specific collection. You can use the -Connection, -ConnectionId, -Task, -TaskId, -User, and -UserId parameters can be used to get collections related to specific objects in the MMP database.

Syntax

```
Get-MMPCollection [-ReturnType <ReturnType>] [<CommonParameters>]
Get-MMPCollection [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPCollection [-User <User>] [<CommonParameters>]
Get-MMPCollection [-UserId <Guid>] [<CommonParameters>]
Get-MMPCollection [-Task <MigrationTask>] [<CommonParameters>]
Get-MMPCollection [-TaskId <Guid>] [<CommonParameters>]
Get-MMPCollection [-Connection <MigrationConnection>] [<CommonParameters>]
Get-MMPCollection [-ConnectionId <Guid>] [<CommonParameters>]
Get-MMPCollection [-Id <String>] [<CommonParameters>]
Get-MMPCollection [-InputObject <MigrationCollection>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
User	Specifying a user with the User parameter will retrieve the list of collections containing that user.
UserId	Specifying a user with the UserId GUID will retrieve the list of collections containing that user.
Task	Specifying a task with the Task parameter will retrieve the list of collections associated with that task.
TaskId	Specifying a task with the TaskId GUID will retrieve the list of collections associated with that task.
ReturnType	Use the ReturnType parameter to retrieve hidden, not hidden, or all collections.

Example

To retrieve a collection from the database, use the following example:

```
$Collection = Get-MMPCollection -Id $Collection.Id
```

Set-MMPCollection

Use the Set-MMPCollection cmdlet to update a user collections on the MMP server.

Syntax

```
Set-MMPCollection [-Name <String>] [<CommonParameters>]
Set-MMPCollection [-Label <String>] [<CommonParameters>]
Set-MMPCollection [-Hidden <Nullable`1[Boolean]>] [<CommonParameters>]
Set-MMPCollection [-PipelineVariable <String>] [<CommonParameters>]
Set-MMPCollection -Id <String> [<CommonParameters>]
Set-MMPCollection -InputObject <MigrationCollection> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Optionally use this parameter to update the name of the collection.
Label	Optionally use this parameter to update the label of the collection.
Hidden	Optionally use this parameter to update the hidden attribute of a collection.

Example

To modify the name on an existing collection use the following example:

```
$Collection = Get-MMPCollection | Where-Object Name -eq "My Collection"
Set-MMPCollection -Id $collection.Id  -Name 'New Name'
To set a collection to hidden in the database, use the following example:
$Collection = Get-MMPCollection | Where-Object Name -eq "My Collection"
Set-MMPCollection -InputObject $Collection -Hidden $True
```

Add-MMPCollectionMember

Use the Add-MMPCollectionMember cmdlet to add a user to an MMP collection. You can get a list of discovered users from the MMP database using the Get-MMPDiscoveredUser cmdlet. You can get a list of discovered PST files from the MMP database using the Get-MMPDiscoveredPSTFile cmdlet.

User collections and PST file collections are mutually exclusive. If you attempt to add a PST file to a user collection or vice versa, an error is generated.

Syntax

```
Add-MMPCollectionMember -User <User> [<CommonParameters>]
Add-MMPCollectionMember -PstFile <PstFile> [<CommonParameters>]
Add-MMPCollectionMember -Collection <MigrationCollection> [<CommonParameters>]
Add-MMPCollectionMember [-PipelineVariable <String>] [<CommonParameters>]
Add-MMPCollectionMember -CollectionId <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
User	Lets you to specify the list of discovered users that you want to add to your collection.
PST File	Lets you to specify the list of discovered PSTs that you want to add to your collection.
CollectionId	The CollectionId parameter allows you to specify the collection by its ID string. You must use either the -CollectionId or the -Collection parameter.
Collection	The Collection parameter allows you set the MigrationCollection as a variable.

Example

To add a user or PST file as a member of the collection you created, use the following example:

```
$User = Get-MMPDiscoveredUser -Email "John.Smith@sitraka.com"
$UserCollection = Get-MMPCollection | Where-Object {$_.Name -eq 'collection name'}
Add-MMPCollectionMember -User $User -Collection $UserCollection
$FileCollection = Get-MMPCollection | Where-Object {$_.Name -eq 'collection name'}
Add-MMPCollectionMember -PstFile $PstFile1 -Collection $FileCollection
```

Get-MMPCollectionMember

Use the Get-MMPCollectionMember cmdlet to get a list of members belonging to an MMP collection. You will need to pass a collection object that you retrieved from the Get-MMPCollection cmdlet.

Syntax

```
Get-MMPCollectionMember [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPCollectionMember -Collection <MigrationCollection> [<CommonParameters>]
Get-MMPCollectionMember -CollectionId <String> [<CommonParameters>]
```

Example

To get all collection members from a collection, run the following command:

```
$Collection = Get-MMPCollection | Where-Object -Property Name -EQ "My Collection"
Get-MMPCollectionMember -CollectionId $Collection.Id
```

Remove-MMPCollectionMember

Use the Remove-MMPCollectionMember cmdlet to remove members belonging from an MMP collection. You can get discovered users with the Get-MMPDiscoveredUser cmdlet.

Syntax

```
Remove-MMPCollectionMember [-PipelineVariable <String>] [<CommonParameters>]
Remove-MMPCollectionMember -User <User> [<CommonParameters>]
Remove-MMPCollectionMember -CollectionId <String> [<CommonParameters>]
Remove-MMPCollectionMember -Collection <MigrationCollection> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
User	Use the User parameter to specify the users to be removed from the collection.

Example

To remove a collection member from a collection, run the following command:

```
$Collection = Get-MMPCollection | Where-Object -Property Name -EQ "My Collection"
$User = Get-MMPDiscoveredUser -Email "John.Smith@sitraka.com"
Remove-MMPCollectionMember -CollectionId $collection.Id -User $user
```

Set-MMPCollectionMember

Use the Set-MMPCollection cmdlet to replace the memberships of a collection on the MMP server.

Syntax

```
Set-MMPCollectionMember -User <User> [<CommonParameters>]
Set-MMPCollectionMember -PSTFile <PSTFile> [<CommonParameters>]
Set-MMPCollectionMember -CollectionId <String> [<CommonParameters>]
Set-MMPCollectionMember [-PipelineVariable <String>] [<CommonParameters>]
Set-MMPCollectionMember -Collection <MigrationCollection> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
User	Use the User parameter to specify the users to be updated in the collection.
PST File	The PST file parameter allows you to specify the list of discovered PSTs that you want to add to your collection.
CollectionId	The CollectionId parameter allows you to specify the collection by its ID string. You must use either the -CollectionId or the -Collection parameter.
Collection	The Collection parameter allows you set the MigrationCollection as a variable.

Example

To modify the users that belong to a collection, use the following example:

```
$User = Get-MMPDiscoveredUser -Email "John.Smith@sitraka.com"
```

```
$UserCollection = Get-MMPCollection | Where-Object {$_.Name -eq 'collection name'}
$Users | Set-MMPCollectionMember -Collection $UserCollection
```

To modify the files that belong to a collection, use the following example:

```
$Files = Get-MMPDiscoveredPstFile | Where-Object Name -like "*.pst*"
$FileCollection = Get-MMPCollection | Where-Object {$_.Name -eq 'collection name'}
$Files | Set-MMPCollectionMember -Collection $FileCollection
```

New-MMPConnection

Use the New-MMPConnection cmdlet to create a source or target connection. You can use the Add-MMPConnection cmdlet to add a new connection to the MMP database.

Syntax

```
New-MMPConnection -Name <String> [<CommonParameters>]
New-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
New-MMPConnection -SourceConnection [<SwitchParameter>] [<CommonParameters>]
New-MMPConnection -TargetConnection [<SwitchParameter>] [<CommonParameters>]
New-MMPConnection -ServerKind <ServerType> [<CommonParameters>]
New-MMPConnection [-NonSsl [<SwitchParameter>]] [<CommonParameters>]
New-MMPConnection -UseAutoDiscovery [<SwitchParameter>] [<CommonParameters>]
New-MMPConnection -EwsUrl <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to set the name of your new connection.
NonSsl	Use this optional parameter if you do not want to use SSL to connect to the MMP web services over HTTP. By default, HTTPS is used.
TargetConnection	Use this switch if you want a new target connection.
SourceConnection	Use this switch if you want a new source connection.
ServerKind	Use the ServerKind parameter to set the type of target server. This parameter only applies to new target connections. Valid values are Exchange2010, Exchange2013, Exchange2016 and Office365.
UseAutoDiscovery	Use this switch to enable the use of autodiscovery on the Exchange target server. This parameter only applies to new target connections.
EwsUrl	Use the EwsUrl parameter to set the URL of the Exchange Web Service on the target Exchange server. This parameter only applies to new target connections.

Example

To create the connection object, use the following example:

```
$sourceConnection = New-MMPConnection -Name "Sitraka Source Connection"
-SourceConnection
```

Add-MMPConnection

Use the Add-MMPConnection cmdlet to add a source or target connection to the MMP database. You can create a connection configuration with the New-MMPConnection cmdlet.

Syntax

```
Add-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
Add-MMPConnection -InputObject <MigrationConnection> [<CommonParameters>]
```

Example

To add the connection to the MMP Database, use the following example.

```
$sourceConnection = New-MMPConnection -Name "Sitraka Source Connection"
-SourceConnection$sourceConnection = $sourceConnection | Add-MMPConnection
```

Get-MMPConnection

Use the Get-MMPConnection cmdlet to retrieve a list of connections from the MMP server. You can use the -Connection, -ConnectionId, -Task, -TaskId, -User, and -UserId parameters to get connections related to specific objects in the MMP database.

Syntax

```
Get-MMPConnection [-ReturnType <ReturnType>] [<CommonParameters>]
Get-MMPConnection [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPConnection [-TargetConnection [<SwitchParameter>]] [<CommonParameters>]
Get-MMPConnection [-SourceConnection [<SwitchParameter>]] [<CommonParameters>]
Get-MMPConnection [-Collection <MigrationCollection>] [<CommonParameters>]
Get-MMPConnection [-CollectionId <Guid>] [<CommonParameters>]
Get-MMPConnection [-Task <MigrationTask>] [<CommonParameters>]
Get-MMPConnection [-TaskId <Guid>] [<CommonParameters>]
Get-MMPConnection [-Id <String>] [<CommonParameters>]
Get-MMPConnection [-InputObject <MigrationConnection>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TargetConnection	Use this switch if you want to retrieve the target connections.
SourceConnection	Use this switch if you want to retrieve the source connections.
Collection	Use the Collection parameter to retrieve the list of connections associated with that collection.
CollectionId	Use the Collection parameter to retrieve the list of connections associated with that collection GUID.
Task	Use the Task parameter to retrieve the list of connections associated with that task.
TaskId	Use the Task parameter to retrieve the list of connections associated with that task GUID.
ReturnType	Use the ReturnType parameter to retrieve hidden, not hidden, or all connections.

Example

To get the connection from the MMP Database, use the following example:

```
$sourceConnection = Get-MMPConnection -SourceConnection | Where-Object -Property
Name -eq "Sitraka Source Connection"
```

Set-MMPSourceConnection

Use the Set-MMPSourceConnection cmdlet to set a source connection on the MMP server.

Syntax

```
Set-MMPSourceConnection -Id <string> [-Name <string>] [-Hidden <bool>]
[<CommonParameters>]
```

```
Set-MMPSourceConnection -InputObject <MigrationConnection> [-Name <string>] [-Hidden
<bool>]  [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to update the name of your connection.
Hidden	Optionally use this parameter to update the hidden attribute of a connection.

Example

To set a source connection to hidden in the database, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Source Connection"
```

```
Set-MMPSourceConnection -InputObject $Connection -Hidden $True
```

Set-MMPTargetConnection

Use the Set-MMPTargetConnection cmdlet to set a target connection on the MMP server.

Syntax

```
Set-MMPTargetConnection -Id <string> [-ServerKind <ServerType> {Exchange2010 |
Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-
UseAutoDiscover] [-Name <string>] [-Hidden <bool>] [<CommonParameters>]
```

```
Set-MMPTargetConnection -Id <string> [-ServerKind <ServerType> {Exchange2010 |
Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>] [-EwsUrl
<string>] [-Name <string>] [-Hidden <bool>] [<CommonParameters>]
```

```
Set-MMPTargetConnection -InputObject <MigrationConnection> [-ServerKind <ServerType>
{Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>]
[-EwsUrl <string>] [-Name <string>] [-Hidden <bool>] [<CommonParameters>]
```

```
Set-MMPTargetConnection -InputObject <MigrationConnection> [-ServerKind <ServerType>
{Exchange2010 | Exchange2013 | Exchange2016 | Office365 | Unknown}] [-UseSsl <bool>]
[-UseAutoDiscover] [-Name <string>] [-Hidden <bool>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Name	Use the Name parameter to update the name of your connection.
ServerKind	Use the ServerKind parameter to set the type of target server. This parameter only applies to new target connections. Valid values are Exchange2010, Exchange2013, Exchange2016 and Office365.
UseAutoDiscover	Use this switch to update the use of autodiscovery on the Exchange target server. This parameter only applies to new target connections.
EwsUrl	Use the EwsUrl parameter to update the URL of the Exchange Web Service on the target Exchange server.This parameter only applies to new target connections.
Hidden	Optionally use this parameter to update the hidden attribute of a connection.
UseSsl	Use a Secure Sockets Layer certificate.

Example

To change the server type of a connection from Exchange 2010 to Exchange 2013, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
Set-MMPTargetConnection -Id $TargetConnection.Id -ServerKind Exchange2013
```

To update a connection to use Autodiscover, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
Set-MMPTargetConnection -Id $TargetConnection.Id -UseAutoDiscover
```

To update a connection to use EWS, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
Set-MMPTargetConnection -Id $TargetConnection.Id -EwsUrl Exchange2010.sitraka.com
```

To set a target connection to hidden in the database, use the following example:

```
$Connection = Get-MMPConnection | Where-Object Name -eq "My Target Connection"
Set-MMPTargetConnection -InputObject $Connection -Hidden $True
```

Confirm-MMPConnection

Use the Confirm-MMPConnection cmdlet to validate source and target connections.

Syntax

```
Confirm-MMPConnection -Credentials <Credential> -Connection <MigrationConnection>
[-Mailbox <string>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Connection	Use the Connection parameter to retrieve the list of tasks associated with that connection.
Credentials	Use this parameter to specify the PowerShell credential object.
Mailbox	

Example

To validate a source connection, use the following example:

```
$connection = Get-MMPConnection -SourceConnection

$credentials = Get-MMPCredential -ConnectionId $connection.Id

Confirm-MMPConnection -Credentials $credentials -Connection $connection -Mailbox
"sitraka.admin@sitraka.com"
```

New-MMPCredential

Use the New-MMPCredential cmdlet to create a credential for a connection.

Syntax

```
New-MMPCredential -ConnectionId <Guid> [<CommonParameters>]

New-MMPCredential -Credentials <PSCredential> [<CommonParameters>]

New-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Use the ConnectionId GUID to specify the connection for this new credential.
Credentials	Use this parameter to specify the PowerShell credential object.

Example

To create a new target credential, use the following example:

```
$cred = New-MMPCredential -Credentials (Get-Credential) -ConnectionId
$sourceConnection.Id
```

Add-MMPCredential

Use the Add-MMPCredential cmdlet to add a credential to the MMP database.

Syntax

```
Add-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]

Add-MMPCredential -InputObject <Credential> [<CommonParameters>]
```

Example

To create a new credential with user input and add it to the MMP database, use the following example:

```
$cred = New-MMPCredential -Credentials (Get-Credential) -ConnectionId
$sourceConnection.Id
$cred = $cred | Add-MMPCredential
```

Get-MMPCredential

Use the Get-MMPCredential cmdlet to retrieve a list of credentials from the MMP server. You can use the -ConnectionId parameter to get the credentials for a specific connection.

Syntax

```
Get-MMPCredential [-ConnectionId <Guid>] [<CommonParameters>]

Get-MMPCredential [-PipelineVariable <String>] [<CommonParameters>]

Get-MMPCredential [-Id <String>] [<CommonParameters>]

Get-MMPCredential [-InputObject <Credential>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
ConnectionId	Specifying a connection with the ConnectionId GUID will retrieve the list of credentials associated with that connection.

Example

To retrieve the credential for the connection, use the following example:

```
$cred = Get-MMPCredential -ConnectionId $targetConnection.Id
```

Get-MMPDiscoveredComputer

Use the Get-MMPDiscoveredComputer cmdlet to retrieve a list of the computers discovered by MMP. You can use the -Id parameter to retrieve a specific computer.

Syntax

```
Get-MMPDiscoveredComputer [-PipelineVariable <String>] [<CommonParameters>]

Get-MMPDiscoveredComputer [-Id <String>] [<CommonParameters>]

Get-MMPDiscoveredComputer [-InputObject <Computer>] [<CommonParameters>]
```

Example

The task to discover PST files requires at least one computer to run on. To retrieve computers from the MMP Database, use the following example:

```
$computers = Get-MMPDiscoveredComputer
```

Get-MMPDiscoveredPstFile

Use the Get-MMPDiscoveredPstFile cmdlet to retrieve a list of the PST files discovered by MMP. You can use the -Id parameter to retrieve a specific file. You can use the -User parameter to find files related to specific users. You can use the -Email, -SamAccountName and -UPN parameters to get files with specific attributes. You can use the -OprhanFiles parameter to find PST files that could not be associated with a specific owner in the MMP database.

Syntax

```
Get-MMPDiscoveredPstFile [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-Email <String>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-SamAccountName <String>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-OrphanFiles [<SwitchParameter>]] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-UPN <String>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [[-User] <User>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [[-Computer] <Computer>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-Id <String>] [<CommonParameters>]
Get-MMPDiscoveredPstFile [-InputObject <PstFile>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Email	Specifying an email address with the Email parameter will retrieve the list of PST files associated with that address.
SamAccountName	Specifying an account name with the SamAccountName parameter will retrieve the list of PST files associated with that identity.
UPN	Specifying a UPN (user principal name) with the UPN parameter will retrieve the list of PST files associated with that identity.
OrphanFiles	Using the the OrphanFiles switch will retrieve the list of PST files that are not associated with owners in the MMP database.
User	Specifying a user will retrieve all of the PST files associated with that user.

Example

To retrieve all of the discerned PST files that are owned by a specific user, use the following example:

```
$User = Get-MMPDiscoveredUser -Email "John.Smith@sitraka.com"
Get-MMPDiscoveredPstFile -User $User
```

Get-MMPDiscoveredUser

Use the Get-MMPDiscoveredUser cmdlet to retrieve a list of the users discovered by MMP. You can use the -Id parameter to retrieve a specific user. You can use the -PstFile, -PstFileId, -Collection, - and -CollectionId parameters to get users related to specific objects in the MMP database. You can use the -Email, -MailboxEnabled, -Ou, -SamAccountName and -UPN parameters to get files with specific attributes.

Syntax

```
Get-MMPDiscoveredUser [-MailboxEnabled [<SwitchParameter>]] [<CommonParameters>]
Get-MMPDiscoveredUser [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPDiscoveredUser -PstFile <PstFile> [<CommonParameters>]
Get-MMPDiscoveredUser -PstFileId <Guid> [<CommonParameters>]
Get-MMPDiscoveredUser -Collection <MigrationCollection> [<CommonParameters>]
Get-MMPDiscoveredUser -CollectionId <Guid> [<CommonParameters>]
Get-MMPDiscoveredUser -Computer <Computer> [<CommonParameters>]
Get-MMPDiscoveredUser -ComputerId <Guid> [<CommonParameters>]
Get-MMPDiscoveredUser -Email <String> [<CommonParameters>]
Get-MMPDiscoveredUser -Ou <String> [<CommonParameters>]
Get-MMPDiscoveredUser -SamAccountName <String> [<CommonParameters>]
Get-MMPDiscoveredUser -UPN <String> [<CommonParameters>]
Get-MMPDiscoveredUser [-Id <String>] [<CommonParameters>]
Get-MMPDiscoveredUser [-InputObject <User>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Email	Specifying an email address with the Email parameter will retrieve the user associated with that address.
MailboxEnabled	Specifying the MailboxEnabled switch will only return users who are mailbox enabled on the target e-mail server.
Ou	Specifying an OU (Organization Unit) with the Ou parameter will retrieve the list of users found in that Active Directory OU.
SamAccountName	Specifying an account name with the SamAccountName parameter will retrieve the user associated with that identity.
UPN	Specifying a UPN (user principal name) with the UPN parameter will retrieve the user associated with that identity.
Collection	Specifying a collection with the Collection parameter will retrieve the list of users who are members of that collection.
CollectionId	Specifying a collection with the CollectionId GUID will retrieve the list of users who are members of that collection.
Computer	Specifying a computer with the Computer parameter will retrieve the list of users who are owners of the discovered PST files on that computer.
Computer ID	Specifying a computer with the ComputerId GUID will retrieve the list of users who are owners of the discovered PST files on that computer.
PstFile	Specifying a PST file with the PstFile parameter will retrieve the owner of that file.
PstFileId	Specifying a PST file with the PstFileId GUID will retrieve the owner of that file.

Example

To retrieve a specific user from the MMP database, use the following example:

```
$User1 = Get-MMPDiscoveredUser -Email "administrator@sitraka.com"
```

Set-MMPDiscoveredUser

Use the Set-MMPDiscoveredUser cmdlet to set the target email address for a discovered user.

Syntax

```
Set-MMPDiscoveredUser [-PipelineVariable <String>] [<CommonParameters>]
Set-MMPDiscoveredUser -TargetEmailAddress <String> [<CommonParameters>]
Set-MMPDiscoveredUser [-DoNotMailboxEnable [<SwitchParameter>]][<CommonParameters>]
Set-MMPDiscoveredUser [-InputObject <User>] [<CommonParameters>]
Set-MMPDiscoveredUser -RevertToDiscoveredTargetEmailAddress [<SwitchParameter>]
[<CommonParameters>]
Set-MMPDiscoveredUser -Id <String> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TargetEmailAddress	The target email address to assign to the specified user.
DoNotMailboxEnable	Do not mailbox enable when the target email address is provided.
RevertToDiscoveredTargetEmailAddress	Reverts to the target email address for the specified user.

Example

Set the TargetEmailAddress for a given user migrating to on-premises Exchange.

```
$user1 = Get-MMPDiscoveredUser -Email John.Doe@sitraka.com
Set-MMPDiscoveredUser -InputObject $user1 -TargetEmailAddress
"William.Smith@sitraka.com"
```

Set the TargetEmailAddress for a given user migrating to O365.

```
$user1 = Get-MMPDiscoveredUser -Email John.Doe@sitraka.com
Set-MMPDiscoveredUser -InputObject $user1 -TargetEmailAddress
"William.Smith@sitraka.onmicrosoft.com"
```

Set the TargetEmailAddress for a given user migrating to on-premises Exchange and **`DoNotMailboxEnable`** when the target email address is provided.

```
$user1 = Get-MMPDiscoveredUser -Email John.Doe@sitraka.com
Set-MMPDiscoveredUser -InputObject $user1 -TargetEmailAddress
"William.Smith@sitraka.com" -DoNotMailboxEnable
```

Revert the TargetEmailAddress for a user to the original target email address discovered in Active Directory.

```
$user1 = Get-MMPDiscoveredUser -SamAccountName "JDOE"
$user1 | Set-MMPDiscoveredUser -RevertToDiscoveredTargetEmailAddress
```

Revert changes to the TargetEmailAddress for multiple users.

```
$SalesOU = Get-MMPDiscoveredUser -Ou "Sales"
$SalesOU | Set-MMPDiscoveredUser -RevertToDiscoveredTargetEmailAddress
```

Get-MMPMigratedPstFile

Use the Get-MMPMigratedPstFile cmdlet to retrieve a list of the PST files migrated with MMP. You can use the -Id parameter to specify a specific file.

Syntax

```
Get-MMPMigratedPstFile [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPMigratedPstFile -Task <MigrationTask> [<CommonParameters>]
Get-MMPMigratedPstFile -TaskId <Guid> [<CommonParameters>]
Get-MMPMigratedPstFile [-Id <String>] [<CommonParameters>]
Get-MMPMigratedPstFile [-InputObject <PstFile>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Task	Optionally use the Task parameter to find the events associated with that Task.
TaskId	Optionally use TaskId parameter allows you to specify the task by its GUID.

Example

To retrieve all of the migrated PST files, use the following example:

```
$AllMigratedFiles = Get-MMPDiscoveredPstFiles
```

Start-MMPMigrationTask

Use the Start-MMPMigrationTask cmdlet to queue a migration task for processing by the MMP agents. Use the MigrateToSubfolder, MigrateToArchive, MigrateEmail, MigrateCalendar, MigrateTasks, MigrateContacts, ExcludeDeleted, ExcludeJunk, ExcludeSent, ExcludeSpecificFolders, MigrateEmailBefore and MigrateEmailAfter parameters to configure the migration process. Use the StartTime, RecurDaysOfWeek, RecurCount, and RecurEndTime parameters to configure the task schedule.

Syntax

```
Start-MMPMigrationTask -Name <String> [<CommonParameters>]
Start-MMPMigrationTask [-StartTimeUtc <DateTime>] [<CommonParameters>]
Start-MMPMigrationTask [-RecurEndTimeUtc <Nullable`1[DateTime]>]
[<CommonParameters>]
Start-MMPMigrationTask [-RecurCount <Nullable`1[Int32]>] [<CommonParameters>]
Start-MMPMigrationTask [-RecurDaysOfWeek <RecurDaysOfWeekEnum>]
[<CommonParameters>]
Start-MMPMigrationTask [-PipelineVariable <String>] [<CommonParameters>]
Start-MMPMigrationTask -CollectionId <Nullable`1[Guid]> [<CommonParameters>]
Start-MMPMigrationTask -TargetConnectionId <Nullable`1[Guid]> [<CommonParameters>]
Start-MMPMigrationTask [-MigrateToSubfolder [<SwitchParameter>]]
[<CommonParameters>]
Start-MMPMigrationTask [-MigrateAssociatedItems [<SwitchParameter>]]
[<CommonParameters>]
Start-MMPMigrationTask [-MigrateToArchive [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-MigrateEmail [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-MigrateCalendar [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-MigrateTasks [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-MigrateContacts [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-ExcludeDeleted [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-ExcludeJunk [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-ExcludeSent [<SwitchParameter>]] [<CommonParameters>]
Start-MMPMigrationTask [-ExcludeSpecificFolders <List`1[String]>]
[<CommonParameters>]
Start-MMPMigrationTask [-MigrateEmailBeforeUtc <Nullable`1[DateTime]>]
[<CommonParameters>]
Start-MMPMigrationTask [-MigrateEmailAfterUtc <Nullable`1[DateTime]>]
[<CommonParameters>]
Start-MMPMigrationTask -TaskId <Nullable`1[Guid]> [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
TargetConnectionId	Use the TargetConnectionId GUID to set the target connection for this task.
TaskId	Use the TaskId parameter to set the task by GUID for the migration task.
StartTimeUtc	Optionally use this parameter to set the UTC time of day when the agent will launch the migration task.
RecurEndTimeUtc	Optionally use this parameter to set the UTC end time for this migration task.
RecurCount	Optionally use this parameter to set the number of times this agent will launch the migration task.
RecurDaysOfWeek	Optionally use this parameter to set which days of the week are on the schedule for the migration task. Options include Everyday, Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, or Saturday.
Name	Use this parameter to set the name of the migration task.
MigrateAssociatedItems	Use this parameter to migrate hidden messages.

PARAMETER	DESCRIPTION
MigrateCalendar	Use this switch if you want to migrate calendar items from the PST file to the target.
MigrateContacts	Use this switch if you want to migrate contacts and personal distribution lists from the PST file to the target.
MigrateEmail	Use this switch if you want to migrate e-mail items from the PST file to the target.
MigrateTasks	Use this switch if you want to migrate task (to-do) items from the PST file to the target.
MigrateToSubfolder	Use this switch if you want to migrate content from the PST file into a subfolder on the target.
MigrateToArchive	Use this switch if you want to migrate content from the PST file into the Personal Archive folder on the target server.
ExcludeDeleted	Use this switch if you do not want to migrate items from the Deleted Items folder of the PST file.
ExcludeJunk	Use this switch if you do not want to migrate items from the Junk E-Mail folder of the PST file.
ExcludeSent	Use this switch if you do not want to migrate items from the Sent Items folder of the PST file.
ExcludeSpecificFolders	Use this optional parameter to list folders that you do not want to migrate items from the PST file.
MigrateEmailAfterUtc	Use this optional parameter to only migrate email received after a specific date and time.
MigrateEmailBeforeUtc	Use this optional parameter to only migrate email received before a specific date and time.

Example

To queue a migration task for processing by the MMP agents, use the following example:

```
$sourceConnection = Get-MMPConnection -SourceConnection | Where-Object -Property
Name -eq "Sitraka Source Connection"

$pstMigration = Start-MMPMigrationTask -Name "Sitraka PST Migration 1" -CollectionId
$collection.id -TargetConnectionId $targetConnection.id
```

Stop-MMPMigrationTask

Use the Stop-MMPMigrationTask cmdlet to halt a migration task that is being or would be processed by the MMP agents.

Syntax

```
Stop-MMPMigrationTask [-PipelineVariable <String>] [<CommonParameters>]
Stop-MMPMigrationTask -Id <String> [<CommonParameters>]
Stop-MMPMigrationTask -InputObject <MigrationTask> [<CommonParameters>]
```

Example

To stop a task when, use the following example:

```
$MigrationTask = Get-MMPTask | Where-Object -Property Name -EQ "My Migration Task"
Stop-MMPMigrationTask -Id $MigrationTask.Id
```

Get-MMPTask

Use the Get-MMPTask cmdlet to retrieve a list of tasks from the MMP server. You can use the -Id parameter to get a specific task. You can use the -Schedule, -ScheduleId, -Collection, -CollectionId, -Connection, -ConnectionId, -Event and -EventId parameters to get tasks related to specific objects in the MMP database. You can use the -Type parameter to restrict the list to a specific type of tasks. You can get more details about subtasks with the -Detailed switch.

Syntax

```
Get-MMPTask [-Detailed [<SwitchParameter>]] [<CommonParameters>]
Get-MMPTask [-Type <Nullable`1[TaskType]>] [<CommonParameters>]
Get-MMPTask [-PipelineVariable <String>] [<CommonParameters>]
Get-MMPTask [-Schedule <MigrationSchedule>] [<CommonParameters>]
Get-MMPTask [-ScheduleId <Guid>] [<CommonParameters>]
Get-MMPTask [-Collection <MigrationCollection>] [<CommonParameters>]
Get-MMPTask [-CollectionId <Guid>] [<CommonParameters>]
Get-MMPTask [-Connection <MigrationConnection>] [<CommonParameters>]
Get-MMPTask [-ConnectionId <Guid>] [<CommonParameters>]
Get-MMPTask [-Event <MigrationEvent>] [<CommonParameters>]
Get-MMPTask [-EventId <Guid>] [<CommonParameters>]
Get-MMPTask [-Id <String>] [<CommonParameters>]
Get-MMPTask [-InputObject <MigrationTask>] [<CommonParameters>]
```

Parameters

PARAMETER	DESCRIPTION
Event	Use the Event parameter to retrieve the list of tasks associated with that event.
EventId	Use the EventId parameter to retrieve the list of tasks associated with that event GUID.
Collection	Use the Collection parameter to retrieve the list of tasks associated with that collection.
CollectionId	Use the Collection parameter to retrieve the list of tasks associated with that collection GUID.
Connection	Use the Connection parameter to retrieve the list of tasks associated with that connection.
ConnectionId	Use the Connection parameter to retrieve the list of tasks associated with that connection GUID.
Detailed	Use the Detailed parameter when you have a task and get details on the task so that you can get associated subtasks.
Schedule	Use the Schedule parameter to retrieve the list of tasks associated with that schedule.
ScheduleId	Use the Schedule parameter to retrieve the list of tasks associated with that schedule GUID.
Type	Use the Type parameter to restrict your results to a specific type of task. Valid values are Migration, UserDiscovery, ComputerDiscovery, and PstFileDiscovery.

Example

To retrieve the migration task from the database, use the following example:

```
Get-MMPTask -Id $pstMigration
```

4

Migration Scenarios

Migrating to an on-premises Exchange target from a network share

The following sections provide instructions for migrating to an on-premises Exchange target. First, you discover what is available for migration, then you choose what to migrate, then choose where to migrate it, then you migrate the PSTs to their Exchange target mailboxes. Note that Migration Manager for PSTs does not create mailboxes in Exchange. You must create the mailboxes in Exchange before migrating PSTs.

Instructions for migrating to an Office 365 Exchange Online target are given in a following section.

Create a new source connection

Complete the following steps to define a new connection to a migration source:

1. Click the **Sources** link and then click **New Source**.
2. In the **Source Name** box, enter the desired name for your source connection.
3. In the **Admin Name** box, enter credentials for an Administrator that can open Administrator folder shares and access files and temp directories on any computer, including the migration server, during discovery and migration. The format must be username@domain or domain\username.
4. In the **Password** box, enter the password for the administrator entered in the **Admin Name** box.
5. Once you have entered all of the source information, click the **Test** link in the **Test Connection** box to test the connection to the source domain.
6. If the connection is valid, click **Save**.
7. If needed, repeat the steps above to create multiple source connections.

Create discovery tasks

Complete the following steps to connect to Active Directory to discover users, create network shares, and discover PST files to be migrated to mailboxes:

1. Click the **Discover** link and then click **New User Discovery**.
2. On the **Settings** tab in the **Name** box, enter a user discovery task name.

3 In the **Source** box, select the desired source connection.

4 Optionally, in the **Advanced Discovery Filters** section of the screen, select containers to restrict the scope of discovery.

5 On the **Schedule** tab, select a radio button to set the schedule on which to run New User Discovery:

- Run Now
- Run Once
- Run Recurring

6 Click **Run** to start new user discovery.

7 Click the **Discover/Computers links** and then click **New Network Share**.

8 Enter a network share name.

9 Select a source that has credentials to access the network share from the drop-down list.

10 Enter the path for the network share and optionally add subfolders to refine the directories that can be scanned by this network share.

11 Click **Test** to confirm that the selected credentials can access the network share.

12 Click **Save**.

13 Click the **Discover** link and then click **New PST Discovery on Network Shares**.

14 On the **Settings** tab in the **Name** box, enter the PST discovery task name.

15 In the **Source** box, select the desired source connection.

16 To narrow your search, in the **Relative Path** box, enter the path to search for PSTs. Otherwise, leave this field blank to search the entire network share. The **Relative Path** can be specified as a comma-delimited list of directories that exist under the network share path. Examples of valid relative paths include:

- foldername, foldername, foldername\subfoldername
- foldername\subfoldername

UNC paths are not allowed in the relative path.

17 Under **Search Options**, if you select the **Recurse** checkbox, then MMP will search through subfolders of the paths entered in the **Relative Path** box.

Select the **Exclude system files** checkbox if you want to exclude system files from the network share search.

18 On the **Network Shares** tab, the **Filter available network shares** field allows you to optionally filter by network share name. In the **Filter by network share name box**, enter a name to narrow down the list of network shares displayed on the left. This makes it easier to select which ones to move to the right to be searched in that particular PST Discovery task.

19 On the **Schedule** tab, click the radio button to select the schedule on which to run the PST Discovery:

- Run Now
- Run Once
- Run Recurring

20 Click **Run** to start new PST discovery.

As files are discovered, the owner of the PST files is determined by the Access Control List (ACL) on the source system. To change the owner, change the owner in the ACL on the source system and perform discovery again to update it. You can also change the owner of the PST after discovery with the Set-MMPPstFileOwner cmdlet.

Create a new user or PST collection

A collection is a grouping of discovered users or PSTs upon which MMP can perform migration operations. Collections can be made up of users or PSTs from a single source connection, or multiple source connections.

Note: User discovery must be completed before creating a user collection, and PST discovery must be completed before creating a PST collection.

Complete the following steps to create a collection:

1. Click the **Collections** link and then click **New User Collection** or **New PST Collection**.
2. On the **Settings** tab in the **Name** box, enter a name for the collection.
3. In the **Label** box, enter a descriptive word or phrase for the collection.
4. On the **Users** tab, you can optionally filter by **Name** or **Email**. On the **PSTs** tab, you can optionally filter by **Name** or **Owner**.
5. Select the users or PSTs you want to include in the collection, move them to the right, and click **Close**.

Create a target connection

Complete the following steps to specify how to define a new connection to a migration target:

1. Click the **Targets** link and then click **New Target**.
2. In the **Target Name** box, enter a name for the target system.
3. In the **System** drop-down list, select Exchange2016, **Exchange 2013** or **Exchange 2010**.
4. Autodiscover is enabled by default but may be unchecked to directly enter the URL for your Exchange server.
5. In the **Admin name** box, enter the administrator name with rights to connect to all of the target mailboxes that will be used for migration. The format must be username@domain or domain\username.
6. In the **Password** box, enter the password for the Admin name you entered above.
7. To work around Exchange limits on concurrent connections, you can use the **Add Additional Credential** box to specify additional accounts to be used for migration
8. Once you have entered all of the target information, enter the name of a target mailbox that you will be migrating to and click the **Test** link in the **Test Connection** box to test the connection.
9. Click **Save**.

Create a migration task

Complete the following steps to create a new migration task:

1. Click the **Migration** link and then click **New Migration**.
2. On the **Task Settings** tab in the **Name** box, enter a name for the migration task.
3. In the **Collection** box, select the name for the collection you will migrate.
4. In the **Target** box, select the name of the target connection you set up to connect to target mailboxes.
5. On the **Migration Settings** tab, select the destination:
    - **Migrate to Primary:** Migrate PSTs to users' primary mailboxes.
    - **Migrate to Archive:** Migrate PSTs to an archive.
    - **Migrate to Subfolder:** Migrate PSTs to a subfolder of users' primary mailboxes or archive.

Then, select the checkboxes (any or all) next to the items you intend to migrate:
Email • **Contacts** • **Calendar** • **Tasks**

If you select email, also select the following options:

- **Date Range** options: **All email** • **Last modified date**
- To increase migration performance:
  Exclude Deleted Items • **Exclude Junk Mail** • **Exclude Sent Mail**
- Exclude specific folders: Enter folder names to exclude, separated by commas.

6 On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**

7 Click **Run** to start the migration task.

Migrating to an on-premises Exchange target

The following sections provide instructions for migrating to an on-premises Exchange target. First, you discover what is available for migration, then you choose what to migrate, then choose where to migrate it, then you migrate the PSTs to their Exchange target mailboxes. Note that Migration Manager for PSTs does not create mailboxes in Exchange. You must create the mailboxes in Exchange before migrating PSTs.

Instructions for migrating to an Office 365 Exchange Online target are given in a following section.

Create a new source connection

Complete the following steps to define a new connection to a migration source:

1 Click the **Sources** link and then click **New Source**.

2 In the **Source Name** box, enter the desired name for your source connection.

3 In the **Admin Name** box, enter credentials for an Administrator that can open Administrator folder shares and access files and temp directories on any computer, including the migration server, during discovery and migration. The format must be username@domain or domain\username.

4 In the **Password** box, enter the password for the administrator entered in the **Admin Name** box.

5 Once you have entered all of the source information, click the **Test** link in the **Test Connection** box to test the connection to the source domain.

6 If the connection is valid, click **Save**.

7 If needed, repeat the steps above to create multiple source connections.

Create discovery tasks

Complete the following steps to connect to Active Directory to discover users, computers, and PST files to be migrated to mailboxes:

1 Click the **Discover** link and then click **New User Discovery**.

2 On the **Settings** tab in the **Name** box, enter a user discovery task name.

3 In the **Source** box, select the desired source connection.

4 Optionally, in the **Advanced Discovery Filters** section of the screen, select containers to restrict the scope of discovery.

5 On the **Schedule** tab, click a radio button to select the schedule on which to run New User Discovery: **Run Now** or **Run Once** or **Run Recurring**

6 Click **Run** to start new user discovery.

7 Click th **Discover** link and then click **New Computer Discovery**.

8 On the **Settings** tab in the **Name** box, enter a computer discovery task name.

9 In the **Source** box, select the desired source connection.

10 On the **Schedule** tab, click the radio button to select the schedule on which to run the Computer Discovery: **Run Now** or **Run Once** or **Run Recurring**

11 Click **Run** to start new computer discovery.

Note: Computer discovery must be completed before running PST discovery.

12 Select the **Discover** link and then click **New PST Discovery on Computers**.

13 On the **Settings** tab in the **Name** box, enter the PST discovery task name.

14 In the **Source** box, select the desired source connection.

15 To narrow your search, in the **Relative Path** box, enter the path to search for PSTs. Otherwise, leave this field blank to search the entire computer. The **Relative Path** can be specified as a comma-delimited list of directories that exist under the network share path. Examples of valid relative paths include:

- foldername, foldername, foldername\subfoldername
- foldername\subfoldername

UNC paths are not allowed in the relative path.

16 Under **Search Options**, if you select the **Recurse** checkbox, then MMP will search through subfolders of the paths entered in the **Relative Path** box.

Select the **Exclude system files** checkbox if you want to exclude system files from the network share search.

17 On the **Computers** tab, the **Filter available computers** field allows you to optionally **Filter by computer name** and/or **Filter by an OU**. In the **Filter by Computer name box**, enter a computer name to narrow down the list of computers displayed on the left. This makes it easier to select which ones to move to the right to be searched in that particular PST Discovery task. In the Filter by an OU drop-down list, filter by selecting an organizational unit.

18 On the **Schedule** tab, click the radio button to select the schedule on which to run the PST Discovery: **Run Now** or **Run Once** or **Run Recurring**

19 Click **Run** to start new PST discovery.

As files are discovered, the owner of the PST files is determined by the Access Control List (ACL) on the source system. To change the owner, change the owner in the ACL on the source system and perform discovery again to update it. You can also change the owner of the PST after discovery with the Set-MMPPstFileOwner cmdlet.

Create a new user or PST collection

Note: User discovery must be completed before creating a user collection, and PST discovery must be completed before creating a PST collection.

Complete the following steps to create a collection:

1 Click the **Collections** link and then click **New User Collection** or **New PST Collection**.

2 On the **Settings** tab in the **Name** box, enter a name for the collection.

3 In the **Label** box, enter a descriptive word or phrase for the collection.

4 On the **Users** tab, you can optionally filter by **Name** or **Email**. On the **PSTs** tab, you can optionally filter by **Name** or **Owner**.

5 Select the users or PSTs that you want to include in the collection, move them to the right, and click **Close**.

Create a target connection

Complete the following steps to specify how to define a new connection to a migration target:

1 Click the **Targets** link and then click **New Target**.

2 In the **Target Name** box, enter a name for the target system.

3 In the **System** drop-down list, select Exchange2016, **Exchange 2013** or **Exchange 2010**.

4 Autodiscover is enabled by default and may be unchecked to directly enter the URL for your Exchange server.

5 In the **Admin name** box, enter the administrator name with rights to connect to all of the target mailboxes that will be used for migration. The format must be username@domain or domain\username.

6 In the **Password** box, enter the password for the Admin name you entered above.

7 To work around Exchange limits on concurrent connections, you can use the **Add Additional Credential** box to specify additional accounts to be used for migration

8 Once you have entered all of the target information, enter the name of a target mailbox that you will be migrating to and click the **Test** link in the **Test Connection** box to test the connection.

9 Click **Save**.

Create a migration task

Complete the following steps to create a new migration task:

1 Click the **Migration** link and then click **New Migration**.

2 On the **Task Settings** tab in the **Name** box, enter a name for the migration task.

3 In the **Collection** box, select the name for the collection you will migrate.

4 In the **Target** box, select the name of the target connection you set up to connect to target mailboxes.

5 On the **Migration Settings** tab, select the destination:

- **Migrate to Primary:** Migrate PSTs to users' primary mailboxes.
- **Migrate to Archive:** Migrate PSTs to an archive.
- **Migrate to Subfolder:** Migrate PSTs to a subfolder of users' primary mailboxes or archive.

Then, select the checkboxes (any or all) next to the items you intend to migrate:
Email • Contacts • Calendar • Tasks

If you select email, also select the following options:

- **Date Range** options: **All email • Last modified date**
- To increase migration performance:
  Exclude Deleted Items • Exclude Junk Mail • Exclude Sent Mail
- Exclude specific folders: Enter folder names to exclude, separated by commas.

6 On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**

7 Click **Run** to start the migration task.

Migrating to an O365 target

The following sections provide instructions for migrating to an O365 target. First, you discover what is available for migration, then you choose what to migrate, then choose where to migrate it, then you migrate the PSTs to their Exchange target mailboxes. Note that Migration Manager for PSTs does not create mailboxes in Exchange Online. You must create the mailboxes in Exchange Online before migrating PSTs.

Instructions for migrating to an on-premises Exchange target are given in an earlier section.

Create a new source connection

Complete the following steps to define a new connection to a migration source:

1. Click the **Sources** link and then click **New Source**.
2. In the **Source Name** box, enter the desired name for your source connection.
3. In the **Admin Name** box, enter credentials for an Administrator that can open Administrator folder shares and access files and temp directories on any computer, including the migration server, during discovery and migration. The format must be username@domain or domain\username.
4. In the **Password** box, enter the password for the administrator entered in the **Admin Name** box.
5. Once you have entered all of the source information, click the **Test** link in the **Test Connection** box to test the connection to the source domain.
6. If the connection is valid, click **Save**.
7. If needed, repeat the steps above to create multiple source connections.

Create discovery tasks

Complete the following steps to connect to Active Directory to discover users, computers, and PST files to be migrated to mailboxes:

1. Click the **Discover** link and then click **New User Discovery**.
2. On the **Settings** tab in the **Name** box, enter a user discovery task name.
3. In the **Source** box, select the desired source connection.
4. Optionally, in the **Advanced Discovery Filters** section of the screen, select containers to restrict the scope of discovery.
5. On the **Schedule** tab, click a radio button to select the schedule on which to run New User Discovery: **Run Now** or **Run Once** or **Run Recurring**
6. Click **Run** to start new user discovery.
7. Select **Discover/New Computer Discovery**.
8. On the **Settings** tab in the **Name** box, enter a computer discovery task name.
9. In the **Source** box, select the desired source connection.
10. On the **Schedule** tab, click the radio button to select the schedule on which to run the Computer Discovery: **Run Now** or **Run Once** or **Run Recurring**
11. Click **Run** to start new computer discovery.
12. Select the **Discover** link and then click **New PST Discovery**.
13. In the **Source** box, select the desired source connection.

14 To narrow your search, in the **Relative Path** box, enter the path to search for PSTs. Otherwise, leave this field blank to search the entire computer. The **Relative Path** can be specified as a comma-delimited list of directories that exist under the network share path. Examples of valid relative paths include:

- foldername, foldername, foldername\subfoldername
- foldername\subfoldername

UNC paths are not allowed in the relative path.

15 Under **Search Options**, if you select the **Recurse** checkbox, then MMP will search through subfolders of the paths entered in the **Relative Path** box.

Select the **Exclude system files** checkbox if you want to exclude system files from the network share search.

16 On the **Computers** tab, the **Filter available computers** field allows you to optionally **Filter by computer name** and/or **Filter by an OU**. In the **Filter by Computer name box**, enter a computer name to narrow down the list of computers displayed on the left. This makes it easier to select which ones to move to the right to be searched in that particular PST Discovery task. In the Filter by an OU drop-down list, filter by selecting an organizational unit.

17 On the **Schedule** tab, click the radio button to select the schedule on which to run the PST Discovery: **Run Now** or **Run Once** or **Run Recurring**

18 Click **Run** to start new PST discovery.

Create a new user or PST collection

A collection is a grouping of discovered users or PSTs that MMP can perform migration operations upon. Collections can be made up of users or PSTs from a single source connection, or from multiple source connections.

Note: User discovery must be completed before creating a user collection, and PST discovery must be completed before creating a PST collection.

Complete the following steps to create a collection:

1 Click the **Collections** link and then click **New User Collection** or **New PST Collection**.

2 On the **Settings** tab in the **Name** box, enter a name for the collection.

3 In the **Label** box, enter a descriptive word or phrase for the collection.

4 On the **Users** tab, you can optionally filter by **Name** or **Email**. On the **PSTs** tab, you can optionally filter by **Name** or **Owner**.

5 Select the users or PSTs that you want to include in the collection, move them to the right, and click **Close**.

Create a target connection

Complete the following steps to specify how to define a new connection to a migration target:

1 Click the **Targets** link and then click **New Target**.

2 In the **Target Name** box, enter a name for the target system.

3 In the **System** drop-down list, select **O365**.

4 In the **Admin name** box, enter the administrator name with rights to connect to all of the target mailboxes that will be used for migration.

5 In the **Password** box, enter the password for the Admin name you entered above.

6 To work around O365 limits on concurrent connections, you can use the **Add Additional Credential** box to specify additional accounts to be used for migration.

7 Once you have entered all of the target information, enter the name of a target mailbox that you will be migrating to and click the **Test** link in the **Test Connection** box to test the connection.

8 Click **Save**.

Create a migration task

Complete the following steps to create a new migration task:

1 Click the **Migration** link and then click **New Migration**.

2 On the **Task Settings** tab in the **Name** box, enter a name for the migration task.

3 In the **Target** box, select the name of the target connection you set up to connect to target mailboxes.

4 On the **Migration Settings** tab, select the destination:

- **Migrate to Primary:** Migrate PSTs to users' primary mailboxes.
- **Migrate to Archive:** Migrate PSTs to an archive.
- **Migrate to Subfolder:** Migrate PSTs to a subfolder of users' primary mailboxes or archive.

Then, select the checkboxes (any or all) next to the items you intend to migrate:
Email • Contacts • Calendar • Tasks

If you select email, also select the following options:

- **Date Range** options: **All email • Last modified date**
- To increase migration performance:
  Exclude Deleted Items • Exclude Junk Mail • Exclude Sent Mail
- Exclude specific folders: Enter folder names to exclude, separated by commas.

5 On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**

6 Click **Run** to start the migration task.

7 On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**

8 Click **Run** to start the migration task.

Migrating to a hybrid target environment

One migration scenario involves using On-Premises Exchange for your primary mailboxes and O365 for your archives. Migration to this type of environment can be accomplished by setting up an on-premises Exchange target in MMP. Then, when configuring a migration task, on the Migration Settings tab selecting **Migrate to Archive**. Configuring the MMP target to an on-premises Exchange server allows the configured hybrid environment to forward the migrated data to the O365 archive. This is described in the migration procedure below. First, discover what is available for migration. Then, select what to migrate and where to migrate it. Lastly, migrate the PSTs to their Exchange target mailboxes. Note that Migration Manager for PSTs does not create mailboxes in Exchange or Exchange Online. You must create the mailboxes before migrating PSTs.

Create a new source connection

Complete the following steps to define a new connection to a migration source:

1. Click the **Sources** link and then click **New Source**.
2. In the **Source Name** box, enter the desired name for your source connection.

   This field is required. MMP performs a validation check on the name entered. If the source name has already been used you are prompted to enter a different source name.
3. In the **Admin Name** box, enter credentials for an Administrator that can open Administrator folder shares and access files and temp directories on any computer, including the migration server, during discovery and migration. The format must be username@domain or domain\username.
4. In the **Password** box, enter the password for the administrator entered in the **Admin Name** box.
5. Once you have entered all of the source information, click the **Test** link in the **Test Connection** box to test the connection to the source domain.
6. If the connection is valid, click **Save**.
7. If needed, repeat the steps above to create multiple source connections.

Create discovery tasks

Complete the following steps to connect to Active Directory to discover users, computers, and PST files to be migrated to mailboxes:

1. Click the **Discover** link and then click **New User Discovery**.
2. On the **Settings** tab in the **Name** box, enter a user discovery task name.
3. In the **Source** box, select the desired source connection.
4. Optionally, in the **Advanced Discovery Filters** section of the screen, select containers to restrict the scope of discovery.
5. On the **Schedule** tab, click a radio button to select the schedule on which to run New User Discovery: **Run Now** or **Run Once** or **Run Recurring**
6. Click **Run** to start new user discovery.
7. Select **Discover/New Computer Discovery**.
8. On the **Settings** tab in the **Name** box, enter a computer discovery task name.
9. In the **Source** box, select the desired source connection.
10. On the **Schedule** tab, click the radio button to select the schedule on which to run the Computer Discovery: **Run Now** or **Run Once** or **Run Recurring**
11. Click **Run** to start new computer discovery.
12. Select **Discover/New PST Discovery**.
13. On the **Settings** tab in the **Name** box, enter the PST discovery task name.
14. In the **Source** box, select the desired source connection.
15. To narrow your search, in the **Relative Path** box, enter the path to search for PSTs. Otherwise, leave this field blank to search the entire computer. The **Relative Path** can be specified as a comma-delimited list of directories that exist under the network share path. Examples of valid relative paths include:

    - foldername, foldername, foldername\subfoldername
    - foldername\subfoldername

    UNC paths are not allowed in the relative path.

16 Under **Search Options**, if you select the **Recurse** checkbox, then MMP will search through subfolders of the paths entered in the **Relative Path** box.

   Select the **Exclude system files** checkbox if you want to exclude system files from the network share search.

17 On the **Computers** tab, the **Filter available computers** field allows you to optionally **Filter by computer name** and/or **Filter by an OU**. In the **Filter by Computer name box**, enter a computer name to narrow down the list of computers displayed on the left. This makes it easier to select which ones to move to the right to be searched in that particular PST Discovery task. In the Filter by an OU drop-down list, filter by selecting an organizational unit.

18 On the **Schedule** tab, click the radio button to select the schedule on which to run the PST Discovery: **Run Now** or **Run Once** or **Run Recurring**

19 Click **Run** to start new PST discovery.

Create a new user or PST collection

A collection is a grouping of discovered users or PSTs that MMP can perform migration operations upon. Collections can be made up of users or PSTs from a single source connection, or from multiple source connections.

Note: User discovery must be completed before creating a user collection, and PST discovery must be completed before creating a PST collection.

Complete the following steps to create a collection:

1 Click the **Collections** link and then click **New User Collection** or **New PST Collection**.

2 On the **Settings** tab in the **Name** box, enter a name for the collection.

3 In the **Label** box, enter a descriptive word or phrase for the collection.

4 On the **Users** tab, you can optionally filter by **Name** or **Email**. On the **PSTs** tab, you can optionally filter by **Name** or **Owner**.

5 Select the users or PSTs that you want to include in the collection, move them to the right, and click **Close**.

Create a Target Connection

Complete the following steps to specify how to define a new connection to a migration target:

1 Click the **Targets** link and then click **New Target**.

2 In the **Target Name** box, enter a name for the target system.

3 In the **System** drop-down list, select Exchange2016, **Exchange 2013 or Exchange 2010**.

4 In the **Admin name** box, enter the administrator name with rights to connect to all of the target mailboxes that will be used for migration. The format must be username@domain or domain\username.

5 In the **Password** box, enter the password for the Admin name you entered above.

6 To work around Exchange and O365 limits on concurrent connections, you can use the **Add Additional Credential** box to specify additional accounts to be used for migration.

7 Once you have entered all of the target information, enter the name of a target mailbox that you will be migrating to and click the **Test** link in the **Test Connection** box to test the connection.

8 Click **Save**.

Create a Migration Task

Complete the following steps to create a new migration task:

1. Click the **Migration** link and then click **New Migration**.
2. On the **Task Settings** tab in the **Name** box, enter a name for the migration task.
3. In the **Target** box, select the name of the target connection you set up to connect to target mailboxes.
4. On the **Migration Settings** tab, select **Migrate to Archive**.

   Then, select the checkboxes (any or all) next to the items you intend to migrate:
   Email • Contacts • Calendar • Tasks

   If you select email, also select the following options:

   - **Date Range** options: **All email • Last modified date**
   - To increase migration performance:
     Exclude Deleted Items • Exclude Junk Mail • Exclude Sent Mail
   - Exclude specific folders: Enter folder names to exclude, separated by commas.
5. On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**
6. Click **Run** to start the migration task.

Migrating to O365 without a hybrid target environment

When you do not have a hybrid configuration as described in the previous section, Active Directory may not know the target email addresses in O365. If this is the case, an MMP user discovery would not have the correct target email address. If the users that you want to migrate to O365 do not have target email addresses in your O365 domain, then you will need to set them. This scenario describes how to set the target email address in the MMP Database.

The following sections provide instructions for migrating to an O365 target. First, you discover what is available for migration, then you choose what to migrate, then choose where to migrate it, then you migrate the PSTs to their O365 target mailboxes. Note that Migration Manager for PSTs does not create mailboxes in O365. You must create the mailboxes in O365 before migrating PSTs.

Instructions for migrating to an on-premises Exchange target are given in an earlier section.

Create a new source connection

Complete the following steps to define a new connection to a migration source:

1. Click the **Sources** link and then click **New Source**.
2. In the **Source Name** box, enter the desired name for your source connection.
3. In the **Admin Name** box, enter credentials for an Administrator that can open Administrator folder shares and access files and temp directories on any computer, including the migration server, during discovery and migration. The format must be username@domain or domain\username.
4. In the **Password** box, enter the password for the administrator entered in the **Admin Name** box.
5. Once you have entered all of the source information, click the **Test** link in the **Test Connection** box to test the connection to the source domain.
6. If the connection is valid, click **Save**.

7 If needed, repeat the steps above to create multiple source connections.

Complete the following steps to connect to Active Directory to discover users, computers, and PST files to be migrated to mailboxes:

1 Click the **Discover** link and then click **New User Discovery**.

2 On the **Settings** tab in the **Name** box, enter a name for the User Discovery Task Name.

3 In the **Source** box, select the desired source connection.

4 On the **Schedule** tab, click a radio button to select the schedule on which to run New User Discovery: **Run Now** or **Run Once** or **Run Recurring**

5 Click **Run** to start new user discovery.

6 Select **Discover/New Computer Discovery**.

7 On the **Settings** tab in the **Name** box, enter a computer discovery task name.

8 In the **Source** box, select the desired source connection.

9 On the **Schedule** tab, click the radio button to select the schedule on which to run the Computer Discovery: **Run Now** or **Run Once** or **Run Recurring**

10 Click **Run** to start new computer discovery.

11 Select **Discover/New PST Discovery**.

12 On the **Settings** tab in the **Name** box, enter the PST discovery task name.

13 In the **Source** box, select the desired source connection.

14 To narrow your search, in the **Relative Path** box, enter the path to search for PSTs. Otherwise, leave this field blank to search the entire computer. The **Relative Path** can be specified as a comma-delimited list of directories that exist under the network share path. Examples of valid relative paths include:

- foldername, foldername, foldername\subfoldername
- foldername\subfoldername

UNC paths are not allowed in the relative path.

15 Under **Search Options**, if you select the **Recurse** checkbox, then MMP will search through subfolders of the paths entered in the **Relative Path** box.

Select the **Exclude system files** checkbox if you want to exclude system files from the network share search.

16 On the **Computers** tab, the **Filter available computers** field allows you to optionally **Filter by computer name** and/or **Filter by an OU**. In the **Filter by Computer name box**, enter a computer name to narrow down the list of computers displayed on the left. This makes it easier to select which ones to move to the right to be searched in that particular PST Discovery task. In the Filter by an OU drop-down list, filter by selecting an organizational unit.

17 On the **Schedule** tab, click the radio button to select the schedule on which to run the PST Discovery: **Run Now** or **Run Once** or **Run Recurring**

18 Click **Run** to start new PST discovery.

Setting the target email address with the Set-MMPDiscoveredUser cmdlet

To set the target email address on a discovered user in the MMP database, use the Set-MMPDiscoveredUser cmdlet. This does not change the target email address in Active Directory, just in the MMP database. See the example below.

Set the TargetEmailAddress for a given user migrating to O365.

```
$user1 = Get-MMPDiscoveredUser -Email John.Doe@ sitraka.com
```

```
Set-MMPDiscoveredUser -InputObject $user1 -TargetEmailAddress
"William.Smith@sitraka.onmicrosoft.com"
```

The Set-MMPDiscoveredUser cmdlet is described in detail in Chapter 4, PowerShell Cmdlets.

Create a user collection

Using the user with the updated target email address, create a user collection.

A collection is a grouping of discovered users that MMP can perform migration operations upon. Collections can be made up of users from a single source connection, or from multiple source connections.

User discovery must be completed before creating a collection.

Complete the following steps to select users' PST files to migrate to mailboxes:

1. Click the **Collections** link and then click **New User Collection**.
2. On the **Settings** tab in the **Name** box, enter a name for the collection.
3. In the **Label** box, enter a descriptive word or phrase for the collection. For example, you may want to use the organizational unit for that collection, such as Marketing.
4. On the **Users** tab in the **Filter by name** box, optionally enter a user name to filter the search.
5. Select the users you want to include in the collection, move them to the right, and click **Close**.

Create a target connection

Complete the following steps to specify how to define a new connection to a migration target:

1. Click the **Targets** link and then click **New Target**.
2. In the **Target Name** box, enter a name for the target system.
3. In the **System** drop-down list, select **O365**.
4. In the **Admin name** box, enter the administrator name with rights to connect to all of the target mailboxes that will be used for migration.
5. In the **Password** box, enter the password for the Admin name you entered above.
6. To work around O365 limits on concurrent connections, you can use the **Add Additional Credential** box to specify additional accounts to be used for migration.
7. Once you have entered all of the target information, enter the name of a target mailbox that you will be migrating to and click the **Test** link in the **Test Connection** box to test the connection.
8. Click **Save**.

Create a migration task

Complete the following steps to create a new migration task:

1. Click the **Migration** link and then click **New Migration**.
2. On the **Task Settings** tab in the **Name** box, enter a name for the migration task.
3. In the **Target** box, select the name of the target connection you set up to connect to target mailboxes.
4. On the **Migration Settings** tab, select the destination:
   - **Migrate to Primary:** Migrate PSTs to users' primary mailboxes.
   - **Migrate to Archive:** Migrate PSTs to an archive.

- **Migrate to Subfolder:** Migrate PSTs to a subfolder of users' primary mailboxes or archive.

Then, select the checkboxes (any or all) next to the items you intend to migrate:
Email • Contacts • Calendar • Tasks

If you select email, also select the following options:

- **Date Range** options: **All email • Last modified date**
- To increase migration performance:
  Exclude Deleted Items • Exclude Junk Mail • Exclude Sent Mail
- Exclude specific folders: Enter folder names to exclude, separated by commas.

5 On the **Schedule** tab, select the schedule to run the migration(s): **Run Now** or **Run Once** or **Run Recurring**

6 Click **Run** to start the migration task.

7 On the **Schedule** tab, select the schedule to run the migration(s):**Run Now** or **Run Once** or **Run Recurring**

8 Click **Run** to start the migration task.

5

Log Files

Overview

Log files provide useful information about your migration. Log files are generated for both the agents and PowerShell cmdlets and are located in the following directory:

- %systemdrive%\ProgramData\Dell\Migration Manager for PSTs\logs

The **logs** directory contains the following subdirectories:

- Configuration Console
- Discovery Agent
- Migration Agent
- Schedule Agent
- WebService
- WindowsPowerShell

Configuration Console log files

- ConfigurationConsole.wlog

Agent Log files

- DiscoveryAgent.wlog
- MigrationAgent.wlog
- ScheduleAgent.wlog

The following are examples of the information logged:

- Agent startup, shutdown
- Communication errors between agent and web services
- Task scheduling, start, stop, and completion status
- Task errors, warnings, and other informational messages
- Discovery of a new user, computer, or PST file
- Migration of a PST file

WebService log files

- WebService.wlog

PowerShell log files

The following log files are generated for any Windows PowerShell cmdlets that are run:

- WindowsPowerShell.wlog
- WindowsPowerShell.wlog2

The Dell Log Viewer

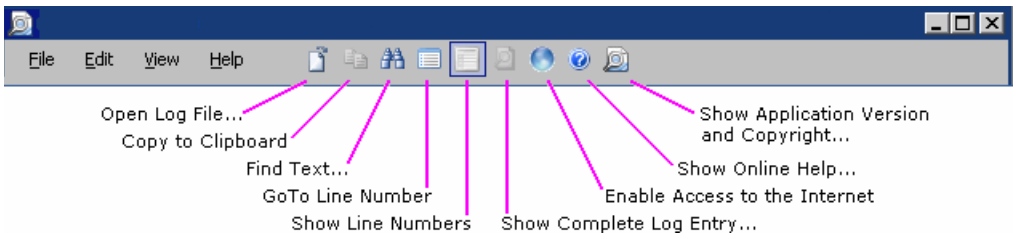
The Dell Log Viewer simplifies the viewing and interpretation of program log files, which document events and warnings in Dell programs.

IMPORTANT: The full benefit of Dell's Log Viewer requires the application to transmit log event codes from the computer hosting the Log Viewer to a remote Dell database of *Solutions* (useful information about the events associated with the log codes). The Dell *Solutions* database then sends a corresponding *Solution* back to the Log Viewer for display to the user.

This feature is enabled by default, but can be disabled by deselecting the **Enable Internet Access** option on the **View** menu. For more information, see the IMPORTANT disclosures in the **View** menu topic below, in the field notes for the **Enable Internet Access** option.

Log Viewer Menus and Toolbar

Most Log Viewer features are accessible by the program's menus and/or the program tool bar, which share a horizontal band across the top of the screen:



Several features are also available directly from the keyboard, and those keyboard shortcuts are displayed in the menus and noted here.

File Menu

- **Open Log File...** (or Ctrl+O): Opens a standard Windows *Open* dialog box, from which you can specify the file you want to open into the Log Viewer. The Log Viewer can open and display *WLog* (optionally compressed) files, and plain text files.

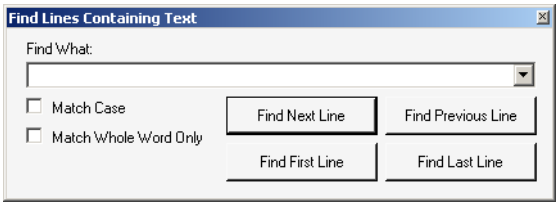
  Drag-and-Drop Option: You can also open a wlog file in the Log Viewer by dragging and dropping a filename from Windows Explorer into the Log Viewer window.

- **Save Copy Of Log File As...** (appears only when a file is open): Opens a standard Windows *Save As* dialog box, from which you can specify the filename and location where you want the open file to be saved. The Log Viewer lets you edit the contents of an open file, but will not replace the original on disk with the edited version (you cannot save it under the same name in the same location).

- **Recent Files:** Shows a list of recently opened files, from which you can select a file to re-open (to quickly re-open a file you have recently viewed and closed).
- **Exit:** Closes the Log Viewer window.

Edit Menu

- **Copy (or Ctrl+C):** Copies the selected line to the Windows clipboard.
- **Find... (or Ctrl+F):** Opens a **Find** dialog box that lets you specify a text string to search for within the open file:



  The dialog box lets you search for the next or preceding occurrence, or for the first or last occurrence in the file. The **Find** feature highlights the entire line that contains the target string.
- **Go To Line Number:** Open a dialog box that lets you jump to a particular line number of the file. (Enter the line number and click **OK**.)

View Menu

- **Show Line Numbers** (or Ctrl+L): Toggles the display of line numbers (within the open file) on and off.
- **Show Complete Log Entry** (or F5): Opens a *Log Detail* window that shows the entire string for the selected item—useful when the item text overruns the Log Viewer's maximum line length (maximum 259 characters), or if the line extends beyond the right edge of the viewer window without wrapping.
- **Enable Internet Access:** Toggles the Internet connection on and off. Note these important disclosures:

  IMPORTANT: The full benefit of Dell's Log Viewer requires the application to transmit log event codes from the computer hosting the Log Viewer to a remote Dell database of *Solutions* (useful information about the events associated with the log codes). The Dell *Solutions* database then sends a corresponding *Solution* back to the Log Viewer for display to the user.

  This feature is enabled by default, but can be disabled by deselecting this **Enable Internet Access** option (on the **View** menu). Note:

  - The Log Viewer sends event codes to Dell servers in the United States, as the codes appear in a log file that has been opened into the Log Viewer window.
  - The Log Viewer sends only event codes and source IP addresses to Dell's server. No personally identifying information other than IP addresses is collected or sent.
  - Dell's transmission of *Solutions* corresponding to received event codes is fully automated (no human intervention or observation). Dell tallies the frequencies of event codes received and *Solutions* sent in reply, but does not associate those tallies with IP addresses or any other personally identifying information, and no other information is logged.
  - Dell servers treat each received event code as a query. Dell uses an event code only to determine which *Solution* to send back to the source of the query, and uses an IP address only to transmit the *Solution* to the source of the query. Dell servers log querying IP addresses as a matter of course, but Dell does not use them for any other purpose and does not disclose them to any other entity.
  - You may opt out of this feature, to prevent the transmission of event codes and IP addresses to Dell, by unmarking the **Enable Internet Access** option on the **View** menu.
- **Goto Line Number:** Prompts for a line number in the file to display.

Help Menu

- **Online Help... (or F1):** Opens Dell's online Help file for the Log Viewer, which documents its features.
- **About...:** Opens a window of information about the Log Viewer—identifying the current release, and asserting Dell's intellectual property rights to the software.

How To ...

This section describes how to complete various tasks in the Log Viewer.

To Open a Specific Log:

- Drag and drop a wlog file name from Windows Explorer into the Log Viewer window.
- Click the **Open Log File** button (in the Toolbar) to view a list of log files that can be opened. In the *File* section of the screen, select a log file and click **OK** to open the log in the Dell Log File Viewer.

To Find a Particular Text String Within an Open File:

- **Edit** menu | **Find...** (or Ctrl+F): Opens a **Find** dialog box that lets you specify a text string to search for within the open file. The **Find** feature highlights the entire line that contains the target string.

To Re-Open a Recently Viewed File:

- **File** menu | **Recent Files:** Shows a list of recently opened files, from which you can select a file to re-open (to quickly re-open a file you have recently viewed and closed).

To Save a Copy of a File:

- **File** menu | **Save Copy Of Log File As...** (appears only when a file is open): Opens a standard Windows *Save As* dialog box, from which you can specify the filename and location where you want the open file to be saved. (This feature does not permit any revisions to the open file. It simply lets you save the file *in its original form* to a new filename and/or a new location.)

To Show or Hide Line Numbers:

- **View** menu | **Show Line Numbers (or Ctrl+L):** Toggles the display of line numbers (within the open file) on and off.

To Jump to a Particular Line Number in the File:

- **Edit** menu | **Go To Line Number:** Opens a dialog box that lets you specify the destination line number. (Enter the number and click **OK**.)

To View an Entire Untruncated Log Entry:

- **View** menu | **Show Complete Log Entry (or F5):** Opens a Log Detail window that shows the entire string for the selected item—useful when the item text overruns the Log Viewer's maximum line length (maximum 259 characters), or if the line extends beyond the right edge of the window without wrapping.

To Turn Internet Access On or Off:

- **View** menu | **Enable Internet Access:** Toggles the Internet connection on and off.

To Copy a Selected Line to the Clipboard:

- **Edit** menu | **Copy** (or Ctrl+C): Copies the selected line to the Windows clipboard.

To Close the Log Viewer:

- **File** menu | **Exit:** Closes the Log Viewer window, or click the Log Viewer **Close** box (**[X]**) to dismiss the window and return to the previous display.

Configuring Logs

The logging system within Migration Manager for PSTs is built on the Apache log4net project (http://logging.apache.org/log4net). The following sections describe the log4net appenders used by MMP in more detail.

Logging configuration files for each agent can be found within the MMP installation directory in the following locations:

- Discovery Agent Service/log4net.config
- Migration Agent Service/log4net.config
- Schedule Agent/log4net.config
- PowerShell
- Web Service
- Configuration Console

EventLogAppender

The EventLogAppender logs any ERROR level messages to the Windows event log. To log warnings to the event log, change the threshold value for this appender:

```
<appender name="EventLogAppender" type="log42net.Appender.EventLogAppender">
   <threshold value="ERROR"/>
   <applicationName value="Dell MMP Discovery Agent"/>
   <layout type="log42net.Layout.PatternLayout">
      <conversionPattern value="%date [%thread] %-5level %logger -
%message%newline"/>
   </layout>
</appender>
```

RestServiceAppender

The RestServiceAppender logs NOTICE, WARN, and ERROR level messages to the MMP REST Service.

```
<appender name="RestAppender"
type="DellSoftware.Log42Net.RestAppender.RestServiceAppender,
DellSoftware.Log42Net.RestAppender" >
   <threshold value="NOTICE" />
   <file type="log42net.Util.PatternString" value="%env{programData}\Dell\Migration
Manager for PSTs\logs\WindowsPowershell\WindowsPowershell.wlog" />
</appender>
```

RollingFileAppender

This appender acts as the rolling file appender which produces the local log files for each agent. The threshold level of this appender controls messages to the rolling file appender only. The threshold value is set to Debug to work with the appender listed below.

```
<appender name="RollingFileAppender" type="log4net.Appender.RollingFileAppender">
    <threshold value="DEBUG" />
    <layout type="log42net.Layout.PatternLayout">
      <conversionPattern value="%date [%thread] %-5level %logger - %message%newline"
/>
    </layout>
```

```
   <file type="log42net.Util.PatternString" value="%env{programData}\Dell\Migration
Manager for PSTs\logs\Migration Agent\MigrationAgent.wlog" />
    <PreserveLogFileNameExtension value="true" />
    <appendToFile value="true" />
    <rollingStyle value="Size" />
    <maximumFileSize value="25MB" />
    <maxSizeRollBackups value="10" />
  </appender>
```

MultithreadedBufferingForwardingAppender

The MultithreadedBufferingForwardingAppender is a custom appender that provides debug-level log messages for errors without running at DEBUG level. The default settings for this appender enable you to view useful troubleshooting information in the logs for errors in without generating large log files that would result from with the appenders at a DEBUG log level.

The MultithreadedBufferingForwardingAppender behaves like a BufferingForwardingAppender. It forwards messages to the attached appender based upon the threshold values that are set. It will forward only LossyLevel or higher messages to the attached appender, or in the case of a log message with a log level greater than or equal to the evaluator threshold, all of the previous log messages. The log level of the attached appender will filter the messages it receives.

The default configuration forwards log message of level INFO and higher to a RollingFileAppender named RollingFileAppender. The log level of the RollingFileAppender is set to DEBUG. INFO, NOTICE and WARN log messages will be forwarded to RollingFileAppender normally. When an ERROR message is handled by the ErrorLogFileAppender, it will send all the log messages to RollingFileAppender, which will write up to 30 (bufferSize) log messages that are DEBUG or higher.

```
<appender name="ErrorLogFileAppender"
type="log42net.Appender.MultithreadedBufferingForwardingAppender" >
    <bufferSize value="30"/>
    <bufferMultiplier value="5"/>
    <lossy value="true"/>
    <evaluator type="log42net.Core.LevelEvaluator">
      <threshold value="ERROR"/>
    </evaluator>
    <LossyEvaluator type="log42net.Core.LevelEvaluator">
      <threshold value="INFO"/>
    </LossyEvaluator>
    <appender-ref ref="RollingFileAppender"/>
  </appender>
```

- **bufferSize value="x"/>:** X is the max number of messages that will be sent on the evaluator threshold.
- **bufferMultiplier value="x"/>:** The internal queue of messages set to the buffer size times the multiplier. In the example below, the bufferSize of 30 times the bufferMutiplier of 5 equals 150.
- **lossy value="true":** The default value is true. if set to false, it will dump bufferSize*bufferMultipler messages to the log every time that number of messages has been queued.

The **bufferMultiplier** is an extension of the ErrorLogFileAppender. See your log4net resources for additional information.

TraceAppender

The TraceAppender logs calls made to the .NET Trace system. By default this appender is inactive. To enable tracing for an MMP agent, edit the log4net.config for that agent and add a reference to the appender to the root configuration node:

```
<appender name="TraceAppender" type="log42net.Appender.TraceAppender">
   <threshold value="ALL"/>
   <layout type="log42net.Layout.PatternLayout">
      <conversionPattern value="%date [%thread] %-5level %logger -
%message%newline"/>
   </layout>
</appender>
```

Enabling Appenders

The appenders listed in the root node will be used for logging. The appenders listed below are enabled by default. You can configure and add other appenders to this list.

```
<root>
   <level value="ALL"/>
   <appender-ref ref="RestAppender"/>
   <appender-ref ref="EventLogAppender"/>
   <appender-ref ref="ErrorLogFileAppender"/>
</root>
```

Additional Information

See the following links for additional information about configuring log4net:

- http://logging.apache.org/log4net/release/manual/configuration.html#syntax
- http://logging.apache.org/log4net/release/config-examples.html

About Dell

Dell listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.software.dell.com.

Contacting Dell

For sales or other inquiries, visit http://software.dell.com/company/contact-us.aspx or call 1-949-754-8000.

Technical Support Resources

Technical support is available to customers who have purchased Dell software with a valid maintenance contract and to customers who have trial versions. To access the Support Portal, go to https://support.software.dell.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. In addition, the portal provides direct access to product support engineers through an online Service Request system.

The site enables you to:

- Create, update, and manage Service Requests (cases)
- View Knowledge Base articles
- Obtain product notifications
- Download software. For trial software, go to http://software.dell.com/trials.
- View how-to videos
- Engage in community discussions
- Chat with a support engineer

Third-Party Contributions

This product contains some third-party components (listed below). Copies of their licenses may be found by referencing http://software.dell.com/legal/license-agreements.aspx. Source code for components marked with an asterisk (*) is available at http://opensource.dell.com.

Table 1. List of third-party contributions

Component	License or Acknowledgement
AlphaFS 2.0.1	MIT License
Angular.js 1.2.16	MIT License
AutoMapper 3.2.1	MIT License
Backbone.js 1.0.0	MIT License
Common.Logging 2.1.2	Apache 2.0 License
Google APIs Client Library for .NET 1.8.2	Apache 2.0 License
Google Data API SDK (1.8.0.0) Setup 1.8	Apache 2.0 License
Google Data API SDK (2.2.0.0) Setup 2.2.0.0	Apache 2.0 License
JQuery 1.7.1	MIT License
JQuery 1.8.2	MIT License
JQuery 2.1.0	MIT License
JQuery Form 3.46	MIT License
jQuery-Placeholder 2.0.7	MIT License
Json.net 6.0.8	MIT License
Log4Net 2.0.3	Apache 2.0 License
Moment.js 2.6.0	MIT License
Newtonsoft.Json.dll 6.0.5	MIT License
Newtonsoft.Json.dll 6.0.8	MIT License
Quartz.NET 2.2.3	Apache 2.0 License
RestSharp 105.0.1	Apache 2.0 License
RestSharp 105.2.3	Apache 2.0 License
Select2 3.4.8	Apache 2.0 License
spin.js 1.2.4	MIT License
Twitter Bootstrap 2.1.1	Apache 2.0 License
Twitter Bootstrap 2.3.1	Apache 2.0 License
Underscore.js 1.5.1	MIT License
Underscore.string 2.3.0	MIT License
WebGrease 1.3	Apache 2.0 License
ZimbraCSharpClient 5.0.96.0	Mozilla Public License (MPL) 1.1