



INTEGRATION MODULE FOR KEVERION AUTOMATION PORTAL

For Keverion Runbook Studio and Azure Automation

User Guide

Version 1.0

Microsoft
Azure

Certified

Kelverion Integration Module for Kelverion Automation Portal

Copyright © 2023 Kelverion Inc. All rights reserved.

Published: October 2023

[*Feedback*](#)

Send suggestions and comments about this document to support@kelverion.com

Contents

Getting Started.....	4
System Requirements.....	4
Deploying the Integration Module	4
Using the PowerShell Gallery.....	4
Manual Installation	5
Licensing the Integration Module	5
Working with Activities in Runbook Studio.....	7
Smart Connections	7
Global Connection Assets.....	8
Activity Properties	9
Smart Discovery.....	9
Smart Parameters.....	9
Smart Filters	11
Retry Behavior	11
Additional Parameters.....	12
Approve-AutomationRequest	13
Get-AutomationRequest.....	13
Get-AutomationRequestData	15
New-AutomationRequestHistory.....	16
Deny-AutomationRequest	17
Set-AutomationRequest.....	17

Getting Started

The following sections outline how to deploy and configure the Keverion Integration Module for Keverion Automation Portal.

System Requirements

The Integration Module for Keverion Automation Portal requires the following software to be installed and configured prior to implementing the integration. For more information on installing Keverion Runbook Studio, please refer to the Keverion Runbook Studio User Guide.

- Keverion Runbook Studio
- Microsoft .NET Framework 4.7.2
- Keverion Automation Portal for Azure

Deploying the Integration Module

The easiest way to install and deploy the Integration Module for Keverion Automation Portal is from the PowerShell Gallery, but you can also download the module from Keverion and perform the steps manually.

You must install and deploy the Integration Module to each Azure Automation Account and hybrid runbook worker host system that you plan to use to run your runbooks. You must also install the Integration Module on any Runbook Studio host systems that you will be using to build and manage your runbooks.

Using the PowerShell Gallery

Using the commands in the **PowerShellGet** module you can download the Keverion Integration Module for Keverion Automation Portal from the PowerShell Gallery and install it on your local computer. You can also deploy the module directly from the PowerShell Gallery to any of your Azure Automation Accounts.

Install the Integration Module on your local computer or hybrid runbook worker:

1. Confirm that the PowerShellGet module is installed.
2. Start a PowerShell window as Administrator and run the command:

```
Install-Module -Name Keverion.AutomationPortal -Scope  
AllUsers
```

Upload the integration module to an Azure Automation account:

1. Go to the [PowerShell Gallery](#).
2. Click the **Azure Automation** tab.
3. Click **Deploy to Azure Automation**. You will be directed to Microsoft Azure.
4. Select the **Automation Account** that you want to deploy the module to.

5. Click **OK**.

Manual Installation

Alternatively, you can download the Integration Module package from Keverion and deploy it manually to your local computer, hybrid workers and Automation Accounts.

The download package from Keverion includes a **.zip** file containing the Integration Module as well as the User Guide and Release Notes. The following instructions assume that you have unzipped the download package and have access to the **.zip** file containing the Integration Module.

Important: When installing the Integration Module on a hybrid runbook worker, you must use a location that is accessible to all users of the computer.

Install the integration module on your local computer or hybrid runbook worker:

1. Copy the **Keverion.AutomationPortal.zip** file to your local computer.
2. Right click on the file and select **Properties**.
3. Click the **General** tab. If necessary, click **Unblock**, and then click **OK**.
4. Unzip the **Keverion.AutomationPortal.zip** file.
5. Copy the **Keverion.AutomationPortal** folder to a location in the %PsModulePath% path.

Upload the integration module to an Azure Automation Account:

1. Sign into [Microsoft Azure](#).
2. Open the Automation Account that you want to upload the module to.
3. Click **Modules** under Shared Resources. The list of installed modules is displayed.
4. Click **Add a module** at the top of the list.
5. In the **Upload File** box, select the **Keverion.AutomationPortal.zip** file that you downloaded.
6. Click **OK**. Importing the module may take several minutes.

Licensing the Integration Module

Licenses for Keverion Integration Modules are managed and deployed using the *Keverion Runbook Studio* and *Automation Connection Assets*.

Important: Entitlements will not display until after the Integration Module has been installed on the Runbook Studio computer.

Register an Integration Module license with Runbook Studio:

1. Open **Keverion Runbook Studio**.
2. On the **File** tab, click **About**.
3. Click **License Information**.
4. Click the **Integration Modules** tab, and then click **Add License**.

5. Select the integration module license file (.kaml) and click **Open**.
6. You should see your entitlements displayed in the list.
7. Click **OK**.

Create a Connection Asset with a license key and upload it to Azure:

1. On the **Home** tab, click **Sign In**. The Sign In dialog appears.
2. Sign into your account.
3. In the **Active Azure Automation Account** box, select the account that you want to add the connection asset to.
4. Click **New Asset** and then click **Connection**. The New Connection dialog appears.
5. In the **Name** field, enter a name to identify the connection.
6. In the **Connection Type** field, select the desired connection type.
7. Enter the appropriate connection information in the provided fields.
8. Click **OK**.

Update all Connection Assets license keys and upload them to Azure:

1. On the **Home** tab, click **Sign In**. The Sign In dialog appears.
2. Sign into your account.
3. In the Explorer panel, click the **Azure (Online)** group.
4. Right-click the Azure Automation Account that contains the connection assets you want to update, and then click **Update License Keys**. A summary is displayed.

Working with Activities in Runbook Studio

The following sections outline some of the common configuration options that are available to you when working with the activities in the Keverion Integration Module for Keverion Automation Portal.

The integration module includes the following activities:

Get-AutomationRequest	Retrieves and filters Automation Portal requests.
Get-AutomationRequestData	Retrieves an Automation Portal request specific data.
Set-AutomationRequest	Updates an existing Automation Portal request
Approve-AutomationRequest	Approves an existing Automation Portal request
Deny-AutomationRequest	Deny an existing Automation Portal request.
New-AutomationRequestHistory	Creates a new history record for the specified Automation Portal request.

The advanced discovery capabilities provided by the activities in this integration module are only supported when authoring runbooks in Keverion Runbook Studio.

When you publish your runbooks from Keverion Runbook Studio to Azure Automation or when you generate PowerShell code snippets for Service Management Automation, Runbook Studio will automatically convert the dynamically generated parameters and filters of Smart activities into the parameters provided by the underlying command activities.

Smart Connections

In Keverion Runbook Studio you can configure one or more Smart Connections to establish reusable links between Runbook Studio and a specific Keverion Automation Portal instance. You can create as many Smart Connections as you require, specifying links to multiple instances. You can also create multiple Smart Connections to the same instance to allow for differences in security privileges for different user accounts.

Add a Smart Connection in Keverion Runbook Studio:

1. Click **Connections** in the Runbook Studio toolbar.
2. In the **Smart Connections** dialog click **Add**.
3. In the **Name** box, type a name for the configuration. This could be the name of the instance or a descriptive name to distinguish the type of configuration.
4. In the **Connection Type** box, select **Keverion.AutomationPortal**.
5. In the **PortalUrl** box, type the URL of the Keverion Automation Portal instance.
6. In the **TenantId** box, type the TenantId provided by your Keverion Automation Portal administrator.

7. In the **ClientId** box, type the ClientId provided by your Keverion Automation Portal administrator.
8. In the **Username** and **Password** boxes, type the credentials that an activity will use to connect to the Automation Portal instance. **It is highly recommended to use a dedicated Automation Portal user for your smart connection, to ensure proper runbook operation during design and runtime.**
9. Click **OK** to close the configuration dialog box, and then click **OK**.

Global Connection Assets

The activities in the Keverion Integration Module for Keverion Automation Portal require connection information to connect to instances of Automation Portal as well as the Keverion Management server.

The recommended way to pass connection information to your activities in your runbooks is to use Global Connection Assets. Global connection assets let you securely define connection information in Azure which can then be retrieved on demand using either the *Get-AutomationConnection* cmdlet or Connection Asset Data Source.

Add a global connection asset in Runbook Studio:

1. In Keverion Runbook Studio, click the **Azure** panel.
2. Select your Azure subscription.
3. Select your Automation account.
4. Select **Connections** and right click.
5. Select **Add New Connection**.
6. In the **Name** box, type a name for the configuration. This could be the name of the instance or a descriptive name to distinguish the type of configuration.
7. In the **Connection Type** box, select *Keverion.AutomationPortal*.
8. In the **PortalUrl** box, type the URL of the Keverion Automation Portal instance.
9. In the **TenantId** box, type the TenantId provided by your Keverion Automation Portal administrator.
10. In the **ClientId** box, type the ClientId provided by your Keverion Automation Portal administrator.
11. In the **Username** and **Password** boxes, type the credentials that an activity will use to connect to the Automation Portal instance. **It is highly recommended to use a dedicated Automation Portal user for your connection, to ensure proper runbook operation during design and runtime.**
12. Click **OK** to close the New Connection dialog box.

Activity Properties

All activities in the Keverion Integration Module for Keverion Automation Portal have the following properties:

Label	A unique label that identifies the activity in the runbook. Runbook Studio will provide a default name for each activity, but you can provide your own labels to make their role in the runbook more obvious.
Description	An optional description of the activity. Providing a description is a fantastic way to let everyone understand the function of the activity in the runbook.
Checkpoint	Indicates whether a checkpoint is set in the runbook workflow after the activity runs. Checkpoints are only available for Graphical PowerShell Workflow runbooks. If the runbook uses Azure cmdlets, you should follow best practices and follow a check-pointed activity with an <i>Add-AzureRMAccount</i> in case the runbook is suspended and restarts from this checkpoint on a different worker.

Smart Discovery

When designing runbooks in Keverion Runbook Studio, you will notice that the activities in the Keverion Integration Module for Keverion Automation Portal include a **Discovery** panel instead of the **Parameter Sets** panel that is present for standard command activities. This is because the activities in the Keverion Integration Module for Keverion Automation Portal support interactive discovery of the Automation Portal assets in your environments.

All activities in the Keverion Integration Module for Keverion Automation Portal have a **Connection** option on the **Discovery** panel which lets you specify how Runbook Studio should connect to Automation Portal.

When connected to Automation Portal, Runbook Studio will provide additional discovery options. Once you have filled in the discovery options Runbook Studio will provide additional parameters and, in some cases, filters which can be used to configure the activity.

Smart Parameters

Unlike standard command activities, whose parameters are determined by the Parameter Set that is selected, the parameters in the Keverion Integration Module Keverion Automation Portal are determined by the Discovery options that you specify.

You must configure all mandatory parameters. To view the optional parameters that are associated with an activity, click **Optional** at the top of the Parameters tab.

In addition, all activities in the Keverion Integration Module for Keverion Automation Portal include a **Connection** parameter which is used to specify information that the activity will use to connect to Automation Portal when it is executed as part of a runbook running on a hybrid runbook worker. Typically, you will assign a Connection Asset data source to this parameter so that the activity can securely use connection information stored in Azure.

The Connection parameter should not be confused with the similarly named Connection option on the Discovery panel which is used to specify how Runbook Studio connects to Automation Portal to provide design-time configuration options.

Several factors determine the data sources that are available when configuring a parameter. They include: the parameter's data type, whether it is linked to another activity and whether the runbook has any input parameters.

Runbook studio supports the following data sources.

Activity output	<p>Specify activity whose output will be assigned to the parameter. You may also provide an optional Path to select a specific property of the output objects that are generated by the activity.</p> <p>Available when the activity is linked to a source activity.</p>
Not configured	<p>Clears any value that was previously configured. You must configure all mandatory parameters.</p>
Certificate asset	<p>Specify the name of the global certificate asset that will be used to provide a value for the parameter.</p> <p>If you have connected to Azure and selected a Subscription and Automation Account on the toolbar, the data source will provide the names of the certificates that are available.</p>
Credential asset	<p>Specify the name of the global credential asset that will be used to provide a value for the parameter.</p> <p>If you have connected to Azure and selected a Subscription and Automation Account on the toolbar, the data source will provide the names of the credentials that are available.</p>
Constant	<p>Specify a constant value to assign to the parameter.</p> <p>Available for parameters that have the following data types:</p> <ul style="list-style-type: none">• String• DateTime• Timespan• Decimal• Double <p>When assigning a constant DateTime and Time values, Runbook Studio assumes the value is in UTC.</p>
Connection asset	<p>Specify the name of the global connection asset that will be used to provide a value for the parameter.</p> <p>If you have connected to Azure and selected a Subscription and Automation Account on the toolbar, the data source will provide the names of the connections that are available.</p>
Empty string	<p>An empty string will be assigned to the parameter. Available when the parameter is type <i>System.String</i></p>

Null	A null (\$null) value will be assigned to the parameter. Available when the parameter type is a reference type.
PowerShell expression	Specify a <i>simple</i> PowerShell expression whose output will be assigned to the parameter. You can use variables in the expression to access the output of an activity or a runbook parameter.
Runbook input	Specify the name of the runbook input parameter whose value will be assigned to the parameter. Available when the runbook has one or more input parameters.
Variable asset	Specify the name of the global variable asset that will be used to provide a value for the parameter. If you have connected to Azure and selected a Subscription and Automation Account on the toolbar, the data source will provide the names of the variables that are available.

Smart Filters

Some of the activities in the Keverion Integration Module for Keverion Automation Portal include a **Filters** panel which lets you specify filters that can be used to retrieve specific records in Automation Portal.

To add a filter to your activity, select the **Filters** panel and click **Add**. Filters have the following properties.

Filter	The name of the filter.
Operation	The operation that is used to evaluate the filter. Different operators will be provided based on the filter that is selected. Operators include: <ul style="list-style-type: none"> • Equals • Is less than • Is greater than • Contains • Starts with
Value	The data source used to retrieve the value used to evaluate the filter. The value used to evaluate the filter will be obtained. For more information on data sources, please refer to the Parameters section for more information on configuring data sources.

Retry Behavior

The activities in the Keverion Integration Module Keverion Automation Portal can be configured to run multiple times until a condition, which you specify, is satisfied. You can use the retry behavior options to configure activities that should run multiple times, which are error prone or may need more than one attempt for success.

When you enable retry for an activity, you can configure the runbook to wait a specified number of minutes or seconds before running the activity again. If no delay is specified the runbook will run the activity again, immediately after it is completed.

The retry condition lets you specify a PowerShell expression that the runbook will evaluate after each time the activity runs. If the result of the expression is true the activity does not run again, and the runbook moves on to the next child activity in the runbook.

When defining the retry conditions for your activity you can take advantage of a global variable called **\$RetryData**. Specific information about the last time the activity ran can be accessed using the following properties.

NumberOfAttempts	Number of times that the activity has ran
Output	Output that was generated by the activity the last time that it ran
TotalDuration	Time elapsed since the activity was started
StartedAt	Time in UTC when the activity was first started

The following are some examples of activity retry conditions.

```
# Run the activity exactly five times
$RetryData.NumberOfAttempts -eq 5

# Run the activity until it produces some output
$RetryData.Output.Count -ge 1

# Run the activity until at least 2 minutes has elapsed
$RetryData.TotalDuration.TotalMinutes -ge 2
```

Additional Parameters

The activities in the Kelverion Integration Module for Kelverion Automation Portal let you specify additional PowerShell parameters that you can use to control the behavior of the activity.

For example, to output detailed information about the operation performed by an activity you would specify **-Verbose:\$True**

Approve-AutomationRequest

The Approve-AutomationRequest activity approves an existing Automation Portal request.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Keverion Automation Portal.
-------------------	--

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Keverion Automation Portal.
Request ID	Identifies the request to be approved.

Outputs

This activity provides the following outputs.

Request ID	Request identifier.
-------------------	---------------------

Get-AutomationRequest

The Get-AutomationRequest activity retrieves and filters Automation Portal requests.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Keverion Automation Portal.
Search By	The method used to retrieve the Business Objects. Select Filters to use filters or Request ID to get by request ID.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Automation Portal.
Request ID	The request IDs of the requests to retrieve.

Optional Parameters

This activity provides the following optional parameters.

Limit	The maximum number of requests to retrieve. When zero is specified, all requests are retrieved. The default value is one thousand.
Ascending	Indicates that request should be sorted in ascending order. This parameter is used only when Order By is specified.
Order By	The field used to order the requests by.

Filters

This activity provides the following filters.

Created	Filter by request create date.
External ID	Filter by external ID.
Message	Filter by message.
Offering ID	Filter by offering ID (if offering not specified at discovery).
Offering Name	Filter by offering name (if offering not specified at discovery).
Requested By	Filter by requester.
Runbook Owner	Filter by runbook owner.
Service ID	Filter by service ID (if service not specified at discovery).
Service Name	Filter by service Name (if service not specified at discovery).
State	Filter by request state.
State Reason	Filter by request state reason.
Tag	Filter by tag.
Updated	Filter by request update date.

Output

This activity provides the following outputs.

Can Approve	Request can be approved.
Created	Created date time of the request.
Can Set Reject Reason	Reject Reason can be set on the request.
External ID	Request external ID.
Message	Request message.
Offering Id	Request offering ID.
Offering Name	Request offering.
Request ID	Request ID.
Requested By	Requestor.
Requires	Specifies if the request requires approval.

Approval	
Runbook Owner	Runbook owner.
Service ID	Request service ID.
Service Name	Request service.
State	Request state
State Reason	Request state reason.
Tag	Request tag.
Updated	Request update date time.

Remarks

We **highly recommend** specifying a limit when retrieving requests from Automation Portal to avoid the negative performance associated with retrieving large request sets.

Get-AutomationRequestData

The Get-AutomationRequestData activity retrieves a request and publishes offering-specific request data. The activity defines smart activity outputs based on the offering that is selected during discovery.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Kolverion Automation Portal.
Service	Service containing the request offering or offering folder.
Offering Folder	Folder containing the request offering.
Offering	Request offering. Used to define smart activity outputs for offering-specific request data.
Include Secure Data	Determines if Secure Text Box Field data is returned with request data.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kolverion Automation Portal.
Request ID	Identifies the request

Outputs

This activity publishes offering-specific request data. The outputs are dynamic based on the offering that is specified at Discovery time. The field names used in the Automation Portal are used as output parameter names.

Date Field

Date fields are returned as DateTime objects and have the time set to 12:00:00 AM.

List (Multiple Selection) Field

List (Multiple Selection) fields are returned as string array of the selected values.

Complex Field Types

The following field types are returned as complex objects: Table (Multiple Selection), Table (Single Selection) and Time.

Table (Multiple Selection)

Table (Multiple Selection) fields are returned as a PSObject array with each PSObject representing a selected row in the table. The column names of the table are used for PSObject property names.

Table (Single Selection)

Table (Single Selection) fields are returned as a PSObject representing the selected row in the table. The column names of the table are used for PSObject property names.

Time

Time fields are returned as a PSObject with the following fields and types.

Time	TimeSpan
Time Zone Offset	TimeSpan

New-AutomationRequestHistory

The New-AutomationRequestHistory activity creates a new history record for the specified Automation Portal request.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Kolverion Automation Portal.
-------------------	---

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kolverion Automation Portal.
-------------------	--

Request ID	Identifies the request to add history for.
Action	Action description of the new history record.

Outputs

This activity provides the following outputs.

Request ID	Request identifier.
-------------------	---------------------

Deny-AutomationRequest

The Deny-AutomationRequest activity deny an existing Automation Portal request.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Kelverion Automation Portal.
-------------------	---

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
Request ID	Identifies the request to be denied.

Optional Parameters

This activity provides the following optional parameters.

Reason	Specifies the reason the request is being denied.
---------------	---

Outputs

This activity provides the following outputs.

Request ID	Request identifier.
-------------------	---------------------

Set-AutomationRequest

The Set-AutomationRequest activity updates an existing Automation Portal request.

Discovery Options

This activity provides the following smart discovery options:

Connection	The name of the Smart Connection used to connect Runbook Studio to Keverion Automation Portal.
-------------------	--

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Keverion Automation Portal.
Request ID	Identifies the request

Optional Parameters

This activity provides the following optional parameters.

State	Specifies the request state.
State Reason	Specifies the request state reason. This parameter is used only when the State parameter is set to Rejected.
Message	Specifies the request message.
Runbook Owner	Specifies the runbook owner.

Outputs

This activity provides the following outputs.

Request ID	Request identifier.
-------------------	---------------------