Kelverion

KELVERION INTEGRATION MODULE FOR AUTOMATION PORTAL

For Kelverion Runbook Studio and Azure Automation

User Guide

Version 2.0

Microsoft Azure

Certified

Kelverion Integration Module for Automation Portal

Published: December 2024

Feedback

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

Contents

Getting Started

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

System Requirements

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

- Kelverion Runbook Studio 5.6
- Microsoft .NET Framework 4.7.2
- Kelverion Automation Portal 4.1

Deploying the Integration Module

The easiest way to install and deploy the Integration Module for Kelverion Automation Portal is from the PowerShell Gallery, but you can also download the module from Kelverion 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 Integration Module for Kelverion 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 Kelverion.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 Kelverion and deploy it manually to your local computer, hybrid workers and Automation Accounts.

The download package from Kelverion 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 **Kelverion.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 **Kelverion.AutomationPortal.zip** file.
5. Copy the **Kelverion.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 **Kelverion.AutomationPortal.zip** file that you downloaded.
6. Click **OK**. Importing the module may take several minutes.

Licensing the Integration Module

Licenses for Kelverion Integration Modules are managed and deployed using the *Kelverion 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 **Kelverion 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 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 Integration Module for Kelverion Automation Portal.

The integration module includes the following activities:

Approve-AutomationRequest	Approves Automation Portal request that is pending approval
Deny-AutomationRequest	Rejects an Automation Portal request that is pending approval
Get-AutomationAttachmentContent	Downloads the contents from an Automation Portal attachment
Get-AutomationOffering	Gets Automation Portal service offerings
Get-AutomationRequest	Gets Automation Portal requests
Get-AutomationRequestAttachment	Gets attachment records for an Automation Portal request
Get-AutomationRequestData	Retrieves field data from an Automation Portal request
Get-AutomationRequestHistory	Gets history records for an Automation Portal request
Get-AutomationService	Gets Automation Portal services
New-AutomationRequest	Creates a new Automation Portal request
New-AutomationRequestHistory	Creates a new history record for the specified Automation Portal request.
Remove-AutomationAttachment	Removes one or more Automation Portal attachments
Remove-AutomationRequest	Removes one or more Automation Portal requests
Set-AutomationRequest	Updates an Automation Portal request

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

When you publish your runbooks from Kelverion 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 Kelverion Runbook Studio you can configure one or more Smart Connections to establish reusable links between Runbook Studio and a specific Kelverion 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 Kelverion 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 **Kelverion.AutomationPortal**.
5. In the **PortalUrl** box, type the URL of the Kelverion Automation Portal instance.
6. In the **TenantId** box, type the TenantId provided by your Kelverion Automation Portal administrator.
7. In the **ClientId** box, type the ClientId provided by your Kelverion Automation Portal administrator.
8. In the **ClientSecret** box, type the client secret provided by your Kelverion Automation Portal administrator.
9. Click **OK** to close the configuration dialog box, and then click **OK**.

Global Connection Assets

The activities in the Integration Module for Kelverion Automation Portal require connection information to connect to instances of Automation Portal as well as the Kelverion 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 Kelverion 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 *Kelverion.AutomationPortal.*
8. In the **PortalUrl** box, type the URL of the Kelverion Automation Portal instance.

9. In the **TenantId** box, type the TenantId provided by your Kelverion Automation Portal administrator.
10. In the **ClientId** box, type the ClientId provided by your Kelverion Automation Portal administrator.
11. In the **ClientSecret** box, type the client secret provided by your Kelverion Automation Portal administrator.
12. Click **OK** to close the New Connection dialog box.

Activity Properties

All activities in the Integration Module for Kelverion 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 *Add-AzureRMAccount* in case the runbook is suspended and restarts from this checkpoint on a different worker.

Smart Discovery

When designing runbooks in Kelverion Runbook Studio, you will notice that the activities in the Integration Module for Kelverion 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 Integration Module for Kelverion Automation Portal support interactive discovery of the Automation Portal assets in your environments.

All activities in the Integration Module for Kelverion 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 Integration Module for Kelverion 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 Integration Module for Kelverion 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	**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.** **Available when the activity is linked to a source activity.**
Not configured	Clears any value that was previously configured. You must configure all mandatory parameters.
Certificate asset	Specify the name of the global certificate 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 certificates that are available.
Credential asset	Specify the name of the global credential 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 credentials that are available.
Constant	Specify a constant value to assign to the parameter. Available for parameters that have the following data types: • String • DateTime • Timespan • Decimal • Double When assigning a constant DateTime and Time values, Runbook Studio assumes the value is in UTC.
Connection asset	Specify the name of the global connection 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 connections that are available.

Empty string	An empty string will be assigned to the parameter. Available when the parameter is type *System.String*
Null	A null ($null) value will be assigned to the parameter. Available when the parameter type is a reference type.
PowerShell expression	Specify a *simple* 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 Integration Module for Kelverion Automation Portal include a **Filters** panel which lets you specify filters that can be used to retrieve specific records in Automation Portal.

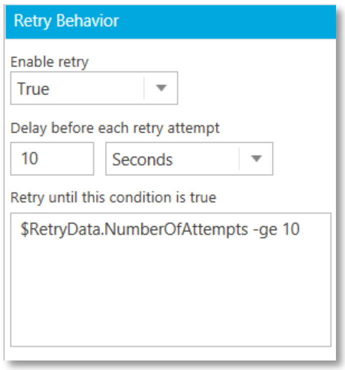
To add a filter to your activity, select the **Filters** panel and click **Add.** Filters have the following properties.

Filter	The name of the filter.
Operation	The operation is used to evaluate the filter. Different operators will be provided based on the filter that is selected. Filter operators include: • 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 Integration Module Kelverion for 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.


Retry Behavior

Enable retry: True

Delay before each retry attempt: 10 Seconds

Retry until this condition is true: $RetryData.NumberOfAttempts -ge 10

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 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`**

Activity Reference

Approve-AutomationRequest

The **Approve-AutomationRequest** activity is used in a runbook to approve an Automation Portal request that is the Pending Approval state.

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.
RequestId	Specifies the ID of the request to be approved.

Outputs

This activity returns an object that represents the Automation Portal request that was approved. Request objects have the following properties.

CanApprove	Indicates whether the request can be approved.
CostSaved	The amount saved by the request.
Created	The date/time that the request was created.
Data	The values that were assigned to the request fields.
Deleted	Indicates whether the request has been deleted.
ExternalId	The external Ref. ID that was assigned to the request.
ID	The identifier used to uniquely identify the request.
Message	The update message that was assigned to the request.
MustSetRejectReason	Indicates whether you can set a reject reason.
OfferingId	The ID of the service offering.
OfferingName	The name of the service offering.
RequestedBy	The email of the user that made the request.
RequiresApproval	Indicates whether the requires approval.
RunbookOwner	The name of the runbook owner.

ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
State	The current state of the request.
StateReason	The reason for the current request state.
Tag	The tag assigned to the request offering.
TimeSaved	The time saved by the request.
Updated	The date/time that the request was last updated.

Deny-AutomationRequest

The **Deny-AutomationRequest** activity is used in a runbook to reject an Automation Portal request that is the Pending Approval state.

Discovery Options

This activity does not provide any discovery options.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
RequestId	Specifies the ID of the request to reject.

Optional Parameters

This activity provides the following optional parameters.

Reason	Specifies the reason for rejecting the request.

Outputs

This activity returns an object that represents the Automation Portal request that was rejected. Request objects have the following properties.

CanApprove	Indicates whether the request can be approved.
CostSaved	The amount saved by the request.
Created	The date/time that the request was created.
Data	The values that were assigned to the request fields.
Deleted	Indicates whether the request has been deleted.
ExternalId	The external Ref. ID that was assigned to the request.
ID	The identifier used to uniquely identify the request.
Message	The update message that was assigned to the request.
MustSetRejectReason	Indicates whether you can set a reject reason.
OfferingId	The ID of the service offering.
OfferingName	The name of the service offering.
RequestedBy	The email of the user that made the request.
RequiresApproval	Indicates whether the requires approval.
RunbookOwner	The name of the runbook owner.

ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
State	The request state.
StateReason	The reason for the current request state.
Tag	The tag assigned to the request offering.
TimeSaved	The time saved by the request.
Updated	The date/time that the request was last updated.

Get-AutomationAttachmentContent

The **Get-AutomationAttachmentContent** activity is used in a runbook to download the content of a request attachment.

Discovery Options

This activity does not provide any discovery options.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
RequestId	Specifies the ID of the attachment to download from.

Outputs

This activity outputs the content of the attachment as an array of bytes.

Get-AutomationOffering

The **Get-AutomationOffering** activity is used in a runbook to retrieve offering records from the Automation Portal. You can retrieve a specific offering by its unique ID, retrieve all offerings or retrieve a subset of offerings using filters.

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.
Search By	Indicates whether to retrieve an offering by its **Offering ID** or to retrieve a collection of offerings using **Filters**.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Automation Portal.
OfferingId	Specifies the ID of the offering to retrieve. Available when **Search By** is set to **Offering ID**.

Optional Parameters

This activity provides the following optional parameters when **Search By** is set to **Filters.**

Descending	Indicates that offering records should be output in descending order according to the **OrderBy** value. The default is ascending order.
Limit	The maximum number of offering records to retrieve. When zero is specified, all offerings are retrieved. The default value is **one thousand**.
OrderBy	Specifies the offering property used to order the results. The default value is **ID**.

Filters

When **Search By** is set to **Filters**, this activity provides filters that can be used to control which offering records to retrieve from the Automation Portal. The following filters are available.

Active	Filter on whether the offering is active.
CostSaved	Filter on the cost that has been saved by using the offering.
Created	Filter on the date/time that the offering was created.
CreatedBy	Filter on the user that created the offering.
Description	Filter on the offering’s description.

FolderId	Filter on the ID of the folder that contains the offering.
ID	Filter on the ID of the offering
MobileActive	Filter on whether the folder is available for mobile devices.
Name	Filter on the name given to the offering.
RequiresExternalId	Filter on whether the offering requires an External Ref. ID.
ServiceId	Filter on the ID of the service that contains the offering.
ServiceName	Filter on the name of the service that contains the offering.
Tag	Filter on the tag that has been assigned to the offering.
TimeSaved	Filter on the time, in minutes, that have been saved by using the offering.
Updated	Filter on the date/time that the offering was last updated.

Outputs

This activity generates objects that represent the Automation Portal offerings that were retrieved. Each offering object has the following properties.

Active	Indicates whether the offering is active.
CostSaved	The cost that has been saved by using the offering.
Created	The date/time that the offering was created.
CreatedBy	The user that created the Offering.
Deleted	Indicates whether the offering has been deleted.
Description	The offering's description
FolderId	The ID of the folder that contains the offering.
ID	The offering's unique ID.
MobileActive	Indicates whether the folder is available for mobile devices.
Name	The offering's name.
RequiresApproval	Indicates whether the offering requires approval.
RequiresExternalId	Indicates whether the offering requires an External Ref. ID.
ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
Tag	The tag that has been assigned to the offering.
TimeSaved	The time, in minutes, that have been saved by using the offering.
Updated	The date/time that the offering was last updated.

Get-AutomationRequest

The **Get-AutomationRequest** activity is used to retrieve requests from the Automation Portal. You can retrieve a specific request by its unique ID, retrieve all requests or retrieve a subset of requests using filters.

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.
Search By	Identifies whether to retrieve a request by its **Request ID** or to retrieve a collection of requests matching one or more **Filters**.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Automation Portal.
RequestId	Specifies the ID of the request to retrieve. Available when **Search By** is set to **Request ID**.

Optional Parameters

This activity provides the following optional parameters when **Search By** is set to **Filters.**

Descending	Indicates that request records should be output in descending order according to the **OrderBy** parameter. The default is ascending order.
Limit	The maximum number of request records to retrieve. When zero is specified, all requests are retrieved. The default value is **1000**.
OrderBy	Specifies the request property used to order the results. The default value is **ID**.

Filters

When **Search By** is set to **Filters**, this activity provides filters that can be used to control which request records are retrieved from the Automation Portal. The following filters are available.

CostSaved	Filter on the cost that has been saved by using the offering.
Created	Filter on the date/time that the request was created.
ExternalId	Filter on the External Ref. ID that was assigned to the request.
ID	Filter on the request ID.
Message	Filter on the update message assigned to the request.

OfferingId	Filter on the ID of the offering that the request is for.
OfferingName	Filter on the name of the offering that the request is for.
RequestedBy	Filter on the name of the user that made the request.
RunbookOwner	Filter on the name of the user that owns the runbook responding to the request.
ServiceId	Filter on the ID of the service that contains the offering that the request is for.
ServiceName	Filter on the name of the service that contains the offering that the request is for.
State	Filter on the state that the request is in.
StateReason	Filter the reason that has been given for the state the request is in.
Tag	Filter on the tag that has been assigned to the request.
TimeSaved	Filter on the time, in minutes, that have been saved by using the offering.
Updated	Filter on the date/time that the request was last updated.

Output

This activity returns objects that represents the Automation Portal requests that were retrieved. Request objects have the following properties.

CanApprove	Indicates whether the request can be approved.
CostSaved	The amount saved by the request.
Created	The date/time that the request was created.
Data	The values that were assigned to the request fields.
Deleted	Indicates whether the offering has been deleted.
ExternalId	The external Ref. ID that was assigned to the request.
ID	The identifier used to uniquely identify the request.
Message	The update message that was assigned to the request.
MustSetRejectReason	Indicates whether you can set a reject reason.
OfferingId	The ID of the service offering.
OfferingName	The name of the service offering.
RequestedBy	The email of the user that made the request.
RequiresApproval	Indicates whether the requires approval.
RunbookOwner	The name of the runbook owner.
ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
State	The request state.

StateReason	The reason for the current request state.
Tag	The tag assigned to the request offering.
TimeSaved	The time that has been saved by the request.
Updated	The date/time that the request was last updated

Get-AutomationRequestAttachment

The **Get-AutomationRequestAttachment** activity is used in a runbook to retrieve attachment records from the Automation Porta. You can retrieve a specific attachment by its unique ID, you can retrieve all attachment records for a request or retrieve a subset of attachment records using filters.

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
Search By	Indicates whether to retrieve an attachment by its unique **Attachment ID** or to retrieve a collection of attachments using **Filters**.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Automation Portal.
AttachmentId	Specifies the ID of the request attachment to retrieve. Available when **Search By** is set to **Attachment ID**.
RequestId	Specifies the ID of the request from which to retrieve attachments. Available when **SearchBy** is set to **Filters**.

Optional Parameters

This activity provides the following optional parameters when **Search By** is set to **Filters.**

Descending	Indicates that attachment records should be output in descending order according to the **OrderBy** parameter. The default is ascending order.
Limit	The maximum number of attachments records to retrieve. When zero is specified, all records are retrieved. The default value is **1000**.
OrderBy	Specifies the attachment property used to order the results. The default value is **ID**.

Filters

When **Search By** is set to **Filters**, this activity provides filters that can be used to control which attachment records to retrieve from the Automation Portal. The following filters are available.

ContentType	Filter on the MIME content type.
Created	Filter on the date/time that the attachment was created.
FieldId	Filter on the attachment field that the attachment is for.
ID	Filter on the attachment ID.

Name	Filter on the name assigned to the attachment.

Output

This activity returns objects that represents the Automation Portal attachments that were retrieved. Request objects have the following properties.

Content	The content of the attachment, encoded as a Base64, UTF-8 string. Only available when **SearchBy** is set to **Attachment ID**.
ContentType	The MIME content type assigned to the attachment.
Created	The date/time that the attachment was created.
Deleted	Indicates whether the attachment has been deleted.
FieldId	The ID of the field that the attachment is for.
ID	The ID of the attachment record.
Name	The name given to the attachment.
RequestId	The ID of the request that the attachment is for.

Get-AutomationRequestData

The **Get-AutomationRequestData** activity is used in a runbook to retrieve request data from the Automation Portal. When used with **Kelverion Runbook Studio**, this activity uses discovery to automatically generate appropriate outputs for the selected offering, making it much easier to use request data in child activities.

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.
Service	Specifies the service that contains the offering that you want to use.
Folder	Specifies the service folder that contains the offering that you want to use. If the offering that you want to use is not in a folder, select **(None)**.
Offering	Specifies the offering that you want to use.
Include Secure Data	Indicates whether you want to retrieve data from Secure Text Box fields. Only provided when the selected offering contains secure fields. Only available when the selected offering contains one or more secure text box fields.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
RequestId	Specifies the ID of the request that you want to retrieve data for.

Outputs

This activity generates a custom where the object's properties correspond to the fields in the selected offering. Property values are determined by the underlying field type.

Date	System.DataTimeOffset
DateTime	System.DateTimeOffset
File Attachment	A custom object represent an attachment object. Each object has **ID, Name, FieldId** and **ContentType** properties.
Hidden	System.String
ListMultipleSelection	An array of System.String objects
ListSingleSelection	System.String
Radio Button	System.String
SecureTextBox	System.String

Switch	System.Boolean
TableMultipleSelection	An array of custom objects representing the table rows. The object properties correspond the columns in the table.
TableSingleSelection	A custom object representing a table row. The object properties correspond to the columns in the table
TextArea	System.String
TextBox	System.String
Time	System.TimeSpan

Get-AutomationRequestHistory

The **Get-AutomationRequestHistory** activity is used in a runbook to retrieve request history records for an Automation Portal request. You can use filter criteria to control which history records to retrieve.

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.
RequestId	Specifies the ID of the request that you want to retrieve history records for.

Optional Parameters

This activity provides the following optional parameters.

Descending	Indicates that history records should be sorted in descending order. The default is ascending order.
Limit	The maximum number of history records to retrieve. When zero is specified, all records are retrieved. The default value is **1000**.
OrderBy	Specifies the history record property used to order the results. The default value is **ID**.

Filters

This activity provides filters that can be used to control which history records to retrieve from the Automation Portal. The following filters are available.

Action	Filter on the action assigned to the history record.
Created	Filter on the date/time that the history record was created.
CreatedBy	Filter on the user that created the history record.
ID	Filter on the request history ID.

Output

This activity generates objects that represent the request history records that were retrieved. Each history record has the following properties.

Action	The action assigned to the history record.
Created	The date/time that the history record was created.
CreatedBy	The user that created the history record.
ID	The ID of the history record.
RequestId	The ID of the request that the history record is for

Get-AutomationService

The **Get-AutomationService** activity is used in a runbook to retrieve service records from the Automation Portal. You can retrieve a specific service by its unique ID, all services or a subset of services using filters.

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.
Search By	Indicates whether you want to retrieve a specific service by its unique **Service ID** or retrieve a collection of services matching one or more **Filters**.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
ServiceId	Specifies the ID of the service that you want to retrieve from the Automation Portal. Available when **Search By** is set to **Service ID**.

Optional Parameters

This activity provides the following optional parameters when **Search By** is set to **Filters.**

Descending	Indicates that service records should be output in descending order. The default is ascending order.
Limit	The maximum number of service records to retrieve. When zero is specified, all requests are retrieved. The default value is **1000**.
OrderBy	Specifies the service property used to order the results. The default value is **ID**.

Filters

This activity provides filters that can be used to control which service records to retrieve from the Automation Portal. The following filters are available.

Active	Filter on whether the service is active.
Created	Filter on the date/time that the service was created.
Description	Filter on the description given to the service.
Name	Filter on the name given to the service.
Update	Filter on the date/time that the service was last updated.

Output

This activity generates objects that represent the service records that were retrieved from the Automation Portal. Each object has the following properties.

Active	Indicates whether the service is active.
Created	The date/time that the service was created.
CreatedBy	The user that created the service.
Deleted	Indicates whether the service has been deleted.
Description	The description given to the service.
ID	The unique ID used to identify the service.
Name	The name given to the service.
Update	The date/time that the service was last updated.

New-AutomationRequest

The **New-AutomationRequest** activity is used in a runbook to submit a new request to the Automation Portal. When used with **Kelverion Runbook Studio**, this activity uses discovery to automatically generate appropriate inputs for the selected offering, making it easy to submit new requests.

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.
Service	Specifies the service that contains the offering that you want to use.
Folder	Specifies the service folder that contains the offering that you want to use. If the offering that you want to use is not in a folder, then select **(None)**.
Offering	Specifies the offering that you want to submit the request for.

Parameters

This activity provides required and optional parameters based on the fields in the offering that you selected. Parameter inputs are determined by the underlying field type.

<table>
<tr><td><b>Date</b></td><td>System.DataTimeOffset, System.DateTime or a date/time string in ISO-8601 or RFC1123 formats. The time portion of the value is ignored.</td></tr>
<tr><td><b>DateTime</b></td><td>System.DataTimeOffset, System.DateTime or a date/time string in ISO-8601 or RFC1123 formats.</td></tr>
<tr><td><b>FileAttachment</b></td><td>A Hashtable or PSObject. The hashtable/object has the following properties:
<table>
<tr><td><b>name</b></td><td>The name given to the attachment</td></tr>
<tr><td><b>content</b></td><td>The attachment content as a Base-64 encoded string</td></tr>
<tr><td><b>contentType</b></td><td>The MIME type (optional).</td></tr>
</table>
You can also use a JSON formatted string (See remarks for the JSON schema).</td></tr>
<tr><td><b>Hidden</b></td><td>System.String</td></tr>
<tr><td><b>ListMultipleSelection</b></td><td>An array of System.String objects or a comma-delimited string.</td></tr>
<tr><td><b>ListSingleSelection</b></td><td>System.String</td></tr>
<tr><td><b>Radio Button</b></td><td>System.String</td></tr>
<tr><td><b>SecureTextBox</b></td><td>System.String</td></tr>
<tr><td><b>Switch</b></td><td>System.Boolean</td></tr>
</table>

TableMultipleSelection	An array of Hashtable and/or PSObject objects, where keys correspond to the columns in the rows you want to insert. You can also use a JSON formatted string (See remarks for the JSON schema).
TableSingleSelection	A Hashtable or PSObject, where keys correspond to the columns in the row that you want to insert. You can also use a JSON formatted string (See remarks for the JSON schema).
TextArea	System.String
TextBox	System.String
Time	System.TimeSpan or a time string in the format HH:mm:ss.

Outputs

This activity returns an object that represents the Automation Portal request that was created. Request objects have the following properties.

CanApprove	Indicates whether the request can be approved.
CostSaved	The amount saved by the request.
Created	The date/time that the request was created.
Data	The values that were assigned to the request fields.
Deleted	Indicates whether the request has been deleted.
ExternalId	The external Ref. ID that was assigned to the request.
ID	The identifier used to uniquely identify the request.
Message	The update message that was assigned to the request.
MustSetRejectReason	Indicates whether you can set a reject reason.
OfferingId	The ID of the service offering.
OfferingName	The name of the service offering.
RequestedBy	The email of the user that made the request.
RequiresApproval	Indicates whether the requires approval.
RunbookOwner	The name of the runbook owner.
ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
State	The current state of the request.
StateReason	The reason for the current request state.
Tag	The tag assigned to the request offering.
TimeSaved	The time saved by the request.
Updated	The date/time that the request was last updated.

Remarks

The following JSON schemas can be used to insert complex-data, such as file attachments and table rows.

File Attachment Schema

The following JSON schema defines the format for strings used to insert attachment data.

```
{
   "$schema": "https://json-schema.org/draft/2020-12/schema",
   "$title" : "Attachment",
   "$description": "A Kelverion Automation Portal attachment record",
   "type": "object",
   "properties": {
      "name": {
         "description": "The name given to the attachment",
         "type": "string"
      },
      "contentType": {
         "description": "The MIME content-type",
         "type": "string"
      },
      "content": {
         "description": "The content as a base-64 encoded string",
         "type": "string"
      }
   },
   "required": ["name", "content"]
}
```

For example:

```
{
   "name": "Test.txt",
   "contentType": "text/plain",
   "content": "TG9yZW0gaXBzdW0gZG9sb3Igc2l0IGFtZXQ="
}
```

Table Single Selection Schema

The following JSON schema defines the format for strings used to insert data for single selection table fields.

```
{
   "$schema": "https://json-schema.org/draft/2020-12/schema",
   "$title" : "Table Row",
   "$description": "A Kelverion Automation Portal table row",
   "type": "array",
   "items": {
      "type": "object",
      "properties": {
         "name": {
            "description": "The column name",
            "type": "string"
         },
         "value": {
                "description": "The value to insert",
                "type": "string"
            }
        },
        "required": ["name", "value"]
    }
}
```

For example:

```
[
    {
            "name": "ProductID",
            "value": "707"
    },
    {
            "name": "Name",
            "value": "Sport-100 Helmet, Red"
    },
    {
            "name": "ProductNumber",
            "value": "HL-UF09-R"
    },
    {
            "name": "Color",
            "value” “Red”
    },
    {
            "name": "ListPrice",
            "value": "34.99"
    }
]
```

Table Multiple Selection Schema

The following JSON schema defines the format for strings used to insert values for multiple selection table fields.

```
{
    "$schema": "https://json-schema.org/draft/2020-12/schema",
    "$title" : "Table Row",
    "$description": "A Kelverion Automation Portal table row",
    "type": "array",
    "items": {
        "type": "array",
        "items": {
            "type": "object",
            "properties": {
                "name": {
                    "description": "The column name",
                    "type": "string"
                },
                "value": {
                    "description": "The value to insert",
                    "type": "string"
                }
            },
            "required": ["name", "value"]
        }
    }
}
```

For example:

```
[
    [
        {
                "name": "ProductID",
                "value": "707"
        },
        {
                "name": "Name",
                "value": "Sport-100 Helmet, Red"
        },
        {
                "name": "ProductNumber",
                "value": "HL-UF09-R"
        },
        {
                "name": "Color",
                "value": "Red"
        },
        {
                "name": "ListPrice",
                "value": "34.99"
        }
    ],
    [
        {
                "name": "ProductID",
                "value": "708"
        },
        {
                "name": "Name",
                "value": "Sport-100 Helmet, Black"
        },
        {
                "name": "ProductNumber",
                "value": "HL-UF09"
        },
        {
                "name": "Color",
                "value": "Black"
        },
        {
                "name": "ListPrice",
                "value": "34.99"
        }
    ]
]
```

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 Kelverion Automation Portal.

Required Parameters

This activity requires the following parameters.

Action	Specifies the action to assign the history record.
Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
RequestId	Specifies the ID of the request to create the history record for.

Outputs

This activity outputs an object the represents that request history record that was created. The history object has the following properties.

Action	The action that was assigned to the history record.
Created	The date/time that the history record was created.
Id	The ID of the history record.
RequestId	The ID of the request that the history is associated with.
User	The user that created the history record.

Remove-AutomationAttachment

The **Remove-AutomationAttachment** activity is used in a runbook to remove an attachment from the Automation Portal.

Discovery Options

This activity does not provide any discovery options.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
AttachmentId	Specifies the ID of the attachment to remove.

Outputs

This activity does not generate any output.

Remove-AutomationRequest

The **Remove-AutomationRequest** activity is used in a runbook to remove a request from the Automation Portal.

Discovery Options

This activity does not provide any discovery options.

Required Parameters

This activity requires the following parameters.

Connection	Hashtable containing information used to connect to Kelverion Automation Portal.
RequestId	Specifies the ID of the request to remove.

Outputs

This activity does not generate any output.

Set-AutomationRequest

The **Set-AutomationRequest activity** is used in a runbook to update an Automation Portal request.

Discovery Options

This activity does not provide any discovery options.

Required Parameters

This activity requires the following parameters.

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

Optional Parameters

This activity provides the following optional parameters.

State	Specifies the new state to assign to the request.
State Reason	Specifies the reason for assigning the date. This parameter is only used when setting the state to **Rejected**.
Message	Specifies a message to add to the request.
Runbook Owner	Specifies the owner of the runbook that is handling the request.

Outputs

This activity returns an object that represents the Automation Portal request that was updated. Request objects have the following properties.

CanApprove	Indicates whether the request can be approved.
CostSaved	The amount saved by the request.
Created	The date/time that the request was created.
Deleted	Indicates whether the request has been deleted.
ExternalId	The external Ref. ID that was assigned to the request.
ID	The identifier used to uniquely identify the request.
Message	The update message that was assigned to the request.
MustSetRejectReason	Identifies whether you can set a reject reason.
OfferingId	The ID of the service offering.
OfferingName	The name of the service offering.
RequestedBy	The email of the user that made the request.

RequiresApproval	Indicates whether the requires approval.
RunbookOwner	The name of the runbook owner.
ServiceId	The ID of the service that contains the offering.
ServiceName	The name of the service that contains the offering.
State	The request state.
StateReason	The reason for the current request state.
Tag	The tag assigned to the request offering.
TimeSaved	The time saved by the request.
Updated	The date/time that the request was last updated.

Notes