



AUTOMATION PORTAL

Administrator Guide

Version 4.6.3

Kelverion Automation Portal

Copyright 2024 Kelverion Inc. All rights reserved.

Published: July 2026

Feedback

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

Table of Contents

Table of Contents	3
Introduction	6
Accessibility Statement.....	7
System Requirements.....	8
System Specs	8
Web Browser	9
Installation	9
Installation Configuration.....	9
Installation Components	10
Azure App Registration.....	10
<i>Create-AppRegistration Syntax</i>	10
<i>Using the App Registration PowerShell Script</i>	12
Automation Portal Installation	13
<i>Preparing for the Installation</i>	13
<i>Running the Installer</i>	14
Automation Portal Update	15
<i>Preparing for the Update</i>	15
<i>Running the Installer</i>	19
Database Setup On-Premises	21
<i>Creating a New Database On-Premises</i>	21
<i>Updating an Existing Database On-Premises</i>	23
<i>Configuring a Database User On-Premises</i>	25
<i>Configuring Portal Administrator Group On-Premises</i>	26
<i>Configuring Proxy Settings On-Premises</i>	27
Database Setup in Azure	30
<i>Creating a New Database in Azure</i>	30
<i>Updating an Existing Database in Azure</i>	34
<i>Configuring a Managed Identity and User in Azure</i>	36
<i>Configuring Proxy Settings in Azure</i>	37
Automation Portal Roles	39
<i>Roles in Azure Entra ID</i>	40
<i>Roles in Active Directory</i>	40
Firewall Configuration	41
Proxy Configuration	41
On-Premises High Availability (HA) Deployment.....	41
<i>Automation Portal Installation (HA)</i>	42

Automation Portal Update (HA)	43
Configuring Azure IIS Website for External Access	43
Configure Inbound Port Rule	43
Configure Inbound Firewall Rule	44
Update App Registration Redirect URLs	44
Configure IIS CORS Rule	44
Configure Content Security Policy	45
Configure Automation Portal API URL	46
Update IIS Server Certificate	46
Licensing your Automation Portal	46
High Availability Licensing	46
License Notifications	47
Getting Started	48
Logging In	48
Overview of the User Interface	48
Application Bar	49
Universal Search	49
Feedback	49
Display Settings	49
Account Profile	50
Navigation Bar	50
Content View	50
Setup and Administration	52
Settings Page	52
License	52
Customization	52
Request	53
Wiki Section	53
Portal Roles	53
Maintenance Tasks	54
Services Page	54
Lists Page	58
Connections Page	58
Queries Page	60
Security Recommendations	61
Wiki Page	61
Integrations	62
Webhooks Page	62
Logs Page	63
Feedback Page	63

Permissions Page	64
Themes Page.....	65
<i>Enable Custom Themes</i>	65
<i>Import/Export</i>	65
<i>Light Mode and Dark Mode</i>	65
<i>Customize Portal Font</i>	66
<i>Customize Specific Theme Settings</i>	66
<i>Customize Portal Logo</i>	66
<i>Custom Themes Settings Reference</i>	67
About Page	68
<i>Details</i>	68
<i>Accessibility Statement</i>	68
<i>Open Source</i>	68
Importing and Exporting Resources	68
<i>Exporting Resources</i>	69
<i>Importing Resources</i>	69
Integration	70
Integration Scope	70
Integration Priority	70
Keverion Automation Monitor Service.....	71
Developer Tools	71
Enabling the Swagger Interface.....	71
Support and Guidance	72
Sending Feedback	73

Introduction

The *Keverion Automation Portal* is an easy to implement self-service front end for your automation solutions. It offers a versatile interface without the lengthy list of prerequisites required by other self-service portals available today.

The Automation Portal ethos is to *keep things simple and flexible*. The portal complements existing Automation platforms and does not attempt to replicate the automation platforms capability. Integration with the automation tools takes place via the Keverion Automation Portal REST API with the support of Keverion integrations for Automation Portal.

The Automation Portal was designed with *System Center Orchestrator and Azure Automation* in mind; however, it equally complements any automation platform or scripting language that can read and write to the Automation Portal database or that can use REST web services.

The Automation Portal is simple to implement, but more importantly it is easy for your end users to navigate, and by adding a simple web interface to your solutions you can allow users to easily interact with your automated solutions.

Accessibility Statement

The [Web Content Accessibility Guidelines \(WCAG\)](#) defines requirements for designers and developers to improve accessibility for people with disabilities. It defines three levels of conformance: Level A, Level AA, and Level AAA. The Automation Portal is partially conformant with WCAG 2.1 level AA.

Important: The Automation Portal provides features to customize themes and colors. Making changes to themes and colors may result in the Automation Portal no longer conforming to WCAG requirements.

System Requirements

The Kelverion Automation Portal has the following system requirements:

- (On-Premises) Windows Server 2022
- (On-Premises) SQL Server 2019, or SQL Server 2022
- (On-Premises) Microsoft Active Directory
- (Azure) Windows Server 2022 Datacenter Azure Edition
- (Azure) Azure Database
- (Azure) Microsoft Entra ID cloud environment with Azure global services
- Web Server (IIS)
- Web Server (IIS) website certificate
- The [ASP.NET Core 10.0 Windows Hosting Bundle](#) must be installed.
- The [CORS](#) and [URL Rewrite 2.1](#) IIS modules must be installed.
- The Server Role **Web Server (IIS) > Web Server > Application Development > ASP.NET 4.8** must be enabled.
- The Server Role **Web Server (IIS) > Web Server > Security > Windows Authentication** must be enabled when using on-premises Active Directory.
- The [Microsoft.Graph.Applications \(2.37.0\)](#) and [Microsoft.Graph.Identity.SignIns \(2.37.0\)](#) PowerShell modules must be installed when using Azure Entra ID. The following PowerShell can be used to install these modules.

```
Install-Module -Scope AllUsers -Name Microsoft.Graph.Applications -Force -RequiredVersion 2.37.0
Install-Module -Scope AllUsers -Name Microsoft.Graph.Identity.SignIns -Force -RequiredVersion 2.37.0
```

System Specs

The following are the minimum recommended system specifications for the Automation Portal on-premises installation.

Portal Version	IIS Server Windows Server 2019 Windows Server 2022	SQL Server SQL Server 2019 SQL Server 2022	SQL Database Auto grow 64MB increments Full Recovery Mode
Starter	2 Cores, 8GB RAM	2 Cores, 8GB RAM	2 GB Data 1 GB Transaction Logs
Professional	2 Cores, 8GB RAM	2 Cores, 8GB RAM	2 GB Data 1 GB Transaction Logs
Enterprise	2 Cores, 16GB RAM each HA Node	2 Cores, 16GB RAM	2 GB Data 1 GB Transaction Logs
Unlimited	4 Cores, 16GB RAM each HA Node	2 Cores, 16GB RAM	2 GB Data 1 GB Transaction Logs

For Production deployments in Microsoft Azure, we recommend:

- A minimum of D series for Virtual Machines
- A minimum of Standard for the Azure SQL Database

Web Browser

The Kelverion Automation Portal has been evaluated with the following web browsers. Whichever browser you use, you must have support for **JavaScript** and **Cookies** enabled.

- Google Chrome 150.0
- Firefox 150.0
- Microsoft Edge 150.0

Installation

Installing the Kelverion Automation portal involves the following steps:

1. Reviewing System Requirements.
2. Selecting an installation configuration.
3. Creating an Azure App Registration (Entra ID only).
4. Running the Automation Portal installer.
5. Setting up the Automation Portal Database.
6. Configuring portal users and roles.
7. Login and install license.

Installation Configuration

The Kelverion Automation Portal can be installed in one of the following configurations:

On-premises Installation, with Microsoft Active Directory

- On-prem IIS server.
- On-prem SQL server database.
- On-prem Active Directory Windows authentication.
- On-prem Active Directory groups for permission roles.
- IIS Windows Authentication role is required.

On-premises Installation, with Azure Entra ID

- On-prem IIS server.
- On-prem SQL server database.
- Azure Entra ID user authentication.
- Azure Entra ID permission roles.
- Azure app registration is required.

Azure Cloud Installation

- Azure IIS server VM.

- Azure SQL server database.
- Azure Entra ID user authentication.
- Azure Entra ID permission roles.
- Azure app registration is required.

Installation Components

To install the Automation Portal, you will need the following installation components:

- **Azure App registration PowerShell script**
 - This is only required when using Entra ID authentication/authorization.
- **Automation portal installer**
 - Required for fresh installations and for upgrade from previous versions.
- **Database configuration utility**
 - Required for database creation/upgrade and other post-installation tasks.

The first two components can be extracted from the ZIP file **Kelverion.AutomationPortal.Installer-4.x.zip**. The database configuration utility will be deployed by the portal installer.

Unblock ZIP: Your **Kelverion.AutomationPortal.Installer-4.x.zip** file may have been blocked if downloaded from the internet. Make sure you unblock the zip file before extracting the required installation components. You can unblock the ZIP file from the General tab, Security section, in the ZIP file properties.

Azure App Registration

Entra ID: This is only required when using Azure Entra ID authentication. On-premises installations that use Active Directory and Windows authentication do not require an Azure app registration.

When using Microsoft Entra ID, the Automation Portal requires an Azure app registration. Use the provided **Create-AppRegistration.ps1** script to assist in creating the app registration. The script creates and configures a new Entra ID app registration that the portal will use for Entra ID authentication and authorization.

The portal uses a certificate credential associated with the app registration when requesting authentication tokens. For production environments, Kelverion recommends using a certificate signed by a trusted certificate authority. You must provide this certificate when creating the app registration. Alternatively, you can let the script configure a self-signed certificate.

Global Administrator: When running the Create-AppRegistration.ps1 script, you will be prompted to sign-in with an Azure account. This account must have the Global Administrator role assigned, in the tenant where the app registration will be created, for the app registration to be created successfully.

Create-AppRegistration Syntax

```
Create-AppRegistration.ps1 -TenantId <string> -AppName <string> [-ApiWebPort
<int>] [-ClientSecretExpiresInMonths <int>] [-CertificateExpiresInMonths <int>]
[-CertificateExportPath <string>] [-CertificatePath <string>] [-
CertificatePassword <SecureString>]
```

```
Create-AppRegistration.ps1 -TenantId <string> -AppName <string> -ApiWebFqdn
<string> -ApiWebIp <string> [-ApiWebPort <int>] [-ClientSecretExpiresInMonths
<int>] [-CertificateExpiresInMonths <int>] [-CertificateExportPath <string>] [-
CertificatePath <string>] [-CertificatePassword <SecureString>]
```

Script parameters:

- **TenantId:** The Azure tenant ID where the app registration will be created.
- **AppName:** The name of the app registration.
- **ApiWebFqdn:** (Optional) The FQDN of the machine hosting the Kelverion Automation portal API. When not specified, the local host name will be used. When installing the portal in a high availability cluster, use this parameter to specify the cluster FQDN.
- **ApiWebIP:** (Optional) The IP address of the machine hosting the Kelverion Automation portal API. When not specified, the local host IP address will be used. When installing the portal in a high availability cluster, use this parameter to specify the cluster IP Address.
- **ApiWebPort:** (Optional) The port number that will be used for the Kelverion Automation portal API. The default value is 8443.
- **CertificateExpiresInMonths:** (Optional) The number of months the certificate is valid. The default is 12.
- **ClientSecretExpiresInMonths:** (Optional) The number of months the client secret is valid. The default is 12 months.
- **CertificatePath:** The file path for an existing certificate file. Kelverion recommends providing a certificate signed by a trusted certificate authority. The certificate must use an RSA 2048-bit (or stronger) key generated with a CNG provider. CAPI-based certificates are not supported.
- **CertificateExportPath:** Must be specified when **CertificatePath** is not specified. The folder where the generated self-signed certificate is saved. Existing files at this location will be overwritten.
- **CertificatePassword:** (Optional) The certificate password for:
 - The existing .pfx or .p12 certificate when using **CertificatePath** parameter.
 - The self-signed certificate generated when using the **CertificateExportPath** parameter.
 When not specified, you will be prompted to provide a password during script runtime.

Script outputs:

- **Directory (tenant) ID:** The Azure tenant ID where the app registration was created.
- **Application (client) ID:** The client ID of the newly created app registration.
- **Certificate Thumbprint:** Thumbprint of the app registration certificate.
- **Certificate will expire on:** Expiry date of the certificate.
- **Certificate exported to:** Location where the certificate was saved.
- **Client Secret:** App registration client secret.
- **Client Secret will expire on:** Expiry date of the client secret.

Important: Make sure to save the information returned from the app registration script. You will need this information later when running the portal installer, and for future reference.

Using the App Registration PowerShell Script

To create a new App Registration:

1. On the IIS server that hosts the Kelverion portal, make sure the [Microsoft.Graph.Applications \(2.37.0\)](#) and [Microsoft.Graph.Identity.SignIns \(2.37.0\)](#) PowerShell modules are installed. See [System Requirements](#).
2. Open a PowerShell command prompt.
3. Navigate to the folder that contains the Create-AppRegistration.ps1 file.
4. Type **.\Create-AppRegistration**.
5. For the **TenantId** parameter enter the Microsoft Entra ID tenant ID where the app registration will be created.
6. For the **AppName** parameter enter the name of the app registration.
7. Optional. For the **ApiWebFqdn** enter the FQDN of the IIS server that will host the Automation Portal or the high availability cluster FQDN.
Required: Specify **ApiWebFqdn** when running the script on a machine which is not IIS server host.
Required: Specify **ApiWebFqdn** when creating an app registration for an HA cluster.
8. Optional. For the **ApiWebIp** enter the IP address of the IIS server that will host the Automation Portal or the high availability cluster IP address.
Required: Specify **ApiWebIp** when running the script on a machine which is not IIS server host.
Required: Specify **ApiWebIp** when creating an app registration for an HA cluster.
9. For the app registration certificate, you can choose to use an existing certificate or to generate a new self-signed certificate. Choose one of the following options:
 - a. For the **CertificatePath** parameter, enter the path to an existing certificate file.
Important: The certificate must use an RSA 2048-bit (or stronger) key generated with a CNG provider. CAPI-based certificates are not supported.
 - b. For the **CertificateExportPath** parameter, enter a folder path where the script generated, new self-signed certificate will be saved.
10. Run the command.

Existing certificate example:

```
.\Create-AppRegistration -TenantID 00000000-0000-0000-0000-000000000000 -AppName "Kelverion Automation Portal" -CertificatePath C:\Temp\yourcert.pfx
```

New self-signed certificate example:

```
.\Create-AppRegistration -TenantID 00000000-0000-0000-0000-000000000000 -AppName "Kelverion Automation Portal" -CertificateExportPath C:\Temp
```

11. The script creates the app registration. Inspect the app registration in the Azure portal to make sure that it was created correctly.

Important: Make sure the app registration contains the configured certificate in the app registration **Certificates and Secrets**.

12. The script returns app registration **Tenant ID**, **Client ID**, **Certificate Thumbprint** and **Client Secret**. Copy this information to a safe place so that it can be used on the next installation step and for future reference.
13. The script returns the app registration certificate and client secret expiry dates. Copy this information to a safe place so that it can be used to schedule a maintenance task.
14. The script saves certificates files at the specified **CertificateExportPath** location. Existing files at this location are overwritten. Copy certificate files to a safe place so that they can be used on the next installation step and for future reference.

Automation Portal Installation

The Automation Portal can be installed on an IIS Windows Server machine that is located either on premises or in Azure. The following main components will be installed:

- Kelverion Automation Portal Website
- Kelverion Automation Monitor Service
- Database configuration utility

On-premises: For on-premises installations, the specified IIS website identity service account should be a domain user, with limited privileges on the IIS web server. The user can be specified either as <domain>\<username> or in UPN format. For example, **example\service.account** or **service.account@example.com**.

Azure: For Azure installations, the specified IIS website identity service account should be a local machine user, with limited privileges on the Azure IIS web server. Note that this should not be the Azure user that was used to create the app registration. The user must be specified as <computername>\<username>, where <computername> is the name of the Azure IIS web server machine, which may be truncated from the Azure VM name.

Preparing for the Installation

Before installing the Kelverion Automation portal in IIS you will need the following information:

- IIS website certificate. This can be a .pfx file or a certificate installed in the personal certificate store on the local machine.
- IIS website identity service account
 - Account is used to access the Automation Portal SQL Server database.
 - Account must have external internet access.
 - Account is also used as the Log On user for the Kelverion Automation Monitor Service.
 - Can be specified in <domain>\<username> or UPN format.
- When using Azure Entra ID authentication:
 - Directory tenant ID
 - App registration client ID
 - App registration certificate
 - App registration client secret
- When using on-premises Windows authentication:

- Domain FQDN

Running the Installer

Follow these steps when installing the Kelverion Automation Portal 4.x for the first time:

1. Run the **Kelverion Automaton Portal Setup Wizard** (Kelverion.AutomationPortal.Installer-4.x.exe).
2. Check **I agree to the license terms and conditions** and then click **Install**.
3. Click **Next**. The **Destination Folder** page opens.
4. In the box, enter or select the destination folder. The default is C:\Inetpub\.
5. Click **Next**. The **IIS Website** page appears.
6. In the **Web Port** box, enter the port that Automaton Portal website will use.
7. In the **API Port** box, enter the port that the Automation Portal REST API will use. Make sure that you use the same port that you used when you created the Azure App Registration.
8. For **Web Certificate** select one of the following options.
 - a) Import from a file. In the **Web Certificate (.pfx)** box, enter the path to the certificate location. Enter the password and confirm.
 - b) Select from certificate store. Using the certificate browser, select an existing certificate from the certificate store.
9. Click **Next**. The **IIS Website Identity/Monitor Service Account** page appears.
10. In the **Username** box, enter the username for the IIS website identity service account. This account will also be configured as the Log On user for the [Kelverion Automation Monitor Service](#).
11. In the **Password** box, enter the password for the user.
12. In the **Confirm Password** box, enter the password for the user again.
13. Click **Next**. The **High Availability (Optional)** page appears. This page is only used when the portal is part of a high availability cluster. See [Installing the Web Site \(HA\)](#) for more information.
 - a) Optionally, in the **Cluster FQDN** box, enter the high availability cluster FQDN.
 - b) Optionally, in the **Cluster IP Address** box, enter the high availability cluster IP address.
14. Click **Next**. The **Authentication** page appears. Select one of the following options.
 - a) Select **Microsoft Entra ID** – for Azure Entra ID authentication with app registration.
 - b) **Windows Integrated Authentication** – for on-premises Active Directory authentication.
15. Click **Next**. The **Microsoft Entra ID** or **Windows Integrated Authentication** page will appear based on your selection.
 - a) The **Microsoft Entra ID** page appears if selected.
 - i. In the **Directory (tenant) ID** box, enter the directory ID that you used when creating the Azure app registration.
 - ii. In the **Application (client) ID** box, enter the app registration client ID that was returned by the Azure app registration.
 - iii. Click **Next**, The Microsoft Entra ID Credentials page appears. Select a credential option. For new installations Certificate credential is recommended.
 1. **Certificate** credential.

Important: This must be the same certificate configured in the Azure app registration when the app registration was created. The certificate must exist in the Azure app registration **Certificates and Secrets**. The certificate must use an RSA 2048-bit (or stronger) key

generated with a CNG provider. CAPI-based certificates are not supported. For details, see [Azure App Registration](#).

Select one of the options below to provide a certificate credential.

- a) Import from a certificate file. In the **Client Certificate (.pfx)** box, enter the path to the certificate file. Enter the password and confirm.
 - b) Select from certificate store. Using the certificate browser, select an existing certificate from the local personal certificate store.
2. **Client Secret** credential. In the **Client Secret Value** box, enter the client secret that was returned from the Azure app registration.
 - iv. Click **Next**. The **Ready to Install Portal** page appears.
 - b) The **Windows Integrated Authentication** page appears if selected.
 - i. In the **Domain** box, enter the Active Directory domain. This value is prepopulated with the current domain.
 - ii. Click **Next**. The **Ready to Install Portal** page appears.
 16. Click **Install**.
 17. Click **Finish**.
 18. Click **Close**.
 19. Once the portal website installation is complete, proceed with portal database configuration.
 - For on-premises deployment or the deployment of the first host in a high availability cluster, please refer to [Database Setup On-Premises](#).
 - For on-premises deployment of additional hosts in a high availability cluster, please refer to step 6 of [Installing the Web Site \(HA\)](#) on how to configure the database connection.
 - For Azure deployments, please refer to [Database Setup in Azure](#).
 20. Start the [Kelverion Automation Monitor Service](#) in the Service Control Manager (SCM).

Automation Portal Update

Updating the Kelverion Automation portal involves the following steps:

- Uninstall the old version of Automation Portal.
- Install the new version of the Automation Portal.
- Update the portal database.

Preparing for the Update

When running the Kelverion Automation Portal installer to update to a new version, you will have to provide the same information you provided during the initial install. You must ensure that certificates and secrets are still valid and not expired.

Before updating the Kelverion Automation portal you will need the following information:

- IIS website certificate. This can be a .pfx file or a certificate installed in the personal certificate store on the local machine. If you are re-using the same certificate, ensure that it is not expired.
- IIS website identity service account
 - This must be the same account that is used in the pre-update portal installation.

- Account is used to access the Automation Portal SQL Server database.
- Account must have external internet access.
- Account is also used as the Log On user for the Kelverion Automation Monitor Service.
- Can be specified in <domain>\<username> or UPN format.
- When using Azure Entra ID authentication:
 - Directory tenant ID
 - App registration client ID
 - App registration certificate – if using Certificate credentials in the pre-update portal.
 - App registration client secret – if using Client Secret credentials in the pre-update portal.
- When using on-premises Windows authentication:
 - Domain FQDN

Migrating From Client Secret to Certificate Credential for Entra ID Authentication

Entra ID: This only applies to installations using Azure Entra ID authentication. On-premises installations that use Active Directory and Windows authentication do not require an Azure app registration.

Starting with version 4.6.1, the Automation Portal supports Certificate credential for Entra ID authentication. If you were initially using Client Secret for Entra ID authentication, you can continue to do so. Alternatively, you can choose to start using Certificate credentials for Entra ID authentication. Certificate credentials are safer because they are harder to copy, leak, or reuse than client secrets. Client secrets work like passwords, while certificates keep the sensitive private key protected on the server.

To configure Entra ID Certificate Credential during the Automation Portal Update:

1. Before updating the portal, configure a certificate.
 - a) Obtain a new certificate. We recommend using a certificate issued by a trusted certificate authority. A self-signed certificate can also be used.
Important: The certificate must use an RSA 2048-bit (or stronger) key generated with a CNG provider. CAPI-based certificates are not supported.
 - b) Import the private key certificate (.pfx) into the local personal credential store on the portal IIS server.
 - c) Upload the public key certificate (.cer) in the existing app registration. You can obtain the public key certificate (.cer) by exporting the certificate from local credential store, after import. For details, see [Add and manage application credentials in Microsoft Entra ID](#).
2. Portal update.
 - a) When running the portal installer, on the **Microsoft Entra ID Credentials** page, select **Certificate** credential.
 - b) Use the certificate store browser to select the certificate that was imported in the local credential store.
3. Complete portal update. The portal will now use the configured certificate for Entra ID authentication.

To configure Entra ID Certificate Credential at any time:

1. Configure a certificate.

- a) Obtain a new certificate. We recommend using a certificate issued by a trusted certificate authority. A self-signed certificate can also be used.
Important: The certificate must use an RSA 2048-bit (or stronger) key generated with a CNG provider. CAPI-based certificates are not supported.
 - b) Import the private key certificate (.pfx) into the local personal credential store on the portal IIS server.
 - c) Grant Read permissions on the imported certificate, for the IIS account identity.
 - i. In your local certificate store, find the imported certificate.
 - ii. Right click on the certificate and select **All Tasks > Manage Private Keys**.
 - iii. In the Permissions dialog add the IIS account identity user and grant permissions.
 - d) Upload the public key certificate (.cer) in the existing app registration. You can obtain the public key certificate (.cer) by exporting the certificate from local credential store, after import. For details, see [Add and manage application credentials in Microsoft Entra ID](#).
2. Update the Automation Portal API configuration, on the IIS server hosting the Automation Portal.
 - a) Open the **appsettings.json** file, in the portal API installation folder. By default, this is **C:\inetpub\Kelverion\Kelverion Automation Portal API**.
 - b) Locate the **ClientSecret** string in the **Authentication/EntraId** section and change it to an empty string.
 - c) Modify the **Authentication/EntraId/ClientCertificates** as shown in the example below. Make sure to specify the certificate thumbprint in the **CertificateThumbprint** string.
 - d) Save the file.
 3. Restart/recycle the Kelverion Automation Portal API website in IIS.
 4. The portal will now use the configured certificate for Entra ID authentication.

Example Authentication/EntraId section in portal API appsettings.json:

```
Authentication": {
  "Mode": "EntraId",
  "EntraId": {
    "Instance": "https://login.microsoftonline.com/",
    "TenantId": "00000000-0000-0000-0000-000000000000",
    "ClientId": "00000000-0000-0000-0000-000000000000",
    "ClientSecret": "",
    "ClientCertificates": [{
      "SourceType": "StoreWithThumbprint",
      "CertificateStorePath": "LocalMachine/My",
      "CertificateThumbprint": "<your certificate thumbprint>"
    }]
  }
}
```

Refreshing the Portal Client Secret

Entra ID: This only applies to installations using Azure Entra ID authentication. On-premises installations that use Active Directory and Windows authentication do not require an Azure app registration.

If your deployment is using Client Secret credential, at some point that Client Secret will expire. Follow the steps below to generate a new Client Secret in your Azure app registration and configure it in your existing portal deployment.

To configure a new Client Secret during the Automation Portal update:

1. Before updating the portal, generate a new Client Secret in your app registration.
 - a) In your existing app registration, go to **Certificates & Secrets**.
 - b) Select the **Client secrets** tab and click **New client secret**.
 - c) Specify a Description and expiry date. Click Add.
 - d) A new Client Secret is created. **Make sure to record the new Client Secret value. It is only available in your Azure portal for a limited time.**
For details, see [Add and manage application credentials in Microsoft Entra ID](#).
2. Portal update.
 - a) When running the portal installer, on the **Microsoft Entra ID Credentials** page, select **Client Secret** credential.
 - b) In the **Client Secret** box enter the value of the new Client Secret that you just created.
3. Complete portal update. The portal will now use the new client secret for Entra ID authentication.

To configure a new Client Secret at any time:

1. Generate a new Client Secret in your app registration.
 - a) In your existing app registration, go to **Certificates & Secrets**.
 - b) Select the **Client secrets** tab and click **New client secret**.
 - c) Specify a Description and expiry date. Click Add.
 - d) A new Client Secret is created. **Make sure to record the new Client Secret value. It is only available in your Azure portal for a limited time.**
For details, see [Add and manage application credentials in Microsoft Entra ID](#).

2. Encrypt the new client secret with Data Protection **LocalMachine** scope. On the IIS server hosting the Automation Portal, run the following PowerShell with Administrator privileges:

```
Add-Type -AssemblyName System.Security
$plainText = '<plain text value>'
$bytes = [System.Text.Encoding]::UTF8.GetBytes($plainText)
$scope = [System.Security.Cryptography.DataProtectionScope]::LocalMachine
$encrypted = [System.Security.Cryptography.ProtectedData]::Protect($bytes, $null, $scope)
$encryptedBase64 = [System.Convert]::ToBase64String($encrypted)
$encryptedBase64
```

3. Save the encrypted Client Secret value in the Automation Portal API configuration, on the IIS server hosting the Automation Portal.
 - a) Open the **appsettings.json** file, in the portal API installation folder. By default, this is **C:\inetpub\Kelverion\Kelverion Automation Portal API**.
 - b) Locate the **ClientSecret** string in the **Authentication/EntraId** section.
 - c) Enter the encrypted Client Secret value and save the file.

4. Restart/recycle the Kelverion Automation Portal API website in IIS.
5. The portal will now use the new client secret for Entra ID authentication.

Example Authentication/EntraId section in portal API appsettings.json:

```
Authentication": {
  "Mode": "EntraId",
  "EntraId": {
    "Instance": "https://login.microsoftonline.com/",
    "TenantId": "00000000-0000-0000-0000-000000000000",
    "ClientId": "00000000-0000-0000-0000-000000000000",
    "ClientSecret": "<encrypted client secret value>"
  }
}
```

Running the Installer

Follow these steps when updating an existing installation of the Kelverion Automation Portal 4.x:

1. Backup your Kelverion Automation Portal database for the current installation.
2. Run the **Kelverion Automaton Portal** uninstall for the old installation. Make sure to backup your environment first. After running the uninstall, make sure all current installation components have been removed:
 - a) Kelverion Automation Portal and API folders (under C:\Inetpub by default)
 - b) In IIS, Kelverion Automation Portal and API websites and corresponding app pools.
 - c) C:\Program Files\Kelverion folders
 - d) Kelverion Automation Monitor Windows service
3. If you are uninstalling a portal version **4.4** or **earlier**:
 - a) Uninstall the ASP.NET Core 8.x Windows Hosting Bundle.
 - b) Install the [ASP.NET Core 10.0 Windows Hosting Bundle](#).
4. Run the **Kelverion Automaton Portal Setup Wizard** for the new version that is being installed (Kelverion.AutomationPortal.Installer-4.x.exe).
5. Check **I agree to the license terms and conditions** and then click **Install**.
6. Click **Next**. The **Destination Folder** page opens.
7. In the box, enter or select the destination folder. The default is C:\Inetpub\.
8. Click **Next**. The **IIS Website** page appears.
9. In the **Web Port** box, enter the port that Automaton Portal website will use.
10. In the **API Port** box, enter the port that the Automation Portal REST API will use. Make sure that you use the same port that you used when you created the Azure App Registration.
11. For Web Certificate select one of the following options.
 - a) Import from a file. In the **Web Certificate (.pfx)** box, enter the path to the certificate location. Enter the password and confirm.
 - b) Select from certificate store. Using the certificate browser, select an existing certificate from the certificate store.
12. Click **Next**. The **IIS Website Identity/Monitor Service Account** page appears.
13. In the **Username** box, enter the username for the IIS website identity service account.
Important: You must provide the same service account that was used during the initial portal installation.

14. In the **Password** box, enter the password for the user.
15. In the **Confirm Password** box, enter the password for the user again.
16. Click **Next**. The **High Availability (Optional)** page appears. This page is only used when the portal is part of a high availability cluster. See [Updating the Web Site \(HA\)](#) for more information.
 - a) Optionally, in the **Cluster FQDN** box, enter the high availability cluster FQDN.
 - b) Optionally, in the **Cluster IP Address** box, enter the high availability cluster IP address.
17. Click **Next**. The **Authentication** page appears.
 - a) Select **Microsoft Entra ID** – for Azure Entra ID authentication with app registration
 - b) **Windows Integrated Authentication** – for on-premises Active Directory authentication.
18. Click **Next**. The **Microsoft Entra ID** or **Windows Integrated Authentication** page will appear based on your selection.
 - a) The **Microsoft Entra ID** page appears if selected.
 - i. In the **Directory (tenant) ID** box, enter the directory ID that you used when creating the Azure app registration.
 - ii. In the **Application (client) ID** box, enter the Application ID that was returned from the Azure app registration.
 - iii. Click **Next**, The Microsoft Entra ID Credentials page appears. Select a credential option. You should use the same credential option as in the previous version of your installation.
 1. **Certificate** credential.

Important: This must be the same certificate configured in the Azure app registration. The certificate must exist in the app registration **Certificates and Secrets**. The certificate must not be expired. The certificate must use an RSA 2048-bit (or stronger) key generated with a CNG provider. CAPI-based certificates are not supported. For details, see [Azure App Registration](#) and [Migrating From Client Secret to Certificate Credential for Entra ID Authentication](#).

Select one of the options below to provide a certificate credential.

 - a) Import from a file. In the **Client Certificate (.pfx)** box, enter the path to the Azure App client certificate. Enter the password and confirm.
 - b) Select from certificate store. Using the certificate browser, select an existing certificate from the personal local certificate store.
 2. **Client Secret** credential. In the **Client Secret Value** box, enter a non-expired app registration client secret.
 - iv. Click **Next**. The **Ready to Install Portal** page appears.
 - b) The **Windows Integrated Authentication** page appears if selected.
 - i. In the **Domain** box, enter the Active Directory domain. This value is prepopulated with the current domain.
 - ii. Click **Next**. The **Ready to Install Portal** page appears.
19. Click **Install**.
20. Click **Finish**.
21. Click **Close**.
22. Once the portal website installation is complete, proceed with updating your portal database.
 - For on-premises deployment or the deployment of the first host in a high availability cluster, please refer to [Updating an Existing Database On-Premises](#).

- For on-premises deployment of additional hosts in a high availability cluster, please refer to step 2 of [Updating the Web Site \(HA\)](#) on how to configure the database connection.
 - For Azure deployments, please refer to [Updating an Existing Database in Azure](#).
23. Once the portal database has been updated, proceed to start the [Kelverion Automation Monitor Service](#) in the Service Control Manager (SCM).

Database Setup On-Premises

The Kelverion Automation Portal requires a Microsoft SQL Server database. To help with the creation and configuration of the database, use the Kelverion **automation-portal-db-cmd** utility. This utility can be found at **C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line**. The command displays help documentation when ran without any parameters.

Important: Make sure you back up your existing database before running the **automation-portal-db-cmd** command utility.

Creating a New Database On-Premises

DbCreate Command

The **automation-portal-db-cmd** provides the **DbCreate** command for creating a new on-premises Automation Portal database.

Important: The **DbCreate** command must be run on the IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges, and with sufficient permissions to configure the database on the specified SQL Server.

Important: The Column Encryption Certificate is created in the Local Machine store, and the configured user is granted read permission on the new certificate. The configured user should be the same as the user specified for the IIS website identity service account. If the IIS website is modified to use a service account different than the configured user, the new service account must be granted read permissions on the Column Encryption certificate.

Important: Exercise caution when working with the Column Encryption Certificate. This certificate is used for encrypting/decrypting data in the Automation Portal database. Deleting or modifying this certificate may result in data no longer being accessible. Make sure to always back up your database before making changes.

Important: Before using the **DbCreate** command to create a new database, you will need to add a firewall rule to allow your IIS server to access your SQL server. Default SQL Server port is 1433.

To create a new Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe DbCreate** command with parameters specific to your environment.
4. Confirm the command ran successfully and the database was created correctly.
5. **Important:** If your on-premises portal installation uses Windows authentication, you must configure the portal Administrator group. For details, please refer to [Configuring Portal Administrator Group On-Premises](#).

6. Restart the **Kelverion Automation Portal** and **Kelverion Automation Portal API** using the IIS Manager.
7. After the IIS Website and database have been correctly configured, you should be able to navigate to the Automation Portal in your browser using `https://<Portal FQDN>` or `https://<Portal IP Address>`.

The **DbCreate** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
User* (-U)	Required. Domain user to be configured for database access and column encryption certificate access, in the form <domain>\<username> . This must be the same as the IIS website identity service account that was specified during the IIS Website Installation .
Host* (-H)	Required. SQL Server host name or IP address.
Port (-P)	Optional. SQL Server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
Replace (-R)	Optional. Specifies whether an existing database with the specified name will be deleted before creating the new database. Default is True .
ColumnMasterKey (-CMK)	Optional. Name for the new Always Encrypted master key that will be created for column encryption. Default value is KelverionCMK .
ColumnEncryptKey (-CEK)	Optional. Name for the new Always Encrypted column encryption key. Default value is KelverionCEK .
CertName (-Cert)	Optional. Name for the Column Encryption Certificate used when creating the new ColumnMasterKey. A new certificate is created, with the specified friendly name, when no certificate is found in the Local Machine certificate store, and if CertCreate = true. Existing certificates with the same name are not deleted or overwritten. Default value is Kelverion CMK Certificate .
CertCreate (-CCr)	Optional. Specifies if a new certificate is created when not found. Default is True .
AppSettingsApiPath (-AA)	Optional. Specifies the path to the Kelverion Automation Portal API appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal API/AppSettings.json .
AppSettingsPortalPath (-AP)	Optional. Specifies the path to the Kelverion Automation Portal appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal/AppSettings.json .
AppSettingsMonitorPath (-AM)	Optional. Specifies the path to the Kelverion Automation Monitor appsettings.json file. Default value is C:/Program Files/Kelverion/Automation Portal/Monitor/AppSettings.json .

DbCreate Examples

This first example creates a new Automation Portal database in SQL Server. The command creates a new self-signed certificate and registers it in the LocalMachine certificate store. This certificate will be used to create SQL Always Encrypted master and column keys. Default names are used for the created certificate and for the master and column keys.

```
automation-portal-db-cmd.exe DBCreate -DBName KelverionPortal -Host
sqlServer.example.com -User example\service.account
```

To force the SQL server host certificate to be trusted, include the **TrustServerCert** option.

```
automation-portal-db-cmd.exe DBCreate -DBName KelverionPortal -Host
sqlServer.example.com -User example\service.account -TrustServerCert
```

You can also specify the names to use for the Always Encrypted master and column keys, using the **CMK** and **CEK** parameters, respectively.

```
automation-portal-db-cmd.exe DBCreate -DBName KelverionPortal -Host
sqlServer.example.com -User example\service.account -CertName PortalServiceCert
-CMK PortalCMK -CEK PortalCEK
```

Updating an Existing Database On-Premises

The **automation-portal-db-cmd** provides the **DbUpdate** command for updating an existing on-premises Automation Portal database. In addition, the **DbConnectionStr** command is used to update connection string in the portal website, after running DbUpdate.

Important: Make sure you back up your existing database before running the DbUpdate and/or DbConnectionStr commands.

Important: The **DbUpdate** and **DbConnectionStr** commands must be run on the IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges, and with sufficient permissions to configure the database on the specified SQL Server.

To update an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe DbUpdate** command with parameters specific to your environment.
4. Run the **automation-portal-db-cmd.exe DbConnectionStr** command with parameters specific to your environment.
5. Confirm the commands ran successfully and the database was updated.
6. After the IIS Website and database have been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

DbUpdate Command

The **DbUpdate** command is used for updating an existing on-premises Automation Portal database. The **DbUpdate** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. SQL Server host name or IP address.
Port (-P)	Optional. SQL Server port. Default port is 1433 .

Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ColumnEncryptKey (-CEK)	Optional. Name for the Always Encrypted column encryption key. Default value is KelverionCEK .

DbUpdate Example

The following updates an existing Automation Portal database in SQL Server. The command specifies the DbName and Host parameters, and uses defaults for Port, Encrypt and TrustServerCert parameters, respectively.

```
automation-portal-db-cmd.exe DbUpdate -DbName KelverionPortal -Host
sqlServer.example.com
```

DbConnectionString Command

The **DbConnectionString** command is used to update connection string configuration in the portal website. If you are performing an update from a previous Automation Portal version, you will have to run the DbConnectionString command after running DbUpdate. The **DbConnectionString** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. SQL Server host name or IP address.
Port (-P)	Optional. SQL Server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
AppSettingsApiPath (-A)	Optional. Specifies the path to the Kelverion Automation Portal API appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal API/AppSettings.json .
AppSettingsPortalPath (-Ap)	Optional. Specifies the path to the Kelverion Automation Portal appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal/AppSettings.json .
AppSettingsMonitorPath (-App)	Optional. Specifies the path to the Kelverion Automation Monitor appsettings.json file. Default value is C:/Program Files/Kelverion/Automation Portal/Monitor/AppSettings.json .

DbConnectionString Example

The following configures the Automation Portal connection string. The command specifies the DbName and Host parameters, and uses defaults for AppSettingsApiPath, AppSettingsPortalPath, as well as Port, Encrypt and TrustServerCert parameters, respectively.

```
automation-portal-db-cmd.exe DbConnectionString -DbName KelverionPortal -Host
sqlServer.example.com
```

Configuring a Database User On-Premises

DbUser Command

The **automation-portal-db-cmd** provides the **DbUser** command for configuring a new user on an existing on-premises Automation Portal database. The command is also used for configuring the existing column encryption certificate for the new user. The previously configured database user is not removed from the database.

Important: Make sure you back up your existing database before running the DbUser command.

Important: When configuring a new database user, make sure the same user is also configured as IIS website identity service account.

Important: The **DbUser** command must be run on the IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges, and with sufficient permissions to configure the database on the specified SQL Server.

Important: Exercise caution when working with the Column Encryption Certificate. This certificate is used for encrypting/decrypting data in the Automation Portal database. Deleting or modifying this certificate may result in data no longer being accessible. Make sure to always back up your database before making changes.

The **DbUser** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. SQL Server host name or IP address.
Port (-P)	Optional. SQL server port. Default port is 1433 .
User* (-U)	Required. New domain user to be configured for database access and column encryption certificate access, in the form <domain>\<username> . This user must also be configured as the IIS website app pool identity service account.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
CertName (-Cert)	Optional. Name for the existing Column Encryption certificate in the Local Machine certificate store. Default value is Kelverion CMK Certificate .

To update the user for an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe DbUser** command with parameters specific to your environment.
4. Confirm the commands ran successfully and the database user was configured correctly.
5. After the database user has been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

DbUser Example

The following adds a new user for an existing Automation Portal database. The command specifies the DbName, Host and User parameters, and uses the default Column Encryption certificate.

```
automation-portal-db-cmd.exe DbUser -DbName KelverionPortal -Host
sqlServer.example.com -User example\another.service.account
```

Configuring Portal Administrator Group On-Premises

Important: This only applies to portal installations that are using on-premises Windows authentication.

SetRoleGroups Command

The **automation-portal-db-cmd** provides the **SetRoleGroups** command for configuring Active Directory role group(s), in an existing Automation Portal database. This command can be used to specify the portal Administrator role group(s). Members of these group(s) will have administrative privileges in the portal.

Important: After configuring new Administrator group(s), members of previously configured group(s) will no longer have administrative privileges, unless they are also members of the newly configured group(s).

Important: Make sure you back up your existing database before running the **SetRoleGroups** command.

Important: The **SetRoleGroups** command must be run on the IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

To set the Admin Group for an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe SetRoleGroups** command with parameters specific to your environment.
4. Use the **GetRoleGroups** command to confirm that the Admin Group was configured correctly.

After the Admin Group has been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>, and log in with a member of the specified group.

The **SetRoleGroups** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .

AdminGroup (-AG)	Required. Active Directory group(s) containing portal administrator users. Multiple comma (,) separated groups can be specified. This only applies to installations using on-premises Windows authentication.
AppSettingsApiPath (-AA)	Optional. Specifies the path to the Kelverion Automation Portal API appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal API/AppSettings.json .

SetRoleGroups Example

The following configures a new Admin group for Automation Portal database. The command specifies the DbName, Host and AdminGroup parameters, and uses the default AppSettingsApiPath.

```
automation-portal-db-cmd.exe SetRoleGroups -DbName KelverionPortal -Host
sqlServer.example.com -AdminGroup Portal_Amins
```

GetRoleGroups Command

The **GetRoleGroups** command retrieves the currently configured Active Directory role groups.

The **GetRoleGroups** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .

GetRoleGroups Example

The following command retrieves the Active Directory role groups, as configured in the specified database.

```
automation-portal-db-cmd.exe GetRoleGroups -DbName KelverionPortal -Host
sqlServer.example.com
```

Configuring Proxy Settings On-Premises

The **automation-portal-db-cmd** provides the **SetProxy** command for configuring proxy settings when using an on-premises database.

Important: Make sure you back up your existing database before running the **SetProxy** command.

Important: The **SetProxy** command must be run on the IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

To configure proxy settings for an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe SetProxy** command with parameters specific to your environment.
4. Optional. Run the **SetProxyPassword** command to set the proxy password securely.

- Run the **GetProxy** command to confirm that the proxy settings were configured correctly.

After the proxy settings have been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

SetProxy Command

The **SetProxy** command configures proxy settings for an existing on-premises portal database. Only specified proxy parameters are configured. The **SetProxy** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ProxyUrl (-Url)	Optional. Specifies the proxy server URL. When not configured, the portal will not use proxy settings.
ProxyUsername (-UN)	Optional. Specifies the proxy username.
ProxyPassword (-PW)	Optional. Specifies the proxy password.
UseDefaultProxyCredentials (-DCred)	Optional. Specifies whether the default system credentials are used for proxy authentication. When set to True, ProxyUsername and ProxyPassword are ignored.
ProxyBypassList (-BP)	Optional. Comma-separated list of addresses that bypass the proxy.
Reset (-R)	Optional. Reset all proxy settings.

SetProxy Examples

The following command sets the proxy URL, username and password.

```
automation-portal-db-cmd.exe SetProxy -DbName KelverionPortal -Host
sqlServer.example.com -ProxyUrl https://portalProxy.example.com -ProxyUsername
username -ProxyPassword password
```

The following command configures proxy settings with default system credentials.

```
automation-portal-db-cmd.exe SetProxy -DbName KelverionPortal -Host
sqlServer.example.com -ProxyUrl https://portalProxy.example.com -Dcred true
```

The following command resets all proxy settings.

```
automation-portal-db-cmd.exe SetProxy -DbName KelverionPortal -Host
sqlServer.example.com -Reset true
```

SetProxyPassword Command

The **SetProxyPassword** command configures the proxy password for an existing on-premises portal database. You can use this command to specify the proxy password securely. The **SetProxyPassword** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ProxyPassword (-PW)	Optional. Specifies the proxy password. When not specified, the command will prompt for the password. Password is not displayed while typing at command prompt.
PasswordConfirm (-PWc)	Optional. Confirm the proxy password. When not specified, the command will prompt for the password. Password is not displayed while typing at command prompt.
Reset (-R)	Optional. Resets the proxy password.

SetProxyPassword Examples

The following command sets the proxy password. The command will prompt for the password and password confirmation. The passwords will not be displayed while typed at command prompt.

```
automation-portal-db-cmd.exe SetProxyPassword -DbName KolverionPortal -Host
sqlServer.example.com
```

The following command resets the proxy password.

```
automation-portal-db-cmd.exe SetProxyPassword -DbName KolverionPortal -Host
sqlServer.example.com -Reset true
```

GetProxy Command

The **GetProxy** command retrieves proxy settings for an existing on-premises portal database. The **GetProxy** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .

GetProxy Examples

The following command retrieves the configured proxy settings.

```
automation-portal-db-cmd.exe GetProxy -DbName KolverionPortal -Host
sqlServer.example.com
```

Database Setup in Azure

The Kelverion Automation Portal requires a Microsoft SQL Server database. To help with the creation and configuration of the database, use the Kelverion **automation-portal-db-cmd** command line utility. This utility can be found at **C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line**. The command displays help documentation when ran without any parameters.

Important: Make sure you back up your existing database before running the **automation-portal-db-cmd** command utility.

Creating a New Database in Azure

The **automation-portal-db-cmd** provides the **DbCreateAz** command for creating a new Automation Portal database in Azure.

Important: When using the **DbCreateAz** command, you will be prompted to sign into the Azure Portal. The user signing in must be configured as an administrator on the target Azure SQL Server, so that the command has the necessary permissions to create and configure the new database. For details, please see [Configuring Azure SQL Server Admin](#).

Important: Before using the **DbCreateAz** command to create a new database in Azure, you will need to add a firewall rule to allow your Azure IIS server to access your Azure SQL server. For details, see [Adding an Azure SQL Server Firewall Rule](#).

Important: The Azure portal IIS server VM uses Azure Managed Identities to connect to the Azure SQL server. Before creating a new Azure database, you must configure either a system assigned, or user assigned managed identity for the portal IIS VM. For details, please refer to [Configuring Automation Portal VM Managed Identity](#).

DbCreateAz Command

The **DbCreateAz** command is used for creating a new Automation Portal database in Azure.

Important: The **DbCreateAz** command must be run on the Azure IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

Important: The Column Encryption Certificate is created in the Local Machine store, and the configured user is granted read permission on the new certificate. The configured user should be the same as the user specified for the IIS website identity service account. If the IIS website is modified to use a service account different than the configured user, the new service account must be granted read permissions on the Column Encryption certificate.

Important: Exercise caution when working with the Column Encryption Certificate. This certificate is used for encrypting/decrypting data in the Automation Portal database. Deleting or modifying this certificate may result in data no longer being accessible. Make sure to always back up your database before making changes.

To create a new Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to **C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line**.

3. Run the **automation-portal-db-cmd.exe DbCreateAz** command with parameters specific to your environment.
4. Confirm the command ran successfully and the database was created correctly.
5. After the IIS Website and database have been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Azure VM name >> or <https://<Azure internal IP Address>>.

The **DbCreateAz** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
User* (-U)	Required. Local machine user to be configured for column encryption certificate access. This must be the same as the IIS website identity service account that was specified during the IIS Website Installation . The user must be specified in the form <computername>\<username> , where <computername> is the computer name of the Azure IIS web server machine, which may be truncated from the Azure VM name. For example, for the VM name Kelverion-Portal-VM , the user should be Kelverion-Portal\service.account .
ManagedIdentityName* (-Min)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database. For details, see Configuring Azure Portal VM Managed Identity .
ManagedIdentityClientId (-Mic)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity. For details, see Configuring Azure Portal VM Managed Identity .
Host* (-H)	Required. Azure SQL server name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
Replace (-R)	Optional. Specifies whether an existing database with the specified name will be deleted before creating the new database. Default is True .
ColumnMasterKey (-CMK)	Optional. Name for the new Always Encrypted master key that will be created for column encryption. Default value is KelverionCMK .
ColumnEncryptKey (-CEK)	Optional. Name for the new Always Encrypted column encryption key. Default value is KelverionCEK .
CertName (-Cert)	Optional. Name for the Column Encryption Certificate used when creating the new ColumnMasterKey. A new certificate is created, with the specified friendly name, when no certificate is found in the Local Machine certificate store, and if CertCreate = true. Existing certificates with the same name are not deleted or overwritten. Default value is Kelverion CMK Certificate .
CertCreate (-CCr)	Optional. Specifies if a new certificate is created when not found. Default is True .
AppSettingsApiPath (-AA)	Optional. Specifies the path to the Kelverion Automation Portal API appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal API/AppSettings.json .

AppSettingsPortalPath (-AP)	Optional. Specifies the path to the Kelverion Automation Portal appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal/AppSettings.json .
AppSettingsMonitorPath (-AM)	Optional. Specifies the path to the Kelverion Automation Monitor appsettings.json file. Default value is C:/Program Files/Kelverion/Automation Portal/Monitor/AppSettings.json .

DbCreateAz Examples

This first example creates a new Automation Portal database in Azure SQL Server. The system assigned managed identity associated with the portal IIS VM will be used to connect to the database server. The command creates a new self-signed certificate and registers it in the LocalMachine certificate store. This certificate will be used to create SQL Always Encrypted master and column keys.

```
automation-portal-db-cmd.exe DbCreateAz -DBName KelverionPortal -Host
sqlServer.database.windows.net -User Portal-VM\localUser -ManagedIdentityName
Portal-VM
```

You can also use a user assigned managed identity for connecting to the database server. In this case, the managed identity client ID GUID must also be provided.

```
automation-portal-db-cmd.exe DbCreateAz -DBName KelverionPortal -Host
sqlServer.database.windows.net -User Portal-VM\localUser -ManagedIdentityName
My-user-MI -ManagedIdentityClientId A96EC3DB-6658-4444-8AFD-E4E6C5B3B5E2
```

You can also specify the names to use for the Always Encrypted master and column keys, using the **CMK** and **CEK** parameters, respectively.

```
automation-portal-db-cmd.exe DbCreateAz -DBName KelverionPortal -Host
sqlServer.database.windows.net -User Portal-VM\localUser -ManagedIdentityName
Portal-VM -CertName PortalServiceCert -CMK PortalCMK -CEK PortalCEK
```

Configuring Azure SQL Server Admin

When using the **DbCreateAz** command to create a new Azure database, you will be prompted to sign into the Azure Portal. The user signing in must be configured as an administrator on the target Azure SQL Server, so that the command has the necessary permissions to create and configure the new database.

To configure the Azure SQL Server Admin:

1. Connect to Azure portal and navigate to the Azure SQL Server where the database will be created.
2. Navigate to **Settings > Microsoft Entra ID**.
3. Click **Set Admin**.
4. Select the user to be set as admin. Alternatively, specify a group containing the user to be set as admin.
5. Click **Save**.
6. Make sure to sign into Azure with this user when using the **DbCreateAz** command.

Adding an Azure SQL Server Firewall Rule

Before using the **DbCreateAz** command to create a new database in Azure, you will need to add a firewall rule to allow your Azure IIS server to access your Azure SQL server.

To add an Azure SQL Server firewall rule:

1. Connect to Azure portal and navigate to the Azure SQL Server where the database will be created.
2. Navigate to **Security > Networking**.
3. In the **Firewall rules** section click **Add a firewall rule**.
4. In the **Add a firewall rule** popup:
 - a. Specify the rule name.
 - b. In the **Start IP** and **End IP** boxes, enter the **public IP address** of the Azure IIS server VM where Automation Portal Website has been installed. You can find the VM public IP address in the **Overview** section of the VM, in the Azure portal.

Configuring Automation Portal VM Managed Identity

The Automation Portal VM uses Azure Managed Identities to connect to the database server. Before creating a new Azure database, you must configure either a system assigned, or user assigned managed identity for the Automation Portal IIS VM.

To configure a system assigned managed identity:

1. Connect to Azure portal and navigate to your Azure VM where the Automation Portal IIS is installed.
2. Navigate to **Security > Identity** and select the **System assigned** tab.
3. Make sure the **Status** setting is turned on. The identity name will be the same as the VM name.
4. To specify the system assigned managed identity when creating the Azure database, specify the identity name (VM name) in the **ManagedIdentityName** parameter.

To configure a user assigned managed identity:

1. Connect to Azure portal and navigate to **Managed Identities**.
2. Create a new managed identity in the same region and resource group where your Azure Portal VM is located.
3. Record the newly managed identity Client ID.
4. Navigate to your Azure Portal VM where the Automation Portal IIS is installed.
5. Navigate to **Security > Identity** and select the **User assigned** tab.
6. Click **+Add** to add the new managed identity to the Portal VM. You may have to re-log into the Azure portal, before the new managed identity is available for assignment.
7. To specify the user assigned managed identity when creating the Azure database, specify the identity name in the **ManagedIdentityName** parameter and the identity Client ID GUID in the **ManagedIdentityClientId** parameter.

Updating an Existing Database in Azure

The **automation-portal-db-cmd** provides the **DbUpdateAz** command for updating an existing Azure Automation Portal database. In addition, the **DbConnectionStrAz** command is used to update connection string in the portal website, after running DbUpdateAz.

Important: Make sure you back up your existing database before running the DbUpdateAz and/or DbConnectionStrAz commands.

Important: The **DbUpdateAz** and **DbConnectionStrAz** commands must be run on the Azure IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

Important: When using the **DbUpdateAz** command you will be prompted to sign into the Azure Portal. Make sure to sign in with an Azure user that has been configured as administrator on the target Azure SQL Server, so that the command has the necessary permissions to create and configure the new database. For details see [Configuring Azure SQL Server Admin](#).

To update an existing Azure Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe DbUpdateAz** command with parameters specific to your environment.
4. Run the **automation-portal-db-cmd.exe DbConnectionStrAz** command with parameters specific to your environment.
5. Confirm the commands ran successfully and the database was updated.
6. After the IIS Website and database have been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

DbUpdateAz Command

The **DbUpdateAz** command is used for updating an existing Azure Automation Portal database. The **DbUpdateAz** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. Azure SQL Server name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
ManagedIdentityName* (-MIIn)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.
ManagedIdentityClientId (-MIc)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ColumnEncryptKey (-CEK)	Optional. Name for the Always Encrypted column encryption key. Default value is KelverionCEK .

DbUpdateAz Example

The following updates an existing Automation Portal database in Azure SQL Server. The command specifies the DbName and Host parameters, and the ManagedIdentityName parameter with the system assigned managed identity (VM name).

```
automation-portal-db-cmd.exe DbUpdateAz -DbName KelverionPortal -Host
sqlServer.database.windows.net -ManagedIdentityName Portal-VM
```

DbConnectionStrAz Command

The **DbConnectionStrAz** command is used to update connection string configuration in the portal website. If you are performing an update from a previous Automation Portal version, you will have to run the DbConnectionStrAz command after running DbUpdateAz. The **DbConnectionStrAz** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
ManagedIdentityName* (-MIn)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.
ManagedIdentityClientId (-MId)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
AppSettingsApiPath (-A)	Optional. Specifies the path to the Kelverion Automation Portal API appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal API/AppSettings.json .
AppSettingsPortalPath (-Ap)	Optional. Specifies the path to the Kelverion Automation Portal appsettings.json file. Default value is C:/inetpub/Kelverion/Kelverion Automation Portal/AppSettings.json .
AppSettingsMonitorPath (-App)	Optional. Specifies the path to the Kelverion Automation Monitor appsettings.json file. Default value is C:/Program Files/Kelverion/Automation Portal/Monitor/AppSettings.json .

DbConnectionStrAz Example

The following configures the Automation Portal connection string. The command specifies the DbName and Host parameters, and the ManagedIdentityName and ManagedIdentityClientId parameters for the user assigned managed identity.

```
automation-portal-db-cmd.exe DbConnectionStrAz -DbName KelverionPortal -Host
sqlServer.database.windows.net -ManagedIdentityName My-user-managed-identity
-ManagedIdentityClientId A96EC3DB-6658-4444-8AFD-E4E6C5B3B5E2
```

Configuring a Managed Identity and User in Azure

DbUserAz Command

The **automation-portal-db-cmd** provides the **DbUserAz** command for configuring a managed identity on an existing Azure Automation Portal database. The command is also used for configuring the existing column encryption certificate for a new user. The previously configured database managed identity is not removed from the database.

Important: Make sure you back up your existing database before running the **DbUserAz** command.

Important: The **DbUserAz** command must be run on the Azure IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

Important: When using the **DbUserAz** command you will be prompted to sign into the Azure Portal. Make sure to sign in with an Azure user that has been configured as administrator on the target Azure SQL Server, so that the command has the necessary permissions to create and configure the new database. For details see [Configuring Azure SQL Server Admin](#).

Important: Exercise caution when working with the Column Encryption Certificate. This certificate is used for encrypting/decrypting data in the Automation Portal database. Deleting or modifying this certificate may result in data no longer being accessible. Make sure to always back up your database before making changes.

The **DbUserAz** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database to be configured.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
User* (-U)	Required. Local machine user to be configured for column encryption certificate access. This must be the same as the IIS website identity service account that was specified during the IIS Website Installation . The user must be specified in the form <computername>\<username> , where <computername> is the computer name of the Azure IIS web server machine, which may be truncated from the Azure VM name. For example, for the VM name Kelverion-Portal-VM , the user should be Kelverion-Porta\service.account .
ManagedIdentityName* (-MIName)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.
ManagedIdentityClientId (-MIC)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
CertName (-Cert)	Optional. Name for the existing Column Encryption certificate, in the Local Machine certificate store. Default value is Kelverion CMK Certificate .

To update the user for an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe DbUserAz** command with parameters specific to your environment.
4. Confirm the commands ran successfully and the database user was configured correctly.
5. After the database user has been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

DbUserAz Example

The following adds a new user/managed identity for an existing Automation Portal database. The command specifies the DbName, Host and User parameters, and uses the default Column Encryption certificate.

```
automation-portal-db-cmd.exe DbUserAz -DbName KelverionPortal -Host
sqlServer.example.com -User example\another.service.account -
ManagedIdentityName Portal-VM
```

Configuring Proxy Settings in Azure

The **automation-portal-db-cmd** provides the **SetProxyAz** command for configuring proxy settings when using a database in Azure.

Important: Make sure you back up your existing database before running the **SetProxyAz** command.

Important: The **SetProxyAz** command must be run on the Azure IIS server where the Automation Portal website has been installed and must be run by a user with local administrator privileges.

To configure proxy settings for an existing Automation Portal database:

1. Open an Admin command prompt.
2. Change the directory to *C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line*.
3. Run the **automation-portal-db-cmd.exe SetProxyAz** command with parameters specific to your environment.
4. Optional. Run the **SetProxyPasswordAz** command to set the proxy password securely.
5. Run the **GetProxyAz** command to confirm that the proxy settings were configured correctly.

After the proxy settings have been correctly configured, you should be able to navigate to the Automation Portal in your browser using <https://<Portal FQDN>> or <https://<Portal IP Address>>.

SetProxyAz Command

The **SetProxyAz** command configures proxy settings for an existing Azure portal database. Only specified proxy parameters are configured. The **SetProxyAz** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
ManagedIdentityName* (-MIn)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.

ManagedIdentityClientId (-Mlc)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ProxyUrl (-Url)	Optional. Specifies the proxy server URL. When not configured, the portal will not use proxy settings.
ProxyUsername (-UN)	Optional. Specifies the proxy username.
ProxyPassword (-PW)	Optional. Specifies the proxy password.
UseDefaultProxyCredentials (-DCred)	Optional. Specifies whether the default system credentials are used for proxy authentication. When set to True, ProxyUsername and ProxyPassword are ignored.
ProxyBypassList (-BP)	Optional. Comma-separated list of addresses that bypass the proxy.
Reset (-R)	Optional. Reset all proxy settings.

SetProxyAz Examples

The following command sets the proxy URL, username and password.

```
automation-portal-db-cmd.exe SetProxyAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM -ProxyUrl
https://portalProxy.example.com -ProxyUsername username -ProxyPassword password
```

The following command configures proxy settings with default system credentials.

```
automation-portal-db-cmd.exe SetProxyAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM -ProxyUrl
https://portalProxy.example.com -Dcred true
```

The following command resets all proxy settings.

```
automation-portal-db-cmd.exe SetProxyAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM -Reset true
```

SetProxyPasswordAz Command

The **SetProxyPasswordAz** command configures the proxy password for an existing Azure portal database. You can use this command to specify the proxy password securely. The **SetProxyPassword** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
ManagedIdentityName* (-Min)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.
ManagedIdentityClientId (-Mlc)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .

TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .
ProxyPassword (-PW)	Optional. Specifies the proxy password. When not specified, the command will prompt for the password. Password is not displayed while typing at command prompt.
PasswordConfirm (-PWc)	Optional. Confirm the proxy password. When not specified, the command will prompt for the password. Password is not displayed while typing at command prompt.
Reset (-R)	Optional. Resets the proxy password.

SetProxyPasswordAz Examples

The following command sets the proxy password. The command will prompt for the password and password confirmation. The passwords will not be displayed while typed at command prompt.

```
automation-portal-db-cmd.exe SetProxyPasswordAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM
```

The following command resets the proxy password.

```
automation-portal-db-cmd.exe SetProxyPasswordAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM -Reset true
```

GetProxy Command

The **GetProxy** command retrieves proxy settings for an existing on-premises portal database. The **GetProxy** command has the following parameters:

Parameter	Description
DbName* (-D)	Required. Name of the database.
Host* (-H)	Required. Azure SQL Server host name or IP address.
Port (-P)	Optional. Azure SQL server port. Default port is 1433 .
ManagedIdentityName* (-MIn)	Required. Specifies the name of the system or user assigned managed identity that is used to connect to the Azure SQL database.
ManagedIdentityClientId (-MId)	Optional. Managed identity client ID, when ManagedIdentityName specifies a user assigned managed identity.
Encrypt (-E)	Optional. Specifies if communication with the database is encrypted. Default is True .
TrustServerCert (-T)	Optional. Specifies whether the SQL server host certificate is trusted. Default is False .

GetProxy Examples

The following command retrieves the configured proxy settings.

```
automation-portal-db-cmd.exe GetProxyPasswordAz -DbName KelverionPortal -Host
sqlServer.example.com -ManagedIdentityName Portal-VM
```

Automation Portal Roles

In the Automation Portal, every portal user must be assigned one of the following roles:

- The **Administrator** role has full permissions for all Automaton Portal pages and tools.

- The **Developer** role has permissions for creating and editing services, offerings, and wiki items.
- The **Operator** role has permissions for submitting and managing requests and viewing wiki content.
- The **Reporter** role has permissions for viewing reports for the entire organization. Users that have been configured with the Developer or Operator role can additionally also receive the Reporter role, to access Automation Portal Reports.

Roles in Azure Entra ID

For portal installations that use Azure Entra ID, portal roles are configured by using the Microsoft Azure portal.

Important: Portal users assigned the **Administrator** or **Developer** roles will need to be assigned the Azure user role **Directory Readers**, in order to access Entra ID group information.

To assign an Entra ID user role:

1. Login to Microsoft Azure portal.
2. On the navigation bar, select **Microsoft Entra ID**.
3. Select **Enterprise applications**.
4. Open the Kelverion Automation Portal enterprise application.
5. Select **Users and groups**.
6. Click **Add user/group**.
7. On the **Add Assignment** page under **Users**. Select **None Selected**.
8. Select one or more users.
9. Click **Select**.
10. On the **Add Assignment** page under **Select a role**. Select **None Selected**.
11. Select a role (e.g., administrator, developer, or operator).
12. Click **Select**.
13. Review the selected users and roles.
14. Click **Assign**.

Roles in Active Directory

For portal installations that use on-premises Windows authentication, portal roles are configured by assigning users to one of the following Active Directory groups:

- **Administrator** role group – members of this group have the portal **Administrator** role.
- **Developer** role group – members of this group have the portal **Developer** role.
- **Operator** role group – members of this group have the portal **Operator** role.
- **Reporting** role group – members of this group have the portal **Reporting** role.

These groups are Active Directory groups that can have arbitrary names but must be configured in the portal as the Administrator, Developer, Operator and Reporting role group, respectively.

To configure the Administrator role group:

The **Administrator** role group is configured by using the **automation-portal-db-cmd** utility. Use the **SetGroupRoles** command to specify one or more Active Directory groups for the portal Administrator role. For details, refer to [Configuring Portal Administrator Group On-Premises](#).

To configure the Developer, Operator and Reporting role groups:

Once the Administrator role group has been configured, log into the Automation Portal with a portal administrator user, and navigate to the **Portal Roles** section on the **Settings Page** to configure the other non-admin roles. For details, refer to [Settings Page > Portal Roles](#).

Firewall Configuration

You must ensure that your firewall has inbound rules for the ports that are used by the Automation Portal website and API, which by default are ports 443 and 8443, respectively. If you used other ports during your Automation Portal installation, ensure that the appropriate inbound rules have been configured.

Failure to add a firewall rule for the API port will result in the portal failing with the error message: "Something went wrong. Cannot read properties of undefined (reading 'data'). Please refer to your system administrator."

Proxy Configuration

If your network environment uses a proxy to connect to the internet, you will have to configure proxy settings, so that the Automation Portal can make external calls. The portal makes external HTTPS calls during the following cases:

- During authentication, when using Azure Entra ID.
- When accessing Azure Entra ID user and group information.
- When making webhook calls to external endpoints, from the Monitor Service.

You can use the **SetProxy** and **SetProxyAz** commands in the **automation-portal-db-cmd** utility to configure proxy settings, for on-premises and Azure environments, respectively. For details, refer to:

- [Configuring Proxy Settings On-Premises](#)
- [Configuring Proxy Settings in Azure](#)

On-Premises High Availability (HA) Deployment

The baseline configuration for a portal High availability (HA) deployment is using the Windows Server Network Load Balancing (NLB) feature, and the instructions below describe its configuration. The portal has also been validated in HA mode deployed to Azure using a Virtual Machines (IaaS), and an Azure Load Balancer. The configuration process for the portal is the same as documented below when allowing for the infrastructure differences between your environment and our documented steps using Windows NLB.

Note: Additional licensing fees apply. Please speak to your account manager regarding support for other HA configurations.

Additional System Requirements:

- Network Load Balancing feature installed.
- The web certificate settings must be configured for the cluster FQDN and cluster IP address.
- Each Server in the cluster must have all System Requirements installed with matching versions of ASP.NET.

Automation Portal Installation (HA)

To deploy the Automation Portal High Availability:

1. Configure a cluster of IIS servers by installing the NLB Windows Server Feature. <https://docs.microsoft.com/en-us/windows-server/networking/technologies/network-load-balancing>
2. Complete [Azure App Registration](#) specifying the high availability cluster FQDN and cluster IP address using the [Create-AppRegistration](#) and specifying the parameters -ApiWebFQDN and -ApiWebIP.
3. On the first host in the cluster, complete the [Automation Portal Installation](#). When running the portal installer, take note of the following:
 - On the IIS Website page, use the same certificate for each host.
 - On the High Availability page, specify the Cluster FQDN and the Cluster IP Address. Use the same values for each host in the cluster.
4. On the host created during step 3, export the following database certificate located in the computer certificate store. See [Export a certificate with its private key](#).
 - Kelverion CMK Certificate.
5. On each additional host in the cluster, complete the [Automation Portal Installation](#). Take note of the following:
 - On the IIS Website page, use the same certificate for each host.
 - On the High Availability page, specify the Cluster FQDN and the Cluster IP Address. Use the same values for each host in the cluster.
6. On each additional host in the cluster, configure the database connection.
 - a. Verify your SQL Server has an inbound port rule for port 1433.
 - b. Open an Admin command prompt.
 - c. Change the directory to C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line.
 - d. Run the **automation-portal-db-cmd.exe** [DBConnectionStr](#) command with parameters specific to your environment.
 - e. Confirm the command ran successfully.
7. On each additional host in the cluster, import the **Kelverion CMK Certificate**.
 - a. Copy the **Kelverion CMK Certificate** exported from the first host during step 4 to each additional host.
 - b. Open the **Manage computer certificates** and navigate to Personal > Certificates.
 - c. Import the database certificate.
 - d. Right-click the certificate Kelverion CMK Certificate, All Tasks -> **Manage Private Key**.
 - e. Add the same service user as IIS identity pool.
8. Restart the IIS servers.
9. From a supported web browser, navigate to the Portal using the cluster FQDN.
10. On each host, obtain the Server ID for licensing and request a license from Kelverion. See [High Availability Licensing](#).
11. Upload a license to the Portal, see [Uploading License](#).
12. Start the [Kelverion Automation Monitor Service](#) in the Service Control Manager (SCM) on each additional host in the cluster.

Automation Portal Update (HA)

To update the Automation Portal High Availability:

1. On the first host and each additional host in the cluster, complete the [Automation Portal Update](#). Take note of the following:
 - On the IIS Website page, use the same certificate for each host.
 - On the High Availability page, specify the Cluster FQDN and the Cluster IP Address. Use the same values for each host in the cluster.
2. On each additional host in the cluster, configure the database connection.
 - f. Verify your SQL Server has an inbound port rule for port 1433.
 - g. Open an Admin command prompt.
 - h. Change the directory to `C:\Program Files\Kelverion\Automation Portal\Tools\Database Command Line`.
 - i. Run the [automation-portal-db-cmd.exe DBConnectionStr](#) command with parameters specific to your environment.
 - j. Confirm the command ran successfully.
3. Restart the IIS servers.

Configuring Azure IIS Website for External Access

After installing the Automation Portal in Azure, additional configuration steps are required to make the portal externally accessible.

As an example, the following steps outline the process for configuring the Automation Portal for external access by using the Azure VM public IP address. This may vary depending on your Azure environment, but the general steps would be similar. You can find the Automation Portal VM public IP address in the **Overview** tab of the VM.

1. [Configure Inbound Port Rule](#)
2. [Update App Registration Redirect URLs](#)
3. [Configure IIS CORS RULE](#)
4. [Configure Content Security Policy](#)
5. [Configure Automation Portal API URL](#)
6. Update IIS Server Certificate
7. After the Azure IIS Website and database have been correctly configured, you should be able to navigate to the Automation Portal in your browser using `https://<Azure external IP Address>`

Configure Inbound Port Rule

An inbound port rule must be configured on the Automation Portal VM to allow incoming traffic to access the website.

To configure the Automation Portal VM inbound port rule:

1. Navigate to **Automation Portal VM > Networking > Network settings**.
2. In the **Rules** section, click on **Create port rule > Inbound port rule**.
3. Set **Source** to **Any**.

4. Set **Source port ranges** to *****.
5. Set **Destination** to **Any**.
6. Configure **Destination port ranges** with the IIS ports configured for the Kelverion Automation Portal and Kelverion Automation Portal API websites, respectively. Default port values in the Automation Portal are **443, 8443**.
7. Set **Protocol** to **TCP**.
8. Set **Action** to **Allow**.
9. Set **Priority** as suggested.
10. Click **Save**.

Configure Inbound Firewall Rule

An inbound Windows firewall rule must be configured on the Automation Portal VM machine to allow incoming traffic to access the website.

To configure the Automation Portal VM inbound Windows firewall rule:

1. On the Automation Portal VM machine, open **Windows Defender Firewall**.
2. Expand the tree view and right-click on Inbound Rules to add a new **Inbound Rule**.
3. Specify Rule Type as **Port**.
4. Configure allowed TCP ports with the IIS ports configured for the Kelverion Automation Portal and Kelverion Automation Portal API websites, respectively. Default port values in the Automation Portal are **443, 8443**.
Your firewall may already allow connection on port 443.
5. Specify **Domain** and **Private** profiles for the rule.
6. Specify a rule **Name** and click **Finish**.

Update App Registration Redirect URLs

The Azure Portal App Registration must be updated to specify new redirect URLs with the Automation Portal VM public IP address:

To configure the VM public IP address in the app registration:

1. Navigate to **Azure Portal Microsoft Entra ID**.
2. Navigate to **Manage > App registrations** and locate your Automation Portal app registration.
3. Navigate to **Manage > Authentication**.
4. Under **Web Redirect URIs** add a new URI with the Portal VM public IP address. For example:
https://20.12.215.184/oauth2-redirect.html
5. Under **Single-page application Redirect URIs** add a new URI with the Portal VM public IP address. For example: **https://20.12.215.184/**
6. Click **Save**.

Configure IIS CORS Rule

A new IIS CORS rule must be added to list the Automation Portal VM public IP address as a valid origin.

To configure a new IIS CORS rule for the VM Public IP Address:

1. Connect to your Portal Azure IIS VM using Remote Desktop.
2. Navigate to the folder where the Kelverion Automation Portal API has been installed. By default, this is **C:\inetpub\Kelverion\Kelverion Automation Portal API**.
3. Use a text editor with Administrator elevated privileges to edit the **web.config** XML file.
4. Under the **configuration/system.webServer/cors** XML element configure a new **add** section. The new section should be like the existing ones but will specify the public IP address as the origin:


```
<add allowCredentials="true" maxAge="120" origin="https://20.12.215.184">
  <allowHeaders allowAllRequestedHeaders="true"/>
  <allowMethods>
    <add method="GET"/>
    <add method="HEAD"/>
    <add method="POST"/>
    <add method="PUT"/>
    <add method="PATCH"/>
    <add method="DELETE"/>
  </allowMethods>
</add>
```
5. Save the **web.config** file.
6. Restart/recycle the **Kelverion Automation Portal API** website.

Configure Content Security Policy

The Automation Portal Content Security Policy must be modified to permit the Automation Portal VM as a valid source.

To configure the Content Security Policy:

1. Connect to your Portal Azure IIS VM using Remote Desktop.
2. Navigate to the folder where the Kelverion Automation Portal has been installed. By default, this is **C:\inetpub\Kelverion\Kelverion Automation Portal**.
3. Use a text editor with Administrator elevated privileges to edit the **appsettings.json** configuration file.
4. Locate **Content-Security-Policy** in the appsettings.json file. This contains a semi-colon separated list of policy settings. You will have to modify the **img-src** and **connect-src** settings.
5. Locate the **img-src** setting and add a URL that includes the Automation Portal VM public IP address. Make sure to also configure the API port (default 8443). For example, the modified img-src should look like this:


```
img-src \u0027self\u0027 https://Portal-VM:8443 https://10.0.0.4:8443
https://20.12.215.184:8443 ...
```
6. Locate the **connect-src** setting and add a URL that includes the Automation Portal VM public IP address. Make sure to also configure the API port (default 8443). For example, the modified connect-src should look like this:


```
connect-src \u0027self\u0027 https://Portal-VM:8443 https://10.0.0.4:8443
https://20.12.215.184:8443 ...
```
7. Save the appsettings.json file.
8. Restart/recycle the **Kelverion Automation Portal** website.

Configure Automation Portal API URL

The Automation Portal API URL must be modified to use the Automation Portal VM IP address.

To configure the Content Security Policy:

1. Connect to your Portal Azure IIS VM using Remote Desktop.
2. Navigate to the folder where the Kelverion Automation Portal has been installed. By default, this is **C:\inetpub\Kelverion\Kelverion Automation Portal**.
3. Navigate into the **wwwroot\configuration** folder and locate the **apiConfig.js** file.
4. Use a text editor with Administrator elevated privileges to edit the **apiConfig.js** file.
5. Locate the **PORTAL_API_URL** setting and update it with a URL that contains the Automation Portal VM public IP address. Make sure to also configure the API port (default 8443). For example, the modified **PORTAL_API_URL** should look like this:
`"PORTAL_API_URL": "https://20.12.215.184:8443/api/v2"`
6. Save the **apiConfig.js** file.
7. Restart/recycle the **Kelverion Automation Portal** website.

Update IIS Server Certificate

To avoid certificate warnings when accessing the Automation Portal website with the public IP address, the public IP address must be included in the IIS certificate.

Licensing your Automation Portal

After installing the portal, you must provide a valid Kelverion license to activate the portal.

Provide the **Server ID** value found on the **Setup > Settings** page to Kelverion when requesting a license.

Alternatively run this PowerShell command on the target IIS server:

```
(Get-CimInstance -ClassName Win32_ComputerSystemProduct).UUID
```

To apply your license:

1. Select **Setup > Settings**.
2. In the **License** section, click **Upload license**.
3. Click **browse for files** or drag and drop a file into the drop-zone.
4. Click **Upload**.

High Availability Licensing

The license for a high availability installation must include the Server ID for each Automation Portal host that is part of the cluster. Run the following command on each host to obtain each host Server ID.

```
(Get-CimInstance -ClassName Win32_ComputerSystemProduct).UUID
```

License Notifications

The Keverion Automation portal will notify administrators when the current license is less than 30 days from expiring and when the user capacity is nearing capacity. License notifications will be displayed every 30 minutes and can be closed by clicking **X**.



Licensing: Your licensed user capacity is at 90 percent. Please contact [Keverion support](#) for assistance.



Getting Started

Logging In

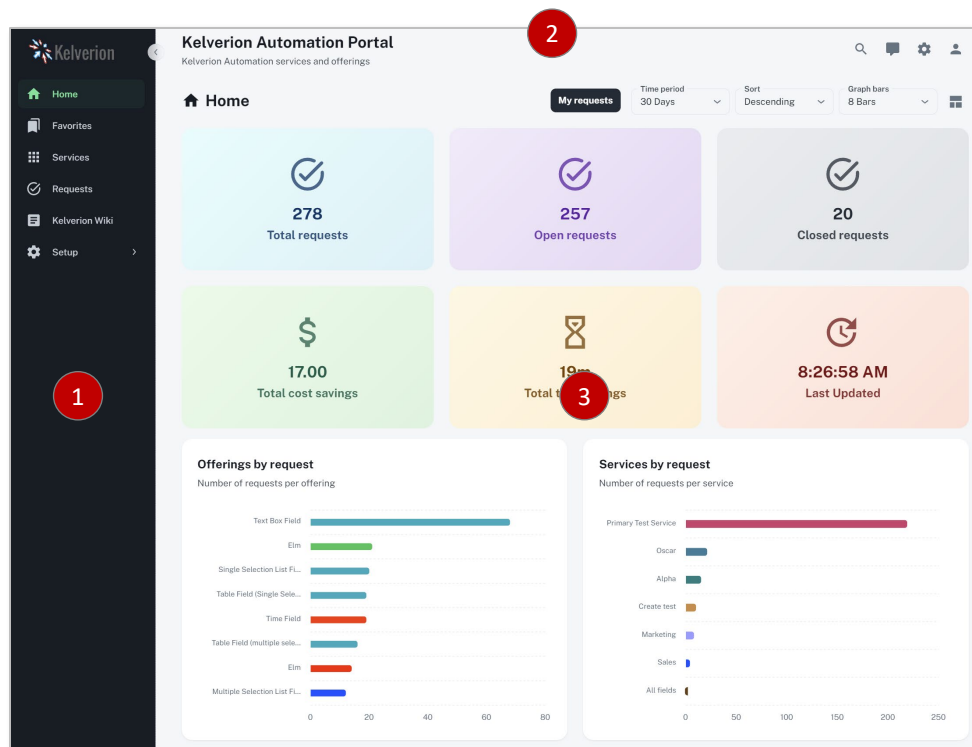
Login to the Kelverion Automation Portal from your web browser using your Microsoft Entra ID account.

1. Open a web browser and go to URL provided by your portal administrator.
2. On the **Login** page, click **Login**.
3. Enter your Microsoft Entra ID credentials.

Overview of the User Interface

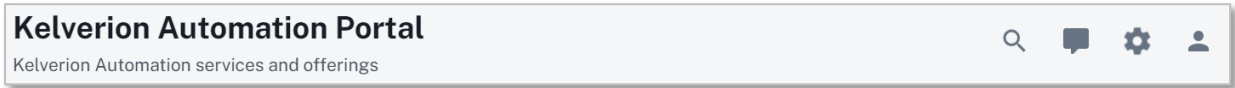
The Kelverion Automation Portal user interface is composed of three main sections.

1. The **Application bar** is located at the top of the user interface and provides access to search, feedback and help, display settings and your user profile.
2. The **Navigation bar** appears on the left side of the user interface by default and provides access to all available pages.
3. The **Content** view takes up the remaining space of the user interface and displays the pages of the portal, including your home page, services, offerings, and requests.



Application Bar

The **application bar** appears at the top of the Automation Portal window. The application bar displays the name of your portal and provides quick access buttons to access universal search, feedback/support, display settings and your user account.



Universal Search

The universal search tool lets you search for Automation Portal resources, including services, offerings, service folders, wiki folders, and wiki documents.

To search for Automation Portal resources:

1. On the application bar, click the **search** button ().
2. On the search dialog, enter the text of the item that you are looking for. The dialog will display a list of items that match the text that you entered. The tag on the right identifies what type of resource the item is.
3. Click on the desired item. The page for that item is opened in the Content frame.

Feedback

The feedback tool provides you with a way to send feedback to Automation Portal administrators.

To send feedback:

1. On the application bar, click the **feedback** button ().
2. In the **Comment** box, enter the feedback that you want to send.
3. Click **Submit**.

Display Settings

As an Automation Portal user, you can customize some aspects of the layout and appearance. You can switch between light and dark modes, improve the contrast between foreground and background, control where the navigation bar appears and choose an accent color.

To change your display settings:

1. On the application bar, click the **display settings** button ().
2. Select between **light** and **dark** modes.
3. Select **hi** or **low** contrast.
4. Select the **layout** of the navigation bar.
5. Select an **accent color**.

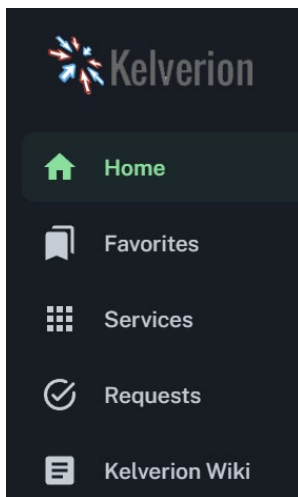
Account Profile

The account profile displays the name of the user that is connected to the Automation Portal and provides a logout option.

To log out of the portal:

1. On the application bar, click the **profile** button ().
2. Click **Logout**.
3. You are taken to the Automation Portal login screen.

Navigation Bar



The **Navigation bar** provides links that you can use to open the pages that are available to you. By default, the navigation bar appears on the left side of the portal window; however, you can also choose to display the Navigation bar horizontally beneath the Application bar.

Collapse or expand the navigation bar by clicking the < button or > button, respectively.

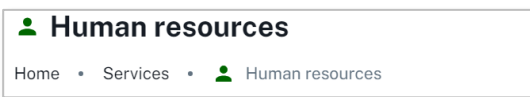
If you collapse the width of the browser window sufficiently, the navigation bar will be hidden and replaced with a navigation button in the top left corner of the portal. Clicking the navigation button opens the navigation bar as a drawer, which will close when you select a page.

Note: If you login as an administrator, the navigation bar will include a **Setup** option that can be expanded to access pages for configuring the portal and for creating and managing the services and offerings that the portal will provide.

Content View

The **Content view** is where the current Automation Portal page is displayed. When you first login to the Automation Portal the Content view will display your **Home** page. Return to the Home page at any time by clicking the **Home** button () on the navigation bar.

Many of the pages that will be displayed in the Content view include **Breadcrumb bar** that contains links that you can use to navigate the Automation Portal. Some Breadcrumb bars, especially those on setup pages, will also include buttons for tasks related to the page.



The Automation Portal is organized into six primary categories:

- The **Home** page provides a landing page for operators and administrators. The Home page provides an overview of the current state of your portal and quick access to your requests.
- The **Favorites** page displays a list of your favorite offerings.
- The **Services** page displays a list of all the services and offerings that are available to you.
- The **Requests** page displays all your requests, requests from other team members and requests that require your approval.
- The **Reports** page is available when you log in either with reporter or administrator role and provides an overview of the Automation Portal operation.
- The **Wiki** page displays wiki documents that were created to help you use the Automation Portal.
- The **Setup** page is available when you log in as an administrator and provides resources and tools for managing your Automation Portal and creating and managing content.

Setup and Administration

The Automation Portal provides administrators with a collection of pages to support customization, building and managing content and for viewing feedback and diagnosing issues.

- The **Settings** page provides options for customizing your portal and managing your license.
- The **Services** page provides tools for creating and managing services and offerings.
- The **Lists** page provides tools for creating and managing lists for list fields.
- The **Connections** page provides tools for creating and managing database connections.
- The **Queries** page provides tools for creating and managing queries for list and table fields.
- The **Permissions** page provides tools for managing group permissions for offerings.
- The **Wiki** page provides tools for creating and managing wiki items.
- The **Integrations** menu provides integration options.
 - The **Webhooks** page provides tools for creating and managing webhook integrations.
- The **Logs** page displays log entries that can be used to diagnose issues with your portal.
- The **Feedback** page displays feedback entries from portal operators.
- The **About** page displays information about your portal instance.

Settings Page

The **Settings** page provides Automation Portal administrators with options for customizing the appearance and behavior of the portal. To open the Settings page, select **Setup > Settings** on the navigation bar. The Settings page is organized into **License**, **Customization**, **Request**, **Wiki** sections.

License

The **License** section displays information about your current Automation Portal license, including status, type, contact information, expiry date, and user count. It also lets you update your license.

To update your Automation Portal license:

1. On the **License** section, click **Upload license**.
2. Click **browse for files** or drag and drop a file into the drop-zone.
3. Click **Upload**.

Customization

The **Customization** section provides options for customizing the appearance of the portal.

1. In the **Portal name** box, enter the name of the portal. This name is displayed in bold font on the Application bar.
2. In the **Portal description** box, enter a brief description. This description is displayed beneath the portal name on the Application bar.
3. Select a **Currency symbol** to be displayed with monetary values.

4. In the **Dashboard refresh interval** box, enter an interval in seconds that will be used to periodically refresh the **Home** page dashboard.
5. In the **Support link** box, enter a support link for your organization. Operators use the support link to contact your organization's technical support team.
6. Click **Save changes**.

Note: You can completely customize the appearance of the portal for all users using a custom theme file. For more information see [Theme Customization](#).

Request

The **Request** section provides options for changing the behavior of the request submission form.

1. In the **Query timeout** box, enter the number seconds for database queries to run before timing out.
2. In the **Request attachment max size** box, enter the maximum file size in MB for file attachment fields.
3. To require approvers to enter a reason when rejecting a request, turn on the **Require a reason to reject a request** switch.
4. To stay on the new request form after submitting a request, turn on the **Stay on request** switch.
5. Click **Save changes**.

Wiki Section

The **Wiki** section provides options for customizing the appearance and behavior of your wiki.

To change the name of the Wiki:

1. In the **Wiki name** box, enter the name of the Wiki. This name will be displayed in the navigation bar.
2. Click **Save changes**.

The portal wiki can be enabled or disabled as needed.

To disable the Wiki page:

1. On the top right corner of the **Wiki** card, turn off the **Enabled** switch.
2. Click **Save changes**.

To enable the Wiki page:

1. On the top right corner of the **Wiki** card, turn on the **Enabled** switch.
2. Click **Save changes**.

Portal Roles

The **Portal Roles** section allows portal administrators to configure the portal role groups, for portal installations that use on-premises Windows authentication.

To specify the Operator role group(s):

1. In the **Operator Groups** box, enter one or more Active Directory groups.
2. Click **Save changes**. Members of the specified groups will have the **Operator** role in the Automation Portal.

To specify the Developer role group(s):

1. In the **Developer Groups** box, enter one or more Active Directory groups.
2. Click **Save changes**. Members of the specified groups will have the **Developer** role in the Automation Portal.

To specify the Reporting role group(s):

1. In the **Reporting Groups** box, enter one or more Active Directory groups.
2. Click **Save changes**. Members of the specified groups will have the **Reporting** role in the Automation Portal.

Maintenance Tasks

The **Settings** page also provides administrators with maintenance tools, including the ability to purge log and feedback entries and deleted items. *Purged items are permanently removed from the portal database and cannot be recovered.*

To purge all log entries:

1. On the toolbar click **Maintenance** and then click **Purge logs**.
2. On the confirmation dialog, click **Purge**.

To purge all feedback entries:

1. On the toolbar click **Maintenance** and then click **Purge feedback**.
2. On the confirmation dialog, click **Purge**.

To purge all deleted resources:

1. On the toolbar click **Maintenance** and then click **Purge deleted items**.
2. On the confirmation dialog, click **Purge**.

Services Page

The **Services** page provides administrators with tools to create and manage the services and offerings that will be provided by the portal. Open the Services page by selecting **Setup > Services** on the navigation bar.

To create a new service:

1. Click **New service**.
2. In the **Name** box, enter the name of the service.
3. In the **Description** box, enter a brief description of what the service does.
4. Select **Change icon** to change the icon that is used to display the service.
5. Select **Change color** to change the color of the icon used to display the service.
6. Click **Create service**.

When you create a new service the Automation Portal automatically opens the service and selects the **Offerings** tab so that you can start adding folders and offerings to your service. Folders are used to organize a collection of related offerings within a service. You cannot create a folder within a folder.

To open an existing service, on the service list click the service's **Name** or click the **more** button (⋮) and then select **View**.

In the Services **Integrations** tab you can configure root, or global, integrations. Root integrations are invoked for all portal requests. For details, please refer to the [Integration](#) section.

To add root level integrations:

1. On the services page select the **Integrations** tab.
2. Click **Add Integration**.
3. In the **Type** box, select the integration type.
4. In the **Integration** box, select the integration.
5. In the **Request state** box, select the request state that will trigger the integration.
6. Continue adding integrations to the list as needed.
7. Click **Save changes**.

To add a folder to a service:

1. Open the service that you want to add the folder to.
2. Click **New folder**.
3. In the **Name** box, enter the name of the folder.
4. In the **Description** box, enter a brief description of the folder.
5. To make the folder available to mobile devices, turn on the **Mobile enabled** toggle.
6. Click **Create folder**.

When you create a new folder the Automation Portal will automatically open the folder and select the **Offerings** tab so that you can start adding offerings to the folder. To open an existing folder, on the service list click the folder's **Name** or click the **more** button (⋮) and then select **View**.

To add an offering to a service or folder.

1. Click **New offering**.
2. In the **Name** box, enter the name of the offering.
3. In the **Description** box, enter a brief description of what the offering does.
4. In the **Tag** box, enter an optional tag to help manage the offering.
5. Select **Change icon** to change the icon used to display the offering.
6. Select **Change color** to change the color of the icon used to display the offering.
7. To make the offering available to mobile devices, turn on the **Mobile enabled** toggle.
8. If the offering requires an external reference identifier, turn on the **Requires External Ref. ID** toggle.
9. Click **User Groups** to select the Microsoft Entra ID groups that will have permission to create requests for this offering. Users in these groups can only view and manage their own requests.
10. Click **Team Groups** to select the Microsoft Entra ID groups that will have permission to create requests for this offering and view and manage the requests made by others in their groups.
11. Click **Approval Groups** to select Microsoft Entra ID groups that have permission to approve and reject the requests made for this offering. Adding one or more groups automatically flags the offering as requiring group approval.

12. In the **Cost Saved** box, enter an optional cost saved by using this offering.
13. In the **Time Saved** box, enter the time saved, in total minutes, by using this offering.
14. Click **Create offering**.

Once you have created an offering, the Automation Portal will automatically open the offering and select the **Fields** tab so that you can start adding fields to your offering. The Automation Portal provides several types of fields including fields for text, fields for date and time and fields for selecting values and database content.

Integrations can also be configured at the service, folder or offering level. Service and folder integrations are invoked for requests corresponding to offerings that are contained that service or folder. Offering integrations are invoked for requests corresponding to the offering. For details, please refer to the [Integration](#) section.

To add an integration to a service, folder or offering:

1. Open the service, folder or offering that you want to add the integration to.
2. On the service, folder or offering page select the **Integrations** tab.
3. Click **Add Integration**.
4. In the **Type** box, select the integration type.
5. In the **Integration** box, select the integration.
6. In the **Request state** box, select the request state that will trigger the integration.
7. Continue adding integrations to the list as needed.
8. Click **Save changes**.

To add new field a to an offering:

1. On the **Offering** page select the **Fields** tab.
2. Click **New field**.
3. In the **Name** box, enter the name of the field.
4. In the **Type** box, select the field type. The default is **Text Box**, but you can also select:
 - **Date** fields provide a date picker.
 - **DateTime** fields provide a date and time picker.
 - **File Attachment** fields provide a browser for uploading attachments.
 - **Hidden** fields are not displayed on the offering form, but the assigned value will be submitted with requests for this offering and can be used by your automation runbooks.
 - **List Multiple Selection** fields provide a drop-down list that can be used to select one or more values.
 - **List Single Selection** fields provide a drop-down list that can be used to select a single value.
 - **Radio Button** fields provide an alternate way of selecting a single value from a concise list of options.
 - **Secure Text Box** fields provide a box for entering sensitive text.
 - **Switch** fields provide a toggle that can be switched on or off.
 - **Table Display Only** fields provide a read-only table for displaying information.
 - **Table Multiple Selection** fields provide a table for selecting multiple rows from a database query.
 - **Table Single Selection** fields provide a table for selecting a single row from a database query.
 - **Text Box** fields provide a single-line text box for entering non-sensitive text.

- **Text Area** fields provide a multiple-line text box or non-sensitive text.
 - **Time** fields from a time picker.
5. In the **Help Text** box, enter an optional help message to be displayed beneath the field.
 6. Depending on the field **Type** that you selected, there may be additional options to configure.
 7. Click **Create field**.

On the **Fields** tab you can use the mouse to change the position of a field on the offering form.

To change the position of a field:

1. On the **Fields** tab, click and hold the **drag** button ().
2. Use the mouse to position the field in the list and then release.

You can also add markdown text that will be displayed to operators when they open the offering. The markdown text will be displayed on the side of the offering page and is a good place for providing instructions on how to use the offering. For more information on formatting offering markdown, refer to the [Common Mark](#) documentation.

To add markdown text to an offering:

9. On the offering page select the **Markdown** tab.
10. In the **Title** box, enter a title to be displayed at the top of the markdown text. Click the markdown editor to start adding content. Use the options in the markdown editor toolbar to help format your content.
11. Click **Save markdown**.

Services and the folders and offerings that they contain can be enabled and disabled. A disabled service, folder or offering will be hidden from operators but can still be accessed by administrators by selecting **Setup > Services**.

To enable/disable a service:

1. On the service list, turn on or off the **Enabled** switch.
2. Alternatively, click the **more** button () and select **View**.
3. Select the **Settings** tab.
4. On the **Details** card, turn the **Enabled** switch on or off.
5. Click **Save changes**.

To enable/disable an offering or folder:

1. Open the service that contains the folder or offering that you want to enable/disable.
2. Select the **Offerings** tab.
3. On the list, turn on or off the **Enabled** switch.
4. Alternatively, click the **more** button () and select **View**.
5. Select the **Settings** tab.
6. On the **Details** card, turn the **Enabled** switch on or off.
7. Click **Save changes**.

If you need to reorganize your offerings and folders, you can move them to another location. When you move a folder to another service, it moves the folder and all the offerings that it contains to the new service.

To move an offering to another service and/or folder:

1. Click the **more** button (⋮) on offering that you want to move and click **Edit**.
2. Select the **Settings** tab.
3. In the **Services** box, select the service that you want to move the offering to.
4. In the **Folders** select the folder that you want to move the offering to.
5. Click **Save changes**.

Lists Page

The **Lists** page provides tools for creating and managing lists for **list** and **radio button** offering fields. Open the Lists page by selecting **Setup > Lists** on the navigation bar.

To create a new list:

1. Click **New list**.
2. In the **Name** box, enter a name for the list.
3. In the **Description** box, enter a brief description of the list.
4. Click **Add Value** and enter a value in the box that is provided.
5. You can optionally also enter a value description.
6. Continue adding new values to the list as needed.
7. When you are finished adding values, click **Create list**.

You can change a list after you have created it, and this includes changing the name and description and the values that are in the list.

To modify an existing list:

1. Click the **More** button (⋮) on the row of the list that you want to modify and then select **Edit**.
2. Modify the list as needed.
3. Click **Save changes**.

You can delete a list *that is not in use*. To determine whether a list is in use, review the **Use count** column of the Lists table.

To delete a list that is not being used:

1. Click the **more** button (⋮) on the row of the list that you want to delete and then select **Delete**.
2. Confirm that you want to delete the list by clicking **Delete**.

Connections Page

The **Connections** page provides tools for creating and managing database connections. Database connections are used by the queries that are used to select database content for list and table fields. Open the Connections page by selecting **Setup > Connections** on the navigation bar.

New: The Connections page is new for Automation Portal 4.0 and is an improvement on previous versions, where connections were configured when you created a query. This new page minimizes duplication and makes it easier to manage connections that are shared by multiple queries.

To create a new database connection:

1. Click **New connection**.
2. In the **Name** box, enter a name for the new connection.
3. In the **Description** box, enter a brief description of the connection.
4. Select an **Authentication Type**.
5. In the **Data Source** box, enter the name or network address of the SQL Server instance to connect to.
6. In the **Initial Catalog** box, enter the name of the database that you want to connect to.
7. If you selected **SQL Authentication**:
 - 7.1. In the **Username** box, enter the username of the user that you want to connect with.
 - 7.2. In the **Password** box, enter the password of the user that you want to connect with.
8. If you selected **Managed Identity**:
 - 8.1. If you are using a **System Assigned Managed Identity**, leave the Managed Identity Client ID box empty.
 - 8.2. If you are using a **User Assigned Managed Identity**, in the **Managed Identity Client ID** box, enter the client ID GUID of the managed identity.
9. Click **Create connection**.

You can change an offering after you have created it; however, any changes that you make will affect every query that is using the connection.

To modify an existing connection:

1. Click the **more** button (**:**) on the row of the connection that you want to modify and then select **Edit**.
2. Modify the connection as needed.
3. Click **Save changes**.

You can delete a connection *that is not in use*. To determine whether a connection is in use, review the **Use count** column of the Lists page.

To delete a connection that is not being used:

1. Click the **more** button (**:**) on the row of the connection that you want to delete and then select **Delete**.
2. On the confirmation dialog, click **Delete**.

Queries Page

The **Queries** page provides tools for creating and managing database queries for **list** and **table** offering fields. Open the Queries page by selecting **Setup > Queries** on the navigation bar.

To create a new database query:

1. Click **New query**.
2. In the **Name** box, enter the name of the new query.
3. In the **Description** box, enter a brief description of the new query.
4. Select a **Connection** from the list of available connections.
5. For **Command Type** select **SQL query** or **Stored procedure**.
 - 5.1. If you selected SQL query as the command type, then in the **SQL query** box, enter the select statement that will be used to fetch the desired data.
 - 5.2. If you selected Store procedure as the command type, then select the **Stored procedure** that will be used to fetch the desired data.
6. If your new query requires input from another query
 - 6.1. Click **Add query input**.
 - 6.2. In the **Input name** box, enter the name of the input. This name should match the name of a parameter in your SQL query or stored procedure. The parameter in your **SQL query** should be the input name prefixed with @.
 - 6.3. In the **Source query** box, select the query that will be used as the source query.
 - 6.4. In the **Source query output** box, select the output from the source query that will be used to provide the value for this input parameter.
7. If your new query requires input from an offering text field:
 - 7.1. Click **Add field input**.
 - 7.2. In the **Input name** box, enter the name of the input. This name should match the name of a parameter in your SQL query or stored procedure.
 - 7.3. In the **Field name** box, select the field that will be used to provide the value for the input parameter.
8. If your new query will provide outputs for other queries:
 - 8.1. Click **Add query output**.
 - 8.2. In the **Output name** box, enter the name of the output. This should match an output from your SQL query or stored procedure.
9. Click **Create query**.

You can change an existing query after you have created it; however, be aware that changes that you make will affect any offerings that make use of the query.

To modify an existing query:

1. Click the **more** button (**:**) at the right side of the row that contains the query that you want to modify and click **Edit**. The **Edit query** page appears.
2. Modify the query as needed.
3. Click **Save changes**.

You can delete a query *that is not in use*. To determine whether a query is in use, review the **Use count** column of the Queries page.

To delete a query that is not being used:

1. Click the **more** button (**:**) on the row of the query that you want to delete and click **Delete**.
2. On the confirmation dialog, click **Delete**.

Security Recommendations

To enable the execution of SQL queries and stored procedures, you must provide a User Account in SQL Server with the necessary permissions. To minimize security risks, we recommend that you:

- Create a special *least privileged* database account for use with the Automation Portal.
- For SQL queries, use a database user with *Select-only* permissions to the required tables.
- Preferably, use stored procedures exclusively and limit the database user permissions to only those stored procedures that are required for Automation Portal.

Guidance on SQL Permissions can be found at: <https://docs.microsoft.com/en-us/sql/relational-databases/security/permissions-database-engine?page=sql-server-ver15>

Wiki Page

The **Wiki** page provides Automation Portal administrators with tools for creating and managing wiki content. Open the Wiki page by selecting **Setup > Wiki** on the navigation bar.

To create a new wiki document:

1. Optionally, open the folder that you want to create the document in.
2. Click **New document**.
3. In the **Name** box, enter the name of the document.
4. In the **Description** box, enter a description of the folder.
5. Click **User groups**. Select the Microsoft Entra ID groups that you want to have access to the document and then click **Done**.
6. Click beneath the toolbar in the **Content** card to start entering content. Use the options in the content toolbar to assist with formatting. For more information on formatting markdown text, refer to the [Common Mark](#) documentation.
7. Click **Create document**.

Folders can be used within the wiki to organize related content. You can create folders with folders to create a hierarchy of wiki items.

To create a new wiki folder:

1. Click **New folder**.
2. In the **Name** box, enter the name of the new folder.
3. In the **Description** box, enter a description of the folder.

4. Click **User groups**. Select the Microsoft Entra ID groups that you want to have access to the folder and then click **Done**.
5. Click **Create folder**.

You can enable and disable wiki documents and folders to make them available and unavailable to portal operators, respectively.

To enable/disable a wiki items:

1. Navigate to the folder that contains the document/folder that you want to enable/disable.
2. In the wiki items list, turn on or off the **Enabled** switch.

You can delete wiki documents and folders that are no longer required.

To delete wiki documents and folders:

1. Navigate to the folder that contains the wiki item that you want to delete.
2. Click the **more** button (**:**) on the row of the wiki item that you want to delete and click **Delete**.
3. On the confirmation dialog, click **Delete**.

Integrations

Webhooks Page

The **Webhooks** page provides Automation Portal administrators with tools for creating and managing webhook integrations. Open the Webhooks page by selecting Setup > Integrations > Webhooks on the navigation bar.

To create a new webhook:

1. Click **New webhook**.
2. In the **Name** box, enter a name for the new webhook.
3. In the **Description** box, enter a brief description of the webhook.
4. In the **URL** box, enter the webhook URL.
5. If your new webhook includes headers:
 - 5.1. Click **Add header**.
 - 5.2. In the **Name** box, enter the name of the header.
 - 5.3. In the **Value** box, enter the value of the header.

You can change an existing webhook after you have created it; however, be aware that changes that you make will affect any integrations that make use of the webhook.

To modify an existing webhook:

1. Click the **more** button (**:**) at the right side of the row that contains the webhook that you want to modify and click **Edit**. The **Edit webhook** page appears.
2. Modify the webhook as needed.
3. Click **Save changes**.

You can delete a webhook *that is not in use*. To determine whether a webhook is in use, review the **Use count** column of the Webhooks page.

Logs Page

The **Logs** page provides Automation Portal administrators with a place to review diagnostic information, which can help monitor the status of the service and investigate and resolve issues. Open the Logs page by selecting **Setup > Logs** on the navigation bar.

You can select a **Level** on the Logs page toolbar to quickly filter diagnostic information by level (e.g., critical, error, warning, etc.). To page the details of a log entry, click the **more** button () on the row of the log that you want to page and select View. Log details include:

- The **Message** property provides a description of the event.
- The **Category** property specifies the event category.
- The **Created** property specifies the date and time of the event.
- The **Level** property specifies the severity of the event (e.g., critical, error, warning, etc.)
- The **ID** specifies the unique ID of the event.
- The **Stack Trace** property specifies the location of the code that generated the event.

You can purge all log entries from the Automation Portal database if they are no longer required. *Purged log entries will be permanently removed* from the Automation Portal database and will no longer be available.

To purge all log entries:

1. Click **Purge logs**.
2. On the confirmation dialog, click **Purge**.

Feedback Page

The **Feedback** page provides Automation Portal administrators with a place to review feedback from Automation Portal operators. Open the Feedback page by selecting **Setup > Feedback** on the navigation bar.

To page the details of a feedback entry, click the **more** button () on and then select **View**. The log details page is organized into two sections: Details and Feedback.

The **Details** section displays information about the feedback entry, including:

- The **User** property specifies the email of the user the submitted the feedback.
- The **URL** property specifies the URL of the Automation Portal page that was open when the feedback was submitted.
- The **Created** property specifies the date and time that the feedback was submitted.

The **Feedback** section displays the content of the feedback message that was submitted.

You can purge all feedback entries from the Automation Portal database if they are no longer required. *Purged feedback entries will be permanently removed* from the Automation Portal database and will no longer be available.

To purge all feedback entries:

1. On the Feedback page, click **Purge feedback**.
2. On the confirmation dialog, click **Purge**.

Permissions Page

The **Permissions** page provides Automation Portal administrators with tools for managing the Microsoft Entra ID groups that are used to control access to offerings. To open the Permissions page, select **Setup > Permissions** on the navigation bar.

New: The Permissions page is new for Automation Portal 4.0, and it lets you manage group permissions for all your offerings in a single place. You can also make batch changes to one or more offerings at the same time.

The permissions list contains all the offerings in your portal and displays the number of User, Team and Approval groups that are used by each one. To review the names of the Microsoft Entra ID groups that are used by an offering, click the usage number.

To change the permissions for a single offering:

1. On the row of the offering that you want to update, click the **more** button (**:**) and then select **Update**.
2. You can quickly remove a group by clicking its delete (**x**) button.
3. To change the groups that can use the offering click the **User groups** box. Select the groups that you want to include and unselect the ones that you want to exclude. When you are finished, click **Done**.
4. To change the groups that can use the offering and review and manage requests made by other team members click **Team groups**. Select the groups that you want to include and unselect the ones that you want to exclude. When you are finished, click **Done**.
5. To change the groups that can approve or reject requests for this offering click the **Approval groups** box. Select the groups that you want to include and unselect the ones that you want to exclude. When you are finished, click **Done**.
6. Click **Save changes**.

The permissions page also lets you change multiple offerings at the same time. You can add or remove one or more users, team or approval groups from multiple offerings. You can also replace the user, team, and approval groups for multiple offerings.

To change the permissions for multiple offerings:

1. On the navigation bar, select **Setup > Permissions**.
2. Select the checkbox of the offerings that you want to update.
3. On the toolbar, click **Update**. The bulk group permissions dialog appears.
 - a. Select the **Operation** that you want to apply.
 - b. Select the **Group type** that you want to apply the changes to.

- c. Click the **Groups** box. Select the groups that you want to add, remove, or replace.
4. To add additional operations, click **Add**.
5. Click **Save changes**.

Themes Page

The **Themes** page provides Automation Portal administrators with tools for customizing the portal display themes, to make the Automation Portal a better fit for your organization. Open the Themes page by selecting **Setup > Themes** on the navigation bar.

Enable Custom Themes

On the Themes page, click on the **Enable Custom Themes** switch to enable or disable theme customization with one click. This way, you can keep theme customization disabled while configuring all the settings and then enable theme customization when it is ready for the entire organization.

To enable/disable custom themes:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Enable Custom Themes** switch.
3. Click **Save changes**.

Import/Export

Import/export your custom theme settings by using the Import and Export buttons on the Themes page. Note that logo images are not imported/exported, only light and dark mode theme colors and fonts.

To export your current custom themes:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Export** button.
3. A Themes export file will be downloaded in your **Downloads** folder.

To import themes from an export file:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Import** button.
3. Select a previously exported themes export file.
4. Click the **Import** button.
5. The themes settings will be imported.

Light Mode and Dark Mode

Customize the Light Mode and Dark Mode themes by selecting the **Light** or **Dark** tab respectively, and then configuring specific settings as needed. You can display the portal in Light Mode or Dark Mode by configuring the Mode setting, in Display Settings, on the Application Bar.

To customize light or dark mode themes:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Light/Dark** tab.
3. Configure specific color settings.
4. Configure Logo and Compact Logo as needed.
5. Click **Save changes**.

Customize Portal Font

In the **Font** tab you can customize font that is used for displaying text throughout the portal.

To change the portal font:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Font** tab.
3. Select one of the available fonts. By default, the portal is using the Public Sans font.
4. Click the switch next to the font picker to enable the custom font.
5. Enable custom themes.
6. Click **Save changes**.

Customize Specific Theme Settings

On the Themes page there are settings for customizing specific aspects of the portal.

To customize a specific setting:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Light/Dark** tab.
3. Click on a specific color setting to display the color picker. Choose the desired color and then click anywhere on the page or click Esc to exit the color picker.
4. Alternatively, in the color text box, you can enter an RGB color value in the format **#RRGGBB**.
5. Click the switch next to the color box to enable the specific custom color.
6. Enable custom themes.
7. Click **Save changes**.

For example, to customize the Application Bar light mode background color:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Light** tab.
3. In the **Application Bar** section, change the **Background** color.
4. Click the switch next to the background color box to enable the custom background color.
5. Enable custom themes.
6. Click **Save changes**.

Customize Portal Logo

Starting with version 4.1 of the Automation Portal, you can customize the logo and compact logo for light mode and dark mode, respectively.

To customize the logo:

1. On the navigation bar, select **Setup > Themes**.
2. On the Themes page click the **Light/Dark** tab and scroll to the **Logo** section.
3. Select a **Logo** image to be displayed at the top of the expanded navigation bar. This image must have a size of 38 x 140 pixels.
4. Select a **Compact logo** image to be displayed at the top of the collapsed navigation bar. This image must have a size of 38 x 38 pixels.
5. Click **Save changes**.

Custom Themes Settings Reference

When configuring Themes settings, color values must be specified in the format **#RRGGBB**. Where RR, GG, BB are hexadecimal values for the red, green, and blue color components, respectively. In some cases, the Portal reduces the opacity of the specified color, for example for navigation bar hover colors, to achieve a semi-transparent effect when hovering over an item.

Application Bar

In this section you can customize the appearance of the Application Bar area in the Automation Portal.

Setting	Description
Portal name	Text color for the portal name.
Portal description	Text color for the portal description.
Background	Background color for the application bar.
Contrast background	Background color for the application bar in high contrast mode.

Navigation Bar

In this section you can customize the appearance of the Navigation Bar area in the Automation Portal.

Setting	Description
Background	Background color for the navigation bar.
Item	Text color for menu items.
Item hover	Background color for menu items when hovered over.
Selection	Text color for selected menu items.
Selection background	Background color for selected menu items.
Selection hover	Background color for selected menu items when hovered over.
Open item	Text color for open menu items.
Open item background	Background color for open menu items.
Open item hover	Background color for open menu items when hovered over.
Subitem selection	Text color for selected menu subitems.
Subitem selection background	Background color for selected menu subitems.
Subitem selection hover	Background color for selected menu subitems when hovered over.

Main Area

In this section you can customize the appearance of the Automation Portal main area.

Setting	Description
Background	Background color for the main area.
Panel Background	Background color for main area panels.
List Header Background	Background color list header bar.
Primary Text	Text color for primary text in the main area.
Secondary Text	Text color for secondary text in the main area.
Disabled Text	Text color for disabled text in the main area.
Accent	Accent item color. Visible on certain portal items, switches for example.

Main Area High Contrast (Light Mode Only)

In this section you can customize the appearance of the Automation Portal main area for high contrast mode. This section is only available for Light Mode.

Setting	Description
Background	Background color for the main area.
Panel Background	Background color for main area panels.
List Header Background	Background color list header bar.

About Page

The **About page** provides Automation Portal administrators with information about their portal instance. The About page is organized into three sections: Details, Accessibility Statement and Open source.

Details

The **Details** section provides information about your Automation Portal instance, including the product name and version.

Accessibility Statement

The **Accessibility statement** section provides details of the [Web Content Accessibility Guidelines \(WCAG\)](#) measures that are provided the Automation Portal. The Automation Portal is partially conformant with WCAG 2.1 level AA.

Open Source

The **Open-source** section provides details of all the open-source code libraries that were used to develop the Automation Portal.

Importing and Exporting Resources

The Automation Portal lets you export and import resources (services, offerings, list, etc.) so that you can share them between multiple instances.

Version support: Only export files from Automation Portal version 4.4 or later are supported for import. If you are currently using an earlier version of the Automation Portal, we recommend updating first to the latest version, and then exporting. For update details, please refer to the update sections applicable for your environment, under the portal [Installation](#) section.

Exporting Resources

You can export a collection of resources – such as a service and all the folders and offerings that it contains – or a single resource.

To export a resource:

1. Open the setup page that contains the resources (e.g., Services, offerings, wiki document).
2. On the resource list toolbar, click the **export** button (.
3. The Automation Portal creates an export file, named after the resource(s) that you are exporting (e.g., Services_1730209409932.export) and saves it to your **Downloads** folder.

Importing Resources

You can import export files containing resources so that they can be incorporated into your Automation Portal instance. Every resource in an Automation Portal export file has a globally unique **Coalesce ID** is used to identify duplicate resources.

To import a resource:

1. Open a setup page that contains a list of resources (e.g., Services).
2. On the resource list toolbar, click the **import** button (.
3. Select how to manage duplicate resources:
 - a. If you select **Keep existing resources** your Automation Portal will ignore any resource that has the same coalesce ID as a resource in your instance.
 - b. If you select **Overwrite existing resources** your Automation Portal will overwrite resources in your instance with the resource data that is contained in the export file.
 - c. If you select **Create duplicate resources** your Automation Portal will create novel resources, with a unique name, for any duplicates that it identifies.
4. Click **browse for files**. Select the export file that you want to import and click **Open**.
5. Click **Import**.

Integration

The Automation Portal provides the ability to integrate request changes with other third-party systems. Configured integrations are invoked when requests are created or when request states are modified.

The following request states that can trigger integrations:

- New
- Pending Approval
- Approved

The first step in configuring an integration is to define connectivity settings, so the portal can invoke the integration. Currently the only supported integration type is **Webhooks**. For details on how to configure webhook integration settings, please refer to the [Webhooks Page](#).

Integration Scope

Once you have created an integration, you must configure which requests will trigger the integration. Integrations can be configured at different scopes in the root-service-folder-offering hierarchy:

- Root – integrations will trigger for all portal requests.
- Service – integrations trigger for requests corresponding to service offerings.
- Folder – integrations trigger for requests corresponding to folder offerings.
- Offering – integrations trigger for requests corresponding to an offering

Important: As a best practice, it is recommended to configure narrow scope integrations, at the offering level, and to avoid wide scope root integrations. This way you can avoid situations when integrations are invoked unintentionally.

For details on how to configure integrations at the root/service/folder/offering level, please refer to the [Services Page](#).

Integration Priority

The Automation Portal will invoke integrations according to the following priority rules:

- **Offering Integrations** take priority over **Folder Integrations**
- **Folder Integrations** take priority over **Service Integrations**
- **Service Integrations** take priority over **Root Integrations**

For example, consider a folder with two offerings O1 and O2. If you configure an integration I1 for the folder, the integration will trigger as follows:

- Offering O1 requests will trigger integration I1.
- Offering O2 requests will trigger integration I1.

Integration I1 triggers for both offerings since the integration is configured on the parent folder and neither O1 nor O2 has any offering level integrations.

Now, if you configure a second integration I2 for offering O2, integrations will trigger as follows:

- Offering O1 requests will trigger integration I1.
- Offering O2 requests will trigger integration I2.

This is because offering integration I2 takes priority over folder integration I1 and will override the folder level integration for O2 requests. Integration I1 will continue to be invoked for offering O1, since O1 does not have any offering level integrations.

Kelverion Automation Monitor Service

The Kelverion Automation Monitor Service is a Windows service that monitors portal requests and invokes integrations configured for those requests. When the monitor service detects a request state change, it determines if that request is configured to trigger an integration, and then proceeds to invoke that integration according to its connection settings. Currently the Monitor Service supports webhook integrations.

The service is installed on the same machine as the Automation Portal is installed, and must be started after installation. The service is configured to run as the same user account that is configured on the IIS app pool.

Developer Tools

Enabling the Swagger Interface

Swagger is an interactive API documentation interface that lists every API endpoint exposed by the Automation Portal, along with the parameters, request/response formats, and authentication requirements for each. It allows developers to view and test API calls directly from a browser, without needing a separate API client.

Warning: The Swagger interface should only be enabled in a non-production, developer environment. It exposes the full structure of the API, including authentication schemes, and should never be made available on a production system.

To enable the Swagger interface:

1. Open the API's appsettings.json configuration file.
2. Locate the Swagger section.
3. Set Enabled to true:

```
"Swagger": {  
    "Enabled": true  
}
```

4. Restart the Automation Portal API website in IIS for the change to take effect.

5. Browse to the root URL of the API (e.g. `https://<server>:8443`) to view the Swagger interface.

Getting a bearer token:

Replace <tenantId> and <clientId> with the values for your Automation Portal. Then run the following steps in PowerShell.

1. Request a device code:

```
$device = Invoke-RestMethod -Method Post -Uri  
"https://login.microsoftonline.com/<tenantId>/oauth2/v2.0/devicecode" -ContentType  
"application/x-www-form-urlencoded" -Body @{ client_id = "<clientId>"; scope =  
"api://<clientId>/user_impersonation" }
```

2. Then run `$device.message` to show the URL + code to enter.
3. Open the URL from `$device.message`, enter the code, sign in.
4. Then request the token:

```
$token = Invoke-RestMethod -Method Post -Uri  
"https://login.microsoftonline.com/<tenantId>/oauth2/v2.0/token" -ContentType "application/x-  
www-form-urlencoded" -Body @{ grant_type = "urn:ietf:params:oauth:grant-type:device_code";  
client_id = "<clientId>"; device_code = $device.device_code }
```

5. Then run `$token.access_token | Set-Clipboard` to copy token to clipboard.

Support and Guidance

If you require any support or guidance with deploying the portal and integrating it with your automation tool, our professional services team are ready to help.

Please contact your local sales representative who can organize a services support engagement.

Sending Feedback

If you encounter a problem while working with the Automation Portal or have an idea for making the Automation Portal even better, we would like to hear from you.

You can send us an e-mail at support@kolverion.com

We look forward to hearing from you.