Bring Your Own Key for Microsoft Azure Key Vault and Entrust KeyControl

Integration Guide

2026-04-15

Member of
Microsoft Intelligent Security Association

Microsoft Security

© Entrust Corporation. All rights reserved.

Table of Contents

1. Introduction . . . 1
    1.1. Documents to read first . . . 1
    1.2. Product configurations . . . 2
    1.3. Features tested . . . 2
    1.4. Requirements . . . 2
2. Install and configure Entrust KeyControl . . . 3
    2.1. Deploy an Entrust KeyControl cluster . . . 3
    2.2. Create an Entrust KeyControl Management Vault . . . 3
3. Configure Microsoft Azure . . . 5
    3.1. Create an app registration in Azure . . . 5
    3.2. Add the app to the subscription Reader Role list . . . 7
    3.3. Create an Azure key vault . . . 8
    3.4. Add the app registration to the key vault access policies . . . 11
4. Configure Entrust KeyControl as Microsoft Azure CSP . . . 14
    4.1. Create an Azure client secret . . . 14
    4.2. Create an Entrust KeyControl CSP account for Azure . . . 15
5. Test integration . . . 17
    5.1. Create a key set in Entrust KeyControl . . . 17
    5.2. Create a cloud key in Entrust KeyControl . . . 19
    5.3. Create a cloud key in Azure key vault . . . 22
    5.4. Rotate a cloud key in Entrust KeyControl . . . 23
    5.5. Remove a cloud key in Entrust KeyControl . . . 25
    5.6. Upload a removed cloud key to Azure in Entrust KeyControl . . . 26
    5.7. Delete a cloud key in Entrust KeyControl . . . 27
    5.8. Cancel a cloud key deletion in Entrust KeyControl . . . 28
6. Additional resources and related products . . . 30
    6.1. nShield HSMs . . . 30
    6.2. nShield as a Service . . . 30
    6.3. KeyControl BYOK . . . 30
    6.4. Entrust products . . . 30
    6.5. nShield product documentation . . . 30

Chapter 1. Introduction

Entrust KeyControl has been rebranded as the Entrust Cryptographic Security Platform (CSP) Key Manager.

The Entrust CSP Key Manager continues to provide a comprehensive solution for discovering and managing the lifecycles of cryptographic keys, secrets, certificates, tokens, libraries, protocols, and configurations:

- The KeyControl Compliance Manager is now the Entrust CSP Compliance Manager. It still integrates with Entrust nShield Hardware Security Modules (HSMs) to protect the master keys for the CSP.
- KeyControl Vault is now the Entrust Cryptographic Security Platform Vault. The Cryptographic Security Platform Vaults also still integrate with Entrust nShield HSMs to provide an optional HSM root of trust.

Because the Entrust integrations are tested against specific product versions, this guide is still branded as a "KeyControl" integration. It was tested against a pre-CSP version of KeyControl.

Exercise caution when using an Entrust Integration Guide with a product version that does not match the tested version, because your version might not function in exactly the same way.

Entrust cannot guarantee the success of integrations in configurations other than those indicated in the guide. This guide remains on the website for customers using pre-CSP versions of KeyControl.

This document describes the integration of Microsoft Azure Key Vault Bring Your Own Key (referred to as Azure BYOK in this guide) with the Entrust KeyControl Key Management Solution (KMS).

1.1. Documents to read first

This guide describes how to configure the Entrust KeyControl server as a KMS in Azure BYOK.

Entrust KeyControl v10.2 supports BYOK as an add-on. You can request a free trial of Entrust KeyControl BYOK here: https://go.entrust.com/

> keycontrol-byok-30-day-free-trial.

To install and configure the Entrust KeyControl server see KeyControl Installation and Upgrade Guide.

Also refer to the documentation and set-up process for Microsoft Azure BYOK in the Microsoft Azure Key Vault online documentation.

1.2. Product configurations

Entrust has successfully tested the integration of KeyControl with Azure BYOK in the following configurations:

System	Version
Entrust KeyControl	10.2

1.3. Features tested

Entrust has successfully tested the following features:

- Create cloud key
- Rotate cloud key
- Remove cloud key
- Upload removed cloud key
- Delete cloud key
- Cancel cloud key deletion

1.4. Requirements

Entrust recommends that you allow only unprivileged connections unless you are performing administrative tasks.

Chapter 2. Install and configure Entrust KeyControl

- Deploy an Entrust KeyControl cluster
- Create an Entrust KeyControl Management Vault

2.1. Deploy an Entrust KeyControl cluster

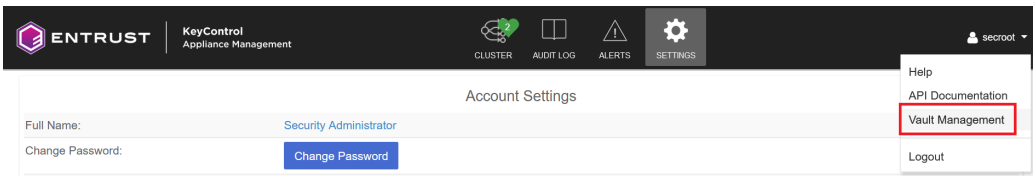
For this integration, Entrust KeyControl was deployed as a two-node cluster on premises. The installation software was downloaded in the form of an OVA file, deployed in VMware ESXi.

Follow the installation and set-up instructions in KeyControl Installation and Upgrade Guide. If using an HSM, the integration guide with the Entrust nShield HSM is available at https://www.entrust.com/documentation. Search for the key phrase **KeyControl nShield HSM**.

2.2. Create an Entrust KeyControl Management Vault

To create an Entrust KeyControl Management Vault:

1. Sign in to the Entrust KeyControl Vault Server Appliance Manager.
2. In the home page, select the user's drop-down menu and select **Vault Management**.

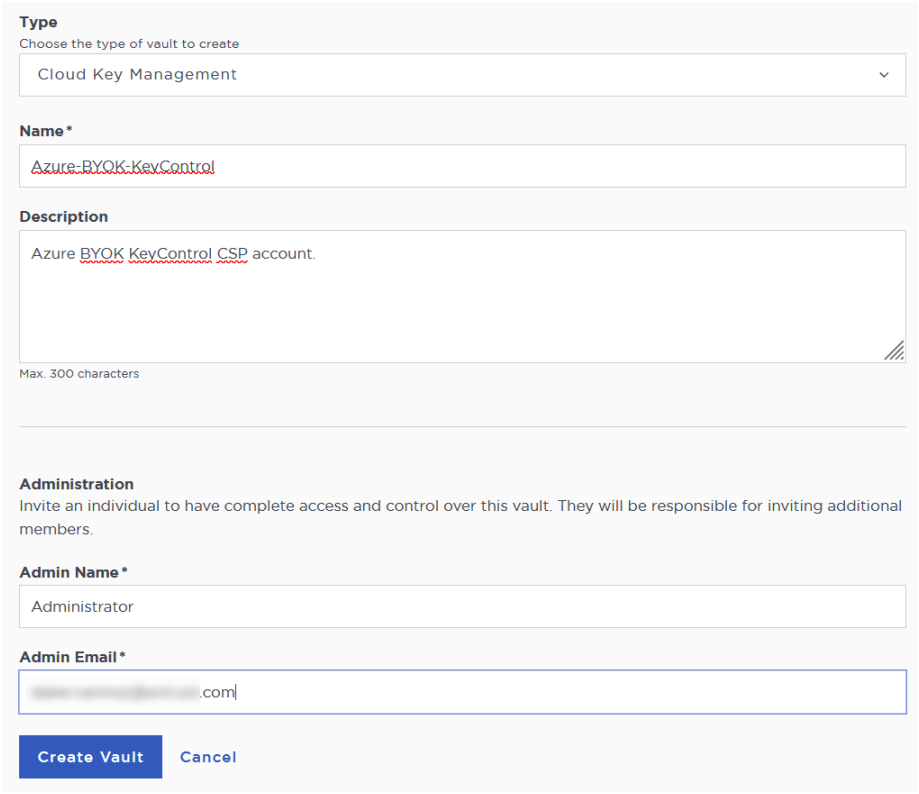


3. Select **Create Vault**.

   The **Create Vault** dialog appears.

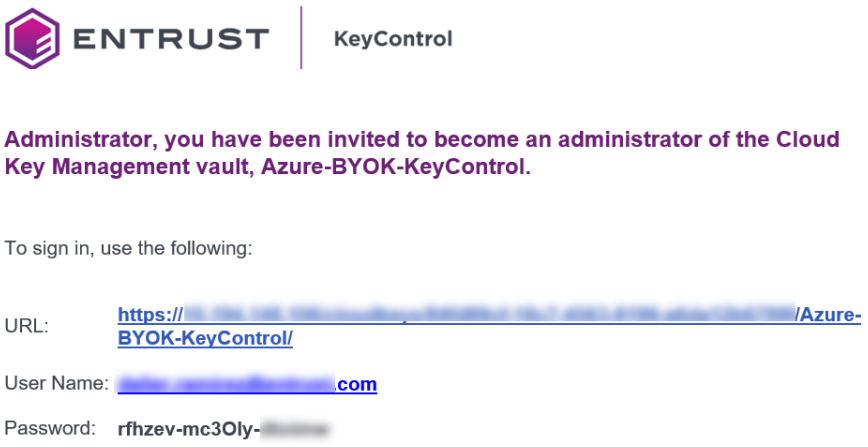
4. In the **Type** drop-down box, select **Cloud Key Management**. Enter the required information.
5. Select **Create Vault**.

   For example:



6. You will receive an email with a URL and login credentials to the Entrust KeyControl vault. Bookmark the URL and save the credentials.

   For example:



7. Sign in to the above URL. Change the one-time password when prompted.

Chapter 3. Configure Microsoft Azure

- Create an app registration in Azure
- Add the app to the subscription Reader Role list
- Create an Azure key vault
- Add the app registration to the key vault access policies

3.1. Create an app registration in Azure

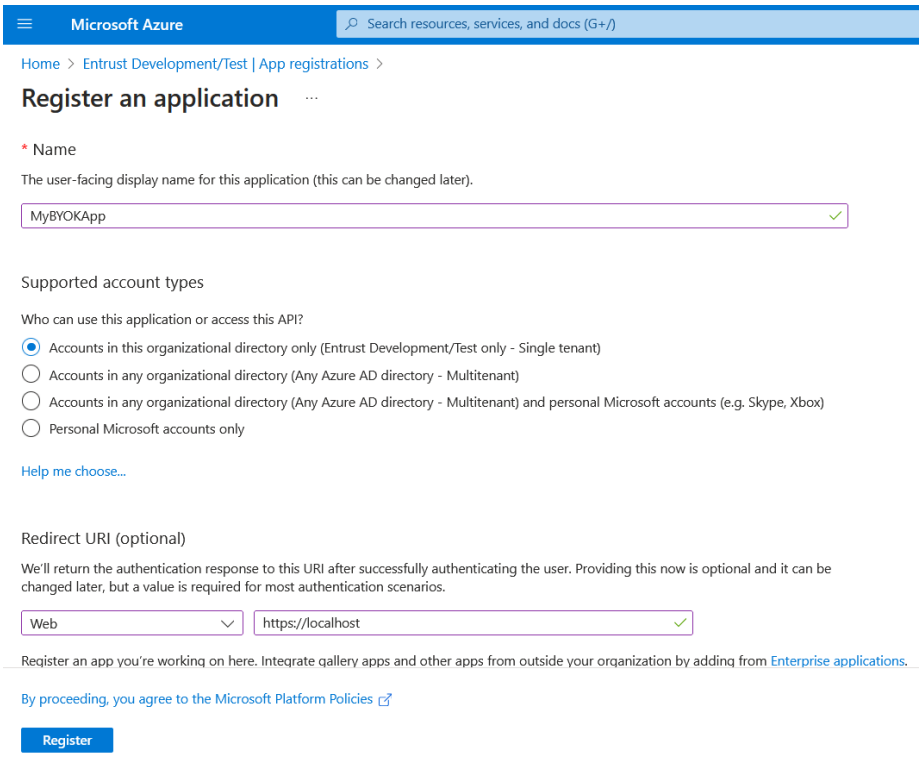
The app registration provides trust between your app and Azure.

1. Open a browser and sign in to the Azure portal https://portal.azure.com/#home.
2. Navigate to **Home** > **Azure Active Directory** > **App registrations**.
3. Select **New registration**.

   The **Register an application** dialog.

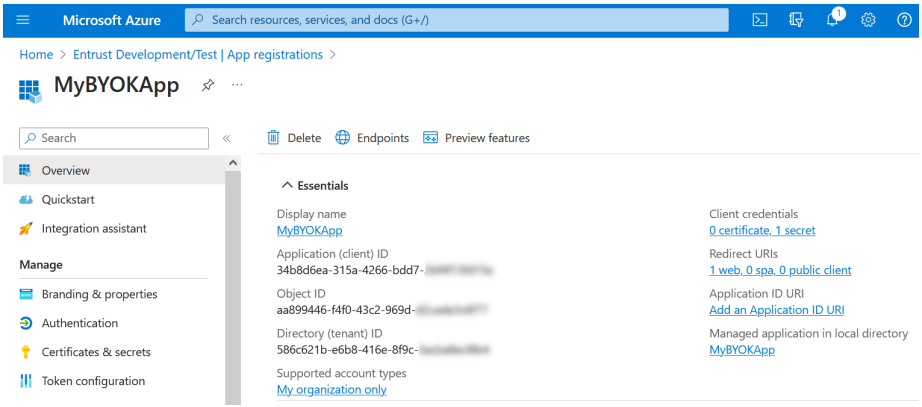
4. Enter the **Name**, a user-facing or friendly name. Select the applicable **Supported account types** and enter a **Redirect URI**.

   For example:



5. Select **Register**.

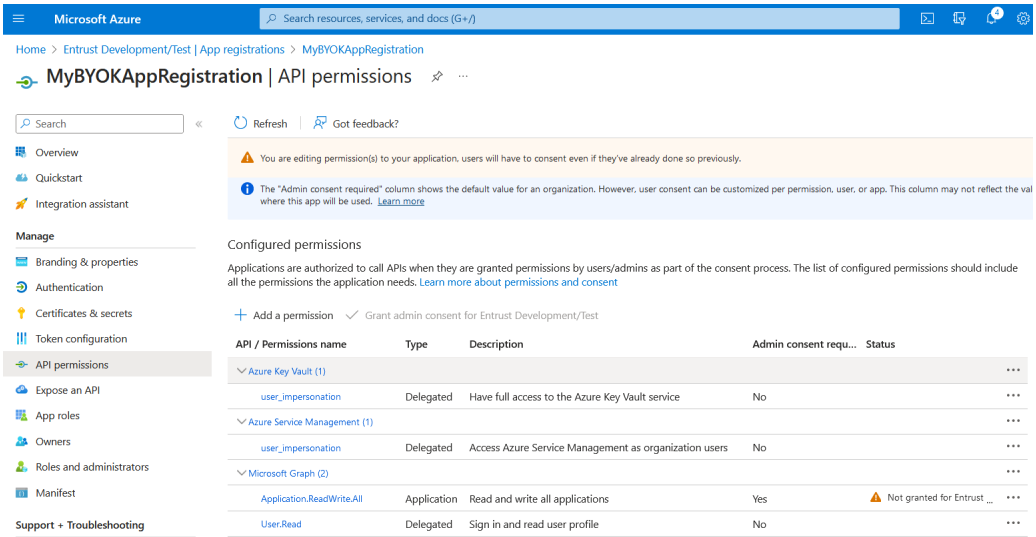
The newly created registration appears.



6. Select **API permissions**. Alternatively, select **Home** > **Azure Active Directory** > **App Registrations** > **<Display name>** > **API permissions**.
7. Select **Add a permission** and add the following permissions:

Microsoft API	Permission	Type
Microsoft Graph	`Application.ReadWrite.All`	Application
Microsoft Graph	`User.Read` (granted by default)	Delegated
Azure Key Vault	`user_impersonation`	Delegated
Azure Service Management	`user_impersonation`	Delegated

For example:

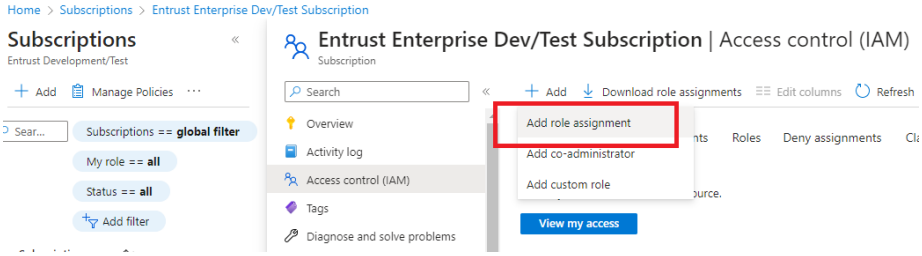


For additional information, see Creating a Service Principal.

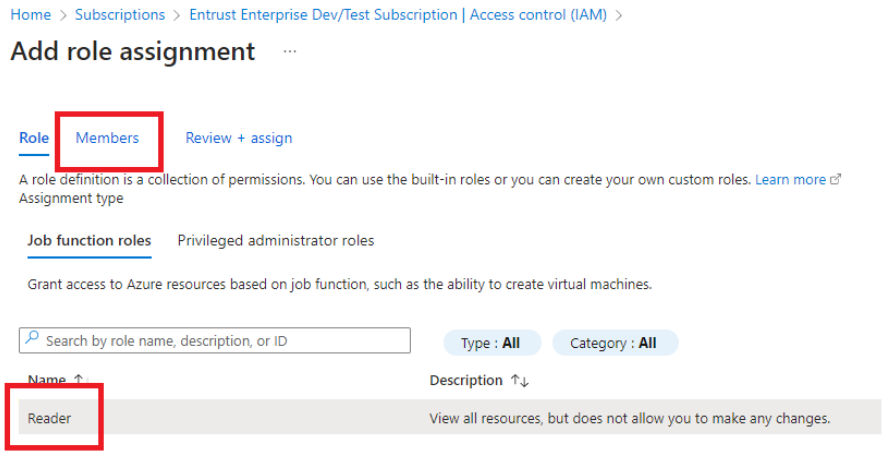
3.2. Add the app to the subscription Reader Role list

The **Owner** permission of the subscription is required to perform this operation.

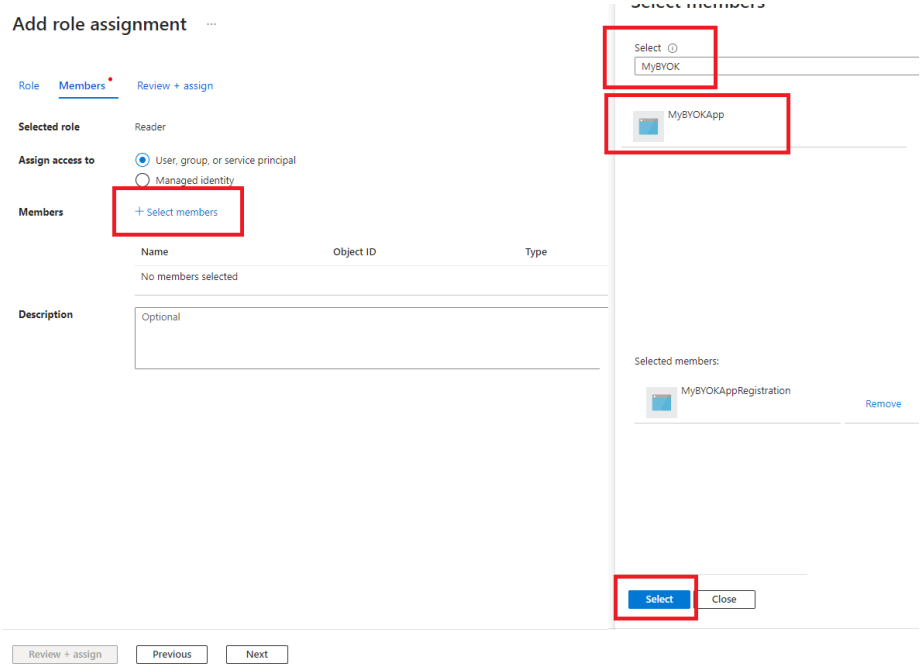
1. Navigate to **Home** > **Subscriptions**.
2. Select your subscription.
3. Select **Access control (IAM)**.
4. Select **Add** and then select **Add role assignment** from the pull-down menu.



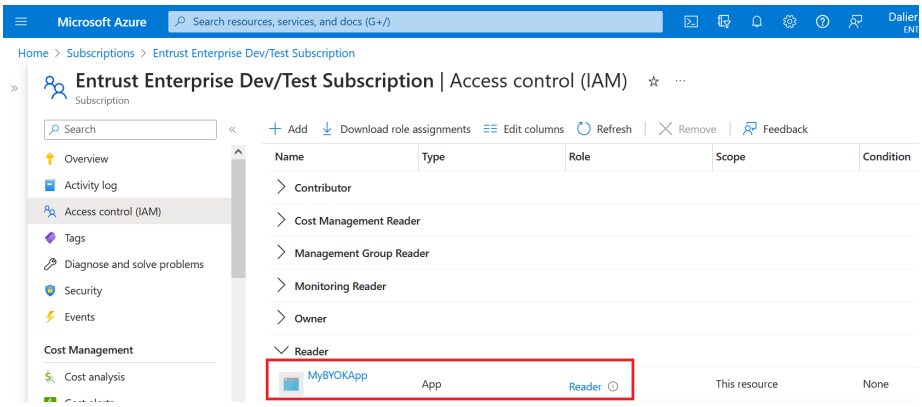
5. In the **Add role assignment** dialog, select the **Reader** role and then select the **Members** tab.



6. Select **Select members**, search for the app **Display name**, and select it.
7. Select **Save**.



The new subscription **Reader** role is added.



3.3. Create an Azure key vault

An existing Azure key vault with **Permission model** equals **Vault access policy** can be used for this integration. A new Azure key vault was created in this integration to show the entire process.

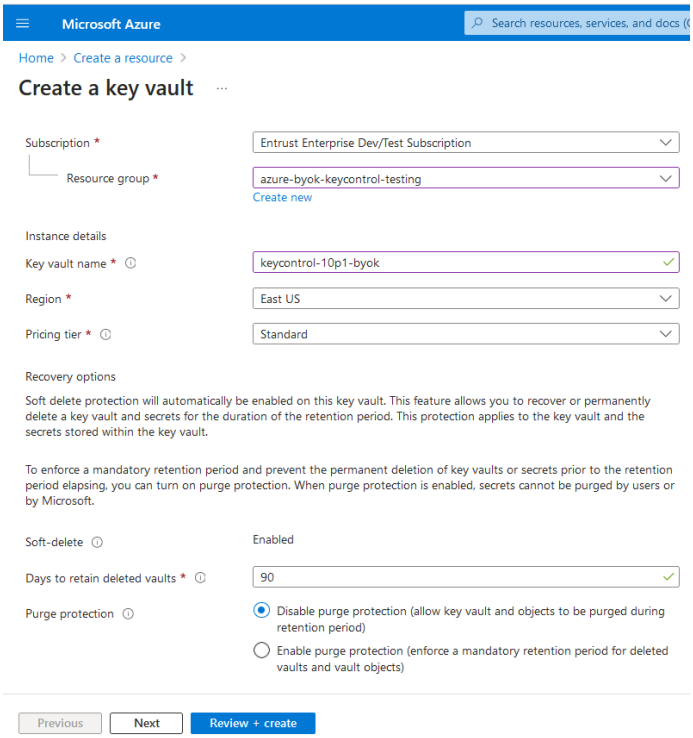
For an existing Azure key vault, proceed to section Add the app registration to the key vault access policies directly, skipping this section entirely.

1. Open a browser and sign in to the Azure portal https://portal.azure.com/#home.
2. In the home page, select the **Create a resource** icon.
3. Select **Key Vault**.

   The **Create a key vault** dialog appears.

4. In the **Basics** tab select the **Subscription** and **Resource group** from the pull-down menu. Enter the instance details.
5. Select **Next**.

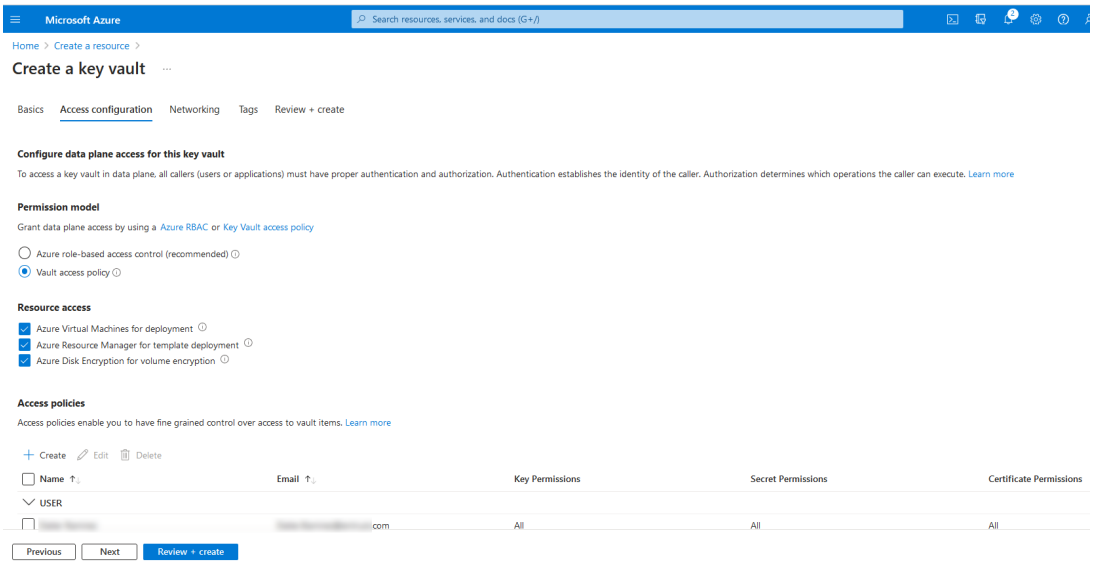
   For example:



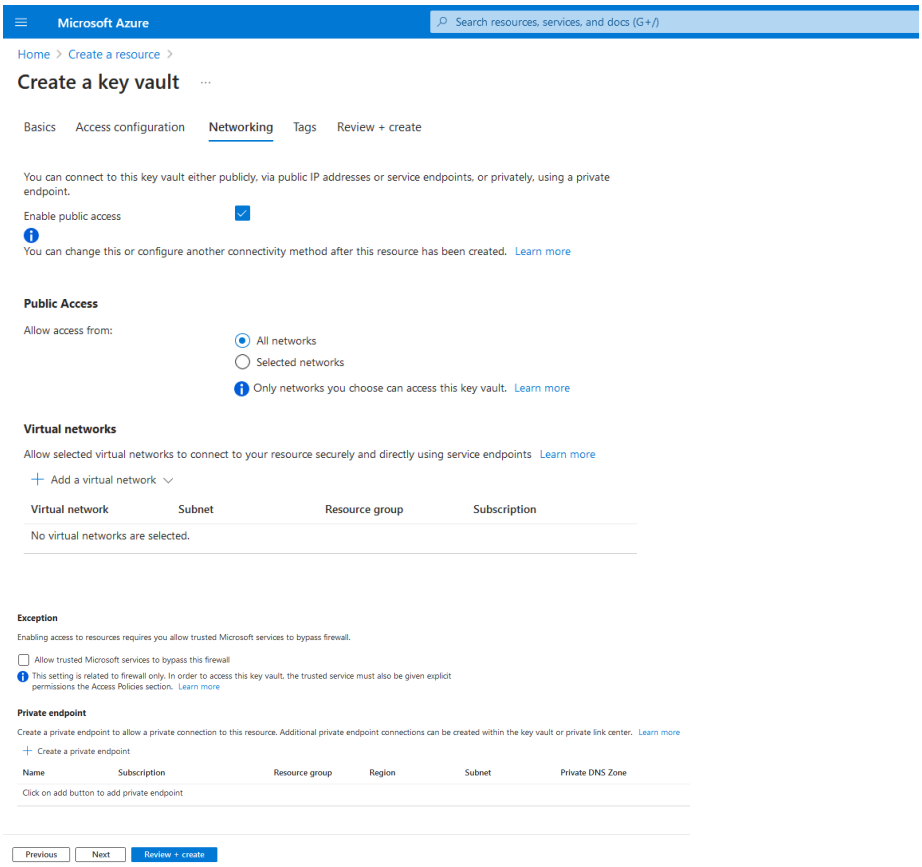
6. In the **Access configuration** tab, select the **Permission model**, **Resource access**, and **Access policies**.
7. If you are using **Vault access policy** for the **Permission model**:

   a. Select the user.
   b. Select **Edit** and select all permissions that apply.
   c. Select **Save** and **Next**.

   All **Key Permissions**, **Secrets Permissions**, and **Certificate Permissions** were selected for the purpose of this integration.

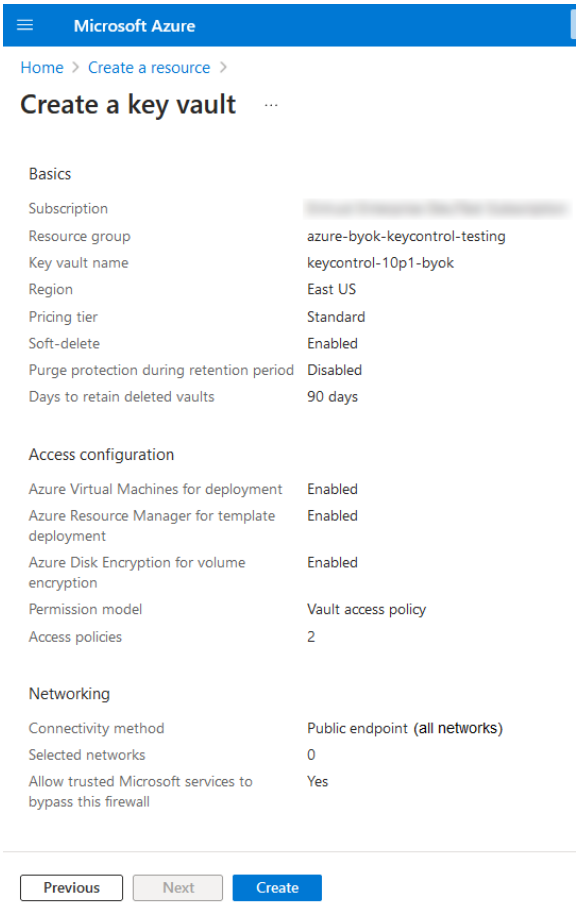


8. In the **Networking** tab, select **Enable public access**.
9. Under **Public access**, select **All networks**.
10. Select **Next**.

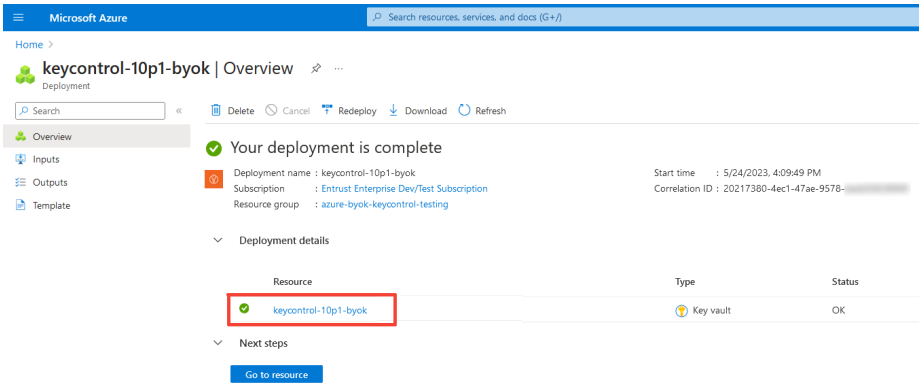


11. In the **Tags** tab enter the required **Name** and **Value**. These were left blank for the purpose of this integration.
12. Select **Next**.

13. Review the information and select **Create**.



14. A deployment page appears. The newly created Azure vault is included.



3.4. Add the app registration to the key vault access policies

These steps configure the key vault policies to allow access by the app.

1. Navigate to **Home** > **Key vault** > **<Key_vault_name>** > **Access policies**.

2. Select **Create**.

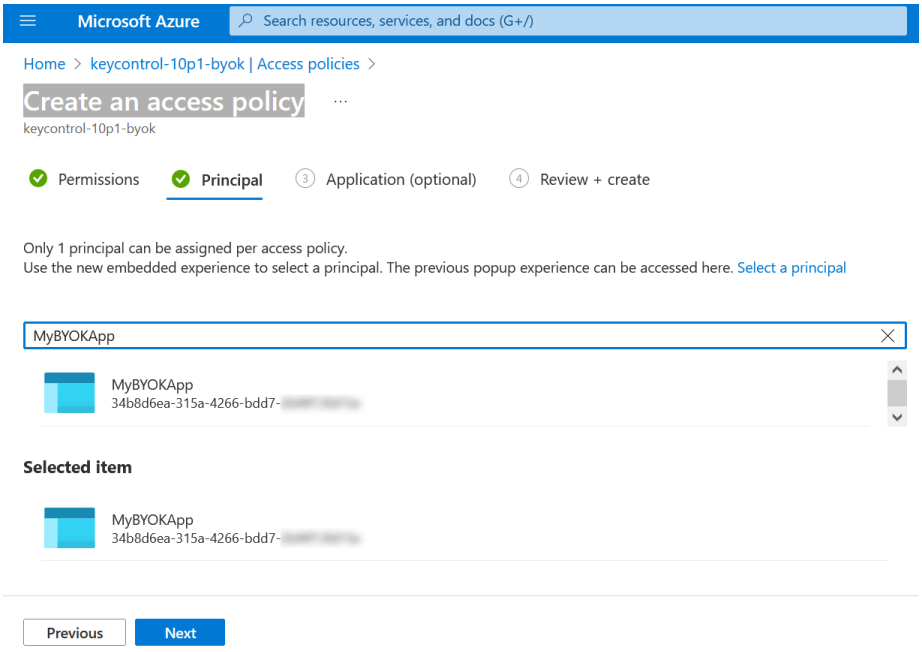
   The **Create and access policy** dialog appears.

3. In the **Permissions** tab select the following **Key permissions**.

Key permissions	Selection
Key Management Operations	All
Privileged Key Operations	All
Rotation Policy Operations	All

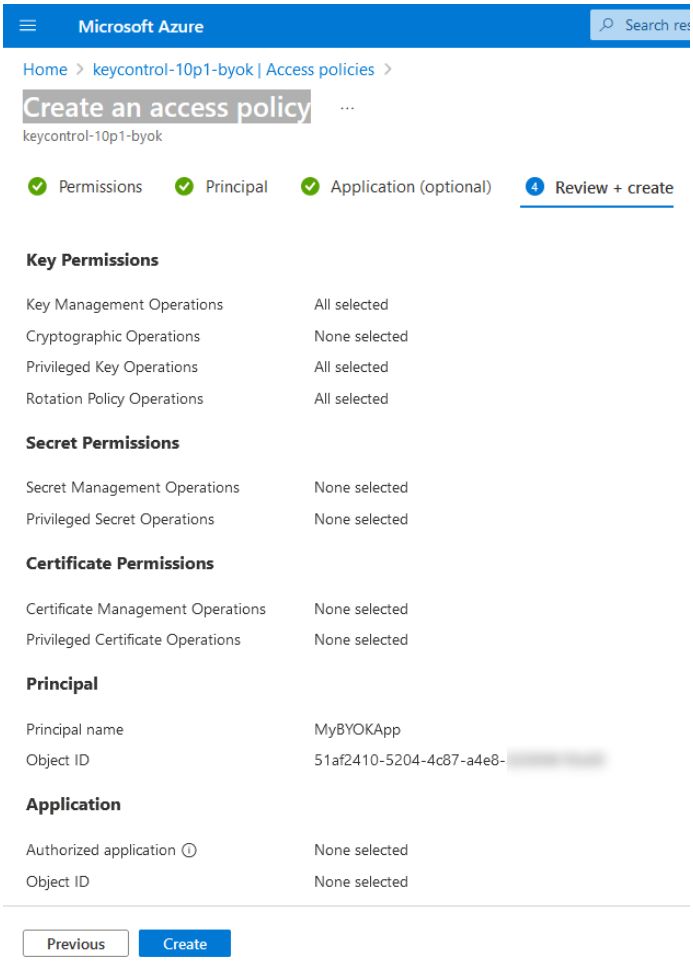
4. Select **Next**.
5. In the **Principal** tab, enter the **Display name** of the app. After the app is found, select the app.
6. Select **Next**.

   For example:

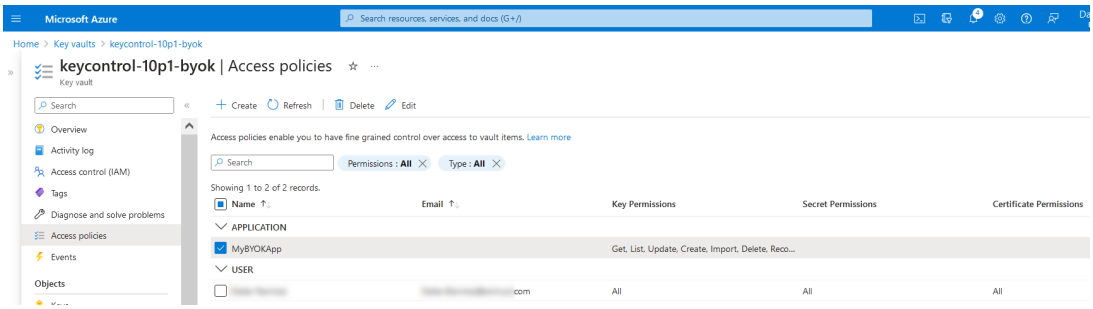


7. Select **Next** in the **Application (optional)** tab.
8. Review the information and select **Create**.

   For example:



9. The **Access policies** page appears. The new vault access policy is included.



For additional information, see Set Permissions for the BYOK Service by Configuring Each Azure Key Vault.

Chapter 4. Configure Entrust KeyControl as Microsoft Azure CSP

- Create an Azure client secret
- Create an Entrust KeyControl CSP account for Azure

4.1. Create an Azure client secret

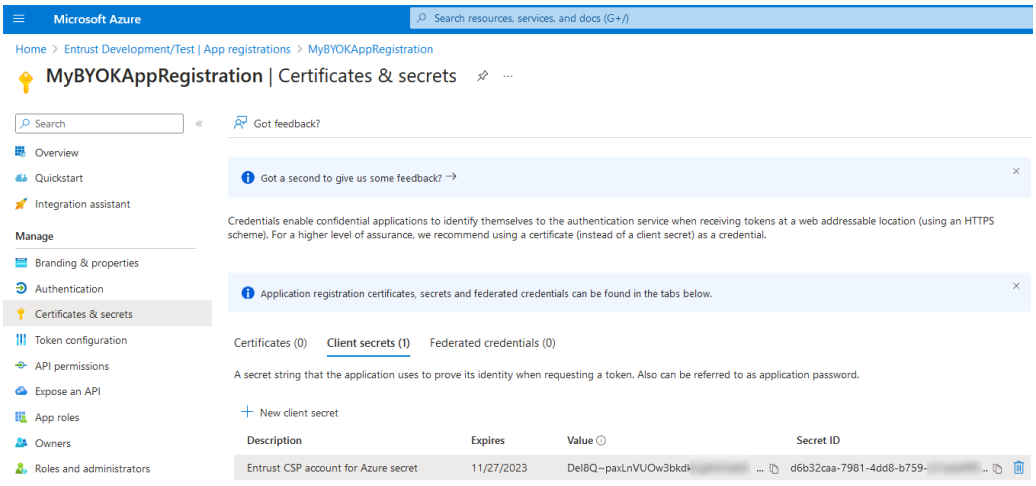
This secret is required to create the Entrust CSP account for Azure. It expires after a set period. You must create the Entrust KeyControl CSP account for Azure before the secret expiration date.

1. Navigate to **Home** > **Azure Active Directory** > **App registrations** > **<App-registration-name>** > **Certificates & secrets**.
2. Select **New client secret**.

   The **Add a client secret** dialog appears.

3. Enter the **Description** and select the expiration date.
4. Select **Add**.

   The **Certificates & secrets** page appears. For example:



5. Copy and save the **Value** of the new client secret.

This value appears in Azure Portal only temporarily. When the portal hides the client secret, it cannot be retrieved and a new secret must be created.

For additional information, see Creating a client secret in Azure Active Directory.

4.2. Create an Entrust KeyControl CSP account for Azure

The following steps establish the connection between Entrust KeyControl and Azure, making Entrust KeyControl the CSP of the Azure application.

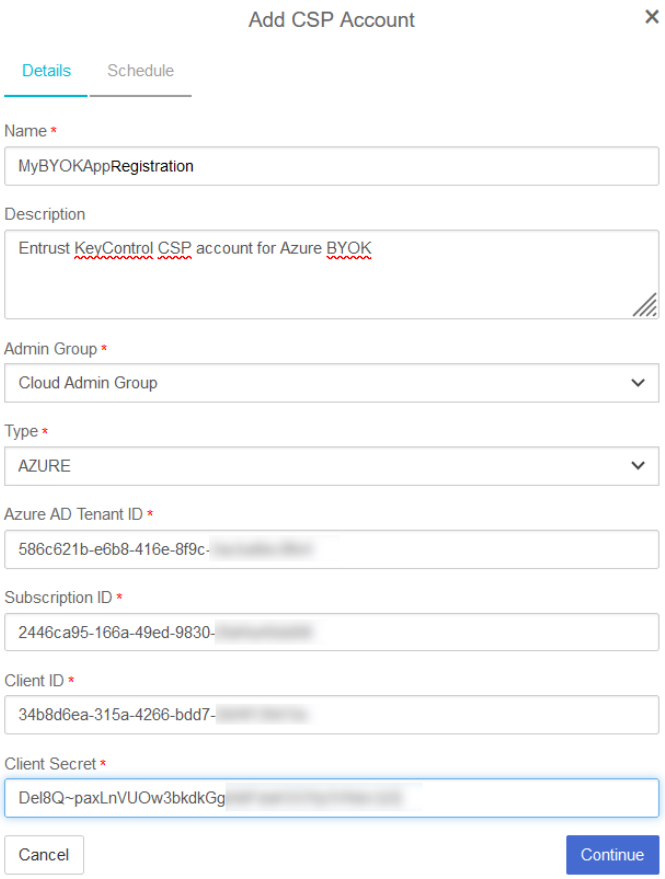
1. Sign in to the Entrust KeyControl Vault URL bookmark from Create an Entrust KeyControl Management Vault.
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CSP Accounts** tab.
4. In the **Action** icon, select **Add CSP Account** in the drop-down menu.

   The **Add CSP Account** dialog appears.

5. In the **Details** tab enter the **Name** and **Description**.
6. In the **Admin Group** drop-down menu box select **Cloud Admin Group**.
7. In the **Type** drop-down menu box select **Azure**.
8. Enter the following from the Azure account:

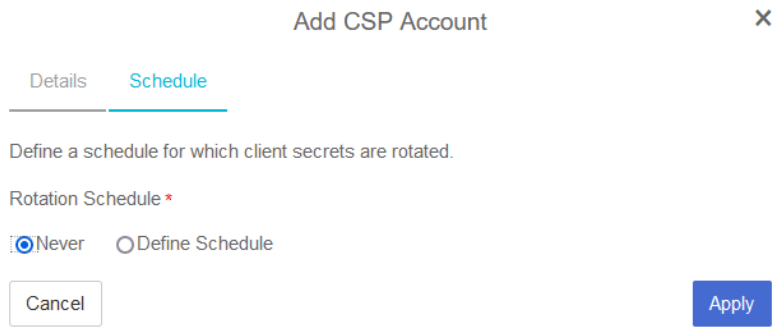
Item	Value
Azure AD Tenant ID	**Home** > **Azure Active Directory** > **App registrations** > **<Display name>** > **Directory (tenant) ID**
Subscription ID	**Home** > **Subscription** > **Subscription ID**
Client ID	**Home** > **Azure Active Directory** > **App registrations** > **<Display name>** > **Application (client) ID**
Client Secret	Value of the secret created in Create an Azure client secret.

For example:

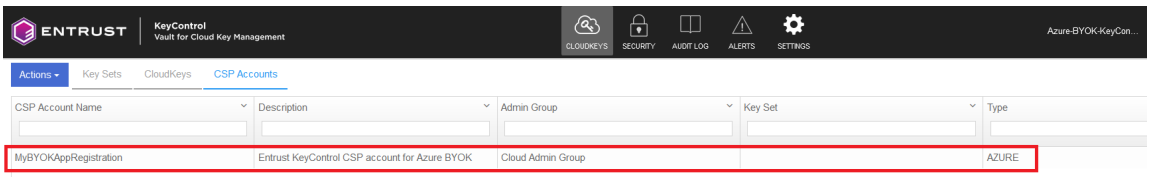


9. Select **Continue**.
10. In the **Schedule** tab, define the rotation schedule.
11. Select **Apply**.

    For example:



    The new CSP account is created.



Chapter 5. Test integration

- Create a key set in Entrust KeyControl
- Create a cloud key in Entrust KeyControl
- Create a cloud key in Azure key vault
- Rotate a cloud key in Entrust KeyControl
- Remove a cloud key in Entrust KeyControl
- Upload a removed cloud key to Azure in Entrust KeyControl
- Delete a cloud key in Entrust KeyControl
- Cancel a cloud key deletion in Entrust KeyControl

5.1. Create a key set in Entrust KeyControl

This key set will be used to create a cloud key in Entrust KeyControl.

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **Key Sets** tab.
4. Select **Actions** > **Create Key Set**.

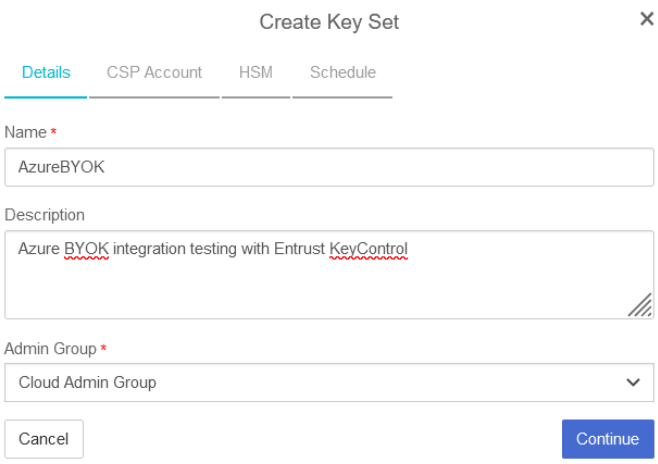
   The **Choose the type of keys...** dialog appears.

5. Choose **Azure Key**.

   The **Create Key Set** dialog appears.

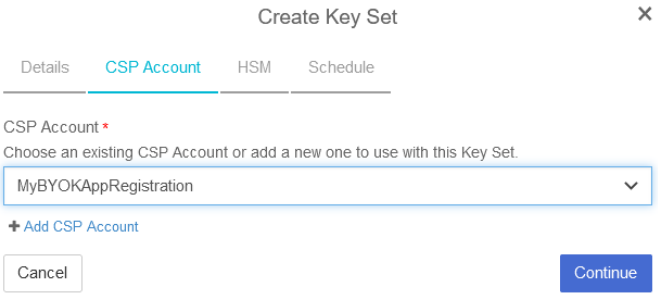
6. In the **Details** tab enter a **Name** and **Description**.
7. In the **Admin Group** menu select **Cloud Admin Group**.

   For example:



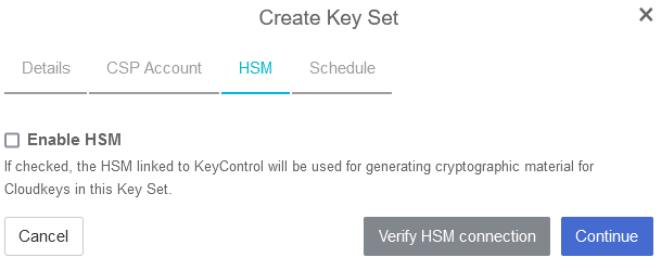
8. Select **Continue**.
9. In the **CSP Account** tab, select the CSP account created in [test-integration:::create-keycontrol-csp-account].

   For example:

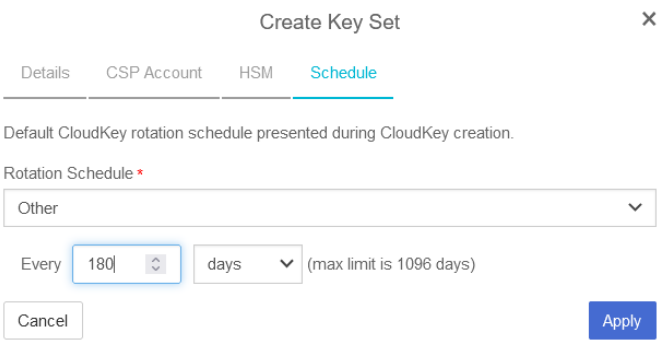


10. Select **Continue**.
11. In the **HSM** tab, select **Enable HSM** if using one. In that case ensure the HSM is configured prior to this step.

    For example:

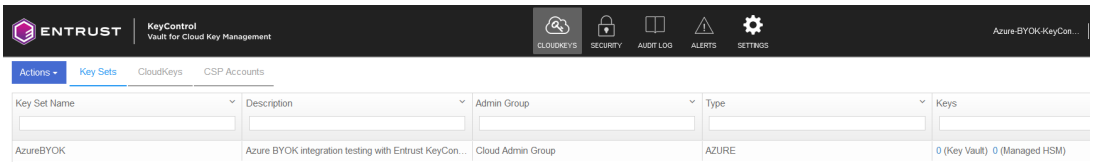


12. Select **Continue**.
13. In the **Schedule** tab, select a **Rotation Schedule** matching the selection made during [test-integration:::create-azure-client-secret]. For example:



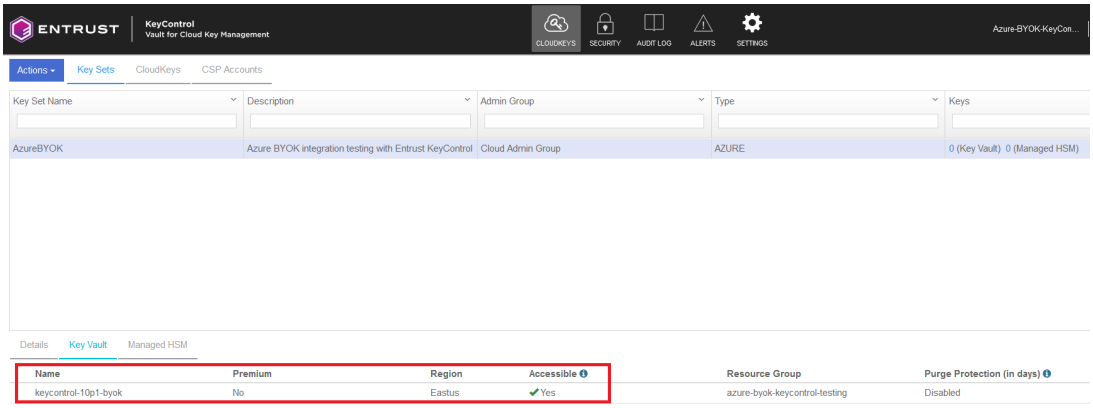
14. Select **Apply**.

    The key set is added. For example:



15. Verify the Azure key vault created in [test-integration:::create-azure-keyvault] is listed in the **Key Vault** tab with setting **Accessible** set to **Yes**.

    For example:



For additional information, see Creating a Key Set.

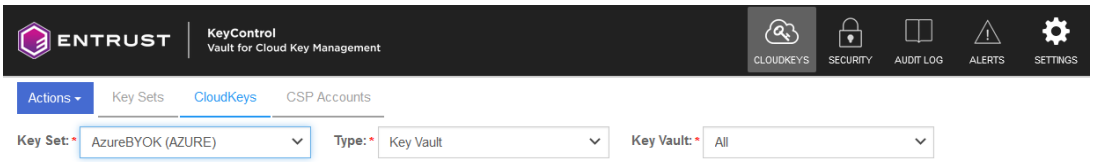
5.2. Create a cloud key in Entrust KeyControl

The following steps create a cloud key in Entrust KeyControl and verify it is available in Azure key vault.

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.

3. Select the **CloudKeys** tab.
4. In the **Key Set** menu, select the **Key Set** created in Create a key set in Entrust KeyControl.
5. In the **Type** menu, select **Key Vault**.

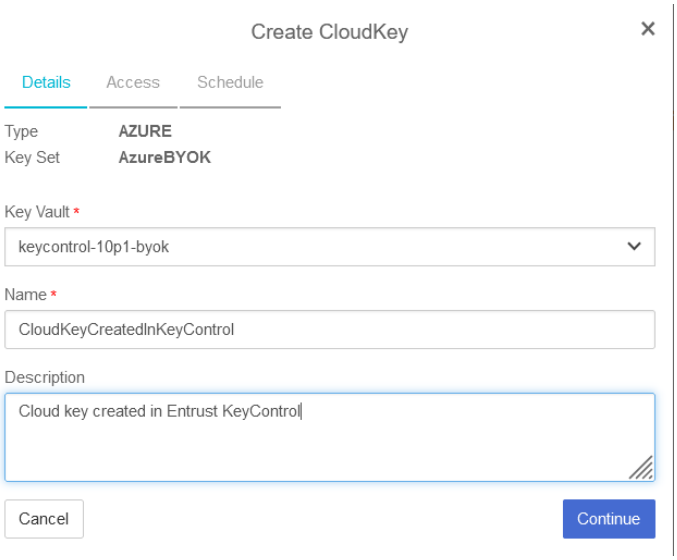
   For example:



6. Select **Actions** > **Create CloudKey**.

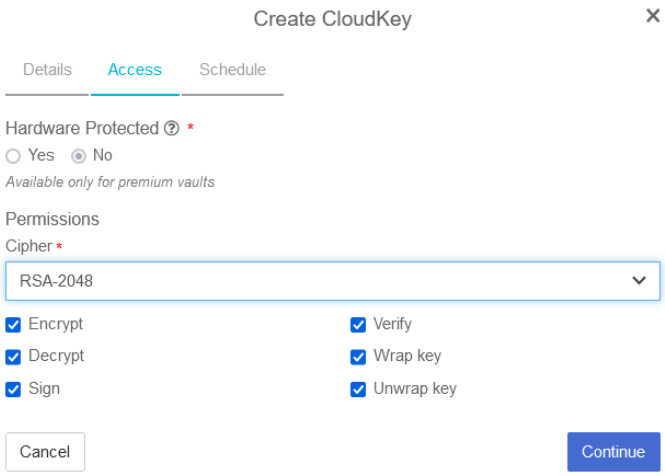
   The **Create CloudKey** dialog appears.

7. In the **Key Vault** menu, select the Azure key vault created in [test-integration:::create-azure-keyvault].
8. In the **Details** tab, enter the **Name** and **Description**. For example:



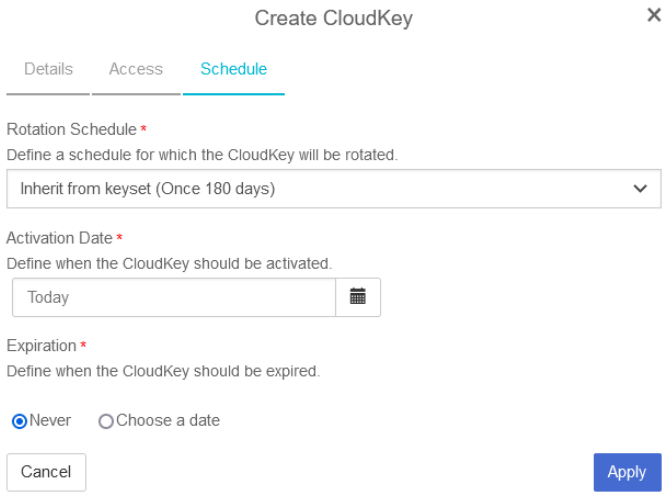
9. Select **Continue**.
10. In the **Access** tab, select the required **Cipher**.

    For example:



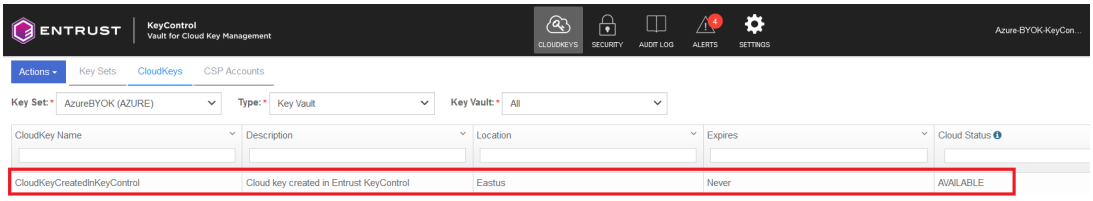
11. Select **Continue**.
12. In the **Schedule** tab, select the **Rotation Schedule**, **Activation Date**, and **Expiration**.

    For example:

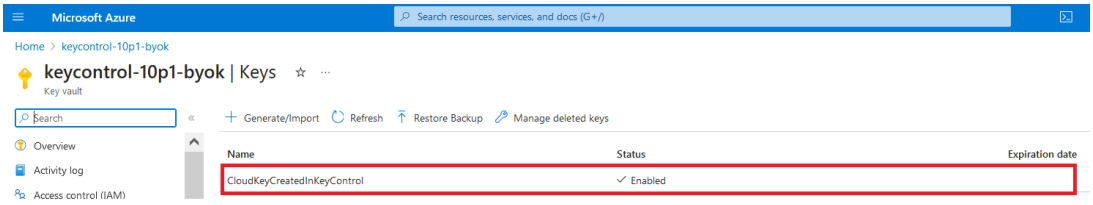


13. Select **Apply**.

    The cloud key is created.



14. Verify the cloud key created in Entrust KeyControl is available in Azure key vault.



For additional information, see Creating a CloudKey.

5.3. Create a cloud key in Azure key vault

The following steps create a cloud key in Azure key vault and import it into Entrust KeyControl.

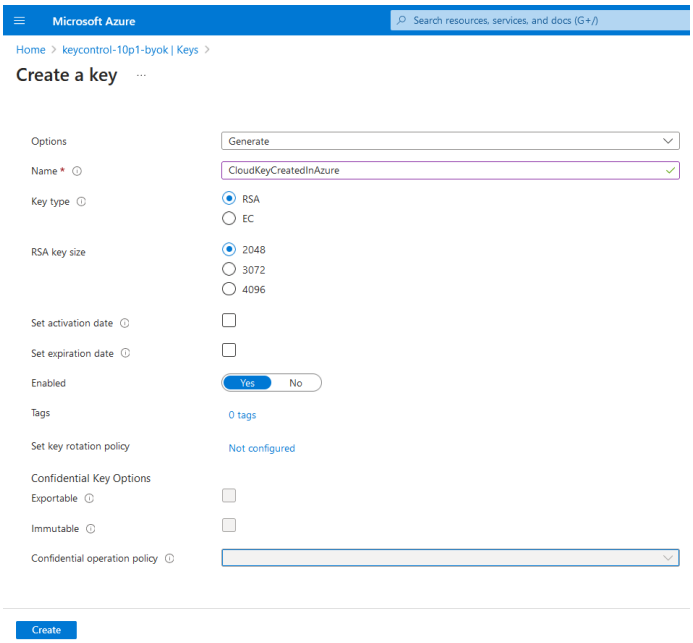
To create a cloud key in Azure Key Vault:

1. Navigate to **Home** > **Key vaults** > **<Key_vault_name>** > **Keys** > **Generate/Import**.

   The **Create a key** dialog appears.

2. Enter the **Name** and the required key properties.

   For example:

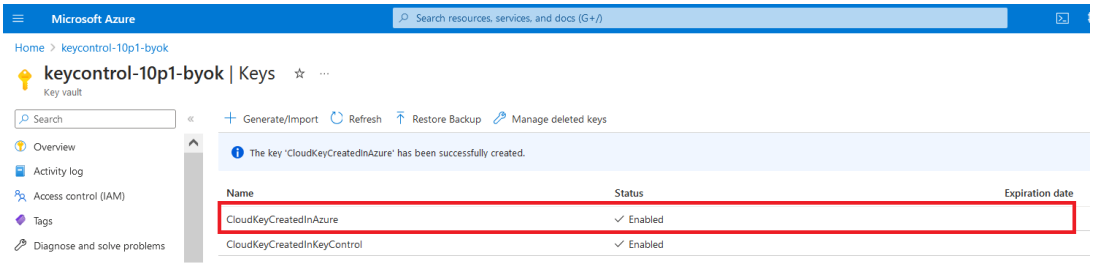


3. Select **Create**.

   The cloud key is created.

4. Verify the newly created key.

   For example:



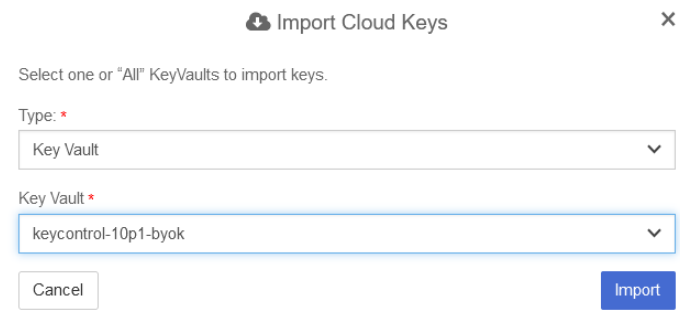
To import the cloud key created in Azure into Entrust KeyControl:

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **Key Sets** tab.
4. Select the key set created in Create a key set in Entrust KeyControl.
5. Select **Actions** > **Import CloudKey**.

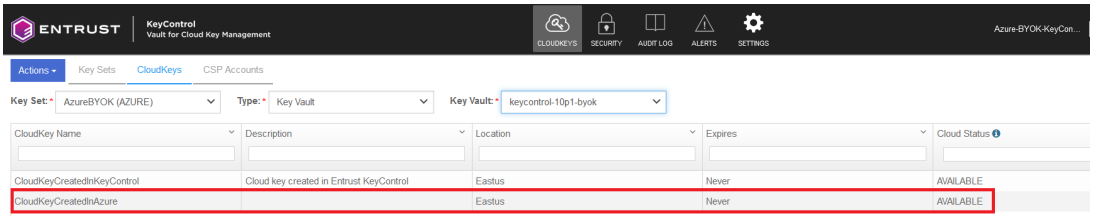
   The **Import Cloud Keys** dialog appears.

6. In the **Type** menu, select **Key Vault**.
7. In the **Key Vault** menu, select the Azure key vault created in [test-integration:::create-azure-keyvault].

   For example:



8. Select **Import**.
9. Verify the cloud key created in Azure key vault is available in Entrust KeyControl.



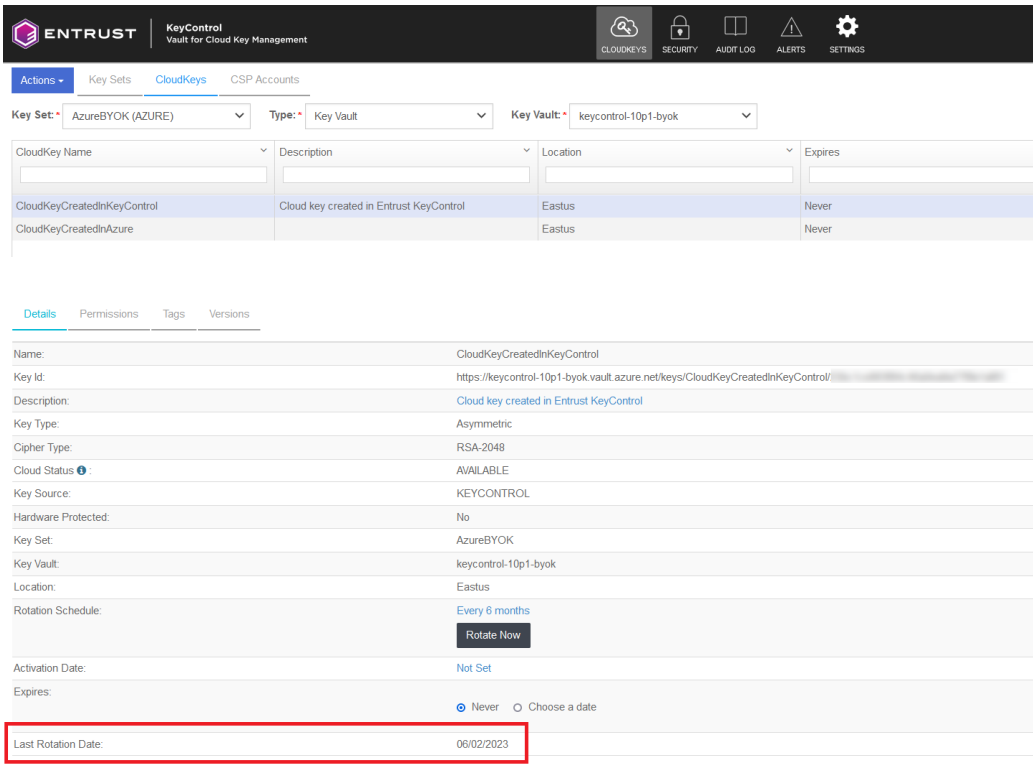
5.4. Rotate a cloud key in Entrust KeyControl

To rotate a cloud key in Entrust KeyControl:

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CloudKeys** tab.
4. Select the key to rotate. Then, scroll down until you see the **Rotate Now** control.
5. Select **Rotate Now**.

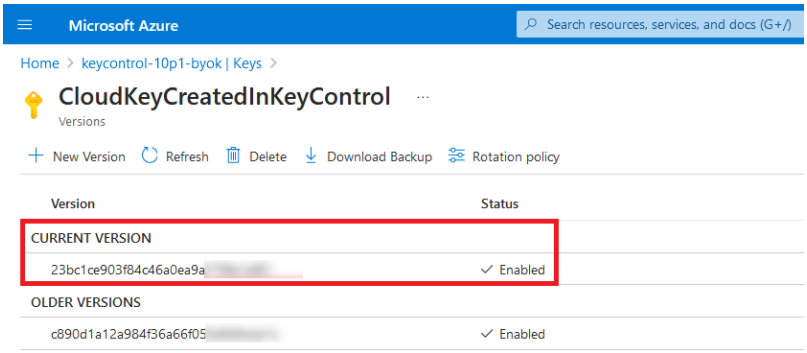
   The key has been rotated.

   For example:



6. In Azure, navigate to **Home** > **Key vaults** > **<Key_vault_name>** > **Keys**.
7. Select the key you want to rotate.
8. Verify that the key has been rotated.

   For example:



5.5. Remove a cloud key in Entrust KeyControl

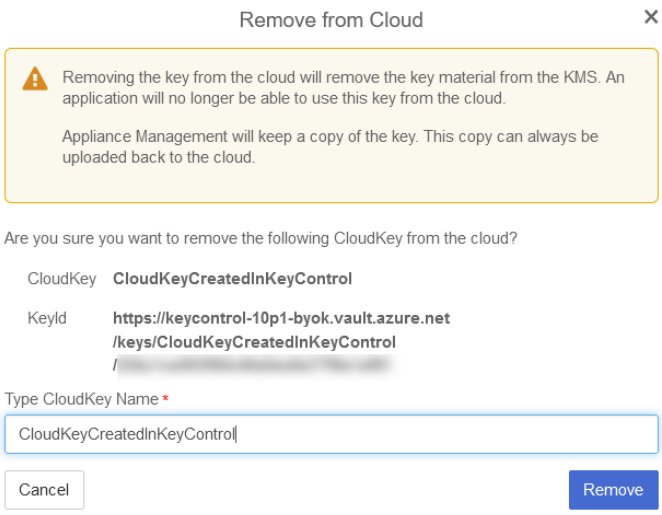
A removed cloud key in Entrust KeyControl will no longer be available for use in Azure. However, Entrust KeyControl will keep a copy of the removed cloud key, which could be reloaded to Azure for use.

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CloudKeys** tab.
4. Select the key to the removed.
5. Select **Actions** > **Remove from Cloud**.

   The **Remove from Cloud** dialog appears.

6. Type the name of the cloud key in **Type CloudKey Name**.

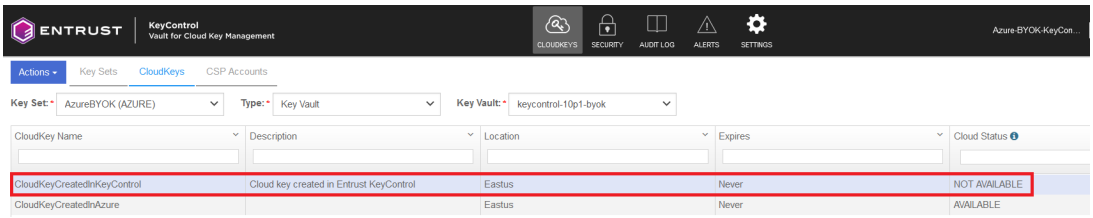
   For example:



7. Select **Remove**.

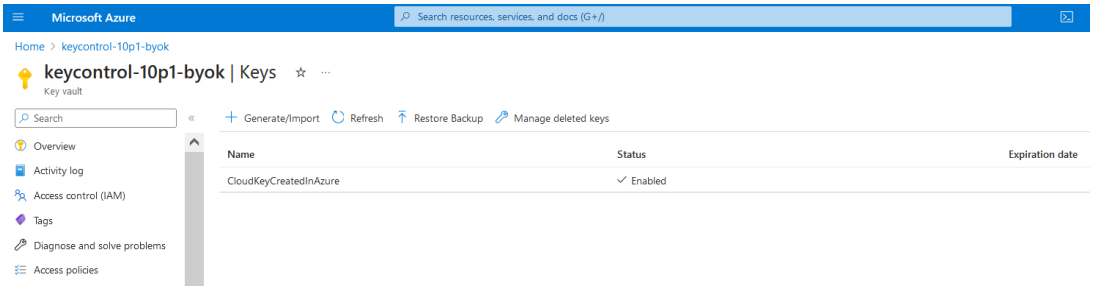
8. Verify the status change in Entrust KeyControl.

   For example:



9. Verify the key is gone from Azure.

   For example:



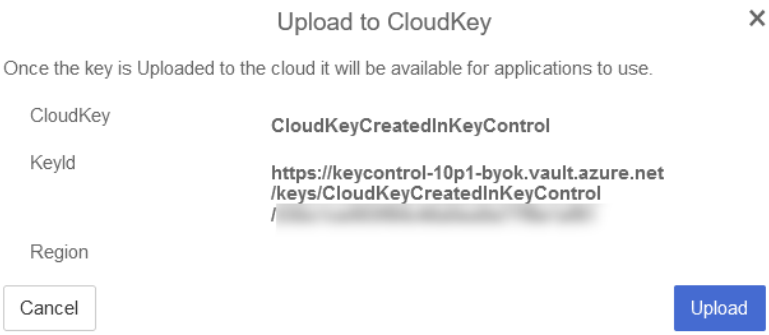
For additional information, see Removing a CloudKey from the Cloud.

5.6. Upload a removed cloud key to Azure in Entrust KeyControl

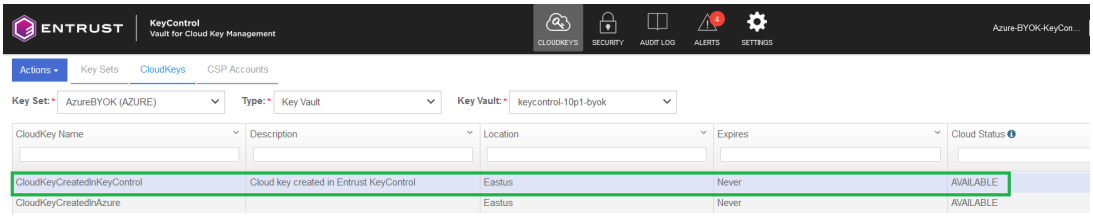
To upload a removed cloud key to Azure in Entrust KeyControl:

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CloudKeys** tab.
4. Select the key to be uploaded.
5. Select **Actions** > **Upload to Cloud**.

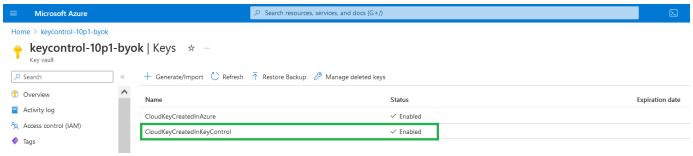
   The **Remove from Cloud** dialog appears. For example:



6. Select **Upload**.
7. Verify the status change in Entrust KeyControl. For example:



8. Verify the key is now available in Azure. For example:



5.7. Delete a cloud key in Entrust KeyControl

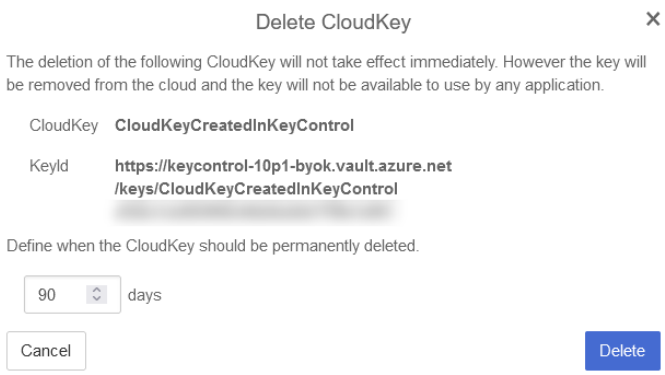
The deletion of a cloud key does not take effect immediately. However, after a user defined interval, the key will be permanently removed.

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CloudKeys** tab.
4. Select the key to deleted.
5. Select **Actions** > **Delete CloudKey**.

   The **Delete CloudKey** dialog appears.

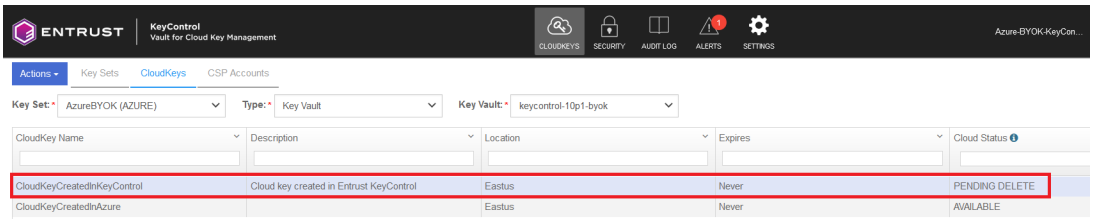
6. Select a time in **Define when the CloudKey should be permanently deleted**.

   For example:

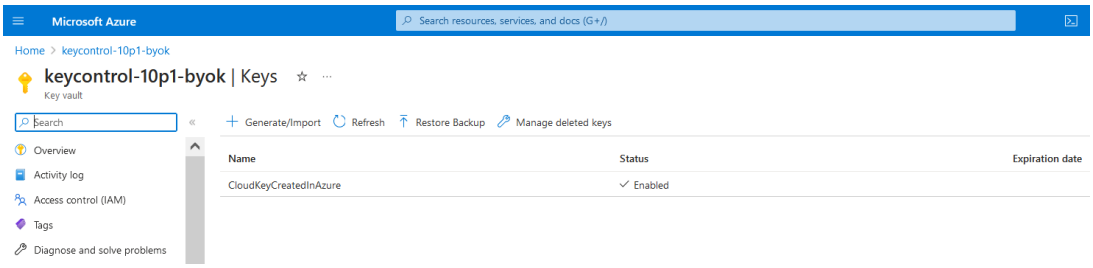


7. Select **Delete**.
8. Verify the status change in Entrust KeyControl.

   For example:



9. Verify the key is gone from Azure. For example:



For additional information, see Deleting a CloudKey.

5.8. Cancel a cloud key deletion in Entrust KeyControl

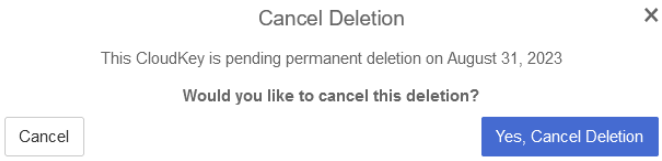
The deletion of a key can be canceled while the time in the **Define when the CloudKey should be permanently deleted** setting has not expired.

1. Sign in to the Entrust KeyControl Vault URL bookmark from [test-integration:::create-keycontrol-vault].
2. Select the **CLOUDKEYS** icon on the toolbar.
3. Select the **CloudKeys** tab.
4. Select the key deletion to be canceled.

5. Select **Actions** > **Cancel Deletion**.

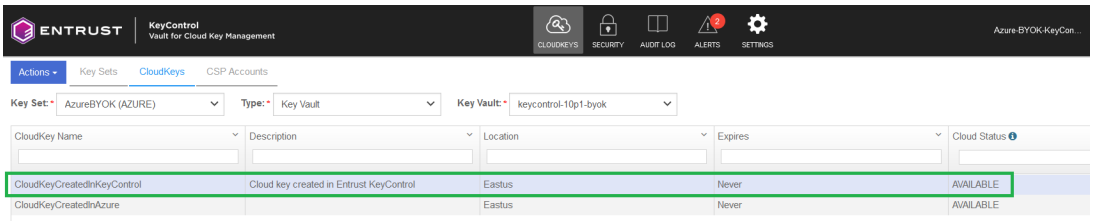
   The **Cancel Deletion** dialog appears.

   For example:

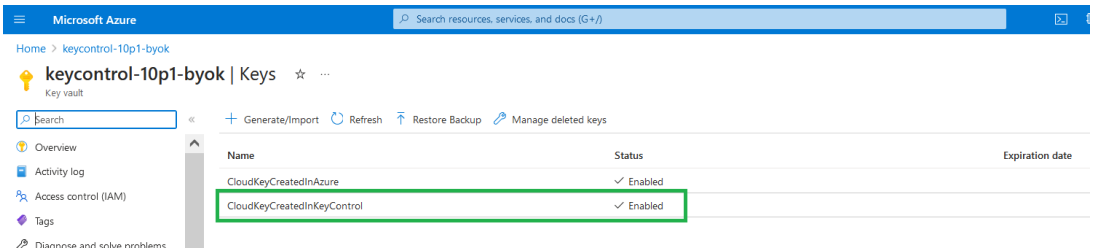


6. Select **Yes, Cancel Deletion**.
7. Verify the status change in Entrust KeyControl.

   For example:



8. Verify the key is now available in Azure. For example:



For additional information, see Canceling a CloudKey Deletion.

Chapter 6. Additional resources and related products

6.1. nShield HSMs

6.2. nShield as a Service

6.3. KeyControl BYOK

6.4. Entrust products

6.5. nShield product documentation