Configure Azure Key Vault

Contents:

For authentication purposes, the Trifacta® platform must be integrated with an Azure Key Vault keystore.

- For more information, see *https://azure.microsoft.com/en-us/services/key-vault/*.

Please complete the following sections to create and configure your Azure Key Vault.

Create a Key Vault resource in Azure

Please complete the following steps in the Azure portal to create a Key Vault and to associate it with the Trifacta registered application.

> **NOTE:** A Key Vault is required for use with the Trifacta platform.

Create Key Vault in Azure

Steps:

1. Log into the Azure portal.
2. Goto: *https://portal.azure.com/#create/Microsoft.KeyVault*
3. Complete the form for creating a new Key Vault resource:
   a. Name: Provide a reasonable name for the resource. Example:

      ```
      <clusterName>-<applicationName>-<group/organizationName>
      ```

      Or, you can use `trifacta`.
   b. Location: Pick the location used by the HDI cluster.
   c. For other fields, add appropriate information based on your enterprise's preferences.
4. To create the resource, click **Create**.

> **NOTE:** Retain the DNS Name value for later use.

Enable Key Vault access for the Trifacta platform

Steps:

In the Azure portal, you must assign access policies for application principal of the Trifacta registered application t o access the Key Vault.

Steps:

1. In the Azure portal, select the Key Vault you created. Then, select **Access Policies**.
2. In the Access Policies window, select the Trifacta registered application.
3. Click **Add Access Policy**.
4. Select the following secret permissions (at a minimum):
    a. Get
    b. Set
    c. Delete
5. Select the Trifacta application principal.
6. Assign the policy you just created to that principal.

Configure Key Vault for WASB

Create WASB access token

If you are enabling access to WASB, you must create this token within the Azure Portal.

For more information, see *https://docs.microsoft.com/en-us/rest/api/storageservices/delegating-access-with-a-shared-access-signature*.

You must specify the storage protocol (`wasbs`) used by the Trifacta platform.

Configure Key Vault key and secret for WASB

In the Key Vault, you can create key and secret pairs for use.

Base Storage Layer	Description
ADLS	The Trifacta platform creates its own key-secret combinations in the Key Vault. No additional configuration is required. Please skip this section and populate the Key Vault URL into the Trifacta platform.
WASB	For WASB, you must create key and secret values that match other values in your Azure configuration. Instructions are below.

WASB: To enable access to the Key Vault, you must specify your key and secret values as follows:

Item	Applicable Configuration
key	The value of the key must be specified as the `sasTokenId` in the Trifacta platform.
secret	The value of the secret should match the shared access signature for your storage. This value is specified as `sasToken` in the T rifacta platform.

Acquire shared access signature value:

In the Azure portal, please do the following:

1. Open your storage account.
2. Select **Shared Access Signature** .
3. Generate or view existing signatures.
4. For a new or existing signature, copy the SAS token value. Omit the leading question mark (?).
5. Paste this value into a text file for safekeeping.

Create a custom key:

To create a custom key and secret pair for WASB use by the Trifacta platform, please complete the following steps:

1. On an existing or newly created Azure Key Vault resource, click **Secrets**.
2. At the top of the menu, click **Generate/Import**.
3. In the Create a secret menu:
    a. Select **Manual** for upload options.
    b. Chose an appropriate name for the key.

    > **NOTE:** Please retain the name of the key for later use, when it is applied through the Trifact a platform as the `sasTokenId` value. Instructions are provided later.

    c. Paste the SAS token value for the key into the secret field.
    d. Click **Create**.

Configure Key Vault Location

For ADLS or WASB, the location of the Azure Key Vault must be specified for the Trifacta platform. The location can be found in the properties section of the Key Vault resource in the Azure portal.

Steps:

1. Log in to the Azure portal.
2. Select the Key Vault resource.
3. Click **Properties**.
4. Locate the DNS Name field. Copy the field value.

This value is the location for the Key Vault. It must be applied in the Trifacta platform.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Specify the URL in the following parameter:

```
"azure.keyVaultURL": "<your key value URL>",
```

Apply SAS token identifier for WASB

If you are using WASB as your base storage layer, you must apply the SAS token value into the configuration of the Trifacta platform.

Steps:

1. You can apply this change through the *Admin Settings Page* (recommended) or `trifacta-conf.json`. For more information, see *Platform Configuration Methods*.
2. Paste the value of the SAS Token for the key you created in the Key Vault as the following value:

```
"azure.wasb.defaultStore.sasTokenId": "<your Sas Token Id>",
```

3. Save your changes.

Configure Secure Token Service

Access to the Key Vault requires use of the secure token service (STS) from the Trifacta platform. To use STS with Azure, the following properties must be specified.

> **NOTE:** Except in rare cases, the other properties for secure token service do not need to be modified.

You can apply this change through the *Admin Settings Page* (recommended) or

`trifacta-conf.json`

. For more information, see *Platform Configuration Methods*.

Property	Description
" secure-token-service .autorestart"	Set this value to `true` to enable auto-restarting of the secure token service.
"secure-token-service.port"	Set this value to `8090`.
"com.trifacta.services.secure_token_service.refresh_token_encryption_key"	Enter a base64 string to serve as your encryption key for the refresh token of the secure token service. A default encryption key is inserted for you. **NOTE:** If a valid base64 string value is not provided here, the platform fails to start.
"secure-token-service.userIdHashingPepper"	Enter a base64 string.