



## AZURE KEY VAULT BYOK

Integration Guide

**Applicable Devices:**

*KMES Series 3*



THIS DOCUMENT CONTAINS CONFIDENTIAL INFORMATION PROPRIETARY TO FUTUREX, LP. ANY UNAUTHORIZED USE, DISCLOSURE, OR DUPLICATION OF THIS DOCUMENT OR ANY OF ITS CONTENTS IS EXPRESSLY PROHIBITED.

## TABLE OF CONTENTS

[1] OVERVIEW OF THE AZURE KEY VAULT BYOK / KMES SERIES 3 INTEGRATION .....	3
[1.1] ABOUT AZURE KEY VAULT .....	3
[1.2] WHAT IS BYOK? .....	3
[1.3] KEY BENEFITS OF THE INTEGRATION .....	3
[2] CREATE CREDENTIALS FOR COMMUNICATION BETWEEN THE KMES AND AZURE .....	5
[2.1] CREATE AN APP REGISTRATION .....	5
[3] CREATE AN AZURE KEY VAULT .....	7
[4] CONFIGURATION ON THE KMES SERIES 3 .....	8
[4.1] CREATE A CLOUD CREDENTIAL .....	8
[4.2] CREATE A NEW KEY GROUP .....	8
[5] KEY OPERATIONS .....	10
[5.1] CREATING A KEY ON THE KMES .....	10
[5.2] PUSHING KEY MATERIAL TO AZURE .....	10
[5.3] ROTATING KEY MATERIAL ON AZURE .....	11
[5.4] DELETING KEY MATERIAL FROM AZURE .....	11
[6] LOGGING .....	12
[6.1] TRACKING THE PROGRESS/STATUS OF JOBS .....	12
[6.2] VIEWING AZURE SERVICE LOGS .....	12
[6.3] EXPORTING AZURE SERVICE LOGS .....	13
APPENDIX A: XCEPTIONAL SUPPORT .....	14

## [1] OVERVIEW OF THE AZURE KEY VAULT BYOK / KMES SERIES 3 INTEGRATION

### [1.1] ABOUT AZURE KEY VAULT

From Microsoft's documentation website: "Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance."

For more general information about Azure Key Vault, please refer to the following article on Microsoft's documentation website: <https://docs.microsoft.com/en-us/azure/key-vault/general/overview>

### [1.2] WHAT IS BYOK?

Azure Key Vault's BYOK (Bring Your Own Key) feature allows importing existing asymmetric keys into a Key Vault. For this integration, this means being able to create asymmetric HSM Protected keys on a KMES Series 3 device, and then pushing those keys to an Azure Key Vault via the KMES application interface.

Keys that are pushed to a Key Vault can be used with other services inside Azure, such as the following:

- Azure Disk Encryption
- The always encrypted and Transparent Data Encryption functionality in SQL server and Azure SQL Database
- Azure App Service

Azure Key Vault also has it's own API that customers can use with their own applications to access and use keys stored in Azure Key Vault.

For this integration, keys will be created and stored on the KMES Series 3, synchronized to an Azure Key Vault, and then subsequently managed via the KMES application interface.

### [1.3] KEY BENEFITS OF THE INTEGRATION

The Azure Key Vault BYOK / KMES Series 3 integration provides several benefits:

- **Key provenance:** You are the sole owner of your keys, so you have the ability to control the location and distribution of them.
- **Added assurance:** Keys that are created on the KMES and imported into Azure never leave the HSM boundary. Because, even once in Azure, the keys are stored on hardware security modules on the backend.
- **Centralized key management:** You can manage your keys and access policies from a single location and user interface, whether the data they protect resides in the cloud or on your premises.

- **Audit compliance:** Many audits require you to escrow keys outside of the cloud provider. This is accomplished with this integration.

## [2] CREATE CREDENTIALS FOR COMMUNICATION BETWEEN THE KMES AND AZURE

Before the KMES Series 3 can push keys to an Azure Key Vault, credentials must be created inside of Azure and configured on the KMES. In Azure, these credentials will take the form of an **App Registration**. On the KMES, the credentials will take the form of a **Cloud Credential**.

### [2.1] CREATE AN APP REGISTRATION

1. Log in to the Azure Portal and navigate to [https://portal.azure.com/#blade/Microsoft\\_AAD\\_RegisteredApps/ApplicationsListBlade](https://portal.azure.com/#blade/Microsoft_AAD_RegisteredApps/ApplicationsListBlade).
2. Create a new App Registration.
3. Once it is created, click on **Certificates & Secrets** in the sidebar.
4. From there, scroll down to **Client Secrets** and add a new secret.
  - a. Note the value of the client secret that is generated. Save it in a plain text file with no additional characters included.

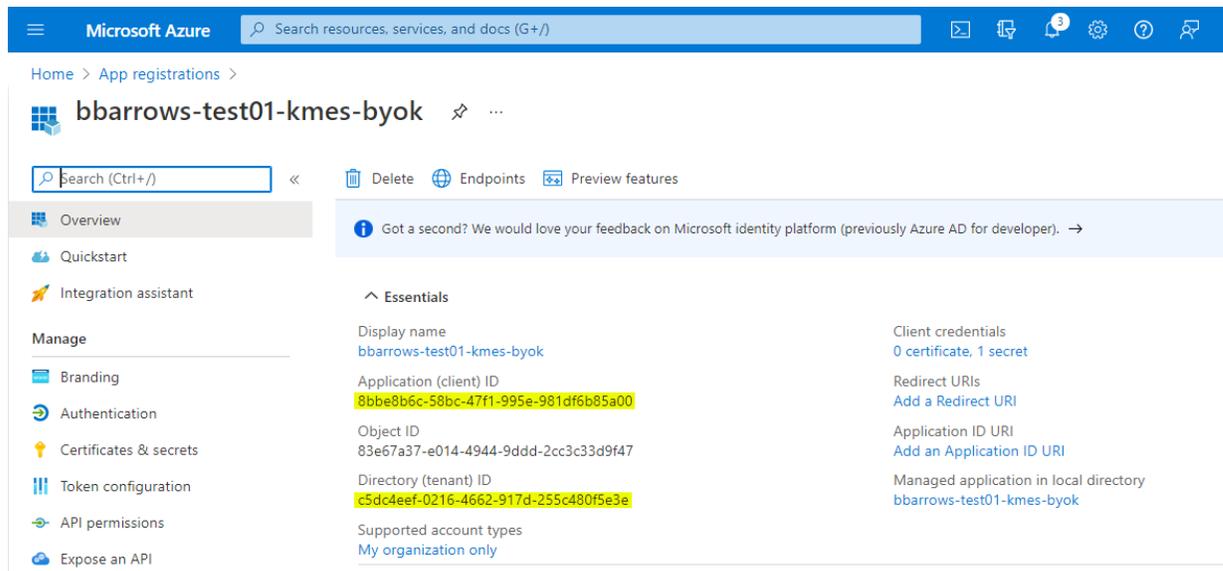
The screenshot shows the Azure portal interface for an application named 'bbarrows-test01-kmes-byok'. The 'Certificates & secrets' section is active, showing a table of client secrets. The table has columns for Description, Expires, Value, and Secret ID. One secret is listed with the description 'bbarrows-test01-secret', an expiration date of '3/21/2022', and a value 'kAa7Q~7vAzibvqaazwrhc3UWhYrVXnL...'. The value is highlighted in yellow. A notification in the top right corner states 'Update application credentials' and 'Successfully updated application bbarrows-test01-kmes-byok credentials'.

Description	Expires	Value	Secret ID
bbarrows-test01-secret	3/21/2022	kAa7Q~7vAzibvqaazwrhc3UWhYrVXnL...	c9d709bd-03f0-4dc8-8df5-4ab50a8e3...

**NOTE:** There is a time limit for being able to view the client secret value, so be sure to copy it immediately.

5. Navigate back to the main page for the App registration by clicking **Overview**.

6. Note the **Tenant ID** and **Client ID**.



**NOTE:** The **Tenant ID**, **Client ID**, and **Client Secret** values will be used when creating a Cloud Credential on the KMES in a later section.

### [3] CREATE AN AZURE KEY VAULT

**NOTE:** An existing Key Vault can be used rather than creating a new one, but it must be in the **Premium** service tier to include support for HSM backed keys.

1. Navigate to <https://portal.azure.com/#blade/HubsExtension/BrowseResource/resourceType/Microsoft.KeyVault%2Fvaults>.
2. Click the **Create** button. This will start the Key Vault creation wizard.
3. The pricing tier must be set to **Premium**. All other fields under the *Basics* tab can be set according to your specific use-case.
4. Under the *Access Policy* tab, either a Vault access policy or Azure role-based access control can be configured. Regardless of which is used, the App Registration created in the previous section needs to be granted the following key permissions:
  - Get (for general operations)
  - List (for general operations)
  - Create (for creating the ephemeral RSA KEK used in BYOK)
  - Import (for importing keys)
  - Delete (for deleting the ephemeral RSA KEK and for deleting your own key material)
  - Purge (Only required if the Key Vault supports soft-delete. The KMES will auto-detect this and not call Purge if it is not needed.)

**NOTE:** The permissions given to the App Registration will be the permissions that the Cloud Credential will have on the KMES.

5. Under the *Networking* tab, the connectivity method needs to be set to either **Public endpoint (all networks)** or **Public endpoint (selected networks)**.

**NOTE:** If setting the connectivity method to **Public endpoint (selected networks)**, you must whitelist in Azure the subnet that the KMES Series 3 will be connecting from.

6. Click the **Review + create** button at the bottom of the page to finish creating the Key Vault.

## [4] CONFIGURATION ON THE KMES SERIES 3

This section will cover creating a **Cloud Credential** and an **Azure Key Group** on the KMES Series 3.

### [4.1] CREATE A CLOUD CREDENTIAL

As mentioned in section 2, the three IDs (i.e., Tenant ID, Client ID, and Client Secret) that were gathered for the App Registration in Azure will now be used to create a Cloud Credential on the KMES.

1. Log in to the KMES Series 3 application interface with the default Admin identities.
2. Select *Identity Management* -> *Cloud Credentials* from the sidebar.
3. Click the **Add Cloud Credential** button.
4. Select **Azure App Registration** from the Service dropdown.
5. Give the Cloud Credential a name and enter the Client Secret, Tenant ID, and Client ID values from the App Registration step.

**NOTE:** For the Client Secret, there is also the option to import it as a plain text file.



6. Click **OK**.

### [4.2] CREATE A NEW KEY GROUP

The key group that will be created on the KMES in this section will essentially be the "key" in the Azure Key Vault. Azure Key Vault's holds keys, and each key has versions. In the KMES' parlance, this translates to the key group representing your key and the keys inside that group representing versions of that key. Furthermore, the KMES only shows keys for which key material was created on the KMES. Pulling private key data generated entirely on the Key Vault is not possible.

1. Select *Key Management* -> *Keys* from the sidebar.
2. Click the **Create** button in the top right-hand side of the menu.
3. Select an **Asymmetric HSM Protected** key group and click **OK**.

4. Select **Azure App Registration** from the service dropdown.
5. Name the key group (this will be the name of the key on the Key Vault), select the Cloud Credential created in section 4.1, and set the key type, rotation policy, and key usages.



6. Select the *Azure Properties* tab and type in the name of the Key Vault that was created in the previous section.



7. Click **OK**.

Keys can now be added to this group per the usual process. No synchronization of key material is done unless manually specified, as in the next step, or when automatic rotation is triggered on the schedule set for the group.

## [5] KEY OPERATIONS

This section will explain the process for creating a key on the KMES, pushing key material to Azure, rotating key material on Azure, and deleting key material from Azure.

**NOTE:** If a firewall is configured in your environment, ensure that the following endpoints are allowed from the KMES out to the internet:

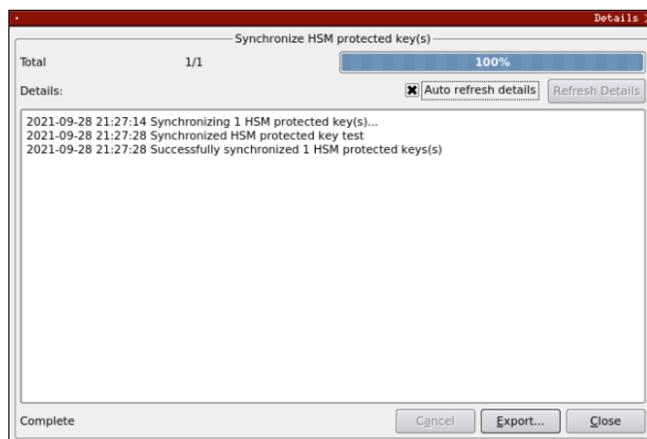
- login.microsoftonline.com:443
- management.azure.com:443
- <vault-name>.vault.azure.net:443 (**NOTE:** Replace <vault-name> with the actual name of your key vault in Azure.)

### [5.1] CREATING A KEY ON THE KMES

1. Log in to the KMES Series 3 application interface with the default Admin identities.
2. Navigate to *Key Management* -> *Keys*.
3. Select the key group that was created in the previous section, then click **Create** -> **Random** in the Keys section of the menu.
4. Specify any name for the key, then click **OK** to finish creating the key. It will now be listed in the Keys section of the menu.

### [5.2] PUSHING KEY MATERIAL TO AZURE

1. Make sure that the KMES is set to be the designated device to push key material (under **Administration** -> **Configuration** -> **HSM Protected Key Options**).
2. Right-click on the key that was just created and select **Cloud** -> **Synchronize**.
3. A job will be started to synchronize this key to the Azure Key Vault that was specified for the key group. Navigate to **Logging and Reporting** -> **Jobs** and double-click on the **Synchronize HSM protected key(s)** job that was just started. If the synchronization is successful, the following message will be shown:



- Once the job is finished, navigate back to the *Keys* view and select the key group for the key that was just synchronized. If the synchronization was successful, the key will have the Azure version assigned to it.



Name	Key Group	Certificate	Version	Storage	Key type	Algorithm	Check Digits
test	byok-integration-test		74d39584c0b14c7eac95a104f0727be1	Protected	None	RSA-2048	97F1

This new key, until another is pushed, will become the active key material for that key name in Azure Key Vault.

### [5.3] ROTATING KEY MATERIAL ON AZURE

When the time for rotation comes (if scheduled during key group creation), a new key is generated locally on the KMES, given a name based on the group's name, and synchronized to Azure. Since pushing a new key to Azure with the same name as the old key is the same thing as pushing a new version of that key, the new key material will become the active material.

Force rotation of the key material can also be done by right-clicking the key group and selecting **Cloud -> Force Rotation**. This will create a new key locally and automatically synchronize it to Azure.

Rotation will also set the "last rotated" timestamp on that particular group, if the rotation succeeds.

### [5.4] DELETING KEY MATERIAL FROM AZURE

Deleting the local key material and deleting the cloud key material are two different actions. Deleting the local key material does not delete the key material on the Key Vault, if it's been pushed.

To delete the cloud key material:

- Right-click the key group you want to delete the cloud key material for, and select **Cloud -> Delete on Cloud Service**.
- A job will be started to delete this key's material on its specified Key Vault. Check the Jobs tab under **Logging and Reporting -> Jobs** to view the status of the operation.

## [6] LOGGING

This section will explain how to track the progress/status of jobs related to Azure, view general Azure Service logs, and export Azure Service logs.

### [6.1] TRACKING THE PROGRESS/STATUS OF JOBS

It has already been mentioned in previous sections that the progress/status of jobs related to Azure can be view under **Logging and Reporting -> Jobs**. Events specific to this integration that would initiate a new job include the following:

- Pushing Key Material to Azure
- Rotating Key Material on Azure
- Deleting Key Material from Azure

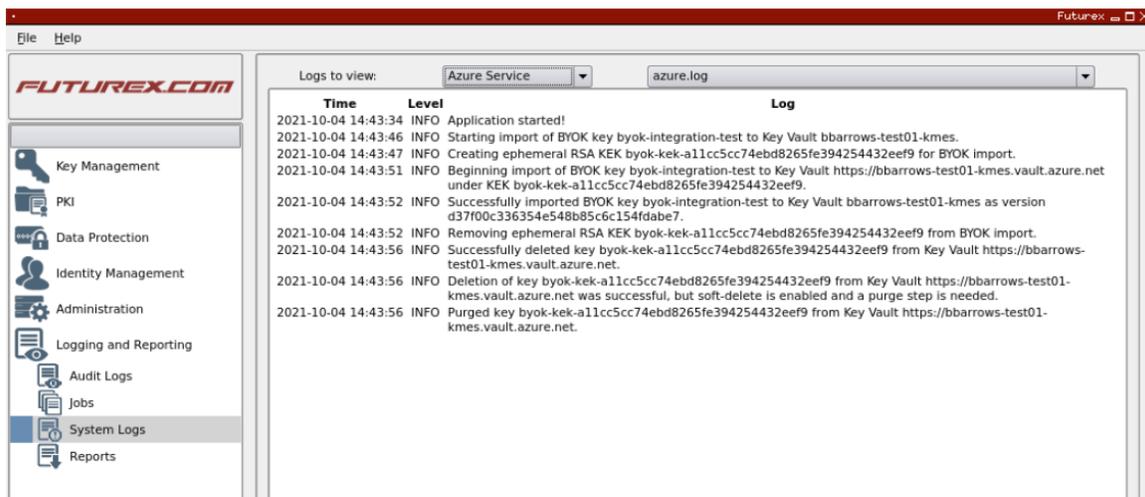
### [6.2] VIEWING AZURE SERVICE LOGS

All log events specific to Azure can be viewed by navigating to **Logging and Reporting -> System Logs** and then selecting **Azure Service** in the dropdown at the top of the window.

The following are the different categories of log events:

- **INFO** - General events such as creating, rotating, or deleting keys, which completed successfully.
- **WARN** - Events that completed successfully but with recoverable errors.
- **\*ERROR\*** - Events where a command was not able to complete successfully, resulting in a failure.

To provide an example of what you could expect to see in the Azure Service logs, see the image below showing all of the logs events that would occur for successfully pushing a key to an Azure Key Vault, and then remotely deleting that key.



### [6.3] EXPORTING AZURE SERVICE LOGS

Azure Service logs can be exported by completing the following steps:

1. Navigate to **Logging and Reporting** -> **System Logs**.
2. Select **Azure Service** in the dropdown.
3. Click the **Export File** button.
4. Specify a file name and the location where to save the file, then click **Open**. There should be a message stating that the logs were exported successfully.



## APPENDIX A: XCEPTIONAL SUPPORT



In today's high-paced environment, we know you are looking for timely and effective resolutions for your mission-critical needs. That is why our Xceptional Support Team does whatever it takes to ensure you have the best experience and support possible. Every time. Guaranteed.

- 24x7x365 mission critical support
- Level 1 to level 3 support
- Extremely knowledgeable subject matter experts

At Futurex, we strive to supply you with the latest data encryption innovations as well as our best-in-class support services. Our Xceptional Support Team goes above and beyond to meet your needs and provide you with exclusive services that you cannot find anywhere else in the industry.

- Technical Services
- Onsite Training
- Virtual Training
- Customized Consulting
- Customized Software Solutions
- Secure Key Generation, Printing, and Mailing
- Remote Key Injection
- Certificate Authority Services

Toll-Free: 1-800-251-5112

E-mail: [support@futurex.com](mailto:support@futurex.com)



#### ENGINEERING CAMPUS

864 Old Boerne Road  
Bulverde, Texas, USA 78163

Phone: +1 830-980-9782

+1 830-438-8782

E-mail: [info@futurex.com](mailto:info@futurex.com)

#### EXCEPTIONAL SUPPORT

24x7x365

Toll-Free: 1-800-251-5112

E-mail: [support@futurex.com](mailto:support@futurex.com)

#### SOLUTIONS ARCHITECT

E-mail: [solutions@futurex.com](mailto:solutions@futurex.com)