All information herein is either public information or is the property of and owned solely by Gemalto and/or its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Gemalto's information.

This document can be used for informational, non-commercial, internal and personal use only provided that:

- The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any publicly accessible network computer or broadcast in any media and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided “AS IS” without any warranty of any kind. Unless otherwise expressly agreed in writing, Gemalto makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Gemalto reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Gemalto hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Gemalto be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Gemalto does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Gemalto be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Gemalto products. Gemalto disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service or loss of privacy.

© 2017-18 Gemalto. All rights reserved. Gemalto and the Gemalto logo are trademarks and service marks of Gemalto and/or its subsidiaries and are registered in certain countries. All other trademarks and service marks, whether registered or not in specific countries, are the property of their respective owners.

**Document Part Number:** 007-013885-001, Rev. B

**Release Date:** June 2018
Preface .......................................................................................................................... 4
Scope ............................................................................................................................... 4
Document Conventions .................................................................................................. 4
   Command Syntax and Typeface Conventions .............................................................. 5
   Support Contacts ......................................................................................................... 6

1 Introduction .................................................................................................................. 7
Overview ......................................................................................................................... 7
Understanding BYOK ....................................................................................................... 7
   Supported HSM ............................................................................................................. 8
   PFX Utility ................................................................................................................... 8
Prerequisites ..................................................................................................................... 9
   Configuring SafeNet Luna Network HSM 7.1 ............................................................... 9
   Initialize the Partition SO and Crypto Officer Roles on a PW-Auth Partition .......... 9
   Initialize the Partition SO and Crypto Officer Roles on a PED-Auth Partition ......... 10
   Key Export Mode ........................................................................................................ 11
   Setting Key Export Mode on a Partition .................................................................... 11

2 Integrating Azure BYOK with SafeNet Luna HSMs .................................................... 12
Setting up SafeNet Luna HSMs with Azure BYOK for use with Azure RMS .............. 12
   Before You Begin ....................................................................................................... 12
   Generating and Exporting the Tenant Key ................................................................ 12
   Configuring Azure RMS to Use Customer Imported Key ....................................... 15
   Verifying Azure RMS Protection Using RMS Client ................................................. 23
Preface

This document is intended to guide administrators through the steps of supporting Azure BYOK integration with on-premises SafeNet Luna HSMs including installation, configuration, and integration.

Scope

This guide provides instructions for setting up a small test lab with SafeNet Luna HSMs for generating the tenant key (RSA-2048 key pair), exporting the tenant key pair from the HSM and importing the tenant keys into Azure Key Vault. Once the tenant key pair has been imported into the Azure Key Vault, it can be used by Azure Services (such as Azure RMS, Azure SQL, etc.) like any other Azure Key Vault key. This guide will also demonstrate how to use the imported key with Azure RMS for protecting Microsoft Office documents.

Document Conventions

This section provides information on the conventions used in this template.

Notes

Notes are used to alert you to important or helpful information. These elements use the following format:

NOTE: Take note. Contains important or helpful information.

Cautions

Cautions are used to alert you to important information that may help prevent unexpected results or data loss. These elements use the following format:

CAUTION: Exercise caution. Caution alerts contain important information that may help prevent unexpected results or data loss.

Warnings

Warnings are used to alert you to the potential for catastrophic data loss or personal injury. These elements use the following format:

WARNING: Be extremely careful and obey all safety and security measures. In this situation you might do something that could result in catastrophic data loss or personal injury.
## Command Syntax and Typeface Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
</table>
| **bold**   | The bold attribute is used to indicate the following:  
• Command-line commands and options (Type `dir /p`.)  
• Button names (Click `Save As`.)  
• Check box and radio button names (Select the `Print Duplex` check box.)  
• Window titles (On the `Protect Document` window, click `Yes`.)  
• Field names (**User Name**: Enter the name of the user.)  
• Menu names (On the `File` menu, click `Save`.) (Click `Menu > Go To > Folders`.)  
• User input (In the `Date` box, type `April 1`.) |
| **italic** | The italic attribute is used for emphasis or to indicate a related document. (See the `Installation Guide` for more information.) |
| Consolas   | Denotes syntax, prompts, and code examples. |
## Support Contacts

<table>
<thead>
<tr>
<th>Contact Method</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
<td>Gemalto</td>
</tr>
<tr>
<td></td>
<td>4690 Millennium Drive</td>
</tr>
<tr>
<td></td>
<td>Belcamp, Maryland 21017, USA</td>
</tr>
<tr>
<td><strong>Phone</strong></td>
<td></td>
</tr>
<tr>
<td>US</td>
<td>1-800-545-6608</td>
</tr>
<tr>
<td>International</td>
<td>1-410-931-7520</td>
</tr>
<tr>
<td></td>
<td>Existing customers with a Technical Support Customer Portal account can log in to manage incidents, get the latest software upgrades, and access the Gemalto Knowledge Base.</td>
</tr>
</tbody>
</table>
Introduction

Overview

Bring Your Own Key (BYOK) is a Microsoft Azure feature designed to support organizations that need greater control over their tenant keys and sensitive data in the Azure Cloud. Azure BYOK enables tenant keys to be generated in on-premises HSMs and then be imported into the Azure Key Vault, offering the following benefits:

- Full control over the key generation process and the conditions under which keys have been created
- Ability to archive a copy of generated keys outside of Microsoft Azure
- Enhanced control over key lifetimes and usage above and beyond what is supported for Azure generated keys

SafeNet Luna Hardware Security Module (HSMs) can be used to generate tenant keys on-premises and imported into Microsoft Azure Key Vault. These imported tenant keys can then be used like any other Azure Key Vault key, making the use of BYOK transparent to Azure services such as Azure RMS, Azure SQL, and other applications that support Azure Key Vault.

Understanding BYOK

This scenario involves creating your tenant key within the HSM, maintaining a master copy on-premises while importing the tenant key into Azure Key Vault. This BYOK method offers the following benefits:

1. Generate your tenant key on your premises, in line with your IT and security policies.
2. Securely transfer the tenant key from a HSM in your possession to HSMs that are owned and managed by Microsoft using Azure Key Vault.
3. Key remains protected in Azure Key Vault after transferring the tenant key to Microsoft Azure.
Supported HSM

SafeNet Luna HSM (Key Export) is available in all SafeNet HSM form factors (Network, PCIe, and USB HSMs). The Cloning variant does not allow Asymmetric keys (i.e. RSA, ECC) to be exported from the HSM in any condition and is the most commonly deployed HSM variant.

The Key Export HSM is designed to support use cases where the HSM will generate keying material that will then be used outside of the HSM, such as the Internet of Things (IoT) and secure manufacturing. With the Key Export variant, asymmetric keys are generated on the HSM and wrapped off the HSM using a symmetric key (keys never leave the HSM in the clear). Support for Azure BYOK requires the use of the Key Export variant as the Azure tenant key is an RSA-2048 key pair.

Please contact the Gemalto support team to obtain the correct variant of SafeNet Luna HSM.

NOTE: SafeNet Luna HSM v7.1 supports both Cloning and Key Export partitions within a single HSM.

PFX Utility

To simplify the key export and import process of tenant keys, Gemalto has created a PFX Utility. The PFX utility is available on the Gemalto Customer Support portal. The Document ID for the download is DOW0002365.
Prerequisite

Configuring SafeNet Luna Network HSM 7.1

SafeNet Luna Network HSM allows to create Per-Partition Security Officer (PPSO) partition. HSM Administrator is not Security Officer (SO) for PPSO partitions. The HSM SO/Administrator elects to create a partition as PPSO-type, which creates an empty structure that is handed to the new owner, who initializes the partition to create the Partition Security Officer (PSO) role or identifies for management functions. The PSO in turn creates the partition Crypto Officer (CO) to control client cryptographic operations on the partition.

Before you get started ensure the following:

- SafeNet Luna Network HSM appliance and a secure admin password.
- SafeNet Luna Network HSM, and a hostname, suitable for your network.
- SafeNet Luna Network HSM network parameters are set to work with your network.
- Initialize the HSM on the SafeNet Luna Network HSM appliance.
- Create and exchange certificates between the SafeNet Luna Network HSM and your Client system.
- Create a partition on the HSM that will be later used to export the key.
- Register the Client with the partition. And run the "vtl verify" command on the client system to display a partition from SafeNet Luna HSM. The general form of command is "C:\Program Files\SafeNet\LunaClient\vtl verify" for Windows and "/usr/safenet/lunaclient/bin/vtl verify" for Unix.
- Initialize the Partition as mentioned in steps below for Password/PED based respectively

Enabled Partition "Activation" and "Auto Activation" (Partition policy settings 22 and 23 (applies to SafeNet Luna Network HSM with Trusted Path Authentication [which is FIPS 140-2 level 3] only).

Initialize the Partition SO and Crypto Officer Roles on a PW-Auth Partition

These instructions assume a password-authenticated SafeNet Luna Network HSM that has been initialized, and an application partition has been created, which is capable of having its own Security Officer.

- Initialize the Partition SO role

Set the active slot to the created, uninitialized, application partition:

```
Type slot set -slot <slot number>
```

```
lunacm:> slot set -slot 0
Current Slot Id:    0     (Luna User Slot 7.0.0 (Password) Signing With Cloning Mode)
Command Result : No Error
```

Initialize the application partition, to create the partition’s Security Officer (SO).

```
Type partition init -label <partition label> -password <part_password>
```

```
lunacm:> par init -label <part_label> -password <part_password>
You are about to initialize the partition.
All partition objects will be destroyed.
Are you sure you wish to continue?
Type 'proceed' to continue, or 'quit' to quit now -> proceed
Command Result: No Error
```
• **Initialize the Crypto Officer role**
  a. The SO of the application partition can now assign the first operational role within the new partition. Type role login -name Partition SO.
     
     lunacm:> role login -name Partition SO
  
b. Type role init -name Crypto Officer.
     
     lunacm:> role init -name Crypto Officer
  
c. The application partition SO can create the Crypto Officer, but only the Crypto Officer can create the Crypto User. Therefore, the SO must log out to allow the Crypto Officer to log in.
     
     Type role logout.
     
     lunacm:> role logout

**Initialize the Partition SO and Crypto Officer Roles on a PED-Auth Partition**

These instructions assume a PED-authenticated SafeNet Luna Network HSM that has been initialized, and an application partition has been created, capable of having its own Security Officer.

Take the following steps to initialize the PSO and CO roles:

• **Initialize the Partition SO role**

  Set the active slot to the created, uninitialized, application partition.
  
  Type *slot set -slot <slot number>*
  
  lunacm:> slot set -slot 0
  
  Current Slot Id: 0 (Luna User Slot 7.0.0 (PED) Signing With Cloning Mode)
  
  Command Result : No Error

  Initialize the application partition, to create the partition's Security Officer (SO).
  
  Type *partition init -label <partition label>*
  
  lunacm:> par init -label <part_label>
  
  You are about to initialize the partition.
  All partition objects will be destroyed.
  Are you sure you wish to continue?
  Type 'proceed' to continue, or 'quit' to quit now -> proceed
  
  Please attend to the PED.
  
  Respond to SafeNet PED prompts...
  
  Command Result : No Error

• **Initialize the Crypto Officer role**

  The SO of the application partition can now assign the first operational role within the new partition.
  
  Type role login -name Partition SO.
  
  Type role init -name Crypto Officer.
  
  lunacm:> role init -name Crypto Officer
  
  Please attend to the PED.
  
  Respond to SafeNet PED prompts...
  
  Command Result: No Error
The application partition SO can create the Crypto Officer, but only the Crypto Officer can create the Crypto User. Therefore, the SO must log out to allow the Crypto Officer to log in. Type `role logout`.

Now, the Crypto Officer or an application using the CO’s challenge secret/password can perform cryptographic operations in the partition, as soon as the Crypto Officer logs in with `role login -name Crypto Officer`. However, the Crypto Officer can create, modify, and delete crypto objects within the partition, in addition to merely using existing crypto objects (sign/verify). You can also create a limited-capability role called Crypto User that can use the objects created by the Crypto Officer, but cannot modify them.

---

**NOTE:** The black Crypto Officer PED key/Crypto Officer password is valid for the initial login only. You must change the initial credential on the key using the command `role changepw` during the initial login session, or a subsequent login. Failing to change the credential will result in a CKR_PIN_EXPIRED error while performing role-dependent actions.

---

**Key Export Mode**

A partition in Key Export mode has the following capabilities and restrictions:

- Private keys cannot be cloned to other partitions nor to a SafeNet Luna Backup HSM.
- The partition cannot be part of an HA group (private keys will not be replicated).
- All keys/objects, including private keys, can be wrapped off the HSM (can be exported to a file encrypted with a wrapping key).

This mode is useful when generating key pairs for identity issuance, where transient key-pairs are generated, wrapped off, and embedded on a device. They are not used on the HSM, but generated and issued securely, and then deleted from the HSM.

**Setting Key Export Mode on a Partition**

The Partition SO can use the following procedure to set Key Export mode. Use `partition showpolicies` to see the current policy settings.

**To manually set Key Export mode on a partition:**

- Launch LunaCM and log in to the partition as Partition SO.
  
lunacm:>slot set slot <slotnum>
lunacm:>role login -name po

- Set partition policy 0: Allow private key cloning to 0 (OFF).
  
lunacm:>partition changepolicy -policy 0 -value 0

- Set partition policy 1: Allow private key wrapping to 1 (ON).
  
lunacm:>partition changepolicy -policy 1 -value 1
Integrating Azure BYOK with SafeNet Luna HSMs

Setting up SafeNet Luna HSMs with Azure BYOK for use with Azure RMS

HSMs provide strong physical protection of secure assets, including keys, and should be considered a best practice when leveraging the cloud.

Before You Begin

Read the Azure RMS document for more information on the BYOK feature.

To use the command-line examples in this guide:

a. Install the Luna Client and create NTLS with the HSM partition.
b. Obtain Microsoft Azure credentials and a subscription that supports Azure Information Protection.
c. Obtain the Azure Key Vault Premium service tier to support HSM-protected keys.
e. Download and install either OpenSSL or hsmpkcs12.exe to create the PFX file.

Refer to Microsoft Documentation for details on planning and implementing your Azure Information Protection tenant key.

Generating and Exporting the Tenant Key

After creating the NTLS connection with HSM, follow the steps below to generate the RSA key pair on HSM, and then export the private key and certificate to create the PFX file.

1. Generate the RSA key pair on SafeNet Luna HSM using the CMU utility provided with Luna Client. Provide the partition password when prompted.

   # cmu generatekeypair -labelprivate=RSAPrivate -labelpub=RSAPublic -keytype=RSA -modulusbits=2048 -sign=T -verify=T -extractable=T

   Select RSA Mechanism Type as [1] PKCS and Select public exponent as [3] 65537

   **NOTE:** Please refer to Luna Documentation for exact commands and keywords.
2. List the contents generated on HSM partition and note down the handle of public/private key. Provide the partition password when prompted.

```
# cmu list
```

3. Generate the self-signed certificate from the generated public/private key. Provide the partition password and certificate attributes when prompted.

```
# cmu selfsigncertificate -privatehandle=<handlePriv> -publichandle=<handlePub> -keyusage=digitalsignature,keyencipherment
```

![Image of cmu selfsigncertificate output]

**NOTE:** Self-signed certificate is used for test purpose. In production environment facing internet, create the certificate request and signed it by the Trusted Certificate Authority.

Please refer to Luna Documentation for exact commands and keywords.

4. List the contents generated on HSM partition and note down the handle of certificate. Provide the partition password when prompted.

```
# cmu list
```

After certificate and key generation on HSM partition now it's time to export the key/certificate in PFX form. You can use one of the following methods for the same:

**Method 1: Using hsmpkcs12.exe**

Download the hsmpkcs12 ([DOW0002365](#)) utility from portal. Extract the contents of the downloaded file on a system which have NTLS connection with SafeNet Luna HSM. The file should have the following contents:

- hsmpkcs12.exe -- application that performs import/export.
- hsmpkcs12Event.dll -- resource that defines custom log.
- hsmpkcs12Event.reg -- script that configures registry for custom log.

1. Copy the DLL file 'hsmpkcs12Event.dll' to directory 'C:\Windows\System32'.
2. Double-click the REG file 'hsmpkcs12Event.reg' and click Yes and then OK to install.
3. Edit the crystoki.ini file and add the following entries:

   [hsmpkcs12]
import.token.label=<partition_label>
export.token.label=<partition_label>

**NOTE:** Where `<partition_label>` is the name of the HSM partition.

4. Now export the certificate and key using the `hsmpkcs12.exe`. Open the command prompt and run the following command:

```
# hsmpkcs12.exe --export
```

5. It will list the available options. Select option [2] **List Certificates** and then hit **Enter**. Provide the partition password when prompted.


7. Enter the certificate handle you want to export and then hit **Enter**.

8. Enter the output file name and then hit **Enter**.

9. Provide PKCS#12 file password and confirm password. Hit **Enter** and remember the password as it will be required while importing the key in azure key vault.

10. Now type 0 and then press **Enter** to exit.

After completion we have the PFX file containing certificate and private key exported from the HSM.
Method 2: Using OpenSSL and ckdemo.exe

Use the below steps when to export the certificate and private in PFX format.

1. Export the certificate from SafeNet Luna HSM using CMU utility. Provide the partition password when prompted.
   
   ```
   # cmu export -handle=<certificate handle> -outputfile MyCert.pem
   ```

2. Run the CKDEMO utility provided with Luna Client to generate an AES key that will be used to wrap the RSA private key and note the handle of the generated AES key for subsequent use. Sequence of choices are provided below to generate an AES key using CKDEMO:
   
   ```
   # ckdemo
   # choose ( 1) Open Session
   # choose ( 3) Login  (login as a Crypto Officer option will be displayed when you enter the choice)
   # choose (45) Simple Generate Key
   # choose [16] AES
   
   Enter the key length as 32 and all attributes TRUE (1) when prompted.
   ```

3. Wrap the RSA private key using the generated AES key. Sequence of choices are provided below to wrap the key.
   
   ```
   # choose (60) Wrap key
   # choose [21] AES-CBC-PAD mechanism
   
   Provide the handle of wrapping key (AES) and key to wrapped (RSA) when prompted.
   
   The outcome would be the `wrapped.key` that is nothing but the RSA key wrapped by AES key.
   ```

4. Decrypt the `wrapped.key` file to get the binary version of RSA private key generated on HSM. Sequence of choices are provided below to decrypt the key.
   
   ```
   # choose (41) Decrypt file
   # choose [29] AES-CBC-PAD mechanism
   
   Enter `wrapped.key` when prompt for name of file to decrypt and then handle of AES key.
   
   The outcome would be the `DECRYPT.TXT` that is nothing but the binary version of the RSA private key.
   ```

5. Use the OpenSSL to convert the private key into PEM format from binary form.
   
   ```
   # openssl pkcs8 -topk8 -in DECRYPT.TXT -inform DER -out privkey.pem -outform PEM -nocrypt
   
   The outcome would be the `privkey.pem` that is nothing but the unencrypted RSA private key.
   ```

6. Create the PFX file from exported certificate and private key.
   
   ```
   # openssl pkcs12 -export -in MyCert.pem -inkey privkey.pem -out MyP12Bag.pfx
   
   Provide the password for the PFX file when prompted. Make sure the password is strong and hard to guess it will be used when key will be imported in Azure Key Vault.
   ```

Configuring Azure RMS to Use Customer Imported Key

After creating the PFX containing the Certificate and Private Key, transfer your own tenant key to azure key vault and configure the Azure RMS to use the azure key vault key for protecting the confidential documents.
Steps for transferring the key and configuring the Azure RMS are as follows:

1. Log on to the system as an administrative account. Open the Windows PowerShell and log on the Azure RMS.
   
   ```powershell
   # Login-AzureRmAccount
   
   Provide your Azure RMS global administrator credentials when prompted.
   
   2. After you have signed up for your Azure account, check your subscription for using the Azure RMS.
      
      ```powershell
      # Get-AzureRmSubscription
      
      If the subscription for using the Azure RMS is not bound to your credentials, set the credentials using the command:
      
      ```powershell
      # Set-AzureRmContext -SubscriptionId <subscription ID>
      
      Where <subscription ID> is the subscription to use the Azure RMS.
      
      3. Create the resource group that will be used to create the Azure Key Vault.
         
         ```powershell
         # New-AzureRmResourceGroup -Name 'SafeNetResourceGroup' -Location 'Central India'
         
         Change the resource group name and location as per your Azure RMS subscription. Azure RMS do not access the key vault if it is created in different location.
         
      4. Register the resource provider to Microsoft.KeyVault to allow Azure RMS to access the key vault services.
         
         ```powershell
         # Register-AzureRmResourceProvider -ProviderNamespace "Microsoft.KeyVault"
         
         5. Create the Azure Key Vault in the resource group you have created above.
            
            ```powershell
            # New-AzureRmKeyVault -VaultName 'MyKeyVaultHSM' -ResourceGroupName 'SafeNetResourceGroup' -Location 'Central India' -SKU 'Premium'
            
            Where SKU is required to use the Azure HSM. Vault Name will the name of your Key Vault.
            
            6. Run the following command to create encrypted password for PFX file that you have created by exporting the key and certificate from SafeNet.
               
               ```powershell
               # $securepfxpwd = ConvertTo-SecureString -String '<password>' -AsPlainText -Force
               
               Where <password> is the password you set for your PFX file.
               
               7. Import your key in to Azure Key Vault using the following command.
                  
                  ```powershell
                  # $key = Add-AzureKeyVaultKey -VaultName 'MyKeyVaultHSM' -Name 'SafeNetFirstHSKey' -KeyFilePath 'MyP12Bag.pfx' -KeyFilePassword $securepfxpwd -Destination 'HSM'
                  
                  It will add your key in to Azure Key Vault with the Name provided by you.
You can see the added key using the following command.

```
# $Key.key.kid
```

![Image showing the command output]

Note down the key vault key URL which will be used later.

9. Before Azure Information Protection can use the key, the Azure Rights Management service must be authorized to use the key in your organization's key vault. To do this, the Azure Key Vault administrator uses the Key Vault PowerShell cmdlet, `Set-AzureRmKeyVaultAccessPolicy` and grants permissions to the Azure Rights Management service principal, by using the GUID `00000012-0000-0000-0000-c000-000000000000`.

```
# Set-AzureRmKeyVaultAccessPolicy -VaultName 'MyKeyVaultHSM' -ResourceGroupName 'SafeNetResourceGroup' -ServicePrincipalName '00000012-0000-0000-0000-c000-000000000000' -PermissionsToKeys decrypt,encrypt,unwrapkey,wrapkey,verify,sign,get
```

10. You are now ready to configure Azure Information Protection to use this key as your organization's Azure Information Protection tenant key. Using Azure RMS cmdlets, first connect to the Azure Rights Management service and sign in.

```
# Connect-AadrmService
```

![Image showing the connection to Azure Rights Management service]

11. Then run the `Use-AadrmKeyVaultKey` cmdlet, specifying the key URL.

For example:

```
# Use-AadrmKeyVaultKey -KeyVaultKeyUrl "https://mykeyvaulthsm.vault.azure.net/keys/SafeNetFirstHSMKey/265b9567c0ca4a7b93378d636144e581"
```

![Image showing the use of the key vault key]

Ensure that you specify the key version, in addition to the key name when you run this command. You can use the Azure Key Vault cmd, `Get-AzureKeyVaultKey`, to get the version number of the current key. For example:

```
# Get-AzureKeyVaultKey -VaultName 'MyKeyVaultHSM' -KeyName 'SafeNetFirstHSMKey'
```

12. Activate the Rights Management service so that your organization can start using Azure Information Protection.

```
# Enable-Aadrm
```

![Image showing the activation of Rights Management service]
13. If the Azure Rights Management service is already activated, run `Set-AadrmKeyProperties` to tell Azure Rights Management to use this key as the active tenant key for your Azure Rights Management service.

```powershell
# Set-AadrmKeyProperties -Force -KeyIdentifier "8c92e1b6-7be3-473e-aaee-fed50315bbfc" -Active $True
```

14. Use the below command to check the Azure Key Vault key used by Azure RMS.

```powershell
# Get-AadrmKeys
```

Now your key is ready to be used to protect the documents using Azure RMS.

15. In a new browser window, sign in to the Azure portal as a global admin for your tenant.

16. On the hub menu, click **New** and then select **Security + Identity** from **Azure Marketplace** list. In the Security + Identify blade, select **Azure Information Protection** from the **Featured** list.
17. Select **Pin to dashboard** to create an Azure Information Protection tile on your dashboard, so that you can skip browsing to the service the next time you sign in to the portal. Click **Create**.

18. Click **Azure Information Protection** on dashboard.
19. Explore the main Azure Information Protection blade, which shows the Global Information Protection policy that is automatically created:

- Labels for classification: **Personal, Public, Internal, Confidential**, and **Highly Confidential**. Read the tooltip for each to understand how the labels are intended to be used. Note that Secret has two sub-labels: All-Employees and My-Group, which provides an example of how a classification can have subcategories.
20. Click **Add a new label**, Enter the **Label name** and **Description**. Click Protect and then select **Azure (cloud key)** on protection blade. Select **Set permissions** and then click **Add permissions**. Here you can select users and group and their permissions. After adding the users and their permissions click **OK**.

![Microsoft Azure](image1.png)

21. Verify that all your permissions are added and then, click **OK** on the **Protection** blade.

![Microsoft Azure](image2.png)
22. On the **Label** blade, change the settings for Header, Footer, Watermark, and other settings as required and click **Save**.

23. Click **Publish** to publish the changes you made. Click **Yes** when the confirmation message displays.
24. Under **Configure settings to display and apply on Information Protection end users**, set **On** for **All documents and emails must have a label (applied automatically or by users)** and select the default label. Click **Save** and then **Publish**.

You are now prepared to apply this label to your documents and protect them using the tenant key imported in Azure Key Vault.

**Verifying Azure RMS Protection Using RMS Client**

To verify the BYOK, first make sure that you have Office365 or your computer must have either Microsoft Office 2013 or Microsoft Office 2016.

Here we have used Office 2013, ensure that you have Office 2013 with Service Pack 1 with all latest updates.

1. Log on to **system** as an Administrator.
2. Download and install **Microsoft Azure Information Protection** client and click **I Agree**.

![Microsoft Azure Information Protection](Image)

3. Click **Close** when installation is finished.

4. Now verify the installation by opening a new Microsoft Word blank document. If you are prompted to enter your user name and password, enter the details for your Azure global administrator account. When the document loads, you should see two new things:

![Microsoft Word Document](Image)

On the **Home** tab, a new Protection group, with a button named **Protect**.

5. Click **Protect** and then **Help and Feedback**. The Microsoft Azure Information Protection dialog box displays to confirm your client status. It should display connected as and your user name. In addition, you should also see a recent time and date for the last connection and when the Information Protection policy was installed. Verify that your displayed user name is correct for your tenant.

A new bar under the ribbon; the Information Protection bar. It displays the title of **Sensitivity**, and the default label that we configured of **General**.

6. Log off from the system and logged in using Azure global administrator.

7. Open a new blank Microsoft Word document and click **Sign in** to login as an Azure Global Administrator.
8. Write something in the document. You can see the Azure Information Protection labels in the document. Apply the Label that you configured for Azure RMS protection which is **Protected-View Only**. Click the label to apply the protection we configured in Azure RMS.

9. **Save** the document at Public location where everyone can access it.

10. Log off from the system and logged in using the user whom you have granted **Viewer** permission.

11. After logging in, open the Public folder and double click to open the protected word file.

12. Click **OK** when prompted for connecting AD RMS service.

13. When prompted, you need to sign in as Azure RMS user that you have provided the Viewer permission.
14. When user logged in to see the document, it displays the document in view only.

![Microsoft Word Protected View Only](image1.png)

SafeNet Key Export HSM can be used for Microsoft Azure BYOK.

15. Except for allowed user, if you try to open the document using another user account, a confirmation message is displayed as given below:

![Microsoft Word Permission Message](image2.png)

Only users who have permissions are allowed to access the document as per their permissions. The content is protected by the tenant key in to the Azure Key Vault and the copy of the key is generated and secured on the SafeNet Key Export HSM.

This guide is used to setup small test lab and demonstrated the use of SafeNet HSM for securing the copy of your tenant key and configuring the Azure RMS to use the imported key for protecting the documents. For more details about Azure RMS Information Protection and production deployment refer to the Microsoft Azure RMS documentation.