

# Alibaba Cloud KMS

Integration Guide

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.

© 2018 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 Number:** 007-000275-001, Rev. A

**Release Date:** December 2018

# Contents

<b>Preface</b> .....	<b>4</b>
Scope .....	4
Document Conventions .....	4
Command Syntax and Typeface Conventions .....	5
Support Contacts .....	6
<b>1 Introduction</b> .....	<b>7</b>
Overview .....	7
3 <sup>rd</sup> Party Application Details .....	7
Supported Platforms .....	7
Prerequisites .....	8
Configure the SafeNet Luna HSM .....	8
Provision your HSM on Demand Service .....	9
Constraints on HSM on Demand Services .....	9
Set up Alibaba Cloud .....	10
Configure the client machine .....	10
<b>2 Integrating Alibaba Cloud KMS with SafeNet HSM</b> .....	<b>11</b>
Importing key to Alibaba Cloud KMS from SafeNet HSM .....	11
Downloading the Public Key and Import Token .....	11
Importing the Public Key into the HSM .....	13
Generating the AES key on the HSM .....	15
Wrapping the AES key with the Public Key .....	16
Uploading Key Material to KMS .....	17
Creating a Resource Access Management User .....	19
Using CMK for encryption/decryption .....	21

# Preface

This document guides administrators through the steps for integrating Alibaba Cloud KMS with SafeNet Luna HSM or an HSM on Demand (HSMoD) service. It provides the necessary information to configure and integrate SafeNet Luna HSM or HSMoD service with Alibaba Cloud KMS.

## Scope

This document outlines the steps to import the key material from SafeNet Luna HSM or an HSMoD service to Alibaba Cloud KMS.

## 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

Convention	Description
<b>bold</b>	The bold attribute is used to indicate the following: <ul style="list-style-type: none"><li>• Command-line commands and options (Type <b>dir /p</b>.)</li><li>• Button names (Click <b>Save As</b>.)</li><li>• Check box and radio button names (Select the <b>Print Duplex</b> check box.)</li><li>• Window titles (On the <b>Protect Document</b> window, click <b>Yes</b>.)</li><li>• Field names (<b>User Name</b>: Enter the name of the user.)</li><li>• Menu names (On the <b>File</b> menu, click <b>Save</b>.) (Click <b>Menu &gt; Go To &gt; Folders</b>.)</li><li>• User input (In the <b>Date</b> box, type <b>April 1</b>.)</li></ul>
<i>italic</i>	The italic attribute is used for emphasis or to indicate a related document. (See the <i>Installation Guide</i> for more information.)
Consolas	Denotes syntax, prompts, and code examples.

## Support Contacts

Contact Method	Contact Information	
Address	Gemalto 4690 Millennium Drive Belcamp, Maryland 21017, USA	
Phone	US	1-800-545-6608
	International	1-410-931-7520
Technical Support Customer Portal	<a href="https://supportportal.gemalto.com">https://supportportal.gemalto.com</a> 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.	

# Introduction

## Overview

---

Key Management Service (KMS) is a service available with Alibaba Cloud that allows you to create and manage encryption keys (master keys) used to encrypt data. KMS enables you to maintain control over who can use master keys and gain access to encrypted data.

Customer master keys (CMKs) are the basic resources of KMS. CMKs are composed of key IDs, basic metadata (such as key state) and key materials used to encrypt and decrypt data. KMS allows you to create a key from external key materials. You can generate and import your own key material to the CMK in KMS console.

This document describes how to import key material generated on SafeNet HSM into the Alibaba Cloud KMS.

The benefits of generating the keys with SafeNet HSM include:

- Secure generation, storage and protection of the encryption keys on FIPS 140-2 level 3 validated hardware\*.
- Full life cycle management of the keys.
- Take advantage of cloud services with confidence.

\*validation for HSMoD services in progress.

## 3<sup>rd</sup> Party Application Details

---

This integration guide uses the following third party applications:

- Alibaba Cloud Account with Key Management Service

## Supported Platforms

---

**SafeNet Luna HSM:** SafeNet Luna HSM appliances are purposefully designed to provide a balance of security, high performance, and usability that makes them an ideal choice for enterprise, financial, and government organizations. SafeNet Luna HSMs physically and logically secure cryptographic keys and accelerate cryptographic processing.

The SafeNet Luna HSM on premise offerings include the SafeNet Luna Network HSM, SafeNet PCIe HSM, and SafeNet Luna USB HSMs. SafeNet Luna HSMs are also available for access as an offering from cloud service providers such as IBM cloud HSM and AWS cloud HSM classic.

**SafeNet DPoD:** SafeNet Data Protection on Demand (DPoD) is a cloud-based platform that provides on-demand HSM and Key Management services through a simple graphical user interface. With DPoD, security is simple, cost effective and easy to manage because there is no hardware to buy, deploy and maintain. As an

Application Owner, you click and deploy services, generate usage reports and maintain just the services you need.



**NOTE:** Alibaba Cloud KMS requires a browser and internet connection. The operating system is only required for the commands used to generate and wrap the CMK. As a result, you can use any platform (Unix or Windows) that are supported by SafeNet Luna Client or the HSM on Demand service client to complete this integration.

## Prerequisites

Before beginning the integration, ensure you have completed the following:

### Configure the SafeNet Luna HSM

Set up and configure the SafeNet Luna HSM device for your system.

1. Ensure the HSM is setup, initialized, provisioned and ready for deployment. Refer to the *SafeNet Luna HSM Product Documentation* for help.
2. Create a partition on the HSM that will be later used by Alibaba Cloud KMS.
3. If using a SafeNet Luna Network HSM, register a client for the system and assign the client to the partition to create an NTLS connection. Initialize Crypto Officer and Crypto User roles for the registered partition.
4. Ensure that the partition is successfully registered and configured. The command to see the registered partition is:

```
# /usr/safenet/lunaclient/bin/lunacm
```

```
lunacm.exe (64-bit) v7.2.0-219. Copyright (c) 2018 SafeNet. All rights reserved.
```

Available HSMs:

```
Slot Id ->          0
Label ->           Alibaba_Cloud_KMS
Serial Number ->   1213475834492
Model ->           LunaSA 7.2.0
Firmware Version -> 7.2.0
Configuration ->   Luna User Partition With SO (PW) Signing With Cloning Mode
Slot Description -> Net Token Slot
```

```
Current Slot Id: 0
```



**NOTE:** Follow the *SafeNet Network Luna HSM Product Documentation* for detailed steps for creating NTLS connection, initializing the partition and initializing the user roles.



## Provision your HSM on Demand Service

This service provides your client machine with access to an HSM Application Partition for storing cryptographic objects used by your applications. Application partitions can be assigned to a single client, or multiple clients can be assigned to, and share, a single application partition.

To use the HSM on Demand service you need to provision your application partition, starting by initializing the following roles:

- **Security Officer (SO)** - responsible for setting the partition policies and for creating the Crypto Officer.
- **Crypto Officer (CO)** - responsible for creating, modifying and deleting crypto objects within the partition. The CO can use the crypto objects and create an optional, limited-capability role called Crypto User that can use the crypto objects but cannot modify them.
- **Crypto User (CU)** - optional role that can use crypto objects while performing cryptographic operations.



**NOTE:** Refer to the *SafeNet Data Protection on Demand Application Owner Quick Start Guide* for procedural information on configuring the HSM on Demand service and create a service client.

The HSM on Demand service client package is a zip file that contains system information needed to connect your client machine to an existing HSM on Demand service.

## Constraints on HSM on Demand Services

Please take the following limitations into consideration when integrating your application software with an HSM on Demand service:

### HSM on Demand Service in FIPS mode

HSMoD services operate in a FIPS and non-FIPS mode. If your organization requires non-FIPS algorithms for your operations, ensure you enable the **Allow non-FIPS approved algorithms** check box when configuring your HSM on Demand service. The FIPS mode is enabled by default.

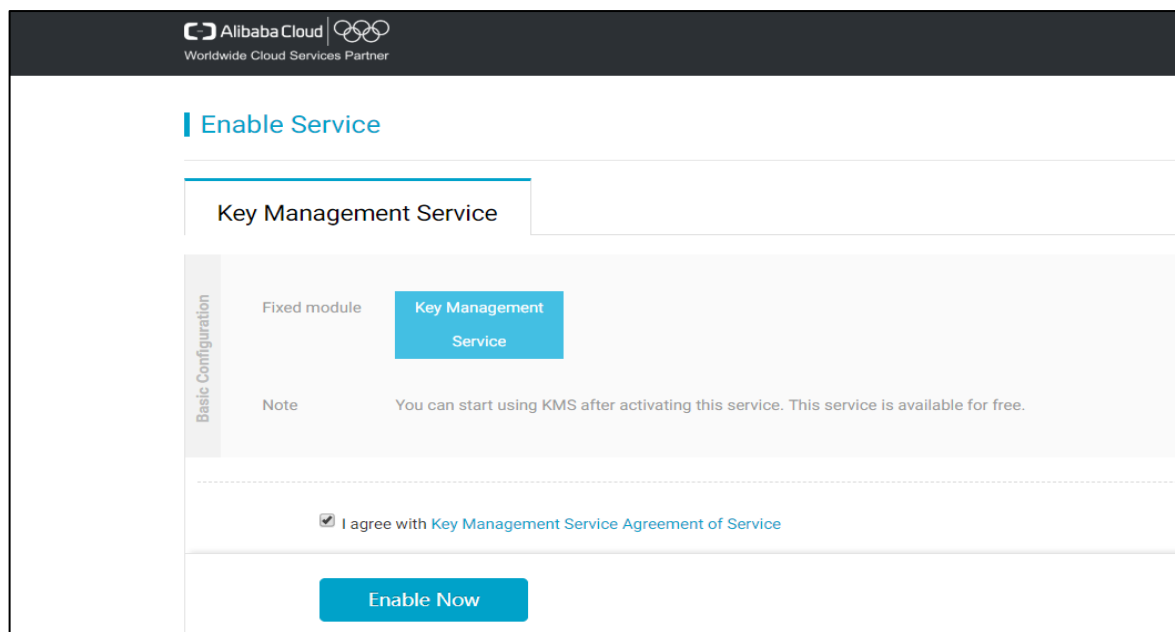
Refer to the *Mechanism List* in the *SDK Reference Guide* for more information about available FIPS and non-FIPS algorithms.

### Verify HSM on Demand <slot> value

LunaCM commands work on the current slot. If there is only one slot, then it is always the current slot. If you are completing an integration using HSMoD services, you need to verify which slot on the HSMoD service you send the commands to. If there is more than one slot, then use the **slot set** command to direct a command to a specified slot. You can use slot list to determine which slot numbers are in use by which HSMoD service.

## Set up Alibaba Cloud

Login to your **Alibaba Cloud** account and enable the **Key Management Service**. See the [Alibaba Cloud Key Management Service Documentation](#) for further information about enabling the KMS.



## Configure the client machine

You require some additional libraries for operating **Alibaba Cloud** over the command-line interface (CLI). Complete the following on the system where you will be managing and accessing the Alibaba Cloud KMS:

- Install Python 2.7.x and pip 7.x on the client machine.
- Run the following command to install the Alibaba Cloud CLI.  

```
# pip install aliyuncli
```
- Alibaba Cloud CLI requires the Alibaba Cloud product Software Development Kit (SDK). Run the following command to install the KMS SDK. See [Online installation of Alibaba Cloud CLI and SDK](#) for further information.  

```
# pip install aliyun-python-sdk-kms
```
- Download and install OpenSSL on the client machine. See the [OpenSSL Compilation and Installation Documentation](#) for further information.

# 2

## Integrating Alibaba Cloud KMS with SafeNet HSM

### Importing key to Alibaba Cloud KMS from SafeNet HSM

---

To import the key material from the SafeNet HSM to the Alibaba Cloud KMS, complete the following:

- Downloading the Public Key and Import Token
- Importing the Public Key into the HSM
- Generating the AES key on the HSM
- Wrapping the AES key with the Public Key
- Uploading Key Material to KMS
- Creating a Resource Access Management User
- Using CMK for encryption/decryption

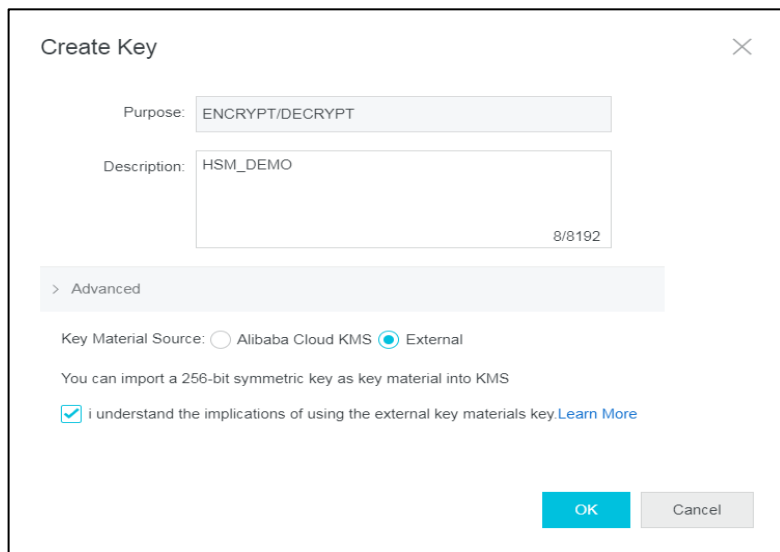
#### Downloading the Public Key and Import Token

You need to download a Public Wrapping Key and Import Token from the **Alibaba Cloud KMS**. You require these objects for wrapping the HSM generated key and importing the HSM generated key into the **Alibaba Cloud KMS**.

#### To download the public key and import token

1. Log in to the **Alibaba Cloud KMS** console.
2. Click **Create Key**.

- Specify the description in **Description** field. Click on **Advanced** and select the **External** radio button for the **Key Material Source**.



Create Key

Purpose: ENCRYPT/DECRYPT

Description: HSM\_DEMO 8/8192

> Advanced

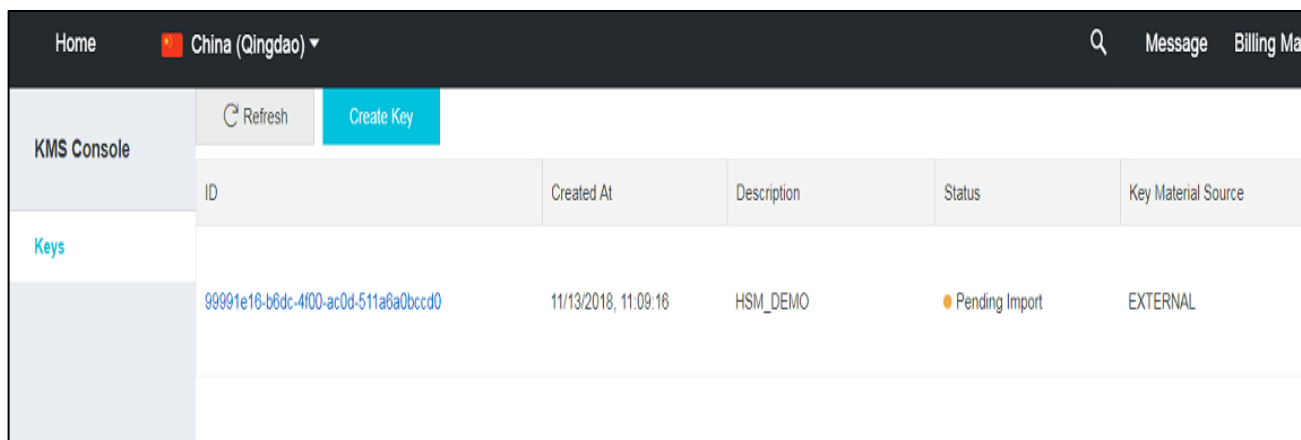
Key Material Source:  Alibaba Cloud KMS  External

You can import a 256-bit symmetric key as key material into KMS

I understand the implications of using the external key materials key. [Learn More](#)

OK Cancel

- Click **OK**. The console will show the generated key with status **Pending Import**.



ID	Created At	Description	Status	Key Material Source
99991e16-b6dc-4f00-ac0d-511a8a0bccd0	11/13/2018, 11:09:16	HSM_DEMO	Pending Import	EXTERNAL

- Click on the **ID** value of the key. The key details display.

6. Click on **Key Encryption Material** at the bottom of the page. Open the **Encryption Algorithm** drop-down menu and select the algorithm that you want to use. Click **Next**.

7. The Public Key and Import Token generate. Click **Download** on both the Public Key and Import Token and click **Close**.

The Public Key is a RSA 2048 wrapping key and the Import Token is bound to the Public Key used to encrypt key material. A single token can only be used to import the key material for the CMK specified at the time of generation.



**NOTE:** The Import Token is valid for 24 hours and can be used multiple times during this period. After the token expires, you must obtain a new import token and public encryption key.

## Importing the Public Key into the HSM

You must import the Public Key generated on the Alibaba Cloud KMS into the SafeNet HSM and then enable wrapping on the imported key. The wrapping operation occurs on the SafeNet HSM.

## To import the public key into the HSM

1. Import the public wrapping key in to the HSM using the **CMU** utility. The **CMU** utility is provided with the HSM client.

```
# ./cmu import -inputfile=pub_key.pem -pubkey=pub_key.pem -label " Alibaba Cloud Public Key "
```

Where `pub_key.pem` is the public key downloaded from KMS console.

Provide the HSM partition password when prompted.



**NOTE:** The public key is downloaded in `.txt` format. To convert it into `.pem`, specify the key in following format :

```
-----BEGIN RSA PUBLIC KEY-----
<content of the file downloaded>
-----END RSA PUBLIC KEY-----
and rename the file to pub_key.pem.
```

2. Run the **cmu list** command to ensure the key imported successfully. You will be prompted for the HSM partition password.

```
# ./cmu list
```

```
handle=3031091932          label= Alibaba Cloud Public Key
```

Copy the handle of the public key, you will require it for a command later.

3. Set the **wrap** key attribute of public key to true using the **ckdemo** utility below:

```
# ckdemo
```

It shows you the available options and prompt for your choice. Below are the numeric values:

### (1) Open Session

```
Enter your choice: 1
```

```
Status: Doing great, no errors (CKR_OK)
```

### (3) Login

```
Enter your choice: 3
```

```
Crypto Officer   [0]
```

```
Crypto User     [1]: 0
```

```
Enter PIN       : *****
```

```
Status: Doing great, no errors (CKR_OK)
```

### (25) Set attribute

```
Which object do you want to modify (0 to list available objects) : 3031091932
```

```
Edit template for set attribute operation.
```

```
(1) Add Attribute   (2) Remove Attribute   (0) Accept Template : 1
```

```
0 - CKA_CLASS          1 - CKA_TOKEN
2 - CKA_PRIVATE        3 - CKA_LABEL
4 - CKA_APPLICATION   5 - CKA_VALUE
```

```

6 - CKA_UNKNOWN          7 - CKA_CERTIFICATE_TYPE
8 - CKA_ISSUER           9 - CKA_SERIAL_NUMBER
10 - CKA_KEY_TYPE        11 - CKA_SUBJECT
12 - CKA_ID              13 - CKA_SENSITIVE
14 - CKA_ENCRYPT          15 - CKA_DECRYPT
16 - CKA_WRAP            17 - CKA_UNWRAP
18 - CKA_SIGN            19 - CKA_SIGN_RECOVER
20 - CKA_VERIFY          21 - CKA_VERIFY_RECOVER
22 - CKA_DERIVE          23 - CKA_START_DATE
24 - CKA_END_DATE        25 - CKA_MODULUS
26 - CKA_MODULUS_BITS    27 - CKA_PUBLIC_EXPONENT
28 - CKA_PRIVATE_EXPONENT 29 - CKA_PRIME_1
30 - CKA_PRIME_2         31 - CKA_EXPONENT_1
32 - CKA_EXPONENT_2     33 - CKA_COEFFICIENT
34 - CKA_PRIME           35 - CKA_SUBPRIME
36 - CKA_BASE            37 - CKA_VALUE_BITS
38 - CKA_VALUE_LEN       39 - CKA_LOCAL
40 - CKA_MODIFIABLE      41 - CKA_ECDSA_PARAMS
42 - CKA_EC_POINT        43 - CKA_EXTRACTABLE
44 - CKA_ALWAYS_SENSITIVE 45 - CKA_NEVER_EXTRACTABLE
46 - CKA_CCM_PRIVATE     47 - CKA_FINGERPRINT_SHA1
48 - CKA_OUID            49 - CKA_X9_31_GENERATED
50 - CKA_PRIME_BITS      51 - CKA_SUBPRIME_BITS
52 - CKA_USAGE_COUNT     53 - CKA_USAGE_LIMIT
54 - CKA_EKM_UID         55 - CKA_GENERIC_1
56 - CKA_GENERIC_2       57 - CKA_GENERIC_3
58 - CKA_FINGERPRINT_SHA256 59 - CKA_WARNING_THRESHOLD
60 - CKA_HW_FEATURE_TYPE

```

Select which one: **16**

Enter boolean value: **1**

CKA\_WRAP=01

(1) Add Attribute (2) Remove Attribute (0) Accept Template : 0

Status: Doing great, no errors (CKR\_OK)

## Generating the AES key on the HSM

Generate an AES key on the HSM to be wrapped by the **Alibaba Cloud KMS** wrapping key.

### To generate the AES key on the HSM

1. Generate an AES 256 key on the HSM partition using the **ckdemo** utility. The **ckdemo** utility is provided with the HSM client.

```
# ckdemo

(45) Simple Generate Key
Enter your choice: 45
Select type of key to generate
[1] DES          [2] DES2   [3] DES3          [5] CAST3
[6] Generic      [7] RSA    [8] DSA   [9] DH   [10] CAST5
[11] RC2         [12] RC4   [13] RC5   [14] SSL3 [15] ECDSA
[16] AES         [17] SEED  [18] KCDSA-1024 [19] KCDSA-2048
[20] DSA Domain Param [21] KCDSA Domain Param
[22] RSA X9.31   [23] DH X9.42   [24] ARIA
[25] DH PKCS Domain Param [26] RSA 186-3 Aux Primes
[27] RSA 186-3 Primes   [28] DH X9.42 Domain Param
[29] ECDSA with Extra Bits
> 16
Enter Key Length in bytes (16, 24, 32): 32
Enter Is Token Attribute [0-1]: 1
Enter Is Sensitive Attribute [0-1]: 1
Enter Is Private Attribute [0-1]: 1
Enter Encrypt Attribute [0-1]: 1
Enter Decrypt Attribute [0-1]: 1
Enter Sign Attribute [0-1]: 1
Enter Verify Attribute [0-1]: 1
Enter Wrap Attribute [0-1]: 1
Enter Unwrap Attribute [0-1]: 1
Enter Derive Attribute [0-1]: 1
Enter Extractable Attribute [0-1]: 1
Generated AES Key:          2688464618 (0xa03eb6ea)
Status: Doing great, no errors (CKR_OK)

The AES key generates on the HSM partition. Execute partition contents in lunacm to verify the key is available.
```

### Wrapping the AES key with the Public Key

Wrap the AES key using the Public Key that you generated on the **Alibaba Cloud KMS** console and imported in to the HSM.



### To wrap the AES key with the Public Key

1. Use the same `ckdemo` session and provide the choices to wrap the AES key using appropriate mechanism.

#### (60) Wrap key

Enter your choice: 60

```
[1]DES-ECB      [2]DES-CBC      [3]DES3-ECB    [4]DES3-CBC
[7]CAST3-ECB   [8]CAST3-CBC
[9]RSA          [10]TRANSLA     [11]DES3-CBC-PAD [12]DES3-CBC-PAD-IPSEC
[13]SEED-ECB   [14]SEED-CBC    [15]SEED-CBC-PAD [16]DES-CBC-PAD
[17]CAST3-CBC-PAD [18]CAST5-CBC-PAD [19]AES-ECB     [20]AES-CBC
[21]AES-CBC-PAD [22]AES-CBC-PAD-IPSEC [23]ARIA-ECB   [24]ARIA-CBC
[25]ARIA-CBC-PAD [26]RSA_OAEP    [27]SET_OAEP    [28]AES-CTR
[29]DES3-CTR    [30]AES-KW      [31]AES-KWP     [34]AES-KEY-WRAP
```

Select mechanism for wrapping: 26

Enter filename of OAEP Source Data [0 for none]: 0

Enter handle of wrapping key (0 to list available objects) : 0

Handle 2688464618 (0xa03eb6ea) -- label: Generated AES Key

Handle 3031091932 (0xb4aacadc) -- label: Alibaba Cloud Public Key

Number of objects found = 2

Enter handle of wrapping key (0 to list available objects) : 3031091932

Enter handle of key to wrap (0 to list available objects) : 2688464618

Wrapped key was saved in file wrapped.key



**NOTE:** wrapped.key is the output file that contains the wrapped AES key.

2. Exit the `ckdemo` session by entering 0.

Enter your choice: 0

Exiting GESC SIMULATION LAB

## Uploading Key Material to KMS

Upload the wrapped AES key in to the **Alibaba Cloud KMS**.

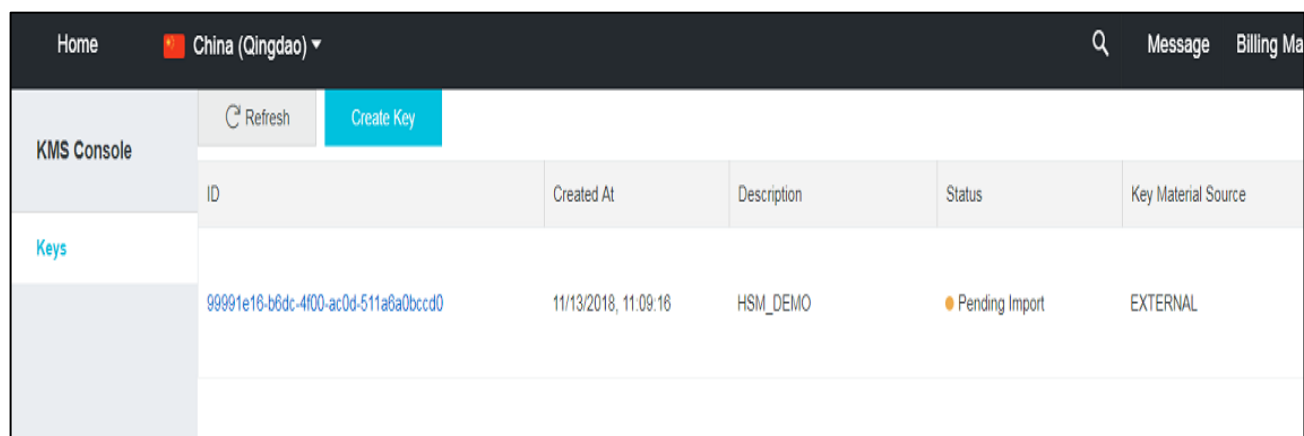
### To upload key material to KMS

1. Perform base64 encoding on the encrypted key material, and save the output as a text file.

```
# openssl enc -e -base64 -A -in wrapped.key -out EncryptedKeyMaterial_base64.txt
```

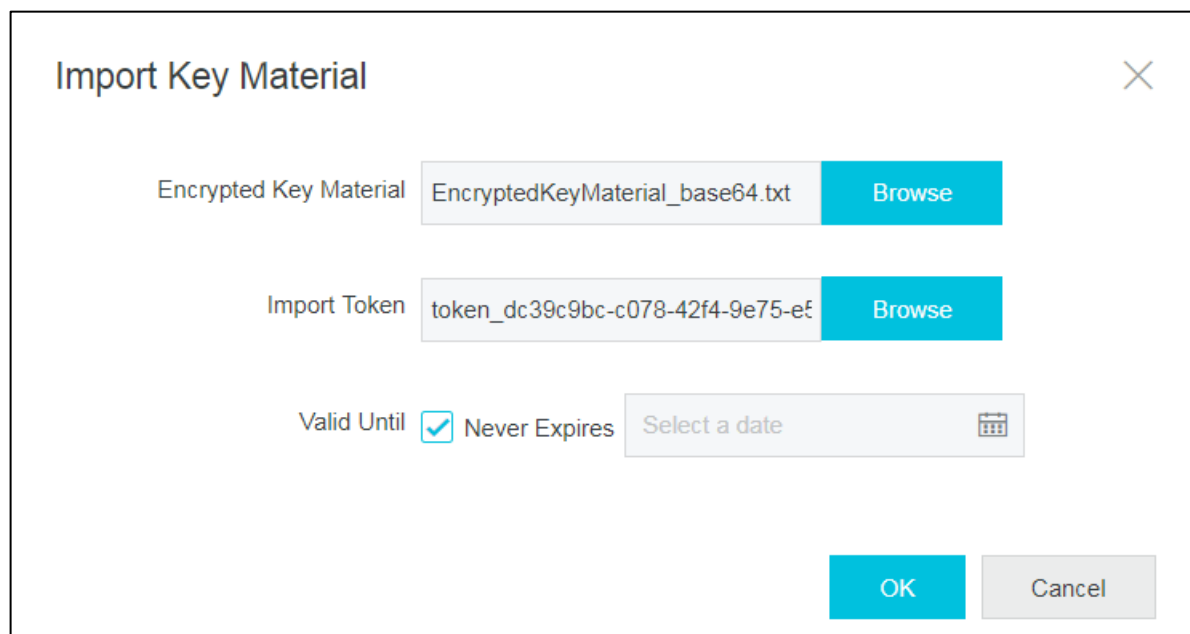
2. Open the **Alibaba Cloud KMS** console.

- Click on the **ID** of the key.

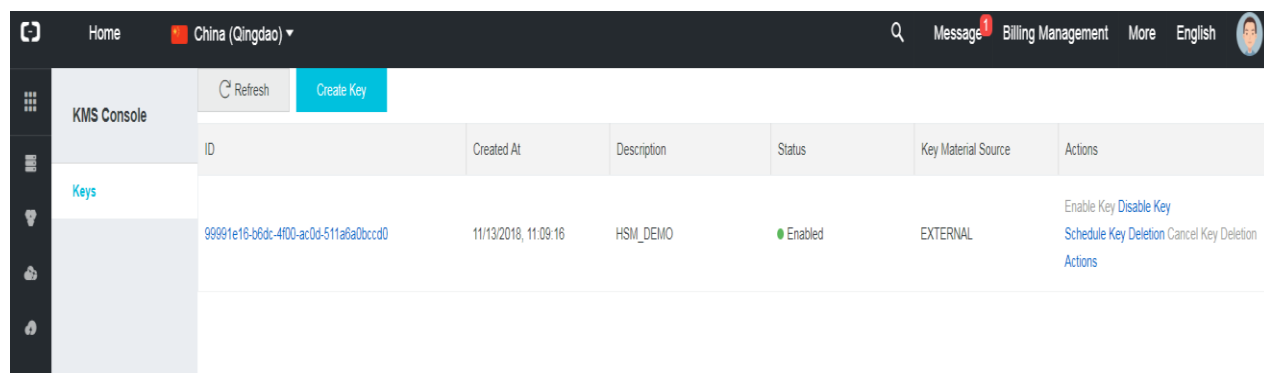


- On **Import Key Material**, click **Browse** for **Encrypted key material** and select the encrypted key material file. Click **Browse** for **Import token** and select the token file that was downloaded with the wrapping key. Select the **Never Expire** check box, or enter an expiry date in the **Valid Until** calendar. When complete, click **OK**.

The key material imports into the **Alibaba Cloud KMS**.



- After successful import, the key is visible in the **Alibaba Cloud KMS** with status **Enabled**.



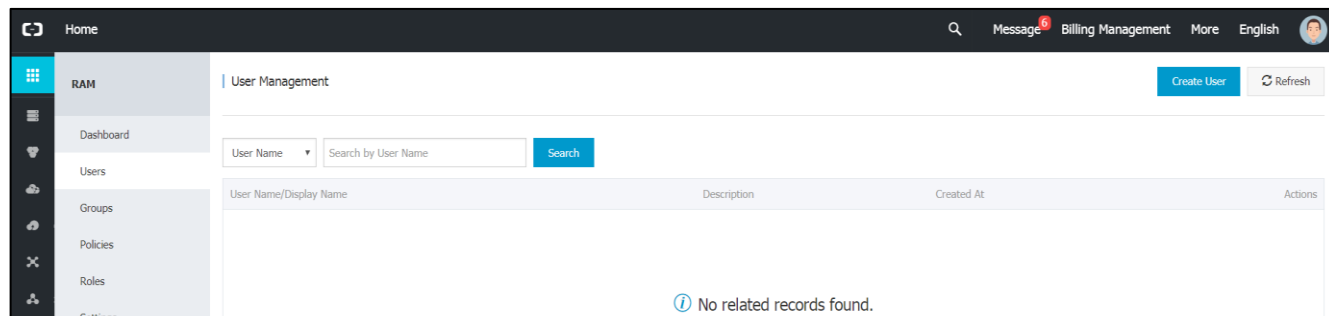
The imported key can now be used for encrypting the Customer Master Keys (CMK).

## Creating a Resource Access Management User

Resource Access Management (RAM) is an Alibaba Cloud service that helps you create and manage user identities and control resources access. You can create and manage RAM users and their access keys.

### To create a Resource Access Management user

- Log into RAM console.



- Click on **Create User** button.

3. Enter **User Name**, **Display Name** and **Description** in the respective fields.

**Create User**

\* User Name :   
 The name can contain 1 to 64 characters, including lowercase letters a-z, uppercase letters A-Z, digits 0-9, and only these special characters: period (.), underscore (\_), and hyphen (-).

Display Name :   
 Display names must contain 1-128 characters. They may include Chinese characters, lowercase letters a-z, numbers 0-9, and these special characters: (@) (.) ( ) ( -).

Description :

Automatically generate an Access key for this user.

**OK** **Cancel**

The user is created and displays on the console.

4. Click on **Manage** under **Actions** for the user.

User Name/Display Name	Description	Created At	Actions
<a href="#">kms_user</a> KMS-User	User to use KMS	2018-11-22 14:04:21	<a href="#">Manage</a>   <a href="#">Authorize</a>   <a href="#">Delete</a> <a href="#">Join Group</a>

5. The user details display. Click the **Create Access Key** button in the **User Access Key** row.

**kms\_user**

**Basic Information** Edit Basic Information

User Name	kms_user	UID	293714942875661771	Created At	2018-11-22 14:04:21
Display Name	KMS-User				
Description	User to use KMS				

**Web Console Logon Management** Enable Console Logon

You must activate MFA Close | Last Logon Time: | On your next logon you must reset the password. Close

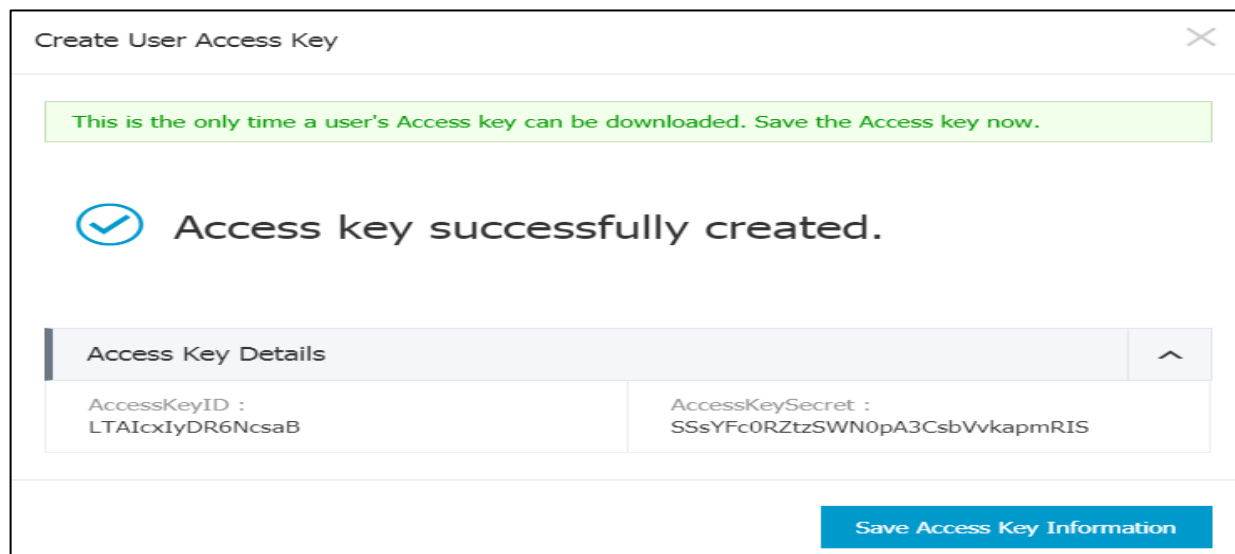
**MFA Device**

Type	Introduction	Enabling Status	Actions
VMFA Device	Application calculates a 6-digit verification code using the TOTP standard algorithm.	Not Enabled	<a href="#">Enable VMFA Device</a>

**User Access Key** Create Access Key

AccessKey ID	Status	Created At	Actions
--------------	--------	------------	---------

- The access key generates and the **Access key successfully created** message displays in the console. Note down the **AccessKeyID** and **AccessKeySecret** or save both in a csv file by clicking the **Save Access Key Information** button.



## Using CMK for encryption/decryption

Once your Resource Access Management (RAM) group user is created you can begin to use the CMK for encryption/decryption operations.

### To use CMK for encryption/decryption

- Configure Alibaba Cloud client on the client machine with the AccessKeyId and AccessKeySecret of the user configured to use KMS SDK.

```
# aliyuncli configure
```

```
Aliyun Access Key ID [None]: <Enter Access Key ID>
```

```
Aliyun Access Key Secret [None]: <Enter Access Key Secret>
```

```
Default Region Id [None]: <Enter the RegionId of your instance>
```

```
Default output format [None]: <Enter your expected output format e.g. json>
```

## 2. List the keys created in the Alibaba cloud account:

```
# aliyuncli kms ListKeys
```

```
[root@NOIHSM1INT-CG04 home]# aliyuncli kms ListKeys
{
  "Keys": {
    "Key": [
      {
        "KeyArn": "acs:kms:cn-qingdao:5836742085772136:key/316c8845-22d2-4c2d-a877-a88e95cde623",
        "KeyId": "316c8845-22d2-4c2d-a877-a88e95cde623"
      },
      {
        "KeyArn": "acs:kms:cn-qingdao:5836742085772136:key/99991e16-b6dc-4f00-ac0d-511a6a0bccd0",
        "KeyId": "99991e16-b6dc-4f00-ac0d-511a6a0bccd0"
      },
      {
        "KeyArn": "acs:kms:cn-qingdao:5836742085772136:key/9eb69ac7-b9d0-4405-87d1-8b1a6a67f7d4",
        "KeyId": "9eb69ac7-b9d0-4405-87d1-8b1a6a67f7d4"
      },
      {
        "KeyArn": "acs:kms:cn-qingdao:5836742085772136:key/dc39c9bc-c078-42f4-9e75-e55711ce9f87",
        "KeyId": "dc39c9bc-c078-42f4-9e75-e55711ce9f87"
      }
    ]
  },
  "TotalCount": 4,
  "PageNumber": 1,
  "RequestId": "bc5a36f7-617c-4b9e-8330-efce59396d1e",
  "PageSize": 10
}
```

3. Encrypt plain text using the **KeyId** of the CMK to obtain an encrypted **CiphertextBlob**. Execute:

```
# aliyuncli kms Encrypt --KeyId <KeyId> --Plaintext "<Text to be encrypted>"
```

For example :

```
# aliyuncli kms Encrypt --KeyId 99991e16-b6dc-4f00-ac0d-511a6a0bccd0 --Plaintext "userpin1"
{
  "KeyId": "99991e16-b6dc-4f00-ac0d-511a6a0bccd0",
  "RequestId": "6f2d6772-8c2c-46a5-943d-df95fc923797",
  "CiphertextBlob":
  "NjAzZTA0N2MtZTBmZC00YTc5LWJlNzMtYWYzNTZiMmI0ZWEdEpLZWIVRnIRME1mRV1tMUIzbVJKOE5GYTBPZjhx2RBQUF
  BQUFBQUFBQktqMGlXekQrYW1PbzFpRmZQZUJjZ1N5V1VtekydzRBPQ=="
}
```

4. Decrypt the encrypted **CiphertextBlob** using the same **KeyId** of CMK. Execute:

```
# aliyuncli kms Decrypt --KeyId <KeyId> --CiphertextBlob "<CiphertextBlob generated during encryption>"
```

For example :

```
# aliyuncli kms Decrypt --KeyId 99991e16-b6dc-4f00-ac0d-511a6a0bccd0 --CiphertextBlob
"NjAzZTA0N2MtZTBmZC00YTc5LWJlNzMtYWYzNTZiMmI0ZWEdEpLZWIVRnIRME1mRV1tMUIzbVJKOE5GYTBPZjhx2RBQUF
  BQUFBQUFBQktqMGlXekQrYW1PbzFpRmZQZUJjZ1N5V1VtekydzRBPQ=="
{
  "Plaintext": "userpin1",
  "KeyId": "99991e16-b6dc-4f00-ac0d-511a6a0bccd0",
  "RequestId": "aeb5e337-b21b-4d4a-a333-6dab3d4485ef"
}
```

This completes the integration of SafeNet HSM with the Alibaba Cloud KMS. This concludes the demonstration of generating an AES256 key on an HSM and wrapping the AES256 key using the public key for importing in to the Alibaba Cloud KMS and using the same for encryption/decryption.