Key Management Service

User Guide

Issue 01

Date 2025-12-02

HUAWEI

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 Service Overview 1
1.1 Concepts 1
1.2 Functions 3
1.3 Advantages 6
1.4 Application Scenarios 6
1.5 Accessing and Using KMS 8
1.5.1 How to Access KMS 8
1.5.2 How to Use KMS 8
2 Key Management 10
2.1 Creating a Key 10
2.2 Creating CMKs Using Imported Key Material 12
2.2.1 Overview 12
2.2.2 Deleting a Key Material 13
2.3 Managing CMKs 14
2.3.1 Querying a Key 14
2.3.2 Changing the Alias and Description of a Key 17
2.3.3 Enabling a Key 18
2.3.4 Disabling a Key 19
2.3.5 Deleting a Key 20
2.3.6 Canceling the Scheduled Deletion of a Key 21
2.4 Encrypting and Decrypting Small-Size Data Online 22
2.5 Managing Tags 24
2.5.1 Adding a Tag 24
2.5.2 Searching for a Custom Key by Tag 26
2.5.3 Modifying Tag Values 26
2.5.4 Deleting Tags 27
3 Permissions Management 29
3.1 Creating a User and Authorizing the User the Permission to Access DEW 29
3.2 Creating a Custom DEW Policy 31
4 FAQs 33
4.1 What Is Key Management Service? 33
4.2 What Is a Customer Master Key? 33

4.3 What Is a Data Encryption Key?..... 34
4.4 Which Cloud Services Can Use KMS for Encryption?..... 34
4.5 Why Can't I Delete a CMK Immediately?..... 34

1 Service Overview

1.1 Concepts

This section describes the basic concepts in DEW, helping you understand and use DEW better.

Symmetric Key Encryption

Symmetric key encryption is also called dedicated key encryption. The sender and receiver use the same key to encrypt and decrypt data.

Advantage: Encryption and decryption are fast.

Disadvantage: Each pair of keys must be unique, making key management difficult when there are a large number of users.

Scenario: Encrypt a large amount of data.

Encryption process: Assume there is a plaintext message "Hello", the sender uses a symmetric key (for example, key123) and a symmetric cryptographic algorithm (for example, AES) to encrypt "Hello" into ciphertext, for example, "# %&*". After receiving the ciphertext, the receiver uses the same key123 and AES algorithm to decrypt "#%&*" back to "Hello".

Asymmetric Key Encryption

Asymmetric key encryption is also called public key encryption. A pair of keys are used for encryption and decryption. One is a public key, and the other is a private key.

Advantage: Different keys are used for encryption and decryption, ensuring high security.

Disadvantage: Encryption and decryption are slow.

Scenario: Encrypt sensitive information.

Encryption process: Assume that the sender needs to send a message "secret plan" to the receiver. The sender obtains the public key of the receiver (for

example, public_key_A) and uses the public key to encrypt "secret plan" into ciphertext "@#$ %^&". After receiving the ciphertext, the receiver uses the private key (for example, private_key_A) to decrypt the ciphertext into the plaintext "secret plan". In this way, even if the public key is obtained by others, the ciphertext cannot be decrypted because they do not have the corresponding private key.

HSM

A Hardware Security Module (HSM) is a type of computer hardware that protects and manages the keys used by strong authentication systems and provides related cryptographic operations.

CMK

A customer master key (CMK), the highest level of keys in a cryptographic system, generates and manages other keys, including session keys and data encryption keys, or directly encrypts important data. It is vital to protect its security and confidentiality. Once a master key is leaked, the entire cryptographic system may be severely threatened.

A master key features the following:

- **High security**: A master key is generally the most sensitive key in a system and needs to be strictly protected. It is usually stored in a secure hardware device, such as an HSM.
- **Long-term use**: A master key has a long lifecycle and will not be frequently changed to ensure system stability and consistency.
- **Multi-usage**: A master key can be used for various encryption operations, including subkey generation, data encryption, and signature verification.
- **Uniqueness**: A master key is unique in a cryptographic system. In a distributed system, each node or region may have its own master key.

Master keys include **custom keys** and **default keys**. You can create, view, enable, disable, schedule the deletion of, and cancel the deletion of custom keys.

Custom keys can be categorized into symmetric keys and asymmetric keys.

- Symmetric keys are most commonly used for data encryption protection.
- Asymmetric keys are used for digital signature verification or sensitive information encryption in systems where the trust relationship is not mutual. An asymmetric key consists of a public key and a private key. The public key can be sent to anyone. The private key must be securely stored and only accessible to trusted users.
- An asymmetric key can be used to generate and verify a signature. To securely transfer data, a signer sends the public key to a receiver, uses the private key to sign data, and then sends the data and signature to the receiver. The receiver can use the public key to verify the signature.

Default Key

A default key is automatically created by another cloud service using KMS, such as Object Storage Service (OBS). The alias of a default key ends with **/default**. The

default key cannot be disabled and does not support scheduled deletion. For details about cloud services that support KMS encryption, see .

Envelope Encryption

Envelope encryption is the practice of encrypting data with a DEK and then encrypting the DEK with a root key that you can fully manage. In this case, CMKs are not required for encryption or decryption.

DEK

A data encryption key (DEK) is used to encrypt data.

1.2 Functions

KMS provides the following functions:

- Manages custom keys.

  You can perform the following operations on custom keys on the KMS console or via APIs:

  - Creating, querying, enabling, disabling, scheduling the deletion of, and canceling the deletion of custom keys
  - Importing keys and deleting key material
  - Modifying the alias and description of a custom key
  - Using the online tool to encrypt and decrypt small-size data
  - Adding, searching for, editing, and deleting tags
- Creates, encrypts, and decrypts DEKs.

  You can create, encrypt, and decrypt a DEK by calling KMS APIs. For details, see *Data Encryption Workshop API Reference*.
- Generates hardware true random numbers.

  You can generate 512-bit hardware true random numbers using a KMS API. The 512-bit hardware true random numbers can be used as or serve as basis for keys and encryption parameters. For details, see *Data Encryption Workshop (DEW) API Reference*.

Key Algorithms Supported by KMS

Symmetric keys created on the KMS console use the AES-256 algorithm. Asymmetric keys created by KMS support the RSA and ECC algorithms.

Table 1-1 Key algorithms supported by KMS

Key Type	Algorithm Type	Key Specifications	Description	Application Scenario
Symmetric key	AES	AES_256	AES symmetric key	• Data encryption and decryption • DEKs encryption and decryption **NOTE** You can encrypt and decrypt a small amount of data using the online tool on the console. You need to call APIs to encrypt and decrypt a large amount of data.

Key Type	Algorithm Type	Key Specifications	Description	Application Scenario
Asymmetric key	RSA	• RSA_2048 • RSA_3072 • RSA_4096	RSA asymmetric password	• Digital signature and signature verification • Data encryption and decryption **NOTE** Asymmetric keys are applicable to signature and signature verification scenarios. Asymmetric keys are not efficient enough for data encryption. Symmetric keys are suitable for encrypting and decrypting data.
	ECC	• EC_P256 • EC_P384	Elliptic curve recommended by NIST	Digital signature and signature verification

Key wrapping algorithms describes the cryptographic key wrapping algorithms supported by imported keys.

Table 1-2 Key wrapping algorithms

Algorithm	Description	Configuration
RSAES_OAEP_SHA_256	RSA algorithm that uses OAEP and has the **SHA-256** hash function	Select an algorithm based on your HSM functions. If your HSM supports the **RSAES_OAEP_SHA_256** algorithm, use **RSAES_OAEP_SHA_256** to encrypt key materials. **NOTICE** The **RSAES_OAEP_SHA_1** algorithm is no longer secure. Exercise caution when performing this operation.
RSAES_OAEP_SHA_1	RSA algorithm that uses Optimal Asymmetric Encryption Padding (OAEP) and has the **SHA-1** hash function	

1.3 Advantages

Extensive Service Integration

- By integrating with OBS, EVS, and IMS, you can use KMS to manage the keys of the services or use KMS APIs to encrypt and decrypt local data.
- By integrating with Cloud Trace Service (CTS), you can use CTS to view recent KMS operation records.

Regulatory Compliance

Keys are generated by third-party validated HSMs. Access to keys is controlled and key operations involving keys are traceable by logs, compliant with international laws and regulations.

Easy to Use

You can use and manage keys easily using the console or APIs, needless to purchase hardware encryption devices.

1.4 Application Scenarios

KMS provides central management and control capabilities of CMKs for Object Storage Service (OBS), Elastic Volume Service (EVS), Image Management Service (IMS), Relational Database Service (RDS), and user applications. It is perfectly suited for data encryption and decryption scenarios.

- For OBS, KMS applies to object encryption on OBS.

> **NOTE**
>
> OBS is an object-based storage service that provides customers with massive, secure, reliable, and cost-effective data storage capabilities, including but not limited to bucket creation, modification, deletion, and management, as well as object upload, download, deletion, and general management. OBS can store all file types, and is suitable for individual subscribers, websites, enterprises, and developers. For details about OBS, see *Object Storage Service (OBS) User Guide*.

- For EVS, KMS applies to data encryption in EVS disks.

  > **NOTE**
  >
  > Based on a distributed architecture, an EVS disk is a virtual block storage device that can be elastically scaled up and down. EVS disks can be operated online. Using them is the same as using common server hard disks. Compared with traditional hard disks, EVS disks have higher data reliability and I/O throughput and are easier to use. EVS disks can be used in file systems, databases, and system software applications that require block storage devices. For more information about EVS, see the *Elastic Volume Service User Guide*.

- For IMS, KMS applies to the creation of encrypted private images.

  > **NOTE**
  >
  > IMS provides easy-to-use self-service image management functions. You can apply for a cloud server using either a private image or a public image. You can also create a private image using an existing ECS or an external image file. For more information about IMS, see the *Image Management Service User Guide*.

- For RDS, KMS applies to disk encryption in RDS database instances.

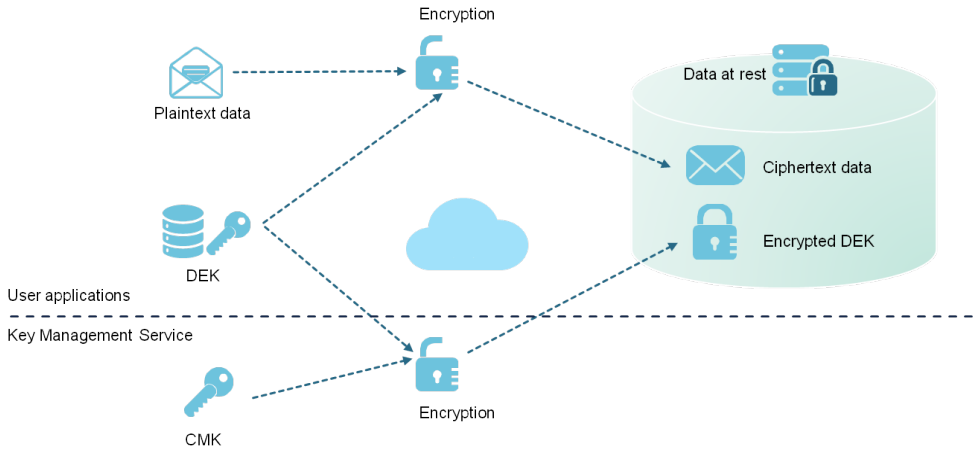
  > **NOTE**
  >
  > RDS is an online relational database service based on the cloud computing platform. RDS is out-of-box, reliable, scalable, and easy to manage. For more information about RDS, see the *Relational Database Service User Guide*.

- For user applications

  To encrypt plaintext data, a user application can call the necessary KMS API to generate a DEK, which can then be used to encrypt the plaintext data. Then the application can store the encrypted data. In addition, the user application can call the necessary KMS APIs to create custom keys. DEKs can be stored in ciphertext after being encrypted with the custom keys. **Figure 1-1** shows envelope encryption working principles.

  To ensure the security of the user's encrypted data, KMS does not save DEKs in plaintext or ciphertext. Instead, it manages users' custom keys so that users can obtain and use DEKs securely.

Figure 1-1 Envelope encryption working principles



1.5 Accessing and Using KMS

1.5.1 How to Access KMS

The cloud service provides a web-based service management platform. You can access KMS using HTTPS-compliant APIs or the management console.

- Management console

  If you have registered with the cloud service, you can log in to the management console directly. In the upper left corner of the console, click . Select a region or project. Choose **Security** > **Data Encryption Workshop**.
- API

  You can access KMS using APIs. For details, see *Data Encryption Workshop (DEW) API Reference*.

1.5.2 How to Use KMS

Working with OBS

Users can upload objects to and download them from Object Storage Service (OBS) in common mode or server-side encryption mode. When users upload objects in encryption mode, data is encrypted at the server side and then securely stored on OBS in ciphertext. When users download encrypted objects, the data in ciphertext is decrypted at the server side and then provided to users in plaintext. OBS supports the server-side encryption with KMS-managed keys (SSE-KMS) mode. In SSE-KMS mode, OBS uses the keys provided by KMS for server-side encryption.

For details about how to upload objects to OBS in SSE-KMS mode, see the *Object Storage Service User Guide*.

Working with EVS

If you enable the encryption function when creating an EVS disk and select a CMK provided by KMS to encrypt the EVS disk, data stored to the EVS disk is automatically encrypted.

For details about how to use the encryption function of EVS, see the *Elastic Volume Service User Guide*.

Working with IMS

When creating a private image using an external image file, you can enable the private image encryption function and select a CMK provided by KMS to encrypt the image.

For details about how to use the private image encryption function of Image Management Service (IMS), see the *Image Management Service User Guide*.

Working with SFS

When creating a file system on SFS, the CMK provided by KMS can be selected to encrypt the file system, so that files stored in the file system are automatically encrypted.

For details about how to use the encryption function of SFS, see the *Scalable File Service User Guide*.

Working with RDS

When creating a database instance, you can enable the disk encryption function of the database instance and select a CMK created on KMS to encrypt the disk of the database instance. The enablement of disk encryption will enhance data security.

For details about how to use the disk encryption function of RDS, see the *Relational Database Service User Guide*.

Working with User Applications

To encrypt plaintext data, a user application can call the necessary KMS APIs to generate a DEK. The DEK can then be used to encrypt the plaintext data. Then the application can store the encrypted data. In addition, the user application can call the necessary KMS APIs to create CMKs. DEKs can be stored in ciphertext after being encrypted with the CMKs. For details, see the *Key Management Service API Reference*.

2 Key Management

2.1 Creating a Key

Scenario

CMKs can be used for:

- Server-side encryption on OBS
- Encryption of data on EVS disks
- Encryption of private images on IMS
- File system encryption on SFS
- Disk encryption for database instances in RDS
- DEK encryption and decryption for user applications

Constraints

- Aliases of default keys end with **/default**. When configuring aliases for your custom keys, the value cannot end with **/default**.

Procedure

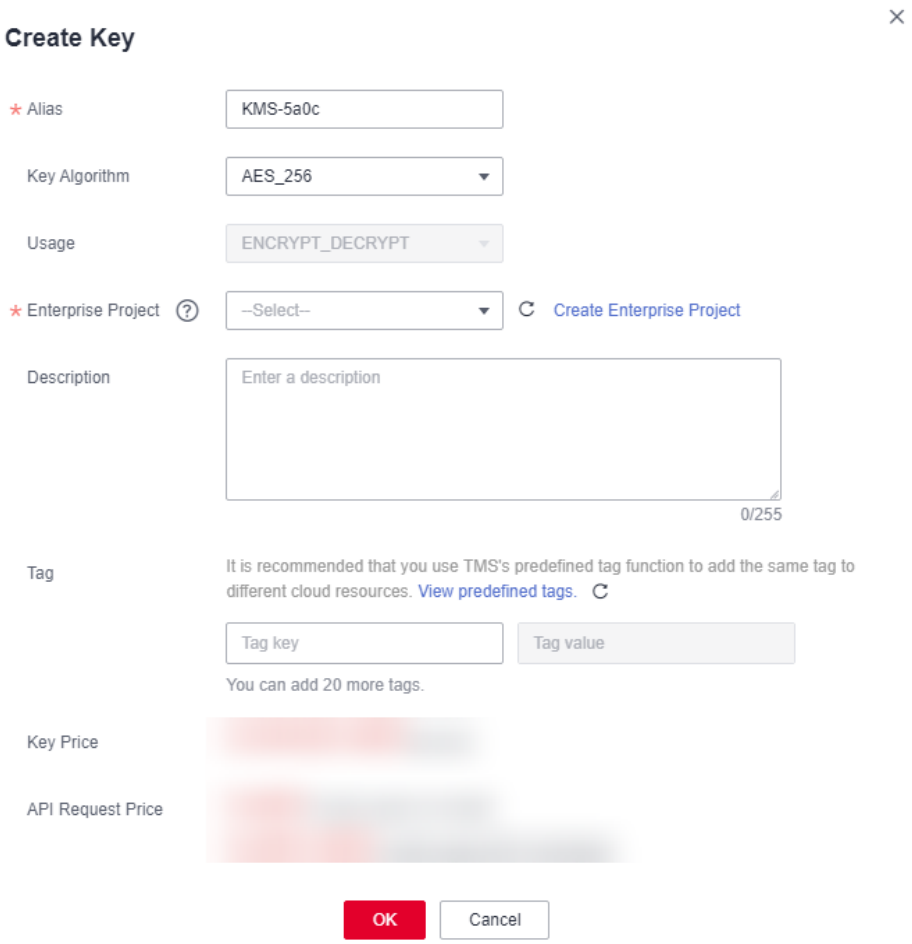
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click **Create Key** in the upper right corner. In the displayed dialog box, enter the alias, names, enterprise project, tags, and description of the key.

Figure 2-1 Create Key dialog box



- **Alias**: Alias of the key to be created
- **Enterprise Project**:

  If you are an enterprise user and have created an enterprise project, select the required enterprise project from the drop-down list. The default project is **default**.

  If there are no **Enterprise Management** options displayed, you do not need to configure it.
- (Optional) **Description** is the description of the custom key.
- (Optional) **Tags**: Add tags to the custom key as needed, and enter the tag key and tag value.

  NOTE

  - If a custom key has been created without any tag, you can add a tag to the custom key later as necessary. Click the alias of the custom key. The page with key details is displayed. Then you can add tags to the custom key.
  - The same tag (including tag key and tag value) can be used for different custom keys. However, under the same custom key, one tag key can have only one tag value.
  - A maximum of 20 tags can be added for one custom key.
  - If you want to delete a tag to be added when adding multiple tags, you can click **Delete** in the row where the tag to be added is located to delete the tag.

Step 5 Click **OK**.

In the custom key list, you can view created custom keys. The default status of a custom key is **Enabled**.

----End

Related Operations

- For details about how to upload objects with server-side encryption, see section **Uploading a File with Server-Side Encryption** in the *Object Storage Service User Guide*.
- For details about how to encrypt data on EVS disks, see section **Creating an EVS Disk** in the *Elastic Volume Service User Guide*.
- For details about how to encrypt private images, see section **Encrypting an Image** in the *Image Management Service User Guide*.
- For details about how to encrypt the file system on SFS, see section **Creating a File System** in the *Scalable File Service User Guide*.
- For details about how to encrypt disks for a database instance in RDS, see section **Creating an RDS MySQL DB Instance** in the *Relational Database Service User Guide*.
- For details about how to create a DEK and a plaintext-free DEK, see sections "Creating a DEK" and "Creating a Plaintext-Free DEK" in *Data Encryption Workshop (DEW) API Reference*.
- For details about how to encrypt and decrypt a DEK for a user application, see sections "Encrypting a DEK" and "Decrypting a DEK" in *Data Encryption Workshop (DEW) API Reference*.

2.2 Creating CMKs Using Imported Key Material

2.2.1 Overview

A custom key contains key metadata (key ID, key name, description, key status, and creation date) and key materials used for encrypting and decrypting data.

- When a user uses the KMS console to create a custom key, the KMS automatically generates a key material for the custom key.
- If you want to use your own key material, you can use the KMS console to create a custom key whose key material source is external, and import the key material to the custom key.

Important Notes

- Security

  You need to ensure that random sources meet your security requirements when using them to generate key materials. When using the import key materials function, you need to be responsible for the security of your key materials. Save the original backup of the key material so that the backup key material can be imported to the KMS in time when the key material is deleted accidentally.

- Availability and durability

  Before importing the key material into KMS, you need to ensure the availability and durability of the key material.

  Differences between the imported key material and the key material generated by KMS are shown in **Table 2-1**.

  Table 2-1 Differences between the imported key material and the key material generated by KMS

Key Material Source	Difference
Imported keys	• You can delete the key material, but cannot delete the custom key and its metadata. • Such keys cannot be rotated. • When importing the key material, you can set the expiration time of the key material. After the key material expires, the KMS automatically deletes the key material within 24 hours, but does not delete the custom key and its metadata. It is recommended that you save a copy of the material on your local device because it may be used for re-import in cases of invalid key materials or key material mis-deletion.
Keys created in KMS	• The key material cannot be manually deleted. • Symmetric keys can be rotated. • You cannot set the expiration time for key material.

- Association

  When a key material is imported to a custom key, the custom key is permanently associated with the key material. Other key materials cannot be imported into the custom key.

- Uniqueness

  If you use the custom key created using the imported key material to encrypt data, the encrypted data can be decrypted only by the custom key that has been used to encrypt the data, because the metadata and key material of the custom key must be consistent.

2.2.2 Deleting a Key Material

Scenario

When importing key material, you can specify the expiration time. After the key material expires, KMS deletes it, and the status of the custom key changes to **Pending import**. You can manually delete the key material as needed. The effect of expiration of the key material is the same as that of manual deletion of the key material.

This section describes how to delete imported key material on the management console.

Constraints

- To re-import a deleted key material, ensure the imported material is the same as the deleted one.
- Data encrypted using a custom key cannot be decrypted if the key material of the custom key was deleted. To decrypt the data, re-import the key material.
- After the deletion, the key will become unavailable and its status will change to **Pending import**.

Prerequisites

- You have imported the key material for a key.
- The material source of the key is **External**.
- The key status is **Enabled** or **Disabled**.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Locate the target key material and choose **More** > **Delete Key Material** in the **Operation** column.

Step 5 In the displayed dialog box, click **Yes**.

After the deletion, the key will become unavailable and its status changes to **Pending import**.

----End

2.3 Managing CMKs

2.3.1 Querying a Key

Scenario

This section describes how to use the management console to view the information about a custom key, such as its alias, status, ID, and creation time. The status of a key can be **Enabled**, **Disabled**, **Pending deletion**, or **Pending import**.

Procedure

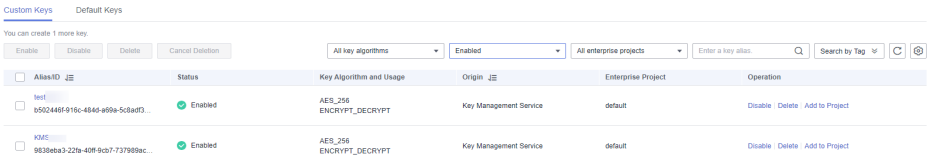
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 View key details in the key list.

Figure 2-2 Key list



NOTE

- To filter keys in a certain status, select the status from the drop-down list of **All statuses**.
- Enter the key alias in the search box in the upper right corner. Click or press **Enter**.
- You can click **Search Tag** to search for the custom key that meets the search criteria.
- You can click in the upper right corner of the key list to set the columns of the list.

Table 2-2 describes the parameters of a key list.

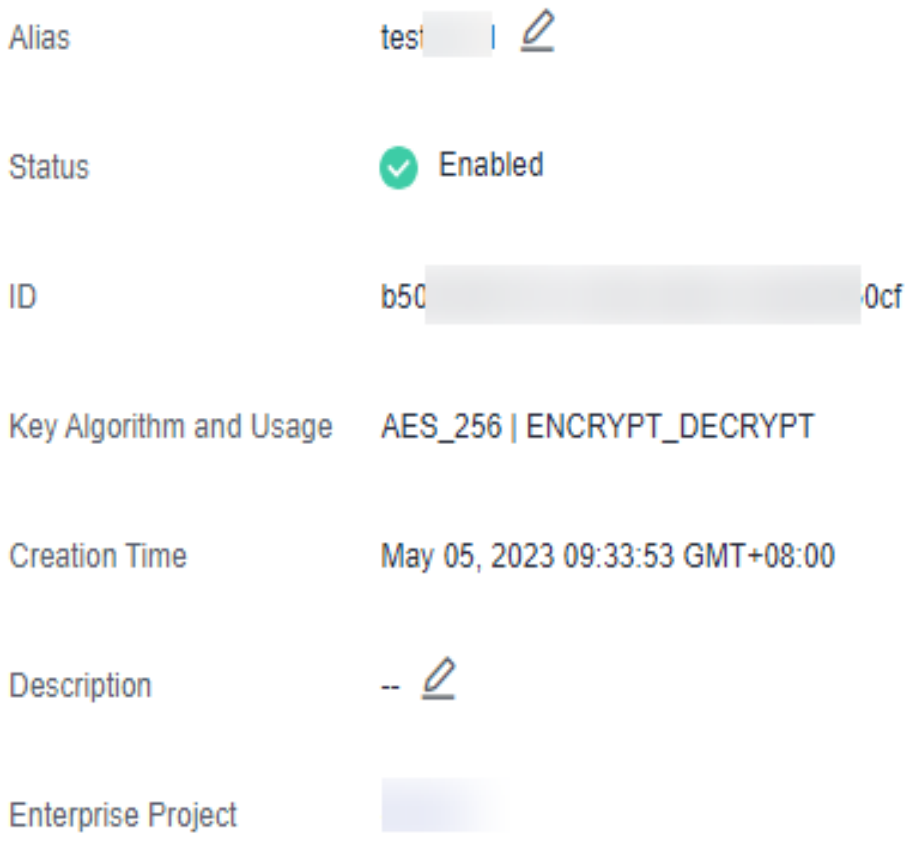
Table 2-2 Key list parameters

Parameter	Description
Alias/ID	of a key and the random ID of a key generated during its creation
Status	Status of a key, which can be one of the following: • **Enabled** The key is enabled. • **Disabled** The key is disabled. • **Pending deletion** The key is scheduled for deletion. • **Pending import** If a key does not have any key material, its status is **Pending import**.
Creation Time	Creation time of the key
Expiration Time	Expiration time of the key material. When the material expires, the custom key becomes an empty key.

Parameter	Description
Origin	Source of key material, which can be one of the following: • **External** The key uses an imported key material. • Key Management Service The key uses KMS-generated material.
Enterprise Project	ID of the enterprise project bound to a key during key creation.

Step 5 Click the key alias to view its details.

Figure 2-3 Viewing key details



NOTE

To change the alias or description of the CMK, click next to **Alias** or **Description**.

- The alias and description of the default key cannot be modified. The alias of the default key ends with **/default**.
- The alias and description of a key cannot be changed if the key is in **Pending deletion** status.

----End

2.3.2 Changing the Alias and Description of a Key

Scenario

Key aliases help you find custom keys more easily.

This section describes how to change the alias and description of a custom key on the KMS management console.

NOTICE

- The alias and description of the default master key cannot be modified. The alias of the default master key ends with **/default**.
- The alias and description of a key cannot be changed if the key is in **Pending deletion** status.

Prerequisites

- The custom key is in **Enabled**, **Disabled**, or **Pending import** status.

Procedure

Step 1 Log in to the management console.

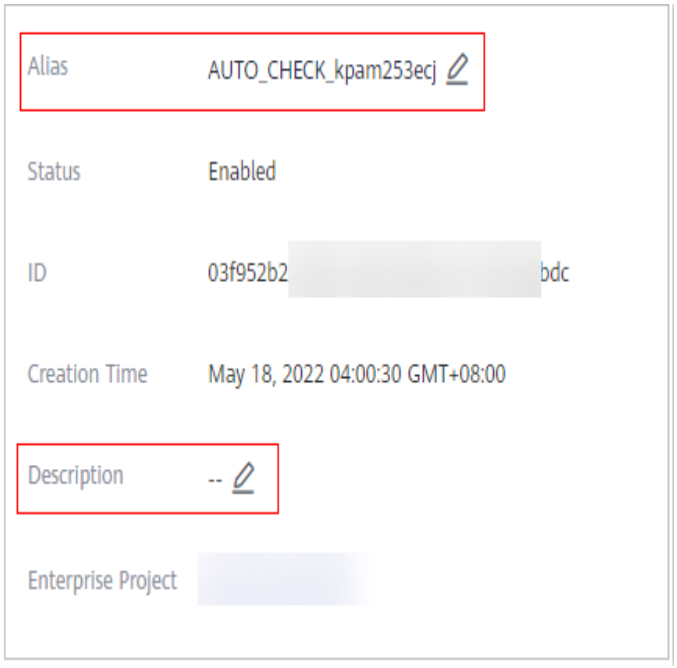
Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click the alias or description of the target key to access its details page.

Step 5 To change the alias or description of a key, click  next to **Alias** or **Description**.

Figure 2-4 Key details



NOTE

- The alias can contain 1 to 255 characters. Only digits, letters, underscores (_), hyphens (-), colons (:), and forward slashes (/) are allowed.
- Length of the description cannot exceed 255 characters.

Step 6 Click ✓ to save the changes.

----End

2.3.3 Enabling a Key

Scenario

This section describes how to use the management console to enable one or multiple custom keys. Only enabled keys can be used to encrypt/decrypt data. A new custom key is in the **Enabled** state by default.

Prerequisites

The key you want to enable is in **Disabled** status.

Procedure

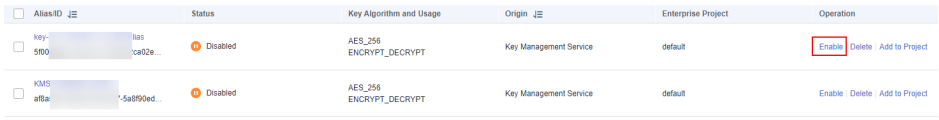
Step 1 Log in to the management console.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 In the row containing the desired key, click **Enable**.

Figure 2-5 Enabling a single key



Step 5 In the dialog box that is displayed, click **Yes** to enable the key.

NOTE

To enable multiple keys at a time, select them and click **Enable** in the upper left corner of the list.

----End

2.3.4 Disabling a Key

Scenario

This section describes how to use the management console to disable one or multiple custom keys, thereby protecting data in urgent cases.

After being disabled, a custom key cannot be used to encrypt or decrypt any data. Before using a disabled key to encrypt or decrypt data, you must enable it by following instructions in **Enabling a Key**.

NOTE

Default keys created by KMS cannot be disabled.

Prerequisites

The key you want to disable is in **Enabled** status.

Procedure

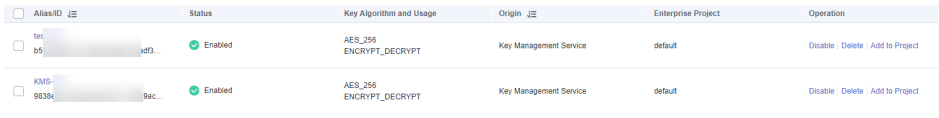
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 In the row containing the desired key, click **Disable**.

Figure 2-6 Disabling a single key



Step 5 In the dialog box that is displayed, select **I understand the impact of disabling keys** and click .

NOTE

To disable multiple keys at a time, select them and click **Disable** in the upper left corner of the list.

----End

2.3.5 Deleting a Key

Scenario

This section describes how to use the management console to schedule the deletion of one or multiple unwanted custom keys.

If deletion is scheduled for a key, the deletion will not take effect immediately. Instead, it will take effect after a waiting period of 7 to 1096 days. Before the specified deletion date, you can cancel the deletion if you want to use the key. Once the scheduled deletion has taken effect, the key will be deleted permanently and you will not be able to decrypt data encrypted by it. Therefore, you are advised to exercise caution when performing this operation.

Before deleting the key, confirm that it is not in use and will not be used.

NOTE

Default Master Keys created by KMS cannot be scheduled for deletion.

Prerequisites

- The key to be deleted is in **Enabled**, **Disabled**, or **Pending Import** status.

Procedure

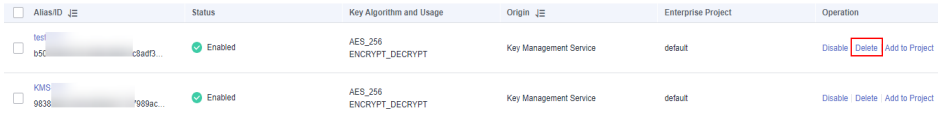
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

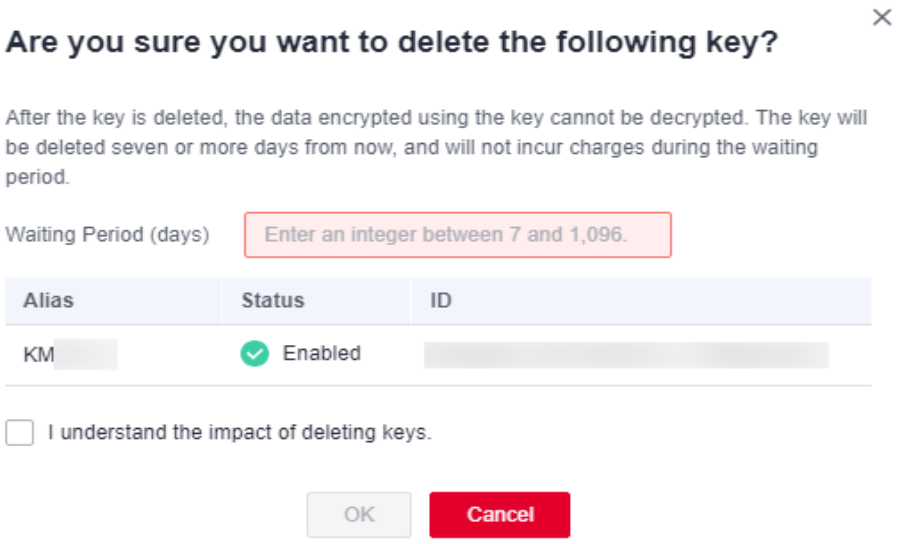
Step 4 In the row containing the desired key, click **Delete**.

Figure 2-7 Scheduling the deletion for a single key



Step 5 On the key deletion dialog box, enter the deletion delay time.

Figure 2-8 Scheduling a deletion time



Step 6 Select **I understand the impact of deleting keys**.

Step 7 Click **Yes** to schedule the deletion.

NOTE

To delete multiple keys at a time, select them and click **Delete** in the upper left corner of the list.

----End

2.3.6 Canceling the Scheduled Deletion of a Key

Scenario

This section describes how to use the management console to cancel the scheduled deletion of a custom key prior to deletion execution.

Prerequisites

The key for which you want to cancel the scheduled deletion is in **Pending deletion** status.

Procedure

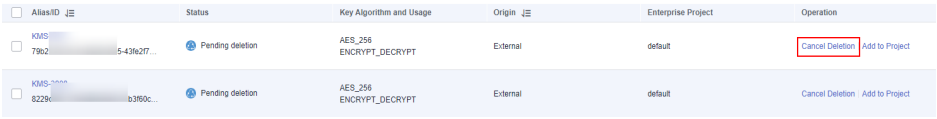
Step 1 Log in to the management console.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 In the row containing the desired key, click **Cancel Deletion**.

Figure 2-9 Canceling the scheduled deletion of a single key



Step 5 In the displayed dialog box, click **Yes** to cancel the scheduled deletion for the key.

- If the key is created using KMS generated material, its status becomes **Disabled** after the cancelation. To enable the key, see **Enabling a Key**.
- If the key is created using imported material, its status becomes **Disabled** after the cancelation. To enable the key, see **Enabling a Key**.
- If the key is created using imported material and no key material has been imported for it, its status becomes **Pending import** after the cancelation. To use the key, perform **Creating CMKs Using Imported Key Material**.

NOTE

To cancel the deletion of multiple keys at a time, select them and click **Cancel Deletion** in the upper left corner of the list.

----End

2.4 Encrypting and Decrypting Small-Size Data Online

This section describes how to use an online tool to encrypt and decrypt data less than or equal to 4 KB on the KMS console.

Prerequisites

The desired custom key is in **Enabled** status.

Constraints

- Default keys cannot be used to encrypt or decrypt such data with the tool.
- Asymmetric keys cannot be used to encrypt or decrypt such data with the tool.
- You can call an API to use a default key to encrypt or decrypt small volumes of data. For details, see the *Key Management Service API Reference*.
- Use the current CMK to encrypt the data.
- Exercise caution when you delete a CMK. The online tool cannot decrypt data if the CMK used for encryption has been deleted.
- After an API is called to encrypt data, the online tool cannot be used to decrypt the data.

Encrypting Data

Step 1 Log in to the management console.

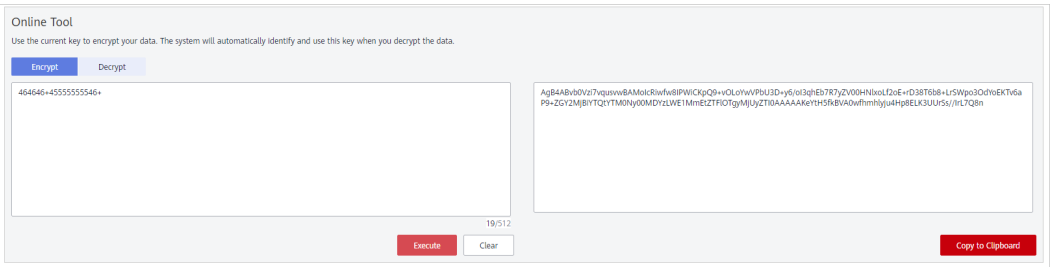
Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click the name of the target custom key to access the key details page. Click the **Tool** tab.

Step 5 Click **Encrypt**. In the text box on the left, enter the data to be encrypted.

Figure 2-10 Encrypting data



Step 6 Click **Execute**. The data encryption result is displayed in the text box on the right.

NOTE

- The key you clicked is used for encryption.
- To clear your input, click **Clear**.
- To copy the encrypted data, click **Copy to Clipboard**. You can then paste and save it to a local file.

----End

Decrypting Data

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click the alias of an enabled key (excepting Default Master Keys) to access its details page.

Step 5 Click the **Tool** tab.

Step 6 Click **Decrypt**. In the text box on the left, enter the data to be decrypted.

NOTE

- The online tool automatically identifies the key used for data encryption, and uses it to decrypt data.
- If the key has been deleted, the decryption will fail.

Step 7 Click **Execute**. The data decryption result is displayed in plaintext in the text box on the right.

NOTE

To copy the decrypted data, click **Copy to Clipboard**. You can then paste and save it to a local file.

----End

2.5 Managing Tags

2.5.1 Adding a Tag

Scenario

Tags are used to identify custom keys. You can add tags to custom keys so that you can classify custom keys, trace them, and collect their usage status according to the tags.

Constraints

Tags cannot be added to default keys.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

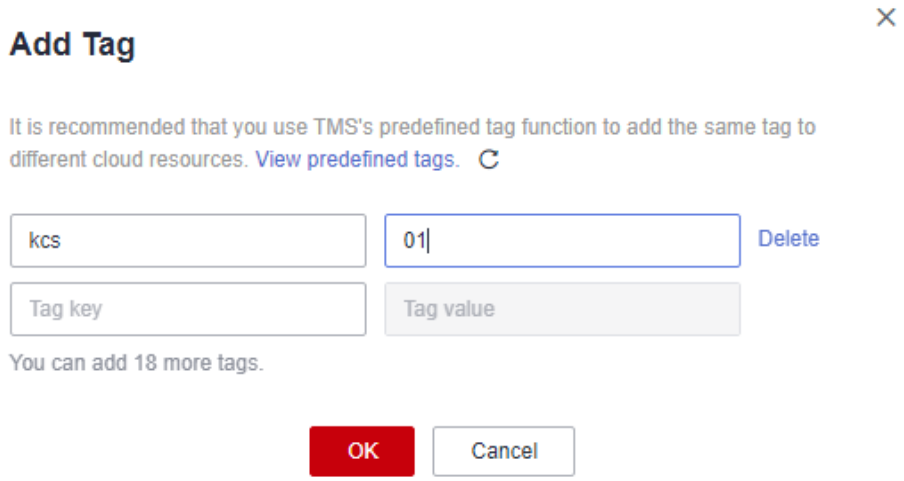
Step 4 Click the alias of the target custom key to view its details.

Step 5 Click **Tags** to go to the tag management page.

Figure 2-11 Managing tags

Step 6 Click **Add Tag**. In the **Add Tag** dialog box, enter the tag key and tag value. **Table 2-3** describes the parameters.

Figure 2-12 Adding a tag



NOTE

If you want to delete a tag to be added when adding multiple tags, you can click **Delete** in the row where the tag to be added is located to delete the tag.

Table 2-3 Tag parameters

Parameter	Description	Value	Example Value
Tag key	Name of a tag. The same tag (including tag key and tag value) can be used for different keys. However, under the same custom key, one tag key can have only one tag value. A maximum of 20 tags can be added for one custom key.	• Mandatory. • Each tag key must be unique under the same custom key. • Contains a maximum of 36 characters. • Only digits, letters, underscores (_), and hyphens (-) are allowed.	cost
Tag value	Value of the tag	• This parameter can be empty. • Can contain a maximum of 43 characters. • Only digits, letters, underscores (_), and hyphens (-) are allowed.	100

Step 7 Click **OK** to complete.

----End

2.5.2 Searching for a Custom Key by Tag

Scenario

This section describes how to search for tags through KMS. You can search for tags of all custom keys that meet the search criteria in the current project.

Prerequisites

Tags have been added.

Procedure

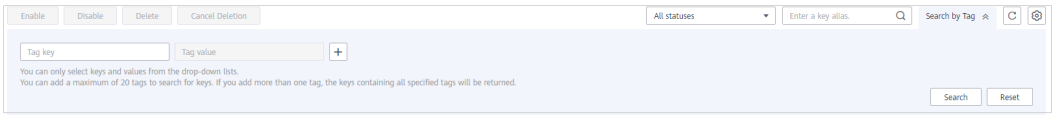
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click **Search by Tag** to show the search box.

Figure 2-13 Searching for tags



Step 5 In the search box, enter the tag key and tag value.

Step 6 Click  to add the input to the search criteria, and click **Search**. The list displays the custom keys that meet the search criteria.

NOTE

- Multiple tags can be added for one search, 20 at most. If multiple tags are added, only custom keys that meet the combined search criteria are displayed.
- If you want to delete an added tag from the search criteria, click  next to the tag.
- You can click **Reset** to reset the search criteria.

----End

2.5.3 Modifying Tag Values

Scenario

This section describes how to modify tag values on the KMS management console.

Procedure

Step 1 Log in to the management console.

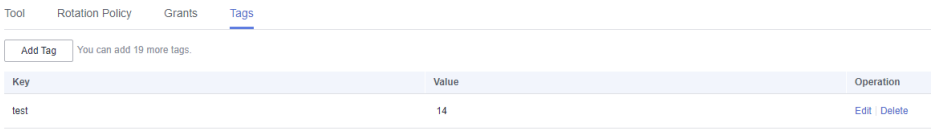
Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click the alias of the target custom key to view its details.

Step 5 Click **Tags** to go to the tag management page.

Figure 2-14 Managing tags



Step 6 Click **Edit** of the target tag, and the **Edit Tag** dialog box is displayed.

Step 7 In the **Edit Tag** dialog box, enter a tag value, and click **OK** to complete the editing.

----End

2.5.4 Deleting Tags

Scenario

This section describes how to delete tags on the KMS management console.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner of the management console and select a region or project.

Step 3 Choose **Security** > **Data Encryption Workshop**. The key management page is displayed.

Step 4 Click the alias of the target custom key to view its details.

Step 5 Click **Tags** to go to the tag management page.

Figure 2-15 Managing tags

Step 6 Click **Delete** of the target tag, and the **Delete Tag** dialog box is displayed.

Step 7 In the **Delete Tag** dialog box, click **Yes** to complete the deletion.

----End

3 Permissions Management

3.1 Creating a User and Authorizing the User the Permission to Access DEW

This chapter describes how to use IAM to implement fine-grained permissions control for your KMS resources. With IAM, you can:

- Create IAM users for employees based on the organizational structure of your enterprise. Each IAM user has its own security credentials to access DEW resources.
- Grant users only the permissions required to perform a task.
- Entrust an account or cloud service to perform efficient O&M on your KMS resources.

If your account does not need individual IAM users, skip this chapter.

This section describes the procedure for granting permissions (see **Figure 3-1**).

Prerequisites

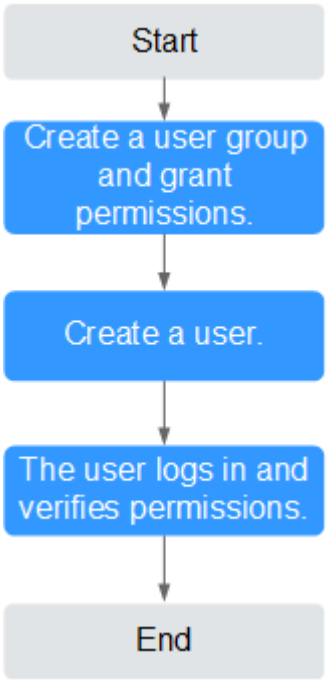
Before granting permissions to a user group, you need to understand the available DEW permissions, and grant permissions based on the real-life scenario. The following tables describe the permissions supported in DEW.

Table 3-1 DEW permissions

Role/Policy	Description	Type
KMS Administrator	Administrator permissions for the encryption key	Role
KMS CMKFullAccess	All permissions for the encryption keys	Policy
KMS CMKReadOnlyAccess	Read-only permission for encryption keys	Policy

Authorization Process

Figure 3-1 Authorizing the DEW access permission to a user



1. **Create a user group and assign permissions**.
2. **Create a user and add it to a user group.**
3. **Log in as the created user** and verify permissions.

   Log in to the console as newly created user, and verify that the user only has the assigned permissions.

Tenant Guest Roles

If you have configured Tenant Guest permissions for the IAM account, apart from the read-only permissions for all cloud services except Identity and Access Management (IAM), you also have the following KMS permissions:

- **kms:cmk:create**: Create a key.
- **kms:cmk:createDataKey**: Create a DEK.
- **kms:cmk:createDataKeyWithoutPlaintext**: Create a plaintext-free DEK.
- **kms:cmk:encryptDataKey**: Encrypt the DEK.
- **kms:cmk:decryptDataKey**: Decrypt a DEK.
- **kms:cmk:retireGrant**: Retire a grant.
- **kms:cmk:decryptData**: Decrypt data.
- **kms:cmk:encryptData**: Encrypt data.
- **kms::generateRandom**: Generate a random number.

If you want to configure the Tenant Guest role for an IAM user but do not want to have the preceding permissions, you need to configure a custom deny policy for

the IAM user. For details about how to configure a custom policy, see **Creating a Custom DEW Policy**.

3.2 Creating a Custom DEW Policy

Custom policies can be created as a supplement to the system policies of KMS. For details about the actions supported by custom policies, see "Permissions Policies and Supported Actions" in *Data Encryption Workshop API Reference*.

You can create custom policies in either of the following ways:

- Visual editor: You can select policy configurations without the need to know policy syntax.

  Custom KMS policy parameters:

  - **Select service**: Select **Data Encryption Workshop**.
  - **Select action**: Set it as required.
  - **(Optional) Select resource**: Set **Resources** to **Specific** and **KeyId** to **Specify resource path**. In the dialog box that is displayed, set **Path** to the ID generated when the key was created. For details about how to obtain the ID, see "Viewing a CMK".
- JSON: Edit JSON policies from scratch or based on an existing policy. For details about how to create custom policies, see **Creating a Custom Policy**. This section describes typical DEW custom policies.

Example Custom Policies of DEW

- Example: authorizing users to create and import keys

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:cmk:create",
                "kms:cmk:getMaterial",
                "kms:cmkTag:create",
                "kms:cmkTag:batch",
                "kms:cmk:importMaterial"
            ]
        }
    ]
}
```

- Example: authorizing users to use keys

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:dek:crypto",
                "kms:cmk:get",
                "kms:cmk:crypto",
                "kms:cmk:generate",
                "kms:cmk:list"
            ]
        }
    ]
}
```


- Example: multi-action policy

  A custom policy can contain actions of multiple services that are all of the global or project-level type. The following is a policy with multiple statements:

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "rds:task:list"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "kms:dek:crypto",
            "kms:cmk:get",
            "kms:cmk:crypto",
            "kms:cmk:generate",
            "kms:cmk:list"
         ]
      }
   ]
}
```


4 FAQs

4.1 What Is Key Management Service?

Key Management Service (KMS) is a secure, reliable, and easy-to-use service that helps users centrally manage and safeguard their Customer Master Keys (CMKs).

This service uses hardware security modules (HSMs) to protect CMKs. HSMs help you create and control CMKs with ease. All CMKs are protected by root keys in HSMs to avoid leakage caused by human error. KMS implements access control and log-based tracking on all operations involving CMKs. Additionally, it provides CMK operation records, meeting your audit and regulatory compliance requirements.

4.2 What Is a Customer Master Key?

A Customer Master Key (CMK) is a Key Encryption Key (KEK) created by a user using KMS. It is used to encrypt and protect Data Encryption Keys (DEKs). One CMK can be used to encrypt one or multiple DEKs.

CMKs are categorized into custom keys and default keys.

- Custom keys

  Keys created or imported by users on the KMS console.

- Default keys

  When a user uses KMS for encryption in a cloud service for the first time, the cloud service automatically creates a key whose alias ends with **/default**.

  On the KMS console, you can query Default Master Keys, but can neither disable them nor schedule their deletion.

Table 4-1 Default Master Keys

Key Alias	Cloud Service
obs/default	Object Storage Service (OBS)
evs/default	Elastic Volume Service (EVS)

Key Alias	Cloud Service
ims/default	Image Management Service (IMS)
sfs/default	Scalable File Service (SFS)

4.3 What Is a Data Encryption Key?

A data encryption key (DEK) is used to encrypt data.

4.4 Which Cloud Services Can Use KMS for Encryption?

Object Storage Service (OBS), Elastic Volume Service (EVS), Image Management Service (IMS), and Relational Database Service (RDS) can use KMS for encryption.

4.5 Why Can't I Delete a CMK Immediately?

The decision to delete a CMK should be taken with caution. Before deletion, confirm that the CMK's encrypted data has all been migrated. Once the CMK is deleted, you will not be able to decrypt data with it. Therefore, KMS offers a waiting period of 7 to 1096 days for the deletion to finally take effect. On the scheduled day of deletion, the CMK will be permanently deleted. However, prior to the scheduled day, you can still cancel the deletion.

